

Article

# Optimization Based on Pattern Search Algorithm Applied to pH Non-Linear Control: Application to Alkalinization Process of Sugar Juice

Jairo Palacio-Morales <sup>1</sup>, Andrés Tobón <sup>2</sup> and Jorge Herrera <sup>3,\*</sup> 

<sup>1</sup> Quality and Production Department, Instituto Tecnológico Metropolitano (ITM), Medellín 050034, Colombia; jairopalacio@itm.edu.co

<sup>2</sup> Electronic Engineering and Telecommunications Department, Instituto Tecnológico Metropolitano (ITM), Medellín 050034, Colombia; andrestobon@itm.edu.co

<sup>3</sup> Facultad de Ciencias Económicas y Administrativas, Universidad Jorge Tadeo Lozano, Bogotá 110311, Colombia

\* Correspondence: jorgea.herrerac@utadeo.edu.co; Tel.: +57-1-242-7030

**Abstract:** In this paper, an approach for the tuning of a model-based non-linear predictive control (NMPC) is presented. The proposed control uses the pattern search optimization algorithm (PSM), which is applied to the pH non-linear control in the alkalinization process of sugar juice. First, the model identification is made using the Takagi Sugeno T-S fuzzy inference systems with multidimensional fuzzy sets; the next step is the controller parameters tuning. The PSM algorithm is used in both cases. The proposed approach allows the minimization of model uncertainty and decreases, in the response, the error in a steady state when compared with other authors who perform the same procedure but apply other optimization algorithms. The results show an improvement in the steady-state error in the plant response.

**Keywords:** optimization; alkalinization; non-linear control



**Citation:** Palacio-Morales, J.; Tobón, A.; Herrera, J. Optimization Based on Pattern Search Algorithm Applied to pH Non-Linear Control: Application to Alkalinization Process of Sugar Juice. *Processes* **2021**, *9*, 2283. <https://doi.org/10.3390/pr9122283>

Academic Editor: Dimitrios I. Gerogiorgis

Received: 5 October 2021

Accepted: 18 November 2021

Published: 20 December 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The existence of productive processes with strong non-linearity forces different control strategies. One of them is adaptive control. NMPC is a non-linear control strategy used to control different systems such as wastewater treatment, chemical, biotechnological, wind turbine systems, wave oscillations, and motion control, among others [1–5].

The pH becomes a crucial magnitude in biotechnological and food sector applications. In these processes, the control of pH goes from linear to non-linear applications, highlighting the importance of the control strategy and its tuning. For example, it is the case of the clarification sugarcane juice process, where the pH is affected by whitewash flow and both by temperature as the flow of the juice.

The NMPC tuning, due to its complexity, is made using optimization methods, and in some cases, the tuning is online. There are several optimization methods to solve different kinds of problems.

In many cases, optimization strategies, such as genetic algorithm (GA), evolution strategic (ES), and particle swarm optimization (PSO) are made up of a population of particles, which are the starting point of an optimal value, which in some cases shows stagnation in local minimums [6–9]. In this range of optimization methods for the tuning of fuzzy control systems with PID structures, classical techniques such as gradient algorithms and Rosenbrok's algorithms are also used [10]. In these cases, the former uses the gradient of the objective function, looking for possible directions that sometimes must be corrected utilizing offsets. The latter is characterized by not employing derivatives; other alternative methods appear, such as the (ACO) and (GSO) algorithms, which present interesting results in particular applications [11].

They use historical information to find new points that allow an adequate solution. These algorithms differ from other techniques in that they use aptitude functions instead of derivatives. Additionally, they work with a set of design points instead of a single point [12]. However, although this optimization structure is widely used, it presents adjustment difficulties in the parameters of the operators, and a significant dependence between the type of problem and the adjustment of said parameters [13–15]. It is presented as a variant of GA, in which crossover operators are not used and have a high capacity for mutations. These mutations are determined exclusively by the mutation operator, where the same parameters are required as the GA, and in turn, have the same disadvantages [16].

Moreover, alternatives, such as ant colony optimization (ACO), are applied in various works, in which the behavior of ant or bee colonies is emulated [17–21] are presented. These colonies are made up of individuals who carry out various tasks. Although an individual may not be able to perform his task, the colony as a whole does. These algorithms are initially inspired by how ants surround obstacles while transporting their food, transmitting information through marker pheromones. However, it has as a disadvantage the tendency to stagnate in local minimums [22,23].

Another recognized alternative is particle swarm optimization (PSO), which is based on the social behavior of animals. The pursuit of a point is carried out starting from these particles that develop a systemic search throughout the operating range. Then, according to mathematical rules that consider their position, the particles know the best point reached by any other particles in the swarm. This optimization strategy is more straightforward than other similar strategies because it does not require excessive adjustments. However, just as the ACO method tends to stay at a local minimum [22,23], or requires too high a population [24].

In recent years, some research has demonstrated the excellent performance of applications using methods, such as DE and PSO algorithms. Some of these strategies also show a solid path in the handling of Membership Functions and parameter tuning for applications in the world of robotic systems [11]. On the other hand, other relationships are presented in the field of control and robotics, where fuzzy optimization strategies based on animal nature, such as grey wolf GWO optimizers, stand out for their behavioral [25]. There is also the appearance of other research-oriented to work on pattern recognition through Sugeno strategies and fuzzy systems [26,27], as well as in the comparison between different optimization algorithms and the adjustment of their parameters [28]. However, a lack of work focused on advanced control applications with multidimensional fuzzy sets combined with methods that can perform non-random searches.

The appearance of an optimization algorithm, such as PSM, is advantageous because it has global convergence as its main characteristic, which means that it does not generate stagnation in local minimum since it presents an exhaustive search throughout the search range. PSM is another great advantage because it can be implemented under simple mathematical operations, allowing relatively easy implementation. As an example, it can be seen how PSM has been implemented in mathematics, and optimization theory [29,30]. The PSM pattern search method is part of the optimization techniques of the family of metaheuristic algorithms. This method consists of an array where a restricted search is performed. Said matrix is defined as a mesh, which is related to a sequence of survey conditions. The conditions generated by the surveys give the guidelines for reducing the current mesh, thus ensuring algorithm convergence. In such a way that it presents a satisfactory operation for applications with multiple local minimum [31] and is flawlessly applied to NMPC techniques such as those presented in this article. The improvements identified after the application of the method are as follows:

- The PSM improves the search for local minima to make the tuning more accurate values;
- When tuning the models obtained from the SIB T-S with CBMD with PSM, the obtained results improve compared with other authors;

- By optimizing the tuning process of control strategies applied to non-linear systems, such as NMPC reflects results with lower steady-state error than most other authors.

The generalized pattern search method is presented in Section 2. Section 3 discusses the problem formulation. Section 4 provides an analysis of results. Finally, our main conclusions are given in Section 5.

## 2. Generalized Pattern Search Method (GPSM)

The pattern search algorithm is mainly made up of two parts, the first one, constraint conditions and the second, an equidistant particle mesh formed between the constraints already given. The evaluation of the particles in the function gives the guidelines to determine the directions the particles must define towards which direction they should start by giving previously determined jumps to search for the entire work area. The Figure 1 presented below is taken from [32].

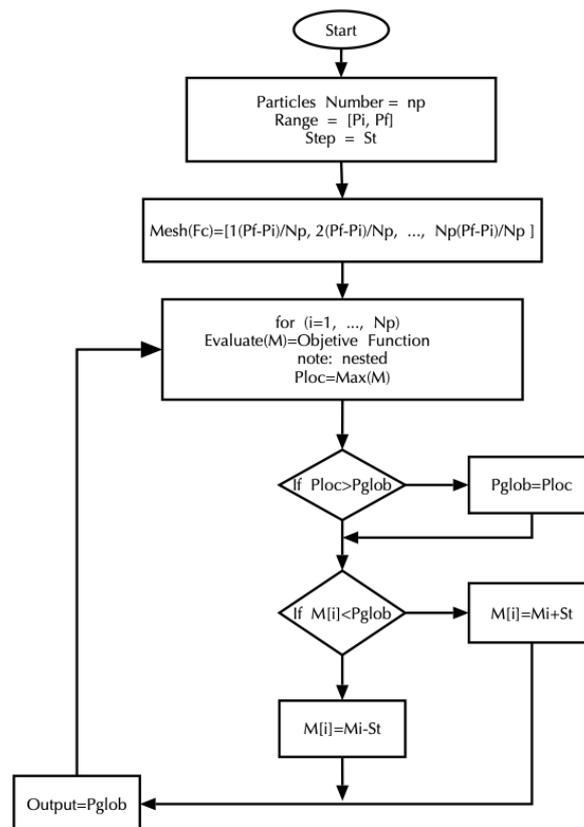
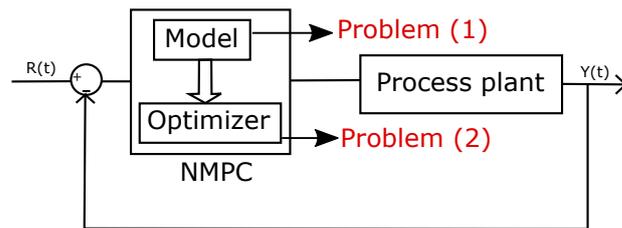


Figure 1. Generalized pattern search algorithm.

## 3. Problem Formulation

Two main problems are presented in the control diagram, the first one is to identify the parameters of the Takagi Sugeno Fuzzy Inference System model defined as hereafter as SIB T-S. These algebraic models use as numerical consequent in each of their rules a linear function for each model input. The second problem is focused on finding the parameters of the control tuner as shown in Figure 2.



**Figure 2.** General representation of a NMPC.

### 3.1. Problem 1: SIB T-S and CBMD Model

The CBMD defined as Multidimensional Fuzzy Sets are used for work development, which for an SIB T-S use the operation of an implicit partition as stated by [33], developing an approach when using a SIB as a model T-S. The tasks related to the structural and parametric definition of the model can be made more flexible using SIB T-S, which allow information to be grouped in a blurred way to obtain a CBMD.

The dynamic system treated in this work is described employing the rules of a SIB T-S that are shown in the Equation (1) as follows:

$$\begin{aligned}
 y^i(k) = & p_0^i + p_{11}^i x_1(k - n_{d1}) + \dots \\
 & + p_{1n_{u1}}^i x_1(k - (n_{d1} + n_{u1} - 1)) + p_{21}^i x_2(k - n_{d2}) + \\
 & \dots + p_{2n_{u2}}^i x_2(k - (n_{d2} + n_{u2} - 1)) + \dots + \\
 & p_{M1}^i x_M(k - n_{dM}) + \dots + p_{Mn_{uM}}^i x_M(k - (n_{dM} + n_{uM} - 1)) \\
 & + b_1^i y(k - n_v) + \dots + b_n^i y(k - (n_v + n_y - 1))
 \end{aligned} \quad (1)$$

Being  $R_i$ : for the Equation (2) to  $A^{(i)}$ ,  $r(k)$  being the input space represented by the following vector:

$$\begin{aligned}
 r(k) = & [x_1(k - n_{d1}), \dots, x_1(k - (n_{d1} + n_{u1} - 1)), \dots, \\
 & x_M(k - n_{dM}), \dots, x_M(k - (n_{dM} + n_{uM} - 1)), y(k - n_{d1}), \\
 & \dots, y(k - n_v), \dots, y(k - (n_y + n_v - 1))]
 \end{aligned} \quad (2)$$

$A^{(i)}$  is the fuzzy set for spaces  $X_1 \times \dots \times X_1 \times \dots \times X_M \times Y \times \dots \times Y$ . In Table 1 description of the variables of Equations (1) and (2) are presented.

We find that,  $r(k)$  is formed by the elements associated to current and lagged values of the input variable at any instant, with dimensions in space defined by  $n$ .

Identification using SIB T-S requires initially determining the structure of the model to subsequently perform the process of tuning the regressors, antecedents, and consequent  $p$ ,  $q$ ,  $r$ ,  $s$ ,  $v$  of the model SIB T-S and CBMD of the Equation (3). These elements represent the tuning of predicted state regressors, determining the maximum number of regressions that each variable can take to obtain a suitable model.

$$\begin{aligned}
 X = & [u(k - p), \dots, u(k - 1), d(k - q), \dots, d(k - 1), \\
 & y(k - r), \dots, y(k - 1), \hat{x}_2(k - s), \dots, \hat{x}_2(k - 1), \\
 & \hat{x}_1(k - v), \dots, \hat{x}_1(k - 1)]
 \end{aligned} \quad (3)$$

**Table 1.** Description of the variables of Equations (1) and (2) NMPC.

Parameter	Definition
$p_0^i$	It is the initial value of the rule output.
$x_j(k)$	Is the value, at the time $k$ , of the input variable .
$y(k)$	Is the value, at time $k$ , of the system output variable
$x_j(k - n)$	Is the value, $n$ instants before $k$ , of the input variable $j$
$n \in \{0 \cup \mathbb{Z}^+\}; (k - n)$	Is the value, $n$ moments before $k$ , of the system output variable
$n \in \mathbb{Z}^+; p_j(i)$	Is the value of the parameter associated with variable $j$ in rule $i$ .
$j = 1, \dots, M; b_j(i)$	Is the value of the parameter associated with the delayed system output variable $j$ times.
$nd_j \in \{0 \cup \mathbb{Z}^+\}$	Is the time it takes for a previous value of an input $j$ to take effect in the output value of the system.
$nv \in \mathbb{Z}^+$	Is the time it takes for the value of a previous system output to take effect in the value of the system output.
$ni(\geq nd_j)j$	Is the number of previous values of the input variable $j$ , to be considered in the regressor; ( $\geq nv$ ) is the number of previous values of the system output, to be considered in the regressor.

The variables used in the sugarcane alkalization process model are defined in Table 2.

**Table 2.** Process input and output variables.

Process Variable	Model Variable
Jacket flow $F_j$	Manipulated variable ( $u$ )
Reagent input flow $F$	Measured disturbance ( $d$ )
reagent concentration $CA$	Controlled variable $y$
Reactor temperature $T$	First state ( $x_1$ )
Jacket temperature $T_j$	Second state ( $x_2$ )

For each variable defined in Table 2, the preliminary regressor of the algorithm includes the input flows that are progressively increased until the  $n$  regressions are reached. Initially, the model is  $\hat{x} = (u, d)$ , to additionally include the output of the process  $pH_3$  in the regressor defined as  $\hat{x} = f(u, d, y)$ , to conclude, including the states estimated by the same model  $\hat{x} = f(u, d, y, \hat{x})$ . For each regressor, a SIB T-S is identified with  $m$  being the number of CBMD validated against actual data to evaluate its performance. To model the SIB in state space, the algorithm simultaneously identifies and validates states  $x_1$  and  $x_2$  to include in the regressor the predicted states  $\hat{x}$  as presented in Equation (3). Therefore,  $p, q, r, s, v$ , are defined as the maximum number of regressions that each variable can take in the algorithm used to optimize the regressor.

### 3.1.1. Objective Function of the Model

In Equation (4), which is the objective function of the model, the actual model is compared with the model identified by tuning SIB T-S with CBMD . Then, the evaluation of the system is executed utilizing performance indices that serve as a tool to determine

relevant characteristics to conclude on the results obtained. The main performance criterion used to determine the behavior of SIB T-S with CBMD is the integral of the quadratic error ICE that is described by the Equation (4), where the error is calculated with the values of difference of process variables. This performance index is adequate for evaluating the tuning of the SIB T-S since the identified model is required to reproduce the actual plant most exactly, looking for a minimum error, leaving the time of processing as the ICE does with the broadest errors.

$$FICE(pH) = \int_0^{\infty} (pH_{real} - pH_{SIB})^2 dt \quad (4)$$

### 3.1.2. Control Algorithm Proposed Using GPSM to Obtain the Parameters of the SIB T-S and CBMD Model

The GPSM is implemented using the so-called multi-model applied by [34], where it can work non-linear systems with multiple variables. For example, in problem 1, where the model variables of the SIB T-S and CBMD must be found, the GPSM is implemented in Figure 3.

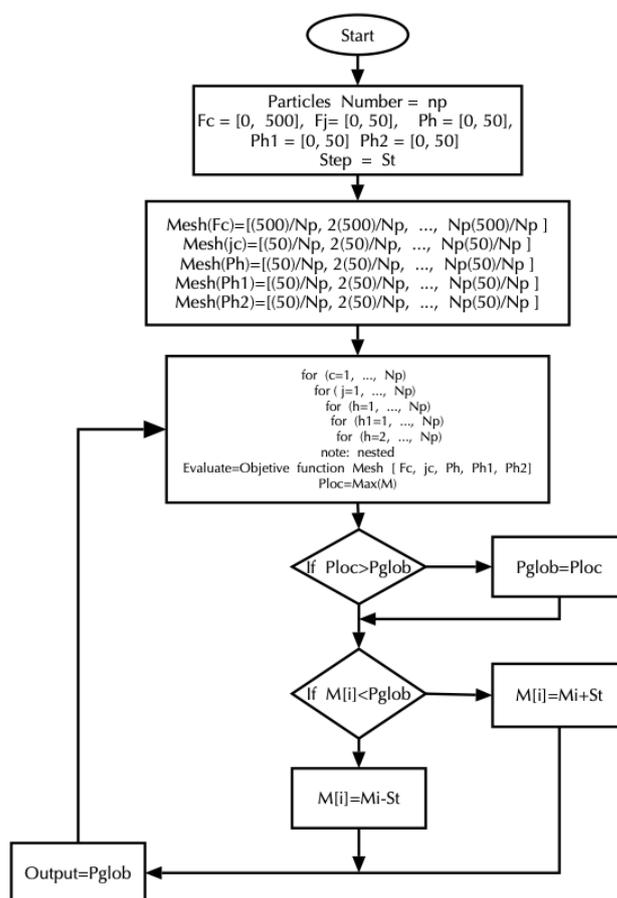


Figure 3. Proposed method based on pattern search to problem 1.

### 3.2. Problem 2, the Optimizer

The objective function  $FC$  represented in the Equation (5), is part of the Optimization Algorithm. This function describes the system's performance through its minimization, becoming a performance index that determines the usefulness of the control policy. Generally, this mathematical expression must be minimized to represent the finding of a specific reasonable control action that is applied in each sampling period. One of the  $FC$  most used in predictive control strategies is the quadratic representation that has expressions that

describe the constraints [35]. Furthermore, the FC structure of one input and one output system extends to multiple inputs and multiple output systems as determined by [36].

$$FC[y(k), u(k)] = \sum_{i=P_1}^{M_1} \alpha_i (y_{ref}(k+i) - \hat{y}(k + \frac{i}{k}))^2 + \sum_{i=P_2}^{M_2} \beta_i [\Delta u(k+i-1)]^2 \quad (5)$$

The definition of variables used in Equation (5) are presented in Table 3.

**Table 3.** Identification of values to tune in the NMPC.

Parameter	Definition
$P_1, P_2, P_3$	Represents the future steps in which the prediction is made.
$M_1, M_2, M_3$	Terms associated with the initial horizon
$\alpha$	Terms associated with the control horizon.
$\beta$	Weighting weight associated with the error
$\hat{y}$	Weight of weight associated with change of action control
$y_{ref}$	Estimated output
$\Delta u(k+i-1)$	Reference value of the output
$\Delta \hat{y}(k + \frac{i}{k})$	Term associated with change of control action
	Term associated with the change of the predicted output

The parameters  $P_1, P_2, P_3, M_1, M_2$  and  $M_3$  depend on the application, which perform scaling. They must always satisfy that  $1 \leq P_i \leq M_i \leq P$  with  $P$  the prediction horizon  $e_i = 1, 2, 3$ . Usually  $P_1, P_2$  and  $P_3$  are taken equal to 1.0 while  $M_1$  and  $M_3$  are taken equal to  $p$  and  $M_2$  becomes equal to the control horizon  $M$ .

The structure of the most used FC in the NMPC is the quadratic that may or may not have restrictions that must be taken into account when executing the optimization [35]. The form represented in the Equation (5) is the essential formulation for non-linear plants that have been applied discretely in case studies addressed by other authors [37].

The main performance criterion used to determine the behavior of NMPC is the integral of the squared error for time  $FICET$  described by the following equation, where it is a function of error and time. The variable  $t$  is the time with which the response speed parameters in the function are penalized. The parameter allows punishing the present errors time after the beginning of the entrance to the system.

This performance index is adequate for evaluating the behavior of the NMPC tuned because errors are heavily penalized and, additionally, the time factor is taken into account, which will allow the relationship with the controller's transient and processing response.

$$FICET(pH) = \int_0^{\infty} t(pH - pH_{setpoint})^2 dt \quad (6)$$

#### Control Algorithm Proposed Using GPSM to Obtain Tuner Parameters

The GPSM is implemented, as in problem 1, using the multi-model applied by [34], but in this time, it is applied to the tuner, as seen in Figure 4.

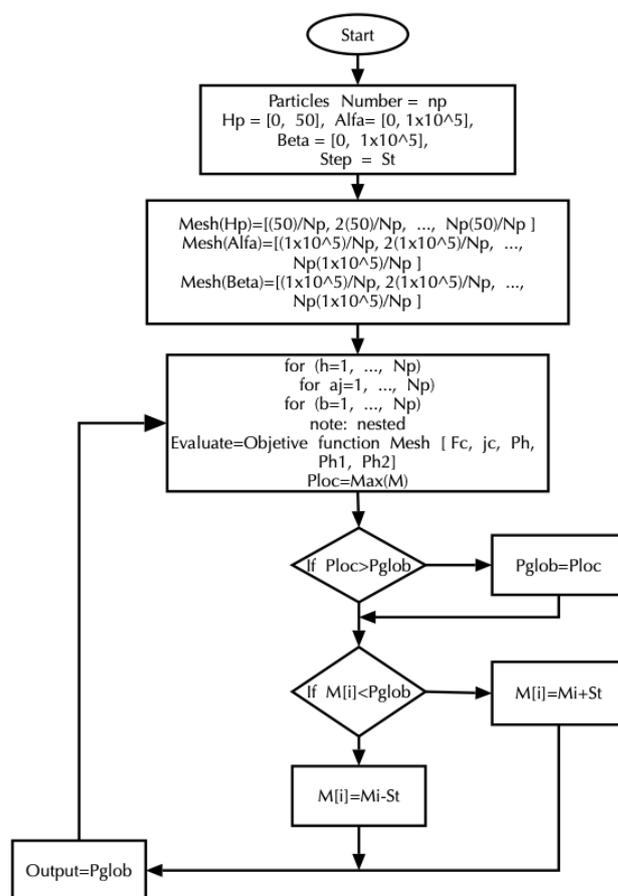


Figure 4. Proposed method based on pattern search to problem 2.

### 3.3. Case Study, Alkalinization of Sugar Cane Juice

NMPC implementation is required in various chemical processes, such as the alkalinization process of the sugar cane juice. An advanced control strategy is required in this process due to the difficulty in adjusting the pH to normal operational conditions, which is reflected in its non-linear behavior. PH is the quantity that describes the acidity/alkalinity of a substance at specific temperature conditions. Its measurement scale is expressed with range values between 0 and 14, which allows determining the hydronium concentration measurement activity in a solution. The measurement of pH can be represented by a function of concentration, which is expressed mathematically with logarithmic elements in base ten and has the following characteristics of its behavior:

- The final control element associated with pH control must guarantee exact base or acid values; this means small or large dosages that become process constraints;
- The pH response to an input that is the addition of reagents shows a non-linear behavior;
- In this type of process, the pH is presented as a controlled variable, and the manipulated variable is the addition of reagents;
- The process dead time changes depending on: the variation in the volume stored in the tank, the reagent inlet flow and the mixing speed;
- When the system gain is small (indicates a high acid content or low base content), generating a response with low speed.

The pH modelling process in an alkalinization process is not the object of study of this work, which is why any reference will be based on the development elaborated by [37]. In Figure 5, a general scheme of cane juice treatment for sugar production is presented. Sulfidation is the stage where adding  $\text{SO}_2$  to the raw cane juice in countercurrent, through the sulfidation tower, to reduce coloring compounds. Figure 6 shows the general alkalinization diagram of cane juice for pH control.

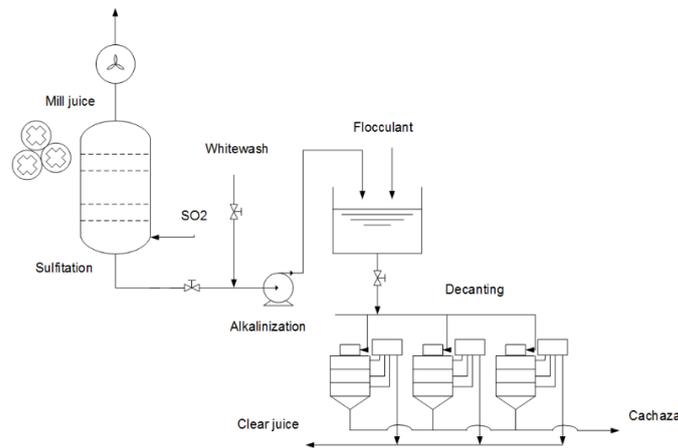


Figure 5. Sugar cane juice treatment scheme.

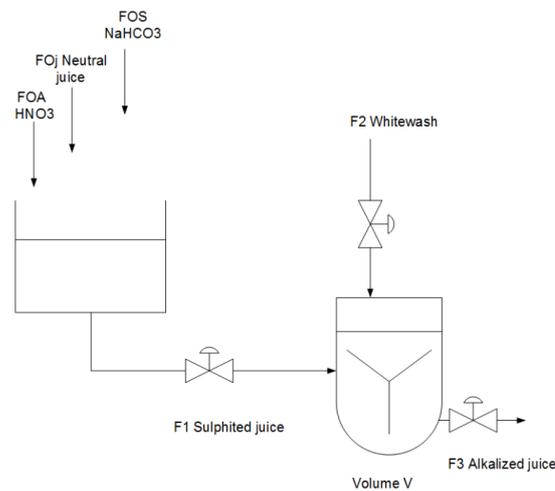


Figure 6. Cane juice alkalization stage for pH control.

Taking into account that the sugarcane juice contains a strong acid  $\text{HNO}_3$  and a weak acid  $\text{NaHO}_3$ . The two dissociation constants  $K_{a1}$  and  $K_{a2}$  will be taken equal to those of the acid carbonic. The chemical reaction is modelled based on the approximation of two reaction invariants, such as the electronic charge of the solution  $W_a$  and the final anion concentration  $W_b$ . The reaction invariants are balanced in the current  $i$  so they will be represented by  $W_{ai}$  and  $W_{bi}$ .

Starting with the variables  $u = F_2$  and the state variables  $X_1 = W_{a3}$  and  $X_2 = W_{b3}$  the model in state variables is represented as follows:

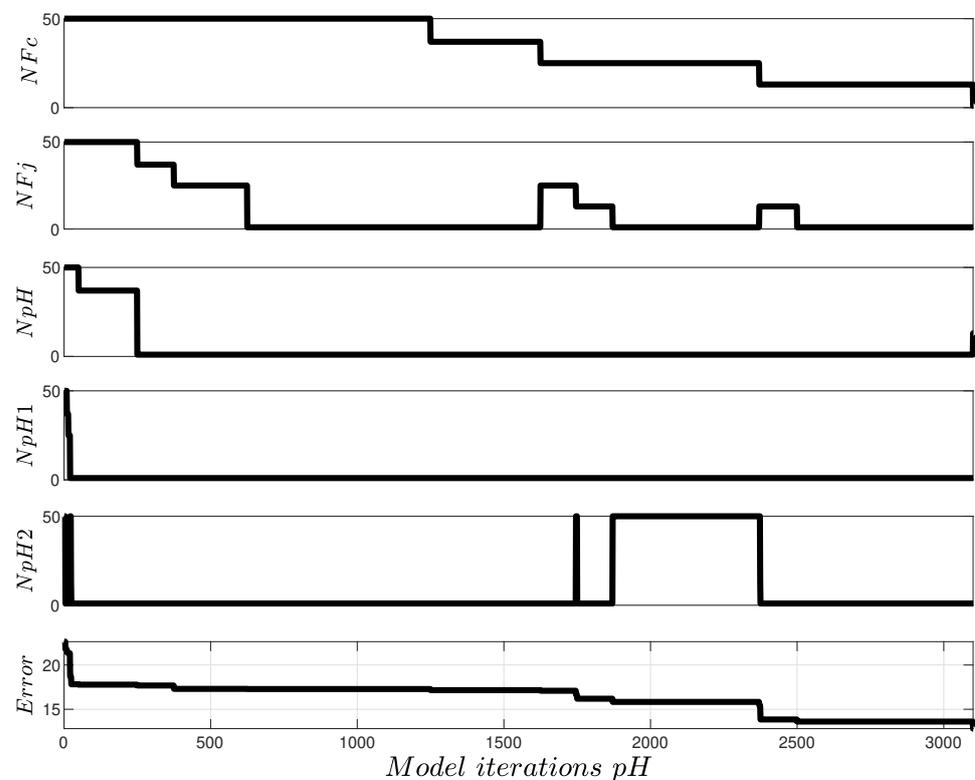
$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \end{bmatrix} = \begin{bmatrix} \frac{F_2}{V} (W_{a1} - W_{a3}) \\ \frac{F_2}{V} (W_{b1} - W_{b3}) \end{bmatrix} + \begin{bmatrix} \frac{W_{a2} - W_{a3}}{V} \\ \frac{W_{b2} - W_{b3}}{V} \end{bmatrix} u \quad (7)$$

It is required to know each one of the values of the process variables to normalize, according to [37] expressed in Table 4 that represents the engineering units for the normalization of the variables.

**Table 4.** Interval of engineering units for the normalization of the variables of the alkalization process of the sugar cane juice.

Magnitude	Unity	Range	Initial Condition
Lime milk flow ( $F_2$ )	GPM	[10, 70]	23.63
Sulfited juice flow ( $F_1$ )	GPM	[500, 2100]	1657.3
pH of lime milk ( $pH_2$ )	pH	[11.7, 12.2]	12.33
pH of sulphited juice ( $pH_1$ )	pH	[4.1, 5]	4.4965
Electric charge ( $W_{a3}$ )	C	$[-9.8090 \times 10^{-6}, -0.0026]$	$-2.6 \times 10^{-4}$
Ultimate anion	$\frac{mol}{gal}$	[1.5776, 6.5704]	$4.06 \times 10^{-4}$
pH of the alkalized juice	pH	[5.7, 8.7]	7.3481

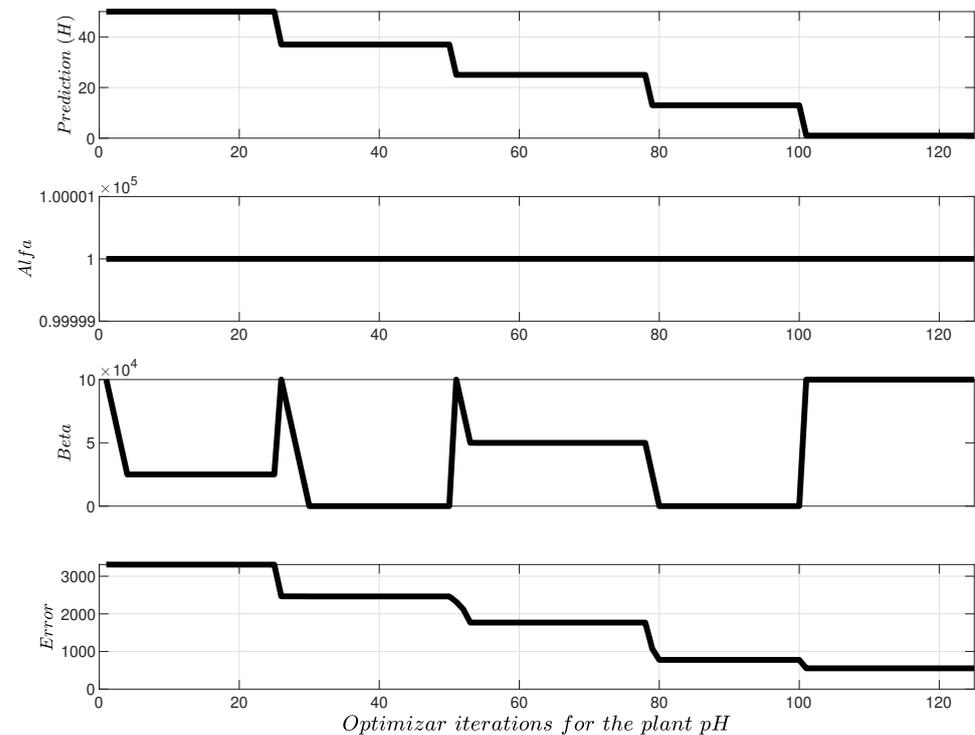
Figure 7 shows the graph of the evolution of the parameters for the second case study, where the optimization algorithm performs an exhaustive search for the minor error to find the tuning parameters of the SIB T-S and CBMD that allows obtaining the most suitable system model. Graphically it is observed how the algorithm PSM is in charge of preserving the value of the minimum obtained and the parameters with which it is obtained until a global minimum is found. As a result of the previous simulations, suitable values are obtained for tuning the SIB T-S and CBMD, which allows the most suitable model to be identified, starting from a comparison between the actual model and the model identified with the tuning parameters. It is defined which identified model presents minor differences or errors concerning the real one. With several fuzzy sets equal to 7, the PSM searches for the ideal values for tuning the SIB T-S and CBMD for the regressors associated with the second case study as shown in Table 5.

**Figure 7.** Evolution of regressor parameters  $pH$ ,  $F_c$ ,  $F_j$ ,  $pH_1$ ,  $pH_2$  with the algorithm PSM.

**Table 5.** Tuning values for the SIB T-S and CBMD .

Regression Variables	Optimal Regressor Values
Lime milk flow $F_c$	1
Sulfited juice flow $F_j$	1
pH of alkalized juice pH	1
pH of lime milk $pH1$	1
pH of sulphited juice $pH2$	1

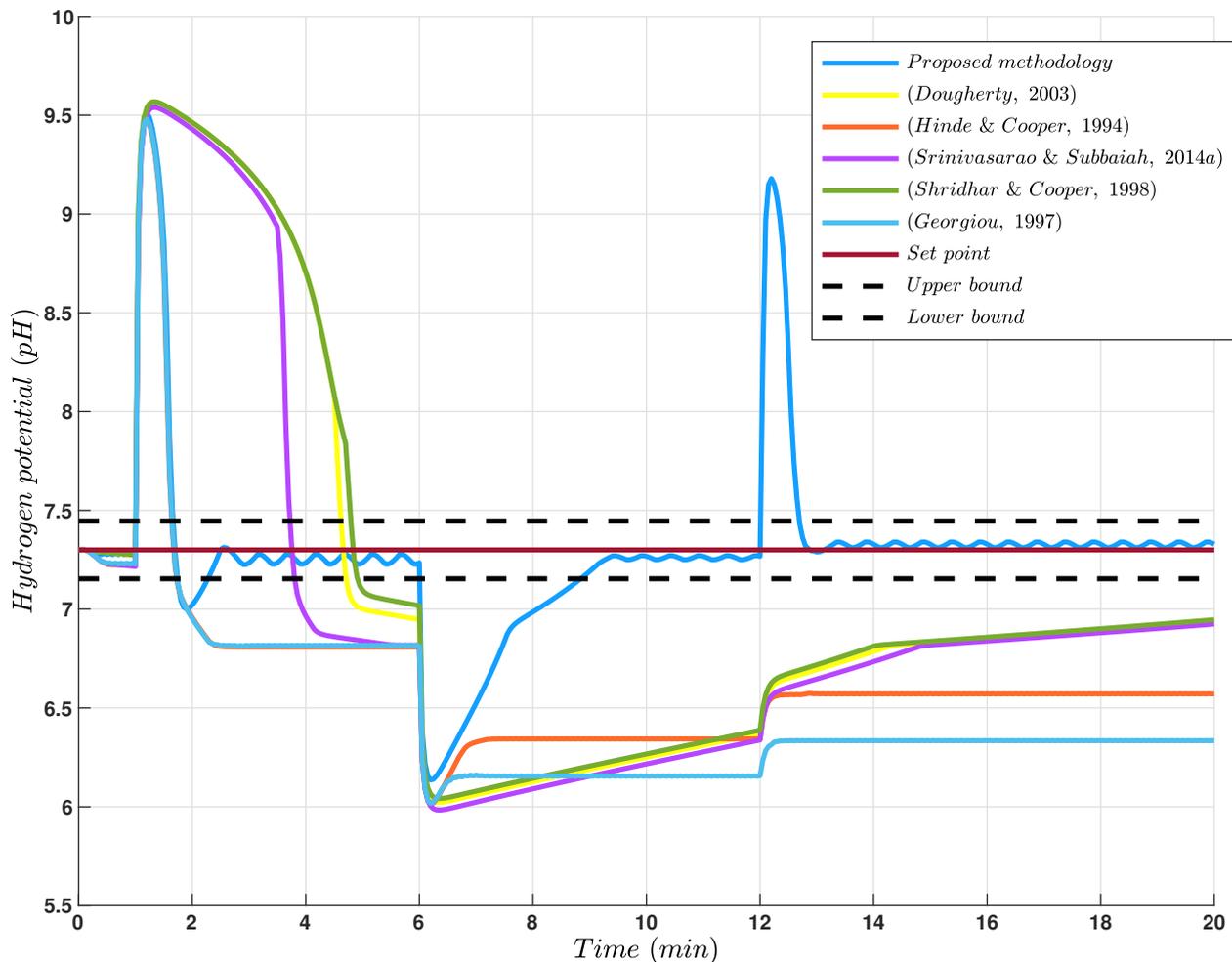
It is required to determine the conditions that are required for a given process for tuning the controller. Among the most relevant conditions to consider are response speed, reduction in disturbances, reduction in error, reduction in excessive controller action and compliance with operating conditions or restrictions. For the case study, the tuning of the prediction horizon  $P$ , the control horizon  $M$ , and the weights of the cost function  $\alpha$ ,  $\beta$ , and  $\gamma$  are presented in Figure 8. First, this tuning is developed with the values previously obtained for each parameter; then, the comparison is made with the behavior of the tuned controller with proposals from other authors to validate it.

**Figure 8.** Evolution of tuning parameters  $P$ ,  $\alpha$  and  $\beta$  with algorithm PSM case Alkalization of sugar cane juice.

#### 4. Analysis of Results

The simulations performed in this work are run on an Intel® Core(TM) i3 CPU 2.53 Ghz. Simulations start with disturbances common in natural sugar cane juice clarification processes, such as the reduction in the flow of sulfite cane and the reduction in the pH of sulfite cane juice. These perturbations generate direct alterations in the pH of the process and can be detailed in Figure 9 at 1 min and 12 min, to obtain the behavior of the time response that we will analyze below. Initially, behavior analysis of the NMPC for pH control is carried out, in line with the proposals that various authors have put forward in comparison with the proposal developed in this article. Due to the requirements of the process, it is necessary to define an upper and lower value or range of permissible error, which is represented graphically with the black strip with the desired pH value in the

central part of Figure 9. This range presents a set point of 7.34, which is the ideal value for maintaining the process's pH.



**Figure 9.** Comparison of the response of NMPC with SIB T-S with CBMD tuned with various proposals case study alkalinization of sugar cane juice.

The main performance criterion used to determine the behavior of NMPC is the integral of the squared error for time  $ICET$  that is described by Equation (8) for pH as a function of error and time. The variable  $t$  is the time with which the response speed parameters in the function are penalized.  $pH$  is the current value of the quantities to be controlled.

$$FICET(pH) = \int_0^{\infty} t(pH - pH_{setpoint})^2 dt \quad (8)$$

This performance index is suitable for evaluating the tuned NMPC behavior because errors are strongly penalized, and the time factor that will allow the relationship with the controller's transient response and processing is taken into account. For example, Figure 9 shows the behavior of NMPC with SIB T-S and CBMD tuned with various methods to validate the results obtained with the proposed PSM algorithm.

The optimization algorithm performs an exhaustive search reducing the number of trials to 230 iterations. When the optimizer does not find a different minimum error, it is responsible for storing the data of the minimum previously obtained and the variables that generate it. As a result of previous experiments, the ideal tuning values for SIB T-S and CBMD are obtained, with minimum errors presented. Optimal tuning values are several sets equal to 3, regressor associated with concentration  $CA$  equal to 9, regressor associated

with reactor temperature  $T$  equal to 9, regressor associated with jacket temperature  $T_j$  equal to 1, regressor associated with reactor inflow  $F$  equal to 9, regressor associated with jacket input flow  $F_j$  equal to 1.

The proposals of [38,39] present considerable perturbation of 31.07% and 30.68%, respectively. Both proposals present a relatively acceptable final control element effort.

However, errors with ICET performance index with values in the order of  $2.4831 \times 10^3$  are still too high, without reaching an optimal value at steady state, being far away from the setpoint and falling outside the desired pH range, as detailed in Table 6. The tuning developed with parameters proposed by [40] presents a minor impulse overshoot when receiving a second perturbation. However, this response does not reach an acceptable pH value for the process. The NMPC behavior of the SIB T-S and CBMD tunes the parameters found with the proposed approach, presenting an adequate process behavior. The error of the ICET performance index presented with the controller, as shown in Tables 3 and 5, is the minimum value presented with the other tunings and is within the acceptable margin of error. Analyzing in the time response other aspects that can be valuable, it is found that the overshoot generated for the first perturbation for all the tunings are very similar, being between 29.8% and 31.07%. The aspect against the response generated by the proposal presented is the ripple in the time response. This aspect can be improved and opens the possibility for future works. The derivative parameter can be used to combine the proposal developed with traditional PID control strategies. In this case, the derivative parameter can allow its characteristics to generate the necessary speed to correct the oscillations. Additionally, it opens the possibility of working with hybrid control strategies and sliding modes.

**Table 6.** Performance indices, CT: Computer time.

Tuning Proposal	ICET pH	ICET Average	$M_p$ [%]	CT [s]	Effort EFC
Proposed methodology	553.2816	0.0690	30.3170	$4.2783 \times 10^{-4}$	79.1858
[39]	$2.4831 \times 10^3$	0.3096	31.0739	$4.2388 \times 10^{-4}$	28.9337
[41]	$2.5321 \times 10^3$	0.3157	29.8152	$4.2881 \times 10^{-4}$	50.5342
[38]	$2.4112 \times 10^3$	0.3006	31.0739	$4.4185 \times 10^{-4}$	28.0000
[42]	$2.5011 \times 10^3$	0.3119	30.6831	$4.6854 \times 10^{-4}$	31.3067
[40]	$3.9504 \times 10^3$	0.4926	29.8718	$4.4517 \times 10^{-4}$	83.4359

## 5. Conclusions

In this article, the practical application for the optimization of the control of a strongly non-linear system with a controlled variable such as pH, using the PSM optimization algorithm is presented.

- It is identified that the models obtained using fuzzy inference systems SIB T-S and CBMD present a favorability for control-oriented applications, demonstrated to be more flexible in chemical processes with application in the industry. Thus improving their efficiency because they avoid the contribution of the uncertainties inherent to the model;
- The emergence of a novel optimization method such as the PSM pattern search method is established, which presents essential advantages that allow the construction of a clear methodology for tuning an NMPC with SIB T-S in combination with a CBMD applied to chemical processes;
- The tuning process of an NMPC is optimized utilizing the PSM, which, unlike other optimization methods, does not present randomness, helping the results to be consistently reliable, avoiding at the same time the stagnation in local minimums, performing an exhaustive search for a minimum value, reducing the number of tests in a significant way. Furthermore, control is obtained over the entire spectrum of values

with the algorithm application without approximating the defined operating points, as most proposals for tuning these control strategies;

- The proposed optimization approach for the NMPC tuning with SIB T-S and CBMD can be a first step for the industry to adopt such techniques, which will allow at a given time to reach the implementation in the processes, as today happens with PID controllers that present acceptable behaviors, however, they can be supported by the alternatives of advanced control when it comes to optimizing the processes.

Based on the results obtained, it can be assured that the PSM is a suitable option for implementation in pH control applications in the alkalization of sugar cane juice. PSM is used to identify the model and subsequent controller tuning that obtains better results for these processes than other authors.

**Author Contributions:** Conceptualization, J.P.-M. and A.T.; investigation, J.P.-M.; writing—original draft preparation, J.P.-M., A.T. and J.H.; writing—review and editing, A.T. and J.H.; funding acquisition, A.T. and J.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was partially supported by the Instituto Tecnológico Metropolitano (ITM) of Medellín, under project—P21101 “Fortalecimiento y consolidación del grupo Automática, Electrónica y Ciencias Computacionales para responder a las necesidades de las Industrias 4.0” and has been funded by the Facultad de Ciencias Económicas y Administrativas de la Universidad de Bogotá Jorge Tadeo Lozano.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Kasi, A.; Velagi, J.; Osmanovi, A. Design of NMPC-Based Framework for Mobile Robot Motion in Unstructured Environments. In Proceedings of the 2018 International Symposium ELMAR, Zadar, Croatia, 16–19 September 2018; pp. 183–186.
2. Roy, K.; Bhati, J.; Paruya, S. Evaluating Successive Linearization in NMPC for Controlling Oscillations in Boiling Channel. In Proceedings of the 18th International Conference on Control, Automation and Systems (ICCAS), Institute of Control, Robotics and Systems–ICROS, PyeongChang, Korea, 17–20 October 2018; pp. 1260–1264.
3. Gros, S.; Quirynen, R.; Diehl, M. An Improved Real-time Economic NMPC Scheme for Wind Turbine Control Using Spline-Interpolated Aerodynamic Coefficients. In Proceedings of the 53rd IEEE Conference on Decision and Control, Los Angeles, CA, USA, 15–17 December 2014. [[CrossRef](#)]
4. Guo, N.; Lenzo, B.; Zhang, X.; Zou, Y.; Zhai, R.; Zhang, T. A Real-Time Nonlinear Model Predictive Controller for Yaw Motion Optimization of Distributed Drive Electric Vehicles. *IEEE Trans. Veh. Technol.* **2020**, *69*, 4935–4946. [[CrossRef](#)]
5. Herrera, J.; Ibeas, A.; Alcántara, S.; Vilanova, R. Identification and adaptive control of delayed unstable systems. In Proceedings of the 2010 IEEE International Symposium on Intelligent Control, Yokohama, Japan, 8–10 September 2010; pp. 767–772. [[CrossRef](#)]
6. Lemonge, A.; Barbosa, H. A new adaptive penalty scheme for genetic algorithms. *Inf. Sci.* **2003**, *3*, 215–251.
7. Schutte, J.; Groenwold, A. Sizing design of truss structures using particle swarms. *Struct. Multidiscip. Optim.* **2003**, *25*, 261–269. [[CrossRef](#)]
8. Herrera, J.; Ibeas, A.; de la Sen, M. Identification and control of integrative MIMO systems using pattern search algorithms: An application to irrigation channels. *Eng. Appl. Artif. Intell.* **2013**, *26*, 334–346. [[CrossRef](#)]
9. Alcántara Cano, S.; Ibeas Hernández, A.; Aurelio Herrera, J.; Vilanova Arbos, R.; Pedret i Ferre, C. Multi-Model Smith Predictor Based Control of Multivariable Systems with Uncertain Bounded External Delays. *IEEE Lat. Am. Trans.* **2009**, *7*, 42–53. [[CrossRef](#)]
10. Demaya, B.; Palm, R.; Boverie, S.; Titli, A. Multilevel qualitative and numerical optimization of fuzzy controller. In Proceedings of the 1995 IEEE International Conference on Fuzzy Systems, Yokohama, Japan, 20–24 March 1995; Volume 3, pp. 1149–1154. [[CrossRef](#)]
11. Zangeneh, M.; Aghajari, E.; Forouzanfar, M. A Review on Optimization of Fuzzy Controller Parameters in Robotic Applications. *IETE J. Res.* **2020**, 1–10. [[CrossRef](#)]
12. Goldberg, D.E. *Genetic Algorithms in Search, Optimization and Machine Learning*; Addison Wesley: Boston, MA, USA; 1989; pp. 95–99, ISBN 0201157675.
13. Adeli, H.; Cheng, N.T. Integrated Genetic Algorithm for Optimization of Space Structures. *J. Aerosp. Eng.* **1993**, *6*, 315–328. [[CrossRef](#)]

14. Norapat, N.; Bureerat, S. Simultaneous topology, shape and sizing optimisation of a three-dimensional slender truss tower using multiobjective evolutionary algorithms. *Comput. Struct.* **2011**, *89*, 2531–2538.
15. Yang, J.; Chee Kiong, S. Structural Optimization by Genetic Algorithms with Tournament Selection. *J. Comput. Civ. Eng.* **1997**, *11*, 195–200. [[CrossRef](#)]
16. Ebenau, G.; Rottschäfer, J. An advanced evolutionary strategy with an adaptive penalty function for mixed-discrete structural optimisation. *Adv. Eng. Softw.* **2005**, *36*, 29–38. [[CrossRef](#)]
17. Li, D.; Guo, W.; Wang, L.; Chen, M. Particle swarm optimization-based solution updating strategy for biogeography-based optimization. In Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC), Vancouver, BC, Canada, 24–29 July 2016; pp. 455–459. [[CrossRef](#)]
18. Feng, C.S.; Cong, S.; Feng, X.Y. A new adaptive inertia weight strategy in particle swarm. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC 2007), Singapore, 25–28 September 2007; pp. 4186–4190.
19. Yasuda, K.; Yazawa, K. Parameter self-adjusting strategy for Particle Swarm Optimization. In Proceedings of the 2011 11th International Conference on Intelligent Systems Design and Applications, Cordoba, Spain, 22–24 November 2011; pp. 265–270. [[CrossRef](#)]
20. Dai, Y.; Lou, Y.; Lu, X. A Task Scheduling Algorithm Based on Genetic Algorithm and Ant Colony Optimization Algorithm with Multi-QoS Constraints in Cloud Computing. In Proceedings of the 2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, China, 26–27 August 2015; Volume 2, pp. 428–431. [[CrossRef](#)]
21. Lee, M.G.; Yu, K.M. Dynamic Path Planning Based on an Improved Ant Colony Optimization with Genetic Algorithm. In Proceedings of the 2018 IEEE Asia-Pacific Conference on Antennas and Propagation (APCAP), Auckland, New Zealand, 5–8 August 2018; pp. 1–2.
22. Camp, C.; Barron, J. Design of Space Trusses Using Ant Colony Optimization. *J. Struct. Eng.* **2004**, *130*, 741–751. [[CrossRef](#)]
23. Sonmez, M. Artificial Bee Colony algorithm for optimization of truss structures. *Appl. Soft Comput.* **2011**, *11*, 2406–2418. [[CrossRef](#)]
24. Xu, F.; Chen, H.; Gong, X.; Mei, Q. Fast Nonlinear Model Predictive Control on FPGA Using Particle Swarm Optimization. *IEEE Trans. Ind. Electron.* **2016**, *63*, 310–321. [[CrossRef](#)]
25. Hernandez, E.; Castillo, O.; Soria, J. Optimization of fuzzy controllers for autonomous mobile robots using the grey wolf optimizer. In Proceedings of the 2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), New Orleans, LA, USA, 23–26 June 2019; pp. 1–6. [[CrossRef](#)]
26. Martínez, G.E.; Melin, P.; Castillo, O. A new approach for an intuitionistic fuzzy Sugeno integral for decision making. *Notes Intuit. Fuzzy Sets* **2019**, *25*, 41–52. [[CrossRef](#)]
27. Ontiveros-Robles, E.; Melin, P.; Castillo, O.; Gonzalez, J. Design and FPGA Implementation of Real-Time Edge Detectors Based on Interval Type-2 Fuzzy Systems. *J. Mult. Valued Log. Soft Comput.* **2019**, *33*, 295–320.
28. Castillo, O.; Valdez, F.; Soria, J.; Amador-Angulo, L.; Ochoa, P.; Peraza, C. Comparative study in fuzzy controller optimization using bee colony, differential evolution, and harmony search algorithms. *Algorithms* **2019**, *12*, 9. [[CrossRef](#)]
29. Bogani, C.; Gasparo, M.G.; Papini, A. Generalized Pattern Search methods for a class of nonsmooth optimization problems with structure. *J. Comput. Appl. Math.* **2009**, *229*, 283–293. [[CrossRef](#)]
30. Liu, L.; Zhang, X. Generalized pattern search methods for linearly equality constrained optimization problems. *ScienceDirect* **2006**, *181*, 527–535. [[CrossRef](#)]
31. Herrera, J.; Ibeas, A.; de la Sen, M.; Alcantara, M.; Serna-Garcés, S. Identification and control of delayed siso systems through Pattern Search Methods. *J. Frankl. Inst. B* **2013**, *350*, 3128–3148.
32. Herrera, J.; Ibeas, A.; De La Sen, M.; Rivera, E.; Peláez, J. Generalized Pattern Search Methods for control of stable, unstable and integrating systems with unknown delay under step input. *Math. Comput. Simul.* **2015**, *115*, 37–48. [[CrossRef](#)]
33. Alvarez, H.; Peña, M. Modelamiento de Sistemas de Inferencia Borrosa Tipo Takagi-Sugeno. *Av. En Sist. Inform.* **2004**, *1*, 1–11.
34. Ibeas, A.; Vilanova, R.; Balaguer, P.; de la Sen, M. Multiple-model based intelligent control techniques for LTI systems with unknown external delays part I: Known rational component. *Int. J. Innov. Comput. Inf. Control* **2008**, *4*, 839–848.
35. Suárez, A.M. Nueva Arquitectura de Control Predictivo Para Sistemas Dinámicos Nolineares Usando Redes Neuronales. Ph.D. Thesis, Tesis de Doctorado en Ciencias de la Ingeniería, Mención Automática, U. de Chile, Santiago, Chile, 1998.
36. Espinosa, J.J.; Vandewalle, J. Predictive Control Using Fuzzy Models. In *Advances in Soft Computing*; Springer: London, UK, 1999; pp. 187–200. [[CrossRef](#)]
37. Isaza, J.A. Evaluación de un Controlador Predictivo Basado en un Modelo Semifísico de Inferencia Borrosa Takagi-Sugeno Con Conjuntos Multidimensionales. Ph.D. Thesis, Universidad Nacional de Colombia, Manizales, Colombia 2012.
38. Hinde, R.; Cooper, D.J. A Pattern-based Approach to Excitation Diagnostics for Adaptive Process Control. *Chem. Eng. Sci.* **1994**, *49*, 1403–1415. [[CrossRef](#)]
39. Shridhar, R.; Cooper, D.J. A Tuning Strategy for Unconstrained Multivariable Model Predictive Control. *Ind. Eng. Chem. Res.* **1998**, *37*, 4003–4016. [[CrossRef](#)]
40. Dougherty, D.; Cooper, D.J. A practical multiple model adaptive strategy for multivariable model predictive control. *Eng. Pract.* **2003**, *11*, 649–664. [[CrossRef](#)]

- 
41. Yamuna, K.; Unbehauen, H. Study of Predictive Controller Tuning Methods \* IL. *Automatica* **1997**, *33*, 2243–2248.
  42. Srinivasarao, P.; Subbaiah, P. Tuning of Nonlinear Model Predictive for qwuadruple tank process. *J. Theor. Appl. Inf. Technol.* **2014**, *67*, 316–326.