NTNU
Norwegian University of
Science and Technology

Well Models for Production Optimization

Marte Arianson

Master of Science in Engineering Cybernetics
Submission date: June 2008
Supervisor: Bjarne Anton Foss, ITK

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Problem Description

Production optimization on a short time horizon, typically a day, requires mathematical models for wells and collection system. The downstream boundary is usually some fixed inlet separator value, like the separator pressure, and the upstream boundary is one or several subsurface hydrocarbon reservoir. In this work the focus is on near well models. A model may be developed from physics or by an empirical approach. This project shall study empirical well models. A polynomial relationship is often used to describe the dependency of the gas oil ratio (GOR) on production rate, providing a simplified representation of the inflow properties of a reservoir. Proper updating of the rate dependent GOR relationship is required to obtain realistic optimization of production. The main activities include:

1. A brief description of the production optimization problem.
2. A literature survey on near well models for production optimization. This survey should end with a set of requirements for a near well model suitable for production optimization.
3. Alternative model structure candidates for empirical near well models should be discussed and at least two of these should be selected for further analysis.
4. Asses the data available for model updating and the need for filtering this data before model identification. Include a discussion on data sources, e.g. may synthetic data from a comprehensive reservoir simulator be applicable for model identification.
5. Choose an appropriate method for model identification and identify at least two alternative models. Asses model validity and uncertainty of the models, including sensitivity towards missing data points. Use data made available by FMC Technologies AS in the analysis.
6. It is important to be able to update a model efficiently as new data become available. Derive a procedure, possibly a recursive method, together with suitable tools or algorithms for accomplishing this.
7. Propose activities for further work, in particular activities necessary to develop promising results into components in FMC's product suite.

Assignment given: 07. January 2008
Supervisor: Bjarne Anton Foss, ITK

Abstract

In coupled reservoir simulations a model of a reservoir and the production system are joined together to obtain realistic responses. One way of performing production optimization is to make use of a gas oil ratio (GOR) estimate when calculating the optimal flow rates. Today this estimate is either not accurate enough or calculating it is a time consuming process.

The thesis tries on an alternative online approach for estimating the GOR. It makes use of basis functions in the form of polynomials and normalized radial basis functions together with a recursive least squares (RLS) algorithm. This yields a simple and effective optimization strategy.

When using a second order polynomial together with the fast convergent recursive least squares algorithm one achieves a suitable fit to the estimated production data. The algorithm has not yet been tested together with a production optimization tool and it has not been compared to the existing methods for estimating the GOR. Still it seems to have a lot of potential and the RLS is fast, convergent and proper for the objective of this thesis.

Preface

This Master's thesis concludes my Master of Science with the Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU).

During the course of this work I have had help from many people. My initial thanks goes to my supervisor at NTNU professor Bjarne A. Foss for excellent guidance. Our frequent meetings have been a great support and drive to me.

Special thanks are also due to my fabulous supervisors senior researcher Berit F. Lund at Sintef and Olav Johan Horgen at FMC Technologies for valuable help and guidance and for always being there on my disposal.

Apart from my supervisors, several people at FMC Technologies deserve to be thanked. Thanks to Knut Vannes and Helge Ruistuen for their contribution and making this Master's thesis possible.

Finally I would like to thank my fellow Master's students at NTNU Jørn S. Nilsson and Øyvind Breyholtz for instructive discussions and support throughout the semester.

Marte Arianson
Trondheim, June 2008

Contents

Abbreviations

ARMA	Autoregressive Moving Average
ARMAX	Autoregressive Moving Average eXogenous
ARX	Autoregressive eXogenous
BJ	Box-Jenkins
FIR	Finite Impulse Response
FVF	Formation Volume Factor
GAP	General Allocation Program
GOR	Gas Oil Ratio
LS	Least Squares
mGOR	Marginal Gas Oil Ratio
NCS	Norwegian Continental Shelf
NRBF	Normalized Radial Basis Function
OE	Output Error
RBF	Radial Basis Function
RIV	Recursive Instrumental Variable
RLS	Recursive Least Squares
RPLR	Recursive Prediction Error
RPM	Recursive Pseudolinear Regression
WC	Water Cut
WFT	Windowed Fourier Transform
WOC	Water Oil Contact

B_o	Formation volume factor
V_ON	Volume oil
V_ON	Volume oil at standard conditions
R_s	Solution gas oil ratio

List of Figures

List of Tables

Chapter 1

Introduction

The Norwegian Continental Shelf (NCS) is experiencing a steeper decrease in production of oil than previously assumed [Bertelsen, 2007]. It has been more than ten years since the last great discovery of oil hence there is a huge focus on increased oil recovery from already discovered reservoirs. This focus is not only about making money but also about recovering the oil before starting to produce gas. The reason for this is that if the oil and gas companies produce gas exclusively, the reservoirs will loose pressure and highly valuable oil reserves will be lost.

In reservoir engineering it is common to simplify the behavior of flow lines, whereas production optimization tools tend to simplify the reservoir behavior. This is often done by two separate tools namely a reservoir simulator for the reservoir behaviour and a production optimizer taking care of the flow lines and production network. The two can be coupled to achieve a more complete model. In this coupling the production optimizer requires an algorithm for efficient calculation of the gas oil ratio (GOR) to obtain realistic responses from the reservoir simulator.

Figure 1.1 presents a typical subsea network. The figure illustrates the wells which produce from the reservoir. They are gathered through manifolds, which may route each of the wells to common gathering flow lines. The estimate of the GOR is one of the elements called for when calculating the optimal flow rates. Today's calculations are either a time consuming process or they yield inaccurate estimates. Hence there is a need for a simple, online algorithm which can estimate the GOR in a more satisfactory manner.

Another use of the GOR estimate is in view of the marginal GOR (mGOR) $\partial q_{gas}/\partial q_{oil}$. The wells in a reservoir often have a common flowline. The shared flowline make the wells compete for the capacity of the line. The optimal marginal GOR is when we have maximum oil production for a given total gas capacity and the marginal GOR is the same for all wells [Aasheim et al., 2004]. Hence the GOR gives an idea of which wells should reduce their production and which wells should increase their production for maximum oil

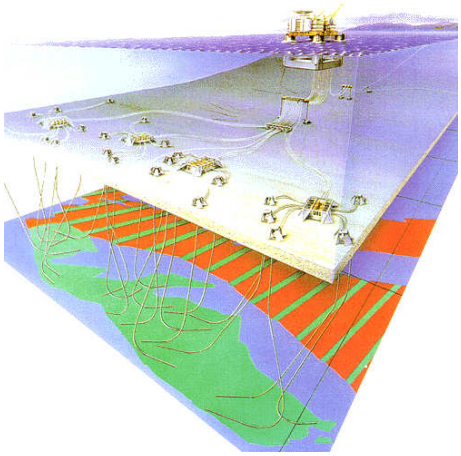


Figure 1.1: A North Sea field - example of wells and flowlines *Source: SPGMediaPLC [2008]*

recovery, see figure 1.2.

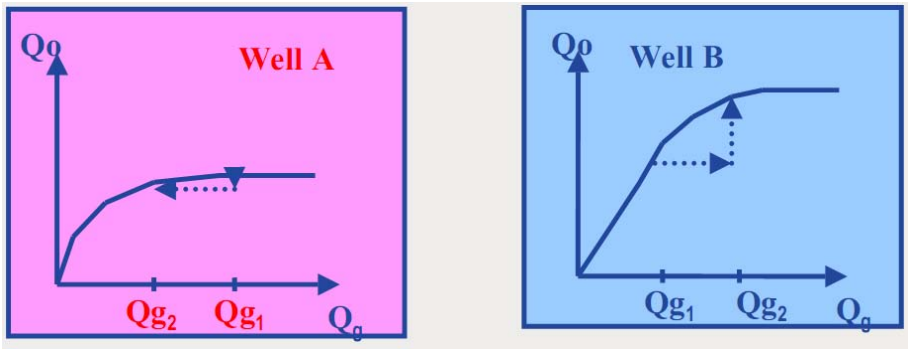


Figure 1.2: Qg_1+Qg_1=Qg_2+Qg_2=Qg_{max} *Source: Aasheim et al. [2004]*

In this thesis a recursive least squares algorithm is used to fit a finite impulse response (FIR) model to the GOR. The model is either presented as a polynomial or as a normalized radial basis function. The algorithm for estimating GOR will make the prevailing production optimization software more efficient and hopefully more accurate in its calculations. It is important to remember that the oil and gas flows used in the algorithm are not measured directly and only represent the best possible estimates of the flows. Hence the GOR estimate will only become as good an estimate as the calculated flow allows.

1.1 Aim and scope of the thesis

As mentioned above it is possible to couple a reservoir simulator taking care of the long term reservoir management and a flow optimizer handling the short term production optimization. ECLIPSE is a renowned software devolped and distributed by Schlumberger which simulates oil and gas reservoirs. ECLIPSE can be coupled with the production optimization tools FlowManager and GAP (General allocation program).

Maximising the utilization of the production systems and avoid losses are key aspects in production optimization. FlowManager and GAP are examples of tools designed to meet the challenges in production optimization taking care of issues like lift rates and well routing. Both tools carry out the production optimization while at the same time honouring the constraints of the system. In their calculations there is a need for an online estimate of the GOR. Among other things the estimate is used for calculating the optimal flow rates.

With GAP the GOR estimate is calculated at different rates by perturbing ECLIPSE. A good estimate calls for perturbing ECLIPSE at numerous rates for each simulation step. This can be a highly time consuming process as ECLIPSE simulation may require as much as three minutes per step.

FlowManager has three choices when estimating the GOR. It can use the same, predefined GOR curve through the entire simulation. The second choice is assuming that the GOR from ECLIPSE is rate independent. Finally it is possible to use a time and rate dependent polynomial, but this option is not yet fully developed.

This thesis will base itself on the third option and focus on developing an effective algorithm for proper updating of the GOR curve. As the other choices of GOR are suboptimal, calculating the GOR as a polynomial or another suitable function is expected to improve the estimate.

1.2 Outline of the thesis

The thesis is divided into three parts.

In Chapter 2 a presentation of hydrocarbon reservoirs and production optimization is made. Chapter 3 contains an overview of system identification with focus on black-box modelling. Possible models and algorithms are presented here. Hence Chapter 2 and 3 present theory and background for the choices made in the thesis.

The algorithm and results are presented in Chapter 4.

Discussion of the results are presented in Chapter 5, in Chapter 6 conclusions are drawn and finally suggestions for future work are found in Chapter 7.

Chapter 2

Hydrocarbon reservoirs

2.1 From reservoir fluid to stock tank volumes

The production of oil and gas from a reservoir can be described as the transformation of volumes of reservoir fluids into stock tank volumes. From the reservoir fluids being trapped in a reservoir to being transformed into stock tank volumes top side, numerous things happen. The temperature and pressure undergo a significant drop. When the dissolved gas is freed at bubble point pressure P_b and comes out of solution the hydrocarbons goes from being a one phase fluid to consisting of two phases. The two phases are liquid oil and the recently liberated gas. This phase change results in a smaller stock tank volume of oil than initially recovered, see figure 2.1.

Example To further illustrate the transformation from one to two phases; imagine a soft drink bottle. When the bottle is unopened you will not be able to see the gas inside. Even though you shake the bottle the gas will remain invisible since the pressure remains the same. But when you unscrew the cap, the pressure drops and the gas will immediately appear as bubbles in your soft drink.

In the primary reservoir production there is no significant change in temperature, but there is a substantial pressure drop near the well at about 100 *bar*. As the pressure drops, the volume of the gas increases faster than the pressure drops.

2.1.1 Volume factors

Since most measurements of oil and gas are made at surface *volume factors* are needed to convert the volumes from stock tank volumes to reservoir volumes. The following terms are employed to convert standard volume quantities to reservoir volumes:

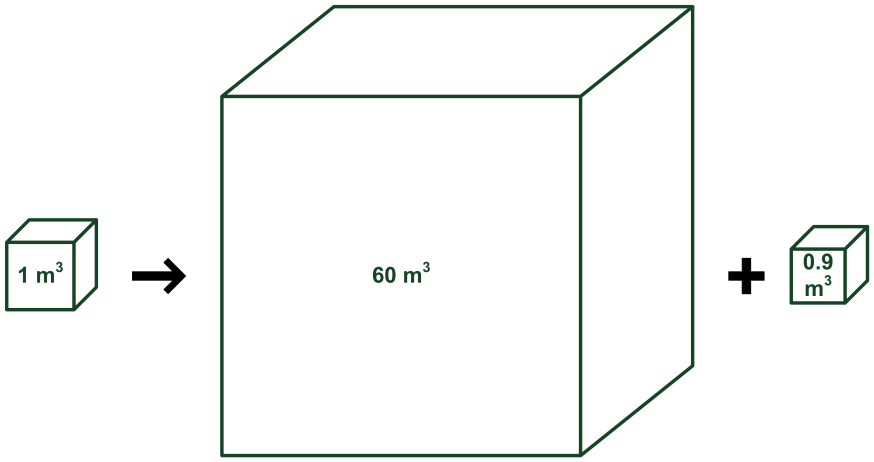


Figure 2.1: 1 m^3 oil in the reservoir becomes approximately 60 m^3 gas and 0.9 m^3 oil topside

Oil formation volume factor (FVF), B_O:

Oil and dissolved gas at reservoir conditions divided by oil volume at standard conditions.

$$B_O = \frac{V_O}{V_{ON}} \quad \left[\frac{Rm^3}{Sm^3}\right] \tag{2.1}$$

Solution Gas-Oil Ratio (GOR), R_S:

The standard volume of gas which will be recovered with a unit of stock tank oil volume.

$$R_S = \frac{V_{OGN}}{V_{ON}} \quad \left[\frac{Sm^3}{Sm^3}\right] \tag{2.2}$$

The R_S differs from the regular GOR in that regular GOR is the ratio of produced gas to produced oil where the gas consists of both initially dissolved gas and excess gas for instance gas from a gas cap. B_O, R_S and GOR are all pressure dependent functions.

2.2 A brief description of production optimization

The ultimate goal for any oil and gas reservoir is to maximize the net present value of the asset. This often means maximizing oil recovery and also maximizing daily production. The goal may be divided into two subproblems; long term reservoir management and short term production optimization. The term *reservoir management* typically means optimization of injection rates and reservoir drainage on the time scale of months and years while *production optimization* aims to maximize production on a daily basis [Saputelli et al., 2003]. The two are connected through constraints from the reservoir management on the production optimization. Production optimization may be posed as a real time optimization problem where one has to measure, calculate and control the system. This thesis will focus on production optimization.

To increase long term recovery from oil and gas fields, water and gas can be injected into the reservoir to increase reservoir pressure. Water flooding is a frequently used technique in the North-Sea [Zolotukhin and Ursin, 2000]. Artificial lift methods, for instance a gas-lift, are used to increase short term production [Schlumberger, 2003]. Such methods are, however, quite complicated to implement. For instance, oil entrapment can occur when heterogeneous formations are flooded with water. Another problem is gas and water coning close to a well. This is due to the higher mobility of water and gas compared to oil, and the coning effect can result in an increased gas-oil ratio (GOR), higher water cut (WC) and a low oil production rate (figure of coning gas?) [Zolotukhin and Ursin, 2000].

To determine the flow rates of the three phases oil, gas and water in the pipes, well tests are performed. Well tests involve routing one of the wells to an independent test separator. With single rate well tests rates are measured for one set point while multirate well tests measure rates for several set points.

The total amount of oil, gas and water which can be processed is constrained by the capacity, typically the separator capacity, of the downstream processing system. Normally some of these capacities are fully utilized. This makes it difficult to test wells since it incurs costs because of reduced capacity during testing. It is worth mentioning that in practice the oil recovery factor rarely exceeds 70 % and it may be as low as 5-10 %.

In production optimization a mathematical model of the production system (near well region, wells and collection system) is an important tool even though the use of such tools is not common practice. The models can be either empirical, black-box models or models based on physics. Further, they may be linear or nonlinear. Either way a model will always be an approximation of the real system. There are several reasons for this:

- The choice of model structure may be wrong, or anyhow suboptimal.
- There may be unmodelled disturbances and measured data may differ from the actual

production rates. For instance pipes may rust and wax can gather inside them and hence become a disturbance.

- Lack of informative data. Multi rate well tests will for instance include more information than data from normal production, but such tests are scarce.

To be able to perform a trustworthy model identification the data provided should be real production data. On the other hand, today's comprehensive reservoir simulators are highly credible and will probably produce quite realistic data. In case one does not have access to real production data, synthetic data from a reservoir simulator could be applied for model identification. If that is the case one should be careful when using it in a real production environment and use real production data to validate and evaluate the model.

This thesis will use an empirical modelling approach. Hence, a suitable model structure must be chosen. Thereafter model parameters are estimated and the model is validated. Typically the data only cover parts of the operating conditions which are of interest to an optimizer. This implies that model uncertainty differs in different parts of the operating regime (i.e. different flow rates) for a well. We are seeking a static model suitable for daily production optimization. This means that very fast dynamics, like impulses, and slow varying dynamics should be filtered out before using them for estimation and validation. Slow varying dynamics will be included by adjusting model parameters.

Chapter 3

Nonlinear black-box modelling

The field of system identification is far too broad to be completely covered in this thesis. The author has focused on nonlinear, black-box modelling as an empirical approach has been employed. For more theory on system identification, please inquire Ljung [1987] or Sjöberg et al. [1995].

3.1 System identification

With system identification, the key problem is to find a suitable model structure so that one is able to create a good model for the system. The construction of a model from data basically involves three parts.

- A data set
- A model structure
- A rule to assess the quality of the models

There are several different model structures to choose from and some of them will be presented here. Before choosing a model structure, one has to look at the data and familiarize oneself with it. As the choice of model structure is the most important and most demanding choice, there is no point in just picking a model out of nowhere. If data is available the choice of model structure should be based on prior knowledge from this data.

One can distinguish between three different types of modelling.

- *White-box modelling* is when one has physical insight and a priori knowledge about the system.

- *Black-box modelling* has no reference to the physical background of the system, either because the physical insight is not available or it is just not used. The model structure chosen will be some standard models which one knows have been successful previously.
- *Grey-box modelling* is the case when some physical insight is available, but there are still a number of unknown, free parameters which are to be determined through system identification. Most systems are to be put in this category.

Models can be either linear or nonlinear. A wise approach is to always try simple things first. So even if your system seems to be a nonlinear system, it might be sufficient to describe it with a simple, linear model. Nonlinear structures are much more complicated than linear ones.

3.2 The problem

The system identification problem is to look for a relationship between past inputs and outputs and future outputs. A finite number of past inputs $u(k)$ and outputs $y(k)$ can be placed in a vector $\varphi(t)$.

$$\varphi(t) = [y(t-1) \dots y(t-n_a)\ u(t-1) \dots u(t-n_b)]^T \tag{3.1}$$

The vector $\varphi(t)$ is called the *regression vector* and the clue is to find the relationship between the next output $y(t)$ and the regression vector $\varphi(t)$. A model has the general structure

$$\widehat{y}(t) = \theta^T \varphi(t, \theta) \tag{3.2}$$

where θ is a finite parameter vector. The hat on $y(t)$ is to emphasize that $\widehat{y}(t)$ is the best guess of $y(t)$ given $\varphi(t, \theta)$.

3.3 Regressors

3.3.1 Linear black-box models

There are several different linear black-box models to choose from, and the most common can be summarized in the generalized model structure described in equation 3.3. Equation 3.3 and table 3.1 are previously presented in Ljung [1987].

$$A(q)y(t) = \frac{B(q)}{F(q)}u(t) + \frac{C(q)}{D(q)}e(t) \tag{3.3}$$

The simplest model is the finite impulse response (FIR) model, where $A = C = D = F = 1$. q is the shift operator, so for instance B(q) in the FIR model is a polynomial in q^{-1}.

Table 3.1: Special cases of equation 3.3: Common linear black-box models

Polynomials used in eq. 3.3	Name of model structure
B	FIR (Finite impulse response)
AB	ARX
ABC	ARMAX
AC	ARMA
BF	OE (Output error)
$BFCD$	BJ (Box-Jenkins)

To calculate the prediction the expression from equation 3.2 is used. The regressors in $\varphi(t, \theta)$ can be chosen amongst

- $u(t-k)$ $[B]$
- $y(t-k)$ $[A]$
- $\widehat{y}_u(t-k|\theta)$ $[F]$
- $\epsilon(t-k) = y(t-k) - \widehat{y}(t-k|\theta)$ $[D]$
- $\epsilon_u(t-k) = y(t-k) - \widehat{y}_u(t-k|\theta)$ $[C]$

A FIR model requires that the past inputs covers the whole dynamic response time. But it also has the advantage of never being unstable in simulation. With the extra regressors yielding an ARX model it becomes possible to cover slow responses with fewer regressors. When the regressors include past predicted the flexibility is increased but it may also introduce instability and result in that convergence to a global minima cannot be guaranteed [Sjöberg et al., 1994].

3.3.2 Nonlinear black-box models

The nonlinear black-box models follow the same nomenclature as for the linear models.

- NFIR
- NARX

- NOE
- NARMAX
- NBJ
- nonlinear state-space models, use past components of virtual outputs

The most commonly used models are NFIR and NARX. As for linear models neither NFIR nor NARX models have any estimated values in their regressors.

In the nonlinear case the prediction has a different structure than in the linear case

$$\widehat{y}(t|\theta) = g(\varphi(t), \theta) \tag{3.4}$$

where $g(\varphi, \theta)$ is a nonlinear function parameterized by θ.

3.4 Model structure

It is natural to think of equation 3.4 as a family of functions

$$g(\varphi(t), \theta) = \sum \alpha_k g_k, \quad \theta = [\alpha_1 \dots \alpha_n]^T \tag{3.5}$$

where g_k is referred to as a basis function. Basis functions give a foundation for nonlinear black-box model structures and there are a number of choices of these functions. This thesis will only focus on a few basis functions and the author refers the reader to Ljung [1987] and Sjöberg et al. [1995] for further reading on the subject.

3.4.1 Single variable basis functions

The basis functions below do also have multi variable properties, but they will be represented as single variable basis functions here.

Radial basis functions

A radial basis function (RBF) can take the form

$$g_k(\varphi) = e^{-\beta_k(\varphi - \gamma_k)} \tag{3.6}$$

and is shaped like a bell.

The RBF offers a localization property. The γ values can be chosen to the areas on

the x axis where one wants to direct the focus. For instance areas with higher concentrations of data points. The β value gives the steepness of the curve. With a large β value the corresponding basis function will have its prevailing area in the close vicinity of the γ while with a small β value the basis function will influence a more extensive area. So the closer the input is to the centre of the bell, the larger the response of the basis function or node.

In normalized RBF (NRBF) networks the output activity is normalized by the total input activity

$$G_k = e^{-\beta_k(\varphi-\gamma_k)} \tag{3.7}$$

$$g_k(\varphi) = \frac{G_k}{\sum_{i=1}^{N} G_k}, \quad N = number\ of\ nodes \tag{3.8}$$

In standard RBF nets the weights, which equals one in equation 3.7, determine how much each node contribute to the output. With NRBF nets the roles are switched and the activity of the nodes determine which weights contribute most to the output. NRBF networks also improves its interpolation skills over standard RBF networks. Even in regions where no nodes produce a strong response, the NRBF can generate a significant output value [Bugmann, 1998].

Sigmoid basis functions

$$g_k(\varphi) = \frac{1}{1 + e^{-\beta_k(\varphi-\gamma_k)}} \tag{3.9}$$

The most common sigmoid function has $\beta = 1$ and $\gamma = 0$ and has an S shape. The sigmoid function has a localization property just like the radial function, where the β value gives the steepness of the curve and the γ value gives the focus area on the x axis.

Polynomials

Polynomials are well known, simple functions and they take the shape of

$$g_k(\varphi) = \sum_{k=0}^{N} \varphi^k = 1 + \varphi + \varphi^2 + \varphi^3 + \ldots + \varphi^N \tag{3.10}$$

If a polynomial is used as basis function, one gets a *linear regression model structure*.

See the paragraph on *B splines* for further use of polynomials.

3.4.2 Named model structures

In this section some popular named structures which can be used as basis functions will be introduced.

Wavelets

According to Kaiser [1994] wavelets are similar to windowed Fourier transform (WFT) and can represent functions with discontinuities and local spikes, but they are better suited than the WFT to zoom in on short lived frequency phenomena. Wavelets also have multi resolution capabilities and they take the form

$$g_{k,j}(\varphi) = 2^{j/2}\kappa\left(2^{j}\varphi - k\right), \quad j,k \in \mathbb{Z} \tag{3.11}$$

Neural networks

Neural nets are relatively young compared to other functions. They have good general capabilities and are a popular choice of model structure. A neural network learns on the gathered data and detects their underlying relationships - the more data, the better [Sjöberg et al., 1994].

Neural networks can be used with a basis of polynomials (Taylor or Volterra expansion), NARX-structure expansion or sigmoid-functions.

$$g_k(\varphi) = \sigma\left(\beta_k\varphi + \gamma_k\right) \tag{3.12}$$

where the most common choice of σ is the sigmoid function

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{3.13}$$

Equation 3.13 gives a smooth, differentiable model and yields, together with equation 3.12, a one hidden layer feedforward sigmoid neural net. Continuous functions with only one input have no advantage in using more than one hidden layer. There exists a large number of neural networks, feedforward is only one example and the only one which will be mentioned here. For more information on neural networks, please read Sjöberg et al. [1994].

B splines

B splines are piecewise polynomials where the connections have continuous derivatives. They are nice, simple functions and they can be made as smooth as desired. A general polynomial is shown in equation 3.10.

3.4.3 The bias-variance trade off

The bias-variance trade off is an important issue in data modelling. It can be explained in simple terms and should not be neglected.

Models with too few parameters become inaccurate due to a large bias. A model with a large bias does not have enough flexibility. For instance fitting a linear function to a set of samples with the shape of a parabola will yield a large bias.

If a model has too many parameters it is inaccurate because of large variance in its predictions. Such models are too sensitive to the details of the sample. Although an overfitted model may appear to be performing well, in reality it follows the data too closely and will perform poorly on new data.

Identifying a good model therefore requires a model with not too few nor too many parameters. Hence a trade off between the bias and the variance must be found by tuning the number of parameters [ACCESS].

3.5 Optimization algorithms

In this section methods for estimating the parameters will be presented. An intuitive approach would be in some way to minimize the error between the real signal and the predicted value.

In many cases it is useful to have an on-line model of the system available. Then the algorithm is called a recursive algorithm, and the update is based only on measurements made up until now. This section will give an introduction to recursive algorithms.

3.5.1 Recursive methods

According to Ljung [1987] there are three classes of recursive methods:

- Recursive prediction error methods (RPM)
- Recursive pseudolinear regressions (RPLR)
- Recursive instrumental-variable methods (RIV)

Recursive least squares

The least squares (LS) method is a special case of the prediction error identification method which yields linear regression. The benefit of the recursive LS (RLS) algorithm is that there

is no need to invert matrices, thereby saving computational power. Another advantage is that it provides intuition behind such results as the Kalman filter. The most beneficial property of the LS method is that it their exists no local minima other than the global ones. The global minima can be found efficiently and straight forward.

Presenting the weighted least squares criterion using the same notation as Ljung [1987]

$$\widehat{\theta} = \arg\min_{\theta} \sum_{k=1}^{t} \beta(t,k) \left[y(k) - \varphi^T(k)\theta \right]^2 \tag{3.14}$$

$$\widehat{\theta}_t = \bar{R}^{-1}(t) f(t) \tag{3.15}$$

$$\bar{R}(t) = \sum_{k=1}^{t} \beta(t,k)\psi(k)\psi^T(k) \tag{3.16}$$

$$f(t) = \sum_{k=1}^{t} \beta(t,k)\psi(k)y(k) \tag{3.17}$$

To avoid inverting $\bar{R}$ at each step the matrix P used is introduced

$$P(t) = \bar{R}^{-1}(t) \tag{3.18}$$

Then the matrix inversion approach can be summarized below.

$$\widehat{\theta}(t) = \widehat{\theta}(t-1) + L(t)\left[y(t) - \psi^T(t)\widehat{\theta}(t-1) \right] \tag{3.19}$$

$$L(t) = \frac{P(t-1)\psi(t)}{\lambda(t) + \psi^T(t)P(t-1)\psi(t)} \tag{3.20}$$

$$P(t) = \frac{1}{\lambda(t)} \left[P(t-1) - \frac{P(t-1)\psi(t)\psi^T(t)P(t-1)}{\lambda(t) + \psi^T(t)P(t-1)\psi(t)} \right] \tag{3.21}$$

Forgetting factor λ

The RLS makes use of a forgetting factor $0 < \lambda < 1$. The smaller λ is, the smaller contribution of previous samples. This makes the filter more sensitive the recent samples, which means more fluctuations in the filter co-efficients. The $\lambda = 1$ case is referred to as the growing window RLS algorithm. Lambda is recommended to take on a value just below 1, typically in the interval $0,95 - 0,99$. One reason for this is that previous data must not be assigned too little importance as the parameters are slowly varying. Another reason is that if the process goes through a phase where nothing happens, that is y_t and u_t and hence ψ is approximately zero, P(t) will grow exponentially [Henriksen, 1998].

Recursive prediction error methods

$$\epsilon(t) = y(t) - \widehat{y}(t) \tag{3.22}$$
$$\widehat{\theta}(t) = \widehat{\theta}(t-1) + \gamma(t)R^{-1}(t)\psi(t)\epsilon(t) \tag{3.23}$$
$$R(t) = R(t-1) + \gamma(t)\left[\psi(t)\psi^T(t) - R(t-1)\right] \tag{3.24}$$

Depending on the underlying model structure and the choice of $R(t)$ the equation above corresponds to specific algorithms in the family of methods called recursive prediction error methods [Ljung, 1987].

Recursive pseudolinear regressions

$$\widehat{y}(t) = \varphi^T(t)\widehat{\theta}(t-1) \tag{3.25}$$
$$\epsilon(t) = y(t) - \widehat{y}(t) \tag{3.26}$$
$$\widehat{\theta}(t) = \widehat{\theta}(t-1) + \gamma(t)R^{-1}(t)\varphi(t)\epsilon(t) \tag{3.27}$$
$$R(t) = R(t-1) + \gamma(t)\left[\psi(t)\psi^T(t) - R(t-1)\right] \tag{3.28}$$

In the RPLR φ contains entries constructed from using past models. The RLS method is also a member of the RPLR family of algorithms [Ljung, 1987].

Recursive instrumental variable method

$$\widehat{\theta}(t) = \widehat{\theta}(t-1) + L(t)\left[y(t) - \psi^T(t)\widehat{\theta}(t-1)\right] \tag{3.29}$$
$$L(t) = \frac{P(t-1)\zeta(t)}{\lambda(t) + \psi^T(t)P(t-1)\zeta(t)} \tag{3.30}$$
$$P(t) = \frac{1}{\lambda(t)}\left[P(t-1) - \frac{P(t-1)\zeta(t)\psi^T(t)P(t-1)}{\lambda(t) + \psi^T(t)P(t-1)\zeta(t)}\right] \tag{3.31}$$

The RIV method is quite analogous to that of RLS. For deeper insight into the RIV method, please inquire Henriksen [1998] or Ljung [1987].

Kalman filter

The Kalman filter is an alternative to the forgetting factor methods. It is a state estimator with a recursive nature and can all the same be used as a parameter estimator. The Kalman filter has its strength in estimating the states from a series of measurements contaminated

with noise and is nice to use if certain parameters vary more than others as they can be weighted separately [Henriksen, 1998]. Shown below are the system equations

$$\theta(t+1) = \theta(t) + w(t) \tag{3.32}$$

$$y(t) = \psi^T(t)\theta(t) + v(t) \tag{3.33}$$

where it is assumed that θ varies like a random walk and w and v are white and Gaussian.

When using the Kalman filter as a parameter estimator the equations may be presented like this

$$\widehat{\theta}(t) = \widehat{\theta}(t-1) + L(t)\left[y(t) - \psi^T(t)\widehat{\theta}(t-1)\right] \tag{3.34}$$

$$L(t) = \frac{P(t-1)\psi(t)}{R_2(t) + \psi^T(t)P(t-1)\psi(t)} \tag{3.35}$$

$$P(t) = P(t-1) - \frac{P(t-1)\psi(t)\psi^T(t)P(t-1)}{R_2(t) + \psi^T(t)P(t-1)\psi(t)} + R_1(t) \tag{3.36}$$

$$R_1 = Ew(t)w^T(t) \tag{3.37}$$

$$R_2 = Ev(t)v^T(t) \tag{3.38}$$

R_1 in prevents the gain L from tending to zero.

If we are sure about the measurements the measurement error covariance R_2 decreases to zero and the gain L decreases and weights residual more than prediction. If we on the other hand are sure about the predictions the prediction error covariance P decreases to zero and L increases and weights prediction more than residual.

For all of the algorithms mentioned above $\widehat{\theta}(0)$ is the initial guess of the parameter vector and $P(0)$ reflects our confidence in this guess.

3.6 Validating the model

An important step in system identification is evaluating and validating the model and algorithm. There are several options when it comes to model validation. Ljung [1987] proposes several ways of comparing models. Visual inspection is one option. Just looking a plots of $y(t)$ and $\widehat{y}(t)$ can be quite helpful. Another way of comparing models is to compare numerical values like the variance

$$J_k(m) = \frac{1}{N}\sum_{t=1}^{N} |y(t) - \widehat{y}_k(t|m)|^2 \tag{3.39}$$

the sum of the residuals

$$J_k(m) = \sum_{t=1}^{N} |y(t) - \widehat{y}_k(t|m)| \tag{3.40}$$

or the standard deviation

$$J_k(m) = \sqrt{\frac{1}{N} \sum_{t=1}^{N} |y(t) - \widehat{y}_k(t|m)|^2} \tag{3.41}$$

Yet another validation method is cross-validation. There are various types of cross-validation. With holdout validation the data are never crossed over. A number of observations are removed from the initial sample and used as validation data. Normally about a third of the initial sample is used for validation. Another variant is the leave-one-out cross-validation. The advantage of this method is that there is no need to save fresh data for the validation. A model is made up of the entire data set except for one observation. This is done once for every N observations in the data set and in the end one can sum up the N corresponding squared errors.

3.7 Some general advice

A selection of models and model structures have been presented. But how does one choose which regressors to use? How does on choose the most suitable model structure? There are no simple answers to these questions. But Sjöberg et al. [1995] present some rules that can be helpful.

1. *Familiarize with the data.* This is an obvious step and can often be very revealing.
2. *Try simple things first.* For instance one should try linear models first; they may be sufficient.
3. *Attain physical insight.* This may give a clue of whether to turn a measurement into a regressor.
4. *Validation data.* If possible some data should be saved for evaluation.

Chapter 4

A case study

4.1 Presenting the hydrocarbon fields

There is data available from 11 different wells, located in three separate anonymous reservoirs. The hydrocarbon fields provided for testing the algorithms are presented below

- Field A - five wells
- Field B - one well
- Field C - five wells

Field A

Field A is a North Sea reservoir with an enormous gas cap. The field spreads out over a large area and has an oil layer which height varies from about 4 to about 30 meters, and has a typical height of about 12 meters. A typical Field A well would go through a short period of time before gas brakes through, the short period stretching from a couple of weeks to months.

It is important to remember that the flow rates for oil an gas are estimated based on the pressure and temperature of the three phase flow and the pressure drop across the choke. Although there are three measurements the algorithm calculating flow can only find estimates for flow of oil and gas and not for the water cut (WC). Hence the WC holds a fixed value periodically.

Five wells from separate locations throughout Field A are provided.

- Well #1 - Well #1 has some variations in choke position. There is data from 103 days of production and the WC is constant for the entire period.

- Well #2 - This well has production data from 127 days. The WC in this well not constant.
- Well #3 - The well has a quite stable production rate with some variations among other things there is some slugging towards the end. Production data from 275 days is provided. The data from this well does not seem reliable and the algorithm will not be applied to Well #3. A graphical representation of Well #3 is presented in figure 4.1(c).
- Well #4 - Well #4 has production data from 89 days and has a close to constant WC.
- Well #5 - The well has a low, stable production and production data from 68 days. The WC holds a constant value throughout the entire production period.

Measurements have been made every ten minutes and so there is quite an amount of observations available from these wells.

Field B

Field B is also a North Sea reservoir. Field B only had one interesting , that is rate dependent, well. This well develops a gas cone. It shuts down for short periods of time to get rid of the gas cone. When it starts back up the GOR value is reduced, but it will only develop another gas cone to produce more and more gas as time moves on. There is data from 131 days for this well and measurements have been made every tenth minute. In Field B as in Field A the flow rates for oil and gas are estimated rates.

Field C

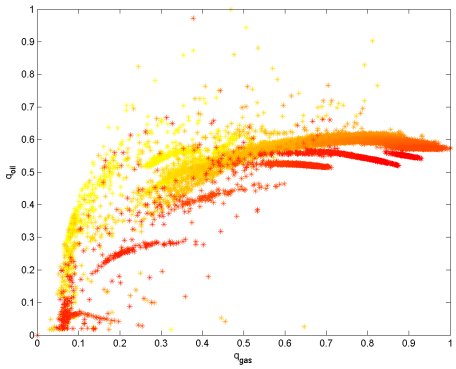
The data from Field C is not gathered from a real reservoir; Field C is a simulated Eclipse reservoir. The field is in production for ten years and measurements are only available every third to tenth day inconsequently. At the beginning of every January, the well is closed for a duration of one month to let a possible gas cone back.

In all of the wells except from Well #1 the WC is significantly higher than the oil rate. In Well #1 the WC is higher than but still comparable to the oil flow rate. The gas rate has been driven to a constant level for most wells except for Well #2. Some degree of coning gas is helpful and often necessary to give the well a natural lift. In Well #2 the fluid becomes heavier and heavier and the gas cone does not yield a sufficient lift. Hence the well closes down after six years of production.

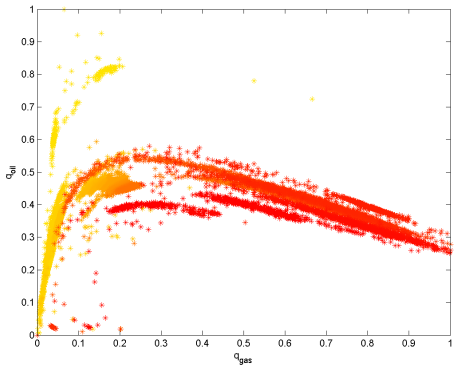
The production data from each well is presented in figure 4.2. After familiarizing with the wells in Field C it is evident that the data from the Eclipse simulations is not suitable for the objective of this thesis.

Graphical display of the gas oil ratios

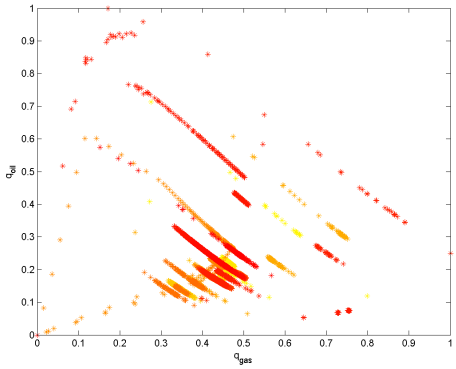
The GORs from the three fields are presented in figures 4.1 and 4.2. Notice that the figures have gas flow q_{gas} along the x-axis, and not time as could be expected. The oil flow q_{oil} is found along the y-axis. The colour in the plots represent the time aspect with the colour yellow marking the early GOR and the colour red marking the final GOR. Due to confidential data the flow rates have been normalized and the well names have been made anonymous.



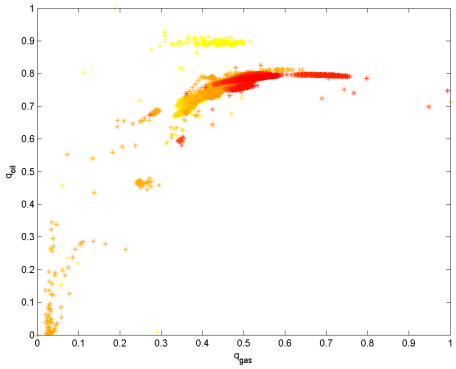
(a) Field A - well #1



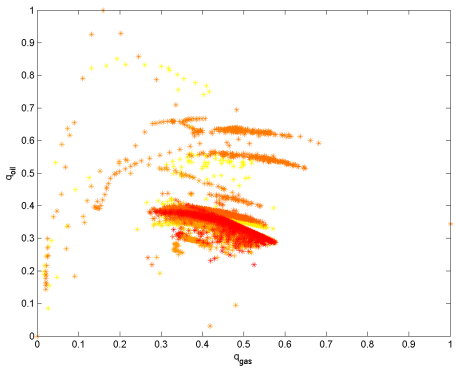
(b) Field A - well #2



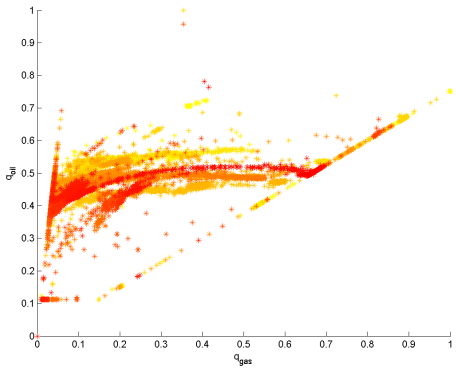
(c) Field A - well #3



(d) Field A - well #4

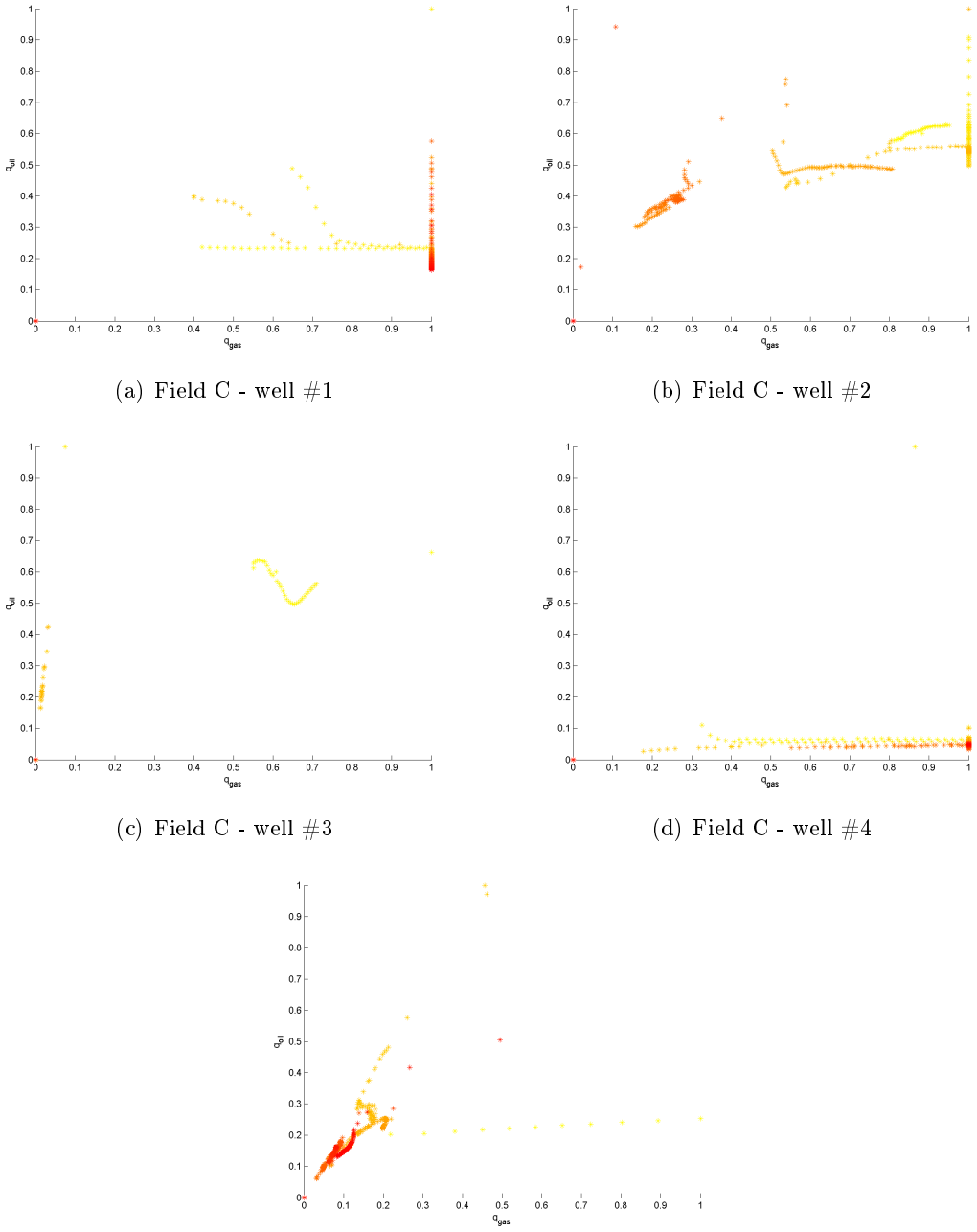


(e) Field A - well #5



(f) Field B - well #1

Figure 4.1: The wells in field A and B



(a) Field C - well #1

(b) Field C - well #2

(c) Field C - well #3

(d) Field C - well #4

(e) Field C - well #5

Figure 4.2: The wells in field C

The shape of the GOR plots can be explained by the well being placed close to the water oil contact (WOC) and in the beginning the produced fluid will contain large amounts of water. After some time of producing lots of water the WOC will move downwards and make room for more oil, hence the increasing oil profile in the beginning of some of the plots. Most of the plots from the the data sets look like polynomials hence a solution containing a polynomial is plausible.

4.2 Choice of model and algorithm

Although a polynomial was the initial choice of function there is no reason why another function should be less fitted in adapting the GOR curve. It is important that the estimate is proper, but another issue is not to make a very complicated model. We must find a balance between a model that is not too complicated and not too simple. In addition we have to choose which regressors to use and a suitable algorithm for optimization.

4.2.1 Regressors

Recall from chapter 3 that there are several choices of regressors to choose from. We have to find a good compromise between the complexity of the model and its efficiency in describing the dynamics of the system. A good starting point would be a nonlinear finite impulse response (NFIR) model. This choice model guarantees stability. One input is selected namely the one step delayed gas. Hence the regression vector $\varphi(t)$ can be displayed.

$$\varphi(t) = [q_{gas}(t-1)] \tag{4.1}$$

4.2.2 Model

The choice of model and model structure is more complicated. With a look at the GOR plots in figures 4.1 and 4.2 a polynomial shape seems to be a reasonable choice. Since the nature of the GOR gives a decreasing oil rate with an increasing gas rate, an even numbered polynomial basis function seems appropriate. According to the bias-variance trade-off a polynomial with too many parameters will become too sensitive to the samples. Hence a second order and a fourth order polynomial are selected.

$$g_k(\varphi) = 1 + q_{gas}(t-1) + q_{gas}(t-1)^2 \tag{4.2}$$

$$g_k(\varphi) = 1 + q_{gas}(t-1) + q_{gas}(t-1)^2 + q_{gas}(t-1)^3 + q_{gas}(t-1)^4 \tag{4.3}$$

The second choice in model structure is the NRBF which yields a radial-basis neural network. With a normalized structure the functions $g_k \geq 0$ and the sum of the functions equals 1. With g_k as in equation 3.7 $g(\varphi)$ will always be convex and the minimization will attain a unique solution.

$$g(\varphi) = \sum g_k \theta_k \tag{4.4}$$

In standard RBF nets, the weights determine how much each hidden node contribute to the output. In normalized RBF nets, see equation 3.7, the activity of the hidden nodes determine which weights contribute most to the output. For instance, in the extreme case where only one of the hidden nodes is active, then the output of the net becomes equal to the weight corresponding to that hidden node, whatever its level of activity.

Different model orders should be tested

$$G_k = e^{-\beta_k(\varphi-\gamma_k)^2} \tag{4.5}$$

$$g_k(\varphi) = \frac{G_k}{\sum_{i=1}^{N} G_k}, \quad N = 2,3,4,5,6 \tag{4.6}$$

A possible modification is to replace the constant θ with a linear function. These functions help smooth the function when the node takes on a constant value and is an alternative to increase the number of parameters or nodes.

$$\theta_1 = \theta_{11} + \theta_{12} q_{gas} \tag{4.7}$$

This gives a nice touch to the model as the bias-variance trade off is still prevailing.

4.2.3 Algorithm

Because of the demand for an online algorithm, the optimization algorithm has to be recursive. A well known algorithm is the recursive least squares (RLS) filter and its simplicity in use and understanding is appealing. From chapter 3 we remember that by implementing the RLS algorithm ensuers a robust and convex implementation and that the global minimum will always be found with this strategy. Nonlinear minimization on the other hand is not convex. It requires numerical techniques like Gauss-Newton for solving and may converge very slowly or not at all and it may continually change directions. Furthermore, Ljung [1987] states that the least-squares algorithm is applicable for most reasonable

nonlinear systems, making it a good idea to try using this scheme before employing some nonlinear search algorithm.

A clue in system identification is not to estimate what we already know. With the linear regression model using a polynomial basis function only one class of parameters θ needs to be estimated. Using an NRBF we obtain two additional sets of parameters namely the localization parameters γ and the scale parameters β. With a priori knowledge of the distribution of the data these parameters can be selected beforehand. For online systems some parameter selection technique should be employed. According to Sjöberg et al. [1995] there are no well suited selection techniques for sigmoid functions while for the NRBF several techniques exist. If such techniques are to be included in the RLS algorithm there will be more parameters than regressors and linear regression will no longer be possible. Due to the limited time scope of this thesis and the fact that data is available to us beforehand no selection techniques will be implemented.

There are several versions of the RLS algorithm. We employ the matrix inversion approach introduced in chapter 3.

4.2.4 Validation

To be able to choose among the implemented models some kind of validation is necessary. We have chosen the hold out validation technique. Ljung [1987] mentions that the sum and the mean of the residuals some of the choices when it comes to evaluating models. Together with summing up the prediction errors the hold out validation will give the foundation for selecting the best possible model. The GOR data used in this thesis is rate and time dependent and so dividing the data in two parts viewed in the light of time would be futile. Hence when estimating the models every second sample of data is saved for validation.

To validate the algorithm it should be compared to the prevailing techniques for estimating the GOR. These are not available to us and such a validation should be performed before implementing the new algorithm with the production optimization software it is intended to work with.

4.2.5 Implementation

The algorithm is implemented in MATLAB. MATLAB is a high-level programming language and numerical computing environment. It is a well known programming tool in the control engineering society and allows for implementation of online algorithms. The code has been produced entirely by the author except for some helpful methods already implemented in MATLAB.

The data production data is presented in Excel files and made available to MATLAB through the function *xlsread*. A FIR model is estimated with the recursive function *rarx*. *rarx* handles only single output models while multiinput models are made possible. It has several options in algorithms. The algorithm selected is a recursive least squares algorithm with forgetting factor λ, the value of λ chosen on behalf of the recommendations in chapter 3. Because of the large number of observations in the data sets λ should be quite close to 1 and it is chosen to $\lambda = 0.9998$. This will make the algorithm consider the approximately last 5000 observations. For the polynomial basis functions where the data is divided into batches λ can be smaller and is chosen to $\lambda = 0.993$ remembering about 140 batches of data. The matrix $P(0)$ is chosen to take on a value of 10 for each $\theta(0)$ as we do not trust the initial guess of $\theta_k(0) = 0$.

4.3 Test system: A random polynomial

A random polynomial was created, equation 4.8, on the basis of the $arctanfunction$ and a data set was generated from the polynomial. The purpose of this was to test out basis functions and a recursive algorithm on a well known system, and manipulate it as desired.

$$f(x) = 0.0080x^3 - 0.1376x^2 + 0.7584x + 5.0468 \tag{4.8}$$

The three different test cases that were selected are

- Data points without noise
- Data points with noise
- Only a few, grouped data points available

4.3.1 Noise free signal

The main function is shown in figure 4.3. Polynomials up to fifth degree and normalized radial functions of 8th and 4th degree have been used as basis functions.

Polynomial

Figure 4.4(a) shows the results from matching the polynomials to the signal. It is obvious from the figure that a polynomial with a degree higher than two is needed to be able to adapt nicely to the main function. But it is also clear that a degree higher than three is futile, see figure 4.4(b).

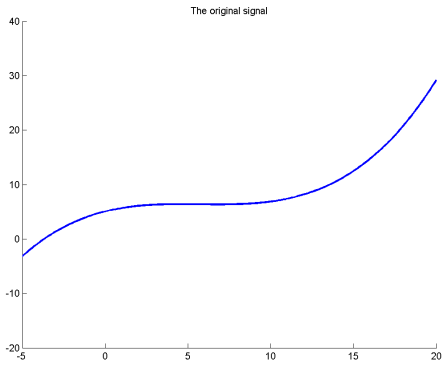
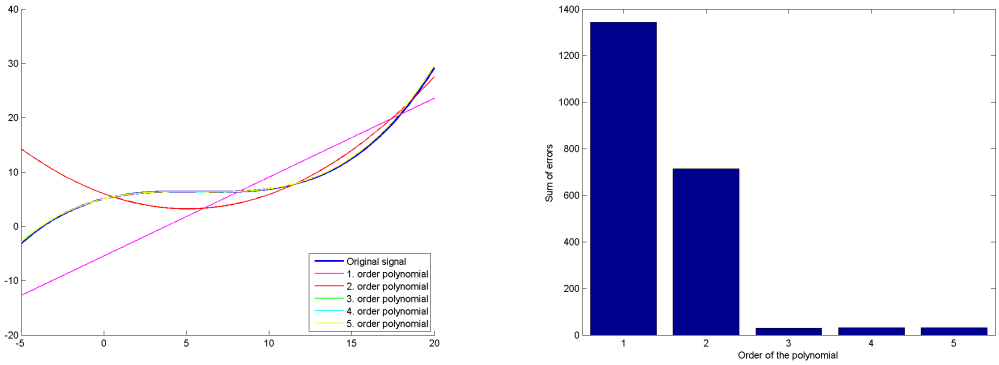


Figure 4.3: Random polynomial



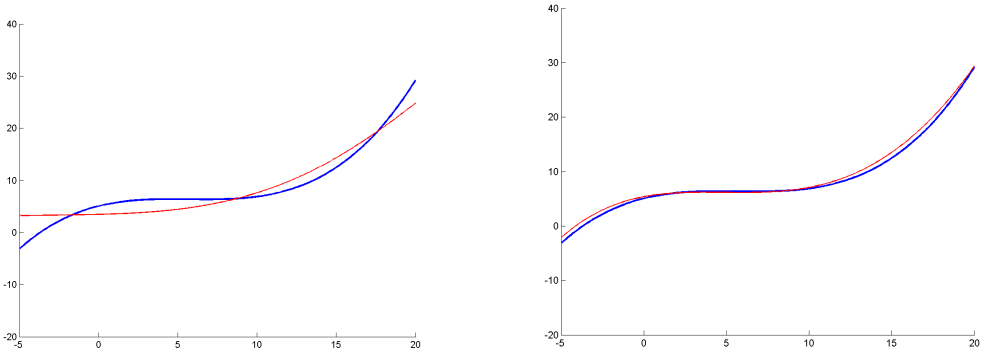
(a) Adapting the five polynomials

(b) Summing up the errors

Figure 4.4: Basis function - Polynomials of degree 1 to 5

Radial function

The match using NRBFs is shown in figure 4.3.1 and the results are not as satisfactory as it was with the polynomials. The signal is not contaminated with noise and even so the normalized radial functions can only trace and not adapt to the main function when it has eight nodes. A reason for this can be that the localization parameters γ and the scale parameters β are not optimal. This goes for both the NRBF and the sigmoid functions as no explicit algorithm for selecting the parameters has been employed. With four nodes the residual sum is decreased, but still not as low as for the three polynomials of highest order.



(a) Normalized radial function of 8th degree (b) Normalized radial function of 4th degree

Figure 4.5: Basis function - Normalized radial functions

Table 4.1: Summarized error - noise free signal

P1	*P2*	*P3*	*P4*	*P5*	*NRBF1*	*NRBF2*
1342.9	714.3	29.9	31.2	31.1	425.8	178.5

4.3.2 Signal with noise

Noise was added to the signal from the main function. The new, noisy signal can be seen together with the noise free signal in figure 4.6. The same system identification as in the previous section was tested on the noisy signal.

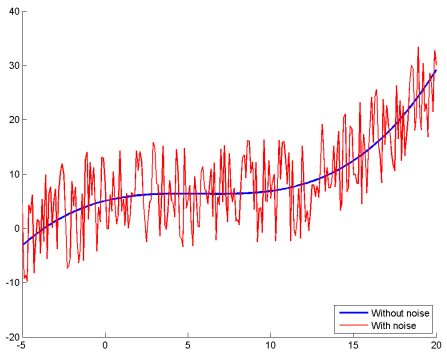
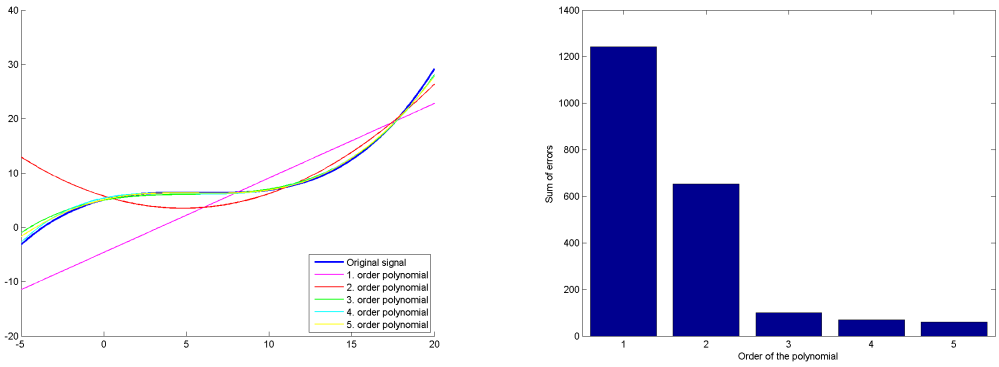


Figure 4.6: Original signal with noise

Polynomial

Figure 4.7(a) shows the results from matching the polynomials to the noisy signal. There is a drop in the sum of the residuals when the polynomial increases from second to third order and it is also clear from the figure that the three highest order polynomials give the best fit.



(a) Adapting the five polynomials

(b) Summing up the errors

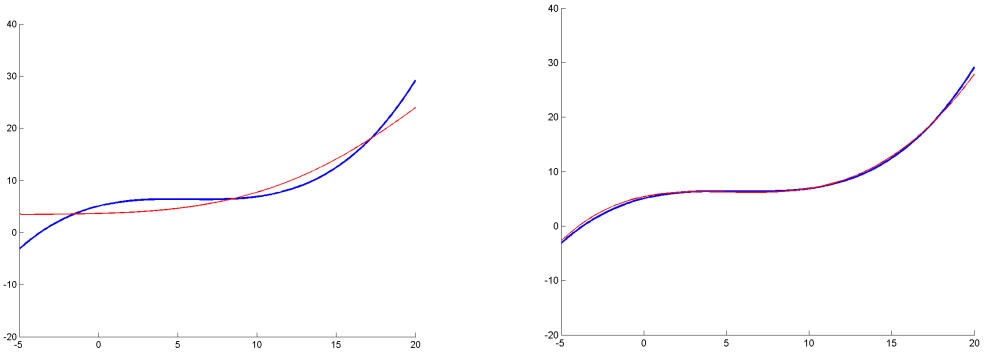
Figure 4.7: Basis function - Polynomials of degree 1 to 5. Noisy signal.

Radial function

With a noisy signal the normalized radial function actually seems to perform better than with a noise free signal. Eight NRBF does not yield a very good fit as can be seen in figure 4.11(a) while with four basis functions or nodes the adaptation is quite smooth.

Table 4.2: Summarized error - noisy signal

P1	*P2*	*P3*	*P4*	*P5*	*NRBF*	*NRBF2*
1242.7	652.7	100.1	70.2	59.2	423.2	70.2



(a) Normalized radial function 1

(b) Normalized radial function 2

Figure 4.8: Basis function - Normalized radial function. Noisy signal.

4.3.3 Missing datapoints

In the next experiment we will test the functions sensitivity towards missing data points and only a third of the initial observations were available to the algorithm. Figure 4.9 shows the areas were data is missing.

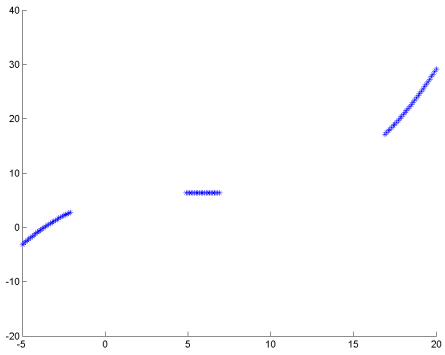
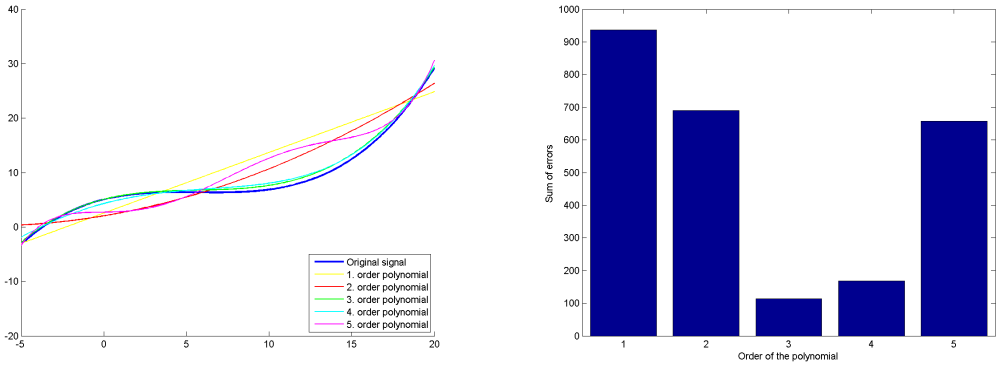


Figure 4.9: Original signal with noise

Polynomial

As in the above example cases a polynomial is implemented first. In figure 4.10(a) the polynomials are fitted to the incomplete function shown in figure 4.9.



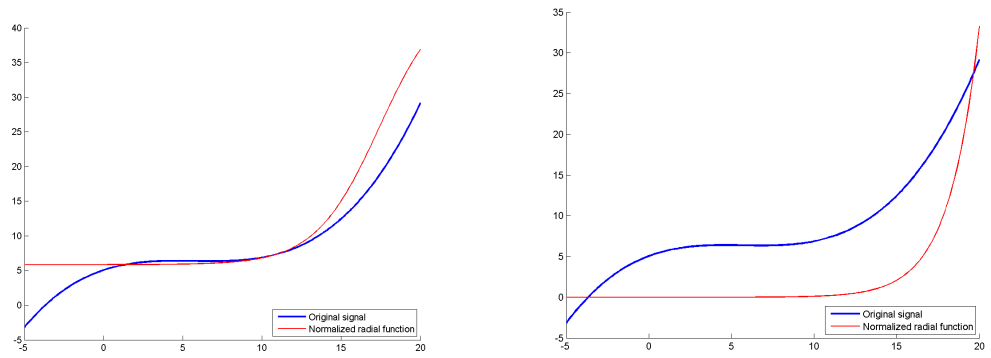
(a) Adapting the five polynomials

(b) Summing up the errors

Figure 4.10: Basis function - Polynomials of degree 1 to 5. Noisy signal.

Normalized radial function

The radial basis functions seem to perform poorly with a restricted number of data available. The error, see table 4.3, could probably have been reduced by changing the γ and β parameters. But we have decided to use the same parameters as above since no technique for selecting the parameters has been employed.



(a) Normalized radial function 1

(b) Normalized radial function 2

Figure 4.11: Basis function - Normalized radial function. Noisy signal.

The sum of the errors for the NRBFs are shown together with the sum of the polynomial errors in table 4.2

Table 4.3: Summarized error - missing data points

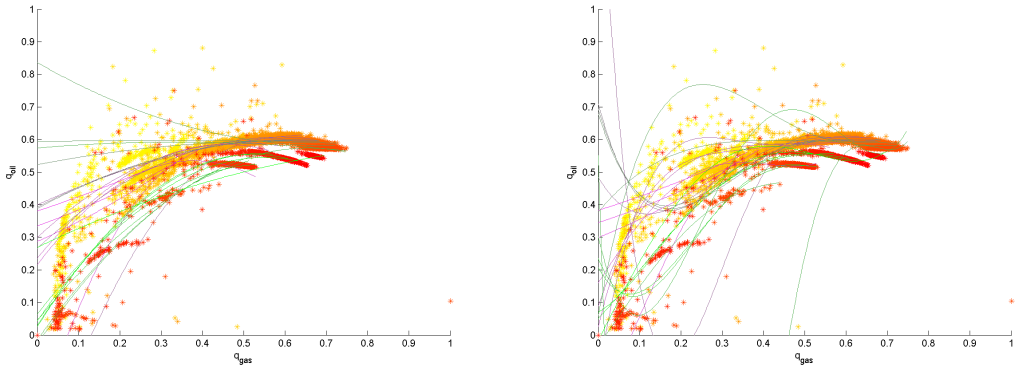
P1	*P2*	*P3*	*P4*	*P5*	*NRBF*	*NRBF2*
936.4	689.7293	113.9	168.2	656.8	600.0	1599.9

4.4 The real case

Only the results from Well #1 in Field A, Well #1 in Field B and Well #1 in Field C will be displayed here. The selection includes the two typical cases of GOR behaviour in addition to one of the wells from Field C. A recursive least squares algorithm is applied to all of the cases, but with different basis functions. Half of every data set, that is every second observation, is used for estimation and the other half is used for validation.

4.4.1 Approach 1: Polynomials

First the linear regression model structure with polynomials of second and fourth order is considered. In figure 4.12 linear regression with a polynomial basis function has been empoyled to Well #1 in Field A. It shows how the 2nd and 4th order polynomials develop with the course of time. The aspect of time is represented by colour code where the initial data points are in the colour yellow and blends to red as time passes. The estimated curves are also colour coded with the purple graphs showing the early adaptation and with time the colour turns to green.

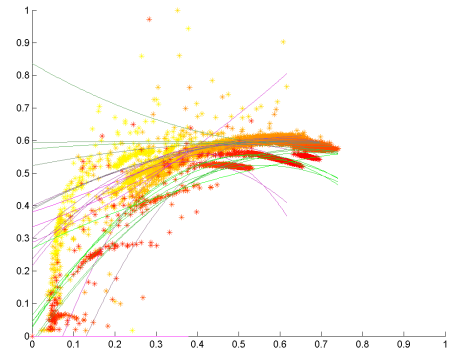


(a) Estimating with second order polynomial (b) Estimating with fourth order polynomial

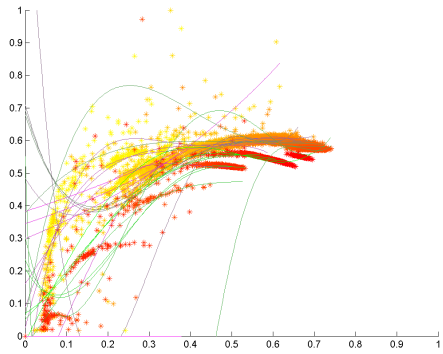
Figure 4.12: Adaptation with polynomials to data from Well #1 in Field A

All of the second order polynomials except for a few are concave downwards. A shape that is concave downwards is the natural curve behaviour for the GOR. When the gas starts to cone towards the inlet of the well more gas will follow due to the higher mobility

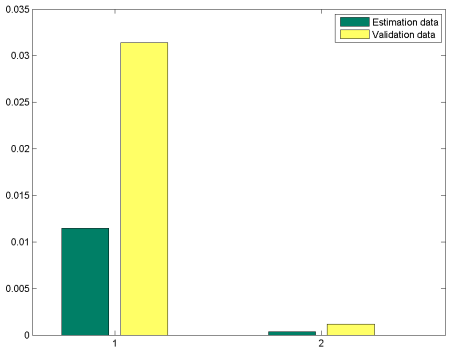
of gas compared to oil and so the oil production will decrease. 2nd order polynomials have a rigid structure and as they are restricted to one inflection point they do not have the freedom of a higher order polynomials. Since the fourth order polynomials are less rigid they should be able to follow the data with more precision than the second order polynomials. Remembering the bias-variance trade off, a higher order still is not always better since the model might overfit to the data. By looking at figure 4.12(b) we see that some of the graphs match the data poorly and this is confirmed by the validation. The estimation yields close to similar mean residual error and standard deviation for the second and fourth order polynomials with the second order polynomial only slightly better and the validation in figure 4.13 shows the same.



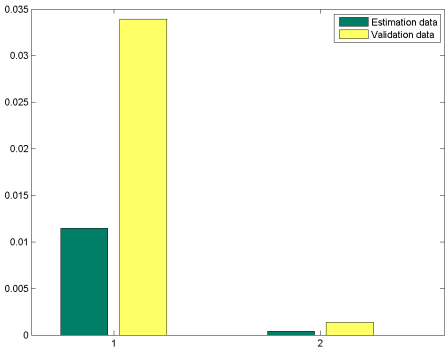
(a) Validating second order polynomial



(b) Validating fourth order polynomial



(c) Second order polynomial: 1.Mean residual error 2.Standard deviation

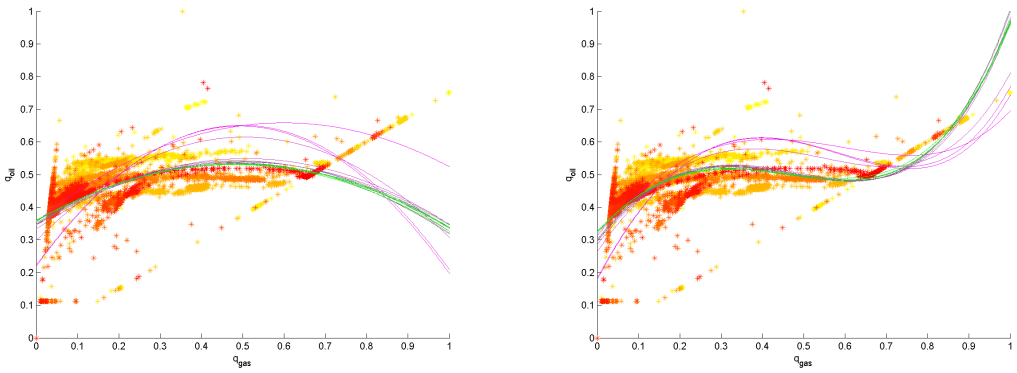


(d) Fourth order polynomial: 1.Mean residual error 2.Standard deviation

Figure 4.13: Validating the estimation of data from Well #1 in Field A

Figure 4.14 shows the estimation of the GOR in Well #1, Field B. With a 2nd order polynomial all of the adapted curves are concave downwards while when using a 4th order polynomial the curves all end with an upswing and hence an erroneous increasing oil flow

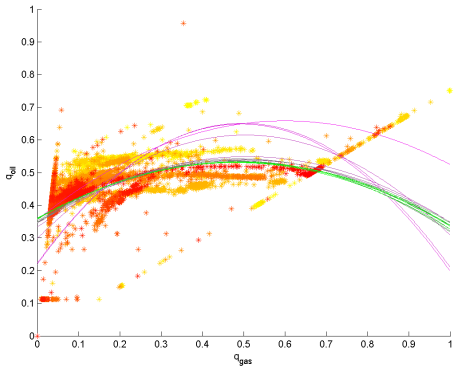
rate. But when validating the estimated parameters we see that the 4th order polynomial yields a smaller standard deviation and mean residual error. Although the difference is minimal it is still present.



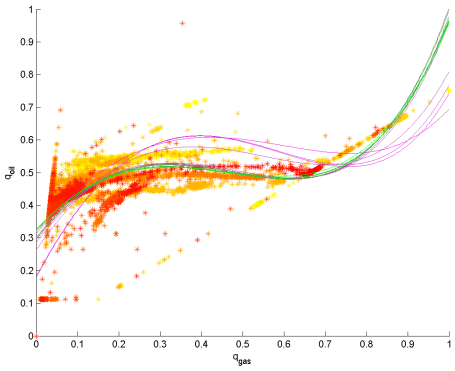
(a) Approach 1: Estimating with second order polynomial

(b) Estimating with fourth order polynomial

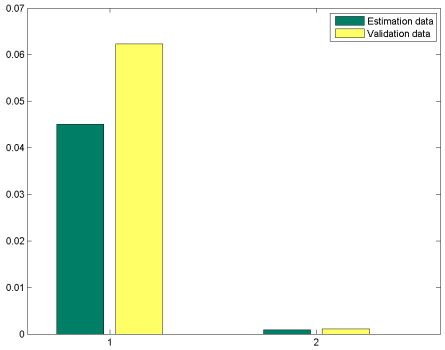
Figure 4.14: Adaptation with polynomials to data from Well #1 in Field B



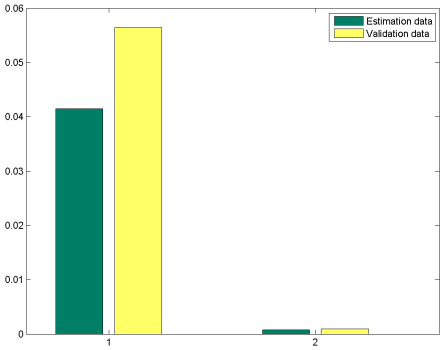
(a) Validating second order polynomial



(b) Validating fourth order polynomial



(c) Second order polynomial: 1.Mean residual error 2.Standard deviation

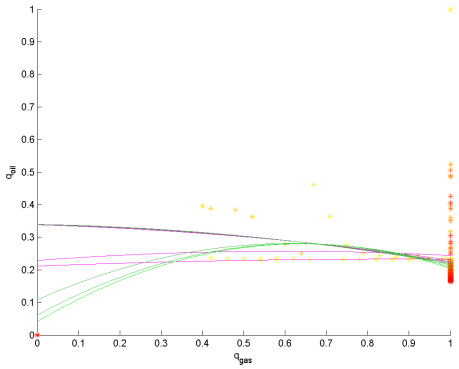


(d) Fourth order polynomial: 1.Mean residual error 2.Standard deviation

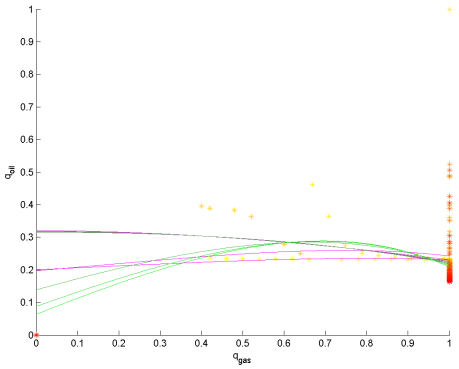
Figure 4.15: Validating the estimation of data from Well #1 in Field B

From Field C Well #1 is presented and the linear regression using polynomials can be viewed in figure 4.16. Previously it was stated that the wells in Field C were not proper for testing the algorithm developed in the thesis. Still an example has been included.

Both the 2nd and 4th order polynomials have curves which are concave downwards. The GOR reach the maximum gas flow q_{gas} constraint $q_{gas} = 1$ each year and after three years the q_{gas} takes on a constant value at its constraint over the remaining seven years. In figure 4.16 we see that the oil flow q_{oil} has a decreasing rate, hence the vertical set of points at $q_{gas} = 1$. Since the goal for the Field C wells is to reach a certain gas constraint and stay there a polynomial adaptation will not cover its behaviour in a correct manner.
By validating the two polynomial functions we see that the standard deviation is the same for both 2nd and 4th order polynomial. The 4th order polynomial gives a slightly larger residual error.

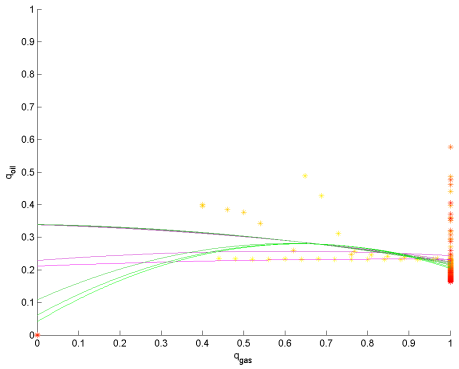


(a) Approach 1: Estimating with second order polynomial

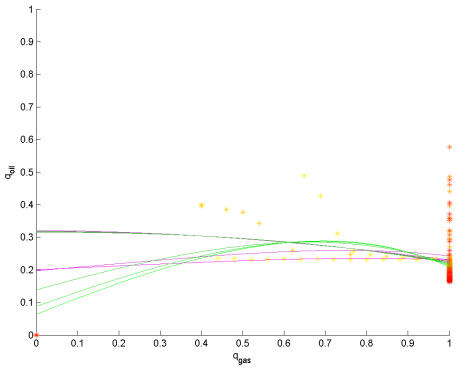


(b) Estimating with fourth order polynomial

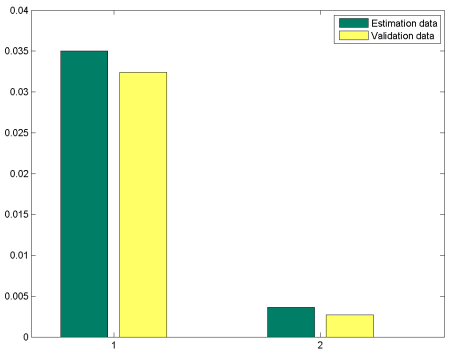
Figure 4.16: Adaptation with polynomials to data from Well #1 in Field C



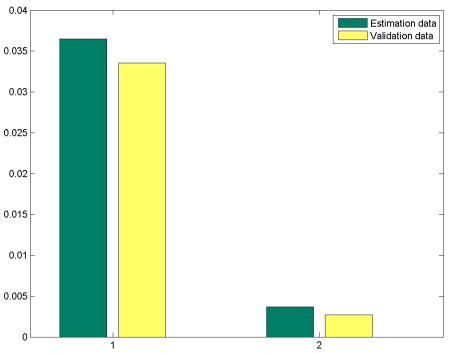
(a) Validating second order polynomial



(b) Validating fourth order polynomial



(c) Second order polynomial: 1.Mean error 2.Standard deviation



(d) Fourth order polynomial: 1.Mean error 2.Standard deviation

Figure 4.17: Validating the estimation of data from Well #1 in Field #3

A comparison of the standard deviation from validating the polynomials is presented in table 4.4. The table shows that 2nd and 4th order polynomials have approximately the same standard deviation. From the visual inspection of the plots presented earlier in this section we see that the 2nd order polynomial give the best fit when including the natural behaviour of the GOR.

Table 4.4: Standard deviation - polynomials

	Well #1, Field A	*Well #1, Field B*	*Well #1, Field C*
2nd order polynomial	0.0012	0.0011	0.0027
4th order polynomial	0.0014	0.0009	0.0027

4.4.2 Approach 2: Normalized radial function

The implemented NRBF has three regressors. Increasing the number of regressors has been tested and gives an increase in residuals and standard deviation. A modification is introduced to the θ parameters making them linear functions for smoother curves. The centres of the nodes are placed within the spread of the gas flow q_{gas}. The β values deciding the steepness of the NRBF are found by testing different values and choosing the one which yields the minimal error.

Consider the adaptation with NRBF to the GOR in Well #1, Field A, shown in figure 4.18. The latest adaptation gives the most smooth curve fitting and it is somehow not very smooth until then. This is not what was expected from the flexible NRBFs which usually give nice fits.

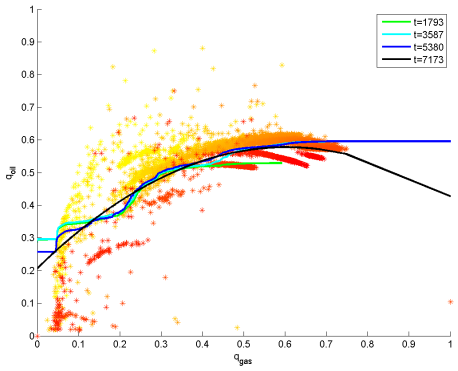
Well #1 from Field B achieves a poorer adaptation using NRBF than Well #1 from the first field. This might have been adjusted by selecting different values for the two additional parameters γ and β. With this in mind the adaptation with the normalized radial functions is still inadequate.

By visual inspection the estimation with NRBF to Well #1 in Field C is not very good. This is not surprising as the observations reach their constraint in q_{gas} after only three years. Even in the first three years the gas constraint is reached eventually.

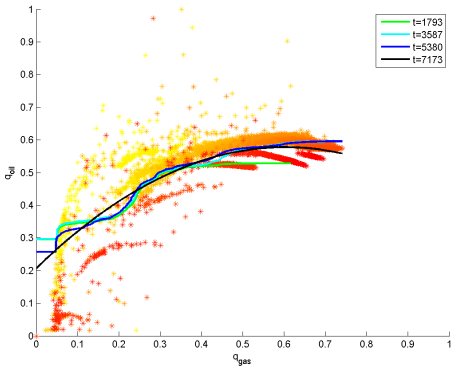
Table 4.5: Standard deviation - normalized radial basis functions

	Well #1, Field A	*Well #1, Field B*	*Well #1, Field C*
NRBF	0.00076	0.00084	0.0044

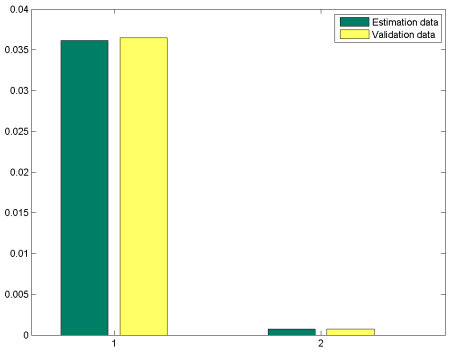
The standard deviation from all of basis functions are gathered in table 4.6.



(a) Estimating with NRBF



(b) Validating NRBF

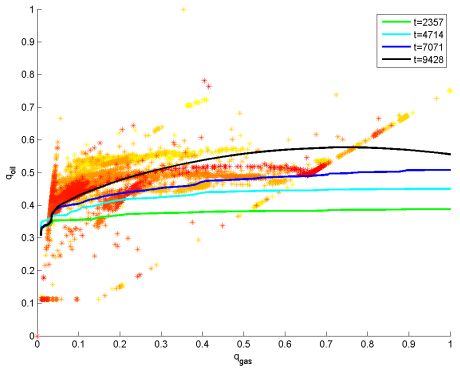


(c) 1.Mean error 2.Standard deviation

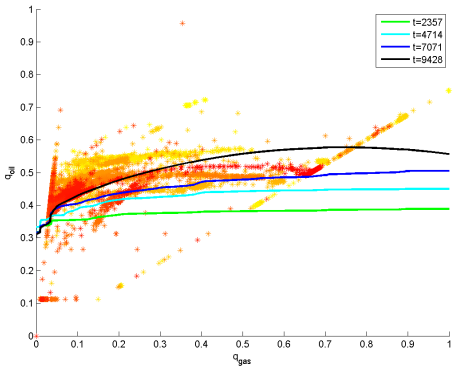
Figure 4.18: Linear regression with NRBF to data from Well #1 in Field #1

Table 4.6: Standard deviation - comparing the choice in basis functions

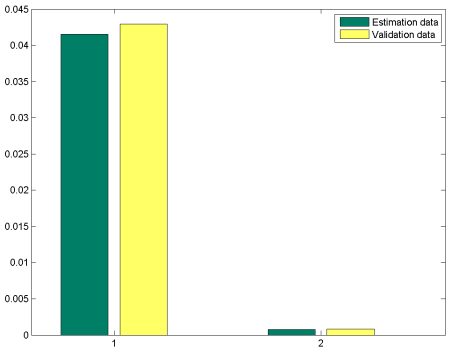
	Well #1, Field A	*Well #1, Field B*	*Well #1, Field C*
2nd order polynomial	0.0012	0.0011	0.0027
4th order polynomial	0.0014	0.0009	0.0027
NRBF	0.00076	0.00084	0.0044



(a) Estimating with NRBF



(b) Validating NRBF

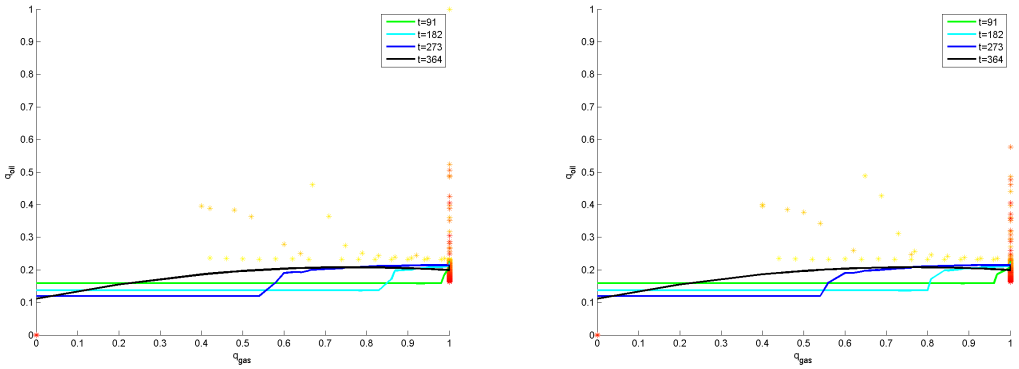


(c) 1.Mean error 2.Standard deviation

Figure 4.19: Linear regression with NRBF to data from Well #1 in Field #2

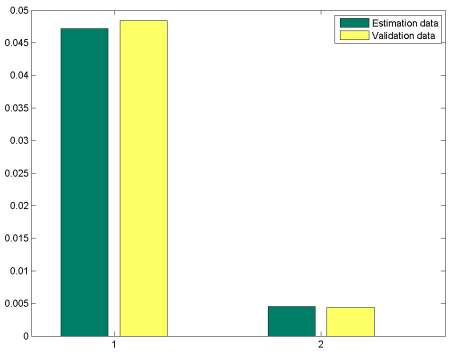
4.4.3 The algorithm

There is no numerical value measuring the efficiency of the recursive least squares algorithm. Still it is a at all times converging algorithm, robust and efficient. The RLS is fast and proper for the objective of this thesis.



(a) Estimating with NRBF

(b) Validating NRBF



(c) 1.Mean error 2.Standard deviation

Figure 4.20: Linear regression with NRBF to data from Well #1 in Field #3

Chapter 5

Discussion

By defining a function and developing an online algorithm that predicts the GOR, the already existing tool FlowManager will hopefully become more efficient and precise in its calculations. An attempt of creating such a procedure has been made in this thesis. Two different models and model structures have been implemented together with a recursive least squares algorithm.

As mentioned in the introduction the oil and gas flows used in the algorithm are not measured directly and only represent the best possible estimates of the flows. Hence the GOR estimate will only become as good an estimate as the calculated flow allows.

5.1 The findings

The findings from chapter 4.4 shows that it is possible to get proper online estimations of the GOR. When comparing the standard deviations from validating the estimations, shown in table 4.6, we see that the normalized radial functions give the smallest standard deviation for the wells in Field A and B.

By visual inspection, examining the figures in chapter 4.4, the overall performance of the 2nd order polynomial seems most proper. As close as all of the curves are concave downwards and they seem to follow the data quite nice. The 4th order polynomials are less predictable. For the first well in Field A the 4th order fit might be good for some of the curves but then they oscillate too much for others. For the well in Field C all of the curves end with an upswing when employing a 4th order polynomial. If we consider the physics of the GOR, such behaviour is not realistic. Although the standard deviation is small it is important to take the physical principles into consideration as well.

It can be argued that with the diminished standard deviation achieved when employ-

ing the NRBF the choice is in choosing the most proper basis function easy. But the curves from the NRBF nets are bumpy until they smooth out with the final graph. For the well in Field B the NRBF net gives the impression of being a slow learner. Although the two last graphs seem to yield nice fits to the data, the first curve is close to linear and do not represent the data properly. Hence, even though linear regression using 2nd order polynomials which are more rigid and generate a larger standard deviation than that of the NRBF nets, the 2nd order polynomial seem to be the best choice in basis function.

Although no other algorithm than the recursive least squares algorithm has been employed, it seems to be working just fine and there is no reason to exchange it with another recursive procedure. The algorithm has not yet been coupled with a production optimization tool for testing and evaluation. Not having been able to compare the estimates of the existing approaches for estimating the GOR one cannot know for sure how much better this approach is over the others.

5.2 Alternative approaches

5.2.1 Polynomial Basis Functions

It can be argued that the estimated GOR curves should pass through the origin. There is a straight line passing through the origin which the GOR never passes. This line is the $1/R_S$ line. The R_S is a slowly varying parameter which changes with the well conditions. So if R_S is a known parameter, the basis function can be altered from a single polynomial into B splines. As stated in chapter 3 B splines are piecewise polynomials where the connections have continuous derivatives. This approach may suit the wells #4 and #5 in Field A (figures 4.1(d) and 4.1(e)).

When the basis function is a polynomial the data is "windowed ". This approach yields smaller prediction error and standard deviation than minimizing over each of the data points. It also makes it easier to employ a possible method for "starting over "when for instance the choke shuts in. More on this method below. There are several ways of choosing the size of the window. A GOR update is not needed more often than once a week, but the window size is reduced to four days in the employed algorithm. During this period of time the GOR can be thought of as stationary. Since the rate dependent GOR vary with the choke position, if the choke is shut in the window size is reduced for that particular batch, a curve is adapted and a new window is started. It can be argued that the window should have a size of one week of observations since the update is needed weekly. A smaller window gave better results. Hence why not make the window size even smaller? A smaller window size gave faulty curves when estimating so this option was left behind.

A modification of this procedure could be to reset the parameters θ and forget all previous data when the well shuts in and start from scratch. A less extreme approach would

be to just make θ smaller to forget a larger quantity of data if the well shuts in. That way the algorithm will not be completely blank when the well starts up again. This last procedure was tested on one of the wells in Field A as those wells shut in from time to time. As opposed to the regular value of $\lambda = 0.993$ the value was reduced to $\lambda = 0.9$ when the wells shut in only remembering measurements from the last 10 windows when starting back up. There was no difference in the standard deviation, but with a more thorough research of this approach it may improve the algorithm.

If the well experiences the phenomena of coning gas, shutting in the well will most likely lead to the cone slowly withdrawing from the inlet of the well. Liquid fluid of oil and water will dominate the initial flow when the well re-opens and the shape of the GOR is not similar to before the shut in. Even so the approach with not forgetting everything the algorithm has learned up until a shut in seems like the most sensible solution as the conditions in the well wont be completely altered.

5.2.2 Normalized Radial Basis Functions

Although using normalized radial functions as basis functions mean a less rigid structure to the estimated curve, it also introduces two additional parameters namely the localization parameter γ and the scale parameter β. Since we know the spread of the data in advance, the centres γ are chosen within the extension of the q_{gas} and the scale parameters are found experimentally by tuning. An algorithm estimating the centres γ and the width of the bell shaped NRBFs β could be employed. Several algorithms exist, among others clustering algorithms. These algorithms are online and apart from the unsupervised first stage they are supervised algorithms [Bors, 2001]. Employing a localization technique for the additional parameters might result in increased performance for the NRBF net, but it would also demand a change in model model and algorithm since linear regression no longer holds.

5.2.3 Prefiltering the data

In theory all data can be important and should be considered when estimating the GOR. In reality not all data should be considered, but there is a fine line separating the observations that should be filtered from the rest of observations. When a well starts up again after a shut in it is typically followed by some spiky and erroneous data points. A way to remove such observations is to just remove data points from the first hour after a start up. Another way is to evaluate the points separately and decide on some pre set ground if the observation is to be used in estimation or not. The reservoir conditions are heterogeneous over time and finding the fine line with proper restrictions on which data should be filtered

and not is not straight forward.

5.2.4 Dividing the data into areas of application

Different application areas for the RLS

The GOR tend to have an uneven spread of data. A proposal is to divide the x axis into parts where the different residuals are weighted by the number of data inside that particular section.

Sigmoid function

When time yields an uneven spread of data, localization function like sigmoid functions may be employed. For instance lets say that there are three clusters of data. The first cluster includes data from time $t_0 < t < t_1$, the next data within the time span $t_1 < t < t_2$ and the last cluster of data arrives at $t_2 < t < t_3$. Due to the forgetting factor λ in the RLS the first points will get minimal weight or they might even be ignored as time passes. By introducing sigmoid functions this problem can be avoided.

$$\sigma_1 = \frac{1}{1 - e^{x - x_0}} \tag{5.1}$$

$$\sigma_2 = \frac{1}{1 - e^{-x - x_0}} \tag{5.2}$$

σ_1 holds for $x(t_0)$ to $x(t_1)$ and σ_2 holds for the rest of the observations. When applying the RLS we can choose that the data affected by σ_1 wont be forgotten, but the rest of the data affected by σ_2 will be treated as usual with the forgetting factor λ.

The downside to this procedure is that the conditions in the reservoir will change and the old data will not be prevailing any more. The technique has been considered redundant to implement on the production data used in this thesis however. Due to alternations in the choke position and some infrequent shut ins a sufficient amount of data points is available for all the prevailing areas of the x axis.

Weighted relative least squares

With the weighted RLS the residuals are as presented in equation 3.19 as
$z_1 = y(t) - \psi^T(t)\widehat{\theta}(t-1)$. If the standard deviations of the measured responses are proportional to the measured value but with different proportional constant, the least squares residual should be altered

$$z_2 = \frac{y(t) - \psi^T(t)\widehat{\theta}(t-1)}{y(t)} \tag{5.3}$$

Alternatively if the variance of the measured responses are proportional to the measured value also here with different proportional constant the residual should become

$$z_3 = \frac{y(t) - \psi^T(t)\widehat{\theta}(t-1)}{\sqrt{y(t)}} \tag{5.4}$$

5.2.5 Changes in choke position

The algorithm does not take into account changes in choke position other than shut ins. Take Well #1 in the first field as an example. In figure 4.1(a) there are six red areas where the data gather in the shape of a line. Due to changes in choke position the observations move from one cluster to another. There is a claim for updates in the GOR once every fourth days and since the choke position may change several times in one day, it does not seem wise to adjust one function for instance a polynomial to each of the clusters. Instead a polynomial is made to fit a more general tendency.

Chapter 6

Conclusion

In this thesis the author has developed an algorithm for estimating and updating the GOR for use in production optimization tools like FlowManager. The application has been tested on production data from three wells.

Two types of basis functions namely polynomials and normalized radial basis functions were adapted to the GOR by a recursive least squares algorithm. From validating the model the normalized radial functions were the most suitable for adaptation and it seems to have good potential, but from visual inspection the a second order polynomial had the best match. Although the fourth order polynomial seems to have a good fit to some of the production data, when taking the physics of a GOR into consideration it is no longer adequate. The recursive least squares algorithm has proven to be a fast convergent algorithm suitable for the purpose of the thesis.

Chapter 7

Future work

Improving the algorithm

Neither of the estimations that have been performed yield optimal responses. The plots are not always a proper fit and this goes for both of the choices in basis functions. This may be corrected by adding constraints to the algorithm. Taking into consideration changes in choke position and water cut should be employed. When the well shuts in to let a possible gas cone back, it should be taken into account how many of the previous observations should be considered. When separating the data into windows, the size of the window could be altered. If new constraints and requirements are added to the algorithm, testing the algorithm without dividing the observations into windows could be an option.

Merging with production optimization tools

The algorithm will need adjustments before working together with a production optimization tool. The author has developed the algorithm without considering the programming language used by the production optimization software and it might need translation to another language before it can communicate and co-exist with other software.

Normalized radial basis functions

The normalized radial basis functions have been successful in other applications and their potential should be explored some more. They may yield a better fit than presented in this thesis with the proper choice in localization parameters γ and scale parameters β. In addition the possibility of employing B splines should be taken into consideration.

Evaluting the algorithm

The performance of the algorithm should be compared to the performance of the prevailing estimation techniques.

Launching the algorithm

When adjustments have been made the algorithm should be tested on fresh observations and finally it should be implemented togehter with a production optimization software in a real time environment optimizing the oil recovery from real reservoirs.

Bibliography

Robert Aasheim, Kari Brusdal, Are Mjaavatten, and Jan Hauge. Production optmization on troll with a rate dependent gor model and gap. In *SPE Offshore Bergen*, 2004.

AI ACCESS. Glossary of data modeling. URL `http://www.aiaccess.net/e_gm.htm`.

Thorsten Bertelsen. Sokkelen krever innsats. *Norsk sokkel*, 4:12–13, 2007.

Adrian G. Bors. Introduction of the radial bsis function (rbf) networks. *Online Symposium for Electronics Engineers*, 1:1–7, 2001.

Guido Bugmann. Normalized radial basis function networks. *Neurocomputing*, 20:97–110, 1998.

Rolf Henriksen. *Stokastiske systemer - Analyse, estimering og regulering*. Institutt for teknisk kybernetikk, 1998.

Gerald Kaiser. *A Friendly Guide to Wavelets*. Birkhaüser, 1994.

Lennart Ljung. *System Identification - Theory for the user*. Prentice Hall PTR, 1987.

L.A. Saputelli, S. Mochizuki, L. Hutchins, R. Cramer, M.B. Anderson, J.B. Mueller, A. Escorcia, A.L. Harms, C.D. Sisk, S. Pennebaker, J.T. Han, A. Brown, C.S. Kabir, R.D. Reese, G.J. Nunez, K.M. Landgren, C.J. McKie, and C. Airlie. Promoting real-time optimization of hydrocarbon producing systems. In *SPE Offshore Europe Aberdeen*, September 2003.

Schlumberger. The schlumberger oilfield glossary, 2003. URL `http://www.glossary.oilfield.slb.com/`.

Jonas Sjöberg, Lennart Ljung, and Håkan Hjalmarsson. Neural networks in system identification. In *IFAC SYSID94*, volume 2, pages 49–72, 1994.

Jonas Sjöberg, Quinghua Zhang, Lennart Ljung, Albert Beneviste, Bernard Delyon, Pierre-Yves Glorennec, Håkan Hjalmarson, and Anatoli Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724, 1995.

SPGMediaPLC. Offshore technology, 2008. URL `http://www.offshore-technology.com`.

Anatoli B. Zolotukhin and Jann-Rune Ursin. *Introduction to Petroleum Reservoir Engineering*. Høyskoleforlaget, 2000.

Appendix A

CD

The appendix is in the form of a CD containing the Master's thesis in PDF format and the m-files (MATLAB).