Statistical Modeling of Separator Processes

An Application of Gaussian Processes with Bayesian Optimization

Tim Svensson

Master's thesis
2020:E54

Lund Institute of Technology
Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Centrum Scientiarum Mathematicarum

Abstract

The separator is a machine with many applications, commonly used to separate liquids or solids into components with different density. Each application demands its own unique set of process parameters to achieve optimal results. Often the procedure of finding the best process parameters is conducted empirically, which can be very time consuming. This thesis aims to address this problem by providing a statistical model of separator processes, which can be used to find the optimal process parameters more efficiently.

Different sensors are mounted on two separators used in regular production. The sensors measure values of the inputs, outputs and process parameters. A Gaussian process is used to model the regression relationships between the process parameters and outputs for two separators. Bayesian optimization is then used to find optimal process parameters, which are shown to be accurate in simulations. In four models, one for each output of the two separators, the optimal process parameters are seen to improve the outputs. In the first separator only small improvements can be seen, as the optimal process parameter is near the middle of the data used to build the model. In the second separator large improvements can be seen. Here, the optimal process parameter is at the upper endpoint of the interval, implicating that a higher value of the process parameter could further improve the outputs. Thus, further experiments with a higher value of the process parameter are needed in order to draw conclusions on the optimal process parameter for the second separator.

These optimal process parameters will be used in the real separators to possibly improve the separator performance. The data used in this thesis is supplied by the manufacturer of the separators, Alfa Laval.

Acknowledgements

This thesis was conducted in cooperation with Decerno and Alfa Laval, which provided me with an opportunity to be part of two exciting working environments of both an IT consultant and an industrial company. During my entire time there, both companies were very supportive and supplied me with everything I needed to conduct this research.

I would first like to thank my thesis supervisor at Lund University, Johan Lindström, for his valuable inputs and suggestions.

I would like to thank Lennart Haggård and Harald Hermansson at Decerno for initiating this thesis, and to express special gratitude to Harald for providing me with continuous feedback and coming up with ideas to test.

I would like to thank Magnus Sundin, Hogir Rasul and Shota Nozadze at Alfa Laval for explaining all the intricacies of the separator and the fruitful and constructive dialogues we have had. I really hope that the results of this thesis will be useful.

Of course, I would also like to thank everyone at both companies who have helped me during these last few months.

5th of June 2020

A Statistical Approach to Separator Optimization

By using real data from regular product production of two separators, it is shown that the statistical model manages to create a good model of the separation process. Further, the results show that the process parameters can be altered in order to improve multiple objectives of the separation process simultaneously.

The separator is a machine with many applications, commonly used to separate liquids or solids into components with different density. In order to improve its performance, testing is often conducted empirically, which can be costly and time-consuming. Each application demands its own unique set of process parameters to achieve optimal results. To find these process parameters deep knowledge of the specific application is needed. By using a statistical approach, the optimal process parameters can be found more efficiently with less specific knowledge required.

In order to build a statistical model, data is needed. This data is collected by mounting many different sensors on a separator. The sensors measure values of the inputs, outputs and process parameters of the separator. It is easy to understand that the outputs depend on both what is put in the separator and the settings of the separator, called process parameters. Therefore, it is of interest for the operator to find the best process parameters, in order to achieve the best possible outputs.

A statistical model which can be used for most data, from financial applications to industrial separation, is the Gaussian process. It can be used to simulate the real separation process. This allows for quick testing of different settings and scenarios. These tests and simulations can be used to shed light on possible improvements of the process parameters.

In this work, two outputs of each separator were regulated with only one process parameter. Four different Gaussian process models, one for each output of the two separators, were built. These models were then used to find four different optimal process parameters. It was shown that both outputs had almost exactly the same optimal process parameter, which means there is a single best setting for both outputs of each separator.

This statistical approach is aimed to increase the understanding of the separator process, but more importantly lessen the need of real testing. It is easy use for any separator application and would be ideal to help guide an inexperienced user.

Contents

List of Abbreviations

C The regulating valve controls the position of the disc stack. The disc decides the position of the interface between LP and HP in the separator bowl.

C1 The regulating valve position of separation step 1.

C2 The regulating valve position of separation step 2.

HP The heavy phase is the liquid in the separator that has a higher density. In this thesis the heavy phase is the cleaning medium.

HP1 The heavy phase of separation step 1.

HP2 The heavy phase of separation step 2.

IN The inlet of the separator is the feed that enters the separator in order to be separated.

IN1 The inlet of separation step 1.

IN2 The inlet of separation step 2. Before $LP1$ is led to $IN2$ more of the cleaning medium is added.

LP The light phase is the liquid in the separator that has a lower density. In this thesis the light phase is the product.

LP1 The light phase of separation step 1.

LP2 The light phase of separation step 2.

T Seconds from last discharge. When the separator is discharging the separator withdraws the disc stack to prevent product loss.

T1 Seconds from last discharge of separation step 1. This variable is not used in any of the Gaussian process models. $T1$ is 10 minutes.

T2 Seconds from last discharge of separation step 2. $T2$ is 10 minutes.

Introduction

Separation of liquids or solids is an essential process in numerous industries all over the world. The fields of application are diverse and include: food and beverage production, oil and grease processing, fuel cleaning and biopharmaceutical production to name a few. The separator is a machine that performs the tasks of separating liquids of different densities or removing solids from liquids. The separators studied in this thesis are separators in one of the above mentioned applications, manufactured by Alfa Laval.

When cleaning and processing the product there are significant losses related to the separation process. These losses occur when steps are taken to remove unwanted solids in order to clean the product. It is possible to reduce the product losses by adjusting and optimizing the separator's process parameters.

Currently the separator's process parameters are improved by long periods of empirical testing and expert knowledge is key to improve the separation process. This makes the procedure of running a separator without much experience a hard task.

The purpose of this thesis is to implement a statistical approach towards optimizing the separation process, which does not require expert knowledge of the separation process. The process parameter optimized is the regulating valve, which is a device that regulates the area where the separation takes place. A Gaussian process is used to model the separation activity and Bayesian optimization is used to determine an optimal position for the regulating valve.

1.1 Related work

Herwin (2019) implemented a method for process parameter optimization of a Marine oil-water through the use of a Gaussian process combined with a basin hopper optimizer. He did this in collaboration with Alfa Laval and Decerno. This thesis is a continuation of Alfa Laval's efforts to optimize separators with a statistical approach.

The use of Gaussian processes in combination with Bayesian optimization for different time series and regression analysis has a wide range of applications. Two of

many applications are shortly summarized.

Frazier P.I. (2016) used Gaussian process regression to simulate the results of different material design options. Bayesian optimization was then used to choose the materials for real life experiments. The purpose of this approach was to reduce the number of experiments, improving the efficiency of the design process.

Gonzalvez et al. (2019) implemented a Gaussian process in combination with Bayesian optimization for two applications, yield curve modeling and construction of trend following strategies. Gaussian processes did an equivalent job of predicting yield curves when compared to traditional econometric methods. Further, the Gaussian process with Bayesian optimization strategy showed improvements for trend following strategies.

The Gaussian process and Bayesian optimization have been extended to include multiple tasks with multiple objectives. Bonilla, Chai, and C. Williams (2008) proposed a model where multiple output variables learn from the same covariance function and only learn from each other when it improves the model performance. Biswajit Paria (2019) implement a flexible framework where the practitioner can choose the priorities of different objectives. This framework would be ideal for a separator with many process parameters affecting multiples output variables.

1.2 Overview

In chapter 2 the basic function of a separator, the measured signals and the variables prepared for the Gaussian process are described.

In chapter 3 the theory and implementation of the Gaussian process and Bayesian optimization are explained. Firstly, a Gaussian process model of the separator process is built. Secondly, this model is used to create an objective function, which is then used to find the optimal regulating valve position with Bayesian optimization. Finally, the procedure of evaluating how the regulating valve position affects the outputs of the separators is described.

In chapter 4 the performance of four Gaussian process models, one for each output of the separators, is shown. Using these models, four optimal regulating valve candidates are presented. Lastly, the conclusions on how the regulating valve position influences the separator performance are given.

In chapter 5 the data, the models and the optimization procedure are discussed, and the final conclusions drawn in this thesis are presented.

Separation & Signals

In this chapter the basic function of a separator is described in section 2.1. Further the setup of the separators used in this thesis are outlined in section 2.2 and a summary of the data collected used in this the thesis is provided in section 2.3.

2.1 Separator

As mentioned in the introduction, the separator is a machine that separates liquids of different densities or removes solids from liquids. The liquid of higher density is called the heavy phase *HP* and the liquid of lower density is called light phase *LP*. The separator's capacity Q is the product of the centrifugal settling velocity V_c and the settling area A:

$$Q = V_c A \tag{2.1}$$

The settling area is given by the separator disc stack. The disc stack consists of a large number of discs (3) stacked upon each other, shown in figure 2.1. A short distance between the discs allows particles to quickly separate. The centrifugal settling velocity is given by Stoke's law:

$$V_c = \frac{d^2(\rho_{HP} - \rho_{LP})}{18\eta} r\omega^2 \tag{2.2}$$

where d is the droplet diameter, ρ_{HP} is the density of *HP*, ρ_{LP} is the density of *LP*, η is the continuous phase viscosity and $r\omega^2$ is the centrifugal acceleration. All variables in equation 2.2 except $r\omega^2$ are properties of the liquids or solids being separated. The centrifugal acceleration $r\omega^2$ is decided by the separator's radius r and angular velocity ω^2.

The separators investigated in this thesis are 3-phase separators. The 3 phases in these separators are: *HP*, *LP* and unwanted solids. A 3-phase separator is illustrated in figure 2.1 and described in the following paragraph.

Through the inlet (1) at the top of the separator, the product enters the separator bowl (2). The separation takes place in the disc stack (3), where centrifugal force pushes the *HP* and solids towards the periphery of the bowl, while the *LP* moves to the centre of the bowl. The *HP* is led over the top discs (4) and exits at the top of the separator through the heavy phase outlet (5). The *LP* moves along the centre of the bowl and exits at the top of the separator through the light phase outlet (6). The solids are collected at the solids holding place (7) and are discharged automatically through the discharge port (8) at regular pre-set time intervals. When the solids are discharged, the regulating valve (9) is adjusted to ensure that there is no *LP* loss. The regulating valve is mounted at the top of the separator and decides the position between LP and HP, called the interface, in the separator bowl.

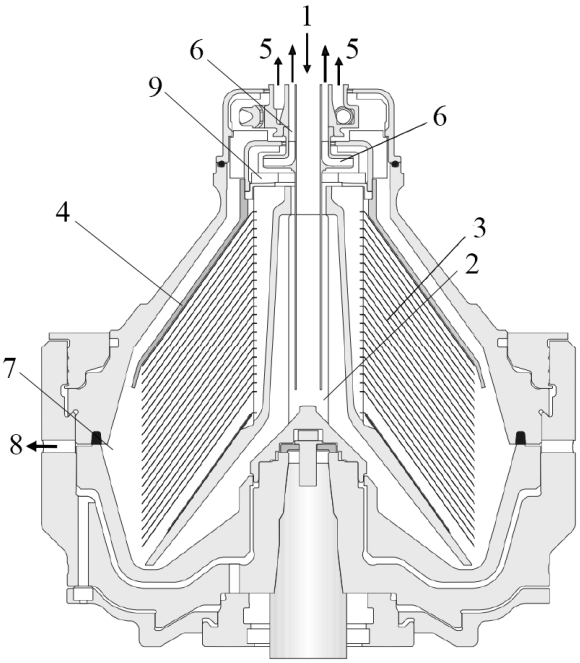


Figure 2.1: A 3-phase disc stack separator by Alfa Laval (Alfa Laval, 2020).
1. Inlet; 2. Separator bowl; 3. Disc stack; 4. Top discs; 5. Heavy phase outlet; 6. Light phase outlet; 7. Solids holding place; 8. Discharge ports; 9. Regulating valve.

2.2 Separator setup

In this thesis two 3-phase separators that work in sequence to clean the product are studied. The separator setup and the signals measured are outlined in the piping and instrumentation diagram in figure 2.2. In both separation steps the product is the *LP* and a medium, called cleaning medium, is used to clean the product is the *HP*. Since the product has a lower density than the cleaning medium, it is the *LP* both separation steps.

In the first separation step, called step 1, cleaning medium is added to remove unwanted solids from the product. Before the *LP* from separation step 1 enters the second separation step, called step 2, more of the cleaning medium is added to further clean the product from unwanted solids and fluids. On both separators the regulating valve position is measured at the top of the separator.

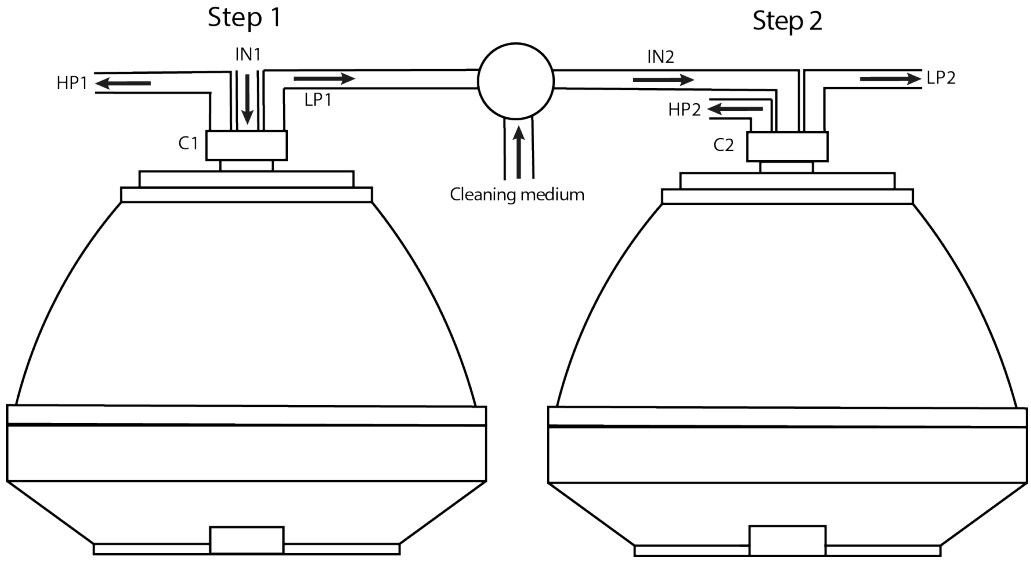


Figure 2.2: Piping and instrumentation diagram of both separation steps working in sequence to clean the product. The signals measured from separation step 1 are denoted 1 and the signals measured from separation step 2 are denoted 2. *IN* is the inlet sensor, *LP* is the heavy phase sensor, *LP* is the light phase sensor and *C* is the regulating valve position.

2.3 Data

The signals measured from separation step 1 are denoted 1 and the signals measured from separation step 2 are denoted 2 (see figure 2.2). The signals are subdivided into two groups, $\mathbf{X}$ and $\mathbf{Y}$. Where $\mathbf{X}$ is the input variable domain, comprised of the input variables $\mathbf{x}$: regulating valve position, inlet and seconds from last discharge. Only the regulating valve position is adjustable while the rest of the input variables are non-adjustable. $\mathbf{Y}$ is the output variable domain, comprised of the output variables $\mathbf{y}$: *LP* and *HP*. The signals measured, their mean, variance and abbreviations are shown in table 2.1.

Table 2.1: Signals measured and their mean, variance and abbreviation. The variables are normalized with equation 2.3. The positions of the different sensors are shown in figure 2.2.

	Abbreviation	Mean	Variance	Signal
X	*C1*	0,89	0,083	Regulating valve 1
	C2	0.54	0.074	Regulating valve 2
	IN1	0.38	0.083	Inlet 1
	IN2	0.54	0.0062	Inlet 2
	T1	0.45	0.096	Seconds from last discharges 1
	T2	0.5	0.084	Seconds from last discharges 2
Y	*LP1*	0.45	0.034	Light phase 1
	LP2	0.75	0.024	Light phase 2
	HP1	0.63	0.0145	Heavy phase 1
	HP2	0.37	0.037	Heavy phase 2

In this thesis a data set measured during 35 hours and 7 minutes of regular product production is used. The signals are measured every second resulting in a data set $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{i=1}^{N}$ of $N = 126\,400$. The data during discharges is removed in order to build a more accurate Gaussian process model. It is of interest to have a model optimal for the separator activity where the regulating valve position is adjustable. The reduction of $\mathcal{D}$ results in a decrease in data points of 10 % for the variables of separator 1 and 3.33 % for the variables of separator 2.

There is a risk of numerical instability when using the variables in the methods of this thesis. In order to reduce the risk of numerical instability the variables are normalized with Min-Max feature scaling. Min-Max feature scaling brings all the values of the variables into the range $[0, 1]$ without distorting differences. This type of scaling is used in order to avoid bias between the variables.

$$\mathcal{D}_{scaled} = \frac{\mathcal{D} - \mathcal{D}_{min}}{\mathcal{D}_{max} - \mathcal{D}_{min}} \tag{2.3}$$

In sections 2.3.1 and 2.3.2 the signals and the variable preparation are described.

2.3.1 Regulating valve & Seconds from last discharge

The regulating valve decides the position of the interface between LP and HP in the separator bowl. Both separation steps have a predetermined regulating valve position deemed optimal. When the separator is discharging unwanted solids, the regulating valve is closed to prevent product loss, which produces a step function signal. The regulating valve positions of both separators have the same appearance and time during discharge (1 minute), the only difference is the time between discharges. Separator 1 has a time between discharges of 9 minutes and separator 2 has a time between discharges of 29 minutes. Though the regulating valve position is predetermined, the positioning mechanism is not exact and the disc stack ends up with a slightly different regulating valve position after every discharge.

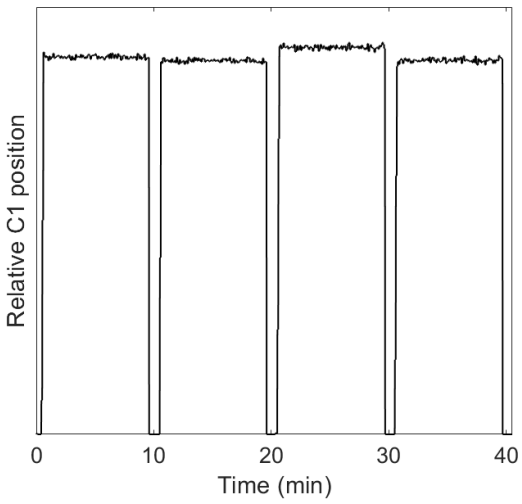


Figure 2.3: The noisy measured regulating valve position.

As the regulating valve position measured is naturally noisy, the measured signal does not correspond to the true regulating valve position. In figure 2.3 the measured noisy regulating valve position is shown before the data during discharges has been cleared.

The regulating valve position is prepared as following: The regulating valve position is assumed to be the average of all the data points in between discharges. To improve the Gaussian process model the seconds from last discharge T is added to the input variable space $\mathbf{X}$. For both separators $T1$ and $T2$ move in the same repeating pattern, since the time between discharges is predetermined. The prepared regulating valve position and the time between discharge for separator 1 are shown in figure 2.4.

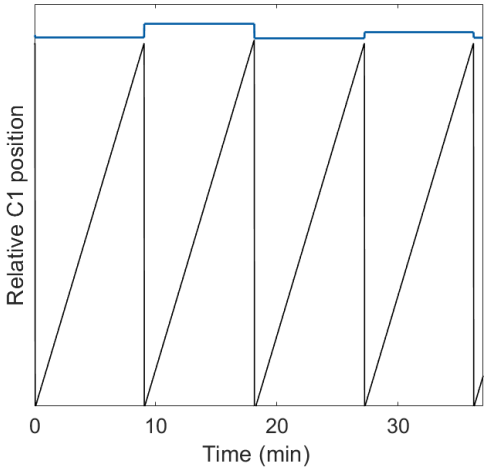


Figure 2.4: The prepared regulating valve position $C1$ in blue and seconds from last discharge $T1$ in black.

2.3.2 Inlet, LP & HP

In this thesis the inlet, LP and HP from the separators are measured with two types of sensors during two different time periods. The two types of sensors are denoted sensor type A and sensor type B. During the time period when the Inlet, LP and HP were measured with sensor B, the regulating valve position sensors were not operational. Thus, the data from sensor B is only used for comparison with the data from sensor A. Both sensors are uncalibrated, hence only the variance of the signals is relevant.

The sampling frequency f_s is the number of samples recorded per second. The sampling frequency is 1 Hz for sensor type A and 0.2 Hz for sensor type B. The signals from sensor type A were measured previously to the signals from sensor type B. The time periods when the signals were measured with the two different sensor types do not overlap. All 6 signals measured with sensor type A and sensor type B are shown in figure 2.5, where the means of the signals have been shifted to improve visibility. The data from sensor type A seen in the figures is a subset of $\mathcal{D}$ matching the time scale of the data from sensor type B.

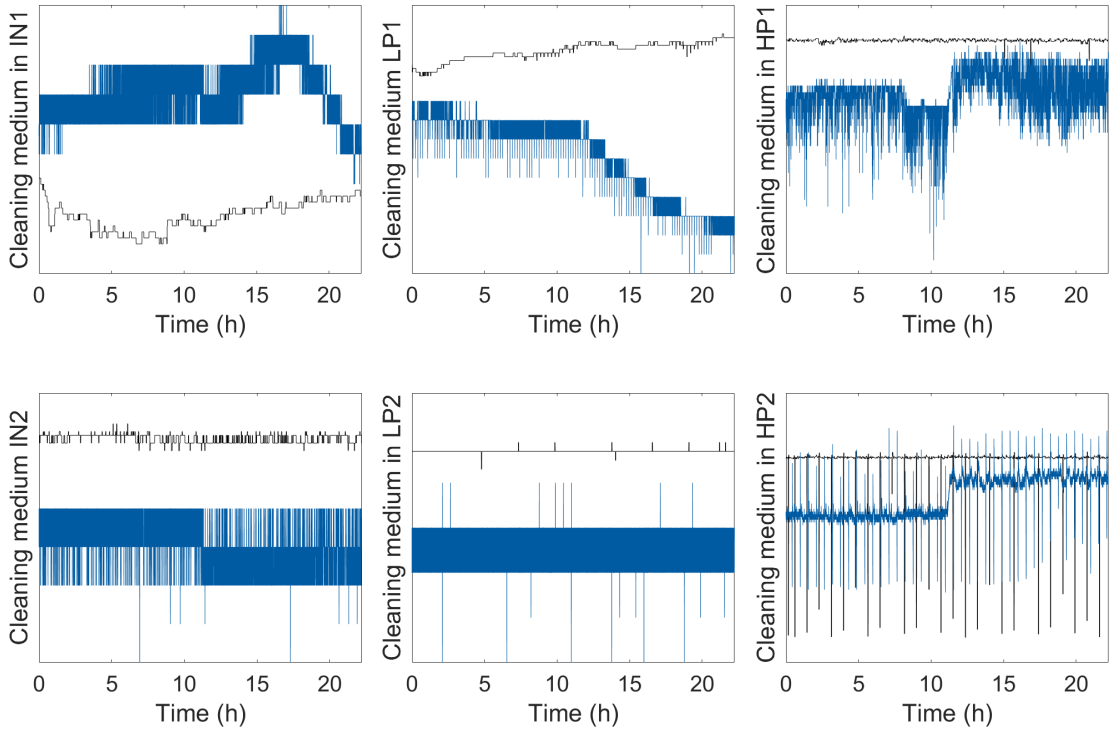


Figure 2.5: The signals from sensor type A are shown in blue and the signals from sensor type B are shown in black.

The signals measured with sensor type A and type B exhibit the same structural behaviors, but the signals measured with sensor type A are much nosier with worse resolution. The noise is caused by temporal aliasing, which causes the sampled signal to distort from the original continuous signal. Temporal aliasing can occur when a continuous signal is sampled in time, giving a noisy signal.

In order to remove the noise and uncover a clearer signal, a low pass filter can be used. The filter removes the higher frequencies of a signal above a cutoff frequency, f_c, while keeping the lower frequencies of the signal. In this thesis a cutoff frequency of $f_c = 0,01\,\text{Hz}$ and a low pass filter of the 10th order is used. The filter used is non-causal, which means it uses information that has not yet occurred to filter the signal. Therefore this type of filter could not be used in a real time application to filter a signal, it is limited to post analysis of signals. The filter is designed with MATLAB (*MATLAB* 2018), a software primarily used for numerical programming.

The original signals for the entire data set $\mathcal{D}$ are shown in figures 2.6 and 2.7, the filtered signals are shown in figures 2.8 and 2.9. The reason that there is less data points for separator 1 (figures 2.6 and 2.8) than separator 2 (figures 2.7 and 2.9) is that separator 1 discharges more often, causing more data to be removed from $\mathcal{D}$.

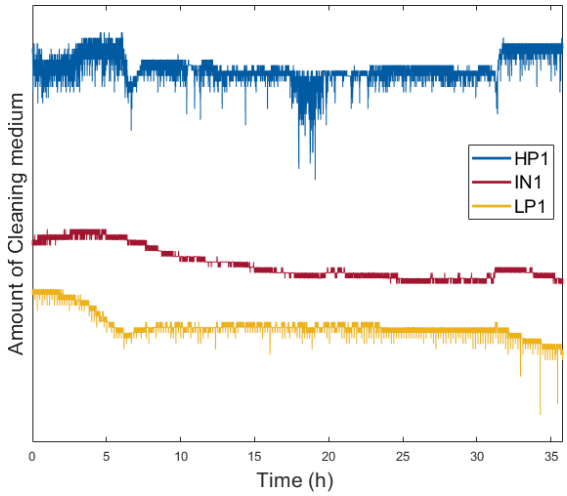


Figure 2.6: The original *IN1*, *LP1* and *HP1*.

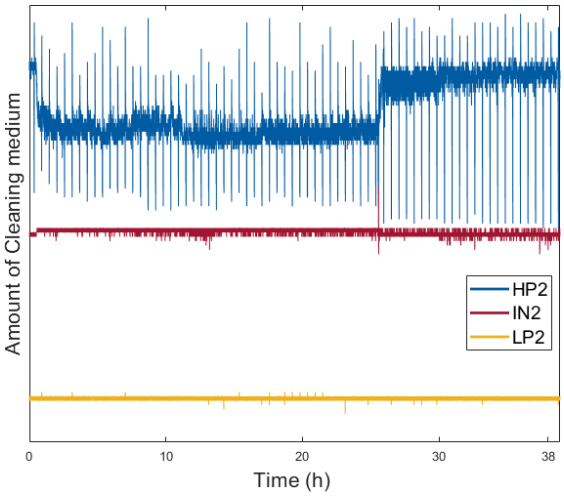


Figure 2.7: The original *IN2*, *LP2* and *HP2*.

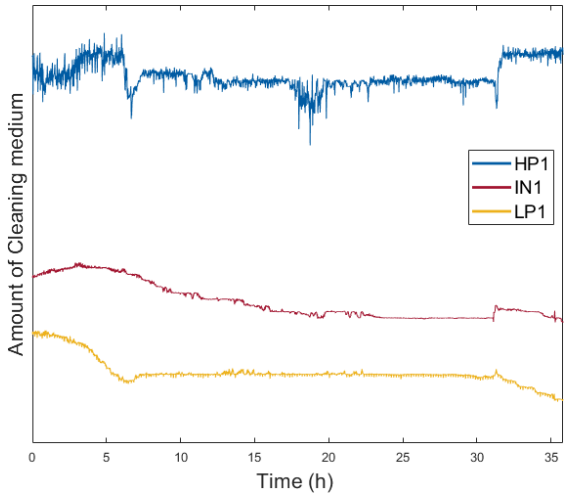


Figure 2.8: The low pass filtered *IN1*, *LP1* and *HP1*.

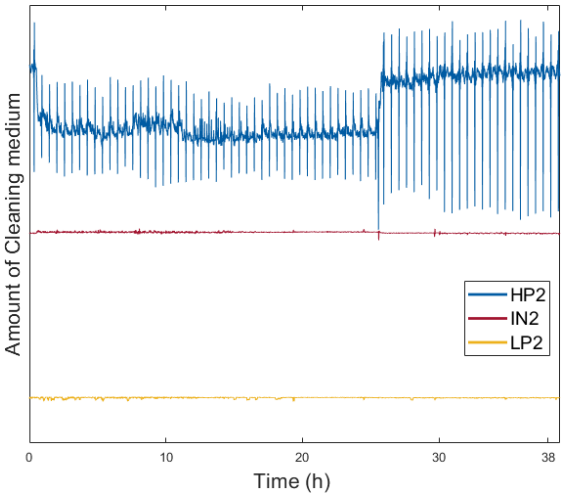


Figure 2.9: The low pass filtered *IN2*, *LP2* and *HP2*.

In separation step 1 *HP1* has a high variance. *LP1* and *IN1* have much smaller variances and are correlated, shown in figure 2.10. Changes in all signals of separator 1 can be easily observed. In separation step 2 *IN2* and *LP2* are almost constant. *HP2* moves in an oscillating pattern with strong transients. The transients occur when the separator is discharging, as shown in figure 2.11.

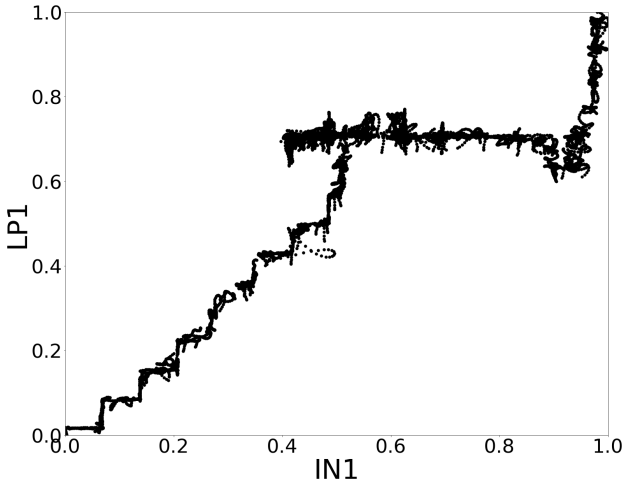


Figure 2.10: Scatter plot of the fraction of the cleaning medium in *LP1* and *IN1* using the same data as in figure 2.8.

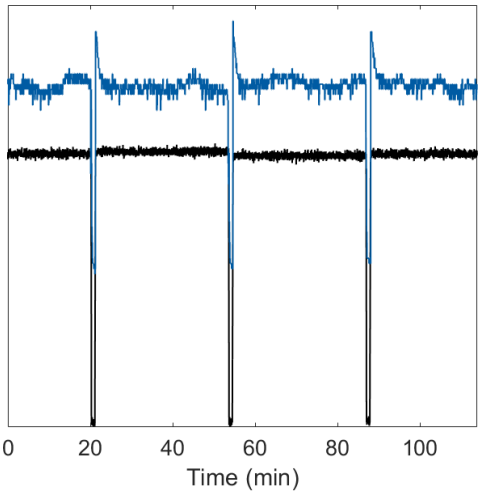


Figure 2.11: *C2* in black and *HP2* in blue. The transients in *HP2* occur when the separator is discharging, which can be seen when *C2* goes to zero as the regulating valve is closed.

Methods & Model

Regression is the statistical modeling of relationships between output and input variables. It can be used for forecasting and prediction. The data set $\mathcal{D}$ is split into a training $\mathcal{D}_{train}$, a validation $\mathcal{D}_{val}$ and a testing set $\mathcal{D}_{test}$. The training set is used to fit the regression model. The validation set is used to evaluate the model fit and to estimate the optimal input variables. The testing set is used to evaluate the estimated optimal input variables. The division of data is: 80 % for training, 10 % for validation and 10 % for testing.

Figure 3.1: The partition of the data $\mathcal{D}$ and the task done with each data set.

In section 3.1 a Gaussian process is used for the regression modeling. The regression model is then used to create an objective function in a Bayesian optimizer to determine optimal input parameters, as described in section 3.2. The results of the optimization and the evaluation of optimal input candidates are given in section 3.3.

The Gaussian process is implemented with GPyTorch (Gardner et al., 2018), a library built on Python, which can be found at `gpytorch.ai`. The Bayesian optimization is implemented with BoTorch (Balandat et al., 2019), a library built on Python, which can be found at `botorch.org`. BoTorch works seamlessly with models from GPyTorch.

3.1 Gaussian process

A Gaussian process is a family of random variables, where every finite collection of them has a multivariate Gaussian distribution (Rasmussen and C. K. I. Williams, 2006a). Thus, it is assumed that all data used is Gaussian distributed. The Gaussian process can be used to fit a function f to a set of data $\mathcal{D}$. The approximation, $f(\mathbf{x})$, of the output $\mathbf{y}$ at input $\mathbf{x}$ is:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \tag{3.1}$$

The Gaussian process is fully defined by its mean function $\mu(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$, commonly referred to as the kernel function.

$$\begin{cases} \mu(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})] \\ k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - \mu(\mathbf{x})(f(\mathbf{x}') - \mu(\mathbf{x}')] \end{cases} \tag{3.2}$$

The mean function of the Gaussian process encodes the central tendency of the underlying function, which is often assumed to be constant. The covariance function encodes the shape and structure of the underlying function. The covariance function must be symmetric and positive definite. The signals measured from the separators are noisy. Therefore, the noisy observations is considered in the Gaussian process inference:

$$\mathbf{y} = f(\mathbf{x}) + \delta \tag{3.3}$$

where f are the noise free observations and $\delta \sim \mathcal{GP}(0, \sigma^2)$ is Gaussian zero-mean noise. Gaussian process regression is often seen as a Bayesian inference problem with a prior distribution $p(\mathbf{y})$ and a posterior distribution $p(\mathbf{y}|\mathcal{D})$. The prior distribution captures a prior belief of probable observations, $\mathbf{y}$, before any data, $\mathcal{D}$, has been observed and the posterior distribution captures the updated belief of probable observations after data has been observed. Given a finite set of input values $\mathbf{X} \in \mathbb{R}^{\mathrm{n}\times\mathrm{d}}$, where n is the number of training inputs and d is the dimension, the Gaussian process prior on the noisy observations $\mathbf{y}$ is:

$$p(\mathbf{y}|\mathbf{X}) = \mathcal{GP}(\mu(\mathbf{X}), K(\mathbf{X}, \mathbf{X})) \tag{3.4}$$

where $K(\mathbf{X}, \mathbf{X})$ is the $n \times n$ covariance matrix. In order to make predictions for a set of testing input values $\mathbf{X}^* \in \mathbb{R}^{\mathrm{n}^*\times\mathrm{d}}$, where n^* is the number of testing inputs, the joint distribution is considered. Given a Gaussian process with gaussian observation noise, the joint distribution of training, $\mathbf{y}$, and function values at the test inputs values, $\mathbf{f}^*$, is also Gaussian:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} \mu(\mathbf{X}) \\ \mu(\mathbf{X}^*) \end{bmatrix}, \begin{bmatrix} K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I} & K(\mathbf{X},\mathbf{X}^*) \\ K(\mathbf{X}^*,\mathbf{X}) & K(\mathbf{X}^*,\mathbf{X}^*) \end{bmatrix} \right) \tag{3.5}$$

here the covariance matrix consists of: the $n^* \times n$ matrix $K(\mathbf{X}^*, \mathbf{X})$, the $n^* \times n^*$ matrix $K(\mathbf{X}, \mathbf{X}^*)$ and the $n^* \times n^*$ matrix $K(\mathbf{X}^*, \mathbf{X}^*)$. For a multivariate Gaussian distribution conditional and marginal distributions are also Gaussian. By conditioning the joint Gaussian prior distribution on $\mathbf{X}^*$ and the training observations, $\mathcal{D} = (\mathbf{X}, \mathbf{y})$, the predictive posterior is obtained as:

$$\begin{aligned} p(\mathbf{f}^*|\mathcal{D}, \mathbf{X}^*) = \mathcal{GP}\Big(&\mu(\mathbf{X}^*) + K(\mathbf{X},\mathbf{X})\mathbf{V}^{-1}(\mathbf{y} - \mu(\mathbf{X}), \\ &K(\mathbf{X}^*,\mathbf{X}^*) - K(\mathbf{X}^*,\mathbf{X})\mathbf{V}^{-1}K(\mathbf{X},\mathbf{X}^*)\Big) \end{aligned} \tag{3.6}$$

where $\mathbf{V} = K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I}$. The Gaussian process defines the probability distribution of the possible outcomes. In order to visualize this, the predictive posterior distribution is calculated with equation 3.6, where the data set $\mathcal{D}_{train}$ is used for fitting the model and an evenly spaced grid of test inputs, $\mathbf{X}^*$, is used for prediction. The procedure of training the model is given in the next section, where the covariance function's parameters are estimated to fit the underlying functions as defined by the training data. In figure 3.2 the conditioned mean of the predictive posterior distribution given the covariance function's fitted parameters is shown.

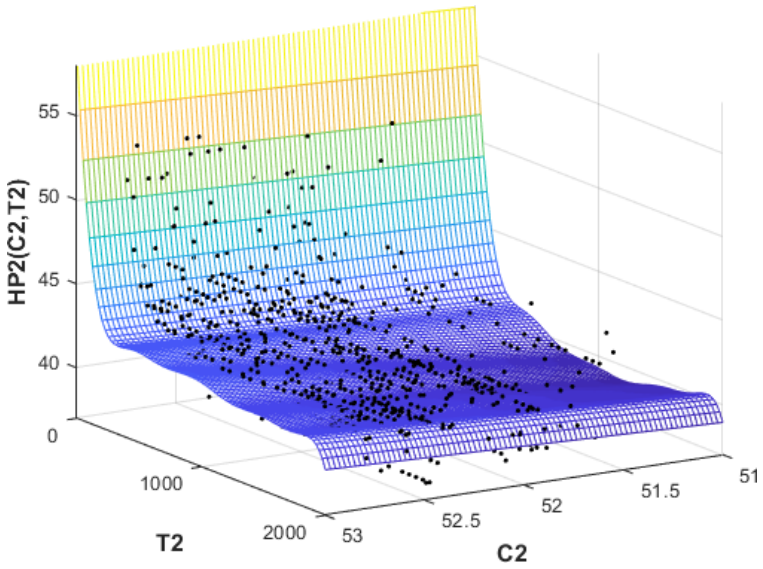


Figure 3.2: The predictive posterior is shown as the colored surface and $\mathcal{D}_{train}$ is shown as the black dots. Three different variables are used in the visualization: *HP2* from the output space and *C2* and *T2* from the input space.

3.1.1 Covariance functions

The covariance function (3.2) measures similarities between two values of a function evaluated at the input pair x and x'. There is a wide range of covariance functions to choose from depending on the characteristics of the underlying function. The two main categories of covariance functions are stationary and non-stationary covariance functions. A stationary covariance function depends only on the relative position of the input pair, while a non-stationary covariance function depends on the absolute position of the input pair. A short summary of commonly used covariance functions follows.

Exponential results a non-differentiable process, which makes it ill suited for smooth underlying functions. It is stationary and has two parameters, the length scale ℓ and the output variance σ^2.

$$k_{\text{Exp}}(x, x') = \sigma^2 \exp\left(-\frac{(x - x')}{\ell}\right) \tag{3.7}$$

ℓ determines how rapidly the function varies. A small ℓ means that function values change quickly and large ℓ means that function values change slowly. σ^2 is the average distance and variation of the function values from their mean, this parameter is shared for all covariance functions. A small σ means that function values stay close to the mean, larger σ means that function values are allowed more variation from the mean. The exponential covariance function is not used for the models in this thesis, but included for completeness.

Squared Exponential, commonly referred to as the the radial basis function, is universal and can be used for most underlying functions. Squared exponential is used as the default covariance function in many Gaussian process applications. It is stationary and has two parameters, the length scale ℓ and the output variance σ^2.

$$k_{\text{SE}}(x, x') = \sigma^2 \exp\left(-\frac{(x - x')^2}{2\ell^2}\right) \tag{3.8}$$

ℓ is the length scale, same as for exponential. σ^2 is the average distance and variation to the function mean, which is the same as for all covariance functions. Squared exponential gives a (infinitely) differentiable process, in contrast to the exponential covariance function. Hence, it is good for smooth underlying functions.

Rational Quadratic is equivalent to adding multiple Squared exponential with dif-

ferent length scales. This allows the function's structure to vary. It is stationary and has three parameters, ℓ, σ and η.

$$k_{\text{RQ}}(x, x') = \sigma^2 \left(1 + \frac{(x - x')^2}{2\eta\ell^2}\right)^{-\eta} \tag{3.9}$$

ℓ is the length scale, σ^2 is the average distance and variation to the function mean and η is the relative weighting of variations, determining how smooth the functions is. The rational quadratic is equal to squared exponential when $\eta \to \infty$.

Matérn is stationary and has three parameters. The average distance and variation to the function mean σ^2, the length scale parameter ℓ and the smoothness parameter ν. Γ is a gamma function and K_ν is a modified Bessel function of the second kind.

$$k_{\text{Matérn}}(x, x') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}(x - x')}{\ell}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}(x - x')}{\ell}\right) \tag{3.10}$$

Matérn is equal to squared exponential when $\nu \to \infty$. For computational efficiency ν is commonly chosen as $\frac{1}{2}, \frac{3}{2}$ or $\frac{5}{2}$. For half integers the Matérn covariance function simplifies to a polynomial (of degree $\nu = \frac{1}{2}$) multiplied by an exponential function avoiding the Bessel function. Smaller ν is used to decrease the smoothness and bigger ν is used to increase the smoothness. When $\nu = \frac{1}{2}$ the Matérn is equal to the exponential covariance function. Matérn is once differentiable when $\nu = \frac{3}{2}$ and is twice differentiable when $\nu = \frac{5}{2}$. In this thesis only $\nu = \frac{3}{2}$ is considered.

Linear is the only non-stationary covariance function tested. It has two parameters, c and σ. c is the offset, which determines the input coordinate x where all the lines of the posterior go though. σ^2 is the average distance and variation to the function mean, same as for all the previous covariance functions.

$$k_{\text{Linear}}(x, x') = \sigma^2(x - c)(x' - c) \tag{3.11}$$

A linear covariance function is equivalent to including a simple linear regression in the mean structure

Combining covariance functions

Covariance functions can be combined to sums or products of multiple covariance functions. This is useful when the data exhibits different features, for example, both a stationary and a non-stationary behaviour. The additive structure and product structure of two covariance functions $k_{\mathrm{a}}(x, x')$ and $k_{\mathrm{b}}(x, x')$ is simply given as:

$$k_{\mathrm{Additive}}(x, x') = k_{\mathrm{a}}(x, x') + k_{\mathrm{b}}(x, x') \tag{3.12}$$

$$k_{\mathrm{Product}}(x, x') = k_{\mathrm{a}}(x, x') \times k_{\mathrm{b}}(x, x') \tag{3.13}$$

The additive covariance function can be thought of as an OR operation, it will have a high value if any of the covariance functions has a high value. The product covariance function on the other hand can be thought of as an AND operation, it will only have a high value if all the covariance functions have a high value.

Anisotropic covariance functions

The stationary covariance functions previously discussed are isotropic, which means that they are equal for all directions. Therefore, the length scales of all the variables in the Gaussian process are equal using an isotropic covariance function.

When there is a large number of input variables they can vary at different scales, some might even impact the model negatively. Thus, the the scaling and choice of input variables becomes very important to ensure a good Gaussian process model.

One way to account for different scaling and handle the potential negative impact from one of the input variables in high dimensional problems is automatic relevance determination (Neal, 1996). Automatic relevance determination gives each input variable a separate length scale, making the covariance functions anisotropic. The scaling is used to reduce the relevance of input variables that affect the model negatively. As can be seen in equations 3.8, 3.9 and 3.10 the inverse of the length scale determines the influence of an input on the predictions. If a length scale has a high value for one of the inputs, the resulting covariance will be close to zero between the two points with similar input values. The zero covariance implies that the input variable will not affect predictions, making that input irrelevant.

3.1.2 Training the Gaussian process model

Every covariance function has a set of parameters, which means that the prior distribution (3.4) itself has parameters, these parameters are called hyperparameters. By using knowledge gained from the training data they can be adjusted in order to better fit the Gaussian process model.

In order to find the best values for the hyperparameters, denoted θ, iterative learning is used. The output of the Gaussian process model is computed over a number N of training iterations i. In each training iteration the log marginal likelihood, denoted $\mathcal{L}(\theta)$, is used to decide the next iteration's hyperparameter values. $\mathcal{L}(\theta)$ measures how well θ fits the model by marginalization over the observed noisy output values, $\mathbf{y}$. $\mathcal{L}(\theta)$ is given as:

$$\mathcal{L}(\theta) = -\log p(\mathbf{y}|\mathbf{X}, \theta) = -\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^\top \mathbf{V}^{-1}(\mathbf{y} - \boldsymbol{\mu}) - \frac{1}{2}\log|\mathbf{V}| - \frac{N}{2}\log 2\pi \quad (3.14)$$

where $\mathbf{V}$ is given in equation 3.6 and $\boldsymbol{\mu}$ is given in equation 3.2. The first term measures the fit of the data, the second term penalizes complexity of the model and the third term is a normalizing constant. The log determinant in the second term is computed using, Cholesky factorisation (Rasmussen and C. K. I. Williams, 2006b). Instead of inverting the matrix to compute the quadratic expression the first term the Cholesky decomposition is used. This is numerically more stable and faster. The time complexity for training the Gaussian process model of n data points is $\mathcal{O}(n^3)$ and the time complexity of computing the predictive poster in equation 3.6 is $\mathcal{O}(n)$, since $\mathbf{V}^{-1}$ can be precomputed from the training data. Thus, the computational expenses to perform Gaussian process inference increase rapidly with n.

One group of iterative optimization methods generally used for tuning the hyperparameters of the Gaussian process model is stochastic gradient descent optimization

algorithms, the basic outline of which is shown in algorithm 1.

Algorithm 1: Stochastic gradient based optimization

Choose a number of training iterations N
Choose a learning rate α
for $i \leftarrow 1$ **to** N **do**
  Compute the gradient of the log marginal likelihood $\nabla_\theta \mathcal{L}(\theta)$
  Update the hyperparameters θ w.r.t $\nabla_\theta \mathcal{L}(\theta)$
    (Update rule for ADAM is shown in equation 3.17)
  Increment i until $\nabla_\theta \mathcal{L}(\theta)$ is close to 0
end
return the solution θ_{opt}

The hyperparameters are updated in the direction of the gradient $\nabla_\theta \mathcal{L}(\theta)$, until the gradient is close to 0. The learning rate α determines each iteration's i step size in order to reach the minimum of $\mathcal{L}(\theta)$.

Stochastic gradient descent can be viewed as the stochastic approximation of standard gradient descent optimization. Instead of using the entire data set to calculate the gradient, the gradient is estimated separately for every value in $\mathcal{D}_{train}$. This allows redundant updates of θ with similar values to be excluded, which reduces the computational burden.

An overview of the most commonly used gradient descent optimization algorithms is given by Ruder (2016). Ruder concludes that the Adaptive Moment Estimation (ADAM) is a good overall choice for sparse data sets, such as signal measurements, as it converges very fast with easy initialization. However, ADAM tends to over adaption, which means it generalizes worse than the slower but more robust stochastic gradient descent optimizer.

ADAM, proposed by Kingma and Ba (2014), stores exponentially decaying average of past gradients m_i and past squared gradients v_i. The first moment m_i (mean) and the second moment v_i (variance) have a tendency towards zero, therefore the bias-corrected moments are computed:

$$\hat{m}_i = \frac{\beta_1 m_{i-1} + (1-\beta_1)\nabla \mathcal{L}(\theta)}{1-\beta_1^i} \tag{3.15}$$

$$\hat{v}_i = \frac{\beta_2 v_{i-1} + (1-\beta_2)\nabla^2 \mathcal{L}(\theta)}{1-\beta_2^i} \tag{3.16}$$

where i is the current training iteration, $i-1$ is the previous training iteration, β_1,

$\beta_2 \in [0, 1)$ are decay rates for the moment estimates and β_1^i, β_2^i are the decay rates to the power of i. Using the bias-corrected moments yields ADAM's update rule for the next iteration's, $i + 1$, hyperparameter values.

$$\theta_{i+1} = \theta_i - \frac{\alpha}{\sqrt{\hat{v}_i} + \epsilon}\hat{m}_i \tag{3.17}$$

In this thesis the values $\alpha = 0.001$, $\epsilon = 10^{-8}$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$ suggested by the authors of ADAM, P. Kingma and Ba are used.

3.1.3 Gaussian process selection

To evaluate the predictive performance of the Gaussian process models using different covariance functions, the error measurement root mean squared error (RMSE) is used:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{\text{y}}_i - \text{y}_i)^2} \tag{3.18}$$

where y_i is the validation set output variable and $\hat{\text{y}}_i$ is the predicted output, from the Gaussian process model. RMSE can be compared with another popular error measurement, mean absolute error. Mean absolute error scales each error linearly, a single error of 10 only contributes twice as much to the total error as single error of 5. RMSE in contrast scales quadratically, which means each single outlier error weights more for the total error than for MAE.

By using the posterior mean and variance (equation 3.6) of every predictive observation, $\hat{y}_i$, 95 % confidence intervals can be computed as:

$$\mu_{\mathbf{f}^*|\mathcal{D},\mathbf{X}^*} \pm 1,96 \cdot \sqrt{K_{\mathbf{f}^*|\mathcal{D},\mathbf{X}^*}} \tag{3.19}$$

As a measure of model performance, the portion of validation data $\mathcal{D}_{val}$ within the confidence is also computed for the models. If 95 % of the validation data is inside the 95 % confidence interval, it indicates a good model. If the validation data inside the 95 % confidence interval is above or below 95 %, either the variance is wrongly estimated or the assumed distribution is incorrect.

The signal to noise ratio (SNR) is calculated by utilizing the signal variance, σ_K^2 (section 3.1.1), relative to the noise variance σ^2 (equation 3.3):

$$\text{SNR} = \frac{\sigma_K^2}{\sigma^2} \tag{3.20}$$

If the SNR is high, the Gaussian process model will fit more variations, rather than explaining them as noise. If the SNR is low the prediction will be flat with constant variances.

3.2 Bayesian optimization

Bayesian optimization is a global optimization technique that is effective when the objective function, $g(c) = \sum_{D_{val}} f^*(c, IN^*_{C'}, T^*_{C'})$, is not explicitly known or expensive to evaluate (Frazier, 2018). A second Gaussian process model is used as a surrogate model for $g(c)$ in Bayesian optimization. To evaluate the Gaussian process model, an acquisition function is used to decide where on the posterior distribution to draw samples. This utility function draws samples sequentially for a number of iterations, N, in order to find the global optimum, while considering the trade-off between exploration and exploitation. Exploration means that samples are drawn where the uncertainty of the prediction is high and exploitation means that samples are drawn where the acquisition function predicts a gain in the objective function. The basic outline of Bayesian optimization is shown in algorithm 2:

Algorithm 2: Bayesian optimization

```
Place a Gaussian prior on g(c)
n initial observations are observed on g(c)
for i ← 1 to N do
    Compute the posterior distribution of g(c) using all samples
    Choose the next query point c_i based on the acquisition function
    Obtain a value of the objective function g(c_i)
    Increment i
end
return optimal candidate c_opt = argmaxc g(c)
```

As the optimization of the acquisition function is non-convex, and since the training of the Gaussian process model is not entirely deterministic, there might be relatively small variations found in c_{opt}. To ensure that the global optimum is found, algorithm 2 is restarted 10 times in this thesis. If the variation in c_{opt} is smaller than 0.01 it

is considered a valid candidate for the global optimum.

The objective function used to find c_{opt} is the total amount of cleaning medium in the output of the data set $\mathcal{D}_{val}$. Using the Gaussian process models built with $\mathcal{D}_{train}$, LP and HP are simulated for 1000 values of C, denoted $C_{[0,1]}$, on an evenly spaced interval [0,1]. The actual value of C in $\mathcal{D}_{val}$ is replaced with $C_{[0,1]}$ while keeping the rest of the input variables in the simulations unchanged. Since $\mathcal{D}$ is normalized with Min-Max feature scaling, the smallest value of C is zero and the biggest is one.

To compute the total amount of cleaning medium during the simulation, the composite trapezoidal rule is used, which approximates the integral of discrete values. The integral is equal to the total amount of cleaning medium in the predictions, $\hat{LP}$ and $\hat{HP}$, during the entire simulation. For each value in $C_{[0,1]}$, there is a corresponding value of the total amount of cleaning medium during the simulation. Thus, the objective function as a function of $C_{[0,1]}$ and the unchanged input variables, is created.

$g(c)$ is used as the objective function to find the optimal regulating valve position with Bayesian optimization, denoted C_{BO}. Thus, a second Gaussian process model is built to surrogate $g(c)$.

3.2.1 Acquisition functions

Acquisition functions have different properties where the effectiveness depends on the compatibility with the objective function $f(x)$. The Gaussian process used to surrogate $f(x)$ has normally distributed posterior mean μ_f and variance σ_f^2. A short summary of the acquisition functions, upper confidence bound and expected improvement, used to find the optimum of $f(x)$ follows:

Upper confidence bound (UCB) combines the posterior mean and the posterior variance. The exploitation-exploration trade-off factor β is the only input parameter (Auer, 2002). A high β favors exploration, while a low β favors exploitation. In this thesis β is chosen as 0.2, which is recommended in (Balandat et al., 2019).

$$\text{UCB}(x, \beta) = \mu_f(x) + \sqrt{\beta}\sigma_f^2(x) \tag{3.21}$$

Expected improvement (EI) considers how much it is possible to improve the ob-

jective. The evaluation of $f(\mathrm{x})$ is based on the previously best observed value f^*, which is usually chosen as the highest value of the data used to train the model. Expected improvement is given as:

$$\mathrm{EI(x)} = \max(\Delta(x), 0) + \sigma_f(x)\phi\left(\frac{\Delta(x)}{\sigma_f(x)}\right) - |\Delta(x)|\Phi\left(\frac{\Delta(x)}{\sigma_f(x)}\right) \tag{3.22}$$

where $\Delta(x) = \mu_f(x) - f^*$, ϕ is the cumulative distribution function and Φ is the probability distribution function. Expected improvement favors exploitation when $\max(\Delta(x), 0)$ is high and exploration when σ_f is high. Noise free observations are assumed in expected improvement (Mockus, Tiesis, and Zilinskas, 2014).

3.3 Multiple objectives & Evaluation

During the separation process there are two objectives, to minimize the cleaning medium in *LP* and to maximize the cleaning medium in *HP*. Both objectives are optimized by finding the optimal *C*. As the input parameter *C* regulates both objectives there might be cases where improving one objective worsens the other.

All the values of *C*, where one of the objectives cannot be improved without worsening the other, are called Pareto optimal values. All the values of *C* where both objectives can be improved simultaneously are called Pareto dominated values. The set of all Pareto optimal values are called the Pareto frontier. The Pareto frontier does not tell what the optimal value of *C* is, but makes the trade-off between the two objectives clear.

First, a regression model was built using the Gaussian process and $\mathcal{D}_{train}$. With the trained model the objective function, cleaning medium in output during the entire simulation of $\mathcal{D}_{val}$, was created. This objective function was surrogated with a new Gaussian process model, in order to find the optimal candidate C_{BO} with Bayesian optimization. After that C_{BO} is evaluated, which is done by additionally simulating 1000 values on an evenly spaced interval [0,1] using $\mathcal{D}_{test}$, while keeping the rest of the input variables unchanged. The amount of cleaning medium is computed with the trapezoidal rule and used to evaluate the impact of *C*, and to find the Pareto frontier of *LP* and *HP*.

The predictions, $\hat{LP}$ and $\hat{HP}$, with the original *C* are compared with the simulations

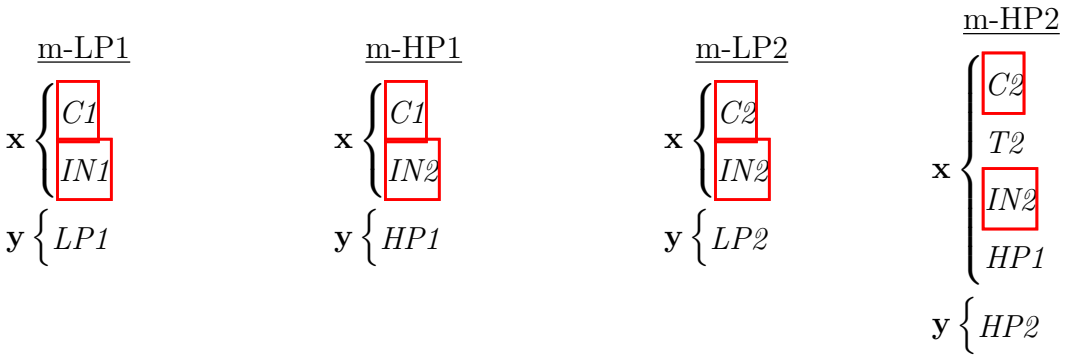
using C_{BO} and $C_{[0,1]}$. By comparing the total amount of cleaning medium during the simulations in $\mathcal{D}_{test}$, the following questions can be answered in section 4.4

1. Whether $\hat{LP}(C_{\text{BO}})$ and $\hat{HP}(C_{\text{BO}})$ provide better results than $\hat{LP}$ and $\hat{HP}$.
2. Whether $\hat{LP}(C_{\text{BO}})$ and $\hat{HP}(C_{\text{BO}})$ manage to provide the best possible results, by comparing them to $\hat{LP}(C_{[0,1]})$ and $\hat{HP}(C_{[0,1]})$.
3. What the trade-off is between minimizing the amount of cleaning medium in *LP* and maximizing the amount of cleaning medium in *HP*, by analyzing the Pareto frontier.

Results

The Gaussian process predictive performance is presented in section 4.1. Using the best covariance functions, the optimal values for *C1* and *C2*, found with Bayesian optimization are given in section 4.2. The optimal values are analyzed through simulation in section 4.3 and the optimal values and conclusions are finally given in section 4.4. Since $\mathcal{D}$ is normalized all variables shown in the figures are in the interval [0,1].

For the variables investigated in this thesis, Anisotropic covariance functions (section 3.1.1) neither manages to remove the impact of irrelevant variables nor provide better models. Several different models were tested, but only the 4 best models are presented. For each model only the best combination of input variables is shown.



4.1 Gaussian process predictive performance

The 4 models were tried with all the covariance functions and all possible additive and product combinations. The models were trained with $\mathcal{D}_{train}$ and validated with $\mathcal{D}_{val}$. The RMSE values are presented in table 4.1.

Using solely the linear covariance function results in a clearly higher RMSE than for all other combinations of covariance functions. The predictions using the best covariance function for each respective model are shown in figures 4.1 to 4.4.

Table 4.1: The RMSE using different covariance functions (section 3.1.1). The lowest RMSE for each model is marked in bold.

RMSE				
Covariance function	m-LP1	m-HP1	m-LP2	m-HP2
Rational Quadratic	0.0309	0.1122	0.0605	0.1021
Squared Exponential	0.0462	0.1225	**0.0560**	0.0898
Linear	0.1379	0.4223	0.1216	0.2401
Matérn	0.0422	0.1073	0.0591	0.1232
Rational Quadratic + Squared Exponential	0.0447	0.1124	0.0615	0.1001
Rational Quadratic + Linear	0.0312	0.1122	0.0615	0.1009
Rational Quadratic + Matérn	0.0426	0.1069	0.0601	0.1120
Linear + Squared Exponential	0.0469	0.1212	0.0577	0.0884
Linear + Matérn	0.0398	0.1071	0.0603	0.1175
Squared Exponential + Matérn	0.0374	0.1074	0.0560	0.1139
Rational Quadratic · Squared Exponential	0.0523	0.1102	0.0599	0.1347
Rational Quadratic · Linear	0.0326	0.1196	0.0712	0.0904
Rational Quadratic · Matérn	0.0501	**0.1059**	0.0584	0.1421
Linear · Squared Exponential	0.0505	0.1230	0.0669	**0.0769**
Linear · Matérn	**0.0279**	0.1140	0.0694	0.1048
Squared Exponential · Matérn	0.0452	0.1064	0.0582	0.1525

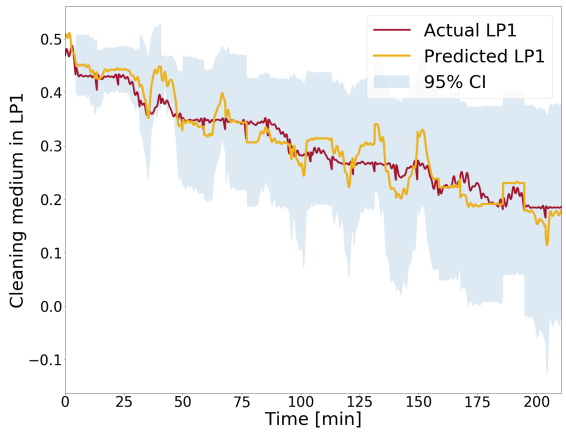


Figure 4.1: The Prediction of *LP1* is yellow using the input variables *C1* and *IN1* in model m-LP1. The covariance function used is the product of linear and Matérn. The actual *LP1* of $\mathcal{D}_{val}$ is red and the predictive 95% confidence interval is gray.

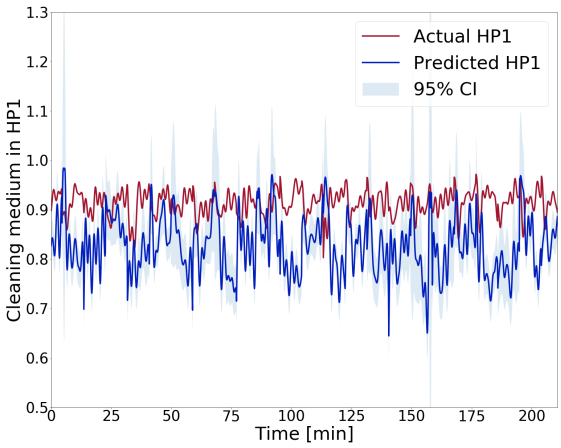


Figure 4.2: The prediction of *HP1* is blue using the input variables *C1* and *IN2* in model m-HP1. The covariance function used is the product of rational quadratic and Matérn. The actual *HP1* of $\mathcal{D}_{val}$ is red and the predictive 95% confidence interval is gray.

In figure 4.1 the prediction of *LP1* follows the linear downward trend while still capturing small variations. The validation data is completely within the confidence interval and the SNR is 2.6. In figure 4.2 the prediction of *HP2* is generally noisy and only 33.8 % of the validation data is inside the confidence interval. Considering that the mean of *HP1* is 0.64 in $\mathcal{D}_{train}$, the prediction with a mean of 0.82 still manages to somewhat capture the actual mean of *HP1* in $\mathcal{D}_{val}$, which is 0.91. The SNR is 3.2.

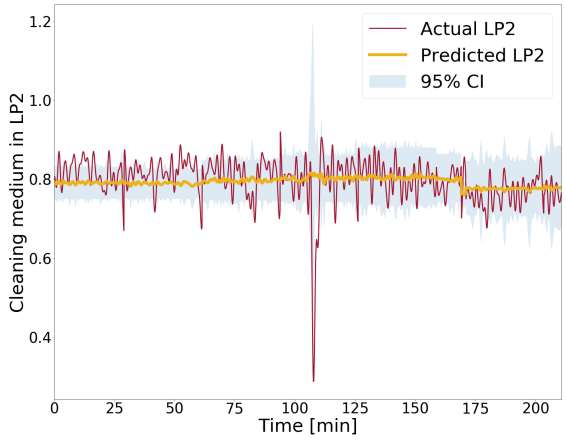


Figure 4.3: The prediction of *LP2* is yellow using the input variables *C2* and *IN2* in model m-LP2. The covariance function used is the squared exponential. The actual *LP2* of $\mathcal{D}_{val}$ is red and the predictive 95% confidence interval is gray.

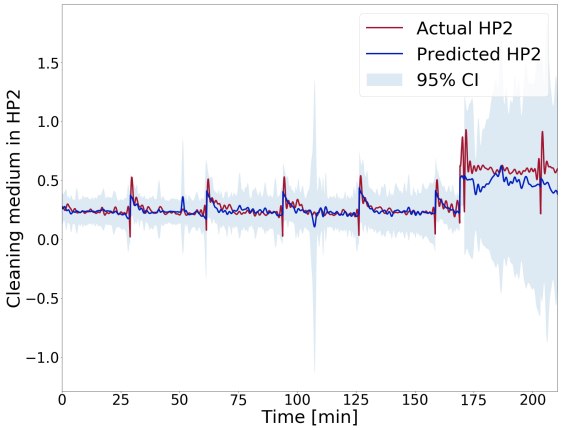


Figure 4.4: The prediction of *HP2* is blue using the input variables *C2*, *T2*, *HP1* and *IN2* in model m-HP2. The covariance function used is the product of linear and squared exponential. The actual *HP2* of $\mathcal{D}_{val}$ is red and the predictive 95% confidence interval is gray.

In figure 4.3 the mean of *LP2* is captured well and captures the downward shift in *LP2* at 165 minutes. The prediction of *LP2* does not manage to capture the downward large spike at minute 107. The confidence interval contains 82 % of the validation data and the SNR is 0.11. In figure 4.4 the oscillating pattern of *HP2* with strong transients is captured quite well until minute 165, where the predictions are unable to follow the upwards shift. In general the upward part of the transient is captured better than the downward spike. 99.4 % of the validation data confidence interval and the SNR is 0.43.

The Gaussian process models, i.e. combination of input variables and covariance functions, which provide the lowest RMSE for each respective model are used for finding the optimal values of *C1* and *C2* in the next section.

4.2 Application of Bayesian optimization

The optimal regulating valve position for *LP* is the position that minimizes the amount of cleaning medium, and for *HP* it is the position that maximizes the amount of cleaning medium. For the separators investigated in this thesis the regulating valve position has a predetermined fixed value deemed optimal. This means that the regulating valve position is not varied during the separation process. *C* was not optimized in relation to the other input variables for the Gaussian process model, as a single fixed value of C could be seen optimizing the output variables. For example, for every value of $IN1$ there might have been a specific $C1$ that optimized $LP1$.

C_{BO} is determined with both the acquisition functions upper confidence bound and expected improvement. A Gaussian process model with the squared exponential covariance function is used as a surrogate for the objective function, the total amount of cleaning medium during simulations of $\mathcal{D}_{val}$. Four different objective functions are created, one for each model. The optimal regulating valve positions acquired with Bayesian optimization, denoted $C1_{\text{BO}}$ and $C2_{\text{BO}}$, are presented in table 4.2.

Table 4.2: $C1_{\text{BO}}$ and $C2_{\text{BO}}$ for the different models acquired with the acquisition functions upper confidence bound and expected improvement (section 3.2.1).

	m-LP1	m-HP1	m-LP2	m-HP2
	$C1_{\text{BO}\|\text{LP}}$	$C1_{\text{BO}\|\text{HP}}$	$C2_{\text{BO}\|\text{LP}}$	$C2_{\text{BO}\|\text{HP}}$
Upper Confidence Bound	0.45	0.41	1.0	1.0
Expected Improvement	$*$	0.41	$*$	1.0

$*$ There are very large variations in the solutions using the acquisition function.

$C2_{\text{BO}|\text{LP}}$ and $C2_{\text{BO}|\text{HP}}$ are denoted $C2_{\text{BO}}$ as they share the same optimal value 1.0, except for the inconclusive result of $C2_{\text{BO}|\text{LP}}$ using expected improvement. The solutions using expected improvement varies randomly over the entire interval of possible solutions [0,1]. Thus, the solutions using expected improvement are inconclusive when minimizing the objective function, i.e. for both *LP1* and *LP2*.

4.3 Simulation & Evaluation

To investigate how the regulating valve position affects *LP* and *HP*, 1000 different values denoted $C1_{[0,1]}$ and $C2_{[0,1]}$ on an evenly spaced interval [0,1] are used in simulations. The endpoints of the interval [0,1] are the minimal and maximal values of C in the complete data set $\mathcal{D}$.

For *LP* a lower amount of cleaning medium is better, as it implies more product. Thus, if $\hat{LP}(C_{\text{BO}})$ (green) is below $\hat{LP}$ it means that C_{BO} manages to improve the results. If $\hat{LP}(C_{\text{BO}})$ (green) is at the bottom of $\hat{LP}(C_{[0,1]})$ (gray), it means that given the interval [0,1], C_{BO} minimizes the possible amount of cleaning medium in *LP*. For *HP* a higher amount of cleaning medium is better. Thus, if $\hat{HP}(C_{\text{BO}})$ (green) is above $\hat{HP}$ it means that C_{BO} manages to improve the results. If $\hat{HP}(C_{\text{BO}})$ (green) is at the top of $\hat{HP}(C_{[0,1]})$ (gray), it means that given the interval [0,1], C_{BO} maximizes the possible amount of cleaning medium in *LP*.

In figures 4.5 to 4.8 the predictions using the unchanged input variables $\hat{LP}$ and $\hat{HP}$

are shown. They are compared with the predictions using the 1000 different regulating valve positions on the interval $\hat{LP}(C_{[0,1]})$ and $\hat{HP}(C_{[0,1]})$ and the predictions using $C1_{\text{BO|LP}}$, $C1_{\text{BO|HP}}$ and $C2_{\text{BO}}$.

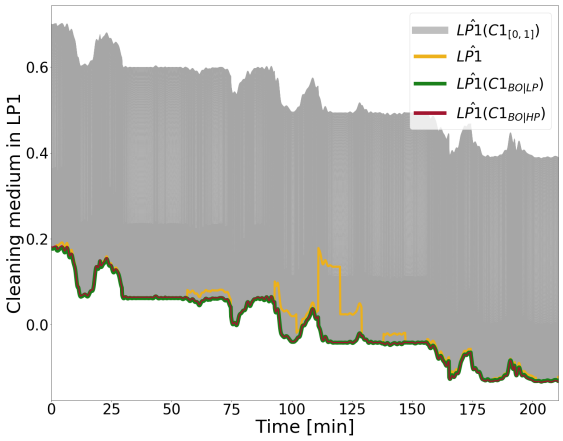


Figure 4.5: $\hat{LP1}$ in yellow is the prediction using the original input variables. $\hat{LP1}(C1_{[0,1]})$ in gray are the simulations using the evenly spaced interval. $\hat{LP1}(C1_{\text{BO|LP}})$ in green is the simulated $\hat{LP1}$ using $C1_{\text{BO|LP}}$. $\hat{LP1}(C1_{\text{BO|HP}})$ in red is the simulated $\hat{LP1}$ using $C1_{\text{BO|HP}}$.

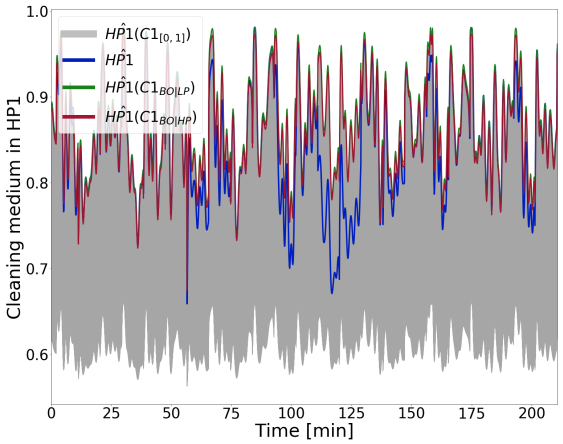


Figure 4.6: $\hat{HP1}$ in blue is the prediction using the original input variables. $\hat{HP1}(C1_{[0,1]})$ in gray are the simulations using the evenly spaced interval. $\hat{LP1}(C1_{\text{BO|LP}})$ in green is the simulated $\hat{LP1}$ using $C1_{\text{BO|LP}}$. $\hat{LP1}(C1_{\text{BO|HP}})$ in red is the simulated $\hat{LP1}$ using $C1_{\text{BO|HP}}$.

In figure 4.5 it can be seen that both $C1_{\text{BO|LP}}$ and $C1_{\text{BO|HP}}$ are close to minimizing the $\hat{LP1}$ in the entire simulation. In figure 4.6 it can be seen that $C1_{\text{BO|HP}}$ does not maximize $\hat{HP1}$ during the simulation. However, it can be seen that $C1_{\text{BO|LP}}$ almost maximizes $\hat{HP1}$ during the simulation. In figures 4.7 and 4.8 it can be seen that $C2_{\text{BO}}$ is close to minimizing $\hat{LP2}$ and close to maximizing $\hat{HP2}$.

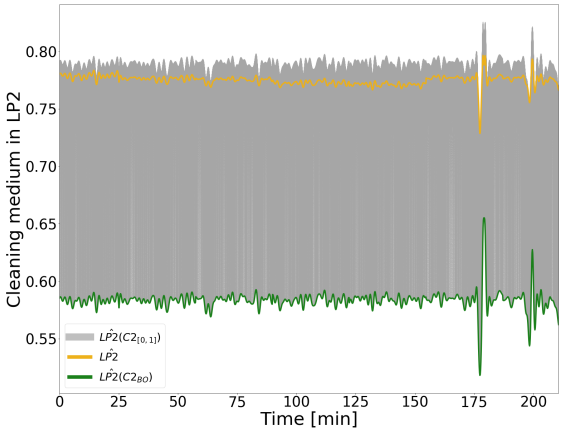


Figure 4.7: $\hat{LP2}$ in yellow is the prediction using the original input variables. $\hat{LP2}(C2_{[0,1]})$ in gray are the simulations using the evenly spaced grid. $\hat{LP2}(C2_{\text{BO}})$ in green is the simulated $\hat{LP2}$ using $C2_{\text{BO}}$.

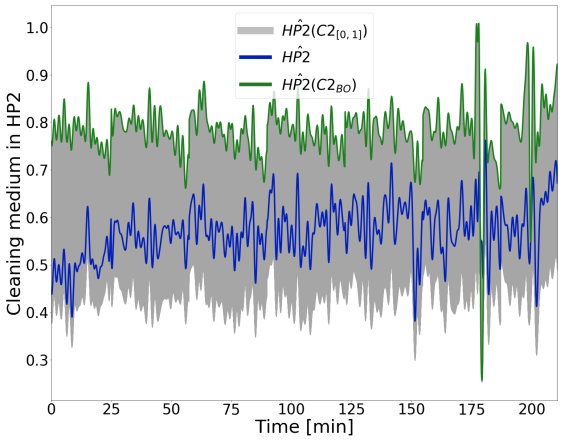


Figure 4.8: $\hat{HP2}$ in blue is the prediction using the original input variables. $\hat{HP2}(C2_{[0,1]})$ in gray are the simulations using using the evenly spaced grid. $\hat{HP2}(C2_{\text{BO}})$ in green is the simulated $\hat{HP2}$ using $C2_{\text{BO}}$.

Using the composite trapezoidal rule, the total amount of cleaning medium in the simulation of *LP* and *HP* is calculated. The total amount of cleaning medium in the predictions using $C_{[0,1]}$ is shown in figures 4.9 and 4.11. Further, the Pareto frontier of total amount of cleaning medium in the predictions using $C_{[0,1]}$ is shown in figures 4.10 and 4.12. In the figures with the Pareto frontier the "Product in *LP*" is computed as: 1 - cleaning medium, since it is the amount of cleaning medium that is measured. For the Pareto frontier the amount of cleaning medium in *HP* increases higher up in the figure, and the amount of product increases further right in the figure. This means that the Pareto optimal points will be in the top right corner of the figure.

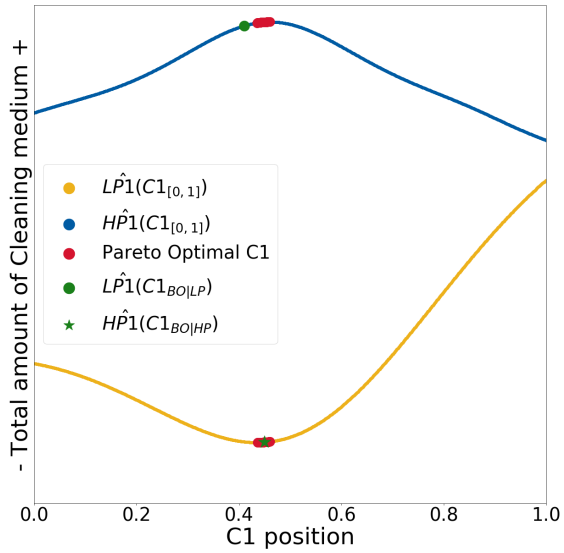


Figure 4.9: The total amount of cleaning medium in $\hat{LP}1$ and $\hat{HP}1$ for $C1_{[0,1]}$.

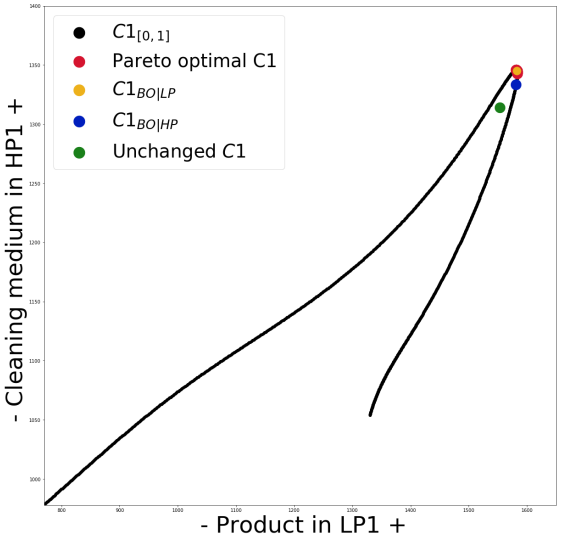


Figure 4.10: The total amount of product in $\hat{LP}1$ and the total amount of cleaning medium in $\hat{HP}1$. The Pareto optimal points are found in the top right corner of the Pareto frontier.

In figure 4.9 it can be seen that the Pareto optimal points are close to maximizing $\hat{HP}1$ while simultaneously minimizing $\hat{LP}1$. This means that there is a very small trade-off between the two objectives of minimizing *LP1* and maximizing *HP1*. In figure 4.10 it can seen that using the unchanged C1 is a Pareto dominated value. $C1_{\text{BO|LP}}$ is close to minimizing $\hat{LP}1$. $C1_{\text{BO|HP}}$ does not maximize $\hat{HP}1$, but still manages to slightly increase the amount of cleaning medium in $\hat{HP}1$ by 1.48 % compared to using the unchanged input variables. The maximum increase of cleaning medium achievable in $\hat{HP}1$ is 2.4 %.

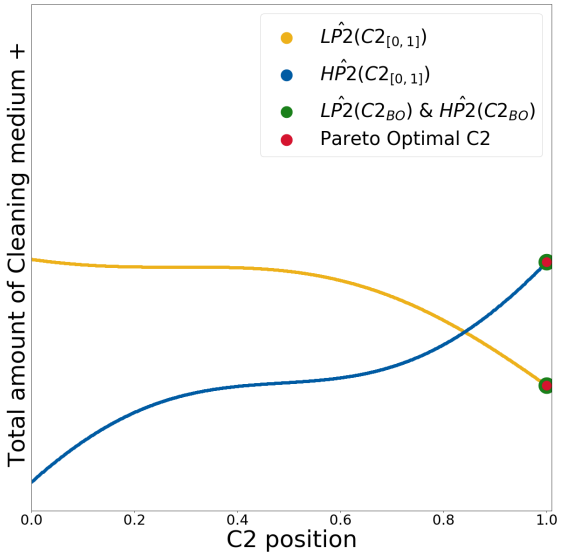


Figure 4.11: The total amount of cleaning medium in $L\hat{P}2$ and $H\hat{P}2$ for the $C2$ on the grid [0,1].

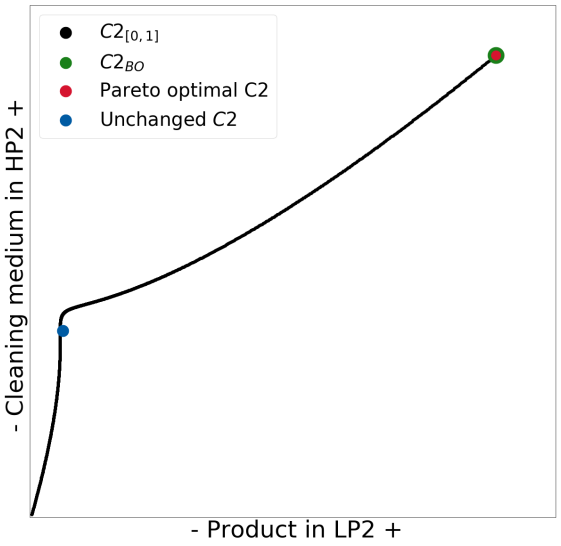


Figure 4.12: The total amount of product in $L\hat{P}2$ and the total amount of cleaning medium in $H\hat{P}2$. The Pareto optimal points are found in the top right corner of the Pareto frontier.

In figure 4.11 it can be seen that the Pareto optimal point maximizes $H\hat{P}2$ while simultaneously minimizing $L\hat{P}2$, which means that there is no trade off between the two objectives of minimizing $LP2$ and maximizing $HP2$. $C2_{\text{BO}}$ finds the global minimum of $L\hat{P}2$ and the global maximum of $H\hat{P}2$. The fact that the global optimum is at $C2 = 1.0$ indicates that that there is a value of $C2$ bigger than 1.0 that would further improve $L\hat{P}2$ and $H\hat{P}2$. In figure 4.12 it can seen that using the unchanged C2 is a Pareto dominated value.

4.4 Optimal regulating valve position

The conclusions to questions asked in section 3.3 can now be drawn:

1. C_{BO} clearly provides better results for $L\hat{P}1$, $L\hat{P}2$ and $H\hat{P}2$. For $H\hat{P}1$, C_{BO} provides only a slight improvement.

2. C_{BO} finds the optimum for $L\hat{P}2$ and $H\hat{P}2$ and is close to the finding the optimum for $L\hat{P}1$. C_{BO} does not find the optimum for $H\hat{P}1$.

3. For $L\hat{P}2$ and $H\hat{P}2$ there is only one Pareto optimal point ($C2 = 1.0$) which also optimizes both outputs simultaneously. For $L\hat{P}1$ and $H\hat{P}1$ the Pareto frontier is in the interval $C1 \in [0.43, 0.46]$. There is a small trade-off between the objectives as the optimums of $L\hat{P}1$ and $H\hat{P}1$ are very close.

The optimal *C1* is chosen in the middle of the interval [0.43, 0.46] (Pareto frontier) as $C1_{Optimal} = 0.445$. The optimal *C2* is chosen as $C2_{Optimal} = 1.0$, since it is the only Pareto optimal point. The two regulating valve positions of the entire data set $\mathcal{D}$, the mean of the regulating valve positions and the optimal regulating valve positions are shown in figures 4.13 and 4.14

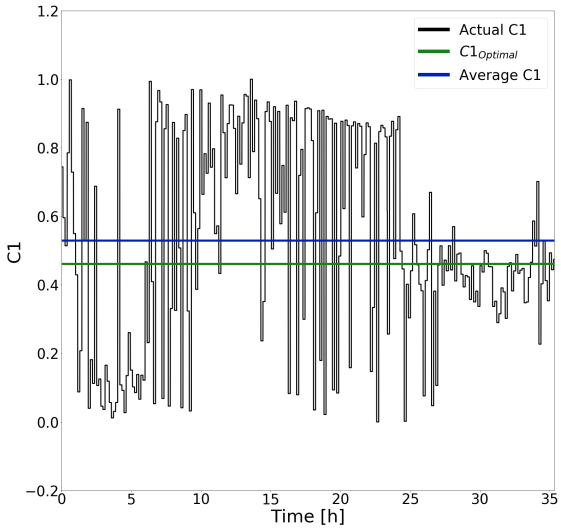


Figure 4.13: *C1* from $\mathcal{D}$ compared with the average and optimal *C1*.

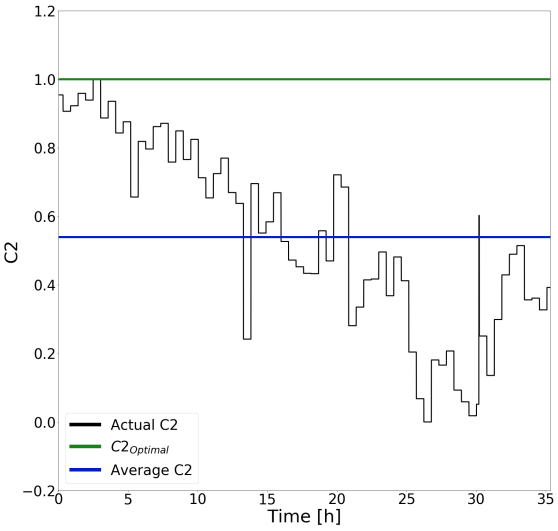


Figure 4.14: *C2* from $\mathcal{D}$ compared with the average and optimal *C2*.

The difference in the appearance of the regulating valve positions *C1* and *C2* stems for the difference in seconds from last discharge *T1* and *T2*. In figure 4.13 it can be seen that $C1_{optimal}$ is very close to the mean. In figure 4.14 it can be seen that $C2_{optimal}$ is at the maximum of the measured values in $\mathcal{D}$, which indicates that *LP2* and *HP2* can be further improved by increasing *C2*.

Discussion

In the discussion the results of this thesis are reviewed and the method limitations and possible improvements are discussed. The data and variables are discussed in section 5.1, the model in section 5.2 and the optimization and simulations in section 5.3. The main points of this thesis are summarized in section 5.4.

5.1 Data review

The quality of the data is the foundation of any statistical model. Without good data it is also impossible to build a good model. The removal of noise from the regulating valve signal, *IN*, *LP* and *HP* was done in order to create better data. The Gaussian process predictive performance improves with less noisy data, even though the predictive posterior (equation 3.6) has a noise component. Further, the structural patterns of the data become easier to see and the differences and similarities of the predictive posterior and validation data become clearer.

The amount of data used when building the models was restricted by the high time complexity $\mathcal{O}(n^3)$ of the Gaussian process model. Using better hardware with GPU-acceleration, as suggested by Gardner et al. (2018), probably would have allowed the models to include much longer time intervals. Sometimes, it is desirable to use longer time periods in the models in order to mimic longer experiments in real life, which can last weeks.

The non-causal filter used to remove the high frequency noise is limited to historical analysis, since it uses information about the future to filter the signal. A causal filter would be needed if the Gaussian process model with Bayesian optimization were to be implemented in a real time system.

There are possibilities to improve the Gaussian process models in this thesis by including more variables in the models. Measuring pressure at different locations such as the inlet, the light phase outlet and the heavy phase outlet could be a good predictive indicator of bigger changes in the separation process. For example, the drastic change in the cleaning medium in *HP2*, seen in figure 4.4 at minute 165, could potentially have been predicted better with pressure measurements. Temperature measurements and flow volumes could fill a similar role of better predicting big

changes in the separation process.

In the results it is concluded that the trade-off between *LP* and *HP* is almost negligible. In future modeling of the separation process a ratio of *LP* and *HP* could be used for simultaneous modeling.

Further testing using different regulating valve positions is of utmost interest. The major limitation in this thesis was the limited variability in the regulating valve position. For *C1* this is of less relevance since the $C1_{optimal}$ is found near the middle of the interval [0,1]. However, since the variability in *C1* is still quite limited there are no guaranties that this is the global optimum. For *C2*, on the other hand, further testing with a bigger variability in *C2* is crucial. Given that the optimal value was found at the boundary of the variable space, and the trend seen in figure 4.11, it is highly likely that the optimal value is larger than 1.

5.2 Model review

As can be seen from the summary of the predictive performance in table 4.1, using only the linear covariance function always gives worse performance than any other covariance function. In figure 2.10 it can be seen *IN1* and *LP1* are correlated. However, for the linear covariance function to work well, all the input variables need to be linearly correlated with the output variable. For the models in this thesis the linear covariance function proves useful in the product structure, where it provides the best models for m-LP1 and m-HP2.

Additive covariance functions neither produce any of the best results for the 4 models, nor manages to outperform the use of single covariance functions. The product structure provides some of the best results and gives the best results for 3 of the cases. This is logical considering the difference in that the additive structure works like an OR operation while the product structure works like an AND operation.

Restricting the choice covariance functions to one metric, RMSE, has limitations. For example looking at m-HP2, one covariance function might be very good at predicting the mean of *HP2*, but worse at predicting the strong transients. In this case a covariance function which catches the transients better but the mean worse, might be a better fit for m-HP2. Preferably the plots of all the predictions of the models using all covariance functions should have been investigated.

The variations of the predictions of *LP1* and *HP1* are bigger than variations in the

validation data. This occurs since the signal variance is much bigger than the noise variance in the Gaussian process model (SNR = 2.6 & 3.2), which causes the model to fit the variations of the input data rather than explain them as noise. When the SNR is low (0.11), as in the prediction of *LP2*, the Gaussian process model explains the variations as noise. The amount of validation data inside the 95 % confidence intervals should be 95 %. None of the models manage to come close to this percentage, which likely occurs since the variance estimations of the predictions are incorrect.

For the separator processes analyzed in this thesis anisotropic covariance functions did not manage to reduce the negative impact from irrelevant input variables. This makes the Gaussian process impractical to implement since very noisy data, data with inconsistencies and uncorrelated input variables have to be avoided. It is very time consuming for the practitioner to find the best combination, due to the high number of possible choices, considering the presence of 4 models and 10 variables, as well as the usage of different data sets.

If the management of the irrelevant input variables was improved, a multi-task Gaussian process, as suggested by Bonilla, Chai, and C. Williams (2008), with multiple output variables could have been incorporated. It is an intuitive way to model the separation process, multiple input variables affect multiple output variables. Using this multi-task Gaussian process speeds up testing, but is limited due to the possibility of only choosing one covariance function for all output variables.

5.3 Optimization & Simulation review

The quality of the Bayesian optimization depends on the quality of the Gaussian process regression model. If the model is not good, the result of the Bayesian optimizing is inconclusive. This can be seen in m-HP1, which is arguably a worse model, where $C1_{\mathrm{BO|HP}}$ does not maximize the cleaning medium in the simulation as shown in figure 4.6.

The restriction of using the interval [0,1] for C_{BO} was based on two reasons: Firstly, since the regulating valve position has not been measured outside this interval, there is no knowledge how the separation process looks like outside the interval. Secondly, the Gaussian process model learns by recognizing similarities between data points. With no previous data, there are no data points the model can learn from. This can be seen in figure 4.4 where the 95 % confidence interval increases drastically when

$HP2$ changes appearance after minute 165.

During the simulations the amount of cleaning medium drastically changes with varying C for all the models. It is not realistic to expect these kinds of changes in the output variables when conducting real life experiments. The Gaussian process model drastically overestimates the importance of the input variable C in all models, highlighting the need of using better data and complimenting it with real life experiments.

Bayesian optimization is primarily used when the objective function is very expensive to evaluate. The objective function in this thesis, the total amount of cleaning medium in *LP* and *HP*, is very easy to evaluate. However, if the model extends to multiple input variables being parameters in the objective function, it quickly becomes very expensive to evaluate and Bayesian optimization would serve an important role.

Multi-objective Bayesian optimization, as suggested by Biswajit Paria (2019), could be used when multiple output variables need to be optimized simultaneously. This could prove very useful when extending the model to optimizing more than 3 variables, since with more than 3 input variables it becomes impossible to visually draw conclusions. For example, it is possible to add energy consumption and sound level as output variables.

5.4 Conclusion

The results of this thesis showed that models of *LP* and *HP* for both separation steps can be created. Bayesian optimization managed to find the optimums for 3 of the 4 models. The results showed that the two objectives of maximizing the cleaning medium in *HP* and minimizing the cleaning medium in *LP* have close to no trade-off, meaning that there is one perfect regulating valve position for both separator steps. The optimal regulating valve position candidates discovered in this thesis will be tested in real life experiments.

Bibliography

Alfa Laval (2020). *Separators*. URL: https://www.alfalaval.com/products/separation/centrifugal-separators/separators/ (visited on 06/07/2020).

Auer, Peter (Jan. 2002). "Using Confidence Bounds for Exploitation-Exploration Trade-offs." In: *Journal of Machine Learning Research* 3, pp. 397–422.

Balandat, Maximilian, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy (2019). "BoTorch: Programmable Bayesian Optimization in PyTorch". In: arXiv: 1910.06403.

Biswajit Paria Kirthevasan Kandasamy, Barnabás Póczos (2019). "A Flexible Framework for Multi-Objective Bayesian Optimization using Random Scalarizations". In: arXiv: 1805.12168v3.

Bonilla, Edwin V, Kian M. Chai, and Christopher Williams (2008). "Multi-task Gaussian Process Prediction". In: *Advances in Neural Information Processing Systems 20*. Curran Associates, Inc., pp. 153–160.

Frazier P.I., Wang J (2016). "Bayesian Optimization for Materials Design". In: *Information Science for Materials Discovery and Design. Springer Series in Materials Science, vol 225. Springer.*

Frazier, Peter I. (2018). "A Tutorial on Bayesian Optimization". In: arXiv: 1807.02811.

Gardner, Jacob R, Geoff Pleiss, David Bindel, Kilian Q Weinberger, and Andrew Gordon Wilson (2018). "GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration". In: *Advances in Neural Information Processing Systems*. arXiv: 1809.11165.

Gonzalvez, Joan, Edmond Lezmi, Thierry Roncalli, and Jiali Xu (Jan. 2019). "Financial Applications of Gaussian Processes and Bayesian Optimization". In: *SSRN Electronic Journal*.

Herwin, Eric (2019). "Optimizing process parameters to increase the quality of the output in a separator: An application of Deep Kernel Learning in combination with the Basin-hopping optimizer". Linköping University.

Kingma, Diederik and Jimmy Ba (Dec. 2014). "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations*.

MATLAB (2018). The MathWorks, Natick, MA, USA.

Mockus, J., Vytautas Tiesis, and Antanas Zilinskas (Sept. 2014). "The application of Bayesian methods for seeking the extremum". In: *Towards Global Optimization* 2, pp. 117–129.

Neal, Radford M. (1996). *Bayesian Learning for Neural Networks*. Springer-Verlag New York.

Rasmussen, Carl Edward and Christopher K. I. Williams (2006a). *Gaussian Processes for Machine Learning*. The MIT Press.

— (2006b). *Gaussian Processes for Machine Learning*. The MIT Press.

Ruder, Sebastian (2016). "An overview of gradient descent optimization algorithms". In: *CoRR* abs/1609.04747. arXiv: 1609.04747.

Master's Theses in Mathematical Sciences 2020:E54
ISSN 1404-6342

LUTFMS-3394-2020

Mathematical Statistics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/