
**Modelling and Optimization of Recirculated Gas Lift
Problem**

TKP4580 - Chemical Engineering, Specialization Project

Author: Kristian Ødegård

Supervisor: Sigurd Skogestad, IKP

Co-supervisor: Risvan Dirza, IKP

December 23, 2022

Abstract

The goal of most producers is to conduct optimal production, while keeping costs at a minimum. This work considers the problem of a recirculated gas lift oil production system, where the source of the gas lift is supplied by a part of the total produced gas. Unlike a variety of previous studies (Krishnamoorthy et al. (2016)^[1] to name a few), where the upstream gas lift pressure is considered constant, this paper will consider this pressure to be dependent on both the separator pressure and the performance of the gas lift compressors. The “recirculated” gas system are found in a large number of oil fields, and is consequently considered to be a more realistic case study.

This paper will present an integrated model consisting of a gas lift well system, a riser system, a separator system and a re-injection compressor train. There have previously been developed various models describing the dynamic behaviour of these systems. This work will consider, among others, the nonlinear well model of Eikrem et al. (2008)^[2], the gravitational separator model of Backi et al. (2018)^[3] and the compressor model of Greitzer (1976)^[4].

The presented model was developed systematically from the mass and energy balances for the different systems. The necessary relations between the entities were also defined. The resulting differential-algebraic equations were solved in Python with the backward differentiation based integrator IDAS from the CasADI framework^[5]. Furthermore, measures were implemented to ensure convergence and reasonable dynamic relationships between the model units. The system was further continuously evaluated for stability and eventual issues were addressed. The objective of the optimization problem of this case study is to maximize oil production while minimizing associated costs, such as energy- and treatment costs. The Interior point optimizer (IPOPT) was implemented to solve the resulting non-linear programming (NLP) problem, and the model was manipulated to operate closer to the optimal values.

In this study we investigate the possibility of modelling a recirculated gas lift oil production system for further implementation of control strategies. Based on this study it is possible to develop the model with manipulated variables that can be used for further implementation. However, the model is non-linear and depends on well suited initial guesses to solve the system of equations. The optimal operation conditions will also be sensitive to small deviations in a process parameter, and the NLP solver will struggle to find the the solution to the optimization problem. Due to the assumptions and approximations applied in the development of this model, studies conducting comparisons between this model and real systems are recommended for validation and improvement of the model.

Contents

1	Introduction	1
2	Theory	2
2.1	Principles of oil production	2
2.2	CasADI	3
	Initial value problems	4
	Nonlinear programming	4
2.3	PID controller	6
2.4	SIMC tuning	7
3	Modelling	10
3.1	Model description	10
3.2	Well system	10
	Mass balances	11
	Pressure	12
	Flow	13
	Density	15
	Equations and states	16
3.3	Manifold and Riser	16
	Mass balances	17
	Pressure	17
	Flow	18
	Density	19
	Equations and states	19
3.4	Separator	19
	Pressure	20
	Flow	21
	Level and volume	21
	Density	22
	Equations and states	22
3.5	Compressor train	23
	Pressure	23
	Flow	24
	Density	25
	Pressure ratio	25
	Polytropic head	26
	Power and efficiency	26
	Equations and states	26
3.6	Gas lift	27
	Mass balances	27
	Pressure	28
	Flow	28
	Density	28
	Equations and states	28
4	Model implementation	29

4.1	Modelling strategy	29
	General modelling procedure	29
	Well system	30
	Riser system	30
	Separator system	30
	Compressor system	31
	Gas lift system	31
4.2	Integrated model	31
4.3	Optimization	32
5	Results	34
5.1	Implementation	34
5.2	Integration	36
	Level control	36
	Model verification	37
5.3	Optimization	42
6	Conclusion	44
7	Further work	45
	References	46
A	Constant parameters	
B	Initial Values and Boundary Conditions	
B.1	Dynamic states(x)	
B.2	Controlled variables(u)	
B.3	Algebraic states(z)	
C	Python-code	
C.1	Parameter file	
C.2	Simulator file	
C.3	Main file	
D	Integration results	

List of Figures

2.1	Step response of first-order plus time delay process ^[27]	8
2.2	Open loop step response on integrating process ^[28]	9
3.1	Sketch of the total model flow-sheet.	10
3.2	Sketch of the simplified gas lift well system.	11
3.3	Sketch of the simplified well system with manifold and riser.	17
3.4	Sketch of the simplified Separator.	19
3.5	Sketch of the simplified Compressor train.	23
3.6	Sketch of the simplified gas lift system.	27
5.1	Equation 3.59 plotted for different rotational speeds against mass flow through the compressor.	34
5.2	Equation 3.61 plotted for different rotational speeds against mass flow through the compressor.	35
5.3	Equation 3.62 plotted for different rotational speeds against mass flow through the compressor.	36
5.4	Height of oil in separator(h_{ls}).	36
5.5	Valve opening(u_{ov}).	36
5.6	Height of oil in separator(h_{ls}).	37
5.7	Valve opening(u_{ov}).	37
5.8	Pressure regions in the total system.	38
5.9	The mass flow of from well 1 plotted against the gas lift choke valve opening. . .	39
5.10	Bottom-hole pressure well 1 plotted against the gas lift choke valve opening. . . .	40
5.11	Power compressors.	40
5.12	Polytropic efficiency compressors.	40
5.13	Pressure ratio compressors.	41
5.14	Polytropic head compressors.	41
5.15	Constraint regions related to the GOR values of each well. The black lines indicate the region of feasibility, while the red dotted lines indicate change of region. . . .	43
D.1	In Appendix D the integration results for the system states and the controlled valve opening at the oil outlet of the separator are presented. The system is solved with IDAS explained in Section 2.2.1, with a simulation time of $t = 10000s$.	

List of Tables

4.1	Tuning parameters separator level controller.	30
4.3	polynomial constants representing the pressure ratio and efficiency of a compressor. . .	31
4.5	Dynamical coefficients of the compressor system.	31
5.1	Optimal valve openings (0-1).	42
A.1	Constant parameters for the model.	
A.3	Constant parameters for the 6 different wells.	
B.1	Initial values, lower and upper boundaries for the differential states(x).	
B.3	Initial values, lower and upper boundaries for the control variables(u).	
B.5	Initial values, lower and upper boundaries for the algebraic states(z).	

Abbreviations

AD	Algorithmic differentiation
BDF	Backward differentiation formula
CAS	Computer-algebra system
DAE	Differential-algebraic equation
GOR	Gas-oil ratio
IPOPT	Interior point optimizer
IVP	Initial value problem
KKT	Karush–Kuhn–Tucker
NLP	Non-linear programming
ODE	Ordinary-differential equation
PID	proportional-integralderivative
SIMC	Skogestad Internal method control

1 Introduction

Optimal production at the lowest possible cost is the objective for most oil and gas companies. Usually, mathematical models are used when optimizing production system performance. The use of mathematical models can cause great uncertainties in the solution, due to the simplifications of the model, and the lack of reliable data. This is an inherent problem in production optimization because of the complex nature of the production fluids, and the large number of processing units^[1]. Thus, the construction of realistic models are essential for reliable optimization results and optimal production.

This work considers the modelling and optimization of a recirculated gas lift oil production system. The modelling aims to construct an integrated model, based on several separate oil production systems, including a gas lift well system, a riser system, a separator and a re-injection compressor train. Furthermore, the model will implement re-circulation of gas for use as gas lift. This results in a gas lift system that is dependent on the gas pressure in the separator and the compressor performance. The model is developed for further implementation of different control strategies, like primal-dual based control and regional based control^[6]. These methods are based on the principle of moving the optimization problem to the control layer proposed by Moriari et al. (1980)^[7]. The methods will then be compared with the NLP solver used in this paper in the continued work of a mater thesis.

This paper will systematically explain the mathematical background of the different subsystems, as well as the related assumptions and simplifications. Furthermore, a general method for the modelling will be described, as well as implementations of the separate subsystems. The model will be developed from mass and energy balances relating to the different subsystems and methods developed in previous work. Several dynamic models have been developed for the subsystems, Eikrem et al.(2008)^[2] among others^[1], proposed a non-linear dynamic well model for the control of instabilities in gas lift wells. A dynamic model of the riser system was developed by Jahanshahi et al.(2011)^{[8][9]}, who proposed a dynamical model of multi-phase flow in risers. For the separator system Backi et al. (2018)^{[3][10]}, proposed a gravity based separator for three-phase separation. The dynamic compressor model derived by Greitzer (1976)^{[4][11]}, proposes a non-linear model for axial-compressor flow.

From this point forward, the paper will be organized as following: Section 2 includes general theory, Section 3 includes the mathematical background and implementation of the model, Section 5 includes results and discussion of the simulation and optimization of the model, Section 6 includes the conclusion and Section 7 includes the proposed further work.

2 Theory

2.1 Principles of oil production

The most important part of a oil production system is the reservoir. Reservoirs are classified as either water-drive reservoirs, gas-cap drive reservoirs or dissolved gas drive reservoirs. In water-drive reservoirs, ground water expands and flows into the reservoirs when pressure is reduced, which forces oil and gas to the top of the reservoir. If the production rate is kept stable, this type of reservoir will maintain its pressure for a longer period of time than other types of reservoirs. In gas-cap drive reservoirs, the gas escapes the oil/gas solution and rises to the top of the reservoir. If the gas in the cap(top of the reservoir) is produced too fast, the reservoir pressure will drop significantly, thus reducing the production potential. For dissolved gas drive reservoirs, the gas dissolves in the oil and remains in liquid phase. It is often necessary to perform early pressure maintenance in the reservoir due to the potential of formation of a two-phase flow resulting from pressure drop^[12].

To be able to access the oil and gas inside petroleum reservoirs, wells need to be drilled to create extraction passages. These production wells consist of packers, a production pipe (tubing), casings, and a wellhead containing numerous chokes. The packers isolate the annulus at the bottom of the tubing, forcing the produced fluid to escape the perforation, and into the bottom of the well. The tubing transports the oil and gas to the surface. The casings are pipes that support the structure. The wellhead consists of various chokes, which are implemented to control the flow of the well. The main choke for flow control is defined as the production choke, which can be manipulated to alter the flow. By closing down a production choke, the bottom-hole flowing pressure increases, thus reducing the pressure difference between reservoir and the bottom-hole of the well. This results in a decrease in production rate. Wells can be classified by their Gas-oil ratio (GOR), which quantifies the relationship between oil and gas in the production fluid. Another way to classify wells is to examine their productivity index. This index formulates a relationship between the flow of the liquid in the well and the pressure difference between the reservoir and the bottom-hole^[12]. Wellheads can be positioned either subsea or at the topside production facilities. If a wellhead is located at the seabed, a riser, which is a pipe section, can be used for transportation to the topside production facilities. At the topside, the production fluid is transported to the inlet separator.

Normally, the production fluid consists of several different compounds. These compounds are for the most part hydrocarbons, both in gas- and liquid-phase, as well as water and solids. The production flow usually has turbulent character, which implies irregular movement of the liquid. To separate the different compounds in the production fluid for further treatment, separators are used. The most common separator to use due to its wide range of applications and low cost is the horizontal separator, which separates the components in the production fluid by exploiting the density difference between the components and gravity^[12].

Artificial lift is a general term for methods that increase oil production from the wells. One of the most prevalent artificial lift methods is called gas lift. The gas lift method is based on injection of gas into the annulus of a well. The injected gas flows to an injection valve at the lower sections of the well, and into the tubing. The gas affects the production fluid by reducing its density, thus reducing the hydro-static pressure which increases the flow. The injected gas also affects the production fluid by pushing it towards the surface due to the effects of expansion^[12]. The gas lift system uses produced gas as injection gas, and is reliant on a compressor system

to re-compress the produced gas before it can be used for injection. The gas lift system is also dependent on a gas lift manifold with piping and chokes connected to the relevant wells, and an injection valve at the bottom of the annulus. According to Hu(2004)^[13], gas lift will increase the production of the well to the point where hydrostatic-pressure drop can not compensate for the increased friction due to the increased mass of gas in the tubing. If more gas is injected at this point, the production of oil will decrease.

Gas lift is often supplied by a part of the produced gas. The pressure loss from the reservoir to the production facilities lead to the need for re-compression of the gas. In most production facilities compressors are chosen for this need. There are several types of compressors available for this purpose. Centrifugal compressors are often implemented due to the reliable nature compared to other compressors. The compressor is also known for its great tolerance for process fluctuations and its ability to operate smoothly. The centrifugal compressor works by forcing gas through an impeller (inlet), this is achieved by fast rotating blades in the impeller that creates suction pressure. In the impeller the energy of the velocity of the mass is converted into pressure, which in turn leads to a pressure rise in the out-flowing gas in a diffuser (outlet). Therefore, the main function of a centrifugal compressor is to add kinetic energy supplied by the impeller rotor blades to the production gas. The kinetic energy is then converted into potential energy represented by an increase in pressure^[14].

According to McMillan et al.(2010, p.52)^[15], the general pressure ratio a single-step centrifugal compressor are able to achieve is approximately 3:1. However, this pressure ratio is not sufficient for a multitude of implementations, which creates the need for introducing compressors in series. The introduction of compressors in series will increase the total achieved pressure-ratio. At each compressor stage, heat must be removed from the gas to avert the risk of overheating and loss of efficiency. The pressure ratio will increase by rotational speed, and decrease with rise in inlet mass flow. The reason for this being that when the speed increases more kinetic energy are added to the gas, and more potential energy are converted. When an increase in mass is introduced to the system, the system must compress more gas with the same amount of energy, thus resulting in a decrease of pressure ratio^[16].

A polytropic process is a thermodynamic process that includes heat transfer for expansion and compression processes. Polytropic compression considers changes in the gas characteristics before and after compression. The polytropic efficiency of a centrifugal compressor typically range between 0.7 - 0.85 %, which is high compared to other types of compressors^[17].

2.2 CasADI

CasADI is an open-source software framework for numerical optimization. CasADI was first implemented as a tool for algorithmic differentiation (AD), with the use of a computer-algebra system (CAS) syntax. The project was started by Joel Andersson and Joris Gillis in 2018^[5].

CasADI is based on a symbolic framework, where variables are defined as symbolic values. The program consider these values as matrices. CasADI can be used to solve a multitude of different problems related to optimal control, and supplies the users with a toolkit for implementation of optimization problems with less effort and no loss of efficiency^[5].

2.2.1 Initial value problems

Initial value problems (IVP) can be solved in CasADI for both ordinary-differential equations (ODE) and differential-algebraic equations (DAE). The general state-space form of a DAE system are shown in Equation 2.1:

$$\begin{aligned} \dot{x} &= f(t, x, z, p) & x(0) &= x_0 \\ 0 &= g(t, x, z, p), \end{aligned} \tag{2.1}$$

where x is the differential states, z is the algebraic states and p is the parameters.

CasADI implements the SUNDIALS suite integrator IDAS for integration of DAE systems^[18]. The IDAS integrator is based on the backward differentiation formula (BDF), which is linear multi-step methods for solving stiff IVP's and DAE's^[19]. The BDF method, defined in Equation 2.2, has high order and good stability, and is defined for $k = 1, \dots, 6$:

$$\sum_{j=0}^k \alpha_j y_{n+j} = h f_{n+k}, \tag{2.2}$$

where α is the region of $A(\alpha)$ stability, y_{n+j} is the value of y at $n + j$, h is the step-size, and f_{n+k} is the function evaluation at $n + k$ ^[19].

The implementation of the IDAS integrator in CasADI can be summarized with the following steps:

1. Defining the differential, algebraic and parameter variables as symbolic.
2. Defining the algebraic and differential equations.
3. Concatenate all the relating variables and equations.
4. Structure the variables and equations into a single variable defining the DAE system.
5. Defining the integrator function with "IDAS" and the DAE system as input variables.
6. Call the function with initial guesses for the variables.

The time horizon of the integration can be changed by including time in the integrator function^[5]. For further information about CasADI integrators and implementation see *Welcome to casadi's documentation!* by Andersson et al^[20].

2.2.2 Nonlinear programming

CasADI can be used to solve a number of nonlinear optimization problems. The most prevalent nonlinear programming (NLP) solver is the Interior Point Optimizer (IPOPT). IPOPT is a primal-dual interior point algorithm with a filter line-search method for NLP^[21]. A generalized optimization problem can be defined on the form shown in Equation 4.3:

$$\begin{aligned}
\min_x \quad & f(x) \\
\text{s.t.} \quad & c(x) = 0 \\
& g(x) \leq 0 \\
& x^L \leq x \leq x^U
\end{aligned} \tag{2.3}$$

where $f(x)$ is the objective function, $c(x)$ is the equality constraints, $g(x)$ is the inequality constraints, x is the states and x^L and x^U are the lower and upper bounds of the states. IPOPT simplifies this general optimization problem by eliminating the inequality constraints by introducing slack variables, thus forming an equality constraint and a bounded variable^[22]. Applying the assumption that all states are only lower-bounded by 0, IPOPT considers the optimization problem after simplification shown in Equation 2.4:

$$\begin{aligned}
\min_x \quad & f(x) \\
\text{s.t.} \quad & c(x) = 0 \\
& x \geq 0
\end{aligned} \tag{2.4}$$

With the re-formulated optimizing problem, IPOPT as an interior point method, considers the barrier problem formulated by Fiacco and McCormick^[23] in Equation 2.5:

$$\begin{aligned}
\min_x \quad & \phi_\mu(x) = f(x) - \mu \sum_{i=1}^n \ln(x_i) \\
\text{s.t.} \quad & c(x) = 0
\end{aligned} \tag{2.5}$$

where $\phi_\mu(x)$ is the barrier objective function, and the logarithmic term replaces the boundary constraint from Equation 2.4. The method works by starting at an initial point x and a moderate value of μ . The equation for the barrier objective function is then solved. The next evaluation starts at the solution of the previous one and the value of μ is decreased. This results in a more accurate approximation for every iteration. If x_i is near its boundary of zero, the barrier objective function will go towards infinity. Consequently, the optimal solution will be inside the region determined by the boundaries of the states. The method continues until a point which satisfies the first-order optimality conditions known as the Karush–Kuhn–Tucker (KKT) conditions has been identified^[22]. The KKT conditions for the barrier problem are presented in Equation 2.6:

$$\begin{aligned}
\nabla f(x) + \nabla c(x)y - z &= 0 \\
c(x) &= 0 \\
XZe - \mu e &= 0 \\
x, z &\geq 0
\end{aligned} \tag{2.6}$$

where y and z are the Lagrangian multipliers for the equality and bound constraints respectively. The matrices X , Z and e are defined as shown in Equation 2.7:

$$X = \begin{bmatrix} x_1 & 0 & \cdots & 0 \\ 0 & x_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & x_n \end{bmatrix}, \quad Z = \begin{bmatrix} z_1 & 0 & \cdots & 0 \\ 0 & z_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & z_n \end{bmatrix}, \quad e = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad (2.7)$$

For each evaluation of Equation 2.5, the Newton method is used. The newton step is found by the use of the KKT equations shown in Equation 2.6, that strictly satisfies $x, z > 0$. The steps of x, y and z are computed for each evaluation k :

$$\begin{bmatrix} W_k + X_k^{-1}Z_k & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{pmatrix} \Delta x_k \\ \Delta y_k \end{pmatrix} = - \begin{pmatrix} \phi_\mu(x_k) + \nabla c(x_k)y_k \\ c(x_k) \end{pmatrix}, \quad (2.8)$$

where $\Delta z = X_k^{-1}\mu e - z_k - X_k^{-1}Z_k\Delta x_k$ and W_k is the Hessian of the Lagrangian function. Further, a line search is performed until a sufficient step-size is achieved. The next point is then computed by the step and the step-size. When the Newton method has found a point that satisfies the KKT conditions, μ is decreased as described above^[22].

The implementation of a NLP solver in CasADI consist of the following steps:

1. Defining the differential, algebraic and parameter variables as symbolic.
2. Defining the objective function and the algebraic and differential equations.
3. Concatenate the algebraic and differential equations, thus defining the equality constraints.
4. Structure the variables, equality constraints, inequality constraints and the objective function into a single variable defining the NLP system.
5. Defining the nlsolve function with "ipopt" and the NLP system as input variables.
6. Call the function with initial guesses and upper/lower boundaries for the variables and constraints.

For further information about CasADI NLP solvers and implementation see *Welcome to casadi's documentation!* by Andersson et al^[20].

2.3 PID controller

One of the most prevalent controller-types for industrial purposes is the proportional-integral-derivative (PID) controller. The proportional (P) part of the controller, the P-controller, is used when the controller action is proportional to the magnitude of the controller error. The proportional control output are defined in Equation 2.9:

$$u(t) = K_p e(t), \quad (2.9)$$

where K_p is the proportional gain and $e(t)$ is the error, the difference between the measured controlled variable and the setpoint of the variable. Due to the simplistic nature of the P-controller, the integral (I) part of the controller, the I-controller, is implemented to reduce steady-state off-set. The integral control output are defined in Equation 2.10:

$$u(t) = K_I \int_0^t e(\tau) d\tau, \quad (2.10)$$

where K_I is the integral gain. The derivative (D) part of the controller, the D-controller, utilizes the rate of change of previous errors to "predict" future behaviour. The derivative control output are defined in Equation 2.11:

$$u(t) = K_D \frac{de}{dt}, \quad (2.11)$$

where K_D is the derivative gain. The total PID control output are defined as the combination of the three control outputs: proportional control, integral control and derivative control. The PID control output are shown in Equation 2.12:

$$u(t) = K_p e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de}{dt} \quad (2.12)$$

The most utilized form of the PID control output equation however, is the time constant form, presented in Equation 2.13:

$$u(t) = K_p \left(e(t) + \frac{1}{\tau_I} \int_0^t e(\tau) d\tau + \tau_D \frac{de}{dt} \right), \quad (2.13)$$

where τ_I is the integral time and τ_D is the derivative time. The resulting tuning parameters are K_p , τ_I and τ_D ^[24].

2.4 SIMC tuning

Parameter tuning is a method for obtaining values for a given parameter. Section 2.3 shows that the PID-controller has three different tuning parameters: K_p , τ_I and τ_D . To be able to tune these three parameters, a systematic approach is necessary. Several methods for tuning of the three control tuning parameters have been purposed throughout time. Ziegler and Nichols (1942)^[25] proposed a method that handles disturbance rejection in integrating processes, but often results in aggressive settings. Another tuning method is the IMC method, proposed by Riviera et al.(1986)^[26]. This paper however, considers the tuning rules of *Skogestad Internal method control* (SIMC). These SIMC rules were developed by Sigurd Skogestad, with the goal of making a set of simple tuning rules that works well for a wide range of processes^[27].

For an open-loop step in the manipulated variable (u) and the resulting change in the controlled variable (y), shown in Figure 2.1, it is possible to estimate the process gain (k), The process time delay (θ) and the dominant lag time constant (τ_1). For a second-order process the second order lag time(τ_2) can also be estimated.

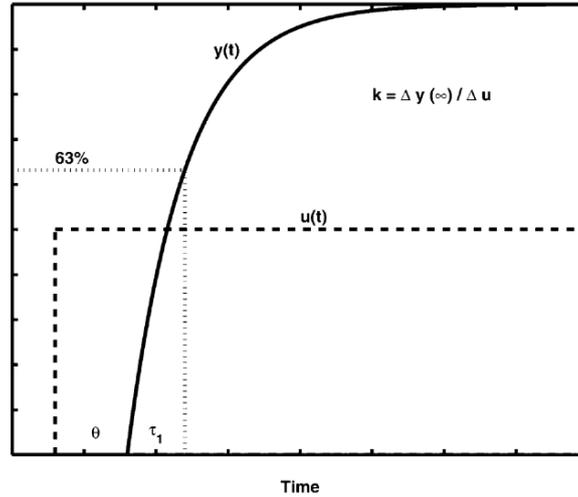


Figure 2.1: Step response of first-order plus time delay process^[27].

The transfer function of a general first-order process with time delay are defined in Equation 2.17:

$$g(s) = \frac{k e^{-\theta s}}{\tau_1 s + 1} \quad (2.14)$$

The SIMC rules can be derived by using the IMC approach, with controller time (τ_c) as the only tuning parameter, on the closed-loop setpoint response. Application of the SIMC rules results in a PI-controller with controller settings presented in Equations 2.15 and 2.16:

$$K_c = \frac{1}{k} \frac{\tau_1}{\tau_c + \theta}, \quad (2.15)$$

$$\tau_I = \min\{\tau_1, 4(\tau_c + \theta)\}, \quad (2.16)$$

where K_C is the controller gain and τ_I is the integral time.

For integral processes, where $\tau_1 \gg \theta$, the process takes a long time to stabilize. The process gain will in this case be estimated from the slope of the controlled variable described in Figure 2.2.

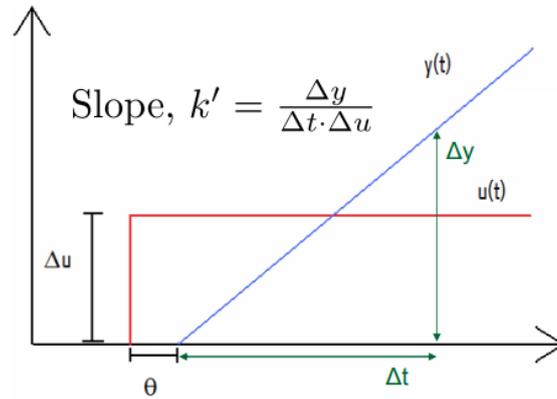


Figure 2.2: Open loop step response on integrating process^[28].

The first-order process with time delay described in Equation 2.17 can then be approximated as an integrating process:

$$g(s) = \frac{k' e^{-\theta s}}{s} \quad (2.17)$$

Application of the SIMC procedure to the process defined in Equation 2.17 results in a PI-controller with controller settings presented in Equation 2.18 and 2.19:

$$K_c = \frac{1}{k'} \frac{1}{\tau_c + \theta} \quad (2.18)$$

$$\tau_I = 4(\tau_c + \theta) \quad (2.19)$$

Equation 2.18 and 2.19 show that an integral process only has one tuning parameter, τ_c ^{[28][27]}.

3 Modelling

3.1 Model description

The model consists of five main sections: the well system, the riser, the separator, the compressor train and the gas lift system. Each section was modelled independently, before being integrated in the total model. The modelling section will describe the dynamics and mathematical background for each subsystem, and explain how the mathematical relations are derived. The main assumptions that has been made are also presented in this section. The constant parameters for each subsystem will be listed, as well as a summary of the equations and the states. The well system consists of six separate wells with gas lift. The wells are connected through a mutual manifold, which are located at the seabed. The produced oil and gas from each of the six wells merges in this manifold, before it is transported to the riser. The riser connects the subsea facilities with the separator at topside, where the oil and gas are separated. The outlet gas of the separator is then either routed to export or to compression. This model only concerns the gas that are sent to compression, so the process of the gas that are routed to export will not be described further. From the compressor the gas is routed to the different wells to be used as lift gas. The outlet oil of the separator is sent to export. A pictorial representation of the entire model is shown in Figure 3.1.

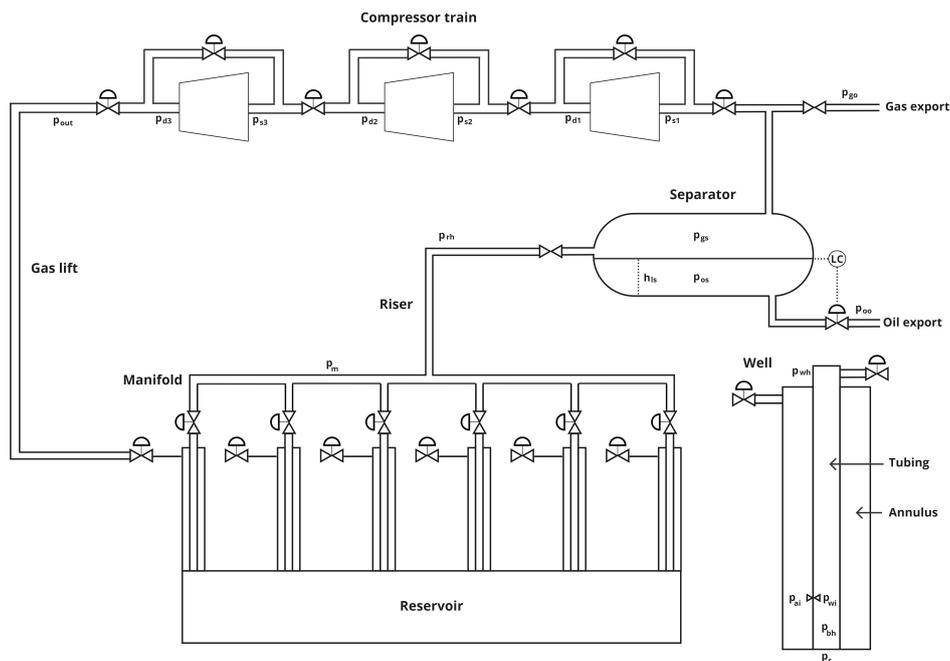


Figure 3.1: Sketch of the total model flow-sheet.

3.2 Well system

This section will present the mathematical background of the entire well system. All relevant assumptions that has been made will also be listed.

A simplified version of the gas lift well system are presented in Figure 3.2. This figure shows how the oil and gas in the reservoir flow into the tubing and up through the production choke. The lift gas enters the annulus through a gas lift choke and is injected into the tubing through an injection valve. The part of the tubing that extends from the reservoir to the injection valve is defined as the bottom hole. The part of the tubing that extends from the injection valve and

up to the wellhead is defined as the well. The main assumptions for the well model is two-phase flow i.e oil and gas and ideal gas behaviour. Further assumptions will be mentioned during the section.

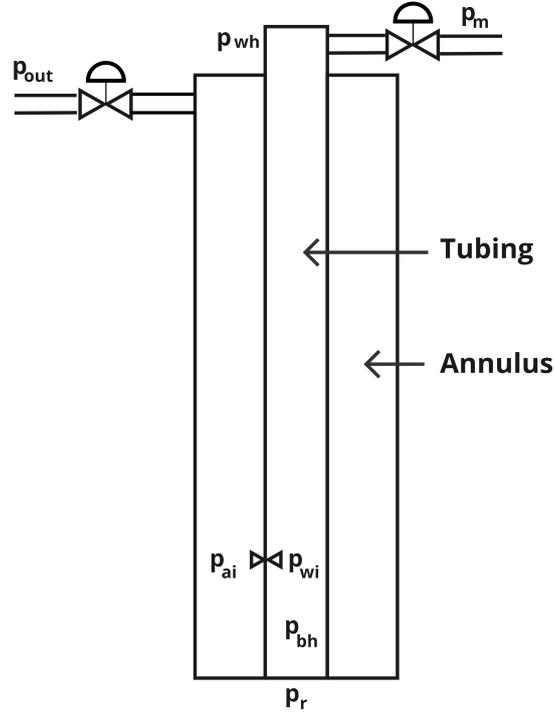


Figure 3.2: Sketch of the simplified gas lift well system.

3.2.1 Mass balances

The generalized mass balance equation, assuming no reactions, are defined in Equation 3.1:

$$\dot{m} = w_{in} - w_{out}, \quad (3.1)$$

where \dot{m} is the mass rate in the system, w_{in} is the mass flow into the system and w_{out} is the mass flow out of the system^[29].

By assuming that the reservoir contains only oil and gas, the component mass balances in the tubing and annulus can be derived from Equation 3.1. These mass balances, which are valid for each of the wells in the system, are presented in Equation 3.2 to 3.4.

$$\dot{m}_{ga_i} = w_{gl_i} - w_{iv_i}, \quad (3.2)$$

$$\dot{m}_{gt_i} = w_{iv_i} + w_{rg_i} - w_{pg_i}, \quad (3.3)$$

$$\dot{m}_{ot_i} = w_{ro_i} - w_{po_i}, \quad (3.4)$$

where i denotes each well, m_{ga} is the mass rate of gas in the annulus, w_{gl} is the gas lift flow, w_{iv} is the gas flow from annulus to tubing, m_{gt} is the mass rate of gas in the tubing, w_{rg} is the gas flow from the reservoir, w_{pg} is the produced gas flow, m_{ot} is the mass rate of oil in the tubing, w_{ro} is the oil flow from the reservoir and w_{po} is the produced oil flow.

3.2.2 Pressure

There are four important pressure regions in the well system: the annulus pressure at the injection point, the bottom hole pressure, the gas injection point pressure and the wellhead pressure. The reservoir pressure is assumed to be known and can be found in Appendix A. To formulate the equations we first need to define Bernoulli's equation. The pressure equations are derived starting from the Bernoulli's equation, presented in Equation 3.5:

$$p_1 + \frac{1}{2}\rho v_1^2 + \rho gh_1 = p_2 + \frac{1}{2}\rho v_2^2 + \rho gh_2, \quad (3.5)$$

where p_1 is the initial pressure, ρ is the density of the fluid, v_1 is the initial velocity, g is the gravitational constant, h_1 is the initial height, p_2 is the final pressure, v_2 is the final velocity and h_2 is the final height. By assuming static gas and the relative height of h_1 to be zero, Equation 3.5 can be simplified to the form shown in Equation 3.6:

$$p_2 = p_1 + \rho gh_2 \quad (3.6)$$

By combining Equation 3.6 and the ideal gas law the annulus pressure at the injection point p_{a_i} can be formulated as shown in Equation 3.7:

$$p_{a_i} = \left(\frac{R T_{a_i}}{V_{a_i} M_w} + \frac{g H_{a_i}}{V_{a_i}} \right) m_{ga_i}, \quad (3.7)$$

where p_a is the annulus pressure, R is the universal gas constant, T_a is the temperature in the annulus, V_a is the volume of the annulus, M_w is the molecular weight of the gas and H_a is the height of the annulus from the injection point.

The pressure of the gas inside the tubing are found by the ideal gas law. To find the volume of gas we need to subtract the volume of oil from the total volume of the tubing. For the wellhead pressure, the pressure of the gas inside the tubing are combined with the the second term in Equation 3.6, assuming the ideal gas pressure is located at the center of the pipe length. This yields a more realistic pressure profile for the wellhead pressure, which is located at the top of the tubing. The equation for the wellhead pressure are presented in Equation 3.8.

$$p_{wh_i} = \frac{RT_{w_i}}{M_w} \left(\frac{m_{gt_i}}{V_{w_i} + V_{bh_i} - \frac{m_{ot_i}}{\rho_o}} \right) - \left(\frac{m_{gt_i} + m_{ot_i}}{V_{w_i}} \right) g \frac{H_{w_i}}{2}, \quad (3.8)$$

where p_{wh} is the wellhead pressure, T_w is the well temperature, V_w is the well volume, V_{bh} is the bottom hole volume, ρ_o is the density of the oil and H_w is the height of the well.

By obtaining the equation for the wellhead pressure, the relationship between this variable and the injection point pressure and the bottom hole pressure that is located further down in the tubing can be derived. To acquire this relationship, the total pressure drop equation must be

evaluated under the assumptions of zero component loss, fittings loss or pump head^[30]. This relationship is shown in Equation 3.9.

$$P[end] = P[start] - \text{Friction loss} + \text{Elevation}[start - end] \quad (3.9)$$

Equation 3.6 gives a relationship for pressure difference related to the static pressure for elevation differences in the pipe. To obtain the pressure drop related to friction, the Hagen-Poiseuille equation assuming laminar flow can be used^[31]. This equation describes the characteristics of laminar flow through a tube, and the pressure loss due to viscosity, and are shown in Equation 3.10.

$$P = \frac{128QL\mu}{\pi D^4}, \quad (3.10)$$

where P is the pressure drop in the pipe, Q is the volumetric flow, L is the length of the pipe segment, μ is the fluid viscosity and D is the diameter of the pipe.

The Hagen-Poiseuille equation, shown in Equation 3.10, provides a relationship for the friction pressure drop. This makes it possible to derive an equation for the injection point pressure. The gas lift injection point is located at the bottom of the well, which is the top of the bottom hole. Therefore, the amount of mass between the wellhead and the injection point will be equal to the total mass in the tubing, minus the amount that is located in the bottom hole. We also use the fact that the volumetric flow equals the mass flow divided by the density. By assuming that the mass of gas in the bottom hole is negligible compared to the mass of oil, the injection point pressure becomes as shown in Equation 3.11:

$$p_{wi} = p_{whi} + \frac{gH_{wi}}{A_{wi}L_{wi}}(m_{oti} + m_{gti} - \rho_o L_{bhi} A_{bhi}) + \frac{128\mu_o L_{wi} w_{pci}}{\pi D_{wi}^4 \left(\frac{(m_{gti} + m_{oti}) p_{whi} M_w \rho_o}{m_{oti} p_{whi} M_w + \rho_o RT_{wi} m_{gti}} \right)}, \quad (3.11)$$

where p_{wi} is the gas injection point pressure, A_w is the area of the well, L_w is the length of the well, L_{bh} is the length of the bottom hole, A_{bh} is the area of the bottom hole, μ_o is the oil viscosity, w_{pc} is the flow through the production choke and D_w is the diameter of the well. The bottom hole pressure is found by the same principles as in Equation 3.11. In this case the pressure is calculated relative to the injection point pressure and by assuming that the bottom hole is filled with oil. The equation for the bottom hole pressure is shown in Equation 3.12:

$$p_{bhi} = p_{wi} + \rho_o g H_{bhi} + \frac{128\mu_o L_{bhi} w_{ro_i}}{\pi D_{bhi}^4 \rho_o}, \quad (3.12)$$

where p_{bh} is the bottom hole pressure, H_{bh} is the height of the bottom hole and D_{bh} is the diameter of the bottom hole.

3.2.3 Flow

Figure 3.2 shows that there is flow through the injection valve, the production choke and flow from the reservoir. The flow from the reservoir contains both oil and gas, which flows up through the tubing and through the production choke. The re-compressed gas from the compressor flows

in through the gas lift choke and down the annulus, before it flows through the injection valve into the well, where it mixes with the reservoir oil and gas flow. The valve equation can be used to find expressions for the flows^[32]. The valve equation for mass flow is shown in Equation 3.13:

$$w = uC_v\sqrt{\rho(P_1 - P_2)}, \quad (3.13)$$

where w is the mass flow, u is the valve opening, C_v is the flow coefficient, ρ is the density of the fluid, P_1 is the inlet pressure and P_2 is the outlet pressure. This equation gives a relationship between flow and pressure difference across a valve. With this, the flow through the injection valve and production choke can be found as well.

From Figure 3.2, we can observe that the inlet pressure of the injection valve is the same as the annulus pressure and the outlet is the same as the injection point pressure found in Equation 3.7.

$$w_{iv_i} = C_{iv}\sqrt{\rho_a(p_{a_i} - p_{iv_i})}, \quad (3.14)$$

where C_{iv} is the injection valve flow coefficient, ρ_a is the density of gas in the annulus, and p_{iv} is the injection point pressure. Figure 3.2 shows that the inlet pressure of the production choke is the same as the wellhead pressure derived in Equation 3.8. The figure also shows that the outlet pressure of the production choke is equal to the manifold pressure. This gives an equation for the mass flow through the production choke, which are shown in Equation 3.15.

$$w_{pc_i} = u_{pc}C_{pc}\sqrt{\rho_m(p_{wh_i} - p_m)}, \quad (3.15)$$

where u_{pc} is the production choke opening, C_{pc} is the production choke flow coefficient, ρ_m is the density of the mixed fluid in the wellhead, and p_m is the manifold pressure.

To find the gas mass flow through the production choke, a relation between the total gas in the tubing, found in Equation 3.3, and the total oil, found in Equation 3.4, is used. The percentage of gas in the total flow, found in Equation 3.15 will be equal to the total mass of gas in the tubing divided by the total mass of oil and gas in the tubing. The equation for the flow of the produced gas is given in Equation 3.16:

$$w_{pg_i} = \frac{m_{gt_i}}{m_{gt_i} + m_{ot_i}}w_{pc_i} \quad (3.16)$$

When deriving the equation for produced oil the same principles as for the produced gas are used. The only difference being we are solving for the amount of oil. The equation for the flow of the produced oil is given in Equation 3.17:

$$w_{po_i} = \frac{m_{ot_i}}{m_{gt_i} + m_{ot_i}}w_{pc_i} \quad (3.17)$$

The reservoir oil flow can be found by utilizing the reservoir productivity index. The productivity index gives an indication of the wells ability to produce hydrocarbons^[33]. The equation for the productivity index given in kg/s is presented in Equation 3.18:

$$PI = \frac{w_o}{p_{res} - p_{bh}}, \quad (3.18)$$

where PI is the productivity index, w_o is the reservoir oil flow, p_{bh} is the bottom hole pressure and p_{res} is the reservoir pressure. Rearranging this equation gives the equation for the reservoir oil flow, shown in Equation 3.19:

$$w_{ro} = \frac{PI}{p_{res} - p_{bh}} \quad (3.19)$$

The gas-oil ratio can be used to find the reservoir gas flow from the reservoir oil flow. GOR is the volumetric ratio between oil and gas in the crude oil. The equation for the reservoir gas flow can thus be related to the reservoir oil flow, and is given in Equation 3.20:

$$w_{rg} = GORw_{ro} \quad (3.20)$$

3.2.4 Density

In the modelling of the density, there are two regions of interest: the density of gas in the annulus and the density of the fluid mixture in the wellhead. The density of the gas in the annulus can be derived by assuming that the gas possess ideal behaviour in the annulus. The density of the gas in the annulus can thus be derived from the ideal gas law, and is presented in Equation 3.21.

$$\rho_{a_i} = \frac{M_w p_{a_i}}{RT_{a_i}} \quad (3.21)$$

The mixed density can be found from assuming that the total density of the mixture equals the total mass, divided by the total volume. The total volume is represented by the volume of each component in the mixture, as shown in Equation 3.22:

$$\rho_t = \frac{M_t}{\frac{M_a}{\rho_a} + \frac{M_b}{\rho_b}}, \quad (3.22)$$

where ρ_t is the total density, M_t is the total mass and a and b represents the density and mass of the two entities. By multiplying with density of a and b the previous expression can be rewritten into the form presented in Equation 3.23:

$$\rho_t = \frac{M_t \rho_a \rho_b}{M_a \rho_b + M_b \rho_a} \quad (3.23)$$

Implementing Equation 3.23 and the ideal gas law for the density of gas, the mixed density in the wellhead can be defined as shown in Equation 3.24:

$$\rho_m = \frac{(m_{gt} + m_{ot}) p_{wh} M_w \rho_o}{m_{ot} p_{wh} M_w + \rho_o R T_w m_{gt}} \quad (3.24)$$

3.2.5 Equations and states

The well model shows that we have both differential- and algebraic equations in the the well system. We consider the GOR of each well as a potential disturbance to the overall system. We assume that we can manipulate the opening of the production chokes. The summarized system can be described for each well (i) as:

$$x_{well} = [m_{ga_i}, m_{gt_i}, m_{ot_i}]^T \quad (3.25a)$$

$$z_{well} = [p_{ai_i}, p_{wh_i}, p_{wi_i}, p_{bh_i}, \rho_{ai_i}, \rho_{m_i}, w_{iv_i}, w_{pc_i}, w_{pg_i}, w_{po_i}, w_{ro_i}, w_{rg_i}]^T \quad (3.25b)$$

$$p_{well} = [GOR_i, p_{res_i}]^T \quad (3.25c)$$

$$u_{well} = [u_{pc_i}]^T \quad (3.25d)$$

where x_{well} represents the differential states of the well system, z_{well} represents the algebraic states of the well system, p_{well} represents the parameters of the well system and u_{well} represents the control variables of the well system. The constant values and parameters of the well system can be found in Appendix A.

3.3 Manifold and Riser

This section will present the mathematical background of the manifold and riser system. All relevant assumptions that has been made will also be listed.

A simplified version of the well system, manifold and riser are presented in Figure 3.3. The figure show how the oil and gas flows from the production choke of each well to a mutual manifold. In the manifold, the production fluids from each well are mixed. From the manifold, the total flow is transported to the riser, which is assumed to be perfectly vertical and rigid. The riser transports the flow from the subsea facilities and up to the surface production facilities.

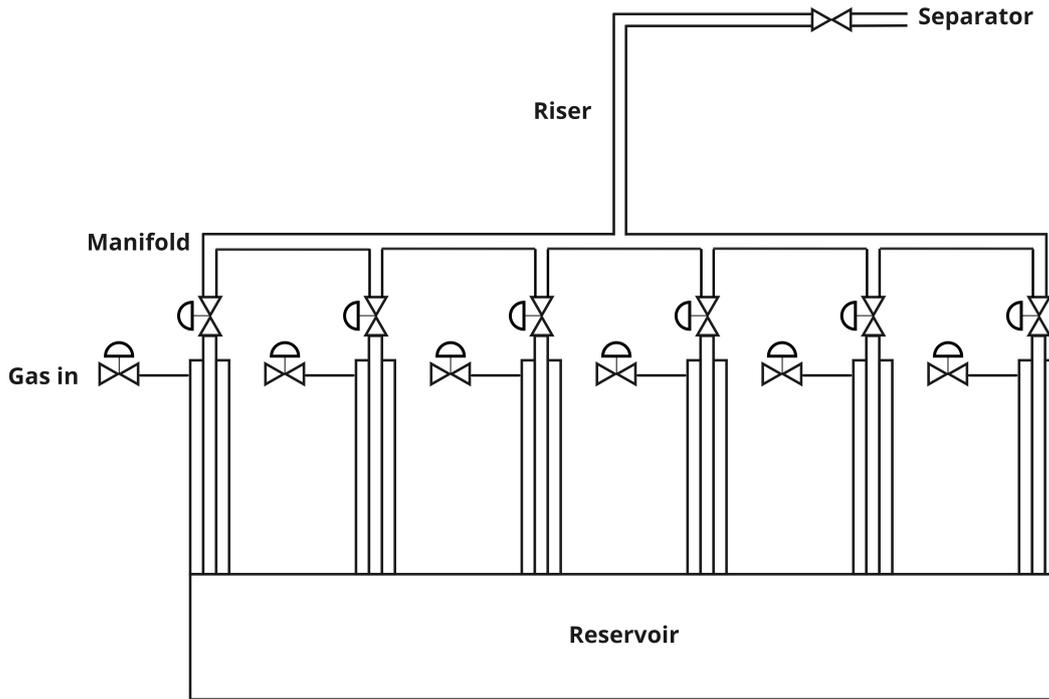


Figure 3.3: Sketch of the simplified well system with manifold and riser.

3.3.1 Mass balances

Figure 3.3 shows that the inlet flow of the riser is equal to the total flow of the six wells. Furthermore, the outlet flow of the riser is the same as the the total flow going through the valve located before the separator. Consequently, the oil and gas mass rate in the riser can be found by the general mass balance equation derived in Equation 3.1. Here, the inlet flow is the sum of oil produced from each well, shown in Equation 3.17, and the gas, shown in Equation 3.20. The equations for the mass of oil and gas in the riser as shown in Equation 3.26 and 3.27 respectively.

$$\dot{m}_{or} = \sum_{i=1}^6 w_{po} - w_{to}, \quad (3.26)$$

$$\dot{m}_{gr} = \sum_{i=1}^6 w_{pg} - w_{tg}, \quad (3.27)$$

where m_{r_o} is the mas rate of oil in the riser, w_{po} is the produced oil from each well and w_{to} is the total oil flow going to the separator, m_{r_g} is the mas rate of gas in the riser, w_{pg} is the produced gas from each well and w_{tg} is the total gas flow going to the separator.

3.3.2 Pressure

For the modelling of the riser there are two important pressure regions: the pressure at the top of the riser (riser head pressure), and the manifold pressure. These pressures are derived with the same logic as the dependent pressures in the well. The riser head pressure is derived from

the ideal gas law and the static pressure drop from Equation 3.6. The static pressure drop is added since we assume that the ideal pressure of gas is in the center of the length of the riser. The equation for the pressure in the riser head is shown in Equation 3.28:

$$p_{rh} = \frac{RT_r}{M_w} \left(\frac{m_{gr}}{L_r A_r} \right) - \left(\frac{m_{gr} + m_{or}}{L_r A_r} \right) g \frac{H_r}{2}, \quad (3.28)$$

where p_{rh} is the riser head pressure, L_r is the length of the riser, A_r is the area of the riser and H_r is the height of the riser.

The manifold pressure can be found by using the riser head pressure shown in Equation 3.28, and the total pressure drop, shown in Equation 3.9. The manifold pressure equation is given in Equation 3.29:

$$p_m = p_{rh} + \frac{gH_r}{A_r L_r} (m_{or} + m_{gr}) + \frac{128\mu_o L_r w_{pr}}{\pi D_r^4 \left(\frac{(m_{gr} + m_{or}) p_{rh} M_w \rho_{ro}}{m_{or} p_{rh} M_w + \rho_o RT_r m_{gr}} \right)}, \quad (3.29)$$

where p_m is the manifold pressure and T_r is the temperature in the reservoir.

3.3.3 Flow

The total flow through the riser can be found by utilizing the valve equation defined in Equation 3.13. From Figure 3.3 it can be observed that the inlet pressure is the same as the riser head pressure, and the outlet pressure is the same as the separator pressure. The equation for the total flow from the riser are presented in Equation 3.30:

$$w_{pr} = C_{pr} \sqrt{\rho_r (p_{rh} - p_{gs})}, \quad (3.30)$$

where w_{pr} is the total production mass flow from the riser, C_{pr} is the valve characteristics, ρ_r is the mixed density in the riser and p_{gs} is the pressure of the gas in the separator. From the equation of total mass flow through the riser valve, the flow of oil and gas can be derived. The oil flow from the riser will be equal to the mass of oil in the riser divided by the total mass times the total flow. The equation for total oil flow through the riser are shown in Equation 3.31:

$$w_{to} = \frac{m_{or}}{m_{gr} + m_{or}} w_{pr}, \quad (3.31)$$

where w_{to} is the total oil flow from the riser. The gas flow from the riser is derived using the same method as for Equation 3.31. Thus, the equation for total gas flow through the riser are shown in Equation 3.32:

$$w_{tg} = \frac{m_{gr}}{m_{gr} + m_{or}} w_{pr}, \quad (3.32)$$

3.3.4 Density

The density at the riser head can be found by implementing Equation 3.23, and the density of gas from the ideal gas law. The resulting equation for the density of oil and gas in the riser head is shown in Equation 3.33:

$$\rho_r = \frac{(m_{gr} + m_{or})p_{rh}M_w\rho_{ro}}{m_{or}p_{rh}M_w + \rho_{ro}RT_r m_{gr}} \quad (3.33)$$

3.3.5 Equations and states

The riser model includes both differential and algebraic states. We do not consider any potential disturbances or controlled variables in the riser system. The system model can be described as shown in Equation 3.34:

$$x_{riser} = [m_{or}, m_{gr}]^T \quad (3.34a)$$

$$z_{riser} = [p_{rh}, \rho_r, p_m, w_{pr}, w_{to}, w_{tg}]^T \quad (3.34b)$$

where x_{riser} are the differential states of the riser system and z_{riser} are the algebraic states of the riser system. The constant values of the riser system can be found in Appendix A.

3.4 Separator

This section will present the mathematical background of the separator. All relevant assumptions that has been made will also be listed.

A simplified version of the separator are presented in Figure 3.4. The production fluid enters the separator where it gets separated into a gaseous phase and a liquid phase due to the difference in density. We assume perfect separation, no thermodynamic effects between the two phases, no foaming and no droplet generation. The level in the tank, and subsequently the pressure, is controlled by a level controller to secure safe operation. The oil level is always assumed to be below the inlet of the separator. The gas is either routed to export or to the compressor train to be used as lift gas. Since perfect separation is assumed, no further treatment of the gas is necessary before compression. The oil is sent to export.

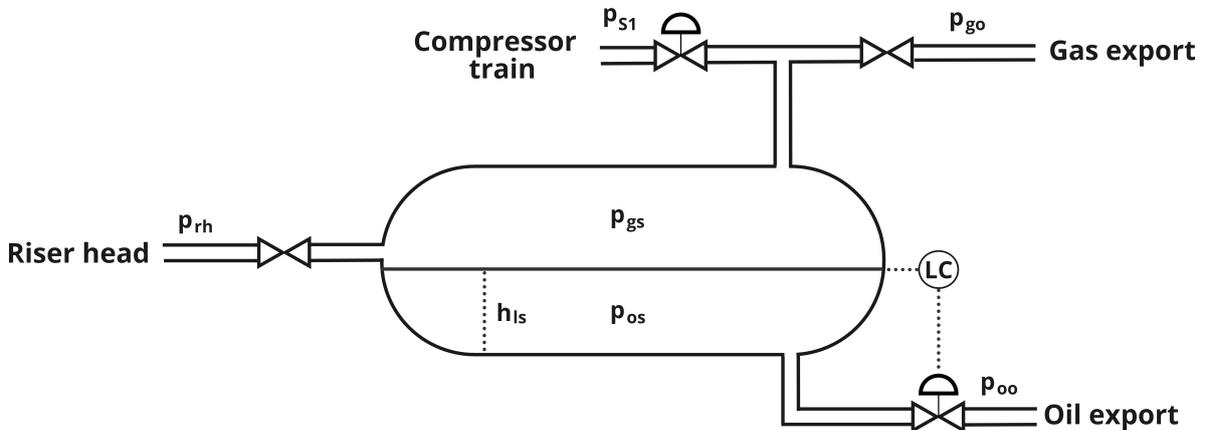


Figure 3.4: Sketch of the simplified Separator.

3.4.1 Pressure

Figure 3.4 shows that the separator has six pressure regions. The oil and gas export pressures are assumed to be constant parameters found in Appendix A, the pressure in the riser head is defined in Equation 3.28 and the suction pressure of the first compressor is derived in the compressor model Section 3.5. The two remaining pressure regions are the pressure of the gaseous phase and the liquid phase in the separator. The pressure of the gas in the separator depends on the inlet flow and the outlet flow of the separator.

Figure 3.4 shows that the separator consist of one inlet flow and three outlet flows. The inlet flow comes from the riser, and consist of partially gas and partially oil. The outlets are one oil outlet, and two gas outlets. By assuming constant temperature over the separator and ideal gas behaviour, Equation 3.1 can be manipulated into the form shown in Equation 3.35:

$$\dot{n}_g = \frac{w_{gin} - w_{gout}}{M_w}, \quad (3.35)$$

where n_g is the mole rate of gas in the system, w_{gin} is the mass flow of gas into the system and w_{gout} is the mass flow of gas out of the system. The molar rate can be related to the pressure by deriving the time-derivative of the ideal gas law^[3]. This time-derivative is shown in Equation 3.36:

$$\dot{n}_g RT = V_g \dot{p}_g + p_g \dot{V}_g, \quad (3.36)$$

where V_g is the volume of the gas in the system and p_g is the pressure of the gas in the system. Based on the dynamics of the system it is assumed that the gas volume rate of change in the system is equal to the negative of the oil volume rate of change. Thus, when the oil volume increases, the gas volume will decrease. The volumetric flow rate of the gas is defined as shown in Equation 3.37:

$$\dot{V}_o = q_{oin} - q_{oout}, \quad (3.37)$$

where \dot{V}_o is the oil volume rate in the separator and q_o is the volumetric flow rate in and out of the system. By adding Equation 3.35 to Equation 3.36, and introducing the relation given in Equation 3.37, the resulting equation for the pressure rate in the separator is defined in Equation 3.38:

$$\dot{p}_{gs} = \frac{RT_s}{V_{gs} M_w} (w_{tg} - w_{gs} - w_{in1}) + \frac{p_{gs}}{V_{gs} r h_{oro}} (w_{to} - w_{os}), \quad (3.38)$$

where p_{gs} is the pressure rate in the separator, V_{gs} is the volume of gas in the separator, w_{gs} is the gas to export, w_{in1} is the gas to the compressor train and w_{os} is the oil to export. The pressure of the liquid phase can be related to the pressure in the gaseous phase by Equation 3.6. Consequently, the equation for the oil pressure in the separator is achieved. This equation is presented in Equation 3.39.

$$p_{os} = p_{gs} + r h_{oro} g h_{ls}, \quad (3.39)$$

where p_{os} is the pressure of the oil in the separator and h_{ls} is the height of the oil in the separator.

3.4.2 Flow

Figure 3.4 shows three separate flows out of the separator. These outlet flows are the export gas, the gas going to the compressors and the export oil. The outlet flows are dependent on the pressure of gas in the separator and by the pressure of the valves positioned on the outlet flow lines. The outlet flows can be derived by the use of the valve equation, given in Equation 3.13. Therefore, the oil flow out of the separator can be obtained by Equation 3.40:

$$w_{os} = u_{os} C_{os} \sqrt{\rho_{ro}(p_{oo} - p_{os})}, \quad (3.40)$$

where u_{os} is the valve opening, C_{os} is the valve characteristics and p_{oo} is the export pressure of the oil. The mass flow of the gas that is sent to export are described by Equation 3.41:

$$w_{gs} = C_{gs} \sqrt{\rho_{gs}(p_{go} - p_{gs})}, \quad (3.41)$$

where C_{gs} is the valve characteristics and p_{go} is the export pressure of the gas. The mass flow of the gas that is transported to the compressor train are obtained from Equation 3.42:

$$w_{in1} = u_{c1} C_{c1} \sqrt{\rho_{gs}(p_{s1} - p_{gs})}, \quad (3.42)$$

where u_{c1} is the valve opening of the inlet valve of the first compressor, C_{c1} is the valve characteristics and p_{s1} is the suction pressure of the first compressor.

3.4.3 Level and volume

Due to the risk of overflowing of the separator, the liquid height must be controlled. Since it is assumed that only gas and oil are present in the model, the separator only contains one single uniform liquid phase. To derive an expression of how the height of the oil changes with time, an expression of the total area of the liquid in the cylindrical separator needs to be derived. Equation 3.43 shows the segment of a circle^[34]:

$$A_{ls} = \frac{r_s^2}{2} (\theta - \sin(\theta)), \quad (3.43)$$

where A_{ls} is the area of the liquid phase in the separator, r_s is the radius of the separator and theta is the angle defined by the segment section. To find the area of the segment, the total area of the section theta spans in the circle is subtracted with the triangle that forms above the section. The angle can be found from the Pythagorean theorem, using the inverse of the cosine function. By dividing the isosceles triangle into two right triangles, the angle can be found with the adjacent side corresponding to the radius, minus the liquid height and the hypotenuse corresponding to the radius. The area of the segment can thus be formulated as shown in Equation 3.44:

$$A_{ls} = \frac{r_s^2}{2} \left(2\cos^{-1} \left(\frac{r_s - h_{ls}}{r_s} \right) - \sin \left(2\cos^{-1} \left(\frac{r_s - h_{ls}}{r_s} \right) \right) \right) \quad (3.44)$$

By differentiating this equation with respect to time, the expression for the area differentiated with respect to time is given in Equation 3.45:

$$\dot{A}_{ls} = \dot{h}_{ls} \left(r_s^2 \left(\frac{1 - \cos \left(2\cos^{-1} \left(\frac{r_s - h_{ls}}{r_s} \right) \right)}{\sqrt{h_{ls}(2r_s - h_{ls})}} \right) \right) \quad (3.45)$$

Based on Equation 3.44 and the relationship between the area and the volume ($A = \frac{V}{L}$) an equation for the liquid height differentiated with time can be derived, with the relation for the volume differentiated with time found in Equation 3.37 as:

$$\dot{h}_{ls} = \frac{w_{os} - w_{to}}{\rho_{ro} 2L \sqrt{h_{ls}(2r_s - h_{ls})}}, \quad (3.46)$$

where L is the length of the separator. Furthermore, there are two volume regions in the separator. These regions are the volume of oil and gas. The equation for the area of liquid in the separator has already been derived. Thus the relationship between volume and the area can be used to find the volume of oil in the separator, which is shown in Equation 3.47.

$$V_{os} = \frac{Lr_s^2}{2} \left(2\cos^{-1} \left(\frac{r_s - h_{ls}}{r_s} \right) - \sin \left(2\cos^{-1} \left(\frac{r_s - h_{ls}}{r_s} \right) \right) \right) \quad (3.47)$$

The total volume of the separator is constant, and due to the nature of gas, which makes it occupy all remaining volume, the gas volume can be related to the volume of oil. The volume of gas, expressed by the volume of oil is shown in Equation 3.48:

$$V_{gs} = V_s - V_{os}, \quad (3.48)$$

where V_{gs} is the volume of gas in the separator and V_s is the separator volume.

3.4.4 Density

There are two density regions in the separator: the density of the oil, and the density of the gas. The density of the oil in the separator is assumed to be equal to the density of the oil in the riser, which is defined as the average of the densities of the oil from the wells. The gas density however, is dependent on the pressure of the gas in the separator. By applying the assumption of ideal gas behaviour, the density of the gas in the separator can be calculated from the ideal gas law. The expression for the density of the gas in the separator is presented in Equation 3.49.

$$\rho_{gs} = \frac{M_w P_{gs}}{RT_s}, \quad (3.49)$$

where ρ_{gs} is the density of the gas in the separator.

3.4.5 Equations and states

The separator system includes both differential and algebraic states and the corresponding equations to calculate them. By assuming that we can manipulate the opening of the control valve at the oil outlet and the control valve to the compressor train (The control valve to the compressor train will be covered later), the summarized system can be described as shown in Equation 3.50:

$$x_{sep} = [p_{gs}, h_{ls}]^T \quad (3.50a)$$

$$z_{sep} = [w_{os}, w_{gs}, \rho_{gs}, p_{os}, V_{os}, V_{gs}]^T \quad (3.50b)$$

$$p_{sep} = [p_{oo}, p_{go}]^T \quad (3.50c)$$

$$u_{sep} = [u_{os}]^T \quad (3.50d)$$

where x_{sep} represents the differential states of the separator system, z_{sep} represents the algebraic states of the separator system, p_{sep} represents the parameters of the separator system and u_{sep} represents the control variables of the separator system. The constant values and parameters of the separator system can be found in Appendix A.

3.5 Compressor train

This section will present the mathematical background of the compressor train system. All relevant assumptions that has been made will also be listed.

A simplified version of the compressor train system are presented in Figure 3.5. The compressor train system consists of three compressors in series. Due to the pressure difference between the gas coming from the separator and the gas used for gas lift, multiple compressors are introduced to support the necessary pressure rise as described in Section 2.1. As Figure 3.5 shows, the gas enters the first compressor in the compressor train through the inlet valve and into the impeller where it is compressed. A recycle line is modelled for each compressor for further implementation of surge constraints. Each compressor step is modelled equally, and the inlet conditions of a compressor is dependent on the outlet conditions of the previous compressor. The equations of the compressor system will be derived for a single compressor, based on the assumption that each compressor is designed equally. We assume polytropic compression, and to respond to the temperature rise over the compressors, we assume that the resulting heat added over the individual compressor stages is removed.

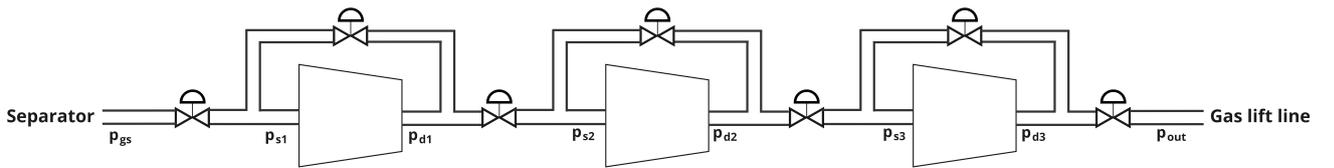


Figure 3.5: Sketch of the simplified Compressor train.

3.5.1 Pressure

Figure 3.5 shows that the compressor train system has two pressure regions for each compressor. These regions are the suction pressure (p_s) and the discharge pressure (p_d). To find the differential change of these pressures with respect to time, a model developed by Greitzer (1976)^[4] and then further developed by Milosavljevic et al. (2020)^[11] can be utilized. This model relates the the time derivative of the pressure to the mass flow in and out of the regions. When relating this to total model, the suction pressure will be dependent on the incoming flow, the recycle flow and the flow through the compressor. The suction pressure rate of each compressor is given in Equation 3.51:

$$\dot{p}_s = C_1(w_{in} + w_{rec} - w_c), \quad (3.51)$$

where w_{in} is the flow of gas either from the separator or the previous compression stage, w_{rec} is the recycled gas flow, w_c is the flow of gas through the compressor and C_1 is a constant parameter determining system dynamics. The discharge pressure rate of change may be obtained similarly. From the total model it can be observed that the discharge pressure rate is dependent on the flow from the compressor, the recycle flow and the flow to the next compressor. The equation for the discharge pressure is can be defined as shown in Equation 3.52:

$$\dot{p}_d = C_2(w_c - w_{out} - w_{rec}), \quad (3.52)$$

where w_{out} is the flow out of the compressor and C_2 is a constant determining system dynamics.

3.5.2 Flow

For each of the compressors in the compressor train there are four different relating mass flows. Due to the serial configuration of the compressors the mass flow out of compressor (i) will equal to the inflow of compressor ($i + 1$). The mass flows, with exception of the mass flow through the compressors, can be found using the valve equation shown in Equation 3.13, assuming laminar flow. The inlet flow of the first compressor will be dependent on the separator gas pressure and the suction pressure of the first compressor. While for the next two compressors the mass flow will be dependent on the discharge pressure of the previous compressor, and the compressors suction pressure. The general equation for the inlet mass flows can be described as presented in Equation 3.53.

$$w_{in} = u_{in}C_{in}\sqrt{\rho_{in}(p_{in} - p_s)}, \quad (3.53)$$

where u_{in} is the valve opening of the inlet valve to the compressors, C_{in} is the valve characteristics of the inlet valve and ρ_{in} is the density of the in flowing gas.

The outlet flow of the compressors are dependent on each compressors discharge pressure and the next compressors suction pressure, with the exception of the third compressor, where the mass flow out will be dependent on the discharge pressure and the pressure in the gas lift line. The general equation for the outlet flows of the compressors are shown in Equation 3.54:

$$w_{out} = u_{out}C_{out}\sqrt{\rho_d(p_d - p_s)}, \quad (3.54)$$

where u_{out} is the valve opening of the outlet valve, C_{out} is the valve characteristics of the outlet valve and ρ_d is the density of the discharge gas. The mass flow of the recirculated gas flow are dependent on each compressors discharge pressure and suction pressure, and can be described as shown in Equation 3.55.

$$w_{rec} = u_{rec}C_{rec}\sqrt{\rho_d(p_d - p_s)}, \quad (3.55)$$

where u_{rec} is the valve opening of the recycle valve and C_{rec} is the valve characteristics of the recycle valve. The mass flow rate of change through the compressors can be formulated

as the difference between the suction pressure times the pressure ratio, which is defined as $\Pi = P_{out}/P_{in}$ and the discharge pressure^[35]. The result of this relationship is that the mass flow through the compressor will stabilize when the compressor manages to compress the gas to the desired discharge pressure. The equation for the mass flow rate of change through the compressor are defined in Equation 3.56.

$$\dot{w}_c = C_3(p_s\Pi - p_d), \quad (3.56)$$

where \dot{w}_c is the mass flow rate of change through the compressor, C_3 is a constant determining system dynamics and Π is the pressure ratio.

3.5.3 Density

The density of the gas through the compressor train will change as the pressure of the gas is increasing. To describe the different density regions, the ideal gas law can be utilized at the different pressure regions of the system. The density relating to the mass flows in of the compressors can be found by looking at the pressure region before the inlet valve of each compressor. Thus the inlet density of the first compressor will be dependent on the separator pressure, while the inlet density of the remaining compressors will depend on the discharge pressure of the previous compressor. The general equation for the inlet density of the compressors is given in Equation 3.57.

$$\rho_{in} = \left(\frac{Mw}{RT_{in}} \right) p_{in}, \quad (3.57)$$

where ρ_{in} is the inlet density, T_{in} is the temperature at the inlet of the compressor. The discharge density can be found for each compressor by the expression shown in Equation 3.58

$$\rho_d = \left(\frac{Mw}{RT_d} \right) p_d, \quad (3.58)$$

where ρ_d is the discharge density and T_d is the discharge temperature of the compressor.

3.5.4 Pressure ratio

The pressure ratio, which for the compressors is the relationship between the inlet pressure and the outlet pressure, can be modelled as a polynomial function of the rotational speed and the mass flow through the compressor. The compressor maps are normally based on manufacturer specification or historical data^[36]. The polynomial function of the pressure ratio is given by Equation 3.59.

$$\Pi = \alpha_1 + \alpha_2\omega + \alpha_3w_c + \alpha_4\omega w_c + \alpha_5\omega^2 + \alpha_6w_c^2, \quad (3.59)$$

where α_i are constant values that describes the form of the polynomial function and ω is the rotational speed of the compressor.

3.5.5 Polytropic head

The polytropic head is a term that can be defined as how many Joules that is needed to compress one kg of gas to the required pressure^[37]. The polytropic head for the compressors can be calculated using Equation 3.60:

$$y_p = \left(\frac{Z_{in}RT_{in}}{M_w} \right) \left(\frac{n_v}{n_v-1} \right) \left(\Pi^{\left(\frac{n_v-1}{n_v} \right)} - 1 \right), \quad (3.60)$$

where y_p is the polytropic head, Z_{in} is the gas compressibility factor and n_v is the polytropic exponent.

3.5.6 Power and efficiency

The polytropic efficiency, which describes the efficiency of the rotating shaft, can according to Cortinovis et al. (2012)^[36] be modelled as a polynomial function of the pressure ratio and the rotational speed. The polynomial function for the polytropic efficiency is given in Equation 3.61:

$$\eta_p = \beta_1 + \beta_2\omega + \beta_3\Pi + \beta_4\omega\Pi + \beta_5\omega^2 + \beta_6\Pi^2, \quad (3.61)$$

where β_i are constant values that describes the form of the polynomial function. When the polytropic head and the polytropic efficiency are defined, the shaft power of each compressor can be found using Equation 3.62:

$$P = \left(\frac{y_p}{\eta_p} \right) w_c, \quad (3.62)$$

where P is the shaft power of the compressor.

3.5.7 Equations and states

The compressor system includes both differential and algebraic states and the corresponding equations to calculate them. We assume that we can manipulate the opening of the control valve at the inlet and outlet of each compressor, and the control valve for the recycle flow. The control valve at the outlet of the previous compressor equals the inlet control valve of the next. The summarized system for each compressor(i) can be described as shown in Equation 4.2:

$$x_{comp} = [p_{s_i}, p_{d_i}, m_{c_i}]^T \quad (3.63a)$$

$$z_{comp} = [w_{in_i}, w_{out_i}, \rho_{in_i}, \rho_{d_i}, \Pi_i, P_{c_i}, y_{p_i}, \eta_{p_i}, w_{rec}]^T \quad (3.63b)$$

$$u_{comp} = [u_{in_i}, u_{out_i}, u_{rec_i}]^T \quad (3.63c)$$

where x_{comp} represents the differential states of the compressor system, z_{comp} represents the algebraic states of the compressor system and u_{comp} represents the control variables of the compressor system. The constant values and parameters of the compressor system can be found in Appendix A.

3.6 Gas lift

This section will present the mathematical background of the gas lift system. All relevant assumptions that has been made will also be listed.

A simplified version of the gas lift system is presented in Figure 3.2. As the figure shows, the compressed gas enters the gas lift line through the outlet valve of the third compressor. The gas then flows through the gas lift line where the gas lift chokes routes the gas to the annulus sections of the wells. The gas lift line is assumed to be a vertical line that stretches from the surface production facilities to the subsea facilities.

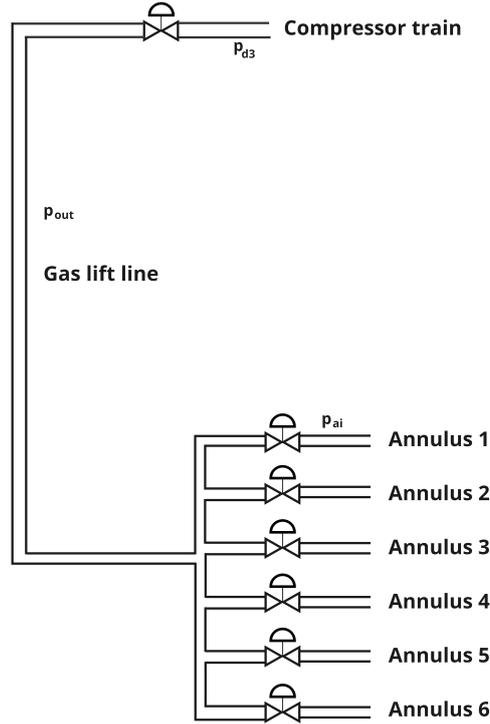


Figure 3.6: Sketch of the simplified gas lift system.

3.6.1 Mass balances

The mass balance of the gas lift line can be found by using the general mass balance shown in Equation 3.1 and information retrieved from Figure 3.2. From the figure it is apparent that the only flow into the system is the flow out of the third compressor. Furthermore, the flows out of this system are defined by the total gas through the gas lift chokes. Therefore, the mass rate of change in the gas lift line can be defined as shown in Equation 3.64:

$$\dot{m}_{gl} = w_{out3} - \sum_{i=1}^6 w_{gl_i}, \quad (3.64)$$

where m_{gl} is the mass rate of change in the gas lift line, w_{out3} is the flow out of the third compressor and w_{gl} is the mass flow of gas to each well.

3.6.2 Pressure

To find the pressure in the gas lift line the ideal gas law can be used. The gas lift line is assumed to be cylindrical. Thus the relation for volume of a cylinder ($V = \pi r^2 L$) and the fact that the number of moles equal the mass divided by the molar mass, can be used to find:

$$p_{out} = \frac{RT_d m_{gl}}{M_w \pi r_{gl}^2 L_{gl}}, \quad (3.65)$$

where p_{out} is the pressure in the gas lift line, r_{gl} is the radius of the gas lift line and L_{gl} is the length of the gas lift line.

3.6.3 Flow

Due to the assumption of uniform pressure throughout the entire pipe section, the mass flow of the gas that is going to be used as lift gas for each well, can be calculated using the valve equation in Equation 3.13. In this equation the outlet pressure of the valve is equal to the annulus pressure, and the inlet pressure is equal to the gas lift line pressure. The mass flow of the gas lift can be defined as shown in Equation 3.64:

$$w_{gl} = c_{gl} u_{gl} \sqrt{\rho_{out} (p_{out} - p_{ai})}, \quad (3.66)$$

where u_{gl} is the valve opening of the gas lift choke, C_{gl} is the valve characteristics of the gas lift choke and ρ_{out} is the density of the gas in the gas lift line.

3.6.4 Density

The density of the gas in the gas lift line can be found from the ideal gas law, and is given in Equation 3.67.

$$\rho_{out} = \left(\frac{M_w}{RT_d} \right) p_{out} \quad (3.67)$$

3.6.5 Equations and states

The gas lift system includes both differential and algebraic states and the corresponding equations to calculate them. We assume that we can manipulate the opening of the gas lift choke valves. The summarized system can be described for each well(i) as shown in Equation 3.68:

$$x_{gl} = [m_{gl}]^T \quad (3.68a)$$

$$z_{gl} = [w_{gl_i}, p_{out}, \rho_{out}]^T \quad (3.68b)$$

$$u_{gl} = [u_{gl_i}]^T \quad (3.68c)$$

where x_{gl} are the differential states of the gas lift system, z_{gl} are the algebraic states of the gas lift system and u_{gl} are the control variables of the gas lift system. The constant values and parameters of the gas lift system can be found in Appendix A.

4 Model implementation

4.1 Modelling strategy

In this section the procedure and methods relating to the development of the model will be described. The first step of modelling the total system consisted of defining the battery limits of multiple smaller parts of the model. As can be observed in Section 3 the total model was divided into five subsystems: a well system, a riser system, a separator system, a compressor system and a gas lift system.

4.1.1 General modelling procedure

The procedure of the modelling consisted of defining the dynamics and the mathematical background of each subsystem. This was based on previous work and the theoretical background of the subsystem. The constant parameters of the system, such as height of wells, were defined from general estimates or previous work and fitted to properly match the model. Furthermore, the system equations were derived from mass and energy balances with respect to the assumptions made. As well as the necessary relations between the regions in the system, as described in Section 3. The model equations were then implemented in Python with the CasADI framework described in Section 2.2. To ensure stability and convergence of the states, the system was further divided into smaller systems by defining variables as constant parameters. This resulted in a new delimited system. The variables and their corresponding equations were then introduced to the model in bulk i.e pressure of annulus and flow through injection valve, due to the dependence. For each introduction of a new equation into the model, the expected value of the new variable was either calculated or estimated. If the necessary data was available, the data were calculated, and if not, the data were estimated based on expected behaviour.

The resulting equation system and the approximated initial values found in Appendix B, were solved with the BDF based integrator IDAS described in Section 2.2.1. The iterative behaviour was then analyzed. If the integration resulted in slower or faster convergence than expected, the dynamic equations of the system were manipulated by multiplying them with a factor, representing the different time scales between the systems. If a system variable didn't converge during the simulation, the integration time was extended, and the initial guess was updated accordingly. If a variable diverged, a PID-controller was introduced and tuned with the SIMC rules described in Section 2.4. The initial guesses were then updated to equal the convergence value.

When the stability and convergence of the system was ensured, the upper and lower boundaries of the variables were defined. The boundaries were either defined from physical limitations or given an arbitrary value that would not affect the solving of the system.

The next step was to define the objective function, relating the oil production to the power consumption. Then inequality constraints were defined based on total capacity. Furthermore, the non-linear program solver IPOPT, which is based on the barrier method described in Section 2.2.2, was implemented to find the optimal value of the states. The initial values were then updated to be equal to the optimal state values. The procedure was repeated until the total subsystem was defined and converged, and the optimizer managed to find a viable solution. When a subsystem was connected to another subsystem, the systems were first implemented separately to verify convergence and expected behaviour. The next step in the merger followed the method of gradually introducing equations of the second system to the first one.

4.1.2 Well system

The well system was modelled after the procedure presented in Section 4.1.1. To be able to define a battery limit for the well system, a temporary compressor was introduced to emulate the compressor system. In addition, the manifold pressure was introduced as a constant parameter. When the general procedure was implemented and one well slot was defined, the model was extended by adding new wells. The model was implemented with a total of six wells to make more decision variables available for further implementation of control strategies. The constant parameters and the overall model were inspired by the model presented by Krishnamoorthy et al. (2016)^[1].

4.1.3 Riser system

The riser system was developed by introducing the wellhead pressure of each well as constant parameters. This decision was based on the knowledge obtained by the well modelling. The separator pressure was assumed to be around 20 bar, following the specification of a medium pressure separator^[38]. With the data obtained from the wells, the approximated flow of oil and gas from the wells could be used to scale the valve, which in turn could give useful information regarding initial values.

4.1.4 Separator system

The separator system was developed by introducing the flow from the integrated well and riser model as a constant inlet parameter. The export pressure of oil and gas was introduced as constant parameters. The system was then solved for the gas phase of the separator by using the general method. Subsequently, the oil phase was introduced and the relations between the phases in the separator introduced. The integration results showed signs of divergence in the oil level, resulting in a PID-controller being introduced to control the level at the optimal value. The controller was tuned with the SIMC rules for integrating processes described in Section 2.4. The tuning parameters used are presented in Table 4.1.

Table 4.1: Tuning parameters separator level controller.

Variable	Value
τ_c	50
τ_I	200
k'	$5.173 \text{ cot } 10^{-6}$
θ	0

When the system was stabilized the general procedure was continued, and the separator model was connected to the main model. The constant parameters and the overall model were inspired by the model presented by Backi et al. (2018)^[3].

4.1.5 Compressor system

The compressor system was developed by introducing the separator and gas lift line pressure as constant parameters. The compressors were gradually introduced to the compressor model by assuming that the surge pressure of the next compressor were a fixed variable. The general method was then used for each introduction of new variables until the total compressor train was described. As described in Section 3.5, the pressure ratio and efficiency was found from a polynomial relationship. Due to the lack of historical data or manufacturer specification, the polynomial constants were found from trial and error, based on expected behaviour. The approximated polynomial variables used in Equation 3.59 and 3.61 are presented in Table 4.3.

Table 4.3: polynomial constants representing the pressure ratio and efficiency of a compressor.

α_1	α_2	α_3	α_4	α_5	α_6
1.8	$9.8 \cot 10^{-3}$	$-1.35 \cdot 10^{-3}$	$2.24 \cdot 10^{-4}$	$2.175 \cdot 10^{-4}$	$-1.1 \cdot 10^{-3}$
β_1	β_2	β_3	β_4	β_5	β_6
74.466	-0.426	0.293	$2.97 \cdot 10^{-3}$	$-2.68 \cdot 10^{-5}$	1.32

The dynamic equations of the compressor system relies on dynamic constants to mimic the dynamic behaviour difference between the different components. The dynamic constants used are presented in Table 4.5.

Table 4.5: Dynamical coefficients of the compressor system.

C_1	C_2	C_3
10^4	10^5	1

The compressor model was then added to the total model. The constant parameters and the overall model were inspired by the model presented by Milosaljevic et al. (2016)^[11].

4.1.6 Gas lift system

The gas lift system was developed by introducing the discharge pressure of the third compressor as a fixed variable, and the annulus pressures of each well as constant parameters for each gas lift line. The optimal gas lift injection rates for each well, supplied by the temporary compressor, were used as initial values for the flows. The valve characteristics of the gas lift chokes were then sized in such a way that the valve opening did not risk saturating when further implementation of control methods were implemented. The final connection with the wells were made with great care, due to the potential issues concerning the integration of recycled systems.

4.2 Integrated model

The final model can be formulated as a semi-explicit DAE on the form displayed in Equation 4.1:

$$\begin{aligned}\dot{x} &= f(x, z, u, p) \\ 0 &= g(x, z, u, p),\end{aligned}\tag{4.1}$$

where the variables \dot{x} , x , z , u and p can be defined from the states of each individual model in the Equations 3.25, 3.34, 3.50, 4.2 and 3.68. We can thus define the total system variables as in Equation 4.2:

$$\dot{x} = [\dot{x}_{well}, \dot{x}_{riser}, \dot{x}_{sep}, \dot{x}_{comp}, \dot{x}_{gl}]^T\tag{4.2a}$$

$$x = [x_{well}, x_{riser}, x_{sep}, x_{comp}, x_{gl}]^T\tag{4.2b}$$

$$z = [z_{well}, z_{riser}, z_{sep}, z_{comp}, z_{gl}]^T\tag{4.2c}$$

$$p = [p_{well}, p_{sep}]^T\tag{4.2d}$$

$$u = [u_{well}, u_{sep}, u_{comp}, u_{gl}]^T\tag{4.2e}$$

The functions f and g from Equation 4.1 represent the differential equations for the \dot{x} variables and the algebraic equations for the z variables derived in Section 3 respectively. The DAE system was solved with the IDAS integrator described in Section 2.2.1. The initial values required by IDAS can be found in Appendix B. The implementation in python can be found in Appendix C.

4.3 Optimization

When the total model was developed, the final non-linear optimization problem could be defined. The objective function was defined with the purpose of maximizing the total oil production, represented by the oil to export, and minimizing the power consumption of the three compressors. The factors multiplied with the variables relate to the price of oil (\$/kg/s) and the price of power (\$/kW). The optimization problem is restricted by the amount of produced gas (10 kg/s), the total amount of gas used for gas lift (6 kg/s) and total power consumption (19 kW). The constraints relate to the capacity of the systems. The system is also constrained by the model equations and upper and lower boundaries of the states and control variables. These are shown in Equation 4.3.

$$\begin{aligned}
\min_{\theta} \quad & -0.6w_{os} + 0.1 \sum_{i=1}^3 P_{c_i} \\
\text{s.t.} \quad & g(\theta) = 0 \\
& f(\theta) = 0 \\
& w_{gs} - w_{gs}^{max} \leq 0 \\
& \sum_{i=1}^6 w_{gl_i} - w_{gl}^{max} \leq 0 \\
& \sum_{i=1}^3 P_{C_i} - P_C^{max} \leq 0 \\
& x^L \leq x \leq x^U \\
& u^L \leq u \leq u^U \\
& z^L \leq z \leq z^U
\end{aligned} \tag{4.3}$$

where θ represents the model states and controlled variables. $g(\theta)$ are the algebraic equations of the model, $f(\theta)$ are the differential equations of the model, w_{gs}^{max} is the maximum produced gas, w_{gl}^{max} is the maximum gas lift, P_C^{max} is the maximum power consumption, x^L and x^U are the lower and upper bounds of the differential states defined in Equation 4.2b. z^L and z^U are the lower and upper bounds of the algebraic states defined in Equation 4.2c, u^L and u^U are the lower and upper bounds of the controlled variables defined in Equation 4.2e. The optimization problem was implemented in CasADI by the use of IPOPT, which is described in Section 2.2.2. The bounds and initial values required by IPOPT can be found in Appendix B. The implementation in python can be found in Appendix C.

5 Results

In this section, all relevant results of the implementation, integration and optimization will be presented and discussed.

5.1 Implementation

As described in Section 4.1.5, the behaviour of the pressure ratio and efficiency was found from a polynomial relationship. Furthermore, it was explained that this relationship is in most cases found either from manufacturer specifications or from historical data. The modelling and implementation in this project doesn't consider any of the mentioned sources of information above. However, it is possible to model this relation by considering general operation limits of the centrifugal compressor presented in Section 2.1. However, this results in major uncertainty and may affect the end results. The assumption that the model operates above the surge constraint will also affect reality of the proposed model. The resulting compressor performance curves for the pressure ratio, efficiency and power plotted for different rotational speeds are found below in Figure 5.1

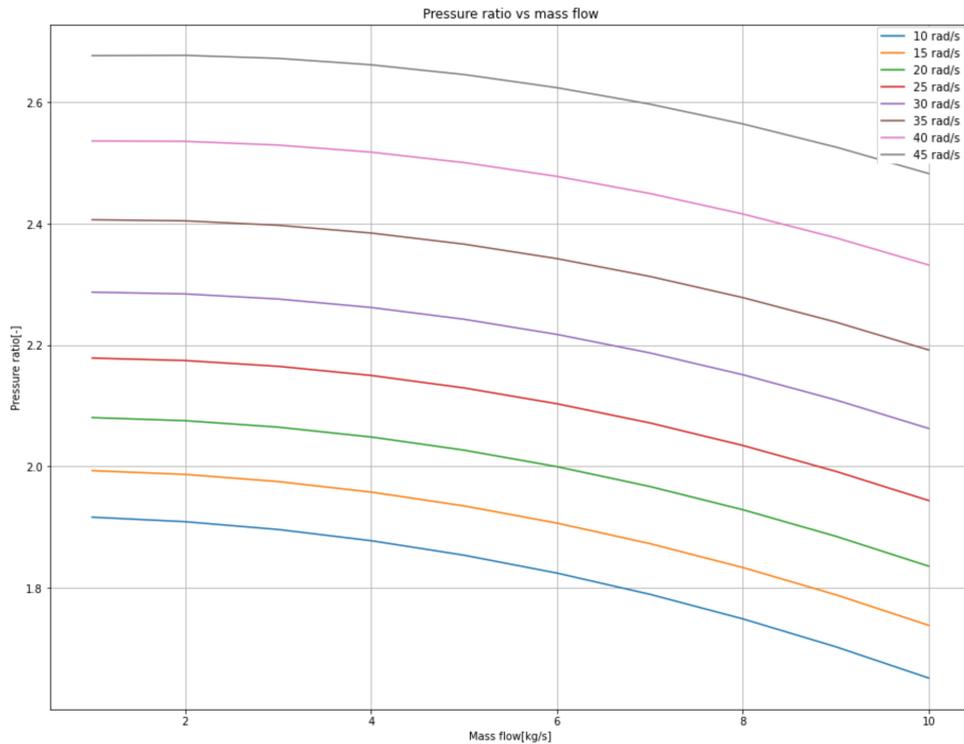


Figure 5.1: Equation 3.59 plotted for different rotational speeds against mass flow through the compressor.

From Figure 5.1 we can observe that the pressure-ratio achieved by the compressor increases with rotational speed, and decreases with inlet mass flowing through the compressor. This is expected from the theory, because the compressor needs to compress more gas with the same speed input, thus resulting in worse performance. The pressure ratio is also located in the normal operation area of a centrifugal compressor as described in Section 2.1.

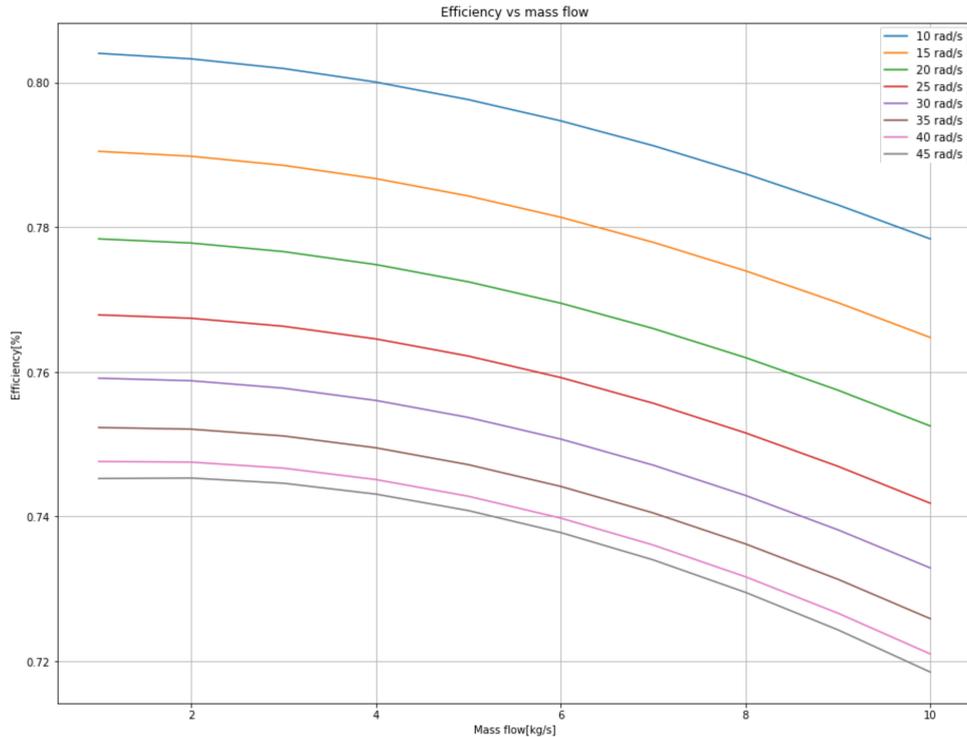


Figure 5.2: Equation 3.61 plotted for different rotational speeds against mass flow through the compressor.

From Figure 5.2 we can observe that the efficiency decreases with higher rotational speed and decreases with input mass flow. For real compressor curves the efficiency would not be this static between the rotational speeds. Furthermore, the efficiency would be different for the rotational speeds at different mass flows i.e higher rad/s would probably be more efficient for larger mass flows than lower rad/s. The optimization problem wants to minimize the power consumption. A consequence of this, is as long as the lower rotational speed delivers an acceptable pressure increase, the optimizer will choose this. From the theory in Section 2.1, we can see that the model operates within the normal efficiency range of centrifugal compressors, which is 0.70 - 0.85.

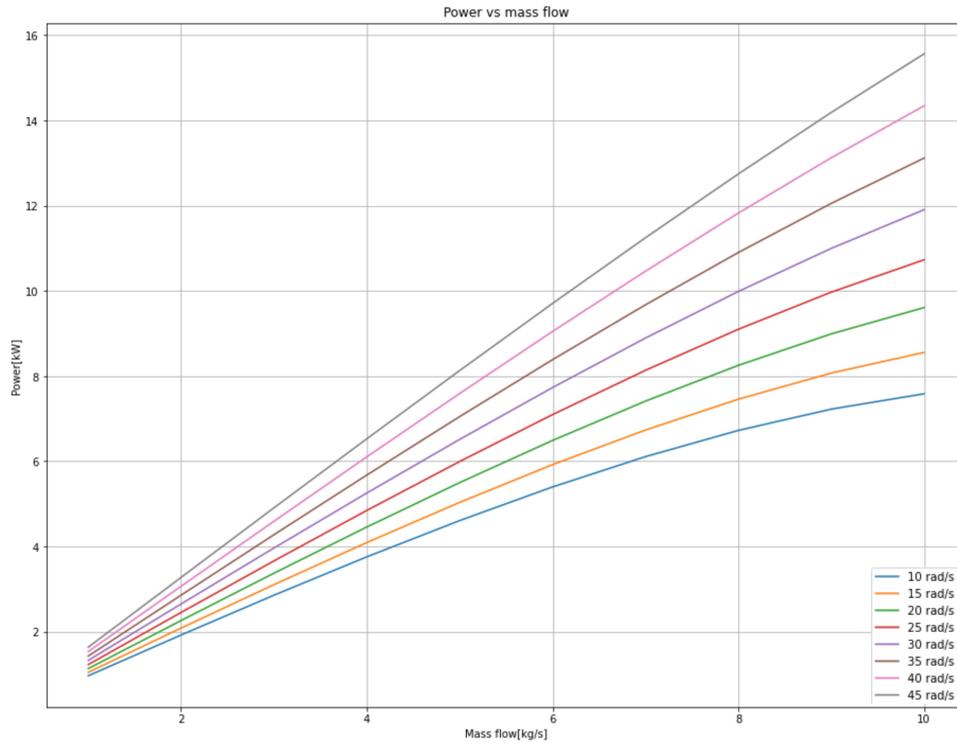


Figure 5.3: Equation 3.62 plotted for different rotational speeds against mass flow through the compressor.

From Figure 5.3 we can observe that the usage of power increases with higher rotational speed and mass flow. This corresponds well with the fact that to achieve higher rotational speeds, more shaft power needs to be introduced.

5.2 Integration

In this section a selection of the integration results will be presented and discussed. The integration results for the remaining variables can be found in Appendix D.

5.2.1 Level control

As mentioned in Section 4.1.4, the height of the oil in the separator did not converge. In response to this, a PID-controller was introduced to control the level at the optimal value. The result of the implementation are presented in Figure 5.4 and Figure 5.5. The total model was integrated with IDAS with a simulation time of 10000 seconds.

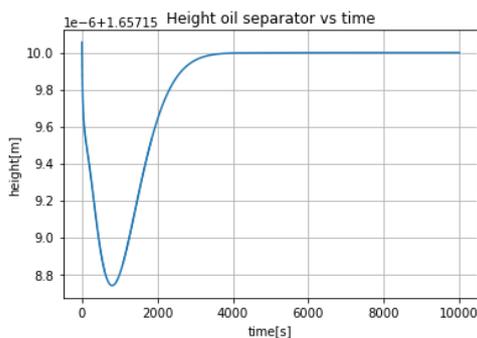


Figure 5.4: Height of oil in separator (h_{ls}).

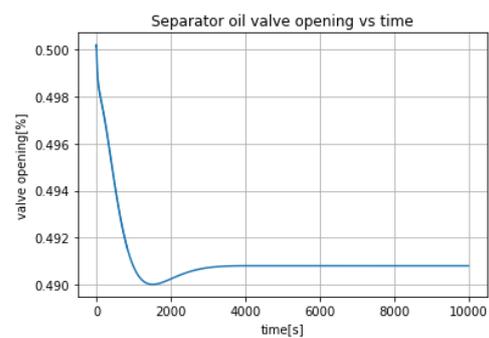


Figure 5.5: Valve opening (u_{ov}).

From Figure 5.4 we can observe that the height of oil in the separator reacts slowly to any changes in the oil output. This can also be observed by the small size of process gain in Section 4.1.4. The most probable explanation for this is that the separator is oversized. Due to the nature of the modelling, the most important part is that the system can be stabilized. For further implementation and comparison with real systems, the model parameters would need to be changed according to the real scenarios specifications. From Figure 5.5 we can observe that the controller gradually decreases the opening of the valve from 0 to 1700 seconds. Then it undershoots slightly before converging to the correct valve opening. From Equation 3.46 we can observe that the level in the separator is dependent on the oil flow from the riser and the outflow of the separator. This relationship can be observed from the dynamic behaviour in the plots, where the oil flow out of the separator is clearly larger than the riser oil flow at the start of the simulation. Thus the controller decreases the valve opening, and the level drops. From the plots it is possible to observe that when the valve opening reaches the desired value before undershooting, the level in the separator starts to increase. This value corresponds to when the inlet and outlet flow are equal, which is essential for level stability.

In Section 3.2.5 we assumed the gas-oil ratio as a potential disturbance to the overall system. Changes in the GOR will result in a change in the composition of the reservoir oil flow. Therefore, if the GOR increases in a reservoir, the amount of gas will increase in relation to the amount of oil. In Figure 5.6 and 5.7, the system is subjected to a drop in the GOR in well 1 (-0.1) after 5000 seconds, and an increase in the GOR in well 2 (+0.1) after 10000 seconds. The system is simulated over 18000 seconds.

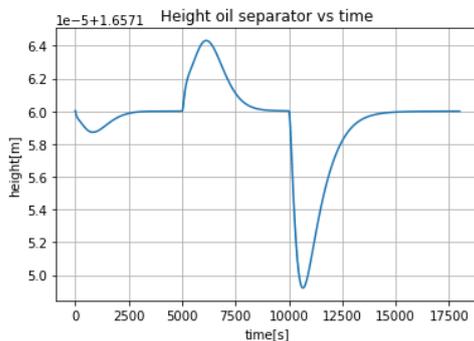


Figure 5.6: Height of oil in separator(h_{ls}).

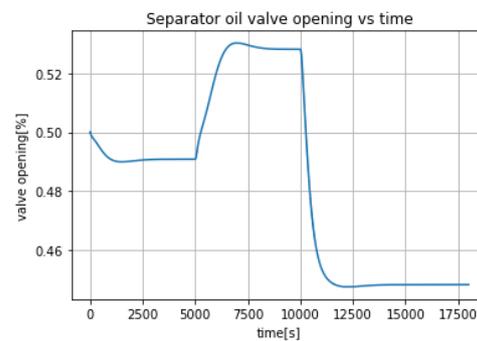


Figure 5.7: Valve opening(u_{ov}).

From the resulting plot in Figure 5.6 we can observe that the level starts increasing as the GOR of well 1 is reduced at $t = 5000s$. The disturbance results in an increase of oil flow from well 1, and naturally an increase of oil flow into the separator. In Figure 5.7 we can observe that the controller counteracts the level change by opening the valve. Furthermore, at $t = 10000s$, the GOR of well 2 is increased, and the oil flow from the well is decreased. The level of oil in the separator drops, and the controller starts closing the valve. The controller manages to stabilize the system after $t = 15000s$.

5.2.2 Model verification

The importance of verifying the model for expected behaviour is of great importance. In Section 5.2.1 we observed that the level changed insignificantly to the change in oil output. This means that the total volume of the separator is too large compared to the amount of oil and gas flowing in. The volume of the separator is about 85 m^3 and as can be observed from the total riser oil

flow in Appendix D, the oil into the separator is approximately 65 kg/s. With the oil density assumed to be around 800 kg/m³, the resulting volume flow of oil is relatively low compared to the total separator volume. As previously mentioned, this will not affect the model in a significant way due to the fact that we are able to control the flow out of the separator at the optimum in such a way that it equals the flow in. This will be the case for a correctly sized separator as well. Due to the high non-linearity of the model which can be observed from the integration results in Appendix D, the modelling error of the size of the separator will be difficult to fix without building the model from the beginning. In a potential comparison study with a real system, the model parameters would need to be changed, and the issues of the slow and oversized separator would be fixed. Furthermore, the controller would be tuned for the new system with the resulting gain. The response to disturbances will be minor, but the relative effect can be analyzed.

As explained in Section 2.1, oil and gas flows from the reservoir due to the pressure difference between the reservoir and the wellhead. This applies to all flowing liquid and gas and can be explained with the second law of thermodynamics, where the system will try to move to thermodynamic equilibrium. Thus, the oil and gas will flow from a high pressure region to a lower pressure region. To validate the behaviour, the different pressure regions can be plotted for one well relating to the total system. The total system was integrated by IDAS with a simulation time of 3000s.

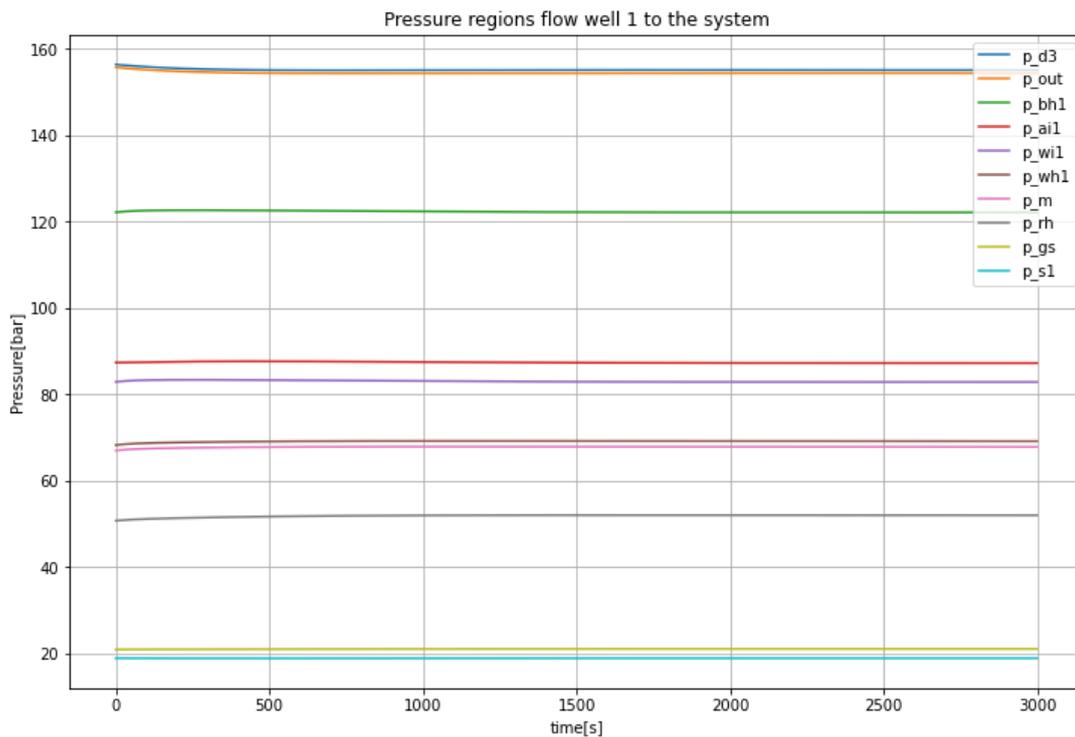


Figure 5.8: Pressure regions in the total system.

Figure 5.8 shows that the discharge pressure of compressor three (p_d3) is the highest pressure region in the system. This is expected due to the specifications of the compressor train. From Figure 3.1 we can observe that the flow from the compressor train (w_out) flows to the gas lift system which can be verified due to the pressure difference between (p_d3) and the gas lift line pressure (p_out). From the gas lift line the gas flows through the gas lift choke and in to the

annulus. From the figure we can observe that (p_{out}) is higher than the annulus pressure of well 1 (p_{ai1}). The gas then flows through the injection valve, into the tubing section at the injection point (p_{wi1}). From the injection point the production fluid and the introduced gas lift continues to the wellhead (p_{wh1}), and further into the manifold (p_m). The oil and gas is then transported to the production facilities by the riser, and as can be observed, the riser head pressure (p_{rh}) is lower than the manifold pressure. From the riser head, the production fluid enters the separator, where the oil and gas is separated. The gas pressure of the separator (p_{gs}) is lower than p_{rh} , so the fluid will flow in the direction of the separator. From the separator, a part of the gas is routed to the compressor train depending on the suction pressure of compressor 1 (p_{s1}). We can also observe that the bottom-hole pressure (p_{bh1}) is higher than p_{iw1} , thus the flow will go in the direction of the injection point. Based on these results, the model possess the expected behaviour of the flow regimes in a real production system.

The gas lift method as explained in Section 2.1, is used to reduce the density of the fluid in the tubing and thus reducing the hydro static pressure in the well. The following plot shows how the flow of oil from well 1 reacts to changes in the valve opening of the corresponding gas lift choke.

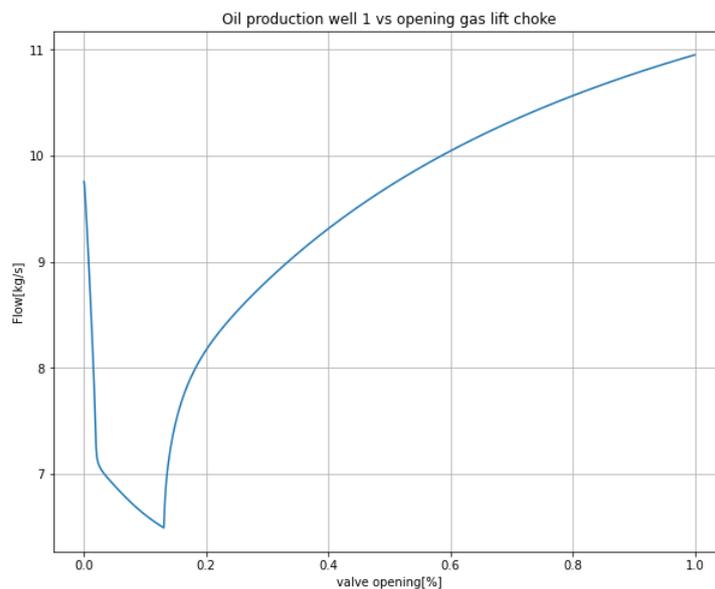


Figure 5.9: The mass flow of from well 1 plotted against the gas lift choke valve opening.

Figure 5.9 shows that the oil flow will increase as more lifting gas is injected into the well tubing. Due to the initial guess of the oil flow not corresponding to a closed valve, the oil flow will immediately decrease due to less gas lift injected. As the valve gradually opens up, the mass flow from the well will increase. The reservoir pressure is assumed as a constant parameter, thus the amount of oil flow from the reservoir will depend on the bottom-hole pressure.

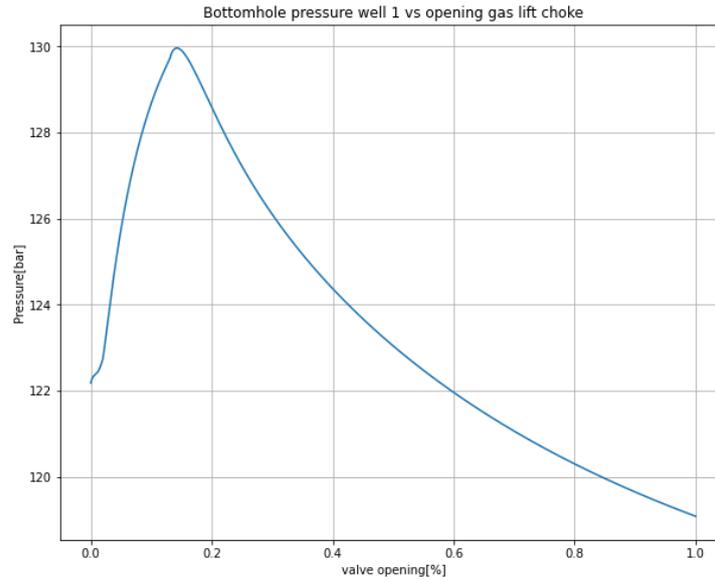


Figure 5.10: Bottom-hole pressure well 1 plotted against the gas lift choke valve opening.

From Equation 3.12 we can observe that the bottom-hole pressure is dependent on the injection point pressure in figure. The increase of the pressure up to $u = 0.15$ is due to the initial guess of the bottom-hole pressure. Furthermore, we can observe that pressure decreases as the valve opens up. This effect leads to an increase in the pressure difference between the reservoir pressure and the bottom-hole pressure, and from Equation 3.19 an increase in the reservoir oil flow.

From the results we can observe that the well produces more oil with increasing gas lift injection. This effect will continue until the hydro static pressure drop can't compensate for the increased friction pressure drop due to increased gas flow in the tubing as explained in Section 2.1. From the plot we can observe that this effect is not active, but the plot shows signs of reduced growth.

The compressor train is a source of uncertainty due to the approximation of the polynomial curve of the pressure ratio and efficiency discussed in Section 5.1. The compressor model was presented in Section 3.5. The assumption made for the compressors, that the compressors operate above surge limit, will affect the model's realism. We also assumed that heat was removed between each compressor stage, which is a normal implementation in real scenarios, due to the risk of overheating and efficiency loss.

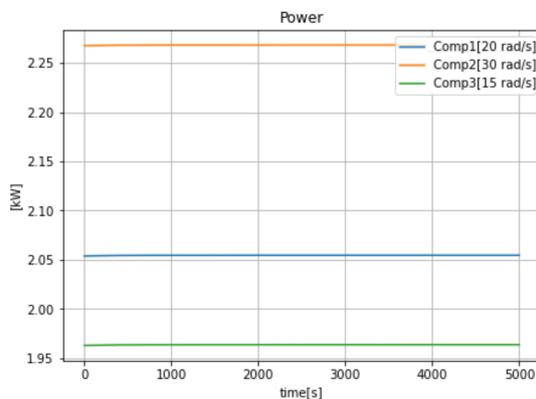


Figure 5.11: Power compressors.

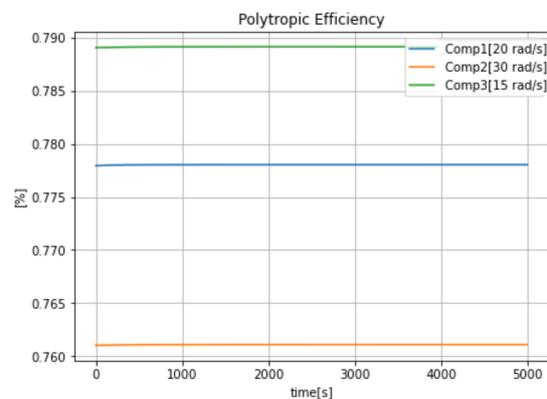


Figure 5.12: Polytopic efficiency compressors.

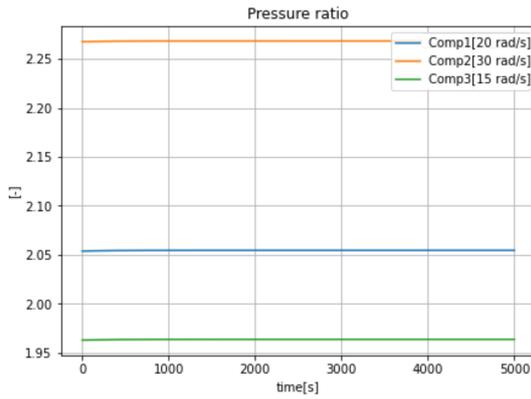


Figure 5.13: Pressure ratio compressors.

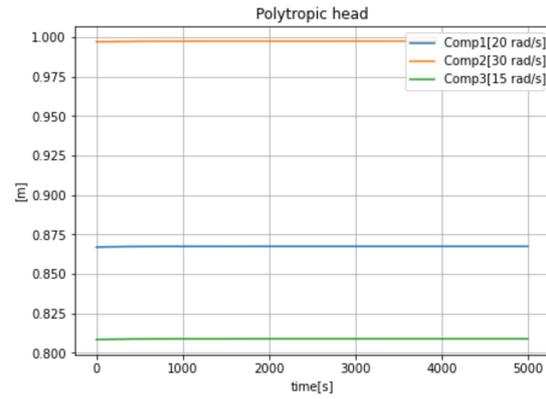


Figure 5.14: Polytropic head compressors.

In Figures 5.11, 5.12, 5.13 and 5.14 the effects of differentiating the speed of the three compressors are presented. The flow through the compressors will be the same due to the serial configuration (4.1 kg/s). By studying the plots we can observe that the variables converge faster than the rest of the system. The reason for being that the compressor train is a faster system than the well and separator systems, and have thus been modelled in that manner. Figure 3.59 shows that the pressure ratio is larger for larger rotational speeds. This is as expected because more work is done on the system, and thus the pressure ratio will be higher. The shaft power of the compressor plotted in Figure 5.11 will also increase for both due to the polytropic head increase and the efficiency reduction as can be observed in Equation 3.62. In Figure 5.14 we observe that the polytropic head increases with rotational speed. From Equation 3.5.5 we can observe that the polytropic head increase with increased pressure ratio. From the equation for shaft power, Equation 3.62, we can observe that as the power increases the efficiency will go down. This corresponds well with Figure 5.12.

5.3 Optimization

When the system was stabilized the system was optimized with the interior point optimizer described in Section 2.2.2. The optimization problem was solved with 15 iterations, due to the continuous update of initial values to fit the optimal convergence. During the modelling, the solver showed signs of high sensitivity to changes in the initial values. Thus, a small change in the input could lead to the problem becoming infeasible. This is due to the non-linearity of the system, which can be observed from the non-linear behaviour of the variables in the integration results Appendix D. The solver is therefore dependent on good quality initial values. This section will present some of the optimization results, and the implications for the further work of implementing control strategies.

For further implementation of control methods it is important that we do not risk saturated valves when controlling the system to optimum. The optimal valve openings are found by solving the optimization problem. The resulting valve openings of the optimized model can be found in Table 5.1.

Table 5.1: Optimal valve openings (0-1).

u_{gl_1}	u_{gl_2}	u_{gl_3}	u_{gl_4}	u_{gl_5}	u_{gl_6}
0.46218	0.305298	0.508811	0.356351	0.355219	0.434294
u_{pc_1}	u_{pc_2}	u_{pc_3}	u_{pc_4}	u_{pc_5}	u_{pc_6}
1	1	1	1	1	1
u_{c_1}	u_{c_2}	u_{c_3}	u_{c_4}	u_{os}	u_{rec1-3}
0.739851	0.593998	0.515123	0.500211	0.472841	$\simeq 0$

From the table we can observe that the gas lift chokes (u_{gl}) optimized values are far from saturation. Thus, they have the opportunity to close or open on demand. The production chokes (u_{pc}) are saturated at a value of 1. The entirely open valves can be explained by the objective function. As long as the production choke valves brings benefit to the objective value they will stay fully open. The compressor valves (u_c) as the gas lift chokes have a safety margin from saturation. The recycle valves (u_{recl}) which are implemented for later implementation of surge constraints are saturated at approximately 0. This is expected due from the perspective of power usage. When the gas is recycled, more gas will need to be compressed and thus more power used.

Another important aspect for further implementation of control strategies, is the active constraint regions. To simulate where the different constraints are active, the variables relating to the inequality constraints were plotted against the potential disturbance (GOR). The GOR values of the wells where assumed equal to simplify the mapping. However, it should be noted that the GOR values won't change simultaneously in a normal scenario. This is because every well is connected to an individual reservoir. The result of the active constraint mapping can be observed in Figure 5.15.

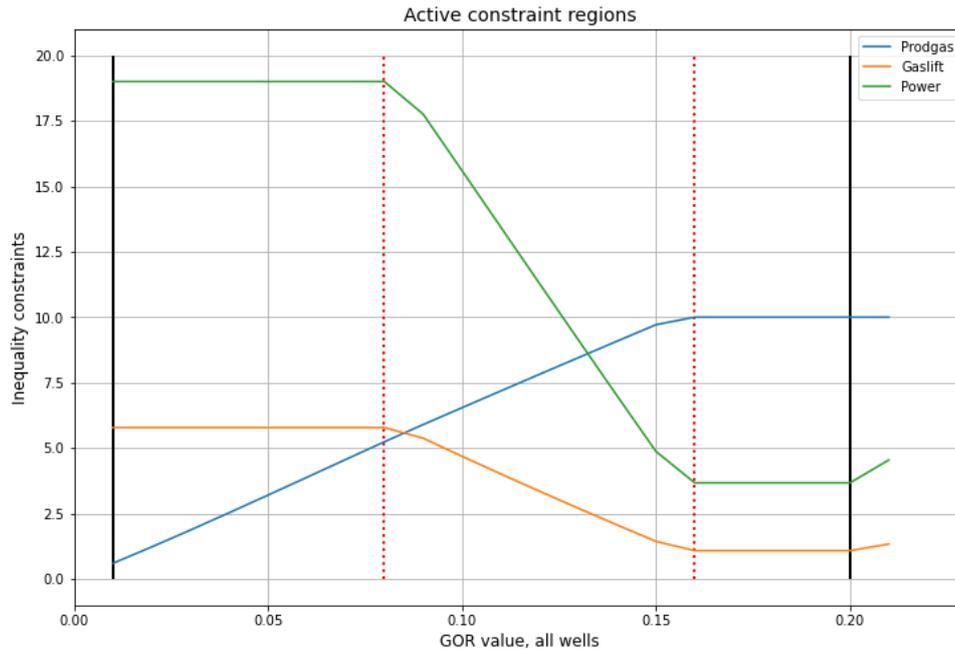


Figure 5.15: Constraint regions related to the GOR values of each well. The black lines indicate the region of feasibility, while the red dotted lines indicate change of region.

From Figure 5.15 we can observe that we have a feasible region from GOR(0.01) to GOR(0.20). Optimizing with values lower or higher than this, resulted in IPOPT converging to a point of local infeasibility. This is due to the solver's sensitivity regarding the initial values for non-linear systems. Furthermore, we have three regions of interest relating to changes in active constraints. As can be observed from the figure, from 0.1 to 0.8 the total power constraint is active. Thus, a manipulated variable should be dedicated to controlling it at this level. The restriction on power usage, leads to a constraint on gas through the compressors. This is expected behaviour, because the compressors can't compress more gas without the introduction of more energy to the system. This in turn leads to the constraint on gas lift becoming obsolete. From 0.08 to 0.16 the problem is unconstrained, meaning that none of the constraints are active. Thus, all the manipulated variables should be used to control the system to optimum. From 0.16 to 0.20 the constraint on produced gas becomes active. From the figure we can observe that this results in a restriction on the power and gas lift. The reason for this correlation, is due to the restriction imposed on the production by the active constraint. The system is therefore not allowed to send more gas to export. Resulting in that the flow through the compressors have no possibility to reduce the amount of gas it processes. The amount of gas through the compressor train won't increase, due to the cost function's objective to reduce power consumption.

From the figure we can also observe that the need for gas lift is more prevalent when the GOR is low. When the GOR is increased, the need for gas lift is reduced. This corresponds well with the notion that the gas lift will be beneficial, until the subsequent friction pressure drop caused by the increase in mass flow of gas, will counteract the benefits.

6 Conclusion

This report has shown that it is possible to model and optimize a recirculated gas lift system. The most important steps for modelling the system are as follows. Assumptions and simplifications based on the aim of the model need to be introduced to make the modelling manageable. Mathematical relationships based on mass and energy balances need to be defined to relate the process variables to each other and simulate expected behaviour. The total system needs to be divided into smaller parts to make sure that the model is stable, due to the non-linearity of the system of mathematical equations. Furthermore, the equations and variables should be introduced gradually to make sure sizing of valves and equipment corresponds to the system dynamics. If a system shows signs of faster or slower convergence than expected relative to another, the system dynamics should be manipulated to mimic the difference in dynamics for the real systems. If the model does not stabilize in a reasonable amount of time, control structures should be implemented to ensure convergence. Initial values need to be estimated by calculation, studying of system behaviour or by engineering intuition. This is due to the optimization problems need for good initial values when solving non-linear problems. When a sub-model is solved, the results should be analyzed to ensure the accuracy of the model. When two subsystems are connected, one should make sure that the outlet properties of the initial system correspond to the inlet conditions of the secondary system.

7 Further work

From the results we can observe that the compressor model was implemented without any specifications or historical data, leading to uncertainty regarding the performance. For further implementation, the compressor curves can be based on the performance of a real compressor to make the model more realistic. The surge-constraints should be implemented. The comparison with a real system also applies to the separator and the other parts of the model.

At this point, the model assumes a two phase flow, constant pressure of the exported oil and gas. To make the model more realistic, three phase flow should be introduced. An oil pump should be modelled for the oil export, and a compressor should be introduced for the gas export.

Methods for feedback-optimizing control (i.e Regional based and Primal-dual control) should be tested on the model, and compared with the numerical results obtained from the optimization.

References

- [1] D. Krishnamoorthy, B. Foss, S. Skogestad, *Processes* **2016**, *4*, DOI 10.3390/pr4040052.
- [2] G. O. Eikrem, O. M. Aamo, B. A. Foss, *SPE Production Operations* **2008**, *23*, 268–279.
- [3] C. J. Backi, B. A. Grimes, S. Skogestad, *Industrial & Engineering Chemistry Research* **2018**, *57*, 7201–7217.
- [4] E. M. Greitzer, *Journal of Engineering for Power* **1976**, *98*, 190–198.
- [5] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, M. Diehl, *Mathematical Programming Computation* **2019**, *11*, 1–36.
- [6] M. Bergounioux, K. Ito, K. Kunisch, *SIAM Journal on Control and Optimization* **1999**, *37*, 1176–1194.
- [7] M. Morari, Y. Arkun, G. Stephanopoulos, *AIChE Journal* **1980**, *26*, 220–232.
- [8] E. Jahanshahi, S. Skogestad, *IFAC Proceedings Volumes* **2011**, *44*, 1634–1639.
- [9] F. Di Meglio, G.-O. Kaasa, N. Petit in Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference, **2009**, pp. 8244–8251.
- [10] A. F. Sayda, J. H. Taylor in 2007 American Control Conference, **2007**, pp. 4847–4853.
- [11] P. Milosavljevic, A. G. Marchetti, A. Cortinovis, T. Faulwasser, M. Mercangöz, D. Bonvin, *Applied Energy* **2020**, *272*, 114883.
- [12] B. Gou, W. C. Lyons, A. Ghalambor, Petroleum production engineering, **2007**.
- [13] B. Hu, *Master of Science Thesis NTNU* **2004**.
- [14] M. P. Boyce in *Gas Turbine Engineering Handbook (Fourth Edition)*, (Ed.: M. P. Boyce), Butterworth-Heinemann, Oxford, **2012**, pp. 253–301.
- [15] G. K. McMillan, Centrifugal and axial compressor control, eng, New York, N.Y. (222 East 46th Street, New York, NY 10017), **2010**.
- [16] J. Ling, K. Wong, S. Armfield, **2007**.
- [17] J. M. Campbell.
- [18] D. J. Gardner, D. R. Reynolds, C. S. Woodward, C. J. Balos, *ACM Transactions on Mathematical Software (TOMS)* **2022**, DOI 10.1145/3539801.
- [19] J. R. Cash in *Encyclopedia of Applied and Computational Mathematics*, (Ed.: B. Engquist), Springer Berlin Heidelberg, Berlin, Heidelberg, **2015**, pp. 97–101.
- [20] J. Andersson, J. Gillis, M. Diehl, Welcome to casadi’s documentation!¶, **2018**.
- [21] A. Wächter, L. Biegler, *Mathematical Programming* **2006**, *106*, Copyright: Copyright 2008 Elsevier B.V., All rights reserved., 25–57.
- [22] A. Wächter in Combinatorial Scientific Computing, (Eds.: U. Naumann, O. Schenk, H. D. Simon, S. Toledo), Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, **2009**, pp. 1–9.
- [23] A. V. Fiacco, G. P. McCormick, *Nonlinear Programming*, Society for Industrial and Applied Mathematics, **1990**, Chapter 3, pp. 39–52.
- [24] J. Crowe, G. Chen, R. Ferdous, D. Greenwood, M. Grimble, H. Huang, J. Jeng, M. A. Johnson, M. Katebi, S Kwong, et al., *PID control: new identification and design methods*, Springer, **2005**.
- [25] J. G. Ziegler, N. B. Nichols, et al., *trans. ASME* **1942**, *64*.
- [26] D. E. Rivera, M. Morari, S. Skogestad, *Industrial & engineering chemistry process design and development* **1986**, *25*, 252–265.
- [27] S. Skogestad, *Journal of process control* **2003**, *13*, 291–309.

- [28] S. Skogestad, C. Grimholt in *PID control in the third millennium*, Springer, **2012**, pp. 147–175.
- [29] E. Gatzke, *Springer Cham* **2022**, 1, 123.
- [30] P. Software, Pipe Pressure Drop Calculations, **2022**, <https://www.pipeflow.com/pipe-pressure-drop-calculations> (visited on 11/21/2022).
- [31] E. T. Gilbert-Kawai, M. D. Wittenberg in *Essential Equations for Anaesthesia: Key Clinical Concepts for the FRCA and EDA*, Cambridge University Press, **2014**, 19–20.
- [32] B. W. Bequette, *Process control: modeling, design, and simulation*, Prentice Hall Professional, **2003**.
- [33] W. C. Lyons in *Working Guide to Reservoir Engineering*, (Ed.: W. C. Lyons), Gulf Professional Publishing, Boston, **2010**, pp. 97–232.
- [34] E. W. Weisstein, "Circular Segment".From MathWorld—A Wolfram Web Resource, **2022**, <https://mathworld.wolfram.com/CircularSegment.html> (visited on 11/30/2022).
- [35] S. Mokhatab, W. A. Poe in *Handbook of Natural Gas Transmission and Processing (Second Edition)*, (Eds.: S. Mokhatab, W. A. Poe), Gulf Professional Publishing, Boston, **2012**, pp. 393–423.
- [36] A. Cortinovis, M. Mercangöz, M. Zovadelli, D. Pareschi, A. De Marco, S. Bittanti, *Computers Chemical Engineering* **2016**, 88, 145–156.
- [37] J. Moore, J. Durham, A. Eijk, E. Karakas, R. Kurz, J. Lesak, M. McBain, P. McCalley, L. Moroz, Z. Mohamed, B. Pettinato, G. Phillippi, H. Watanabe, B. Williams in *Machinery and Energy Systems for the Hydrogen Economy*, (Eds.: K. Brun, T. Allison), Elsevier, **2022**, pp. 333–424.
- [38] Schlumberger, Separator.

A Constant parameters

Table A.1: Constant parameters for the model.

Variable	Unit	Value	
R	$\text{m}^3\text{PaK}^{-1}\text{mol}^{-1}$	8.314	
M_w	kg/mol	0.02	
Riser	L_r	m	500
	H_r	m	500
	D_r	m	0.121
	A_r	m	$\pi \cdot \left(\frac{D_r}{2}\right)^2$
	T_r	K	303
	C_{pr}	m^2	0.003
	ρ_{ro}	kg/m^3	802.5
Separator	m	L_s	10
	r_s	m	1.65
	T_s	K	302
	V_s	m^3	85.53
	C_{gs}	m^2	0.0055
	C_{os}	m^2	0.0006875
	p_{go}	bar	20
	p_{oo}	bar	20
Compressor	C_{gs}	m^2	0.0055
	n_c	-	1
	T_d	K	298
	C_{in}	m^2	0.002637
	T_{in}	K	298
	Z_{in}	-	0.9
	n_v	-	1.27
	C_{out}	m^2	0.001201
	C_{rec}	m^2	0.0000385
Gas lift	L_{gl}	m	500
	r_{gl}	m^2	0.15

B Initial Values and Boundary Conditions

B.1 Dynamic states(x)

Table B.1: Initial values, lower and upper boundaries for the differential states(x).

Variable	Unit	Equipment	Initial value	Lower boundary	Upper boundary
m_{ga}	ton	Well 1	1.60858	0.01	100000000
		Well 2	1.59738	0.01	100000000
		Well 3	1.66102	0.01	100000000
		Well 4	1.64335	0.01	100000000
		Well 5	1.59797	0.01	100000000
		Well 6	1.62939	0.01	100000000
m_{gt}	ton	Well 1	1.07745	0.01	100000000
		Well 2	1.06683	0.01	100000000
		Well 3	1.05517	0.01	100000000
		Well 4	1.05187	0.01	100000000
		Well 5	1.08088	0.01	100000000
		Well 6	1.06824	0.01	100000000
m_{ot}	ton	Well 1	6.06796	0.01	100000000
		Well 2	6.31651	0.01	100000000
		Well 3	6.38467	0.01	100000000
		Well 4	6.62452	0.01	100000000
		Well 5	6.31118	0.01	100000000
		Well 6	6.34885	0.01	100000000
m_{gr}	ton	Riser	0.26819	0.01	100000000
m_{or}	ton		1.59629	0.01	100000000
p_{gs}	bar	Separator	20.9459	0	100000000
h_{ls}	m		1.65716	0	3.30000
p_{s1}	bar	Compressor 1	18.716	0.01	100000000
p_{d1}	bar		38.8325	0.01	100000000
w_{c1}	kg/s		4.04456	0.01	100000000
p_{s2}	bar	Compressor 2	36.9852	0.01	100000000
p_{d2}	bar		76.7379	0.01	100000000
w_{c2}	kg/s		4.04456	0.01	100000000
p_{s3}	bar	Compressor 3	75.4106	0.01	100000000
p_{d3}	bar		156.464	0.01	100000000
w_{c3}	kg/s		4.04456	0.01	100000000
m_{gl}	ton	Gas lift	4.44312	0.01	100000000

B.2 Controlled variables(u)Table B.3: Initial values, lower and upper boundaries for the control variables(u).

Variable	Unit	Equipment	Initial value	Lower boundary	Upper boundary
u_{gl}	-	Well 1	0.515567	0	1
		Well 2	0.36287	0	1
		Well 3	0.570887	0	1
		Well 4	0.422087	0	1
		Well 5	0.412979	0	1
		Well 6	0.494184	0	1
u_{pc}	-	Well 1	1	0	1
		Well 2	1	0	1
		Well 3	1	0	1
		Well 4	1	0	1
		Well 5	1	0	1
		Well 6	1	0	1
u_{ov}	-	Separator	0.5	0	1
u_1	-	Compressor	0.789901	0	1
u_2	-	Compressor	0.637382	0	1
u_3	-	Compressor	0.524895	0	1
u_4	-	Compressor	0.50497	0	1
u_{rec1}	-	Compressor	0	0	1
u_{rec2}	-	Compressor	0	0	1
u_{rec3}	-	Compressor	0	0	1
omega1	rad/s	Compressor	20	20	45
omega2	rad/s	Compressor	20	20	45
omega3	rad/s	Compressor	20	20	45

B.3 Algebraic states(z)

Table B.5: Initial values, lower and upper boundaries for the algebraic states(z).

Variable	Unit	Equipment	Initial value	Lower boundary	Upper boundary
p_{ai}	bar	Well 1	87.4014	0.1	1500000
		Well 2	86.7925	0.1	1500000
		Well 3	90.2506	0.1	1500000
		Well 4	89.2902	0.1	1500000
		Well 5	86.825	0.1	1500000
		Well 6	88.5319	0.1	1500000
p_{wh}	bar	Well 1	68.3118	0.1	700000
		Well 2	68.5671	0.1	700000
		Well 3	68.5333	0.1	700000
		Well 4	68.7875	0.1	700000
		Well 5	68.5411	0.1	700000
		Well 6	68.5365	0.1	700000
p_{wi}	bar	Well 1	82.9172	0.1	1500000
		Well 2	84.5356	0.1	1500000
		Well 3	85.1482	0.1	1500000
		Well 4	86.4296	0.1	1500000
		Well 5	83.9043	0.1	1500000
		Well 6	84.5319	0.1	1500000
p_{bh}	bar	Well 1	122.169	30	1500000
		Well 2	123.789	30	1500000
		Well 3	123.911	30	1500000
		Well 4	125.684	30	1500000
		Well 5	124.138	30	1500000
		Well 6	124.030	30	1500000
ρ_{ai}	$\text{kg}10^{-2}/\text{m}^3$	Well 1	0.698509	0.01	9000000
		Well 2	0.693642	0.01	9000000
		Well 3	0.721280	0.01	9000000
		Well 4	0.713604	0.01	9000000
		Well 5	0.693902	0.01	9000000
		Well 6	0.707544	0.01	9000000

Continues on next page

Variable	Unit	Equipment	Initial value	Lower boundary	Upper boundary
ρ_m	$\text{kg}10^{-2}/\text{m}^3$	Well 1	2.59054	0.01	9000000
		Well 2	2.67293	0.01	9000000
		Well 3	2.69532	0.01	9000000
		Well 4	2.77441	0.01	9000000
		Well 5	2.66950	0.01	9000000
		Well 6	2.68263	0.01	9000000
w_{iv}	kg/s	Well 1	0.7555444	0.01	500000
		Well 2	0.534137	0.01	500000
		Well 3	0.818986	0.01	500000
		Well 4	0.609945	0.01	500000
		Well 5	0.607743	0.01	500000
		Well 6	0.718193	0.01	500000
w_{pc}	kg/s	Well 1	11.47060	0.01	500000
		Well 2	12.76900	0.01	500000
		Well 3	12.67950	0.01	500000
		Well 4	13.9177	0.01	500000
		Well 5	12.65170	0.01	500000
		Well 6	12.66330	0.01	500000
w_{pg}	kg/s	Well 1	1.72964	0.01	500000
		Well 2	1.84501	0.01	500000
		Well 3	1.79830	0.01	500000
		Well 4	1.90709	0.01	500000
		Well 5	1.84996	0.01	500000
		Well 6	1.82382	0.01	500000
w_{po}	kg/s	Well 1	9.74092	0.01	500000
		Well 2	10.924	0.01	500000
		Well 3	10.88120	0.01	500000
		Well 4	12.0106	0.01	500000
		Well 5	10.80180	0.01	500000
		Well 6	10.83950	0.01	500000
<i>Continues on next page</i>					

Variable	Unit	Equipment	Initial value	Lower boundary	Upper boundary
w_{ro}	kg/s	Well 1	9.74092	0.01	500000
		Well 2	10.924	0.01	500000
		Well 3	10.88120	0.01	500000
		Well 4	12.0106	0.01	500000
		Well 5	10.80180	0.01	500000
		Well 6	10.83950	0.01	500000
w_{rg}	kg/s	Well 1	0.974092	0.01	500000
		Well 2	1.31088	0.01	500000
		Well 3	0.979311	0.01	500000
		Well 4	1.29715	0.01	500000
		Well 5	1.24220	0.01	500000
		Well 6	1.10563	0.01	500000
P_{rh}	bar	Riser	50.8006	20	1500000
ρ_r	kg10 ⁻² /m ³		2.15827	0.01	9000000
P_m	bar		67.0421	0.1	1500000
w_{pr}	kg/s		76.1519	0.01	500000
w_{to}	kg/s		65.1981	0.01	500000
w_{tg}	kg/s		10.95380	0.01	500000
w_{os}	kg/s	Separator	65.19810	0.01	500000
w_{gs}	kg/s		6.90926	0.01	500000
ρ_{gs}	kg10 ⁻² /m ³		0.166844	0.01	9000000
P_{os}	bar		21.0763	0.01	1500000
V_{os}	m ³		42.7694	0.0	85
V_{gs}	m ³		42.7694	0.0	85
w_{gl}	kg/s	Gas lift-Well 1	0.7555444	0.01	500000
		Gas lift-Well 2	0.534137	0.01	500000
		Gas lift-Well 3	0.818986	0.01	500000
		Gas lift-Well 4	0.609945	0.01	500000
		Gas lift-Well 5	0.607753	0.01	500000
		Gas lift-Well 6	0.718193	0.01	500000
P_{out}	bar	Gas lift	155.733	0.01	500000
ρ_{out}	kg10 ⁻² /m ³		1.25715	0.01	500000

Continues on next page

Variable	Unit	Equipment	Initial value	Lower boundary	Upper boundary
w_{in}	kg/s	Compressor 1	4.04456	0.01	500000
w_{out}	kg/s		4.04456	0.01	500000
ρ_{in}	kg10 ⁻² /m ³		0.169084	0.01	9000000
ρ_d	kg10 ⁻² /m ³		0.313472	0.01	9000000
Π	-		2.07483	0	3
Power	kW		4.92105	0	100
y_p	m		0.880277	0	1
n_p	%		0.723489	0	100
w_{rec}	kg/s		0.00001	0	500000
w_{in}	kg/s	Compressor 2	4.04456	0.01	500000
w_{out}	kg/s		4.04456	0.01	500000
ρ_{in}	kg10 ⁻² /m ³		0.313472	0.01	9000000
ρ_d	kg10 ⁻² /m ³		0.619461	0.01	9000000
Π	-		2.07483	0	3
Power	kW		4.92105	0	100
y_p	m		0.880277	0	1
n_p	%		0.723489	0	100
w_{rec}	kg/s		0.00001	0	500000
w_{in}	kg/s	Compressor 3	4.04456	0.01	500000
w_{out}	kg/s		4.04456	0.01	500000
ρ_{in}	kg10 ⁻² /m ³		0.619461	0.01	9000000
ρ_d	kg10 ⁻² /m ³		1.26304	0.01	9000000
Π	-		2.07483	0	3
Power	kW		4.92105	0	100
y_p	m		0.880277	0	1
n_p	%		0.723489	0	100
w_{rec}	kg/s		0.00001	0	500000

C Python-code

The python files included in this project consists of the mathematical model presented in Section 3. The code was implemented with the CasADI framework described in Section 2.2. The code consists of three separate files. The parameter file implements a function returning a dictionary of all the constant parameters in the model. The simulator file constructs the system of differential-algebraic equations and initializes the IDAS integrator described in Section 2.2.1. The main file retrieves initial values, upper bounds and lower bounds from excel files listed in Appendix A and B. Based on the data and the function returned from the simulation file the code then integrates the system. The main file also implements the nlp solver IDAS described in Section 2.2.2 and solves the optimization problem. The code is the modified and improved version of the previous work done by Risvan Dirza.

C.1 Parameter file

```

1 #Parameter file
2 #Function returns a dictionary with the models constant parameters
3
4 import numpy as np
5
6 def Params_6wells():
7     par = {} #Dictionary to store the parameters
8     par['n_w'] = 6 #Number of wells
9
10     ##### Well Parameters #####
11
12     #Length, height and diameter of wells[m]
13     par['L_w'] = np.array([1500, 1500, 1500, 1500, 1500, 1500])
14     par['H_w'] = np.array([1000,1000,1000,1000,1000,1000])
15     par['D_w'] = np.array([0.121,0.121,0.121,0.121,0.121,0.121])
16
17     #Length, height and diameter of bottom hole[m]
18     par['L_bh'] = np.array([500, 500, 500, 500, 500, 500])
19     par['H_bh'] = np.array([500, 500, 500, 500, 500, 500])
20     par['D_bh'] = np.array([0.121,0.121,0.121,0.121,0.121,0.121])
21
22     #Length, height and diameter of annuluses[m]
23     par['L_a'] = par['L_w'] # Lenght of annuls equals length of well
24     par['H_a'] = par['H_w'] # Height of annuls equals length of well
25     par['D_a'] = np.array([0.189, 0.189, 0.189, 0.189, 0.189, 0.189])
26
27     #Density oil, injection valve char and production choke valve char
28     par['rho_o'] = np.array([8,8,7.9,8,8.2,8.05]) *1e2 #[kg/m^3]
29     par['C_iv'] = np.array([0.1e-3,0.1e-3,0.1e-3,0.1e-3,0.1e-3,0.1e-3]) #[m^-2]
30     par['C_pc'] = np.array([2e-3,2e-3,2e-3,2e-3,2e-3,2e-3]) #[m^-2]
31
32     #Gas-oil ratio of wells[kg/kg], possible disturbance
33     par['GOR'] = np.array([0.1,0.12,0.09,0.108,0.115,0.102])
34
35     par['p_res'] = np.array([150,155,155,160,155,155]) #Reservoir pressure[bar]
36     par['PI'] = np.array([7,7,7,7,7,7])* 0.5 #Productvity index wells[kg s^-1
37     bar^-1]
38     par['T_a'] = np.array([273, 273, 273, 273, 273, 273]) + 28 #Annulus
39     temperature[K]
40     par['T_w'] = np.array([273, 273, 273, 273, 273, 273]) + 32 #Well temperature
41     [K]
42
43     #Area of well, bottom hole and volume of annulus
44     par['A_w'] = np.pi*(par['D_w']/2)**2 #[m^2]
45     par['A_bh'] = np.pi*(par['D_bh']/2)**2 #[m^2]
46     par['V_a'] = par['L_a']*(np.pi*(par['D_a']/2)**2 - np.pi*(par['D_w']/2)**2)
47     #[m^3]
48     #Volume of annulus will equal the area of the total well and annulus minus
49     the well

```

```

46
47 #Constraints
48 par['wmax_gl'] = np.array([5]) #Max gas lift
49 par['wmax_pg'] = np.array([7]) #Max produced gas
50 par['Powmax_glcom'] = np.array([18]) #Max power
51
52 #General parameters
53 par['R'] = 8.314 #Gas constant [m^3 Pa K^-1 mol^-1]
54 par['Mw'] = 20e-3 #Molar weighgt kg/mol
55 par['tf'] = 1 #Simulation time
56 par['mu_oil'] = 0.001 #Oil viscosity[kg m^-1 s^-1 ]
57
58
59 ##### Riser System #####
60 par['L_r'] = 500 #Length of riser[m]
61 par['H_r'] = 500 #Height of riser[m]
62 par['D_r'] = 0.121 #Diameter of riser[m]
63 A_r = np.pi*(par['D_r']/2)**2
64 par['A_r'] = A_r #Area of riser[m^2]
65 par['T_r'] = 30+273 #Temperature riser[K]
66 par['C_pr'] = 0.003 #Valve char riser valve[m^2]
67 rho_ro = np.sum(par['rho_o'])/6
68 par['rho_ro'] = rho_ro #Density of oil in riser[kg/m^3]
69
70
71 ##### Separator #####
72 par['L_s'] = 10 #length Separator[m] #Oversized
73 par['r_s'] = 1.65 #radius Separator[m] #Oversized
74 par['T_s'] = 29 + 273 #Temperature Separator[K]
75 V_sep = np.pi * par['r_s']**2 * par['L_s']
76 par['V_s'] = V_sep #Volume of Separator[m3]
77 par['C_gs'] = 5.5*0.001 #Valve char gas outlet[m^2]
78 par['C_os'] = 5.5*0.001*0.5*0.5 #Valve char oil outlet[m^2]
79 par['p_go'] = 20 #pressure gas out[bar]
80 par['p_oo'] = 20 #pressure oil out[bar]
81
82 ##### Compressors #####
83 par['n_c'] = 1 #This parameter is just an early implementation error.
84 par['T_d'] = 298 #Temperature out of compessor(assume heat is removed)[K]
85 par['C_in'] = 9e-4*2.93 #Valve char is equal for all in/out valves[m^2]
86 par['T_in'] = 298 #Temperature inlet compressors[K]
87 par['Z_in'] = 0.9 #Compression factor(difference from ideal behaviour)[-]
88 par['n_v'] = 1.27 #Polytropic coefficient[-]
89 par['C_out'] = 1.201e-3 #Valve char out, not used at this impelementation[m^2]
90 par['C_rec'] = 1.1*3.5e-5 #Recycle valve char[m^2]
91
92 #alpha values for the approximation of pressure ratio
93 par['alpha_1'] = 1.05 * 0.745 *2.3
94 par['alpha_2'] = 0.7 * -1.4e-2 *-1
95 par['alpha_3'] = 0.3 * 0.11 * -4.09e-2
96 par['alpha_4'] = 1.75 * 0.13* 9.86e-4
97 par['alpha_5'] = 1.0 * 0.5* -4.25e-4 *-1
98 par['alpha_6'] = 300* (-0.15)* 2.45e-5 *2
99
100 #beta values for the approximation of efficiency
101 par['beta_1'] = 0.7* 9*5.91e-2 *200
102 par['beta_2'] = -2.13e-1 *2
103 par['beta_3'] = 2.93e-1
104 par['beta_4'] = 2.97e-3
105 par['beta_5'] = -2.68e-5
106 par['beta_6'] = -1.1e1 *(-0.1)*1.2 *2
107
108 #Dynamic coefficients for compressor dynamic equations
109 par['Coef_1'] = 1e4
110 par['Coef_2'] = 1e5
111 par['Coef_3'] = 1
112
113 #gamma values for further implementation of surge and choke constraints
114 par['gamma_1'] = 1

```

```
115 par['gamma_2'] = 1
116 par['gamma_3'] = 1
117
118 ##### Gas lift #####
119 par['L_gl'] = 500 #Length gas lift line[m]
120 par['r_gl'] = 0.15 #radius gas lift line[m]
121 par['C_gl'] = np.array([5e-5,5e-5,5e-5,5e-5,5e-5,5e-5]) #Valve char gas lift
    valves[m^2]
122 par['C_iv'] = np.array([0.1e-3,0.1e-3,0.1e-3,0.1e-3,0.1e-3,0.1e-3]) * 1.35
    #Valve char injection valves[m^2]
123
124 return par
```

C.2 Simulator file

```

1 #Simulation file
2 #Constructs the integrator
3
4
5 import numpy as np
6 from casadi import *
7
8 def CentralizedSimulator_F(par):
9
10     ## Retriving the parameters from Param function ##
11
12     #Wells
13     n_w = par['n_w']
14     L_w = par['L_w']
15     H_w = par['H_w']
16     D_w = par['D_w']
17     L_bh = par['L_bh']
18     H_bh = par['H_bh']
19     D_bh = par['D_bh']
20     L_a = par['L_a']
21     H_a = par['H_a']
22     D_a = par['D_a']
23     rho_o = par['rho_o']
24     C_iv = par['C_iv']
25     C_pc = par['C_pc']
26     mu_oil = par['mu_oil']
27     A_w = par['A_w']
28     A_bh = par['A_bh']
29     V_a = par['V_a']
30     p_res = MX.sym('p_res', n_w)
31     PI = MX.sym('PI', n_w)
32     T_a = MX.sym('T_a', n_w)
33     T_w = MX.sym('T_w', n_w)
34     R = par['R']
35     Mw = par['Mw']
36
37     #Riser
38     T_r = par['T_r']
39     L_r = par['L_r']
40     A_r = par['A_r']
41     H_r = par['H_r']
42     D_r = par['D_r']
43     C_pr = par['C_pr']
44     rho_ro = par['rho_ro']
45
46     #Separator
47     L_s = par['L_s']
48     r_s = par['r_s']
49     T_s = par['T_s']
50     C_gs = par['C_gs']
51     C_os = par['C_os']
52     v_s = par['V_s']
53
54
55     #Compressor
56     n_c = par['n_c']
57     C_in = par['C_in']
58     C_out = par['C_out']
59     C_rec = par['C_rec']
60     T_in = par['T_in']
61     T_d = par['T_d']
62     Z_in = par['Z_in']
63     n_v = par['n_v']
64
65     #alphas
66     alpha_1 = par['alpha_1']
67     alpha_2 = par['alpha_2']
68     alpha_3 = par['alpha_3']
69     alpha_4 = par['alpha_4']
70     alpha_5 = par['alpha_5']

```

```

70 alpha_6 = par['alpha_6']
71 #beta
72 beta_1 = par['beta_1']
73 beta_2 = par['beta_2']
74 beta_3 = par['beta_3']
75 beta_4 = par['beta_4']
76 beta_5 = par['beta_5']
77 beta_6 = par['beta_6']
78 #gammas(Further implementation)
79 gamma_1 = par['gamma_1']
80 gamma_2 = par['gamma_2']
81 gamma_3 = par['gamma_3']
82 #Dynamic Coefficients for compressor dynamic equations
83 Coef_1 = par['Coef_1']
84 Coef_2 = par['Coef_2']
85 Coef_3 = par['Coef_3']
86
87 #Gaslift
88 C_iv = par['C_iv']
89 C_gl = par['C_gl']
90 L_gl = par['L_gl']
91 r_gl = par['r_gl']
92
93
94 #Differential states
95 #Well system
96 m_ga = MX.sym('m_ga',n_w) #Mass gas in annulus[ton]
97 m_gt = MX.sym('m_gt',n_w) #Mass gas in tubing[ton]
98 m_ot = MX.sym('m_ot',n_w) #Mass oil in tubing[ton]
99 #Riser
100 m_gr = MX.sym('m_gr',1) #Mass gas in riser[ton]
101 m_or = MX.sym('m_or',1) #Mass oil in riser[ton]
102 #Separator
103 p_gs = MX.sym('p_gs',1) #Pressure of gas in separator[bar]
104 h_ls = MX.sym('h_ls',1) #Height of oil in separator[bar]
105 #Compressor 1
106 p_s1 = MX.sym('p_s1',n_c) #Suction Pressure Gas lift Compressor 1[bar]
107 p_d1 = MX.sym('p_d1',n_c) #Discharge Pressure Gaslift Compressor 1[bar]
108 w_c1 = MX.sym('w_c1',n_c) #Gas massflow rate Gas-lift Compressor 1[kg/s]
109 #Compressor 2
110 p_s2 = MX.sym('p_s2',n_c) #Suction Pressure Gas lift Compressor 2[bar]
111 p_d2 = MX.sym('p_d2',n_c) #Discharge Pressure Gas lift Compressor 2[bar]
112 w_c2 = MX.sym('w_c2',n_c) #Gas massflow rate Gas lift Compressor 2[kg/s]
113 #Compressor 3
114 p_s3 = MX.sym('p_s3',n_c) #Suction Pressure Gas lift Compressor 3[bar]
115 p_d3 = MX.sym('p_d3',n_c) #Discharge Pressure of Gas lift Compressor 3[bar]
116 w_c3 = MX.sym('w_c3',n_c) #Gas massflow rate in Gas-lift Compressor 3[kg/s]
117 #Gas Lift
118 m_gl = MX.sym('m_gl',1) #Mas gas in gas line[ton]
119
120
121 #Algebraic states
122 #Well
123 p_ai = MX.sym('p_ai',n_w) #Annulus pressure at injection[bar]
124 p_wh = MX.sym('p_wh',n_w) #Wellhead pressure[bar]
125 p_wi = MX.sym('p_wi',n_w) #Injection point pressure in tubing[bar]
126 p_bh = MX.sym('p_bh',n_w) #Bottom-hole pressure[bar]
127 rho_ai = MX.sym('rho_ai',n_w) #Density of gas annulus injection point[bar]
128 rho_m = MX.sym('rho_m',n_w) #Density mixed oil/gas in tubing[kg/m^3]
129 w_iv = MX.sym('w_iv',n_w) #Flow gas through injection valve[kg/s]
130 w_pc = MX.sym('w_pc',n_w) #Flow through production choke[kg/s]
131 w_pg = MX.sym('w_pg',n_w) #Flow gas through production choke[kg/s]
132 w_po = MX.sym('w_po',n_w) #Flow oil through production choke[kg/s]
133 w_ro = MX.sym('w_ro',n_w) #Flow oil from reservoir[kg/s]
134 w_rg = MX.sym('w_rg',n_w) ##Flow gas from reservoir[kg/s]
135 #Riser
136 p_rh = MX.sym('p_rh', 1) #Pressure riser head[bar]
137 rho_r = MX.sym('rho_r',1) #density oil/gas riser[kg/m^3]
138 p_m = MX.sym('p_m', 1) #Manifold pressure[bar]
139 w_pr = MX.sym('w_pr', 1) #Flow through riser valve[kg/s]

```

```

140 w_to = MX.sym('w_to', 1) #Flow oil through riser valve[kg/s]
141 w_tg = MX.sym('w_tg', 1) #Flow gas through riser valve[kg/s]
142 #Separator
143 w_os = MX.sym('w_os', 1) #Produced oil out of separator[kg/s]
144 w_gs = MX.sym('w_gs', 1) #Produced gas out of separator[kg/s]
145 rho_gs = MX.sym('rho_gs', 1) #Gas density in separator[kg/m^3]
146 p_os = MX.sym('p_os', 1) #Separator oil pressure[bar]
147 v_os = MX.sym('v_os', 1) #Volume of oil in separator[m^3]
148 v_gs = MX.sym('v_gs', 1) #Volume of gas in separator[m^3]
149 #Compressor 1
150 w_in1 = MX.sym('w_in1', n_c) #Flow gas in compressor 1[kg/s]
151 w_out1 = MX.sym('w_out1', n_c) #Flow gas out compressor 1[kg/s]
152 rho_in1 = MX.sym('rho_in1', n_c) #Density gas in compressor 1[kg/m^3]
153 rho_d1 = MX.sym('rho_d1', n_c) #Density gas out compressor 1[kg/m^3]
154 Phi1 = MX.sym('Phi1', n_c) #Pressure Ratio compressor 1[-]
155 Pow1 = MX.sym('Pow1', n_c) #Power consumption compressor 1[kW]
156 y_p1 = MX.sym('y_p1', n_c) #Polytropic Head compressor 1[m]
157 n_p1 = MX.sym('n_p1', n_c) #Polytropic Efficiency 1[%]
158 w_rec1 = MX.sym('w_rec1', n_c) #Recycle mass flow[kg/s]
159 #Further implementation
160 Phi_max1 = MX.sym('Phi_max1', n_c) #Max Pressure ratio
161 gamma_2_dummy1 = MX.sym('gamma_2_dummy1', n_c) #constraint
162 #Compressor 2
163 w_in2 = MX.sym('w_in2', n_c) #Flow gas in compressor 2[kg/s]
164 w_out2 = MX.sym('w_out2', n_c) #Flow gas out compressor 2[kg/s]
165 rho_in2 = MX.sym('rho_in2', n_c) #Density gas in compressor 2[kg/m^3]
166 rho_d2 = MX.sym('rho_d2', n_c) #Density gas out compressor 2[kg/m^3]
167 Phi2 = MX.sym('Phi2', n_c) #Pressure Ratio compressor 2[-]
168 Pow2 = MX.sym('Pow2', n_c) #Power consumption compressor 2[kW]
169 y_p2 = MX.sym('y_p2', n_c) #Polytropic Head compressor 2[m]
170 n_p2 = MX.sym('n_p2', n_c) #Polytropic Efficiency 2[%]
171 w_rec2 = MX.sym('w_rec2', n_c) #Recycle mass flow 2[kg/s]
172 #Further implementation
173 Phi_max2 = MX.sym('Phi_max2', n_c) #Max pressure ratio
174 gamma_2_dummy2 = MX.sym('gamma_2_dummy2', n_c) #constraint
175 #Compressor 3
176 w_in3 = MX.sym('w_in3', n_c) #Flow gas in compressor 2[kg/s]
177 w_out3 = MX.sym('w_out3', n_c) #Flow gas out compressor 2[kg/s]
178 rho_in3 = MX.sym('rho_in3', n_c) #Density gas in compressor 2[kg/m^3]
179 rho_d3 = MX.sym('rho_d3', n_c) #Density gas out compressor 2[kg/m^3]
180 Phi3 = MX.sym('Phi3', n_c) #Pressure Ratio compressor 2[-]
181 Pow3 = MX.sym('Pow3', n_c) #Power consumption compressor 2[kW]
182 y_p3 = MX.sym('y_p3', n_c) #Polytropic Head compressor 2[m]
183 n_p3 = MX.sym('n_p3', n_c) #Polytropic Efficiency 2[%]
184 w_rec3 = MX.sym('w_rec3', n_c) #Recycle mass flow 2[kg/s]
185 #Further implementation
186 Phi_max3 = MX.sym('Phi_max3', n_c) #Max pressure ratio
187 gamma_2_dummy3 = MX.sym('gamma_2_dummy3', n_c) #constraint
188 #Gl system
189 w_gl = MX.sym('w_gl', n_w) #Flow through gas lift choke[kg/s]
190 p_out = MX.sym('p_out', 1) #Pressure in gas lift line[bar]
191 rho_out = MX.sym('rho_out', 1) #density of gas in gas lift line[kg/m^3]
192
193
194 #Control input
195 #Wells
196 u_pc = MX.sym('u_pc', n_w) #Valve opening production chokes[0-1]
197 #Gas lift
198 u_gl = MX.sym('u_gl', n_w) #Valve opening gas lift chokes[0-1]
199 #Separator
200 z_ov = MX.sym('z_ov', 1) #Valve opening separator oil out[0-1]
201 #Compressor
202 u_1 = MX.sym('u_1', n_c) #Valve opening inlet compressor 1[0-1]
203 u_2 = MX.sym('u_2', n_c) #Valve opening inlet compressor 2, outlet 1[0-1]
204 u_3 = MX.sym('u_3', n_c) #Valve opening inlet compressor 3, outlet 2[0-1]
205 u_4 = MX.sym('u_4', n_c) #Valve opening outlet compressor 3[0-1]
206 u_rec1 = MX.sym('u_rec1', n_c) #Valve opening recycle compressor 1[0-1]
207 u_rec2 = MX.sym('u_rec2', n_c) #Valve opening recycle compressor 2[0-1]
208 u_rec3 = MX.sym('u_rec3', n_c) #Valve opening recycle compressor 3[0-1]
209 omega1 = MX.sym('omega1', n_c) #Speed of compressor 1[rad/s]

```

```

210 omega2 = MX.sym('omega2',n_c) #Speed of compressor 2[rad/s]
211 omega3 = MX.sym('omega2',n_c) #Speed of compressor 3[rad/s]
212
213 #Parameters(Possible to introduce more)
214 #Wells
215 GOR = MX.sym('GOR',n_w)
216 #Separator
217 p_go = MX.sym('p_go',1)
218 p_oo = MX.sym('p_oo', 1)
219
220 #Constraints
221 wmax_gl = MX.sym('wmax_gl',1)
222 wmax_pg = MX.sym('wmax_pg',1)
223 Powmax_glcom = MX.sym('Powmax_glcom',1)
224
225
226
227 #Algebraic equations
228 #Wells
229 f1 = -p_ai*1e5 + ((R*T_a/(V_a*Mw) + 9.81*H_a/V_a)*m_ga*1e3) #Bernoulli
230 f2 = -p_wh*1e5 + ((R*T_w/Mw)*(m_gt*1e3/(L_w*A_w + L_bh*A_bh - m_ot*1e3/rho_o
))) - ((m_gt*1e3+m_ot*1e3)/(L_w*A_w))*9.81*H_w/2 #Bernoulli
231 f3 = -p_wi*1e5 + (p_wh*1e5 + 9.81/(A_w*L_w)*fmax(0,(m_ot*1e3+m_gt*1e3-rho_o*
L_bh*A_bh))*H_w + 128*mu_oil*L_w*w_pc/(3.14*D_w**4*((m_gt*1e3 + m_ot*1e3)*
p_wh*1e5*Mw*rho_o)/(m_ot*1e3*p_wh*1e5*Mw + rho_o*R*T_w*m_gt*1e3))) #
Bernoulli/Hagen-Poiseuille
232 f4 = -p_bh*1e5 + (p_wi*1e5 + rho_o*9.81*H_bh + 128*mu_oil*L_bh*w_ro/(3.14*
D_bh**4*rho_o)) #Bernoulli/Hagen-Poiseuille
233 f5 = -rho_ai*1e2 + (Mw/(R*T_a)*p_ai*1e5) #Ideal gas law
234 f6 = -rho_m*1e2 + ((m_gt*1e3 + m_ot*1e3)*p_wh*1e5*Mw*rho_o)/(m_ot*1e3*p_wh*1
e5*Mw + rho_o*R*T_w*m_gt*1e3)#Relationship oil/gas
235 f7 = -w_iv + C_iv*sqrt(rho_ai*1e2*fmax(0.001,(p_ai*1e5 - p_wi*1e5)))#Valve
equation
236 f8 = -w_pc + u_pc*C_pc*sqrt(rho_m*1e2*fmax(0.001,(p_wh*1e5 - p_m*1e5)))#
Valve equation
237 f9 = -w_pg + (m_gt*1e3/fmax(1e-3,(m_gt*1e3+m_ot*1e3)))*w_pc #massfraction of
gas
238 f10 = -w_po + (m_ot*1e3/fmax(1e-3,(m_gt*1e3+m_ot*1e3)))*w_pc #massfraction
oil
239 f11 = -w_ro + PI*1e-6*(p_res*1e5 - p_bh*1e5)#From definition of Productivity
index
240 f12 = -w_rg + GOR*w_ro #From definition of Gas-oil ratio
241 #Riser system
242 f15 = -p_rh*1e5 + ((R*T_r/Mw))*(m_gr*1e3/(L_r*A_r)) - ((m_gr*1e3+m_or*1e3)/(
L_r*A_r))*9.81*H_r/2 #Bernoulli
243 f16 = -rho_r*1e2 + ((m_gr*1e3 + m_or*1e3)*p_rh*1e5*Mw*rho_ro)/(m_or*1e3*p_rh
*1e5*Mw +rho_ro*R*T_r*m_gr*1e3)
244 f17 = -p_m*1e5 + (p_rh*1e5 + 9.81/(A_r*L_r)*(m_or*1e3+m_gr*1e3)*H_r + 128*
mu_oil*L_r*w_pr/(np.pi*D_r**4*((m_gr*1e3+m_or*1e3) * p_rh*1e5*Mw*rho_ro) / (
m_or*1e3*p_rh*1e5*Mw+rho_ro*R*T_r*m_gr*1e3))) #Realationship oil/gas
245 f18 = -w_pr + 1*C_pr * np.sqrt(rho_r*1e2*fmax(0.001,(p_rh*1e5-p_gs*1e5))) #
Valve equation
246 f19 = -w_to + (m_or*1e3/(m_gr*1e3 + m_or*1e3))*w_pr #massfraction oil
247 f20 = -w_tg + (m_gr*1e3/(m_gr*1e3 + m_or*1e3))*w_pr #massfraction gas
248 #Separator system
249 f21 = -w_os + z_ov*C_os*sqrt(rho_ro*1e2*fmax(0.001,(p_os*1e5 - p_oo*1e5))) #
Valve equation
250 f22 = -w_gs + C_gs*np.sqrt(rho_gs*1e2 *fmax(0.001,(p_gs*1e5 - p_go*1e5))) #
Valve equation
251 f23 = -rho_gs*1e2 + (Mw/(T_s * R) * p_gs*1e5) #Ideal gas law
252 f24 = -p_os*1e5 + p_gs*1e5 + rho_ro * 9.81 * h_ls #Bernoulli equation
253 f25 = -v_os + ((0.5*r_s**2)*((2*np.arccos(fmax(0,(r_s-h_ls)/r_s)))-np.sin
((2*np.arccos(fmax(0,(r_s-h_ls)/r_s))))))*L_s #Based on equation of segment/
circle, derivative of the Area
254 f26 = -v_gs + fmax(0,(v_s - v_os)) #Based on relation volume of gas/oil in
separator
255 #Compressor 1
256 f27 = -w_in1 + C_in*u_1*np.sqrt(rho_in1*1e2*fmax(0.001,(p_gs*1e5 - p_s1*1e5)
))#Valve equation
257 f28 = -w_out1 + C_in*u_2*np.sqrt(rho_d1*1e2*fmax(0.001,(p_d1*1e5-p_s2*1e5)))

```

```

#Valve equation
258 f29 = -rho_in1*1e2 + (Mw/(R*T_in)*p_gs*1e5)#Ideal gas law
259 f30 = -rho_d1*1e2 + (Mw/(R*T_d)*p_d1*1e5)#Ideal gas law
260 f31 = -Phi1 + alpha_1 + alpha_2*omega1 + alpha_3*w_c1 + alpha_4*omega1*w_c1
+ alpha_5*omega1*omega1 + alpha_6*w_c1*w_c1#Polynomial relationship/
approximation
261 f32 = -Pow1 + (y_p1/(n_p1))*w_c1#Based on how much of the potential power
that can be utilized
262 f33 = -y_p1*1e5 + (Z_in *R *T_in/(Mw))*(n_v/(n_v-1)) *((Phi1**((n_v-1)/n_v))
-1)#Equation for polytropic head
263 f34 = -n_p1*1e2 + beta_1 + beta_2*omega1 + beta_3*Phi1 + beta_4*omega1*Phi1
+ beta_5*omega1*omega1 + beta_6*Phi1*Phi1#Polynomial relationship/
approximation
264 f35 = -Phi_max1 + gamma_1*(w_c1-gamma_2) + gamma_3 #Further work
265 f36 = - gamma_2_dummy1 + w_c1 - ((Phi1 - gamma_3)/gamma_1) #Further work
266 f37 = -w_rec1 + C_rec*u_rec1*np.sqrt(rho_d1*1e2*fmax(0.0001,(p_d1*1e5 - p_s1
*1e5)))#Valve equation
267 #Compressor 2
268 f38 = -w_in2 + C_in*u_2*np.sqrt(rho_in2*1e2*fmax(0.001,(p_d1*1e5 - p_s2*1e5)
))#Valve equation
269 f39 = -w_out2 + C_in*u_3*np.sqrt(rho_d2*1e2*fmax(0.001,(p_d2*1e5-p_s3*1e5)))
#Valve equation
270 f40 = -rho_in2*1e2 + (Mw/(R*T_in)*p_d1*1e5)#Ideal gas law
271 f41 = -rho_d2*1e2 + (Mw/(R*T_d)*p_d2*1e5)#Ideal gas law
272 f42 = -Phi2 + alpha_1 + alpha_2*omega2 + alpha_3*w_c2 + alpha_4*omega2*w_c2
+ alpha_5*omega2*omega2 + alpha_6*w_c2*w_c2#Polynomial relationship/
approximation
273 f43 = -Pow2 + (y_p2/n_p2)*w_c2#Based on how much of the potential power that
can be utilized
274 f44 = -y_p2*1e5 + (Z_in *R *T_in/(Mw))*(n_v/(n_v-1)) *((Phi2**((n_v-1)/n_v))
-1)#Equation for polytropic head
275 f45 = -n_p2*1e2 + beta_1 + beta_2*omega2 + beta_3*Phi2 + beta_4*omega2*Phi2
+ beta_5*omega2*omega2 + beta_6*Phi2*Phi2#Polynomial relationship/
approximation
276 f46 = -Phi_max2 + gamma_1*(w_c2-gamma_2) + gamma_3 #Further work
277 f47 = - gamma_2_dummy2 + w_c2 - ((Phi2 - gamma_3)/gamma_1) #Further work
278 f48 = -w_rec2 + C_rec*u_rec2*np.sqrt(rho_d2*1e2*fmax(0.0001,(p_d2*1e5 - p_s2
*1e5)))#Valve equation
279 #Compressor 3
280 f49 = -w_in3 + C_in*u_3*np.sqrt(rho_in3*1e2*fmax(0.001,(p_d2*1e5 - p_s3*1e5)
))#Valve equation
281 f50 = -w_out3 + C_in*u_4*np.sqrt(rho_d3*1e2*fmax(0.001,(p_d3*1e5-p_out*1e5))
)#Valve equation
282 f51 = -rho_in3*1e2 + (Mw/(R*T_in)*p_d2*1e5)#Ideal gas law
283 f52 = -rho_d3*1e2 + (Mw/(R*T_d)*p_d3*1e5)#Ideal gas law
284 f53 = -Phi3 + alpha_1 + alpha_2*omega3 + alpha_3*w_c3 + alpha_4*omega3*w_c3
+ alpha_5*omega3*omega3 + alpha_6*w_c3*w_c3#Polynomial relationship/
approximation
285 f54 = -Pow3 + (y_p3/n_p3)*w_c3#Based on how much of the potential power that
can be utilized
286 f55 = -y_p3*1e5 + (Z_in *R *T_in/(Mw))*(n_v/(n_v-1)) *((Phi3**((n_v-1)/n_v))
-1)#Equation for polytropic head
287 f56 = -n_p3*1e2 + beta_1 + beta_2*omega3 + beta_3*Phi3 + beta_4*omega3*Phi3
+ beta_5*omega3*omega3 + beta_6*Phi3*Phi3#Polynomial relationship/
approximation
288 f57 = -Phi_max3 + gamma_1*(w_c3-gamma_2) + gamma_3 #Further work
289 f58 = - gamma_2_dummy3 + w_c3 - ((Phi3 - gamma_3)/gamma_1) #Further work
290 f59 = -w_rec3 + C_rec*u_rec3*np.sqrt(rho_d3*1e2*fmax(0.001,(p_d3*1e5 - p_s3
*1e5)))#Valve equation
291 #Gas Lift
292 f60 = -w_gl + C_gl*u_gl*np.sqrt(rho_out*1e2*fmax(0.001,(p_out*1e5 - p_ai*1e5
)))#Valve equation
293 f61 = -p_out*1e5 + R*T_d*m_gl*1e3/(Mw*np.pi*r_gl*r_gl*L_gl)#Ideal gas law
294 f62 = -rho_out*1e2 + (Mw/(R*T_d)*p_out*1e5)#Ideal gas law
295
296 #Differential equations
297 #Wells
298 df1 = (w_gl - w_iv)*1e-3 #m_ga, massbalance of gas annulus
299 df2 = (w_iv + w_rg - w_pg)*1e-3 #m_tg, massbalance of gas tubing
300 df3 = (w_ro - w_po)*1e-3 #m_to, massbalance of oil tubing

```

```

301 #Riser
302 df4 = (sum(w_pg.nz) - w_tg)*1e-3 #m_gt, massbalance gas riser
303 df5 = (sum(w_po.nz) - w_to)*1e-3 #m_ot, massbalance oil riser
304 #Separator
305 df6 = ((R*T_s/(v_gs*Mw))*(w_tg - w_gs - w_in1)*1e-4) + (p_gs/(v_gs*rho_ro))
306 *((w_to - w_os)*1e-4)# p_gs, based on derivative of ideal gas law
307 df7 = (((w_to - w_os)*1e-3)/rho_ro)/(2*L_s * np.sqrt(h_ls* fmax(0,((2 *r_s)-
308 h_ls))))#h_ls, based on equation of segment/circle, derivative of the Area
309 #Compressor system(diff equations)
310 #Compressor 1
311 df8 = (w_in1 - w_c1 + w_rec1) * Coef_1 #p_s1, based on gas in/out of system
312 df9 = (w_c1 - w_out1 - w_rec1) *Coef_2 #p_d1, based on gas in/out of system
313 df10 = (p_s1*Phi1 - p_d1) * Coef_3 #w_c1, based on pressure difference
314 between in/out
315 # Define variables for combined systems (needed only for decomposition case)
316 #Compressor 2
317 df11 = (w_in2 - w_c2 + w_rec2) * Coef_1 #p_s2, based on gas in/out of
318 system
319 df12 = (w_c2 - w_out2 - w_rec2) *Coef_2 #p_d2, based on gas in/out of system
320 df13 = (p_s2*Phi2 - p_d2) * Coef_3 #w_c2, based on pressure difference
321 between in/out
322 #Compressor 3
323 df14 = (w_in3 - w_c3 + w_rec3) * Coef_1#p_s3, based on gas in/out of system
324 df15 = (w_c3 - w_out3 - w_rec3) *Coef_2#p_d3, based on gas in/out of system
325 df16 = (p_s3*Phi3 - p_d3) * Coef_3#w_c3, based on pressure difference
326 between in/out
327 #Gas lift(diff equations)
328 df17 = (w_out3 - sum(w_gl.nz))*1e-3 #m_gl, based on massbalance
329
330 #Form the DAE system
331 dif = vertcat(df1,df2,df3,df4, df5,df6,df7,df8,df9,df10,df11,df12,df13,df14,
332 df15,df16,df17) #Differential equations
333 alg = vertcat(f1,f2,f3,f4,f5,f6,f7,f8,f9,f10,f11,f12,f15,f16,f17,f18,f19,f20
334 ,f21,f22,f23,f24,f25,f26,f27,f28,f29,f30,f31,f32,f33,f34,f35,f36,f37,f38,f39
335 ,f40,f41,f42,f43,f44,f45,f46,f47,f48,f49,f50,f51,f52,f53,f54,f55,f56,f57,f58
336 ,f59,f60,f61,f62) #Algebraic equations
337 x_var = vertcat(m_ga,m_gt,m_ot,m_gr, m_or,p_gs,h_ls,p_s1,p_d1, w_c1, p_s2,
338 p_d2,w_c2,p_s3,p_d3,w_c3,m_gl) #Differential states
339 z_var = vertcat(p_ai,p_wh,p_wi,p_bh,rho_ai,rho_m,w_iv,w_pc,w_pg,w_po,w_ro,
340 w_rg, p_rh,rho_r,p_m,w_pr,w_to,w_tg,w_os,w_gs,rho_gs,p_os,v_os,v_gs,w_in1,
341 w_out1,rho_in1,rho_d1, Phi1,Pow1,y_p1,n_p1,Phi_max1,gamma_2_dummy1,w_rec1,
342 w_in2,w_out2,rho_in2,rho_d2, Phi2,Pow2,y_p2,n_p2,Phi_max2,gamma_2_dummy2,
343 w_rec2,w_in3,w_out3,rho_in3,rho_d3,Phi3,Pow3,y_p3,n_p3,Phi_max3,
344 gamma_2_dummy3,w_rec3,w_gl,p_out,rho_out)#Algebraic states
345 u_var = vertcat(u_gl,z_ov,u_1,u_pc,u_2,u_3,u_4,u_rec1,u_rec2,u_rec3,omega1,
346 omega2,omega3)#Control variables
347 p_var = vertcat(GOR,wmax_gl,wmax_pg,Powmax_glcom,p_go,p_oo)#Parameters/
348 constraints
349
350 #Inequality constraints
351 g_var = vertcat((w_gs-wmax_pg),((Pow1 + Pow2 + Pow3)-Powmax_glcom),(sum(w_gl
352 .nz)- wmax_gl))
353
354 #Objective function(Whant to maximize oil production and minimize power con
355 sumption)
356 L = -0.6*w_os + 0.1*Pow1 + 0.1*Pow2 + 0.1*Pow3
357
358 #Free variables need to be added
359 alg = substitute(alg,p_res,par['p_res'])
360 alg = substitute(alg,PI,par['PI'])
361 alg = substitute(alg,T_a,par['T_a'])
362 alg = substitute(alg,T_w,par['T_w'])
363
364 #Constructing the total DEA system, into CasADI framework
365 dae = {'x': x_var, 'z': z_var, 'p': vertcat(u_var,p_var), 'ode': dif, 'alg': alg
366 , 'quad': L}
367
368 #Define integration time
369 opts = {'tf': par['tf']}
370

```

```
350 #Create IDAS integrator for the DAE system
351 F = integrator('F','idas',dae,opts)
352
353 #Returns values
354 return F,x_var, z_var, u_var, p_var, alg, dif, L, g_var
```

C.3 Main file

```

1 #Main file
2 #Coding based on and inspired by model made by Risvan Dirza(NTNU).
3 #Integrates the system of equations with the use of the CasADI framework IDAS
  integrator.
4 #Optimize the system of equations with the use of the CasADI framework IPOPT nlp
  solver.
5
6
7 import numpy as np
8 from sys import path
9 path.append(r"C:/Users/Bruker/Documents/CASADIPython/casadi-windows-py38-v3
  .5.5-64bit")
10 from casadi import *
11 import casadi as ca
12
13 # Call the parameters
14 import ParamFordelivery
15
16 #par now represents the dictionary defined in parameter function
17 par = ParamFordelivery.Params_6wells()
18
19
20
21
22 import pandas as pd
23 #Retrieve initial guesses for the differential states(x0), algebraic states(z0)
  and
24 #controlled variables(u0). Data listed in excel, comma separated files.
25 x0 = pd.read_csv('Data Folder 6wellsSeptestCompGL32/x06Sep.csv',header=None).
  values.reshape(-1)
26 z0 = pd.read_csv('Data Folder 6wellsSeptestCompGL32/z06Sep.csv',header=None).
  values.reshape(-1)
27 u0 = pd.read_csv('Data Folder 6wellsSeptestCompGL32/u06Sep.csv',header=None).
  values.reshape(-1)
28
29 #Retrieve the lower and upper bounds for the differential states(x), algebraic
  states(z) and
30 #controlled variables(u). Data listed in excel, comma separated files.
31 lbx = pd.read_csv('Data Folder 6wellsSeptestCompGL32/lbx6Sep.csv',header=None).
  values.reshape(-1)
32 lbz = pd.read_csv('Data Folder 6wellsSeptestCompGL32/lbz6Sep.csv',header=None).
  values.reshape(-1)
33 lbu = pd.read_csv('Data Folder 6wellsSeptestCompGL32/lbu6Sep.csv',header=None).
  values.reshape(-1)
34 ubx = pd.read_csv('Data Folder 6wellsSeptestCompGL32/ubx6Sep.csv',header=None).
  values.reshape(-1)
35 ubz = pd.read_csv('Data Folder 6wellsSeptestCompGL32/ubz6Sep.csv',header=None).
  values.reshape(-1)
36 ubu = pd.read_csv('Data Folder 6wellsSeptestCompGL32/ubu6Sep.csv',header=None).
  values.reshape(-1)
37
38 #Define the parameter intial values(constant, if not manually changed)
39 p0 = ca.vertcat(par['GOR'],par['wmax_gl'],par['wmax_pg'],par['Powmax_glcom'],par
  ['p_go'],par['p_oo'])
40
41 #Call the simulator
42 import SimulatorFordelivery
43
44 #Retrieve return variables of the integrator function
45 F,x_var, z_var, u_var, p_var, alg, dif, L, g_var = SimulatorFordelivery.
  CentralizedSimulator_F(par)
46 """
47 #Define time span of simulation
48 t_span = np.arange(10)
49
50 #Initialize initial values
51 uk = u0
52 xf = x0
53 zk = z0

```

```

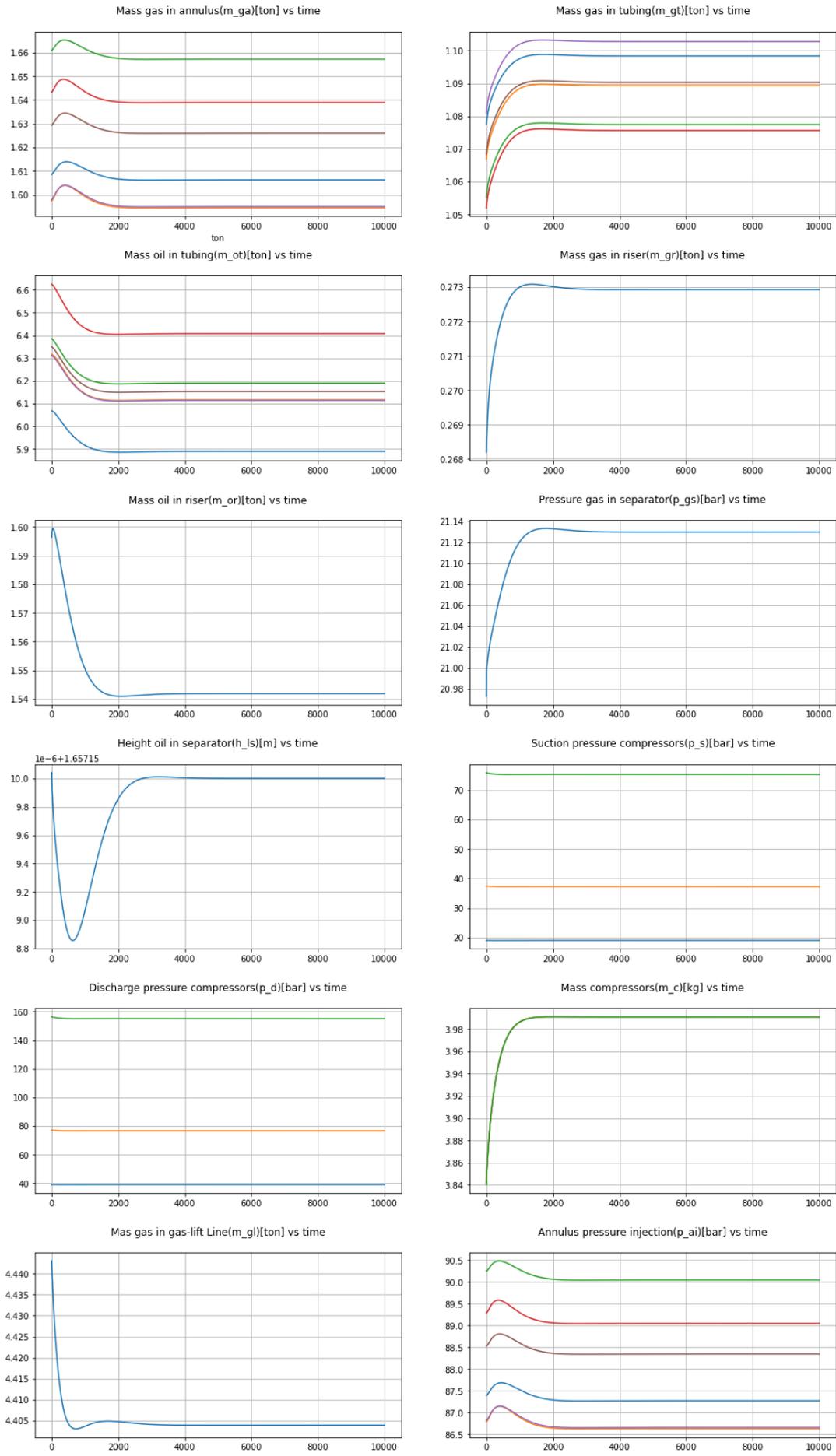
54
55 #Make containers for storing integrator/control output
56 x_store = []
57 z_store = []
58 u_store = []
59 p_store = []
60 error_store = [0]
61
62 ##### Integrator #####
63
64 for k in t_span:
65     #Change to simulate disturbance in GOR(Possible to implement disturbance in
66     #if k == 2000:
67         #p0[0] -= 0.1
68         #p0[1] += 0.1
69         #p0[2] += 0.1
70         #p0[3] += 0.1
71         #p0[4] += 0.1
72         #p0[5] += 0.1
73
74     #Simulate step change in the gas lift valves(u_gl)
75     #if k == 1000:
76         #uk[0] += 0.2
77         #uk[1] += 0.1
78         #uk[2] += 0.4
79         #uk[3] += 0.2
80         #uk[4] += 0.2
81         #uk[5] += 0.2
82
83     #Solving the initial value problem
84     inputs = ca.vertcat(uk, p0)
85     Fk = F(x0 = xf, z0 = zk, p = inputs)
86     #Retrieving the differential states
87     xf = (Fk['xf']).full()
88     #Retrieving the algebraic states
89     zk = (Fk['zf']).full()
90
91     #Append results
92     x_store.append(xf)
93     u_store.append(uk)
94
95     #PID controller, tuned with SIMC rules(integration process)
96     h = x_store[k][21] #Value of height og oil at current iteration
97     h_sp = 1.65716 #Optimal height, thus used as setpoint
98     tauC = 50 #Controller time, can be changed up or down depending on needs for
99     #fast control or smooth control
100     tauI = 4*tauC #Integral time, corresponding to SIMC rules for integration
101     #processes.
102     Kp1 = 1/(5.1732222222222735e-06*tauC)#Proportional gain
103     Ki1 = Kp1/tauI #Integral gain
104     error = (h_sp - h) #Difference between setpoint and measured height
105     #Calculate new controller output
106     u = ca.fmax(0, ca.fmin(1, (u_store[k-1][6] - (Kp1*error + Ki1*error - Kp1*
107     error_store[k-1] ))))
108     #Update the controller output for z_ov(oil valve separator)
109     uk[6] = u
110     #Store all errors, to be used for previous errors
111     error_store.append(error)
112
113 ##### Optimizer #####
114
115 #Get shape of controlled, and differential states
116 nu = u_var.shape[0]
117 nx = x_var.shape[0]
118 #Define equality constraints of nlp, equals the model equations
119 eqcon = ca.vertcat(alg, dif)

```

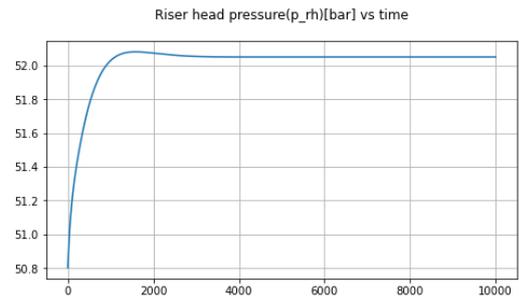
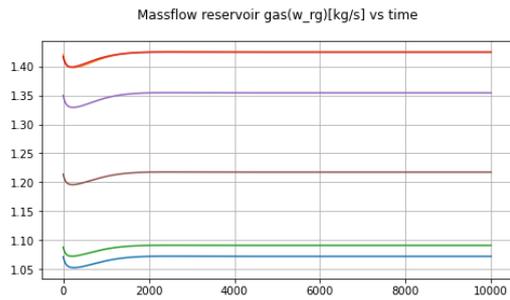
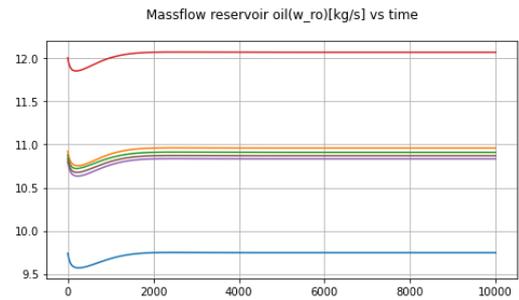
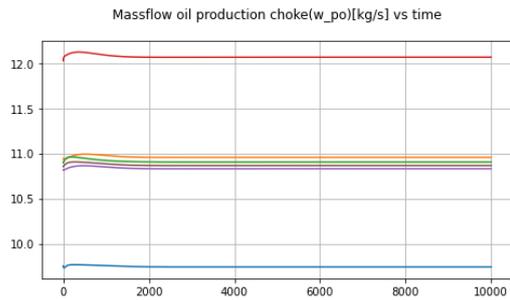
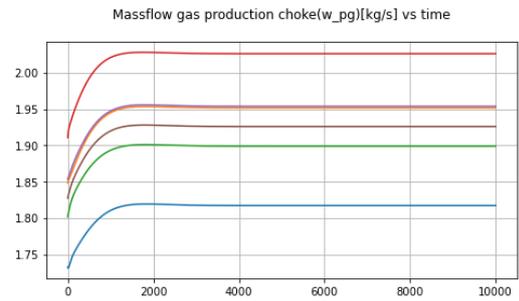
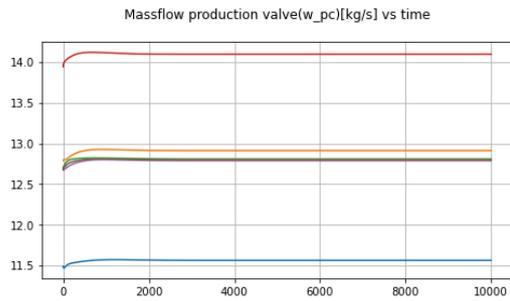
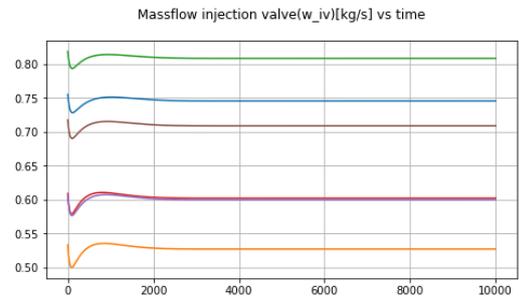
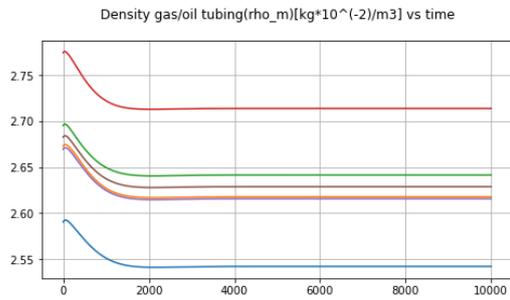
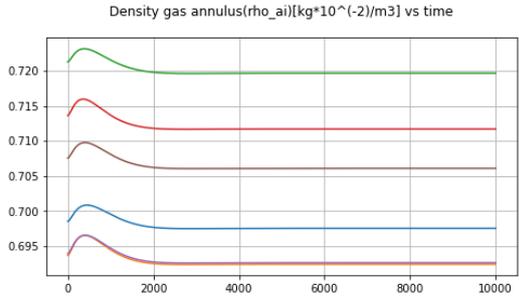
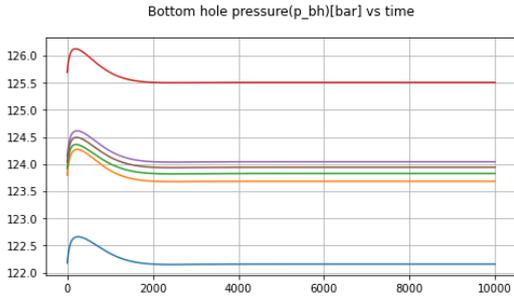
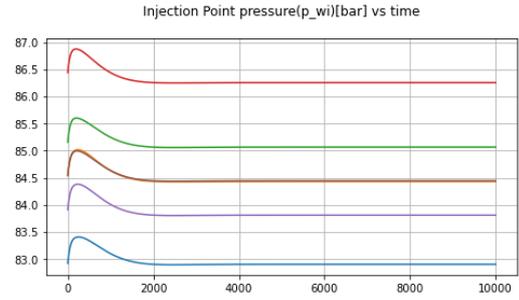
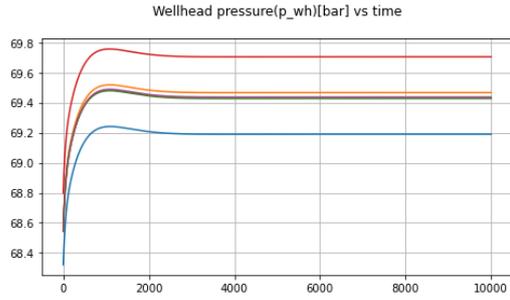
```
120
121 #Define the optimization problem(x = states, L= objective function, g =
    inequality constraints, p = parameters)
122 nlp = {
123     'x': ca.vertcat(u_var, x_var, z_var),
124     'f': L,
125     'g': ca.vertcat(eqcon, g_var),
126     'p': p_var,
127 }
128
129 #Define upper/lower bounds for the inequality constraints
130 lbg = ca.vertcat(np.full(eqcon.shape, 0), np.full(g_var.shape, -ca.inf))
131 ubg = ca.vertcat(np.full(eqcon.shape, 0), np.full(g_var.shape, 0))
132
133 #Use IPOPT(interior point optimizer) fromt the CasADI framework to solve the nlp
134 opt_inst = ca.nlpsol('opt_inst', 'ipopt', nlp)
135
136 #Extract the solution of the optimization, feed ipopt initial values, lower and
    upper bounds
137 opt_res = (opt_inst(p=p0,x0 = ca.vertcat(u0,x0,z0), lbx = ca.vertcat(lbu,lbx,lbz
    ), ubx = ca.vertcat(ubu,ubx,ubz), lbg=lbg, ubg=ubg))
138
139 states = opt_res['x'] #Optimal controlled variables(u), differential states(x)
    and algebraic states(z)
140 cost = opt_res['f'] #Value of objective function at optimal point
141
142 print(states)
```


D Integration results

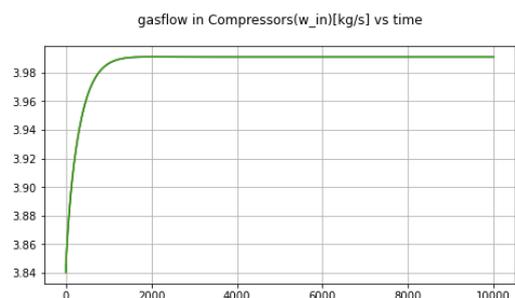
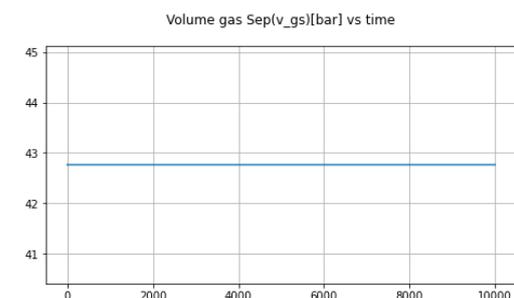
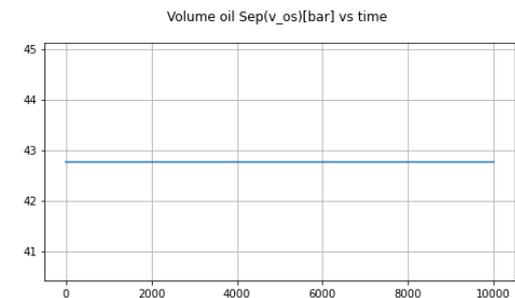
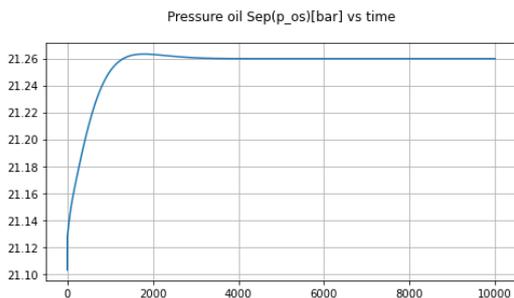
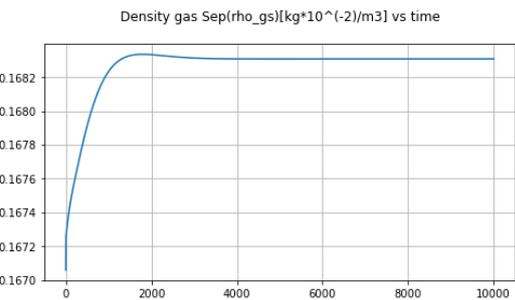
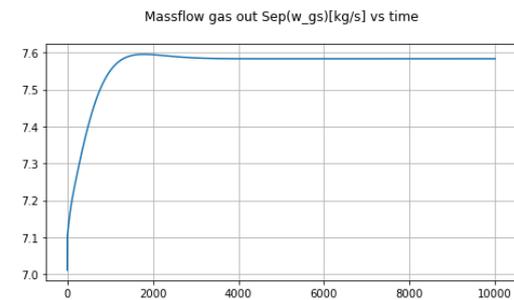
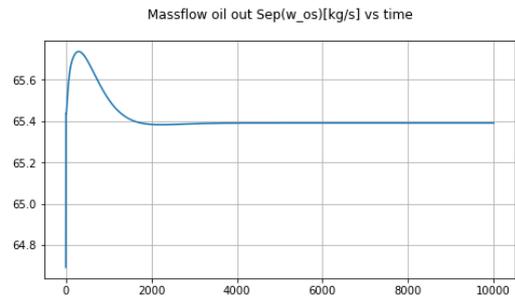
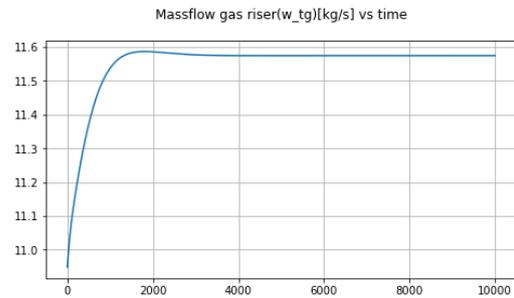
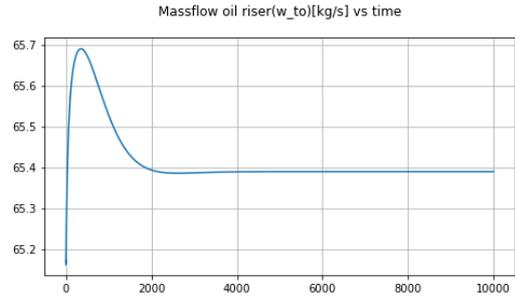
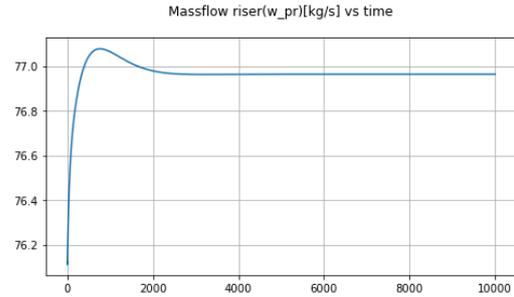
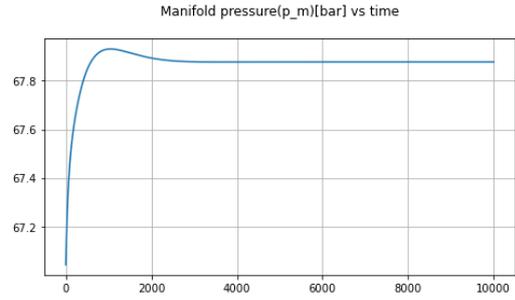
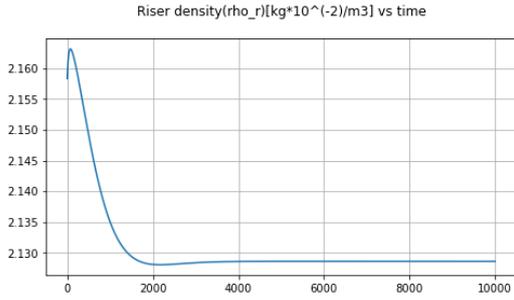
Figure D.1: In Appendix D the integration results for the system states and the controlled valve opening at the oil outlet of the separator are presented. The system is solved with IDAS explained in Section 2.2.1, with a simulation time of $t = 10000$ s.



D INTEGRATION RESULTS



D INTEGRATION RESULTS



D INTEGRATION RESULTS

