NTNU

Norwegian University of
Science and Technology

Modelling and Control of Coalescence Based Gravity Separators in a Subsea Separation System

with Models of Produced Water Treatment

Sindre Johan Heggheim

Chemical Engineering and Biotechnology
Submission date: June 2018
Supervisor: Johannes Jäschke, IKP
Co-supervisor: Tamal Das, IKP

Norwegian University of Science and Technology
Department of Chemical Engineering

Abstract

Subsea separation of well fluids is a way to reduce the cost of topside processing, and to simplify and reduce the export of products. Subsea separation consist of bulk separators and purifiers, to separate the multi-phase well stream and to get as clean phases as possible. Models and simulations of the subsea process is needed to optimize the process, and to assure that the purity restriction on the produced water to be injected, is held.

The aim of this work has been to develop dynamic models of the separators in the bulk separation of the well stream. The developed models was two coalescence driven gravity separators with main focus on oil-water separation. The oil product of the first stage gravity separator (three phase) was fed to the second stage gravity separator (two phase). The models of the bulk separators were combined with existing models of inline de-oiling hydrocyclones and compact flotation units, which are used in produced water treatment. The combined system was simulated and controlled with proportional integral (PI) controllers.

The steady state gravity separators were implemented and solved in MATLAB with CasADi. The models uses Stokes' terminal velocity adjusted by the water cut and coalescence of droplets with 10 different sizes in the range of 150 μm to 300 μm in diameter. The steady state is solved for wanted pressure, liquid level and water level. Interior point optimizer (IPOPT), with slacks on the overconstrained equality constraints, was used to solve the steady state. The dynamic models, with product flows as system inputs, were solved and simulated with ODE15s-solver in Simulink.

The gravity separator models showed reasonable steady state solutions and dynamic behavior with the guessed model parameters. The model would benefit from better tuned model parameters from measured data. The simulation of the combined bulk and water treatment showed that the water treatment was able to control the purity of the produced water and that a restructuring and tuning would improve the control even further.

Sammendrag

Subsea separasjon gjør det mulig å redusere kostnadene i topside prosesser, og transport av unødvendig produkt. Subsea separasjon består av bulk separasjonen og rensing av fasene, slik at de blir så rene som mulig. Modeller og simuleringer av prosessen gjør det mulig å optimalisere prosessen og sørge for at renhetskravet til utslipp av vannet blir overholdt.

Målet med dette arbeidet har vært å lage en dynamisk model av bulk separasjonen. Modellen består av to gravitasjonsseparatorer med koaleserende dråper, med fokus på væske-væske separasjon. Olje produktet fra den første separatoren (trefase-separator) ble ført inn i den andre separatoren (tofase-separator). Modellen av bulk separasjonen ble satt sammen med modeller av eksisterende modeller av oljeseparerende hydrosykloner og kompakte flotasjonsenheter som brukes til å rense det produserte vannet. Separatorene ble regulert med proporsjonal integral (PI) regulatorer.

Steady state modellene av bulk separatorene ble implementert og løst i MATLAB med CasADi. Modellene bruker Stokes' terminal hastighet justert med vannkuttet, og koalesering med 10 dråpestørrelser mellom 150 μm og 300 μm i diameter. Steady state er løst for ønsket trykk, væskenivå og vannivå. Den valgte løseren var IPOPT ("interior point optimizer") med slakk på de overspesifiserte likhetsbegrensningene. De dynamiske modellene ble løst og simulert med ODE15s i Simulink.

Modellene av gravitasjonsseparatorene viste rimelige resultat og dynamisk atferd med de valgte modellparameterene. Modellen kunne dratt nytte av bedre parametere fra målte data. Simuleringen av den kombinerte systemet viste at vannrenseanlegget klarte å holde renheten stabil ved referansepunktet og at det er rom for forbedringer i innstillingene og strukturen til reguleringen av systemet.

Preface

This master's thesis was done at Department of Chemical Engineering at Norwegian University of Science and Technology (NTNU). This project was done together with supervisor, Associate Professor Johannes Jäschke and co-supervisor, PhD. Candidate Tamal Das. The project was performed in cooperation with Das' PhD project within the SUBPRO project.

The goal of the thesis was to develop a simple model of the bulk separation of the well stream, in oil and gas production. The model consist of two dynamic coalescence driven gravity separators. The model of the bulk separation was then combined, simulated and tuned together with already developed models of the separators used in produced water treatment.

I would like to thank Tamal Das for the help I have gotten during the project and that I could be a part this interesting project.

Declaration of Compliance

I declare that this is an independent work according to the exam regulations of the Norwegian University of Science and Technology (NTNU).

Sindre Johan Heggheim

Trondheim, June 11, 2018

Table of Contents

List of Tables

List of Figures

Symbols and Abbreviations

Latin Letters

A	Area	m^2
A	Droplet mass	g
d	Diameter	m, $\mu\mathrm{m}$
d	System disturbance	$\mathrm{m}^3\,\mathrm{s}^{-1}$,(-)
F	Volumetric Flow	$\mathrm{m}^3\,\mathrm{s}^{-1}$
g	Gravitational acceleration constant	$\mathrm{m}\,\mathrm{s}^{-2}$
h	Height	m
h	Level	m
J	Cost function in optimization problem	-
k	Process gain	output unit/input unit
k	Rate constant, coalescence	$\mathrm{m}^3\,\mathrm{s}^{-1}\,\mathrm{kg}^{-1}$
k	Rate constant, interfacial transfer	$\mathrm{m}\,\mathrm{s}^{-1}$
k'	Process slope	output unit/input unit/s
K_c	Controller gain	input unit/output unit
L	Separator length	m
l	Length of travel	m
m	Mass	kg
$\dot{m}$	Rate of mass	$\mathrm{kg}\,\mathrm{s}^{-1}$
$\mathbf{m}$	Vector with mass of droplet sizes	kg
$\dot{\mathbf{m}}$	Vector with rates of mass of droplet sizes	$\mathrm{kg}\,\mathrm{s}^{-1}$
M_m	Molar mass	$\mathrm{kg}\,\mathrm{mol}^{-1}$
$\mathbf{N}$	Matrix of stoichiometric coefficients	-
n	Number of control volumes in a direction	-
n	Stokes adjustment parameter	-

n	Amount of substance	mol
$\dot{n}$	Rate of substance	$\mathrm{mol\,s^{-1}}$
P	Gas pressure	bar
q	Internal convective flow through control volumes	$\mathrm{m^3\,s^{-1}}$
$\mathbf{r}$	Vector of reaction rates, coalescence	$\mathrm{kg\,m^{-3}\,s^{-1}}$
R	Change in partial densities from coalescence	$\mathrm{kg\,m^{-3}\,s^{-1}}$
R	Gas constant	$\mathrm{J\,K^{-1}\,mol^{-1}}$
r	Radius	m
r	Reaction rate, coalescence	$\mathrm{kg\,m^{-3}\,s^{-1}}$
r	Reaction rate, interfacial transfer	$\mathrm{m^3\,s^{-1}}$
$\mathbf{s}$	Scaling vector with droplet size indices	-
T	Temperature	K
t	Time	s
u	System input	$\mathrm{m^3\,s^{-1}}$
V	Volume	$\mathrm{m^3}$
v	Velocity	$\mathrm{m\,s^{-1}}$
$\dot{V}$	Rate of volume	$\mathrm{m^3\,s^{-1}}$
X	State vector	$\mathrm{kg\,m^{-3}}$, bar, m
$\dot{X}$	Change in state vector	$\mathrm{kg\,m^{-3}\,s^{-1}}$, $\mathrm{bar\,s^{-1}}$, $\mathrm{m\,s^{-1}}$

Greek Letters

ϵ_{in}	Water cut in feed	-
γ	Initial separation of oil and water	-
θ	Angle	rad
μ	Viscosity	$\mathrm{kg\,s^{-1}\,m}$
ρ	Density	$\mathrm{kg\,m^{-3}}$
$\bar{\rho}$	Partial density	$\mathrm{kg\,m^{-3}}$
τ	Residence time	s
τ_1	First order time constant	s
τ_c	Closed loop time constant	s
τ_I	Integral time	s

Superscripts

a	Adjusted Stokes

$conv$	Convective transfer
g	Gas
$gen/lost$	Generated or lost from coalescence
in	Feed to separator or control volume
int	Interface
l	Liquid
$left$	Left of current control volume
out	Out of separator or control volume
$over$	Over current control volume
p	Phase inversion
s	Stokes
top	Control volume at top ($j = n_z$)
$under$	Under current control volume

Subscripts

α	Size class of a coalescing droplet
A	First size class of droplets
β	Size class of a coalescing droplet
c	Continuous phase
c	Cross section
cv	Control volume
δ	Single droplet
d	Dispersed phase
d	Droplet
G	Gas product
g	Gas
h	Height in a circle
i	i-th row of control volumes from inlet
j	j-th layer of control volumes from bottom
j	Layer index in z direction
ξ	Coalescence reaction index
L	Water product, left of weir
l	Liquid level

l	Liquid
lw	Water level
max	Maximum
min	Minimum
o	Oil feed to Oil phase fraction
o	Oil
op	Oil phase
R	Oil product, right of weir
s	States
sp	Setpoint
t	Horizontal transfer
t	Terminal
w	Water feed to water phase fraction
w	Water
w	Weir
wp	Water phase
χ	State vector index
x	x direction
y	Droplet size class index
z	z direction

Abbreviations

CFU	Compact flotation unit
GS1	First stage gravity separator
GS2	Second stage gravity separator
HC	Hydrocyclone
IMC	Internal model control
IPOPT	Interior point optimizer
OC	Oil cut
OiW	Oil dispersed in water
PI	Proportional integral
PWT	Produced water treatment
RTO	Real time optimizer

WC	Water cut
WiO	Water dispersed in oil

Chapter 1

Introduction

Subsea production system is an alternative to ordinary offshore platforms, to make oil and gas production more efficient and enable utilization of reservoir in deep water seabed or reservoir in areas with harsh conditions [1]. Subsea processes such as subsea separation and injection systems reduce the amount of produced water in export and in topside processing. There is governmental restriction on the discharge of hydrocarbons in the sea and the purity of water recommended by OSPAR (Convention for the Protection of the Marine Environment of the North-East Atlantic, Recommendation 2001/1) is 30 mg/l as monthly average [2].

Models of the different separators in the subsea separator system can be used to optimize the process and find optimal control structures. It is beneficial to use simplified models for estimation of unmeasured process parameters, since measurements in subsea might be difficult or impossible. Combining the separator models and performing simulation on the combined system will show how the purified produced water reacts to disturbances that enters the initial bulk separator and if the product is held at the acceptable purity.

1.1 Subsea Separation System

The well stream to be separated consists of several phases, mainly gas, oil and produced water. The bulk separation of the three phases takes place in two gravity separator. In the first stage gravity separator, water settles to the bottom and gas is flashed out in the top. The oil product is fed into a second stage gravity separator to achieve further separation of the dispersed water droplets. The produced water from the bulk separation goes through produced water treatment (PWT), consisting of compact separators to achieve water purity within the restricted limit [3,

4]. The produced gas also goes through compact separators to separate remaining liquids.

A possible water treatment system of compact separators uses inline deoiling hydrocyclones (HC) and compact flotation units (CFU) [5, 6]. A subsea separation system with bulk separation and produced water treatment is shown in Figure 1.1. The produced water in bulk separation is divided into three parallel HCs from where the underflows (water product) are combined and divided into four parallel inlet flows for CFUs. The water product of the flotation units are ready for rejection and the overflow of the hydrocyclones and the reject of the flotation units are combined with the oil product and exported to topside processing or processing plants on shore. The separation system being studied in this work does not look at purification of gas.

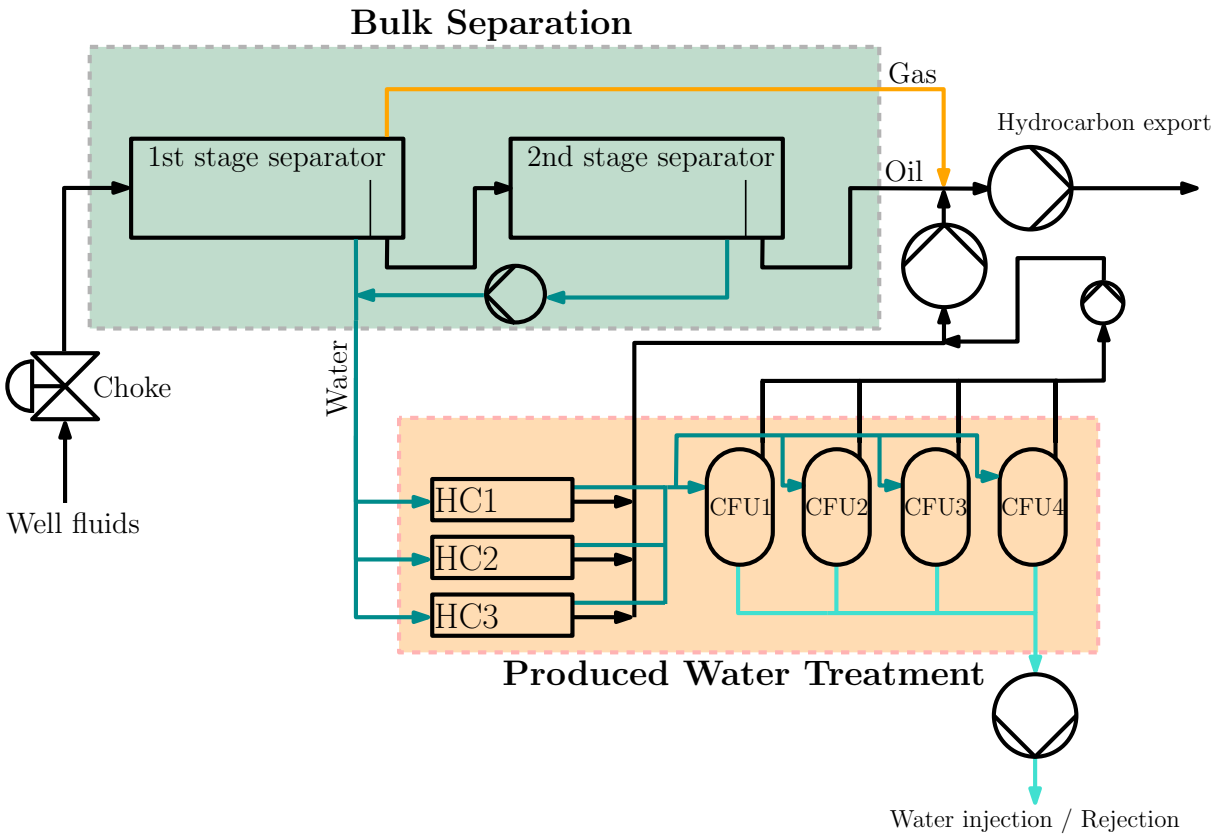


Figure 1.1: Subsea separation with bulk separation and water purification. The multi-phase feed is fed to a three phase gravity separator (1st stage separator), and the light phase product of oil with dispersed water is fed to a two phase gravity separator (2nd stage separator). The gas produced in the 1st stage separator and the oil product from the 2nd stage separator exported to further processing. The water products goes through produced water treatment which consist of three inline deoiling hydrocyclones (HC) and four compact flotation units (CFU). The separated oil is combined with exported hydrocarbons.

1.2 Previous Work and Contribution in This Work

Models of three phase gravity separators have already been developed earlier [7, 8]. These models are based on Stokes' terminal settling velocity which depends on the droplet size. The models are either not dynamic or has not yet implemented, the dynamic of coalescence and breakage of the dispersed droplets. Models of the inline hydrocyclones [5, 7] and the compact flotation units [6] in the produced water treatment have also been developed.

The objective of this work has been to develop dynamic models of the gravity separators in the bulk separation. The models focus on the liquid-liquid separation and consider the coalescence of the dispersed phases. The models do not consider dispersed gas in the liquid or dispersed liquid in the gas. The model of the bulk separation was then combined with the already existing models of the produced water treatment separators developed by Tamal Das [5, 6]. The combined system was then simulated to study the separation of the produced water.

The work in this thesis and further work on the system will be used as second author in a publication titled "*Optimal control for a hydrocarbon separation system including a coalescence based gravity separator model*". The publication extends the work by optimizing the separation and implementing an real time optimizer (RTO) on the separation system.

This work is structured into six chapters. In chapter two the basic theory of separation of dispersed phases is given. This includes the theory of settling and coalescence of dispersed droplets and the importance of residence time. In chapter 3 the steady state and dynamic models of the bulk separators are described. The chapter also studies the steady state solutions and the dynamic behavior of the bulk separators. In chapter 4 the models of the inline deoiling hydrocyclones and compact flotation units, used in the produced water treatment, are described. In chapter 5 the bulk separation and PWT system is combined and a study of the dynamics of the entire system is performed. Chapter 6 gives a conclusion to the study of the bulk separators and the combined separation system.

Chapter 2

Theory: Separation of Dispersed Phases

This chapter will go through the basic theory needed in the development of the gravity separators. This includes the main concepts in separation of dispersed phases, from sedimentation and coalescence of particles to the residence time of the particle movements. Some theory of interfacial transfer and oil-water emulsions will also be described.

The liquid-liquid mixture of oil and water to be separated is a colloidal suspension. The separation in the gravity separator is mechanical-physical separation process and uses the difference in gravity forces on the two phases to separate the phases [9]. The classification is then a settling and sedimentation process. Centrifugal settling and sedimentation has similarities but uses instead centrifugal forces to increase the gravitational force.

2.1 Settling and Sedimentation

The settling of the dispersed phase has either free settling or hindered settling depending on how cramped the droplets are [9]. The settling velocity stabilizes rapidly by the drag forces between the dispersed droplet and the continuous phase and thereby the separation will depend on the terminal velocity, v_t, of the droplets compared to the horizontal movement. Stokes' law of terminal velocity gives:

$$v_t = \frac{gd^2(\rho_d - \rho_c)}{18\mu_c}. \tag{2.1}$$

g is the gravitational acceleration, d is the diameter of the droplet. ρ_d and ρ_c are the density of the dispersed and continuous phase respectively and μ_c is the viscosity of the continuous phase. The terminal velocity given by Stokes (2.1) will in this work be given as v^s. Stokes' terminal velocity is derived assuming free settling of spherical droplets in laminar flow [9].

In a gravity separator with liquid-liquid separation, there is not free settling of droplets, and the concentration of droplets will increase near the interface of the two liquid phases. The settling velocity can be scaled or adjusted by the fraction of water, called water cut (WC), in the mixture:

$$v^a = v^s(1 - WC^{receiver})^n \tag{2.2}$$

where v^a is the adjusted stokes velocity and n is a parameter, which can be used to adjust the behaviour. Higher water cut in front of the droplet will decrease the terminal velocity. More on the adjusting of terminal velocity and the parameter is given Appendix A.

The residence time, τ, of the droplets is the time it uses to travel a specific distance. In the gravity separator the droplets has a residence time through the separator following the convective flow and another residence time where the droplets settles down (or up) to the interface. If the settling time or residence time in vertical direction, τ_z, is much higher than the horizontal direction, τ_x, the separation will be poor. Assuming the interfacial transfer is not limiting factor, a fully separation of the dispersed droplets is achieved if the residence time of all droplets is shorter in vertical direction than in the horizontal direction.

For constant cross section area, the residence times is given as:

$$\tau_z = \frac{l}{v} \tag{2.3}$$

$$\tau_x = \frac{V}{F} \tag{2.4}$$

where l is the length to be traveled with velocity v, and V is the volume the flow F flows through.

2.2 Coalescence

Coalescence of dispersed droplets is the process of two droplets colliding and merging together [10]. The colliding droplets will first be separated by a film and after the film is drained to a critical thickness, the droplets merge together to a single larger droplet. The coalescence process is shown in Figure 2.1. Breakage is the reversed process where a droplets is broken into two smaller droplets.

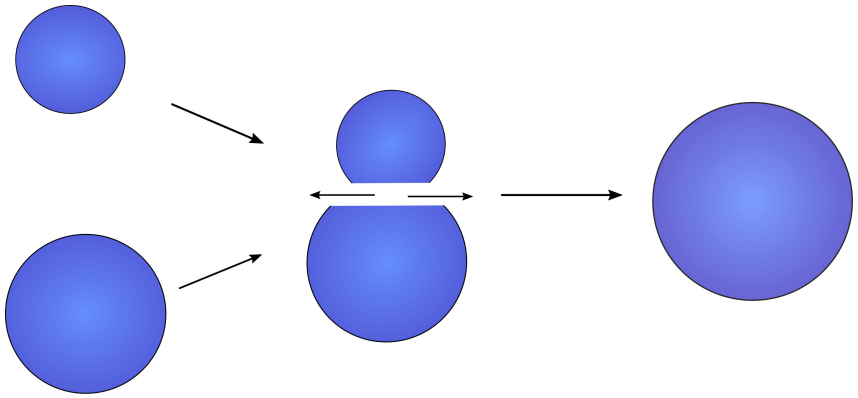


Figure 2.1: Illustration of two coalescing droplets.

Two coalescing droplets of same size A gives droplet of size $2A$. The diameter of the droplet $2A$ is $\sqrt[3]{2}$ times the diameter of A. If the new droplet coalesces again with a droplet of size A again the size becomes $\sqrt[3]{3}$ times the diameter of A and so on. Since the coalescence depends on collisions of the droplets the rate of the coalescence can be described by the concentrations of the two droplet sizes being considered.

2.3 Interfacial Transfer

The separation of the two phases also depends on the ability to transfer dispersed phase to its continuous medium across the interface between the continuous phases. The interfacial transfer can be compared to the coalescence of the droplets, where the continuous phase can be looked at as a droplet of infinitely large diameter. As the dispersed droplets reaches the interface of the continuous phase the droplets will collide with it and coalesce.

The rate of the interfacial transfer will be given by the concentration of the droplets and the available area of the interface, A^{int}.

2.4 Oil-Water Emulsions

In oil-water mixtures, there will be an emulsion layer present between the oil and water phase. The point where the dispersed phase in the emulsion layer switches to its continuous phase are determined by the phase inversion point [7, 11]. The mixture viscosity increases at this

point and it can be used to find the water cut where the phase inversion takes place. The phase inversion point will depend on the droplet sizes in the emulsion. The water cut at the inversion point, WC^p, was in this work chosen to be 0.7.

Since the viscosity changes with the water cut [11], Stokes' terminal velocity should also be affected by the water cut. The adjusted velocity will adjust the velocity by the increased viscosity.

Chapter 3

Modelling of Bulk Separators

The gravity separators is modelled with the assumption that there is an initial separation at the inlet of the separators. In this initial separation the feed stabilizes and reaches equilibrium. The bulk separator models in this work focus on oil-water separation and it is assumed no dispersed gas in the liquids after the initial separation. It is therefore not modelled any flash of gas inside the first stage separator (GS1) after the initial separation, and the second stage gravity separator (GS2) was modelled as a two phase separator. It was also assumed that no liquids is dispersed in the gas in the first stage gravity separator.

The separation of liquids was mainly affected by the difference in horizontal residence time (convective flow) compared with the vertical residence time (adjusted Stokes' terminal velocity). The model solves the transfer and coalescence of 10 droplet sizes in the range $150\,\mu\mathrm{m}$ to $300\,\mu\mathrm{m}$ in diameter. It was assumed that the coalescence and separation happens inside the whole vessel volume. It is therefor no separate coalescer inside the separator and the initial separation happens before entering the vessel. The size distribution of droplets after the initial separation in GS1 and GS2 was chosen to the smallest size, except for the oil phase in GS2 where distribution from GS1 was know.

A steady state three phase gravity separator was first developed and then with small changes modified to steady state GS2 and the two dynamic models. The steady state models was implemented and solved using CasADi in MATLAB with the IPOPT-optimizer to solve the steady state problem, shown in section 3.1.4. The dynamic models was solved and simulated in Simulink with the ODE15s solver. This chapter will start by describing the three phase model and study the steady state solutions. The chapter also shows an analysis of the dynamic model.

3.1 First Stage Gravity Separator

This section describes the development of the models of the steady state and dynamic three phase gravity separator. The model is developed with focus on dynamics and most of the model applies in both transient cases and steady state. The section ends with analysis of the steady state solutions and analysis of the dynamic simulation.

3.1.1 Model Assumptions and Description

It is assumed that an initial separation happens at the inlet of the separator, which stabilizes the feed into one layer of oil and one layer of water. The oil phase is fed with oil dispersed with water droplets of the smallest droplet size and the water phase is fed with water dispersed with oil droplets of the smallest droplet size. The initial separation is determined by the fractions γ_o and γ_w. γ_o is the fraction of oil in feed going into the oil phase and γ_w is the fraction of water in feed going into the water phase. The idea of using γ_o and γ_w to describe the initial separation, is based on the same principles used by Backi [8].

It was assumed that the liquid level and the water level are uniformly distributed through the separator. It was also assumed that there is no volume in the separator taken up by the oil-water emulsion. The emulsion layer was not taken into consideration other than the possible influence on the interfacial transfer rate constant and the adjusted Stokes' velocity. An illustration of the gravity separator modelled in this work, with the flows and levels, are shown in Figure 3.1.

The disturbances to the system is the volumetric feed flow of gas, F_{in}^{g}, volumetric feed flow of liquid, F_{in}^{l} and the water cut in the liquid feed, ϵ_{in}.

The gas is taken out at an outlet in the top of the separator. The oil product goes over the weir and out right of the weir, F_R. The produced water is taken out left of the weir, F_L. The model calculates the amount of water dispersed in oil, (OiW), and the amount of water dispersed in oil, (WiO). In the steady state model, the liquid product flows (F_L and F_R) are calculated from volume balance to keep the water level, h_{lw} and the liquid level, h_l, at wanted steady state values. The dynamic model takes the outflows as inputs and calculates the levels. The product flows was modelled with perfect flow control, without coke dynamics and influence on flow

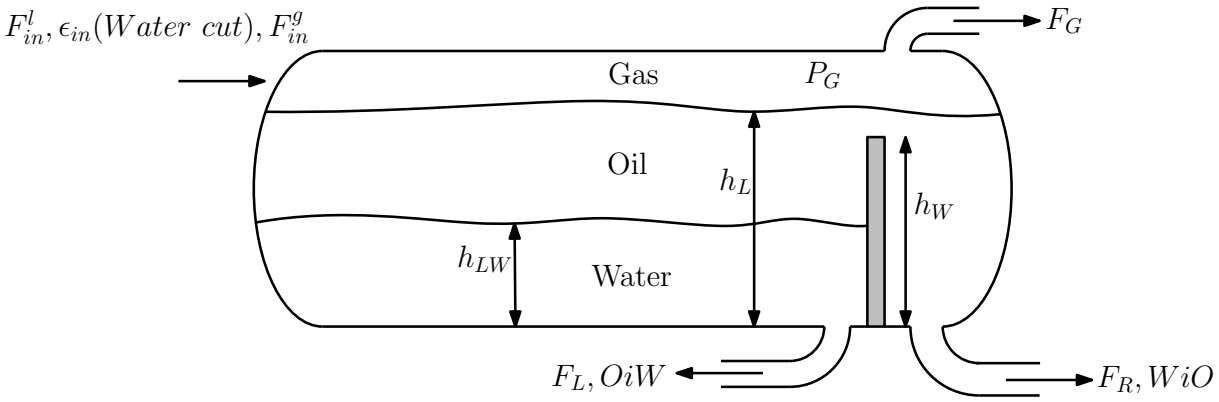


Figure 3.1: Illustrations of three phase gravity separator with flows and levels. h_{lw} and h_l is the water level and liquid level respectively. P_G is the gas pressure. h_w is the height of the weir. F_G, F_R and F_L is the flow of produced gas, oil and water respectively. F^l_{in} and ϵ_{in} is the volumetric flow of liquid feed and water cut of the liquid feed. F^g_{in} is the volumetric flow of gas feed.

from changes in pressure. The pressure will still be influenced by the flows and levels due to the change in available volume for gas under transient conditions.

The transfer of the dispersed droplets through the control volumes in the model are either through convective flow in horizontal direction, non-convective flow in vertical direction or through interfacial transfer. At the weir, the convective flows was modelled to turn into vertical direction. The non-convective flows and interfacial transfer was removed because they became unnecessary. The convective flows through the oil phase and the water phase depend on their respective product flows. The velocity profile was assumed to be uniform over the cross section of the phase. It was also assumed that the oil phase is combined at the weir and flows over the weir while the water is combined into the water outlet. An illustration of the flows and the control volumes in the separator model is shown in Figure 3.2.

3.1.1.1 Control Volumes

The separator model is divided into control volumes such that there is one layer of control volumes over the weir height and one column of control volumes to the right of the weir. The size of the control volumes depends on the two levels. The top layer of control volumes will depend on the liquid level and the control volumes under the weir height will depend on the

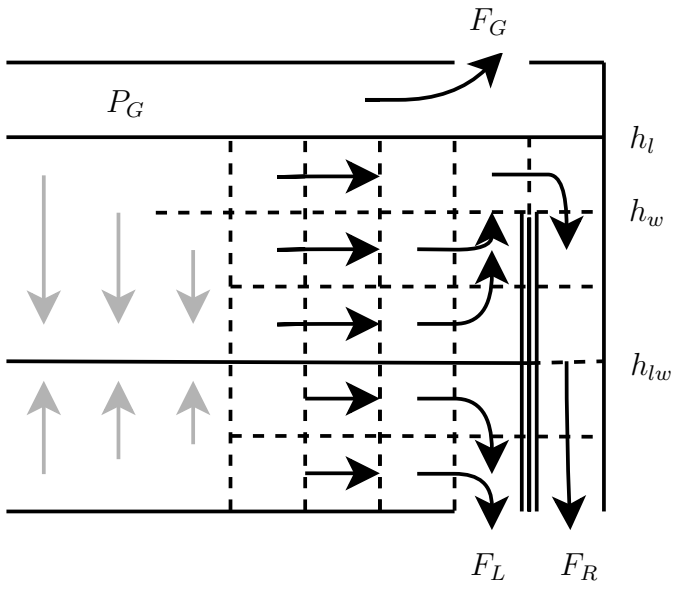


Figure 3.2: Illustrations of flows and control volumes. The convective flows of the two liquid phases depends on their respective product flow (F_R, F_L). The velocities in the two phases are uniform in their cross section and the flows are combined at the weir. The OiW has non-convective flow upward and the WiO has non-convective flow downwards. The gas is modelled in one control volume and the pressure (P_G) depends in the product gas flow (F_G). The size of the control volumes under the weir height h_w depends on the water level h_{lw} while the size of the control volumes in the top layer depends on the liquid level.

water level. The control volumes are shown in Figure 3.2. The water phase and oil phase are divided into $n_{z,w}$ and $n_{z,o}$ number of layers in vertical (z) direction, where the top oil layer is over the weir height. There is n_x control volumes in the horizontal (x) direction, where the rightmost control volumes are right of the weir.

The layers of control volumes under the water level have equal thickness and the layers over the water level and under the weir height have equal thickness. The heights from the separator bottom to the top of layer j are given by

$$h_j = \begin{cases} \frac{j}{n_{z,w}} h_{lw} & \text{if} \quad 0 < j \leq n_{z,w} \\ h_{lw} + \frac{j - n_{z,w}}{n_{z,o} - 1}(h_w - h_{lw}) & \text{if} \quad n_{z,w} < j < n_{z,w} + n_{z,o} \\ h_l & \text{if} \quad j = n_{z,w} + n_{z,o} \end{cases} \tag{3.1}$$

The cross section area A_j under the layer heights are given by

$$A_j = r^2 \cos^{-1}\left(1 - \frac{h_j}{r}\right) - (r - h_j)\sqrt{h_j(2r - h_j)} \tag{3.2}$$

where r is the radius of the separator. A_j is the area under the height h_j as a part of the cross section circle of the separator. This is explained further in Appendix B.1. The cross section area of a specific layer j, A_{cj} is given by

$$A_{cj} = \begin{cases} A_j & \text{if} \quad j = 1 \\ A_j - A_{j-1} & \text{if} \quad j > 1 \end{cases} \tag{3.3}$$

The heights and cross section of the control layers in the separator model and how they changes depending on the water level is illustrated in Figure 3.3. If the water level increases the thickness of the layers under the water level increases evenly and the thickness of the layers over the water level (and under h_w) decreases evenly.

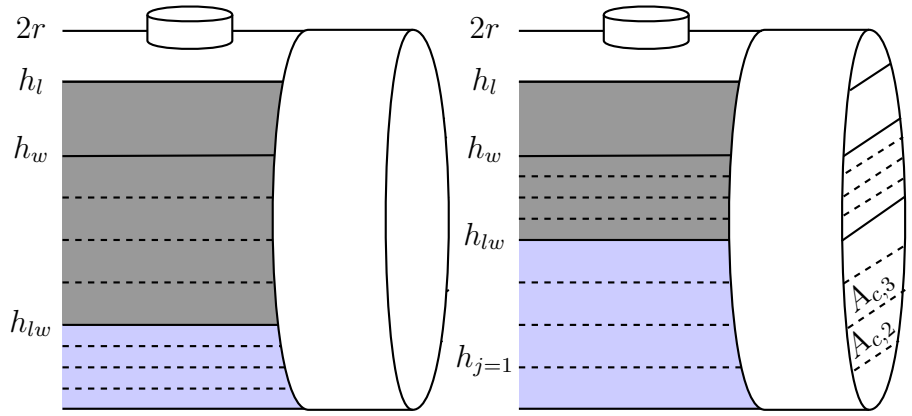


Figure 3.3: Illustration of how the layers of control volumes depend on the water level. h_l, h_w and h_{lw} are liquid level, weir height and water level respectively. r is the radius of the separator. h_j is the height of layer number j from the bottom, and $A_{c,j}$ is the cross section area of layer j. The thickness of the layers under h_{lw} is the same and likewise is the layers under h_w and over h_{lw}. The water level does not differ in horizontal direction, and the figure shows two different water levels at different time instants.

The volumes of the control volumes at layer j, $V_{cv,j}$, is given by

$$V_{cv,j} = A_{cj}\frac{L}{n_x} \tag{3.4}$$

where L is the length of the separator and the time derivatives of the volumes at layer j under the weir height are given by

$$\frac{dV_j}{dt} = \frac{L}{n_x}(\frac{dA_j}{dt} - \frac{dA_{j-1}}{dt}) \tag{3.5}$$

$$\frac{dA_j}{dt} = \frac{dA_j}{dh_j}\frac{dh_j}{dt} = \frac{dA_j}{dh_j}\frac{dh_j}{dh_{lw}}\frac{dh_{lw}}{dt} \tag{3.6}$$

$$\frac{dA_j}{dh_j} = 2\sqrt{h_j(2r - h_j)} \tag{3.7}$$

$$\frac{dh_j}{dh_{lw}} = \begin{cases} \frac{j}{n_{z,w}} & \text{if } 0 < j \leq n_{z,w} \\ 1 - \frac{j - n_{z,w}}{n_{z,o} - 1} & \text{if } n_{z,w} < j < n_{z,w} + n_{z,o} \end{cases} \tag{3.8}$$

Deriving the time derivative of the volumes of the control volumes is shown in Appendix B.2. The control volumes over the weir height depends on the total liquid volume and the time derivative of the control volumes over the weir height is given by

$$\frac{dV_{cv,j}}{dt} = \frac{d}{dt}\frac{V_l - V_w}{n_x} = \frac{1}{n_x}\frac{dV_l}{dt} \qquad \text{if } j = n_{z,w} + n_{z,o} \tag{3.9}$$

The horizontal area $A_{t,j}$ over the control volumes at layer j, is given by

$$A_{t,j} = \frac{L}{n_x} 2\sqrt{h_j(2r - h_j)} \tag{3.10}$$

and is derived from the cross section circle and geometric relation in Appendix B.1.

3.1.2 Transfer and Coalescence of Droplets

The main part of this work was the description of the droplets as they move and interacts through the separator. The transfer of the droplets are either due to convective flow in horizontal direction or non-convective flow which depends on Stokes' terminal velocity. The stokes terminal velocity depends on the droplet size, and this model therefore also keeps track of different droplet sizes formed as a result of coalescence. The amount of the droplets of a size class in a control volume is controlled by use of the partial densities of that class. The partial density $\bar{\rho}_y$ of size y in a control volume is

$$\bar{\rho}_y = \frac{m_y}{V_{cv}} \tag{3.11}$$

where m_y is the total mass of the droplets of that size in the control volume and V_{cv} is the volume of the control volume.

The possible transfers of the droplets are through convective flow, non-convective flow or through interfacial transfer through the water-oil interface. Because the droplets also can coalesce into other size classes, the coalescence is also added to the mass balances.

3.1.2.1 Partial Densities

In the oil phase the partial density $\bar{\rho}_y$ of a water droplet size y is given as

$$\bar{\rho}_y = \frac{\sum_\delta m_{y,\delta}}{V_{cv}} = \frac{\rho_w \sum_\delta V_{y,\delta}}{V_{cv}} \tag{3.12}$$

where ρ_w is the real density of clean water and δ indicate the single droplets. The sum of all the partial densities of the water droplet sizes gives the partial density of water $\bar{\rho}_w$

$$\bar{\rho}_w = \sum_y \bar{\rho}_y \tag{3.13}$$

The water cut, WC, which is used both as model output but also used to adjust the Stokes' velocity and the interfacial transfer is given as

$$WC = \frac{V_w}{V_{cv}} = \frac{\bar{\rho}_w}{\rho_w} \tag{3.14}$$

where V_w is the total volume of water in the control volume and V_{cv} is the volume of the control volume. The same calculations of partial densities of oil and the oil cut, $OC = 1 - WC$, can be calculated in the same way using the density of clean oil.

3.1.2.2 Interfacial Transfer

The droplets near the interface can transfer through the interface into their continuous phase. The rates of transfer of the water and oil droplets through the interface are calculated by

$$r_{w,i} = k_w^{int} A_i^{int} WC_{o,i} \tag{3.15}$$

$$r_{o,i} = k_o^{int} A_i^{int} (1 - WC_{w,i}) \tag{3.16}$$

where k_w^{int} and k_o^{int} is chosen rate constants for the interfacial transfer of water and oil respectively. A_i^{int} is the area of the oil-water interface between the control volumes. i indicate the i-th control volume in horizontal direction. $WC_{o,i}$ and $WC_{w,i}$ is the water cut in the oil and water phase side of the interface respectively. A_i^{int} is equal for all i, because of the assumed uniform level. $WC_{o,i} = 0$ means there is no water in the oil and there will therefore be no transfer of water droplets. The same goes for the transfer of oil droplets. There is no transfer of oil droplets

through the interface if $WC_{w,i} = 1$. The interfacial transfer of the dispersed droplet size classes is calculated by multiplying the rate of transfers with the partial densities:

$$\dot{\mathbf{m}}_i^{int} = r_i \bar{\rho}_i \tag{3.17}$$

Here $\dot{\mathbf{m}}_i^{int}$ and $\bar{\rho}_i$ are vectors with elements for each size class, y, and i is the location in horizontal direction. The net volumetric flow over the interface into the water phase at location i is given by

$$F_i^{int} = \frac{\sum_y \dot{\mathbf{m}}_{i,w,y}^{int}}{\rho_w} - \frac{\sum_y \dot{\mathbf{m}}_{i,o,y}^{int}}{\rho_o} \tag{3.18}$$

where $\sum_y \dot{\mathbf{m}}_{i,w,y}^{int}$ and $\sum_y \dot{\mathbf{m}}_{i,o,y}^{int}$ is the total interfacial transfer of mass of water and oil at the interface at i respectively. The net volumetric flow into the water phase through interfacial transfer is

$$F_{tot}^{int} = \sum_i F_i^{int} \tag{3.19}$$

The interfacial transfer just left of the weir is not considered, as the convective flow turns vertically as shown in Figure 3.2.

3.1.2.3 Flows and Velocities

The separator consists of two continuous liquid phases, oil and water. The two phases have different flows and velocities and the flow of oil depends on the oil product which flows over and out right of the weir, F_R , and the water depends on the flow of water going out left of the weir, F_L. In a vertical cross section the horizontal velocities profiles are uniform in each of the flows. The uniform velocity profile and the different cross section areas, A_{cj}, for the layers, j, causes the volumetric flows to be different in each layer. The volumetric flow, q_j, at layer j is given by

$$q_j = \frac{A_{cj}}{A_{lw}} F_L \tag{3.20}$$

A_{lw} is the area under the water level.

Just left of the weir, the flows under the water level are combined into the F_L product flow, and the flows over the water level is combined into the highest layer which goes over and right of the weir into the F_R product flow. The convective flows just left of and right of the weir was shown in Figure 3.2.

The flows will not be constant through the separator. If the interfacial transfer of water is larger than oil, the convective flow under the interface increases while the flow above the interface decreases. The reason for the increased and decreased flows is that the water level is assumed equal in horizontal direction. The net volumetric flows through the interface are added and subtracted from the two flows. The changes in the two flows are distributed to the layers, so that the velocity profile is kept uniform.

Special for the steady state model is that it calculates the product flows after calculating the interfacial flow of liquid. Assuming incompressible fluid the mass balance over the water phase, and the total liquids are held when the two products are calculated from

$$F_L = F_{in}^{wp} + F_{tot}^{int} \tag{3.21}$$

$$F_R = F_{in}^{l} - F_L \tag{3.22}$$

F_{tot}^{int} is the total volumetric flow through the oil-water interface and is calculated with equation (3.19). F_{in}^{wp} is the feed flow of water phase with dispersed oil droplets and is calculated with:

$$F_{in}^{wp} = (1 - \gamma_o)(1 - \epsilon_{in})F_{in}^{l} + \gamma_w \epsilon_{in} F_{in}^{l} \tag{3.23}$$

where ϵ_{in} is the water cut in the liquid feed and γ_o and γ_w is the fraction of oil to oil phase and water to water phase, in the initial separation, respectively.

In transient condition where F_L is lower than F_{tot}^{int} the model allows temporary back flow in the water phase. The feed flow to the first control volumes are still calculated from the liquid feed and not from the product flows and there is no back flow out of the separator. During the transient back flow, the dispersed oil droplets can travel back to the separator inlets. This is further described in Section 3.1.3.4.

Using just Stokes' terminal velocity given in equation (2.1) to calculate the vertical velocity of the droplets, might result in unreal densities of the dispersed phase. If the interfacial transfer of dispersed water droplets is much slower than the non-convective flow of the water droplets, the solved water densities in the oil above the interface can be higher than clean water, which is an unreal solution. Water cuts in the oil continuous phase higher than the phase inversion point, would also be an unreal solution. The problems of unreal water densities are solved by adjusting the Stokes' terminal velocity by the water cut, so that the terminal velocity also

consider hindering of the droplets. The velocity is also adjusted by the water cut, WC^p, of the phase inversion point, to avoid getting to high water cuts in oil phase and to low water cuts in the water phase. The adjusted Stokes' terminal velocity is given by

$$v^a = \begin{cases} v^s \left(\frac{WC^p - WC^{\text{receiver}}}{WC^p} \right)^n & \text{for Water in Oil} \\ v^s \left(\frac{WC^{\text{receiver}} - WC^p}{1 - WC^p} \right)^n & \text{for Oil in Water} \end{cases} \tag{3.24}$$

3.1.2.4 Coalescence of Droplets

The coalescence of the droplets will play a part in the separation because the vertical velocity of the droplets depends on the squared diameter. The model keeps track of the mass of 10 droplet classes and the coalescence reactions between these 10 droplet sizes. The model calculates the changes in partial densities for every droplet size for every control volume. The partial density of a drop size is the total mass of all the droplets with that size divided by the control volume. The 25 possible reactions of coalescence is given in (3.25a) to (3.25y).

$$A + A \xrightarrow{k_1} 2A \quad (3.25a)$$
$$2A + A \xrightarrow{k_2} 3A \quad (3.25b)$$
$$3A + A \xrightarrow{k_3} 4A \quad (3.25c)$$
$$4A + A \xrightarrow{k_4} 5A \quad (3.25d)$$
$$5A + A \xrightarrow{k_5} 6A \quad (3.25e)$$
$$6A + A \xrightarrow{k_6} 7A \quad (3.25f)$$
$$7A + A \xrightarrow{k_7} 8A \quad (3.25g)$$
$$8A + A \xrightarrow{k_8} 9A \quad (3.25h)$$
$$9A + A \xrightarrow{k_9} 10A \quad (3.25i)$$
$$2A + 2A \xrightarrow{k_{10}} 4A \quad (3.25j)$$
$$3A + 2A \xrightarrow{k_{11}} 5A \quad (3.25k)$$
$$4A + 2A \xrightarrow{k_{12}} 6A \quad (3.25l)$$
$$5A + 2A \xrightarrow{k_{13}} 7A \quad (3.25m)$$
$$6A + 2A \xrightarrow{k_{14}} 8A \quad (3.25n)$$
$$7A + 2A \xrightarrow{k_{15}} 9A \quad (3.25o)$$
$$8A + 2A \xrightarrow{k_{16}} 10A \quad (3.25p)$$
$$3A + 3A \xrightarrow{k_{17}} 6A \quad (3.25q)$$
$$4A + 3A \xrightarrow{k_{18}} 7A \quad (3.25r)$$
$$5A + 3A \xrightarrow{k_{19}} 8A \quad (3.25s)$$
$$6A + 3A \xrightarrow{k_{20}} 9A \quad (3.25t)$$
$$7A + 3A \xrightarrow{k_{21}} 10A \quad (3.25u)$$
$$4A + 4A \xrightarrow{k_{22}} 8A \quad (3.25v)$$
$$5A + 4A \xrightarrow{k_{23}} 9A \quad (3.25w)$$
$$6A + 4A \xrightarrow{k_{24}} 10A \quad (3.25x)$$
$$5A + 5A \xrightarrow{k_{25}} 10A \quad (3.25y)$$

The 10 possible droplet sizes are given as

$$d_y = \sqrt[3]{y} \cdot d_A \tag{3.26}$$

where d_A is the size of the smallest droplet.

The rates of net changes in the 10 densities from coalescence is given in a vector R. The rates of changes are calculated assuming first order dynamics in the reactions in (3.25a) to (3.25y).

The rate of a coalescence reaction, r_ξ, where a droplet of size $y = \alpha$ and a droplet of size $y = \beta$ are merging are given by:

$$r_\xi = k_\xi \bar{\rho}_\alpha \bar{\rho}_\beta \tag{3.27}$$

$\bar{\rho}_\alpha$ and $\bar{\rho}_\beta$ are the partial densities of droplet classes. The rates in R are calculated by

$$R = \mathbf{s} \circ (\mathbf{Nr}) \tag{3.28}$$

where $\mathbf{N}$ is a matrix of the stoichiometric coefficients of the reactions and $\mathbf{s}$ is a vector from 1 to 10 used to scale the rates. $\circ$ indicate element wise multiplication ($a = b \circ c$ such that $a_i = b_i c_i$). The rates are scaled to take the mass into consideration. It is not concentration or number of droplets which are used to calculate the rates, and the partial density of y changes y times faster for each formed droplet than for the smallest droplet size A.

3.1.3 Model States and Equations

The three phase gravity separator model calculates the liquid level, h_l, the water level, h_{lw}, the gas pressure in the tank P and 10 partial densities for every control volume in the separator model.

3.1.3.1 Liquid Level Calculation

It is assumed that no gas will flash from the liquid volume, V_l, under the liquid level. Assuming incompressible fluids the change in liquid volume is given by

$$\frac{dV_l}{dt} = F_{in}^l - F_R - F_L \tag{3.29}$$

where F_{in}^l is the volumetric flow of liquid feed and F_R and F_L are liquid product flows. The volume depends on the area under the level in the cross section circle of the separator, which again depends on the level. The change in volume described by the angle to the liquid level, θ, and liquid level is given by

$$\frac{dV_l}{dt} = \frac{d(A_l L)}{dt} = L\frac{dA_l}{dt} = L\frac{dA_l}{dh_l}\frac{dh_l}{dt} = L\frac{dA_l}{d\theta}\frac{d\theta}{dh_l}\frac{dh_l}{dt} \tag{3.30}$$

$\frac{dA}{d\theta}$ and $\frac{d\theta}{dh_l}$ are trigonometric relations given in Appendix B.1. Combining the geometric relations with (3.30) and solving for $\frac{dh_l}{dt}$ gives the equation

$$\frac{dh_l}{dt} = \frac{1}{2L\sqrt{h_l\left(2r - h_l\right)}}\frac{dV_l}{dt} \tag{3.31}$$

used to solve the liquid level in the separator. r and L is the radius and length of the separator.

3.1.3.2 Water Level Calculation

As in the previous section, the assumptions are that there is no gas in the liquids and that the liquids are incompressible. The change in volume, V_{lw}, under the water level, h_{lw}, is therefore

$$\frac{dV_{lw}}{dt} = \text{Volumetric flow in} - \text{Volumetric flow out} \tag{3.32}$$

$$\frac{dV_{lw}}{dt} = F_{in}^{wp} + F_{in}^{int} - F_L \tag{3.33}$$

where F_{in}^{wp} is the part of the feed flow that goes directly into the water phase. F_{tot}^{int} is the volumetric flow into the water phase from interfacial transfer and F_L the product flow of water.

The geometric relations are the same as for the liquid level, except that the length is to weir and not the whole separator length L. The equation for solving the water level is

$$\frac{dh_{lw}}{dt} = \frac{1}{2L\frac{n_x-1}{n_x}\sqrt{h_{lw}\left(2r - h_{lw}\right)}}\frac{dV_{lw}}{dt} \tag{3.34}$$

where n_x is the number of control volumes in horizontal direction and r is the separator radius.

3.1.3.3 Gas Pressure Calculation

The equation describing the change in the gas pressure is calculated from the molecular balance of the gas.

$$\frac{dn^g}{dt} = \dot{n}_{in}^g - \dot{n}_{out}^g \tag{3.35}$$

It is assumed that no gas is flashed out of the liquid in the tank. The feed rate, $\dot{n}_{in}^g$, is given by

$$\dot{n}_{in}^g = \frac{\dot{m}_{in}^g}{M_g} = \frac{\rho_g F_{in}^g}{M_g} \tag{3.36}$$

where M_g is the molar mass. ρ_g is the density of the gas, at inlet pressure, and F_{in}^g is the volumetric flow of gas after the gas is flashed out before inlet of separator. The rate of gas

moles going out of the tank, $\dot{n}^g_{out}$, is given by ideal gas law and the volumetric outflow of gas, F_G.

$$\dot{n}^g_{out} = \left(\frac{P\dot{V}}{RT}\right)^g_{out} = \frac{P}{RT}F_G \tag{3.37}$$

R is the universal gas constant and T is the temperature in the tank. $\dot{V}$ is the rate of gas volume going out. Using the product rule of derivation on the ideal gas law gives

$$\frac{dn^g}{dt} = \frac{d}{dt}\frac{PV_g}{RT} = \frac{1}{RT}\left(V_g\frac{dP}{dt} + P\frac{dV_g}{dt}\right) \tag{3.38}$$

The volume of the gas, V_g is the difference between the total volume of the tank and the liquid volume. The time derivative of the gas volume is therefore the negative of the rate of change in V_l that was given in (3.29). Solving for the time derivative of pressure gives the equation used for solving the gas pressure in the tank

$$\frac{dP}{dt} = \frac{RT}{V_g}\left(\dot{n}^g_{in} - \dot{n}^g_{out}\right) + \frac{P}{V_g}\frac{dV_l}{dt} \tag{3.39}$$

3.1.3.4 Partial Density Calculation

The mass balances for each droplets size class, in a control volume is

$$\frac{d\mathbf{m}}{dt} = \dot{\mathbf{m}}^{in} - \dot{\mathbf{m}}^{out} + \dot{\mathbf{m}}^{gen/lost} \tag{3.40}$$

The vector of masses, $\mathbf{m}$, and the corresponding rates, $\dot{\mathbf{m}}$, consists of 10 elements corresponding to the 10 droplet sizes. The rates going in and out of the control volume is both the convective flow and the non convective flow of droplets. $\dot{\mathbf{m}}^{gen/lost}$ is the vector of the rates at which the droplets are being generated or lost due to coalescence. The masses and densities used in the rest of this section are vectors even tho they will not be written on the vector for. The are still representing all 10 droplet size classes and multiplication of vectors are element wise if not stated otherwise.

The control volumes in the separator is divided into different parts depending on what goes on in the control volume. The coalescence takes place in each of the control volumes.

1. **Under Water Level, Left of Weir:**

   Dispersed oil in water phase floats up and transfer through the interface. Convective

flow increases with $F^{int} > 0$. Flow is combined to vertical flow downwards just left of weir and goes out with F_L. Interfacial transfer and non-convective flow at weir are not considered. Temporary backflow of droplets are allowed when F_L is low.

2. **Over Water Level, Left of Weir:**

   Dispersed water in oil phase fall down and transfer through interface. Convective flow decreases with $F^{int} > 0$. Flow in layers under weir height is combined to vertical flow upwards just left of weir and flows into top layer, which flows over the weir. Interfacial transfer and non-convective flow at weir is neglected.

3. **Right of Weir:**

   Water is dispersed in oil phase. Only convective flows are considered. In the top layer the flow comes from left and turns downwards, while the other control volumes have only vertical flows. The volumetric flow is equal to F_R. Control volumes under water level is occupied by oil with dispersed water and the solved partial densities are of WiO.

Common all control volumes is that the model allows changes in size. The control volumes under weir height increase or decrease in size with change in water level and the top layer increase in size with increase in liquid level. With the help of the product rule of derivation on the left hand side of the mass balance in (3.40) gives

$$\frac{dm}{dt} = \frac{d(\bar{\rho}V_{cv})}{dt} = \bar{\rho}\frac{dV_{cv}}{dt} + V_{cv}\frac{d\bar{\rho}}{dt} \tag{3.41}$$

where V_{cv} is the volume of the control volume and $\bar{\rho}$ is the vector of the partial densities of the 10 droplet classes. The change in the control volume size is given in equation (3.9).

3.1.3.5 Under Water Level, Left of Weir

The part of the separator in focus is shown in Figure 3.4.

Higher interfacial transfer of water than oil from the dispersed phases to their continuous phase, gives a total volumetric flow of water over the interface. It will in these cases be necessary to take out more water product than the flow to water phase in the initial separation at the outlet. It can in transient cases be lower outflow of water than the interfacial flow of water. For example when we want rise the water level rapidly. If the water outlet is zero, and there is gained water through the interface, the internal flows might change direction and the flow goes back up the

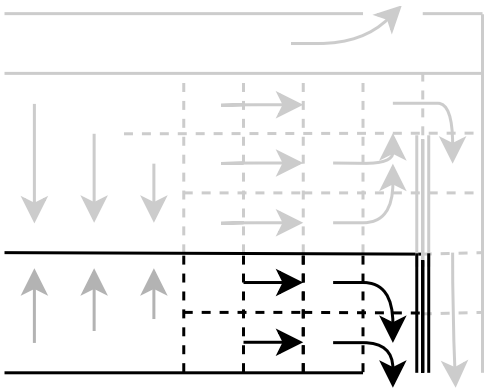


Figure 3.4: Control volumes under water level and left of weir.

separator. The dispersed oil droplets will then transfer back up the separator with the new convective flow. It is assumed that the convective flow through the oil phase is to high for it to have any backflow present.

Since the balance should allow backflow, it is important to know which side the droplets come from. The rate of mass transfer from convective flow, $\dot{m}^{conv}$, is therefore given by

$$\dot{m}^{conv} = (\bar{\rho} v_x A_c)^{in} - (\bar{\rho} v_x A_c)^{out} \tag{3.42}$$

$$\dot{m}^{conv} = \max(q_{i,j}\bar{\rho}^{left}, -q_{i,j}\bar{\rho}^{right}) - \max(q_{i+1,j}\bar{\rho}, -q_{i+1,j}\bar{\rho}) \tag{3.43}$$

A_c is the cross section area through which the convective flow, with velocity, v_x, passes through. $\bar{\rho}^{left}$ and $\bar{\rho}^{right}$ are the densities in the control volume left and right of the current control volume. At the left most control volume ρ^{left} is equal the feed densities. The mass balance (3.40) combined with (3.43), non-convective transfer and coalescence term gives

$$\bar{\rho}\frac{dV_{cv}}{dt} + V_{cv}\frac{d\bar{\rho}}{dt} = \dot{m}^{conv} + \bar{\rho}^{under} v_t^{under} A_t^{under} - \bar{\rho} v_t^{over} A_t^{over} + RV_{cv} \tag{3.44}$$

v_t indicates the terminal velocity of the non-convective transfer, either in or out of the control volume. v_t is calculated using the adjusted Stokes' velocity (3.24). R is a vector specifying the rates of the coalescence. A_t is the horizontal area over or under the control volume and is calculated using (3.10).

Solving (3.44) for the change in density gives

$$\frac{d\bar{\rho}}{dt} = \frac{\dot{m}^{conv}}{V_{cv}} + \left(\bar{\rho}^{under} v_t^{under} A_t^{under} - \bar{\rho} v_t^{over} A_t^{over}\right)\frac{1}{V_{cv}} + R - \frac{\bar{\rho}}{V_{cv}}\frac{dV_{cv}}{dt} \tag{3.45}$$

which is used to solve the partial densities.

At the bottom layer $j = 1$ there is no non-convective transfer from the bottom and $\bar{\rho}^{under} v_t^{under} A_t^{under}$ is zero. In the control volume just under the interface $\bar{\rho} v_t^{over} A_t^{over}$ is changed to the $\dot{m}^{int}$-vector of the interfacial mass transfer rates, calculated by equation (3.17).

Just left of the weir all non-convective and interfacial transfer are neglected and equation (3.45) are replaced with

$$\frac{d\bar{\rho}}{dt} = \bar{\rho}^{left} \frac{q_j}{V_{cv}} + \bar{\rho}^{over} \frac{\sum_{k=j+1}^{n_{zw}} q_k}{V_{cv}} - \bar{\rho} \frac{\sum_{k=j}^{n_{zw}} q_k}{V_{cv}} + R - \frac{\bar{\rho}}{V_{cv}} \frac{dV_{cv}}{dt} \tag{3.46}$$

The first term on the right hand side, is the transfer from the left. Since the interfacial transfer is neglected there, is no backflow in these control volumes and $\dot{m}^{conv}$ is not needed. The first and the second term on the right hand side, is the transfer into the control volume from the left and from the top. The third term is the transfer out of the bottom of the control volume. The second term is zero at the interface.

3.1.3.6 Over Water Level, Left of Weir

The part of the separator in focus is shown in Figure 3.5.

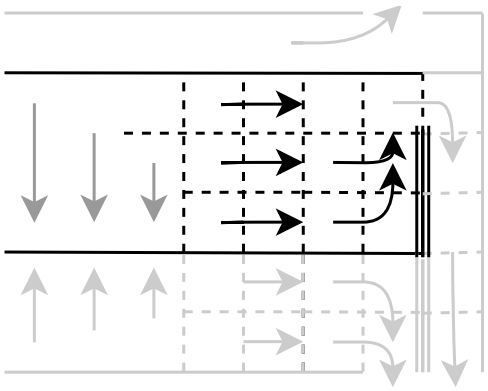


Figure 3.5: Control volumes over water level and left of weir.

The equation used to solve the partial densities are found as in the last section. The differences are that it is assumed no backflow, the non-convective terminal velocity changes direction and the convective flow at the weir goes up and over the weir instead of down into the water outlet. The equation used to solve the densities is

$$\frac{d\bar{\rho}}{dt} = (\bar{\rho}^{left} - \bar{\rho}) \frac{q_j}{V_{cv}} + \left(\bar{\rho}^{over} v_t^{over} A_t^{over} - \bar{\rho} v_t^{under} A_t^{under}\right) \frac{1}{V_{cv}} + R - \frac{\bar{\rho}}{V_{cv}} \frac{dV_{cv}}{dt} \tag{3.47}$$

where $\bar{\rho}^{over} v_t^{over} A_t^{over}$ is zero in the top layer and at the interface $\bar{\rho} v_t^{under} A_t^{under}$ is changed with the interfacial mass transfer rates given by the $\dot{m}^{int}$-vector.

At the weir the densities are calculated by

$$\frac{d\bar{\rho}}{dt} = \bar{\rho}^{left} \frac{q_j}{V_{cv}} + \bar{\rho}^{under} \frac{\sum_{k=n_{zw}+1}^{j-1} q_k}{V_{cv}} - \bar{\rho} \frac{\sum_{k=n_{zw}+1}^{j} q_k}{V_{cv}} + R - \frac{\bar{\rho}}{V_{cv}} \frac{dV_{cv}}{dt} \tag{3.48}$$

where the first, second and third term on the right hand side, represent the convective transfer into the control volume from the left, from the bottom and transfer out of the top of the control volume, respectively. The second term is zero at the interface.

3.1.3.7 Right of Weir

The part of the separator in focus is shown in Figure 3.6.

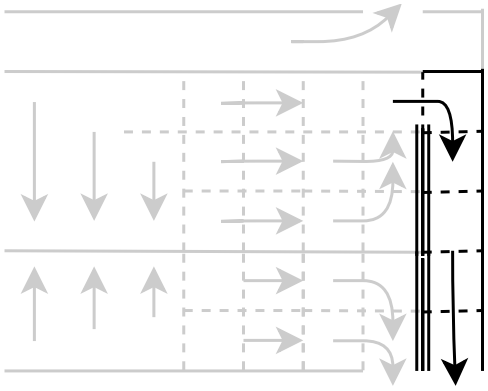


Figure 3.6: Control volumes right of weir.

In all the control volumes right of the weir, the non-convective transfer is neglected and all convective flows are equal to the oil product flow F_R. The coalescence term is still added to the equations, but since the flow has passed the weir the coalescence will no longer have an effect to the separation in this separator.

The partial densities, in the control volume over the weir height, are calculated by

$$\frac{d\bar{\rho}}{dt} = \left(\bar{\rho}^{left} - \bar{\rho}\right) \frac{F_R}{V_{cv}} + R - \frac{\bar{\rho}}{V_{cv}} \frac{dV_{cv}}{dt} \tag{3.49}$$

and the partial densities in the other control volumes right of the weir is calculated by

$$\frac{d\bar{\rho}}{dt} = \left(\bar{\rho}^{over} - \bar{\rho}\right) \frac{F_R}{V_{cv}} + R - \frac{\bar{\rho}}{V_{cv}} \frac{dV_{cv}}{dt} \tag{3.50}$$

3.1.4 Solving the Steady State

The model was solved for steady state by implementing and solving the model in MATLAB with CasADi. The model calculates the needed product flows at desired levels setpoints. The model solves the different partial densities for each control volume and the gas pressure and levels.

The solver selected to solve the steady state was IPOPT (Interior Point Optimizer) which is able to constrain the states and outputs. Solvers without state constraints might have difficulties solving the steady state solution at different initial guesses, because negative partial densities lead to an intermediate solution it is unable to recover from. The cost function J to be minimized in the solver was set to zero, so that IPOPT becomes a root-finder.

The model has to solve $10(n_{zo} + n_{zw})n_x$ partial densities, the liquid level, the water level and the gas pressure. The number of states to be solved is $n_s = 10(n_{zo} + n_{zw})n_x + 3$. The steady state problem is given as

$$\min_{X} \quad J = 0 \cdot X^\intercal X \tag{3.51a}$$

$$\textbf{s.t.} \quad \dot{X} = 0, \qquad \chi = 1, ..., n_s \tag{3.51b}$$

$$X_{min} \leq X \qquad \chi = 1, ..., n_s - 3 \tag{3.51c}$$

$$X_{min} \leq X \leq X_{max} \qquad \chi = n_s \tag{3.51d}$$

$$X = X_{sp} \qquad \chi = n_s - 2, n_s - 1 \tag{3.51e}$$

$$u^{min} \leq u \leq u^{max} \tag{3.51f}$$

$$d = d^{given} \tag{3.51g}$$

where X is the vector of states, and $\dot{X}$ is the time derivatives of the states. u and d are vectors of the system input and disturbance respectively. χ indicate the element in the vectors, where $\chi = n_s - 2$, $\chi = n_s - 1$ and $\chi = n_s$ indicate liquid level, water level and pressure respectively. Since the goal is to solve the steady state solution, all inputs and disturbances are given. Desirable liquid level and water level is also given. The gas is not the main goal and the input was set to an inequality constraint, with the boundaries (u^{min}, u^{max}) of $\pm 1\%$ of an initial guess, $F_{G,0}$, and gas pressure somewhere between $70\,\mathrm{bar}$ and $90\,\mathrm{bar}$.

Since the liquid product flows is calculated directly in the model and not given as inputs, the

problem (3.51) is over-constrained. The problem has $n_s + 1$ variables and $n_s + 2$ equality constraints. The solver algorithm relaxes the bounds on the equality constraints, when the problem is over-constrained. The equality constraints is therefore made into inequality constraints such that the solver are able to find the solution to the problem and then find its way back to the equality constraints. The parameters used in the model are given in Table 3.1.

Table 3.1: Parameters, system inputs and disturbances for nominal case. The parameters describing the feed is from [8].

Parameter	Symbol	Value	Unit
Separator radius	r	2	m
Separator length	L	15	m
Weir height	h_w	2	m
Molar mass of gas	M_g	0.01604	$\mathrm{kg\,mol^{-1}}$
Density of gas	ρ_g	49.7	$\mathrm{kg\,m^{-3}}$
Density of oil	ρ_o	831.5	$\mathrm{kg\,m^{-3}}$
Density of water	ρ_w	1000	$\mathrm{kg\,m^{-3}}$
Viscosity of oil	μ_o	0.001	$\mathrm{kg\,s^{-1}\,m^{-1}}$
Viscosity of water	μ_w	0.0005	$\mathrm{kg\,s^{-1}\,m^{-1}}$
Temperature	T	328.5	K
Adjustment of Stokes' velocity	n	1	-
Gas constant	R	8.314	$\mathrm{J\,K^{-1}\,mol^{-1}}$
Gas feed flow	F_{in}^g	0.456	$\mathrm{m^3\,s^{-1}}$
Liquid feed flow	F_{in}^l	0.590	$\mathrm{m^3\,s^{-1}}$
Initial Gas product flow	$F_{G,0}$	0.456	$\mathrm{m^3\,s^{-1}}$
Water cut in feed	ϵ_{in}	0.15	-
Initial droplet size	d_A	150	µm
Fraction water to water	γ_w	0.4	-
Fraction oil to oil	γ_o	0.99	-
Rate constant, coalescence	k_ξ	1 for all ξ	$\mathrm{m^3\,s^{-1}\,kg^{-1}}$
Rate constant, interfacial flow of water	k_w^{int}	0.5	$\mathrm{m\,s^{-1}}$
Rate constant, interfacial flow of oil	k_o^{int}	0.5	$\mathrm{m\,s^{-1}}$

3.1.5 Steady State Solutions

The nominal steady state solution was solved with water level and liquid level at $1.8\,\mathrm{m}$ and $3.5\,\mathrm{m}$, respectively. How the steady state product flow rates and their WiO and OiW concentrations depends on the water level and liquid levels are shown in Figure 3.7 and 3.8. The water product was purest with water level as close as possible to the weir height. Higher water level resulted in water product with lower flow rate and OiW concentration, and oil product with higher flow rate and WiO concentration.

The nominal levels was chosen with some back off, to keep the water phase from flowing into the oil product and to avoid to high interaction between level changes and gas pressure. The nominal product flows of oil and water were calculated to 0.5244 $\mathrm{m^3\,s^{-1}}$ and 0.0656 $\mathrm{m^3\,s^{-1}}$ respectively. The water cut solution inside the separator is given in Table 3.2.

Table 3.2: Nominal water cut solutions. The water cuts are given in percentage. The vertical line indicate the weir position and the horizontal lines indicate the weir height and oil-water interface.

8.71	7.84	7.04	6.30	5.63	5.02	4.46	4.38	4.38
8.75	7.91	7.15	6.45	5.81	5.22	4.67	3.82	4.38
4.57	3.89	3.65	3.47	3.30	3.13	2.97	2.97	4.38
96.61	98.52	99.19	99.49	99.63	99.72	99.77	99.77	4.38
98.22	99.75	99.96	99.99	100.00	100.00	100.00	99.86	4.38

How the steady state products depended on different feed conditions; liquid flow and water cut, are shown in Figure 3.9 and 3.10 respectively. The residence time of the droplets inside the separator is shown in 3.11 and 3.12, for changes in the water level and liquid level respectively.

The scripts used to solve the steady state solution with CasADi are given in Appendix C.1 and C.2. This includes implemented model functions and definition of the steady state problem and solver with constraints. The parameters are given in C.3. The Stokes' terminal velocity and the rate of coalescence is calculated with the function C.4 and C.5, respectively.

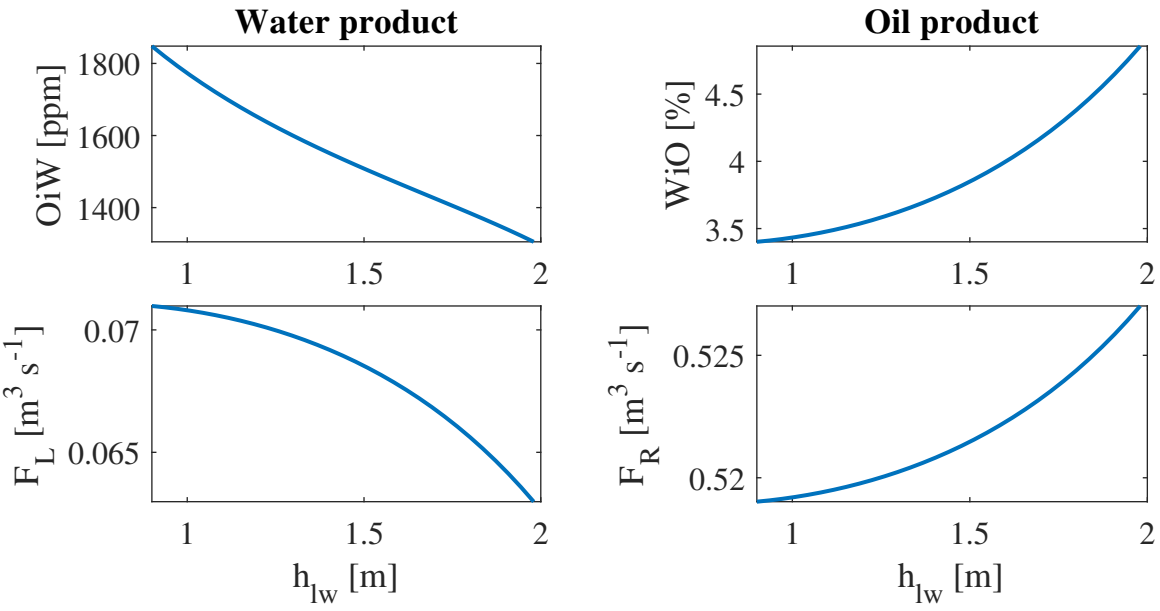


Figure 3.7: Liquid steady state products against water level. Increased water level decreases the OiW concentration and water product flow rate. WiO concentration and oil product flow rate increases. The separation of both dispersed phases depends on the water level as the interfacial area and the horizontal residence time of both phases depends on the water level.

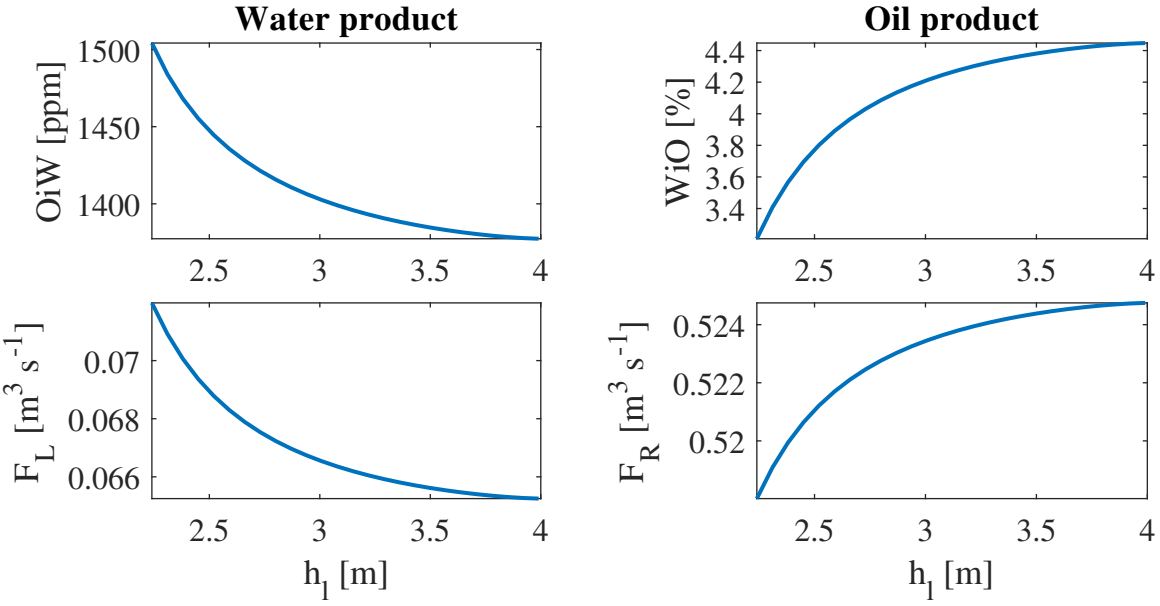


Figure 3.8: Liquid steady state products against liquid level. Increased water level decreases the OiW concentration and water product flow rate. WiO concentration and oil product flow rate increases. The separation of both dispersed phases depends on the liquid level as change in horizontal residence time in oil phase increases the interfacial flow of which leads to lower horizontal residence time in the water. Interfacial area do not depend on the liquid level.

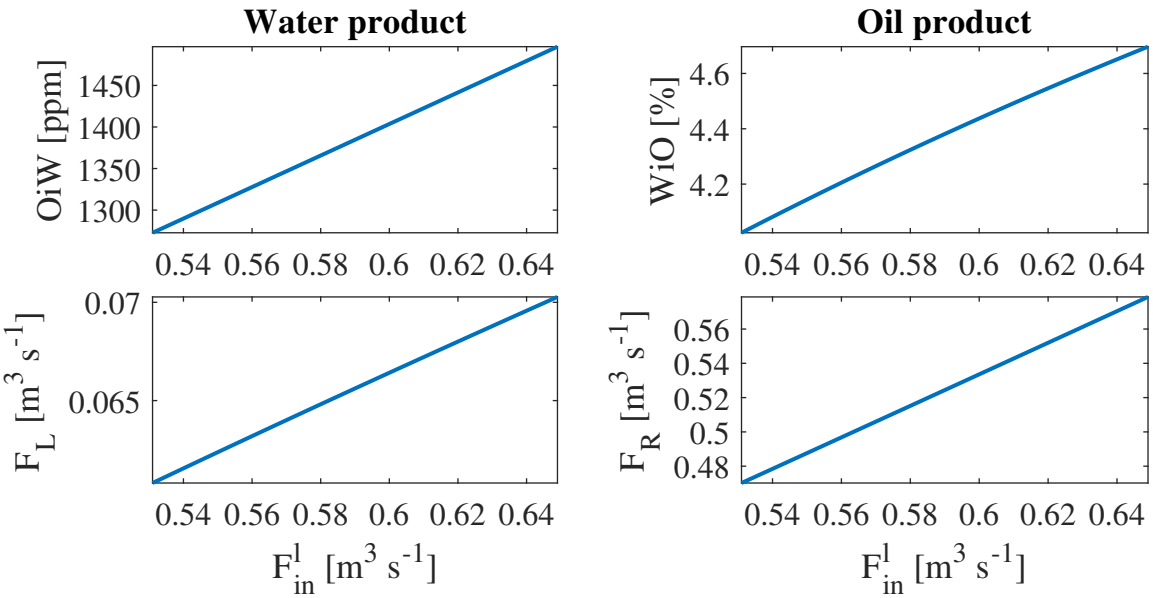


Figure 3.9: Steady state products against liquid feed flow. The increased feed flow resulted in increased flows in both phases. The increased flow resulted in shorter residence time and poorer separation.

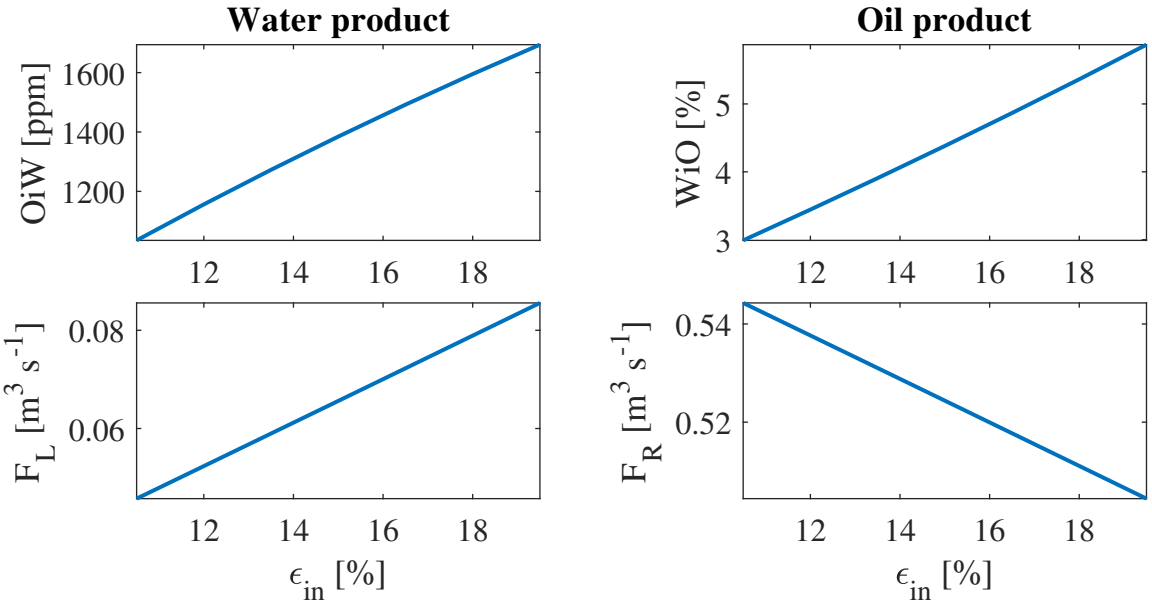


Figure 3.10: Steady state products against inlet water cut. The increased water cut at the inlet resulted in increased flow of water phase and decreased flow of oil phase. This was because the model assumes a fraction of the water goes directly into the water phase. Since the amount of dispersed WiO also increased at the inlet the OiW is increased in the product. The OiW concentration also increased because of the larger flow of water phase and smaller residence time.

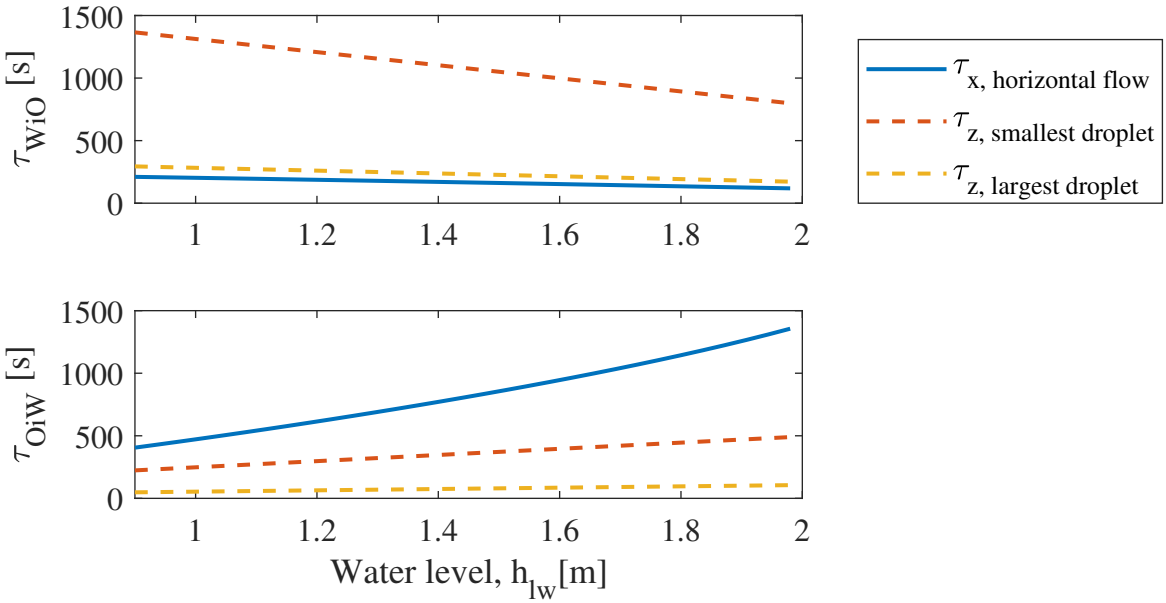


Figure 3.11: Horizontal and vertical residence time against changing water level. τ_{WiO} and τ_{OiW} indicate the residence time for water droplets in oil phase and oil droplets in water phase, respectively. The vertical residence time is shown for the smallest and largest droplet. The volumetric horizontal flow used is through the fifth horizontal control volumes and the vertical velocity is adjusted with the mean WC in the respective phase.

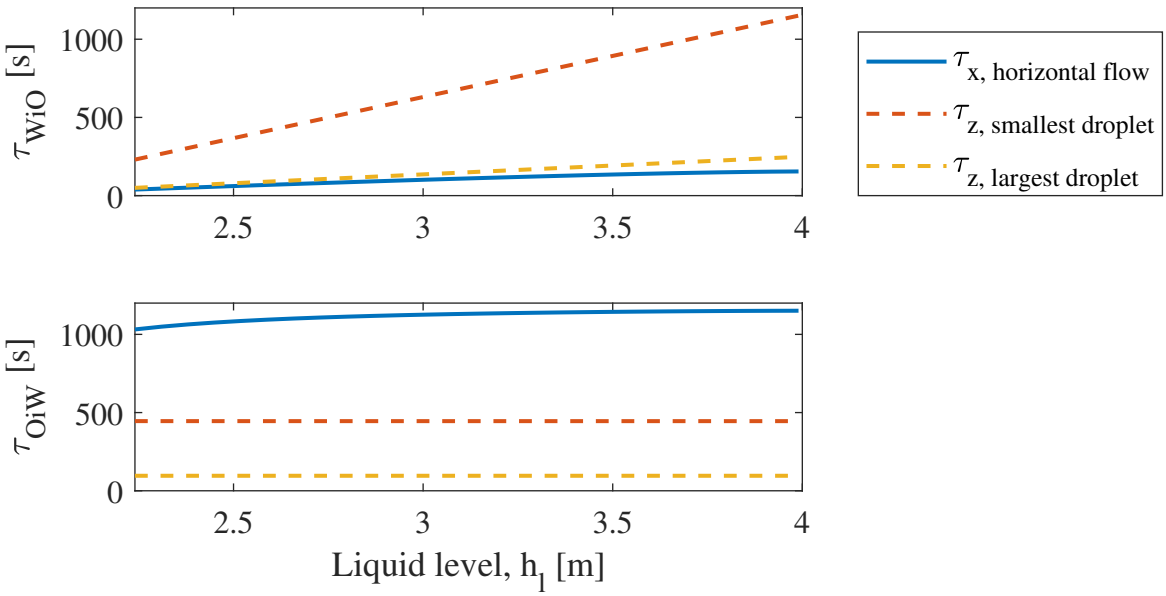


Figure 3.12: Horizontal and vertical residence time against changing liquid level. τ_{WiO} and τ_{OiW} indicate the residence time for water droplets in oil phase and oil droplets in water phase, respectively. The vertical residence time is shown for the smallest and largest droplet. The volumetric horizontal flow used is through the fifth horizontal control volumes and the vertical velocity is adjusted with the mean WC in the respective phase.

3.1.5.1 Discussion

The water cuts should decrease in the oil and increase in the water phase as the liquid flows through the separator. The steady state water cuts in the solution shown in Table 3.2, have this trend. The water cuts should also increase from the top to the bottom as a result of non-convective flow of dispersed phase. This trend was also found in the solution, except for the control volumes at the oil-water interface. This is because of high interfacial transfer, and is shows that the model parameters should be fitted to real data. The water cuts no not change on the right side of the weir. This is accordance with theory, as there no possible separation after the flow has passed the weir. The partial densities might still change as a result of the coalescence, but the lack of separation means the water cut must be unchanged.

The trends shown in Figure 3.7 and 3.8 are explained by studying the residence time of the droplets. Less residence time in horizontal directions compared to the residence time in vertical direction gives poorer separation. The residence time in horizontal and vertical direction are shown against the changes in the water level and liquid level in Figure 3.11 and 3.12, respectively.

For the chosen model parameters and levels the study show that the residence time for OiW through the convective flow is longer than the residence time in vertical direction. The limiting factor of the separation of oil from water must be the interfacial transfer, as the residence times indicate that the oil should be able to completely separate from the water phase. Interfacial transfer as limiting factor corresponds with the water cuts shown in Table 3.2, where the bottom is pure water. Increased water level or liquid level causes the horizontal residence time of OiW to increase faster than the vertical residence time of OiW, and the separation is improved, as shown in the results.

Increased water level causes the vertical residence time of WiO droplets to decrease faster than the horizontal residence time. This is because the distance the droplets need to travel decreases and more of the droplets reaches the oil-water interface before leaving the separator, and the separation improves. Increased liquid level causes the vertical residence time of WiO droplets to increase faster than the horizontal residence time. The distance, the droplets have to settle, increases and the separation of WiO droplets is poorer.

How the steady state products depends on the disturbances in Figure 3.9 and 3.10 corresponds with how the respond should be. The increased liquid feed flow does not change the feed concentrations, but the increased flow decreases the horizontal residence time in both phases and both separations is poorer. The increased water cut in the feed ϵ_{in} results in more water (determined by the γ_w-fraction) going into the water phase. The increased flow of water phase, decrease the horizontal residence time and the separation of OiW is poorer. Larger flow to water phase in the initial separation causes smaller flow to oil phase and the rate of oil product goes down. Increased water cut in the feed, causes more water to disperse in the oil phase and the WiO concentration of the oil product increases.

The model could be improved by adding better size distribution in the initial separation, such that not only the smallest size is distributed in the feed to the phase. This combined with better reaction rate constants for coalescence, would give a distribution in the separator more similar to real world.

3.1.6 Dynamic Model Analysis

The dynamic model was achieved from minor changes to the steady state model, because it was developed with focus on dynamics. The flow rates of the product was changed and equation (3.21) and (3.22) is in the dynamic model given as

$$F_G = u_1 \tag{3.52}$$

$$F_R = u_2 \tag{3.53}$$

$$F_L = u_3 \tag{3.54}$$

The dynamic model do not constrain the states and the pressure and levels are instead controlled by proportional integral (PI) controllers. The dynamic model do not use CasADi and the model is given as a ordinary MATLAB model/function and is solved and simulated with ODE15s solver in Simulink. The script of the dynamic model is given in Appendix C.6.

The changes in the product flows changes the internal flows and it is important that the dynamic model allows temporary backflow of the dispersed OiW droplets in transient conditions. The backflow of dispersed OiW droplets is described in section 3.1.3.5.

3.1.6.1 Simulation and Control

The developed dynamic model has a control structure with three PI controllers. The pressure is controlled with the flow of gas product, the liquid level is controlled with the flow of oil product and the water level is controlled with the flow of water product. Some of the model parameters was changed during the development of the dynamic model. This is because the parameters is guesses and to achieve result closer to the case used to develop the produced water treatment models. The new parameters are given in Table 3.3, and the rest of the parameters are given in Table 3.1.

Table 3.3: Parameters and setpoints for dynamic GS1 and GS2. The rest of the parameters are shown in Table 3.1.

Parameter	Symbol	Value GS1	Value GS2	Unit
Separator length	L	15	5	m
Rate constant, interfacial flow of water	k_w^{int}	0.4	1	$\mathrm{m\,s^{-1}}$
Fraction of water to water phase	γ_w	0.4	0.15	-
Fraction of oil to oil phase	γ_o	0.99	0.999	-
Pressure setpoint	P_{sp}	84.6	-	bar
Liquid level setpoint	$h_{l,sp}$	3.5	2.2	m
Water level setpoint	$h_{lw,sp}$	1.8	1.8	m

The open loop responses was found by simulating 10% steps in each of the manipulated variables. The first order response in pressure was used to find the parameters describing the response (process gain k, time constant τ_1), and the pressure controller was tuned with Skogestad internal model control (IMC) tuning rules for first order response [12]. The integrating response in the levels was used to find the slope k' and used to tune the level controllers with Skogestad IMC tuning rules for integrating processes [13].

The close loop time constant τ_c was chosen such that the Pressure controller was tuned tighter than the liquid level controller, which again was tuned tighter than the water level. The tuning gave good closed loop responses and the tuning was kept as they were. The tuning parameters for the first stage gravity separator are given in Table 3.4. The controller was implemented with Anti-windup, where Clamping was the chosen method. The Simulink controller block did not

work with Clamping and the controller block was modified to fix the problem. The modified controller block is described in Appendix D.

Table 3.4: Tuning parameters for control of GS1. Time unit is in seconds.

GS1 tuning	Closed loop time, τ_c	Controller gain, K_c	Integral time, τ_I
Pressure, P	5	-0.0341	20
Liquid level, h_l	10	-4.3017	40
Water level, h_{lw}	15	-3.5570	60

The controlled GS1 was studied with steps in the setpoints, and the responses in the separator from setpoint changes in pressure, liquid level and water level are respectively shown in Figure 3.13, 3.14 and 3.15.

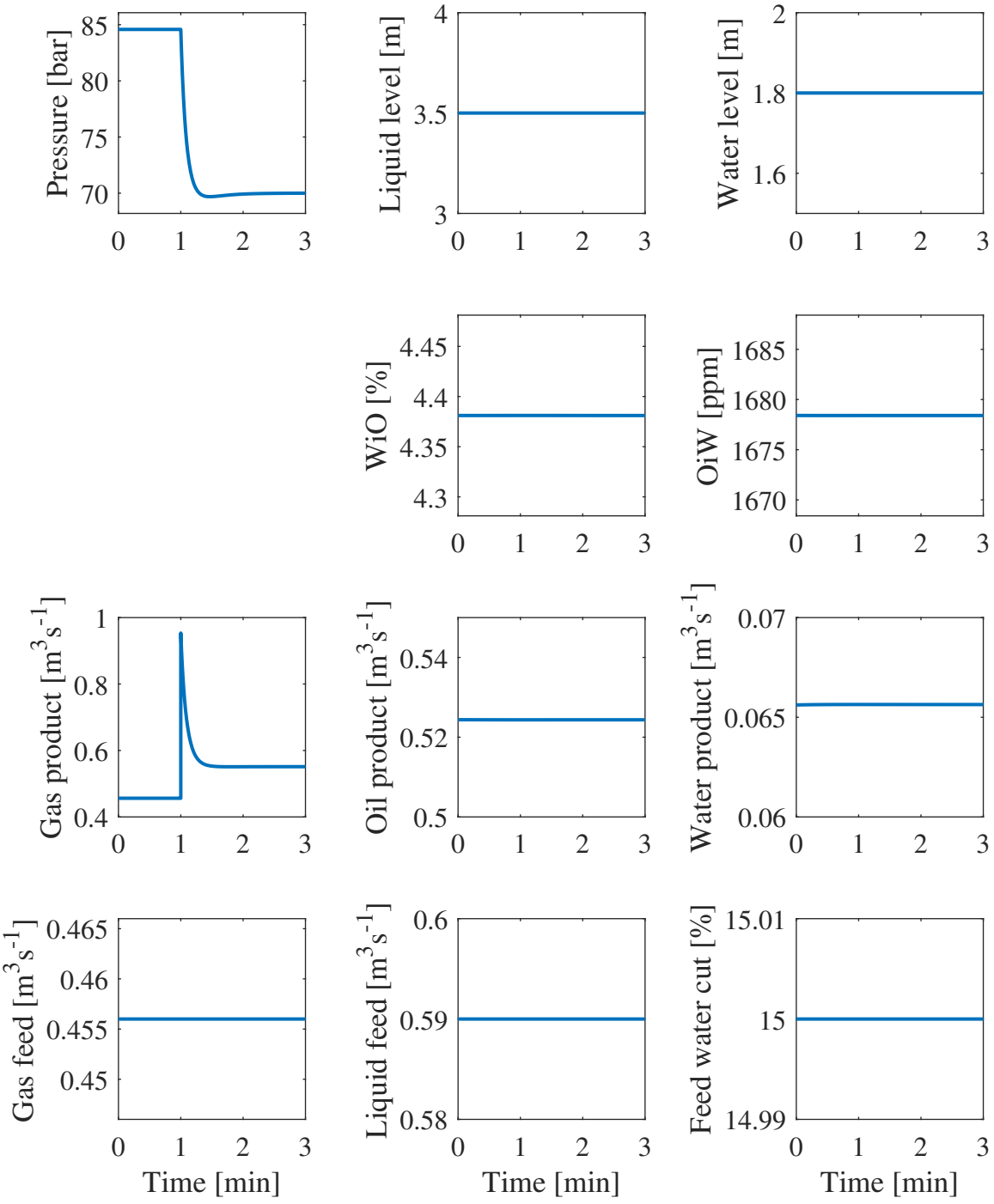


Figure 3.13: Closed loop responses in GS1 with change in pressure setpoint. The three controlled variables are shown in the first row. The purity of the products are shown on the second row. The manipulated variables are shown on the third row. The bottom row shows the disturbance in the feed. It is shown that changes in the pressure does not interact with the liquid separation or the levels.

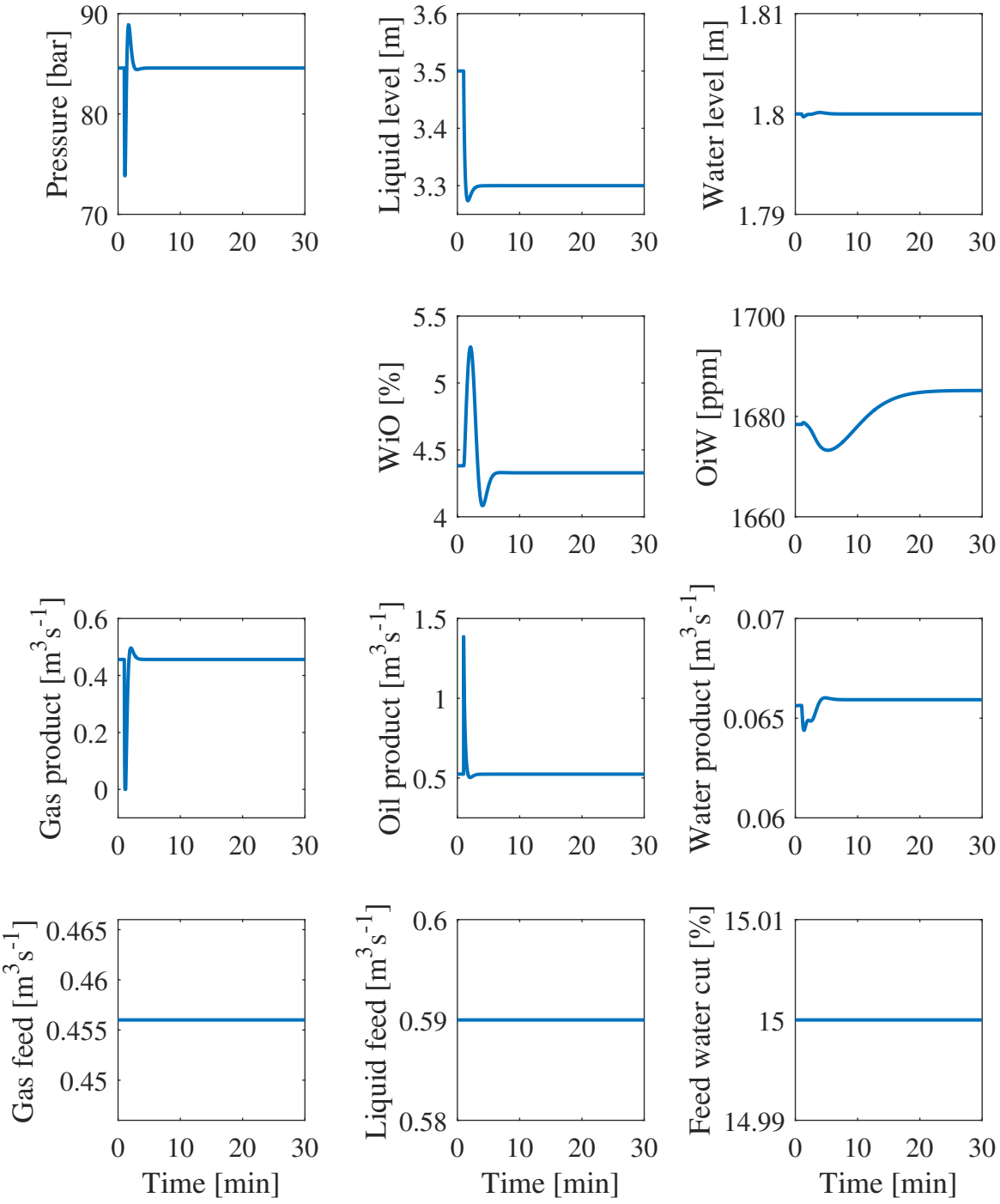


Figure 3.14: Closed loop responses in GS1 with change in liquid level setpoint. The three controlled variables are shown in the first row. The product purity are shown on second row. The manipulated variables are shown on the third row. The bottom row shows the disturbance in the feed. It is shown that changes in the liquid level mainly interact with the pressure and the WiO concentration. Small interactions with water level and OiW concentration are shown.

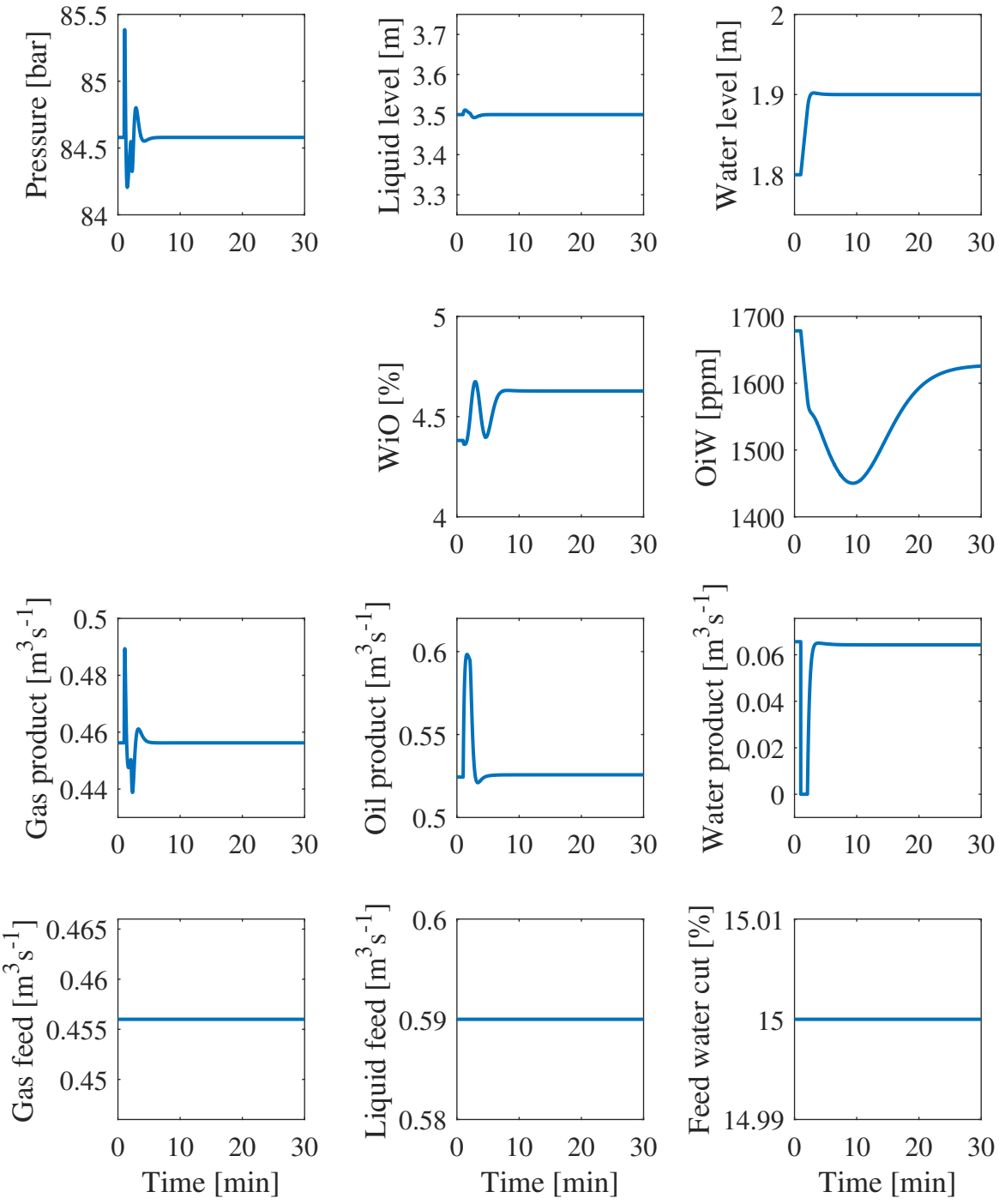


Figure 3.15: Closed loop responses in GS1 with change in water level setpoint. The three controlled variables are shown in the first row. The product purity are shown on second row. The manipulated variables are shown on the third row. The bottom row shows the disturbance in the feed. It is shown that changes in the water level interacts with all other variables. Liquid level and pressure is controlled but WiO and OiW have large changes.

3.1.6.2 Discussion

The simulation shows acceptable control on the three phase gravity separator. The closed loop constants was chosen such that they increased from pressure to liquid level and to water level. This was because the pressure depends on the liquid level and the liquid level again depends on the water level. The tuning shows good results and was unchanged.

The pressure do not show any response in the other process variables, because flow control was assumed perfect. The pressure would in reality interact with the product flows. Adding a pressure dependent valve equation would give better simulations. The step changes in the setpoint is also not a very realistic approach to desired changes in the separator. Adding dispersed gas to the liquids would also give pressure dependent separation of oil and water, as it would interact with the viscosity. The responses in the concentrations are a result of the changes in the flows, which again changes the residence times as discussed in section 3.1.5.1. Changes to the water level also changes the interfacial area and the interfacial transfer between oil and water phase.

The allowed backflow of dispersed droplets can also be used to describe the large change in OiW when the water product flow is temporary decreased in Figure 3.15. The same responses between in OiW, water level and water product flow are shown in Figure 3.16. The figure shows that when the water product flow is temporary turned off, during changes in water level, the OiW decreases and overshoot its new steady state value. When the water flow is turned of the internal water phase flow is stopped and the interfacial flow of water allows backflow. The low internal water phase flow also increases the separation because the horizontal residence time increases and the WC at then interface increases, resulting in a higher interfacial transfer. It is shown in Figure 3.11 and 3.12, that the interfacial transfer is the limiting factor, for OiW separation.

The three phase gravity separator does not have any degrees of freedom left as it is controlling the pressure and levels. Since OiW concentration depends on the water level and the OiW dynamics is slower than the water level, it would in theory be possible to use the water level controller as a slave in a OiW cascade controller. Figure 3.16 shows the interactions between the setpoint change in the slave (water level) and the OiW. The response in the figure shows an overshoot of over 300% and it is therefore not possible to easily tune a cascade controller. The steady state solutions in section 3.1.5 shows that a OiW controller is in fact not needed, as the optimal water concentration is as close to the weir as possible. Optimizing the GS1 together

with other separators might give other operational points not at the weir height and GS1 would then be controlled to the optimal operational point and is a OiW setpoint.

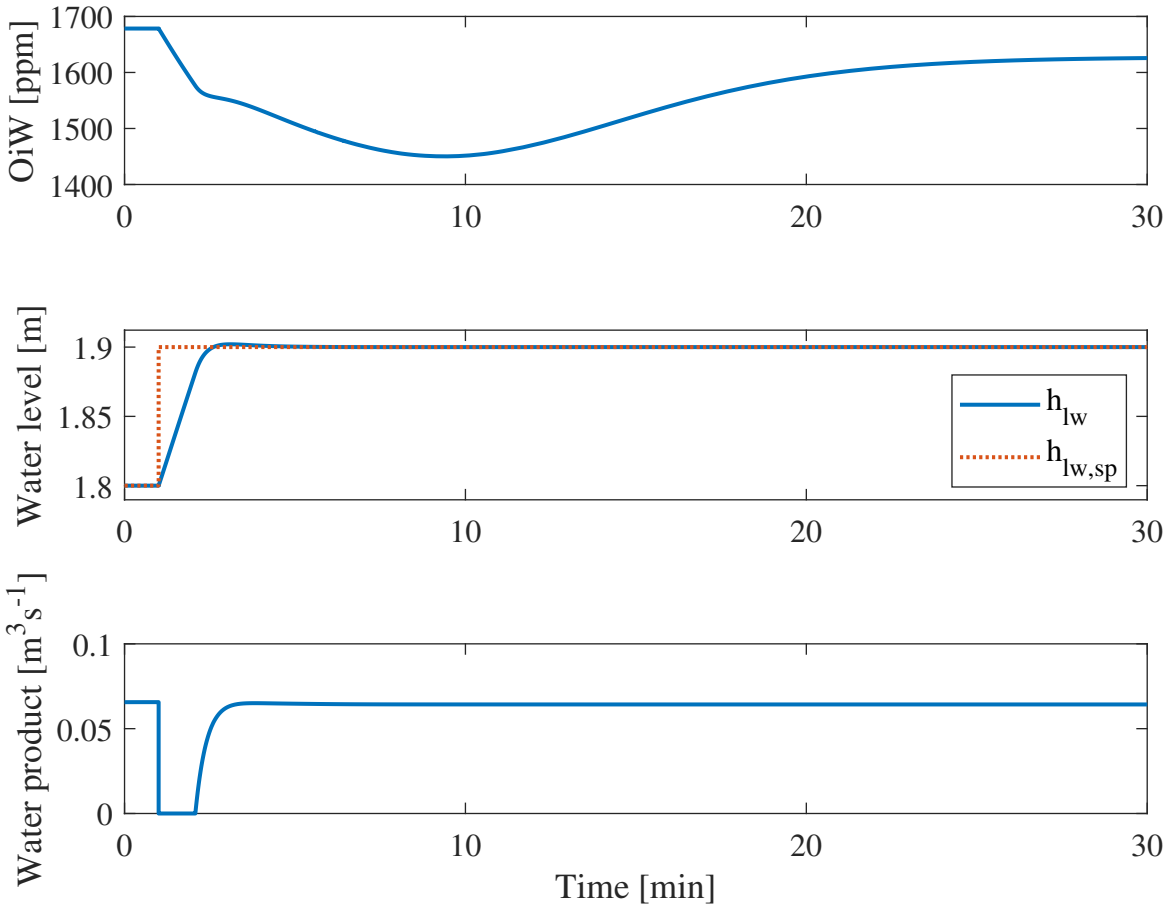


Figure 3.16: Change in OiW from step in water level setpoint. The given step in water level setpoint result in an overshoot larger than 300% in OiW. This is a result of the reduced internal water phase flow and increased horizontal residence time. The stop in water product flow rate also allows temporary backflow of the dispersed oil droplets.

The closed loop responses in GS1 from disturbances are not included in this dynamic analysis as they are shown in the simulation of the subsea system in Chapter 5.

3.2 Second Stage Gravity Separator

The second stage gravity separator was implemented the same way as the first stage gravity separator. The main difference in GS2 compared to GS1 is that GS2 utilizes the known droplet distribution of the GS1 oil product, and uses it in the feed to the oil phase. The second stage

gravity separator does not calculate any gas pressure.

3.2.1 Steady State Model

$\gamma_{o,2}$ and $\gamma_{w,2}$ is used to determine the initial separation at the inlet, just as in GS1. The fraction $\gamma_{w,2}$ is used to determine the flow of water to water phase and it takes a portion out of each volume of the size classes y. The rest of the volume of water is kept dispersed in the oil phase. The volumetric water feed flow, $F_{w,y}$, of each size class y is given by

$$F_{w,y} = \frac{\bar{\rho}_{w,y}}{\rho_w} F_{in}^l \tag{3.55}$$

where $\bar{\rho}_{w,y}$ is the partial density of the size class in the feed and ρ_w is the density of pure water. F_{in}^l is the liquid feed flow. The total feed of oil, F_o, to GS2 is given by

$$F_o = F_{in}^l - \sum_y F_{w,y}. \tag{3.56}$$

The liquid feed to oil phase, F_{op}, and water phase, F_{wp}, after the initial separation is respectively given by

$$F^{op} = \gamma_{o,2} F_o + \sum_y (1 - \gamma_{w,2}) F_{w,y} \tag{3.57}$$

$$F^{wp} = (1 - \gamma_{0,2}) F_o + \sum_y \gamma_{w,2} F_{w,y} \tag{3.58}$$

and the partial feed densities is

$$\bar{\rho}_{WiO,y} = \frac{F_{w,y}\rho_w}{F^{op}} \tag{3.59}$$

$$\bar{\rho}_{OiW,y=1} = \frac{(1 - \gamma_{o,2}) F_o \rho_o}{F^{wp}} \tag{3.60}$$

The partial feed densities of WiO, $\bar{\rho}_{WiO,y}$, have densities in all classes, while the partial feed density of OiW, $\bar{\rho}_{OiW,y=1}$, is just added as the smallest size. The changes in the steady state second stage separator model are given in Appendix C.7.

It was not performed any studies on the steady state solutions of the second stage gravity separator model. The model is implemented and solves as GS1. GS2 is solved without pressure and with the solved steady state product from GS1 as feed. The parameters used in the model is given in Table 3.1 and 3.3.

3.2.2 Dynamic Model Analysis

The dynamic model of the second stage separator was developed the same way as the dynamic first stage separator and with the changes in the second stage separator given in last section. Since the model parameters, size parameters and feed composition and the flow are different in the second stage separator is not equal the first separator and the controller is tuned separately. The method used to tune the dynamic GS2 model was the same as for GS1. The open loop responses was found by performing -10% step in the flow of oil product and +10% step in the flow of water product. The tuning parameters of GS2 are given in Table 3.5.

Table 3.5: Tuning parameters for control of GS2. Time unit is in seconds.

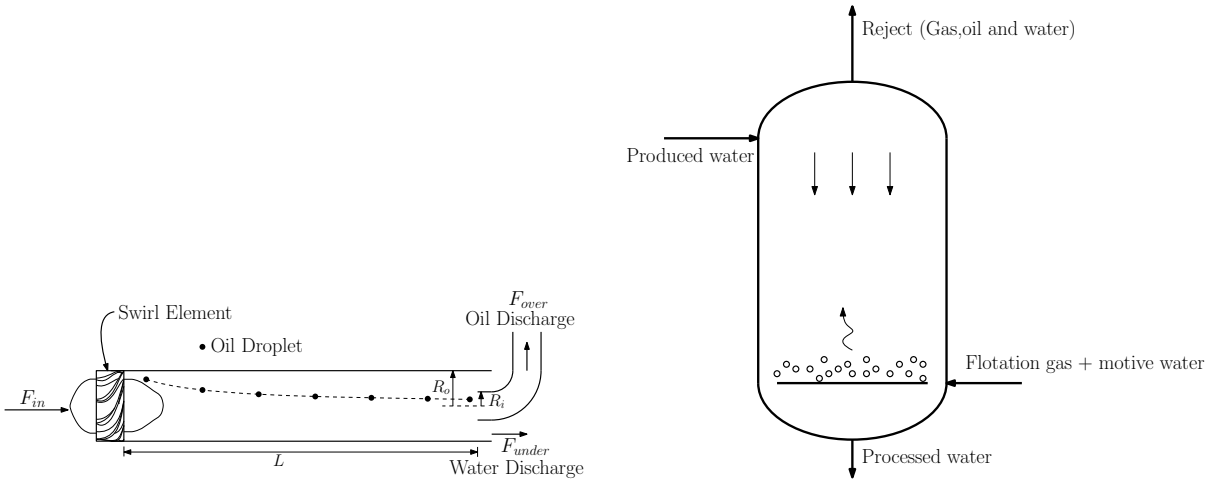
GS2 tuning	Closed loop time, τ_c	Controller gain, K_c	Integral time, τ_I
Liquid level, h_l	10	-1.9472	40
Water level, h_{lw}	15	-1.2347	60

The dynamic behavior of GS2 was similar to GS1. The closed loop responses in the model, from changes in the setpoint are not included in this work. The closed loop responses in GS2, as a result of disturbances in the feed to GS1 are shown in the simulation of the subsea system in Chapter 5.

Chapter 4

Modelling of Produced Water Treatment

The model of the produced water treatment uses existing models of hydrocyclones and compact flotation units. Basic theory on the separators and the models is given in this chapter, but for details on how the models was developed it is refereed to the respective papers [5, 6]. Illustrations of the two separators are given in Figure 4.1.



(a) Inline deoiling hydrocyclone, HC

(b) Compact Flotation Unit, CFU

Figure 4.1: Illustration of the modelled separators in the produced water treatment [5, 6].

4.1 Inline Deoiling Hydrocyclone

The inline deoiling hydrocycones is mechanical-physical separator where centrifugal settling and sedimentation is used. The separator increases the gravitational force by forcing the flow of water through the swirl element shown in Figure 4.1a. The gained swirl in the separator, forces the heavy water phase to the outer parts of the separator cylinder as liquid flows through the

separator. The dispersed oil droplets of lighter phase, will be forced to the inner parts of the cylinder from buoyancy forces. The oil rich flow in the center of the separator goes out in the oil discharge flow, called overflow and the rest of the liquid, in the outer part of the cylinder, flows out in the flow called underflow.

The inline deoiling hydrocyclone is a simpler model than the GS1, GS2 and CFU, as it has only the overflow as manipulated variable and only OiW concentration as controlled variable. The steady state solution of the HC model was solved with the produced water product from the bulk separation as feed. The OiW setpoint in the underflow was chosen to 120 ppm, which is a concentration the CFU should be able to reduce to the restriction of 30 ppm. The tuning parameters used to control the HC are given in Table 4.1. The open loop response in HC, was found by performing a 10% step in the overflow. The response was roughly estimated to a first order response and tuned with the Skogestad IMC tuning rules [12]. The controller was first tuned smoothly with τ_c = 60 s, and then changed to a more tightly tuned setting, with 4 times larger K_c and an integral time only a fourth of the first tuned parameter.

Table 4.1: Tuning parameters for control of HC. Time unit is in seconds.

HC tuning	Controller gain, K_c	Integral time, τ_I
OiW control	$-1.78 \times 4 \times 10^{-6}$	13.8/4

4.2 Compact Flotation Unit

Compact flotation units mixes a gas, called flotation gas, into the water of dispersed oil and uses agglomeration effects between the oil and gas droplets to separate the phases. The agglomeration effect causes flotation of the oil together with the gas, and the oil can be separated out together with the gas in the reject flow at the top of the separator. The purified water, flows out in the bottom of the separator. An illustration of the CFU is shown in Figure 4.1b.

The CFU is a more difficult system, as it have several manipulated and controlled variables. The manipulated variables of the CFU model are the two outflows and the feed flow of flotation gas. The reject flow is used to control the pressure, the flotation gas is used to control the OiW concentration in the cleaned water and the product flow rate of cleaned water is used to control

the liquid hold up inside the separator. The control pairs and their open loop responses are shown in Table 4.2.

Table 4.2: Open loop responses in CFU.

Controlled variable	Manipulated variable	Open loop response
Pressure	Reject flow	$\frac{-0.303}{845s+1}$
OiW	Flotation gas flow	$\frac{20.2}{400s+1}e^{-30s}$
Liquid hold up	Water flow	$\frac{-0.0167}{271s+1}e^{-2s}$

The controllers was tuned using the transfer functions of the responses in Table 4.2 and Skogestad IMC tuning rules [12]. The used closed loop constant τ_c is $24\,\mathrm{s}$. The tuning parameters are given in Table 4.3. The setpoints was found from optimizing the CFU model with the feed concentration. The setpoint of the pressure was 1.4 bar, the setpoint of the OiW was 25 ppm and the setpoint of the liquid hold up was 0.9.

Table 4.3: Tuning parameters for control of CFU. Time unit is in seconds and close loop time constant, τ_c, is equal $24\,\mathrm{s}$.

CFU tuning	Controller gain, K_c	Integral time, τ_I
Pressure control	-111.5	96
OiW control	-0.8251	96
Liquid hold up control	-676.14	96

Chapter 5

Subsea Separation System: Bulk and Produced Water Treatment

The developed models and controllers in the bulk separation was combined with the models of the hydrocyclones and compact flotation units in the produced water treatment. The combined separation system was then simulated for disturbances in the feed to the first stage gravity separator.

5.1 Results

The possible disturbances in the feed to GS1, is the flow of gas, the flow of liquid and the water cut in the liquid feed. The simulation performed simulated 120 minutes where the disturbances happens at $15\,\mathrm{min}$, $30\,\mathrm{min}$ and $75\,\mathrm{min}$ respectively. The disturbances are a 10% increase of the nominal values. The responses in the first stage gravity separator (GS1) and the second stage separator (GS2) are shown in Figure 5.1 and 5.2, respectively. The figures shows the controlled variables, manipulated variables and the disturbances in the feed to GS1. The figures also shows the concentrations of the dispersed phases in liquid products. The disturbance in the gas flow to GS1, is for simplicity not shown in Figure 5.2, because it does not interact with GS2 and any of the other liquid downstream processes. The responses in the HC and CFU models are shown in Figure 5.3 and 5.4, respectively.

The tuning parameters used in GS1 and GS2 is described in Chapter 3 and the tuning parameters used in the produced water treatment models is given in Chapter 4. The feed flow and OiW concentration to the produced water treatment is shown in Figure 5.5. A closer look on the CFU response, between $70\,\mathrm{min}$ and $90\,\mathrm{min}$, is shown in Figure 5.6.

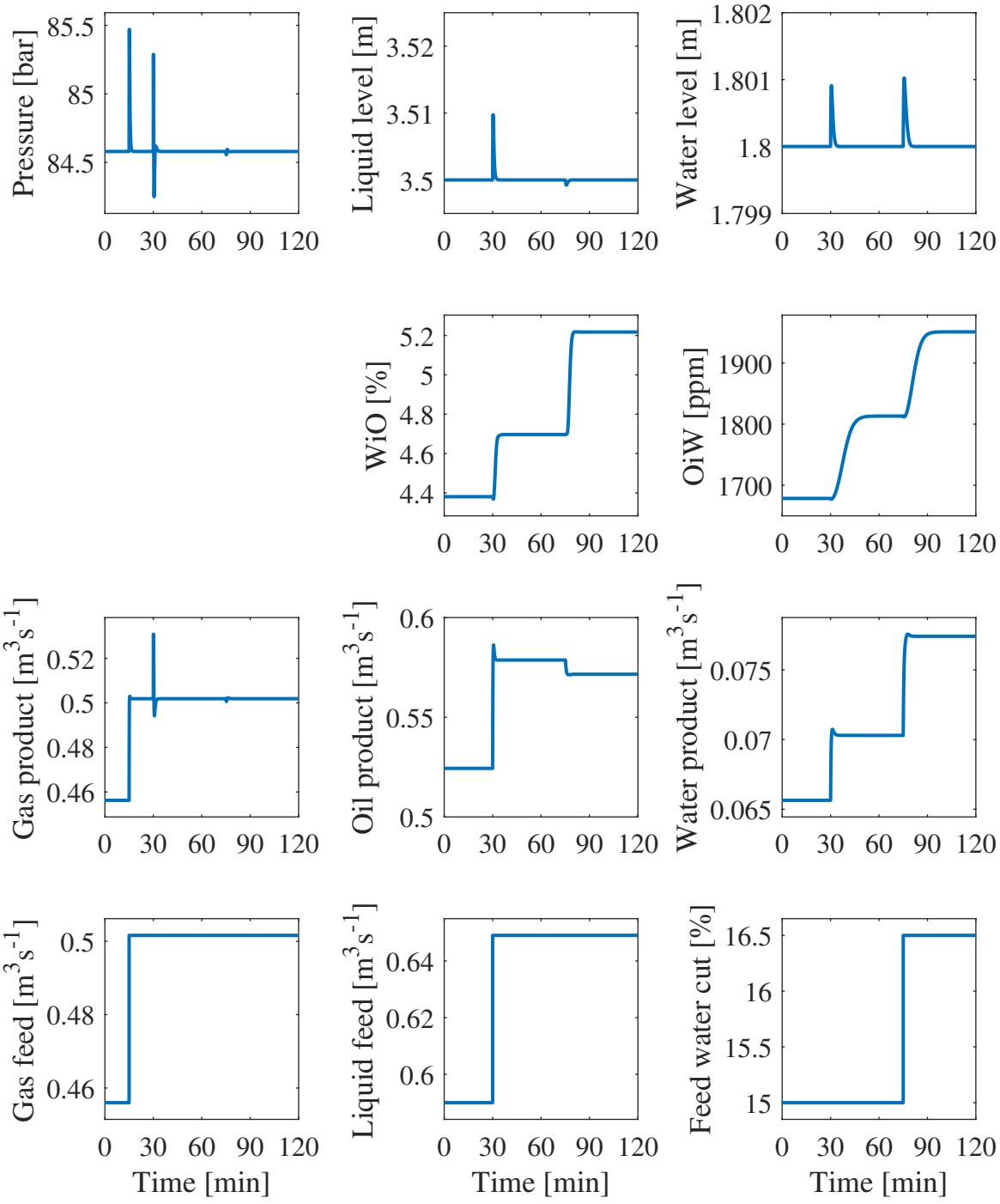


Figure 5.1: Simulation of first stage gravity separator. The pressure, liquid level and water level are controlled by the gas product, oil product and the water product, respectively. WiO and OiW are the concentrations of the dispersed phases in the products and the feed with disturbances is shown on the fourth row. The simulation show good control of the pressure and levels for the given disturbances.

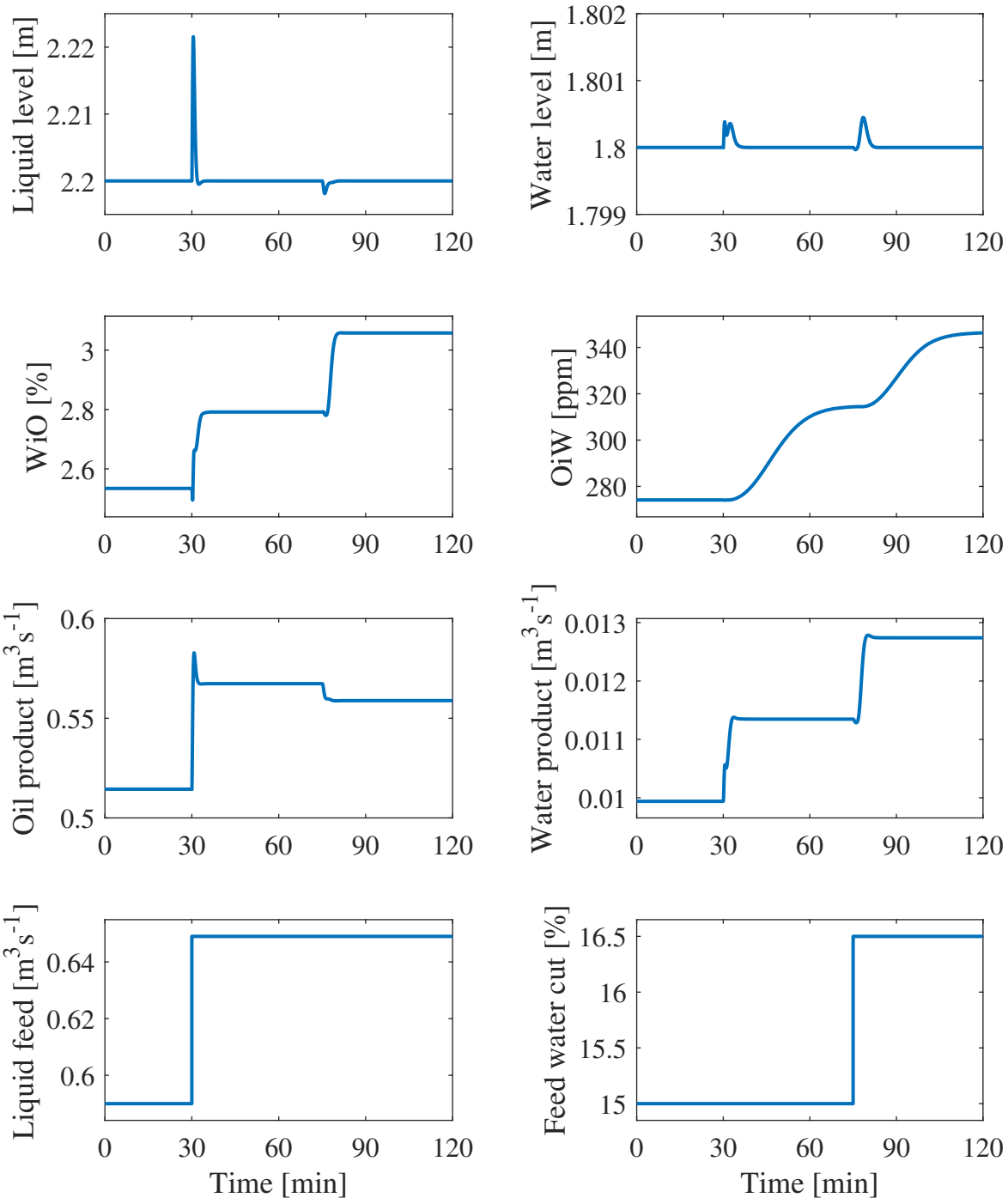


Figure 5.2: Simulation of second stage gravity separator. The liquid level and water level are controlled by the oil product and water product, respectively. WiO and OiW are the concentrations of the dispersed phases in the products and the feed with disturbances to GS1 is shown on the fourth row. The gas flow to GS1 is not shown, as it does not interact with GS2. The simulation show good control of levels for the given disturbances. The spike in liquid level is twice the size as in GS1, while the spike in water level is smaller than in GS1.

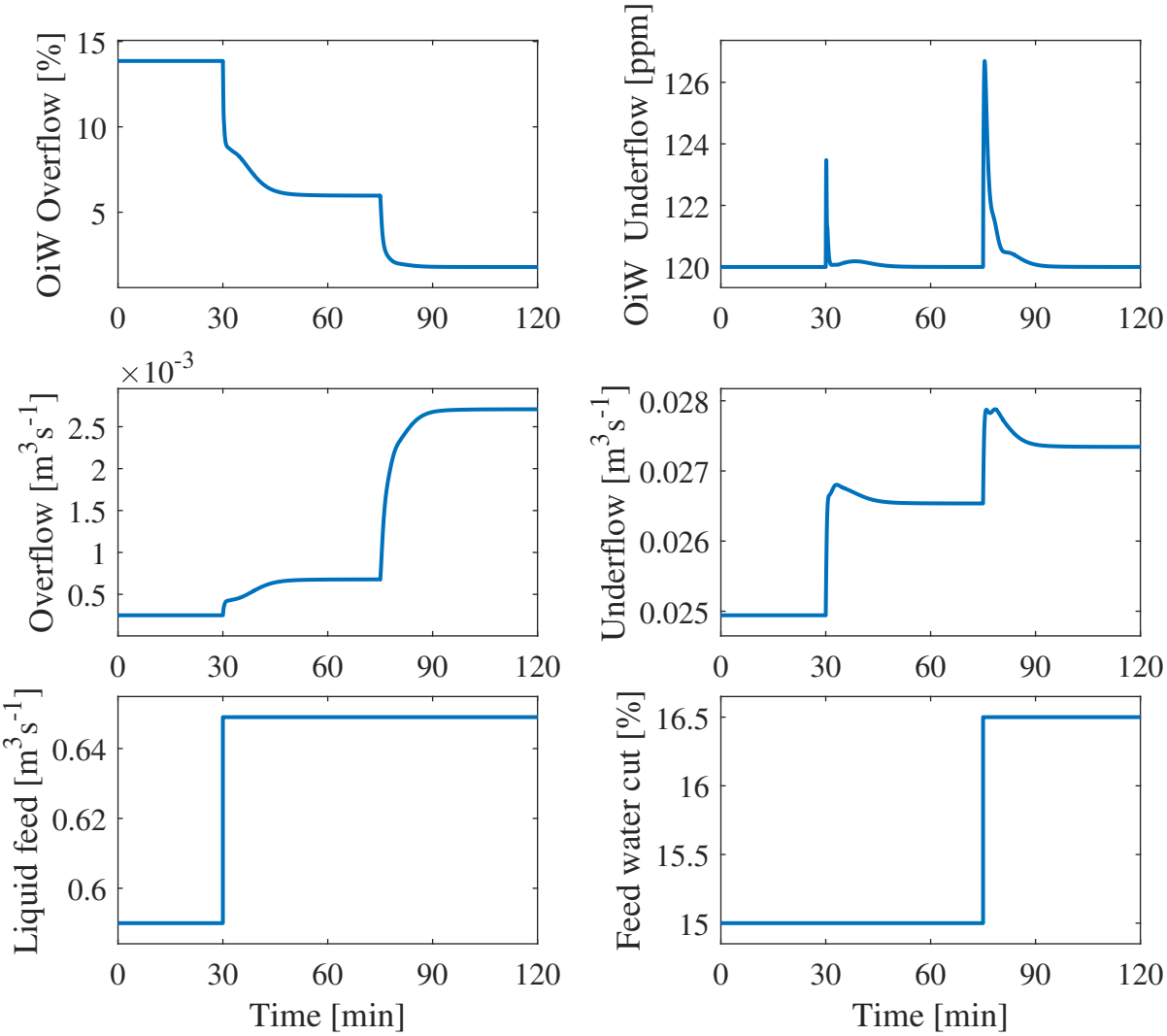


Figure 5.3: Simulation of inline deoiling hydrocyclones. The OiW in the underflow (water product) is controlled by the rate of overflow (oil rich product). The rate and concentration of dispersed phase is shown in both product flows. The liquid feed and feed water cut shown on the forth row, is the feed to feed GS1.

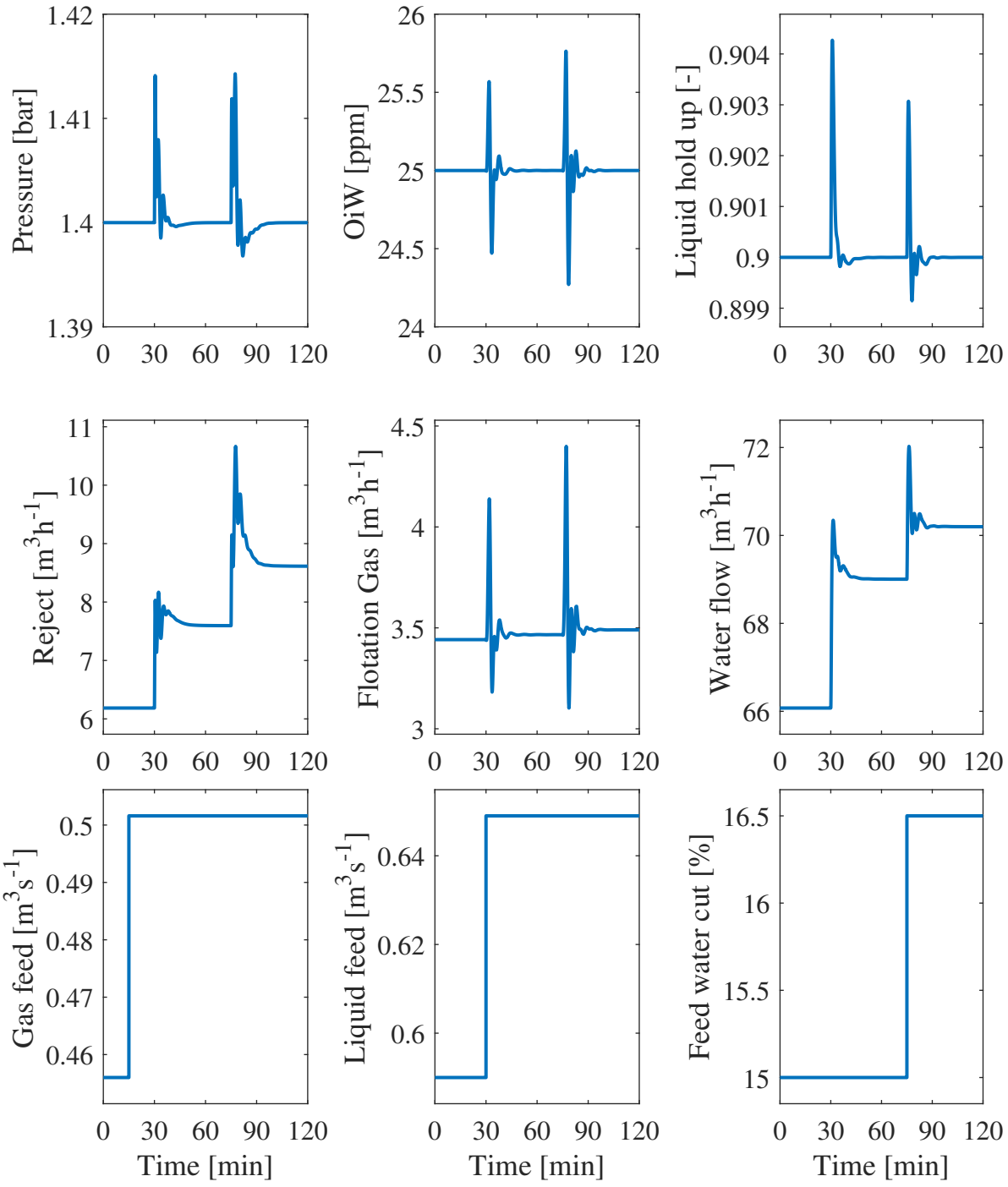


Figure 5.4: Simulation of compact flotation units. The pressure, OiW and liquid hold up are controlled by the rate of the reject flow, flotation gas flow and the water product flow, respectively. The flow rates and water cut on the forth row shows the feed to GS1. The simulation shows oscillation in the CFU with small deviations from the desired setpoint value.

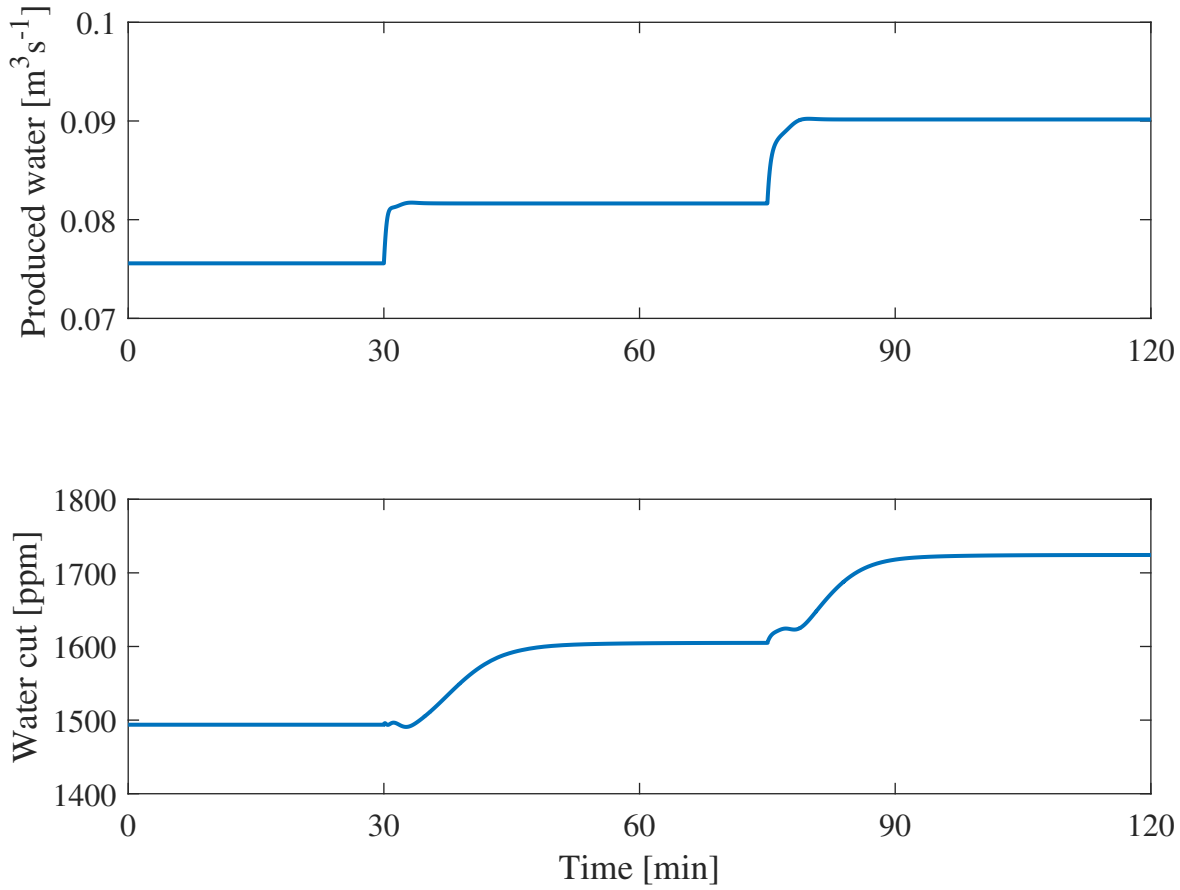


Figure 5.5: Flow rate and water cut in produced water from bulk separation.

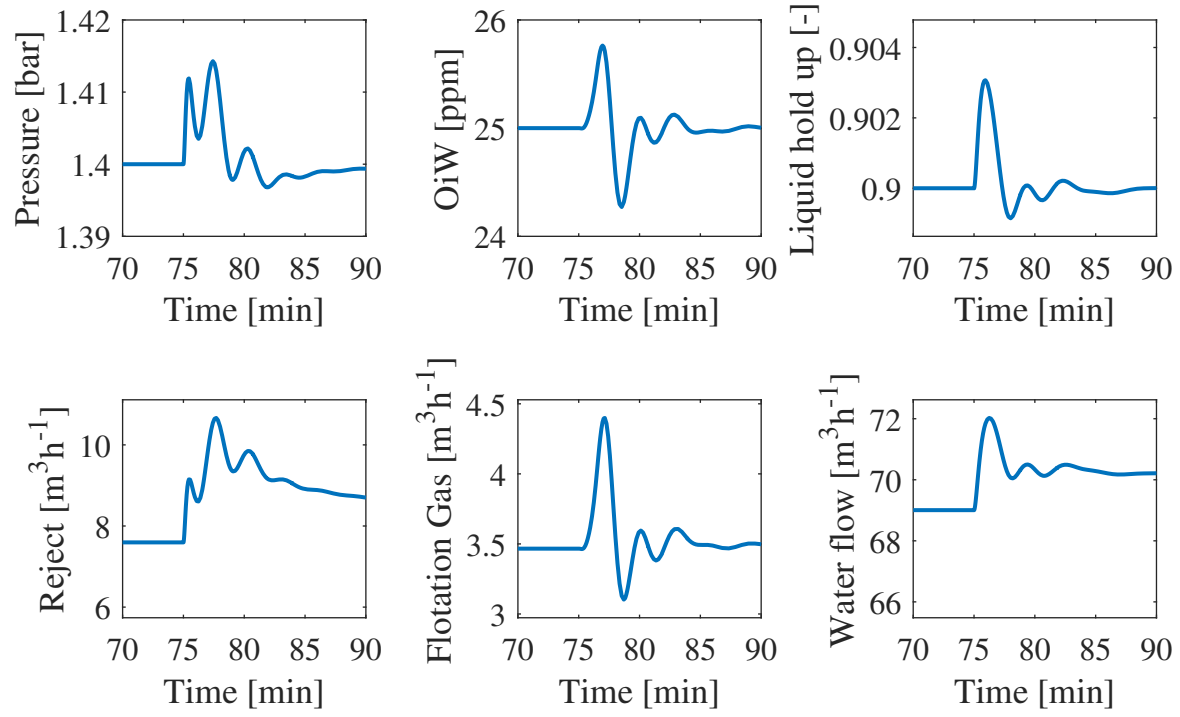


Figure 5.6: Closer look at CFU response between 70 min and 90 min. The oscillations have reasonable amplitudes and periods.

5.2 Discussion

The response in GS1 is very fast and the disturbance in the feed flow is sent to the downstream separators. The pressure depends heavily on the transient response in liquid level and a smoother control of the levels will need a tight control of the pressure. In this model the interaction goes only from the liquid level to the pressure and the pressure does not interact with the liquids. A better model of the valves such as a pressure dependent valve equation would give a more realistic simulation.

The differences in the spikes in the levels in GS1 and GS2, are because the two separators are different. The fractions γ_o and γ_w are also chosen to a different value than in the GS1 model. The γ fraction was in this work, guessed to give believable initial separation. The parameters should be compared and fitted to a real system and its measured data.

The response in HC shows that the OiW will react to the disturbances, but that the OiW is relatively, quickly controlled back to the setpoint. The feed to the produced water treatment, shows that the change in flow rate is much faster than the change in the OiW concentration in the feed. The spike in the controlled OiW concentration, observed in the simulation, must be from the increased flow and not from the increased OiW. Smoother control of GS1 and GS2 would allow the separators to work as buffer tanks and therefore relax the changes in the flow rates. The slower changes in the feed rate to the PWT, would then improve the control of the PWT.

Implementing a feed forward controller to the HC would also improve the control of the PWT. The OiW controller, in the existing HC, is a feedback composition controller and such controller are normally slower than flow controllers. Since the flow rates of the produces water in the bulk separation should be known, or at least estimated in the control of GS1 and GS2, the implementation of a feed forward controller using the feed flow, should allow the HC to react earlier to the increased feed disturbance.

The closer look at the simulation of the CFU, in Figure 5.6, shows that the oscillation in the CFU is within reasonable periods and amplitudes. The oscillations is stabilized relatively fast and the deviations after $10\,\mathrm{min}$ is within deviations from the setpoint.

The OiW concentration in the purified water show small deviation from the setpoint, where the largest deviations are less than $1\,\mathrm{ppm}$. In this case the water treatment would benefit from shifting the setpoint closer to the restriction at $30\,\mathrm{ppm}$, and save flotation gas. It should not be a problem moving the setpoint to $30\,\mathrm{ppm}$, as the restriction is monthly average and temporary overshoots will not damage equipment.

The chosen operational point in the simulation and developed models was chosen, and the system would benefit from an optimization of the separation system. This would give the optimal levels in the bulk separation and the optimal OiW setpoint for HC. The optimization of the separation system would not change the OiW setpoint in the CFU, as it will be an active constraint kept at $30\,\mathrm{ppm}$ (or as close to the $30\,\mathrm{ppm}$ as possible). The separation system is optimized by minimizing the flow of water to topside processing. This is the same as maximizing the flow rate of purified produced water.

The GS1 and GS2 level setpoints was in this case, chosen to minimize the OiW in GS1 and minimize the WiO in GS2. This will not be the case in optimization of the bulk separation together with the PWT. Minimizing OiW in GS1 gives level setpoint as high as possible, which will give higher WiO. The increased WiO will result in higher flow of water sent to the topside processing.

Chapter 6

Conclusion

This thesis has been studying the system of bulk separators and compact purifiers used in a subsea separation system. The bulk separation consist of two gravity separators and the produced water treatment consist of inline deoiling hydrocyclones and compact flotation units.

6.1 Bulk separation

It was developed a dynamic coalescence based gravity separator, with droplets with diameters between $150\,\mu\mathrm{m}$ and $300\,\mu\mathrm{m}$. The hindering of the droplets and the change in viscosity from increased water cut, was taken into consideration in the adjusted Stokes' velocity used as non-convective velocity in horizontal direction. The bulk separation consist of a three phase gravity separator and a two phase gravity separator. The model rely on parameters describing the initial separation of the phases, the interfacial transfer and the rate of the coalescence. These parameters were not fitted to any preexisting data and the chosen parameters do not necessary reflect any real separators. The dispersed droplets were set to the smallest size in the initial separation in the separators, except for the oil continuous phase in the second stage gravity separator, where the product densities from the upstream gravity separator was known.

It was shown in the dynamic analysis that the pressure in transient conditions, depends heavily on the changes in the liquid level. Since pressure dynamic valves and dispersed gas in the liquid was not modelled the gas flow and pressure in the first stage gravity separator does not interact with the separation and the downstream separators.

6.2 Subsea Separation System

The simulation of the combined bulk separation and produced water treatment, shows the flow disturbance in the feed to the first stage gravity separator is sent to the downstream separators and that a smoother tuning on the gravity separator would be beneficial for the system.

The produced water treatment was able to control the OiW concentration with the 10% increases in the feed to the first stage gravity separator. The largest deviation from the OiW setpoint was less than $1\,\mathrm{ppm}$ and it would in this case be possible to shift the setpoint closer to the purity restriction.

The hydrocyclones depends heavily on the flow of produced water, and the tight responses in the gravity separators can be seen in the control of the hydrocyclones. The PWT could benefit from a feed forward control of the hydrocyclones. The flow rate of the produced water should be known from the control of the bulk separators and the overflow can be increased earlier.

6.3 Further Work

The gravity separator model is not validated with any experimental data, and the model should be comparing with real separators and the model parameters fitted to data from real separation. Extending the model of the gravity separator to also consider dispersed gas and pressure dependent valves, would improve the model. The dispersed gas would interact with the separation and influence the downstream processes. The second stage gravity separator can then be modelled as a three phase gravity separator.

The subsea separation system can be optimized such that the setpoints for the levels in the gravity separators and the setpoint for the OiW in the underflow of the hydrocyclones, is optimized. The controller tuning in the gravity separators should be reevaluated and tested with smoother tuning, allowing it to work as a buffer tank to the downstream processes.

Bibliography

[1] O.T. McClimans and R. Fantoft. “Status and New Developments in Subsea Processing”. In: *Offshore Technology Conference*. May 2006. URL: https://www.onepetro.org/conference-paper/OTC-17984-MS.

[2] Nesse S., Garpestad E., and Dragsund E. “Produced Water Management Under the Norwegian Zero Harmful Discharge Regime - Benefits With the Risk Based Approach”. In: *SPE International Conference and Exhibition on Health, Safety, Security, Environment, and Social Responsibility*. Society of Petroleum Engineers, Apr. 2016. URL: https://www.onepetro.org/conference-paper/SPE-179326-MS.

[3] SPE International. *Subsea separation*. PetroWiki. June 2015. URL: http://petrowiki.org/Subsea_separation.

[4] Ahmadun Fakhru’l-Razi et al. “Review of technologies for oil and gas produced water treatment”. eng. In: *Journal of Hazardous Materials* 170.2 (2009), pp. 530–551.

[5] Tamal Das and Johannes Jäschke. “Modeling and control of an inline deoiling hydrocyclone”. In: *accepted for publication in IFAC-PapersOnLine* (2018).

[6] Tamal Das and Johannes Jäschke. “Modeling and control of a compact flotation unit”. In: *To be submitted to Industrial & Engineering Chemistry Research* (2018).

[7] Preben Fürst Tyvold. “Modeling and Optimization of a Subsea Oil-Water Separation System”. MA thesis. Trondheim: Norwegian University of Science and Technology, June 2015. URL: https://brage.bibsys.no/xmlui/handle/11250/2351693.

[8] C.J. Backi and S. Skogestad. “A simple dynamic gravity separator model for separation efficiency evaluation incorporating level and pressure control”. In: Institute of Electrical and Electronics Engineers Inc., June 2017, pp. 2823–2828.

[9] Christie J Geankoplis. *Transport processes and separation process principles (includes unit operations)*. eng. Upper Saddle River, N.J, 2003.

[10] Hugo A Jakobsen. *Chemical Reactor Modeling: Multiphase Reactive Flows*. Cham: Springer International Publishing: Cham, 2014.

[11] S. Arirachakaran et al. “An Analysis of Oil/Water Flow Phenomena in Horizontal Pipes”. In: *SPE Production Operations Symposium, Oklahoma City, Oklahoma, March 13-14*. SPE Society of Petroleum Engineers, 1989. URL: https://www.onepetro.org/conference-paper/SPE-18836-MS.

[12] Sigurd Skogestad. “Simple analytic rules for model reduction and PID controller tuning”. eng. In: *Journal of Process Control* 13.4 (2003), pp. 291–309.

[13] Sigurd Skogestad and Chriss Grimholt. “The SIMC Method for Smooth PID Controller Tuning”. In: *PID Control in the Third Millennium: Lessons Learned and New Approaches*. Ed. by Ramon Vilanova and Antonio Visioli. London: Springer London, 2012, pp. 147–175. URL: https://doi.org/10.1007/978-1-4471-2425-2_5.

[14] Mathworks Documentation. *Anti-Windup Control Using a PID Controller*. [Accessed: 2018-06-07]. URL: https://se.mathworks.com/help/simulink/examples/anti-windup-control-using-a-pid-controller.html.

Appendix A

Adjusted Stokes Terminal Velocity

Stokes' terminal velocity (2.1) can be used to calculate the non-convective vertical velocity of the droplets [9]. Stokes' terminal velocity gives the free settling velocity and it depends in the viscosity of the continuous phase. The settling of dispersed water droplets in oil is to be modelled in a closed tank. If just the Stokes' terminal velocity is used to model the settling of the water droplets and there is no continuous phase of water modelled, the droplets will settle and accumulate in the bottom of the reactor and the water cut will get higher than 1. The partial density of water in the oil phase will be higher than clean water.

We introduce an adjusted Stokes' velocity where the velocity is scaled with water cut. Using the adjusted Stokes' velocity, consider that the increased amount of water around the water droplet will hinder the water droplet from settle. The first proposed adjusted Stokes' velocity, v^a is given as

$$v^a = \begin{cases} v^s(1 - WC^{\text{receiver}})^n & \text{for Water in Oil} \\ v^s(1 - OC^{\text{receiver}})^n = v^s(WC^{\text{receiver}})^n & \text{for Oil in Water} \end{cases} \tag{A.1}$$

where v^s is the Stokes' terminal velocity and both OiW and WiO dispersion is considered. n is a parameter which can be used change how much the water cut hinders the settling. WC^{receiver} and OC^{receiver} is the water cut and oil cut in the control volume the droplet is about to enter. If the water cut in the control volume under the dispersed water droplet is high the water droplet has difficulties moving into the control volume. If the water cut is as high as 1, the water droplet is not able to enter the control volume. The same applies for oil in water droplets.

The viscosity in a oil-water mixture depends on the water cut [11]. The water cut at the phase inversion point is around 0.7. Since viscosity depends on the water cut, the Stokes' terminal velocity will also change with the increased water cut. Using the adjusted Stokes' terminal

velocity allows us to keep using the viscosity of the clean oil phase, since the changes is already taken into consideration.

Figure A.1 shows how the n-parameter be used to change how the Stokes' velocity should be adjusted. Lower n-parameter let the droplet travel easier through low water cuts than the same water cut with larger n-parameter. The n-parameter does not change that the Stokes' velocity is zero for $WC = 1$. n-parameters different from 1 will result in a more difficult problem to be solved and the time needed to solve the steady state might increase.

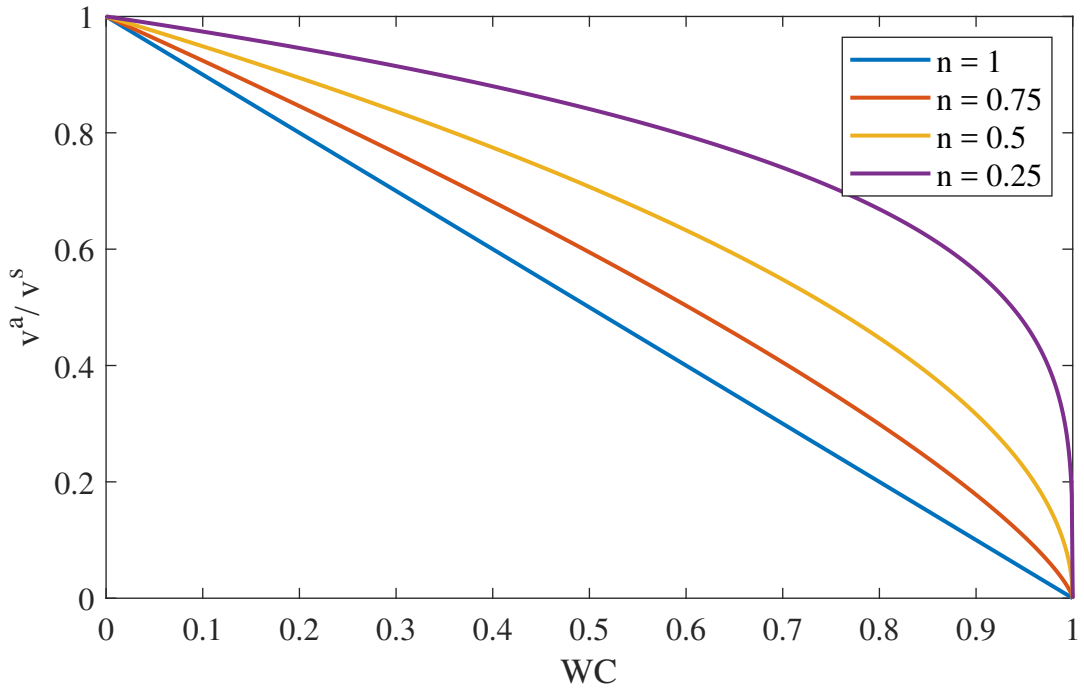


Figure A.1: Illustration of how the n-parameter adjusts the stokes velocity.

Modelling the oil-water separation with modelled interfacial flow between the two phases, and the adjusted Stokes' velocity in (A.1) can still give unreal solutions. If the interfacial transfer is much slower than the adjusted Stokes' velocity, the water cut in the oil phase above the oil-water interface can get higher than the WC at the phase inversion point. This is avoided by instead using

$$v^a = \begin{cases} v^s \left(\frac{WC^p - WC^{\text{receiver}}}{WC^p} \right)^n & \text{for Water in Oil} \\ v^s \left(\frac{WC^{\text{receiver}} - WC^p}{1 - WC^p} \right)^n & \text{for Oil in Water} \end{cases} \tag{A.2}$$

where the water cut at the phase inversion point, WC^p, is used.

Appendix B

Geometric Relationships

Some geometric and trigonometric relation is needed when deriving the level equations and volumes of the control volumes in the gravity separator. The relation between the height h, the angle θ to the height, and the area, A_h, of the cross section under the height is shown in Figure B.1.

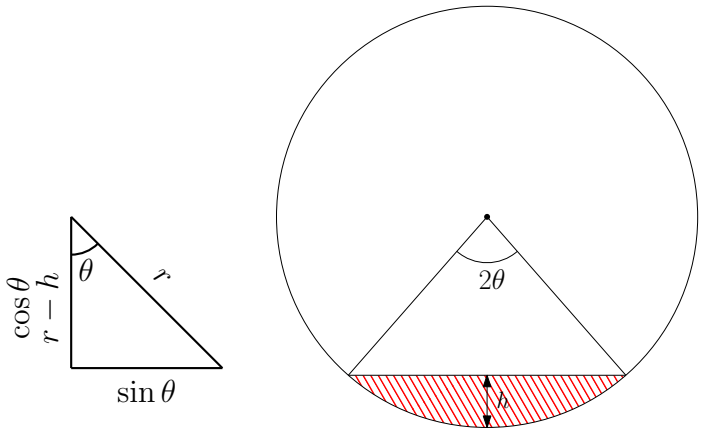


Figure B.1: Circle with height, angle to height and area under height. The trigonometric relations are also shown.

B.1 Change in Cross Section Area

The height given by the angle is

$$h = r(1 - \cos\theta) \tag{B.1}$$

where r is the radius. The derivative of the height with respect to the angle, and the derivative of the angle with respect to the height is

$$\frac{dh}{d\theta} = rsin\theta \tag{B.2}$$

$$\frac{d\theta}{dh} = \frac{1}{r\csc\theta} \tag{B.3}$$

The area of the segment (shaded region in Figure B.1) is the area of the sector minus the area of the triangle.

$$\{Area\ of\ sector\} = \pi r^2 \frac{2\theta}{2\pi} = r^2\theta \tag{B.4}$$

$$\{Area\ of\ triangle\} = \frac{1}{2}(r\cos\theta)(2r\sin\theta) = \frac{1}{2}r^2\sin 2\theta \tag{B.5}$$

$$\{Area\ of\ segment\} = r^2\theta - \frac{1}{2}r^2 sin2\theta = r^2(\theta - \frac{1}{2}\sin 2\theta) \tag{B.6}$$

The trigonometric relation in the triangle gives

$$\cos\theta = \left(1 - \frac{h}{r}\right) \tag{B.7}$$

$$\theta = \cos^{-1}\left(1 - \frac{h}{r}\right) \tag{B.8}$$

The trigonometric relations

$$\sin 2\theta = 2\sin\theta\cos\theta \tag{B.9}$$

$$\sin\theta = \sqrt{1 - \cos^2\theta} \tag{B.10}$$

combined with (B.7) gives

$$\sin 2\theta = 2\sqrt{1 - \left(1 - \frac{h}{r}\right)^2}\left(1 - \frac{h}{r}\right) \tag{B.11}$$

Inserting (B.8) and (B.11) into (B.6) gives:

$$A = r^2\cos^{-1}\left(1 - \frac{h}{r}\right) - (r - h)\sqrt{h(2r - h)} \tag{B.12}$$

The derivative of (B.6) gives

$$\frac{dA}{d\theta} = r^2(1 - \cos 2\theta) \tag{B.13}$$

The trigonometric relation

$$\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b) \tag{B.14}$$

gives

$$\cos(2\theta) = \cos^2(\theta) - \sin^2(\theta) \tag{B.15}$$

$$\cos(2\theta) = \cos^2(\theta) - (1 - \cos^2(\theta)) = 2\cos(\theta) - 1 \tag{B.16}$$

which with the help of (B.7) solves to

$$\cos 2\theta = 2\left(1 - \frac{h}{r}\right)^2 - 1 \tag{B.17}$$

Inserting (B.17) into (B.13) gives

$$\frac{dA}{d\theta} = 2r^2\left(1 - \left(1 - \frac{h}{r}\right)^2\right) \tag{B.18}$$

Inserting (B.10) and (B.7) in (B.3) gives that the height derivative of the angle is

$$\frac{d\theta}{dh} = \frac{1}{r\sin\theta} = \left(r\sqrt{1-\cos^2\theta}\right)^{-1} = \left(r\sqrt{1-\left(1-\frac{h}{r}\right)^2}\right)^{-1} \tag{B.19}$$

The product of $\frac{dA}{d\theta}$ and $\frac{d\theta}{dh}$ becomes

$$\frac{dA}{d\theta}\frac{d\theta}{dh} = \frac{2r^2\left(1-\left(1-\frac{h}{r}\right)^2\right)}{r\sqrt{1-\left(1-\frac{h}{r}\right)^2}} = 2r\sqrt{1-\left(1-\frac{h}{r}\right)^2} \tag{B.20}$$

$$\frac{dA}{d\theta}\frac{d\theta}{dh} = 2\sqrt{r^2-(r-h)^2} = 2\sqrt{r^2-(r^2-2hr+h^2)} \tag{B.21}$$

$$\frac{dA}{d\theta}\frac{d\theta}{dh} = 2\sqrt{h(2r-h)} \tag{B.22}$$

B.2 Level Dependent Control Volumes

The volume of the control volumes depends on the cross section area, A_c, of the control volumes. The size of the control volumes are constant in horizontal direction, but they are different for each layer, j. The layers in the water have the same thickness, but the area is different. The same is for the oil layer under the weir height. An illustration of the layers is shown in Figure

3.3. Since the cross section area depends on the levels, the size of the control volumes depends on the water level. The volume in the control volumes at layer j is

$$V_j = \frac{L}{n_x} A_{c,j} = \frac{L}{n_x}(A_j - A_{j-1}) \tag{B.23}$$

and the time derivative is

$$\frac{dV_j}{dt} = \frac{L}{n_x}(\frac{dA_j}{dt} - \frac{dA_{j-1}}{dt}) \tag{B.24}$$

The area under layer j is calculated with (B.12) with the height of the layer h_j, $A_j = f(h_j)$. The heights depends on the water level h_{lw}, $h_j = f(h_{lw})$, and the heights are given by the water volume, $h_{lw} = f(V_{lw})$. The time derivative of the area A_j is therefor given by

$$\frac{dA_j}{dt} = \frac{dA_j}{dh_j}\frac{dh_j}{dt} = \frac{dA_j}{dh_j}\frac{dh_j}{dh_{lw}}\frac{dh_{lw}}{dt} \tag{B.25}$$

The derivative of (B.12) is

$$\frac{dA_j}{dh_j} = 2\sqrt{h_j(2r - h_j)} \tag{B.26}$$

The heights h_j is given by

$$h_j = \begin{cases} \frac{j}{n_{z,w}} h_{lw} & \text{if } 0 < j \leq n_{z,w} \\ h_{lw} + \frac{j - n_{z,w}}{n_{z,o} - 1}(h_w - h_{lw}) & \text{if } n_{z,w} < j < n_{z,w} + n_{z,o} \end{cases} \tag{B.27}$$

and their derivatives is

$$\frac{dh_j}{dh_{lw}} = \begin{cases} \frac{j}{n_{z,w}} & \text{if } 0 < j \leq n_{z,w} \\ 1 - \frac{j - n_{z,w}}{n_{z,o} - 1} & \text{if } n_{z,w} < j < n_{z,w} + n_{z,o} \end{cases} \tag{B.28}$$

Combining (B.25), (B.26) and (B.28) gives the time derivative of the areas A_j.

Appendix C

MATLAB Scripts and Functions

The model scripts are given in this Appendix. The steady state problem of GS1 is solved using the script in section C.1. The parameters are given in C.3. The model equations to be solved is defined with CasADi in the script in C.2. The two functions, **rateOfFormation.m** and **stokesVelocity.m**, are used to calculate the rates of coalescence and the Stokes' terminal velocity. The dynamic model of GS1 is shown in section C.6, and the changes between the GS1 model and GS2 model is shown in section C.7.

C.1 solve_init_GS1.m

```
%% Solve Steady State solution of the three phase Gravity Separator.
% The model equation defined with CasADi are given in script: casADi_GS1.m

% Author: Sindre Johan Heggheim
% Date: 03.06.2018

close all; clear; clc;
addpath('Z:\MASTEROPPGAVE\casadi-matlabR2014b-v3.3.0');
import casadi.*
run('init_param') % Initialize Parameters
run('casADi_GS1') % Initialize CasADi model

hl_sp = 3.5;
hlw_sp = 1.8;

```

```
%% Solve problem:
problem.f = 0*X'*X;%(1-WC_w(end-1,1));
problem.x = [X;U];
problem.g = [f(X,U,D);y(X,U,D)];
problem.p = D;

% Limits on states (Densities, Levels, Pressure):
x_lb = [zeros(ns_o+ns_w,1);    hl_sp; hlw_sp; 70 ];
x_ub = [inf*ones(ns_o+ns_w,1); hl_sp; hlw_sp; 90 ];
% Limits on system inputs: (Product flows):
u_lb = [U_val(1)*0.99];
u_ub = [U_val(1)*1.01];
% Limits on x_dots and function returns y:
g_lb = [zeros(ns_tot,1);-inf;  zeros(ny-1,1)   ; -inf*ones(2,1) ];
g_ub = [zeros(ns_tot,1); inf; inf*ones(ny-1,1) ;  inf*ones(2,1) ];

% problem Solver
ipopt_opt.ipopt.max_iter = 3000;
solver = nlpsol('solver','ipopt',problem,ipopt_opt);
load('x0_gs1.mat');
sol=solver('p',D_val,'x0',[x0;U_val(1)],'lbx',[x_lb;u_lb],...
    'ubx',[x_ub;u_ub],'lbg',g_lb,'ubg',g_ub);

%% RESULTS:
g_sol = full(sol.g);
xdot_sol = g_sol(1:ns_tot);
y_sol = g_sol(ns_tot+1:end);

% Interfacial flow, Product flows and flows inside separator:
solution.Fint = y_sol(1);       % Total interfacial volumetric flow
solution.F = y_sol(2:4)';       % Product flows
solution.q = rot90(reshape(y_sol(5:4+(nx-1)*(nz_w+nz_o)), nx-1,nz_w+nz_o));

% Checking mass balance:
solution.mb = y_sol(end-1:end)';% Formation of water and oil

xdottop_sol = xdot_sol(1:ns_o);
xdotbottom_sol = xdot_sol(ns_o+1:ns_o+ns_w);
```

```
54 for i = 1:10
55 rhosdot_t_sol(:,:,i)=...
56     rot90(reshape(xdottop_sol((i-1)*nx*nz_o+1:i*nx*nz_o),nx,nz_o));
57 rhosdot_b_sol(:,:,i)=...
58     rot90(reshape(xdotbottom_sol((i-1)*nx*nz_w+1:i*nx*nz_w),nx,nz_w));
59 end
60
61 x_sol = full(sol.x);     % States and input solutions:
62 x0 = x_sol(1:ns_tot);
63 save('x0_gs1','x0')
64 solution.hl = x_sol(ns_tot-2);
65 solution.hlw= x_sol(ns_tot-1);
66 solution.P  =  x_sol(ns_tot);
67 solution.U  = x_sol(ns_tot+1:end);
68 xtop_sol    = x_sol(1:ns_o);
69 xbottom_sol = x_sol(ns_o+1:ns_o+ns_w);
70 for i = 1:10
71 rhos_t_sol(:,:,i)=...
72     rot90(reshape(xtop_sol((i-1)*nx*nz_o+1:i*nx*nz_o),nx,nz_o));
73 rhos_b_sol(:,:,i)=...
74     rot90(reshape(xbottom_sol((i-1)*nx*nz_w+1:i*nx*nz_w),nx,nz_w));
75 end
76 rho_w_sol = sum(rhos_t_sol,3);  % Partial density WiO
77 rho_o_sol = sum(rhos_b_sol,3);  % Partial density OiW
78 solution.WC=...
79     [rho_w_sol/rho_w;1-rho_o_sol(:,1:end-1)/rho_o,rho_o_sol(:,end)/rho_w];
80 solution
81 WCsol = solution.WC
82 OCsol = 1-WCsol;
83
84 % Save initial simulation states:
85 gs1.WC_R = WCsol(end,end);
86 WC_L = WCsol(end,end-1);
87 gs1.ppm_L = (1-WC_L)*1e6;
88 gs1.F_G0 = solution.F(1);
89 gs1.F_R0 = solution.F(2);
90 gs1.F_L0 = solution.F(3);
91 gs1.hl_sp = hl_sp;
```

```
92 gs1.hlw_sp = hlw_sp;
93 gs1.x0 = x0;
94 save('gs1_simParam','gs1')
95
96 % Saving partial densities in oil product, which are used in second stage
97 % separator calculations:
98 rhos_gs2feed(:,1) = rhos_b_sol(end,end,:);
99 Fin_gs2feed = solution.F(2);
100 save('GS2_feed','rhos_gs2feed','Fin_gs2feed');
```

C.2 casADi_GS1.m

```
% Steady state three phase gravity separator model with coalescing droplets
% This script defines the model equations and function with casadi which
% are to be solved using the IPOPT solver.
% steady state problem

% Author: Sindre Johan Heggheim
% Date: 03.06.2018
%% Initialize CasADi Model:
% Initializing casADi model equations and functions.
addpath('Z:\MASTEROPPGAVE\casadi-matlabR2014b-v3.3.0');
import casadi.*
%% Definition of CasADi variables :
X = SX.sym('x',ns_tot,1);                    % States
X_dot= SX.sym('x_dot',ns_tot,1);             % Change in states
D = SX.sym('D',nd,1);                        % Disturbance
U = SX.sym('U',nu,1);                        % System inputs

% OIL SIDE:
Xo = X(1:ns_o);
rho1_w = reshape(Xo(1:1*nz_o*nx),nx,nz_o);
rho2_w = reshape(Xo(1*nz_o*nx+1:2*nz_o*nx),nx,nz_o);
rho3_w = reshape(Xo(2*nz_o*nx+1:3*nz_o*nx),nx,nz_o);
rho4_w = reshape(Xo(3*nz_o*nx+1:4*nz_o*nx),nx,nz_o);
rho5_w = reshape(Xo(4*nz_o*nx+1:5*nz_o*nx),nx,nz_o);
rho6_w = reshape(Xo(5*nz_o*nx+1:6*nz_o*nx),nx,nz_o);
rho7_w = reshape(Xo(6*nz_o*nx+1:7*nz_o*nx),nx,nz_o);
rho8_w = reshape(Xo(7*nz_o*nx+1:8*nz_o*nx),nx,nz_o);
rho9_w = reshape(Xo(8*nz_o*nx+1:9*nz_o*nx),nx,nz_o);
rho10_w= reshape(Xo(9*nz_o*nx+1:10*nz_o*nx),nx,nz_o);
Xo_dot = X_dot(1:ns_o);
rho1_dot_w = reshape(Xo_dot(1:1*nz_o*nx),nx,nz_o);
rho2_dot_w = reshape(Xo_dot(1*nz_o*nx+1:2*nz_o*nx),nx,nz_o);
rho3_dot_w = reshape(Xo_dot(2*nz_o*nx+1:3*nz_o*nx),nx,nz_o);
```

```
rho4_dot_w = reshape(Xo_dot(3*nz_o*nx+1:4*nz_o*nx),nx,nz_o);
rho5_dot_w = reshape(Xo_dot(4*nz_o*nx+1:5*nz_o*nx),nx,nz_o);
rho6_dot_w = reshape(Xo_dot(5*nz_o*nx+1:6*nz_o*nx),nx,nz_o);
rho7_dot_w = reshape(Xo_dot(6*nz_o*nx+1:7*nz_o*nx),nx,nz_o);
rho8_dot_w = reshape(Xo_dot(7*nz_o*nx+1:8*nz_o*nx),nx,nz_o);
rho9_dot_w = reshape(Xo_dot(8*nz_o*nx+1:9*nz_o*nx),nx,nz_o);
rho10_dot_w= reshape(Xo_dot(9*nz_o*nx+1:10*nz_o*nx),nx,nz_o);
WC_o = (rho1_w + rho2_w + rho3_w + rho4_w + rho5_w ...
      + rho6_w + rho7_w + rho8_w + rho9_w + rho10_w)/rho_w;

% WATER SIDE:
Xw = X(ns_o+1:ns_o+ns_w);
rho1_o = reshape(Xw(1:1*nz_w*nx),nx,nz_w);
rho2_o = reshape(Xw(1*nz_w*nx+1:2*nz_w*nx),nx,nz_w);
rho3_o = reshape(Xw(2*nz_w*nx+1:3*nz_w*nx),nx,nz_w);
rho4_o = reshape(Xw(3*nz_w*nx+1:4*nz_w*nx),nx,nz_w);
rho5_o = reshape(Xw(4*nz_w*nx+1:5*nz_w*nx),nx,nz_w);
rho6_o = reshape(Xw(5*nz_w*nx+1:6*nz_w*nx),nx,nz_w);
rho7_o = reshape(Xw(6*nz_w*nx+1:7*nz_w*nx),nx,nz_w);
rho8_o = reshape(Xw(7*nz_w*nx+1:8*nz_w*nx),nx,nz_w);
rho9_o = reshape(Xw(8*nz_w*nx+1:9*nz_w*nx),nx,nz_w);
rho10_o= reshape(Xw(9*nz_w*nx+1:10*nz_w*nx),nx,nz_w);
Xw_dot = X_dot(ns_o+1:ns_o+ns_w);
rho1_dot_o = reshape(Xw_dot(1:1*nz_w*nx),nx,nz_w);
rho2_dot_o = reshape(Xw_dot(1*nz_w*nx+1:2*nz_w*nx),nx,nz_w);
rho3_dot_o = reshape(Xw_dot(2*nz_w*nx+1:3*nz_w*nx),nx,nz_w);
rho4_dot_o = reshape(Xw_dot(3*nz_w*nx+1:4*nz_w*nx),nx,nz_w);
rho5_dot_o = reshape(Xw_dot(4*nz_w*nx+1:5*nz_w*nx),nx,nz_w);
rho6_dot_o = reshape(Xw_dot(5*nz_w*nx+1:6*nz_w*nx),nx,nz_w);
rho7_dot_o = reshape(Xw_dot(6*nz_w*nx+1:7*nz_w*nx),nx,nz_w);
rho8_dot_o = reshape(Xw_dot(7*nz_w*nx+1:8*nz_w*nx),nx,nz_w);
rho9_dot_o = reshape(Xw_dot(8*nz_w*nx+1:9*nz_w*nx),nx,nz_w);
rho10_dot_o= reshape(Xw_dot(9*nz_w*nx+1:10*nz_w*nx),nx,nz_w);
rho_oiw_tot= rho1_o + rho2_o + rho3_o + rho4_o + rho5_o ...
            +rho6_o + rho7_o + rho8_o + rho9_o + rho10_o;
WC_w = [1 - rho_oiw_tot(1:end-1,:)/rho_o; rho_oiw_tot(end,:)/rho_w];

% Other:
```

```
72 h_l  = X(ns_tot-2);
73 h_lw = X(ns_tot-1);
74 P    = X(ns_tot-0);
75 Fin_g = D(1);
76 Fin_l = D(2);
77 e_in  = D(3);
78 d_A   = D(4);
79
80 %% Feed calculations
81 Fin_w_tot = e_in*Fin_l;          % Total feed flow of water
82 Fin_o_tot = (1-e_in)*Fin_l;      % Total feed flow of oil
83
84 Fin_oo = g_o*Fin_o_tot;          % Feed of oil to oil phase
85 Fin_ow = (1-g_o)*Fin_o_tot;      % Feed of oil to water phase
86 Fin_ww = g_w*Fin_w_tot;          % Feed of water to water phase
87 Fin_wo = (1-g_w)*Fin_w_tot;      % Feed of water to oil phase
88
89 Fin_op = Fin_oo+Fin_wo;          % Liquid feed to oil phase
90 Fin_wp = Fin_ow+Fin_ww;          % Liquid feed to water phase
91
92 e_in_op = Fin_wo/Fin_op;         % Water cut in oil phase feed
93 e_in_wp = Fin_ww/Fin_wp;         % Water cut in water phase feed
94
95 % Partial densities in feeds
96 rhosFeed_waterPhase = [(1-e_in_wp)*rho_o; zeros(9,1)];
97 rhosFeed_oilPhase =   [   e_in_op*rho_w;  zeros(9,1)];
98
99 %% Geometric calculations
100 % Segment heights [m]
101 hs_o = (1:nz_o-1)'/(nz_o-1)*(h_w-h_lw) + h_lw ;
102 hs_w = (1:nz_w)'/nz_w*h_lw ;
103 hs = [hs_w;hs_o;h_l];
104
105 % Cross section areas [m2]
106 A_w = r^2*acos(1-h_w/r)-(r-h_w).*sqrt(h_w.*(2*r-h_w));
107 A_hs =  r^2*acos(1-hs/r)-(r-hs).*sqrt(hs.*(2*r-hs));
108 A_l = A_hs(nz_w+nz_o);
109 A_lw = A_hs(nz_w);
```

```
110 Ac_hs = A_hs-[0; A_hs(1:end-1)];
111 A_tank = pi*r^2;
112
113 % Horizontal transfer areas [m2]
114 At_o = 2*L/nx*sqrt(hs_o.*(2*r-hs_o));
115 At_w = 2*L/nx*sqrt(hs_w.*(2*r-hs_w));
116 A_int = At_w(end);
117 % A_int = 1;
118
119 % Control volumes [m3]
120 Vcv_w = Ac_hs(1:nz_w)*L/nx;                  % Under water level
121 Vcv_o = Ac_hs(nz_w+1:nz_w+nz_o-1)*L/nx;      % Over water level under weir
122 Vcv_top = Ac_hs(end)*L/nx;                   % Over weir
123
124 %% Coalescence of droplets to continuous phases:
125 i = 1:nx-2;
126 rhos_w = [rho1_w(i,1),rho2_w(i,1),rho3_w(i,1),rho4_w(i,1),rho5_w(i,1),...
127     rho6_w(i,1),rho7_w(i,1),rho8_w(i,1),rho9_w(i,1),rho10_w(i,1)]';
128 rhos_o=[rho1_o(i,end),rho2_o(i,end),rho3_o(i,end),rho4_o(i,end),...
129         rho5_o(i,end),rho6_o(i,end),rho7_o(i,end),rho8_o(i,end),...
130         rho9_o(i,end),rho10_o(i,end)]';
131
132 rateVec_water = (k_int_w*A_int*WC_o(i,1))';
133 rateVec_oil = (k_int_o*A_int*(1-WC_w(i,end)))';
134 rateMat_water = repmat(rateVec_water,10,1);
135 rateMat_oil = repmat(rateVec_oil,10,1);
136
137 m_coal_water = rhos_w.*rateMat_water; % Rate of mass, water to water phase
138 m_coal_oil   = rhos_o.*rateMat_oil;   % Rate of mass, oil to oil phase
139
140 %% Flow calculations:
141 % Calculating steady state product flows for given levels
142
143 % Interfacial volumetric flows [m3/s]:
144 F_int = (sum(m_coal_water)/rho_w - sum(m_coal_oil)/rho_o)';
145 Fint_sum = tril(ones(nx-2))'*F_int; % Reverse cumulative sum of F_int
146 F_int_tot = sum(F_int);             % Total flow
147
```

```
148 % Calculated Product flows: (Steady state)
149 F = [ U(1); Fin_l-(Fin_wp+F_int_tot); (Fin_wp+F_int_tot)];
150
151 % Horizontal flows through separator [m3/s]:
152 q_out = [ F(3)*Ac_hs(1:nz_w)/A_lw;
153           F(2)*Ac_hs(nz_w+1:nz_w+nz_o)/(A_l-A_lw)];
154 q_waterphase = repmat(q_out(1:nz_w),1,7) - (Ac_hs(1:nz_w)/A_lw)*Fint_sum';
155 q_oilphase = repmat(q_out(nz_w+1:nz_w+nz_o),1,7) + ...
156                (Ac_hs(nz_w+1:nz_w+nz_o)/(A_l-A_lw))*Fint_sum';
157 q = [q_waterphase; q_oilphase]';
158 q = [q;q_out'];
159 % q gives the horizontal feed to each control volume left of the weir
160
161 % Change to real inlet flows:
162 q(1,1:nz_w)=Fin_wp*Ac_hs(1:nz_w)/A_lw;
163 q(1,nz_w+1:nz_w+nz_o) = Fin_op*Ac_hs(nz_w+1:nz_w+nz_o)/(A_l-A_lw);
164
165 %% Level calculations:
166 Vl_dot = Fin_l-F(2)-F(3);
167 hl_dot = Vl_dot/(2*L*sqrt(h_l*(2*r-h_l)));
168
169 Vlw_dot = Fin_wp + F_int_tot - F(3);
170 hlw_dot = Vlw_dot/(2*L*(nx-1)/nx*sqrt(h_lw*(2*r-h_lw)));
171
172 % Change in level result in changed control volumes:
173 j = (1:nz_w)';              % Water phase
174     dAdt_water = 2*sqrt(hs(j).*(2*r-hs(j))).*(j)/nz_w*hlw_dot;
175 j = (nz_w+1:nz_w+nz_o-1)';  % Oil phase:
176     dAdt_oil = 2*sqrt(hs(j).*(2*r-hs(j))).*(1-(j-nz_w)/(nz_o-1))*hlw_dot;
177 dAdt = [dAdt_water; dAdt_oil];
178 dVcvdt = L/nx*(dAdt-[0; dAdt(1:end-1)]);
179
180 %% Gas pressure calculations:
181 m_g_in = Fin_g*rho_g;
182 n_g_in = m_g_in/Mm_g;                   % Molar rate of gas in  [mol/s]
183 n_g_out = P*1e5/(R_gas*T)*F(1);         % Molar rate of gas out [mol/s]
184 V_l = L*A_l;                            % Total liquid volume   [m3]
185 V_g = pi*r^2*L-V_l;                     % Total gas volume      [m3]
```

```
186 P_dot = (R_gas*T/V_g*(n_g_in-n_g_out) + P*1e5/V_g*(Fin_l-F(2)-F(3)))*1e-5;
187
188 %% Density calculations:
189 v_stokes = -stokesVelocity(rho_o,rho_w,mu_w,d_A);
190
191 % % % % Main separator part, water phase
192 for j = 1:nz_w % Under water level:
193 for i = 1:nx-1 % Left of weir:
194 rhos = [rho1_o(i,j),rho2_o(i,j),rho3_o(i,j),rho4_o(i,j),rho5_o(i,j),...
195         rho6_o(i,j),rho7_o(i,j),rho8_o(i,j),rho9_o(i,j),rho10_o(i,j)]';
196 R = rateOfFormation(rhos,k);
197 v_adp_in = v_stokes*((WC_w(i,j)-0.7)/0.3)^n_adp;
198
199 if j == 1           % At bottom
200 rhos_Over=[rho1_o(i,j+1),rho2_o(i,j+1),rho3_o(i,j+1),rho4_o(i,j+1),...
201             rho5_o(i,j+1),rho6_o(i,j+1),rho7_o(i,j+1),rho8_o(i,j+1),...
202             rho9_o(i,j+1),rho10_o(i,j+1)]';
203 v_adp_out = v_stokes*((WC_w(i,j+1)-0.7)/0.3)^n_adp;
204 elseif j == nz_w    % At interface
205 rhos_Under=[rho1_o(i,j-1),rho2_o(i,j-1),rho3_o(i,j-1),rho4_o(i,j-1),...
206             rho5_o(i,j-1),rho6_o(i,j-1),rho7_o(i,j-1),rho8_o(i,j-1),...
207             rho9_o(i,j-1),rho10_o(i,j-1)]';
208 else
209 rhos_Over=[rho1_o(i,j+1),rho2_o(i,j+1),rho3_o(i,j+1),rho4_o(i,j+1),...
210             rho5_o(i,j+1),rho6_o(i,j+1),rho7_o(i,j+1),rho8_o(i,j+1),...
211             rho9_o(i,j+1),rho10_o(i,j+1)]';
212 rhos_Under=[rho1_o(i,j-1),rho2_o(i,j-1),rho3_o(i,j-1),rho4_o(i,j-1),...
213             rho5_o(i,j-1),rho6_o(i,j-1),rho7_o(i,j-1),rho8_o(i,j-1),...
214             rho9_o(i,j-1),rho10_o(i,j-1)]';
215 v_adp_out = v_stokes*((WC_w(i,j+1)-0.7)/0.3)^n_adp;
216 end%if
217
218 if i == 1  % At separator inlet
219   rhos_Left = rhosFeed_waterPhase;
220 else
221   rhos_Left=[rho1_o(i-1,j),rho2_o(i-1,j),rho3_o(i-1,j),rho4_o(i-1,j),...
222             rho5_o(i-1,j),rho6_o(i-1,j),rho7_o(i-1,j),rho8_o(i-1,j),...
223             rho9_o(i-1,j),rho10_o(i-1,j)]';
```

```
224 end%if
225
226 if i == nx-1 % at weir: (vertical convective flow)
227   if j == nz_w % At interface
228    rho_dots=(rhos_Left-rhos)*q_out(j)/Vcv_w(j) +R -rhos*dVcvdt(j)/Vcv_w(j);
229   else
230    rho_dots=(rhos_Left*q_out(j)+rhos_Over*sum(q_out(j+1:nz_w)))/Vcv_w(j)...
231             - rhos*sum(q_out(j:nz_w))/Vcv_w(j) +R -rhos*dVcvdt(j)/Vcv_w(j);
232   end%if
233 else % main part of separator
234 rhos_Right=[rho1_o(i+1,j),rho2_o(i+1,j),rho3_o(i+1,j),rho4_o(i+1,j),...
235     rho5_o(i+1,j),rho6_o(i+1,j),rho7_o(i+1,j),rho8_o(i+1,j),...
236     rho9_o(i+1,j),rho10_o(i+1,j)]';
237    conv_trans = rhos_Left*max(q(i,j),0)  - rhos*max(q(i+1,j),0) ...
238               + rhos_Right*max(-q(i,j),0)- rhos*max(-q(i+1,j),0);
239   if j == nz_w % At interface
240  rho_dots=  (                      conv_trans              )/Vcv_w(j)...
241     + ( rhos_Under.*v_adp_in*At_w(j-1) - m_coal_oil(:,i)  )/Vcv_w(j)...
242     + (                     R*Vcv_w(j)  - rhos*dVcvdt(j)   )/Vcv_w(j);
243   elseif j == 1 % At bottom
244  rho_dots=   (                         conv_trans          )/Vcv_w(j)...
245             +(                   - rhos.*v_adp_out*At_w(j) )/Vcv_w(j)...
246             +(     R*Vcv_w(j)    - rhos*dVcvdt(j)          )/Vcv_w(j);
247   else
248  rho_dots=   (     rhos_Left*q(i,j) - rhos*q(i+1,j)          )/Vcv_w(j)...
249    +(rhos_Under.*v_adp_in*At_w(j-1) - rhos.*v_adp_out*At_w(j))/Vcv_w(j)...
250             +(         R*Vcv_w(j)    - rhos*dVcvdt(j)         )/Vcv_w(j);
251   end%if
252 end%if
253 rho1_dot_o(i,j) = rho_dots(1);
254 rho2_dot_o(i,j) = rho_dots(2);
255 rho3_dot_o(i,j) = rho_dots(3);
256 rho4_dot_o(i,j) = rho_dots(4);
257 rho5_dot_o(i,j) = rho_dots(5);
258 rho6_dot_o(i,j) = rho_dots(6);
259 rho7_dot_o(i,j) = rho_dots(7);
260 rho8_dot_o(i,j) = rho_dots(8);
261 rho9_dot_o(i,j) = rho_dots(9);
```

```
rho10_dot_o(i,j) = rho_dots(10);
end
end

% % % % Main separator part, Oil phase
v_stokes = stokesVelocity(rho_w,rho_o,mu_o,d_A);
for j = 1:nz_o-1 % Under weir:
for i = 1:nx-1 % Left of weir:
rhos = [rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
        rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
rhos_Over=[rho1_w(i,j+1),rho2_w(i,j+1),rho3_w(i,j+1),rho4_w(i,j+1),...
    rho5_w(i,j+1),rho6_w(i,j+1),rho7_w(i,j+1),rho8_w(i,j+1),...
    rho9_w(i,j+1),rho10_w(i,j+1)]';
R = rateOfFormation(rhos,k);
v_adp_in = v_stokes*((0.7-WC_o(i,j))/0.7)^n_adp;

if i == 1
  rhos_Left = rhosFeed_oilPhase;
else
  rhos_Left=[rho1_w(i-1,j),rho2_w(i-1,j),rho3_w(i-1,j),rho4_w(i-1,j),...
      rho5_w(i-1,j),rho6_w(i-1,j),rho7_w(i-1,j),rho8_w(i-1,j),...
      rho9_w(i-1,j),rho10_w(i-1,j)]';
end%if

if i == nx-1 % at weir: (vertical convective flow)
  if j == 1 % convective vertical flow.
  rho_dots =( (rhos_Left - rhos)*q_out(nz_w+j))/Vcv_o(j) ...
               + R - rhos*dVcvdt(nz_w+j)/Vcv_o(j);
  else
    rhos_under=[rho1_w(i,j-1),rho2_w(i,j-1),rho3_w(i,j-1),rho4_w(i,j-1),...
            rho5_w(i,j-1),rho6_w(i,j-1),rho7_w(i,j-1),rho8_w(i,j-1),...
            rho9_w(i,j-1),rho10_w(i,j-1)]';
    rho_dots = (rhos_Left*q_out(nz_w+j) ...
                +  rhos_under*sum(q_out(nz_w+1:nz_w+j-1)))/Vcv_o(j) ...
                - rhos*sum(q_out(nz_w+1:nz_w+j))/Vcv_o(j) + R ...
                - rhos*dVcvdt(nz_w+j)/Vcv_o(j);
  end%if
else % main part of separator
```

```
300   if j == 1 % At interface
301   rho_dots = (   rhos_Left*q(i,nz_w+j) - rhos*q(i+1,nz_w+j)  )/Vcv_o(j)...
302       + (  rhos_Over.*v_adp_in*At_o(j) - m_coal_water(:,i)   )/Vcv_o(j) ...
303             + (              R*Vcv_o(j) - rhos*dVcvdt(nz_w+j) )/Vcv_o(j);
304   else
305   v_adp_out = v_stokes*((0.7-WC_o(i,j-1))/0.7)^n_adp;
306   rho_dots = ( rhos_Left*q(i,nz_w+j) - rhos*q(i+1,nz_w+j)    )/Vcv_o(j) ...
307     +(rhos_Over.*v_adp_in*At_o(j) - rhos.*v_adp_out*At_o(j-1))/Vcv_o(j) ...
308             + (            R*Vcv_o(j) - rhos*dVcvdt(nz_w+j)   )/Vcv_o(j);
309   end%if
310 end%if
311 rho1_dot_w(i,j) = rho_dots(1);
312 rho2_dot_w(i,j) = rho_dots(2);
313 rho3_dot_w(i,j) = rho_dots(3);
314 rho4_dot_w(i,j) = rho_dots(4);
315 rho5_dot_w(i,j) = rho_dots(5);
316 rho6_dot_w(i,j) = rho_dots(6);
317 rho7_dot_w(i,j) = rho_dots(7);
318 rho8_dot_w(i,j) = rho_dots(8);
319 rho9_dot_w(i,j) = rho_dots(9);
320 rho10_dot_w(i,j) = rho_dots(10);
321 end
322 end
323
324 % % % % Top left
325 j = nz_o;         % Over weir:
326 for i = 1:nx-1 % Left of weir:
327 rhos = [rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
328         rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
329 R = rateOfFormation(rhos,k);
330 v_adp_out = v_stokes*((0.7-WC_o(i,j-1))/0.7)^n_adp;
331
332 if i == 1
333   rhos_Left = rhosFeed_oilPhase;
334 else
335 rhos_Left=[rho1_w(i-1,j),rho2_w(i-1,j),rho3_w(i-1,j),rho4_w(i-1,j),...
336             rho5_w(i-1,j),rho6_w(i-1,j),rho7_w(i-1,j),rho8_w(i-1,j),...
337             rho9_w(i-1,j),rho10_w(i-1,j)]';
```

```
end%if
if i==nx-1
  rhos_under = [rho1_w(i,j-1),rho2_w(i,j-1),rho3_w(i,j-1),rho4_w(i,j-1),...
             rho5_w(i,j-1),rho6_w(i,j-1),rho7_w(i,j-1),rho8_w(i,j-1),...
             rho9_w(i,j-1),rho10_w(i,j-1)]';
  rho_dots = (rhos_Left*q_out(end)...
                + rhos_under*sum(q_out(nz_w+1:end-1)))/Vcv_top...
                - rhos*F(2)/Vcv_top + R - rhos/(Vcv_top*nx)*Vl_dot;
else
   rho_dots =   (rhos_Left*q(i,nz_w+j)-rhos*q(i+1,nz_w+j))/Vcv_top ...
       -(rhos.*v_adp_out)*At_o(j-1)/Vcv_top + R - rhos/(Vcv_top*nx)*Vl_dot;
end%if

rho1_dot_w(i,j) = rho_dots(1);
rho2_dot_w(i,j) = rho_dots(2);
rho3_dot_w(i,j) = rho_dots(3);
rho4_dot_w(i,j) = rho_dots(4);
rho5_dot_w(i,j) = rho_dots(5);
rho6_dot_w(i,j) = rho_dots(6);
rho7_dot_w(i,j) = rho_dots(7);
rho8_dot_w(i,j) = rho_dots(8);
rho9_dot_w(i,j) = rho_dots(9);
rho10_dot_w(i,j) = rho_dots(10);
end%for

% % % % Top Right
j = nz_o;    % Over weir:
i = nx;      % Right of weir:
rhos = [rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
    rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
rhos_Left = [rho1_w(i-1,j),rho2_w(i-1,j),rho3_w(i-1,j),rho4_w(i-1,j),...
             rho5_w(i-1,j),rho6_w(i-1,j),rho7_w(i-1,j),rho8_w(i-1,j),...
             rho9_w(i-1,j),rho10_w(i-1,j)]';
R = rateOfFormation(rhos,k);
rho_dots = (rhos_Left-rhos)*F(2)/Vcv_top + R - rhos/(Vcv_top*nx)*Vl_dot;
rho1_dot_w(i,j) = rho_dots(1);
rho2_dot_w(i,j) = rho_dots(2);
rho3_dot_w(i,j) = rho_dots(3);
```

```
376 rho4_dot_w(i,j) = rho_dots(4);
377 rho5_dot_w(i,j) = rho_dots(5);
378 rho6_dot_w(i,j) = rho_dots(6);
379 rho7_dot_w(i,j) = rho_dots(7);
380 rho8_dot_w(i,j) = rho_dots(8);
381 rho9_dot_w(i,j) = rho_dots(9);
382 rho10_dot_w(i,j) = rho_dots(10);
383
384 % % % % Oil out :
385 i = nx;              % Right of Weir
386 for j = 1:nz_o-1     % Under Weir, over interface
387 rhos=[rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
388     rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
389 rhos_Over = [rho1_w(i,j+1),rho2_w(i,j+1),rho3_w(i,j+1),rho4_w(i,j+1),...
390              rho5_w(i,j+1),rho6_w(i,j+1),rho7_w(i,j+1),rho8_w(i,j+1),...
391              rho9_w(i,j+1),rho10_w(i,j+1)]';
392 R = rateOfFormation(rhos,k);
393
394 rho_dots =(rhos_Over-rhos)*F(2)/Vcv_o(j) +R -rhos*dVcvdt(nz_w+j)/Vcv_o(j);
395 rho1_dot_w(i,j) = rho_dots(1);
396 rho2_dot_w(i,j) = rho_dots(2);
397 rho3_dot_w(i,j) = rho_dots(3);
398 rho4_dot_w(i,j) = rho_dots(4);
399 rho5_dot_w(i,j) = rho_dots(5);
400 rho6_dot_w(i,j) = rho_dots(6);
401 rho7_dot_w(i,j) = rho_dots(7);
402 rho8_dot_w(i,j) = rho_dots(8);
403 rho9_dot_w(i,j) = rho_dots(9);
404 rho10_dot_w(i,j) = rho_dots(10);
405 end%for
406
407 for j = 1:nz_w  % Under interface, all rhos are water!!!!!!!
408 rhos = [rho1_o(i,j),rho2_o(i,j),rho3_o(i,j),rho4_o(i,j),rho5_o(i,j),...
409     rho6_o(i,j),rho7_o(i,j),rho8_o(i,j),rho9_o(i,j),rho10_o(i,j)]';
410  if j == nz_w
411  rhos_Over=[rho1_w(i,1),rho2_w(i,1),rho3_w(i,1),rho4_w(i,1),rho5_w(i,1),...
412              rho6_w(i,1),rho7_w(i,1),rho8_w(i,1),rho9_w(i,1),rho10_w(i,1)]';
413  else
```

```
414   rhos_Over=[rho1_o(i,j+1),rho2_o(i,j+1),rho3_o(i,j+1),rho4_o(i,j+1),...
415              rho5_o(i,j+1),rho6_o(i,j+1),rho7_o(i,j+1),rho8_o(i,j+1),...
416              rho9_o(i,j+1),rho10_o(i,j+1)]';
417  end%if
418 R = rateOfFormation(rhos,k);
419 rho_dots = (rhos_Over-rhos)*F(2)/Vcv_w(j) + R - rhos*dVcvdt(j)/Vcv_w(j);
420 rho1_dot_o(i,j) = rho_dots(1);
421 rho2_dot_o(i,j) = rho_dots(2);
422 rho3_dot_o(i,j) = rho_dots(3);
423 rho4_dot_o(i,j) = rho_dots(4);
424 rho5_dot_o(i,j) = rho_dots(5);
425 rho6_dot_o(i,j) = rho_dots(6);
426 rho7_dot_o(i,j) = rho_dots(7);
427 rho8_dot_o(i,j) = rho_dots(8);
428 rho9_dot_o(i,j) = rho_dots(9);
429 rho10_dot_o(i,j) = rho_dots(10);
430 end%for
431
432 %% Check if mass balance is held:
433 % Water Balance:
434 WaterIn  = q(2,:)*([WC_w(1,:),WC_o(1,:)]');
435 WaterOut = q(8,:)*([WC_w(7,:),WC_o(7,:)]');
436 WaterInOilProduct   = WC_w(end,1)*F(2);
437 WaterInWaterProduct = WC_w(end-1,1)*F(3);
438 WaterInFeed = e_in*Fin_l;
439 % mb_w = WaterOut - WaterIn;                                % Local balance
440 mb_w = WaterInOilProduct + WaterInWaterProduct -WaterInFeed;% Total balance
441
442 % Oil Balance:
443 OilIn  = q(2,:)*(1-[WC_w(1,:),WC_o(1,:)]');
444 OilOut = q(8,:)*(1-[WC_w(7,:),WC_o(7,:)]');
445 OilInOilProduct     = (1-WC_w(end,1))*F(2);
446 OilInWaterProduct   = (1-WC_w(end-1,1))*F(3);
447 OilInFeed  = (1-e_in)*Fin_l;
448 % mb_o = OilOut - OilIn;                                % Local Balance
449 mb_o = OilInOilProduct + OilInWaterProduct - OilInFeed; % Total Balance
450
451 %% Model function returns and problem solver:
```

```
452 dots = [
453     reshape(rho1_dot_w,nx*nz_o,1);
454     reshape(rho2_dot_w,nx*nz_o,1);
455     reshape(rho3_dot_w,nx*nz_o,1);
456     reshape(rho4_dot_w,nx*nz_o,1);
457     reshape(rho5_dot_w,nx*nz_o,1);
458     reshape(rho6_dot_w,nx*nz_o,1);
459     reshape(rho7_dot_w,nx*nz_o,1);
460     reshape(rho8_dot_w,nx*nz_o,1);
461     reshape(rho9_dot_w,nx*nz_o,1);
462     reshape(rho10_dot_w,nx*nz_o,1);
463     reshape(rho1_dot_o,nx*nz_w,1);
464     reshape(rho2_dot_o,nx*nz_w,1);
465     reshape(rho3_dot_o,nx*nz_w,1);
466     reshape(rho4_dot_o,nx*nz_w,1);
467     reshape(rho5_dot_o,nx*nz_w,1);
468     reshape(rho6_dot_o,nx*nz_w,1);
469     reshape(rho7_dot_o,nx*nz_w,1);
470     reshape(rho8_dot_o,nx*nz_w,1);
471     reshape(rho9_dot_o,nx*nz_w,1);
472     reshape(rho10_dot_o,nx*nz_w,1);
473     hl_dot;
474     hlw_dot;
475     P_dot  ] ;
476 output = [F_int_tot; F; reshape(q,(nx-1)*(nz_w+nz_o),1); mb_w; mb_o];
477 ny = 1+3+(nx-1)*(nz_w+nz_o);
478
479 % Model functions:
480 f = Function('f', {X,U,D}, {dots});
481 y = Function('y', {X,U,D}, {output});
```

C.3 init_param.m

```
%% Initialize Parameters:
% Numbers:
nx = 9;                     % Number of horizontal control volumes
nz_o = 3;                   % Number of vertical control volumes oil
nz_w = 2;                   % Number of vertical control volumes water
nz_t = nz_o + nz_w;
ns_o = nz_o*nx*10;          % Number of states in oil.
ns_w = nz_w*nx*10;          % Number of states in water.
ns_tot = ns_o + ns_w + 3;   % Total number of states in model.
nd = 4;                     % Number of disturbances (F_g, F_l, e_in, d_A)
nu = 1;                     % Number of system inputs (Outflows)

% Separator sizes:
r = 2;                      % Radius of separator [m]
L = 15;                     % Lenght of separator [m]
h_w = r;                    % Height of weir [m]

% Other
R_gas = 8.314;              % Gas constant J/(K mol)
Mm_g = 0.01604;             % Molar mass of gas [kg/mol]
rho_g = 49.7;               % Density gas [kg/m3]
rho_o = 831.5;              % Density oil [kg/m3]
rho_w = 1000;               % Density water [kg/m3]
mu_o = 0.001;               % Viscosity oil [kg/sm]
mu_w = 0.0005;              % Viscosity water [kg/sm]
T = 328.5;                  % Temperature [K]
k = ones(25,1)*1e-0;        % reaction rate constants of coalescence
n_adp = 1;                  % Adapting parameter stokes velocity
k_int_w = 0.5;              % Rate constant, water droplet to water
k_int_o = 0.5;              % Rate constant, oil droplet to oil
g_o = 0.99;                 % Fraction of oilfeed to oil phase
g_w = 0.4;                  % Fraction of waterfeed flowing to water phase
g_o_2 = 0.999;              % Fraction of oilfeed to oil phase (gs2)
```

```
34 g_w_2 = 0.15;                % Fraction of waterfeed to water phase (gs2)
35
36 % Setpoints
37 % P_sp = 70;                 % Pressure [bar]
38 % hl_sp = 3.5;               % Liquid level [m]
39 % hlw_sp = 1.8;              % Water level [m]
40
41 % Disturbances:
42 F_in_val = [0.456,0.59]';    % Feed flow of gas and liquid [m3/s]
43 e_in_val = 0.15;             % Water cut in feed [-]
44 d_A_val = 150e-6;            % Diameter of water droplets in feed [m]
45 D_val = [F_in_val; e_in_val; d_A_val ];
46
47 % Input variables:
48 F_g_val = F_in_val(1);
49 F_o_val = F_in_val(2)*(1-e_in_val);
50 F_w_val = F_in_val(2)*e_in_val;
51 U_val = [F_g_val;F_o_val;F_w_val];
```

C.4 rateOfFormation.m

```
function rate = rateOfFormation(rho,k)
% Calculates the rate of change in partial densities due to coalescence.
% rho:  Vector of partial densities of droplets (10 sizes)
% k:    Vector of 25 reaction rate constants
% The 25 different equations of possible coelescence reaction of 10 droplet
% sizes are given in the N matrix.

% Author Sindre Johan Heggheim
% Date 03.06.2018
N = [ ...   5             10              15              20              25
-2 -1 -1 -1 -1 -1 -1 -1 -1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0;
 1 -1  0  0  0  0  0  0  0 -2 -1 -1 -1 -1 -1 -1  0  0  0  0  0  0  0  0  0;
 0  1 -1  0  0  0  0  0  0  0 -1  0  0  0  0  0 -2 -1 -1 -1 -1  0  0  0  0;
 0  0  1 -1  0  0  0  0  0  1  0 -1  0  0  0  0  0 -1  0  0  0 -2 -1 -1  0;
 0  0  0  1 -1  0  0  0  0  0  1  0 -1  0  0  0  0  0 -1  0  0  0 -1  0 -2;
 0  0  0  0  1 -1  0  0  0  0  0  1  0 -1  0  0  1  0  0 -1  0  0  0 -1  0;
 0  0  0  0  0  1 -1  0  0  0  0  0  1  0 -1  0  0  1  0  0 -1  0  0  0  0;
 0  0  0  0  0  0  1 -1  0  0  0  0  0  1  0 -1  0  0  1  0  0  1  0  0  0;
 0  0  0  0  0  0  0  1 -1  0  0  0  0  0  1  0  0  0  0  1  0  0  1  0  0;
 0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  1  0  0  0  0  1  0  0  1  1;
];  % 10 sizes times 25 reactions
r1   = rho(1)*rho(1:9).*k(1:9);
r2   = rho(2)*rho(2:8).*k(10:16);
r3   = rho(3)*rho(3:7).*k(17:21);
r4   = rho(4)*rho(4:6).*k(22:24);
r5   = rho(5)*rho(5).*k(25);
r = [r1;r2;r3;r4;r5];
rate=(N*r).*(1:10)'; % rate of formation with the relative changes in rho
end%fun
```

C.5 stokesVelocity.m

```
function v_t = stokesVelocity(rho_d,rho_c,mu_c,d_A)
% Calculates Stokes' terminal velocities of 10 different dispersed droplets
% The sizes of the droplets are the smallest 10 different sizes formed by
% coalescence from the smallest size d_A.
%
% rho_d:    Density of droplet phase
% rho_c:    Density of continuous phase
% mu_c:     Viscosity of continuous phase
% d_A:      Diameter of smallest droplet

% Author: Sindre Johan Heggheim
% Date: 03.06.2018
d = (1:10).^(1/3)*d_A;              % Diameters of the 10 droplet sizes [m]
g = 9.81;                           % Gravity acceleration [m/s2]
v_t=(g*d.^2*(rho_d-rho_c)/(18*mu_c))';% Stokes terminal velocity [m/s]
end%fun
```

C.6 GS1_model_sim.m

```
% Dynamic model of three phase gravity separator with coalescing droplets.
% X: Vector of states (partial densities, pressure and levels)
% D: Vector of disturbances (Gas and liquid feed flows and feed water cut)
% U: System inputs (Product flows of Gas, Oil and Water)

% Author: Sindre Johan Heggheim
% Date: 03.06.2018
function dx_and_y = GS1_model_sim(X,D,U)
run('init_param')
k_int_o = 0.4;

X_dot = zeros(ns_tot,1);
% OIL SIDE:
Xo = X(1:ns_o);
rho1_w = reshape(Xo(1:1*nz_o*nx),nx,nz_o);
rho2_w = reshape(Xo(1*nz_o*nx+1:2*nz_o*nx),nx,nz_o);
rho3_w = reshape(Xo(2*nz_o*nx+1:3*nz_o*nx),nx,nz_o);
rho4_w = reshape(Xo(3*nz_o*nx+1:4*nz_o*nx),nx,nz_o);
rho5_w = reshape(Xo(4*nz_o*nx+1:5*nz_o*nx),nx,nz_o);
rho6_w = reshape(Xo(5*nz_o*nx+1:6*nz_o*nx),nx,nz_o);
rho7_w = reshape(Xo(6*nz_o*nx+1:7*nz_o*nx),nx,nz_o);
rho8_w = reshape(Xo(7*nz_o*nx+1:8*nz_o*nx),nx,nz_o);
rho9_w = reshape(Xo(8*nz_o*nx+1:9*nz_o*nx),nx,nz_o);
rho10_w= reshape(Xo(9*nz_o*nx+1:10*nz_o*nx),nx,nz_o);
Xo_dot = X_dot(1:ns_o);
rho1_dot_w = reshape(Xo_dot(1:1*nz_o*nx),nx,nz_o);
rho2_dot_w = reshape(Xo_dot(1*nz_o*nx+1:2*nz_o*nx),nx,nz_o);
rho3_dot_w = reshape(Xo_dot(2*nz_o*nx+1:3*nz_o*nx),nx,nz_o);
rho4_dot_w = reshape(Xo_dot(3*nz_o*nx+1:4*nz_o*nx),nx,nz_o);
rho5_dot_w = reshape(Xo_dot(4*nz_o*nx+1:5*nz_o*nx),nx,nz_o);
rho6_dot_w = reshape(Xo_dot(5*nz_o*nx+1:6*nz_o*nx),nx,nz_o);
rho7_dot_w = reshape(Xo_dot(6*nz_o*nx+1:7*nz_o*nx),nx,nz_o);
rho8_dot_w = reshape(Xo_dot(7*nz_o*nx+1:8*nz_o*nx),nx,nz_o);
```

```
34 rho9_dot_w = reshape(Xo_dot(8*nz_o*nx+1:9*nz_o*nx),nx,nz_o);
35 rho10_dot_w= reshape(Xo_dot(9*nz_o*nx+1:10*nz_o*nx),nx,nz_o);
36 WC_o = (rho1_w + rho2_w + rho3_w + rho4_w + rho5_w ...
37       + rho6_w + rho7_w + rho8_w + rho9_w + rho10_w)/rho_w;
38
39 % WATER SIDE:
40 Xw = X(ns_o+1:ns_o+ns_w);
41 rho1_o = reshape(Xw(1:1*nz_w*nx),nx,nz_w);
42 rho2_o = reshape(Xw(1*nz_w*nx+1:2*nz_w*nx),nx,nz_w);
43 rho3_o = reshape(Xw(2*nz_w*nx+1:3*nz_w*nx),nx,nz_w);
44 rho4_o = reshape(Xw(3*nz_w*nx+1:4*nz_w*nx),nx,nz_w);
45 rho5_o = reshape(Xw(4*nz_w*nx+1:5*nz_w*nx),nx,nz_w);
46 rho6_o = reshape(Xw(5*nz_w*nx+1:6*nz_w*nx),nx,nz_w);
47 rho7_o = reshape(Xw(6*nz_w*nx+1:7*nz_w*nx),nx,nz_w);
48 rho8_o = reshape(Xw(7*nz_w*nx+1:8*nz_w*nx),nx,nz_w);
49 rho9_o = reshape(Xw(8*nz_w*nx+1:9*nz_w*nx),nx,nz_w);
50 rho10_o= reshape(Xw(9*nz_w*nx+1:10*nz_w*nx),nx,nz_w);
51 Xw_dot = X_dot(ns_o+1:ns_o+ns_w);
52 rho1_dot_o = reshape(Xw_dot(1:1*nz_w*nx),nx,nz_w);
53 rho2_dot_o = reshape(Xw_dot(1*nz_w*nx+1:2*nz_w*nx),nx,nz_w);
54 rho3_dot_o = reshape(Xw_dot(2*nz_w*nx+1:3*nz_w*nx),nx,nz_w);
55 rho4_dot_o = reshape(Xw_dot(3*nz_w*nx+1:4*nz_w*nx),nx,nz_w);
56 rho5_dot_o = reshape(Xw_dot(4*nz_w*nx+1:5*nz_w*nx),nx,nz_w);
57 rho6_dot_o = reshape(Xw_dot(5*nz_w*nx+1:6*nz_w*nx),nx,nz_w);
58 rho7_dot_o = reshape(Xw_dot(6*nz_w*nx+1:7*nz_w*nx),nx,nz_w);
59 rho8_dot_o = reshape(Xw_dot(7*nz_w*nx+1:8*nz_w*nx),nx,nz_w);
60 rho9_dot_o = reshape(Xw_dot(8*nz_w*nx+1:9*nz_w*nx),nx,nz_w);
61 rho10_dot_o= reshape(Xw_dot(9*nz_w*nx+1:10*nz_w*nx),nx,nz_w);
62 rho_oiw_tot= rho1_o + rho2_o + rho3_o + rho4_o + rho5_o ...
63             +rho6_o + rho7_o + rho8_o + rho9_o + rho10_o;
64 WC_w = [1 - rho_oiw_tot(1:end-1,:)/rho_o; rho_oiw_tot(end,:)/rho_w];
65
66 % Other:
67 h_l  = X(ns_tot-2);
68 h_lw = X(ns_tot-1);
69 P    = X(ns_tot-0);
70 Fin_g = D(1);
71 Fin_l = D(2);
```

```
72 e_in  = D(3);
73 d_A   = D_val(4);
74 F = [U(1);U(2);U(3)];
75
76 %% Feed calculations
77 Fin_w_tot = e_in*Fin_l;          % Total feed flow of water
78 Fin_o_tot = (1-e_in)*Fin_l;      % Total feed flow of oil
79
80 Fin_oo = g_o*Fin_o_tot;          % Feed of oil to oil phase
81 Fin_ow = (1-g_o)*Fin_o_tot;      % Feed of oil to water phase
82 Fin_ww = g_w*Fin_w_tot;          % Feed of water to water phase
83 Fin_wo = (1-g_w)*Fin_w_tot;      % Feed of water to oil phase
84
85 Fin_op = Fin_oo+Fin_wo;          % Liquid feed to oil phase
86 Fin_wp = Fin_ow+Fin_ww;          % Liquid feed to water phase
87
88 e_in_op = Fin_wo/Fin_op;         % Water cut in oil phase feed
89 e_in_wp = Fin_ww/Fin_wp;         % Water cut in water phase feed
90
91 % Partial densities in feeds
92 rhosFeed_waterPhase = [(1-e_in_wp)*rho_o; zeros(9,1)];
93 rhosFeed_oilPhase =   [   e_in_op*rho_w;  zeros(9,1)];
94
95 %% Geometric calculations
96 % Segment heights [m]
97 hs_o = (1:nz_o-1)'/(nz_o-1)*(h_w-h_lw) + h_lw ;
98 hs_w = (1:nz_w)'/nz_w*h_lw ;
99 hs = [hs_w;hs_o;h_l];
100
101 % Cross section areas [m2]
102 A_w = r^2*acos(1-h_w/r)-(r-h_w).*sqrt(h_w.*(2*r-h_w));
103 A_hs =  r^2*acos(1-hs/r)-(r-hs).*sqrt(hs.*(2*r-hs));
104 A_l = A_hs(nz_w+nz_o);
105 A_lw = A_hs(nz_w);
106 Ac_hs = A_hs-[0; A_hs(1:end-1)];
107 A_tank = pi*r^2;
108
109 % Horizontal transfer areas [m2]
```

```
110 At_o = 2*L/nx*sqrt(hs_o.*(2*r-hs_o));
111 At_w = 2*L/nx*sqrt(hs_w.*(2*r-hs_w));
112 A_int = At_w(end);
113 % A_int = 1;
114
115 % Control volumes [m3]
116 Vcv_w = Ac_hs(1:nz_w)*L/nx;                 % Under water level
117 Vcv_o = Ac_hs(nz_w+1:nz_w+nz_o-1)*L/nx;     % Over water level under weir
118 Vcv_top = Ac_hs(end)*L/nx;                  % Over weir
119
120 %% Coalescence of droplets to continuous phases:
121 i = 1:nx-2;
122 rhos_w = [rho1_w(i,1),rho2_w(i,1),rho3_w(i,1),rho4_w(i,1),rho5_w(i,1),...
123     rho6_w(i,1),rho7_w(i,1),rho8_w(i,1),rho9_w(i,1),rho10_w(i,1)]';
124 rhos_o =[rho1_o(i,end),rho2_o(i,end),rho3_o(i,end),rho4_o(i,end),...
125         rho5_o(i,end),rho6_o(i,end),rho7_o(i,end),rho8_o(i,end),...
126         rho9_o(i,end),rho10_o(i,end)]';
127
128 rateVec_water = (k_int_w*A_int*WC_o(i,1))';
129 rateVec_oil = (k_int_o*A_int*(1-WC_w(i,end)))';
130 rateMat_water = repmat(rateVec_water,10,1);
131 rateMat_oil = repmat(rateVec_oil,10,1);
132
133 m_coal_water = rhos_w.*rateMat_water; % Rate of mass, water to water phase
134 m_coal_oil   = rhos_o.*rateMat_oil;   % Rate of mass, oil to oil phase
135
136 %% Flow calculations:
137 % Interfacial volumetric flows [m3/s]:
138 F_int = (sum(m_coal_water)/rho_w - sum(m_coal_oil)/rho_o)';
139 Fint_sum = tril(ones(nx-2))'*F_int; % Reverse cumulative sum of F_int
140 F_int_tot = sum(F_int);             % Total flow
141
142 % Horizontal flows through separator [m3/s]:
143 q_out = [ F(3)*Ac_hs(1:nz_w)/A_lw;
144           F(2)*Ac_hs(nz_w+1:nz_w+nz_o)/(A_l-A_lw)];
145 q_waterphase = repmat(q_out(1:nz_w),1,7) - (Ac_hs(1:nz_w)/A_lw)*Fint_sum';
146 q_oilphase = repmat(q_out(nz_w+1:nz_w+nz_o),1,7) + ...
147               (Ac_hs(nz_w+1:nz_w+nz_o)/(A_l-A_lw))*Fint_sum';
```

```
148 q = [q_waterphase; q_oilphase]';
149 q = [q;q_out'];
150 % q gives the horizontal feed to each control volume left of the weir
151
152 % Change to real inlet flows:
153 q(1,1:nz_w)=Fin_wp*Ac_hs(1:nz_w)/A_lw;
154 q(1,nz_w+1:nz_w+nz_o) = Fin_op*Ac_hs(nz_w+1:nz_w+nz_o)/(A_l-A_lw);
155
156 %% Level calculations:
157 Vl_dot = Fin_l-F(2)-F(3);
158 hl_dot = Vl_dot/(2*L*sqrt(h_l*(2*r-h_l)));
159
160 Vlw_dot = Fin_wp + F_int_tot - F(3);
161 hlw_dot = Vlw_dot/(2*L*(nx-1)/nx*sqrt(h_lw*(2*r-h_lw)));
162
163 % Change in level result in changed control volumes:
164 j = (1:nz_w)';              % Water phase
165     dAdt_water = 2*sqrt(hs(j).*(2*r-hs(j))).*(j)/nz_w*hlw_dot;
166 j = (nz_w+1:nz_w+nz_o-1)';  % Oil phase:
167     dAdt_oil = 2*sqrt(hs(j).*(2*r-hs(j))).*(1-(j-nz_w)/(nz_o-1))*hlw_dot;
168 dAdt = [dAdt_water; dAdt_oil];
169 dVcvdt = L/nx*(dAdt-[0; dAdt(1:end-1)]);
170
171 %% Gas pressure calculations:
172 m_g_in = Fin_g*rho_g;
173 n_g_in = m_g_in/Mm_g;                    % Molar rate of gas in  [mol/s]
174 n_g_out = P*1e5/(R_gas*T)*F(1);          % Molar rate of gas out [mol/s]
175 V_l = L*A_l;                             % Total liquid volume   [m3]
176 V_g = pi*r^2*L-V_l;                      % Total gas volume      [m3]
177 P_dot = (R_gas*T/V_g*(n_g_in-n_g_out) + P*1e5/V_g*(Fin_l-F(2)-F(3)))*1e-5;
178
179 %% Density calculations:
180 v_stokes = -stokesVelocity(rho_o,rho_w,mu_w,d_A);
181
182 % % % % Main separator part, water phase
183 for j = 1:nz_w % Under water level:
184 for i = 1:nx-1 % Left of weir:
185 rhos = [rho1_o(i,j),rho2_o(i,j),rho3_o(i,j),rho4_o(i,j),rho5_o(i,j),...
```

```
186         rho6_o(i,j),rho7_o(i,j),rho8_o(i,j),rho9_o(i,j),rho10_o(i,j)]';
187 R = rateOfFormation(rhos,k);
188 v_adp_in = v_stokes*((WC_w(i,j)-0.7)/0.3)^n_adp;
189
190 if j == 1           % At bottom
191 rhos_Over=[ rho1_o(i,j+1),rho2_o(i,j+1),rho3_o(i,j+1),rho4_o(i,j+1),...
192             rho5_o(i,j+1),rho6_o(i,j+1),rho7_o(i,j+1),rho8_o(i,j+1),...
193             rho9_o(i,j+1),rho10_o(i,j+1)]';
194 v_adp_out = v_stokes*((WC_w(i,j+1)-0.7)/0.3)^n_adp;
195 elseif j == nz_w    % At interface
196 rhos_Under=[rho1_o(i,j-1),rho2_o(i,j-1),rho3_o(i,j-1),rho4_o(i,j-1),...
197             rho5_o(i,j-1),rho6_o(i,j-1),rho7_o(i,j-1),rho8_o(i,j-1),...
198             rho9_o(i,j-1),rho10_o(i,j-1)]';
199 else
200 rhos_Over=[rho1_o(i,j+1),rho2_o(i,j+1),rho3_o(i,j+1),rho4_o(i,j+1),...
201             rho5_o(i,j+1),rho6_o(i,j+1),rho7_o(i,j+1),rho8_o(i,j+1),...
202             rho9_o(i,j+1),rho10_o(i,j+1)]';
203 rhos_Under=[rho1_o(i,j-1),rho2_o(i,j-1),rho3_o(i,j-1),rho4_o(i,j-1),...
204             rho5_o(i,j-1),rho6_o(i,j-1),rho7_o(i,j-1),rho8_o(i,j-1),...
205             rho9_o(i,j-1),rho10_o(i,j-1)]';
206 v_adp_out = v_stokes*((WC_w(i,j+1)-0.7)/0.3)^n_adp;
207 end%if
208
209 if i == 1  % At separator inlet
210   rhos_Left = rhosFeed_waterPhase;
211 else
212   rhos_Left=[rho1_o(i-1,j),rho2_o(i-1,j),rho3_o(i-1,j),rho4_o(i-1,j),...
213             rho5_o(i-1,j),rho6_o(i-1,j),rho7_o(i-1,j),rho8_o(i-1,j),...
214             rho9_o(i-1,j),rho10_o(i-1,j)]';
215 end%if
216
217 if i == nx-1 % at weir: (vertical convective flow)
218   if j == nz_w % At interface
219    rho_dots=(rhos_Left-rhos)*q_out(j)/Vcv_w(j) +R -rhos*dVcvdt(j)/Vcv_w(j);
220   else
221    rho_dots=(rhos_Left*q_out(j)+rhos_Over*sum(q_out(j+1:nz_w)))/Vcv_w(j)...
222             - rhos*sum(q_out(j:nz_w))/Vcv_w(j) +R -rhos*dVcvdt(j)/Vcv_w(j);
223   end%if
```

```
224 else % main part of separator
225    rhos_Right=[ rho1_o(i+1,j),rho2_o(i+1,j),rho3_o(i+1,j),rho4_o(i+1,j),...
226                 rho5_o(i+1,j),rho6_o(i+1,j),rho7_o(i+1,j),rho8_o(i+1,j),...
227                 rho9_o(i+1,j),rho10_o(i+1,j)]';
228    conv_trans = rhos_Left*max(q(i,j),0)  - rhos*max(q(i+1,j),0) ...
229               + rhos_Right*max(-q(i,j),0)- rhos*max(-q(i+1,j),0);
230   if j == nz_w % At interface
231  rho_dots=  (                     conv_trans              )/Vcv_w(j)...
232     + ( rhos_Under.*v_adp_in*At_w(j-1) - m_coal_oil(:,i)  )/Vcv_w(j)...
233     + (                     R*Vcv_w(j)  - rhos*dVcvdt(j)  )/Vcv_w(j);
234   elseif j == 1 % At bottom
235  rho_dots=   (                         conv_trans          )/Vcv_w(j)...
236             +(                   - rhos.*v_adp_out*At_w(j) )/Vcv_w(j)...
237             +(     R*Vcv_w(j)    - rhos*dVcvdt(j)          )/Vcv_w(j);
238   else
239  rho_dots=   (     rhos_Left*q(i,j) - rhos*q(i+1,j)          )/Vcv_w(j)...
240    +(rhos_Under.*v_adp_in*At_w(j-1) - rhos.*v_adp_out*At_w(j))/Vcv_w(j)...
241             +(         R*Vcv_w(j)    - rhos*dVcvdt(j)        )/Vcv_w(j);
242   end%if
243 end%if
244 rho1_dot_o(i,j) = rho_dots(1);
245 rho2_dot_o(i,j) = rho_dots(2);
246 rho3_dot_o(i,j) = rho_dots(3);
247 rho4_dot_o(i,j) = rho_dots(4);
248 rho5_dot_o(i,j) = rho_dots(5);
249 rho6_dot_o(i,j) = rho_dots(6);
250 rho7_dot_o(i,j) = rho_dots(7);
251 rho8_dot_o(i,j) = rho_dots(8);
252 rho9_dot_o(i,j) = rho_dots(9);
253 rho10_dot_o(i,j) = rho_dots(10);
254 end
255 end
256
257 % % % % Main separator part, Oil phase
258 v_stokes = stokesVelocity(rho_w,rho_o,mu_o,d_A);
259 for j = 1:nz_o-1 % Under weir:
260 for i = 1:nx-1 % Left of weir:
261 rhos = [rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
```

```
262         rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
263 rhos_Over=[ rho1_w(i,j+1),rho2_w(i,j+1),rho3_w(i,j+1),rho4_w(i,j+1),...
264             rho5_w(i,j+1),rho6_w(i,j+1),rho7_w(i,j+1),rho8_w(i,j+1),...
265             rho9_w(i,j+1),rho10_w(i,j+1)]';
266 R = rateOfFormation(rhos,k);
267 v_adp_in = v_stokes*((0.7-WC_o(i,j))/0.7)^n_adp;
268
269 if i == 1
270   rhos_Left = rhosFeed_oilPhase;
271 else
272 rhos_Left=[ rho1_w(i-1,j),rho2_w(i-1,j),rho3_w(i-1,j),rho4_w(i-1,j),...
273             rho5_w(i-1,j),rho6_w(i-1,j),rho7_w(i-1,j),rho8_w(i-1,j),...
274             rho9_w(i-1,j),rho10_w(i-1,j)]';
275 end%if
276
277 if i == nx-1 % at weir: (vertical convective flow)
278   if j == 1 % convective vertical flow.
279   rho_dots =( (rhos_Left - rhos)*q_out(nz_w+j))/Vcv_o(j) ...
280                + R - rhos*dVcvdt(nz_w+j)/Vcv_o(j);
281   else
282 rhos_under=[rho1_w(i,j-1),rho2_w(i,j-1),rho3_w(i,j-1),rho4_w(i,j-1),...
283             rho5_w(i,j-1),rho6_w(i,j-1),rho7_w(i,j-1),rho8_w(i,j-1),...
284             rho9_w(i,j-1),rho10_w(i,j-1)]';
285 rho_dots=    (    rhos_Left*q_out(nz_w+j) ...
286                     +rhos_under*sum(q_out(nz_w+1:nz_w+j-1))  )/Vcv_o(j) ...
287           -rhos*sum(q_out(nz_w+1:nz_w+j))/Vcv_o(j) ...
288           + R - rhos*dVcvdt(nz_w+j)/Vcv_o(j);
289   end%if
290 else % main part of separator
291   if j == 1 % At interface
292   rho_dots = (   rhos_Left*q(i,nz_w+j) - rhos*q(i+1,nz_w+j)  )/Vcv_o(j)...
293       + (  rhos_Over.*v_adp_in*At_o(j) - m_coal_water(:,i)   )/Vcv_o(j) ...
294            + (              R*Vcv_o(j) - rhos*dVcvdt(nz_w+j) )/Vcv_o(j);
295   else
296   v_adp_out = v_stokes*((0.7-WC_o(i,j-1))/0.7)^n_adp;
297   rho_dots = ( rhos_Left*q(i,nz_w+j) - rhos*q(i+1,nz_w+j)      )/Vcv_o(j)...
298     +(rhos_Over.*v_adp_in*At_o(j) - rhos.*v_adp_out*At_o(j-1) )/Vcv_o(j)...
299            + (            R*Vcv_o(j) - rhos*dVcvdt(nz_w+j)     )/Vcv_o(j);
```

```
300   end%if
301 end%if
302 rho1_dot_w(i,j) = rho_dots(1);
303 rho2_dot_w(i,j) = rho_dots(2);
304 rho3_dot_w(i,j) = rho_dots(3);
305 rho4_dot_w(i,j) = rho_dots(4);
306 rho5_dot_w(i,j) = rho_dots(5);
307 rho6_dot_w(i,j) = rho_dots(6);
308 rho7_dot_w(i,j) = rho_dots(7);
309 rho8_dot_w(i,j) = rho_dots(8);
310 rho9_dot_w(i,j) = rho_dots(9);
311 rho10_dot_w(i,j) = rho_dots(10);
312 end
313 end
314
315 % % % % Top left
316 j = nz_o;         % Over weir:
317 for i = 1:nx-1 % Left of weir:
318 rhos = [rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
319         rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
320 R = rateOfFormation(rhos,k);
321 v_adp_out = v_stokes*((0.7-WC_o(i,j-1))/0.7)^n_adp;
322
323 if i == 1
324   rhos_Left = rhosFeed_oilPhase;
325 else
326 rhos_Left=[ rho1_w(i-1,j),rho2_w(i-1,j),rho3_w(i-1,j),rho4_w(i-1,j),...
327             rho5_w(i-1,j),rho6_w(i-1,j),rho7_w(i-1,j),rho8_w(i-1,j),...
328             rho9_w(i-1,j),rho10_w(i-1,j)]';
329 end%if
330 if i==nx-1
331 rhos_under=[rho1_w(i,j-1),rho2_w(i,j-1),rho3_w(i,j-1),rho4_w(i,j-1),...
332             rho5_w(i,j-1),rho6_w(i,j-1),rho7_w(i,j-1),rho8_w(i,j-1),...
333             rho9_w(i,j-1),rho10_w(i,j-1)]';
334 rho_dots = (  rhos_Left*q_out(end)...
335             +rhos_under*sum(q_out(nz_w+1:end-1))  )/Vcv_top...
336             - rhos*F(2)/Vcv_top + R - rhos/(Vcv_top*nx)*Vl_dot;
337 else
```

```
338     rho_dots =(rhos_Left*q(i,nz_w+j)-rhos*q(i+1,nz_w+j))/Vcv_top ...
339         -(rhos.*v_adp_out)*At_o(j-1)/Vcv_top + R - rhos/(Vcv_top*nx)*Vl_dot;
340 end%if
341 rho1_dot_w(i,j) = rho_dots(1);
342 rho2_dot_w(i,j) = rho_dots(2);
343 rho3_dot_w(i,j) = rho_dots(3);
344 rho4_dot_w(i,j) = rho_dots(4);
345 rho5_dot_w(i,j) = rho_dots(5);
346 rho6_dot_w(i,j) = rho_dots(6);
347 rho7_dot_w(i,j) = rho_dots(7);
348 rho8_dot_w(i,j) = rho_dots(8);
349 rho9_dot_w(i,j) = rho_dots(9);
350 rho10_dot_w(i,j) = rho_dots(10);
351 end%for
352
353 % % % % Top Right
354 j = nz_o;    % Over weir:
355 i = nx;      % Right of weir:
356 rhos = [rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
357     rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
358 rhos_Left = [rho1_w(i-1,j),rho2_w(i-1,j),rho3_w(i-1,j),rho4_w(i-1,j),...
359              rho5_w(i-1,j),rho6_w(i-1,j),rho7_w(i-1,j),rho8_w(i-1,j),...
360              rho9_w(i-1,j),rho10_w(i-1,j)]';
361 R = rateOfFormation(rhos,k);
362 rho_dots = (rhos_Left-rhos)*F(2)/Vcv_top + R - rhos/(Vcv_top*nx)*Vl_dot;
363 rho1_dot_w(i,j) = rho_dots(1);
364 rho2_dot_w(i,j) = rho_dots(2);
365 rho3_dot_w(i,j) = rho_dots(3);
366 rho4_dot_w(i,j) = rho_dots(4);
367 rho5_dot_w(i,j) = rho_dots(5);
368 rho6_dot_w(i,j) = rho_dots(6);
369 rho7_dot_w(i,j) = rho_dots(7);
370 rho8_dot_w(i,j) = rho_dots(8);
371 rho9_dot_w(i,j) = rho_dots(9);
372 rho10_dot_w(i,j) = rho_dots(10);
373
374 % % % % Oil out :
375 i = nx;              % Right of Weir
```

```
376 for j = 1:nz_o-1    % Under Weir, over interface
377 rhos = [rho1_w(i,j),rho2_w(i,j),rho3_w(i,j),rho4_w(i,j),rho5_w(i,j),...
378     rho6_w(i,j),rho7_w(i,j),rho8_w(i,j),rho9_w(i,j),rho10_w(i,j)]';
379 rhos_Over =[rho1_w(i,j+1),rho2_w(i,j+1),rho3_w(i,j+1),rho4_w(i,j+1),...
380             rho5_w(i,j+1),rho6_w(i,j+1),rho7_w(i,j+1),rho8_w(i,j+1),...
381             rho9_w(i,j+1),rho10_w(i,j+1)]';
382 R = rateOfFormation(rhos,k);
383
384 rho_dots = (rhos_Over-rhos)*F(2)/Vcv_o(j) +R -rhos*dVcvdt(nz_w+j)/Vcv_o(j);
385 rho1_dot_w(i,j) = rho_dots(1);
386 rho2_dot_w(i,j) = rho_dots(2);
387 rho3_dot_w(i,j) = rho_dots(3);
388 rho4_dot_w(i,j) = rho_dots(4);
389 rho5_dot_w(i,j) = rho_dots(5);
390 rho6_dot_w(i,j) = rho_dots(6);
391 rho7_dot_w(i,j) = rho_dots(7);
392 rho8_dot_w(i,j) = rho_dots(8);
393 rho9_dot_w(i,j) = rho_dots(9);
394 rho10_dot_w(i,j) = rho_dots(10);
395 end%for
396
397 for j = 1:nz_w  % Under interface, all rhos are water!!!!!!!
398 rhos = [rho1_o(i,j),rho2_o(i,j),rho3_o(i,j),rho4_o(i,j),rho5_o(i,j),...
399         rho6_o(i,j),rho7_o(i,j),rho8_o(i,j),rho9_o(i,j),rho10_o(i,j)]';
400  if j == nz_w
401  rhos_Over=[rho1_w(i,1),rho2_w(i,1),rho3_w(i,1),rho4_w(i,1),rho5_w(i,1),...
402             rho6_w(i,1),rho7_w(i,1),rho8_w(i,1),rho9_w(i,1),rho10_w(i,1)]';
403  else
404  rhos_Over=[rho1_o(i,j+1),rho2_o(i,j+1),rho3_o(i,j+1),rho4_o(i,j+1),...
405             rho5_o(i,j+1),rho6_o(i,j+1),rho7_o(i,j+1),rho8_o(i,j+1),...
406             rho9_o(i,j+1),rho10_o(i,j+1)]';
407  end%if
408 R = rateOfFormation(rhos,k);
409 rho_dots = (rhos_Over-rhos)*F(2)/Vcv_w(j) + R - rhos*dVcvdt(j)/Vcv_w(j);
410 rho1_dot_o(i,j) = rho_dots(1);
411 rho2_dot_o(i,j) = rho_dots(2);
412 rho3_dot_o(i,j) = rho_dots(3);
413 rho4_dot_o(i,j) = rho_dots(4);
```

```
414 rho5_dot_o(i,j) = rho_dots(5);
415 rho6_dot_o(i,j) = rho_dots(6);
416 rho7_dot_o(i,j) = rho_dots(7);
417 rho8_dot_o(i,j) = rho_dots(8);
418 rho9_dot_o(i,j) = rho_dots(9);
419 rho10_dot_o(i,j) = rho_dots(10);
420 end%for
421
422 %% Check if mass balance is held:
423 % Water Balance:
424 WaterIn  = q(2,:)*([WC_w(1,:),WC_o(1,:)]');
425 WaterOut = q(8,:)*([WC_w(7,:),WC_o(7,:)]');
426 WaterInOilProduct   = WC_w(end,1)*F(2);
427 WaterInWaterProduct = WC_w(end-1,1)*F(3);
428 WaterInFeed = e_in*Fin_l;
429 % mb_w = WaterOut - WaterIn;                                % Local balance
430 mb_w = WaterInOilProduct + WaterInWaterProduct -WaterInFeed;% Total balance
431
432 % Oil Balance:
433 OilIn  = q(2,:)*(1-[WC_w(1,:),WC_o(1,:)]');
434 OilOut = q(8,:)*(1-[WC_w(7,:),WC_o(7,:)]');
435 OilInOilProduct     = (1-WC_w(end,1))*F(2);
436 OilInWaterProduct   = (1-WC_w(end-1,1))*F(3);
437 OilInFeed  = (1-e_in)*Fin_l;
438 % mb_o = OilOut - OilIn;                                % Local Balance
439 mb_o = OilInOilProduct + OilInWaterProduct - OilInFeed; % Total Balance
440
441 %% Water product:
442 ppm_oiw = (1-WC_w(end-1,1))*1e6;
443 rho_oiw = [ rho1_o(end-1,1);rho2_o(end-1,1);rho3_o(end-1,1);...
444             rho4_o(end-1,1);rho5_o(end-1,1);rho6_o(end-1,1);...
445             rho7_o(end-1,1);rho8_o(end-1,1);rho9_o(end-1,1);...
446             rho10_o(end-1,1)];
447 %% Oil product:
448 WC_wio = WC_w(end,1);
449 rho_wio = [ rho1_o(end,1);rho2_o(end,1);rho3_o(end,1);rho4_o(end,1);...
450             rho5_o(end,1);rho6_o(end,1);rho7_o(end,1);rho8_o(end,1);...
451             rho9_o(end,1);rho10_o(end,1)];
```

```
452
453 %% Model function returns and problem solver:
454 dots = [
455     reshape(rho1_dot_w,nx*nz_o,1);
456     reshape(rho2_dot_w,nx*nz_o,1);
457     reshape(rho3_dot_w,nx*nz_o,1);
458     reshape(rho4_dot_w,nx*nz_o,1);
459     reshape(rho5_dot_w,nx*nz_o,1);
460     reshape(rho6_dot_w,nx*nz_o,1);
461     reshape(rho7_dot_w,nx*nz_o,1);
462     reshape(rho8_dot_w,nx*nz_o,1);
463     reshape(rho9_dot_w,nx*nz_o,1);
464     reshape(rho10_dot_w,nx*nz_o,1);
465     reshape(rho1_dot_o,nx*nz_w,1);
466     reshape(rho2_dot_o,nx*nz_w,1);
467     reshape(rho3_dot_o,nx*nz_w,1);
468     reshape(rho4_dot_o,nx*nz_w,1);
469     reshape(rho5_dot_o,nx*nz_w,1);
470     reshape(rho6_dot_o,nx*nz_w,1);
471     reshape(rho7_dot_o,nx*nz_w,1);
472     reshape(rho8_dot_o,nx*nz_w,1);
473     reshape(rho9_dot_o,nx*nz_w,1);
474     reshape(rho10_dot_o,nx*nz_w,1);
475     hl_dot;
476     hlw_dot;
477     P_dot  ];  % 453
478 outputs = [WC_wio; ppm_oiw; rho_wio; rho_oiw; mb_o; mb_w];%1+1+10+10+1+1=24
479 dx_and_y = [dots; outputs];
480 end%function
```

C.7 Changes from GS1 to GS2

The part of the GS2 model given under corresponds to line 77-93 in GS1_model_sim.m above. The GS2 model does not calculate the pressure on line 171-177 in GS1_model_sim.m.

```
%% Feed calculations
Fin_ws = (rhos_in/rho_w)*Fin_l; % volumetric flows of dispersed water in feed
Fin_w2wp = sum(Fin_ws*g_w_2);
Fin_w2op = (1-g_w_2)*Fin_ws;

Fin_w_tot = sum(Fin_ws);
Fin_o_tot = Fin_l-Fin_w_tot;
Fin_o2op = g_o_2*Fin_o_tot;
Fin_o2wp = (1-g_o_2)*Fin_o_tot;

Fin_wp = Fin_o2wp + Fin_w2wp;
Fin_op = Fin_o2op + sum(Fin_w2op);

rhosFeed_oilPhase = Fin_w2op*rho_w/Fin_op;
rhosFeed_waterPhase = [Fin_o2wp*rho_o/Fin_wp; zeros(9,1)];
```

Appendix D

Simulink Diagrams

The Simulink diagram of the combined bulk separation and produced water treatment is shown in Figure D.2. The monitors shows the nominal case.

The bulk separation consist of the two models, GS1 and GS2, and their respective controllers C1 and C2. The controllers are feedback PI controllers. The level controllers in C1 and C2 was implemented with anti-windup based on clamping [14]. The controller block in Simulink did not work with this option and they was modified to fix the problem. The modified controller block is shown in Figure D.1.

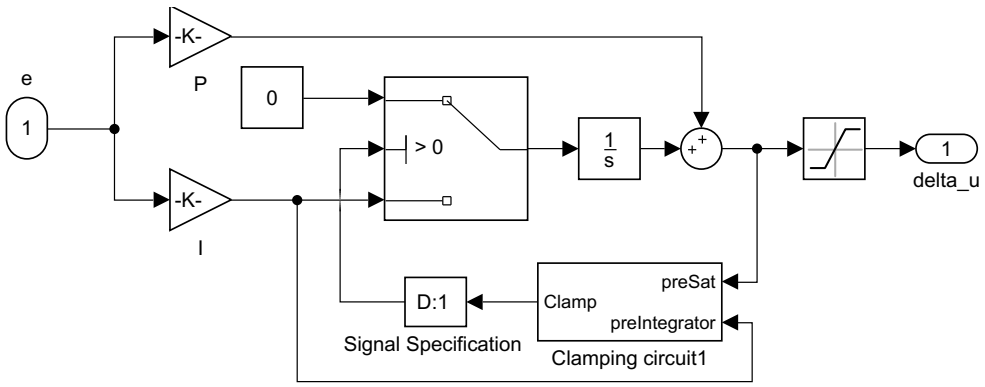


Figure D.1: Modified PI controller block with working Clamping. The controller did not work because of dimension problems. The problem was fixed by introducing the Signal Specification block. The rest of the controller and the Clamping box is the same as the default Simulink PI controller with clamping, which was not working [14].

The two flows of produced water from GS1 and GS2 are then combined and sent to the produced water treatment. The HC and CFU blocks simulate three hydrocyclones in parallel and four compact flotation units in parallel. The controllers C3 and C4 are PI controllers.

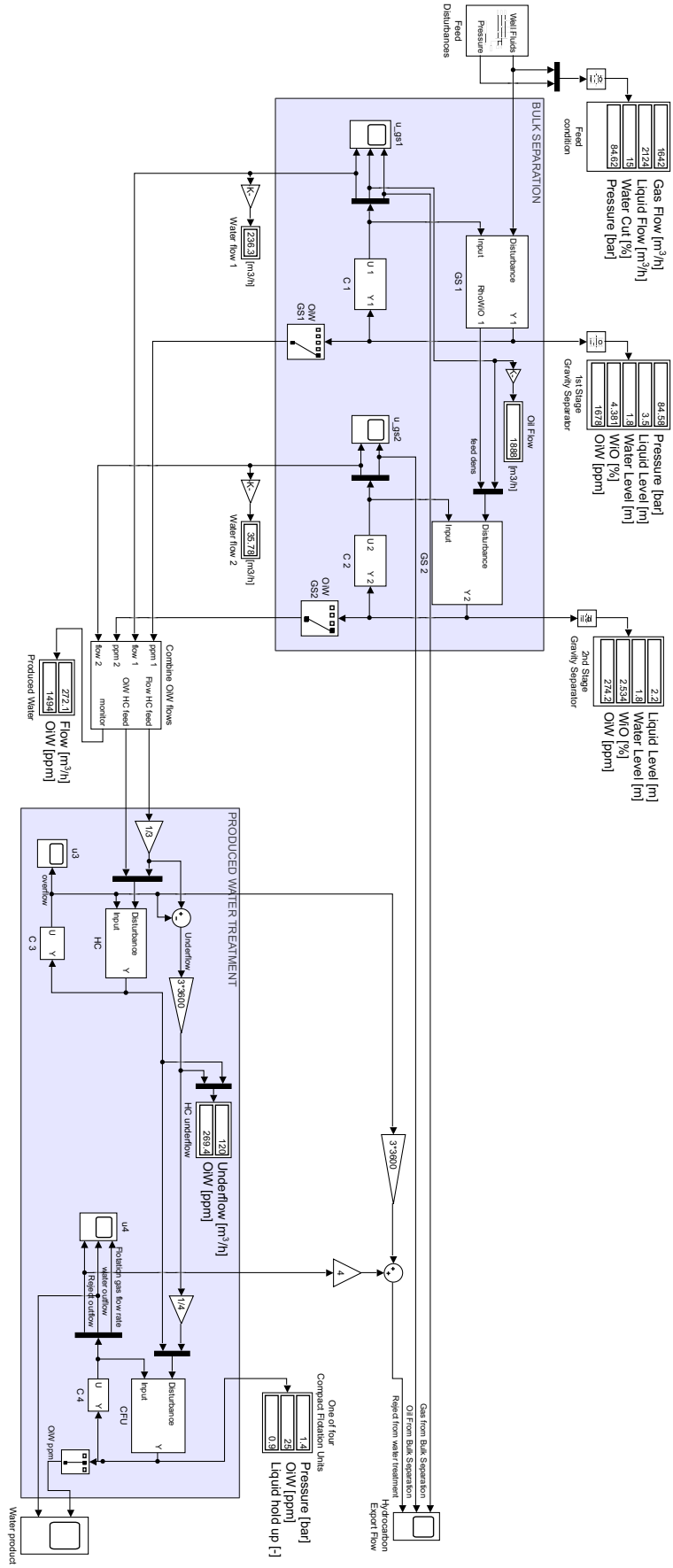


Figure D.2: Developed Simulink diagram of subsea separation system. The displayed values are the nominal values.