NTNU
Norwegian University of
Science and Technology

Estimation of Bottomhole Pressure for Managed Pressure Drilling

Comparison of Nonlinear Estimators

Pål Skønberg Løvik

Master of Science in Engineering Cybernetics

Submission date:	June 2011
Supervisor:	Tor Arne Johansen, ITK
Co-supervisor:	Lars Imsland, ITK
	Glenn-Ole Kaasa, Statoil ASA
	Alexey Pavlov, Statoil ASA

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Problem Description

It is crucial for high-precision managed pressure drilling (MPD) operations that the pressure at critical locations is estimated accurately. To determine the pressure profile of the well, certain parameters should be estimated as well. In particular, choke flow characteristics and annular friction and density stand out as good candidates for estimation as they are encumbered with uncertainties, difficult to tune offline, and slowly varying.

Objective

The objective of the thesis is to continue the work on observers for estimation of the bottom hole pressure during drilling. The sub tasks are to:

1. Compare the moving horizon observer with an unscented Kalman filter.
2. Use information on process and measurement noise to further improve the moving horizon observer by utilizing covariance information from the unscented Kalman filter as weighting.
3. Develop a non-linear friction model for the annular friction using basis functions, and implement this into the moving horizon observer and compare with a simpler friction model.

As far as the laws of mathematics refer to reality, they are not certain, and as far as they are certain, they do not refer to reality.

- Albert Einstein

Preface

The assignment for this master's thesis was given in the spring of 2011, in cooperation with Statoil ASA as part of a M.Sc. program in engineering cybernetics, by the Department of Engineering Cybernetics at NTNU. The goal was to continue investigation of estimation methods for estimation of the bottom hole pressure during a regular drilling scenario. It has demanded a lot of hours, both at and away from the office, but the work has truly increased my interest in the world of estimation, automation and drilling.

Several people have helped me and guided me towards an end result: Tor Arne Johansen has been my supervisor for the last year, both in my project assignment last fall and in this master's thesis, and he has been more accommodating, encouraging and easy to deal with than I could have ever asked. Through swift and precise responses in meetings and electronic correspondence, he has facilitated my work considerably. A special thanks to Marcel Paasche who has been extremely helpful with my understanding of the system and its implementation in MATLAB. Also, Øyvind Nistad Stamnes, Tomas Poloni and co-supervisor Lars Imsland have been involved using their expertise with this particular system to provide me with valuable advice. Glenn-Ole Kaasa and Alexey Pavlov have been my external supervisors from Statoil and they have been available for assistance any time it was needed. My gratitude to you all.

Further, I would like to take this opportunity to thank my friends and family. I am truly blessed with many friends from all parts of Norway, who every day make life even more enjoyable. My utmost appreciation to my parents who always encourage me to strive to be the best that I can be, and who never stop believing in me. To my dear brother, who has cleared the way for me and inspired me with his M.Sc. in Information and Communication Technology; You will always be my best friend. The rest of my family have also been sensational in their support and interest in what I do.

Last, I would like to posthumously thank the great person and fantastic personality that was my uncle, Franz Johnny Skønberg, who always inspired me to pursue higher education. Your intelligence, broad knowledge and loving presence will always be missed.

Pål Skønberg Løvik
Trondheim, June 2011

Abstract

To avoid hole stability problems in an increasingly fierce drilling environment, the demand for accurate control of the pressure profile during drilling operations is rising. As standard instrumentation of drilling rigs have poor measurement of the bottom hole pressure, there is a need for estimation. However, a precise model of a drilling process is difficult to obtain, so a competent observer, using a simpler, lower order model, should be satisfactory.

In this master thesis several approaches on estimation are discussed together with a suggested improvement in the annular friction model. The estimators tested are: First, the moving horizon observer, which is presented together with prior work by the author and Marcel Paasche [19]; Second, the unscented Kalman filter, which is a new estimation candidate introduced together with regularization to compensate for the slow update and lack of availability in bottom hole pressure measurements. Last, different combinations of the two observers are proposed.

All observers are tested in simulations and good performance is found for both the MHE and UKF. Parameter adaptation is found to be effective for both observers, but the UKF encounters some minor observability issues when the system is not sufficiently exciting. Different combinations of the two observers increase computational complexity, unfortunately without achieving better accuracy in estimates. The estimates are deteriorated when the alternative friction model is tested, and it is thus considered a failed attempt to improve the simple third order Kaasa model.

Contents

List of Figures

Abbreviations

BFGS	Broyden-Fletcher-Goldfarb-Shanno
BHP	Bottom Hole Pressure
IADC	International Association of Drilling Contractors
NMHE	Nonlinear Moving Horizon State Estimator
MAEE	Mean Absolute Estimation Error
MATLAB	Matrix Laboratory
MHE	Moving Horizon State Estimator
MPC	Model Predictive Control
MPD	Managed Pressure Drilling
NMPC	Nonlinear Model Predictive Control
PA	Parameter Adaptation
PDF	Power Density Function
PE	Persistence of Excitation
RNMHE	Regularized Moving Horizon State Estimator
RUKF	Regularized Unscented Kalman Filter
SVD	Singular Value Decomposition
UKF	Unscented Kalman Filter

Chapter 1

Background

1.1 Motivation and Introduction to Drilling

Although there has been a minor decrease in worldwide oil consumption over the last two years, the growth in Asia Pacific, Africa and Middle East continues (Figure 1.1) and is inevitable for years to come. In particular, their strive for a better standard of living increases energy demand. Renewable energies have become a primary focus area, but hydrocarbons still account for a large part of the total primary energy supply [6].

As of today, the already produced reservoirs still hold large amounts of crude oil and gas that potentially can be extracted and new reservoirs with complex formations are yet to be discovered. Thus, the importance of exploiting greater percentages of these complex and difficult reservoirs is compelling and precise drilling methods are essential in the progress.

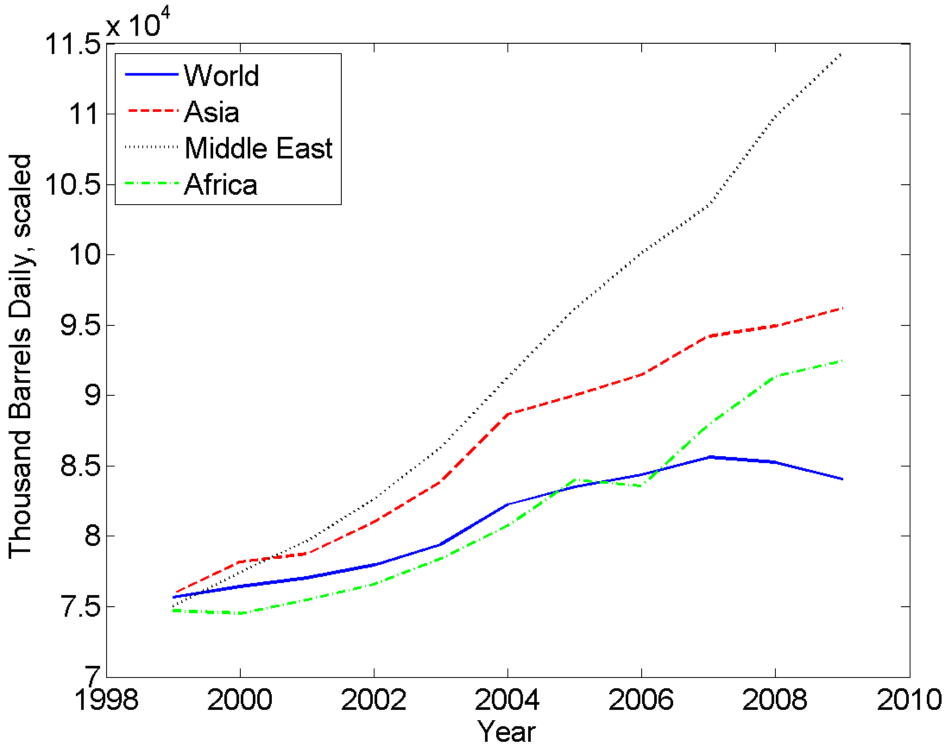

Figure 1.1: World Oil Consumption. The area-specific plots are scaled to more easily grasp the development in consumption

To better understand this work, consider the simplified drill rig illustrated in Figure 1.2. The drill bit cuts the rocks, either by using compressive failure or by shearing off slices, while drill collars provide weight on the bit to help ease the task. Collars are part of the entire drill string, coupled with a motor attached to the top drive of the derrick, which makes the bit turn at the bottom of the well bore.

During drilling, the length of the drill string is gradually increased by adding stands of pipe (approximately every 27 meters), referred to as making a pipe connection. Sections of steel pipe, *casings*, are placed in the bore hole and cement is often filled in the open space outside the casings to help maintain robustness of the well. By adding sets of subsequently smaller hole sizes drilled inside each other, a smaller bit can be used further into the well, where the potentially more uncertain and unstable formations can be found.

There is a constant need to remove generated cuttings and maintain pressure down hole. Drilling fluid, known as "mud", circulates through the bit and carries the cuttings up through the annulus where it exits through a choke. After exiting, the fluid returns to the mud tanks, where the cycle starts over. The “Rotating control device” in Figure 1.2 seals off the annulus from the outside while mud-flow out of the annulus is controlled by a choke. This allows for better pressure control, but it is not standard in conventional drilling techniques.

Controlling the pressure is important to prevent uncontrolled reservoir influx and, among other issues, prevent bore holes from collapsing, minimize loss of mud when drilling into depleted sections of reservoirs, reduce danger when drilling into high pressure zones and avoid skin damage which can lower production later on. Kicks (Oil, natural gas and/or water flowing up the annulus) may occur and eventually turn into blowouts (uncontrolled release of crude oil/natural gas from an oil/gas well), which leads to large financial losses, devastating environmental disasters and possible loss of human and animal lives.

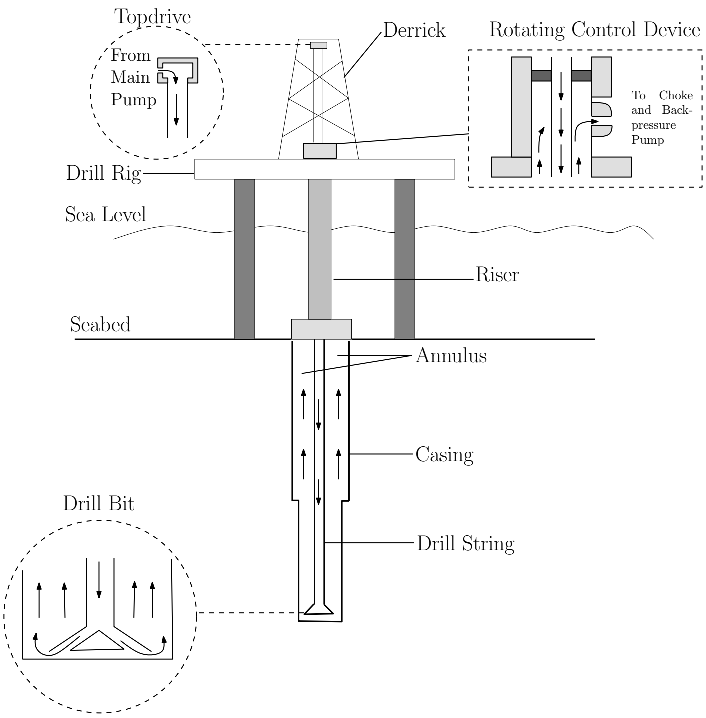

Figure 1.2: Example of Drilling System

1.2 Estimation of Well Pressure

As described in Section 1.1, accurate control of annular pressure during drilling is of great importance. Managed pressure drilling (MPD) has emerged as a result of this demand, and IADC Underbalanced Operations Committee

& Managed Pressure Drilling Committee have defined MPD as *"an adaptive drilling process used to more precisely control the annular pressure profile throughout the well bore."* Further, they define the objectives to be *"to ascertain the down hole pressure environment limits and to manage the annular hydraulic pressure profile accordingly"*.

The annular pressure profile is difficult to obtain, and the bottom hole pressure (BHP) is therefore often the variable chosen for control. Since the measurements are gathered normally by using mud-pulse telemetry, they are not only characterized by slow sampling, but also by their absence at certain points of the drilling procedure, e.g. during pipe connections. Pressure drop due to friction and movement of the drill string, and reservoir influx are factors that affect the certainty of the BHP, and the pressure should therefore be estimated.

OLGA [5] is a powerful, market-leading multiphase simulator for engineering the flow of oil, water and gas in wells, pipelines and receiving facilities. However, simulations often have to be combined with tuning of certain parameters which can be done via automatic methods such as parameter estimation or via a well trained, experienced operator. The former is desirable, as it in the long term will have reduced costs in comparison to the latter. Therefore, several attempts to estimate and control the BHP have been carried out, often using low order models. [4] uses an unscented Kalman filter (UKF) to estimate friction parameters and choke coefficients for an MPD application. Down hole pressure predictions are shown to be fairly accurate, despite the estimates showing unwanted behavior. [16] uses the UKF together with nonlinear model predictive control (NMPC) to keep the well pressure within the pressure restrictions of a reservoir formation. Successful control of the downhole pressure is provided, but again unwanted behavior, i.e oscillations are found in state and parameter estimates. [20], [24] and [25] uses the moving horizon observer, an adaptive observer and the extended Kalman filter respectively to estimate the bottom hole pressure during drilling. [17] demonstrates automatic coordinated control of pump rates and choke valve during surge and swab (increase and decrease of pressure in well respectively) operations, and results are very promising compared to both manual control and the case of only automated choke line pump control.

The wired drill pipe technology [27] has been of major interest in combination with MPD, and makes use of electrical wires, built into every component of the drill string, so that electrical signals can be carried to and from the surface. The data transmission rate is much greater than that of telemetry and mud-pulse. National Oilwell Varco delivers the product "Intelliserv" which is *"the only high-speed, high-volume, high-definition, bi-directional broadband data transmission system that enables downhole conditions to be measured,* *evaluated, monitored and actuated in real time.”* Technology similar to the wired drill pipe can be important in MPD. However, difficult drilling conditions motivate further development of accurate estimation methods that also help to keep costs at a minimum.

1.3 Scope and Emphasis

This thesis simulates and continues to develop good estimation methods for a standard drilling scheme. The unscented Kalman filter is proposed as an alternative to the moving horizon observer, previously presented in the author’s project assignment (See Section 1.4). Main goals are to:

1. Test performance of the unscented Kalman filter on a standard drilling scheme.
2. Compare performance of the unscented Kalman filter with the moving horizon observer.
3. Model annulus friction using basis functions.
4. Use information from the unscented Kalman filter to further improve performance of the moving horizon observer.

1.4 Previous Work in Project Assignment

Prior to this thesis, the author has written a project assignment on the moving horizon observer. The assignment focused on the regularized nonlinear moving horizon observer presented in [2] and work done by Marcel Paasche [20]. Further, it discussed the implications of process noise modeling i.e acknowledging the presence of model errors. The observer theory on the moving horizon observer presented in the project assignment is re-used in this thesis and results obtained are presented to compare performance to the unscented Kalman filter.

1.5 Report Outline

The project is divided into four main sections:

1. In Section 2, the well known “Kaasa Model” is presented and described. Also, minor changes that affect this thesis are made, including the updated model of annulus friction;

2. In Section 3, the regularized nonlinear moving horizon observer and unscented Kalman filter are presented along with changes that may positively affect the accuracy of the estimation. Also, some thoughts around different aspects of implementation are given;

3. In Section 4, simulations and results are presented to demonstrate performance of the estimators;

4. In Section 5, conclusions are drawn and recommendations for future work are given.

Chapter 2

Modeling

2.1 Model Summary

The model used in this project is based on the Kaasa model presented in [23], originally developed in [11], with minor changes. It is a fairly simple third order model that shows both pressure and flow dynamics for the system in Figure 2.1.

The Kaasa model defines the system with the following three first-order differential equations and outputs:

$$\dot{p}_c = \frac{\beta_a}{V_a}(q_{bit} - q_{choke} + q_{back} + q_{res} - \dot{V}_a) \tag{2.1}$$

$$\dot{p}_p = \frac{\beta_d}{V_d}(q_p - q_{bit}) \tag{2.2}$$

$$\begin{aligned} \dot{q}_{bit} = {} & \frac{1}{M}(p_p - p_c - \theta_1 q_{bit} - \theta_2 \left|q_{bit} - q_{res}\right| (q_{bit} + q_{res}) \\ & +(\rho_d - \rho_a) g h_{bit}) \end{aligned} \tag{2.3}$$

$$y_1 = p_c \tag{2.4}$$

$$y_2 = p_p \tag{2.5}$$

$$y_3 = p_c + \theta_1 q_{bit} + \rho_a g h_{bit} \tag{2.6}$$

where p_c is the choke pressure, p_p is pump pressure, q_{bit} is the flow rate out of the drilling bit, q_{choke} and q_p are the choke (Section 2.1.1) and pump flow rates respectively, q_{back} is the back-pressure pump flow rate and q_{res} is reservoir influx or out flux. h_{bit} is the bit's depth and g is gravity. β_a, β_d , V_a, V_d and ρ_a,ρ_d are the bulk modulus', volumes and average densities for the annulus and drill string, respectively and $\dot{V}_a$ is the change in volume of the annulus. The transition to resulting differential change in q_{bit} from net pressure difference is done by $M = M_a + M_d$,where $M_i = \bar{\rho}_i \int_0^{l_i} \frac{1}{A_i(x)} dx$, integrating the inverse of the area over the entire length of the annulus/drill string. Last, θ_1 and θ_2 are friction parameters for the annulus and drill string respectively. It is notable that the flows in the annulus and drill string originally were modeled as turbulent, whereas later the annulus flow, as in this project, was modeled laminar. This is seen in 2.3 where θ_1 is multiplied by $q_{bit} - q_{res}$ (laminar), while θ_2 is multiplied with $|q_{bit} - q_{res}|(q_{bit} - q_{res})$ (turbulent). Imsland [7] concludes that a quadratic model for annulus flow is not optimal, and different test data obtained from the North Sea points in the direction of laminar flow. The third output is a measurement of p_{bit}, i.e.

$$p_{bit} = p_c + \theta_1 q_{bit} + \rho_a g h_{bit} \tag{2.7}$$

The above model is considered as

$$\begin{aligned} x_{t+1} &= f(x_t, u_t) && (2.8) \\ y_t &= h(x_t, u_t) && (2.9) \end{aligned}$$

with state vector $x_t \in \mathbb{R}_x^n$, input vector $u_t \in \mathbb{R}_u^n$ and output vector $y_t \in \mathbb{R}_y^n$ at discrete time t. $f(x_t u_t)$ describe how the state vector propagates from x_t to x_{t+1} with time step Δt and $h(x_t)$ maps the state vector to its corresponding output. For parameter estimation, the state vector is augmented with n_p parameters with $p_t = p_{t+1}$. Hence, the dimension of the state vector is $n_x + n_p$. For simplicity, n_x will denote the number of states and augmented parameters in the following.

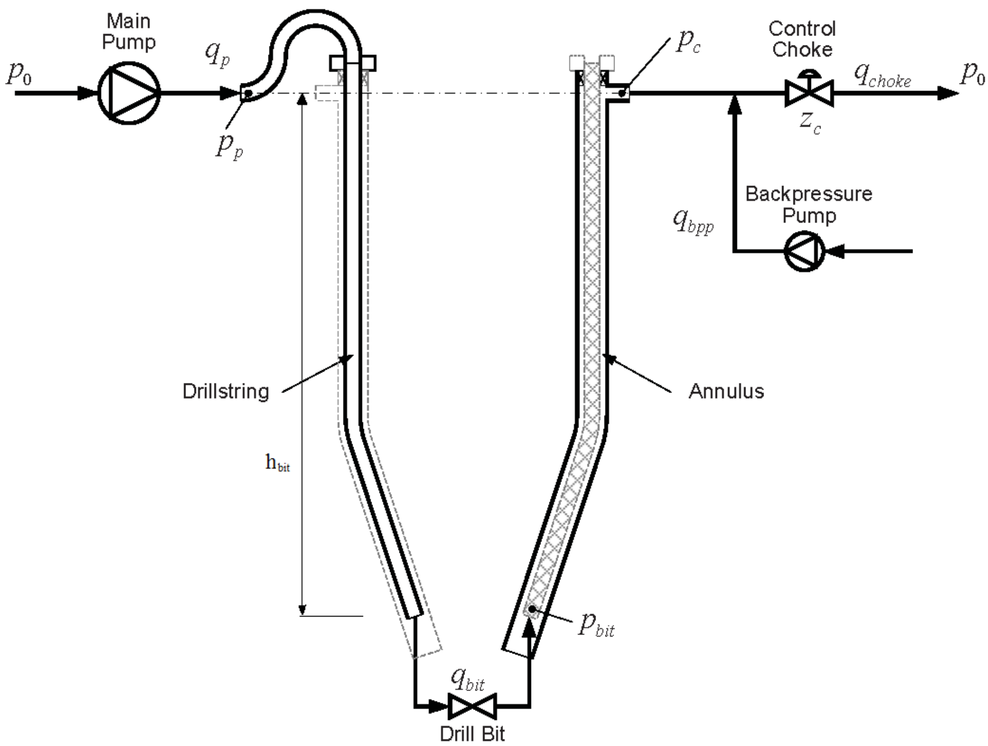

Figure 2.1: Modeled system

2.1.1 Choke Model

The flow through the choke is an important part of the model, and the plant-model mismatch is highly dependent on this term. In this project, as in [23], the classic orifice equation

$$q_{choke} = K_c z_z \sqrt{\frac{2}{\rho_a}(p_c - p_0)} \tag{2.10}$$

where $K_c = A_c C_d$, A_c being the opening of a fully open valve and C_d being the choke valve discharge coefficient, has been used. z_z is the normalized valve opening taking values between zero and one, and p_0 is the pressure at vena contracta (the location of minimum cross closest to the orifice in a fluid stream). A couple of assumptions have been made. First, p_0 has been approximated with the pressure downstream. Second, the orifice equation is based on assumptions of incompressible and steady flow, which is not valid for our system. Therefore, these assumptions have been neglected.

For implementation and fitting purposes, (2.10) is modified to

$$q_{choke} \quad = \quad K_c g_z(z_z)\sqrt{(p_c - p_{0)}} \tag{2.11}$$

where $K_c = A_c C_d \sqrt{\frac{2}{\rho_a}}$ and $g_z(z_z)$ is fitted to experimental data. Note that ρ_a here is lumped into K_c.

2.1.2 Back Pressure Pump

The back pressure pump is used to control the choke pressure p_c if it is necessary after the pump is fully stopped, i.e. $q_p = 0$. The term enters therefore directly into equation (2.1). A back pressure pump is not present in the scenario used for simulations in this work.

2.1.3 Reservoir Influx and Out flux

Reservoir influx ($q_{res} > 0$) and out flux ($q_{res} < 0$) represent possible unwanted gasses and fluids pushing into the well bore, and mud losses due to formation holes or areas with low pressure. A reservoir influx or out flux is expected to enter at the drilling bit and is thus added to the annulus friction term since it has to travel to the choke through the annulus. Intuitively, the term also enters directly into equation (2.1). Under normal conditions q_{res} can be neglected and it is thus for simplicity set to zero in this work.

2.1.4 Drill Bit Check Valve

Drill bit check valves are added to the drill string to prevent mud fluids from traveling up to the surface, i.e. keeping the bit flow positive ($q_{bit} > 0$). Equation (2.3) should therefore be modified to

$$\dot{q}_{bit} = \begin{cases} \frac{1}{M}(p_p - p_c - \theta_1 q_{bit} - \theta_2 \left|q_{bit} - q_{res}\right| (q_{bit} + q_{res}) & \\ +(\rho_d - \rho_a) g h_{bit}), & q_{bit} > 0 \\ max(\frac{1}{M}(p_p - p_c + (\rho_d - \rho_a) g h_{bit}),\ 0), & q_{bit} \leq 0 \end{cases} \tag{2.12}$$

which prevents q_{bit} from decreasing once it is zero.

2.1.5 Friction Model

Friction is the single most complex and uncertain factor to model and a wide varity of techniques can be applied. Previously, this thesis presented the laminar and turbulent model for the annulus and drill string frictions respectively as they appear in the Kaasa model [11]. This section discusses another modeling technique which makes use of function approximation.

2.1.5.1 Approximating Functions

An unknown function $h(x)$ that describes the friction is approximated with a normalized weighted average of N local approximators $\hat{h}_k(x)$, that is

$$\hat{h}(x) = \sum_{k=}^{N} \phi_k(x)\hat{h}_k(x) = \Phi(x) \begin{bmatrix} \hat{h}_1(x) \\ \hat{h}_2(x) \\ \vdots \\ \hat{h}_N(x) \end{bmatrix} \tag{2.13}$$

where

$$\phi_i(x) = \frac{\omega_i(x)}{\sum_{k=1}^{N} \omega_k(x)} \tag{2.14}$$

is a basis funcion which for each x forms a partition of unity, i.e the sum of all the function values at x is 1 and there is a neighborhood of x where all but a finite number of the functions are 0. $\omega_i(x)$ is the local weighting function which in this thesis is chosen as

$$\omega_i(x) = \begin{cases} \left(1 - \frac{|x-c_i|}{\mu_i}\right) & \text{, if } |x - c_i| < \mu_i, \\ 0 & \text{, otherwise} \end{cases} \tag{2.15}$$

which gives a pyramid form for each $\omega_i(x)$. c_i marks the top of the pyramid and μ_i the width at its base.

2.1.5.2 Approximating Friction

The "real friction" is assumed to be a function of the flow at the bit (and slowly varying, but this is omitted in further analysis for simplicity) and is denoted as $\mathcal{F}(q_{bit})$. Since the model of the friction, $F(q_{bit})$, is not accurate, i.e. $F(q_{bit}) \neq \mathcal{F}(q_{bit})$, model error is parametrized with a parameter vector $\theta \in \mathbb{R}^{n_\theta}$ and accompanying basis functions, $\Phi(q_{bit}) = (\phi_1(q_{bit}), ..., \phi_{n_\theta}(q_{bit}))^T$. The local approximator introduced in the prior section is defined as

$$\hat{h}_k(q_{bit}) \quad = \quad F(q_{bit})^T \theta_k \tag{2.16}$$

Further, an approximator $F(q_{bit})\Phi(q_{bit})^T\hat{\theta}$ is constructed, where there exist a θ such that

$$F(q_{bit})\Phi(q_{bit})^T\theta = \mathcal{F}(q_{bit}) \tag{2.17}$$

Last, annulus and drill string frictions are defined as $f_a(q_{bit})^T\theta_a$ and $f_d(q_{bit})^T\theta_d$ respectively, where $f(q_{bit}) = \Phi(q_{bit})F(q_{bit})^T$. The state equation for q_{bit} and measurement equation for p_{bit} are modified according to the above:

$$\dot{q}_{bit} = \frac{1}{M}(p_p - p_c - f_a(q_{bit})^T\theta_a - f_d(q_{bit})^T\theta_d + (\rho_d - \rho_a)gh_{bit}) \tag{2.18}$$

$$y_3 = p_c + f_a(q_{bit})^T\theta_a + \rho_a g h_{bit} \tag{2.19}$$

Chapter 3

Observer Theory

3.1 Nonlinear Moving Horizon Observer (NMHE)

This section discusses the "Regularized Nonlinear Moving Horizon Observer" (RNMHE) as it has been developed in [2], based on the moving horizon principle. The goal is to apply the theory on the model discussed in Chapter 2, similar to what has been done in [20].

The principle of a moving horizon in control and estimation theory is widely known. In model predictive control (MPC), one optimizes a cost function over a forward prediction horizon to acquire an optimal input over a control horizon [15]. In a moving horizon estimator however, the horizon is backwards in time. The goal is to obtain a state trajectory which fits measured data well, and is consistent with an assumed model. Unfortunately, data may not excite all outputs for every point in time which can be problematic, especially for combined state and parameter estimation with an augmented state vector, which is central in this project.

A convergent estimator has been investigated earlier [20] , seeking to minimize the weighted and regularized least squares criterion

$$\begin{aligned} J(\hat{x}_{t-N,t}, \bar{x}_{t-N}, I_t) &= ||W_t(Y_t - H_t(\hat{x}_{t-N,t}, U_t))||^2 \\ &+ ||M_t(\hat{x}_{t-N,t} - \bar{x}_{t-N}||^2, \end{aligned} \tag{3.1}$$

where

$$
\begin{aligned}
Y_t &= \begin{bmatrix} y_{t-N} \\ y_{t-N+1} \\ \vdots \\ y_t \end{bmatrix}, && (3.2)\\
\bar{x}_{t-N} &= f(\hat{x}^o_{t-N-1,t-1}, u_{t-N-1}), && (3.3)\\
H_t(\hat{x}_{t-N,t}, U_t) &= \begin{bmatrix} h(\hat{x}_{t-N,t}) \\ h \circ f^{u_{t-N}}(\hat{x}_{t-N,t}) \\ \vdots \\ h \circ f^{u_{t-1}} \circ \cdots \circ f^{u_{t-N}}(\hat{x}_{t-N,t}) \end{bmatrix}, && (3.4)\\
f^{u_t} &= f(x_t, u_t), && (3.5)\\
h^{u_t} &= h(x_t, u_t), && (3.6)
\end{aligned}
$$

$\hat{x}^o_{t-N,t}$ is the optimal state estimate, $I_t = col(y_{t-N}, ..., y_t, u_{t-N}, ..., u_t)$ carries all prior outputs and inputs for the horizon N, and M_t and W_t are time-varying weight matrices. In brief, M_t is the penalizes errors in every state the same and it is implemented as $\Lambda \cdot I_{n_x}$, where Λ is an adjustable gain. To force the observer to penalize errors in particular states or parameters more/less, one can increase/decrease the corresponding value on the diagonal of M_t, i.e. $\Lambda = [w_0 \dots w_{n_x}]^T$.

W_t, which weights error in measurements, is not as simple. Sui Dan et al. [2] derives how W_t shall be calculated for systems which are not asymptotically stable and have model errors, as is often the case with mixed state and parameter estimation. In short, it aims for zero weight on components that are either unobservable or unexcited. The adaptive law for W_t is defined as

$$||W_t \frac{\partial H}{\partial x}(\hat{x}^o_{t-N,t}, U_t)^+|| = \alpha, \qquad (3.7)$$

where

$$U_t = \begin{bmatrix} u_{t-N} \\ u_{t-N+1} \\ \vdots \\ u_t \end{bmatrix}, \qquad (3.8)$$

the Jacobian $\frac{\partial H}{\partial x}(\hat{x}^o_{t-N,t}, U_t)$ describes sensitivity of output changes towards changes of the different states, and α is a sufficiently small scalar. The elements ($\sigma'_{i,t}s$, $i = 1, .., n_x$) of the diagonal matrix S_t in the singular value decomposition (SVD) of the Jacobian

$$\frac{\partial H}{\partial x}(\hat{x}^o_{t-N,t}, U_t) \;=\; U_{SVD,t} S_t V^T_{SVD,t} \tag{3.9}$$

that are zero or close to zero point out modes that are not observable nor have exciting input. The weight on these modes are effectively reduced to zero by setting

$$\frac{1}{\sigma_{\delta,i,t}} \;=\; \begin{cases} \frac{1}{\sigma_{i,t}} & , \text{if } \sigma_{i,i} \geq \delta > 0 \\ 0 & , \text{otherwise} \end{cases} \tag{3.10}$$

where δ is a tuning parameter, and choosing

$$W_t \;=\; \frac{1}{\alpha} V_{SVD,t} S^+_{\delta,t} U^T_{SVD,t} \tag{3.11}$$

where $S^+_{\delta,t} = diag(\frac{1}{\sigma_{\delta,1,t}}, ..., \frac{1}{\sigma_{\delta,n_x,t}})$. A more in-depth explanation is provided by Sui Dan et al. [2].

3.2 Unscented Kalman Filter (UKF)

The Extended Kalman filter (EKF) has been the most common way to deal with estimation of nonlinear systems. It propagates the power density function (PDF) through a linearization around the equilibrium of the nonlinear system. This means that one has to linearize the system at each time step. Julier et al. [9] discusses the limitations of the EKF, briefly summarized here:

- Calculating the linearizations can be difficult, error-prone and time-consuming as the Jacobian of higher order systems can be hard to obtain, or may not exist.
- A linear approximation of the error propagation has to be more accurate than what is often achievable for the linearized transformation to be reliable.

This section discusses another approach for state estimation widely known as the unscented Kalman filter (UKF) or sigma-point Kalman filter.
The algorithm used for the UKF in this thesis is presented in [12], originally developed by Julier et. al. ([8], [10]), and repeated below.

The augmented state vector is defined as

$$x_k^a = \begin{bmatrix} x_k \\ x_v \\ x_m \end{bmatrix}, \tag{3.12}$$

where x_k, x_v and x_n holds process states, process noise and measurement noise respectively. Augmentation is not necessary, but makes further calculations more straight forward as sigma points for process noise and measurement noise covariance also has to be generated later on which will become clearer when Equations (3.14)-(3.28) are examined.
The augmented state dimension is given by the sum of the process, process noise and measurement noise dimensions respectively

$$N = n_x + n_v + n_m \tag{3.13}$$

and the augmented covariance matrix is a diagonal matrix defined as

$$P^a = \begin{bmatrix} P_x & 0 & 0 \\ 0 & P_v & 0 \\ 0 & 0 & P_m \end{bmatrix}. \tag{3.14}$$

where P_x, P_v and P_m are the process, process noise and measurement noise covariance matrices respectively.
$2N+1$ sigma points are then calculated based on the present state covariance:

$$\mathbf{X}_{i,k-1}^a \begin{cases} = & \hat{x}_{k-1}^a, \quad i = 0 \\ = & \hat{x}_{k-1}^a + \gamma S_i, \quad i = 1, ..., N \\ = & \hat{x}_{k-1}^a - \gamma S_i, \quad i = N+1, ..., 2N \end{cases}, \tag{3.15}$$

where S_i is the ith column of the square root of the covariance matrix

$$S = \sqrt{P_{k-1}^a} \tag{3.16}$$

and

$$\gamma = \sqrt{N + \lambda} \quad , \quad \lambda = \alpha^2(N + \kappa) - N, \tag{3.17}$$

where α and κ are tuning parameters. κ is chosen $\geqslant 0$ to ensure semi-positive definiteness of the covariance matrix, and $0 \leq \alpha \leq 1$ controls the sigma point distribution. Van der Merwe [26] concludes that α ideally should be a small number, i.e. the sigma point distribution should be kept dense.
The ith sigma point is the ith column of the sigma point matrix

$$\mathbf{X}^a_{i,k-1} = \begin{bmatrix} \mathbf{X}^x_{i,k-1} \\ \mathbf{X}^v_{i,k-1} \\ \mathbf{X}^m_{i,k-1} \end{bmatrix}, \tag{3.18}$$

where the superscripts x, v and m refer to states, process noise and measurement noise respectively.
The sigma points are transformed through the state-update function

$$\mathbf{X}^x_{i,k/k-1} = f(\mathbf{X}^x_{i,k-1}, \mathbf{X}^v_{i,k-1}, u_{k-1}), \quad i = 0, 1, ..., 2N. \tag{3.19}$$

The a priori state estimate and a priori covariance are calculated as the weighted sum of the sigma points:

$$\hat{x}^-_k = \sum_{i=0}^{2N} (w_{i,m} \mathbf{X}^x_{i,k/k-1}), \tag{3.20}$$

$$P^-_{x_k} = \sum_{i=0}^{2N} (w_{i,c} \mathbf{X}^x_{i,k/k-1} - \hat{x}^-_k)(w_{i,c} \mathbf{X}^x_{i,k/k-1} - \hat{x}^-_k)^T, \tag{3.21}$$

where the weights $w_{i,m}$ and $w_{i,c}$ are defined as

$$w_{0,m} = \frac{\lambda}{N+\lambda}, \quad i = 0, \tag{3.22}$$

$$w_{0,c} = \frac{\lambda}{N+\lambda} + (1 - \alpha^2 + \beta) \quad i = 0, \tag{3.23}$$

$$w_{i,m} = w_{i,c} = \frac{1}{2(N+\lambda)} \quad i = 1, ..., 2N, \tag{3.24}$$

where β is a non-negative weighting parameter used to incorporate the zeroth sigma point for the calculation of covariance. Van der Merwe [26] states that the optimal value is $\beta = 2$ for Gaussian priors.
The mean and covariance of the measurement vector are calculated as

$$\hat{y}^-_k = \sum_{i=0}^{2N} (w_{i,m} \mathbf{Y}_{i,k/k-1}) \tag{3.25}$$

$$P^-_{\bar{y}_k} = \sum_{i=0}^{2N} (w_{i,c} \mathbf{Y}_{i,k/k-1} - \hat{y}^-_k)(w_{i,c} \mathbf{Y}_{i,k/k-1} - \hat{y}^-_k)^T \tag{3.26}$$

where

$$\mathbf{Y}_{i,k/k-1} = h(\mathbf{X}^x_{i,k/k-1}, \mathbf{X}^m_{i,k-1}, u_k), \quad i = 0, 1, ..., 2N \tag{3.27}$$

Finally, the cross variance and Kalman gain are calculated according to

$$P_{x_k y_k} = \sum_{i=0}^{2N} (w_{i,c}\mathbf{X}^x_{i,k/k-1} - \hat{x}_k^-)(w_{i,c}\mathbf{Y}_{i,k/k-1} - \hat{y}_k^-)^T \tag{3.28}$$

$$K_k = P_{x_k y_k} P_{\bar{y}_k}^{-1} \tag{3.29}$$

and the unscented Kalman filter estimate and its covariance are calculated using the standard Kalman update equations (for derivation see Appendix A):

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - \hat{y}_k^-) \tag{3.30}$$

$$P_{x_k} = P_{x_k}^- - K_k P_{\bar{y}_k} K_k^T \tag{3.31}$$

3.2.1 UKF with Missing Measurements

The unscented Kalman filter as presented in the previous section requires that measurements are available at all times. If one or more measurements are absent, the calculation of the state estimate in (3.30) will have difficulties as the measurement vector y_k will contain components without given values.

An immediate solution is to zero out corresponding columns in the Kalman gain matrix K_k so that the particular components in question do not take part in Equations(3.30)-(3.31). However, this will greatly affect Equation(3.31), as semi-positive definiteness of the covariance matrix no longer can be guaranteed, potentially introducing complex numbers in Equation (3.16). Only applying the Kalman gain with zeroed out columns in the state update equation (3.30) could be an option, but it is not desirable as the obvious relation between Equation (3.30) and (3.31) will be disturbed. There are better solutions which will be investigated further.

3.2.1.1 Applying Last Available Measurement

The easiest solution is to continue to run the filter with the last known measurements, which will work satisfactory in periods when new measurements are arriving frequently. However, there may be times when measurements are absent for longer periods of time and consequently much of the dynamics can to a large extend be neglected. Pipe connection during drilling is a valid example of this particular situation with lasting absence of the bottom hole pressure (BHP) and therefore requires a different, more complex solution. Also, if measurement update frequencies are low for a given system, estimates in between updates may neglect a big part of that system's dynamics and requirements for precision may not be met as the estimates will try to track the measurements.

3.2.1.2 UKF with Regularization (RUKF)

A third solution is to utilize the a priori estimates calculated in (3.20) and approximate replacements for the missing measurements. This can be done by

$$\bar{y}_k = h(\bar{x}_k, m, u_k) \tag{3.32}$$

where m is measurement noise, u_k is the current input and $\bar{x}_k$ holds the state measurements still available together with the a priori estimates of the remaining states. More precisely

$$\bar{x}_{i,k} = \begin{cases} y_{i,k} & \text{if } y_{i,k} \in \mathbb{R} \text{ is a measurement of } x_{i,k} \\ \hat{x}_{i,k}^- & \text{if the measurement of } x_{i,k} \text{ is unavailable} \end{cases} \quad i = 1, ..., n_x. \tag{3.33}$$

Further, the measurement vector is updated according to

$$y_{i,k} = \begin{cases} y_{i,k} & \text{if } y_{i,k} \in \mathbb{R} \text{ excists} \\ \bar{y}_{i,k} & \text{if } y_{i,k} \text{ is unavailable} \end{cases} \quad i = 1, ..., n_y \tag{3.34}$$

This solution is the most clever one as it makes use of all the information available and it does not require interchanging of more filters, which easily can complicate implementation.

To demonstrate the performance of this solution consider the van der Pool oscillator given by the two first order equations

$$\dot{x}_1 = x_2 \tag{3.35}$$

$$\dot{x}_2 = \mu(1 - x_1^2)x_2 + x_1 \quad \mu = 0.2 \tag{3.36}$$

with output vector $y = [x_1 \quad x_2]^T$. If direction of time is reversed, i.e. both equations are multiplied with -1, this system is unstable for certain initial conditions, in particular those which start outside an unstable limit cycle [13]. Those which starts within, will lead the system to converge to the equilibrium $[0 \quad 0]^T$. This example is carried out on the van der Pool oscillator with reversed time as it demonstrates the differences better.

Process and measurement noise is added to the system, both with covariance of $10^{-3}I_2$. Since it is not the point of this example to illustrate convergence from offset initial conditions, they are chosen equal to one another, $x_0 = \hat{x}_0 = [0.1 \quad 0.1]^T$. Also, $P_{x_o} = I_2$ and α, β, and κ are set to 1, 2 and 0 respectively. μ is set to 0.2 in the plant and 0.3 in the model to get a small, but still significant plant-model mismatch.

Figure 3.1 illustrates how poorly the observer performs when the measurement of x_2 is lost for the period $200 - 400$s. During this part of simulation, estimates are calculated based on the last measurement that was available to the observer.

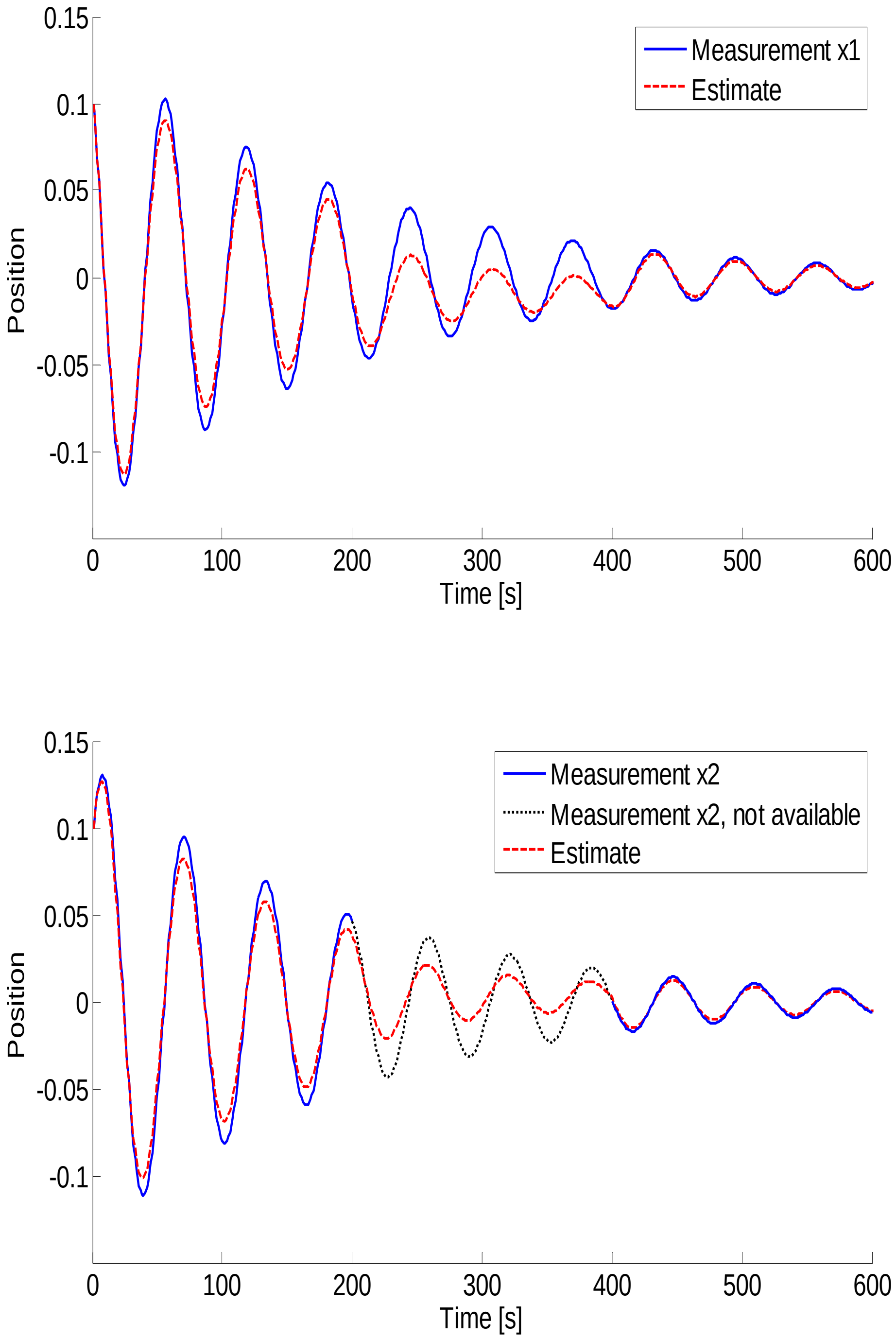

Figure 3.1: States for van der Pool oscillator where observer applies last available measurement at all times

Comparing that result to Figure 3.3, which make us of the regularization presented in Equations (3.32)-(3.34), it is probable that utilizing the information the measurement of x_1 provides and carry out estimation with regularized UKF instead of just applying the last available measurement is increasing performance considerably. By examining Figure 3.2, which plots the absolute differences between estimation errors for the two cases presented above, it is clear that for this test scenario performance is indeed enhanced. The peaks in Figure 3.2 indicate where the errors are largest, i.e at every local maximum of the curve in Figures 3.1 and 3.3 during the period $200 - 600$s. The oscillative behavior in the error difference is easily explainable by observing that the amplitude of the estimates are lower than for the measurements for the said period and thus they have to cross path at some point, forcing the error difference to zero at the point of intersection.

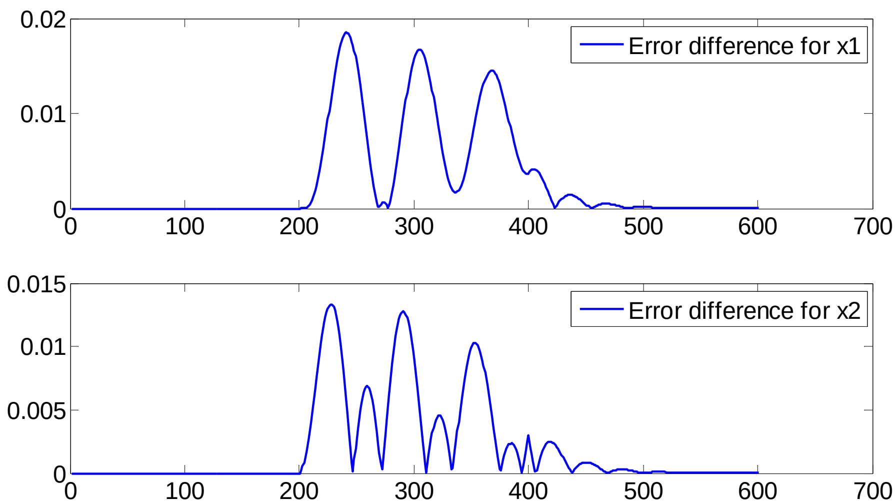

Figure 3.2: Additional estimation error for van der Pool oscillator without regularization, i.e. $||Error_{w/regularization} - Error_{wo/regularization}||$

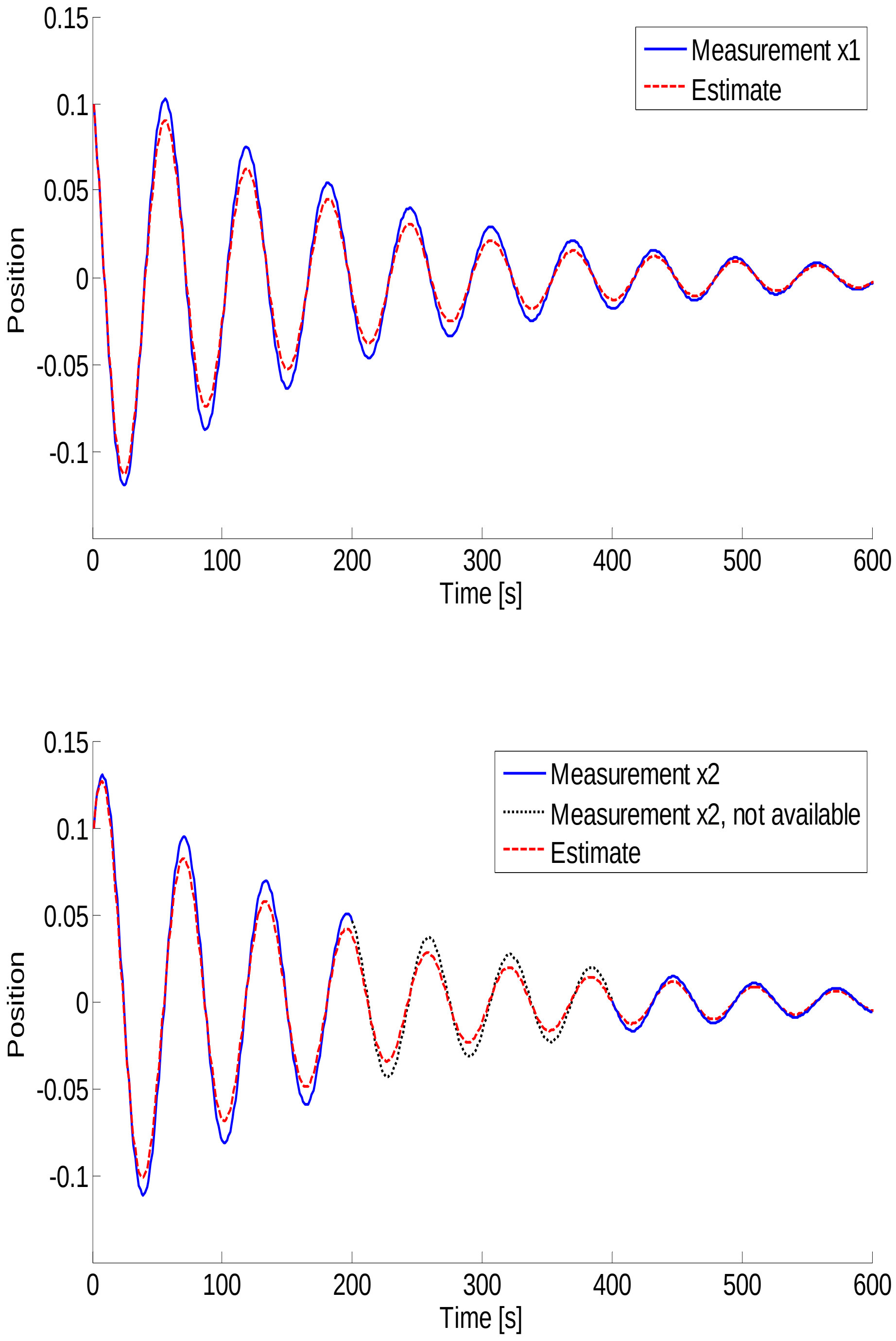

Figure 3.3: States for van der Pool oscillator with regularized UKF

3.2.2 Covariance Tuning

To achieve satisfactory estimation with a Kalman filter, tuning of the covariance matrices is crucial. For the unscented Kalman filter presented in Section 3.2 these matrices are denoted P_v and P_m, where the subscripts indicate process noise and measurement noise respectively. Primarily, it is the ratio between P_v and P_m that is decisive for good performance, but it is also important to have a fitting, mutual relation among the elements in the matrices. This is usually done by the inverse quadratic method of Bryson & Ho [1]. Here, the matrices are defined as

$$P_{v,o} = \begin{bmatrix} p_1^v & & 0 \\ & \ddots & \\ 0 & & p_n^v \end{bmatrix}; \qquad P_m = \begin{bmatrix} p_1^m & & 0 \\ & \ddots & \\ 0 & & p_n^m \end{bmatrix}$$

where $P_{v,o}$ is the unadjusted version of P_v, i.e

$$P_v = \zeta P_{v,o}$$

ζ is introduced for the purpose of being able to adjust the ratio between P_v and P_m as discussed earlier in this section.

The elements on the diagonal of $P_{v,o}$ are calculated according to

$$\left\{ p_i^v = \frac{1}{(\delta x_i)^2} \right\}_{i=1}^{n}$$

where δx_i is the maximum allowed estimation error in states, i.e $max|x_i - \hat{x}_i|$.

Similarly, the elements on the diagonal of P_m are calculated according to

$$\left\{ p_i^m = \frac{1}{(\delta y_j)^2} \right\}_{j=1}^{n}$$

where $\delta y_j = max|y_j - \bar{y}_j|$. $\bar{y}_j$ is the mean value of the stationary measurement and y_j is the largest observed deviation from this value. It is important to stress that these calculated values will only work as initial guesses for the covariance matrices and additional tuning of the diagonal elements and ζ is required. This is also emphasized by Bryson & Ho [1].

3.3 Observer Combinations

Two separate observers are evaluated in this thesis, but it is also interesting to look into possible combinations of the two. This Section presents four different ways to utilize the unscented Kalman filter in the moving horizon observer.

3.3.1 Prefiltering MHE with UKF

There are two terms in the least squares criterion (3.1) for the moving horizon observer introduced in Section 3.1. The first term penalizes model errors for an entire optimization horizon in the sense that real measurements are compared to estimates the model provides for a particular initial state at the horizon's beginning. The second term minimizes the deviation between the state estimate $\hat{x}_{t-N,t}$ and an initially calculated $\bar{x}_{t-N}$ (3.3). It forces the solver to find a solution close to the model-based estimate $\bar{x}_{t-N}$ which may not be optimal at all. Therefore, calculating $\bar{x}_{t-N}$ using a one-step unscented Kalman filter may increase performance considerably and estimates will potentially converge more rapidly to the optimal state estimate $\hat{x}^o_{t-N,t}$. The latter may easily prove to be false as the UKF simulation itself may increase computational complexity more than the MHE decreases it and this will be investigated thoroughly when results are presented.

3.3.2 Utilize Covariance Information

The unscented Kalman filter does not require much computer utility and it can thus be run in parallel with the moving horizon observer. Rao et al. [21] suggests the use of covariance information from the extended Kalman filter as the weighting matrix M_t in Equation (3.1). In particular

$$M_t = RP_x^{-1} \tag{3.37}$$

where P_x is the process covariance and R is a scaling factor. The covariance matrix is obtained from the one step UKF that prefilters $\bar{x}_{t-N}$. As $\bar{x}_{t-N}$ is an estimate provided by the Kalman filter, it is intuitive to vary the weighting of the error according to the covariance related to that particular estimate. If the covariance is high, M_t will contain smaller elements which will weight their corresponding errors less. Doing so is intuitive since the difference between $\bar{x}_{t-N}$ and $\hat{x}_{t-N,t}$ far from represents deviation from any optimal estimate as $\bar{x}_{t-N}$ indeed is just an estimate itself.

3.3.3 UKF in MHE

As discussed in Section 1.4, the project assignment explored different aspects of the moving horizon observer. Its main goal was to better the observer by looking into changes to the cost function, combined with improvements to the third order model. In particular, the effects of process noise in the model was examined by modifying the system equations to

$$\dot{p}_c = \frac{\beta_a}{V_a}(q_{bit} - q_{choke} + q_{back} - \dot{V}_a) + v_1 \tag{3.38}$$

$$\dot{p}_p = \frac{\beta_d}{V_d}(q_p - q_{bit}) + v_2 \tag{3.39}$$

$$\dot{q}_{bit} = \frac{1}{M}(p_p - p_c - \theta_1 q_{bit} - \theta_2 |q_{bit}| q_{bit} + (\rho_d - \rho_a) g h_{bit}, \tag{3.40}$$

with v_1 and v_2 being the modeled error in the states p_c and p_p respectively. All other variables and output equations were the same as previously defined in Section 2.1. Due to the projects time frame, this simple implementation was all that was tested, but other approaches were briefly discussed. In summary, some parts of the system equations are more undetermined than others, and one can therefore aim to target the parts that are associated with higher uncertainties. For this particular set of equations, several variables are, or contain, candidates for estimation, and they are thus also potential targets, e.g. the $\theta_i's$, $\rho_i's$ and q_{choke}. One can therefore model the process noise in sum with these variables, e.g. $-(q_{choke} + v_1)$ in (3.38).

The new proposed constrained, weighted and regularized least squares criterion presented in the project was

$$\begin{aligned} J(\hat{x}_{t-N,t}, \bar{x}_{t-N}, V_t, I_t) &= ||W_t(Y_t - H_t(\hat{x}_{t-N,t}))||^2 \\ &+ ||M_t(\hat{x}_{t-N,t} - \bar{x}_{t-N}||^2 + ||N_t V_t||^2, \end{aligned} \tag{3.41}$$

where

$$V_t = \begin{bmatrix} v_{t-N} \\ v_{t-N+1} \\ \vdots \\ v_t \end{bmatrix}, \tag{3.42}$$

$I_t = [y_{t-N} \dots y_t, u_{t-N} \dots u_t]^T$, and M_t, W_t and N_t are time-varying weight matrices. M_t and W_t are defined as before in Section 3.1, while $N_t = \Gamma\Omega$ for some scalar $\Gamma \geq 0$ and $\Omega \in \mathbb{R}^{n_v \times N \cdot n_v}$ that has a 1 at every entry, weighting the sum of the process noise for the entire window. The noise was free to vary to minimize the errors in estimates and states, but the direct weighting $||N_t V_t||^2$ were added to prevent the values in V_t from blowing up.

The modifications discussed above did provide promising results, but the additional time complexity was a considerable drawback. The number of optimization variables became huge (number of variables added to the system equations multiplied by the size of the moving horizon), and available machine power could not carry out simulations in a satisfactory way. In particular, the moving horizon had to be decreased to close to a fourth of its original size to keep a 3 hour real-time simulation within 24 hours.

As a result of these problems, combined with the promising results provided, this thesis seeks to find another approach that can possibly achieve similar performance. Therefore, using the unscented Kalman filter in combination with the MHE can be a clever solution, even though it is possible that time complexity again will be an issue. Section 3.1, in particular Equation (3.4) explains how the estimate is propagated through the model during optimization and by simply exchanging it the idea of process noise can be incorporated in the MHE without having to add several more optimization variables. The unscented Kalman filter will consequently, for each iteration of the optimization problem, estimate states and measurements for the entire horizon that minimizes the cost function (3.1).

3.3.4 Using UKF to Obtain $\hat{x}_t$ from $x^{\mathbf{o}}_{t-N,t}$

An even simpler approach which is far less time consuming then the one described in Section 3.3.4 is available. Here, the actual optimization problem is carried out normally by the regularized nonlinear moving horizon observer presented in Section 3.4 and the optimal estimate $\hat{x}^o_{t-N,t}$ is found accordingly. However, the current estimate $\hat{x}_t$ is not found by iterating $\hat{x}^o_{t-N,t}$ through the state model, but rather by using $\hat{x}^o_{t-N,t}$ as initial state condition for a separately tuned UKF. In this way, the effects of process noise can be incorporated nicely without a large time punishment.

3.4 Parameter Adaptation

To improve estimation, certain parameters in the model are estimated. When looking for parameters to estimate there are several options available, but to obtain an accurate model, it is important that one chooses those that are encumbered with high uncertainty. In this thesis a combination of the following parameters have been estimated:

Parameter	Description
K_c	Flow gain for choke-flow model
ρ_a	Density in annulus
θ_1	Slowly varying friction parameter

Table 3.1: Estimated parameters

3.4.1 Choke-flow Model Flow Gain K_c

In Section 2.1.1 the choke model is presented. It is emphasized that the flow gain K_c is a parameter with great uncertainty, and it is therefore a first priority for estimation.

3.4.2 Density of Mud in Annulus ρ_a

To come up with a mud mixture so that the density is equal throughout the mud is a near impossible task. Further, the availability of measurements that can paint a good picture of the changes in mud density in the annulus is limited and uncertain. Therefore, ρ_a seems like another natural choice for adaptive estimation. The fact that ρ_a is tuned offline based on measurements not available online, further motivates this choice.

3.4.3 Slowly Varying Friction Parameter θ_1

Section 2.1 briefly discusses the modeling of friction, in particular the fact that annulus friction for simplicity is modeled as laminar. Together with the fact that θ_1 enters directly into Eq. 2.7 for p_{bit}, the friction factor seems like another natural choice for estimation. Similar to ρ_a, θ_1 is tuned offline and in turn depends directly on ρ_a. This motivates a combination of these two parameters for estimation, together with the obvious choice of K_c.

3.5 Implementation

To test the performance of the observers thoroughly they have to be implemented in a simulation environment like MATLAB. Marcel Paasche ([19]) explains in more detail how this is done, but some important aspects are also discussed in this thesis, among them the implementation of the unscented Kalman filter.

3.5.1 Model Iteration

Several methods are available to integrate a discrete model from one time step to another, but especially for the MHE, the observer itself demands considerable computational power. Therefore, a couple of simple methods are evaluated for use, namely the Euler and midpoint methods which iterate the model according to

$$x_{k+1} = x_k + \Delta t \cdot f(t, x),$$

and

$$x_{k+1} = x_k + \Delta t \cdot f(t + \frac{\Delta t}{2}, x_k + \frac{\Delta t}{2} \cdot f(t, x)),$$

respectively. They are examined on the simple test system

$$\dot{x} = x, \quad x(0) = 1,$$

which has the exact solution

$$x(t) = e^t.$$

Figure 3.4 illustrates how the two different methods perform for different integration step lengths compared to the exact solution. A few key observations are made: First, it is evident that the midpoint method outperforms the Euler method considerably with all four step lengths (Δt). Second, none of the methods are close to the exact solution for large Δts and the Euler method requires an even smaller Δt to approach the exponential.

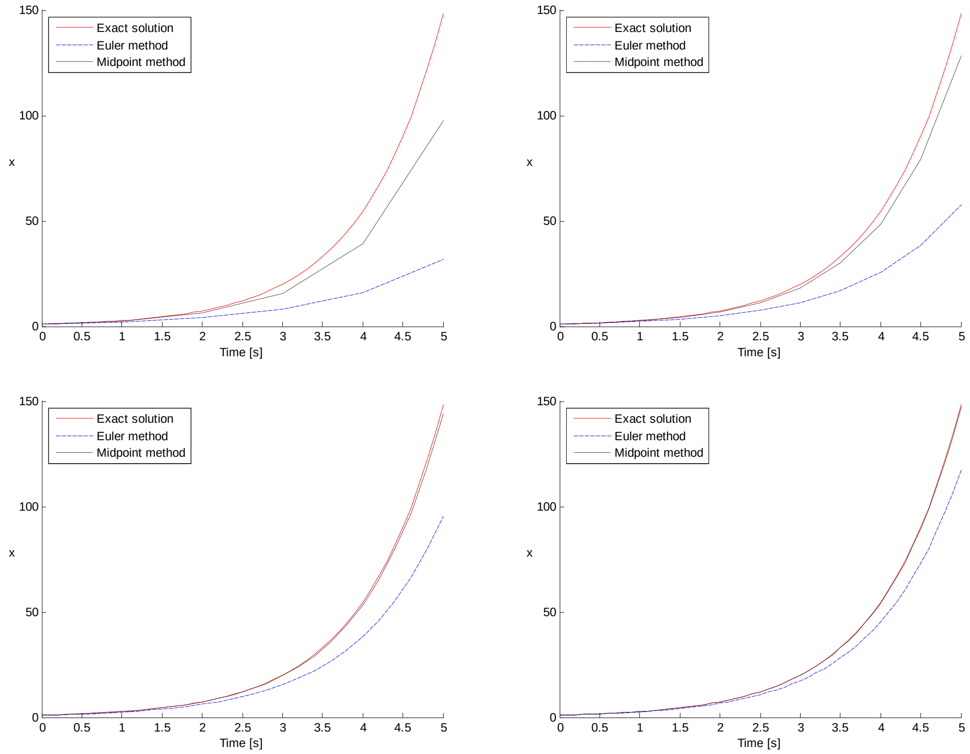

Figure 3.4: Accuracy of Euler (dashed, blue) and midpoint (dotted, black) methods for $\dot{x} = x$, Upper left: $\Delta t = 1$, upper right: $\Delta t = 0.5$, lower left: $\Delta t = 0.2$, lower right: $\Delta t = 0.1$

Shannon's version of the Nyquist-Shannon sampling theorem states that "if a function $x(t)$ is limited to the band from 0 to W cycles per second it is completely determined by giving its ordinates at a series of discrete points spaced $\frac{1}{2W}$ seconds apart" [22]. This theorem is complex to apply to the system in this work, so a sufficient integration step length is best found by trial and error. Running the simulations with a too high Δt will cause unwanted oscillatory behavior and Δt is therefore decreased until performance enhancement is saturated. The same approach is applied when the integration method is chosen. It is found that the simple Euler method with a small integration step length performs satisfactorily and it is thus no use in applying the more complex midpoint method. This concur with the results presented in Figure 3.4, in particular the two lower plots which demonstrates behavior with the lowest integration time steps.

3.5.2 Normalization

The pressure at the bit p_{bit} has a magnitude of around $240barg$ while the choke pressure p_c is varying between 10 and $30barg$. Consequently, the solver will penalize errors in p_{bit} more since its magnitude is higher and all errors are combined in a scalar penalty in the cost function. This scaling problem affects several other variables too and thus some sort of scaling is required. Ranges are thus defined for the variables that enter the cost function and normalization is carried out by dividing the variables with the maximum value of their range.

3.5.3 Solver for Least Squares Criterion in MHE

The least squares criterion in the moving horizon observer has to be minimized. For this project the TOMLAB Optimization Environment [18] is used for fast and robust large-scale optimization in MATLAB. A wide variety of solvers are available, among them ucSolve which is chosen for this particular optimization problem. ucSolve solves unconstrained nonlinear optimization problems with simple bounds on the variables, using several of the most popular search step methods (e.g. the Newton and Broyden-Fletcher-Goldfarb-Shanno (BFGS) methods) for unconstrained optimization.

Almost every classical optimization problem makes use of an objective gradient and a Jacobian matrix to minimize/maximize a particular cost function. More optimization variables give larger Jacobians, which is time consuming to work with from a computational complexity point of view. These matrices may be quite sparse and efficient methods that utilize the sparsity is available [3]. However, in this project, ucSolve is used as solver at all times.

3.5.4 Additional UKF Implementation Changes

The framework used in this work is developed by Marcel Paasche to simulate performance of the moving horizon observer. Briefly, it contains scripts and functions for model iteration, input controller, the moving horizon observer and cost function, parameter calculations, data loading, simulation evaluations and more. Also, the framework has the great benefit of being able to set up several simulations in an excel spreadsheet before execution so that multiple simulations can be carried out without any problems.

Marcel Paasche's work made it easier to implement and simulate the performance of the unscented Kalman filter as it just had to be integrated with the scripts and functions that already existed. The filter was first implemented on the less complex van der Pool oscillator (Section 3.2) to simplify troubleshooting and initial testing. Further, a similar function to what already existed for the MHE in the framework was created together with a variable that chose which of the two observers to apply during simulation.

Chapter 4

Simulation and Results

4.1 Grane Data

Simulations are performed in combination with data from Statoil's Grane field. However, all parameters are not known, so their values have to be decided. First, mud density is assumed to be constant throughout the annulus, and can therefore be calculated using $p = \rho g h + p_0$, which for conditions during zero flow gives

$$\rho_a = \frac{p_{bit,pd} - p_{c,pd}}{g h_{bit}}. \tag{4.1}$$

Then, average drill string mud density is calculated as

$$\rho_d = \rho_a - \frac{p_{diff}}{g h_{bit}}, \tag{4.2}$$

where p_{diff} makes up for the difference in pressure between pump and choke. This number is a tuning parameter, as the logs can not determine it precisely due to an assumed pressure release valve. Further, the annulus and drill string volumes are calculated with basic geometric considerations using data obtained from Grane, but these will not be displayed here. Last, friction factors are calculated using steady state values obtained from the Grane data, resulting in the simple equations

$$\theta_1 = \frac{p_{bit,ss} - p_{c,ss} - \rho_a g h_{bit}}{q_{bit,ss}}, \tag{4.3}$$

$$\theta_2 = \frac{-p_{bit,ss} + p_{p,ss} + \rho_d g h_{bit}}{q_{bit,ss}^2}. \tag{4.4}$$

Grane data also provides the normalized check-valve opening (Figure 4.1), which is equal for all simulations in this section.

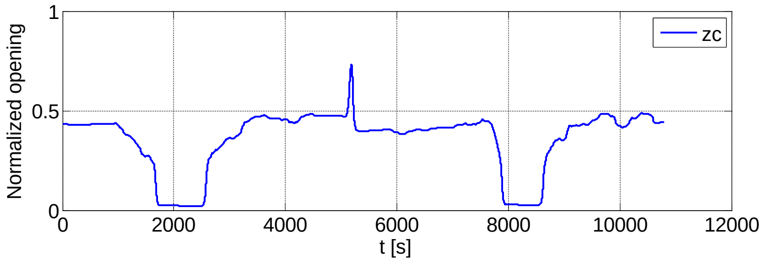

Figure 4.1: Normalized choke valve opening (0: closed, 1: fully open)

All parameters used are summarized in Table 4.1, with their respective descriptions.

Parameter	Value	Description
V_a	11.8	Annulus volume
V_d	4.6	Drill string volume
β_a	8000	Annulus bulk modulus
β_d	10000	Drill string bulk modulus
$\bar{\rho}_a$	0.0118	Average density annulus
$\bar{\rho}_d$	0.0115	Average density drill string
K_c	1	Flow gain for choke-flow model
p_0	0	Pressure downstream
θ_1	420	Annulus friction factor
θ_2	135310	Drill string friction factor
M	65000	Volume flow to pressure transition
h_{bit}	1827.6	Vertical depth of bit at $t = 0$
g	9.81	Gravitational force

Table 4.1: Chosen and tuned parameter values, calculated from data from Statoil's Grane field

4.1.1 Measurement Updates

In addition to parameter values Grane also provides measurements of p_c, p_p and p_{bit}. However, different update frequencies has to be handled as measurements of new choke and pump pressures are available at $f_{p_c,p_p} = 1Hz$, while pressure at the bit only is updated at $f_{p_{bit}} = 0.05Hz$. Estimation commences when a new measurement of p_c and p_p arrives, but the 19 seconds in between has to be handled in a clever way. The approaches are different for the two observers tested in this thesis.

4.1.1.1 Moving Horizon Observer

The moving horizon observer has a receding horizon which reduces the consequences of this problem if the size of the window is carefully decided. Choosing a large window will include more information about how the system is evolving with time, as several measurements of p_c and p_p will be included. Also, it is desirable to include a certain number of p_{bit} measurements, but because of the low update frequency the horizon will be intolerable large from a computational complexity point of view. Thus, it is even more important to keep the horizon so that it just includes a new measurement and by defining

$$N = \frac{\text{included measurements of } p_{bit}}{f_{P_{bit}}}$$

this particular requirement is met. Unless otherwise stated, $N = 40$ is used throughout this thesis, i.e. two measurements of p_{bit} are included in each estimation.

It follows from the different update frequencies that the measurement vector, Y_t (3.2), has to be modified since most of its entries do not hold a value. This is done by first defining the matrix

$$Y_{error} = Y_t - H(x_{t-N,t}, U_t)$$

where Y_t and $H(x_{t-N,t}, U_t)$ are defined as before (Section 3.1). Further, each element in Y_{error} that corresponds to an empty element in Y_t, is assigned a zero value. Last, to compensate for the arisen, skewed weighting in the least squares criterion (3.1), Y_{error} is modified according to

$$Y'_{error} = \frac{Y_{error} \cdot N}{N - M}$$

where M is equal to the number of zeros in Y_{error}. This ensures that the terms in the cost function are weighted correctly.

4.1.1.2 Unscented Kalman Filter

The unscented Kalman filter does not make use of a horizon and is thus even more vulnerable to lack of measurements. Therefore, this thesis presented different approaches to manage the problem in Section 3.2.1. Particularly, the UKF with regularization (Section 3.2.1.2) demonstrated promising behavior in initial tests and is thus the main focus for the pure UKF results presented in this section.

Another solution, that was introduced in Section 3.2.1.1, is to keep up estimation with the last known measurement. This is a solution that seemingly will provide good performance, but as mentioned earlier, concerns of neglecting system dynamics in the relatively long period between updates are present.

4.2 Parameter Estimation and Freezing

The regularized NMHE developed by Sui Dan et al. [2] and summarized in Section 3.1 possess a unique ability to freeze parameters when there is little information available in the system. This is done, as mentioned in Section 3.1 and thoroughly explained by Sui Dan et al. [2], by effectively setting the weight on these parameters to zero when observability and excitation are low. Consequently, the observer can estimate a larger number of variables satisfactory without doing so at the sacrifice of performance.

However, this is not the case for the unscented Kalman filter where every process state and parameter is estimated without regard to observability. Particularly, when measurements of p_{bit} are absent there may not be enough information in the system to properly estimate K_c, ρ_a , θ_1 or other desirable variables. These assumptions are taken into consideration when evaluating the UKF results.

4.3 Nonlinear Moving Horizon Observer (NMHE)

Most of the work and results on the nonlinear moving horizon observer were done in the project assignment written prior to this thesis. Therefore, the results presented in this section are given mostly for comparison purposes, but they will be as carefully examined and discussed as the rest. First, a simulation of the system with no parameter adaptation is executed, using calculated values from the Grane data (Section 4.1). Then, a combination of three adaptive parameters are tested to hopefully improve performance.

4.3.1 No Adaptation

The first simulation is of the system without any modeled process noise nor parameter adaptation, and it shows promising behavior for the RNMHE (Figure 4.2). The first two states, p_c and p_p, quickly converge to the measured values, and the same holds for the estimate of the bottom hole pressure, p_{bit}.

There is no measurement for the flow at the bit, and consequently nothing of particular interest can be said about the third state, q_{bit}. However, as anticipated from the system equations, clear correlations are found with p_c and p_c.

Simulation of the first pipe connection, which takes place during the second half hour of simulation, gives reliable estimates for p_{bit}. It is crucial to get good estimates during this particular phase, as availability of online measurements normally is absent. However, as seen from Figure 4.2, there do exist measurements in this particular data set. They are included solely for comparison and validation purposes, and consequently not used as inputs to the observer.

After the first pipe connection and throughout the rest of the simulation, estimation accuracy worsens dramatically, and $\hat{p}_{bit}$ is off by values approaching 10 bargs which is completely unacceptable. This easily traces to the error in the estimate of p_c, which in turn is affected greatly by q_{choke}. Confirmation is to a great extent found by comparing Figure 4.2 with 4.3, where a distinct correlation between q_{choke} and the estimation error for p_c can be seen . This highly motivates estimation of the flow gain for the choke-flow model, K_c.

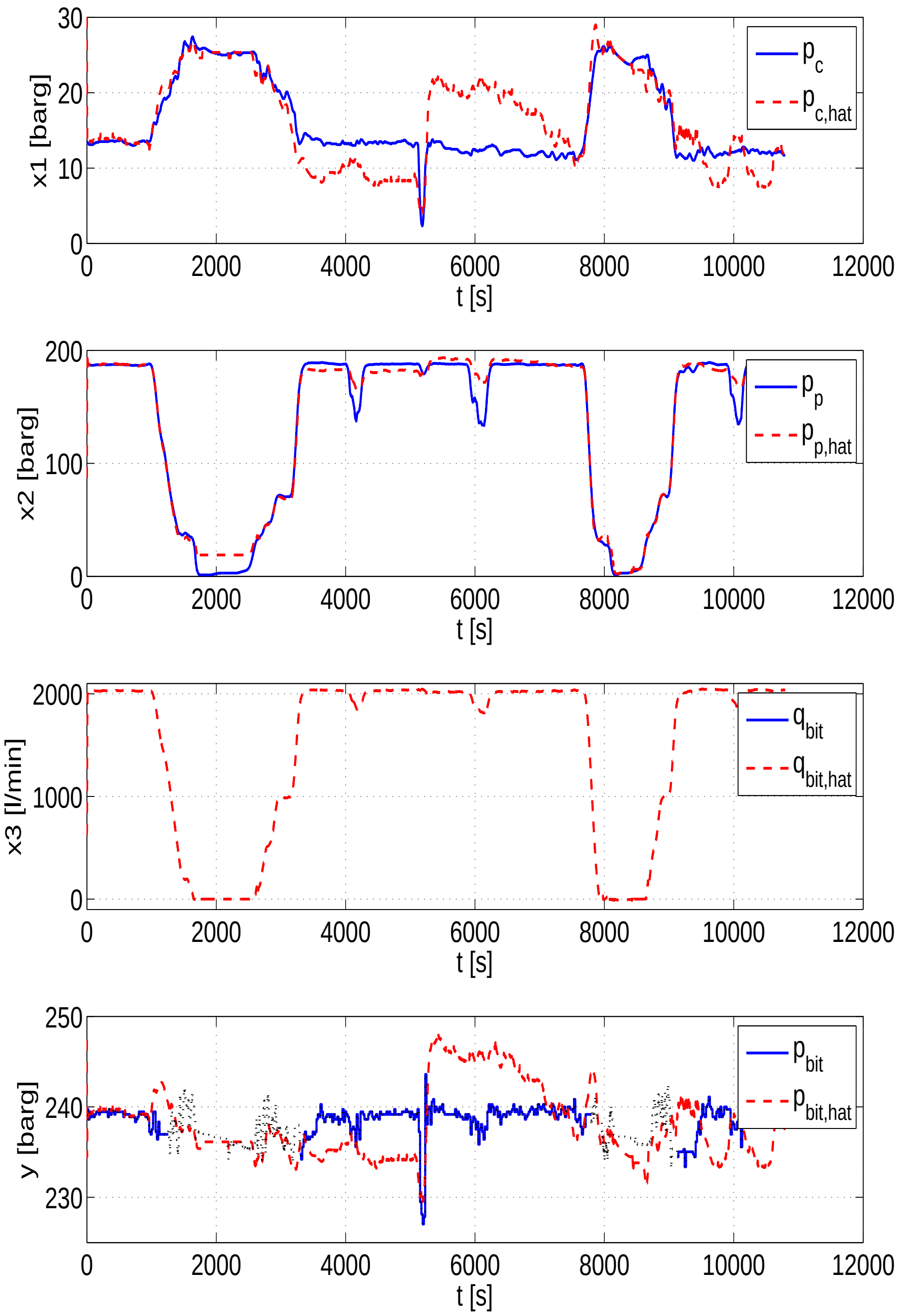

Figure 4.2: MHE without adaptation: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

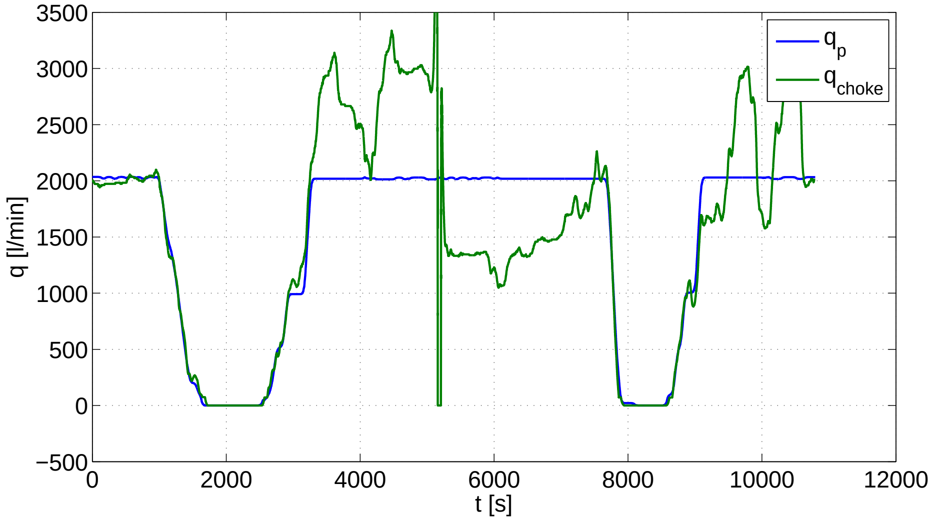

Figure 4.3: MHE without adaptation: Volume flow in pump (solid, blue) and choke (solid, green)

4.3.2 Adaptation of Flow Gain K_c

The preceding Section motivated estimation of flow gain K_c as changes in estimates were correlated with q_{choke}. Figure 4.4 clearly demonstrates how adaptation to K_c dramatically increases performance and the changes are easily traced to q_{choke} (Figure4.5) as expected. In particular, the estimates of p_c are significantly better throughout simulation and $\hat{p}_p$ also improves slightly, especially in the period between the two pipe connections of the simulated scenario. During pipe connections on the other hand, performance is about the same, which can be explained by a relatively good tuning of ρ_a and θ_1 in combination with a decent overall state estimation.

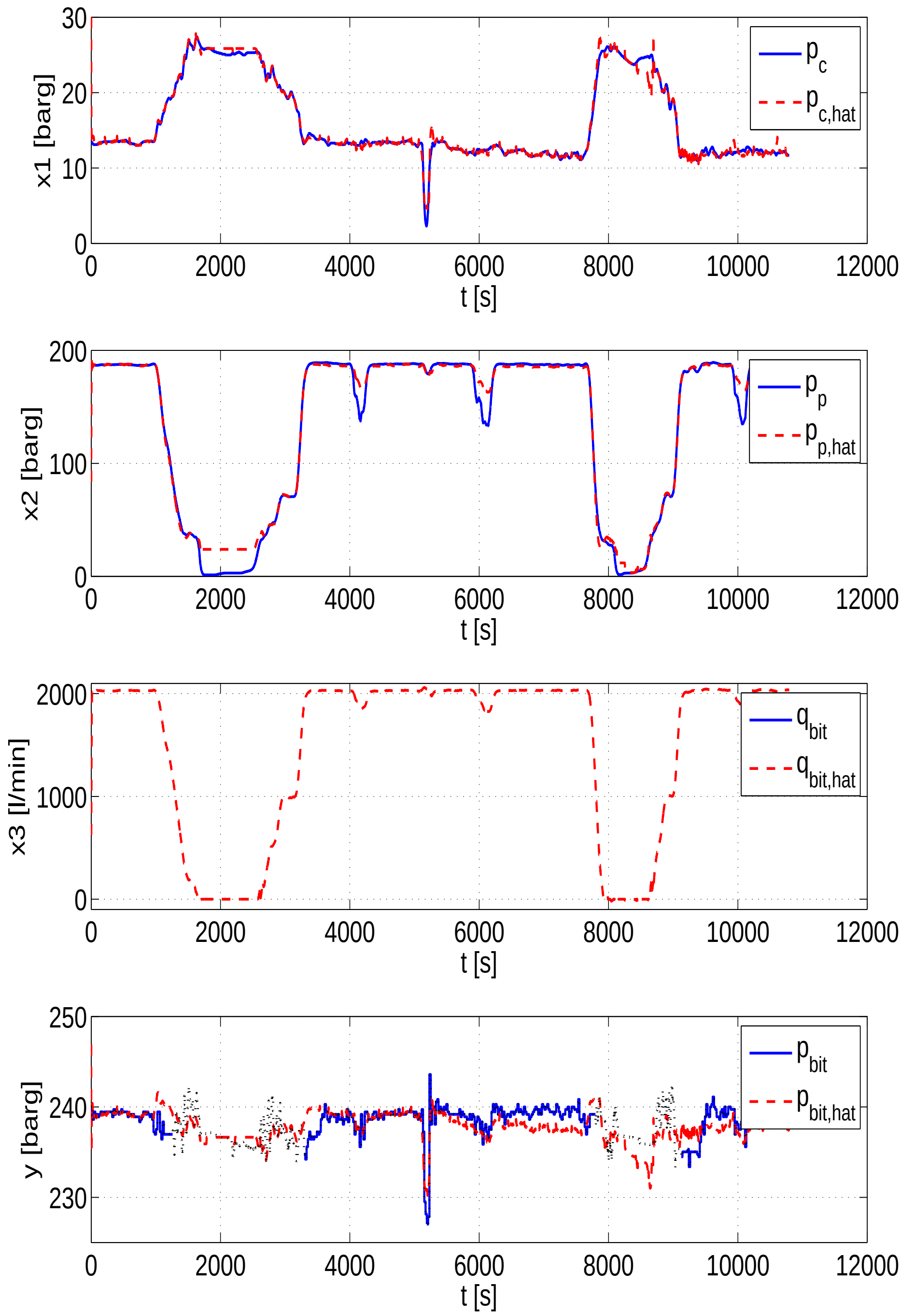

Figure 4.4: MHE with adaptation to flow gain K_c: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

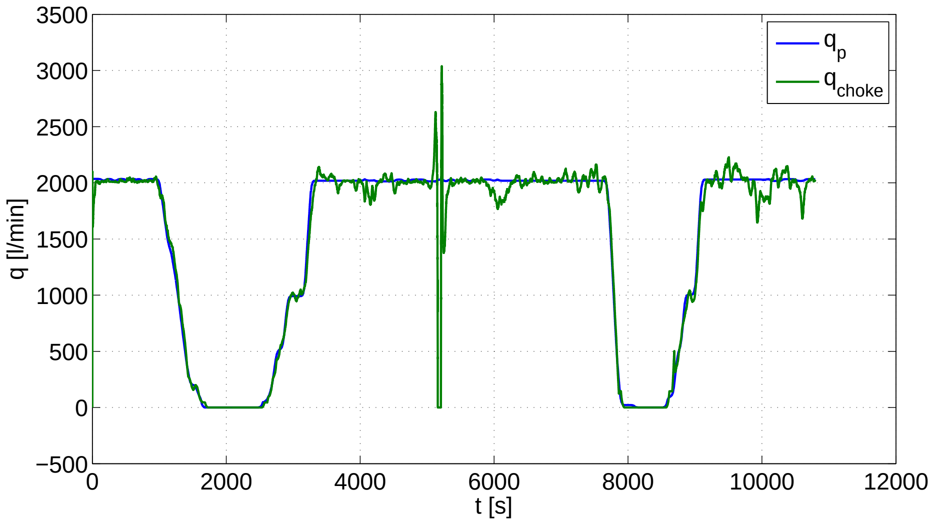

Figure 4.5: MHE with adaptation to flow gain K_c: Volume flow in pump (solid, blue) and choke (solid, green)

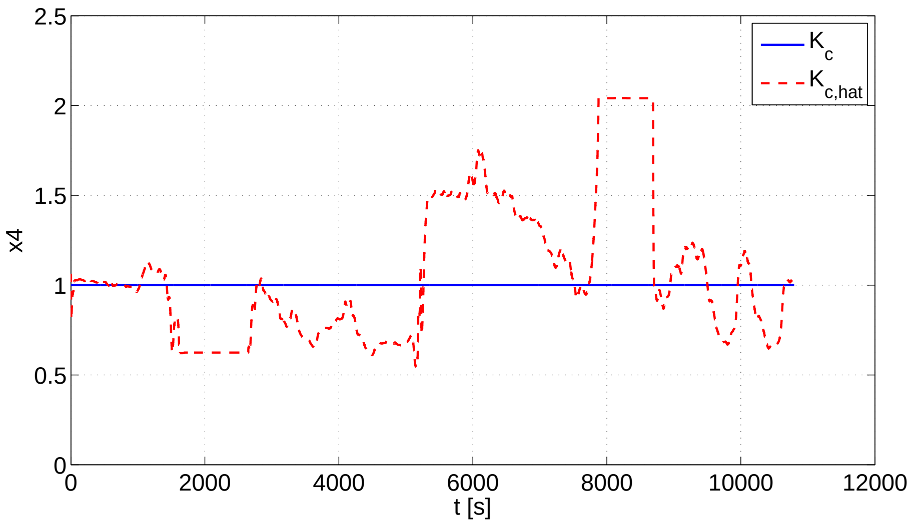

Figure 4.6: MHE with adaptation to K_c: Adapting K_c (dotted, red) and offline tuned K_c (solid, blue)

4.3.3 Adaptation of Flow Gain K_c, Annulus Density ρ_a and Annulus Friction Factor θ_1

In pursuit of an even better estimate, a combination of adaptive K_c, ρ_a and θ_1 is tested. The rationales behind this choice are many. First, the annulus density directly affects Equation (2.3) for q_{bit} and (2.7) for p_{bit}. Second, the density and friction factor are tuned to steady state information, which it is reasonable to anticipate does not hold for transient behavior that occurs during pipe connections. It is also important to recognize the way the choke model is implemented where ρ_a is lumped into K_c, as mentioned in Section 2.1.1. As steady state values used for offline tuning of θ_1 and ρ_a are directly related, it makes sense to try out a combination of these two parameters for estimation.

Unfortunately, the overall accuracy of the observer is only barely increased. By examining Figure 4.7 and comparing it to Figure 4.4 only a small improvement in the estimate of p_p is found when down-links (a communication from the surface to the bottom hole assembly (BHA), e.g. instructions to change drilling direction) are sent at approximately 4000, 6000 and 10 000 seconds. This affects the estimate of q_{bit} which enters directly into Equation (2.7) for p_{bit}, and by careful examination of the plots of p_{bit}, with and without adaption of ρ_a and θ_1, a minor decrease in estimation error is found.

However, the improvements are small, and it may be that the enhancements obtained by only adapting K_c are close to what is achievable. That being said, the estimated parameters can be hard to obtain online and adaptation is therefore desirable for all of them. This section provides valuable results that demonstrate that performance can be kept high, despite uncertainties in several different model parameters.

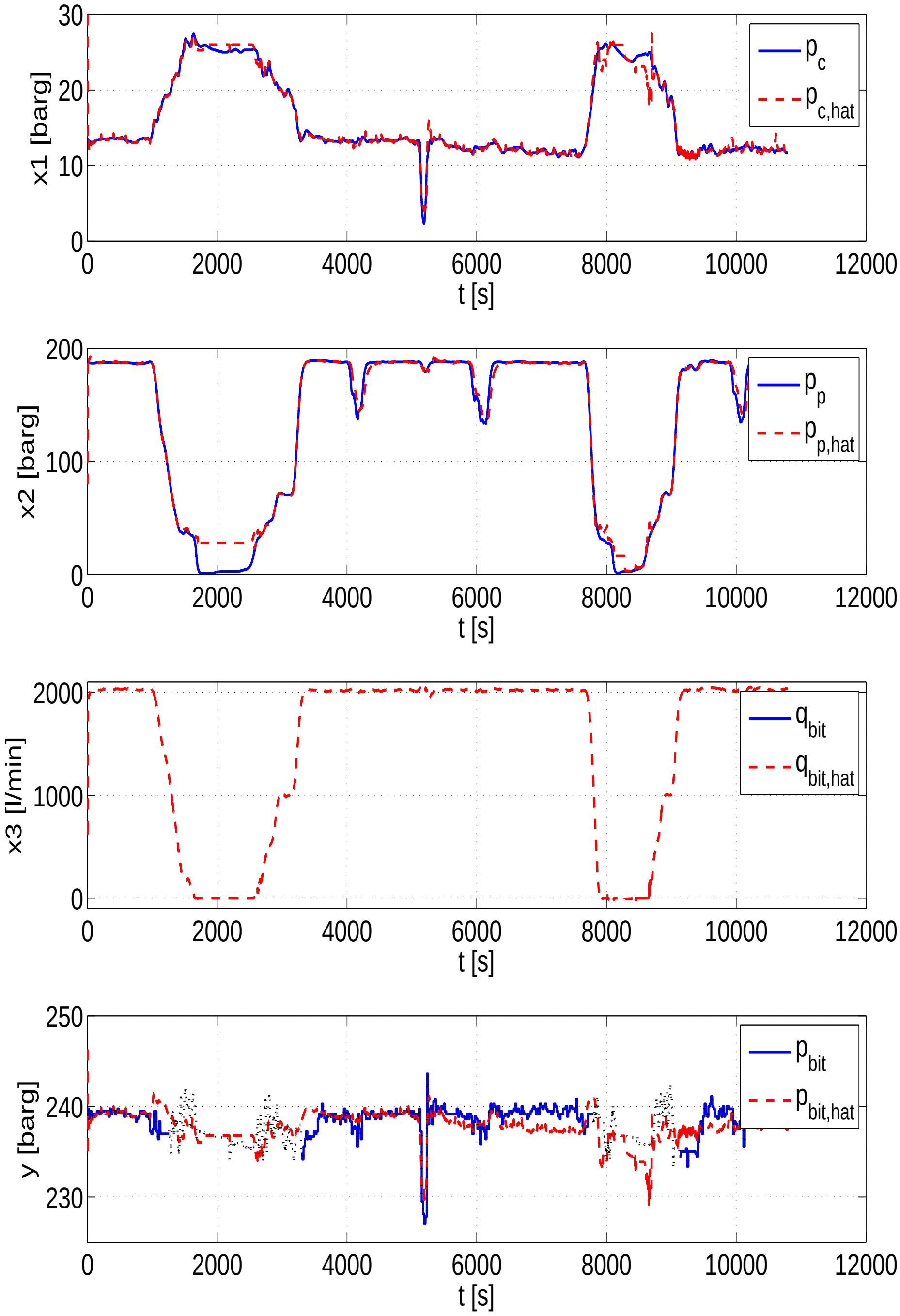

Figure 4.7: MHE with adaptation to flow gain K_c, annulus density ρ_a and annulus friction factor θ_1: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

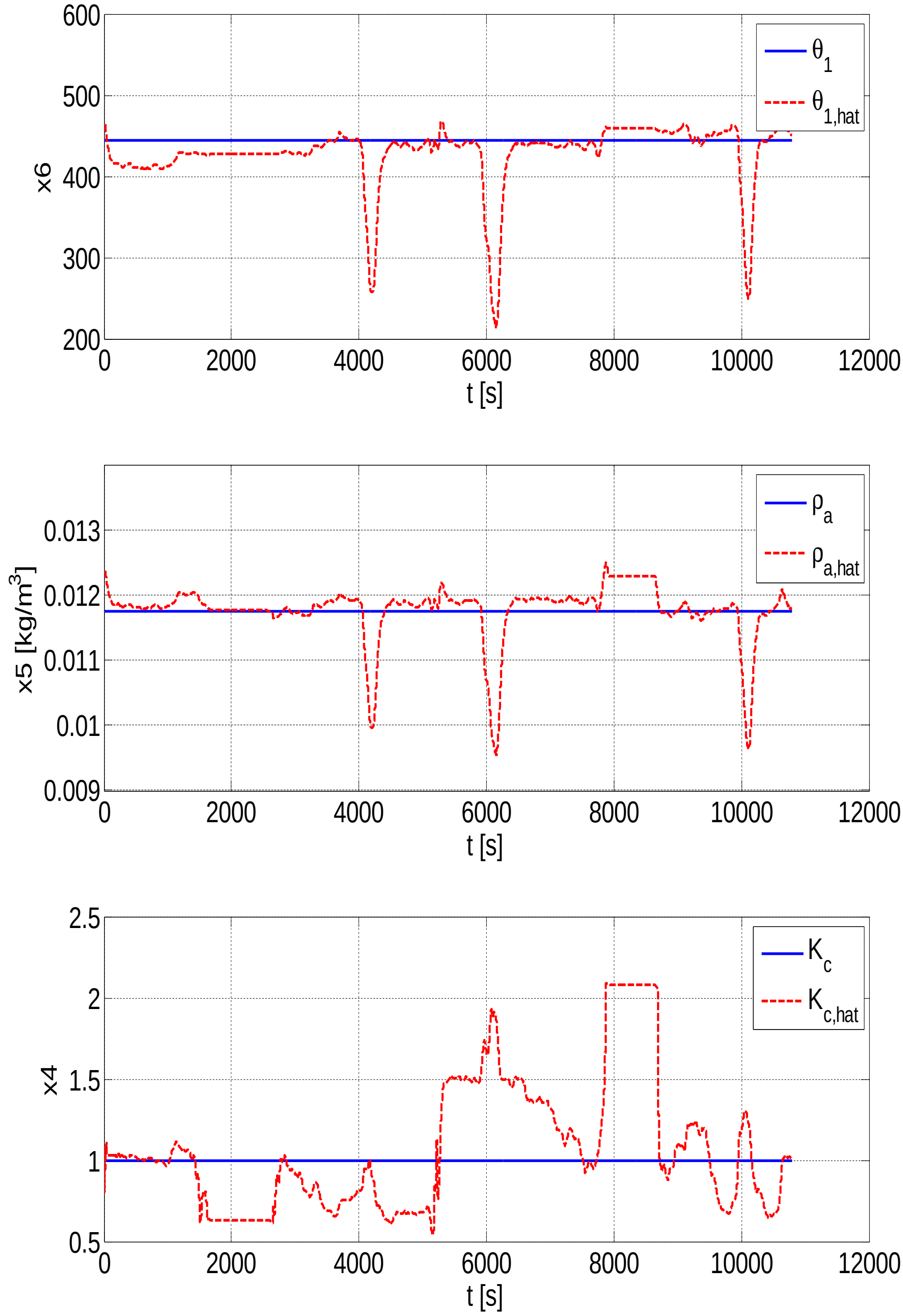

Figure 4.8: MHE with adaptation to flow gain K_c, annulus density ρ_a and annulus friction factor θ_1: Estimated parameter (dotted, red) and value used if parameter is not estimated (solid, blue)

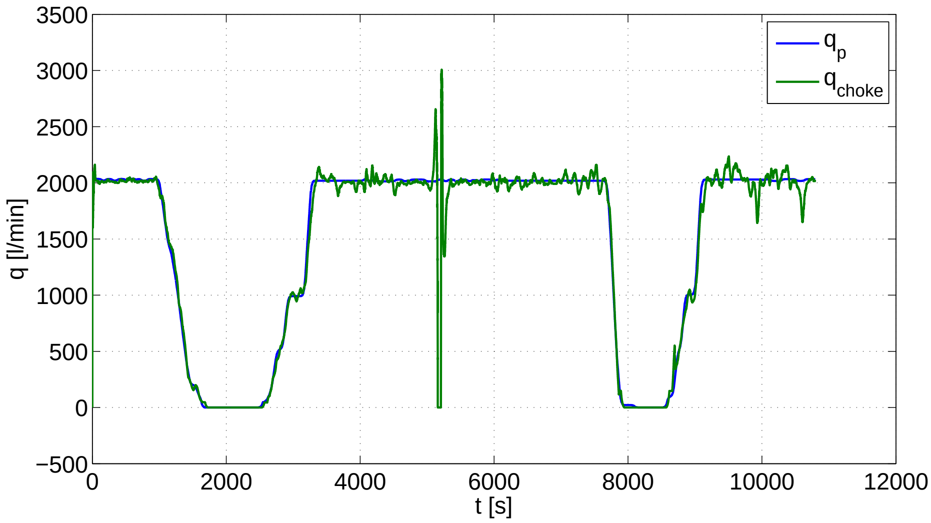

Figure 4.9: MHE with adaptation to flow gain K_c, annulus density ρ_a and annulus friction factor θ_1: Volume flow in pump (solid, blue) and choke (solid, green)

4.4 Annulus Friction Modeling

The modified friction model of the annulus was introduced in Section 2.1.5 and makes use of error parametrization and basis functions to approximate annulus friction. Figure (4.10) shows how the moving horizon observer, with adaptation to K_c and θ_i for $i = 1, \ldots, 4$, performs with the new proposed friction model. Unfortunately, the estimates with this friction modeling attempt deteriorate, but some minor promising behavior is found. After the last pipe connection to about the $10000s$ mark the estimate of p_{bit} is better than in most of the other simulations. Also, during the first pipe connection the estimate tracks the measurement well. However, the large peaks seen at around $1500s$, $4000s$, $6000s$, $8500s$ and $10000s$ are completely unacceptable. It may be that tuning and further investigation can improve the estimates, but due to particularly large computational load (Section 4.7) this was not prioritized and the attempt was considered a failure.

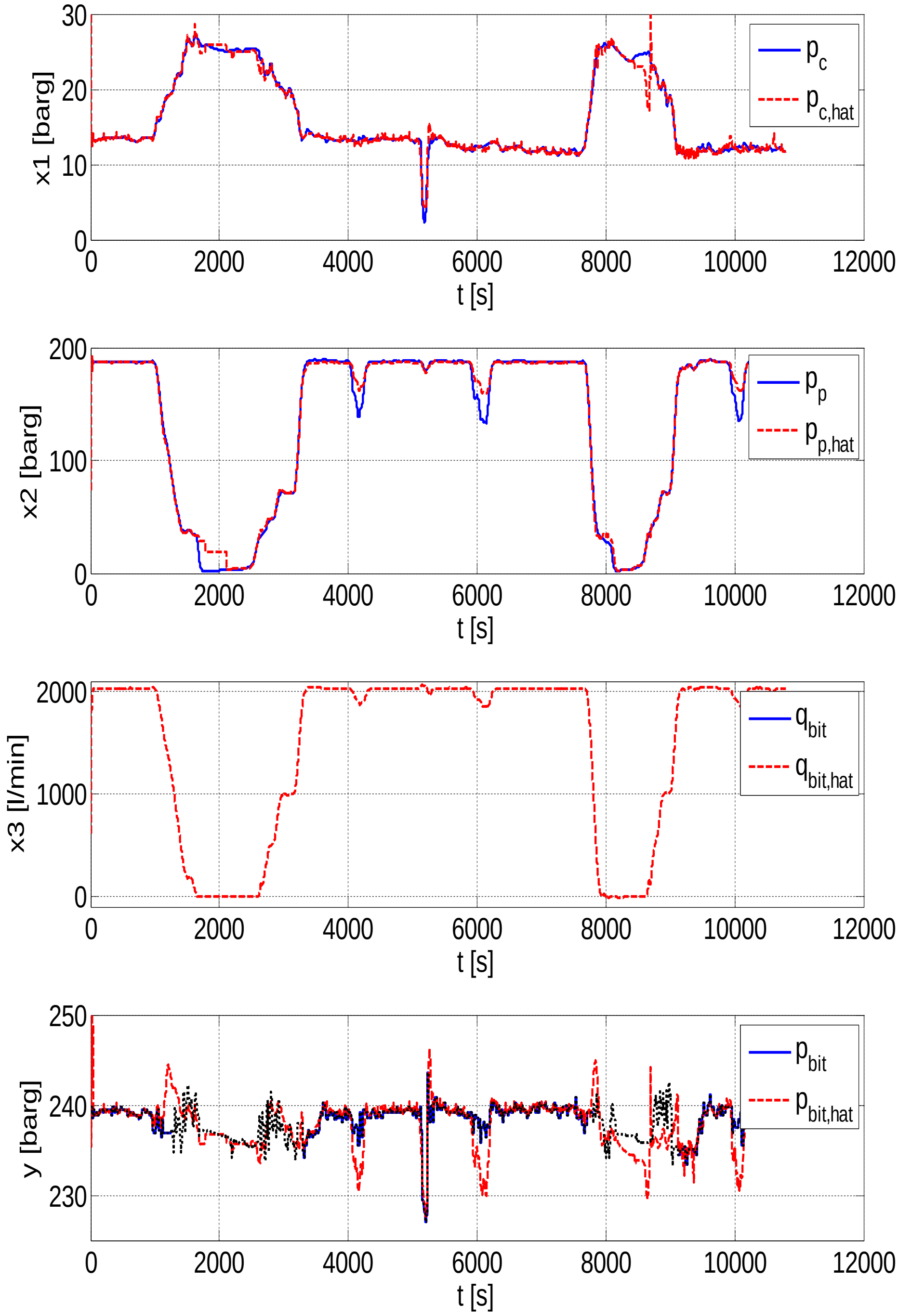

Figure 4.10: MHE with adapting friction model and adaptation to flow gain K_c: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

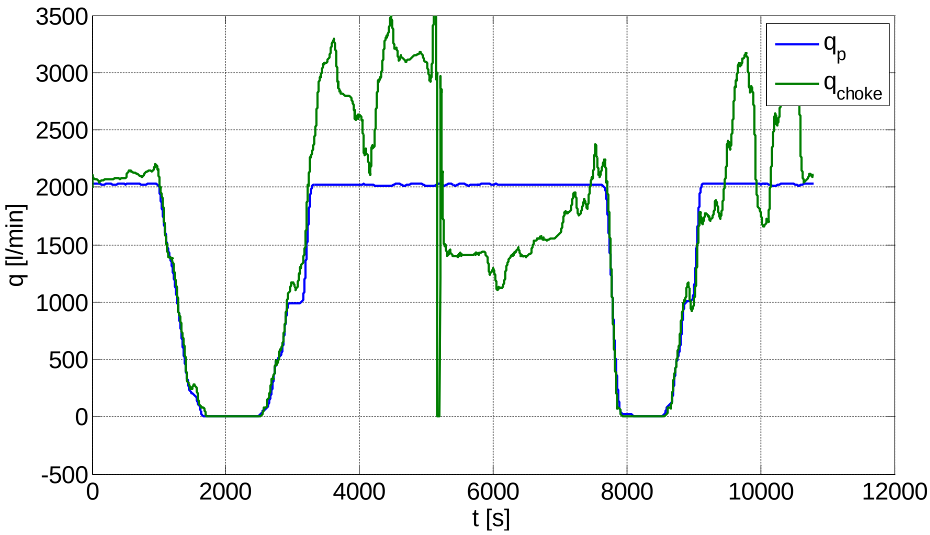

Figure 4.11: MHE with adapting friction model and adaptation to flow gain K_c:Volume flow in pump (solid, blue) and choke (solid, green)

4.5 Unscented Kalman Filter (UKF)

To see if the unscented Kalman filter can perform as well as the moving horizon observer, the UKF is tested on the same drilling scenario as in the prior section. Two filters are presented that utilize the measurement of p_{bit} in slightly different ways. The first UKF only updates the measurement for p_{bit} when a new measurement is available, whereas the second one applies the measurement that was received at the prior update at all times. The latter opens the possibility of a unique tuning of the filter.

4.5.1 Ignoring Last Available Measurement

The results presented in this section is of the unscented Kalman filter with the regularization introduced in Section 3.2.1.2 and it only utilizes measurements of p_{bit} when they arrive. The said regularization is applied both when measurements of p_{bit} are arriving at $0.05 Hz$ and during pipe connection when they do not exist at all. Simulations with no parameter estimation, adaptation to K_c, and adaptation to K_c, θ_1 and ρ_a are presented.

4.5.1.1 No Adaptation

Simulations of the unscented Kalman filter without any parameter adaptation, i.e. pure UKF, is presented in Figure 4.12. Immediately it is evident that the UKF performs better than the MHE (Figure 4.2) when no parameters are estimated. Both $\hat{p}_c$ and $\hat{p}_p$ have very high precision which to a large extent improves $\hat{p}_{bit}$, in contrast to the MHE results presented in Section 4.3.1. By further examination, a few similarities are found among the two observers. The jump in $\hat{p}_{bit}$ at approximately $1500s$ is found in both simulations together with the peak just before the $8000s$ mark. These errors are most likely a consequence of unmodeled phenomena and without parameter adaptation they are near impossible to remove.

During pipe connections, the pure UKF outperforms the pure MHE slightly with more consistency in the estimates. The overall performance is good and promising for further analysis.

The choke and pump flows (Figure 4.13) do not differ noticeably from the ones associated with the pure MHE. However, the correlation between errors in estimates of choke and pump pressures on the one hand, and choke flow on the other, is not found for the UKF. This can be explained as simple as that the UKF and MHE are two different observers, but it may also be that tuning is a decisive factor for the MHE that needs closer consideration.

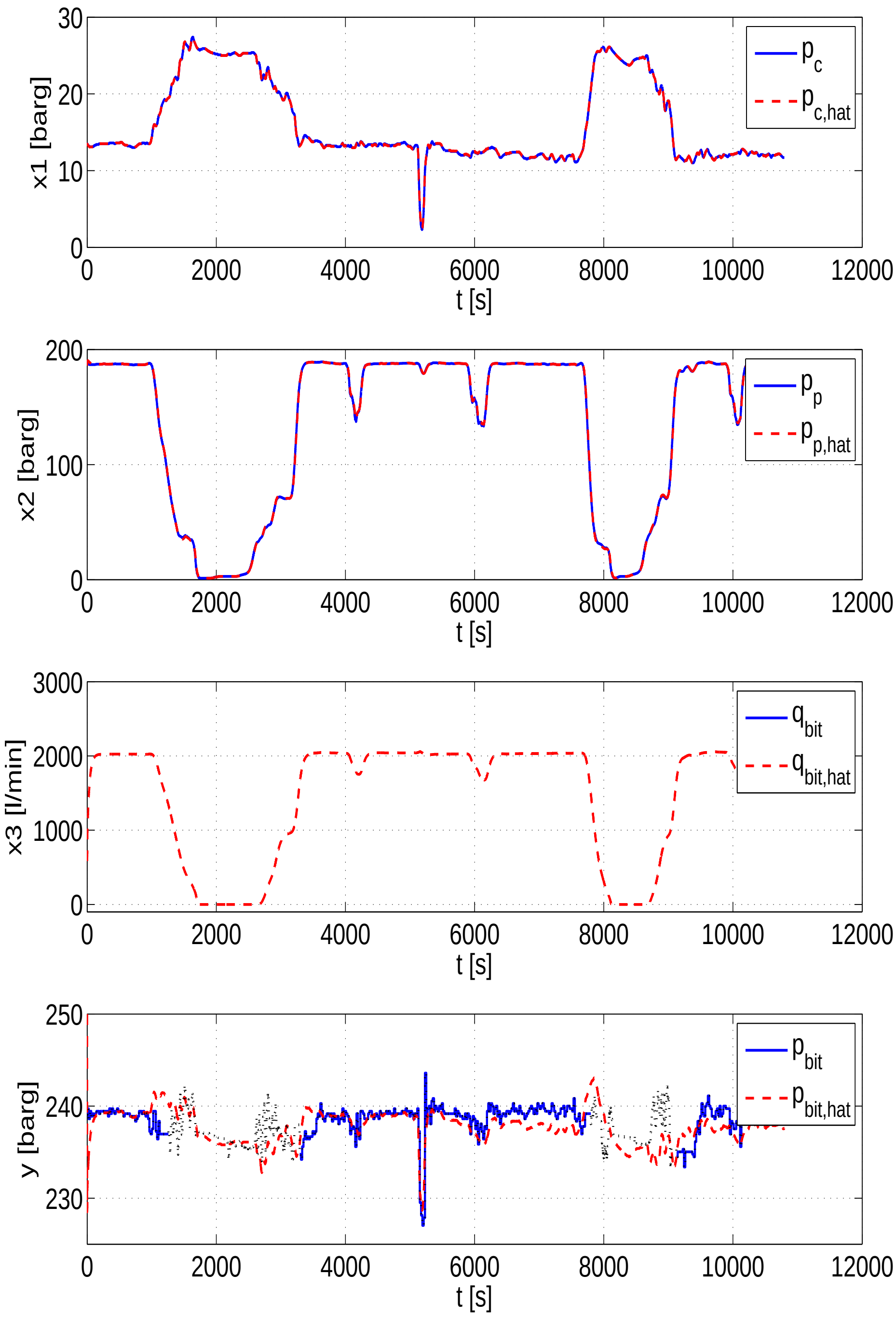

Figure 4.12: UKF without adaptation: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit}(dotted, black)

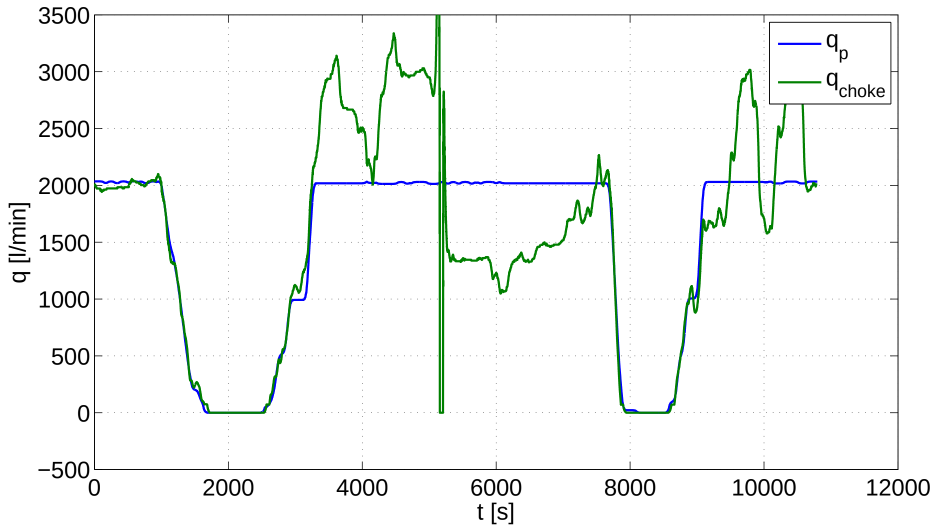

Figure 4.13: UKF without adaptation: Volume flow in pump (solid, blue) and choke (solid, green)

4.5.1.2 Adaptation of Flow Gain K_c

Introducing parameter adaptation enhances performance for the UKF as well as it did for the MHE, but as the UKF already performed relatively well the improvements are minor. Still, changes are found in key parts of simulation, i.e. during pipe connections. In particular, at around $3000s$, $8000s$, and $9000s$ (Figure 4.14) the estimates track the logged measurements of p_{bit} closer than in the prior section. Also, as experienced with the moving horizon observer, adapting to K_c changes the choke flow considerably.

Figure 4.16 illustrates how K_c varies with time and resemblance to K_c estimated by the MHE (Figure 4.5) is immediately recognized. It seems as though K_c decreases with time which can be explained with sediment in the choke. This is supported in that after de-clogging at approximately $5200s$, where the choke is fully opened for a short period of time , the estimate rises considerably before it decreases slowly again. This is an unmodeled progress that K_c seemingly incorporates in its estimate.

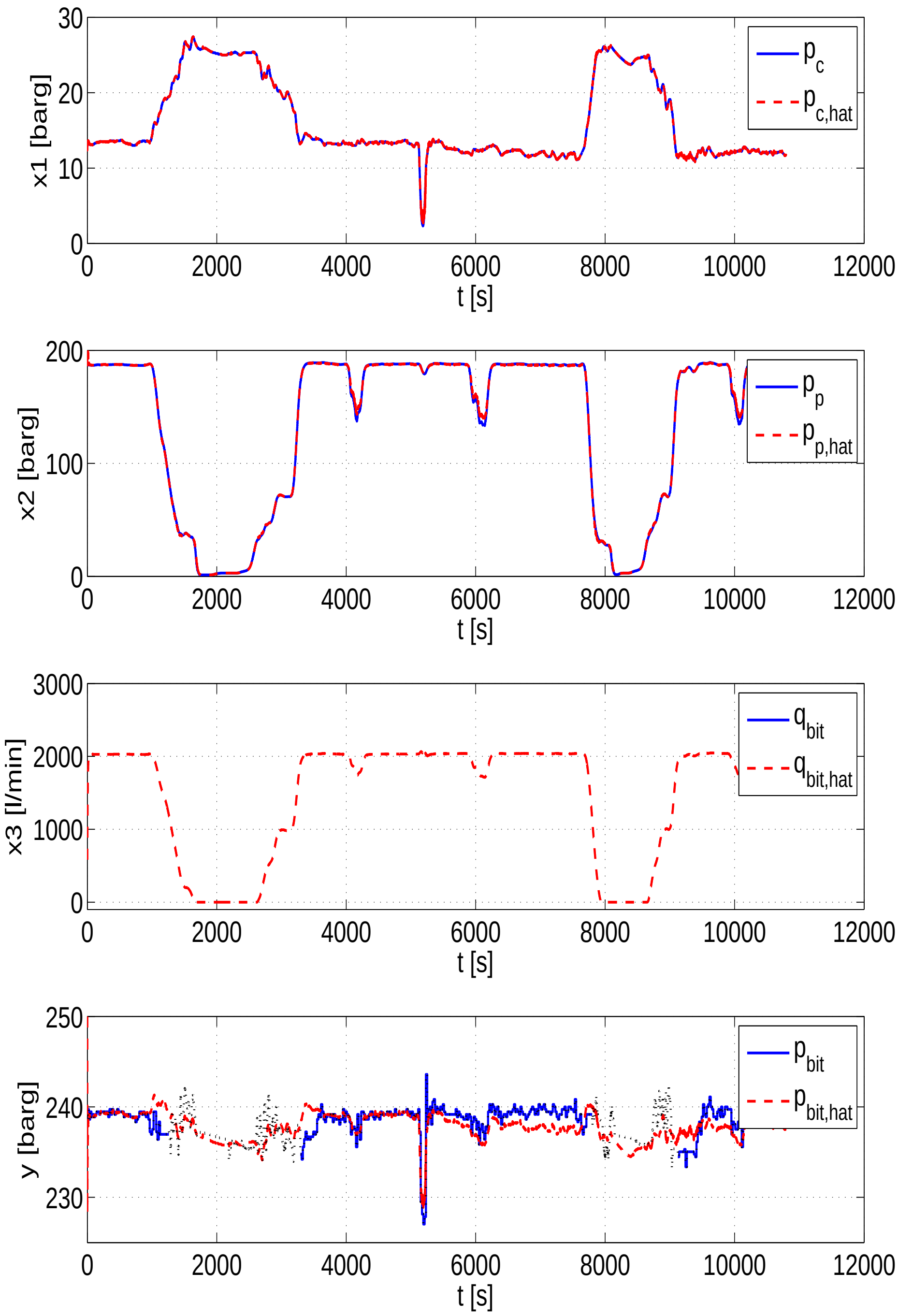

Figure 4.14: UKF with adaptation to flow gain K_c: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

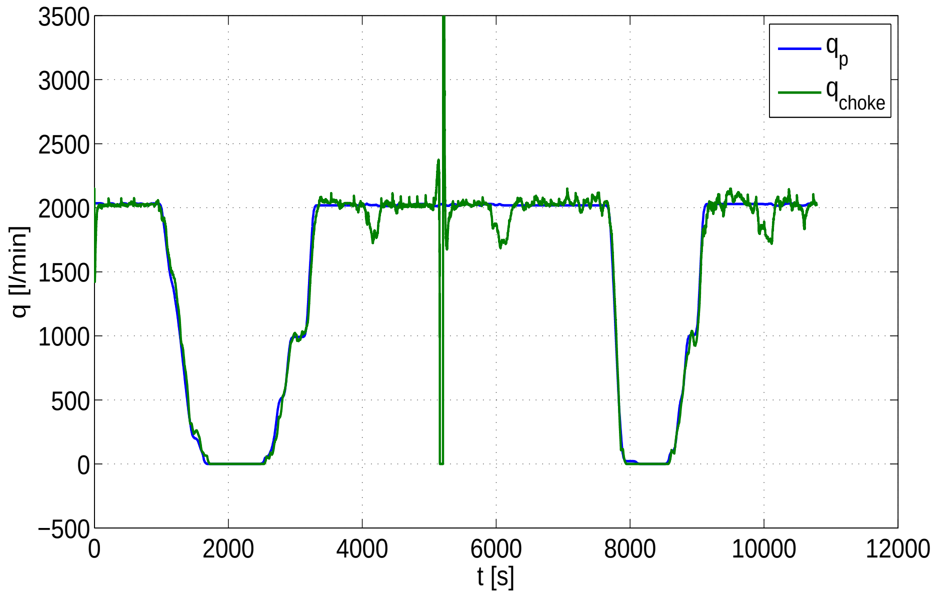

Figure 4.15: : Volume flow in pump (solid, blue) and choke (solid, green)

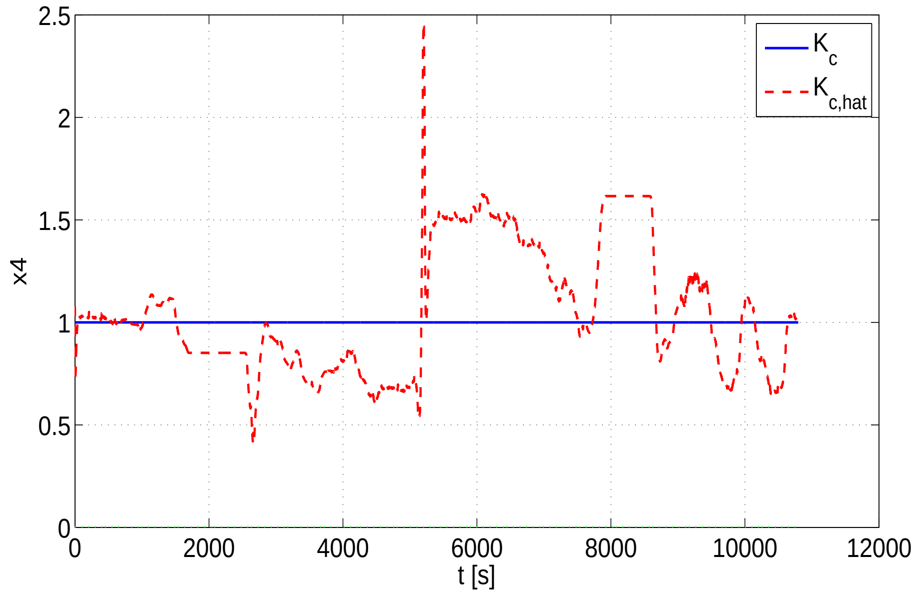

Figure 4.16: UKF with adaptation to K_c: Adapting K_c (dotted, red) and offline tuned K_c (solid blue)

4.5.1.3 Adaptation of Flow Gain K_c , Annulus Density ρ_a and Annulus Friction Factor θ_1

Section 4.2 discussed the possible effects of an increasing number of estimated variables with regards to observability. By evaluating Figure 4.17, the results of this Section confirm the expressed concern. Performance is unfortunately not enhanced, but more or less kept steady. Still, the estimates for p_c and p_p are impressively accurate while $\hat{p}_{bit}$ has the same errors at approximately the same places as before. However, it is satisfying to see that the parameter adaptation also works well for the unscented Kalman filter.

From the parameters (Figure 4.18) a few key observations can be made: First, ρ_a barely changes at all, mostly due to a tuning that penalize its derivative. This was done because a more freely varying $\hat{\rho}_a$ deteriorated the estimate of p_{bit}, probably as a result of the observability issues discussed in Section 4.2. Second, the estimated friction factor is slowly increasing throughout simulation after an initial jump at the beginning of the first pipe connection. This jump can be explained in that no substantial change in inputs and outputs are seen before this point and that the observability increases with growing excitation. The slowly increasing estimate may be traced to changes in the volume, $\dot{V}_a$, as drilling proceeds since larger volume in this case implies longer ways for the mud to travel which in turn implies more friction in the system. Last, the estimate of K_c differ from the prior section. However, the same trend with the decreasing value before and after de-clogging of the choke is found. The choke (Figure 4.19) flow is similar to what is seen in prior sections.

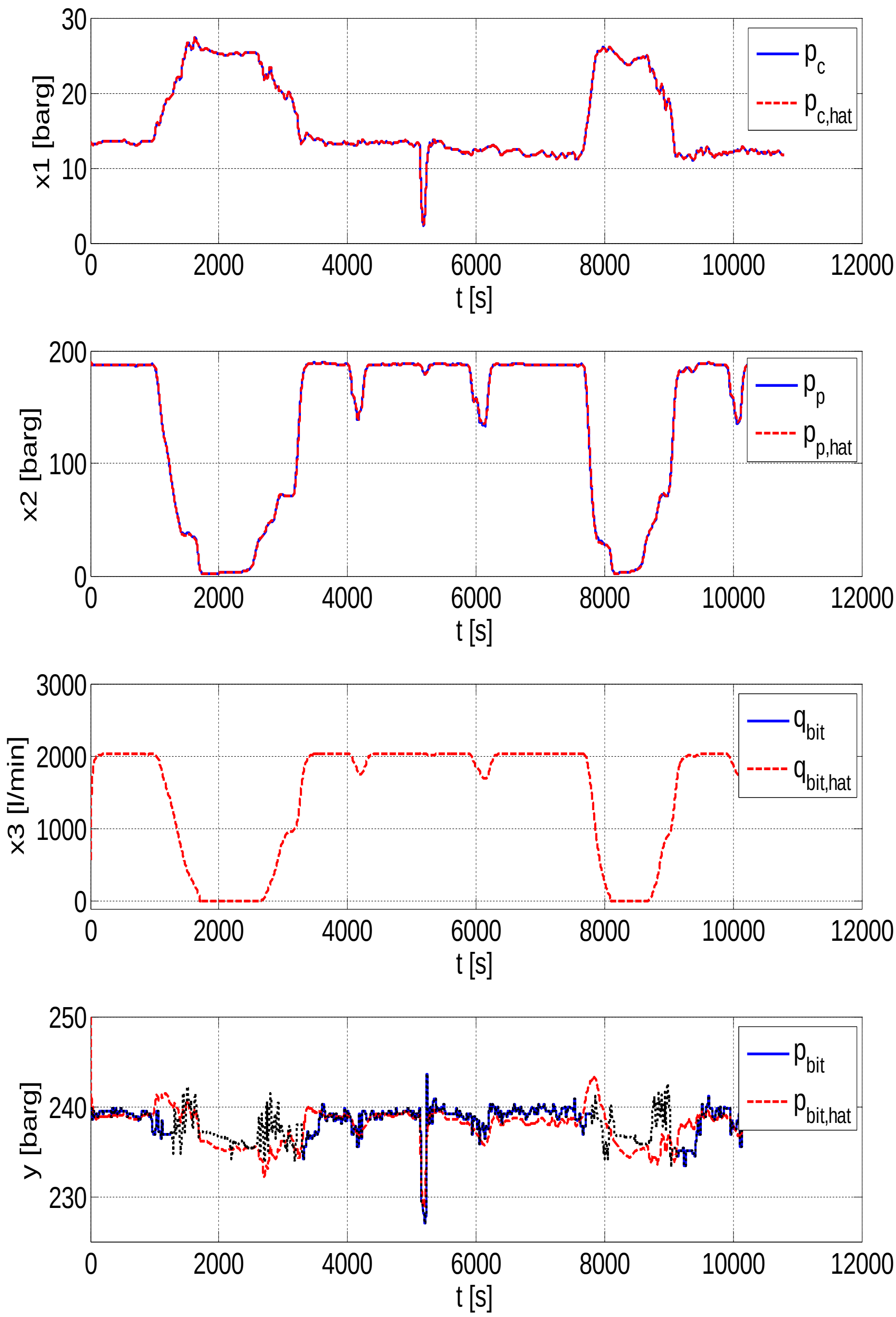

Figure 4.17: UKF with adaptation to flow gain K_c, annulus density ρ_a and annulus friction factor θ_1: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

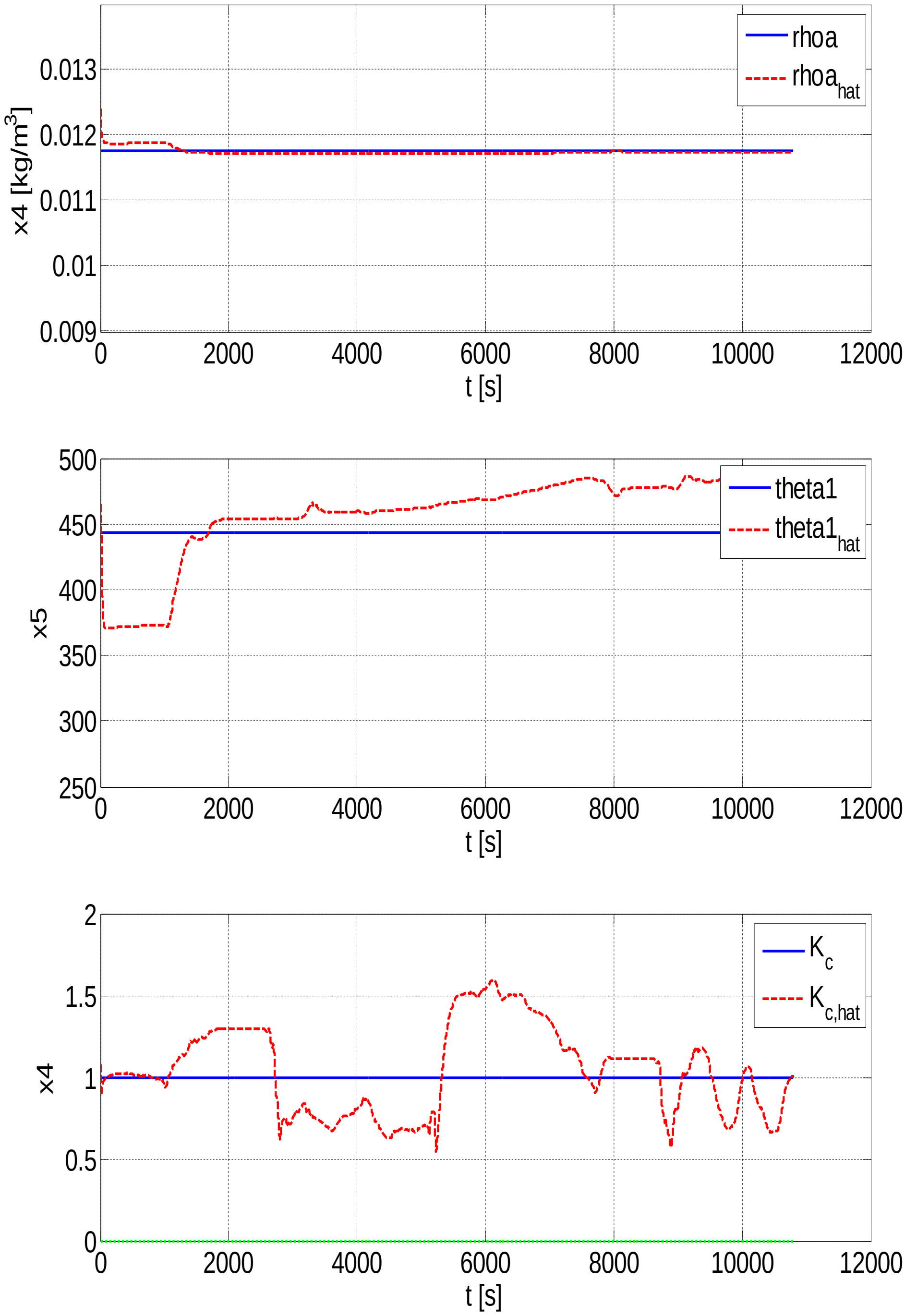

Figure 4.18: UKF with adaptation to flow gain K_c, annulus density ρ_a and annulus friction factor θ_1: Estimated parameter (dotted, red) and value used if parameter is not estimated (solid, blue)

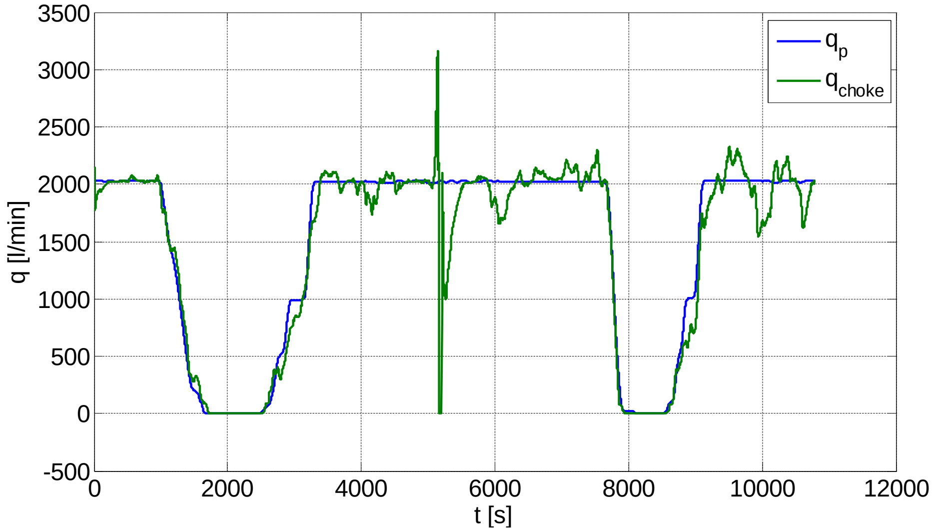

Figure 4.19: UKF with adaptation to flow gain K_c, annulus density ρ_a and annulus friction factor θ_1: Volume flow in pump (solid, blue) and choke (solid, green)

4.5.2 Applying Last Available Measurement

Since the unscented Kalman filter so heavily depends on consistent measurements, the less complex solution is to continue the filter with the last available measurement, as discussed in Section 3.2.1.1. Simulation of this particular approach provides the seemingly best results for the observer, but by further investigation this is a statement with modifications. However, it opens possibilities for some alternative tuning which will be discussed in this section.

4.5.2.1 No Adaptation

Figure 4.20 and 4.21 indicates how the estimate tracks the measurement of p_{bit} with very high precision. It clearly demonstrates how well the UKF performs with consistent measurements and a worthy model. However, by examining the estimate and measurement of p_{bit} closer (Figure 4.21)and remembering the update frequency of the latter, it becomes evident that much of the dynamics in the 20s periods between arrival of new measurements are lost. Also, the general worsened performance seen in the estimates of the states weakens the observers credibility. The poor performance in $\hat{p}_c$ and $\hat{p}_p$ is very similar to that of the moving horizon observer without any parameter adaptation, while the estimate of q_{bit} is very different from anything seen in other simulations with its rapid changes and oscillations.

Figure 4.23 reveals that during pipe connection the observer yields unique results. None of the other well tuned estimations carried out in this thesis manage to capture the oscillatory dynamics of the pipe connection this precise. If similar results are demanded for prior observers it will require tuning that will increase performance at the expense of accuracy during other parts of simulation which is not desirable. However, the results are included since they clearly indicate that phenomena during pipe connections can be reproduced with a simple model.

More specifically, the ratio between the covariances of the process noise and measurement noise are changed dramatically in magnitudes of up to 10^5 with far less uncertainty put on the model. Since a measurement for p_{bit} is available at all times, there is seemingly so much information in the system that the estimate of p_{bit} does not drift off during regular drilling, i.e. no pipe connection. Room is therefore left to rely more heavily on the model when this particular measurement is unavailable, allowing $\hat{q}_{bit}$ to change more freely.

Towards the end of the pipe connection the estimate seems to drift off to higher values, which is hard to explain intuitively. However, as this phenomena just appears in this particular section, one is lead to believe that it is connected to the increased model dependency. Landet [14] describes a similar phenomena where no immediate pressure increase is found in p_p after flow is initiated at the pump after a period of zero flow. For the particular result in this section this specific argument can not be used as the estimated pump pressure actually is lower than the measured pressure. However, it is expected that the deviation in $\hat{p}_{hat}$ can be traced to some model error related to initiation of flow from zero flow.

This section is only included as an interesting side result and is thus not explored further. The filter is implemented in a way that can cause unwanted behavior in other drilling scenarios when p_{bit} may be absent for even longer periods of time and the general performance does not increase reliability for the observer.

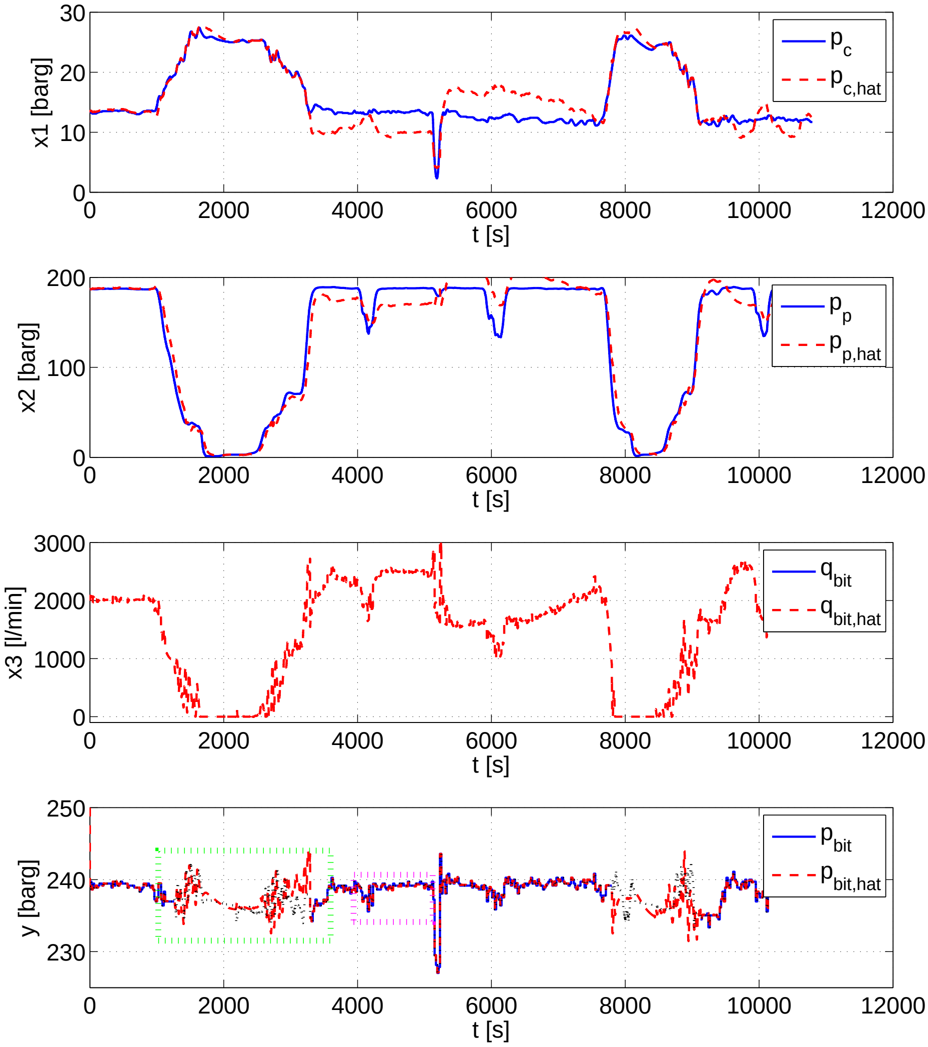

Figure 4.20: UKF without adaptation, using last measurement: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

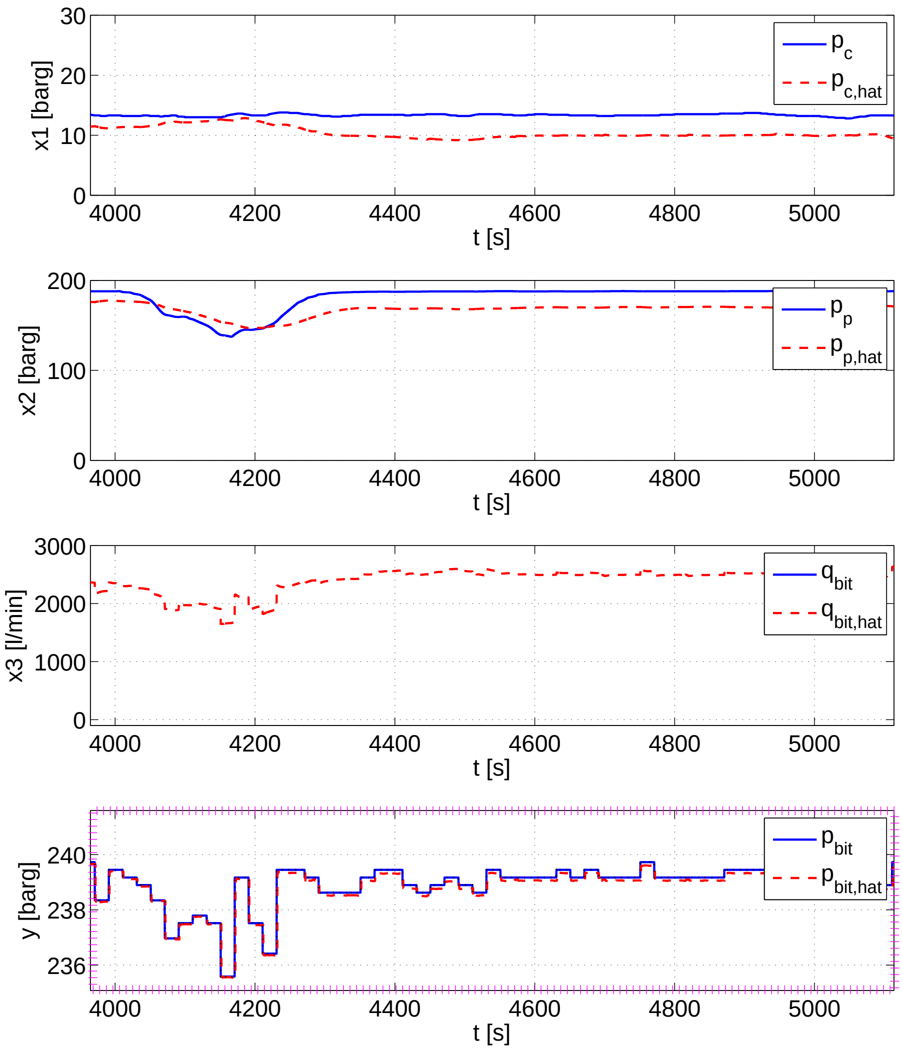

Figure 4.21: UKF without adaptation, using last measurement: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red)

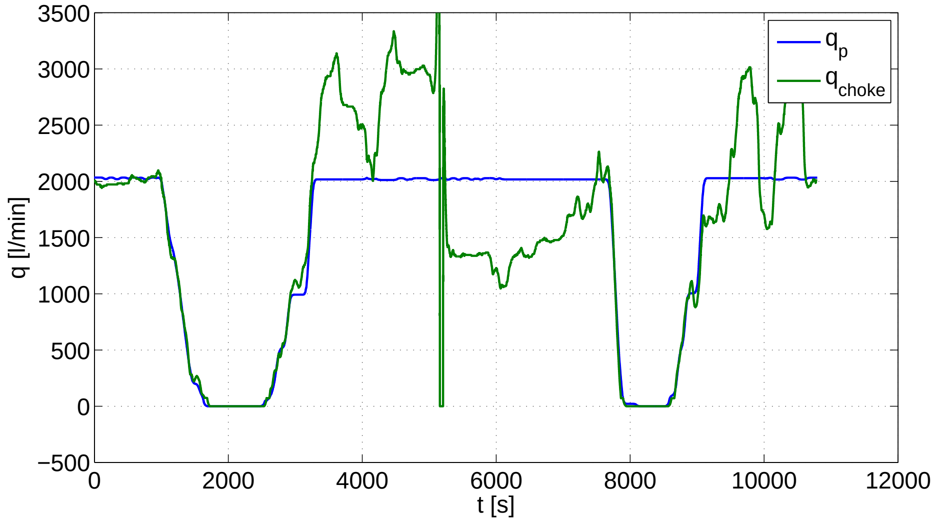

Figure 4.22: UKF without adaptation, using last measurement: Volume flow in pump (solid, blue) and choke (solid, green)

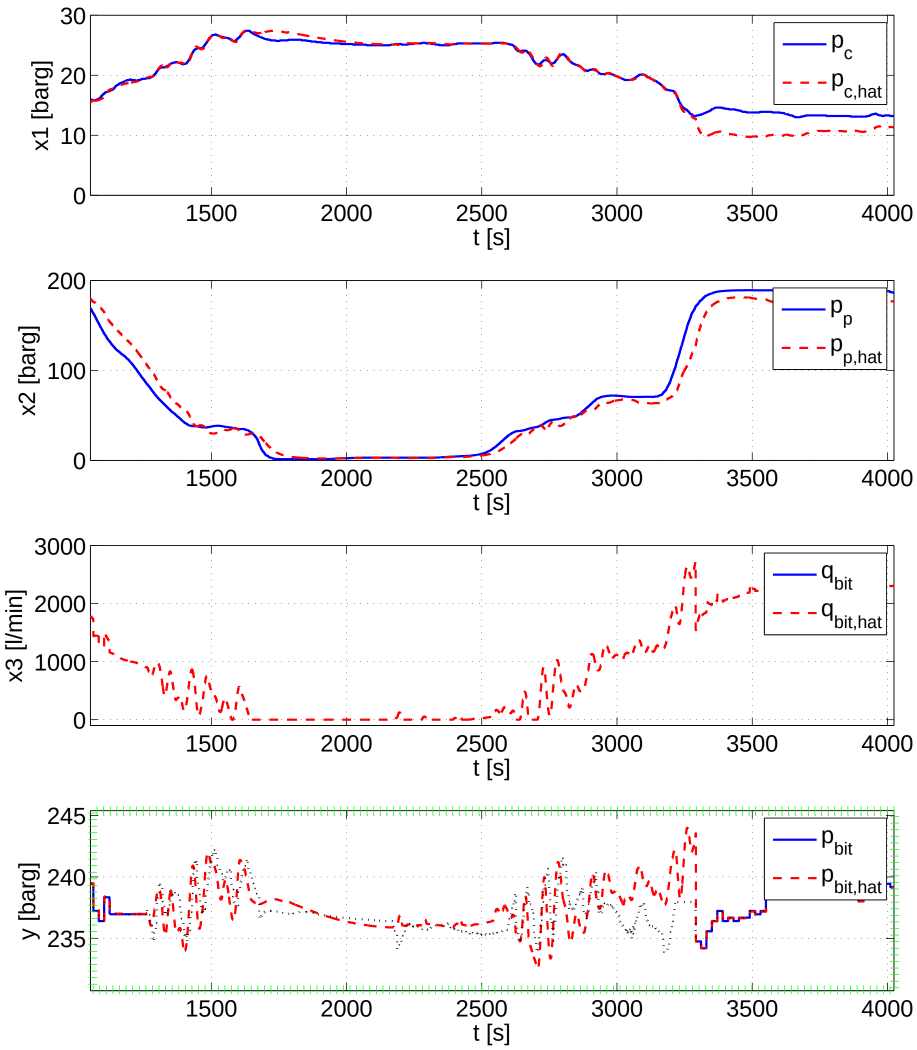

Figure 4.23: UKF without adaptation, using last measurement: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

4.6 Observer Combinations

To see if different combinations of the two observers can have a positive effect on the estimate of p_{bit}, further simulations are carried out. First, instead of calculating $\bar{x}_{t-N}$ with simulative propagation, prefiltering is applied. Second, the covariance calculated by the UKF during the prefiltering is utilized as state error weight matrix in the MHE. Last, UKF is used in two different ways inside the moving horizon estimation as a way to model process and measurement noise.

4.6.1 Prefiltering with UKF in MHE

Unfortunately, all predictions made in Section 3.3.1 are not confirmed by the results presented in this section. Examining Figure 4.24 reveals a very similar simulation to what has been seen before and in comparing it to Figure 4.4 it is hard to spot severe improvements in the estimate of p_{bit}. Overall performance may even be slightly worsened, but this can relate to poor tuning.

The only noticeable betterment occurs in $\hat{p}_p$ and $\hat{p}_c$ from $1800s - 2500s$ which only improves $\hat{p}_{bit}$ some, but enhances the observers liability. It may be that during zero flow in q_{bit} the prefiltering technique provides the optimization problem with an $\bar{x}_{t-N}$ that makes different, more fitting solutions feasible.

However, by closer inspection a significant improvement is found. Figure 4.26 plots the first $100s$ of the simulation scenario used in this thesis and clearly indicates that the prefiltering causes the estimate of p_{bit} to converge more rapidly. Consequently, it is probable that the prefiltering causes the moving horizon observer to be more robust to changes and sudden discrepancies as it will converge to an area around the true value of the pressure again faster. This observation also holds for the other estimates, but only Figure 4.26 is included. Estimation of K_c remains nearly unchanged (Figure 4.25).

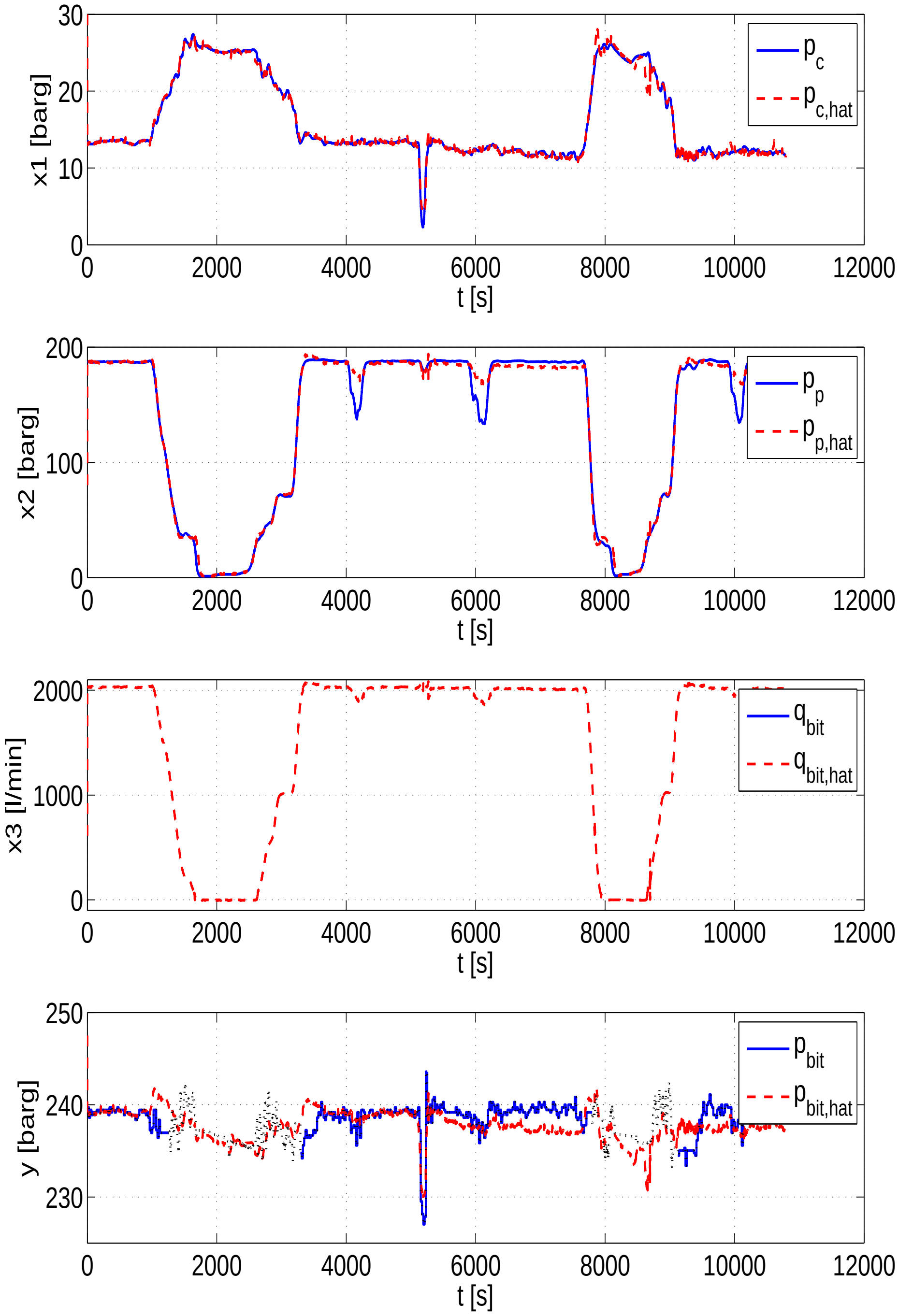

Figure 4.24: MHE with prefiltering and adaptation to flow gain K_c: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

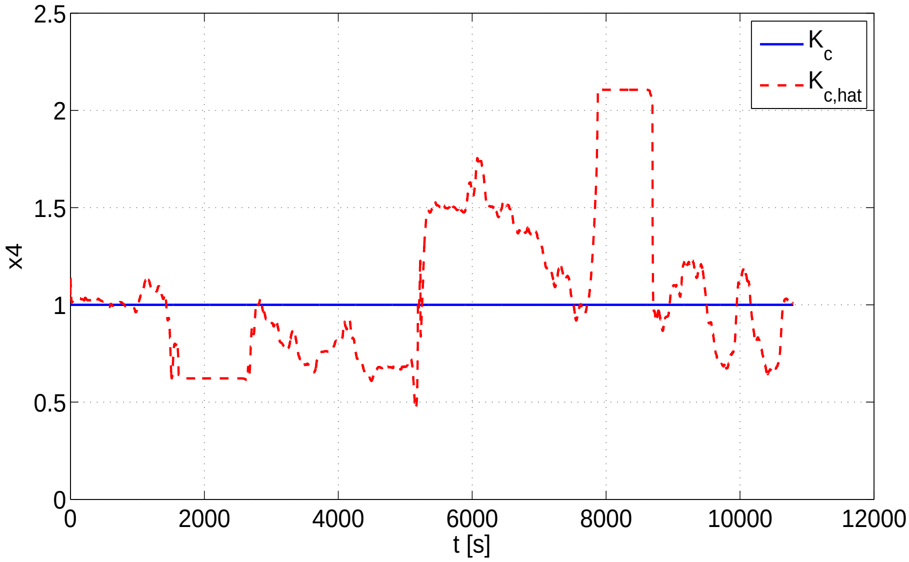

Figure 4.25: MHE with prefiltering and adaptation to K_c: Adapting K_c (dotted, red) and offline tuned K_c (solid blue)

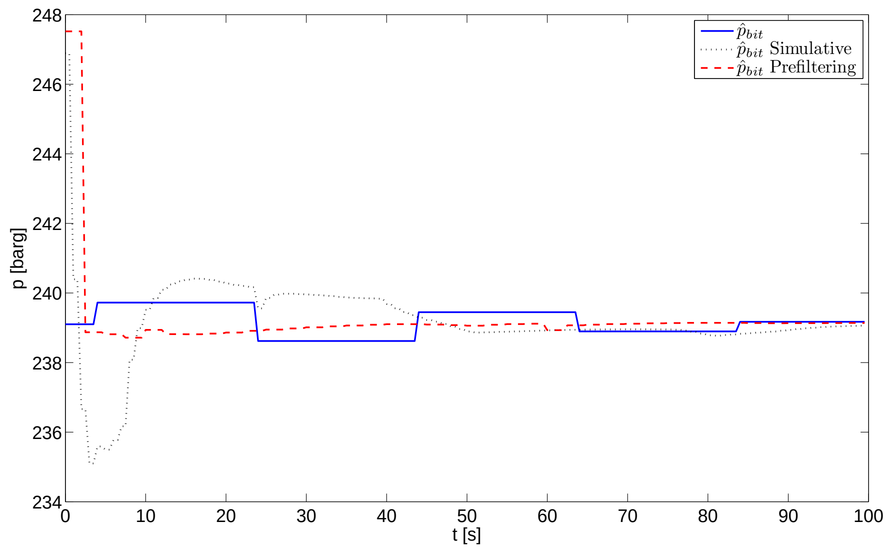

Figure 4.26: MHE with prefiltering and adaptation to flow gain K_c: Measured bottom hole pressure (solid, blue), estimate with prefiltering of $\bar{x}_{t-N}$ (dashed, red), estimate with simulative propagation (dotted, black)

4.6.2 Utilize Covariance Information

The weight suggested in Section 3.3.2 is easy to implement as the prefiltering already is present in the code. However, the results are disappointing as they do not improve measurements at any point of the simulation. Actually, performance is overall worsened as seen by comparison to the initial MHE presented in Section 4.3.2 (Figure 4.4).

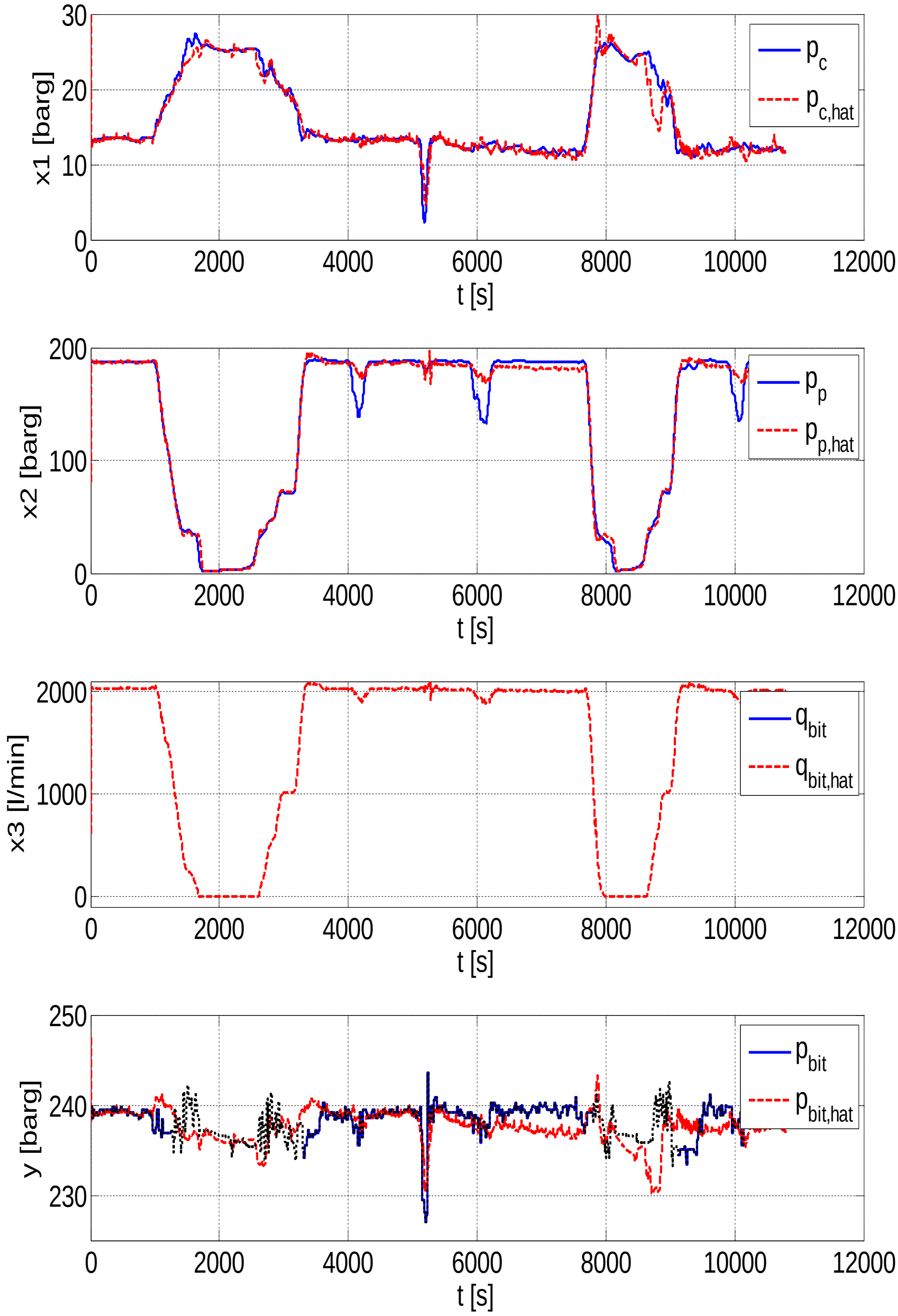

Figure 4.27: MHE with prefiltering and covariance weighting, with adaptation to flow gain K_c: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

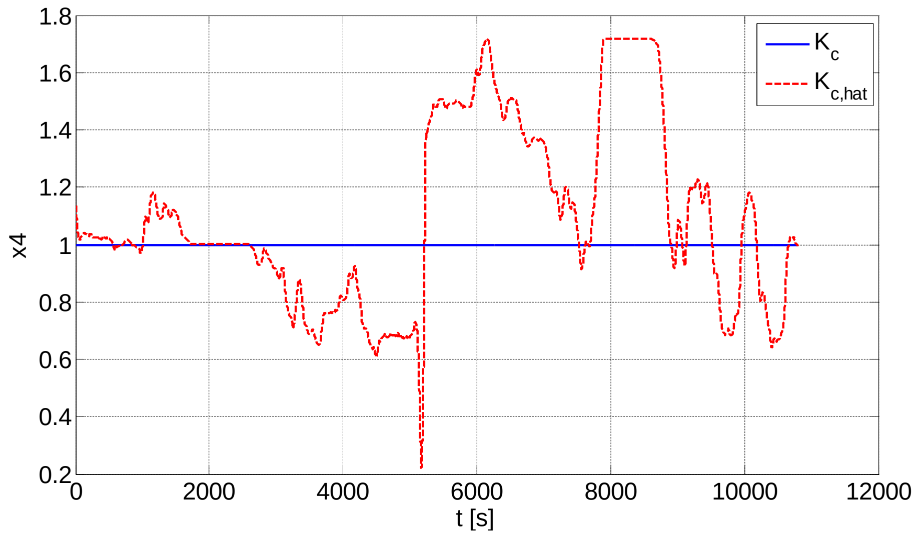

Figure 4.28: MHE with prefiltering and covariance weighting, with adaptation to flow gain K_c: Adapting K_c (dotted, red) and offline tuned K_c (solid blue)

4.6.3 UKF in MHE

Completely integrating the unscented Kalman filter in the moving horizon observer as discussed in Section 3.3.3 is by far the most complex combination of the two observers, which is reflected in the duration of the simulation (A time complexity analysis is presented in Section 4.7). Unfortunately, there is little correlation between performance and time complexity and the results are yet again not very distinguishable from prior results, with relatively good performance during pipe connections and satisfactory accuracy elsewhere, but slightly more oscillatory behavior overall is found. As it was briefly discussed in the project assignment, reaching a threshold for how well the MHE or any other observer can perform based on the simple Kaasa model is very likely and the several new and different approaches investigated in this thesis strengthen the suspicion.

The estimate of K_c seems to vary more than before, but this can be a tuning issue. As a consequence of the long simulation times for this particular approach, careful tuning was downgraded. It is also noticeable that for this combination the choke flow (Figure 4.31) acts as if K_c was not estimated.

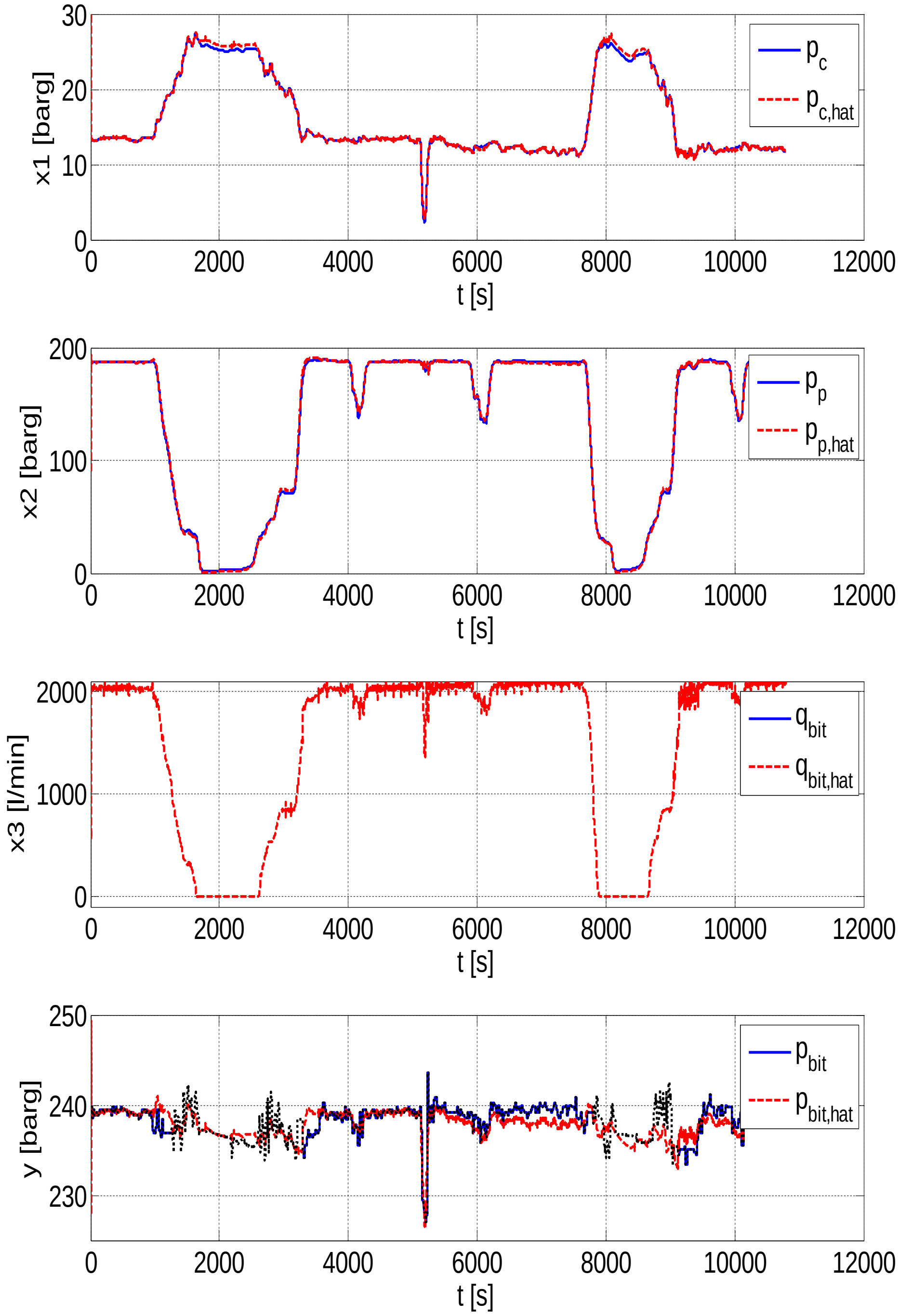

Figure 4.29: MHE combined with UKF, with adaptation to flow gain K_c: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

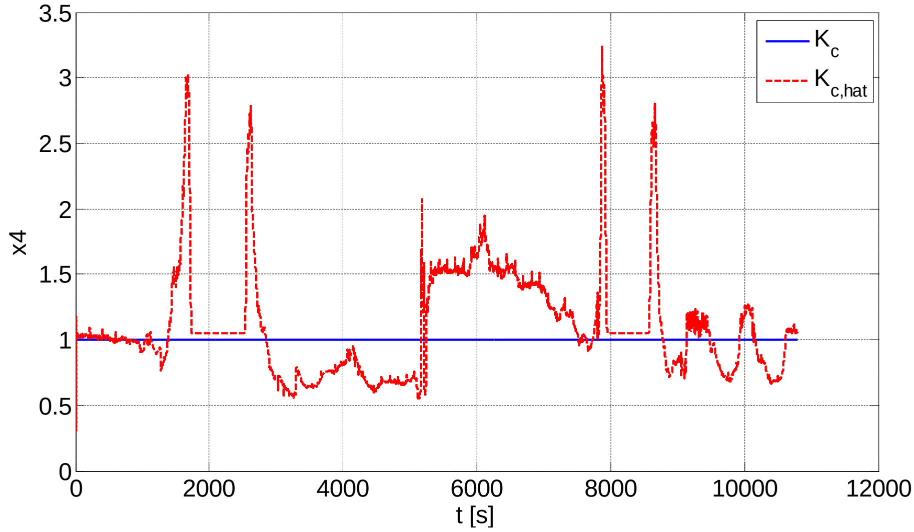

Figure 4.30: MHE combined with UKF, with adaptation to flow gainK_c: Adapting K_c (dotted, red) and offline tuned K_c (solid blue)

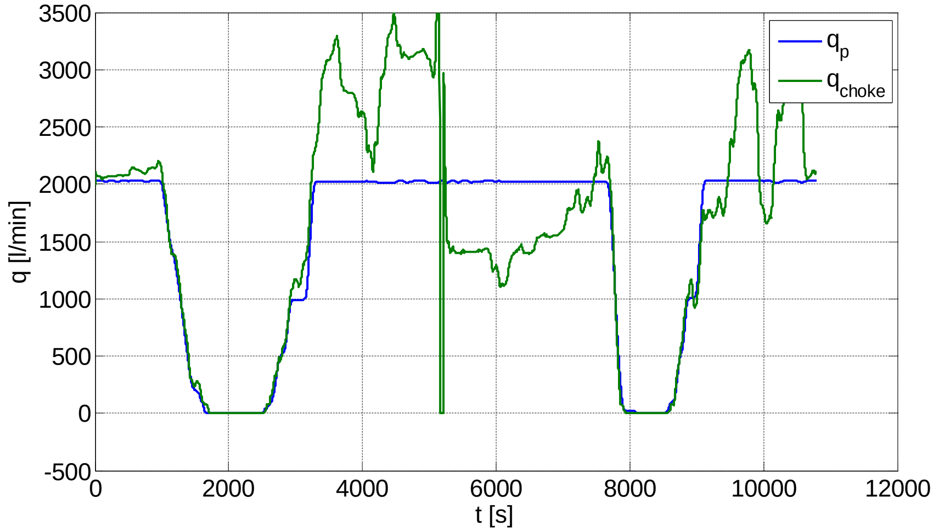

Figure 4.31: MHE combined with UKF, with adaptation to flow gain K_c: Volume flow in pump (solid, blue) and choke (solid, green)

4.6.4 Using UKF to Obtain $\hat{x}_t$ from $x^{\mathbf{o}}_{t-N,t}$

Figure (4.32) shows how the simple approach to incorporate noise in the moving horizon observer performs. Again, the results do not differ very much from previous ones, and bearing in mind the increased computational load, this observer does not exceed expectations neither. However, the results are important as they can close some of the many open doors and narrow the search for a better observer.

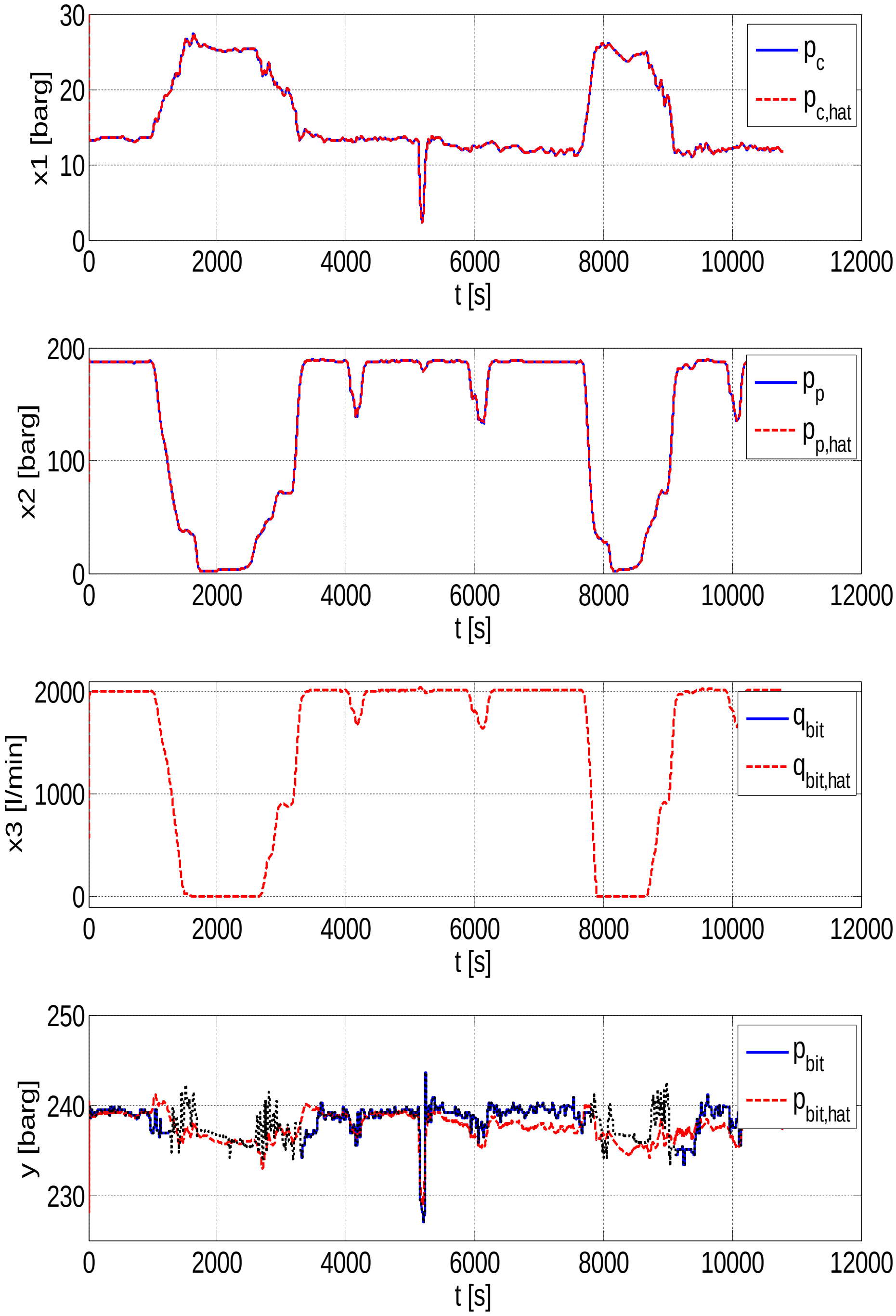

Figure 4.32: MHE, using UKF to Obtain $\hat{x}_t$ from $x^{o}_{t-N,t}$, with adaptation to flow gain K_c: Measured states and bottom hole pressure (solid, blue), estimates (dashed, red), logged measurements p_{bit} (dotted, black)

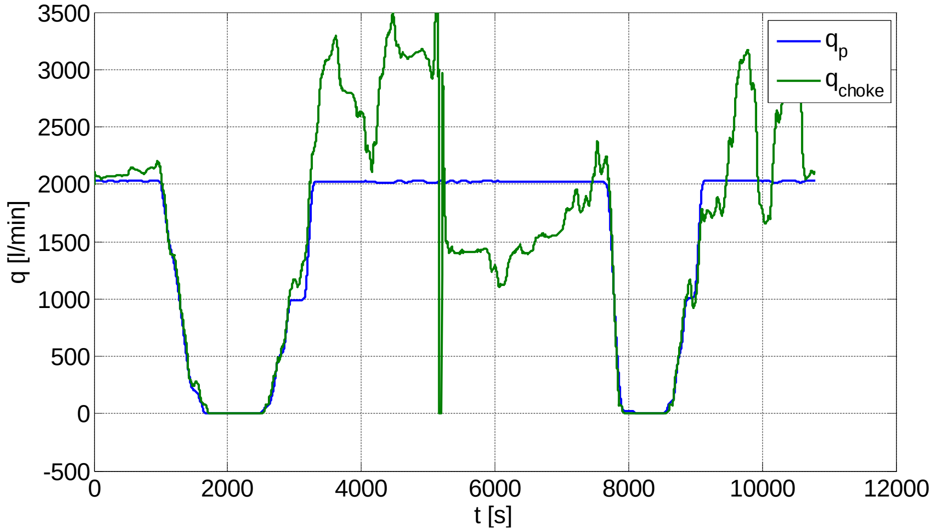

Figure 4.33: MHE, using UKF to Obtain $\hat{x}_t$ from $x^o_{t-N,t}$, with adaptation to flow gain K_c: Adapting K_c (dotted, red) and offline tuned K_c (solid blue)

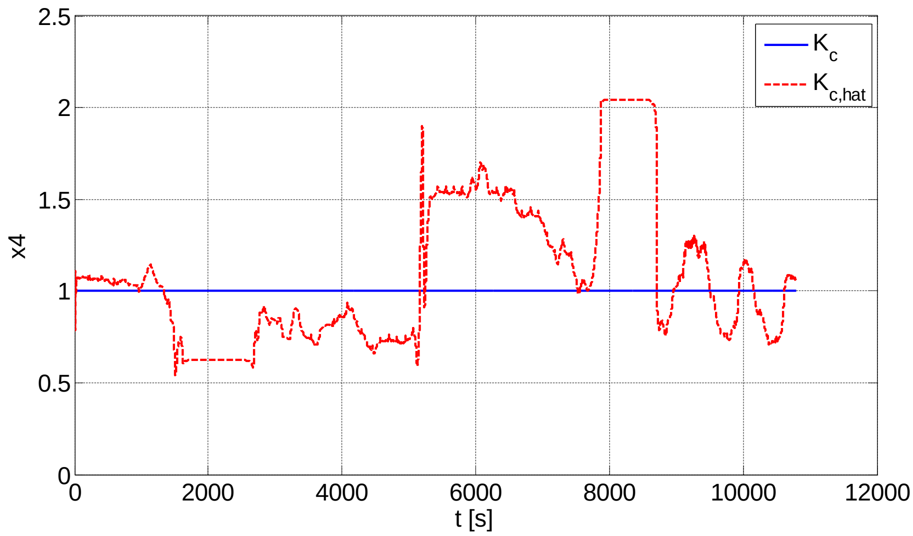

Figure 4.34: MHE, using UKF to Obtain $\hat{x}_t$ from $x^o_{t-N,t}$, with adaptation to flow gain K_c: Volume flow in pump (solid, blue) and choke (solid, green)

4.7 Comparison of Time Complexity

For near all systems that make use of an observer there are requirements on time complexity that has to be met. For the drilling scenario presented in this thesis estimation commence every time a new measurement of p_p and p_c is available, more precisely every second. Consequently, it is strongly desired that each iteration of the observer does not exceed $1s$. Of course, implementations and simulations performed in MATLAB have great potential for improvement, but analysis of time complexity still paint an important picture of how achievable it is to run the observer in real time.

Observer	Horizon	Duration/estimation
MHE	40	≈0.5s
MHE + K_c	40	≈0.7s
MHE + K_c, ρ_a and θ_1	40	≈1.5s
MHE + K_c + adaptive friction model	40	≈7s
UKF	-	≈0.04s
UKF + K_c	-	≈0.04s
UKF +K_c, ρ_a and θ_1	-	≈0.05s
UKF using last measurement	-	≈0.04s
MHE + K_c + prefiltration w/UKF	40	≈1s
MHE + K_c + covariance weighting	40	≈1.3s
MHE + K_c + integrated UKF	10	≈3.7s
MHE + K_c + iteration with UKF	40	≈4.7s

Table 4.2: Time Complexity for every estimation scheme, all with integration time step $\Delta t = 0.2$

Table 4.2 presents approximate values for simulations done on the entire $10795s$ long ($\sim$ 3 hours) drilling scenario provided by the Grane data. The numbers are taken from one single run and has to be considered as tentative and are only included for discussion purposes, especially for the MHE where weighting in the cost function influences computational load drastically. Also, what other tasks the computer performed during simulations affect the duration. The numbers in the table clearly demonstrates that the unscented Kalman filter is far superior to the moving horizon observer with regards to computational complexity. Whereas several of the simulations with a variant of the moving horizon observer exceed an average duration time per iteration of $1s$, the unscented Kalman filter is not even close to this limit. In particular, the moving horizon observers with no adaptation and adaptation to K_c are well within the requirement, while adding adaptation to ρ_a and θ_1 demands just a bit too much computational power. However, implementation on a lower level is likely to increase the efficiency noticeably so minor exceedings are tolerable.

Chapter 5

Conclusion and Future Work

5.1 Conclusion and Future Work

To conclude, several aspects of bottom hole pressure estimation have been discussed and tested in this master's thesis. First, it has been confirmed that work in the author's project assignment "Estimation of Bottom hole Pressure During Drilling using Parameter Adaptation and Modeled Process Noise" and work done by Marcel et al. ([19]) provide good estimates, and that the regularized nonlinear moving horizon observer has a unique ability to adapt to uncertain parameters in the model.

Second, an alternative friction model has been presented, using a basis function technique, and simulations have been carried out. This attempt turned out as a complete failure in that it was impossible to get satisfactory estimates, but it was not unexpected as similar work has been indicating the same.

Third, and most important, it has been shown that the unscented Kalman filter is a simple and accurate observer that is just as suitable as the moving horizon observer, and by acknowledging that much work point to the unscented Kalman filter outperforming the extended Kalman filter, the UKF is a good candidate for bottom hole estimation. It was anticipated that the unscented Kalman filter would have minor observability issues when several parameters was adapted and simulations have proven this to be correct. However, the importance of being able to estimate parameters online is considerable so this is an area for future work. The unscented Kalman filter performed particularly good during regular drilling, but as expected, it encountered the same problems as the moving horizon observer during pipe connections. Still, the largest deviations in estimates are found during stopping and re-initiation of the mud flow where oscillatory behavior that may be unmodeled, or impossible to capture with good overall tuning, is observed. In this regard, it was shown that by letting the UKF use the last available measurement of p_{bit} for estimation in the $20s$ gap between each update, alternative tuning could be applied and the oscillatory behavior could be captured.

Last, several combinations of the two observers were tried out, but unfortunately none of them provided particularly good results: By prefiltering $\bar{x}_{t-N}$ used in the moving horizon observer with a 1-step unscented Kalman filter instead of ordinary simulative propagation, estimates were shown to converge faster to an area around the measurements. Utilizing the covariance information provided by the 1-step UKF as weight in the moving horizon observer had shown good results in other work, but did not enhance performance for simulations in this thesis. Combining the two observers completely, using the unscented Kalman filter instead of model iteration inside the optimization problem solved by the moving horizon observer, only proved to be computationally demanding without any improvements in measurements. A simpler approach where the optimization problem was carried out as normal, but the iteration from an optimal estimate N steps ago to a current estimate was performed by the UKF, did not provide any refreshing neither. It is evident that the unscented Kalman filter is a competent candidate for estimation of the bottom hole pressure during drilling.

There is of course much work yet to be done, other than the already mentioned, with regard to the problems presented in this thesis. It may seem as though improvements to the simple Kaasa model used is needed to further enhance performance of any observer applied. Also, there is currently some activity on suitable controllers for the bottom hole pressure for somewhat higher order models that require an sufficiently accurate estimator, and it would be interesting to try to pair the two, especially since all observers tested in this thesis adapt well to the choke flow K_c, which gives a more accurate choke characteristic for the controller. With more time in hand, initial tests and simulations could have been performed, but the author is confident that future candidates and researchers will complete this task.

Bibliography

[1] Jr. Arthur E. Bryson and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation and Control.* Taylor I& Francis.

[2] Sui Dan and Tor A. Johansen. Moving horizon observer with regularization for detectable systems without persistence of excitation. 2010.

[3] Moritz Diehl, Hans Joachim Ferreau, and Nils Haverbeke. Effiecient numerical methods for nonlinear mpc and moving horizon estimation. 2009.

[4] J. E. Gravdal, R. J. Lorentzen, K. K. Fjelde, and E. H. Vefring. Tuning of computer model parameters in managed-pressure drilling applications using an unscented-kalman-filter technique. 2005.

[5] SPT Group. Olga. `http://www.sptgroup.com/en/Products/olga/Multiphase-Flow-Simulator/`, December 2010. [cited 19 Dec 2010].

[6] IEA. Share of total primary energy supply in 2008. `http://www.iea.org/stats/pdf_graphs/29TPESPI.pdf`. [cited June 8th, 2011].

[7] Lars Imsland. Modeling of drilling hydraulics for adaptive pressure estimation. 2008.

[8] Simon J. Julier and Jeffrey K. Uhlmann. A general method for approximating nonlinear transformations of probability distributions. 1996.

[9] Simon J. Julier and Jeffrey K. Uhlmann. Unscented filtering and nonlinear estimation. 2004.

[10] Simon J. Julier, Jeffrey K. Uhlmann, and H.F. Durrant-Whyte. A new approach for the nonlinear transformation of means and covariances in linear filters. 1996.

[11] G. O. Kaasa. A simple dynamic model of drilling for control. 2007.

[12] Rambabu Kandepu, Bjarne Foss, and Lars Imsland. Applying the unscented kalman filter for nonlinear state estimation. 2007.

[13] Hassan K. Khalil. *Nonlinear Systems Third Edition*. Prentice Hall.

[14] Ingar Landet. Advanced modeling for managed pressure drilling, 2010.

[15] J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education Limited.

[16] G. Nygaard, L. Imsland, and E. A. Johannessen. Using nmpc based on a low-order model for controlling pressure during oil well drilling. 2007.

[17] G. H. Nygaard, E. Johannessen, and F. Iversen J. E. Gravdal. Automatic coordinated control of pump rates and choke valve for compensating pressure fluctuations during surge-and-swab operations. 2007.

[18] TOMLAB Optimization. Tomlab. `http://tomopt.com/tomlab/`.

[19] Marcel Paasche. Nonlinear moving horizon estimation of bottomhole pressure during oil drilling, 2011. [not yet published, June 8th, 2011].

[20] Marcel Paasche, Tor A. Johansen, and Lars Imsland. Estimation of bottomhole pressure during oil well drilling using regularized adaptive nonlinear moving horizon observer. 2010.

[21] Christopher V. Rao, James B. Rawlings, and David Q. Mayne. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. 2003.

[22] C. E. Shannon. The mathemathical theory of communication. 1948.

[23] Oyvind Nistad Stamnes. Adaptive observer for bottomhole pressure during drilling, 2007.

[24] Oyvind Nistad Stamnes, Jing Zhou, Glenn-Ole Kaasa, and Ole Morten Aamo. Adaptive observer design for the bottomhole pressure of a managed pressure drilling system. 2008.

[25] Bard Arve Valstad. Parameter estimation and control of a dual gradient managed pressure drilling system, 2009.

[26] Rudolph van der Merwe. Sigma-point kalman filters for probabilistic inference in dynamic state-space models. 2004.

[27] Liu Xuanchao and Cheng Guojian. The research of information and power transmission property for intelligent drillstring. 2010.

Appendix A

MMSE Derivation of Kalman Filter

The estimation error is defined as

$$\tilde{x} = x_k - \hat{x}_k \ . \tag{A.1}$$

Substituting in the estimator

$$\hat{x}_k = \hat{x}_k^- + K_k \tilde{y}_k \tag{A.2}$$

where K_k is the Kalman gain and $\tilde{y}$ is the error between the measurement and its prediction, i.e.

$$\tilde{y} = y_k - \hat{y}_k^- \tag{A.3}$$

yields

$$\tilde{x}_k = \tilde{x}_k^- - K_k(y_k - \hat{y}_k^-) \tag{A.4}$$

where

$$\tilde{x}_k = x_k - \hat{x}_k^- \ . \tag{A.5}$$

Here, the fact that $E[\tilde{y}_k] = 0$ is utilized under the assumption of an unbiased estimator.

The covariance of the update is found by taking outer products and expectations of the update (A.2) resulting in

$$P_{x_k} = P_{x_k}^- - P_{x_k \tilde{y}_k} K_k^T - K_k P_{\tilde{y}_k x_k} + K_k P_{\tilde{y}_k \tilde{y}_k} K_k^T \tag{A.6}$$

where the error covariance P_{x_k} is defined as

$$P_{x_k} \doteq E[\tilde{x}_k \tilde{x}_k^T] = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] \tag{A.7}$$

and the cross covariance $P_{x_k\tilde{y}_k}$ as

$$P_{x_k\tilde{y}_k} \doteq E[\tilde{x}_k^- \tilde{y}_k^T] = E[(x_k - \hat{x}_k^-)(y_k - \hat{y}_k^-)^T] \ . \tag{A.8}$$

Further, minimizing the expected value of the magnitude of the estimation error is the same as minimizing the trace of the covariance matrix, i.e

$$\begin{aligned} \frac{\partial tr(P_{x_k})}{\partial K_k} &= 0 - P_{x_k\tilde{y}_k} - P_{\tilde{y}_k x_k} + 2K_k P_{\tilde{y}_k\tilde{y}_k} && \text{(A.9)} \\ K_k &= P_{x_k\tilde{y}_k}(P_{\tilde{y}_k\tilde{y}_k})^{-1} && \text{(A.10)} \end{aligned}$$

Last, substituting (A.10) into (A.6) gives

$$\begin{aligned} \hat{x}_k &= \hat{x}_k^- - K_k(y_k - \hat{y}_k^-) && \text{(A.11)} \\ P_{x_k} &= P_{x_k}^- - K_k P_{\tilde{y}_k\tilde{y}_k} K_k^T && \text{(A.12)} \end{aligned}$$

Appendix B

MATLAB Code

B.1 Unscented Kalman Filter

```
11-05-11 GUNI UKF + Kc estimation commence when measurement
   arrives
function [ ukf transfercache ] =...
    ODobserver_UKF( y, u, ukf, transfercache )
    %% Get variables from memory

    % Dimensions
    nx = transfercache.model.nx;
    ny = transfercache.model.ny;
    np = transfercache.model.np;
    nv = ukf.nv;
    nn = ukf.nn;
    N  = nx + np + nv + nn;

    transfercache.Nmemory(1) = [];
    transfercache.Nmemory(end+1).u = u;

    % Integration step length
    T_delta = transfercache.T_delta;

    % Get UKF variables from memory
    alpha    = ukf.alpha;
    beta     = ukf.beta;
    kappa    = ukf.kappa;
    lambda   = ukf.lambda;
    gamma    = ukf.gamma;

    % Get noise from memory
    v        = ukf.v;
    n        = ukf.n;
```

```

    % Get additional variables
    X_a      = ukf.X_a;
    P_x      = ukf.P_x;
    P_v      = ukf.P_v;
    P_n      = ukf.P_n;
    w_m      = ukf.w_m;
    w_c      = ukf.w_c;
    x_hat    = ukf.x_hat;
    xa_hat   = [x_hat; v; n];
    P_a      = blkdiag(P_x, P_v, P_n);
    P_a      = 0.5*(P_a+P_a');
    S        = sqrtm(P_a);
    Y_k      = zeros(ny,2*N+1);

    % Obtain 1/step length
    Mcycle = size(transfercache.Nmemory(end-1).u,2);

    % Allocation
    y_k_     = zeros(ny,1);
    P_ky     = zeros(ny,ny);
    P_kxy    = zeros(nx+np,ny);
    x_k_     = zeros(nx+np,1);
    P_kx_    = zeros(nx+np,nx+np);

    % if checks if new measurement arrives
    if transfercache.memory.flag_NM(1)
    %% For k = 1,2,....,inf Calculate 2N+1 sigma points
    for k = 1:2*N+1
        if k==1
            X_a(:,k) = xa_hat;
        else if k <= N+1
            X_a(:,k) = xa_hat + gamma*S(:,k-1);
            else
                X_a(:,k) = xa_hat - gamma*S(:,k-(N+1));
            end
        end
    end

    % Divide into respective matrices
    X_x = X_a(1:(nx+np),:);
    X_v = X_a((nx+np)+1:(nx+np)+nv,:);
    X_n = X_a((nx+np)+nv+1:(nx+np)+nv+nn,:);

    %% Time-update equatendions and set up weights
    for k = 1:2*N+1
        for j=1:Mcycle
            X_x(:,k) = ODmodel(X_x(:,k), ...
                transfercache.Nmemory(end-1).u(:,j),...
```

```
                  X_v(:,k), X_n(:,k), transfercache);
        end
        if k == 1
            w_m(1,k) = lambda/(N+lambda);
            w_c(1,k) = lambda/(N+lambda)...
            + (1 - alpha^2 + beta);
        else
            w_m(1,k) = 1/(2*(N+lambda));
            w_c(1,k) = w_m(1,k);
        end
    end

    %% Calculating apriori state estimate and apriori covariance
    for k=1:2*N+1
        x_k_  = x_k_ + w_m(1,k)*X_x(:,k);
    end
    for k=1:2*N+1
        P_kx_ = P_kx_ ...
            + w_c(1,k)*(X_x(:,k)-x_k_)*(X_x(:,k)-x_k_)';
    end

    %% Measurement-update equations:
    for k = 1:2*N+1
        Y_k(:,k) ...
            = ODcalch(X_x(:,k), X_n(:,k), transfercache);
    end

    %% Calculating mean and covariance of measurement vector

    % Mean of measurement vector
    for k=1:2*N+1
        y_k_   = y_k_ + w_m(1,k)*Y_k(:,k);
    end

    % Covariance for measurement vector and cross variance
    for k=1:2*N+1
        P_ky  = P_ky + w_c(1,k)*...
            (Y_k(:,k) - y_k_)*(Y_k(:,k) - y_k_)';
        P_kxy = P_kxy + w_c(1,k)*...
            (X_x(:,k) - x_k_)*(Y_k(:,k) - y_k_)';
    end

    % Calculating Kalman gain
    K = P_kxy/P_ky;

    % Use a priori estimates to calculate missing measurement
```

```
    if isnan(y(3))...
            || (transfercache.memory.dT_Count(3) ~= 1)
        y_bar    = ...
            ODcalch([y(1:2);x_k_(3)],n,transfercache);
        y(3)     = y_bar(3);
    else
        transfercache.memory.dT_Count(3) = ...
            transfercache.memory.dT_Count(3) + 1;
    end

    % UKF estimate and covarians
    x_hat    = x_k_ + K*(y - y_k_);

    % Make sure that qbit isn't negative
    if x_hat(3) < 0
        x_hat(3) = 0;
    end

   % Calculate covariance matrix
   P_x      = P_kx_ - K*P_ky*K';

   % Save variables
   ukf.x_hat    = x_hat;
   ukf.P_x      = P_x;
   ukf.y_hat    = y_k_;

   end

end
```

B.2 Friction Modeling

```
function [Fr_a Fr_d transfercache] = ...
    ODfrictionFunction(q_bit, transfercache)

    % Get variables from memory
    a        = transfercache.model.theta1;
    d        = transfercache.model.theta2;
    N        = transfercache.friction.N;
    c        = transfercache.friction.c;
    u        = transfercache.friction.u;
    theta    = transfercache.friction.theta;

    % Define model of friction
    F_a      = a*q_bit;
    F_d      = d*abs(q_bit)*q_bit;

    % Initialization
    sum_w    = 0;

    % Calculate Omega
    for j=1:N
        if abs(q_bit - c(j)) < u
            w(j,1) = (1 - ((abs(q_bit-c(j))/u)));
        else
            w(j,1) = 0;
        end
        sum_w = sum_w + w(j,1);
    end

    % Calculate Phi
    for k=1:N
        phi(k,1) = w(k,1)./sum_w;
    end

    % Calculate frictions
    f_a = phi*F_a';
    f_d = phi*F_d';
    Fr_a = f_a'*theta;
    Fr_d = f_d'*theta;

%     % Use regular model
%     Fr_a = F_a;
    Fr_d = F_d;
```