UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA QUÍMICA

Luís Felipe Safady

Control of a neutralization reactor using extremum-seeking approaches

Controle de um reator de neutralização utilizando o método extremum-seeking

CAMPINAS

2018

LUÍS FELIPE SAFADY

CONTROL OF A NEUTRALIZATION REACTOR USING EXTREMUM-SEEKING APPROACHES

CONTROLE DE UM REATOR DE NEUTRALIZAÇÃO UTILIZANDO O MÉTODO EXTREMUM-SEEKING

Dissertação apresentada à Faculdade de Engenharia Química da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Química.

Dissertation presented to the Faculty of Chemical Engineering of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Chemical Engineering.

Orientadora/Advisor: Profa. Dra. Ana Maria Frattini Fileti

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA PELO ALUNO LUÍS FELIPE SAFADY, E ORIENTADA PELA PROFA. DRA. ANA MARIA FRATTINI FILETI

CAMPINAS

2018

Agência(s) de fomento e nº(s) de processo(s): CNPq, 131712/2016-0

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Luciana Pietrosanto Milla - CRB 8/8129

Sa17c

Safady, Luís Felipe, 1991-
Control of a neutralization reactor using extremum-seeking approaches / Luís Felipe Safady. – Campinas, SP : [s.n.], 2018.

Orientador: Ana Maria Frattini Fileti.
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de Engenharia Química.

1. Controle de processos químicos - Automação. 2. Controle de processos químicos - Simulação por computador. 3. Controle automático. 4. Controladores auto-ajustáveis. I. Fileti, Ana Maria Frattini, 1965-. II. Universidade Estadual de Campinas. Faculdade de Engenharia Química. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Controle de um reator de neutralização utilizando o método extremum-seeking
Palavras-chave em inglês:
Chemical process control - Automation
Chemical processo controle - Computer simulation
Automatic control
Self-adjusting controllers
Área de concentração: Engenharia Química
Titulação: Mestre em Engenharia Química
Banca examinadora:
Ana Maria Frattini Fileti [Orientador]
Flávio Vasconcelos da Silva
Argimiro Resende Secchi
Data de defesa: 16-04-2018
Programa de Pós-Graduação: Engenharia Química

FOLHA DE APROVAÇÃO

Dissertação de Mestrado defendida por Luís Felipe Safady e aprovada em 16 de abril de 2018 pela banca examinadora constituída pelos doutores:

Profa. Dra. Ana Maria Frattini Fileti
FEQ – UNICAMP

Prof. Dr. Flávio Vasconcelos da Silva
FEQ – UNICAMP

Prof. Dr. Argimiro Resende Secchi
COPPE – UFRJ

A Ata de defesa com as respectivas assinaturas dos membros encontra-se no processo de vida acadêmica do aluno.

To my loving family.

ACKNOWLEDGEMENTS

I thank my family for all their support, especially my wife, Caroline, for her thoughtful help and comprehension, and my mother, Silmara, and my father, Alexandre for being who they are and their inspirational thoughts.

My advisor Professor Doctor Ana Maria Frattini Fileti for her help, support, suggestions and tutoring. I also thank Professor Doctor Flávio Vansconcelos da Silva, especially with his tips about the experimental plant. I thank all my colleagues from LCAP (Laboratório de Controle e Automação de Processos) group for all the laughs and support.

I thank all the professors that taught me at UNICAMP for all the knowledge bestowed upon me, and also CNPq for the financial support.

Resumo

Extremum-Seeking Control (ESC) é uma estratégia de controle que independe de um modelo e possui resultados promissores em diversas aplicações, como no aumento da eficiência energética de usinas eólicas (CIRI et al., 2017), controle da posição de balões de alta altitude (VANDERMEULEN; GUAY; MCLELLAN, 2017), aumento da eficiência de células a combustível (BIZON, 2017), entre outros. Porém essa técnica não é muito explorada na área de engenharia química. O potencial de hidrogênio, ou pH, é um fator importante que deve ser controlado em diversos sistemas, como nos sistemas de tratamento de água e efluentes e na produção de açúcar. Entretanto, seu controle pode ser difícil não apenas devido a seu forte comportamento não-linear, mas também devido ao conhecimento esparso das propriedades do líquido a ser tratado, além do fato que elas podem se alterar com o tempo durante o procedimento de controle. Esse trabalho propõe duas técnicas baseadas em *extremum-seeking* para resolver esses problemas. A primeira consiste em um controlador PI sintonizado através de uma série de experimentos onde seus parâmetros são calculados através de um algoritmo do tipo *extremum-seeking*, já a segunda consiste na injeção direta de perturbações no sistema para estimar seu estado atual e definir, em média, o valor ótimo das variáveis manipuladas. Ambas técnicas foram avaliadas através de simulações e experimentos que foram feitos com o uso do *software SciLab* e seu módulo *XCos*. Os experimentos de controle de pH foram realizados em uma planta piloto montada no Laboratório de Controle e Automação de Processos (LCAP) na UNICAMP. Uma interface de comunicação personalizada foi desenvolvida para facilitar a troca de dados entre qualquer programa, incluindo o *SciLab*, e o Controlador Lógico Programável (CLP) da planta. Um supervisório também foi desenvolvido. O reator contínuo de mistura estudado possui três entradas: 1) Efluente a ser tratado (solução tamponada), 2) Solução de hidróxido de sódio diluído (base) e 3) Solução de nitrato de hidrogênio diluído (ácido). O pH da mistura reacional foi medido e seu valor controlado manipulando-se a vazão de entrada de ácido e/ou base. No caso do ESC, simulações mostraram como a saturação da variável manipulada possui grande influência na robustez do sistema de controle, mas com esse problema resolvido o sistema pôde ser controlado com sucesso pelo ESC com ajuste da vazão de ácido, de base e ambas simultaneamente tanto em simulações como experimentalmente. O controlador PI sintonizado através de *Extremum-Seeking* mostrou-se capaz de controlar com sucesso o pH do sistema por meio de simulações.

Abstract

Extremum-Seeking Control (ESC) is a model-free control strategy that has promising results in a range of applications such as increasing the power capture of wind turbines (CIRI et al., 2017), position control of high-altitude balloons (VANDERMEULEN; GUAY; MCLELLAN, 2017), increasing the efficiency of fuel cells (BIZON, 2017) and so on, but is not yet very explored in the area of chemical engineering. The potential of hydrogen, or pH, is an important factor to be controlled in various systems, such as water treatment and sugar processing. However, its control can be difficult not only due to its strong non-linear behaviour but also due to the lack of full or even partial knowledge of the properties of the liquid to be treated, which can change tremendously during the control procedure. This work proposes two techniques based on extremum-seeking to solve those problems. For the first technique a PI controller is tuned with a series of experiments and its parameters are calculated through a extremum-seeking algorithm, for the second technique dither is injected directly into the system to estimate its current state and set, on average, the proper value of the manipulated variables. Both techniques were evaluated through simulations and experiments that were carried out using SciLab and its XCos module. The experiments were carried out with a pilot plant assembled at the Laboratory of Automation and Process Control (LCAP) at UNICAMP. A custom communication interface was developed to allow easy data transfer between any software, including SciLab, and the Programmable Logic Controller (PLC) of the plant. A graphical interface was also developed. The continuous stirred reactor studied has three inlets: 1) Effluent to be treated (buffered medium), 2) Diluted sodium hydroxide (base), and 3) Diluted hydrogen nitrate (acid). The pH of the reactional mixture was measured and its value controlled by manipulating the flow rate of the acid and/or base inlets. The level of the reactor is controlled via a pump at the outlet of the reactor. For ESC, simulations showed how manipulated variable saturation played important part on the robustness of the control system. After this problem is dealt with, simulations and experiments showed that ESC was able to successfully control the pH of the system when adjusting only one flow rate at a time, or both at the same time. Simulations also showed that the PI controller tuned through extremum-seeking was also able to successfully control the pH of the system.

Figure Index

Table Index

Symbology

Latin

e - system reference error

f_c - continuous-time frequency

f_d - normalized frequency

h_f - discrete-time high-pass filter constant

k - continuous-time integrating factor

K - proportional constant of a PID

q_i - flow rate of fluid in line i

r - controller reference signal

s - Laplace domain complex variable

T_d - derivative constant of a PID

T_i - integral constant of a PID

T_{p_i} - period in seconds of dither i

T_s - sampling time in seconds

u - control action signal

$VAR\text{-}nd$ - value of variable VAR without dither (-*nd* suffix)

y - system response

z - Z domain complex variable

Greek

α_i - dither amplitude

γ - discrete-time integrating factor

θ - parameter vector

$J(\theta)$ - cost function that depends on a parameter vector

θ^* - values of θ where J is at a minimum or a maximum

$\hat{\theta}$ - parameter estimate of θ^*

ω_l - continuous-time high-pass filter constant

ω_h - continuous-time low-pass filter constant

ω_i - dither frequency of factor i

Summary

1. Introduction

Most systems have a property that needs to be kept at or set to a given value, be it a level, a temperature, a concentration, or any other. This can be achieved by changing at least one factor that influences the state of the property, such as closing a valve or increasing the speed of rotation of the rotor of a pump.

While a person is able to do so, it is desirable to automate this process with a computer to not only decrease costs but also to increase its quality and safety. People are more prone to errors than machines as the performance of the latter is not affected by age, state of mind, physical health, emotions etc. It is very important to remove the human factor when controlling processes to increase its security, reduce errors and increase production rate or reduce costs (LI; POWELL; HORBERRY, 2012).

There are various types of automated control systems, from a simple on/off switch to set the level of a tank to a complex mechanistic model predictive controller to control fluid catalytic cracking. The choice of the right type of control system greatly depends on the system to be controlled.

The potential of hydrogen, or pH, is not only a difficult property to be controlled but also one with great industrial importance owing to the fact that it needs to be controlled in at least one unit operation on most industries. Some examples are wastewater treatment (CHEREMISINOFF, 2002), pharmaceuticals production, sugar processing (CHOU, 2000), and for chemical and biochemical processes (OBUT; ÖZGEN, 2008).

The difficulty of the control comes mainly from strong process nonlinearities (OBUT; ÖZGEN, 2008) Several techniques were developed to help control it, such as nonlinear H_∞ control (LONGHI et al., 2004), a sequential model predictive control based on a Wiener-Laguerre model (MAHMOODI et al., 2009), a fuzzy approximator (SALEHI; SHAHROKHI; NEJATI, 2009), an adaptive control with a linearizing controller (HENSON; SEBORG, 1994), and a nonlinear controller tuned with the iterative feedback tuning technique (MA; ZHANG, 2016).

The control strategies generally used in chemical plants are based on a mathematical model of the system, or mostly based on data that can be collected from them. The present work proposes the use of two extremum-seeking (ES) techniques, which are not usual in the scope of chemical engineering. These techniques are data-based with no prior knowledge of the system. The first directly controls a neutralization reactor and the second

tunes a PI controller for regulator and servo control of the same reactor. Computational simulation of the system allows prior performance assessment, while experimental tests shows the potential of ES control techniques in chemical process control.

This work is organized as follows. Section 2 presents its objective. Section 3 presents the theoretical framework with emphasis on the neutralization process and on the extremum-seeking techniques. In Section 4 the experimental system and the simulation environment is shown along with the control structure and the pH model used to simulate the system. Section 5 presents the simulation and experimental results along with their discussion, and is followed by the conclusion on Section 6. In Section 7 future works are proposed to augment and better analyze the control system.

2. Objective

In this section the objective is divided into two categories, the main which should be achieved and the specifics, which may be necessary to reach the main objective.

2.1. Main

Evaluate the viability of two different control strategies to control the pH of a neutralization reactor, one where extremum-seeking (ES) tunes a PI controller which in turn controls the pH, and another where extremum-seeking directly controls the pH. No prior knowledge of the system is required using ES strategies, which is the main attractive of this technique.

2.2. Specifics

– Develop a generic method of communication between the computer and the plant Programmable Logic Controller (PLC). Also develop the software where the algorithm will be implemented and use this method to communicate;

– Create a graphical user interface to manage and view the current state of the plant and ease its usage;

– Develop the first principles model of the neutralization reactor and simulate the plant under each control strategy;

– Simulate the system to estimate suitable parameters to be used on the experimental procedure;

– Run closed-loop experiments to test each algorithm and evaluate each, along with how to choose the best parameters to cover the desired pH operating range.

3. Theoretical framework

This section presents a literature survey to better understand the work proposal, introduce the notions of extremum-seeking and show how it can be used to control the system or tune a PI controller.

3.1. Neutralization Process

The process of neutralization is of great importance for the industry, especially for wastewater treatment and the production of pharmaceuticals (HENSON; SEBORG, 1994).

It consists on the reaction of a base and an acid until there is no excess hydroxide or hydrogen ions in the solution. This process is usually represented like the simplified equation

$$acid + base \rightarrow salt + water$$

Or

$$XH + Y \rightarrow X + YH$$

One example of a neutralization reaction is the neutralization of hydrochloric acid with sodium hydroxide, as shown

$$HCl + NaOH \rightarrow NaCl + H_2O$$

Neutralization is highly nonlinear and this property can be clearly seen with the titration curve of a solution, which usually is sigmoid-shaped and can have various gains depending on the composition of the solution. Figure 1 shows examples of titration curves and the difference on the behaviour of the potential of hydrogen, or pH, for different types of solutions.

The pH can be estimated with a probe that measures both the electrical potential between two electrodes and the temperature, as it is also a function of the temperature.

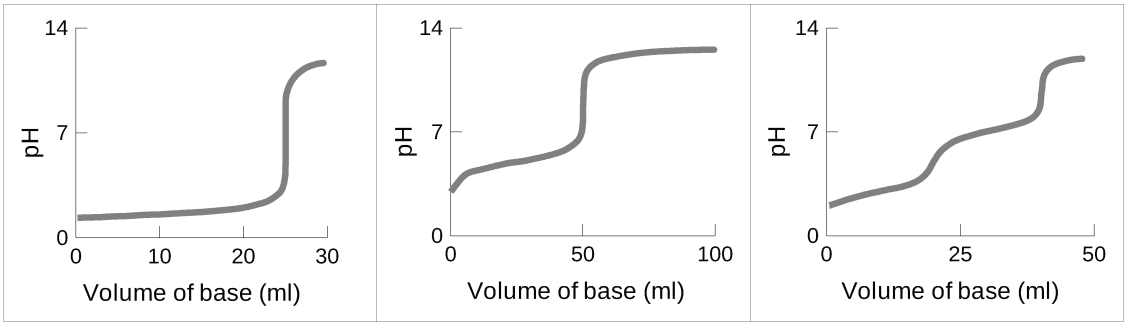

Figure 1 – Some examples of titration curves. From left to right, strong base with strong acid, strong base with weak acid, strong base with *polyfunctional* acid (SKOOG, 2012)

3.1.1. Chemical engineering applications

This section will illustrate how the neutralization process, or pH control in general, is used as an industrial operation along with what type of industry can employ this process to attain a desired property, such as respecting quality standards.

3.1.1.1. Wastewater treatment

The pH of the effluent to be treated is of special interest on wastewater treatment, the effluent should not only be released to the environment or returned to the system at a specific range of pH but its magnitude also interferes with the function of each operational unit, such as disinfection, biological treatment (aerobic or anaerobic), coagulation, clarification and neutralization. Some other important aspects to consider are corrosion control, adsorption and oxidation (CHEREMISINOFF, 2002).

Some studies addresses pH control in industrial wastewater anaerobic treatment (ANDERSON; YANG, 1992), in wastewater with acetic and propionic acids (LIU; CHEN; ZHOU, 2007) and sludge digestion (AI-GHUSAIN et al., 1994).

A process diagram example for wastewater treatment can be seen on Figure 2, where the digestion, disinfection and clarification operations are clearly shown.

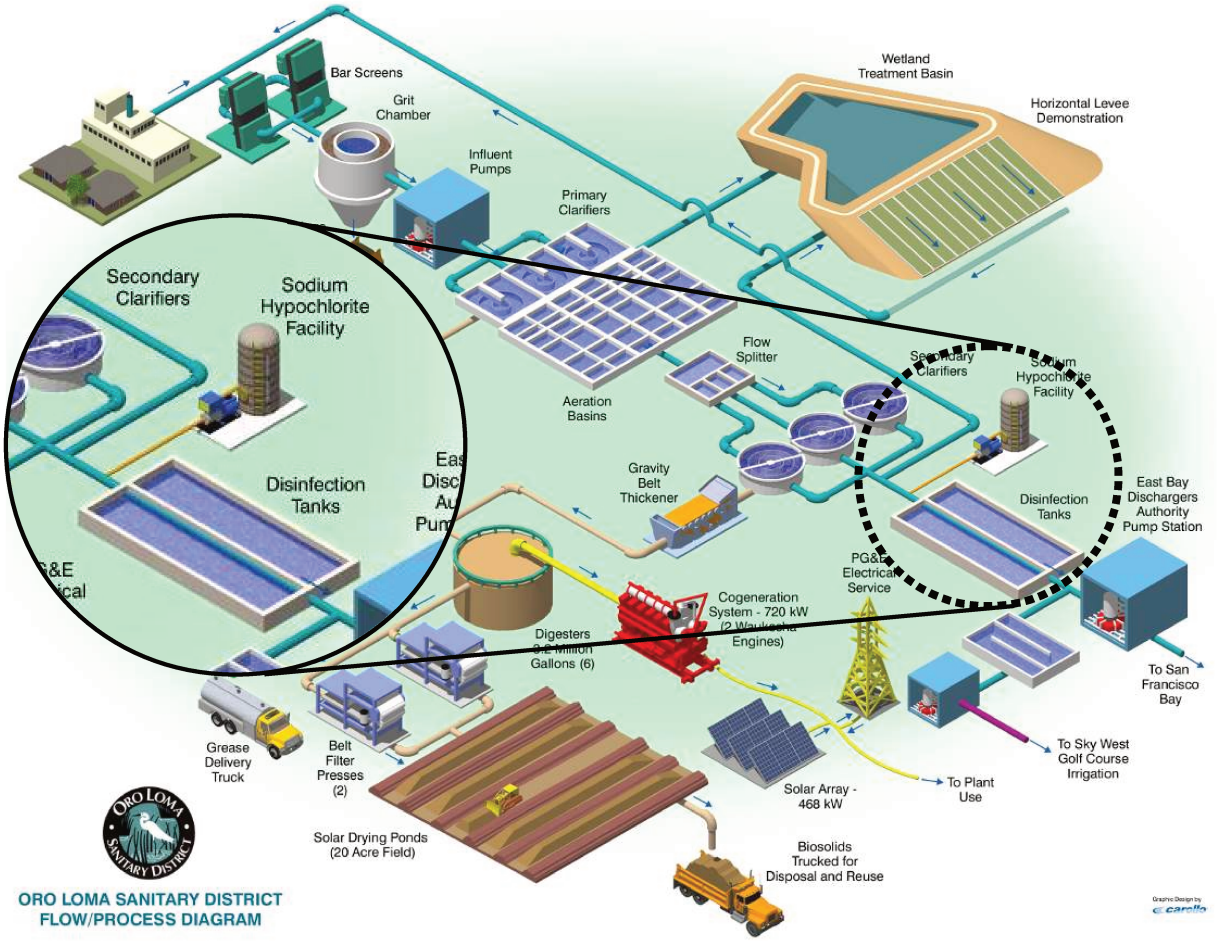

Figure 2 – Wastewater treatment process diagram (SANITARY DISTRICT, n/a).

3.1.1.2. Sugar industry

The control of the pH is important not only to increase the edibility of the sugar, but also on various intermediate steps to obtain the refined sugar, especially on clarification, decolourization and ion exchange (CHOU, 2000).

In respect to edibility, pH control is essential on every food processing industries and should be maintained within a range to respect safety and quality limits even with the difficulties associated its control, strong non-linearity, multi-input and time-varying properties (VIJAYARAGAVAN et al., 2015).

The diagram on Figure 3 is an example of the process flow diagram for sugar processing with sugar cane as feedstock, and the clarification step can be clearly seen.

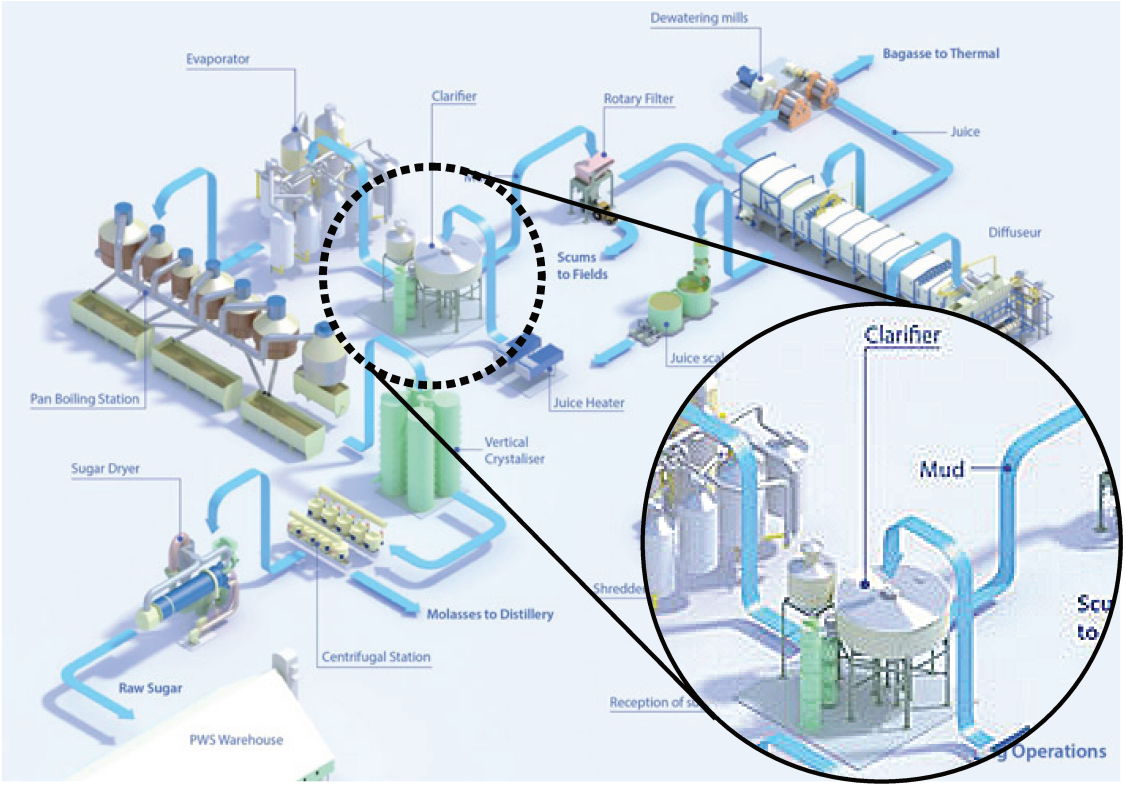

Figure 3 – Sugar cane processing diagram (OMNICANE, n/a).

3.1.1.3. Pharmaceutical manufacturing

There are various methods to manufacture drugs, and pH control is most important on production processes where the synthesis of the drug comes from a living being, such as yeast or bacteria. Thus its control is especially necessary on the fermentation step, as there is a natural shift on its pH while the nutrients are consumed and its magnitude affects the product yield (JERMAINE, 2014).

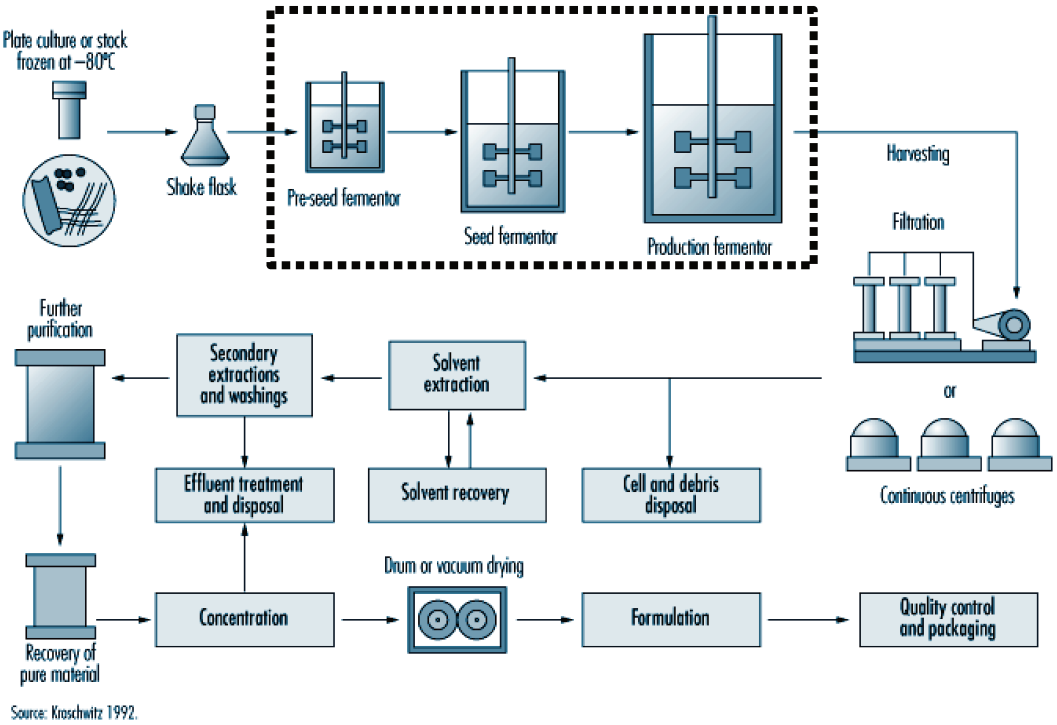

Figure 4 – Diagram of a fermentation process (D. TAIT, n/a).

3.1.2. Process control

There is diverse literature about the neutralization process due its challenges, with various proposals of techniques to control the system, such as an adaptive control strategy that uses an input-output linearizing controller with a reduced-order, open-loop observer (HENSON; SEBORG, 1994), a Model Predictive Control based on a Wiener-Laguerre model (MAHMOODI et al., 2009), a neural network-based predictive controller (ELARAFI; HISHAM, 2008) and a nonlinear controller tuned with the iterative feedback tuning technique (MA; ZHANG, 2016). These techniques will be quickly shown in this section to give an idea of their implementation and complexity to the reader.

3.1.2.1. Input-Output Linearizing Controller

Henson and Seborg (1994) proposed the use of a nonlinear control strategy using an augmented nonlinear controller with indirect parameters estimation to account for any unmeasured buffering changes on the effluent to be treated, which will be shown in this section.

The model is based on the Reaction Invariant Model, the same used in this study as shown on Section 4.8. The nonlinear controller design is based on a modified input-output linearization approach to account the implicit nature of the pH estimation equation, which is

similar to Equation 37 shown later. After development the actual non-adaptive input-output linearizing controller equation can be seen in Equation 1, and its terms in Equation 2, Equation 3, Equation 4, Equation 5, and Equation 6. (HENSON; SEBORG, 1994)

To reach these equations it is assumed that $x \triangleq [W_{a4} \;\; W_{b4} \;\; h_1]^T$, $d \triangleq q_2$, $u_a \triangleq q_3$, and $y \triangleq pH$.

$$u_a = \frac{\varepsilon^{-2}\int_0^t (y_{sp}-y)d\tau - 2\varepsilon^{-1}y + c_y^{-1}(x,y)c_x(y)[f(x)+p(x)d]}{-c_y^{-1}(x,y)c_x(y)g(x)} \tag{1}$$

$$f(x)=\left[\frac{q_1}{A_1 x_3}(W_{a1}-x_1) \;\; \frac{q_1}{A_1 x_3}(W_{b1}-x_2) \;\; \frac{1}{A_1}(q_1 - C_{v4}(h_1+z)^n)\right]^T \tag{2}$$

$$g(x)=\left[\frac{1}{A_1 x_3}(W_{a3}-x_1) \;\; \frac{1}{A_1 x_3}(W_{b3}-x_2) \;\; \frac{1}{A_1}\right]^T \tag{3}$$

$$p(x)=\left[\frac{1}{A_1 x_3}(W_{a2}-x_1) \;\; \frac{1}{A_1 x_3}(W_{b2}-x_2) \;\; \frac{1}{A_1}\right]^T \tag{4}$$

$$c_x(y) = \left[1 \;\; \frac{1+2\times 10^{y-pK_2}}{1+10^{pK_1-y}+10^{y-pK_2}} \;\; 0\right]^T \tag{5}$$

$$c_y(x,y) = (\ln 10)\left[10^{y-14}+10^{-y}+x_2\frac{10^{pK_1-y}+10^{y-pK_2}+4\{10^{pK_1-y}\}\{10^{y-pK_2}\}}{\{1+10^{pK_1-y}+10^{y-pK_2}\}^2}\right] \tag{6}$$

Where x is a state variable, W_i an invariant i of the system, h_1 the level of liquid, q_1 the acid stream flow-rate, q_2 the buffer stream flow-rate, q_3 the base stream flow-rate, ε is the controller tuning parameter, y_{sp} is the desired set-point of the control variable, y the current state of the control variable, $f(x)$, $g(x)$, $p(x)$, and $c_x(y)$ are vectors based on the model of the system, C_{v4} is the constant of the outlet valve, z is the vertical distance between the bottom of the reactor to the outlet valve, n is a constant valve exponent, A_1 is the reactor base area, and pK_1 and pK_2 are equilibrium constants of the system.

Equation 7 is the obtained closed-loop transfer function if the model perfectly matches the plant.

$$\frac{y(s)}{y_{sp}(s)} = \frac{1}{(\epsilon s+1)^2} \tag{7}$$

The linearized pH model has an unobservable mode, thus a closed-loop observer cannot exist and an open-loop observer is used to sequentially estimate the invariants of the

system. After further developing of the system, including discretizations, the resulting control law is shown on Equations 8, 9, 10, and 11. (HENSON; SEBORG, 1994)

$$\gamma(\hat{x}_k, y_k)u_{c,k} = \gamma(\hat{x}_{k-1}, y_{k-1})u_{a,k-1} + \varepsilon^2 \Delta t(y_{sp} - y_k) - 2\varepsilon^{-1}(y_k - y_{k-1}) + \alpha(\hat{x}_k, y_k) - \alpha(\hat{x}_{k-1}, y_{k-1}) + \beta(\hat{x}_k, y_k)d_k - \beta(\hat{x}_{k-1}, y_{k-1})d_{k-1} \quad (8)$$

Where:

$$\alpha(\hat{x}_k, y_k) \triangleq c_y^{-1}(\hat{x}_k, y_k)c_x(y_k)f(\hat{x}_k) \quad (9)$$

$$\beta(\hat{x}_k, y_k) \triangleq c_y^{-1}(\hat{x}_k, y_k)c_x(y_k)p(\hat{x}_k) \quad (10)$$

$$\gamma(\hat{x}_k, y_k) \triangleq c_y^{-1}(\hat{x}_k, y_k)c_x(y_k)g(\hat{x}_k) \quad (11)$$

The results of this technique with even further development of the equations, including the addition of a parameter estimator of unmeasured buffer flow rate, can be seen in Figure 5. This figure shows its performance when the buffer flow rate suddenly changes, a stable system even with low buffer flow rate (not shown). Note that q_3 is the base flow rate.

This controller depends on the availability of a model but has acceptable performance characteristics on set-point change and disturbance rejection, especially when there is change on the buffer flow rate or its properties.

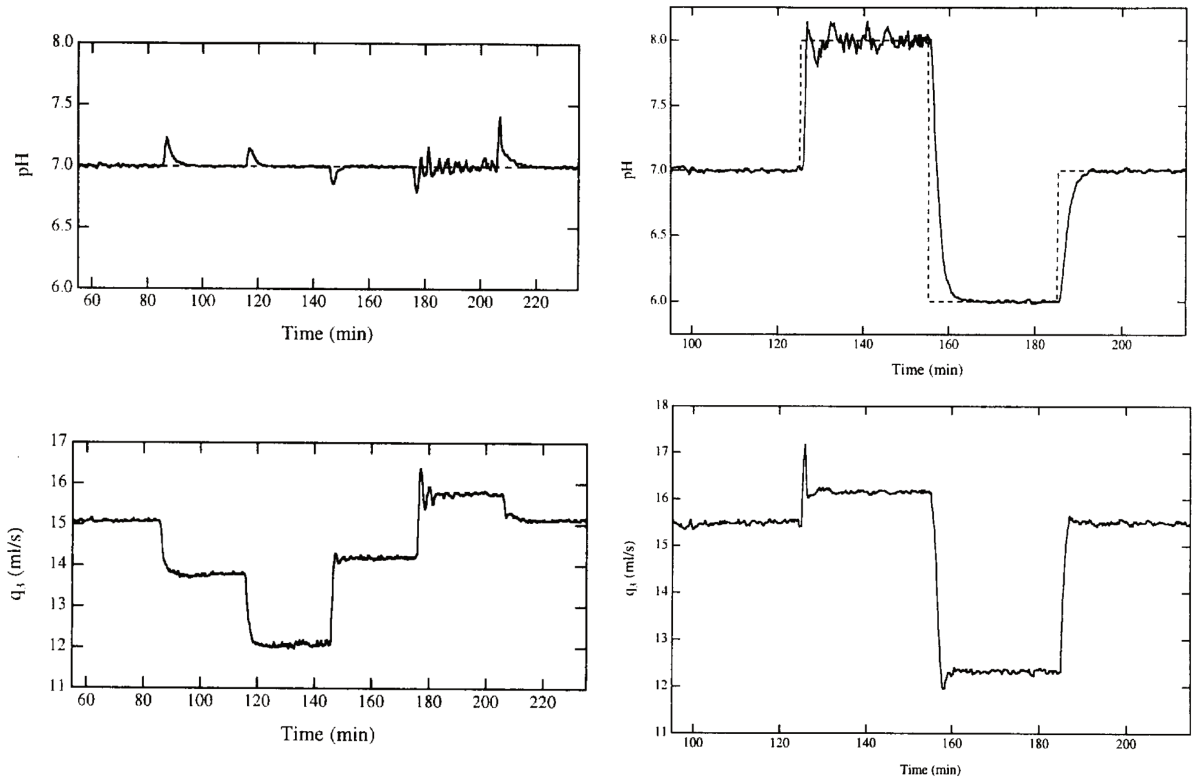

Figure 5 – Performance of the adaptive nonlinear controller. Left: regulatory action on buffer flow rate change. Right: servo action. (HENSON; SEBORG, 1994).

3.1.2.2. MPC based on Wiener-Laguerre model

Model Predictive Control is usually employed to control the future state of a system through an explicit process model, which can be either linear or nonlinear. Most systems are nonlinear, and while a linear model is enough for rejection of disturbances, a complete nonlinear model should be used on set point change to not result in poor control performance (MAHMOODI et al., 2009).

Mahmoodi *et al.* (2009) proposed the use of a MPC with a model based on Wiener-Laguerre, where a Wiener structure is coupled with Laguerre filters for its linear part, and polynomials for its nonlinear part. This controller is shown in this section.

The resulting structure can be seen on Equation 12, which can be written as shown on Equations 13, 14, 15, and 16. Even though it is a simplified structure when compared to first-principle models, it is still complex and its parameters still must be identified through experiments.

$$\hat{y}(k)=h_0+\sum_{i=1}^{N} h_i l_i(k)+\sum_{i=1}^{N}\sum_{j=i}^{N} h_{ij} l_i(k) l_j(k) + \cdots + \sum_{i=1}^{N}\sum_{j=i}^{N}\cdots\sum_{k=r}^{N}\sum_{m=k}^{N} h_{ij\ldots km} l_i(k) l_j(k)\ldots l_k(k) l_m(k) \tag{12}$$

$$\hat{y}(k)=H^T \varphi(k)+\varepsilon(k) \tag{13}$$

Where:

$$H=[h_0, h_1, \ldots, h_R]^T \tag{14}$$

$$R=\sum_{i=1}^{P}\binom{N+i-1}{i-1} \tag{15}$$

$$\varphi(k)=[1 l_1, \ldots, l_N, l_1^2 l_1 l_2, \ldots, l_N^2, l_1^3 l_1 l_2 l_3, \ldots, l_N^3, \ldots, l_N^M]^T \tag{16}$$

And $\varepsilon(k)$ is the equation error. On those equations $l_i(k)$ is the output of the *i*th order Laguerre filter at *k*th sampling time, N the number o Laguerre networks and H are weights of the sum shown on Equation 12. The experiments used to identify the Laguerre model parameters consisted in a series of step tests according to a Generalized Binary Noise (GBN) technique, and to identify the Wiener and the Wiener-Laguerre models consisted in a series of step tests according to a Generalized Multi-level Noise (GMN) technique. According to Mahmoodi *et al.* (2009) each simulated experiment took 100 simulated minutes (1h40) to complete.

The servo control performance of a simulated reactor with buffer ($NaHCO_3$), acid (HNO_3) and base (NaOH) inlets can be seen on Figure 6, and shows that it can quickly control

the pH with little to no overshoot. The flow rate of the base inlet is the manipulated variable for the simulation.

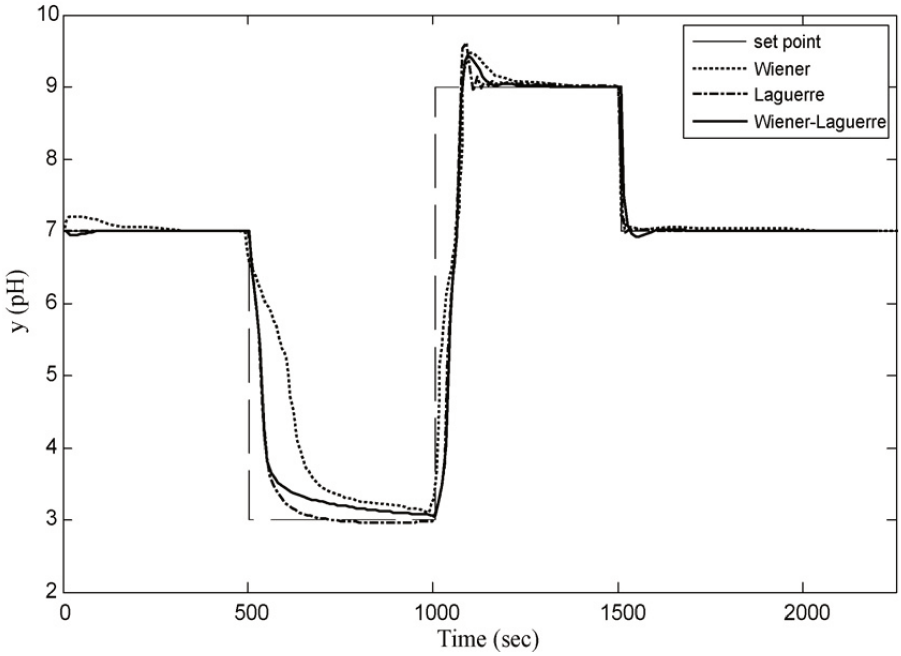

Figure 6 – pH system controlled with method proposed by Mahmoodi *et al.* (2009)

3.1.2.3. Neural-network predictive controller

The use of artificial intelligence and neural networks is expanding to all areas, from simple systems modelling to generating intelligible human speech. Neural network is a versatile tool to model a plethora of systems, from linear to nonlinear. Elarafi and Hisham (2008) proposed the use of a neural network predictive controller to control a pH neutralization process, and it will be shown in this section.

The system is modelled though a neural network composed of three hidden layers, two delayed inputs, and two delayed outputs. The network is trained offline in batch mode with experimental data collected from the operation of the process using Levenberg-Marquardt backpropagation training function.

The proposed MPC is a receding horizon control and the predictions are used to minimize the cost function show on Equation 17.

$$J=\sum_{j=N_1}^{N_2}\left(y_r(t+j)-y_m(t+j)\right)^2+\rho\sum_{j=1}^{N_u}\left(u'(t+j-1)-u'(t+j-2)\right)^2 \quad (17)$$

Where y_r is the desired system response, y_m is the network model system response, u' variable is the tentative control signal, the value of p determines the

contribution that of the sum of the squares of the control increments to the performance index, N_1, N_2 and N_u define the horizons over which the tracking error and the control increments are evaluated,

According to Elarafi and Hisham (2008), a comparison of the MPC controller performance with PI and PID can be seen on Figure 7. The parameters of the MPC controller are set to $N_1=1$, $\rho=0.001$, $N_2=6$, $N_u=2$ with a PI and PID controllers tuned using Ziegler-Nichols closed loop tuning method. The base flow rate is the manipulated variable.

The Neural Network MPC managed to achieve fast response time and fast decay time with little overshoot when compared to the PI and PID controllers, and should not suffer any problems with noise when compared to a controller with derivative action.

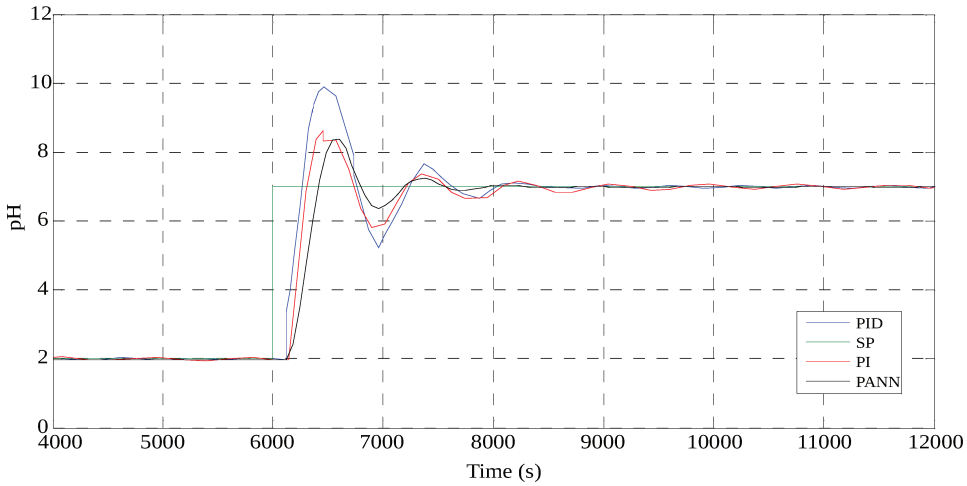

Figure 7 – Performance comparison between the neural network MPC (PANN), PI and PID (ELARAFI; HISHAM, 2008)

3.1.2.4. Nonlinear controller tuned with Iterative Feedback Tuning

This section shows the nonlinear controller proposed by Ma and Zhang (2016). It consists in an iterative feedback tuned PI controller to control the flow rate of the inlet of the system, while the acid or base flow rate is set according to a ratio converter created from experimental data collected from the plant. The control diagram where the acid flow rate is the manipulated variable can be seen on Figure 8, where the converter that calculates the desired flow-rate of acid can be seen. This converter is installed with off-line data that allows the estimation of the desired acid:base ratio, which is then multiplied by the base reference flow-rate to result in the desired acid flow-rate.

The off-life data in the converter can be seen in Figure 9 with the pH as a function of the ratio between the flow rate of base and acid. The behaviour of the pH depends on the characteristics of the inlet, thus new pH-ratio experiments must be done if there is any change on its properties.

Through the iterative feedback tuning (IFT) process it was possible to tune the PI controller in five iterations, with a total of ten experiments, and the proportional coefficient increased from 5.00 (first guess) to 20.3, while the integral coefficient increased from 1.00 to 32.1. The performance of the control scheme can be seen on Figure 10, and it can quickly control the system even though the strong overshoot of the response signal can be a problem..

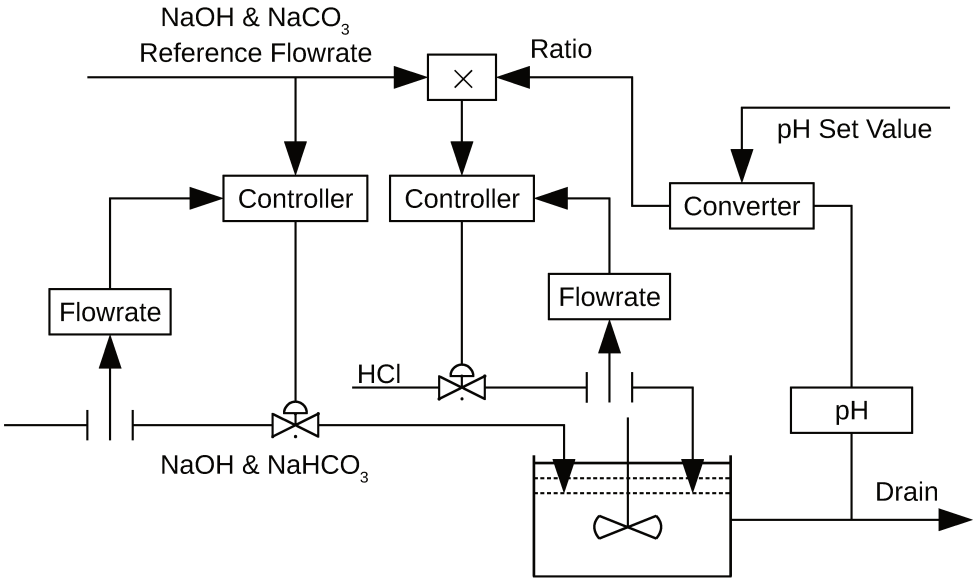

Figure 8 – Adapted diagram with the control scheme proposed by Ma and Zhang (2016).

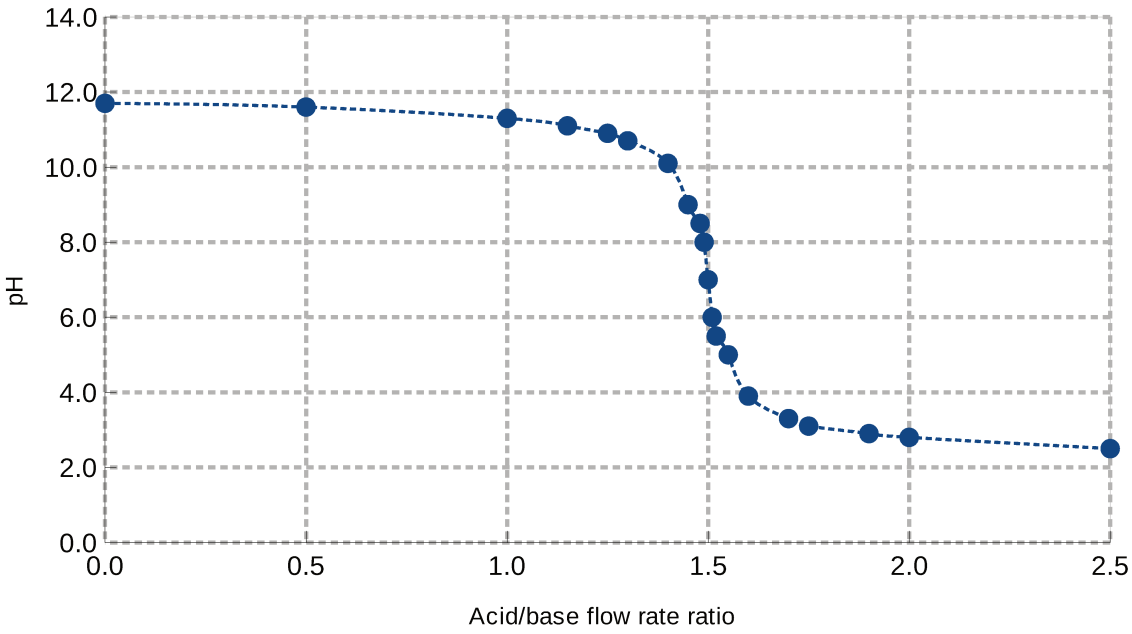

Figure 9 – pH flow-rate ratio

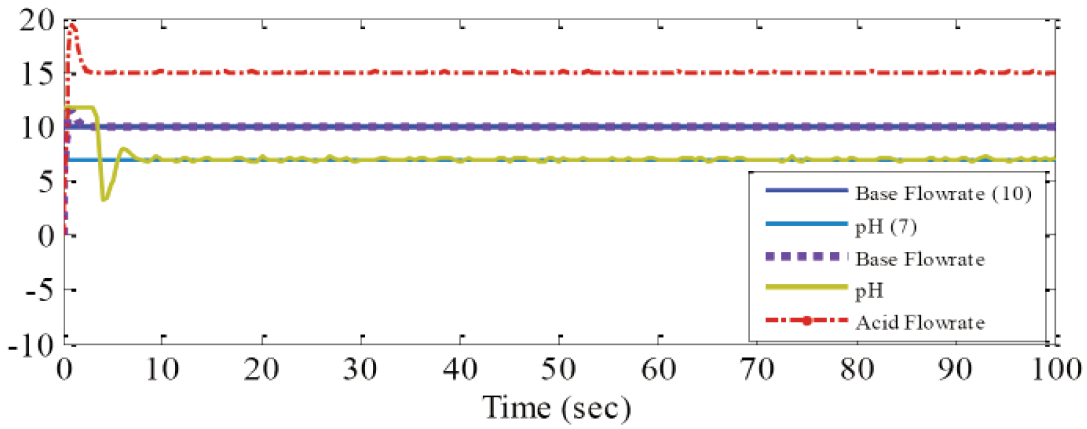

Figure 10 – Performance of the control scheme as proposed by Ma and Zhang (2016).

3.1.2.5. Discussion

As shown, most of those techniques relies on a model of the system to be controlled, and as such the quality of the model directly influences the quality of the control system. On the other hand, some techniques, like iterative feedback tuning and neural networks, can be used with a data-driven approach. This approach considers some knowledge of the system and can model a complex process with limited input-output information. However, both techniques need a previous data set, either to train the neural network or to estimate the gradient descent of a cost function. That is not the case of extremum-seeking

control, which is a data-driven technique that can estimate the gradient descent *in loco* to control a system or tune a controller. Due to this advantage this last technique will be evaluated on this work.

3.2. Extremum seeking

Extremum-seeking got a lot of attention and a large amount of research effort was spent in the last century on it (DOCHAIN; PERRIER; GUAY, 2011; KRSTIĆ, 2000), but how it worked was not fully understood giving way to other techniques that had better theoretical foundation. The interest for extremum-seeking was renewed with the mathematical proof of its stability (KRSTIĆ, 2000; KRSTIĆ; WANG, 2000). This technique was already successfully applied in various systems, such as fuel flow control to achieve maximum pressure, combustion process control for internal combustion engines and gas furnaces, and anti-lock braking system control (DOCHAIN; PERRIER; GUAY, 2011).

Extremum-seeking control has two different approaches, one entirely based on data from the process (KRSTIĆ, 2000), and other based on not only data but also on a minimum knowledge of the process model, even with unknown parameters (DOCHAIN; PERRIER; GUAY, 2011).

The first approach is based on perturbations of the system where a gradient of a cost function can be estimated and used to minimize the cost function value. This approach can be used with black-box cost functions that are measured online and works better on systems where the cost function quickly reaches steady-state, but a proper gradient is not guaranteed to be generated so the cost function must be carefully chosen.

The second approach assumes that the cost function is known as a function of the system state and uncertain parameters from the system dynamic equations. This approach not only identifies and regulates the system simultaneously, but also guarantees some degree of proper gradient generation (DOCHAIN; PERRIER; GUAY, 2011).

3.2.1. How Extremum-Seeking Control works – simplified

The search for the optimum using extremum-seeking can be done in continuous-time or discrete-time. As the cost function, the function to be minimized or maximized, needs to reach steady-state before another perturbation is injected into the system it is important to note the dither timescale. If the cost function is able to quickly reach steady-state a continuous-time extremum seeking algorithm can be used, potentially simplifying its tuning. However, in cases where steady-state is not reached quickly a discrete-time extremum-

seeking algorithm is recommended with the purpose that each perturbation happens after steady-state is reached.

In this section both the continuous-time and discrete-time extremum-seeking algorithms will be explained having as basis the stability proof paper by Krstić and Wang (2000).

Various assumptions are made to prove its stability. Both the evaluated system and control law have to be smooth, there is a smooth function such that the derivative system is zero if and only if the state of the system is a value of the function; the control law is robust and locally stabilizes the system without any knowledge of it; and, most importantly, there is a value assumed by the control law which maximizes (or minimizes) the system output (KRSTIĆ; WANG, 2000).

The diagram shown on Figure 11 exemplifies the continuous time extremum-seeking control algorithm, where J is the cost function, the function whose maximum or minimum must be found, which is a function of the system state $\dot{x}$ which in turn is a function of the manipulated variable, or signal, θ.

ES control is based on a peak seeking scheme and employs periodic perturbations ($\alpha\cos[\omega t]$) to estimate the system gradient. These perturbations have to be slow compared to the system dynamics to not interfere with the peak seeking scheme itself, and they are added to the signal $\hat{\theta}$, the estimate of θ^*, which is the value of the parameter that lead to the extremum.

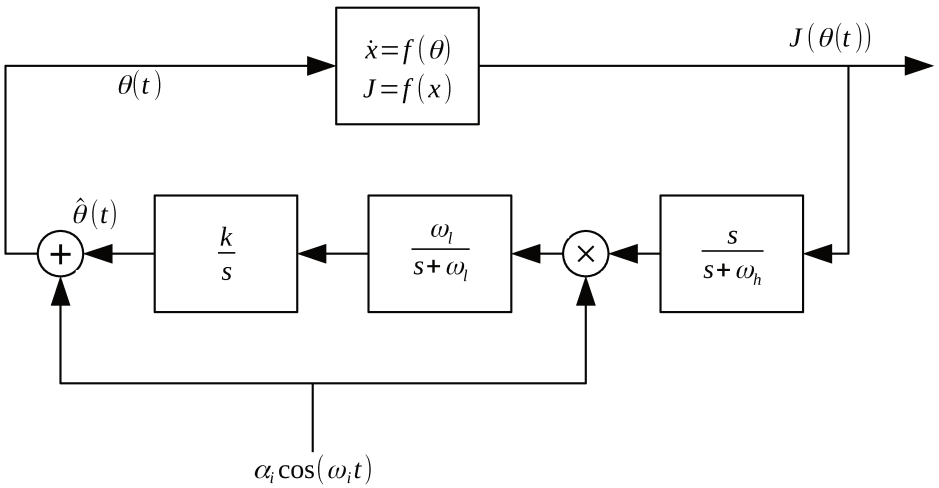

Figure 11 – Continuous time extremum-seeking control algorithm, adapted (KRSTIĆ; WANG, 2000)

A high-pass filter ($s/(s+\omega_h)$) is used to eliminate any DC-component[1] of the periodic system response $y=J(\theta(t))$ in order to approximate the perturbation and the periodic response to two sinusoids. If $\hat{\theta}$ is larger than θ^* the periodic response will be out of phase with the perturbation, while it will be in phase when $\hat{\theta}$ is smaller than θ^*. The product of the high-pass filtered system response and the perturbation will have another DC-component which is then extracted with a low-pass filter ($\omega_l/(s+\omega_l)$). The integrator (k/s, an integral controller) is then used to drive the gradient to zero and tune $\hat{\theta}$ to θ^* (KRSTIĆ; WANG, 2000).

The discrete-time algorithm is similar to the continuous-time algorithm with two main differences. First, while it is interesting to low-pass the signal before integration, this step is not required for the discrete variant and may introduce undesired dynamics. Second, the algorithm depends on the sample rate, not on time. A diagram of the discrete-time algorithm can be seen on Figure 12.

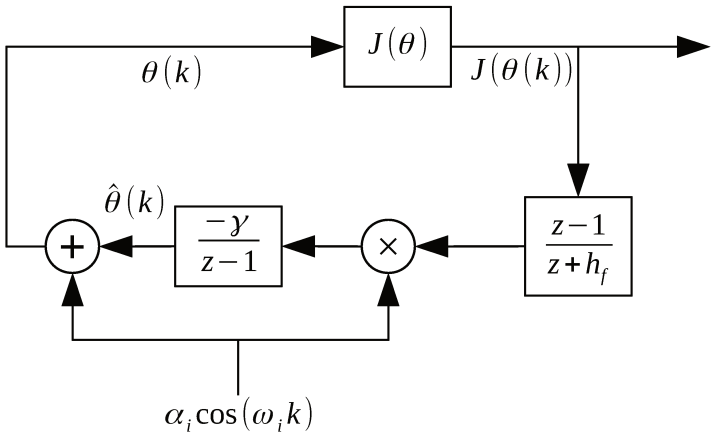

Figure 12 – Discrete ES scheme proposed by Killingsworth and Krstic (2006).

Where $J(\theta)$ is a cost function designed to be a function of θ, $J(\theta(k))$ is the cost function but sampled, and $\hat{\theta}(k)$ the sampled estimate of the value that leads $J(\theta)$ to its extremum.

For discrete-time ES control, k is the sample number count, $z-1/z+h_f$ the high pass filter, and $-\gamma/z-1$ the integrator. The sampling time is arbitrary, and should be set according to the time-scale of the system and the filter.

1. Frequency component when there is no or almost no change of the signal in the time domain, can also be understood as the average of the signal.

On both algorithms the entire system has three time scales, the plant which needs to be the fastest, the periodic perturbations, and then the filters, which needs to be the slowest (KRSTIĆ; WANG, 2000).

3.2.2. Control structures

In this section two control structures are proposed, in the first a PID controller is tuned after a series of experiments where the best parameters for the proportional, integral and derivative actions are estimated using extremum-seeking; and in the second extremum-seeking is used to directly control a parameter by adjusting another.

3.2.2.1. PID Tuning

A possible control structure with extremum-seeking was proposed by Killingsworth and Krstic (2006), where the parameters of a PID controller are computed after a step change on the set-point to minimize the cost function shown on Equation 18, which is assumed to have a minimum.

$$J(\theta) \triangleq \frac{1}{(T-t_0)} \int_{t_0}^{T} (e(t,\theta))^2 \, dt \tag{18}$$

Where $e(t,\theta) \triangleq r(t) - y(t,\theta)$ is the difference between the reference and the output signal of the closed control loop, $\theta \triangleq [K_c, T_i, T_d]^T$ is the parameters of the PID controller, T is the final time and t_0 is the time of the first peak to effectively place zero weighting on the initial transient portion of the response to not take it in account on the cost function calculation. The tuning scheme can be seen on Figure 13.

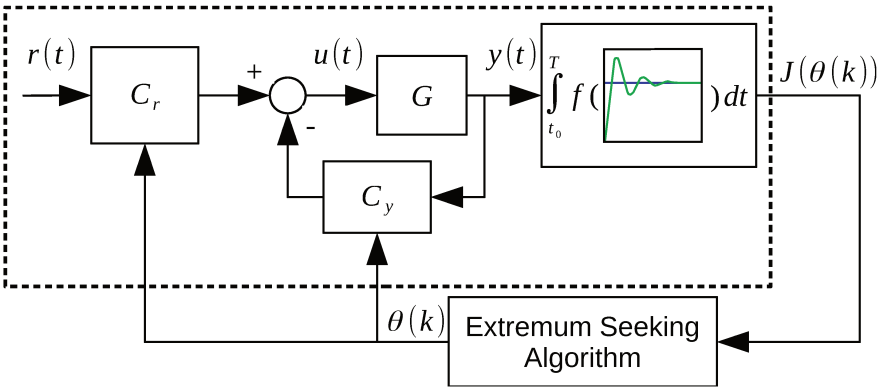

Figure 13 – General ES PID tuning scheme proposed by Killingsworth and Krstic (2006)

In Figure 13 the function G represents a model or transfer function of the plant, C_y and C_r a two-degree of freedom PID controller where the derivative action does not act upon changes on the reference signal, $r(t)$ represents the reference signal, $u(t)$ the control

signal, and $y(t)$ the output of the plant. The integral represents the cost function and θ the tuning parameters of the controller, such as the proportional gain K_c, the integral time T_i and the derivative time T_d. The transfer functions in Laplace domain of the controllers C_r and C_y are shown on Equation 19 and Equation 20.

$$C_r(s) = K_c\left(1 + \frac{1}{T_i s}\right) \tag{19}$$

$$C_y(s) = K_c\left(1 + \frac{1}{T_i s} + T_d s\right) \tag{20}$$

Each parameter of the PID controlled must be estimated with its own perturbation, which in turn has its own amplitude and frequency. The algorithm generates the gradient by changing the parameters of the PID controller after each step experiment.

3.2.2.2. Extremum-seeking control

Extremum-seeking can also be used to directly adjust the manipulated variable to control the controlled variable, hence the name extremum-seeking control. This strategy can be used either on continuous-time or as discrete evaluations according to Figure 11 and Figure 12, respectively. It requires fast plant dynamics or large sampling times to allow the system to reach steady-state before another perturbation is made.

An example control system can be seen on Figure 14, where the level of a vessel is controlled by changing the flow rate of fluid at the vessel outlet.

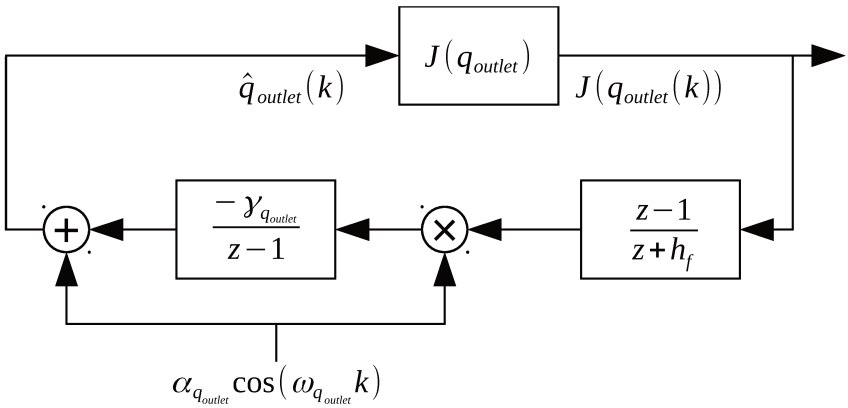

Figure 14 – Example of an extremum-seeking controller to control the level of a vessel.

Just like when used for PID tuning, each parameter to be estimated must have its own perturbation frequency and amplitude, low-pass filter and integrator.

3.3. Discussion

This section addressed the process of neutralization, its challenge and some techniques used to overcome them. Among the control techniques shown, those with a data-based approach were discussed, especially extremum-seeking which will be the focus of this work and doesn’t see much work in the scope of chemical engineering.

This technique can be used with no prior knowledge of the system to be controlled, just the grasp of a cost function that has an extremum, that depends on the state of the system, and that a gradient can be generated by somehow modifying the input of the system. This makes the technique very appealing on systems that are difficult to model or with strong nonlinear behaviour. The next section will show the proposed methodology to evaluate the effectiveness of this technique to control the process of neutralization.

4. Methodology

This section will describe the experimental neutralization plant, its current control system, the control process proposed on this study, along with implementation details such as the simulation and experimental methods, and the software implementation of the control loop.

4.1. Experimental pH neutralization plant

This study proposes a neutralization system of an effluent, a standard buffered medium, by a strong base and a strong acid similar to work done previously by Vaz da Costa (2014), whose output variable (pH) has a complex behaviour due to its high nonlinearity caused by the property of a strong base/strong acid neutralization

The proposed control strategy was evaluated using an experimental neutralization plant already available at LCAP at Unicamp where some modifications were made to better fit the desired analysis. A picture of the plant is shown on Figure 15, and its instrumentation diagram is shown on Figure 16.

Figure 15 – Experimental plant for the neutralization reaction. (LCAP/FEQ/Unicamp)

The reactor is fed by three streams, one carries a strong acid, another carries a strong base and a third carries the effluent, the buffered medium to have its pH corrected. The buffered medium is a solution of $NaHCO_3$ at 0.03 M or 0.02 M, the acid is a solution of nitric acid at 0.0034 M, and the base is a solution of sodium hydroxide at 0.0057 M. Three barrels are used to store each up to 100 L of each solution.

The management and the monitoring of the plant is done through the computer UC-101, which is also responsible for both extremum-seeking strategies shown on Section 3.2.2.

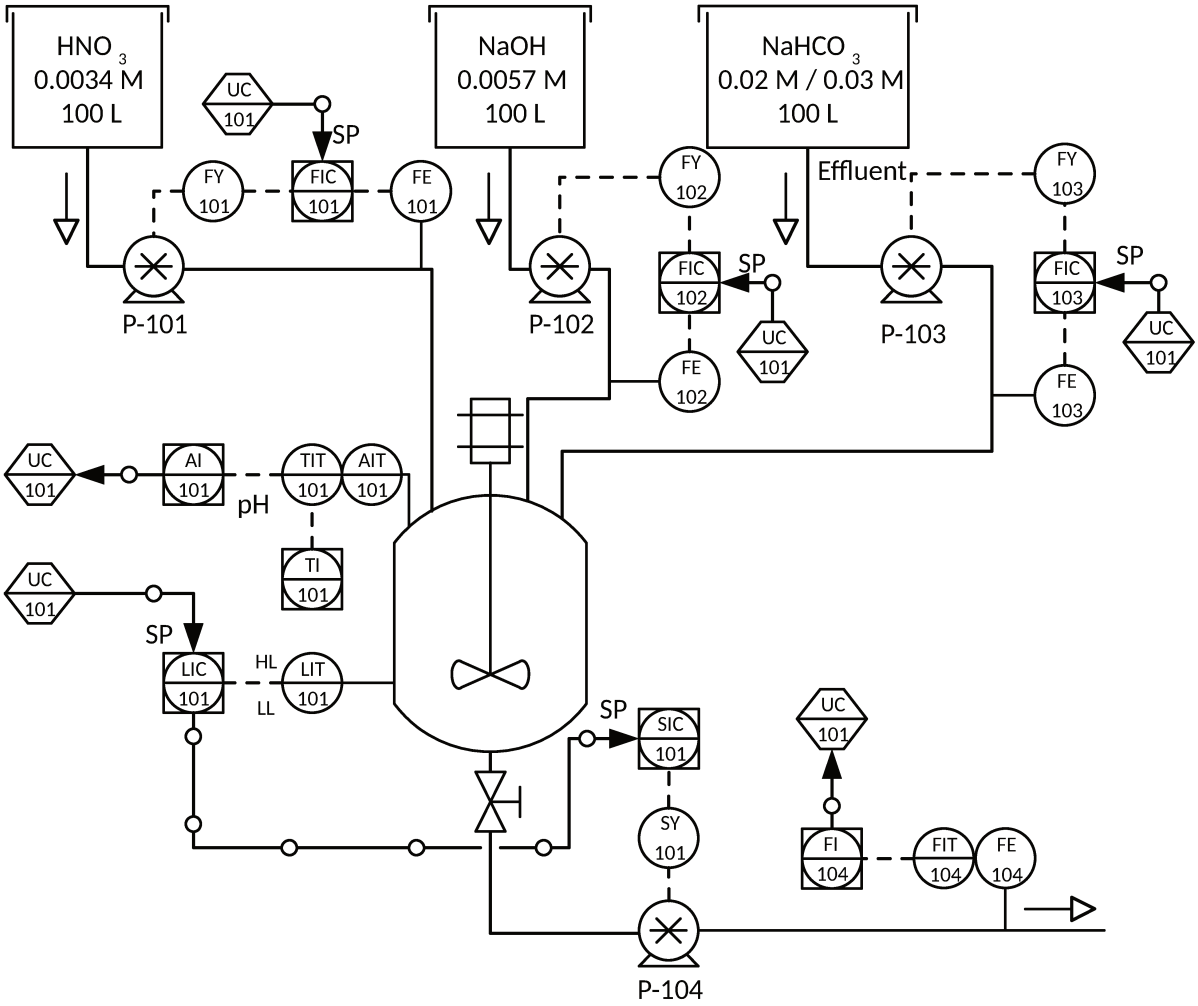

Figure 16 – Instrumentation diagram.

The flow rate from two tanks is controlled using pumps from *Coleparmer* with magnetic coupling, as shown on Figure 17a. The head of P-101 and P-102 is the *Micropump* model *RZ-73004* while the head of P-103 is the *Micropump* model *1800-00* (Figure 17b). They are coupled to the motors model *RZ-75211-55* and *000-336*, respectively. 4-20 mA analog signals actuate these pumps.

The buffered medium flow rate is controlled by FIC-103, a PI controller. Its flow rate was fixed on most experiments, except when a disturbance was applied. The acid and base flow rates are controlled by FIC-101 and FIC-102, respectively. Both are also PI controller whose references are set accordingly to control the pH of the solution in the reactor. A forth PI controller, LIC-101, controls the level of the solution inside the reactor by changing the speed of P-104, the pump at the exit of the reactor. The level is strongly coupled with the inlet but its control loop is fast enough to not suffer much performance degradation from this effect.

The pH is going to be controlled through two techniques by using the computer UC-101, which consists of commodity hardware (x86-64 processor, 4 GB DDR2 memory, on-board graphics). The flow rate is measured by positive displacement sensors of model *Signet 2507-2V* (FE-101, FE-102 and FE-103) connected to transmitters of model *Signet 8550-1* (FIT-101, FIT-102) and *Signet 9010 Intelek-Pro* (FIT-103). The transmitters convert pulse signals from the sensors into a 4-20 mA signal that is then transmitted to the PLC, which is described in Section 4.2. A picture of both the sensors and the transmitter can be seen on Figure 17c.

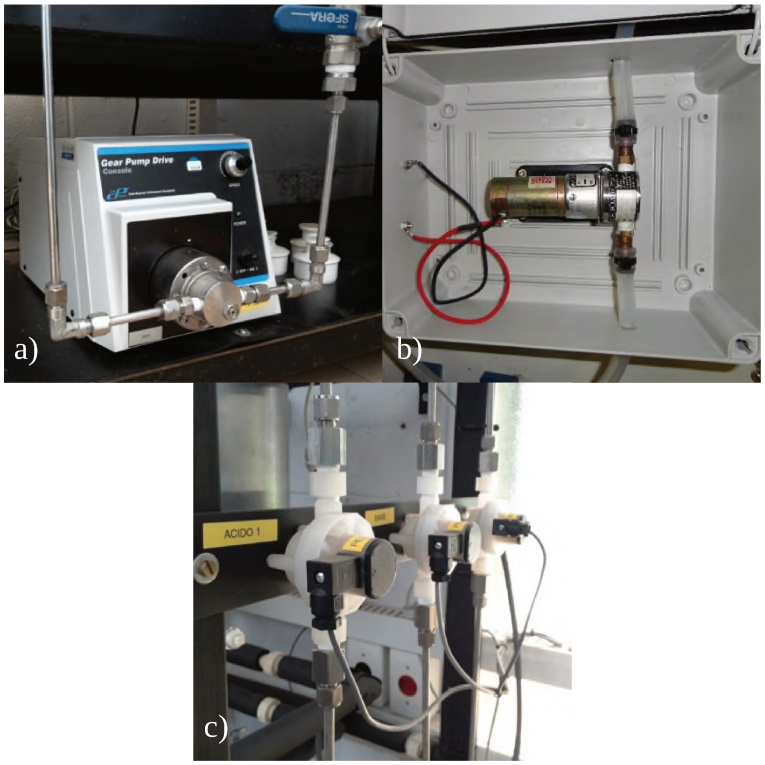

Figure 17 – a) Coleparmer pump. b) Micropump pump. (VAZ DA COSTA, 2014) c) Signet flow rate sensors

All streams are fed to a 6 L cylindrical reactor whose dimensions are shown on Table 1.

Table 1 – Reactor dimensions (VAZ DA COSTA, 2014).

Variable	Description	Value
h	Height	30 cm
A_b	Base area	201.06 cm^2
V	Volume	6032 cm^3

A sensor that is immersed in the solution measures the pH, and it is connected to a pH indicator and transmitter (AIT-101) model *M-300* from *Mettler Toledo*. The pH sensor is an industrial electrode whose model is *Inpro (3250/ISG/D/325)*, as shown on Figure 18a, which has a long rod probe (325 mm of length) for better access to the solution in the reactor.

The liquid level (LIT-101) in the reactor is measured with an absolute pressure sensor from *Smar*, model *LD-301*, which is located at the reactor base and can be seen on Figure 18b.

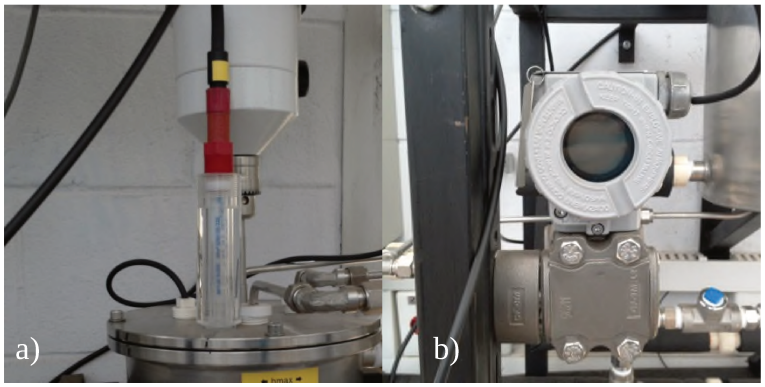

Figure 18 – a) pH sensor. b) Reactor liquid level indicator and transmitter.

The reactor outlet stream is continuously stored during the experiment in a 200 L tank in case any further treatment is needed before releasing it as waste. The liquid level of the reactor is controlled by a pump at the reactor outlet, model RZR-500 shown on Figure 19a, whose rotation is controlled by a frequency converter from *Danfoss* model *VLT 2800*. A positive displacement sensor (FE-104, *Signet GF-2502*, Figure 19b) is used to measure the flow rate and is connected to a *Signet 9010* transmitter.

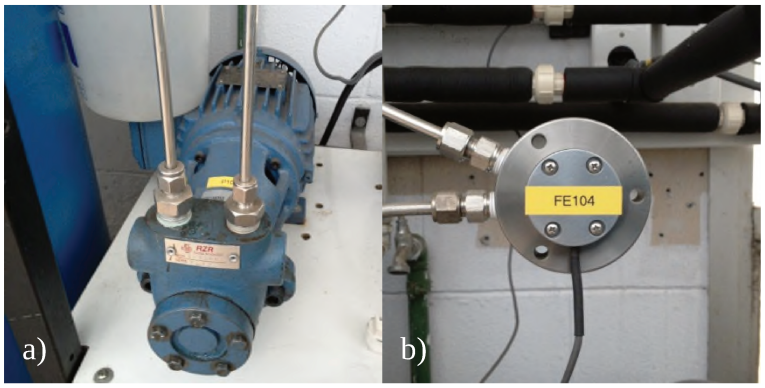

Figure 19 – a) Reactor outlet pump. b) Flow rate sensor.

4.2. Control system

The programmable logic controller (PLC) used to control the system is from *B&R*, model *CP-X20-1485* which has a 400 MHz *Celeron* processor. All the programming, configuration and maintenance was done using the software *Automation Studio 3.9*, whose interface is shown on Figure 20.

The PLC was programmed with structured text as programming language.

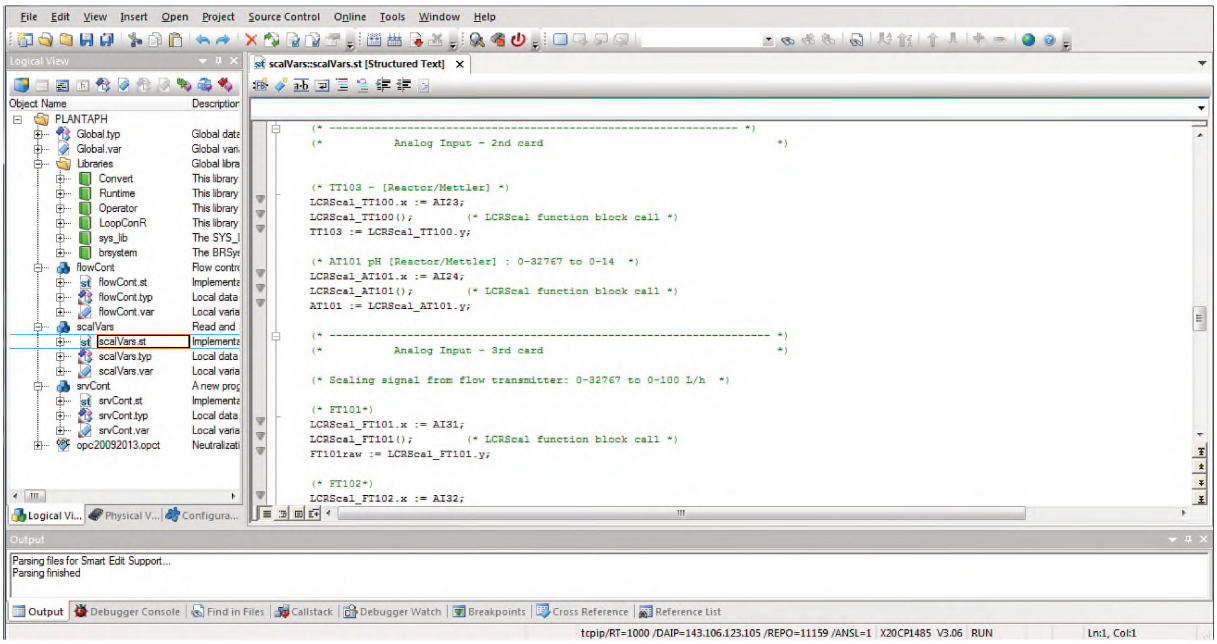

Figure 20 – Automation Studio 3.9 graphical user interface (VAZ DA COSTA, 2014).

A total of five control loops are used to properly control the plant. One loop controls the level of liquid in the reactor (LIC-101) by changing the speed of the pump P-104 at the outlet of the reactor, three control the flow rate of each stream that flows into the reactor (FIC-101, FIC-102, FIC-103) and the fifth (UC-101, the computer acting as a PI controller or an extremum-seeking controller) changes the reference of the base and/or acid flow rate

controllers or the speed of the pumps P-101 and/or P-102 to control the pH. The use of the speed as controlled variable was desired to allow the extremum-seeking algorithm to account for any additional non-linearity between the pump speed and the flow rate automatically.

The controllers FIC-101 and FIC-102 are disabled when UC-101 is used to directly manipulate the speed of the pumps P-101 and/or P-102. With exception of the last controller, all of them are PI controllers with fixed parameters and are run by the PLC. The last controller is run by a computer connected to the PLC.

The computer UC-101 can be set to six different modes and a summary of all its operating modes can be seen in Table 2.

Table 2 – Summary of all control loops and strategies.

Tag	Controlled variable	Manipulated variable	Type
FIC-101	Acid flow rate	P-101 speed	PI
FIC-102	Base flow rate	P-102 speed	PI
FIC-103	Buffer flow rate	P-103 speed	PI
LIC-101	Liquid level	P-104 speed	PI
UC-101 (Mode 1)	pH	FIC-101 or FIC-102 set-point	PI
UC-101 (Mode 2)	pH	FIC-101 or FIC-102 set-point	PI with ES tuning
UC-101 (Mode 3)	pH	FIC-101 or FIC-102 set-point	SISO ES control
UC-101 (Mode 4)	pH	FIC-101 and FIC-102 set-point	MISO ES control
UC-101 (Mode 5)	pH	P-101 or P-102 speed	SISO ES control
UC-101 (Mode 6)	pH	P-101 and P-102 speed	MISO ES control

On the first mode UC-101 acts as a simple PI controller to control the pH by independently changing the set-point of either FIC-101 or FIC-102, not both at the same time. If FIC-101 is the manipulated variable, the controller does not take in account the value set for FIC-102, and vice versa.

On the second mode the computer automatically tunes the PI controller used on the first mode by running a series of experiments on the reactor. These experiments are run offline and will be explained on Section 4.5.1.

The third mode uses extremum-seeking to control the pH by changing the reference of FIC-101 or FIC-102, and just like the first mode if the reference of FIC-101 is the manipulated variable the value of the reference of FIC-102 is not used to calculate the desired value of the reference of FIC-101.

Extremum-seeking is used on the fourth mode to manipulate both FIC-101 and FIC-102 references at the same time to control the pH.

Modes five and six are equivalent to mode number three and mode number four, respectively, but, instead of manipulating the reference of FIC-101 and FIC-102, the speed of the pumps P-101 and P-102 are manipulated to control the pH.

4.3. Graphical user interface and data interface

A cross-platform graphical user interface (GUI) and communication system was developed to ease the communication between the PLC, the user, and the control software, and to reduce the dependency on proprietary software. As a side effect it allowed not only the strengthening the security of the control computer but also the possibility of the use of an internet connection to keep everything updated and allow remote access.

The GUI was developed using Python and Qt, a programming language and an application framework, respectively. Both were regarded as having enough features to develop the GUI and the data interface to the PLC. The GUI running on Kubuntu 17.04, a Linux distribution, can be seen on Figure 21.

The PLC used in the plant uses OPC-DA, previously known as Object linking and embedding for Process Control Data Access, developed by the OPC Foundation as the protocol of communication.

This standard uses a technology developed by Microsoft to communicate with the PLC, thus one Windows XP (or newer) computer must be used as there is no easy way to communicate directly from any non-Windows OS using this standard. Also, there are great security implications due to the security changes needed to make this protocol work.

To workaround this problem an emulated computer, a virtual machine (VM), with Windows XP installed was used to communicate with the PLC, and a free light-weight server was installed on this VM to allow any type of computer to access the PLC data and manipulate its variables.

This free and open-source server modified by Safady (2017) allows any OPC variable to be easily read and written using an application programming interface (API) based on representational state transfer (REST), a way of providing interoperability between computer systems. A sample Python script to read and show the data can be seen on Frame 1.

SciLab is used to process the signal sent by the PLC and apply extremum-seeking or act as a PI controller. The communication with the PLC can be analogously achieved, but the integrated programming language, Tool Command Language (Tcl), can be used rather than Python.

```
import json, requests
payload = {"method": "read", "params": ["TAG1", "TAG2"]}
response = requests.get("http://IP:PORT/", payload, timeout=1)
values = json.loads(response.content)

# print TAG1 and TAG2
print(values(0)(1))
print(values(1)(1))
```

Frame 1 – Using the REST API to read and display data from the server (SAFADY, 2017).

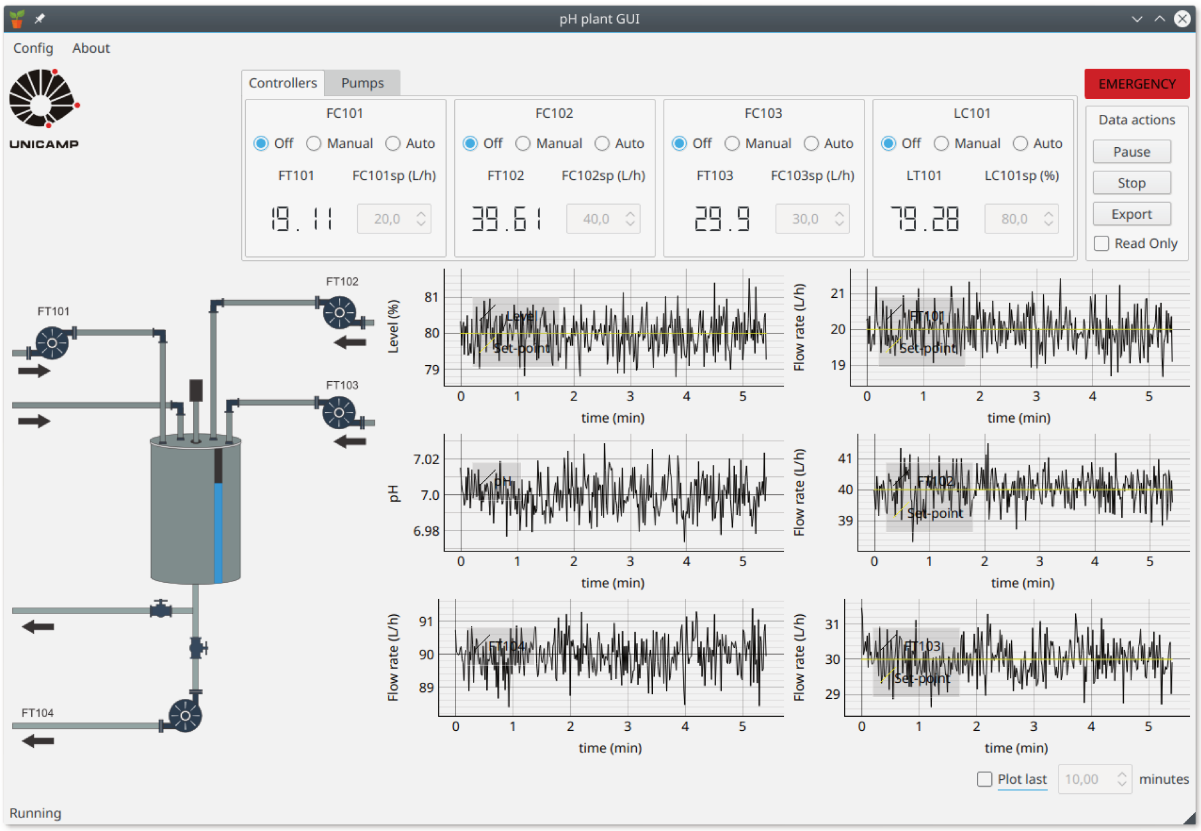

Figure 21 – New GUI developed for the neutralization plant in debug mode.

4.4. Proportional Integral controllers

One of the proposed control schemes is the use of a PI controller to control the pH of the system through UC-101. A derivative controller was not used to avoid the effects of noise on the measurements.

The PI controller is tuned using extremum-seeking. Equation 19 shows the transfer function of the PI controller that is used on this project.

As the flow rate signal from FE-101, FE-102, FE-103 and FE-104 can be noisy, they should be filtered before any control is attempted. The internal filter of FE-103 and FE-104 could not be disabled, but the internal filters of FE-101 and FE-102 were disabled and the filtering was done by the PLC.

The flow rate signal FE-101 and FE-102 were filtered according to Equation 21 before sending them to the slave flow rate controllers FIC-101 and FIC-102, respectively. This is the same filter suggested by Vaz da Costa (2014).

$$G(z) = \frac{1}{10}\frac{z^{10}-1}{z^{10}-z^{9}} = \frac{1}{10}\frac{1-z^{-10}}{1-z^{-1}} \tag{21}$$

While the PI controller that controls the pH is going to be tuned with the use of extremum-seeking, all other controllers have their parameters set according to Table 3. The parameters used for the flow controllers are the same used by Vaz da Costa (2014) for the same plant, while the parameters for the level controller were set aiming fast system response and low control effort.

Table 3 – PI controllers parameters (VAZ DA COSTA, 2014).

Controller	K_c	T_i **(s)**
FIC-101	0.25	1.00
FIC-102	0.25	1.00
FIC-103	0.25	1.00
LIC-101	15.0	13.3

4.5. Extremum-seeking

One of the extremum-seeking approaches used in this study is based on perturbations in the parameters of a PI controller, called “PI controller tuning”. Another approach is based on perturbations in the flow rate of the base and/or acid inlets, called “extremum-seeking control”. The second set of perturbations were achieved by manipulating the reference of flow rate controllers or simply by modifying the speed of rotation of pumps. Both approaches aims to approximate a set cost function to its minimum.

Therefore, two different extremum-seeking strategies are proposed, the first strategy is based on properly tuning a PI controller using extremum-seeking, and the second strategy tracks the reference and tries to directly control the pH by adjusting the flow rate of one or more inlets. Both are implemented through the computer UC-101.

4.5.1. PI controller tuning

This scheme tunes the two parameters of a PI controller (K_c and T_i) using an extremum-seeking algorithm. It varies the parameters of the PI controller to minimize the value of Equation 18, shown in Section 3.2.2.1, which is the mean integrated square error (MISE) cost function. (KILLINGSWORTH; KRSTIC, 2006)

However, unlike the proposal by Killingsworth and Krstic (2006), the full transient response is considered to find an optimum where the set-point is quickly reached while reducing overshoot. Therefore the actual cost function gets simplified to Equation 22, which is the integrated square error (ISE) cost function.

$$J(\theta) \triangleq \int_{t_0}^{T} (e(t,\theta))^2 dt \qquad (22)$$

Where $e(t,\theta)$ is the difference between the reference $r(t)$ and the output signal $y(t)$ of the closed control loop, T is the final time of the experiment and t_0 is approximately the time of the first peak of the initial transient response. Also $\theta \triangleq [K_c, T_i]^T$ is the parameters of the PI controller whose transfer function in Laplace domain can be seen on Equation 19 on Section 3.2.2.1.

A tuning strategy of the controller can be seen on Figure 22. Figure 23 shows the extremum-seeking algorithm strategy, both based on the proposal of Killingsworth and Krstic (2006). In place of the two-degree of freedom controller where the derivative action acts only over $y(t)$ and not over $r(t)$, shown on Figure 13 on Section 3.2.2.1, a simple PI controller will be used and tuned.

The scheme on Figure 22 is similar to the one shown on on Figure 13, but instead of the two-degree of freedom controller a simple PI controller is used. Function G represents the model or transfer function of the plant, PI the PI controller, $r(t)$ the reference signal, $u(t)$ the control signal, and $y(t)$ the output of the plant. The integral represents the calculation of the cost function and θ the parameters of the controller, such as the gain K_c and the integral time T_i.

Figure 23 is based on the diagram shown on Figure 12. h_f is the constant of the high-pass filter, γ_i represents the integration factor, $\hat{K}_c$ the estimated proportional parameter and $\hat{T}_i$ the estimated reset time. The algorithm generates the gradient by changing the parameters of the controller each iteration.

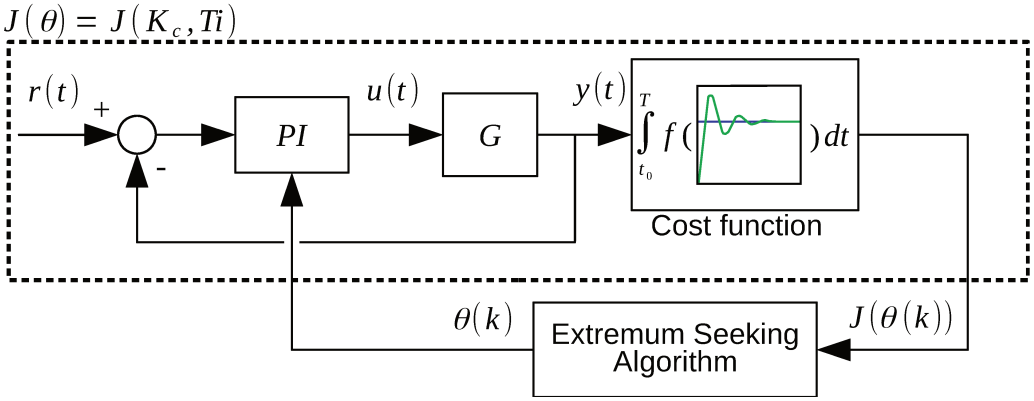

Figure 22 – ES PI tuning scheme based on the proposal by Killingsworth and Krstic (2006).

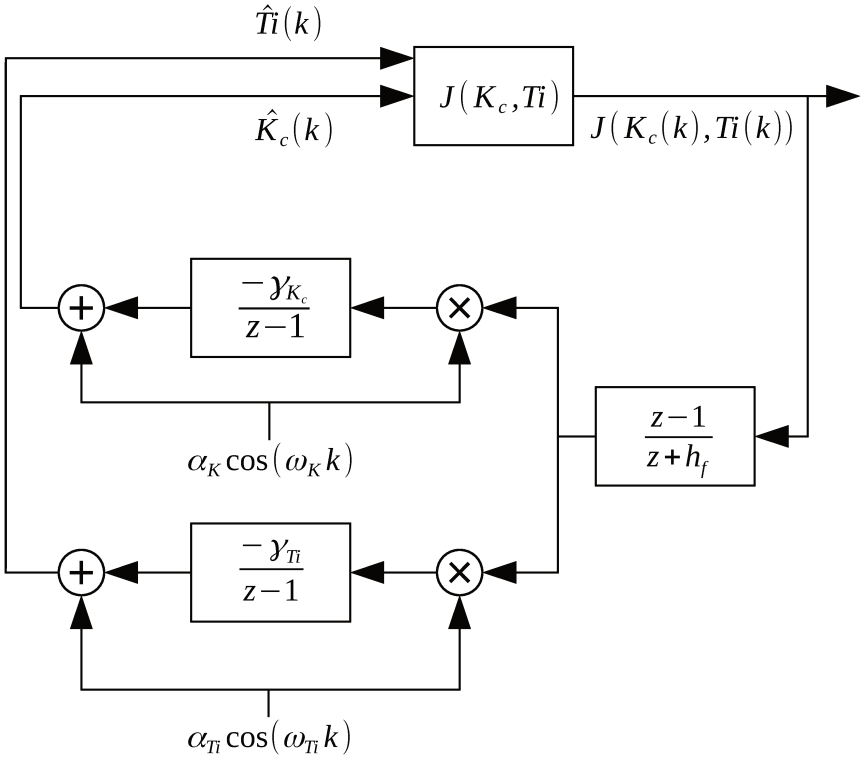

Figure 23 – Discrete ES scheme based on the proposal by Killingsworth and Krstic (2006).

A step-response experiment is carried out for each iteration of the parameters estimation by the extremum-seeking algorithm. Each experiment is carried out by first flushing the current content of the reactor and filling it to 60% of its capacity with the effluent to be treated with the pH control loop disabled. Afterwards the pH control loop is enabled and its pH set point is set to 7. The system is allowed to run for 50 minutes, and the cost function shown on Equation 18 is calculated and used to compute the future values of K_c and T_i for the next experiment where the whole process is repeated.

This process is carried out until the values of the cost function, K_c, and T_i oscillates over a stable average value.

4.5.2. Extremum-seeking control

This strategy directly modifies the flow rate of base, acid, or both inlet streams of the system to reach the desired pH. The discrete version of extremum-seeking, as shown on Figure 12 on Section 3.2.1, was evaluated with one or two inlets as control variables, a SISO or MISO system. The cost function used can be seen on Equation 23, where only the error is considered.

$$J(\theta) \triangleq \frac{1}{h_D}[e(t,\theta)]^2 \tag{23}$$

The element $e(t,\theta)=y(t,\theta)-r(t)$ is the error and h_D represents the sampling time in seconds.

This equation has only one minimum, but as the acid and base flow rates are correlated there are various ratios of base:acid that are able to solve this equation when the system is operating as MISO. This effect was not taken in consideration when choosing the cost function, but will be shown.

The diagram shown on Figure 24 is similar to the one shown on Figure 12. h_f is the constant of the high-pass filter, γ_i represents the integration factor, $\hat{q}_a$ the estimated inlet of acid and $\hat{q}_b$ the estimated inlet of base. Two high-pass filters are shown on the diagram, one for each manipulated variable, however both filters were tuned with the same cut-off frequency, the lowest dither frequency. The algorithm estimates the gradient by changing the flow rate of the acid and base inlets.

The flow rate can be modified by manipulating the flow rate set-point of the flow controllers FIC-101 and FIC-102, or by directly manipulating the speed of the pumps P-101 and P-102. Both are tested as there is no need to trust the flow rate sensor when changing the speed of the pump, and the algorithm should take into account any non-linearity automatically.

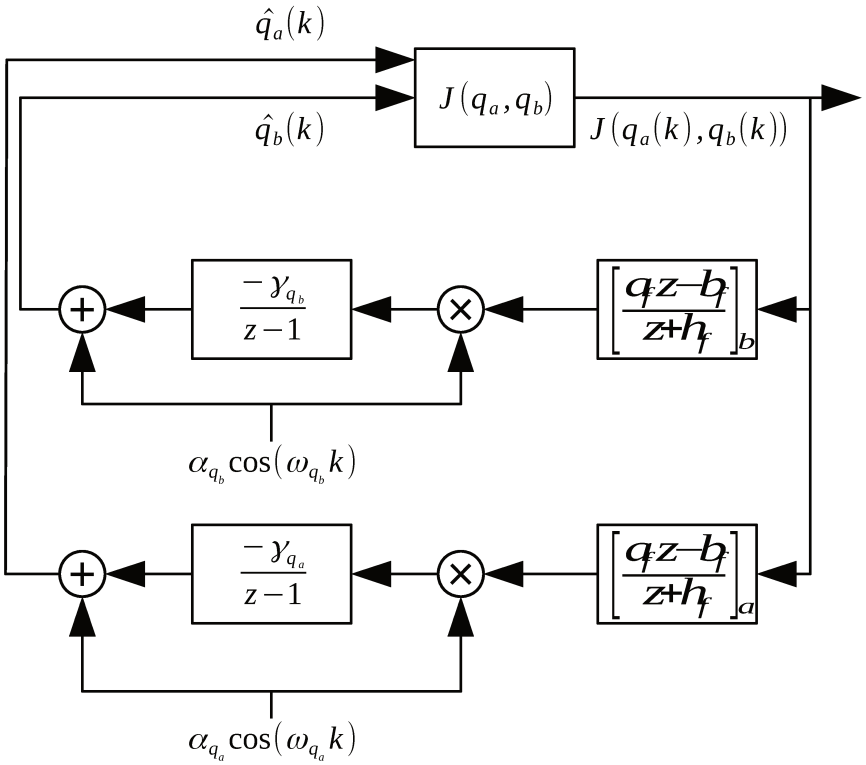

Figure 24 – Discrete Extremum-Seeking Control algorithm based on the proposal by Killingsworth and Krstic (2006).

4.6. ES parameters design

The technique depends on detecting a change on a measure state or output of the system, and thus its parameters must be set to be able to generate a detectable disturbance. This section exposes how the extremum seeking parameters were designed in this work.

In general, the convergence speed is proportional to the perturbation period (ω_i), the perturbation amplitude (α_i), and the integration factor (γ_i). (ZHANG; ORDÓÑEZ, 2012)

Section 4.6.1 shows rules of thumb to help the design of ES parameters, while the sections following explain in greater detail each step shown.

4.6.1. Extremum-seeking parameters design rules of thumb

- Find the time scale of the system to set the perturbation period, the perturbation amplitude, and the sampling time.
  - The period should be found through an open-loop perturbation, such as a step test. It is recommended to be two to three times the system time constant. (MU et al., 2016)

- The period and sampling time must be large enough so the system has time to react to changes in the input. Longer sampling times may help slow systems, especially when filtering.
- Larger perturbations generate larger signals, possibly facilitating its detection. Should be set equivalent to the amplitude of the step test used to find the system time constant.
- Large amplitudes with long periods might increase the convergence rate to the optimum, but may also create an undesired offset at the optimum. Usually one might need to compromise on the amplitude for the period, or vice-versa.

- Find the high-pass filter to best filter the signal DC-component.
  - It should have low delay, and low phase shift on the operating frequency, which can be verified on a Bode plot.
  - A first order IIR (Infinite Impulse Response) filter is recommended for its low-delay and strong DC-removal properties.
- Correct modulator phase shift
  - The phase-lead or phase-lag of the filter and the system must be accounted when modulating (multiplying) the signal with the perturbation.
  - The phase-lag of the system can be estimated with a periodic perturbation over the operating point, which is recommended to have the same frequency as the perturbation. If this is not possible a phase-locked loop (PLL) can also be used to estimate it in real-time.
- Find the integration factor.
  - On the discrete form it also acts as a weak low-pass filter, but a low-pass filter might still be desired when there is strong noise or harmonics even though it can change the dynamics of the system.
  - Higher values will increase the speed of the algorithm, but may also cause instabilities. Should preferably be small.

4.6.2. System time scale, perturbation frequency, amplitude, and sampling time

The system time scale greatly influences the parameters used on the extremum-seeking algorithm, and they must be correctly set to allow proper search of the extremum of the cost function. The perturbation frequency must be low enough and the amplitude high enough to cause a change in the system, which is then used to estimate the gradient of the cost

function. However, low frequencies may be mistakenly removed by the high-pass filter used by the algorithm or be too slow causing large steady-state oscillations. Large amplitudes may destabilize the system and also cause large steady-state offset.

These problems can be in part workaround with the proposed discrete extremum-seeking strategy, as longer sampling times can be used to give the system more time to show any change, but might slow down the convergence to the extremum. In this work a step-response test is used to evaluate the time scale of the system.

4.6.2.1. Step-response test

The same technique used to evaluate a first-order system time constant was used to estimate the time scale of the system, where it is the time the system takes to change by 63.2% (SEELER, 2014) from its current state to its new steady-state value after a step change on its set-point. While dead time can affect the measurement, it was considered sufficiently small compared to the system time constant.

As a pH control system is non-linear, the calculated time constant of a 15% change of the pump speed may be different than a change of 30% or even 10%. However, the time constant is used only as an estimate of the time scale, thus this procedure is only needed if there is no prior knowledge of the system behaviour. The test was carried out over the main operating condition, which for this work is pH equal to 7.

4.6.2.2. Dither period and amplitude

The dither period was set from two to three times the estimated time constant (MU et al., 2016), and its frequency, in radians, was calculated according the Equation 24 and Equation 25. Its amplitude was set to overcome the measurement noise at the dither frequency, but up to a certain extend to reduce the steady-state error. The same amplitude used on the step-response test was considered sufficient and therefore selected.

$$f_c = \frac{2\pi}{T_p} \tag{24}$$

$$f_d = f_c T_s \tag{25}$$

Where f_c is the continuous frequency, T_p the period in seconds, f_d the normalized frequency, and T_s the sampling time in seconds.

According to the Nyquist theorem the largest possible normalized frequency is 0.5 Hz, or π rad/s, lest undesired aliasing occur. The sampling time and the perturbation frequency must agree to not generate a normalized frequency higher than that value.

Two sources of dither were used to modulate and demodulate the signal, one before and another after the integrator, as shown on Figure 24. The phase angle of the first was selected to compensate for the delay caused by the plant dynamics and the high-pass filter (MU et al., 2016), while the second dither has zero phase correction.

While the high-pass filter phase distortion can be easily calculated through frequency response analysis or a Bode plot, the phase shift caused by the neutralization system was estimated through a senoidal perturbation over the operating condition on the manipulated variable and its frequency was set to the selected dither frequency. This operation was done only over the operating condition, which is shown on Section 5, and the phase angle was estimated by normalizing and comparing the phase of the input and output waves. If no information of the phase angle for the modulation is available a phase-locked loop (PLL) can be used to estimate it in real-time, as that information can increase the convergence rate of the algorithm to the optimum (PEIXOTO et al., 2015).

4.6.3. High-pass filter design

The high-pass filter must be especially designed to not allow most if not all DC-component to pass-through, albeit allowing the frequency of the disturbance to pass. It should have little to no delay and preferably low phase distortion. The filter used in this work for the ESC scheme has the form shown on Equation 26, a recursive filter with a single sample of delay, where a_f , b_f , and h_f are design parameters. While this type of filter can cause phase distortion, the low delay is more desired.

$$H(z)=\frac{a_f z-b_f}{z-h_f} \tag{26}$$

The filter was designed so that the cutoff frequency, which is defined as the frequency where the amplitude is reduced by 3 dB, is the same as the perturbation frequency. This was accomplished with the Infinite Impulse Response (IIR) design tool available in SciLab and the Butterworth design technique.

A bode plot of a filter with that form with can be seen on Figure 25 to exemplify. Its cut-off frequency is approximately 0.4 rad, thus frequencies lower than that will get attenuated and higher frequencies may get boosted or remain constant, as shown on the magnitude plot.

The filter phase-lead is approximately 60 degrees at the cut-off frequency, but changes for other frequencies. This factor should be accounted for when modulating the signal afterwards.

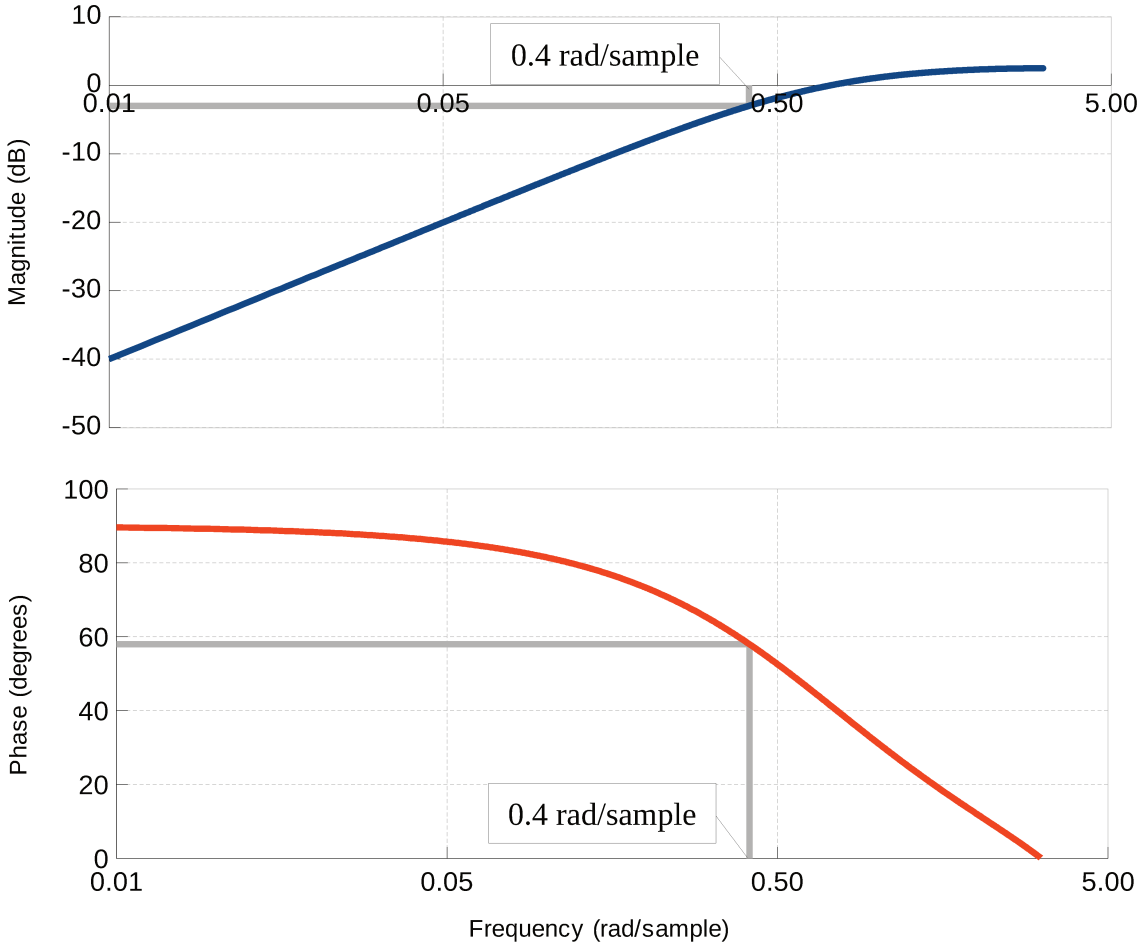

Figure 25 – Example bode plot of the filter from Equation 26 with a_f = 1, b_f = 1, and h_f = 0.5.

4.7. Low-pass filter and integrator design

While extremum-seeking usually employs a low-pass filter before the integrator, it should be used only when needed, such as removing harmonics or the measurement noise, as its action can change the dynamics of the algorithm.

As this work uses the discrete form of extremum-seeking, its integrator also acts as a weak low-pass filter and it was deemed enough. Thus, no additional low-pass filter was used. The equation for the integrator can be seen on Equation 27.

$$I(z)=\frac{-\gamma_i}{z-1} \tag{27}$$

Where γ_i is the integration factor.

Lower integration factors are recommended lest destabilize the system, but higher values may increase the convergence rate to the extremum. A Bode plot of an integrator can be seen on Figure 26.

Lower frequencies being boosted can be clearly seen on the magnitude plot, along with the phase-lead shown on the phase plot.

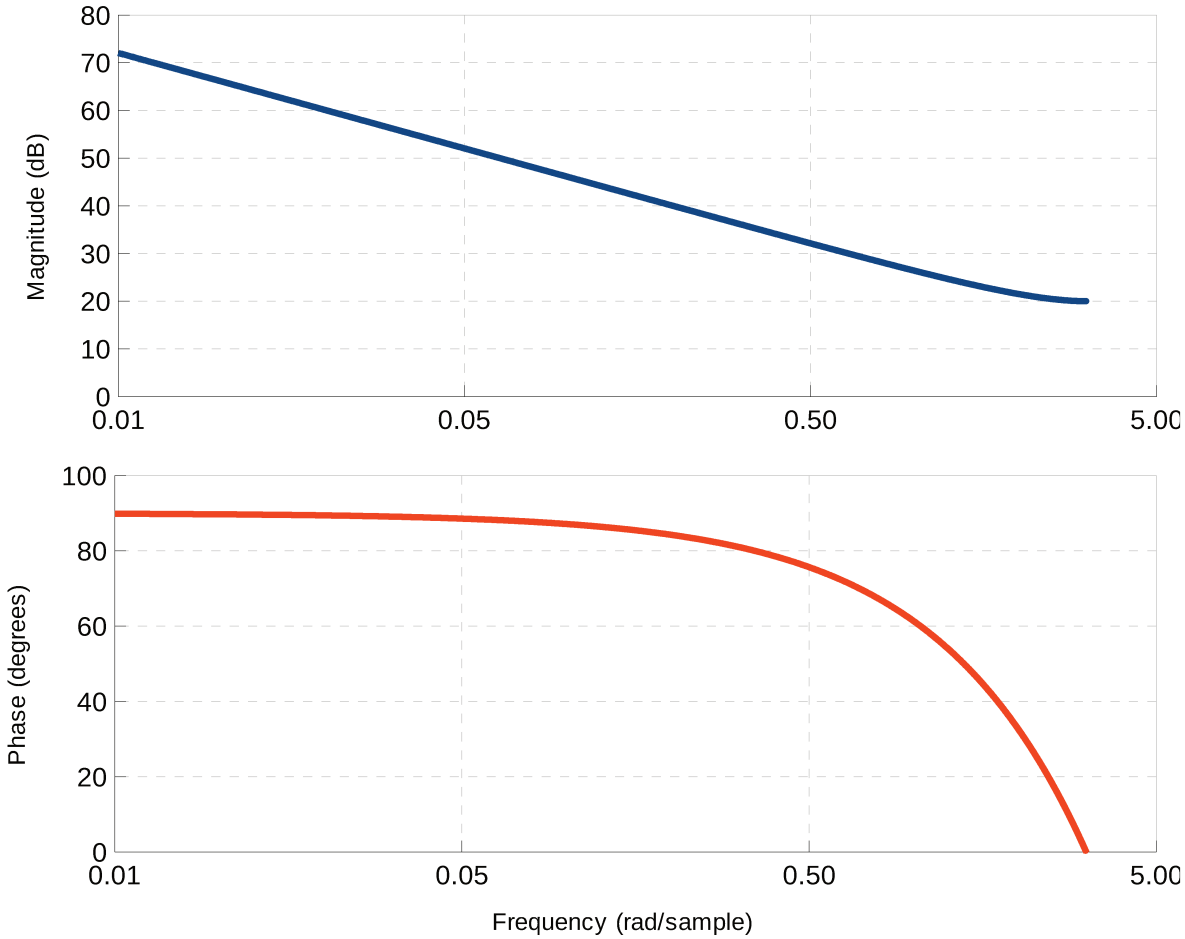

Figure 26 – Example integrator bode plot with $\gamma_i = 20$.

4.7.1. Anti-windup

An anti-windup technique must also be implemented to account for manipulated variable saturation, which can be done thorough a penalty function (TAN; LI; MAREELS, 2012) or through a non-linearity such as a saturation. A saturation was used in this work as it is easier to use and another design constant is not necessary.

The saturation was designed such as the peak of the perturbation does not go over the upper saturation limit, and its valley can not go under the lower saturation limit. No hysteresis was considered and its implementation was done by adding upper and lower limits to the state of the filter.

4.8. pH model for the simulation

The pH model used to simulate the system is based on the Reaction Invariant Model used by Vaz da Costa (2014) with some modifications, which in turn was based on the work by Gustafsson and Waller (1992). Three equilibrium reactions are considered to happen in the reactor, and they are shown in Equation 28, Equation 29, and Equation 30. The equilibrium relationship between each compound is shown in Equation 31, Equation 32 and Equation 33.

The model uses three differential equations to model the system. Equation 34, Equation 35 and Equation 36 are used to calculate the evolution of the invariants over time while Equation 37 is used to calculate the pH. The values of the parameters were calculated by Vaz da Costa (2014) and can be seen on Table 4.

The outlet flow rate is controlled via a PI controller to maintain a constant liquid level inside the reactor.

$$H_2O \Leftrightarrow H^+ + OH^- \tag{28}$$

$$H_2CO_3 \Leftrightarrow H^+ + HCO_3^- \tag{29}$$

$$HCO_3^{1-} \Leftrightarrow H^+ + CO_3^{2-} \tag{30}$$

$$K_w = [H^+][OH^{-1}] \tag{31}$$

$$K_{a1} = \frac{[HCO_3^-][H^+]}{[H_2CO_3]} \tag{32}$$

$$K_{a2} = \frac{[CO_3^{2-}][H^+]}{[HCO_3^{-1}]} \tag{33}$$

$$A_b h \frac{dw_a}{dt} = q_1(w_{a1} - w_a) + q_2(w_{a2} - w_a) + q_3(w_{a3} - w_a) \tag{34}$$

$$A_b h \frac{dw_b}{dt} = q_1(w_{b1} - w_b) + q_2(w_{b2} - w_b) + q_3(w_{b3} - w_b) \tag{35}$$

$$A_b \frac{dh}{dt} = q_1 + q_2 + q_3 - q \tag{36}$$

$$w_a + 10^{pH-14} - 10^{-pH} + w_b \times \frac{1 + 2 \times 10^{pH - pK_{a2}}}{1 + 10^{pK_{a1} - pH} + 10^{pH - pK_{a2}}} = 0 \tag{37}$$

The initial condition of the invariants was calculated by randomly guessing them and letting the system enter steady state while q_1, q_2, q_3, and q are fixed.

Table 4 – Parameters of the dynamic model for the simulation (VAZ DA COSTA, 2014).

Variable	Description	Value
w_{a1}	Invariant a for stream 1	+0.0034 M
w_{a2}	Invariant a for stream 2	-0.0057 M
w_{a3}	Invariant a for stream 3	-0.03 M
w_{b1}	Invariant b for stream 1	0 M
w_{b2}	Invariant b for stream 2	0 M
w_{b3}	Invariant b for stream 3	+0.03 M
pK_{a1}	Antilog of K_{a1} at 25 °C	6.35
pK_{a2}	Antilog of K_{a2} at 25 °C	10.33
A_b	Reactor base area	201.6 cm^2

4.9. Simulation software

The simulations of the neutralization system and the calculations of the extremum-seeking method were made with SciLab through XCos and custom calculations blocks.

The XCos diagram on Figure 27 shows a sample of the simulation procedure to simulate the extremum-seeking control of the system, while Figure 28 shows a sample of the procedure to simulate the PI tuning using extremum-seeking.

On Figure 27, region one shows the blocks responsible for emulating the pump behaviour according to its speed, region two shows the blocks used to simulate the neutralization reactor itself, region three shows the blocks used to control the level of the simulated reactor, and regions four and five indicate the blocks responsible for calculating the cost function and the extremum-seeking algorithm, respectively. The pump emulation in region one is done by experimentally getting a plot of the flow-rate according to the speed of the pump in the actual pilot plant. The level controller of the simulated reactor uses the same proportional and integral parameters as used in the pilot plant. The simulation in region three is done as shown on Section 4.8, and the MISO system can be create analogously.

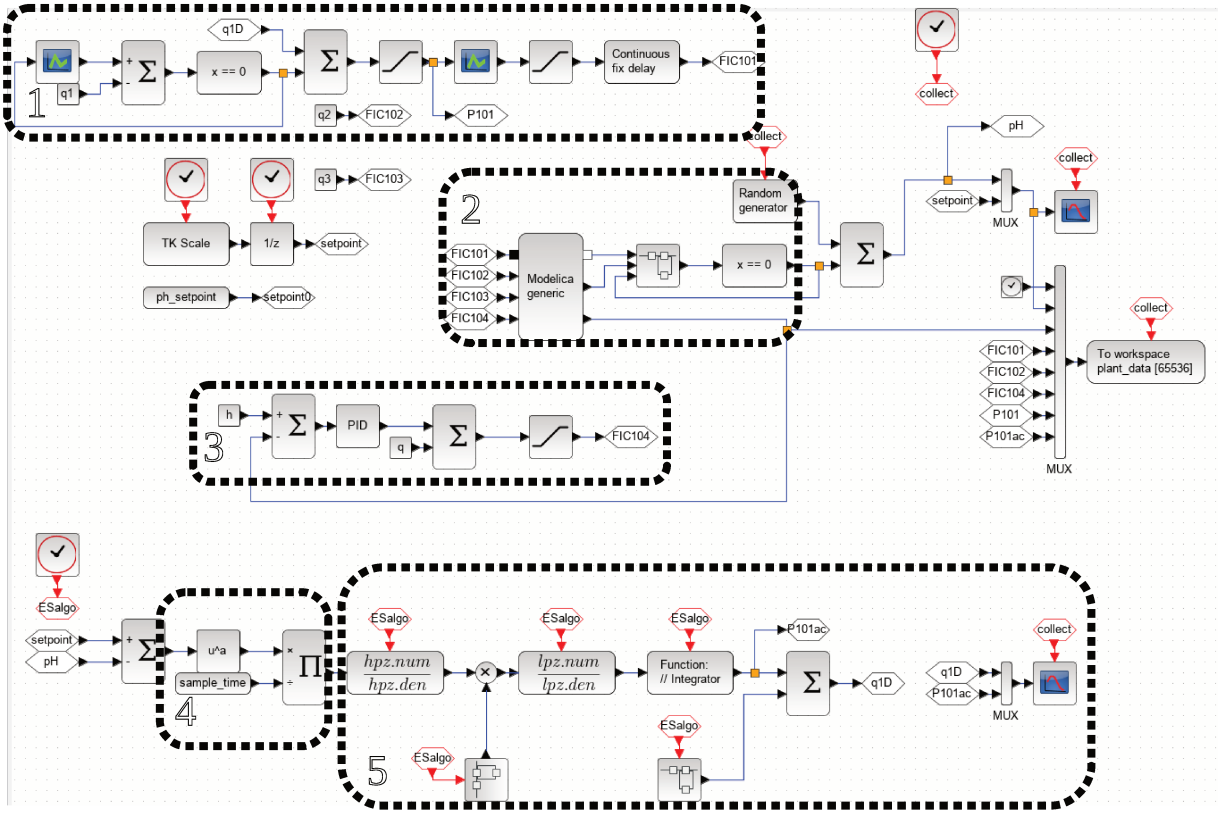

Figure 27 – XCos diagram of an extremum-seeking control simulated SISO system

On Figure 28, regions number one, two and three indicate the simulation of the system, and regions number four, five and six indicate the extremum-seeking PI tuning algorithm.

Region number one, just like the ES control simulation, shows an approximation of the flow-rate behaviour on changes of the pump speed, region number two shows the block of the simulation of the reactor with the model shown on Section 4.8, and region number three shows the PI controller that controls the level of the reactor. Region number four shows the PI controller that is tuned using extremum-seeking, and it is responsible to control the pH of the reactor by changing FIC-101 reference or P-101 speed. Region number five shows the cost function calculation and region number six shows the extremum-seeking calculation to estimate the parameters used in the PI controller from box four.

Between each experiment the manipulated variable is set to zero while to buffer flow rate is maintained until the pH stabilizes. This was done to ease the development of the simulation algorithm.

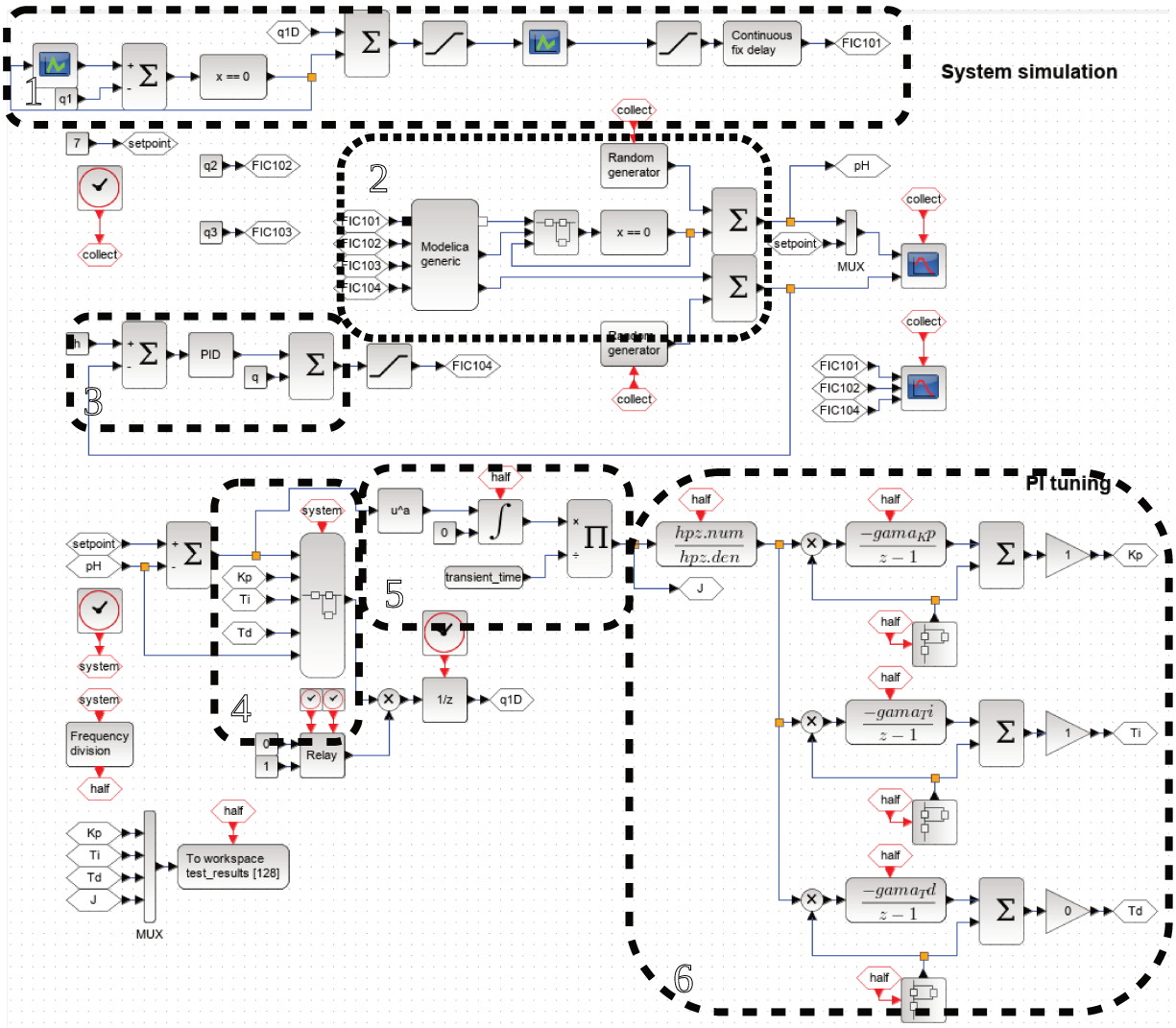

Figure 28 – ES PI tuning simulation diagram

4.10. Control software

While the flow rate PI controllers were programmed into the PLC, SciLab along with its XCos module were used to interact with the PLC and calculate everything related to the extremum-seeking algorithm.

This allowed fast prototyping and evolution of the algorithm due to the simplicity of changing any calculation step in XCos and the use of its pre-programmed blocks, and also removed the need to use *Automation Studio* to re-program the PLC on every change.

As XCos is also used to simulate the system its is possible to sync the numerical method used on both the simulation and the experiment, reducing the probability of error.

XCos is more of a tool for simulating systems instead of controlling systems in real-time, but this was achieved by setting the “simulation” time-scale to the value of one. One script file is used to set all parameters to their desired values, such as initial flow rates,

turn on all controllers in the PLC, start the XCos module to control the system, and then disable all controllers on error or when the operation finishes.

The XCos diagram for an extremum-seeking control experimental procedure can be seen on Figure 29. Region number one shows blocks that read data from the PLC, region number two shows blocks that write data to the PLC, region number three shows blocks that calculate the cost function, and region number four shows blocks that calculate the extremum-seeking algorithm. The system can be easily toggled between SISO and MISO by switching the switches in region five.

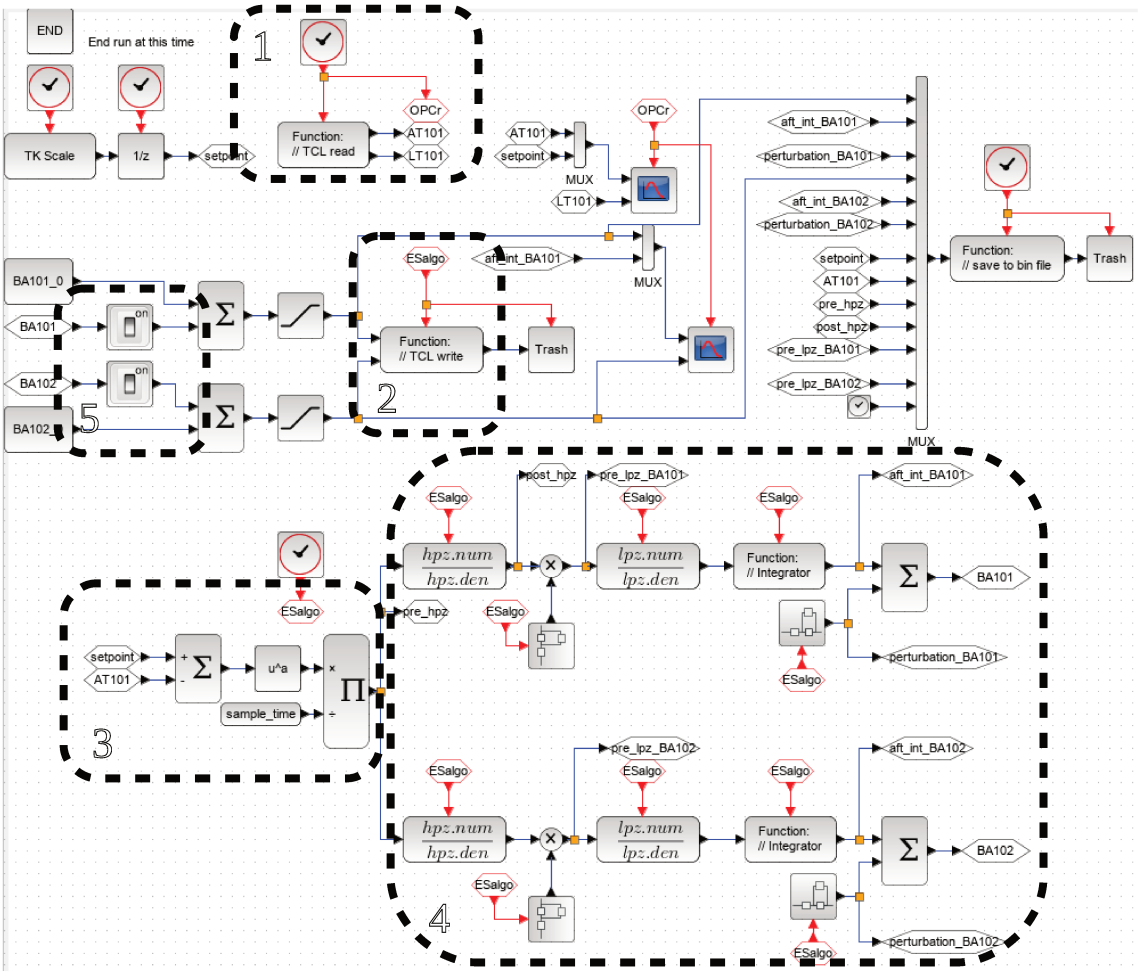

Figure 29 – SISO and MISO extremum-seeking control SciLab diagram used to control the pH.

5. Results

This section will show both simulation and experimental results of the PI tuning and the extremum-seeking control. Extremum-seeking control is evaluated with the algorithm manipulating the acid inlet flow rate, the base inlet flow rate, and both at the same time. Only the base or the acid inlet flow rate is manipulated by the PI controller tuned using extremum-seeking.

5.1. Simulations

This section focuses on the simulation of the closed-loop reactional system using the proposed control system. PI tuning using extremum-seeking is shown first, followed by extremum-seeking control.

5.1.1. PI Tuning using Extremum-seeking

The entire transient response of each iteration, or closed-loop experiment, was used to calculate the cost function, as it was desirable to not only increase the settling speed but also to reduce any oscillations that may arise after the first transient response.

The extremum-seeking parameters used to estimate the optimum are shown on Table 5 and were chosen through trial-and-error, while the plant operating points were: base and buffer flow rates set to 5 L/h; set-point of the level of the reactor set to 60%.

Table 5 – ES parameters used to optimize the parameters of the PI controller.

h_f (-)	γ_K (h/L)	γ_{T_i} (s)	α_K (-)	α_{T_i} (-)	ω_K (rad)	ω_{T_i} (rad)
0.5	0.02	-0.01	0.1	0.5	0.8π	$0.8^2\pi$

On Figure 30 one can see the simulation result of the evolution of the PI parameters K an T_i after each iteration, where the proportional action mostly oscillates over an average and the integral action reduces after each iteration.

The initial guess values for the proportional constant K was set to 0.1 and for the reset time T_i was set to 0.5 seconds, while the optimized values found by the extremum-seeking algorithm are approximately 23.3 and 294.5 seconds, respectively.

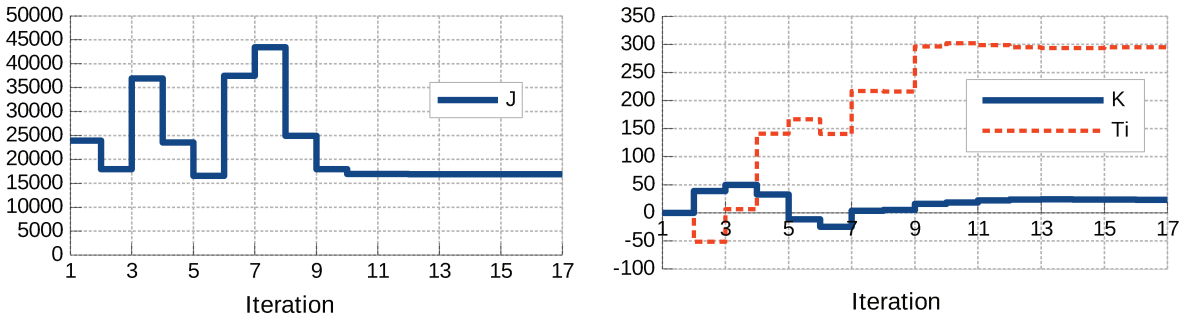

Figure 30 – PI tuning details over each iteration. Left: Evolution of the cost function (Equation 22). Right: Evolution of the calculated PI parameters after each trial..

A comparison of the performance of a PI controller whose parameters are set to the initial guess and the optimized values can be seen on Figure 31.

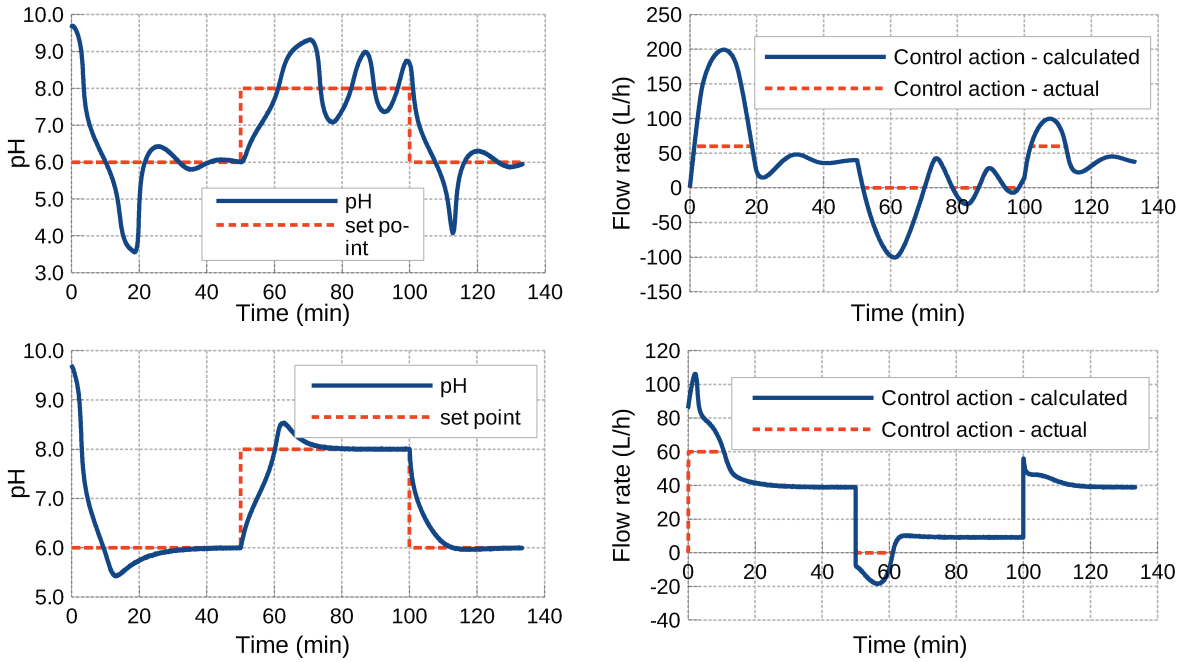

Figure 31 – PI performance before (top) and after (bottom) optimization.

As shown, the resulting overshoot of the optimized values is significantly lower than the initial guess values, the system does not oscillate anymore over the tested operating points, and it also has faster settling time.

While the extremum-seeking algorithm was able to optimize the cost function, during the simulations it was noticed that the ES parameters are very sensitive when using the high-pass filter suggested by Killingsworth and Krstic (2006), while if the high-pass filter is designed with its cut-off as the lower dither frequency the choice of the ES parameters can be more relaxed.

5.1.2. Extremum-seeking Control

In contrast to the previous scheme where extremum-seeking is used to tune a controller, this method uses extremum-seeking to directly control the system as shown in Section 4.5.2.

Extemum-seeking control can be used on both Single-Input Single-Output (SISO) and Multiple-Input Single-Output (MISO) systems. For the SISO scheme the flow rate of the acid or the base inlets are manipulated to control the pH, while on the MISO scheme both are manipulated at the same time. The manipulation happens by changing the speed of the acid or base pumps, be it directly or though a flow rate controller. This section will firstly estimate suitable tuning parameters for the ES control technique, and then show ES control simulations with parameters selected through trial-and-error without prior knowledge of the system behaviour, and another using the parameters found in Section 5.1.2.1 to show how they affect the convergence rate.

5.1.2.1. ES control parameters

The parameters of the algorithm were selected as shown on Section 4.6, and this section shows the step-response test result along with the phase analysis.

A simulated step-test with 20 % change on the speed of the acid pump can be seen in Figure 32, while a simulated step-test with 9 L/h change on the acid flow rate can be seen on Figure 33. The calculated time-constant for 0.02 M is approximately 7.7 minutes (pump speed) and 8.5 minutes (flow rate), while for 0.03 M it is approximately 6.7 minutes (pump speed) and 7.1 minutes (flow rate).

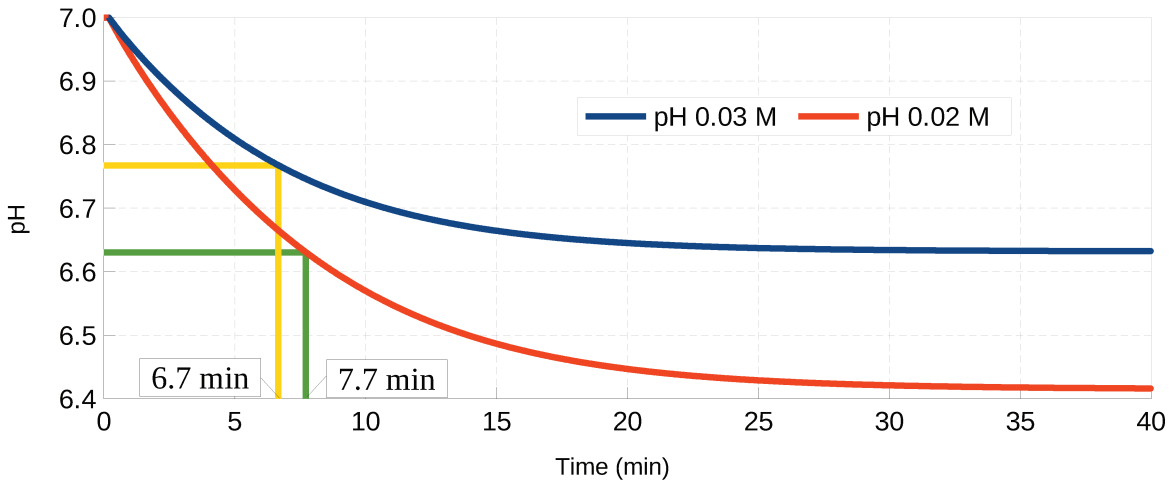

Figure 32 – Step-response test, speed of the acid pump increased from 0% to 20%. Acid pump speed initial conditions: buffer at 0.02 M, 38.7%; buffer at 0.03 M, 45.2%.

The perturbation peak-to-peak amplitude was set to the same magnitude as the step-response test to evaluate the phase lag, and the period was set to 13.3 minutes, close to two times the system time constant.

For both buffer concentrations, and for both manipulated variables (slave PI reference or pump speed) the phase lag was estimated to be 2.8 minutes for the acid inlet and 3.5 minutes for the base inlet. This was tested by setting the system close to its default operating condition (pH equal to seven) through changes on the base or acid inlets, and then injecting dither into the system with the desired frequency on the base or acid inlet.

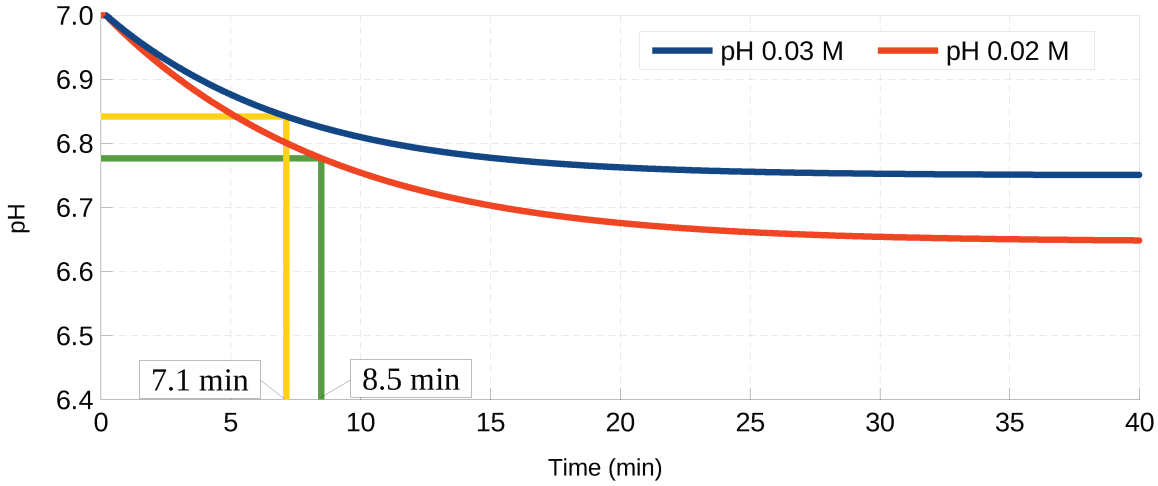

Figure 33 – Step-response test, 9 L/h increase on the acid flow rate. Acid flow rate initial conditions: buffer at 0.02 M, 10.7 L/h; buffer at 0.03 M, 16.1 L/h.

5.1.2.2. Trial-and-error parameters

These parameters were chosen through trial-and-error without prior knowledge of the results obtained with the tuning technique.

5.1.2.2.1 Flow-rate as manipulated variable

The parameters used for this SISO scheme where the acid flow rate is the manipulated variable can be seen on Table 6, while the simulation with those parameters can be seen on Figure 34. The oscillations caused be the inlet can be clearly seen on the pH, including the phase lag, and the “-nd” suffix represents the signal just after the integrator, before adding more dither.

The buffer concentration was set to 0.03 M, the base flow rate was set to zero, the buffer flow rate to 10 L/h, the set-point of the level of the reactor to 60%. No phase-correction was used for the modulation.

While not shown here for short, the results are equivalent with the buffer concentration set to 0.02 M but slower.

Table 6 – Trial-and-error parameters used in the SISO ES control simulation with FIC-101 reference as manipulated variable.

h_f (-)	γ_{q_a} (L/h)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
0.50	55	4.5	4.25	0.25

While the system starts by quickly reaching the neighbourhood of the desired set-point, it takes a long time for it to reach the exact value of the set-point. The derivative of the cost function is near zero close to the set-point as it is based on a quadratic function, thus the control action is also almost zero. If dither with higher amplitude is used the extremum-seeking controller speed will be increased, but brings the risk of overshoot, the manipulated variable saturation limits might be reached reducing the control efficiency, and might even decrease the working life of the pump.

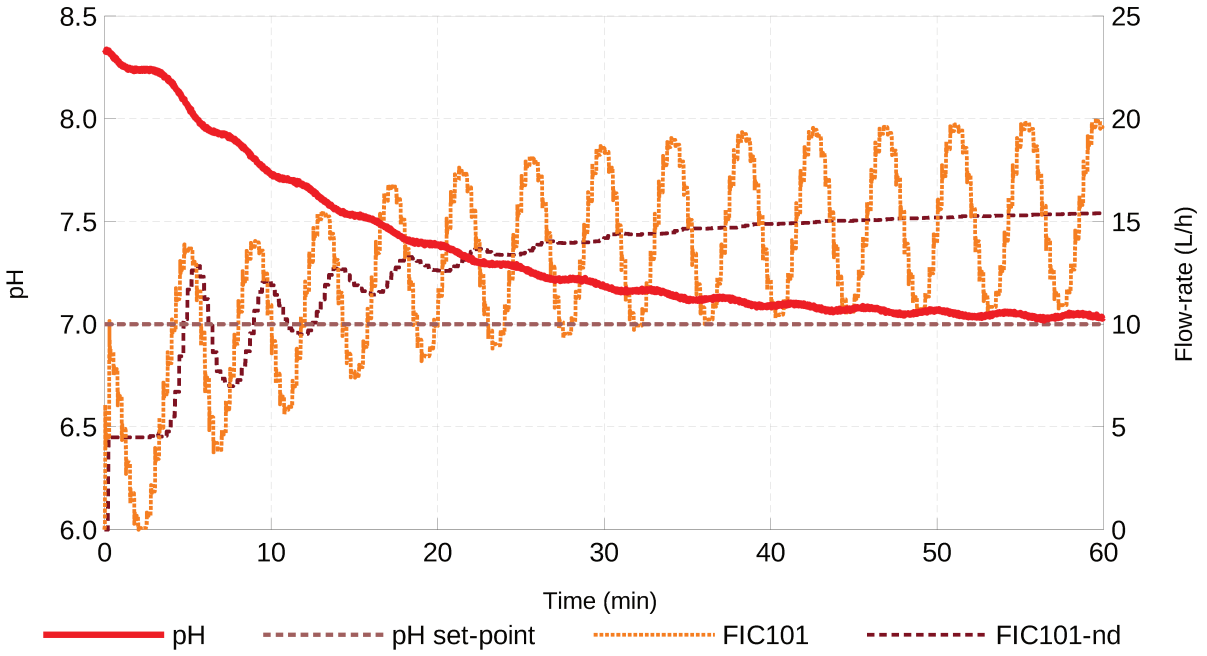

Figure 34 – SISO extremum-seeking control with FIC-101 reference as manipulated variable, trial-and-error parameters.

During the simulations it was noticed that the speed of the set-point change can interfere with the scheme, especially on the MISO scheme. This effect is shown on Figure 35 and Figure 36, where in the first graph the reference is slowly set to the desired value by filtering the set-point change with the transfer function shown on Equation 38, while in the second graph the reference is instantaneously set to the desired value.

$$F(s)=\frac{1}{v\ s+1} \tag{38}$$

In this equation v is equal to 10 seconds.

The slow change causes the algorithm to reach the optimum at a slower pace, while the fast change even causes overshoot. The parameters used for both simulations can be seen on Table 7.

Table 7 – Trial-and-error parameters used in both MISO ES control simulations with FIC-101 and FIC-102 references as manipulated variables.

h_f (-)	γ_{q_a} (L/h)	γ_{q_b} (L/h)	α_{q_a} (-)	α_{q_b} (-)	T_{p_a} (min)	T_{p_b} (min)	T_s (min)
0.50	55	35	4.0	3.5	4.25	3.00	0.25

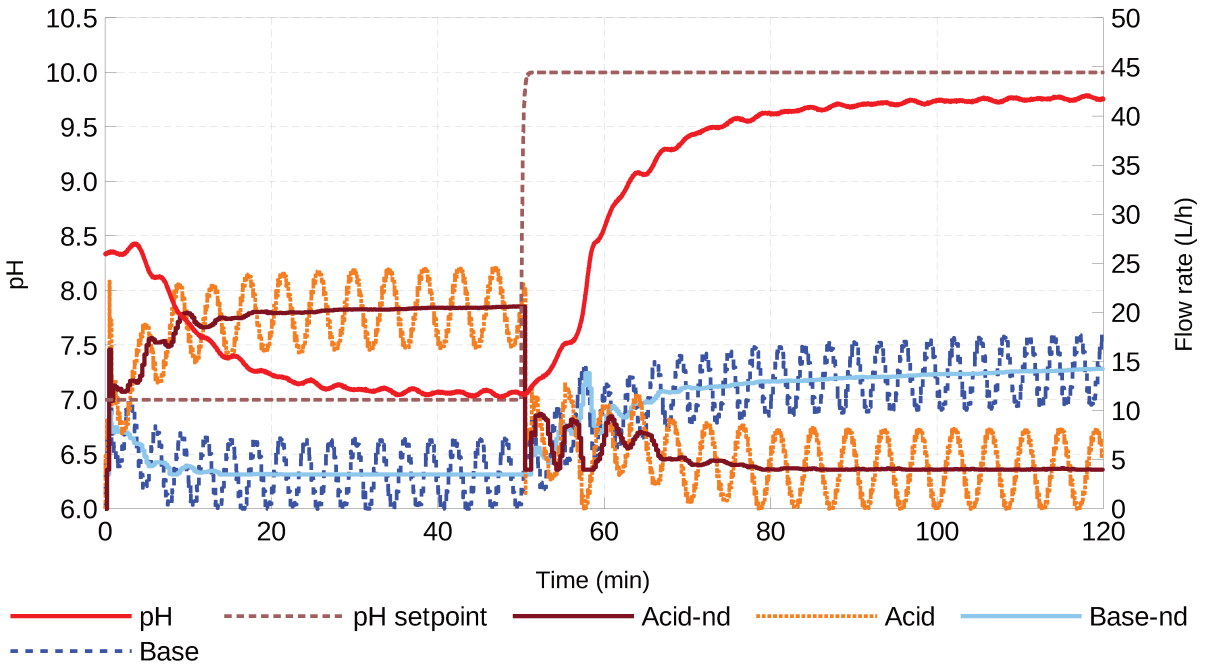

Figure 35 – MISO extremum-seeking control with FIC-101 and FIC-102 references as manipulated variables and slow set-point change, trial-and-error parameters.

The same effect shown on the SISO scheme was noticed on the MISO scheme, the cost function convergence also slows down when close to the extremum (when its value is zero, the minimum), which is expected due to the same reason. The workaround to this effect is the same proposed for the SISO scheme, and has the same consequences.

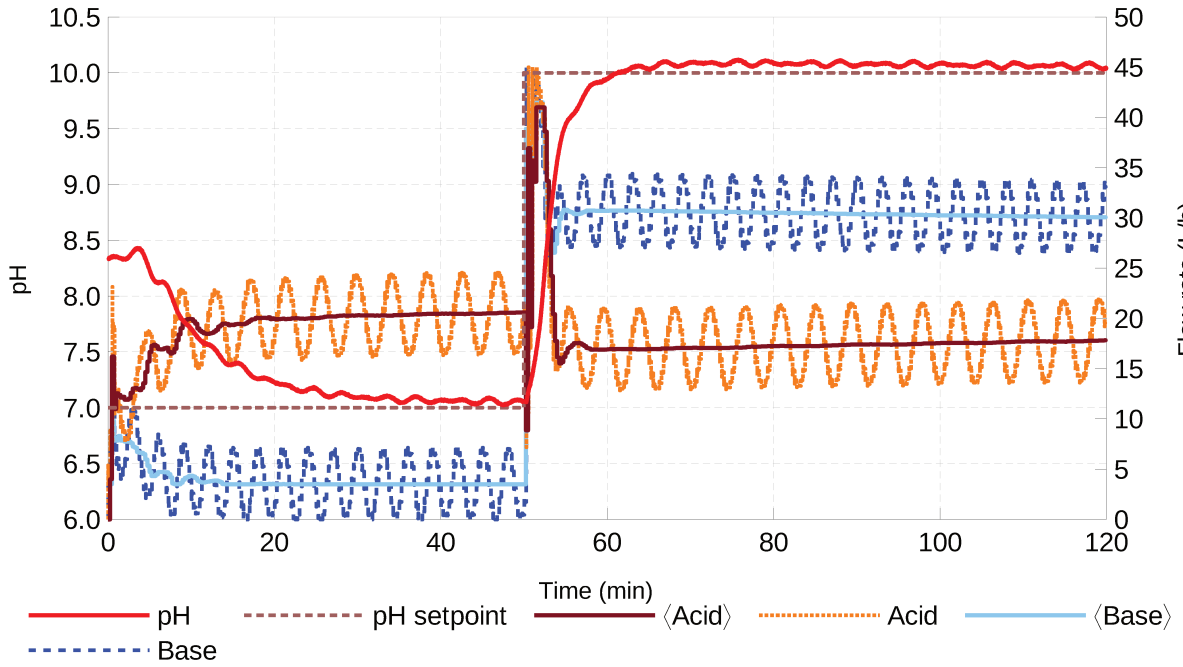

Figure 36 – MISO extremum-seeking control with FIC-101 and FIC-102 references as manipulated variables and instantaneous set-point change, trial-and-error parameters.

5.1.2.2.2 Pump speed as manipulated variable

The parameters used for this SISO scheme where the acid flow rate is the manipulated variable can be seen on Table 8, while the simulation with those parameters can be seen on Figure 37. The parameters are nearly the same as those used when the FIC-101 reference is the manipulated variable, but the perturbation amplitude is doubled. This was done to simplify the tuning as the flow rate is 45 L/h when the pump is at its maximum (100 % speed), and thus almost half in value. The oscillations caused by the inlet can be clearly seen on the pH, including the phase lag.

The buffer concentration was set to 0.03 M, the base flow rate was set to zero, the buffer flow rate to 10 L/h, the set-point of the level of the reactor to 60%. No phase-correction was used for the modulation.

Just like in 5.1.2.2.1, the results are equivalent with the buffer concentration set to 0.02 M but slower, which are not shown here for short. Also the same observations can be made here, the pH stabilizes near the zero of the cost function and the offset can take a long to time ro be removed. The same workaround of this effect can be used.

Table 8 – Trial-and-error parameters used in the SISO ES control simulation with P-101 speed as manipulated variable.

h_f (-)	γ_{q_a} (%)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
0.50	55	9.0	4.25	0.25

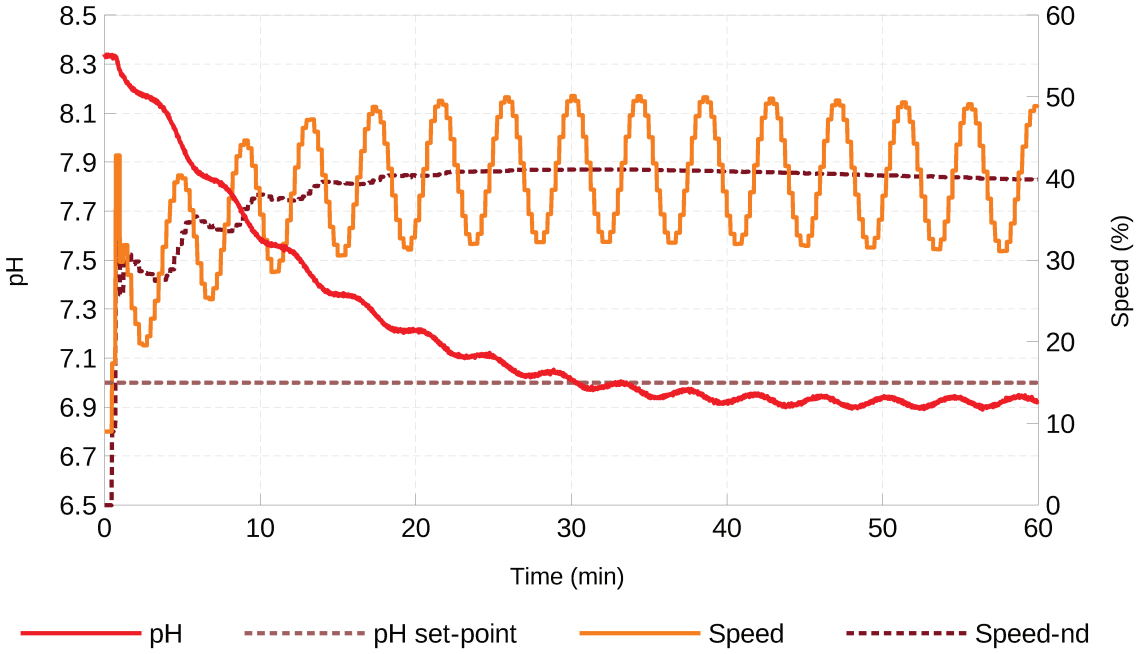

Figure 37 – SISO extremum-seeking control with P-101 speed as manipulated variable, trial-and-error parameters.

In Table 9 the parameters for the MISO scheme can be seen, and its plot can be seen in Figure 38. Only a test with instantaneous set-point is shown.

Table 9 – Trial-and-error parameters used in the MISO ES control simulation with P-101 and P-102 speeds as manipulated variables

h_f (-)	γ_{q_a} (%)	γ_{q_b} (%)	α_{q_a} (-)	α_{q_b} (-)	T_{p_a} (min)	T_{p_b} (min)	T_s (min)
0.50	55	35	8.0	7.0	4.25	3.00	0.25

As can be seen, both plots are similar to those obtained when manipulating the references of FIC-101 and FIC-102. This is expected and desired, as this shows that it is possible to directly manipulate the speed of the pump, simplifying the design of the entire control system, and one of the reason is the no need for PI controllers to control the flow-rate.

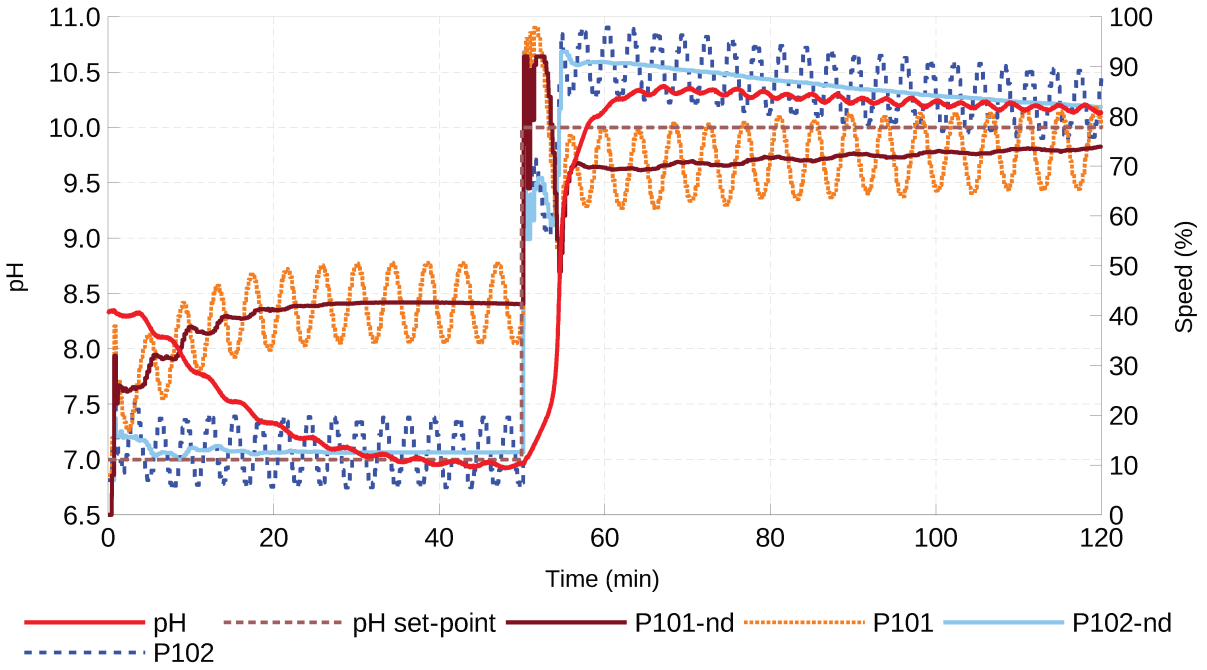

Figure 38 – MISO extremum-seeking control with P-101 and P-102 speeds as manipulated variables, trial-and-error parameters.

5.1.2.3. Designed parameters

Unlike Section 5.1.2.2, all simulations shown will use the parameters designed on Section 5.1.2.1. It will be clear by the end of this section that the convergence rate increases with the use of longer dither periods, as expected, and lower integrating factor can be used. Also, as we suppose there is knowledge of the system behaviour, the phase lag known and considered is not zero, which greatly influences the convergence rate.

5.1.2.3.1 Flow-rate as manipulated variable

The parameters used for this SISO scheme where the acid flow rate is the manipulated variable can be seen on Table 10, while the simulation with those parameters can be seen in Figure 39. The oscillations caused by the inlet can be clearly seen on the pH, including the phase lag. Unlike the simulation with trial-and-error parameters, the steady-state off-set is lower and the system quickly reaches the desired set-point. Unlike Section 5.1.2.2, the high-pass filter was designed with the dither frequency as its cut-off frequency.

The base flow rate was set to zero, the buffer flow rate to 10 L/h, the set-point of the level of the reactor to 60%. Phase-correction was used for the modulation.

Table 10 – Designed parameters used in the SISO ES control simulation with FIC-101 reference as manipulated variable.

Phase lag (min)	γ_{q_a} (L/h)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
2.83	20	5	13.3	0.25

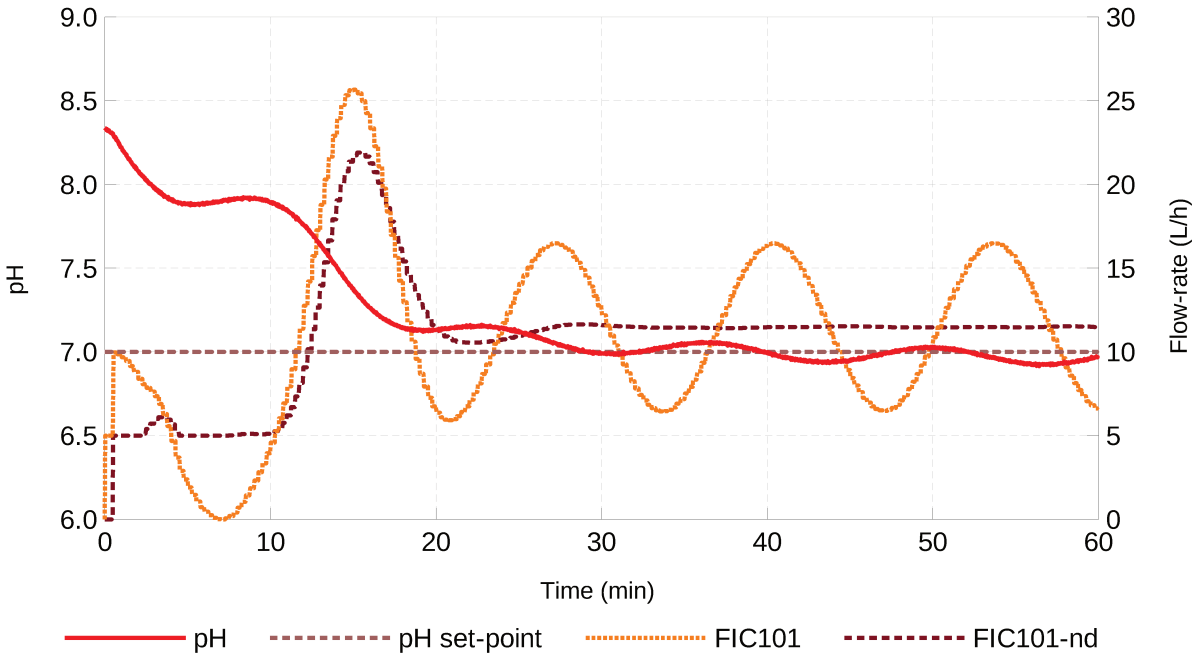

Figure 39 – SISO extremum-seeking control with FIC-101 reference as manipulated variable, designed parameters.

In Table 11 it is possible to see the parameters used for the MISO experiment, with the exception of the phase lag for each manipulated variable. The behaviour of the system changes when changing the pH with acid or base, therefore it is expected for the phase lag caused by each to be different. The estimated phase lag caused by perturbations with the dither frequencies shown in Table 11 is 170 seconds for the acid, and 210 seconds for the base.

A plot of the MISO system can be seen in Figure 40, along with the interaction between both dithers in the system response. The step change in the set-point is made approximately at 130 minutes.

Table 11 – Designed parameters used in the MISO ES control simulation with FIC-101 and FIC-102 references as manipulated variables.

γ_{q_a} (L/h)	γ_{q_b} (L/h)	α_{q_a} (-)	α_{q_b} (-)	T_{p_a} (min)	T_{p_b} (min)	T_s (min)
5.0	2.5	5.0	2.5	13.3	20	0.25

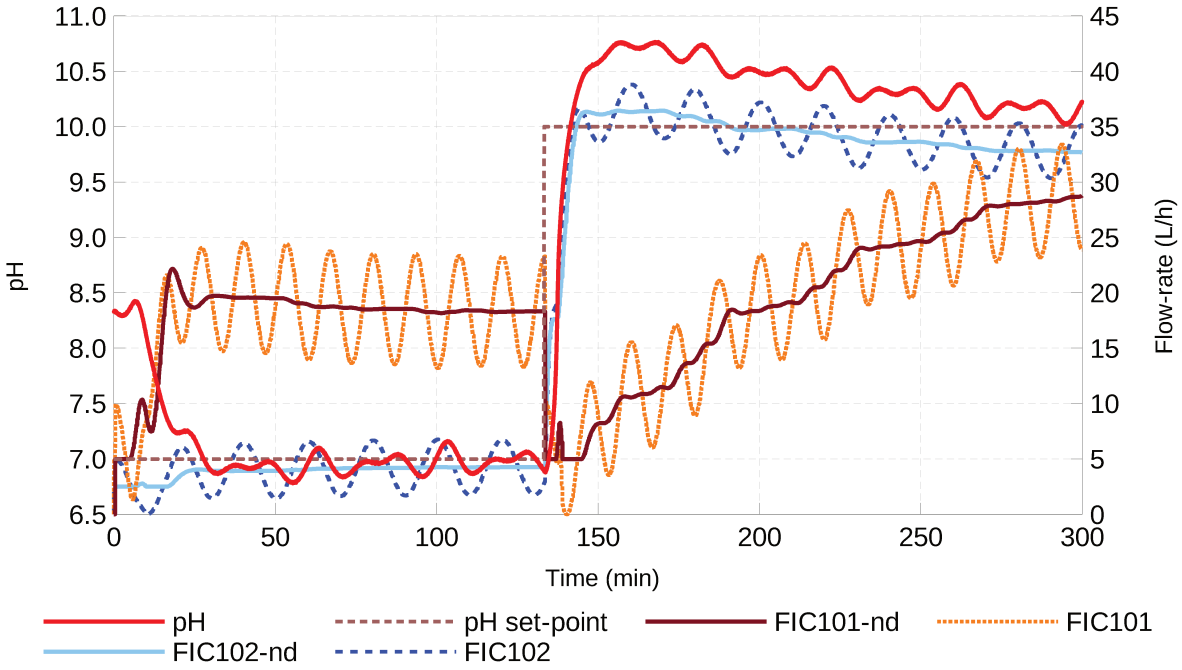

Figure 40 – MISO extremum-seeking control with FIC-101 and FIC-102 references as manipulated variables and instantaneous set-point change, designed parameters.

The system was allowed to run up until 300 minutes to clearly show the behaviour of the system response, and how the algorithm finds the minimum of the cost function by increasing the flow rate of acid instead of reducing the flow rate of base. This is expected as the cost function is designed to only look at the set-point error and does not take in account the quantity of base and acid used.

5.1.2.3.2 Pump speed as manipulated variable

The parameters used for this SISO scheme where the acid flow rate is the manipulated variable can be seen on Table 12, while the simulation with those parameters can be seen in Figure 41. The oscillations caused by the inlet can be clearly seen on the pH, including the phase lag. Unlike the simulation with trial-and-error parameters, the steady-state off-set is lower and the system quickly reaches the desired set-point. Unlike Section 5.1.2.2, the high-pass filter was designed with the dither frequency as its cut-off frequency.

The base flow rate was set to zero, the buffer flow rate to 10 L/h, the set-point of the level of the reactor to 60%. Phase-correction was used for the modulation.

Table 12 – Designed parameters used in the SISO ES control simulation with P-101 speed as manipulated variable.

Phase lag (min)	γ_{q_a} (%)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
2.83	20	10	13.3	0.25

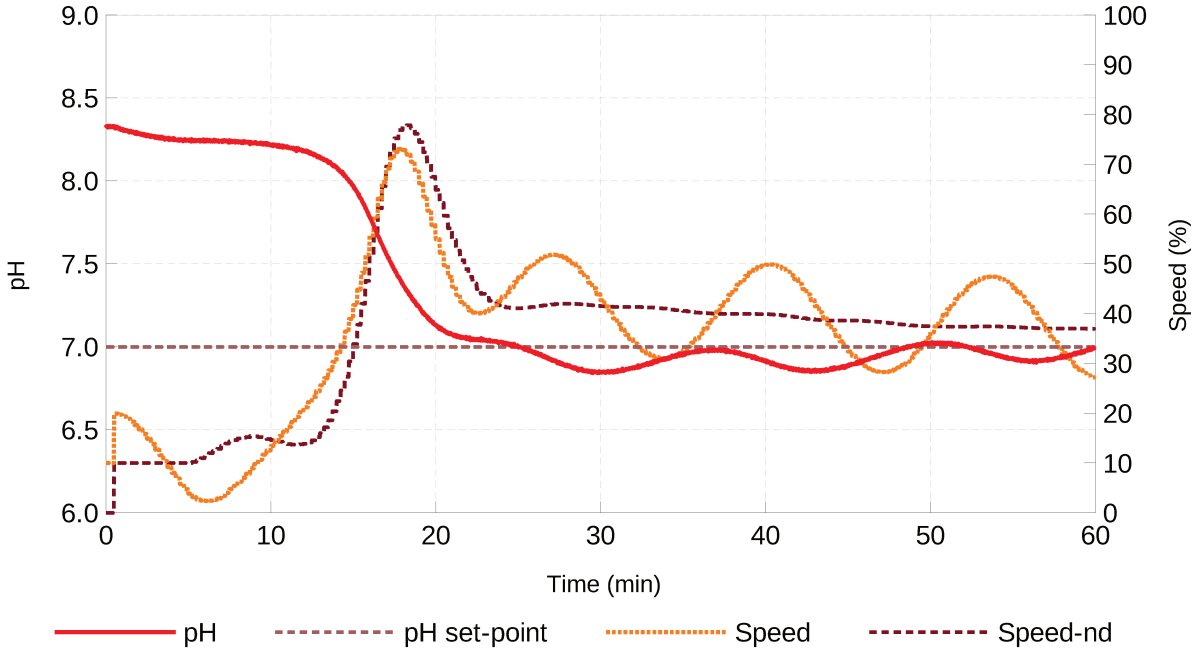

Figure 41 – SISO extremum-seeking control with P-101 speed as manipulated variable, designed parameters.

As can be seen, the system response is similar to the one shown in Figure 39, where the flow rate is the manipulated variable. This shows that it is valid to manipulate the pump speed directly instead of the controller set-point.

In Table 13 it is possible to see the parameters used for the MISO experiment, with the exception of the phase lag for each manipulated variable. Just like when manipulating the reference of the flow rate controller in Section 5.1.2.3.1, the behaviour of the system changes when changing the pH with acid or base, therefore it is expected for the phase lag caused by each to be different. The estimated phase lag caused by perturbations with the dither frequencies shown in Table 13 is 170 seconds for the acid, and 210 seconds for the base.

A plot of the MISO system can be seen in Figure 42, along with the interaction between both dithers in the system response, even though they are harder to see when compared to Figure 40 The step change in the set-point is made approximately at 130 minutes.

Table 13 – Designed parameters used in the MISO ES control simulation with P-101 and P-102 speeds as manipulated variables

γ_{q_a} (%)	γ_{q_b} (%)	α_{q_a} (-)	α_{q_b} (-)	T_{p_a} (min)	T_{p_b} (min)	T_s (min)
5.0	2.5	10	5.0	13.3	20	0.25

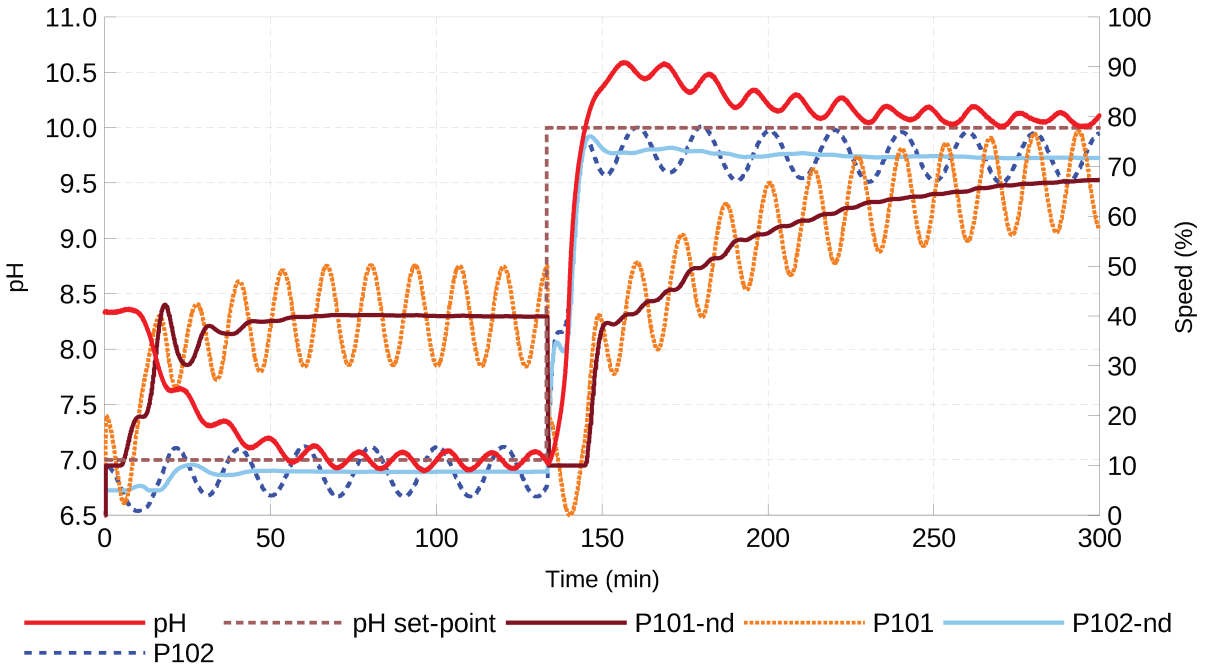

Figure 42 – MISO extremum-seeking control with P-101 and P-102 speeds as manipulated variables, designed parameters.

Just like in Section 5.1.2.3.1, the system was allowed to run up until 300 minutes to clearly show the behaviour of the system response, and how the algorithm finds the minimum of the cost function by increasing the flow rate of acid instead of reducing the flow rate of base.

5.2. Experiments

This section focuses on the experimental results obtained from the neutralization plant with the proposed control system. PI tuning using extremum-seeking is shown first, followed by extremum-seeking control.

5.2.1. PI Tuning using Extremum-seeking

To save reactants this type of extremum-seeking was performed only through simulations, thus no experiment is shown.

5.2.2. Extremum-seeking Control

Just like the simulations shown on Section 5.1.2, both SISO and MISO schemes where evaluated, along with the trial-and-error and designed parameters. For the SISO scheme only the FIC-101 reference or the P-101 speed was evaluated as manipulated variable, while for the MISO scheme the set-point of the controllers FIC-101 and FIC-102 were used as manipulated variables.

5.2.2.1. Trial-and-error parameters

This section shows experiments carried out in the real pH control plant with parameters set through trial-and-error.

5.2.2.1.1 Flow rate as manipulated variable

The experimental parameters for the SISO scheme can be seen in Table 14. Before starting the experiment the reactor was drained and filled to 60% of fullness with the buffered medium. The buffer concentration was set to 0.03 M, the base flow rate was set to zero, the buffer flow rate to 10 L/h, and no phase-correction was used for the modulation.

Table 14 – Trial-and-error parameters used for the experimental SISO ES control with the acid flow rate as manipulated variable through FIC-101.

h_f (-)	γ_{q_a} (L/h)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
0.5	55	4.5	4.25	0.25

The experimental result in Figure 43 is very similar to the result obtained from the simulation shown on Figure 34, even tough the flow rate signal can be really noisy. ES control was able to control the pH of the reactor even though it was slower than the simulation.

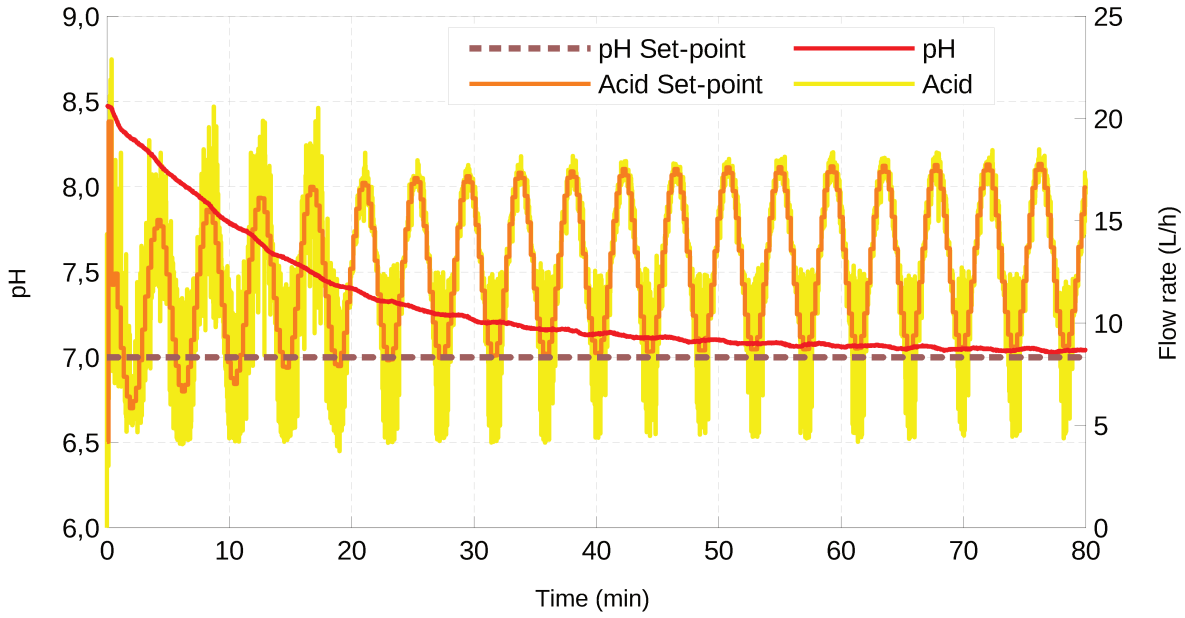

Figure 43 – SISO extremum-seeking control experiment with FIC-101 reference as manipulated variable, trial-and-error parameters.

The MISO experiments were also made with a buffer concentration of 0.03 M. The system was set to three set-points in series, first starting at seven, then to nine and then back to seven. This was done to show that ES control is able to control the pH, but, as the cost function is not optimizing for the flow rate of base and acid, a different minimum is found by the algorithm when setting the pH back to seven. The experimental result can be seen on Figure 44. The parameters used can be seen in Table 15, which are the same used on the MISO simulations shown on Section 5.1.2.2.1.

The MISO ES control strategy was able to control the pH of the reactor, even though it did not have the fastest control action.

Table 15 – Trial-and-error parameters used for the experimental MISO ES control with the acid and base flow rate as manipulated variables through FIC-101 and FIC-102, respectively.

h_f (-)	γ_{q_a} (L/h)	γ_{q_b} (L/h)	α_{q_a} (-)	α_{q_b} (-)	T_{p_a} (min)	T_{p_b} (min)	T_s (min)
0.5	55	35	4.0	3.5	4.25	3.00	0.25

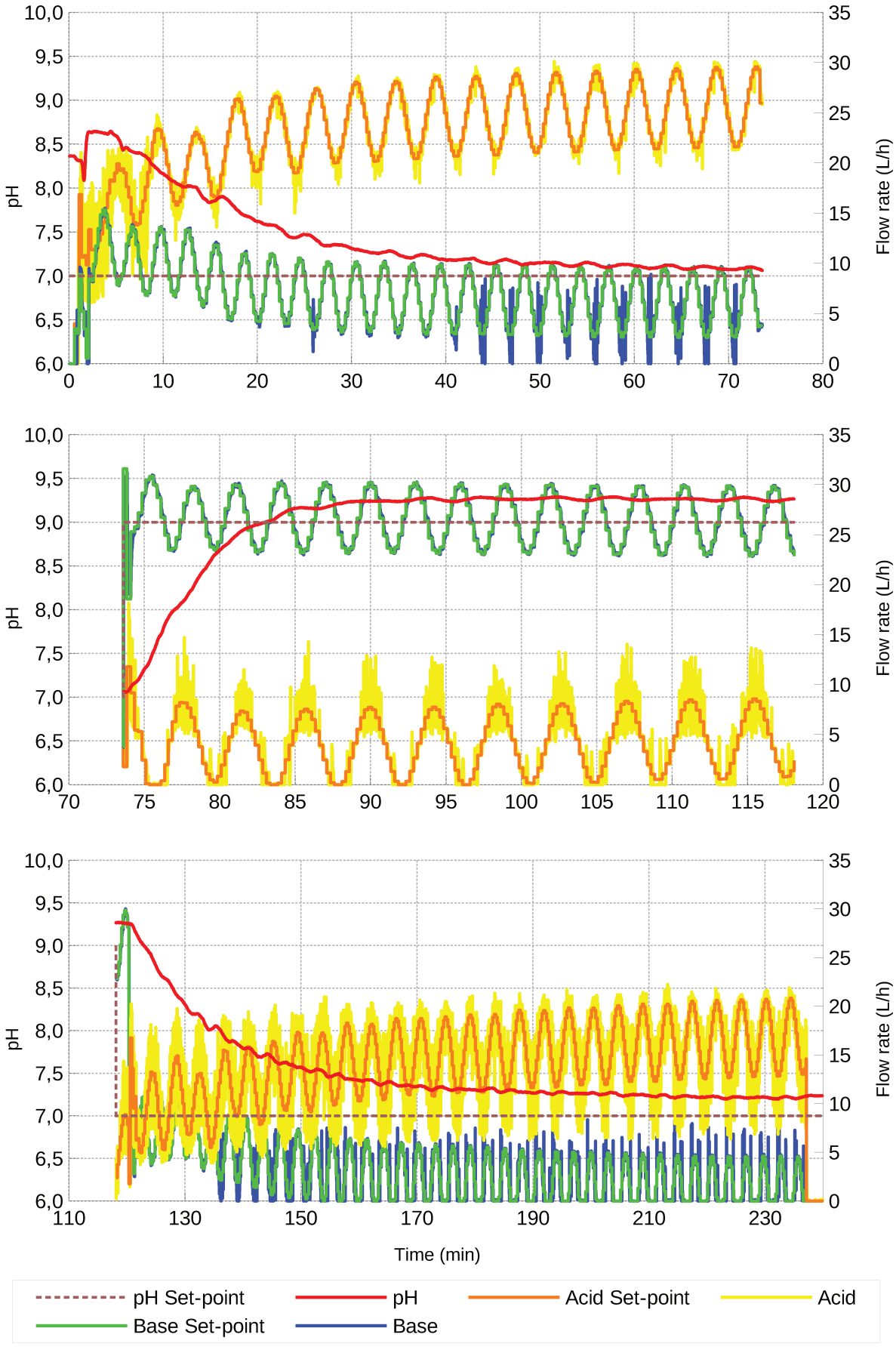

Figure 44 – MISO experiment with different set-points in series.

5.2.2.1.2 Pump speed as manipulated variable

The following experiment was made by manipulating the speed of the pump with a buffer concentration of 0.02 M.

On Figure 45 it is possible to see an ES control SISO experiment with high frequency dither to illustrate what can happen in this case, such as the large control effort. At first the algorithm tries high and low values until it is able to find finally stabilize around an average. After reaching the average, the manipulated variable value stabilizes and floats over the optimum of the cost function. The initial instability can be mitigated with smaller integration factors, but slows the algorithm down.

Table 16 – Trial-and-error parameters with high frequency dither for the ES control SISO experiment where the speed of P-101 is the manipulated variable.

h_f (-)	γ_{q_a} (%)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
0.20	3500	10.0	0.555	0.25

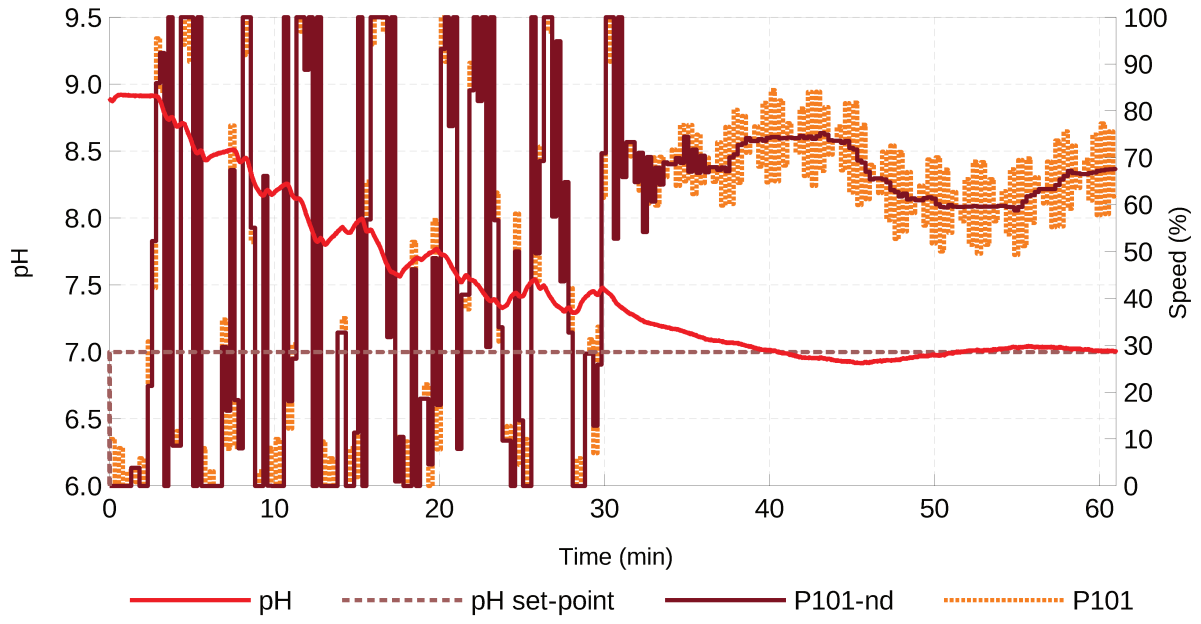

Figure 45 – SISO extremum-seeking control experiment with P-101 speed as manipulated variable and high frequency dither.

By increasing the dither period it is possible to increase both the convergence rate and the stability of the system, as shown in Figure 46. The parameters for this experiment can be seen in Table 17.

Table 17 – Trial-and-error parameters with lower frequency dither for the ES control SISO experiment where the speed of P-101 is the manipulated variable.

h_f (-)	γ_{q_a} (%)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
0.50	30	15.0	5.00	0.25

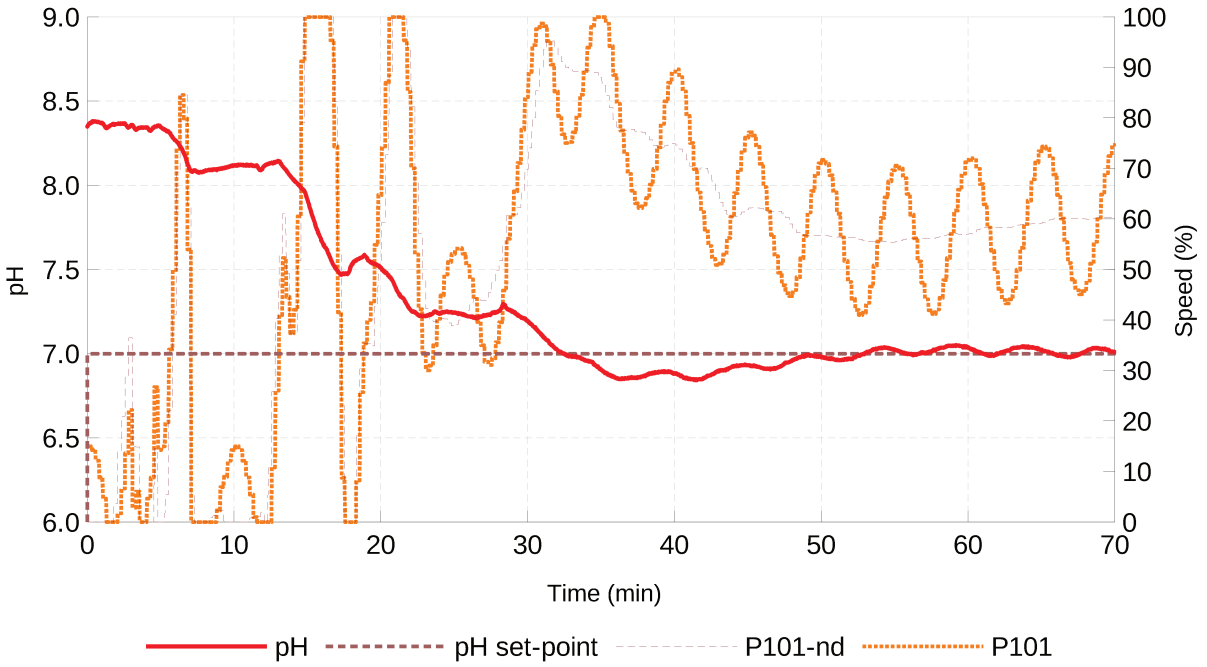

Figure 46 – SISO extremum-seeking control experiment with P-101 speed as manipulated variable and lower frequency dither.

5.2.2.2. Designed parameters

The base flow rate was set to zero, the buffer flow rate to 10 L/h, the set-point of the level of the reactor to 60%. Phase-correction was used for the modulation, and the high-pass filter was designed with the dither frequency as its cut-off frequency. The oscillations caused by the inlet can be clearly seen on the pH response, including the phase lag.

Table 18 – Designed parameters for the ES control SISO experiment where the speed of P-101 is the manipulated variable.

Phase lag (min)	γ_{q_a} (L/h)	α_{q_a} (-)	T_{p_a} (min)	T_s (min)
2.67	20	10	13.3	0.25

As can be seen the system response shown in Figure 47 is similar to the simulation in Figure 41 from Section 5.1.2.3.2, albeit a little slower and with higher average speed to reach the set-point. This can be explained with modelling deficiencies of the pump used in the simulation.

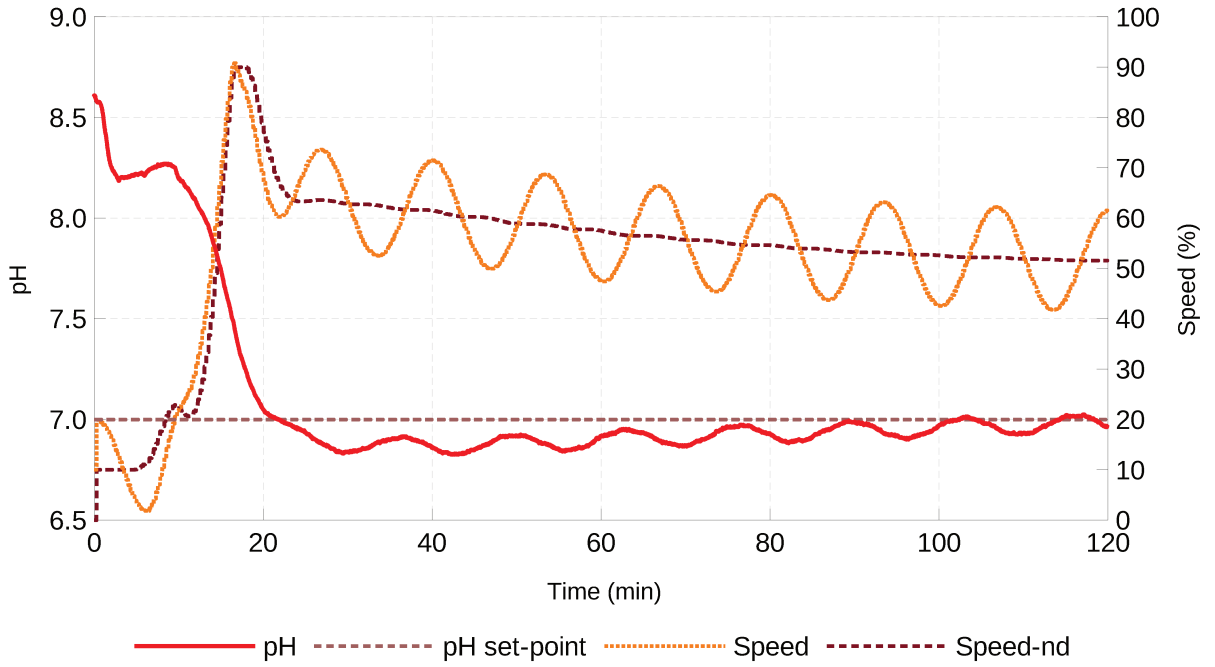

Figure 47 – SISO extremum-seeking control experiment with P-101 speed as manipulated variable and designed parameters.

The behaviour of the system can be explained as following. As a cosine perturbation is used, its values is at its peak at the beginning of the operation and the pH starts to reduce almost immediately, as expected. Then the perturbation enters its first valley and the pH starts to increase, which is detected by the integrator which in turn increases its own value and consequently the speed of the pump. Afterwards the pH decreases until it gets close to its set-point, which is also detected by the integrator so its starts slowly regulating its own value to reduce the remaining offset. As can be seen the period and the perturbation are constant, only the integrator output ("Speed-nd") changes to control the system.

The experimental MISO system can be seen in Figure 48 and its parameters can be seen in Table 19. The system can be successfully controlled but the convergence rate is quite slow, this can be explained with the use of small dither amplitudes, especially for the P-102 (base) pump, as the sum of both senoidal perturbations almost can not be seen in the system response, unlike the response shown in Figure 40. This effect was expected as shown in Figure 42, but was stronger than anticipated.

Table 19 – Designed parameters for the ES control MISO experiment where the speed of P-101 and P-102 are the manipulated variables.

γ_{q_a} (L/h)	γ_{q_b} (L/h)	α_{q_a} (-)	α_{q_b} (-)	T_{p_a} (min)	T_{p_b} (min)	T_s (min)
5.0	2.5	4.0	3.5	13.3	20	0.25

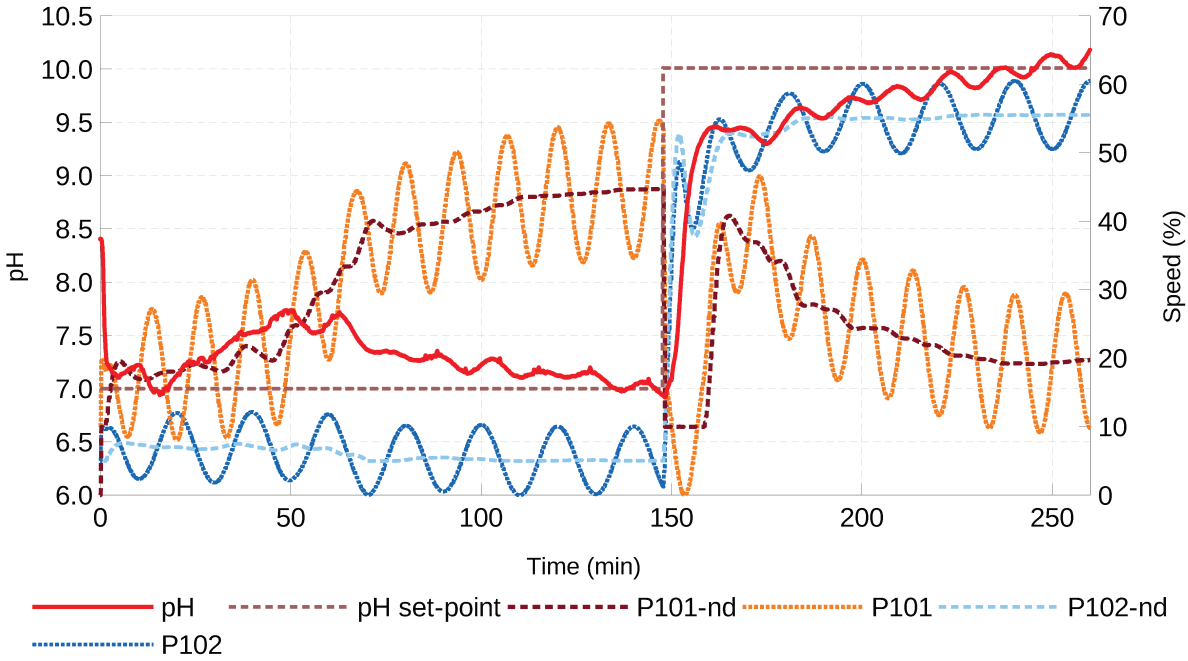

Figure 48 – MISO extremum-seeking control experiment with P-101 and P-102 speed as manipulated variables and designed parameters.

The steep pH reduction at the beginning may be explained by imperfect mixing of the buffering medium, but as can be seen the algorithm later manages to detect that the pH is rising and reduces the amount of base while increasing the amount of acid. The behaviour shown after changing the set-point did not show this anomaly and run as expected.

6. Conclusion

Extremum-seeking is an interesting technique as it does not rely on a model of a system to be able to find any available extremum, be it a maximum or a minimum. This is especially useful for systems without a model or for nonlinear systems, such as the neutralization of pH. This fact allow one to approximate the complete system to a black-box equation or function to be optimized, where its minimum or maximum correspond to the desired state of the system.

While the technique did not yet see much use in the area of chemical engineering, this work managed to apply this technique to a reactor to neutralize the pH of a buffered medium, not only in simulations but also in experiments in a pilot plant.

Other techniques could be used, but they either linearize the system, limiting its use to a range close to the state the system was linearized, or depend on a nonlinear model of the system, which may not be available especially on more complex reactional systems. As shown, one example of that is when neutralizing waste water of some industries.

This work tested extremum-seeking to tune a PI, which can be used for systems with simpler behaviours, and tested extremum-seeking control, which can be used for systems with unknown or complex behaviours.

To achieve the main objective, a simple and easy communication interface between a computer and the plant controller was created, a GUI was made to facilitate the evaluation of the current state of all plant parameters, a simulation system was created to help find suitable parameters and do a preliminar analysis of the algorithm viability, and experiments were made to evaluate the algorithms with real settings.

Even though there were some differences between the simulation and experimental results, they mostly matched each other, indicating that the approximations made to model the system were valid and the simulation could be used to further assess it.

The simulations not only helped in finding suitable trial-and-error parameters for both extremum-seeking approaches, but also allowed to test the parameters designed through rules of thumb before expending reactants and energy on the plant.

As a result the main objective was achieved, extremum-seeking was able to both tune a PI controller and directly control the system. The tuned PI controller managed to have fast response time with little overshoot, and the ES controller was able to quickly control the system with comparable performance to the PI controller, especially when using the designed

parameters. Also, ES control has the benefit of being able to adapt to changes in the buffered medium that has its pH controlled.

7. Future works

The algorithm used in this work may be considered one of the simplest forms of extremum-seeking, therefore there are various changes that can be made to increase its convergence rate to the extremum of the cost function or its stability, such as time varying ES control (GUAY, 2014) and PI-ESC (GUAY; BURNS, 2017). PI-ESC (Proportional Integral Extremum-Seeking Control) is especially interesting as it can act in the control signal before a gradient is even detected, greatly increasing the convergence rate.

Future works may put those techniques to use in the same pH control plant to compare their performance with the basic ES algorithm, and even surpass complex control systems such as model predictive control.

Also, the stability and effectiveness of the ES algorithm with changes on the buffered medium, whose pH is to be controlled, when its concentration or flow-rate suddenly changes, a common occurrence in real-life scenarios, can be tested.

8. Bibliography

AI-GHUSAIN, Ibrahim A. et al. Using pH As A Real-Time Control Parameter For Wastewater Treatment And Sludge Digestion Processes. **Water Science and Technology**, [s. l.], v. 30, n. 4, p. 159, 1994.

ANDERSON, G. K.; YANG, G. pH Control in Anaerobic Treatment of Industrial Wastewater. **Journal of Environmental Engineering**, [s. l.], v. 118, n. 4, p. 551–567, 1992.

BIZON, Nicu. Energy Harvesting from the Fuel Cell Hybrid Power Source Based on Extremum Seeking Control Schemes. In: BIZON, Nicu et al. (Eds.). **Energy Harvesting and Energy Efficiency**. Cham: Springer International Publishing, 2017. v. 37p. 329–370.

CHEREMISINOFF, Nicholas P. **Handbook of water and wastewater treatment technologies**. Boston: Butterworth-Heinemann, 2002.

CHOU, Chung-Chi (ED.). **Handbook of sugar refining: a manual for design and operation of sugar refining facilities**. New York: Wiley, 2000.

CIRI, Umberto et al. Large-eddy simulations with extremum-seeking control for individual wind turbine power optimization: LES with ESC for power optimization. **Wind Energy**, [s. l.], v. 20, n. 9, p. 1617–1634, 2017.

D. TAIT, Keith. **Fermentation diagram**. n/a. Disponível em: <http://www.ilocis.org/documents/chpt79e.htm>. Acesso em: 13 out. 2017.

DOCHAIN, Denis; PERRIER, Michel; GUAY, Martin. Extremum seeking control and its application to process and reaction systems: A survey. **Mathematics and Computers in Simulation**, [s. l.], v. 82, n. 3, p. 369–380, 2011.

ELARAFI, Mohamed Gaberalla Mohamed Khair; HISHAM, Suhaila Badarol. Modeling and control of pH neutralization using neural network predictive controller. In: INTERNATIONAL CONFERENCE ON CONTROL, AUTOMATION AND SYSTEMS, 2008. ICCAS 2008. 2008, Seoul, South Korea. **Anais**... Seoul, South Korea: IEEE, 2008. Disponível em: <http://ieeexplore.ieee.org/document/4694329/>. Acesso em: 6 out. 2017.

GUAY, Martin. A time-varying extremum-seeking control approach for discrete-time systems. **Journal of Process Control**, [s. l.], v. 24, n. 3, p. 98–112, 2014.

GUAY, Martin; BURNS, Daniel J. A proportional integral extremum-seeking control approach for discrete-time nonlinear systems. **International Journal of Control**, [s. l.], v. 90, n. 8, p. 1543–1554, 2017.

GUSTAFSSON, Tore K.; WALLER, Kurt V. Nonlinear and adaptive control of pH. **Industrial & Engineering Chemistry Research**, [s. l.], v. 31, n. 12, p. 2681–2693, 1992.

HENSON, M. A.; SEBORG, D. E. Adaptive nonlinear control of a pH neutralization process. **IEEE Transactions on Control Systems Technology**, [s. l.], v. 2, n. 3, p. 169–182, 1994.

JERMAINE, Jill. **The Importance of pH Control Throughout Drug Manufacturing**. 2014. Disponível em: <https://www.analyticexpert.com/2014/06/the-importance-of-ph-control-throughout-drug-manufacturing/>. Acesso em: 13 out. 2017.

KILLINGSWORTH, N. J.; KRSTIC, M. PID tuning using extremum seeking: online, model-free performance optimization. **IEEE Control Systems Magazine**, [s. l.], v. 26, n. 1, p. 70–79, 2006.

KRSTIĆ, Miroslav. Performance improvement and limitations in extremum seeking control. **Systems & Control Letters**, [s. l.], v. 39, n. 5, p. 313–326, 2000.

KRSTIĆ, Miroslav; WANG, Hsin-Hsiung. Stability of extremum seeking feedback for general nonlinear dynamic systems. **Automatica**, [s. l.], v. 36, n. 4, p. 595–601, 2000.

LI, Xilin; POWELL, Malcolm S.; HORBERRY, Tim. Human Factors in Control Room Operations in Mineral Processing: Elevating Control From Reactive to Proactive. **Journal of Cognitive Engineering and Decision Making**, [s. l.], v. 6, n. 1, p. 88–111, 2012.

LIU, Yan; CHEN, Yinguang; ZHOU, Qi. Effect of initial pH control on enhanced biological phosphorus removal from wastewater containing acetic and propionic acids. **Chemosphere**, [s. l.], v. 66, n. 1, p. 123–129, 2007.

LONGHI, L. G. S. et al. Nonlinear H∞ control of an experimental pH neutralization system. Latin American applied research. [s. l.], v. 34, n. 3, Latin American applied research, p. 141–148, 2004.

MA, Ruizi; ZHANG, Guoshan. Output tracking control of pH neutralisation with iterative feedback tuning. **International Journal of Modelling, Identification and Control**, [s. l.], v. 25, n. 1, p. 62, 2016.

MAHMOODI, Sanaz et al. Nonlinear model predictive control of a pH neutralization process based on Wiener–Laguerre model. **Chemical Engineering Journal**, [s. l.], v. 146, n. 3, p. 328–337, 2009.

MU, Baojie et al. Experimental evaluation of anti-windup extremum seeking control for airside economizers. **Control Engineering Practice**, [s. l.], v. 50, p. 37–47, 2016.

OBUT, Salih; ÖZGEN, Canan. Online Identification and Control of pH in a Neutralization System. **Industrial & Engineering Chemistry Research**, [s. l.], v. 47, n. 13, p. 4394–4404, 2008.

OMNICANE, Omnicane. **Sugar mill diagram**. n/a. Disponível em: <http://www.omnicane.com/sugarcane-mill>. Acesso em: 13 out. 2017.

PEIXOTO, Alessandro Jacoud et al. Modelling and Extremum Seeking Control of Gas Lifted Oil Wells. **IFAC-PapersOnLine**, [s. l.], v. 48, n. 6, p. 21–26, 2015.

SAFADY, Luís Felipe. **OPC REST API**. Software. 2017. Disponível em: <https://github.com/LAGonauta/opc-rest-api>. Acesso em: 14 out. 2017.

SALEHI, Shahin; SHAHROKHI, Mohammad; NEJATI, Ali. Adaptive nonlinear control of pH neutralization processes using fuzzy approximators. **Control Engineering Practice**, [s. l.], v. 17, n. 11, p. 1329–1337, 2009.

SANITARY DISTRICT, Oro Loma. **Sewage treatment diagram**. n/a. Disponível em: <https://oroloma.org/sewage-treatment/>. Acesso em: 12 out. 2017.

SEELER, Karl A. **System dynamics: an introduction for mechanical engineers**. New York, NY: Springer, 2014.

SKOOG, Douglas A. **Fundamentals of analytical chemistry**. 9th Ed ed. Belmont, CA: Cengage - Brooks/Cole, 2012.

VANDERMEULEN, Isaac; GUAY, Martin; MCLELLAN, P. James. Formation control of high-altitude balloons experiencing real wind currents by discrete-time distributed extremum seeking control. In: AMERICAN CONTROL CONFERENCE (ACC) 2017, Seattle, USA. **Anais**... Seattle, USA: IEEE, 2017. Disponível em: <http://ieeexplore.ieee.org/document/7963082/>. Acesso em: 21 set. 2017.

VAZ DA COSTA, Thiago. **Estudo e Implementação de Estruturas de Controle Reconfigurável aplicado a Processos Químicos**. 2014. Universidade de Campinas, Campinas, 2014. Disponível em: <http://repositorio.unicamp.br/jspui/handle/REPOSIP/266123>

VIJAYARAGAVAN, V. et al. pH automation in sugar industries. In: 2015 INTERNATIONAL CONFERENCE ON COMPUTER COMMUNICATION AND INFORMATICS (ICCCI) 2015, Coimbatore, India. **Anais**... Coimbatore, India: IEEE, 2015. Disponível em: <http://ieeexplore.ieee.org/document/7218160/>. Acesso em: 12 out. 2017.

Y. TAN; Y. P. LI; I. MAREELS. Anti-windup for Extremum Seeking. In: 2012 2ND AUSTRALIAN CONTROL CONFERENCE 2012, Sydney, Australia. **Anais**... Sydney, Australia: IEEE, 2012. Disponível em: <http://ieeexplore.ieee.org/document/6613207/>

ZHANG, Chunlei; ORDÓÑEZ, Raúl. **Extremum-seeking control and applications: a numerical optimization-based approach**. New York ; London: Springer, 2012.