Problem Description

With global increased energy demands, the continued depletion of fossil fuels and problems associated with global warming and emissions, it is necessary to focus research on development of renewable fuel alternatives. Biodiesel is a carbon-neutral alternative which can be used in most automotive engines without need for modification. Reactive distillation is an integrated reactive and separation unit which offers several potential benefits; less waste processing, higher yield of product, better separation and large savings on both OPEX and CAPEX. Combination of these two prospects could allow for a clean, efficient and economically superior technology.

There have been several publications on control structures for reactive distillation, but almost all of them analyse the problem from a controller performance point of view. In this project we propose to investigate the plantwide control problem with focus on the economics. The first part of the systematic top-down/bottom-up procedure of Skogestad will be used to optimise the process and identify regions of active constraints.

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

Emilie Ø. Houge, Trondheim

Emilie Øritsland Houge

Preface

This thesis is a compulsory part of the master degree in Chemical Engineering from the Norwegian University of Technology and Science (NTNU). The work on the thesis has allowed me to further explore the aspect of renewable energy, while also giving me the chance to work with modelling and programming. I am very grateful to have had this opportunity, and I am certain that what I have learned during this working period will benefit me later.

For the completion of my master's thesis I would like to thank my supervisor professor Sigurd Skogestad of the system engineering group at NTNU for his valuable help and insight with completing this task. My co-supervisor PhD-student Chriss Grimholt has also been of great assistance with frequent check-ups, bug-searching and cheerful consultations. Professor Amornchai Arpornwichanop also deserves a thank you for providing the modelling file from his similar research. Having the file available helped a lot in validation of the model.

I would also like to thank my good friends and classmates at the office for keeping spirits high, coffee available and providing valuable discussion/venting partners. I would especially like to thank my good friend Nicola Stewart for keeping me calm and for her support and help when my computer crashed 1 week prior to the deadline.

A big thank you is also in order for my boyfriend and family for all the support and guidance they have given me throughout my entire education, including this last half-year. They assisted in the proof-reading of this report even though my mother claimed that she hardly understood a word.

Lastly, thank you for taking the time to read through my thesis, I hope you enjoy!

Abstract

This thesis modifies an already existing dynamic Matlab Simulink model to correlate with a Hysys model from a similar research [1]. The models are concerned with the reactive distillation for production of biodiesel from linoleic esters of soybean oil through transesterification reactions. The Matlab model was constructed using molar balances, Francis' weir equation for the liquid dynamics, no vapour dynamics, kinetics according to the rate law and temperature estimations according to the UNIQUAC method. Very similar results were presented for the two models besides the Matlab being of simpler origin, and having no inclusion of energy balances compared to the Hysys model. Both a dynamic Simulink and a DAE-system Matlab model were produced, with the DAE system offering reduced simulation time and an additional steady-state version.

The steady-state version of the DAE-model was used for optimisation according to the Skogestad Economic Plantwide Optimisation principles. The first part of the systematic top-down/bottom-up method was applied, focusing on the economic control. Four distinct active constraint regions were identified; varying with the feed flowrate and molar ratio. The first region could not be solved with regard to quality constraints, while the second had maximum flowrate and maximum impurity of total glycerol in the biodiesel product as the active constraints. The third region had active constraints of maximum temperature in the reboiler and maximum impurity of glycerol. The fourth region would not be solved by the Matlab fmincon interior-point algorithm, and applies very high feed ratios outside of the range normally applied by biodiesel production processes. The optimal point for operation was found to be on the boundary of region 3 and 4, with a feed of 333:17 kmol/h of Methanol:TG. The corresponding setpoints for the liquid reflux, vapour boilup, distillate and bottoms molar flowrates per minute equal 1.12, 5.93, 4.43 and 1.40 respectively.

List of Contents

1 Introduction

This section will provide a brief introduction to the master thesis; the motivation behind it, and an overview of the work performed.

1.1 Motivation

The global energy consumption is expected to grow by 47% by 2035 [2]. Fossil fuels dominate the energy supplies by 87% while renewable energy sources only contribute with 2% to the global energy market [3]. However, liquid fuels besides petroleum are expected to triple by 2035 largely due to the projected increase in fossil fuel prices in the coming years [2]. Increasing energy demands coupled with incentives to avoid greenhouse gas emissions have contributed in making renewable energy the fastest growing energy source, expected to increase by an average of 3% per year from 2010 to 2035 [3].

Biodiesel is made from plant oils, animal fats or even waste cooking oils through transesterification and esterification reactions with alcohols. It constitutes a renewable, biodegradable and carbon neutral alternative to petroleum diesels, and can be used either on its own or in blends with regular diesel. Biodiesel can be used on most automotive engines without any modifications and can actually contribute to improving the engines' general performance [4].

The traditional process for production of biodiesel is through reactors followed by separators. However, in recent years more studies have been focusing on the prospect of using integrated reactive separation technologies in the manufacturing of biodiesel [5]. Reactive distillation is the merging of a reactor and a distillation column in one, and can offer several benefits, such as: increased yield, avoidance of azeotropes, reduced capital costs of the plant (CAPEX), more effective separation and reduced need for solvents [6].

However, reactive distillation often showcases complex behaviour such as steady state multiplicity, process gain sign changes and strong interactions between process variables [7]. A consequence of this is that reactive distillation processes are often non-linear in nature. This presents an interesting challenge when it comes to the construction of a working control structure. Although an increasing number of papers on reactive distillation columns have been published in the later years [5], there is still a gap in the market regarding a systematic plant-wide control procedure with focus on optimisation [7].

1.2   Scope of work

For my specialization project last semester I produced a dynamic Simulink model as well as a DAE-system in Matlab representing the production of biodiesel through reactive distillation. These models were based largely on the work of Simasatitkul et al [1]. After being in contact with Professor Amornchai Arpornwichanop of Chulalongkorn University in Thailand, the steady-state Hysys model of the reactive distillation column used in their research was made available to me.

The work of this thesis is therefore divided into two main segments; the first being a modification of the Matlab models with regard to the Hysys model obtained, while the second part is the economic plantwide optimisation of the now modified Matlab models. The tasks surrounding the two main segments are roughly summarised below:

1. Modification of the Simulink and Matlab model to better correlate with the Hysys model.
    1.1. Identification of differences between the models, and implementation of mild modifications.
    1.2. Testing of several temperature-estimation methods.
    1.3. Implementation of an appropriate temperature estimation method in the Simulink and DAE model.
2. Optimisation of the model with regard to relevant disturbances using the first part of the Skogestad top-down/bottom-up method.
    2.1. Formulation of a cost function and constraints.
    2.2. Optimization with regard to relevant disturbances and identification of active constraint regions.

The thesis has the following set-up: A general theory section on biodiesel production and reactive distillation, then Part 1 is covered, followed by Part 2 and then a common discussion and conclusion chapter. Part 1 consists of a presentation of the models and results produced during the specialisation project as well as documentations of tasks 1.1-1.3. Part 2 starts off with some general theory on economic plantwide control and is followed by the results of tasks 2.1-2.2.

2 Background

This background section is meant to provide an introduction to biodiesel and reactive distillation; the current state of the technologies and the challenges associated with them. The background is not meant to cover these subjects completely, but to provide sufficient information to understand the modelling and design decisions made, the results obtained in the first part of this report and suggestions for further work and modifications.

2.1 Biodiesel

Biodiesel is vegetable or plant-based oils that can be used as fuels, and normally consists of long chain alkyl esters. The first use of vegetable oils as fuels was demonstrated by Rudolph Diesel, the inventor of the diesel engine in the start of the 20th century. He was of the opinion that vegetable oils were the fuel of the future [4]. However, because of the cheap fossil fuel prices, biodiesel was not considered a viable alternative until recently, when fossil fuel prices have increased and are estimated to keep rising [2]. Biodiesel is promising in that it represents an environmentally friendly alternative or additive to regular fossil fuels that can be used on present engines with no or little modification. Biodiesel can also improve the performance of the engine and even prolong the engines' life as it showcases both increased solvent effect and lubrication properties [4].

Environmentally Friendly Fuel
Using biodiesel as a fuel significantly reduces harmful emissions to the atmosphere. The emissions of sulphur dioxide are reduced by 100%, while emissions of carbon monoxide, particulates and unburned hydrocarbons are reduced by 48%, 47% and 67% respectively. Also for blends of biodiesel with regular petroleum diesel, the biodiesel will improve the overall biodegradability and promote reduction of harmful emissions from the fuel [4].

One of the most important greenhouse gases is CO_2, and also here biodiesel offers an advantage: biodiesel derived from vegetable oils can be considered carbon neutral as the plant attains its carbon through absorption of carbon dioxide from the atmosphere during its lifetime. The carbon released during burning should thus be equal to the carbon previously absorbed by the plant and no excess carbon dioxide would be emitted. In total, the emissions of carbon dioxide from biodiesel are reduced by 78% from regular diesel fuels [4].

The only environmental issue with applying biodiesel as a fuel is the potential increase in NO_x emissions from fossil fuels by 6-9%. However, on-going research is exploring the possibilities of reducing these emissions with regard to

changes in the injection times and temperatures of the diesel engine. The difference in formation of NO_x during combustion is most likely due to differences in the structure of the fuels, as biodiesel have a larger degree of unsaturation compared to regular fuels [4].

Challenges
One of the challenges associated with employing biodiesel as a fuel is that the source of the biodiesel in some cases often is edible and thus competes with markets for producing food for human or livestock use. Also, there is a problem with producing enough biodiesel to be able to satisfy the demands. There simply isn't enough crop space for producing biodiesel with present technology using common raw materials such as soybean oil or palm oil. To give an idea of the areas required; if all of the fats and oils produced in the US were converted to biodiesel, it would only cover 8% of the market for diesel worldwide [4].

Research has been exploring the possibilities of employing alternative sources of biodiesel with larger oil to surface area ratios required such as algae [4], or by utilising waste oils or non-edible vegetable oils that are not a source of food [5].

Soybean Oil
The number one source of biodiesel today is soybean oil (US) and palm oil [8]. Soybean oil contains a favourable energy ratio of 3 energy units produced to each energy unit consumed. However, soybean crops need large surface areas in order to grow, and hence an acre (4046 m^2) of soybean crops yields only 50 gallons (190 litres) per year [4].

The soybean consists of approximately 21% oils, whereas 94.4% is triacylglycerol and 3.7% is phospholipids. The amount of free fatty acids in the oil can vary with the age and soundness of the beans [8], but is usually quite low (less than 0.3%) [4]. The composition of the average soybean methyl ester is given in Table 2-1 below.

Table 2-1: The physical properties of the fatty acid esters of soybean oil.

Methyl ester	Formula [9]	Average composition [8]	Boiling point [K] [9]
Linoleate	$C_{19}H_{34}O_2$	55%	619.15
Oleate	$C_{19}H_{36}O_2$	23%	617
Linolenate	$C_{19}H_{32}O_2$	7%	620.15
Palmitate	$C_{17}H_{34}O_2$	11%	597.7
Stearate	$C_{19}H_{38}O_2$	4%	623.7

2.2 Production of Biodiesel

This section provides a rough overview of the process of producing biodiesel; the reactions, the process conditions and emphasis of current research.
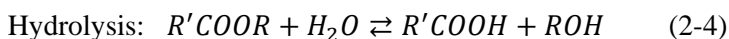
The Reaction

Oils and fats can be converted to biodiesel through either a transesterification or an esterification process with a low weight alcohol (usually methanol or ethanol) and a basic or acidic catalyst. The two different reactions are presented below [4, 10]:

$$\text{Transesterification:} \quad TAG + 3ROH \rightleftarrows 3RCOOR' + C_3H_5(OH)_3 \quad (2\text{-}1)$$

$$\text{Esterification:} \quad R'COOH + ROH \rightleftarrows R'COOR + H_2O \quad (2\text{-}2)$$
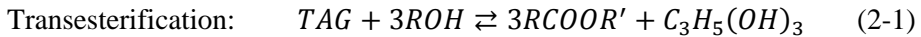
The transesterification reaction is the most common process for raw materials that have a low percentage of free fatty acids (FFAs), such as pure soybean oil or palm oil. The reaction raw materials consist of an alcohol reacting with a triacylglycerol (TAG) to produce glycerol and alkyl esters (the biodiesel). The process normally applies a homogeneous alkali catalyst and occurs at atmospheric pressure, modest temperatures and with an excess of alcohol present [4]. The heat of reaction is exothermic, but varies with varying composition of the biodiesel and the alcohol involved. Transesterification is a reversible reaction but in general the reverse reaction does not occur as glycerol and the alkyl ester form immiscible phases [10].

The esterification reaction is used for raw materials that have a modest to high degree of free fatty acids such as waste cooking oil. This reaction converts the free fatty acids to alkyl esters in one single step, and normally employs an acid catalyst such as HCl or sulphuric acid. Because it is uncommon to have only FFAs present in a raw material source, it is normal to have the transesterification and esterification reactions occur simultaneously to target both the free fatty acids and the TAGs at the same time, or to use the esterification process as a pre-treatment step before the transesterification reaction [4].

There are two known competing reactions to the biodiesel production, called the saponification (soap-formation) reaction, and the hydrolysis reaction. These are shown below as equations 2-3 and 2-4 [4].

$$\text{Saponification:} \quad R'COOH + NaOH \rightleftarrows R'COONa + H_2O \quad (2\text{-}3)$$

$$\text{Hydrolysis:} \quad R'COOR + H_2O \rightleftarrows R'COOH + ROH \quad (2\text{-}4)$$

Both of these reactions are unwanted and may propose difficulties with cleaning the equipment, and dilution of the product, deactivation of the catalyst and interference with the main reaction. The saponification and hydrolysis reactions depend on the presence of free fatty acids (FFAs), but research has shown that if the amount of FFAs is less than 0.5% the reduction in reaction efficiency is negligible [10].

The Catalyst
It is most common to use homogeneous alkali catalysts for the transesterification reaction. This yields high reaction rates at low operational costs (ambient temperatures and pressures). However, the homogeneous catalyst can be difficult to recover, they cause damage to the column due to corrosive properties and waste treatment and contamination can be a challenge post-reaction [10]. The catalysts are usually present at concentrations of approximately 1 percent, and the most common catalysts are NaOH, KOH or sodium methoxide, with sodium hydroxide being the cheapest and most popular alternative. However, NaOH is also the catalyst with the highest costs associated with waste disposal, cleaning, damage to the column and also the one which has the most negative impact on the environment [4].

Research is constantly being conducted in trying to find new alternative catalysts and especially popular is the research for heterogeneous catalysts [4]. A heterogeneous catalyst reduces the number of purification steps for the biodiesel post-reaction, it is reusable, it offers an improved environmental profile [10] and it is also very beneficial for reactive distillation purposes where the catalyst can be fixed on every tray [5]. There has also been a development of processes that apply ultracritical conditions or enzymes to avoid the use of catalysts completely.

The Process
There are several methods for producing biodiesel, such as batch, continuous, enzymatic, supercritical and reactive separation [5]. The advantages and disadvantages of each process are described in Table 2-2.

Strong incentives for efficient production of renewable fuels have shifted the interest to larger scale production of biodiesel. Because of this, the processes mostly used in industry today are the multistep and the continuous method. The enzymatic method and the supercritical method show promise but do not offer enough of an economic advantage to be applied commercially at the time being [5].

A lot of research is being developed on reactive separations at the present and this group consists of: reactive distillation, reactive absorption, membrane reactors and reactive extraction. Reactive distillation is the reactive separation process with the most applications [5], the focus of this paper and is explained in more detail in the next subsection.

Table 2-2: Overview of the different processes for producing biodiesel [5].

Process	Advantages	Disadvantages
Batch	Good flexibility with regard to feed composition.	Low productivity and high operational costs.
Continuous	Combination of transesterification and esterification reaction which leads to high productivity.	Normally uses homogeneous catalysts, which means it is subject to corrosion and extensive cleaning is necessary.
Supercritical	No catalyst and no hindrance to the transesterification kinetics due to oil-alcohol miscibility.	Severe conditions that require special equipment (costly).
Enzymatic	Low energy requirements.	Low yields and productivity, long reaction times.
Multistep	Applies both transesterification and esterification. High purity glycerol is obtained as a by-product. Solid catalyst can be used.	High capital costs
Reactive Separation	High conversion and yields, reduced post-processing	Damage of equipment if combined with homogenous catalysts

2.3   Reactive Distillation

Integrated reactive and separation processes is not a new idea, it has been used in the petroleum industry for decades, but advances in commercial applications helped spark new interest in the technology in the 1980s. These were largely due to the successful construction of a reactive distillation process for production of methyl tert-butyl-ether (MTBE) and methyl acetate, where the latter was able to comprise a traditional process consisting of 9 distillation columns and a reactor into a single integrating column, saving a lot of money on capital costs and

energy. Since then, the number of patents and research papers published on the matter has increased by tenfolds, and new applications for reactive distillation are continuously being researched [6, 11, 12].

Reactive distillation is the combination of a distillation column and a reactor in a single unit. As mentioned in the previous paragraph, the most apparent advantages are decreases in capital and operational costs. Simplified, the reactive distillation unit showcases the following characteristics as described in Table 2-3, which are categorized as advantages and disadvantages accordingly [13].

On the basis of the information in Table 2-3, one can deduce that the advantage of introducing reactive distillation over the traditional reactor followed by a distillation column has the greatest potential if the process involved has limitations when it comes to conversion or separation or both. For example it is an excellent alternative for reactions with very low equilibrium rate constants and for mixtures with azeotropes [5, 11]. Often, if the process has no such problems, there is no financial benefit in changing the design to reactive distillation and it may not even be feasible [11]. E.g. the introduction of a reactive distillation column where there has previously never been any problems with separation, may introduce new "reactive azeotropes" and thus serve as a hindrance for successful separation instead of acting as a promoter. A reactive azeotrope gives a constant boiling mixture with constant liquid and vapour compositions [11, 13].

Also, it is important to note that the operational conditions for the reaction and the distillation column must overlap in order to use a combined process. A regular process that has vast differences in the limiting conditions for temperatures and pressures in the distillation column and the reactor will not yield a feasible reactive distillation process. The compromised process conditions of the reactive distillation column may also be far from the optimum conditions of the segregated units and give worse overall performance [5].

Table 2-3: Characteristics of the reactive distillation process [13]

	Advantages	Disadvantages
Continuous removal of product	- The continuous removal of products in the reaction phase shifts the equilibrium to the right to achieve higher conversion. - The concentration of volatile products is kept low, reducing the rates of side-reactions and improving selectivity.	- The volatilities of the reagents and the products must be suitable in order to get the desired concentrations at desired positions in the column.
Simplification of separation system	- The elimination or reduction in separation systems results in great capital savings. - The flowsheet is simplified which leads to a simpler process which is easier to understand.	- The process conditions for the reactor and the distillation process must overlap. In some cases the conditions in the reactive distillation unit is far from optimal, and may give less efficient operation.
Heat integration	- An exothermic reaction can easily be used to supply heat of vaporisation to reduce reboiler duty. This is more efficient heat transfer than traditional heat exchangers.	
Azeotropes	- Azeotropes of the product mixture from the reactor can be avoided by reacting the mixture under specific conditions in the column.	- The presence of a reaction may introduce reactive azeotropes where there previously was none.
Liquid distribution		- Liquid distribution problems make it difficult for scale up to large flowrates.
Residence time		- A long residence time for the reaction will require a very large column with corresponding large capital costs.

Feasibility

The transesterification reaction is not a traditional choice for a reactive separation process as the conventional process showcases neither slow reaction kinetics nor problems with separating methanol from the glycerol/biodiesel mixture. In this case, reactive distillation is used to overcome the equilibrium limited transesterification reaction and obtaining a higher conversion and yield [1]. Reactive distillation also contributes to making a more economic process by removing some post-processing equipment, as biodiesel is produced and methanol removed in a single step [5].

As mentioned earlier, the process conditions for the reactor unit and the distillation column must overlap in order for the integrated unit to be a feasible solution. For a normal transesterification process the reactor conditions are usually at ambient temperatures and pressures with an excess amount of methanol in the range of 6:1 to 20:1 for alkali catalysts. Higher temperatures can also be used as the reaction speeds up at higher temperatures and viscosity of the oils is reduced [4]. However, the temperature should not exceed 150°C or 250°C as this will lead to the decomposition of glycerol and methyl linoleate respectively [1]. The reactive distillation column will operate at the bubble point temperature of the mixture, which is around 100°C [1]. This fits well inside the specifications on the reactor. By looking at the respective boiling points of the compounds, one can see that there are large differences in the volatilities of the compounds. Hence there is no need to operate at higher pressures; methanol is easily recovered as the light key in the distillation process.

3   The Reactive Distillation Models

This chapter presents a summary of the principles behind the reactive distillation models used to describe the production of biodiesel. The Matlab model design is covered in its own subsection and is not meant to be a detailed account of the modelling procedure, but rather a brief presentation of previous work. This presentation should be sufficient in order to understand the alterations performed in later sections of the thesis. The last part of this section presents the results achieved from simulation of these models and a comparison with the published work of Simasatitkul et al [1].

3.1   Reactive Distillation System

The Simulink model and the DAE-system model in Matlab R2012a were constructed during the specialisation project in the autumn of 2012, while the Hysys model was used in the work of Simasatitkul et al. [1]. An illustration of the reactive distillation system is given below in Figure 3-1.
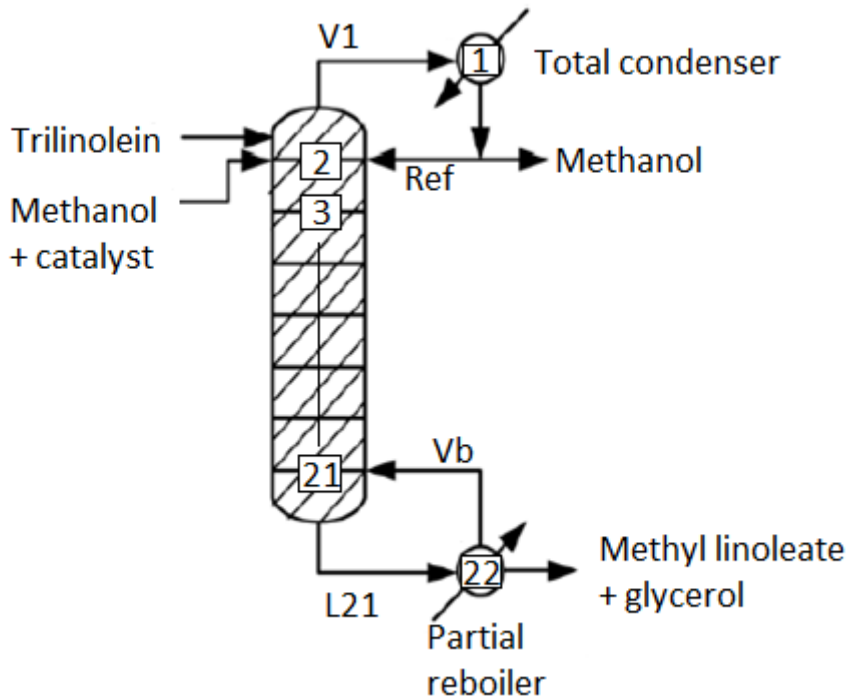

Figure 3-1 : Modified flowsheet of the RD process as published by Simasatitkul et al. [1]

Initial Conditions

The initial conditions applied for the models are summarized in Table 3-1 below:

Table 3-1: Initial conditions for the reactive distillation of biodiesel from Simasatitkul et al. [1].

Condition	Value	Comment
Temperature of feed	323.15 K	
Pressure	1 atm	The pressure drop was considered negligible.
Reflux ratio	3	
Duty of reboiler	$1.2*10^7$ kJ/h	
Number of trays	20	This excludes the reboiler and the condenser.
Feed tray	1	The feed is introduced at the first tray (counted from the top) because the feed contains a mixture of methanol and the catalyst. If the feed was to be introduced at a lower point in the column the number of reactive stages would be reduced as no catalyst would be present.
Catalyst	NaOH	NaOH was chosen as the catalyst because of the availability of kinetic data.

In addition to the applied conditions given in Table 3-1, the following assumptions were also made to ease calculations in the model:

1. The soybean oil only consists of linoleic esters.

2. The pressure drop in the column is considered negligible.

3. The rate of the saponification and hydrolysis reactions are considered negligible.

4. The course of the reactions is completely described by their kinetic equations

5. The mass of the homogeneous catalyst is negligible.

6. The catalyst will have no effect on the phase equilibrium.

7. The reaction will only occur in the liquid phase.

8. Phase equilibrium is established on every tray.

9. The reaction will only occur on the actual trays and not in the reboiler or in the condenser.

10. The efficiency of the trays is 100%.

11. The vapour holdup is considered negligible.

12. The vapour behaves as an ideal gas.

The Reactions

As mentioned in section 2.2, the trilinolein ester reacts to form biodiesel through the transesterification reaction. The consecutive steps of the transesterification reaction are given below as reactions 3-1 to 3-3 [4].

$$TG + CH_3OH \xrightleftharpoons[k_2]{k_1} DG + RCOOCH_3 \qquad (3\text{-}1)$$

$$DG + CH_3OH \xrightleftharpoons[k_4]{k_3} MG + RCOOCH_3 \qquad (3\text{-}2)$$

$$MG + CH_3OH \xrightleftharpoons[k_6]{k_5} GL + RCOOCH_3 \qquad (3\text{-}3)$$
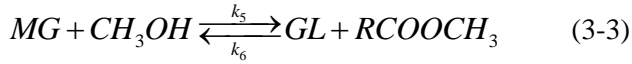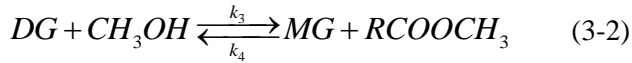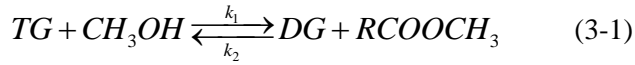
The values for the rate constants and the activation energies of the reactions 3-1 to 3-3 are given in Table 3-2, in the form of the Arrhenius Equation which is displayed below as equation 3-4 [14].

$$k_i = A_i \cdot \exp\left(\frac{Ea_i}{RT}\right) \qquad (3\text{-}4)$$

Table 3-2: The kinetic parameters for the transesterification reactions [1]

Rate constant	$A \ [\text{m}^3 \ \text{kmol}^{-1} \ \text{h}^{-1}]$	$E_a \ [\text{kJ mol}^{-1}]$
k_1	$1.4040 \cdot 10^{11}$	54.999
k_2	$2.0808 \cdot 10^9$	41.555
k_3	$2.1262 \cdot 10^{16}$	83.094
k_4	$3.5597 \cdot 10^{13}$	61.250
k_5	$1.9206 \cdot 10^7$	26.865
k_6	$7.5600 \cdot 10^7$	40.116

3.2 The Matlab Model

The model constructed during the specialisation project was made using Matlab R2012a Simulink. The model is dynamic and was based on and compared to the published results by Simasatitkul et al [1]. The template used was the multicomponent column A model by Skouras [15], with added reactive properties. To model a distillation column it is necessary to include material balances, liquid dynamics and the vapour phase equilibrium, while a reactor requires information on the kinetics and the reactions occurring.

A visualisation of the column was shown previously in Figure 3-1, where the stages are numbered from top to bottom. The feed is introduced at stage 2 (with the condenser drum as stage 1). The inputs to the distillation column are the feed and the feed composition. The manipulated variables consist of the vapour boilup (Vb), the reflux (Ref), the bottoms product flowrate (B) and the distillate flowrate (D).

There are 6*22 available degrees of freedom in the column, from the number of components and the number of trays available. It is desirable to calculate both the total molar holdup and the molar fractions on each tray. Therefore the states used for the iteration were the molar fractions of NC-1 of the components and the molar holdup at every stage. This way, the molar composition of the last component could be found by overall mass balances.

The Reactions
The transesterification reactions were given earlier as equations 3-1 to 3-3. From the rate law [14] of these reactions and by applying the kinetic constants of Table 3-2, equations for the rate of change per species are given as equations 3-5 to 3-9:

$$r_{TG} = -k_1[TG][MetOH] + k_2[DG][BD] \tag{3-5}$$

$$r_{DG} = k_1[TG][MetOH] - k_2[DG][BD] - k_3[DG][MetOH] \\ +k_4[MG][BD] \tag{3-6}$$

$$r_{MG} = k_3[DG][MetOH] - k_4[MG][BD] - k_5[MG][MetOH] \\ +k_6[BD][GL] \tag{3-7}$$

$$r_{GL} = k_5[MG][MetOH] - k_6[BD][GL] \tag{3-8}$$

$$r_{MetOH} = -k_1[TG][MetOH] + k_2[DG][BD] - k_3[DG][MetOH] \\ +k_4[MG][BD] - k_5[MG][MetOH] + k_6[BD][GL] \tag{3-9}$$

The Material Balances

In the reactive distillation column, the number of moles is changing, and hence the molar balances must only be applied in the sections where it is assumed to be no reaction. This will constitute the reboiler and the condenser, whose molar flows are shown below in equations 3-10 and 3-11.

$$V_1 = Ref + D \qquad (3\text{-}10)$$

$$L_{NT+1} = B + Vb \qquad (3\text{-}11)$$

Also, the first liquid flowrate and the last vapour flowrate are set by the reflux and the vapour boilup accordingly:

$$L_1 = Ref \qquad (3\text{-}12)$$

$$V_{NT+1} = Vb \qquad (3\text{-}13)$$

In the reactive parts of the column there is also a requirement for balances, and molar balances were applied for simplicity and consistency. To correct for the change in moles due to reactions occurring, an extra parameter was added ($M_{i,j}$). The reactive parameter is defined below, where the value VR_i represents the volume holdup of component i [14]:

$$M_{i,j} = r_{i,j} \cdot VR_{i,j} \qquad (3\text{-}14)$$

$$M_j = \sum_{i=1}^{NC} r_{i,j} \cdot VR_{i,j} \qquad (3\text{-}15)$$

The resulting balances for the total molar holdup and the individual molar holdup on the reactive trays are shown below as equations 3-16 and 3-17.

$$\frac{dHR_j}{dt} = V_j - V_{j-1} + L_{j-1} - L_j + M_j \qquad (3\text{-}16)$$

$$\frac{dHR_j x_{i,j}}{dt} = V_j y_{i,j} - V_{j-1} y_{i,j-1} + L_{j-1} x_{i,j-1} - L_j x_{i,j} + M_{i,j} \qquad (3\text{-}17)$$

On the feed tray, the molar balance must also account for the additional feed molar flowrate. The subscript F is used to indicate that this balance is only for the feed tray.

$$\frac{dHR_F}{dt} = V_F - V_{F-1} + L_{F-1} - L_F + M_F + F \qquad (3\text{-}18)$$

$$\frac{dHR_F x_{i,F}}{dt} = V_F y_{i,F} - V_{F-1} y_{i,F-1} + L_{F-1} x_{i,F-1} - L_F x_{i,F} + M_{i,F} + F x_{i,F} \qquad (3\text{-}19)$$

The individual molar holdup differential can be rewritten in terms of the derivatives of the composition with time and the derivative of the total molar holdup:

$$\frac{dHRx}{dt} = HR\frac{dx}{dt} + x\frac{dHR}{dt} \qquad (3\text{-}20)$$

To find the change in molar fraction with time, (dx/dt), from the change in individual molar holdup with time $(dHRx/dt)$, algebraic manipulation of equation 3-20 yields:

$$\frac{dx}{dt} = \frac{\dfrac{dHRx}{dt} - x\dfrac{dHR}{dt}}{HR} \qquad (3\text{-}21)$$

Vapour Equilibrium
Raoult's law is an ideal law that describes vapour-liquid phase-behaviour at equilibrium. The law is cited in equations 3-22 and 3-23 which give relations for the vapour (y_i) and liquid (x_i) compositions in regard to the vapour pressure (P_i^*) exerted by the liquid in a binary mixture [16]:

$$P = x_i P_i^* + x_j P_j^* \qquad (3\text{-}22)$$

$$y_i = \frac{x_i P_i^*}{P} \qquad (3\text{-}23)$$

Equation 3-22 can easily be extended to cover multiple component mixtures [16].

Dynamics

For the liquid dynamics, an adaptation of the Francis' weir formula was applied which calculates the liquid crest over the weir (h_{ow}) as an equation of the weir length (l_w), the density of the liquid (ρ_L) and the liquid flowrate (L_w) [17]. The equation for the liquid flow from each tray was derived and is shown below as function 3-24.

$$L_w = \left[\frac{h_{ow} \cdot 10^3}{750} \right]^{1.5} \cdot \rho_L \cdot l_w \qquad (3\text{-}24)$$

The Francis' weir equation is further illustrated in Figure 3-2 which shows the flow behaviour of the liquid on a rectangular tray in a distillation column.
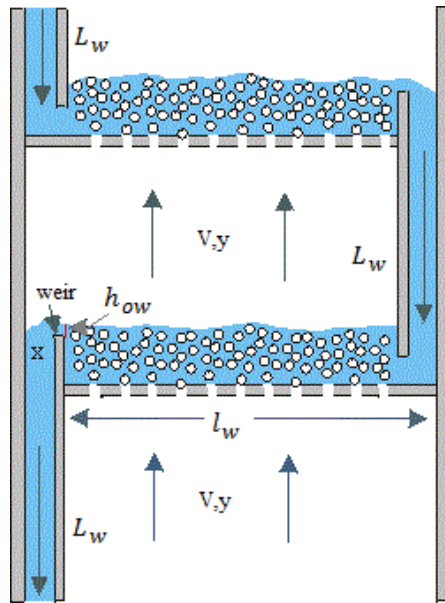
Figure 3-2: A visualisation of the liquid dynamics [18].

3.3   Results

This subsection represents the results obtained for the composition profile by simulation of the Matlab model, the Hysys model and the published results by Simasatitkul et al [1]. A comparison of the composition profiles is given in Figure 3-3.
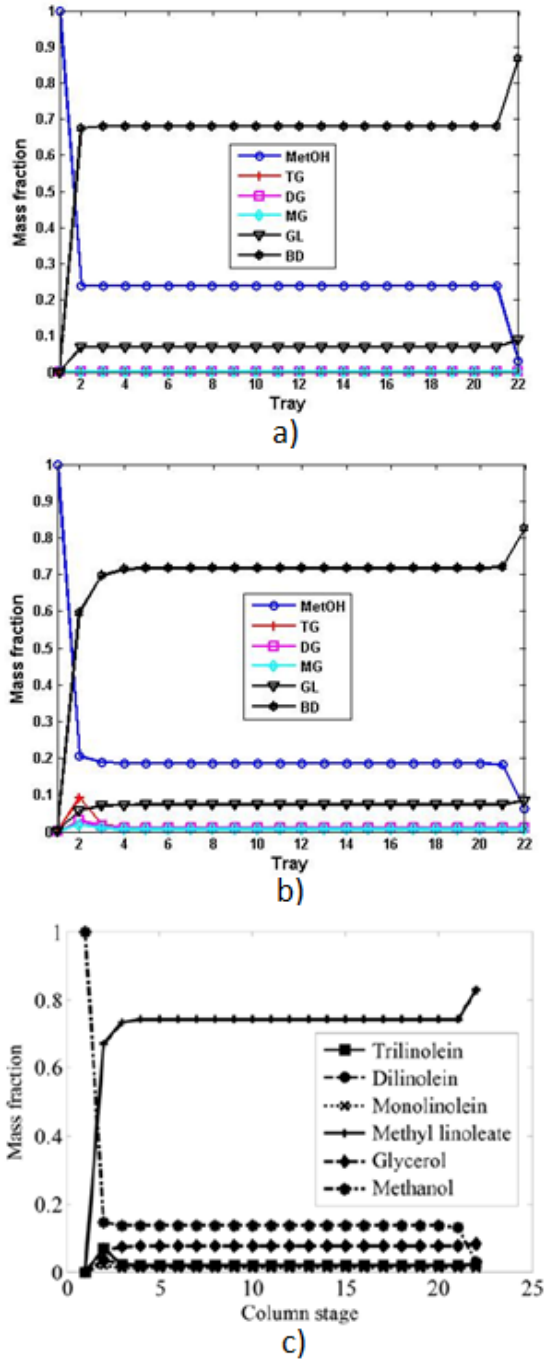
Figure 3-3: Comparison of mass fraction profile of a)Matlab Simulink model, b)Hysys model and c) published results by Simasatitkul et al.[1].

From the analysis of the performance of the Matlab model compared to the published results and the Hysys model, the Matlab model gives a similar shape on the profile and compositions. However, some excess methanol is present in the column, and the product is too pure compared with the other graphs.

The Matlab model was based on the same assumptions and initial conditions as the published results and the Hysys model. The shape of the Hysys model graphs are 'smoother' and more curved than the Matlab model, and there is also a slight peak of glycerides in the first trays which is absent from the Matlab profile. It appears that in the Matlab model, the reaction reaches chemical equilibrium faster than in the Hysys model (requires less stages). This can be seen as the soybean oil (TG) and reaction intermediates are present in much smaller concentrations in the first stages. This may be due to larger holdups in the Matlab model, or more complex interactions in the Hysys model.

As expected for the Hysys model and the published results, the simulations are very similar. Unfortunately they are not exactly the same and there are some small visible deviations; the mass fraction of methanol is higher in the Hysys simulation, and on the second tray the biodiesel mass fraction is smaller in the Hysys simulation than in the published results. Seeing as the Hysys model is identical to the one used in the paper by Simasatitkul et al. [1], the results should also be identical. The deviations may be due to the fact that the model is very complex and gives slightly different results when introduced to new initial values, or it could be a print error.

Ideally, simulations in the Hysys model, the Matlab model and the published results should give almost identical profiles. From this analysis it was clear that some alterations were necessary to achieve a sufficient representation of the system by the dynamic Matlab model.

4 Modifications

This chapter covers the modifications made to the Hysys and Matlab models and presents the results obtained following these alterations. The basis of the modifications made is strongly influenced by the analysis of the results covered in the previous chapter. A print of the modified versions of the Simulink and Hysys models have been provided in Appendix A and B respectively.

The purpose of this chapter is not to obtain the most scientifically correct data for the Matlab model, but rather create a common basis for comparison between the two models.

4.1 Alterations

The results obtained from the Hysys model and the Matlab model deviated both from each other and from the published results. In order to get a more accurate basis for the analysis, some small alterations had to be made to both models regarding information on the chemical compounds and their interactions. These alterations are listed explicitly in the following subsections, followed by a section describing the results achieved after performing these modifications.

The Matlab Alterations

The first alteration made to the Matlab model was to update the boiling temperatures (Tb) to those of the Hysys model, denoted as Tb_H and posted in Table 4-1 below. The original boiling temperatures (Tb_O) were from the DIPPR project 801 database [9] as recommended by the American Institute for Chemical Engineers (AIChE). However, the values for monolinoleate [19] and dilinoleate [20] were not covered in the DIPPR 801 database at the time of writing the specialisation project, and hence are from independent sources.

Table 4-1: The boiling points for the components of the process [9, 19, 20]

	MetOH	TG	DG	MG	GL	BD
Tb_O [K]	337.85	895.3	942.6	758.2	561.0	619.15
Tb_H [K]	337.63	1199.65	922.35	696.35	561.0	653.15

As one can see from Table 4-1, the biggest differences in the boiling point temperatures were for the linoleic esters, with deviations as high as 300 K for trilinolein. This illustrates the uncertainties associated with determining the boiling points for the linoleic esters, and that the actual temperatures may deviate quite a lot from the points given here. The boiling points for glycerol and

methanol were approximately the same in both models and didn't represent any large changes, as these are well-known and established compounds.

The vapour pressure equations were also altered in the same manner as the boiling points, with the Hysys parameters replacing the original parameters of the Matlab model. The vapour pressure parameters used in the specialisation project for the Matlab model were also from the DIPPR database for TG, BD, MetOH and GL. However, the parameters for DG and MG had to be estimated. This was accomplished by applying the Joback method to find the critical temperatures and pressures, and then consecutively the Reidel method was used to find the vapour pressure parameters. The vapour pressure equation is given below as equation 4-1.

$$P^* = F \cdot \exp\left[A + \frac{B}{T} + C \cdot \ln(T) + D \cdot T^E \right] \qquad (4\text{-}1)$$

An extra parameter F was added to equation 4-1 to achieve the desired pressure dimensions. F represents conversion from kPa (Hysys model and Reidel estimations) and Pa (DIPPR 801 database) to atmospheric (atm) which is the default pressure dimension of the Matlab model. An overview of the parameters of the vapour pressure equation is given in Table 4-2 below. The subscript O is used to demonstrate the "old" values as applied in the specialisation project, while the subscript H shows the Hysys parameters.

Table 4-2: New and old parameters for the vapour pressure equations

	MetOH	TG	DG	MG	GL	BD
A_O	82.718	234.71	-15.931	118.95	99.986	105.47
A_H	59.837	505.21	18.155	145.25	169.21	107.53
B_O	-6904.5	-34699	-2111.0	-20181	-13808	-14531
B_H	-6282.9	-94126	-7165.4	-21154	-16893	-15604
C_O	-8.8622	-27.250	2.4303	-14.318	-10.088	-10.980
C_H	-6.3787	-59.788	-0.84781	-16.874	-21.813	-12.214
D_O	$7.4664E^{-6}$	$1.5475E^{-18}$	$8.0567E^{-21}$	$9.1481E^{-19}$	$3.5712E^{-19}$	$2.5753E^{-18}$
D_H	$4.6175E^{-6}$	$5.8959E^{-19}$	$3.1834E^{-20}$	$1.7403E^{-18}$	$1.1407E^{-5}$	$1.8000E^{-18}$
E_O	2	6	6	6	6	6
E_H	2	6	6	6	2	6
F_O	$9.869E^{-6}$	$9.869E^{-6}$	$9.869E^{-3}$	$9.869E^{-3}$	$9.869E^{-6}$	$9.869E^{-6}$
F_H	$9.869E^{-3}$	$9.869E^{-3}$	$9.869E^{-3}$	$9.869E^{-3}$	$9.869E^{-3}$	$9.869E^{-3}$

The validity of the vapour pressure equations for the linoleic compounds in Hysys is questionable as these are only estimations based on group contribution. This type of estimation method was proven to be very inaccurate for DG and MG during the specialisation project with errors of more than 50% when calculating the vapour pressure at respective boiling points. Also the temperatures for the reactive distillation process are outside of the temperature intervals for most of the vapour pressure equations given by Hysys. However, as this part of the thesis deals largely with modification of the Matlab model to give the same behaviour as the Hysys model, all old vapour pressure equations were replaced by the ones utilised by Hysys, even though the validity is highly questionable.

To further simplify the code, relative volatilities were implemented instead of Raoult's law and vapour pressure equations when calculating the composition in the vapour phase. The relative volatility is a measure for how volatile one compound is compared to a reference compound. The equations are shown below for a reference compound c [21]:

$$\alpha_i = \frac{y_i / x_i}{y_c / x_c} \quad (4\text{-}2)$$

$$y_i = \frac{\alpha_i \cdot x_i}{\sum\limits_{i=1}^{NC} \alpha_i \cdot x_i} \quad (4\text{-}3)$$

Calculations were performed by using the compositions of an intermediate tray (tray 11) in the Hysys reactive distillation column with TG as a reference compound. The relative volatilities calculated are shown below:

$$\alpha = \begin{bmatrix} \alpha_{MetOH} & \alpha_{TG} & \alpha_{DG} & \alpha_{MG} & \alpha_{GL} & \alpha_{BD} \end{bmatrix}$$

$$\alpha = \begin{bmatrix} 1.45 \cdot 10^{23} & 1.00 & 2.35 \cdot 10^{18} & 3.07 \cdot 10^{17} & 4.48 \cdot 10^{19} & 2.81 \cdot 10^{18} \end{bmatrix}$$

The Hysys Alterations
The Hysys model was used as a platform for comparison with the Matlab models, and thus not many alterations were made. The largest modifications performed were the removal of the catalyst feed to the distillation column and the removal of the heat exchanger for the oil feed. These removals were

performed to simplify the model as much as necessary for easy comparison with the Matlab model.

The catalyst feed (2.150 kmol/h) was very small compared to the oil and methanol feedrate (350 kmol/h). Also the catalytic action of NaOH is already taken into consideration in the kinetics of the reactions and its presence does not exert a large effect on the temperature in the column. As the catalyst feedrate had not been included in the Matlab models, it was decided to remove this federate also in the Hysys model. This also makes optimisation easier later on, as one can disregard the change in catalyst feed with changes in feed flowrates and ratios.

The heat exchanger on the oil feed was also removed to further simplify the model. The heat exchanger does not represent an essential part of the process and presented some unnecessary difficulties when introducing disturbances in the oil feedrate.

4.2   Results

The modifications of the last subsection were implemented and the resulting composition profiles for the Hysys and the Matlab model are shown below in Figure 4-1.
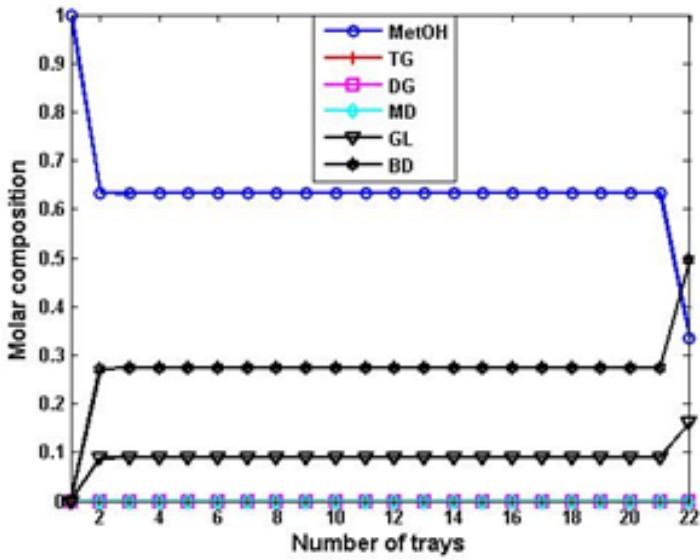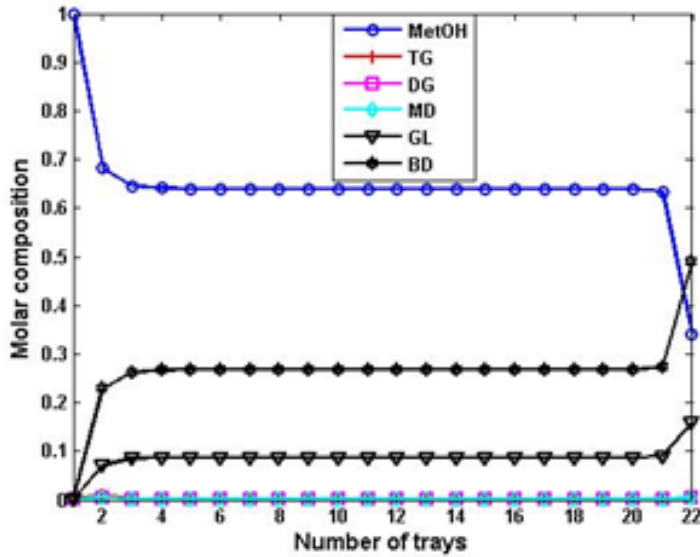
Figure 4-1: Comparison of molar fraction profiles from a) Matlab simulation and b) Hysys simulation

As one can see from Figure 4-1, the composition profiles for the two models are very similar following the alterations. The biggest difference is as mentioned earlier, that the Hysys model has "smoother transitions" and is more curved than

the Matlab model, probably due to more complex modelling equations or smaller holdups; the compositions on tray 2-3 and tray 21 deviate slightly from compositions on trays 4-20. The compositions of the three main components are summarized in Table 4-3 below for easy comparison:

Table 4-3: Molar compositions of MetOH, GL and BD from Hysys and Matlab simulations.

	Simulink model			Hysys model		
	x_{MetOH}	x_{BD}	x_{GL}	x_{MetOH}	x_{BD}	x_{GL}
Tray 1	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000
Tray 2	0.6342	0.2716	0.0884	0.6909	0.2192	0.0674
Tray 3	0.6318	0.2739	0.0894	0.6486	0.2598	0.0839
Tray 4-20	0.6318	0.2740	0.0894	0.6398	0.2682	0.0876
Tray 21	0.6318	0.2740	0.0894	0.6334	0.2726	0.0893
Tray 22	0.3336	0.4958	0.1618	0.3408	0.4906	0.1601

From Table 4-3 one can see the differences more explicitly. The values estimated by the Matlab model on the trays 3-21 are very similar to the compositions on tray 21 for the Hysys model. There are no visible differences in the condenser, but the product is slightly cleaner for the Matlab model than for the Hysys model.

5 Temperature Estimation

Energy balances were not included in the simple Matlab model, and hence it was necessary to find an alternative way of estimating the temperatures in the column. For the specialisation project this was done in two different ways: by a simple empirical method using only the compositions and the boiling point temperatures and in a differential algebraic equation (DAE) system. The Hysys model uses the activity-coefficient calculating method UNIQUAC for these calculations.

This chapter will briefly cover the basis of these temperature estimation methods and how they were implemented in the different models. The results of the different methods and a discussion of their accuracy are embedded in the same section to increase readability.

5.1 The Matlab Model

This subsection gives a brief introduction to the two main methods of temperature estimation used on the Matlab model during the specialisation project. For more information on these methods, please see the sources referenced to in the text.

Empirical Method

Two different empirical methods were employed in the pursuit of accurate temperature estimations. The first method will estimate the temperature on each tray based on the liquid compositions present, while the second method also takes the vapour compositions into account. Both of these temperature estimation methods were implemented in the s-function of the Simulink model and are described in the works by Halvorsen and Skogestad (2000) [22]. The equations for the first and second method are given below as equations 5-1 and 5-2 respectively:

$$T_j = \sum_{i=1}^{NC} x_{j,i} Tb_i \qquad (5\text{-}1)$$

$$T_j = \sum_{i=1}^{NC} \frac{(x_{j,i} + y_{j,i})}{2} \cdot Tb_i \qquad (5\text{-}2)$$

Equation 5-1 normally gives a higher temperature estimate than what is expected, while method 5-2 usually provides fairly accurate values for ideal mixtures [22].

DAE system

An alternative to the empirical methods is to use the vapour pressure equations to iterate on the temperatures on every tray until Raoult's law as shown in equation 5-3 converges.

$$0 = P - \sum_{i=1}^{NC} x_i \cdot P_i^* (T) \quad (5\text{-}3)$$

This method was applied to a differential algebraic equation (DAE) system with a mass matrix to speed up the computation time with regard to a regular fsolve system. At the time there is no way of solving DAE systems in Simulink, so the code was moved to pure m-files.

The implementation of equation 5-3 to the Matlab code merges a set of differential equations with an algebraic equation, as an extra set of states are introduced. This requires a DAE solver once the brute-force method is eliminated. A DAE solver is based on equation 5-4 below [23]:

$$M \frac{dy}{dt} = f(y, u) \quad (5\text{-}4)$$

Here the parameter M represents the mass matrix, which indicates the presence of algebraic and differential equations by a 0 or a 1, respectively, and usually diagonally. Equation 5-4 represents a general approach which allows for faster simulations [23]. The solver applied is ode15s, which is designed to handle stiff ODEs and DAEs.

5.2 Results

The temperature estimation methods were simulated for the composition profiles as given by the Hysys model in Table 4-3 and were compared to the published results of Simasatitkul et al.[1] in Figure 5-1 below.
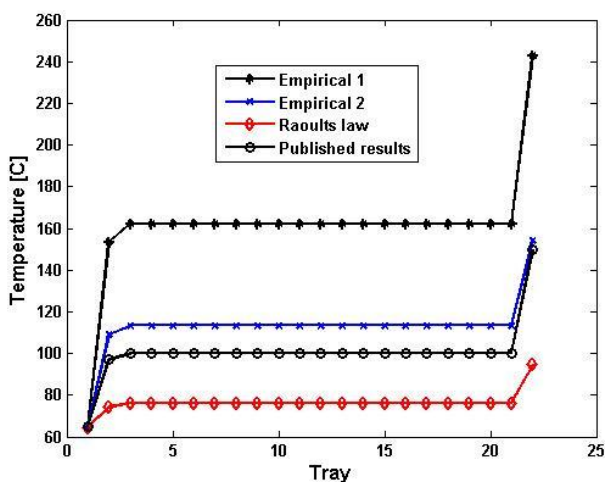
Figure 5-1: Comparison of temperature estimation methods with the published results by Simasatitkul et al [1].

From Figure 5-1 it could be seen that of the estimation methods, empirical method 2 gave the results with the smallest deviation from the desired value. However, the temperature was still overestimated by more than 10°C in the midst of the column. It was also demonstrated that Raoult's law did not provide a very good fit, and underestimated the temperature by almost 20°C in the middle of the column. This probably means that the system of methanol, glycerol and linoleic esters is not ideal and/or that the vapour pressure parameters are inaccurate, as was discussed earlier in section 4.1. The worst estimation method was estimation method 1 which overestimated the temperature by more than 60°C in the column.

5.3   Results - The Hysys Model

The Hysys model uses the UNIQUAC method in its temperature estimations, which calculates a parameter (γ) which corrects for non-ideal behaviour of the mixture. UNIQUAC is described in further detail in chapter 6.1.

One big issue with the Hysys model was that the published temperature estimations at initial conditions did not match the results attained by simulation. Although the condenser/reflux drum was estimated quite accurately, the deviation in the results equalled -10°C in the column and -25°C in the reboiler and is shown in Figure 5-2 below:
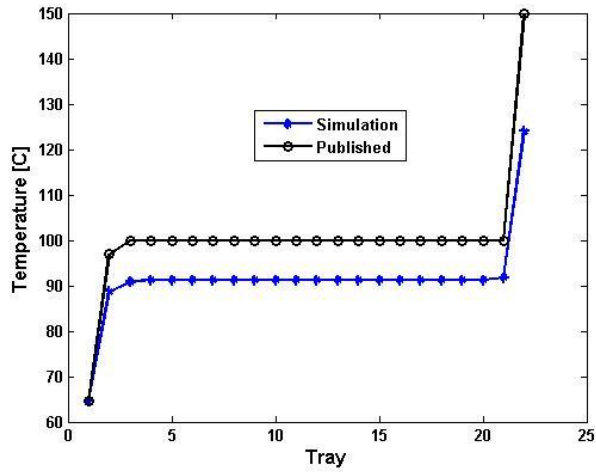
Figure 5-2: Comparison of published and simulated temperature estimations for the Hysys model.

Regarding these issues and also the differences in composition profiles, the Hysys model was chosen for comparison and evaluation of the Matlab model in this thesis instead of the published results. This was mostly because the availability of a model enabled more opportunities as the two models can be compared against each other for a variety of conditions. This will help discover the shortcomings and highlights of a simpler dynamic Matlab code versus involving energy balances and vapour dynamics in the Hysys model.

6 Modifications – Temperature Estimation

This chapter will introduce the theory behind some new temperature estimation methods, cover the implementation of these into Matlab models and present the results of this modification. A print of the UNIQUAC implementation has been provided in Appendix C, while the empirical temperature estimation is covered in Appendix D.

6.1 Background

From the analysis of the temperature estimation methods used during the specialisation project it was clear that some alterations were necessary in order to achieve higher accuracy in the results. Two different new approaches were tested; an adaptive empirical method and the UNIQUAC method.

Empirical Method

Two empirical methods for estimating the temperature were tested for the reactive distillation Matlab model. These empirical methods were combinations of compositions in the liquid and vapour phase and the respective boiling points of the components present, given by equations 5-1 and 5-2 in subsection 5.1.

For the alterations it was decided to test a custom empirical method for temperature estimation. The method evaluated is a modification of equation 5-2 with different weighting of the liquid and the vapour compositions, as given by equation 6-1:

$$T_j = \sum_{i=1}^{NC} \left(a x_{i,j} + (1-a) \, y_{i,j} \right) \cdot Tb_i \qquad (6\text{-}1)$$

In equation 6-1, the parameter a represents the weighting factor of the liquid composition. The weighting factor was found by iteration as the best fit to the system by applying temperatures and compositions from the steady-state results of the Hysys model. The resulting values for the weighting factors are given below in Table 6-1.

Table 6-1: The weighting factors for the empirical temperature estimation method:

	MetOH	TG	DG	MG	GL	BD
Weighting Factor	0.2354	0.2501	0.2502	0.2502	0.2559	0.2711

UNIQUAC

In the analysis of the Hysys and the Matlab model, it was shown that the system is most likely non-ideal. Because the Hysys model uses UNIQUAC to estimate the temperatures in the process, it seemed a natural step to implement this method for the Matlab model as well. The activity model interaction parameters were readily available from Hysys, and the template used for the model was based on the UNIQUAC.m model developed by Skouras (2005) [24] for a multicomponent non-ideal distillation column.

UNIQUAC is a model made to describe phase equilibrium in terms of providing activity coefficients which show deviations from ideal behaviour. UNIQUAC is an acronym for Universal Quasi-chemical and is based on a lattice model of statistical mechanics of interacting surfaces. The activity coefficients are combinations of a residual and a combinatorial contribution as described in equation 6-2 below for component i [25]:

$$ln\gamma_i = ln\gamma_i^C + ln\gamma_i^R \qquad (6\text{-}2)$$

The activity coefficients can be used to calculate the temperature by combining with the pressure equation as given below:

$$P \cdot y_i = \gamma_i \cdot x_i \cdot P_i^* \qquad (6\text{-}3)$$

Combinatorial Contribution

The combinatorial contribution is based on the lattice theory and is concerned with the shape differences of the molecules and how this contributes to the overall entropy [25]. The combinatorial contribution is given by equation 6-4 below:

$$ln\gamma_i^C = \left(1 - V_i + lnV_i\right) - \frac{z}{2}q_i\left(1 - \frac{V_i}{F_i} + ln\frac{V_i}{F_i}\right) \quad (6\text{-}4)$$

Here, z is the coordination number which corresponds to the number of close interacting molecules around a central molecule and varies from 6 (cubic packing) to 12 (hexagonal packing). An intermediate value of 10 was used in this implementation. The parameters V_i and F_i represent the volume fraction and the surface area fraction per mixture mole fraction respectively. The equations for V_i and F_i are given as 6-5 and 6-6 below:

$$V_i = \frac{r_i}{\sum_j^{NC} r_j x_j} \qquad (6\text{-}5)$$

$$F_i = \frac{q_i}{\sum_j^{NC} q_j x_j} \qquad (6\text{-}6)$$

The parameters r_i and q_i from equations 6-5 and 6-6 correspond to the relative Van der Waals volume and surface areas of the pure compounds respectively.

Residual Contribution

The residual contribution is an enthalpic correction of ideal behaviour by considering the change in energy as the molecules are mixing [25]. The residual activity parameter is calculated by the following equation:

$$ln\gamma_i^R = q_i \left(1 - ln\left(\frac{\sum_j^{NC} q_j x_j \tau_{ji}}{\sum_j^{NC} q_j x_j} \right) - \sum_j^{NC} \left(\frac{q_j x_j \tau_{ij}}{\sum_k^{NC} q_k x_k \tau_{kj}} \right) \right) \qquad (6\text{-}7)$$

Here the parameter τ_{ij} is an empirical coefficient which is derived from activity coefficients:

$$\tau_{ij} = exp\left(\frac{-\Delta u_{ij}}{RT} \right) \qquad (6\text{-}8)$$

The parameter Δu_{ij} is known as the binary interaction energy coefficient, and the values used for this implementation were derived from the Hysys model through UNIFAC estimation.

6.2   Results

The temperature profiles for the Hysys model compared with the empirical and the UNIQUAC estimation methods for the Matlab model is given below in Figure 6-1 for the steady state compositions of the Hysys simulation.
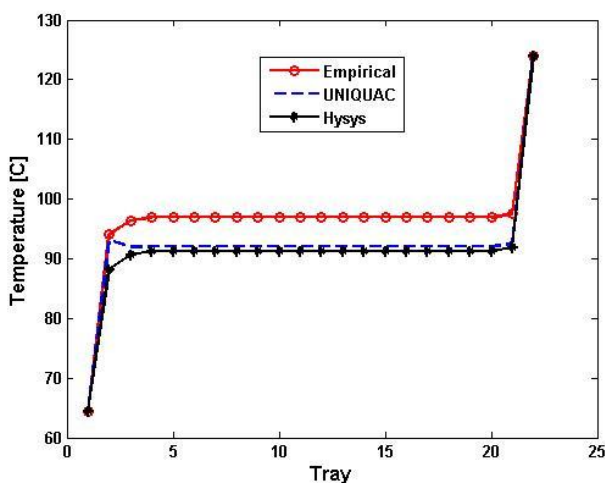
Figure 6-1: Comparison of UNIQUAC and empirical temperature estimation methods with Hysys simulations

From Figure 6-1 one can see that the UNIQUAC implementation is the best approximation for the temperature in the column, but it is not identical to the Hysys simulation results. Particularly trays 2 and 3 show some deviations in Hysys calculations and pure UNIQUAC estimations. This is probably due to the inclusion of energy balances in the Hysys model, as the deviations occur on the column stages that are most influenced by the reactions occurring. As the reaction is slightly exothermic, this may result in a slightly lower total temperature than estimated by UNIQUAC for these trays with the highest reactivity.

The empirical method did not provide a completely accurate estimate for the temperature in the column, with a deviation of approximately 5.6°C. However, the empirical method did estimate fairly accurate temperatures in the reboiler and reflux drum. The empirical method also has the same shape as the Hysys profile. All of the results of the simulations are summarised below in Table 6-2.

Table 6-2: Temperature estimations by different methods

Tray	Empirical	UNIQUAC	Hysys
1	64.48	64.49	64.48
2	94.05	93.04	88.18
3	96.40	91.94	90.67
4-20	96.91	91.99	91.27
21	97.49	92.58	91.85
22	123.86	124.35	124.01

From the comparison of the different temperature estimation methods above, UNIQUAC gave the smallest deviations compared to the Hysys model. However, both methods are discussed also in the next section which covers implementation into the Matlab model and behaviour of these methods with regard to disturbances.

7 Behaviour of Matlab Model

This chapter will discuss implementation of the temperature estimation methods into the Matlab model, and challenges associated with this. The section will also cover behaviour of the model with regard to disturbances and evaluation of the dynamics. A print of the DAE-system with corresponding steady-state and optimisation models developed due to this implementation is shown in Appendix E.

7.1 Implementation of UNIQUAC

Implementation of the empirical temperature estimation method was standard as it could be inserted directly into the s-function of the Matlab model. However, the UNIQUAC method is a bit more rigorous as it involves iteration to find the corresponding temperatures.

Implementation of UNIQUAC directly into the s-function of the Simulink model is possible through using the inbuilt fsolve function, but this makes simulation very slow. Also, only very small disturbances could be introduced before the system is made unstable. The obvious choice for faster calculations would be to use a DAE system as with the Raoult's law iterations in the specialisation project. However, ode15s requires a valid initial value for starting iterations, and after a long unsuccessful period of trial-and-error testing, a different approach was attempted.

Steady-state Model
To find valid initial values it was decided to simulate the system by an algorithm with no requirement for a valid iteration starting point. Steady state values from this simulation could then be applied as initial values for the DAE system. The steady state system was based on the modified version of column A, developed by Antonio Arauju (2008) [26]. This template is designed to carry out steady state- and dynamic simulation as well as optimisation of the same system in one script.

Some modifications of the code were necessary in order to convert the Simulink Matlab model into this new format. Among the changes was a complete vectorisation of the model including removal of all for-loops to provide faster iterations.

The relative volatilities calculated from the Hysys model also had to be altered as they made the system too stiff and the solvers available were not able to solve the problem. Testing different versions of alpha made negligible differences to

the results but greatly increased the speed of the iterations, also in the Simulink model. The new, modified relative volatilities are given below as α_2:

$$\alpha_2 = \begin{bmatrix} 1.45 \cdot 10^3 & 1.00 & 2.35 \cdot 10 & 3.07 \cdot 10 & 4.48 \cdot 10 & 2.81 \cdot 10 \end{bmatrix}$$

To further increase iteration speed and simplify the model the densities and the kinetic constants were assumed to be constant throughout the model, as calculated at 91.27°C by respective equations. The specific temperature was chosen as it represents the column temperature at initial conditions for the Hysys model. The dynamics and the kinetics were converted to minutes instead of hours also to increase iteration speed.

As a consequence of the implementation of constant densities, the Francis weir equation for liquid dynamics was altered to another popular form giving the liquid flowrate in the form of kmol/min. This form is given below as equation 7-1 and includes a constant K and molar holdup of the tray (HR) and molar holdup under the weir (HR_{uw}).

$$L = K \left(HR - HR_{uw} \right) \quad (7\text{-}1)$$

The constant K is calculated based on equation 3-24, where the constant A_t represents the area of the respective tray and is given in equation 7-2 below.

$$K = 60 \cdot \rho_l \cdot l_w \cdot \left(\frac{10^3}{750 \cdot A_t \cdot \rho_l} \right)^{1.5} \qquad (7\text{-}2)$$

Dynamic Model
The original dynamic model of the modified column A file was developed for Simulink. As earlier mentioned, Simulink results in very slow iterations when involving an iterative temperature estimation method, hence it was decided to use a DAE system-method instead. The DAE system was very briefly described in section 5.1. Also, the results of the steady-state model were applied as initial values for the dynamic model to ensure a valid starting point for iteration.

7.2 Results

This subsection provides the results achieved with the steady-state and dynamic model following the alterations made.

Steady-state

Simulation of the modified column A script gives steady state values in less than 10 seconds compared to the dynamic simulations of the Simulink model (almost 38 minutes). The composition profile of the Matlab model compared to the results of the Hysys model is given in Figure 7-1 below.
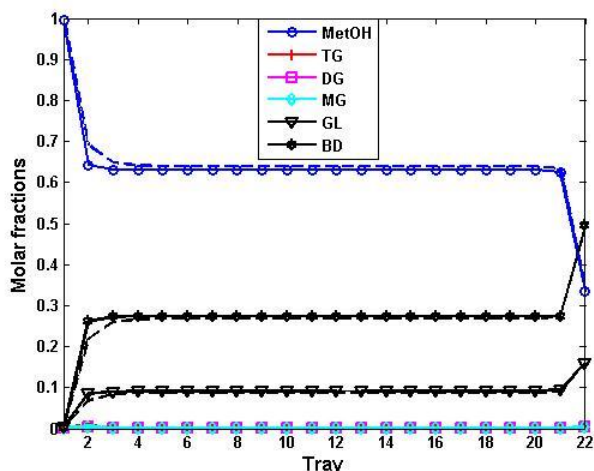
Figure 7-1: Composition profile of Matlab simulation (solid lines) and Hysys simulation (dashed lines)

Figure 7-1 shows that the composition profile of the Matlab model is very similar to the Hysys profile, even though it slightly underestimates the amount of methanol and overestimates the amount of esters and glycerol in the column. The curvature is also not exactly the same, as the Matlab model has a more rigid shape. Increasing the flowrates and decreasing the holdups in the column did help in achieving more of the curved shape. Hence the Matlab model can be manipulated into having an even more similar shape through further modification of these variables.

The temperature profile of the Matlab model is given in Figure 7-2 where it is compared to the Hysys results. Here the composition used as a basis for finding the temperatures belong to the respective models and therefore are not identical.
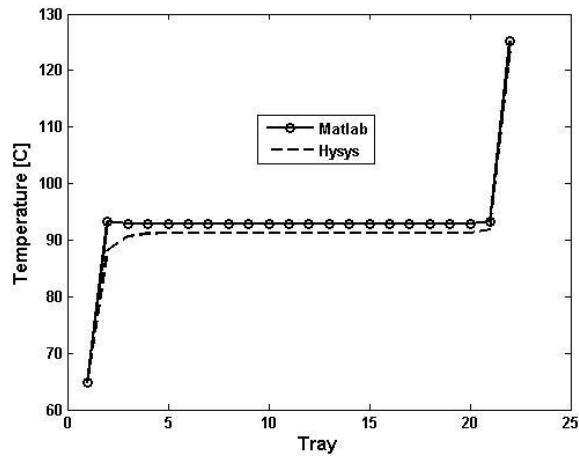
Figure 7-2: Comparison of the Matlab and Hysys temperature profiles.

A similar deviation occurs for the steady-state Matlab model in the figure above as with the results given for the Simulink model in Figure 6-1; the temperatures are slightly overestimated in the column while they are very accurate for the reboiler and condenser. The largest deviation is for the first tray in the column (tray 2 in the model).

7.3   Dynamic Model - Disturbances

The dynamic model was tested for some disturbances and its behaviour compared to the Hysys models' behaviour. The disturbances introduced were:

- D1: +10% in the feed F [kmol/min]
- D2: +10% in feed of triglycerides [kmol/min]
- D3: +10% in feed of methanol [kmol/min]

The composition profiles for the three disturbances are given below as Figure 7-3 compared to the corresponding profiles of the Hysys model.
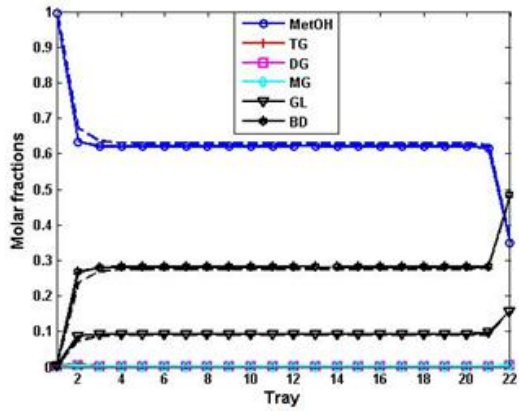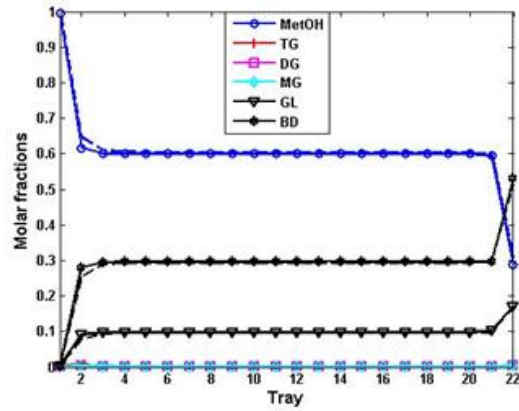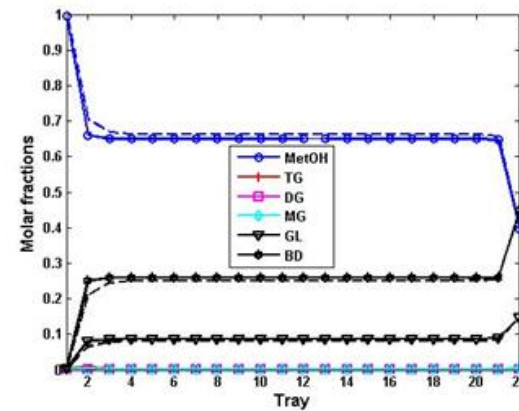
Figure 7-3: Composition profiles comparison of Matlab (solid lines) and Hysys (dotted lines) subject to disturbances: a) D1, b) D2 and c) D3.

From Figure 7-3 one could see that the dynamic model behaves in a similar fashion as the Hysys model for all three disturbances. Especially the increase in TG gave almost identical results for the two simulations.

With regard to the effect of the disturbances compared to the steady-state values, an increase in the feed has little effect on the composition profile. However, increases in the oil and in the methanol had a larger effect. Increases in the soybean oil lead to an overall increased concentration of biodiesel and glycerol in the column and in the products, while increasing the methanol lead to an overall decreased concentration of esters in the column. This is as expected.

Below in Figure 7-4 the temperature profiles for the system are shown subject to disturbances D1-D3.

Figure 7-4 shows that the Matlab model follows the same behaviour as the Hysys simulation with temperature estimation. The temperature estimations showcases similar tendencies as with the composition profile; that the behaviour is more similar for the two models when it comes to an increase in the TG feedrate than for the other disturbances. Clearly, the two models response to the disturbances are not identical, but they show the same tendencies; an increase in TG gives an overall increase in temperature while an increase in methanol gives an overall decrease in temperature.

More surprising is the fact that an increase in the overall feedrate yields a small increase of approximately 1°C in the overall temperature for both models. This is probably due to the duty on the reboiler being set, which means that even though the feed is increased, the vapour boilup flowrate will stay constant. This will decrease the vapour/liquid flowrate ratio and lead to a slight increase in temperature.

Part 1 - Modification
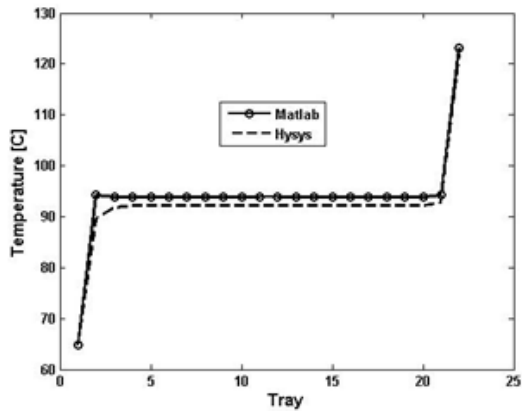
a)
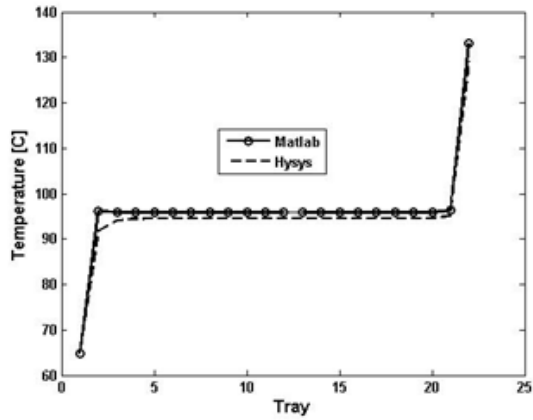
b)
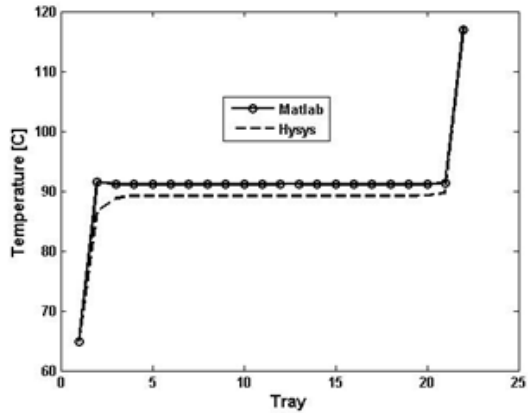
c)

Figure 7-4: Temperature profiles comparison of Matlab and Hysys simulations subject to disturbances: a) D1, b) D2 and c) D3.

Dynamic Model - Dynamics

Of course, an essential part of the dynamic model is also to test the behaviour of the dynamics. The disturbance applied to the system is a +10% increase in TG (D2) occurring at time 2 minutes for a simulation occurring for a total of 10 minutes. The effect of the disturbance on the composition of the three key components MetOH, BD and GL at tray 10 and tray 22 in the column is shown in Figure 7-5 below.
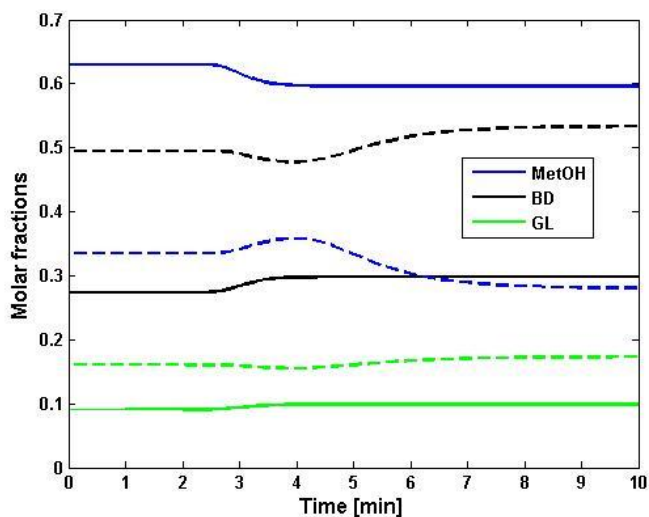
Figure 7-5: The composition profile with time after being subject to a disturbance of +10 % TG in feed at time 2 minutes. Stapled lines represent tray 22 while solid lines represent tray 10.

From Figure 7-5 one can see that the dynamics are working and that the changes are gradual. To reach a new steady state takes a couple of minutes, and the change is slower in the reboiler than in the middle of the column. There is also a slight delay from when the disturbance first occurs to the system responds, which increases further down the column. However, Figure 7-5 displays fast dynamics compared to a regular distillation column. Also, in a regular distillation column, the change would be quicker in the reboiler compared to the midst of the column. But of course, this is not a regular column in that the feed is introduced at the top and that reactions are occurring simultaneously.

One can also see that the change in the reboiler includes a slight overshoot before steady states are achieved. This is probably due to the increase in

volumetric feed which would lead to more methanol flowing down from the trays of the column and reaching the reboiler.

8   Optimisation Background

This chapter marks the start of the second part of the master thesis, as there is a shift in focus from modification of the model over to optimisation and regulation. The section will briefly explain the systematic procedure of economic plantwide control as described by Skogestad and Larsson [27], and compare this to other corresponding methods. Lastly, this chapter will include some theory on the post-processing required to achieve the finished biodiesel product.

8.1   Plantwide Control

Plantwide control is the overall control philosophy of a plant. There are several methods on how to structure the plantwide control, the main two being the process oriented and the mathematical oriented approach. The process oriented approach considers all possible structural decisions and uses engineering experience to eliminate certain options, while the mathematical oriented approach is more focused on mathematical optimisation [27].

The control system of a plant is often separated into layers; scheduling, site-wide optimisation, local optimisation, supervisory control and regulatory control. Each layer normally operates on a specific time frame. A visualisation of the control hierarchy with the corresponding time frames for normal chemical plants is given in Figure 8-1 [27].
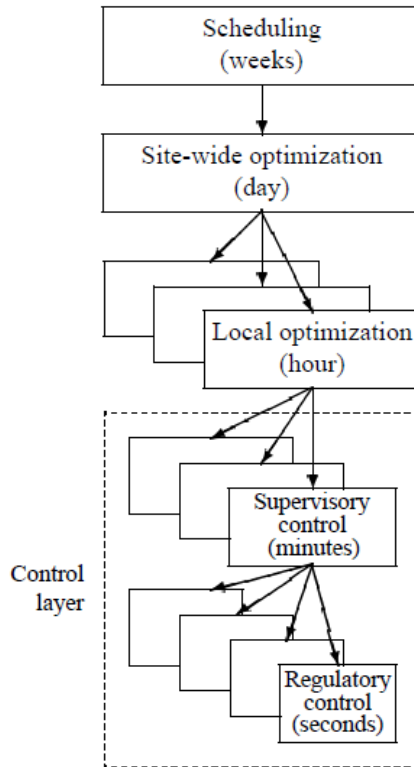
Figure 8-1: Control hierarchy of a chemical plant [27]

Regardless of what approach is utilised for providing a plantwide control structure, the following structural decisions have to be made [28]:

1. Select economic primary controlled variables (CV1) for the supervisory control layer.
2. Select stabilizing secondary controlled variables (CV2) for the regulatory control layer.
3. Position the throughput manipulator (TPM).
4. Select pairings for the stabilising layer.

As the supervisory and regulatory layer operates on separate time scales, the setpoints of the secondary controlled variables link the two control layers while the setpoints of the primary controlled variables link the control layer to the optimisation layer.

Sigurd's Method

The process oriented approach has been criticised for not being very systematic and for having little focus on economics, while the mathematical oriented approach has been criticised for its optimisation problems being too difficult to apply for an entire plant. Skogestad's method is a hybrid between the two main approaches, as it represents a systematic approach combining optimisation and stabilising control all in one.

Skogestad's method is the so-called top-down, bottom-up approach which first creates an economic optimisation and formulates a common control philosophy (top-down) in the optimisation layer, while later looking at stabilizing control (bottom-up) in the control layer. There are seven steps to this method which are given explicitly in Table 8-1 below [28].

Table 8-1: The seven steps of the Skogestad top-down, bottom-up method [28]

Top-down approach (steady-state)	
Step 1	Define operational objectives
Step 2	In steady-state: find degrees of freedom and determine optimal operating conditions and active constraints.
Step 3	Identify candidate measurements and hence primary controlled variables.
Step 4	Select the location of the TPM.
Bottom-up approach (dynamic)	
Step 5	Select stabilising controlled variables as well as inputs and pairings to control them.
Step 6	Select the structure of the supervisory control layer.
Step 7	Assess the need for a real-time optimisation (RTO) layer and select structure if required.

Steady-state in parenthesis for the top-down approach is noted because normally it is sufficient to use a steady-state model for the plantwide control optimisation as this section is mainly concerned with economics. However, for the bottom-up approach which is focused on regulatory control, it is important to have a dynamic model to test with regard to disturbances.

This master thesis will only focus on the first steps of the top-down approach because of time constraints.

8.2   Top-down Approach

This section will cover the first steps of Skogestad's top-down approach in more detail.

Step 1

The first step of the Skogestad method is to "define the operational objectives" of the plant. This includes formulating a cost function and finding the operational constraints for the process. A cost function represents a scalar function to be minimised, of all variable costs associated with the process. Fixed costs and capital costs are not included as they are considered a prerequisite. The cost function is normally formulated as in equation 8-1 below with dimensions of [currency/time unit] [28]:

$$J = \text{cost feed} + \text{cost utilities} - \text{value products} \qquad (8\text{-}1)$$

In addition to minimising the cost function, optimal operation also has to satisfy certain constraints. This could be purity constraints on the products, safety and environmental constraints and constraints on the process itself.

Step 2

The second step is to find the degrees of freedom, uncover regions of active constraints and determine the optimal operation.

The steady-state degrees of freedom (DOF) can be found using several methods, but the method of valve-counting will be used in this example. The valve-counting method works by counting all physical degrees of freedom and subtracting the ones that have no steady-state effect (such as valves used to control liquid levels) [28]. A distillation column has 6 physical valves: F, D, B, Ref, Vb and the amount of cooling prior to the condenser. Because two valves are used to control the liquid levels in the reflux drum and in the reboiler, and the pressure and feed flowrate are set, there are only two free DOFs.

The active constraints will change with varying disturbances and economic conditions. Hence regions of active constraints must be uncovered for expected disturbances. Every region may have specific optimal conditions. There are two very well-known situations known as *Given Throughput* and *Maximum Throughput* [28]. Given throughput means that the optimal point is a trade-off of the value of the product and the cost of the utilities. Maximum throughput occurs when the prices of the product are sufficiently high, and it is desirable to

produce at maximum capacity. The maximum throughput is often constrained by bottlenecks.

8.3   Post-Processing

It may seem strange to include a chapter on post-processing of the plant here. However, it is necessary to have covered this area before applying the principles of economic *plantwide* optimisation.

To further process the biodiesel to satisfy the standards (these standards are mentioned more thoroughly in the next chapter), it is required to remove excess glycerol, methanol and trace amounts of other contaminants from the product. Because of fear that the reaction may reverse, it is common practice in traditional continuous biodiesel processes to first remove the glycerol. The glycerol has little solubility in the esters and can be removed in a settling tank or a centrifuge. Some of the excess methanol is removed with the glycerol, and some is also removed in an acid cleaning step. The acid scrubbing (normally with sulphuric acid) is used to stabilize the pH, dissolving any soap that may have been formed from the saponification reactions and also removing some methanol. The acid cleaning step is applied for both the biodiesel and the glycerol, following the initial separation. The glycerol phase may also need to go through a flash to remove the last bit of methanol [10, 29].

An example of a process flowsheet is shown below in Figure 8-2. This is a modification of the normal continuous process flowsheet described in the biodiesel literature [10, 29]. As little documentation has been found on the subject, it is purely a visualisation of what the transesterification reactive distillation process might look like. The largest alterations from the regular biodiesel production process are the elimination of a reactor and large methanol-removal units for the glycerol and the biodiesel.

The settling tank and the methanol flash will be implemented in the Matlab DAE model. Even though the acid cleaning units are an essential step in the biodiesel production process, these will not be included.
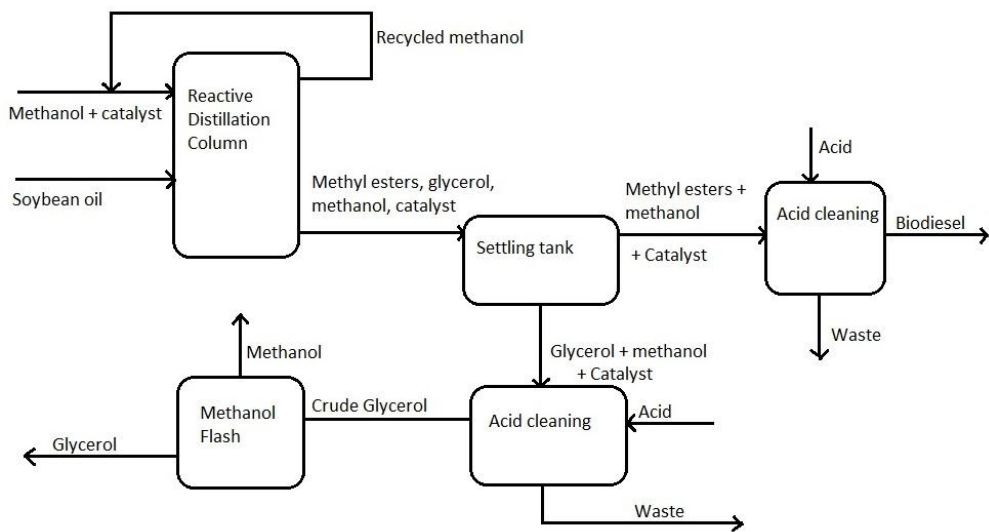
Figure 8-2: Visualised flowsheet of the reactive distillation transesterification process.

Settling Tank

The settling tank will separate the bottoms product based on the densities. Glycerol and biodiesel will form two immiscible phases and hence almost all of the glycerol present will be removed [10]. Research has shown that usually the heavy phase will consist of almost exclusively glycerol and very little biodiesel [30]. Methanol has shown to be approximately 45-50 times more soluble in glycerol than biodiesel at 75 °C with a KOH catalyst (assumed to showcase similar behaviour as NaOH) at a molar ratio between 3:1:3 and 0:1:3 of MetOH:BD:GL [31].

TG, DG and MG are combinations of glycerol and biodiesel esters and therefore vary in their distributions in the heavy and light phases. MG is most similar to glycerol and thus is mostly present in the heavy phase, while the opposite is true for TG. This behaviour is demonstrated in the research by Cernoch et al. [32] where the phase distribution between the glycerol-rich and the ester-rich phase is investigated for ethanol-glycerides.

Based on the research mentioned above, the settling tank is roughly modelled as a split with specifications as summarised in Table 8-2.

Table 8-2: Specifications for the settling tank

Phase	Specifications
Glycerol-rich phase (GP)	$GP_{GL} = 0.999B_{GL}$
	$GP_{BD} = 0.001B_{BD}$
	$GP_{MetOH} = 0.98B_{MetOH}$
Ester-rich phase (EP)	$EP_{TG} = B_{TG}$
	$EP_{DG} = 0.6B_{DG}$
	$EP_{MG} = 0.2B_{MG}$

Methanol Flash

The heavy phase from the settling tank is normally washed and then excess methanol is removed in a flash tank/heater to a purity of approximately 85 weight%. The glycerol can then be sent to a glycerol refinery which using distillation will produce glycerol of high purity [33]. However, the purity of the glycerol produced in this process is likely much higher than the reference amount because of the continuous removal of methanol in the reactive column.

The split factors used in the Matlab model are shown below in Table 8-3 for the crude glycerol (CGL) and the crude methanol (CMet) phases. These split factors were derived from a simulation of a simple flash tank in Hysys assuming ideal behaviour. More details on the flash tank simulation are provided in Appendix F.

Table 8-3: Specifications for the glycerol-methanol flash tank

Phase	Specifications
Crude glycerol (CGL)	$CGL_{GL} = 0.99GP_{GL}$
	$CGL_{BD} = 0.95GP_{BD}$
Crude methanol (CMet)	$CMet_{Met} = 0.99GP_{Met}$
	$CMet_{DG} = 0.02GP_{DG}$
	$CMet_{MG} = 0.01GP_{MG}$

9 The Cost Function and Constraints

This chapter will look at how the cost function was formulated and the constraints associated with it. The cost function is implemented in the cost_c.m file of the DAE Matlab model and is shown in Appendix E.

9.1 The Cost Function

This subsection will look at the prices associated with the products, utilities and raw materials, and how these were used to formulate the cost function.

Prices

From the definition of the cost function it is clear that prices on different items such as steam, cooling water and biodiesel are essential when it comes to its formulation. The prices for relevant raw materials, products and utilities for the biodiesel reactive distillation process are given in Table 9-1. Most of the prices were derived from the article on economic assessment of biodiesel production [34], while the price for soybean oil was found on an internet site applying the reported values of ISTA Mielke GmbH, Oil World [35]. This source was checked against the current prices reported by the CME group [36] to confirm validity.

Table 9-1: Prices of raw materials/utilities relevant for biodiesel production [34]

Item	Specification	Price
Methanol	99.85 %	180 [$/ton]
Biodiesel		600 [$/ton]
Glycerol	92 wt%	1200 [$/ton]
	85 wt%	750 [$/ton]
Cooling water	400 kPa, 6 °C	0.007 [$/m^3]
Low pressure steam (superheated)	450 kPa, 210 °C	6.8 [$/ton]
Soybean oil [35]		338 [$/ton]

The temperatures of the process vary at steady-state between 64.50 °C to approximately 125 °C, hence, only low pressure steam is required. All the prices listed in Table 9-1 are from 2000, and therefore are not up to date. However, the optimisation will mainly look at the relationship between the costs, and thus it did not make sense to update the prices based on price indexes.

Formulating the Cost Function

After having looked at the different prices, the different sections of the cost function are ready to be defined. The cost of the feed is defined as the amount of feed multiplied by the price as shown below for methanol and soybean oil:

$$\text{cost feed} = \left(F_{MetOH} \cdot 180 \right) + \left(F_{TG} \cdot 338 \right) \qquad (9\text{-}1)$$

The value of the products consists of the methanol recycled and the biodiesel and glycerol produced. Because different prices are available based on the purity of glycerol, two equations are given for the two cases:

$$\text{value products} = \left(P_{BD} \cdot 600 \right) + \left(P_{GL} \cdot 1200 \right) + \left(D_{MetOH} \cdot 180 \right) \qquad (9\text{-}2)$$

$$\text{value products} = \left(P_{BD} \cdot 600 \right) + \left(P_{GL} \cdot 750 \right) + \left(D_{MetOH} \cdot 180 \right) \qquad (9\text{-}3)$$

Here the parameter P represents the product flowrate of biodiesel and glycerol in [kg/h], after post-processing. The methanol produced as distillate in equation 9-2 and 9-3 is not really sold, but recycled back to the feed. Hence this part represents the cost saved on recycling the raw material. The cost of utilities is formulated in a similar manner as shown in equation 9-4:

$$\text{cost utilities} = \left(CW \cdot 0.007 \right) + \left(LPS \cdot 6.8 \right) \qquad (9\text{-}4)$$

In equation 9-4, the parameters CW and LPS represent the amount of cooling water [m^3/h] and low pressure steam [ton/h] respectively. Because energy balances are not included in the Matlab model, the utilities are not given directly. Therefore, correlations were found between vapour flowrates and required utilities in the Hysys model, which are shown below as equations 9-5 and 9-6. Details on the derivation of these equations are available in Appendix G.

$$LPS = \frac{2316.9 \cdot \exp\left(0.0037 \cdot Vb \right)}{1000} \qquad (9\text{-}5)$$

$$CW = \frac{1714.3 \cdot V_1 + 6476.8}{60 \cdot 1000} \qquad (9\text{-}6)$$

9.2 Constraints

The constraints posed on the process by quality demands and operational criteria are discussed and formulated in this subsection.

Biodiesel Quality

The standards for biodiesel ensures that the fuel is safe to use, and reduces the possibility of it having harmful effects on the automotive engine [37]. There are separate standards for different biodiesel plants and engine manufacturers [10]. These standards normally contain restrictions on the amount of alcohol, free fatty acids, catalyst, unconverted soybean oil, sulphur and glycerol the biodiesel can contain. The standards also specifies acceptable values for most physical properties of the biodiesel such as cloud point, viscosity, flashpoint and many more [37]. Table 9-2 below gives an overview of the specifications in the standard for Europe for fatty-acid methyl ester (FAME) biodiesel [10]. The specifications concerning the physical properties were not included in this table.

Table 9-2: The International Standards for FAME Biodiesel [10]

Specification		European standard, EN 14214:2003
Water	[mg/kg]	500 max
Total contamination	[mg/kg]	24 max
Methanol	[weight%]	0.20 max
Ester content	[weight%]	96.5 min
Monoglyceride	[weight%]	0.8 max
Diglyceride	[weight%]	0.2 max
Triglyceride	[weight%]	0.2 max
Free Glycerol	[weight%]	0.02 max
Total Glycerol	[weight%]	0.25 max

Water and other contaminations are irrelevant for this model considering that the saponification and hydrolysis reactions have not been included, and it was assumed that only pure linoleic esters were used for the feed.

Operational Constraints

Operational constraints on the process normally include maximum and minimum flowrates such as for the vapour boilup and liquid reflux. This is to prevent "uncontrollable" states such as flooding or weeping in the column. Flooding was estimated as +50% of the nominal vapour flowrate of the column, which was calculated as approximately 8.5 kmol/min. Because the vapour flowrate in the Matlab model below feed is set by the vapour boilup, this is a constraint directly

on the Vb. The minimum constraint on the vapour boilup was set as 0.01 kmol/min, to avoid it from reaching 0.

Often, there are also operational constraints on temperatures, pressures and compositions. For this system the pressure is already set at 1 atm, but there are limitations on the temperature; as glycerol and biodiesel can decompose at 150 °C and 250 °C respectively [1]. The compositions and molar holdups have a lower boundary of 0 as their operational constraint. All the constraints are summarised in Table 9-3 below.

Table 9-3: Overview of the constraints applied to the process

Type of constraint	Constraint	Number
Fundamental	$x > 0$	1
	$HR > 0$	2
	$T > 0$	3
	$D > 0$	4
	$Ref > 0$	5
	$B > 0$	6
Decomposition	$T_{Reb} < 423.15$	7
Quality of biodiesel	$xEP_{MetOH} < 0.002$	8
	$xEP_{TG} < 0.002$	9
	$xEP_{DG} < 0.002$	10
	$xEP_{MG} < 0.008$	11
	$xEP_{GL} < 0.0002$	12
	$xEP_{TG} + xEP_{DG} + xEP_{MG} + xEP_{GL} < 0.002$	13
	$xEP_{BD} > 0.965$	14
Quality of glycerol	$xCGL > 0.92$	15a
	$xCGL > 0.85$	15b
Operational constraint	$Vb > 0.01$	16
	$Vb < 8.5$	17

Each constraint of Table 9-3 was also given a number. This number was introduced in order to quickly recognise the active constraints in various scenarios during the optimisation.

10 Optimisation – Results

This chapter presents the results found by optimisation of the system. The optimisation is inspired by Skogestad's stepwise method [27] as well as the optimisation performed by Simasatitkul et al [1]. The optimisation code is part of the DAE-Matlab model shown in Appendix E, and finds the minimum value of the cost function by applying the fmincon function in Matlab and interior-point algorithm.

Post-processing
The implementation of the post-processing equipment prior to optimisation gave the following results for the glycerol-rich phase (GP), ester-rich phase (EP) from the settling tank and the crude glycerol (CGL) from the flash tank as shown in Table 10-1.

Table 10-1: The composition of the GP, EP and CGL prior to optimisation

	EP [wt%]	GP [wt%]	CGL [wt%]
Methanol	0.11	33.32	0.50
Trilinolein	1.81	0.00	0.00
Dilinolein	1.55	2.38	3.52
Monolinolein	0.38	3.55	5.31
Free Glycerol	0.01	60.15	89.82
Biodiesel	96.14	0.59	0.85
Total Glycerol	3.75	-	-

From Table 10-1 one can see that there are some issues with the results, as EP does not satisfy the biodiesel quality constraints. The total amount of glycerol is too high; specifically the intermediates TG, DG and MG. This is probably due to the conversion and yield being too low. The glycerol recovered after the flash, CGL, satisfies the quality constraint 15b with a concentration of more than 85 wt% glycerol.

10.1 Finding Initial Conditions

From the analysis of the plant at starting conditions it became clear that the conversion in the column and the purity of the biodiesel is too low. The optimisation on the reactive distillation column by Simasatitkul et al.[1] was performed on the basis of maximising conversion, yield and purity of biodiesel. This is closely related to the goal of the plantwide optimisation and may provide a useful starting point for iteration. Therefore, this section will describe the

optimal points found by Simasatitkul et al.'s research, and the results obtained after implementation in the Matlab model.

The optimisation by Simasatitkul et al.[1] looked at how the number of trays, feed temperature, feed ratio and reboiler heat duty affected the yield and conversion of the biodiesel. The definition of the yield and conversion are shown in equations 10-1 and 10-2 below:

$$\text{Conversion} = \frac{F_{TG} - B_{TG}}{F_{TG}} \qquad (10\text{-}1)$$

$$\text{Yield} = \frac{B_{BD}}{3 \cdot F_{TG}} \qquad (10\text{-}2)$$

The optimal values reported from Simasatitkul et al. [1]'s research are shown in Table 10-2 below with corresponding values of conversion and yield.

Table 10-2: The optimal values found of parameters by Simasatitkul et al [1].

Parameter	Optimal value	Conversion	Yield
Feed molar ratio (MetOH:TG)	9 or higher[1]	close to 100 %	99 %
Feed temperature	25 °C	98.45 %	95.8 %
Reboiler heat duty	$1.6 \cdot 10^7$ kJ/h	98.7 %	96.6 %
Number of reactive stages	3 or more	98.45 %	95.8 %

Feed Temperature
Because the Matlab model does not include energy balances, the only way to change the feed temperature in the Matlab model is by changing the feed liquid fraction. The feed liquid fraction q is defined by equation 10-3:

$$q = \frac{H_V - H_F}{H_V - H_L} \qquad (10\text{-}3)$$

---

[1] The optimum feed molar ratio reported is from optimisation based only on achieving high conversion and yield, not product purity. This is because the product purity is defined differently when considering post-processing.

From equation 10-3 one can see that q depends on the enthalpies of the feed at dew point (H_V), bubble point (H_L) and at the feed conditions (H_F) [16]. The value of the feed liquid fraction was calculated for the specialisation project for various temperatures and a feed ratio of 6, which was equal to 1.064. However, because the feed liquid fraction varies with the feed molar fraction, this value is not correct for the entire range of optimisation as performed here.

Reboiler Heat Duty
The optimal reboiler heat duty with regard to yield and conversion found by Simasatitkul et al. of $1.6*10^7$ kJ/h was applied to the Hysys model. The corresponding vapour boilup of 309.4 kmol/h was then used as input to the Matlab model. The optimal heat duty found by Simasatitkul et al, was actually $2*10^7$ kJ/h, but this gives such a high reboiler temperature that glycerol would be in danger of decomposing, (violation of constraint 7).

Number of Reactive Stages
Decreasing the number of reactive stages does not increase the conversion and yield, but rather finds the smallest number of stages resulting in the same conversion as before. The Hysys model only needed 3 stages (5 counting the reboiler and condenser drum), but the Matlab model might need less as chemical equilibrium is achieved faster. Therefore the number of reactive stages was plotted against the conversion for the Matlab model as shown below in Figure 10-1.
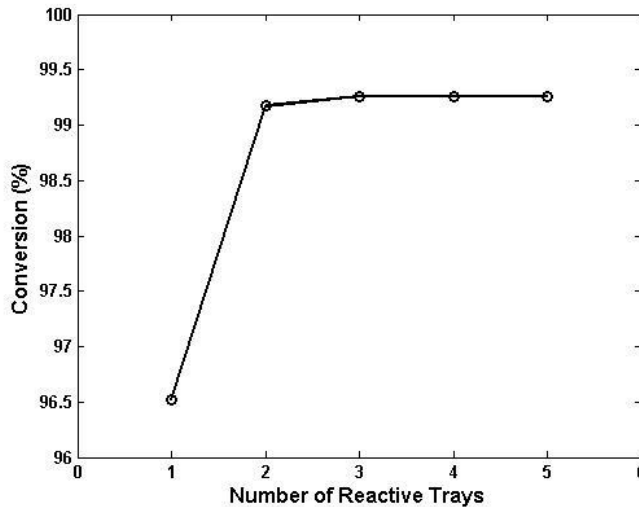
Figure 10-1: The number of reactive stages plotted against conversion of TG.

From Figure 10-1, one can see that three reactive stages are necessary for the Matlab model, in agreement with the Hysys results. Increasing the number of reactive stages beyond this does not increase the overall conversion and is therefore unnecessary. The graph for the yield plotted against the number of reactive stages necessary gave the same results, and hence this graph was not included. However, it should be noted that the yield is slightly lower in value compared to the conversion and gives a value of 97.62% from 3 reactive stages and above.

Feed Molar Ratio
The feed molar ratio of 9 or higher was optimal for the Hysys model, giving a conversion of close to 100% and a yield of 99%. However, it gave a very low purity of biodiesel, and hence a ratio of 2.5 was used instead. This is not as big of a concern with the Matlab model as post-processing equipment have been implemented to separate the methanol from the esters. A bigger problem is satisfying the quality constraints with regard to glycerides as could be seen from Table 10-1. A low feed ratio, and hence a low conversion will not satisfy the quality constraints on the total amount of glycerides in the biodiesel, so it is necessary to increase the conversion. The molar feed ratio plotted against conversion is shown in Figure 10-2 below.
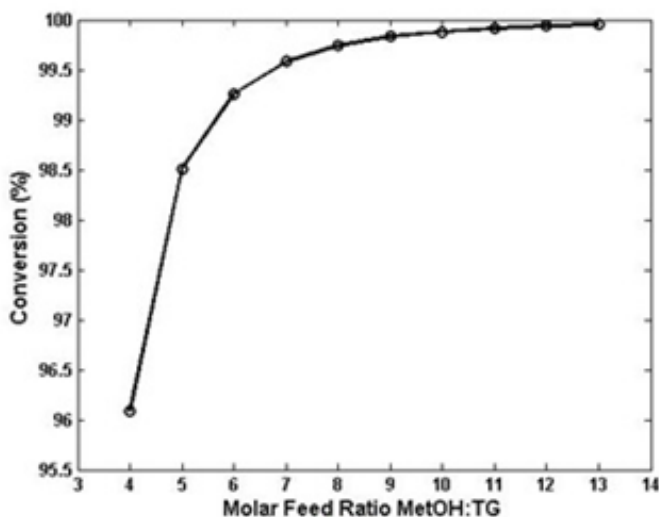
Figure 10-2: The molar feed ratio plotted against conversion

From the plot it is clear that the conversion increases with increasing molar feed ratio as with the Hysys model. Because the glycerides in the intermediates and in the reactant oil cannot be removed at a later point from the biodiesel, the standards apply directly to the values of glycerides in the EP. However, some methanol will be removed during the washing and drying stages [10], so if the value of methanol is a little higher in the EP than stated in the standards, this isn't a big problem. Based on this, the molar feed ratio applied as an initial input before starting the optimisation was 13, with corresponding feed of 325:25 of MetOH:TG.

The yield is slightly lower in percentage compared to the conversion, but the shape of the graph is very similar compared to Figure 10-2. For this reason, this graph was not included here, as it does not present any new information.

Results
The resulting compositions of EP, GP and CGL following the shift in feed molar ratio, number of reactive trays and feed temperature are summarised in Table 10-3 below:

Table 10-3: Compositions of EP, GP and CGL subject to new conditions

	EP [wt%]	GP [wt%]	CGL [wt%]
Methanol	0.55	71.70	2.50
Trilinolein	0.05	0.00	0.00
Dilinolein	0.11	0.19	0.64
Monolinolein	0.05	0.63	2.18
Glycerol	0.01	27.21	93.81
Biodiesel	99.22	0.26	0.87
Total Glycerol	0.22	-	-

The change in the results is large, and now almost all of the quality constraints are met on both the glycerol and the biodiesel products. Even though the amount of methanol in the column is increased, the purity of CGL is also increased, and fulfils the criteria of high purity glycerol with a higher price (constraint 15a). The only issue with these results is that the biodiesel contains too much methanol. But as stated earlier, it is possible to remove this in later stages.

10.2 Plantwide Optimisation

This subsection will include optimisation of the process at different scenarios. The first optimisation occurs at the initial conditions derived in the last subsection.

Initial Optimisation

Optimisation was performed by applying the cost function as defined in equation 8-1, using the vapour boilup and the liquid reflux as the values to be optimised. At first the quality constraints on methanol were not included, and operational constraints did not consider weeping and flooding points. However, the results from the first sets of optimisation were already satisfying the operational constraints, so there was no problem with introducing constraints also on methanol. The results are shown below in Table 10-4 and Table 10-5.

Table 10-4: Flowrates and cost function as results of optimisation

Parameter	Value	Dimension
Cost Function	-788.56	[$/min]
Vapour boilup	8.50	[kmol/min]
Liquid Reflux	4.42	[kmol/min]
Distillate	3.70	[kmol/min]
Bottoms	2.13	[kmol/min]

Table 10-5: Compositions of EP, GP and CGL after first optimisation

	EP [wt%]	GP [wt%]	CGL [wt%]
Methanol	0.08	27.62	0.38
Trilinolein	0.06	0.00	0.00
Dilinolein	0.11	0.52	0.71
Monolinolein	0.06	1.72	2.36
Glycerol	0.01	69.46	95.65
Biodiesel	99.67	0.68	0.90
Total Glycerol	0.25	-	-

From Table 10-5 one can see that all quality constraints are satisfied, and the crude glycerol is even overpurified. The active constraints for the initial scenario are: 13 (maximum amount of total glycerol in biodiesel) and 17 (maximum vapour boilup).

Increased Energy Costs

One of the most common scenarios is to test the effect of reducing/increasing the energy prices. However, implementing a change in energy prices lead to no other change than in the value of the cost function itself. The amount of vapour boilup and liquid reflux stayed constant, even when the costs were increased by many times the original value. This makes sense when one considers that the utility costs are only dependent on the behaviour of the reactive distillation column, as other utilities during post-processing were not included in this simple model. Also, the utilities are defined in such a way that they are both dependent on the vapour flowrates which also makes them dependent on each other.

10.3 Active Constraint Regions

From section 10.1 it became apparent that increasing the feed ratio increased the conversion of the system. Because of the strict constraints on the biodiesel quality, it is interesting to look at how changing the feed flowrate and the feed ratio will change the active constraints.

By considering these two disturbances at the same time, the applicable regions of active constraints were uncovered for the system. The feed flowrate was varied between 300 and 400 kmol/h as the column was designed for a flowrate of 350 kmol/h. The feed ratio was varied for a larger range, but only gave feasible solutions between 9 and 21, though this varied with feed flowrate.

The variation of the cost function with the feed ratio and flowrate is given below in Figure 10-3.
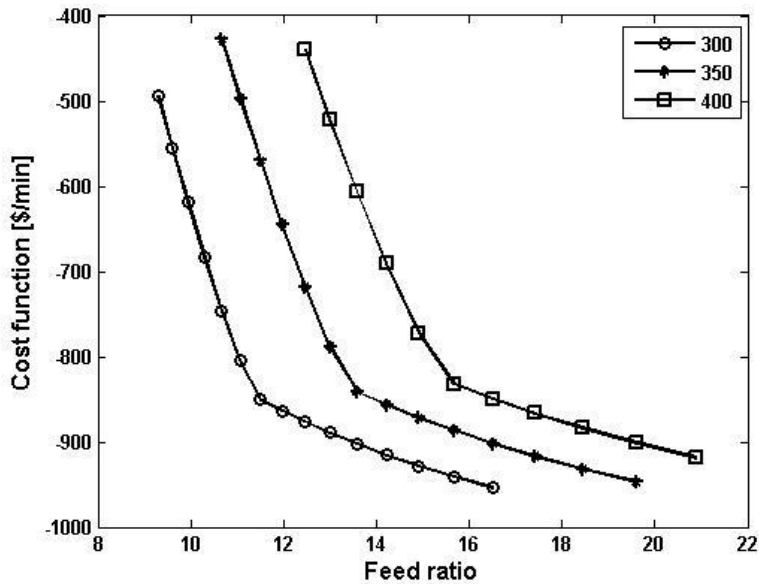
Figure 10-3: The variation of the cost function with feed ratio and feed flowrate.

From Figure 10-3 it is easily seen that there are two distinct regions of operation in the reactive distillation column, as is marked by the change in slope of the graphs. One can also see that the cost function decreases with increasing feed ratio, meaning that optimal operation favours high feed ratios. Lower feed flowrates also yielded lower costs, but this is largely contributed to how the utilities are estimated in the model.

Based on the simulation results, one was also able to produce an outline of the active constraint regions with the feed rate and ratio as the two disturbances. The active constraint diagram is shown below in Figure 10-4. However, one should note that the lines separating areas of active constraints are not absolute. In the areas where constraints transition from active to non-active or the other way around, the optimisation is not stable and won't always find a solution, even though a solution might exist. This especially corresponds to the lines separating region 1&2 and 3&4 as regions 1 and 4 represent infeasible regions for the Matlab optimisation procedures.
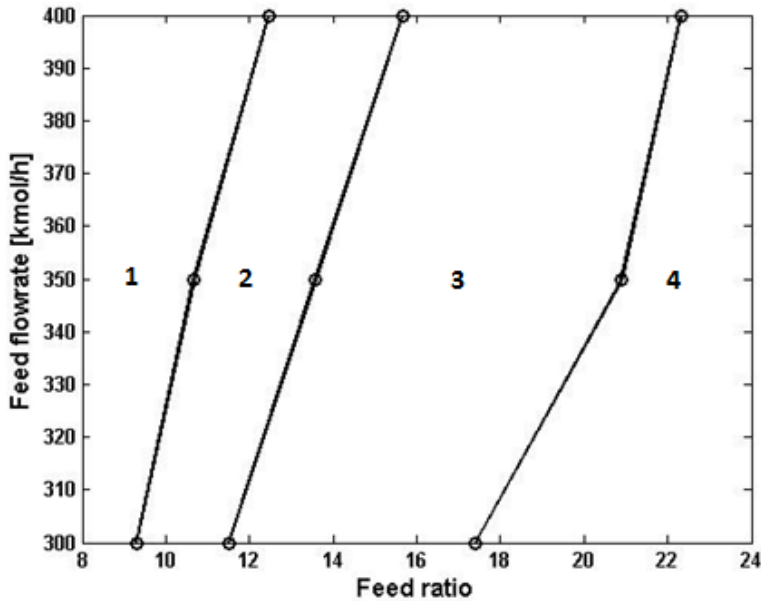
Figure 10-4: Active constraint regions found by varying the feed ratio and flowrate

The regions uncovered through simulation are each given a specific number, and are described in more detail in Table 10-6 below.

Table 10-6: Overview of the active constraint regions

Region	Active constraints	Comment
1	None	This region cannot meet the quality constraints for the biodiesel product
2	Constraints 13 & 17; maximum amount of total glycerol in biodiesel and maximum vapour boilup.	This region corresponds to the steeper slope in the cost function graph, and is less profitable than the following region.
3	Constraints 7 & 13; maximum temperature in reboiler and maximum amount of total glycerol in biodiesel.	This region corresponds to the slope with the highest profits/lowest costs.
4	Unknown	The optimisation could not be solved by Matlab beyond this point.

From Table 10-6 and Figure 10-4 one can see that the neighbouring regions 2 and 3 have the same number of active constraints. This probably means that the two constraints are related to the same unit operation [38], which makes sense for this system as the "dominating" unit in optimisation is the reactive distillation column. The constraint on the maximum amount of total glycerol in the biodiesel also seems reasonable from an economic perspective, as there is little economic gain in overpurification of the product. The constraint on maximum vapour boilup also naturally transcends into a constraint on the maximum temperature in the reboiler, as this becomes the limiting factor. The temperature in the reboiler increases with increasing feed ratio.

Region 4

A clear issue with the active constraint region 4 is that the system is no longer able to find a feasible optimum after a specific feed ratio, dependent on the feed flowrate applied. Two different methods were attempted in order to solve this problem; continuously updating the initial values and introducing a soft constraint. The continuous update of initial values occurred by applying the results of simulation obtained by neighbouring regions as initial values for the subsequent simulation, but didn't have very much effect.

The problem might occur because Matlab has a problem with overcoming constraints in this transition region. Hence, a soft constraint can be introduced one by one on the active constraints that are possibly inflicted. A soft constraint is defined as the difference between the constraint and the simulation value, and is denoted by an ε:

$$\varepsilon = GL_{tot} - 0.0025 \qquad (10\text{-}4)$$

The cost function is penalised with an extra cost when the soft constraint is broken, according to the modified cost function shown below in equation 10-5. The price put on the soft constraint, α_E can be tuned to give desirable loss.

$$J = \text{cost feed} + \text{cost utilities} - \text{value products} + \left(\alpha_E \cdot \varepsilon^2\right) \quad (10\text{-}5)$$

A soft constraint on the total glycerol content in the product was applied with a very high penalty price of 30,000,000, as the constraints on the total glycerol are very strict and the constraint value is very small. However, this gave little results with regard to learning more about region 4, as the region remained infeasible. On the other hand, it was found that the reflux continuously decreased as the feed ratio approached the region 4 border.

Therefore, the next soft constraint to be tested would have been the minimum flowrate of liquid reflux, but this would give mean a negative flowrate and hence did not provide very reasonable results. The maximum temperature in the reboiler was also tested as a soft constraint, with a penalty price of 50 but did not provide any more information on region 4.

Region 4 still remains unfamiliar to us, as Matlab fmincon interior-point algorithm won't solve the optimisation beyond this boundary. It may have certain active constraints, or be infeasible for solution. What is known is that feed ratios above 20 is outside the range used for regular biodiesel production, and hence the region may be unstable or costly.

Optimal Operation
The optimal point for operation of this system was found as the minimisation of the cost function, which was visualised in Figure 10-3. The optimum point is situated at the boundary between region 3 and 4, which constitutes a problem as we don't know the active constraints of region 4 or if there are any at all. In actual implementation of setpoints for regulation, one would need to back-off slightly from this optimal point to account for errors in measurements and implementation.

In any case, the actual optimum represents the second segment of the cost function, equal to region 3 where the active constraints are 7 and 13, on the very boundary of region 4. The cost function for the optimal point equals $ -950 per minute with the corresponding setpoints of flowrates as given in Table 10-7 below.

Table 10-7: Optimal operation of the reactive distillation column

Parameter	Value	Dimension
Feed Ratio	333:17	[kmol/kmol]
Cost Function	-949.98	[$/min]
Vapour boilup	5.93	[kmol/min]
Liquid Reflux	1.12	[kmol/min]
Distillate	4.43	[kmol/min]
Bottoms	1.40	[kmol/min]

10.4 Possible Control Structure

A detailed analysis of the optimum control structure is outside the scope of this thesis. However, based on available research and the active constraint regions uncovered in the previous subsection, possible control structures for the reactive distillation system will be briefly discussed here.

All the feasible regions for operation have two active constraints. The active constraints should always be controlled, as the optimum with regard to these variables is not flat [39]. Back-off is defined as the economic loss achieved by moving the optimal operating point away from the optimum to gain better control. Hence, tight control of the active constraints is desirable in order to get as small an economic loss as possible. After controlling the active constraints, it is normal to look for self-optimising constraints to control next. But seeing as there are only 2 free DOF's available, these will go only to controlling the active constraints.

The vapour boilup is indirectly controlled by the heat input by the warm utility, so it makes sense to use the heat utility flowrate as the manipulated variable (MV) here. The temperature in the reboiler is also determined by the heat supplied by the low pressure steam. Hence, one could easily pair a temperature sensor with the flowrate of LPS as the MV.

The intuitive measurement for the quality constraint on the biodiesel in terms of total glycerol allowed is to use a composition measurement. However, these are generally quite slow [40]. Perhaps a better way to ensure the correct composition would be to integrate a temperature loop in the midst of the column as discussed in the article *The do's and don'ts of distillation column control* by Skogestad (2007) [40]. Integrating a temperature loop will also help stabilise the column, reducing sensitivity to disturbances and making the column behave in a more linear manner, while giving a faster response than e.g. a composition controller would have provided. As a rule-of-thumb, the position of the temperature measurement is normally set in the column close to the end of the valuable product. Temperatures chosen for control loops should be sensitive to inputs and avoid positions where the temperature change from one stage to another is very small, and the temperature measurement should be paired with one of the flowrates close to the bottom of the column.

11 Discussion

This chapter provides a discussion of the results achieved in the master thesis. The discussion is divided into two parts to follow the same structural pattern as the rest of the report; first the modifications of the model are discussed, then the results of the optimisation.

11.1 Modification of Model

The modification of the model included updating the physical properties, such as the boiling points and vapour pressures to the values used in the research by Simasatitkul et al[1]. The exchange occurred because it was desirable to have the same basis for comparison of the Hysys model and the Matlab model. However, the boiling points varied by as much as 300 °C for trilinolein, and the vapour pressures for the esters found from Hysys were all outside of the range of the process temperatures. These physical properties should all be validated, and ideally exchanged with experimental values of higher accuracy.

One of the strange problems during this part of the thesis was discovering that the published results did not match the results achieved from the Hysys model; as there were noticeable differences in both temperature and composition estimations. There may be many reasons for this behaviour, one of them being that the Hysys reactive distillation column is complex and requires good initial values in order to run. Hence it could be that having different initial values resulted in these deviations, or that a different version of Hysys/Unisim was used by Simasatitkul et al.[1] compared to in this thesis.

After the modifications were made, the composition profiles of the Matlab and Hysys simulations were very similar. However, there are still some small differences between them; one of the most prominent being the shape of the column profiles. While the Hysys model has smooth transitions, the Matlab model has clearly defined segments. One reason for the deviation could be the interaction of the more complex dynamics and energy balances in the Hysys model, resulting in a smoother graph. However, it was observed that a significant increase of the flowrate or decrease of the weir heights achieved a similar shape in the Matlab profiles, suggesting that this may be due to larger molar holdups in the Matlab model. Hence, if it is desirable to achieve a smoother shape in the Matlab profiles, one can alter the design by changing the feed flowrates and the weir heights (although the large changes this would

require would change the design of the column from as recommended by Chemical Engineering Design literature [17]).

Temperature Estimation

Temperature estimation proved difficult with the Matlab model, as no empirical method gave very accurate results. Even an empirical method tailored to the specific problem could not provide a decent estimate for the column, reboiler and condenser at the same time. However, the tailored empirical method did present the exact same shape of the temperature profile as the Hysys model. The UNIQUAC method was also implemented and was expected to give very good results as UNIQUAC is also used in the Hysys model. However, while it provided the most accurate temperature estimations yet, the shape of the temperature profile does not match the shape of the empirical method or the results from Hysys. The deviation in the shape is mostly due to an unexpected surge in temperature on the first tray, which is not present in the Hysys results. The actual deviation at this point is approximately 5°C. The deviation may be due to the interference of the exothermic chemical reaction on this step, as this is not included in the Matlab model. Tweaking the temperature estimation method, trying different thermodynamic models or introducing energy balances may reduce this deviation.

Dynamic Behaviour

Introduction of disturbances yielded similar responses in the Matlab and Hysys models. An increase in feed ratio of the soybean oil resulted in an increase in temperature and reduction of methanol in the column while the reverse occurred for an increase in the feed ratio of methanol. Increasing the feed also resulted in a small increase in the temperature as the vapour boilup was kept constant.

The Matlab model's dynamic behaviour in response to a disturbance in the feed ratio was shown in Figure 7-5. The reboiler and plates further down in the column used longer to obtain new steady-state values, while very short times were necessary for plates close to the feed tray. Also, in the reboiler the compositions had an overshoot before reaching the new steady states. This could be explained as a change in feed composition will also include a change in the volumetric feed rate as the oil represents bigger molecules. Hence the liquid flowrate inside the column will momentarily increase and will consist mainly of methanol, leading to a quick surge of methanol in the reboiler. This is quickly reduced and steady states are then obtained. The dynamics are fast compared to ordinary distillation columns, but it is unknown how this behaviour correlates with other reactive distillation columns.

Two different dynamic Matlab models were constructed during this thesis: a Simulink model and a DAE model. The Simulink model is very slow compared to the DAE model due to nested for-loops, but provides the same results. The DAE model also has a steady-state version, and for this reason the DAE system was the model of choice for optimisation and testing with regard to disturbances. However, the Simulink model is a good option to have if it is desirable to later construct a control system, as it presents a graphic interpretation of the process.

11.2 Optimisation

The optimisation of the process occurred by applying the economic plantwide control of Skogestad to the process. Seeing as the focus of the master thesis was on the reactive distillation column, the settling tank and methanol flash were modelled as simple splits. However, one should note that this is not a very accurate way of presenting the plant, as both the flash and the settling tank split factors will vary with disturbances in feed rate, temperature and compositions. Also, the split factors used for the settling tank were based on numbers from research covering biodiesel production under different conditions to what is considered in this thesis. The split factors for the methanol flash was also obtained assuming ideal behaviour, which may not be true, especially if we consider the correlation of the system to Raoult's law in subsection 5.2.

Cost-function

The flooding and weeping constraints on the distillation column are actually functions dependent on liquid and vapour flowrates and the ratios between them. Because these values are always changing during optimisation, it was decided to only use a rough estimate. However, this is not very accurate, as the flooding and weeping points are actually functions dependent on feed composition, flowrates and the ratio between liquid and vapour flowrates within the column.

The amount of utilities required in the reactive distillation column was estimated from equations dependent on the vapour flowrates. These equations were found by simulating the Hysys model for various duties and fitting a trend-line to the plotted data. However, the amount of utilities required will also vary with varying feed rates and compositions. Hence, to get more accurate estimations for the utilities required, it would be necessary to either extend the equations to also account for changes in the feed or to include energy balances. More, the way that the utilities are estimated mean that they are both dependent on the vapour flowrates within the column, which again means that they are co-dependent on each other.

Optimisation

The feed temperature was altered by simply changing the feed liquid fraction. However, the feed liquid fraction calculated is actually only valid for the feed fraction of 6. Hence, the feed liquid fraction is actually a function of the feed composition, and thus this is not a very accurate way of estimating this number. Having more methanol in the feed would actually decrease the feed fraction slightly. Due to the fact that the feed temperature does not have a very large effect on the overall conversion, compared to for example changing the feed ratio or the vapour boilup, finding valid values of the feed liquid fraction was not a priority with regard to time constraints.

One of the most obvious problems with the optimisation was concerning the last constraint region as feed ratio increased above 20. For this region Matlab struggled with finding a feasible optimum. Several methods were tested in order to approach this boundary, such as introducing soft constraints on the active constraints near the boundary and using continuously updated initial values. Though both of these methods gained inconclusive results, it may appear that the last region includes a reduction in the reflux to the point of it reaching minimum value, but this is very uncertain. This problem should be looked at in more detail to be able to draw conclusions.

The economic scenario did not have a large effect on the system even if the price on utilities was multiplied by a factor of 5. This is because of the way that the utilities in the distillation model are defined. As mentioned earlier, adjusting one utility without simultaneously setting the other is not possible, as they are both dependent on the vapour flowrates. The hot utility is also determined by an active constraint in both feasible active constraint regions, which means that there is no "wriggle-room" for altering the utilities even if prices are increased many times. Besides from defining the utilities differently, modelling the post-processing equipment in more detail could provide changes due to alterations in prices. More detailed models of the flash or settling tank would provide more degrees of freedom, and may even uncover more active constraint regions.

The cost function also gives a very high profit per time unit of almost $1000 per minute. If this was accurate it would be fantastic, but this is highly unlikely. Of course we must also remember that other utilities such as heat exchangers, pumps, and cleaning/scrubbing sections were not included in the cost function. These would increase the costs to reach a more realistic level of the cost function.

12 Conclusion and Further Work

This last section concludes the thesis and gives suggestions as to further work on the model, as well as optimisation and design of a control structure.

12.1 Conclusion

The reactive distillation model produced during the specialisation project and modified in this thesis correlates well with the more complex behaviour of the Hysys model as provided by Professor Amornchai Arpornwichanop and used in the research by Simasatitkul et al[1]. The UNIQUAC model gives decent estimates for the temperature profiles besides from a slight overestimation on the temperature of the feed tray. The composition profile is also a good match, albeit having a slightly more stiff shape than the Hysys' profile as chemical equilibrium is achieved faster in the Matlab version. The dynamics used behave as expected, although a bit fast compared to the behaviour of regular distillation columns.

Optimisation of the model occurred through application of the first part of the Skogestad top-down/bottom-up economic plantwide control method. The model was optimised first with regard to the results found in the research by Simasatitkul et al, and then later by application of the cost function and varying the vapour boilup and liquid reflux. The model was found to have four distinct active constraint regions when subject to disturbances in the feed ratio and flowrate. The first region occurred at low feed molar ratios and could not satisfy the constraints posed on the process. The second and third region had two active constraints, with the maximum impurity of glycerol in the product being common for both. The second constraints consisted of the flooding constraint for section 2 and maximum temperature in reboiler to avoid decomposition for section 3. Increasing the feed ratios above that of region 3 yielded region 4 which could not be solved by the Matlab fmincon interior-point algorithm, even with implemented soft-constraints. The feed ratios of region 4 also lie outside of normal biodiesel production ranges.

The optimal point for operation, with the lowest cost function of -950 $/minute was found to be on the boundary of region 3 and 4, with a feed of 333:17 kmol/h of Methanol:TG. The corresponding setpoints for the liquid reflux, vapour boilup, distillate and bottoms molar flowrates per minute equal 1.12, 5.93, 4.43 and 1.40 respectively. Please not however, that costs regarding extra heat exchangers, pumps and cleaning sections have not been included so the actual cost function is much higher.

12.2 Further Work

Renewable fuels is an interesting topic in these times where the energy demands are increasing and the usage of fossil fuel reserves is causing a lot of environmental concerns. Biodiesel is a potential substitute for regular diesel, but some concerns are directed towards how the sources of biodiesel compete with crop space for food. There is also the problem that much-used biodiesel sources such as palm oil and soybean oil need huge areas to produce little oil. Hence, this model was created with that in mind; that it would be great to extend the usage for the model with other types of oils that do not pose these types of issues on the food market and crop space. Current research is looking into production of biodiesel from algae, which looks very promising, but is a great deal further into the future than current commercial applications.

Modification of Model

The Matlab and Hysys model showcase similar behaviours and follow the same trends, although not being entirely identical. However, some questions have been posed concerning much of the data on which the Hysys model was based. There were very large deviations on boiling points for the linoleic esters, suggesting that these may be best found from experimental observation. The same thing goes for the vapour pressure equations of the esters which were merely estimated by UNIFAC, and not really valid for the temperature intervals of the process. These vapour pressures need to be determined using a more accurate method or through experiments, as the validity is very low.

The catalyst used in the process is homogeneous NaOH, which has a low purchase cost, but has higher environmental and cleaning consequences than the other alternatives. It would be very interesting to test the model for another catalyst. This would also enable the model to be tested for other kinetic data.

Optimisation

The focus of this thesis was on the reactive distillation column, and therefore the settling tank for separation between glycerol and esters was modelled as a simple split. In reality, the split factors will vary with compositions in the tank, and hence this is not an accurate representation. For more detailed data of the biodiesel production process it would be interesting to model the entire plant in Hysys/Matlab including all heat exchangers, washers, dryers and flash/distillation columns. The recycle loop with methanol should also be closed, and the catalyst should be included in the flowrates. This would influence both the dynamics and the optimisation, and give a more realistic representation of the process. The optimisation in general would probably get another set of active

constraint regions and we would most likely get to see how a shift in prices would uncover the given throughput and maximum throughput situations.

Another issue with the very simple model representation is that no costs related to other utilities, heat exchangers and pumps were included in the optimisation. These would normally be part of the plant, as the settling tank and methanol flash will normally occur at lower temperatures compared to the reactive distillation column. Including these additional utilities would greatly improve the validity of the cost function, and increase the costs to reach a more reasonable level.

Lastly, an obvious step of the further work would be to continue using Skogestad's seven step plantwide control method to arrive at the finished control structure for the biodiesel reactive distillation process. This includes finishing the top-down steps and finding the ideal combinations of measurements and manipulated variables, as well as looking into the bottom-up section of stabilising control. It could be interesting to look into how a good stabilising control structure of a reactive distillation column compares with that of a regular distillation column.

13 Nomenclature

Table 13-1: Abbreviations and Acronyms

AIChE	American Institute for Chemical Engineers
BD	Biodiesel, here also known as methyl linoleate
CAPEX	Capital costs
CGL	Crude glycerol: recovered after flash tank.
CMet	Crude methanol: recovered after flash tank
CV	Controlled variable
DAE	Differential algebraic equation
DG	Diglyceride, but usually refers to dilinolein
DOF	Degrees of freedom
EP	Ester-rich phase after settling tank.
FAME	Fatty acid methyl ester
FFA	Free fatty acids
GP	Glycerol-rich phase after settling tank
MetOH	Methanol
MG	Monoglyceride, but usually refers to monolinolein
MTBE	Methyl tert-butyl-ether
NC	Number of compounds
NT	Number of trays
OPEX	Operational costs
RD	Reactive distillation
RTO	Real time optimisation
TAG	Triacylglycerol
TG	Trilinolein
TPM	Throughput Manipulator
UNIQUAC	Universal quasichemical method

Table 13-2: Greek symbols

α_i	Relative volatility for component i.
α_E	The penalty of breaking the soft constraint
ε	Soft constraint
γ_i	Liquid phase activity coefficient
γ_i^C	Combinatorial contribution of the activity coefficient
γ_i^R	Residual contribution of the activity coefficient
ρ_L	Liquid density [kg/m^3]
ρ_l	Liquid molar density [kmol/m^3]
τ_{ij}	Empirical coefficient connected to the residual contribution.

Table 13-3: Symbols

A_i	Arrhenius constant for specific reaction [m^3/kmol h]
A_t	Area of the tray [m^2]
B	Bottoms flowrate [kmol/h]
CW	Cooling water [m^3/h]
D	Distillate flowrate [kmol/h]
E_a	Activation energy [kJ/mol]
F	Feed flowrate [kmol/h]
F_i	Surface area fraction per mixture mole fraction
H_F	Enthalpy of feed at feed conditions [kJ/kmol]
H_L	Enthalpy of feed at bubble point [kJ/kmol]
h_{ow}	Height over weir [m]
HR	Molar holdup [kmol]
H_V	Enthalpy of feed at dew point [kJ/kmol]
h_w	Height of weir [m]
J	Cost function [$/min]
K	Constant of the Francis weir equation
k_i	Rate constant for transesterification equations
L	Liquid flowrate [kmol/h]
L_w	Liquid mass flowrate [kg/s]
l_w	Length of weir [m]
LPS	Low pressure steam [ton/h]
M	Mass matrix
M_i	The molar change of component i due to reaction [kmol]
Mw	Molecular weight [kg/kmol]
P	Pressure [atm]
P_i^*	Vapour pressure of compound i [atm]
P_i	Price of corresponding flowrate i in dollars.
q	Feed liquid fraction
q_i	Relative surface area of pure compound i
R	Gas constant [J/K mol]
r_i	Rate of change due to reaction of component i [mol/l*h]
r_i	Relative Van der Waals volume of pure component i
Ref	The reflux [kmol/h]
T	Temperature [K]
Tb	Boiling temperature [K]
Δu_{ij}	Binary interaction energy coefficient of UNIQUAC
V	Vapour flowrate [kmol/h]
V_i	Volume fraction
Vb	Vapour boilup [kmol/h]
VR	Volume holdup [m^3]

14 Bibliography

1.   Simasatitkul, L., et al., *Reactive distillation for biodiesel production from soybean oil.* Korean Journal of Chemical Engineering, 2011. **28**(3): p. 649-655.

2.   *Annual Energy Outlook 2012 - with Projections to 2035*. June 2012; Available from: www.eia.gov/forecasts/aeo.

3.   *BP Statistical Review of World Energy June 2012*. Available from: http://goo.gl/5wTL9.

4.   Drapcho, C.M., N.P. Nhuan, and T.H. Walker, *Biofuels Engineering Process Technology*. 2008: McGraw-Hill.

5.   Kiss, A.A. and C.S. Bildea, *A review of biodiesel production by integrated reactive separation technologies.* Journal of Chemical Technology & Biotechnology, 2012. **87**(7): p. 861-879.

6.   Sundmacher, K. and A. Kienle, *Reactive Distillation - Status and Future Directions*. 2003: Wiley-VCH.

7.   Sharma, N. and K. Singh, *Control of Reactive Distillation Column: A Review.* International Journal of Chemical Reactor Engineering, 2010. **8**(1).

8.   Hammond, E.G., et al., *Soybean Oil*, in *Bailey's Industrial Oil and Fat Products*, F. Shahidi, Editor. 2005. p. 577-604.

9.   *DIPPR 801 Database*. Available from: http://dippr.byu.edu/students/.

10.   Knothe, G., J. Krahl, and J.V. Gerpen, *The Biodiesel Handbook*. 2 ed. 2010: AOCS Press.

11.   Malone, M.F. and M.F. Doherty, *Reactive Distillation.* Industrial & Engineering Chemistry Research, 2000. **39**(11): p. 3953-3957.

12.   Leet, W.A. and S. Kulprathipanja, *Reactive Separation Processes*, in *Reactive Separation Process*, S. Kulprathipanja, Editor. 2002, Taylor & Francis. p. 1-16.

13.   Towler, G.P. and S.J. Frey, *Reactive Distillation*, in *Reactive Separation Process*, S. Kulprathipanja, Editor. 2002, Taylor & Francis. p. 18-48.

14.   Issariyakul, T. and A.K. Dalai, *Comparative kinetics of transesterification for biodiesel production from palm oil and mustard oil.* The Canadian Journal of Chemical Engineering, 2012. **90**(2): p. 342-350.

15.   Skouras, S. *MulticomA - nonlinear model of a continuous distillation column with multicomponent mixture*. 2001; Available from: http://goo.gl/OF7qu.

16.   Geankoplis, C.J., *Vapor-Liquid Separation Processes*, in *Transport Processes and Separation Process Principles*. 2003, Prentice Hall. p. 696-759.

17. Sinnott, R. and G. Towler, *Chemical Engineering Design*. Coulson & Richardson's Chemical Engineering Series. Vol. 6. 2009: Elsevier.

18. *Vapor-Liquid Equilibria (VLE) - Background*. Available from: http://www.et.byu.edu/~rowley/VLEfinal/background.htm.

19. *1-Monolinoleoyl-rac-glycerol*. Available from: http://goo.gl/Ku39P.

20. *1,2-dilinoleoylglycerol*. Available from: http://goo.gl/45b0m.

21. Buckley, P.S., W.L. Luyben, and J.P. Shunta, *Design of Distillation Control Systems*. 1985: Edward Arnold.

22. Halvorsen, I.J. and S. Skogestad, *Distillation Theory*, in *Encyclopedia of Separation Science*, I.D. Wilson, Editor. 2000, Academic Press. p. 1117-11134.

23. Skogestad, S., *Chemical and Energy Process Engineering*. 2008: CRC Press Taylor & Francis Group.

24. Skouras, S. *colamod_nonideal.m - A model for 3 components and nonideal Wilson or Uniquac liquid activity*. 2005; Available from: http://www.nt.ntnu.no/users/skoge/distillation/nonideal_skouras/multicomponent/.

25. Abrams, D.S. and J.M. Prausnitz, *Statistical thermodynamics of liquid mixtures: A new expression for the excess Gibbs energy of partly or completely miscible systems.* AIChE Journal, 1975. **21**(1): p. 116-128.

26. Arauju, A. *multicomA.m - NEW VERSION, contains Simulink and steady-state*. 2008; Available from: http://www.nt.ntnu.no/users/skoge/distillation/multicomp_araujo/.

27. Larsson, T. and S. Skogestad, *Plantwide control - A review and a new design procedure.* Modeling, Identification and Control, 2000. **21**(4): p. 209-240.

28. Skogestad, S., *Economic Plantwide Control*, in *Plantwide Control recent Developments and Applications*, G.P. Rangaiah and V. Kariwala, Editors. 2012, Wiley: Chichester. p. 229-251.

29. Leung, D.Y.C., X. Wu, and M.K.H. Leung, *A review on biodiesel production using catalyzed transesterification.* Applied Energy, 2010. **87**(4): p. 1083-1095.

30. Felice, R.D., et al., *Component Distribution between Light and Heavy Phases in Biodiesel Processes.* Ind. Eng. Chem. Res., 2008. **47**(20): p. 7862-7867.

31. Chiu, C.-W., M.J. Goff, and G.J. Suppes, *Distribution of Methanol and Catalysts between Biodiesel and Glycerin Phases.* AIChE Journal, 2005. **51**(4): p. 1274-1278.

32. Cernoch, M., et al., *Distribution of Ethyl Esters and Glycerides between Ester and Glycerol Phase after Ethnolysis of Rapeseed Oil*, in *44th International Petroleum Conference*. 2009: Bratislava, Slovak Republic.

33. Gerpen, J.V., *Biodiesel processing and production.* Fuel Processing Technology, 2005. **86**(10): p. 1097-1107.

34. Zhang, Y., et al., *Biodiesel production from waste cooking oil: 2. Economic assessment and sensitivity analysis.* Bioresource Technology, 2003. **90**(3): p. 229-240.

35. Mongabay.com. *Soybean oil (Historical Annual Data).* 28/04/2013; Available from: http://www.mongabay.com/commodities/prices/soybean_oil.php.

36. *Soybean Oil Futures.* 23/05/2013; Available from: http://goo.gl/4SsRI.

37. Costenoble, O., et al. (2008) *Improvements needed for the biodiesel standard EN 14214*.

38. Jacobsen, M.G. and S. Skogestad, *Active constraint regions for a natural gas liquefaction process.* Journal of Natural Gas Science and Engineering, 2013. **10**: p. 8-13.

39. Downs, J.J. and S. Skogestad, *An industrial and academic perspective on plantwide control.* Advanced Control of Chemical Processes, 2011. **7**(1): p. 117-128.

40. Skogestad, S., *The do's and don'ts of distillation column control.* Chemical Engineering Research and Design, 2007. **85**(1): p. 13-23.

A Simulink Model

This Appendix includes the Matlab files and a screen print of the Simulink model used to simulate the production of biodiesel by reactive distillation system. The dynamic versions of the files were submitted electronically.

The files included in this appendix are listed and given a brief explanation in Table A-1 below.

Table A-1: The Matlab Simulink model files

File	About	Pages
main_script.m	The main file used to run the simulation from. Contains inputs to the Simulink file.	II
reactivedistillation_sfcn2.mdl	The Simulink model; contains the s-function Distillation_sfcn2 as the reactive distillation column.	III
Distillation_sfcn2.m	The s-function called upon in the Simulink model that describes the behavior of the reactive distillation column	IV-XV
steadystate.mat	The matrix containing the initial values used in the s-function.	-

Please also note that the s-function will also call upon UNIQUAC files, as this was the temperature estimation method of choice embedded in the Simulink files. Because the UNIQUAC files will have their own appendix, this section of the code has been made inactive here. But one can still see how it is implemented. The steadystate matrix is not included here either, as it only represents a long vector of numbers. However, this can be found electronically.

A.1 main_script.m

```matlab
%Basic model for reactive distillation column for biodiesel
production
%By Emilie Øritsland Houge
clc, clear all, close all
format long

%% Feed flows
%   MetOH/TG/DG/MG/GL/BD
F = [300 50 0 0 0 0];        % Feed flowrates [kmol/h]
F0 = sum(F);                 % Total feed flowrate [kmol/h]
x_in = F/F0;                 % Molar fractions
Mw = [32.042 878 616 354 92.095 294]; % Molecular weights
[kg/kmol]

%% Distillation data, set equal to results from HYSYS file
Ntrays = 20;        % Number of trays
NC = 6;             % Number of compounds
D = 56.9;           % Distillate [kmol/h]
Vb = 237.4;         % Vapour boil up [kmol/h]
B = 293.1;          % Bottoms, [kmol/h]
Ref = 3;            % Reflux ratio
Con_set = 5;        % Holdup condenser [kmol]
Reb_set = 5;        % Holdup reboiler [kmol]


%% Simulation
t0 = 0;      %[h]
tfin = 20;   %[h]

tic
sim('reactivedistillation_sfcn2')
toc
```

A.2 reactivedistillation_sfcn2
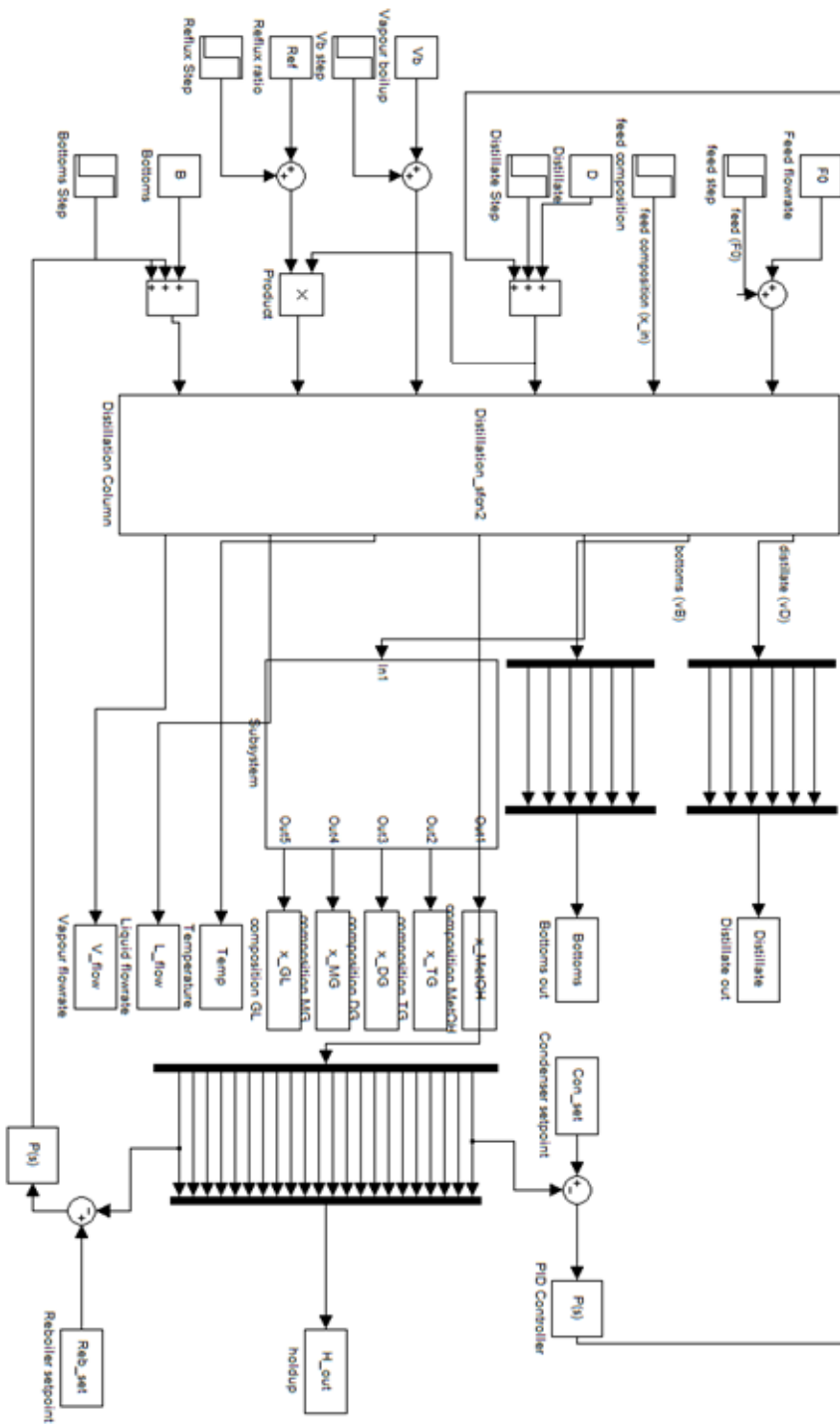
A.3 Distillation_sfcn2

```matlab
function Distillation_sfcn2(block)
% This is a level 2 sfcn in simulink describing the
production of biodiesel
% through reactive distillation
% The compounds are always listed in the following order:
metOH/TG/DG/MG/GL/BD

%% Establishing the number of inputs and outputs:
nu = 6;
ny = 7;
Ntrays = 20;
NC = 6;

% block definitions
block.NumInputPorts  = nu;
block.NumOutputPorts = ny;

% Setup port properties to be inherited or dynamic
block.SetPreCompInpPortInfoToDynamic;
block.SetPreCompOutPortInfoToDynamic;

for i = 1:nu
    block.InputPort(i).Dimensions = 1;
    block.InputPort(i).DirectFeedthrough = true;
end
block.InputPort(2).Dimensions = NC; %The second input port
is the feed molar fraction


for i = 1:ny
    block.OutputPort(i).Dimensions = 1;
end

block.OutputPort(1).Dimensions = NC;
block.OutputPort(2).Dimensions = NC;
block.OutputPort(3).Dimensions = (Ntrays+2)*(NC-1);
block.OutputPort(4).Dimensions = Ntrays+2;
block.OutputPort(5).Dimensions = Ntrays+2;
block.OutputPort(6).Dimensions = Ntrays+1;
block.OutputPort(7).Dimensions = Ntrays+1;

block.SampleTimes = [0 0];

% Setup Dwork
block.NumContStates = (Ntrays+2)*(NC-1)+Ntrays+2;
```

```matlab
block.RegBlockMethod('InitializeConditions',
@InitializeConditions);
block.RegBlockMethod('Outputs',                   @Outputs);
block.RegBlockMethod('Derivatives',
@Derivatives);
block.RegBlockMethod('SetInputPortSamplingMode',@SetInpPort
FrameData);


function SetInpPortFrameData(block, idx, fd)

block.InputPort(idx).SamplingMode = fd;
for i = 1:block.NumOutputPorts
    block.OutputPort(i).SamplingMode = fd;
end


function InitializeConditions(block)

load steadystate.mat
block.ContStates.Data = steadystate;

function Outputs(block)
global l XU
% Important Parameters
Ntrays = 20;                      %Number of trays
NC = 6;                           %Number of compounds
Mw = [32.042 878 616 354 92.095 294]; %Molecular weights
[kg/kmol]
alpha = [1.45*10^3 1 2.35*10 3.06*10 4.48*10 2.81*10];
Dc = 1.500;                       %Diameter of column [m]
lw = 0.77*Dc;                     %Length of weir [m]
hw = 0.20;                        %Heigth of weir [m]
q = 1.028;                        %Feed liquid fraction

%% Key Variables
% State variables
x = block.ContStates.Data(1:(Ntrays+2)*(NC-1)); %Molar
fractions
H = block.ContStates.Data(((Ntrays+2)*(NC-
1)+1):NC*(Ntrays+2)); %Molar holdup
xL = reshape(x,Ntrays+2,5);   %Reshaping into a matrix
HL = reshape(H,Ntrays+2,1);   %Reshaping into a vector
%We still use the sequence: MetOH/TG/DG/MG/GL/BD

% Disturbances
F0  = block.InputPort(1).Data;
```

V

```matlab
x_in  = block.InputPort(2).Data;

% Manipulation
D   = block.InputPort(3).Data;
Vb  = block.InputPort(4).Data;
Ref = block.InputPort(5).Data;
B   = block.InputPort(6).Data;

%% Calculation of compositions and temperature by method 1
%Setting the initial temperature
Te = 364.15*ones(Ntrays+2,1);
Te(1,1) = 337.63; Te(Ntrays+2,1) = 397.15;
XX = zeros(Ntrays+2,NC);   x_BD = zeros(Ntrays+2,1);
y = zeros(Ntrays+2,NC-1); yy = zeros(Ntrays+2,NC);

for i = 1:Ntrays+2
    %Molar fraction of BD:
    x_BD(i,1) = 1 - sum(xL(i,:));
    XX(i,:) = [xL(i,:) x_BD(i,1)];

    %Vapour composition by relative volatilities:
    yy(i,:) = (alpha.*XX(i,:))/sum(alpha.*XX(i,:));
    y(i,1) = yy(1,1); y(i,2) = yy(1,2); y(i,3) = yy(1,3);
y(i,4) = yy(1,4); y(i,5) = yy(1,5);
end

%% Calculations of densities in the liquid phase:
% The constants
%       Ar        Br      Cr      Dr
AR = [2.3267    0.27073 512.5 0.24713    % MetOH
      0.026085 0.14259 934.6 0.28571    % TG
      0.92382  0.24386 850   0.22114    % GL
      0.20469  0.23737 767.4 0.28571];  % BD

i = 1:Ntrays+2;
rho_MetOH = AR(1,1)./(AR(1,2).^(1+(1-
(Te(i,1)/AR(1,3))).^AR(1,4)));
rho_TG = AR(2,1)./(AR(2,2).^(1+(1-
(Te(i,1)/AR(2,3))).^AR(2,4)));
rho_GL = AR(3,1)./(AR(3,2).^(1+(1-
(Te(i,1)/AR(3,3))).^AR(3,4)));
rho_BD = AR(4,1)./(AR(4,2).^(1+(1-
(Te(i,1)/AR(4,3))).^AR(4,4)));
rho_DG = rho_TG;
rho_MG = rho_TG;
rho = [rho_MetOH rho_TG rho_DG rho_MG rho_GL rho_BD];

% kmol of components per tray [kmol]:
```

```
N(:,1) = HL.*XX(:,1);    % MetOH
N(:,2) = HL.*XX(:,2);    % TG
N(:,3) = HL.*XX(:,3);    % DG
N(:,4) = HL.*XX(:,4);    % MG
N(:,5) = HL.*XX(:,5);    % GL
N(:,6) = HL.*XX(:,6);    % BD


V = N./rho;              % volume of component per tray [m3]
Vtot = sum(V,2);         % total volume per tray [m3]

% Volume fraction [m3/m3]
ve(:,1) = V(:,1)./Vtot(:,1);     % MetOH
ve(:,2) = V(:,2)./Vtot(:,1);     % TG
ve(:,3) = V(:,3)./Vtot(:,1);     % DG
ve(:,4) = V(:,4)./Vtot(:,1);     % MG
ve(:,5) = V(:,5)./Vtot(:,1);     % GL
ve(:,6) = V(:,6)./Vtot(:,1);     % BD

% Common liquid density for the tray [kmol/m3]
rho_l = sum(rho.*ve,2);   % common liquid density for the
tray [kmol/m3]

% Weights per tray [kg]:
wt(:,1) = XX(:,1).*HL*Mw(1,1);   % MetOH
wt(:,2) = XX(:,2).*HL*Mw(1,2);   % TG
wt(:,3) = XX(:,3).*HL*Mw(1,3);   % DG
wt(:,4) = XX(:,4).*HL*Mw(1,4);   % MG
wt(:,5) = XX(:,5).*HL*Mw(1,5);   % GL
wt(:,6) = XX(:,6).*HL*Mw(1,6);   % BD
wt_tot = sum(wt,2);

% Weight Fractions
x_wt(:,1) = wt(:,1)./wt_tot;     % MetOH
x_wt(:,2) = wt(:,2)./wt_tot;     % TG
x_wt(:,3) = wt(:,3)./wt_tot;     % DG
x_wt(:,4) = wt(:,4)./wt_tot;     % MG
x_wt(:,5) = wt(:,5)./wt_tot;     % GL
x_wt(:,6) = wt(:,6)./wt_tot;     % BD

% A common liquid density for the tray [kg/m3]:
rho_Li(:,1) = rho(:,1).*x_wt(:,1)*Mw(1,1);   % MetOH
rho_Li(:,2) = rho(:,2).*x_wt(:,2)*Mw(1,2);   % TG
rho_Li(:,3) = rho(:,3).*x_wt(:,3)*Mw(1,3);   % DG
rho_Li(:,4) = rho(:,4).*x_wt(:,4)*Mw(1,4);   % MG
rho_Li(:,5) = rho(:,5).*x_wt(:,5)*Mw(1,5);   % GL
rho_Li(:,6) = rho(:,6).*x_wt(:,6)*Mw(1,6);   % BD
rho_L(i,1) = sum(rho_Li,2);
```

```matlab
%% Temperature Estimation

% % Initializing sizes
% T = Te;
%
% % Temperature Estimation by UNIQUAC:
% for l = 1:Ntrays+2
%     XU = [xL(:,2) xL(:,3) xL(:,4) x_BD(:,1) xL(:,5)
xL(:,1)];
%     T(l,1) = fzero('sumY_UNIQUAC2',Te(l,1));        %
Equilibrium temperature [K]
% end


%% Calculating the flowrates in the column
% Initializing sizes
V = ones(Ntrays+1,1);  %molar vapour flowrates.
L = ones(Ntrays+1,1);  %molar liquid flowrates.

% Molar balances over the top and the bottom:
L(1,1) = Ref;

% Applying Francis weir for the liquid flowrates
how = ones(Ntrays,1); Lw = ones(Ntrays,1);
for i = 1:Ntrays
    how(i,1) = (HL(i+1,1)-
(pi*hw*rho_l(i+1,1)*0.88*(Dc/2)^2))/(pi*0.88*rho_l(i+1,1)*(
Dc/2)^2);    %Calculation of how
    if how(i,1) <0
        Lw(i+1,1) = 0;
        L(i+1,1) = 0;
    else
        Lw(i+1,1) =
((how(i,1)*10^3)/750)^1.5*rho_L(i+1,1)*lw;    %Francis weir
[kg/s]
        L(i+1,1) = (Lw(i+1,1)*3600)/sum(XX(i+1,:).*Mw);
%Liquid flow [kmol/h]
    end
end

% Vapour flowrates
V(Ntrays+1,1) = Vb;

% Assuming constant molar overflow for vapour flowrates
for i = 2:Ntrays
    V(i,1) = V(Ntrays+1,1);
end
```

```matlab
% Correction for the feed:
V(1,1) = V(2,1) + ((1-q)*F0);



%% Determining distillate and bottoms flow
%Determination of the distillate flow
vD = D*XX(1,:);          % partial molar flows (kmol/h)


%Determination of the bottoms flow
vB = B*XX(Ntrays+2,:); % partial molar flows (kmol/h)



%% Output variables
block.OutputPort(1).Data = vD;
block.OutputPort(2).Data = vB;
block.OutputPort(3).Data = x;
block.OutputPort(4).Data = H;
block.OutputPort(5).Data = T;
block.OutputPort(6).Data = L;
block.OutputPort(7).Data = V;



function Derivatives(block)
%% Important parameters
global l XU

Ntrays = 20;                    %Number of trays
NC = 6;                         %Number of compounds
R = 8.314472*10^(-3);           %Gas constant [kJ/K*mol]
Mw = [32.042 878 616 354 92.095 294]; %Molecular weights
[kg/kmol]
q = 1.028;                      %Liquid feed fraction
Dc = 1.500;                     %Diameter of column [m]
lw = 0.77*Dc;                   %Length of weir [m]
hw = 0.20;                      %Heigth of weir [m]
alpha = [1.44*10^3 1 2.35*10 3.06*10 4.48*10 2.81*10];



%% Variables
% State variables
  x = block.ContStates.Data(1:(Ntrays+2)*(NC-1));
  H = block.ContStates.Data(((Ntrays+2)*(NC-
1)+1):NC*(Ntrays+2));
  xL = reshape(x,Ntrays+2,NC-1);
  HL = reshape(H,Ntrays+2,1);


% Disturbances
```

```matlab
  F0  = block.InputPort(1).Data;
  x_in = block.InputPort(2).Data;
%The input F0 is the flowrate [kmol/h] and x_in is the
composition.
%The composition is given in the following order:
MetOH/TG/DG/MG/GL/BD

% Manipulation
  D   = block.InputPort(3).Data;
  Vb = block.InputPort(4).Data;
  Ref = block.InputPort(5).Data;
  B = block.InputPort(6).Data;

%% Calculation of temperatures and compositions

% Setting the initial temperature
Te = 364.15*ones(Ntrays+2,1); Te(1,1) = 337.63;
Te(Ntrays+2,1) = 397.15;

XX = zeros(Ntrays+2,NC);  x_BD = zeros(Ntrays+2,1);
y = zeros(Ntrays+2,NC-1); yy = zeros(Ntrays+2,NC);

for i = 1:Ntrays+2
    x_BD(i,1) = 1 - sum(xL(i,:));
    XX(i,:) = [xL(i,:) x_BD(i,1)];

    %Vapour phase composition by relative volatilities:
    yy(i,:) = (alpha.*XX(i,:))/sum(alpha.*XX(i,:));
    y(i,1) = yy(1,1); y(i,2) = yy(1,2); y(i,3) = yy(1,3);
y(i,4) = yy(1,4); y(i,5) = yy(1,5);
end

%% Calculations of densities in the liquid phase:
% The constants
%        Ar        Br    Cr      Dr
AR = [2.3267    0.27073 512.5 0.24713    % MetOH
      0.026085 0.14259 934.6 0.28571    % TG
      0.92382  0.24386 850   0.22114    % GL
      0.20469  0.23737 767.4 0.28571];  % BD

i = 1:Ntrays+2;
rho_MetOH = AR(1,1)./(AR(1,2).^(1+(1-
(Te(i,1)/AR(1,3))).^AR(1,4)));
rho_TG = AR(2,1)./(AR(2,2).^(1+(1-
(Te(i,1)/AR(2,3))).^AR(2,4)));
rho_GL = AR(3,1)./(AR(3,2).^(1+(1-
(Te(i,1)/AR(3,3))).^AR(3,4)));
```

X

```matlab
rho_BD = AR(4,1)./(AR(4,2).^(1+(1-
(Te(i,1)/AR(4,3))).^AR(4,4)));
rho_DG = rho_TG;
rho_MG = rho_TG;
rho = [rho_MetOH rho_TG rho_DG rho_MG rho_GL rho_BD];

% kmol of components per tray [kmol]:
N(:,1) = HL.*XX(:,1);    % MetOH
N(:,2) = HL.*XX(:,2);    % TG
N(:,3) = HL.*XX(:,3);    % DG
N(:,4) = HL.*XX(:,4);    % MG
N(:,5) = HL.*XX(:,5);    % GL
N(:,6) = HL.*XX(:,6);    % BD


V = N./rho;              % volume of component per tray [m3]
Vtot = sum(V,2);         % total volume per tray [m3]

% Volume fraction [m3/m3]
ve(:,1) = V(:,1)./Vtot(:,1);    % MetOH
ve(:,2) = V(:,2)./Vtot(:,1);    % TG
ve(:,3) = V(:,3)./Vtot(:,1);    % DG
ve(:,4) = V(:,4)./Vtot(:,1);    % MG
ve(:,5) = V(:,5)./Vtot(:,1);    % GL
ve(:,6) = V(:,6)./Vtot(:,1);    % BD

% Common liquid density for the tray [kmol/m3]
rho_l = sum(rho.*ve,2);    % common liquid density for the
tray [kmol/m3]

% Weights per tray [kg]:
wt(:,1) = XX(:,1).*HL*Mw(1,1);   % MetOH
wt(:,2) = XX(:,2).*HL*Mw(1,2);   % TG
wt(:,3) = XX(:,3).*HL*Mw(1,3);   % DG
wt(:,4) = XX(:,4).*HL*Mw(1,4);   % MG
wt(:,5) = XX(:,5).*HL*Mw(1,5);   % GL
wt(:,6) = XX(:,6).*HL*Mw(1,6);   % BD
wt_tot = sum(wt,2);

% Weight Fractions
x_wt(:,1) = wt(:,1)./wt_tot;    % MetOH
x_wt(:,2) = wt(:,2)./wt_tot;    % TG
x_wt(:,3) = wt(:,3)./wt_tot;    % DG
x_wt(:,4) = wt(:,4)./wt_tot;    % MG
x_wt(:,5) = wt(:,5)./wt_tot;    % GL
x_wt(:,6) = wt(:,6)./wt_tot;    % BD

% A common liquid density for the tray [kg/m3]:
rho_Li(:,1) = rho(:,1).*x_wt(:,1)*Mw(1,1);   % MetOH
```

```matlab
rho_Li(:,2) = rho(:,2).*x_wt(:,2)*Mw(1,2);    % TG
rho_Li(:,3) = rho(:,3).*x_wt(:,3)*Mw(1,3);    % DG
rho_Li(:,4) = rho(:,4).*x_wt(:,4)*Mw(1,4);    % MG
rho_Li(:,5) = rho(:,5).*x_wt(:,5)*Mw(1,5);    % GL
rho_Li(:,6) = rho(:,6).*x_wt(:,6)*Mw(1,6);    % BD
rho_L(i,1) = sum(rho_Li,2);



%% Calculating the flowrates in the column
% Initializing sizes
V = ones(Ntrays+1,1);   % molar vapour flowrates
L = ones(Ntrays+1,1);   % molar liquid flowrates
% Molar balances over the top:
L(1,1) = Ref;    %The first liquid molar flow is equal to
the reflux.

% Applying Francis weir for the liquid flowrates
how = ones(Ntrays,1); Lw = ones(Ntrays,1);
for i = 2:Ntrays+1
    %Calculation of height over weir [m]:
        how(i,1) = (HL(i,1)-
(pi*hw*rho_l(i,1)*0.88*(Dc/2)^2))/(pi*0.88*rho_l(i,1)*(Dc/2
)^2);
        if how(i,1) <0
            %Ensures no negative flowrates within column
            Lw(i,1) = 0;
            L(i,1) = 0;
        else
            %Francis weir [kg/s]
            Lw(i,1) =
((how(i,1)*10^3)/750)^1.5*rho_L(i,1)*lw;
            %Liquid flowrate [kmol/h]
            L(i,1) = (Lw(i,1)*3600)/sum(XX(i,:).*Mw);
        end
end

% Vapour flowrates
V(Ntrays+1,1) = Vb;

%Assume constant molar flowrate:
for i = 2:Ntrays
    V(i,1) = V(Ntrays+1,1);
end

% Correction for the feed:
V(1,1) = V(2,1) + ((1-q)*F0);
```

```matlab
%% Calculating the vapour compositions by vapour pressures
+ temperature by method 2
T = Te;
%        TG       DG       MG       BD       GL     MetOH
XU = [xL(:,2) xL(:,3) xL(:,4) x_BD(:,1) xL(:,5) xL(:,1)];

% for l = 1:Ntrays+2
%    % Function ''fzero('sumY',value) returns the T value
(equilibrium temperature) that gives 1-sumY=0
%   T(l,1) = fzero('sumY_UNIQUAC2',Te(l,1));        %
Equilibrium temperature [K]
% end

%% Calculation of the kinetics in the liquid phase:

%The sizes of the kinetic constants:
k1 = ones(Ntrays+2,1); k2 = ones(Ntrays+2,1); k3 =
ones(Ntrays+2,1);
k4 = ones(Ntrays+2,1); k5 = ones(Ntrays+2,1); k6 =
ones(Ntrays+2,1);

for i = 1:Ntrays+2
%Kinetic data for the reaction, rate constants [m3/kmol*h]
k1(i) = 3.9*3600*(10^7)*exp(-54.9987/(R*T(i,1)));
k2(i) = 5.78*3600*(10^5)*exp(-41.5555/(R*T(i,1)));
k3(i) = 5.906*3600*(10^12)*exp(-83.0942/(R*T(i,1)));
k4(i) = 9.888*3600*(10^9)*exp(-61.2496/(R*T(i,1)));
k5(i) = 5.335*3600*(10^3)*exp(-26.8655/(R*T(i,1)));
k6(i) = 2.1*3600*(10^4)*exp(-40.1162/(R*T(i,1)));
end

%We need to convert the molar input to concentrations in
order to calculate
%the reaction outputs:
C = zeros(Ntrays+2,6);
R = zeros(Ntrays+2,6);
M = zeros(size(C));
Msum = zeros(Ntrays+2,1);
for i = 1:Ntrays+2
    vH = HL(i,1)*XX(i,:);      % Molar holdup per component
    vr = sum(vH./rho(i,:));    % volume holdup
    C(i,:) = vH/vr;            % concentration
    % The reactions. R is the gain/loss in [kmol/h]
    R_TG = -(k1(i)*C(i,2)*C(i,1)) + (k2(i)*C(i,3)*C(i,6));
    R_DG = (k1(i)*C(i,2)*C(i,1)) - (k2(i)*C(i,3)*C(i,6)) -
(k3(i)*C(i,3)*C(i,1)) + (k4(i)*C(i,4)*C(i,6));
    R_MG = (k3(i)*C(i,3)*C(i,1)) - (k4(i)*C(i,4)*C(i,6)) -
(k5(i)*C(i,4)*C(i,1)) + (k6(i)*C(i,5)*C(i,6));
```

XIII

```
    R_BD = (k1(i)*C(i,2)*C(i,1))-
(k2(i)*C(i,3)*C(i,6))+(k3(i)*C(i,3)*C(i,1))-
(k4(i)*C(i,4)*C(i,6))+(k5(i)*C(i,4)*C(i,1))-
(k6(i)*C(i,5)*C(i,6));
    R_GL = (k5(i)*C(i,4)*C(i,1)) - (k6(i)*C(i,5)*C(i,6));
    R_MetOH = -(k1(i)*C(i,2)*C(i,1))+(k2(i)*C(i,3)*C(i,6))-
(k3(i)*C(i,3)*C(i,1))+(k4(i)*C(i,4)*C(i,6))-
(k5(i)*C(i,4)*C(i,1))+(k6(i)*C(i,5)*C(i,6));
    % We store the molar gain of each component in an R
matrix:
    R(i,1) = R_MetOH; R(i,2) = R_TG; R(i,3) = R_DG; R(i,4)
= R_MG;
    R(i,5) = R_GL; R(i,6) = R_BD;
    M(i,:) = R(i,:)*vr; %The change in molar holdup of each
component saved in a vector
    Msum(i,1) = sum(M(i,:));
end

%% Molar balances in the column (the states)
dHxdt = zeros(Ntrays+2,5);
dHdt = zeros(Ntrays+2,1);

% Column
i = 2:Ntrays+1;
dHdt(i,1) = L(i-1,1) - L(i,1) + V(i,1) - V(i-1,1) +
Msum(i);
% Correction for the feed:
dHdt(2,1) = dHdt(2,1) + F0;

%The molar holdup composition balance for the feed tray:
for j = 1:5;
    for i = 2
        dHxdt(i,j) = ((F0*x_in(j,1)) + (V(i,1)*y(i,j)) +
L(i-1,1)*xL(i-1,j) - (V(i-1,1)*y(i-1,j)) - (L(i,1)*xL(i,j))
+ M(i,j));
    end
end

%The molar holdup composition balance for the other
reactive trays:
for j = 1:5;
    for i = 3:Ntrays+1
        dHxdt(i,j) = (V(i,1)*y(i,j)) + (L(i-1,1)*xL(i-1,j)
- (V(i-1,1)*y(i-1,j)) - (L(i,1)*xL(i,j)) + M(i,j));
    end
end

%Reboiler (assumed to be an equilibrium stage)
i = Ntrays+2;
```

```
dHdt(i,1) = L(Ntrays+1,1) - V(Ntrays+1,1) - B;


j=1:5;
dHxdt(i,j)= (L(Ntrays+1,1)*xL(Ntrays+1,j) -
V(Ntrays+1,1)*y(Ntrays+2,j) - B*xL(Ntrays+2,j));

%Total condenser (not an equilibrium stage)
i = 1;
dHdt(i,1) = V(1,1) - L(1,1) - D;

j = 1:5;
dHxdt(i,j)= V(1,1)*y(i,j) - L(1,1)*xL(i,j) - D*xL(i,j);

%Computing the derivative for the mole fractions from d(Mx)
= x dM + M dx
dxdt=(dHxdt - (xL.*(dHdt*ones(1,5))))./(HL*ones(1,5));

dxdtout = reshape(dxdt,(Ntrays+2)*5,1);
dxdtout = [dxdtout; dHdt];


block.Derivatives.Data = dxdtout;
```

B Hysys Model

The files in this appendix belong to the Hysys model as provided by Simasatitkul et al. during their research. The model is provided electronically, but some screen shots of the flow-diagram and corresponding flowrates, temperatures and compositions are provided below.

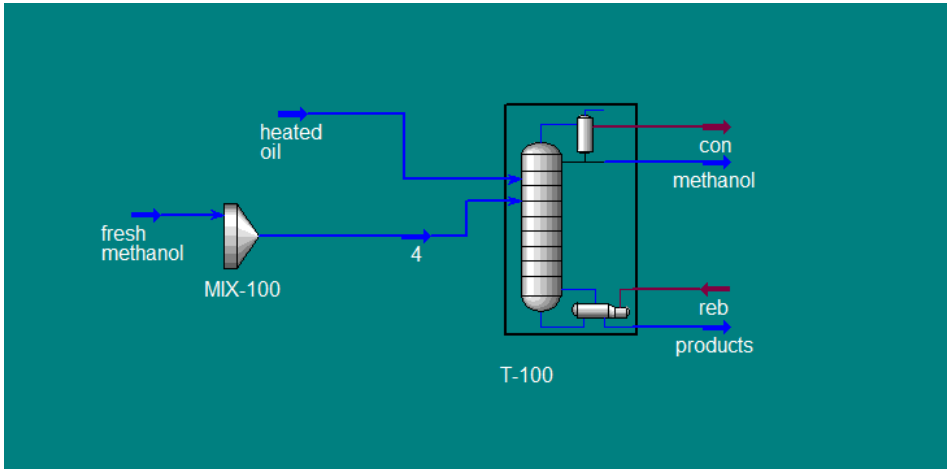*Figure B-1: Flow diagram of Hysys model.*

A print of the flowrates and compositions from the Hysys model are provided below in Figure B-2 and Figure B-3. The names of the flowrates should be self-explanatory, except for ov which corresponds to the overhead product (mainly methanol) and bot which is the bottom product (biodiesel). Streams 1 and 4 correspond to the soybean oil and methanol respectively.

Name	Reflux @COL1	To Condenser (	Boilup @COL1	To Reboiler @C	ov @COL1	
Vapour Fraction	0,0000	1,0000	1,0000	0,0000	0,0000	
Temperature [C]	64,48	88,75	124,0	91,83	64,48	
Pressure [kPa]	101,3	101,3	101,3	101,3	101,3	
Molar Flow [kgmole/h]	160,4	213,8	237,6	534,1	53,46	
Mass Flow [kg/h]	5139	6852	7634	5,943e+004	1713	
Liquid Volume Flow [m3/h]	6,458	8,611	9,583	66,83	2,153	
Heat Flow [kJ/h]	-3,766e+007	-4,241e+007	-4,679e+007	-1,942e+008	-1,255e+007	
Name	bot @COL1	1 @COL1	4 @COL1	** New **		
Vapour Fraction	0,0000	0,0000	0,0000			
Temperature [C]	124,0	50,00	50,00			
Pressure [kPa]	101,3	121,0	121,0			
Molar Flow [kgmole/h]	296,5	50,00	300,0			
Mass Flow [kg/h]	5,180e+004	4,390e+004	9613			
Liquid Volume Flow [m3/h]	57,25	47,98	12,08			
Heat Flow [kJ/h]	-1,354e+008	-8,122e+007	-7,095e+007			

Figure B-2: Overview of the flowrates.

Name	Reflux @COL1	To Condenser (	Boilup @COL1	To Reboiler @C	ov @COL1	
Comp Mole Frac (tri*)	0,0000	0,0000	0,0000	0,0010	0,0000	
Comp Mole Frac (di*)	0,0000	0,0000	0,0000	0,0018	0,0000	
Comp Mole Frac (mono*)	0,0000	0,0000	0,0000	0,0019	0,0000	
Comp Mole Frac (fame*)	0,0000	0,0000	0,0001	0,2724	0,0000	
Comp Mole Frac (NaOH)	0,0000	0,0000	0,0000	0,0000	0,0000	
Comp Mole Frac (H2O)	0,0000	0,0000	0,0000	0,0000	0,0000	
Comp Mole Frac (Glycerol)	0,0000	0,0000	0,0009	0,0893	0,0000	
Comp Mole Frac (Methanol)	1,0000	1,0000	0,9990	0,6336	1,0000	
Name	bot @COL1	1 @COL1	4 @COL1	** New **		
Comp Mole Frac (tri*)	0,0018	1,0000	0,0000			
Comp Mole Frac (di*)	0,0033	0,0000	0,0000			
Comp Mole Frac (mono*)	0,0034	0,0000	0,0000			
Comp Mole Frac (fame*)	0,4905	0,0000	0,0000			
Comp Mole Frac (NaOH)	0,0000	0,0000	0,0000			
Comp Mole Frac (H2O)	0,0000	0,0000	0,0000			
Comp Mole Frac (Glycerol)	0,1601	0,0000	0,0000			
Comp Mole Frac (Methanol)	0,3409	0,0000	1,0000			

Figure B-3: Overview of the compositions.

C   UNIQUAC

This Appendix includes the Matlab files used to estimate the temperature profile by applying the UNIQUAC method. The dynamic versions of the files were submitted electronically. The files used here are as implemented in the Simulink model, and are listed in Table C-1 below.

Table C-1: The UNIQUAC Matlab files

File	About	Pages
sumY_UNIQUAC2.m	The main function used, sends the temperature input to the UNIQUAC2 function and iterates until the total vapour composition fractions are equal to 1.	XIX
UNIQUAC2.m	The function containing the UNIQUAC model, calculates the activity coefficient and the corresponding vapour molar composition fractions.	XX - XXII

Please note that the UNIQUAC files were used in combination with the regular models. The UNIQUAC method is implemented differently in the DAE system and the Simulink model. This version only shows the files used in the Simulink implementation, so please also take a look at the implementation used for the DAE-system.

C.1 sumY_UNIQUAC.m

```
function Y = sumY_UNIQUAC2(T);
global l XU
% This function returns the vector of differences (1-sumY)
at given values of
% temperature (T), pressure (P) and liquid composition (X)

Y = UNIQUAC2(T);
Y = 1-sum(Y);
```

C.2 UNIQUAC2.m

```matlab
function Y = UNIQUAC2(T)
% This function uses the UNIQUAC model at given values of
% temperature (T), pressure (P) and liquid composition (X)
% Returns the vector of vapour compositions (Y)
global l XU

% Necessary data:
R = 1.98721;      % universal gas constant [cal/molK]
NC = 6;           % number of components
P = 1;            % pressure [atm]

% UNIQUAC Parameters:
%       TG        DG        MG        BD       GL        MetOH
W = [    0     -48.519   213.876   47.076   836.044   3030.060
      59.273      0      -79.159   26.437   757.300    -89.061
    -139.052  114.633      0      219.802   317.795   -113.482
     -45.259  -16.759  -125.764     0       823.302      0
    -266.224 -274.223  -60.390  -262.612     0        411.932
    -704.965  870.689   534.174     0      -267.337      0  ];

% Coefficients for the vapour pressure equations:
%       A           B            C            D           E
A = [505.205  -9.41262*10^4  -59.7879   5.8959*10^-19   6
%TG
     18.1551  -7.16544*10^3  -0.847809  3.18337*10^-20  6
%DG
     145.254  -2.11542*10^4  -16.8740   1.74025*10^-18  6
%MG
     107.534  -1.56037*10^4  -12.2136   1.79996*10^-18  6
%BD
     169.211  -1.68937*10^4  -21.8125   1.14067*10^-5   2
%GL
     59.8373  -6.28289*10^3  -6.37873   4.61746*10^-6   2];
%MetOH

% Volume parameters of pure components:
%       TG        DG       MG       BD     GL     MetOH
RP = [39.6176 28.0103 16.403 13.5084 4.7957 1.4311];

% Area parameters of pure components:
%      TG     DG     MG     BD     GL    MetOH
Q = [32.094 23.032 13.97 11.11 4.908 1.432];
```

```matlab
% Vapour pressure equation where Psat [mmHg] and
temperature T [K]
Ps(1,1) = 760*9.869*(10^-3)*exp(A(1,1) + (A(1,2)/T) +
(A(1,3)*log(T)) + (A(1,4)*T^A(1,5)));    %TG
Ps(1,2) = 760*9.869*(10^-3)*exp(A(2,1) + (A(2,2)/T) +
(A(2,3)*log(T)) + (A(2,4)*T^A(2,5)));    %DG
Ps(1,3) = 760*9.869*(10^-3)*exp(A(3,1) + (A(3,2)/T) +
(A(3,3)*log(T)) + (A(3,4)*T^A(3,5)));    %MG
Ps(1,4) = 760*9.869*(10^-3)*exp(A(4,1) + (A(4,2)/T) +
(A(4,3)*log(T)) + (A(4,4)*T^A(4,5)));    %BD
Ps(1,5) = 760*9.869*(10^-3)*exp(A(5,1) + (A(5,2)/T) +
(A(5,3)*log(T)) + (A(5,4)*T^A(5,5)));    %GL
Ps(1,6) = 760*9.869*(10^-3)*exp(A(6,1) + (A(6,2)/T) +
(A(6,3)*log(T)) + (A(6,4)*T^A(6,5)));    %MetOH

% Setting the coordination number
z = 10;

% Calculating the surface area fractions:
F = Q./sum(Q.*XU(l,:));

% Calculating the volume fraction per mixture mole
fraction:
PHI = RP./sum(RP.*XU(l,:));

% Calculate the matrix of UNIQUAC coefficients TAF_ij
TAF = ones(NC,NC);
for i=1:NC
  for j=1:NC
    if TAF(j,i) == 0;
     TAF(j,i) = 1;
    else
     TAF(j,i) = exp(-W(j,i)/(R*T));
    end
  end
end

% Calculating the combinatorial contribution:
lnGammaC = ones(1,NC);
for i = 1:NC
    lnGammaC(1,i) = (1-PHI(1,i)+log(PHI(1,i)))-
((z/2)*Q(1,i)*(1-(PHI(1,i)/F(1,i))+log(PHI(1,i)/F(1,i))));
end


% Calculating the residual part
NOM = ones(1,NC); RATIO = ones(1,NC); lnGammaR =
ones(1,NC);
```

```
for i = 1:NC
        NOM(1,i) = sum(Q.*XU(l,:).*TAF(i,:));
        DENOM = sum(Q.*XU(l,:));
        RATIO(1,i) = sum((Q.*XU(l,:).*TAF(i,:))./NOM(1,i));
        lnGammaR(1,i) = Q(1,i)*(1-log(NOM(1,i)/DENOM)-
RATIO(1,i));
end


% Calculating the activity coefficient:
lnGamma = lnGammaC + lnGammaR;
Gamma = exp(lnGamma);

Y = (Ps.*XU(l,:).*Gamma)./(P*760);
```

D   Empirical Temperature Estimation

The purpose of this appendix is to introduce the Matlab files used to find the parameters of the empirical temperature estimation method. The main file introduces the steady-state molar compositions in the vapour and liquid phase and tray temperatures found from Hysys simulation. The file also gives initial values for the empirical estimation method and tries to solve the system by applying fsolve and the function Temp_est.

The main file is given below:

```
% Matlab code to find an empirical temperature estimation
method
clc, clear all, close all
global X Y Tb T

% The molar compositions in the liquid phase:
%      MetOH    TG         DG        MG         GL         BD
X = [1.0000 1.656e-24 7.9e-8    1.294e-8 2.784e-5 4.972e-6
%1
     0.6909 1.133e-2  5.203e-3 5.958e-3 6.738e-2 0.2192
%2
     0.6486 2.367e-3  2.650e-3 2.785e-3 8.386e-2 0.2598
%3
     0.6412 1.131e-3  1.905e-3 2.002e-3 8.696e-2 0.2668
%4
     0.6398 9.082e-4  1.730e-3 1.823e-3 8.761e-2 0.2682
%11
     0.3408 1.760e-3  3.306e-3 3.434e-3 0.1601    0.4906];
%22

% The molar compositions in the vapour phase:
%      MetOH    TG         DG        MG         GL         BD
Y = [1.0000 5.367e-24 6.326e-13 3.693e-16 2.405e-10 2.649e-
13
     1.0000 1.440e-28 7.900e-8  1.294e-8  2.784e-5  4.972e-
6
     1.0000 0.0000    6.266e-8  8.560e-9  3.900e-5  7.461e-
6
     1.0000 6.883e-28 4.774e-8  6.537e-9  4.175e-5  8.006e-
6
     0.9999 9.794e-27 4.382e-8  6.026e-9  4.237e-5  8.123e-
6
     0.9990 8.759e-25 2.724e-7  1.365e-7  8.913e-4  1.455e-
4];

% The boiling points for the respective components [K]:
Tb = [337.63 1199.65 922.35 696.35 561.0 653.15];
```

```
% The temperatures for these compositions [K]:
% Tray  1       2        3        4        11       22
T = [ 337.63; 361.33; 363.82; 364.31; 364.42;  397.15];

% Initial conditions:
u0 = [0.25 0.25 0.25 0.25 0.25 0.25];

% Calculating the weighting factors:
u = fsolve('Temp_est',u0)

% Checking that the temperatures are ok:
uy = 1-u;
Te = ones(6,1);
for i = 1:6
      Te(i,1) = (sum(Tb.*X(i,:).*u) + sum(Tb.*Y(i,:).*uy));
end
```

The main file also calculates the tray temperatures by applying the empirical method, for a quality check and to compare with the temperatures from Hysys.

The function Temp_est varies the weighting factors by minimising the difference between the temperature of the Hysys model and temperatures as calculated using equation 4.5.

```
function f = Temp_est(u)
% This function calculates the weighting factors of the
% empirical temperature estimation method
global X Y Tb T

uy = 1 - u;
for i = 1:6
      f = T(i,1) - (sum(Tb.*X(i,:).*u) +
sum(Tb.*Y(i,:).*uy));
end
```

E DAE-model

This Appendix includes the Matlab files used to model the production of biodiesel by reactive distillation. The dynamic versions of the files were submitted electronically. The matrix described in the table is only provided electronically.

Table E-1: The DAE Matlab files

File	About	Pages
main.m	The main script used for running the steady-state, dynamic and optimisation files for the model. Inputs and disturbances to the model are also defined here	XXVI - XXVII
parameters_c.m	A file containing the necessary parameters used in the process.	XXVIII - XXX
steadystate_5.mat	Matrix used as initial values in the main script, as input to the steady-state simulation.	-
cost_c.m	The cost function containing the optimisation criterias. Used for the steady-state and optimization phases.	XXXI - XXXII
post_processing.m	An m-code for the settling tank and the methanol/glycerol flash.	XXXIII
reactivedist_c.m	The non-linear constraints on the process. Calls on the reactive distillation model model_c for the steady-state and optimisation simulations.	XXXIV - XXXV
model_c.m	The reactive distillation model.	XXVI - XLI
output_c.m	A file for providing the necessary outputs of the model.	XLII
print_results_c.m	A file for printing outputs after ended simulation.	XLIII
reactivedist_c_DAE.m	The reactive distillation model provided for the DAE system.	XLIV - XLIX
output_dynamic.m	A file for providing the outputs of the dynamic simulation.	L
model_c_simple.m	A simplified version of model_c.	LI - LIII

E.1 main.m

```matlab
%% Main program
clear all; clc; close all

parameters_c

% Inputs and disturbances:
LT = 226.8/60;          % Reflux [kmol/min]
VB = 309.4/60;          % Vapour boilup [kmol/min]
F  = 350/60;            % Feedrate [kmol/min]
qF = 1.064;             % Feed liquid fraction
%      MetOH    TG   DG MG GL BD
zF = [332/350 18/350 0 0 0 0]; % Feed molar compositions
u1 = [LT VB];
d  = [F qF zF];

% Initial values:
load steadystate_5 % y0
% load steadystate_high_350
T0 = (91.27+273.15)*ones(NT,1);
x0 = [y0; T0];

%% Steady-state calculations
options = optimset('algorithm','interior-
point','Display','iter','Diagnostics','on');
tic
x  =
fmincon(@cost_c,x0,[],[],[],[],[],[],@reactivedist_c,option
s,u1,d);
toc
f1 = output_c(x,u1,d);
print_results_c(f1)

x0 = x;

%% Optimization calculations
options = optimset('algorithm','interior-
point','Display','iter','Diagnostics','on','maxfunevals',15
0000, 'maxiter',5000);
lb = zeros(length([x0;u1']),1);
tic
y  =
fmincon(@cost_c,[x0;u1'],[],[],[],[],lb,[],@reactivedist_c,
options,d);
toc
z0 = y(1:length(x0)); u2 = y(length(x0)+1:end);
f2 = output_c(z0,u2,d);
```

```
print_results_c(f2);

LT = f2{1};          % Reflux
VB = f2{2};          % Boilup
u1 = [LT VB];

%% Dynamic calculations
% Using DAE system and mass matrix
% F  = 350;          % Introducing disturbance in feed
% d  = [F qF zF];      % Disturbances
%
% % The mass matrix:
% temporary = ones(132,1);
% temporary1 = zeros(22,1);
% m_matrix = diag([temporary;temporary1]);
% options = odeset('mass', m_matrix);
%
% % Time:
% t0 = 0;
% tfin = 10;
%
% % Solving the system:
% tic
% [t,zf] = ode15s(@reactivedist_c_DAE,[t0
tfin],z0,options);
% toc
% f3 = output_dynamic(zf(end,:),u1,d);
```

E.2 parameters_c.m

```
%% Essential parameters
% Number of stages (including reboiler and total condenser:
NT = 5;

% Location of feed stage (stages are counted from the
bottom):
NF = NT-1;

% Relative volatilities:
alpha1 = [1.45*10^3 1 2.35*10 3.06*10 4.48*10 2.81*10];
R = 8.314; %i.g. constant
% Number of components
NC = length(alpha1);
Mw = [32.042 878 616 354 92.095 294];



%% Dynamics
Dc = 1.5;                              % Diameter of
column [m]
hw = 0.05;                             % Heigth of weir
[m]
lw = 0.77*Dc;                          % Length of weir
[m]
area_plate = pi*0.88*(Dc/2)^2;         % Area of plate
[m^2]
volume_plate = area_plate * hw;        % Volume of plate
[m^3]

%tau_1 = 0.063; %min

%% Density

% Density data:
%         Ar          Br    Cr      Dr
AR = [2.3267    0.27073 512.5 0.24713    % MetOH
      0.026085 0.14259 934.6 0.28571     % TG
      0.026085 0.14259 934.6 0.28571     % DG
      0.026085 0.14259 934.6 0.28571     % MG
      0.92382  0.24386 850   0.22114     % GL
      0.20469  0.23737 767.4 0.28571];   % BD

%% UNIQUAC parameters:
Re = 1.98721;    % universal gas constant [cal/molK]
z = 10;          % the coordination number
```

```matlab
% UNIQUAC Parameters:
%        TG        DG        MG        BD        GL        MetOH
W = [     0      -48.519   213.876   47.076    836.044   3030.060
         59.273     0      -79.159   26.437    757.300   -89.061
       -139.052  114.633     0       219.802   317.795   -113.482
        -45.259  -16.759  -125.764     0        823.302     0
       -266.224 -274.223  -60.390   -262.612     0        411.932
       -704.965  870.689   534.174     0       -267.337     0
];

% Coefficients for the vapour pressure equations [kPa]:
%        A             B            C              D             E
A = [505.205  -9.41262*10^4  -59.7879   5.8959*10^-19   6
%TG
     18.1551  -7.16544*10^3  -0.847809 3.18337*10^-20  6
%DG
     145.254  -2.11542*10^4  -16.8740   1.74025*10^-18  6
%MG
     107.534  -1.56037*10^4  -12.2136   1.79996*10^-18  6
%BD
     169.211  -1.68937*10^4  -21.8125   1.14067*10^-5   2
%GL
     59.8373  -6.28289*10^3  -6.37873   4.61746*10^-6   2];
%MetOH

% Volume parameters of pure components:
%        TG        DG       MG        BD       GL      MetOH
RP = [39.6176 28.0103 16.403 13.5084 4.7957 1.4311];

% Area parameters of pure components:
%        TG      DG      MG      BD      GL    MetOH
Q = [32.094 23.032 13.97 11.11 4.908 1.432];

%% Optimisation and control
% Prices
pMetOH = 180*10^-3;      % [$/kg]
pTG    = 338*10^-3;      % [$/kg]
pBD    = 600*10^-3;      % [$/kg]
pGL    = 1200*10^-3;     % [$/kg]
% pGL   = 750*10^-3;      % [$/kg]
pLPS   = (6.8*10^-3);    % [$/ton]
pCW    = (0.007*10^-3);  % [$/kg]

% Soft constraint for the GL_tot:
% alpha_e = 30000000;
alpha_e = 100;
```

```matlab
% Maximum and minimum boilup:
VBmax = 8.5;
VBmin = 0.01;

% P-controllers for control of reboiler and condenser hold
up.
KcB = -10;  KcD = -10;                    % controller gains
MDs = 5; MBs = 5;                         % Nominal holdups -
these are rather small
Ds  = 57/60; Bs  = 293/60;                % Nominal flows
```

E.3 cost_c.m

```matlab
function f = cost_c(t,varargin)

parameters_c
switch nargin
    case 2 % Optimization
        X  = t;
        U  = X((NC*NT)+NT+1:end);
        Dd = varargin{1};
        % Splitting the states
        x  = X(1:(NC-1)*NT);
        M  = X((NC-1)*NT+1:NC*NT);
        T  = X((NC*NT)+1:(NC*NT)+NT);
        xB = x(NT+1);
        xx = reshape(x,NT,NC-1);
        xx = [xx,1-sum(xx,2)];

        % inputs and disturbances:
        D  = Ds + (M(NT) - MDs)*KcD;
        B  = Bs + (M(1) - MBs)*KcB;
        F  = Dd(1);
        qF = Dd(2);
        zF = Dd(3:end);
        LT = U(1);
        VB = U(2);
        V1 = LT + D;
        % utilities:
        CW = ((1714.3*V1) + 6476.8)/(60*1000);  % cooling
water [m^3/min]
        LPS = (2316.9*exp(0.0037*VB))/60;       % low
pressure steam [kg/min]
        % post-processing:
        post_processing
        % Cost function
        %epsilon = GL_tot - 0.0025;
%       epsilon = 0 - LT;
        cost_feed = (F*zF(1,1)*Mw(1,1)*pMetOH) +
(F*zF(1,2)*Mw(1,2)*pTG);
        val_prod  = (pBD*EP_tot) + (pGL*CGL_tot) +
(D*Mw(1,1)*pMetOH);
        cost_util = (pLPS*LPS) + (pCW*CW);
        f = cost_feed + cost_util - val_prod;
    case 3 % Steady state
        X  = t;
        U  = varargin{1};
        Dd = varargin{2};
        f  = 1;
    case 4 % Dynamic
```

```
        X  = varargin{1};
        U  = varargin{2};
        Dd = varargin{3};
end
```

E.4 post_processing.m

```matlab
%% Settling tank
% Glycerol-rich phase:
GP_MetOH = 0.98*B*xx(1,1);
GP_TG = 0*xx(1,2);
GP_DG = 0.4*B*xx(1,3);
GP_MG = 0.8*B*xx(1,4);
GP_GL = 0.999*B*xx(1,5);
GP_BD = 0.001*B*xx(1,6);


GP = [GP_MetOH GP_TG GP_DG GP_MG GP_GL GP_BD];
GP_wt = GP.*Mw;
GP_tot = sum(GP_wt);
GP_per = GP_wt/GP_tot;



% Ester-rich phase:
EP = ((B*xx(1,:))-GP);
EP_wt = EP.*Mw;
EP_tot = sum(EP_wt);
EP_per = EP_wt/EP_tot;


%            TG              DG              MG              GL
GL_tot = EP_per(1,2) + EP_per(1,3) + EP_per(1,4) +
EP_per(1,5);

%% Methanol Flash
CGL_MetOH = 0.01*GP_MetOH;
CGL_TG = GP_TG;
CGL_DG = 0.98*GP_DG;
CGL_MG = 0.99*GP_MG;
CGL_GL = 0.99*GP_GL;
CGL_BD = 0.95*GP_BD;
CGL = [CGL_MetOH CGL_TG CGL_DG CGL_MG CGL_GL CGL_BD];
CGL_wt = CGL.*Mw;
CGL_tot = sum(CGL_wt);
CGL_per = CGL_wt/CGL_tot;

CMet = GP - CGL;
CMet_wt = CMet.*Mw;
CMet_tot = sum(CMet_wt);
CMet_per = CMet_wt/CMet_tot;
```

E.5 reactivedist_c.m

```matlab
function varargout = reactivedist_c(t,varargin)

parameters_c
switch nargin
    case 2 % Optimization
        X  = t;
        U  = X((NC*NT)+NT+1:end);
        Dd = varargin{1};
    case 3 % Steady state
        X  = t;
        U  = varargin{1};
        Dd = varargin{2};

end

% Splitting the states
x = X(1:(NC-1)*NT);                  % Liquid
compositions from btm to top
M = X((NC-1)*NT+1:NC*NT);            % Liquid hold up
from btm to top
TL = X((NC*NT)+1:(NC*NT)+NT);        % Temperatures from
btm to top

% Rearrange elements of composition vector (x) for later
use
Iu = reshape(1:(NC-1)*NT,NT,NC-1);
x  = x(Iu);

% Inputs
LT = U(1);         % Reflux
VB = U(2);         % Boilup

% Disturbances
F  = Dd(1);        % Feedrate
qF = Dd(2);        % Feed liquid fraction
zF = Dd(3:end);    % Feed compositions

% The distillation model
flag = 0; % This flag signals the inclusion of equations
for liquid levels. If flag~=0, Simulink is being used and
no liquid level is calculated in the code.
model_c

% Output
xprime = [dxdt; dMdt; f3];
x = x(:);
```

XXXIV

```matlab
% Extra constraint which was transferred into a soft
constraint:
% GL_tot - 0.0025;


c = [-D + 0; -B + 0; -LT + 0; -VB + VBmin; VB - VBmax;
EP_per(1,1) - 0.002; EP_per(1,2) - 0.002; EP_per(1,3) -
0.002; EP_per(1,4) - 0.008; EP_per(1,5) - 0.0002; GL_tot -
0.0025; -EP_per(1,6) + 0.965; -CGL_per(1,5) + 0.92; TL(1) -
423.15];
ceq = xprime;

switch nargin
    case 2 % Optimization
        varargout = {c,ceq};
    case 3 % Steady state
        varargout = {[],xprime};
end
```

E.6 model_c.m

```
%% The reactive distillation model

% All mole fractions in the sequence: MetOH/TG/DG/MG/GL/BD:
xx = [x,1-sum(x,2)];

% Vapour-liquid equilibria (multicomponent ideal VLE,
Stichlmair-Fair, 'Distillation', p. 36, 1998)
y = (x*diag(alpha1(1:NC-1)))./((x*(alpha1(1:NC-1) - 1)' +
1)*ones(1,NC-1));

%% Density calculations
% Assuming constant temperature:
T = 91.27+273.15;              % [K]

% Molar density
rho = AR(:,1)./AR(:,2).^(1+(1-(T./AR(:,3))).^AR(:,4)); %
[kmol/m^3]
N = (M*ones(1,NC)).*xx;        % molar holdup of components
per tray [kmol]

% Volume fractions:
V = N./(ones(NT,1)*rho');      % volume of component per tray
[m3]
Vtot = sum(V,2);               % total volume per tray [m3]
ve = V./(Vtot*ones(1,NC));     % volume fraction [m3/m3]
rho_l = sum((ones(NT,1)*rho').*ve,2);   % Common liquid
density for the tray [kmol/m3]

% Weight fractions:
wt = xx.*(M*ones(1,NC)).*(ones(NT,1)*Mw);
wt_sum = sum(wt,2);
x_wt = wt./(wt_sum*ones(1,NC));

% Common liquid density for each tray [kg/m3]:
r_wt = (ones(NT,1)*rho').*x_wt.*(ones(NT,1)*Mw);
rho_L = sum(r_wt,2);

% v = diag(xx/diag(rho)*ones(6,1))\(xx/diag(rho)); %molar
frac
% avg_rho = v*rho; %average molar denity on each tray
% vol = diag(avg_rho)\M; %volume on each tray

vol= volume_plate*ones(NT,1);
```

```matlab
%% Dynamics
% Vapor flows assuming constant molar flows: [kmol/h]
V = VB*ones(NT-1,1); V(NF:NT-1) = V(NF:NT-1) + (1-qF)*F;


% Liquid flows are given by Franci's Weir Formula
L(i)=K*Mow(i)^1.5
% Liquid flow L(i) dependent only on the holdup over the
weir Mow(i)
% M(i)= Mow(i) + Muw(i) (Total holdup = holdup over weir +
holdup below weir)
Muw = volume_plate*rho_l;              % Molar holdup
under weir [kmol]
Mw_avg = sum(xx.*(ones(NT,1)*Mw),2);    % Average molecular
weight per tray [kg/kmol]


% Constants for the Francis weir equation:
Kbf = 60*rho_l*lw.*(10^3./(750*area_plate*rho_l)).^1.5;


L(2:NF,1)        = (Kbf(2:NF,1).*(max(M(2:NF) -
Muw(2:NF,1),0))).^1.5;                 % Liquid flows below
feed (Kmol/min)
%L(NF+1:NT-1,1) = (Kof*(max(M(NF+1:NT-1) - Muw(NF+1:NT-
1,1),0))).^1.5;   % Liquid flows above feed (Kmol/min)
L(NT,1)          = LT;                           %
Condenser's liquid flow (Kmol/min)


%Liquid control in the refulx drum and reboiler: flow
equations for D and B
if flag == 0 % This flag signals the inclusion of equations
for liquid levels. If flag=1, Simulink is being used and no
liquid level is calculated in the code.
    D = max(Ds + (-M(NT) + MDs)*KcD,0);       % Distillate
flow
    B = max(Bs + (-M(1) + MBs)*KcB,0);        % Bottoms flow
end


%% UNIQUAC model

% Vapour pressure equation where Psat [mmHg] and
temperature TL [K]
j = 1:NT;
Ps1(:,1) = (A(1,2)./TL) + (A(1,3)*log(TL)) +
A(1,4)*TL.^A(1,5); % TG
Ps1(:,2) = (A(2,2)./TL) + (A(2,3)*log(TL)) +
A(2,4)*TL.^A(2,5); % DG
Ps1(:,3) = (A(3,2)./TL) + (A(3,3)*log(TL)) +
A(3,4)*TL.^A(3,5); % MG
```

XXXVII

```matlab
Ps1(:,4) = (A(4,2)./TL) + (A(4,3)*log(TL)) +
A(4,4)*TL.^A(4,5); % BD
Ps1(:,5) = (A(5,2)./TL) + (A(5,3)*log(TL)) +
A(5,4)*TL.^A(5,5); % GL
Ps1(:,6) = (A(6,2)./TL) + (A(6,3)*log(TL)) +
A(6,4)*TL.^A(6,5); % MetOH
Ps = 760*9.869*(10^-3)*exp((ones(NT,1)*A(:,1)') + Ps1);

%        TG      DG       MG       BD       GL       MetOH
XU = [xx(j,2) xx(j,3) xx(j,4) xx(j,6) xx(j,5) xx(j,1)];

% Calculating the surface area fractions:
Q_tot = (ones(NT,1)*Q).*XU;
Q_sum = sum(Q_tot,2);
F1 = (ones(NT,1)*Q)./(Q_sum*ones(1,NC));

% Calculating the volume fraction per mixture mole
fraction:
RPt = (ones(NT,1)*RP).*XU;
RP_sum = sum(RPt,2);
PHI = (ones(NT,1)*RP)./(RP_sum*ones(1,NC));

% Calculating matrix of UNIQUAC coefficients TAF_ij:
TAF1 = exp(-W/(Re*TL(1,1)));
TAF2 = exp(-W/(Re*TL(2,1)));
TAF3 = exp(-W/(Re*TL(3,1)));
TAF4 = exp(-W/(Re*TL(4,1)));
TAF5 = exp(-W/(Re*TL(5,1)));

% Calculating the combinatorial contribution:
lnGammaC = (1-PHI+log(PHI))-((z/2)*(ones(NT,1)*Q).*(1-
(PHI./F1)+log(PHI./F1)));

% Calculating the residual contribution:
Q1 = ones(NC,1)*Q;            % Q as a (6,6) matrix
NOM(1,:) = sum(Q1.*(ones(NC,1)*XU(1,:)).*TAF1,2)';
DENOM(1,1) = sum(Q.*XU(1,:));
RATIO(1,1) = sum((Q.*XU(1,:).*TAF1(1,:))./NOM(1,1));
RATIO(1,2) = sum((Q.*XU(1,:).*TAF1(2,:))./NOM(1,2));
RATIO(1,3) = sum((Q.*XU(1,:).*TAF1(3,:))./NOM(1,3));
RATIO(1,4) = sum((Q.*XU(1,:).*TAF1(4,:))./NOM(1,4));
RATIO(1,5) = sum((Q.*XU(1,:).*TAF1(5,:))./NOM(1,5));
RATIO(1,6) = sum((Q.*XU(1,:).*TAF1(6,:))./NOM(1,6));
lnGammaR(1,:) = Q.*(1-log(NOM(1,:)./DENOM(1,1))-
RATIO(1,:));

NOM(2,:) = sum(Q1.*(ones(NC,1)*XU(2,:)).*TAF2,2)';
DENOM(2,1) = sum(Q.*XU(2,:));
```

```
RATIO(2,1) = sum((Q.*XU(2,:).*TAF2(1,:))./NOM(2,1));
RATIO(2,2) = sum((Q.*XU(2,:).*TAF2(2,:))./NOM(2,2));
RATIO(2,3) = sum((Q.*XU(2,:).*TAF2(3,:))./NOM(2,3));
RATIO(2,4) = sum((Q.*XU(2,:).*TAF2(4,:))./NOM(2,4));
RATIO(2,5) = sum((Q.*XU(2,:).*TAF2(5,:))./NOM(2,5));
RATIO(2,6) = sum((Q.*XU(2,:).*TAF2(6,:))./NOM(2,6));
lnGammaR(2,:) = Q.*(1-log(NOM(2,:)./DENOM(2,1))-
RATIO(2,:));

NOM(3,:) = sum(Q1.*(ones(NC,1)*XU(3,:)).*TAF3,2)';
DENOM(3,1) = sum(Q.*XU(3,:));
RATIO(3,1) = sum((Q.*XU(3,:).*TAF3(1,:))./NOM(3,1));
RATIO(3,2) = sum((Q.*XU(3,:).*TAF3(2,:))./NOM(3,2));
RATIO(3,3) = sum((Q.*XU(3,:).*TAF3(3,:))./NOM(3,3));
RATIO(3,4) = sum((Q.*XU(3,:).*TAF3(4,:))./NOM(3,4));
RATIO(3,5) = sum((Q.*XU(3,:).*TAF3(5,:))./NOM(3,5));
RATIO(3,6) = sum((Q.*XU(3,:).*TAF3(6,:))./NOM(3,6));
lnGammaR(3,:) = Q.*(1-log(NOM(3,:)./DENOM(3,1))-
RATIO(3,:));

NOM(4,:) = sum(Q1.*(ones(NC,1)*XU(4,:)).*TAF4,2)';
DENOM(4,1) = sum(Q.*XU(4,:));
RATIO(4,1) = sum((Q.*XU(4,:).*TAF4(1,:))./NOM(4,1));
RATIO(4,2) = sum((Q.*XU(4,:).*TAF4(2,:))./NOM(4,2));
RATIO(4,3) = sum((Q.*XU(4,:).*TAF4(3,:))./NOM(4,3));
RATIO(4,4) = sum((Q.*XU(4,:).*TAF4(4,:))./NOM(4,4));
RATIO(4,5) = sum((Q.*XU(4,:).*TAF4(5,:))./NOM(4,5));
RATIO(4,6) = sum((Q.*XU(4,:).*TAF4(6,:))./NOM(4,6));
lnGammaR(4,:) = Q.*(1-log(NOM(4,:)./DENOM(4,1))-
RATIO(4,:));

NOM(5,:) = sum(Q1.*(ones(NC,1)*XU(5,:)).*TAF5,2)';
DENOM(5,1) = sum(Q.*XU(5,:));
RATIO(5,1) = sum((Q.*XU(5,:).*TAF5(1,:))./NOM(5,1));
RATIO(5,2) = sum((Q.*XU(5,:).*TAF5(2,:))./NOM(5,2));
RATIO(5,3) = sum((Q.*XU(5,:).*TAF5(3,:))./NOM(5,3));
RATIO(5,4) = sum((Q.*XU(5,:).*TAF5(4,:))./NOM(5,4));
RATIO(5,5) = sum((Q.*XU(5,:).*TAF5(5,:))./NOM(5,5));
RATIO(5,6) = sum((Q.*XU(5,:).*TAF5(6,:))./NOM(5,6));
lnGammaR(5,:) = Q.*(1-log(NOM(5,:)./DENOM(5,1))-
RATIO(5,:));

% Calculating the activity coefficents:
lnGamma = lnGammaC + lnGammaR;
Gamma = exp(lnGamma);

% Calculating the total pressure:
Ptot = sum(Gamma.*XU.*Ps,2);
```

```
%% Kinetics

%              A          |   e^(-Ea/R/T)       | [min] |[x-
>kmol]
k1 = 3.9       *(10^7)    *exp(-54998/(R*T))   *60
./vol.*M.^2;
k2 = 5.78      *(10^5)    *exp(-41555/(R*T))   *60
./vol.*M.^2;
k3 = 5.906     *(10^12)   *exp(-83094/(R*T))   *60
./vol.*M.^2;
k4 = 9.888     *(10^9)    *exp(-61249/(R*T))   *60
./vol.*M.^2;
k5 = 5.335     *(10^3)    *exp(-26865/(R*T))   *60
./vol.*M.^2;
k6 = 2.1       *(10^4)    *exp(-40116/(R*T))   *60
./vol.*M.^2;


r(:,2) = -(k1.*xx(:,2).*xx(:,1)) + (k2.*xx(:,3).*xx(:,6));
r(:,3) =  (k1.*xx(:,2).*xx(:,1)) - (k2.*xx(:,3).*xx(:,6)) -
(k3.*xx(:,3).*xx(:,1)) + (k4.*xx(:,4).*xx(:,6));
r(:,4) =  (k3.*xx(:,3).*xx(:,1)) - (k4.*xx(:,4).*xx(:,6)) -
(k5.*xx(:,4).*xx(:,1)) + (k6.*xx(:,5).*xx(:,6));
r(:,6) =  (k1.*xx(:,2).*xx(:,1)) - (k2.*xx(:,3).*xx(:,6)) +
(k3.*xx(:,3).*xx(:,1)) - (k4.*xx(:,4).*xx(:,6)) +
(k5.*xx(:,4).*xx(:,1)) - (k6.*xx(:,5).*xx(:,6));
r(:,5) =  (k5.*xx(:,4).*xx(:,1)) - (k6.*xx(:,5).*xx(:,6));
r(:,1) = -(k1.*xx(:,2).*xx(:,1)) + (k2.*xx(:,3).*xx(:,6)) -
(k3.*xx(:,3).*xx(:,1)) + (k4.*xx(:,4).*xx(:,6)) -
(k5.*xx(:,4).*xx(:,1)) + (k6.*xx(:,5).*xx(:,6));
r_tot = r*ones(6,1);

%% Molar balances:
% Time derivatives from material balances for: (1) total
holdup and (2) component holdup
% Column
j = 2:NT-1; dMdt(j,1)  = L(j+1) - L(j) + V(j-1) - V(j) +
r_tot(j);
i = 1:NC-1; dMxdt(j,i) = diag(L(j+1))*x(j+1,i) -
diag(L(j))*x(j,i) + diag(V(j-1))*y(j-1,i) -
diag(V(j))*y(j,i) + r(j,i);

% Correction for feed at the feed stage: The feed is
assumed to be mixed into the feed stage
dMdt(NF)     = dMdt(NF) + F;
dMxdt(NF,:) = dMxdt(NF,:) + F*zF(1,1:NC-1);

% Reboiler (assumed to be an equilibrium stage)
```

```matlab
dMdt(1)      = L(2) - V(1) - B;
dMxdt(1,i)   = L(2)*x(2,i) - V(1)*y(1,i) - B*x(1,i);

% Total condenser (no equilibrium stage)
dMdt(NT)     = V(NT-1) - L(NT) - D;
dMxdt(NT,i) = V(NT-1)*y(NT-1,i) - L(NT)*x(NT,i) -
D*x(NT,i);

% Compute the derivative for the mole fractions from d(Mx)
= x dM + M dx
dxdt = diag(M)\(dMxdt - diag(dMdt)*x);

% Rearrange elements of composition vector derivatives
(dxdt)
dxdt = dxdt(:);

f3 = 760 - Ptot;

post_processing
```

E.7 output_c.m

```matlab
function f = output_c(X,U,D)


parameters_c
% Splitting the states
x = X(1:(NC-1)*NT);                        % Liquid
compositions from btm to top
M = X((NC-1)*NT+1:NC*NT);                   % Liquid hold up
from btm to top
TL = X((NC*NT)+1:(NC*NT)+NT);              % Temperatures from
btm to top


% Rearrange elements of composition vector (x) for later
use
Iu = reshape(1:(NC-1)*NT,NT,NC-1);
x  = x(Iu);


% Inputs
LT = U(1);      % Reflux
VB = U(2);      % Boilup


% Disturbances
F  = D(1);      % Feedrate
qF = D(2);      % Feed liquid fraction
zF = D(3:end);  % Feed compositions


% The distillation model
flag = 0; % This flag signals the inclusion of equations
for liquid levels. If flag~=0, Simulink is being used and
no liquid level is calculated in the code.
model_c
% epsilon = GL_tot - 0.0025;
epsilon = 0 - LT;


%     1   2  3  4     5           6           7          8        9
10   11    12 13
f = {LT, VB, D, B, x(end,:), x(1,:), L(2:NT), V(1:NT-1), x,
M, TL, GP_per, EP_per, CGL_per, y, rho_l, GL_tot};
```

E.8 print_results_c.m

```matlab
function print_results_c(f)

fprintf('\n\nSolution Point\n')
fprintf('**************\n\n')
fprintf('LT \t %.2f\n', f{1})
fprintf('VB \t %.2f\n', f{2})
fprintf('D  \t %.2f\n', f{3})
fprintf('B  \t %.2f\n', f{4})
fprintf('------------------------------------------\n')
fprintf('xD \t')
for i = 1:length(f{5})
   fprintf(' %.2f\t',f{5}(i))
end


fprintf('\nxB \t')
for i = 1:length(f{6})
   fprintf(' %.2f\t',f{6}(i))
end
fprintf('\n------------------------------------------
\n')

end
```

E.9 reactivedist_c_DAE.m

```matlab
function F = reactivedist_c_DAE(t,Y)
% Essential parameters:
parameters_c

% Inputs and disturbances:
LT = 226.8/60;          % Reflux [kmol/min]
VB = 309.4/60;          % Boilup [kmol/min]

% if t >= 2
%         F = 355/60;
%         zF = [300/355 55/350 0 0 0 0];
%     else
%         F = 350/60;
%         zF = [300/350 50/350 0 0 0 0];
% end
F  = (350/60);          % Feedrate [kmol/min]
qF = 1.064;             % Feed liquid fraction
zF = [300/350 50/350 0 0 0 0]; % Feed compositions for
component x1 to xNC

% II. Extract present value of states
x0 = Y(1:NT*(NC-1),1);          % Compositions of compounds
1-5 on each tray
M0 = Y((NT*(NC-1))+1:NT*NC,1);  % Molar holdup on each tray
[kmol]
T0 = Y((NT*NC)+1:(NT*NC)+NT,1); % Temperature on each tray
[K]
x = reshape(x0,NT,NC-1);        % Reshaping the
compositions
M = reshape(M0,NT,1);           % Reshaping the molar
holdups
TL = reshape(T0,NT,1);          % Reshaping the temperature

% All mole fractions in the sequence: MetOH/TG/DG/MG/GL/BD:
xx = [x,1-sum(x,2)];

% Vapour-liquid equilibria (multicomponent ideal VLE,
Stichlmair-Fair, 'Distillation', p. 36, 1998)
y = (x*diag(alpha1(1:NC-1)))./((x*(alpha1(1:NC-1) - 1)' +
1)*ones(1,NC-1));

%% Density calculations
% Assuming constant temperature:
T = 91.27+273.15;               % [K]

% Molar density
```

```matlab
rho = AR(:,1)./AR(:,2).^(1+(1-(T./AR(:,3)))).^AR(:,4)); %
[kmol/m^3]
N = (M*ones(1,NC)).*xx;      % molar holdup of components
per tray [kmol]

% Volume fractions:
V = N./(ones(NT,1)*rho');   % volume of component per tray
[m3]
Vtot = sum(V,2);            % total volume per tray [m3]
ve = V./(Vtot*ones(1,NC));  % volume fraction [m3/m3]
rho_l = sum((ones(NT,1)*rho').*ve,2);    % Common liquid
density for the tray [kmol/m3]

% Weight fractions:
wt = xx.*(M*ones(1,NC)).*(ones(NT,1)*Mw);
wt_sum = sum(wt,2);
x_wt = wt./(wt_sum*ones(1,NC));

% Common liquid density for each tray [kg/m3]:
r_wt = (ones(NT,1)*rho').*x_wt.*(ones(NT,1)*Mw);
rho_L = sum(r_wt,2);

vol= volume_plate*ones(22,1);



%% Dynamics
% Vapor flows assuming constant molar flows: [kmol/h]
V = VB*ones(NT-1,1); V(NF:NT-1) = V(NF:NT-1) + (1-qF)*F;

% Liquid flows are given by Franci's Weir Formula:
Muw = volume_plate*rho_l;                % Molar holdup
under weir [kmol]
Mw_avg = sum(xx.*(ones(NT,1)*Mw),2);     % Average molecular
weight per tray [kg/kmol]

% Constants for the Francis weir equation:
Kbf = 60*rho_l*lw.*(10^3./(750*area_plate*rho_l)).^1.5;
L(2:NF,1)     = (Kbf(2:NF,1).*(max(M(2:NF) -
Muw(2:NF,1),0)).^1.5);  % Liquid flows below feed
(Kmol/min)
L(NT,1)       = LT;                               %
Condenser's liquid flow (Kmol/min)

%Liquid control in the refulx drum and reboiler: flow
equations for D and B
D = max(Ds + (-M(NT) + MDs)*KcD,0);      % Distillate flow
B = max(Bs + (-M(1) + MBs)*KcB,0);       % Bottoms flow
```

```matlab
%% UNIQUAC model

% Vapour pressure equation where Psat [mmHg] and
temperature T [K]
j = 1:NT;
Ps1(:,1) = (A(1,2)./TL) + (A(1,3)*log(TL)) +
A(1,4)*TL.^A(1,5); % TG
Ps1(:,2) = (A(2,2)./TL) + (A(2,3)*log(TL)) +
A(2,4)*TL.^A(2,5); % DG
Ps1(:,3) = (A(3,2)./TL) + (A(3,3)*log(TL)) +
A(3,4)*TL.^A(3,5); % MG
Ps1(:,4) = (A(4,2)./TL) + (A(4,3)*log(TL)) +
A(4,4)*TL.^A(4,5); % BD
Ps1(:,5) = (A(5,2)./TL) + (A(5,3)*log(TL)) +
A(5,4)*TL.^A(5,5); % GL
Ps1(:,6) = (A(6,2)./TL) + (A(6,3)*log(TL)) +
A(6,4)*TL.^A(6,5); % MetOH
Ps = 760*9.869*(10^-3)*exp((ones(NT,1)*A(:,1)') + Ps1);

%         TG     DG      MG      BD      GL     MetOH
XU = [xx(j,2) xx(j,3) xx(j,4) xx(j,6) xx(j,5) xx(j,1)];

% Calculating the surface area fractions:
Q_tot = (ones(NT,1)*Q).*XU;
Q_sum = sum(Q_tot,2);
F1 = (ones(NT,1)*Q)./(Q_sum*ones(1,6));

% Calculating the volume fraction per mixture mole
fraction:
RPt = (ones(NT,1)*RP).*XU;
RP_sum = sum(RPt,2);
PHI = (ones(NT,1)*RP)./(RP_sum*ones(1,NC));

% Calculating matrix of UNIQUAC coefficients TAF_ij:
TAF1 = exp(-W/(Re*TL(1,1)));
TAF2 = exp(-W/(Re*TL(2,1)));
TAF3 = exp(-W/(Re*TL(3,1)));
TAF4 = exp(-W/(Re*TL(4,1)));
TAF5 = exp(-W/(Re*TL(5,1)));

% Calculating the combinatorial contribution:
lnGammaC = (1-PHI+log(PHI))-((z/2)*(ones(NT,1)*Q).*(1-
(PHI./F1)+log(PHI./F1)));

% Calculating the residual contribution:
Q1 = ones(NC,1)*Q;            % Q as a (6,6) matrix
NOM(1,:) = sum(Q1.*(ones(NC,1)*XU(1,:)).*TAF1,2)';
```

```
DENOM(1,1) = sum(Q.*XU(1,:));
RATIO(1,1) = sum((Q.*XU(1,:).*TAF1(1,:))./NOM(1,1));
RATIO(1,2) = sum((Q.*XU(1,:).*TAF1(2,:))./NOM(1,2));
RATIO(1,3) = sum((Q.*XU(1,:).*TAF1(3,:))./NOM(1,3));
RATIO(1,4) = sum((Q.*XU(1,:).*TAF1(4,:))./NOM(1,4));
RATIO(1,5) = sum((Q.*XU(1,:).*TAF1(5,:))./NOM(1,5));
RATIO(1,6) = sum((Q.*XU(1,:).*TAF1(6,:))./NOM(1,6));
lnGammaR(1,:) = Q.*(1-log(NOM(1,:)./DENOM(1,1))-
RATIO(1,:));

NOM(2,:) = sum(Q1.*(ones(NC,1)*XU(2,:)).*TAF2,2)';
DENOM(2,1) = sum(Q.*XU(2,:));
RATIO(2,1) = sum((Q.*XU(2,:).*TAF2(1,:))./NOM(2,1));
RATIO(2,2) = sum((Q.*XU(2,:).*TAF2(2,:))./NOM(2,2));
RATIO(2,3) = sum((Q.*XU(2,:).*TAF2(3,:))./NOM(2,3));
RATIO(2,4) = sum((Q.*XU(2,:).*TAF2(4,:))./NOM(2,4));
RATIO(2,5) = sum((Q.*XU(2,:).*TAF2(5,:))./NOM(2,5));
RATIO(2,6) = sum((Q.*XU(2,:).*TAF2(6,:))./NOM(2,6));
lnGammaR(2,:) = Q.*(1-log(NOM(2,:)./DENOM(2,1))-
RATIO(2,:));

NOM(3,:) = sum(Q1.*(ones(NC,1)*XU(3,:)).*TAF3,2)';
DENOM(3,1) = sum(Q.*XU(3,:));
RATIO(3,1) = sum((Q.*XU(3,:).*TAF3(1,:))./NOM(3,1));
RATIO(3,2) = sum((Q.*XU(3,:).*TAF3(2,:))./NOM(3,2));
RATIO(3,3) = sum((Q.*XU(3,:).*TAF3(3,:))./NOM(3,3));
RATIO(3,4) = sum((Q.*XU(3,:).*TAF3(4,:))./NOM(3,4));
RATIO(3,5) = sum((Q.*XU(3,:).*TAF3(5,:))./NOM(3,5));
RATIO(3,6) = sum((Q.*XU(3,:).*TAF3(6,:))./NOM(3,6));
lnGammaR(3,:) = Q.*(1-log(NOM(3,:)./DENOM(3,1))-
RATIO(3,:));

NOM(4,:) = sum(Q1.*(ones(NC,1)*XU(4,:)).*TAF4,2)';
DENOM(4,1) = sum(Q.*XU(4,:));
RATIO(4,1) = sum((Q.*XU(4,:).*TAF4(1,:))./NOM(4,1));
RATIO(4,2) = sum((Q.*XU(4,:).*TAF4(2,:))./NOM(4,2));
RATIO(4,3) = sum((Q.*XU(4,:).*TAF4(3,:))./NOM(4,3));
RATIO(4,4) = sum((Q.*XU(4,:).*TAF4(4,:))./NOM(4,4));
RATIO(4,5) = sum((Q.*XU(4,:).*TAF4(5,:))./NOM(4,5));
RATIO(4,6) = sum((Q.*XU(4,:).*TAF4(6,:))./NOM(4,6));
lnGammaR(4,:) = Q.*(1-log(NOM(4,:)./DENOM(4,1))-
RATIO(4,:));

NOM(5,:) = sum(Q1.*(ones(NC,1)*XU(5,:)).*TAF5,2)';
DENOM(5,1) = sum(Q.*XU(5,:));
RATIO(5,1) = sum((Q.*XU(5,:).*TAF5(1,:))./NOM(5,1));
RATIO(5,2) = sum((Q.*XU(5,:).*TAF5(2,:))./NOM(5,2));
RATIO(5,3) = sum((Q.*XU(5,:).*TAF5(3,:))./NOM(5,3));
RATIO(5,4) = sum((Q.*XU(5,:).*TAF5(4,:))./NOM(5,4));
```

```matlab
RATIO(5,5) = sum((Q.*XU(5,:).*TAF5(5,:))./NOM(5,5));
RATIO(5,6) = sum((Q.*XU(5,:).*TAF5(6,:))./NOM(5,6));
lnGammaR(5,:) = Q.*(1-log(NOM(5,:)./DENOM(5,1))-
RATIO(5,:));

% Calculating the activity coefficents:
lnGamma = lnGammaC + lnGammaR;
Gamma = exp(lnGamma);

% Calculating the total pressure:
Ptot = sum(Gamma.*XU.*Ps,2);

%% Kinetics

%          A           |   e^(-Ea/R/T)       | [min] |[x-
>kmol]
k1 = 3.9      *(10^7)   *exp(-54998/(R*T))   *60
./vol.*M.^2;
k2 = 5.78     *(10^5)   *exp(-41555/(R*T))   *60
./vol.*M.^2;
k3 = 5.906    *(10^12)  *exp(-83094/(R*T))   *60
./vol.*M.^2;
k4 = 9.888    *(10^9)   *exp(-61249/(R*T))   *60
./vol.*M.^2;
k5 = 5.335    *(10^3)   *exp(-26865/(R*T))   *60
./vol.*M.^2;
k6 = 2.1      *(10^4)   *exp(-40116/(R*T))   *60
./vol.*M.^2;




r(:,2) = -(k1.*xx(:,2).*xx(:,1)) + (k2.*xx(:,3).*xx(:,6));
r(:,3) =  (k1.*xx(:,2).*xx(:,1)) - (k2.*xx(:,3).*xx(:,6)) -
(k3.*xx(:,3).*xx(:,1)) + (k4.*xx(:,4).*xx(:,6));
r(:,4) =  (k3.*xx(:,3).*xx(:,1)) - (k4.*xx(:,4).*xx(:,6)) -
(k5.*xx(:,4).*xx(:,1)) + (k6.*xx(:,5).*xx(:,6));
r(:,6) =  (k1.*xx(:,2).*xx(:,1)) - (k2.*xx(:,3).*xx(:,6)) +
(k3.*xx(:,3).*xx(:,1)) - (k4.*xx(:,4).*xx(:,6)) +
(k5.*xx(:,4).*xx(:,1)) - (k6.*xx(:,5).*xx(:,6));
r(:,5) =  (k5.*xx(:,4).*xx(:,1)) - (k6.*xx(:,5).*xx(:,6));
r(:,1) = -(k1.*xx(:,2).*xx(:,1)) + (k2.*xx(:,3).*xx(:,6)) -
(k3.*xx(:,3).*xx(:,1)) + (k4.*xx(:,4).*xx(:,6)) -
(k5.*xx(:,4).*xx(:,1)) + (k6.*xx(:,5).*xx(:,6));
r_tot = r*ones(NC,1);

%% Molar balances:
% Time derivatives from material balances for: (1) total
holdup and (2) component holdup
```

```matlab
% Column
j = 2:NT-1; dMdt(j,1)  = L(j+1) - L(j) + V(j-1) - V(j) +
r_tot(j);
i = 1:NC-1; dMxdt(j,i) = diag(L(j+1))*x(j+1,i) -
diag(L(j))*x(j,i) + diag(V(j-1))*y(j-1,i) -
diag(V(j))*y(j,i) + r(j,i);

% Correction for feed at the feed stage: The feed is
assumed to be mixed into the feed stage
dMdt(NF)    = dMdt(NF) + F;
dMxdt(NF,:) = dMxdt(NF,:) + F*zF(1,1:NC-1);

% Reboiler (assumed to be an equilibrium stage)
dMdt(1)    = L(2) - V(1) - B;
dMxdt(1,i)  = L(2)*x(2,i) - V(1)*y(1,i) - B*x(1,i);

% Total condenser (no equilibrium stage)
dMdt(NT)    = V(NT-1) - L(NT) - D;
dMxdt(NT,i) = V(NT-1)*y(NT-1,i) - L(NT)*x(NT,i) -
D*x(NT,i);

% Compute the derivative for the mole fractions from d(Mx)
= x dM + M dx
dxdt = diag(M)\(dMxdt - diag(dMdt)*x);

% Rearrange elements of composition vector derivatives
(dxdt)
dxdt = dxdt(:);

Temp = 760 - Ptot;

F = [dxdt; dMdt; Temp];
```

E.10 output_dynamic.m

```matlab
function f = output_dynamic(X,U,D)

parameters_c

% Inputs
LT = U(1);                              % Reflux
VB = U(2);                              % Boilup

% Disturbances
F  = D(1);                              % Feedrate
qF = D(2);                              % Feed liquid
fraction
zF = D(3:end);                          % Feed compositions

% Splitting the states
x = X(1,1:(NC-1)*NT);                   % Liquid
compositions from btm to top
M = X(1,(NC-1)*NT+1:NC*NT)';            % Liquid hold up
from btm to top
TL = X(1,(NC*NT)+1:(NC*NT)+NT)';        % Temperatures from
btm to top

% Rearrange elements of composition vector (x) for later
use
Iu = reshape(1:(NC-1)*NT,NT,NC-1);
x  = x(Iu);

% The reactive distillation model
model_c_simple


%    1  2  3  4     5        6      7  8  9    10      11
12
f = {L, V, D, B, GP_per, EP_per, GP, EP, CGL, CMet,
CGL_per, CMet_per};
```

E.11 model_c_simple.m

```matlab
%% The reactive distillation model
parameters_c
% All mole fractions in the sequence: MetOH/TG/DG/MG/GL/BD:
xx = [x,1-sum(x,2)];

% Vapour-liquid equilibria (multicomponent ideal VLE,
Stichlmair-Fair, 'Distillation', p. 36, 1998)
y = (x*diag(alpha1(1:NC-1)))./((x*(alpha1(1:NC-1) - 1)' +
1)*ones(1,NC-1));

%% Density calculations
% Assuming constant temperature:
T = 91.27+273.15;              % [K]

% Molar density
rho = AR(:,1)./AR(:,2).^(1+(1-(T./AR(:,3))).^AR(:,4)); %
[kmol/m^3]
N = (M*ones(1,NC)).*xx;       % molar holdup of components
per tray [kmol]

% Volume fractions:
V = N./(ones(NT,1)*rho');     % volume of component per tray
[m3]
Vtot = sum(V,2);              % total volume per tray [m3]
ve = V./(Vtot*ones(1,NC));    % volume fraction [m3/m3]
rho_l = sum((ones(NT,1)*rho').*ve,2);    % Common liquid
density for the tray [kmol/m3]

% Weight fractions:
wt = xx.*(M*ones(1,NC)).*(ones(NT,1)*Mw);
wt_sum = sum(wt,2);
x_wt = wt./(wt_sum*ones(1,NC));

% Common liquid density for each tray [kg/m3]:
r_wt = (ones(NT,1)*rho').*x_wt.*(ones(NT,1)*Mw);
rho_L = sum(r_wt,2);



vol= volume_plate*ones(NT,1);



%% Dynamics
% Vapor flows assuming constant molar flows: [kmol/h]
V = VB*ones(NT-1,1); V(NF:NT-1) = V(NF:NT-1) + (1-qF)*F;
```

```matlab
% Liquid flows are given by Franci's Weir Formula
Muw = volume_plate*rho_l;                % Molar holdup
under weir [kmol]
Mw_avg = sum(xx.*(ones(NT,1)*Mw),2);     % Average molecular
weight per tray [kg/kmol]

% Constants for the Francis weir equation:
Kbf = 60*rho_l*lw.*(10^3./(750*area_plate*rho_l)).^1.5;


L(2:NF,1)      = (Kbf(2:NF,1).*(max(M(2:NF) -
Muw(2:NF,1),0))).^1.5;             % Liquid flows below
feed (Kmol/min)
L(NT,1)        = LT;                          %
Condenser's liquid flow (Kmol/min)

%Liquid control in the refulx drum and reboiler: flow
equations for D and B
D = max(Ds + (-M(NT) + MDs)*KcD,0);      % Distillate flow
B = max(Bs + (-M(1) + MBs)*KcB,0);       % Bottoms flow



%% Kinetics

%           A          |   e^(-Ea/R/T)        | [min] |[x-
>kmol]
k1 = 3.9     *(10^7)   *exp(-54998/(R*T))  *60
./vol.*M.^2;
k2 = 5.78    *(10^5)   *exp(-41555/(R*T))  *60
./vol.*M.^2;
k3 = 5.906   *(10^12)  *exp(-83094/(R*T))  *60
./vol.*M.^2;
k4 = 9.888   *(10^9)   *exp(-61249/(R*T))  *60
./vol.*M.^2;
k5 = 5.335   *(10^3)   *exp(-26865/(R*T))  *60
./vol.*M.^2;
k6 = 2.1     *(10^4)   *exp(-40116/(R*T))  *60
./vol.*M.^2;




r(:,2) = -(k1.*xx(:,2).*xx(:,1)) + (k2.*xx(:,3).*xx(:,6));
r(:,3) =  (k1.*xx(:,2).*xx(:,1)) - (k2.*xx(:,3).*xx(:,6)) -
(k3.*xx(:,3).*xx(:,1)) + (k4.*xx(:,4).*xx(:,6));
r(:,4) =  (k3.*xx(:,3).*xx(:,1)) - (k4.*xx(:,4).*xx(:,6)) -
(k5.*xx(:,4).*xx(:,1)) + (k6.*xx(:,5).*xx(:,6));
r(:,6) =  (k1.*xx(:,2).*xx(:,1)) - (k2.*xx(:,3).*xx(:,6)) +
(k3.*xx(:,3).*xx(:,1)) - (k4.*xx(:,4).*xx(:,6)) +
(k5.*xx(:,4).*xx(:,1)) - (k6.*xx(:,5).*xx(:,6));
```

```
r(:,5) =   (k5.*xx(:,4).*xx(:,1)) - (k6.*xx(:,5).*xx(:,6));
r(:,1) = -(k1.*xx(:,2).*xx(:,1)) + (k2.*xx(:,3).*xx(:,6)) -
(k3.*xx(:,3).*xx(:,1)) + (k4.*xx(:,4).*xx(:,6)) -
(k5.*xx(:,4).*xx(:,1)) + (k6.*xx(:,5).*xx(:,6));
r_tot = r*ones(6,1);

%% Molar balances:
% Time derivatives from material balances for: (1) total
holdup and (2) component holdup
% Column
j = 2:NT-1; dMdt(j,1)  = L(j+1) - L(j) + V(j-1) - V(j) +
r_tot(j);
i = 1:NC-1; dMxdt(j,i) = diag(L(j+1))*x(j+1,i) -
diag(L(j))*x(j,i) + diag(V(j-1))*y(j-1,i) -
diag(V(j))*y(j,i) + r(j,i);

% Correction for feed at the feed stage: The feed is
assumed to be mixed into the feed stage
dMdt(NF)     = dMdt(NF) + F;
dMxdt(NF,:) = dMxdt(NF,:) + F*zF(1,1:NC-1);

% Reboiler (assumed to be an equilibrium stage)
dMdt(1)     = L(2) - V(1) - B;
dMxdt(1,i)  = L(2)*x(2,i) - V(1)*y(1,i) - B*x(1,i);

% Total condenser (no equilibrium stage)
dMdt(NT)    = V(NT-1) - L(NT) - D;
dMxdt(NT,i) = V(NT-1)*y(NT-1,i) - L(NT)*x(NT,i) -
D*x(NT,i);

% Compute the derivative for the mole fractions from d(Mx)
= x dM + M dx
dxdt = diag(M)\(dMxdt - diag(dMdt)*x);

% Rearrange elements of composition vector derivatives
(dxdt)
dxdt = dxdt(:);

post_processing
```

F   Methanol Flash

This appendix shows the simulation of the methanol flash tank in Hysys. The split fractions found by this simulation were used to model the flash tank in the Matlab model.

The flash was simulated as an ideal separation unit at 75°C and atmospheric pressure. The temperature was chosen as the settling tank operates at approximately this temperature. The flash tank is shown in Figure F-1 below.
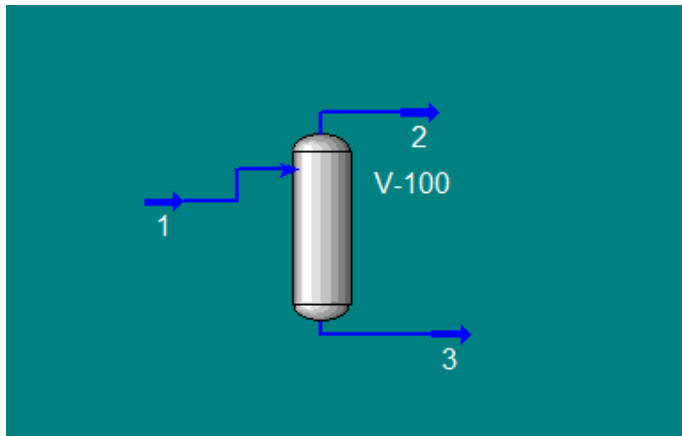

Figure F-1: The flash as simulated in Hysys

The flowrates and compositions for the different streams are given in … below. The inflow used is equal to the GP composition after the settling tank prior to optimization with a basis of 100 kmol/h flowrate.

Table F-1: Flowrates and molar composition of streams 1, 2 and 3

	Stream 1	**Stream 2**	**Stream 3**
Flowrate [kmol/h]	100	60.70	39.30
MetOH [mol%]	60.85	99.59	1.02
TG [mol%]	0.00	0.00	0.00
DG [mol%]	0.22	0.02	0.55
MG [mol%]	0.59	0.00	1.49
GL [mol%]	38.22	0.37	96.68
BD [mol%]	0.12	0.02	0.26

On the basis of Table F-1 the split fractions were calculated and are shown below, specified in terms of the crude glycerol and crude methanol phases:

Table F-2: Split fractions from stream 1 to streams 2 and 3 in terms of molar flowrates

Composition in stream 3	Split fraction stream 1	Split fraction stream 2
MetOH	0.01	0.99
TG	0.00	0.00
DG	0.98	0.02
MG	0.99	0.01
GL	0.99	0.01
BD	0.95	0.05

G Utilities

This appendix represents the calculations and assumptions made in order to find the amount of utilities used for the Matlab model. As the Matlab model contains no energy balances, the amount of low pressure steam and cooling water was instead calculated from the surrounding flows.

G.1 Low Pressure Steam

The low pressure steam (LPS) is used as a heat source in the reboiler. To find a correlation between the necessary amounts of LPS and the affected flowrates, the duty was varied in the Hysys model covering the range of operation for the column. The values of the vapour boilup, the liquid flowrate to the reboiler and the amount of LPS were documented for all duties. These values are summarised in below.

Table G-1: Reported values of LPS, L_{NT+1} and Vb for various duties of reboiler

Duty [kJ/h]	LPS [kg/h]	L_{NT+1} [kmol/h]	Vb [kmol/h]
$1.2 \cdot 10^7$	5,463	533.8	237.3
$1.3 \cdot 10^7$	5,919	547.0	256.1
$1.4 \cdot 10^7$	6,374	559.7	274.4
$1.5 \cdot 10^7$	6,829	572.0	292.3
$1.6 \cdot 10^7$	7,285	584.0	309.7
$1.7 \cdot 10^7$	7,740	595.6	326.5
$1.8 \cdot 10^7$	8,195	606.7	342.7
$1.9 \cdot 10^7$	8,651	617.2	358.3
$2.0 \cdot 10^7$	9,106	627.2	373.1
$2.1 \cdot 10^7$	9,561	636.4	387.0
$2.2 \cdot 10^7$	10,020	644.9	400.0
$2.3 \cdot 10^7$	10,470	652.6	411.9
$2.4 \cdot 10^7$	10,930	659.3	422.8
$2.5 \cdot 10^7$	11,380	665.1	432.4

Simple trend lines were tested for the plots, and one of the best fits proved to be a simple exponential correlation between LPS and Vb. The plot is shown below in Figure G-1**Error! Reference source not found.**.
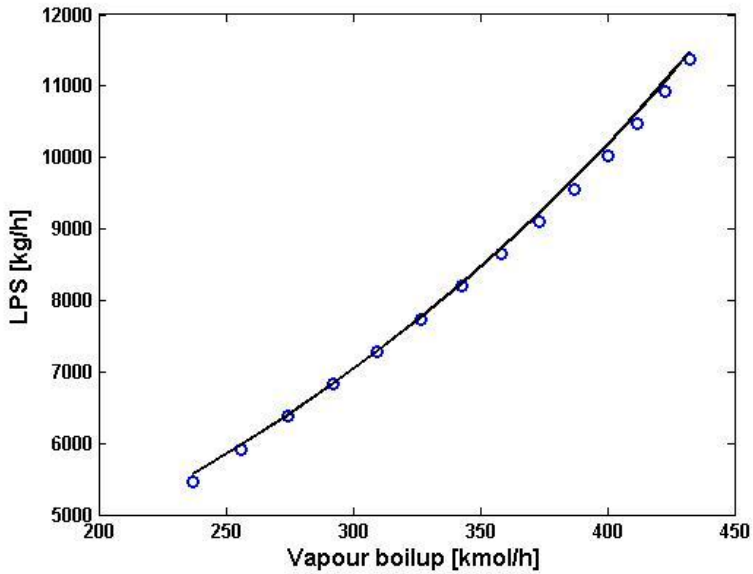
Figure G-1: Plot of vapour boilup against corresponding amount of low pressure steam with added trendline

The trendline found for this case is given by equation G-1 below with an R^2 value of 0.9993.

$$LPS = 2316.9 \exp(0.0037 \cdot Vb) \qquad \text{(G-1)}$$

G.2 Cooling Water

The method for finding a correlation between the amount of cooling water and the surrounding flowrates is similar to the method used for the LPS. Also here, the vapour boilup was varied for the system and the corresponding cooling water requirements and vapour flowrates into the condenser were documented. These values are given below in Table G-2.

Table G-2: Reported values of LPS, L_{NT+1} and Vb for various duties of reboiler

Duty [kJ/h]	CW [kg/h]	V_1 [kmol/h]
$1.2 \cdot 10^7$	372,600	214.0
$1.3 \cdot 10^7$	411,800	236.5
$1.4 \cdot 10^7$	450,400	258.9
$1.5 \cdot 10^7$	488,200	280.9
$1.6 \cdot 10^7$	525,300	302.5
$1.7 \cdot 10^7$	561,500	323.6
$1.8 \cdot 10^7$	596,800	344.2
$1.9 \cdot 10^7$	631,100	364.2
$2.0 \cdot 10^7$	664,300	383.6
$2.1 \cdot 10^7$	696,200	402.3
$2.2 \cdot 10^7$	727,000	420.3
$2.3 \cdot 10^7$	756,300	437.5
$2.4 \cdot 10^7$	784,200	453.8
$2.5 \cdot 10^7$	810,600	469.3

The plot of V1 against the cooling water required is completely linear as shown in Figure G-2 below.
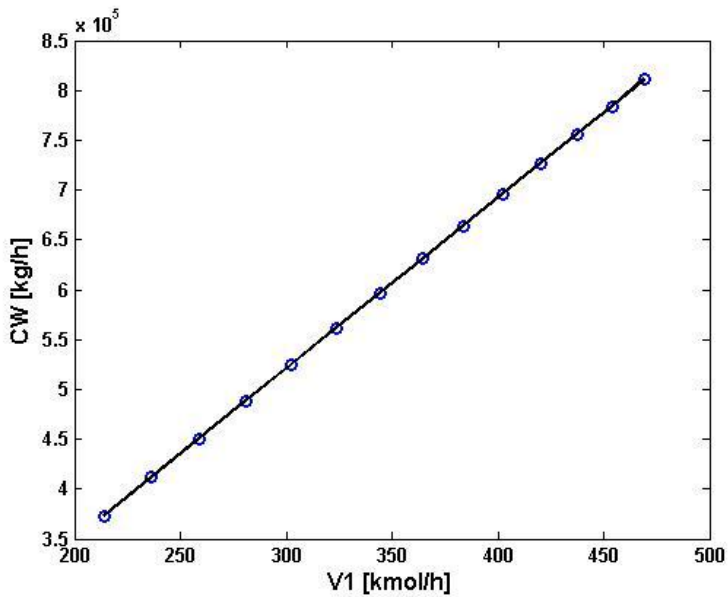

Figure G-2: Plot of vapour flowrate against corresponding amount of cooling water with added trendline

LVIII

The linear trend line of Figure G-2 is given as equation G-2 below:

$$CW = 1714.3 \cdot V_1 + 6476.8 \qquad \text{(G-2)}$$

As with the LPS, the amount of cooling water is also given in kg/h. In order to convert this to m³/h, the water is assumed to have a density of 1000 kg/m³.