

## Article

# Embedded MPC Strategies for ESP-Lifted Oil Wells: Hardware-in-the-Loop Performance Analysis of Nonlinear and Robust Techniques

Bruno A. Santana <sup>1,2,\*</sup> , Victor S. Matos <sup>1</sup> , Daniel D. Santana <sup>3</sup>  and Márcio A. F. Martins <sup>1,3,\*</sup> 

<sup>1</sup> Mechatronics Program, Polytechnic School, Federal University of Bahia, Salvador 40170-110, Brazil; victorsm@ufba.br

<sup>2</sup> Department of Engineering and Computing, State University of Santa Cruz, Ilhéus 45662-900, Brazil

<sup>3</sup> Department of Chemical Engineering, Polytechnic School, Federal University of Bahia, Salvador 40170-110, Brazil; daniel.diniz@ufba.br

\* Correspondence: bruno.aguiar@ufba.br (B.A.S.); marciomartins@ufba.br (M.A.F.M.)

**Abstract:** This paper proposes embedded model predictive control strategies for oil-production processes equipped with electric submersible pump (ESP) installations. The novelty of this paper is the robustness and computational performance analysis of the Robust Infinite-Horizon Model Predictive Controller (RIHMPC) and Nonlinear Model Predictive Controller (NMPC) strategies, which have not yet been documented by the oil and gas exploration and production literature. The proposed method to embed the control laws is flexible with different hardware and is based on automatic code generation, which facilitates the project workflow. Hardware-in-the-loop simulation cases were used to compare the performance of both control strategies embedded in the Teensy 4.1 microcontroller, using key indices for real applications. The results showed that the RIHMPC strategy is a very promising alternative for real-time operation in ESP-lifted oil wells, with overall performance similar to the NMPC controller, even in noisy and plant-model mismatch scenarios, and using only linear models in its formulation.

**Keywords:** artificial lift; electrical submersible pump; zone control; robust model predictive control; nonlinear model predictive control; embedded control



**Citation:** Santana, B.A.; Matos, V.S.; Santana, D.D.; Martins, M.A.F. Embedded MPC Strategies for ESP-Lifted Oil Wells: Hardware-in-the-Loop Performance Analysis of Nonlinear and Robust Techniques. *Processes* **2023**, *11*, 1354. <https://doi.org/10.3390/pr11051354>

Academic Editors: Anthony Rossiter and Maurício Bezerra De Souza Jr.

Received: 23 March 2023

Revised: 8 April 2023

Accepted: 12 April 2023

Published: 28 April 2023



**Copyright:** © 2023 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

### 1.1. Overview of ESP-Lifted Oil-Well System

The ESP is the second most widely used artificial lift method for offshore and onshore oil wells and the first in volume of oil produced [1,2]. The economic success of oil wells is heavily dependent on the availability of their artificial lift equipment, which is related mainly to the frequency of failures in ESP-equipped oil wells due to unfavorable operating conditions during the production process. As outlined in Takacs [3], ensuring the safe and stable operation of ESP-lifted oil wells involves adhering to its nonlinearities and a set of physical and safety constraints. Such ESP systems' constrained multivariable characteristics demand therefore automation and control solutions to ensure proper and efficient operation.

Although advanced process control techniques, e.g., model predictive control (MPC), can systematically deal with the particularities of ESP-lifted oil wells, providing an optimal and safer approach, it is a common practice to adopt a manual operation in fields equipped with this artificial lift method, as highlighted by Krishnamoorthy et al. [4]. In the oil and gas exploration and production (E&P) field, implementing model predictive control (MPC) strategies in an actual ESP facility is a scarce subject, with a few examples treated in the literature, such as Pavlov et al. [5] and Patel et al. [6].

On the other hand, some authors are embedding their techniques in relevant hardware for real-time applications in this field, as Binder et al. [7], Santana et al. [8] and Matos et al. [9]. These formulations are usually based on linear models and adaptive schemes without incorporating stabilizing properties. To work around this, the contribution of this work is to investigate the embedded application of two control strategies for oil-production processes equipped with ESP installations, namely: (i) a robust stabilizing MPC based on a multi-linear model description of uncertainty; (ii) an MPC strategy based on a nonlinear phenomenological model (NMPC). Additionally, aiming at practical applications, these strategies incorporate the ESP operational envelope-type constraints in an implementable and softening zone-control scheme.

### 1.2. State of the Art and Related Works

Regarding MPC-based strategies for ESP-lifted oil wells, Pavlov et al. [5] proposed an MPC controller for a laboratory full-scale ESP-lifted oil-well facility. This controller aimed to track the ESP intake pressure and minimize power consumption by explicitly incorporating a choke valve opening target in the objective function of the controller. A step-response-based linear model was identified from a third-order nonlinear model considering homogeneous fluid under isothermal conditions and choke valve opening around 0–20%. Their formulation considered constraints as a compact convex set: wellhead pressure, ESP intake pressure, ESP motor current, bounds on the ESP rotational speed, and its rate of change. They also included ESP operating envelope-type constraints, limiting the down-thrust and up-thrust regions. Subsequently, Krishnamoorthy et al. [4] proposed another MPC application for such a system, based on the same control objectives of Pavlov et al. [5], but with a data-driven linearized model from a high-fidelity ESP simulator that considers multiphase flow and viscous fluid associated with gas and water under non-isothermal condition. They considered tracking the ESP's reference flow rate instead of explicitly imposing (down and up)-thrust force constraints, resulting in a less computationally intensive optimization problem, even with performance loss. Their results proved to be robust to deal with realistic fluid viscosity changes. Binder et al. [10] included feedforward actions related to the measured disturbance of the plant, such as reservoir pressure, into their MPC formulation to improve the control performance in an ESP-lifted oil-well system.

Binder et al. [7] proposed the first embedded MPC in a programmable logic controller (PLC) for ESP-equipped systems through hardware-in-the-loop (HiL) simulations, in which a nonlinear phenomenological model represented the oil-production process to be controlled. The control law was based on the step-response model identified in a region near 90–100% choke valve opening. They included reference flow rate constraints to indirectly minimize ESP power consumption by regulating the motor current. Furthermore, the first practical MPC large-scale implementation in a real onshore oilfield (more than 100 wells), with multiple ESP-lifted oil wells, was presented by Patel et al. [6]. Their formulation applied a data-driven step-response model and used the rotational speed, the choke valve opening, and the ESP voltage as manipulated variables to track the reference flow rate setpoint, considering practical ESP constraints, achieving a significant power saving (10% to 20%).

Despite ongoing advances, the strategies mentioned so far do not consider either guarantee of stability or feasibility for MPC control laws. Fontes et al. [11] was the first to propose a stabilizing MPC strategy with guaranteed feasibility for such systems. This strategy is based on an infinite-horizon MPC (IHMPC) controller that explicitly includes the down-thrust and up-thrust constraints such as a softening target zone scheme. The stabilizing properties are guaranteed by slacked terminal equality constraints, which also enforces the feasibility of the resulting optimization problem.

Regarding robustness, some developments with adaptive MPC schemes have been observed to extend the operating region and accommodate the typical nonlinearities and uncertainties of the ESP-equipped oilfields. In this scenario, Delou et al. [12] introduced an adaptive MPC control law that uses two step-response linear models, one for a low-gain region with a large choke valve opening and the other for a high-gain region with a

small choke valve opening. However, to enhance the system's robustness and incorporate more dynamic information, Delou et al. [13] proposed a control law that employs multiple linear models, including an updated state estimator, to create a robust approach that can handle missing state variable measurements, as previously suggested by Binder et al. [14]. However, the authors do not explicitly consider down-thrust and up-thrust constraints in their control laws.

To overcome the limitations from [12,13], Santana et al. [15] proposed an adaptive and feasible zone IHMPC scheme based on Fontes et al. [11], applying successive linearization of a nonlinear phenomenological model to accommodate the complexities and uncertainties present in the ESP-equipped systems and directly dealing with the pump envelope constraints. However, this strategy does not have robustly stabilizing properties, despite the performance improvement compared to the linear model-based IHMPC [11]. Matos et al. [9] extended that approach, treating the adaptation of the prediction model by a fuzzy structure, which combines data-based local linear models as a linear parameter-varying model. This technique can be directly embedded in ESP installations since it needs minimum available measured data without requiring phenomenological knowledge or flow rate estimation.

Another promising class of MPC controllers associated with ESP-lifted oil-well systems is the nonlinear MPC (NMPC) strategies. Jordanou et al. [16] designed a nonlinear control law using a data-based echo-state neural network as the prediction model. However, their approach does not explicitly consider ESP operational envelope constraints in their controller formulation. Moreover, the average execution time to solve the NMPC law in a simulation environment is more than five times the adopted sampling time, limiting real-time embedded implementation.

In contrast, Santana et al. [17] presented a zone NMPC formulation that can handle these limitations for ESP-lifted oil-well systems. Their approach explicitly deals with ESP operational envelope constraints via an implementable zone-control scheme. Additionally, the method can accommodate the lack of state measurements by coupling a nonlinear state estimator. The most significant advantage of their approach is its real-time applicability for embedded implementation since the computation time is in the order of milliseconds on the simulation reported, following Allamaa et al. [18], which presented recently an NMPC strategy formulated and validated for a real-time application that suggests its implementation on an embedded system. The results showed a low-level controller working at 25 Hz in a constrained and disturbance application.

With recent advances in industrial MPC implementations, automatic code generation techniques have been increasingly improved, aiming to easily provide a solver implementation that is compact, self-contained, efficient, and possibly customized to the embedded hardware on which it needs to run. In [19], a GRAMPC library was used for the low-level integration of an NMPC in assisted driving and obstacle avoidance. The controller was implemented in an ARM-based microcontroller using entirely the C/C++ language, and validated through Processor-in-the-Loop tests. The author [20] presented the real-time implementation of NMPC on embedded systems and its performance evaluation using ACADO toolkit code generation, and had as a case study the position control of hovercraft. Other code generation tools such as Casadi [21] and IPOPT [22] are also excellent for PC-based environments but not as suitable and ready to port for embedded implementation. Moreover, the ACADO toolkit generates highly efficient C/C++ code but it is generally library dependent.

### 1.3. Contributions

The main contribution of this work is the proposition and systematic analysis of an embedded robust MPC strategy for ESP-lifted oil-well systems from the preceding discussion, not yet documented in the E&P literature. Through a hardware-in-the-loop simulation, its performance is compared against the NMPC scheme proposed by [17]. Both strategies encapsulate the zone-control approach to deal directly with the ESP operational

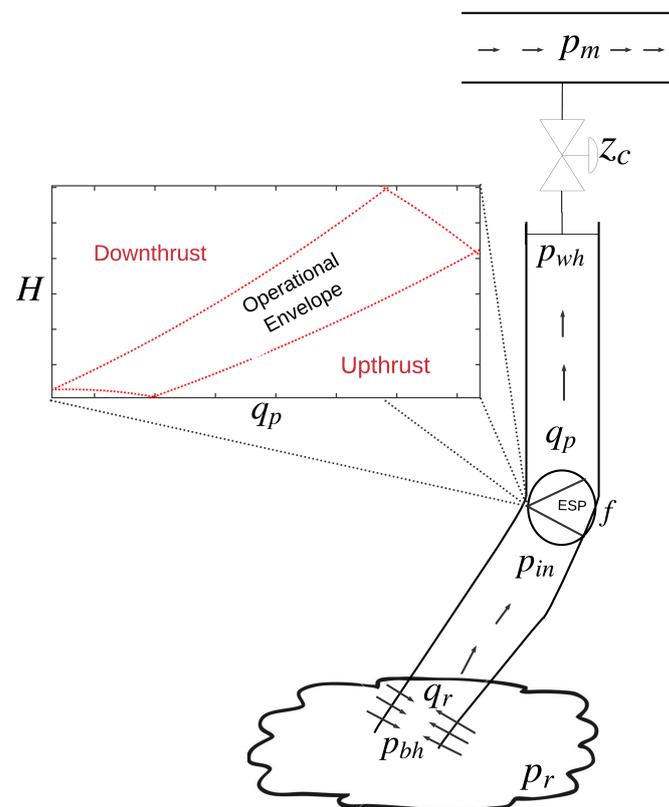
envelope constraints and incorporate economic targets to improve their performance. As a further contribution, the proposed MPC implementation method makes use of the Model-Based Development approach, in which the automatic C/C++ code generation tool facilitates the project workflow and has no dependency on external libraries.

## 2. Robust and Nonlinear MPC Schemes for Embedded Applications on ESP-Equipped Oil-Well Systems

This section presents the challenges of controlling the ESP-lifted oil wells and the proposed MPC techniques designed for this system: (i) the nonlinear MPC approach and (ii) the robust infinite-horizon MPC (RIHMPC) approach.

### 2.1. ESP-Lifted Oil-Well System

Figure 1 depicts a typical ESP-lifted oil-well system. The ESP operates by adjusting the rotational frequency of the pump,  $f$ , and the production choke valve opening,  $z_c$ , to apply additional pressure to the oil and bring it to the surface through the production column and manifold. The fluid properties, such as flow type, the average flow rate in the production column,  $q_p$ , and ESP head dynamics,  $H$ , defines the operational envelope constraint, which consists of time-varying up-thrust and down-thrust constraints. A typical operational envelope curve is shown in Figure 1, usually available from pump manufacturer datasheets. In this study, the average flow rate of the production column was treated as a measured variable. At each sample interval, the maximum head value,  $H_{max}$ , and minimum head value,  $H_{min}$ , were established for the current  $q_p$  value on the operational envelope curve.



**Figure 1.** Typical scheme of an ESP-lifted oil-production system [11].

A typical lumped-element dynamic model for an ESP-lifted oil well was used for simulating the dynamic behavior of the ESP-lifted oil well system. It is represented by a set of differential-algebraic equations as follows [5]:

$$\begin{cases} \frac{dp_{wh}}{dt} = 1.54 \times 10^8 (q_p(t) - q_c(t)) \\ \frac{dp_{bh}}{dt} = 0.8584(p_r(t) - p_{bh}(t) - 3.7 \times 10^8 q_p) \\ \frac{dq_p}{dt} = 5.02 \times 10^{-9} [p_{bh}(t) - p_{wh}(t) - 6.30 \times 10^8 q_p(t)^{1.75} \\ + 9.32 \times 10^3 (H(t) - 1 \times 10^3)] \end{cases} \quad (1)$$

$$\begin{cases} p_{in}(t) = p_{bh}(t) - 1.85 \times 10^8 q_p^{1.75}(t) - 1.9 \times 10^6 \\ H(t) = 0.2664f(t)^2 + 133.09f(t)q_p(t) - 1.41 \times 10^6 q_p^2(t) \end{cases} \quad (2)$$

$$q_c(t) = 2 \times 10^{-5} z_c(t) \sqrt{p_{wh}(t) - p_m(t)} \quad (3)$$

$$\begin{aligned} P(t) = & 4.77 \times 10^{-1} f^3(t) + 1.41 \times 10^3 q_p(t) f^2(t) - 3.74 \times 10^5 q_p^2(t) f(t) \\ & - 3.12 \times 10^9 q_p^3(t) \end{aligned} \quad (4)$$

where  $p_{wh}$ ,  $p_{bh}$ ,  $p_{in}$ ,  $p_m$  and  $p_r$  are the wellhead, bottom hole, intake, manifold, and reservoir pressures in Pa, respectively;  $q_c$  is the flow rate in the production choke in  $m^3/s$ ; and  $P$  is the ESP power in W.

The ESP’s commonly manipulated variables are its rotational frequency and choke valve opening. The control problem involves adjusting them to reach a desired production flow while respecting the safe and physical constraints, especially those defined by the operational envelope. To reach this objective, the intake pressure is taken as a controlled variable that must track a setpoint. Additionally, the head can be treated as another controlled variable, requiring the zone-control approach to softly define its optimal setpoint based on the limits defined by the operational envelope [11].

### 2.2. Nonlinear MPC

This section presents the NMPC approach for ESP systems proposed by Santana et al. [17]: a zone control approach incorporating the model (1)–(4), resulting in a soft optimization problem. In this sense, the constraints of the zone map the operational envelope, as presented in Figure 2, which avoids infeasible solutions for the optimization problem. The optimization problem is given by [17]:

#### Problem 1

$$\begin{aligned} \min_{\Delta \mathbf{u}_k, \mathbf{y}_{sp,k}} V_k = & \sum_{j=1}^p \left\| \hat{\mathbf{y}}(k+j|k) - \mathbf{y}_{sp,k} + \boldsymbol{\epsilon}(k|k) \right\|_{\mathbf{Q}_y}^2 + \sum_{j=0}^{m-1} \left\| \Delta \mathbf{u}(k+j|k) \right\|_{\mathbf{R}}^2 \\ & + \sum_{j=0}^{m-1} \left\| \mathbf{u}(k+j|k) - \mathbf{u}_{tg} \right\|_{\mathbf{Q}_u}^2, \end{aligned} \quad (5)$$

subject to:

$$\Delta \mathbf{u}_k \in \mathcal{U}, \quad (6)$$

$$\mathcal{U} = \left\{ \begin{aligned} & \mathbf{u}_{\min} \leq \mathbf{u}(k-1) + \sum_{i=0}^j \Delta \mathbf{u}(k+i|k) \leq \mathbf{u}_{\max} \\ & -\Delta \mathbf{u}_{\max} \leq \Delta \mathbf{u}(k+j|k) \leq \Delta \mathbf{u}_{\max} \\ & \Delta \mathbf{u}(k+j|k) = 0, \forall j \geq m, \end{aligned} \right\} \quad (7)$$

$$\hat{\mathbf{x}}(k+j|k) = \mathbf{G}(\hat{\mathbf{x}}(k+j-1|k), \mathbf{u}(k+j-1|k), \mathbf{d}(k|k)) \quad (8)$$

$$\hat{\mathbf{y}}(k+j|k) = \mathbf{F}(\hat{\mathbf{x}}(k+j|k)) \quad (9)$$

$$\mathbf{y}_{\min}(k|k) \leq \mathbf{y}_{sp,k} \leq \mathbf{y}_{\max}(k|k) \quad (10)$$

where  $\hat{\mathbf{y}}(k+j|K)$  and  $\hat{\mathbf{x}}(k+j)$  are output and state predictions at time step  $k+j$  from the plant measurement at time step  $k$  ( $\mathbf{y}(k)$ ), respectively;  $\mathbf{u}$  and  $\mathbf{d}$  are the manipulated and disturbance

variables, respectively;  $\Delta \mathbf{u}_k$  is the sequence of control actions calculated along the control horizon, namely  $\Delta \mathbf{u}_k = [\Delta \mathbf{u}(k|k)^\top, \dots, \Delta \mathbf{u}(k+m-1|k)^\top]^\top$ ;  $\epsilon(k|k) = \mathbf{y}(k|k) - \hat{\mathbf{y}}(k|k)$  is a constant output-type disturbance model which results in an offset-free control law and is indicated by the  $\mathbf{y}(k|k)$  feedback dotted-line in Figure 2.  $\mathbf{y}_{sp,k}$  is the artificial setpoint within the proposed zone-control scheme and is a decision variable in the *Problem 1*;  $\mathbf{u}_{tg}$  is the optimizing target of the manipulated variables;  $\mathbf{u}_{max}$ ,  $\mathbf{u}_{min}$ ,  $\Delta \mathbf{u}_{max}$ , are the constraints on manipulated variables and their increments, respectively.  $\mathbf{y}_{max}(k|k)$  and  $\mathbf{y}_{min}(k|k)$  are the upper and lower bounds of the zone control included in constraint (10), evaluated from the operational envelope according to [17].  $p$  and  $m$  are the prediction and control horizons, respectively;  $\mathbf{Q}_y$ ,  $\mathbf{R}$  and  $\mathbf{Q}_u$  are weighting matrices on controlled, control actions and manipulated variables, respectively. Equation (8) represents the discretized nonlinear state equations, Equation (9) the discretized nonlinear output equations. The economic input target ( $\mathbf{u}_{tg}$ ) is also addressed in this control law, without the need for a related constraint.

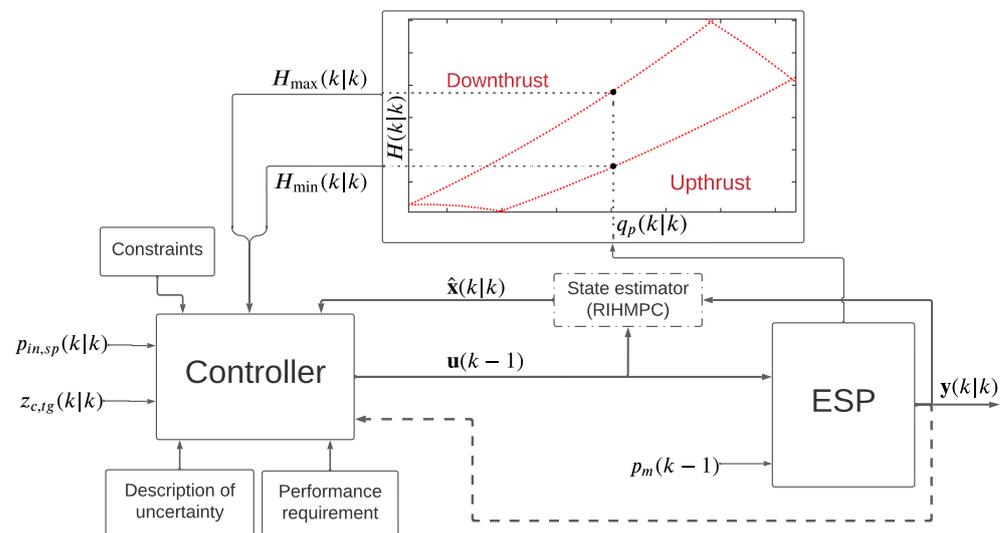


Figure 2. Control scheme elucidating the zone-control strategy.

In this work, the state and other variables needed to evaluate the operational envelope were considered measured, and state estimation was not necessary—presented as a block with a dot-dashed line in Figure 2. In a general scenario, an Extended Kalman Filter (EKF) can be applied to estimate unmeasured variables, for example, the flow rate, as described by Santana et al. [17].

The Figure 2 presents the zone-control scheme for the proposed controllers, where the zone is defined as  $\mathbf{y}_{min}(k) = [p_{in,sp}(k|k), H_{min}(k|k)]$  and  $\mathbf{y}_{max}(k) = [p_{in,sp}(k|k), H_{max}(k|k)]$ , the MVs  $\mathbf{u}(k-1) = [f(k-1), z_c(k-1)]$ , the controlled variables  $\mathbf{y}(k|k) = [p_{in}(k|k), H(k|k)]$ , and the economic target  $\mathbf{u}_{tg}$  is only applied for choke valve opening  $z_{c,tg}(k|k)$ .

One challenging aspect of NMPC techniques is accommodating a nonlinear model to describe the process. Such a task is not trivial in some cases since it requires a mathematical description of phenomenological phenomena. Additionally, the complexity of these models can increase the computational cost and limit real-time applications, so alternatives such as surrogate models or robust MPC strategies must be employed. In this sense, Section 2.3 presents the robust MPC approach investigated in this work.

### 2.3. Robust Infinite-Horizon MPC

An implementable RIHMPC strategy was designed for ESP-lifted oil-well systems applying a multi-model uncertainty description. This implementable RIHMPC was first introduced by Odloak [23] and extended by Gonzalez et al. [24] to the zone-control approach. It imposes non-increasing cost constraints for each model in the set, with only the

selected nominal model having its cost function minimized [24]. The RIHMPC strategy attains robustly stabilizing properties and the feasibility of the optimization problem (implementable feature) by incorporating slack variables [23]. Introducing slack variables into controller formulation is a desired property for practical applications since it can provide a feasible solution to the optimization problem at each time step.

Considering a set of  $L$  linear models of the ESP selected for specific operational points, the operational envelope mapped by setpoint constraints  $(\mathbf{y}_{\min,k}, \mathbf{y}_{\max,k})$ , and input targets  $(\mathbf{u}_{tg,k})$  to track economic performance objectives, the RIHMPC with zone control is presented as the following optimization problem at each time step  $k$ :

*Problem 2*

$$\min_{\Delta \mathbf{u}_k, \mathbf{y}_{sp,k}(\Theta_n=1,\dots,L), \delta_{y,k}(\Theta_n=1,\dots,L), \delta_{u,k}} V_k(\Theta_N) \tag{11}$$

$$\begin{aligned} V_k(\Theta_N) = & \sum_{j=0}^m \left\| \mathbf{y}(k+j|k) - \mathbf{y}_{sp,k}(\Theta_N) - \delta_{y,k}(\Theta_N) \right\|_{\mathbf{Q}_y}^2 + \\ & + \left\| \mathbf{x}_n^{st}(k+m|k) \right\|_{\tilde{\mathbf{Q}}(\Theta_N)}^2 + \sum_{j=0}^{m-1} \left\| \mathbf{u}(k+j|k) - \mathbf{u}_{tg,k} - \delta_{u,k} \right\|_{\mathbf{Q}_u}^2 + \\ & + \sum_{j=0}^{m-1} \left\| \Delta \mathbf{u}(k+j|k) \right\|_{\mathbf{R}}^2 + \left\| \delta_{y,k}(\Theta_N) \right\|_{\mathbf{S}_y}^2 + \left\| \delta_{u,k} \right\|_{\mathbf{S}_u}^2, \end{aligned} \tag{12}$$

subject to:

$$-\Delta \mathbf{u}_{\max} \leq \Delta \mathbf{u}(k+j|k) \leq \Delta \mathbf{u}_{\max}, \quad j = 0, \dots, m-1, \tag{13}$$

$$\mathbf{u}_{\min} \leq \mathbf{u}(k+j|k) \leq \mathbf{u}_{\max}, \quad j = 0, \dots, m-1, \tag{14}$$

$$\mathbf{u}(k+m-1|k) - \mathbf{u}_{tg,k} - \delta_{u,k} = \mathbf{0}, \tag{15}$$

and, for each  $n = 1, \dots, L$ ,

$$\mathbf{y}_{\min}(k|k) \leq \mathbf{y}_{sp,k}(\Theta_n) \leq \mathbf{y}_{\max}(k|k), \tag{16}$$

$$\mathbf{x}_n^s(k+m|k) - \mathbf{y}_{sp,k}(\Theta_n) - \delta_{y,k}(\Theta_n) = \mathbf{0}, \tag{17}$$

$$V_k(\Theta_n) \leq \tilde{V}_k(\Theta_n), \tag{18}$$

where  $\Theta_N$  represents the nominal (or most likely) model of the set of  $L$  models,  $m$  is the control horizon,  $\mathbf{y}(k+j|k)$  are the predictions of the controlled variables at time step  $k+j$  given the information at time step  $k$ ,  $\Delta \mathbf{u}(k+j|k)$  are increments of the manipulated variables.  $(\mathbf{u}_{\min}, \mathbf{u}_{\max})$ ,  $\Delta \mathbf{u}_{\max}$  are the bounds of manipulated variables and increments on manipulated variables, respectively.  $(\mathbf{y}_{\min}, \mathbf{y}_{\max})$  are the zone specification for the controlled variables, and  $\mathbf{y}_{sp,k}$  are the output setpoints within the zone-control scheme imposed by (16).  $\mathbf{Q}_y$  and  $\mathbf{R}$  are the weighting matrices of ( $ny$ ) controlled variables and the increment of ( $nu$ ) manipulated variables, respectively.  $\tilde{\mathbf{Q}}(\Theta_N)$  is the terminal weight detailed.  $\Delta \mathbf{u}_k = [\Delta \mathbf{u}^\top(k|k) \dots \Delta \mathbf{u}^\top(k+m-1|k)]^\top$  is the control actions vector;  $\delta_{y,k} \in \mathbb{R}^{ny}$  and  $\delta_{u,k} \in \mathbb{R}^{nu}$  are the slack variables used to guarantee the feasibility of the optimization Problem 2.  $\mathbf{S}_y \in \mathbb{R}^{ny \times ny}$  and  $\mathbf{S}_u \in \mathbb{R}^{nu \times nu}$  are the weighting matrices of the slack variables.  $\mathbf{u}_{tg}$  are input targets and  $\mathbf{Q}_u$  is their weighting matrix.  $V_k(\Theta_n)$  is the actual cost function value for each model and  $\tilde{V}_k(\Theta_n)$  is the cost function obtained with a solution inherited from Problem 2 at time step  $k-1$  and translated to time  $k$ . More details in [23–25].

The cost-contraction constraint (18) is imposed for all models to guarantee robustly stabilizing properties of the controller. The predictive model must be on the canonical state-space based on the analytical form of the step response of the system outlined in [23], in which  $\mathbf{x}_n^s(k|k)$  is artificial integrating states vector and  $\mathbf{x}_n^{st}(k|k)$  is the stable states vector. The former is used to obtain an offset-free control law, originating from the incremental form of inputs and corresponding to the system outputs at the steady

state. As a result, the terminal constraints (17) and (15) must be incorporated into the formulation to avoid an unbounded cost. Slack variables soften these constraints to guarantee the optimization problem's feasibility. It is worth noting that *Problem 2* is a nonlinear optimization problem. However, it is convex and less computationally intensive compared to non-convex optimization problems commonly found in nonlinear model predictive control formulations.

The artificial integrating states make it necessary to use a linear state estimator only to obtain these unmeasured states, and one Kalman filter was used for each of the  $L$  models—the state estimator block in Figure 2. The model states are considered measured.

### 3. The Implementation Method of Embedded MPC Strategies

This section presents the deployment steps on Teensy 4.1 (<https://www.pjrc.com/store/teensy41.html>, accessed on 6 March 2023) hardware for an embedded control strategy with hardware-in-the-loop (HiL) validation. It also explains how the aforementioned MPC strategies were C-code generated, tested in MATLAB environment, and deployed as a standalone static library in runtime on the microcontroller.

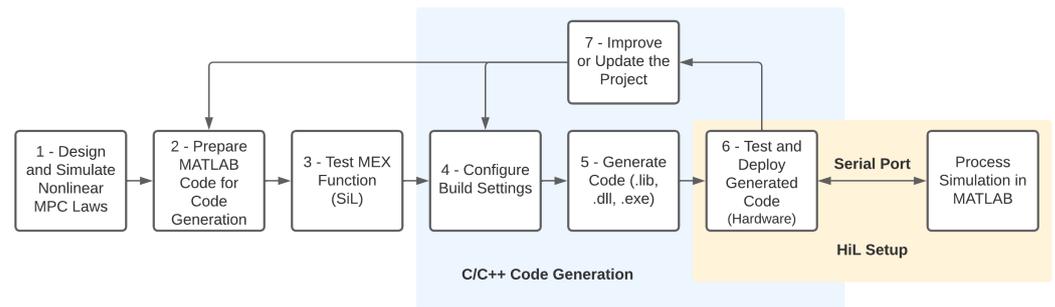
#### 3.1. Code Generation for MPC Controllers

The first step was to write the controller code in MATLAB (R2020a) and solve them using an optimization algorithm. In this work, Sequential Quadratic Programming (SQP) was applied. Following the Model-Based Development approach [26], the next step was executing several simulations to tune the controllers and the optimization algorithm. In this sense, this work focused on defining the maximum limit of iterations of the optimization algorithm and its maximum number of objective function evaluations because they are required for the subsequent automatic generation of C++ code. Furthermore, it is essential to point out that these parameters directly influence the computational times of the embedded code on the hardware. A sensitivity analysis was carried out in several simulated scenarios to ensure the problems converge to the optimal solution.

The next step was generating and verifying the software for the applications referring to the projected control strategies. Then, the functions created for each controller were translated into MEX-type executables using the MATLAB Coder toolbox (<https://www.mathworks.com/products/matlab-coder.html>, accessed on 6 March 2023). Tests were performed with the control executables in the simulated environment, thus characterizing the software-in-the-loop (SiL) stage.

After performing all the previous steps, the C++ code was generated through the MATLAB Coder toolbox and integrated into the hardware that embedded the MPC strategies. Among the extensions available in Coder Toolbox, this work created static libraries (.lib) from the control functions to be integrated into the application embedded in the Teensy 4.1 microcontroller. Thus, it was not necessary to include external libraries in the project, e.g., matrix algebra package, since all mathematical operations related to the solution of *Problems 1* and *2* were encapsulated in the routines generated by the Coder Toolbox. With the embedded control routines, the hardware was tested in closed-loop through the HiL stage: the embedded controller was executed in real time, and only the process remained simulated in MATLAB.

The diagram in Figure 3 summarizes the necessary steps to embed MPC control strategies. This workflow facilitates application improvements and updates using the automatic code generation tool.



**Figure 3.** Diagram of the implementation process of embedded MPC strategies.

### 3.2. Hardware-in-the-Loop Setup

The hardware used in this work was the Teensy 4.1 microcontroller, which has an ARM cortex-M7 processor, 7936K Flash memory, 1024K RAM, 18 analog inputs, 35 PWM pins, 55 I/O pins, and eight serial ports. This hardware is suitable for advanced real-time control applications with computational resources superior to most microcontrollers. Preliminary tests were carried out on other hardware, which did not meet the requirements of this work. Therefore, the choice of Teensy 4.1 converged mainly due to its memory and processing resources.

In the HiL tests, the communication between the microcontroller and MATLAB was performed through the serial port, with a baud rate of 115,200. Therefore, the NMPC and RIHMPC problems were calculated in Teensy 4.1, and the optimal control actions were sent via serial to MATLAB, where the nonlinear model of the ESP-lifted oil-well process was simulated. After simulating the oil-production system, the process outputs were also sent to the microcontroller via the serial port, thus ending one communication cycle.

### 3.3. Solver and Controllers Tuning Parameters

Table 1 summarizes both controllers' tuning parameters. The zone configuration ( $\mathbf{y}_{\min}(k) = [p_{in,sp}(k) \ H_{min}(k)]$ ,  $\mathbf{y}_{\max}(k) = [p_{in,sp}(k) \ H_{max}(k)]$ ) specifies setpoint tracking for the intake pressure and min/max head from the ESP operational envelope. Furthermore, a Runge–Kutta 4th order algorithm was used in the closed-loop system of both MPC controllers for the numerical integration of the nonlinear model of the ESP-equipped oil well, with a sampling time of  $\Delta t = 4$  s.

**Table 1.** Tuning parameters of controllers and constraints on inputs.

	NMPC	RIHMPC
$m$	2	4
$p$	10	$\infty$
$\mathbf{R}$	diag[ $10^{-3}$ $10^{-3}$ ]	diag[10 1]
$\mathbf{Q}_y$	diag[ $10^3$ 10]	diag[ $10^3$ 10]
$\mathbf{Q}_u$	diag[0 10]	diag[0 1]
$\mathbf{S}_y$	-	diag[ $10^2$ 1]
$\mathbf{S}_u$	-	diag[0 10]
$\mathbf{u}_{\min}$	[35 Hz 0 %]	[35 Hz 0 %]
$\mathbf{u}_{\max}$	[65 Hz 100 %]	[65 Hz 100 %]
$\Delta \mathbf{u}_{\max}$	[2 Hz 2 %]	[2 Hz 2 %]

The multi-model set  $\Theta$  that comprises the RIHMPC description of uncertainty was defined by linearization of the nonlinear model (1)–(3) for three different (steady-state) operating points:  $f_{ss,1} = 35.29$  Hz,  $z_{c_{ss,1}} = 14.53\%$ ;  $f_{ss,2} = 55.44$  Hz,  $z_{c_{ss,2}} = 31.58\%$ ;  $f_{ss,3} = 65$  Hz,  $z_{c_{ss,3}} = 98.97\%$ . They were discretized with a sampling time of 4 s, making up  $L = 3$  linear models in the canonical form. These models map a wide operating range of the system, representing probable operating conditions of an ESP-lifted oil-well system. Furthermore, the smallest number of models was chosen to avoid increasing the

computational burden. The typical tuning parameters of the linear Kalman filters were defined as  $\mathbf{Q}_{kf} = 0.5 \cdot \mathbf{I}_{14}$  and  $\mathbf{R}_{kf} = \text{diag}[0.5 \ 0.5]$  in the RIHMPC controller design.

The following main properties were used in the SQP solver compilation: maximum of 30 iterations; maximum of 470 objective function evaluations; primal and dual infeasibility threshold equal to  $1 \times 10^{-6}$ .

#### 4. HiL Application of the Implementable Zone RIHMPC and NMPC Strategies in an ESP-Lifted Oil-Well System

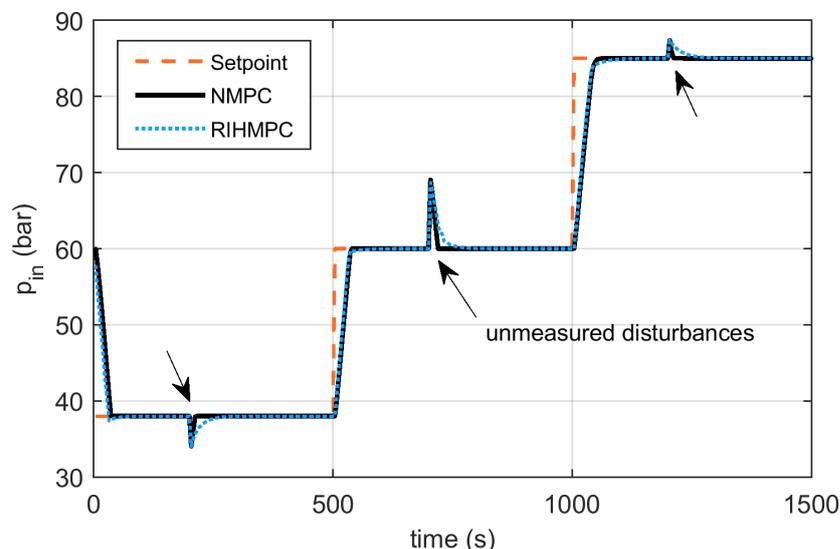
This section investigates the capabilities of NMPC and RIHMPC strategies to perform the following tasks: (i) tracking the intake pressure to regulate the oil production, (ii) systematically treating the ESP head (down and up-thrust) operational envelope to ensure a safe operation, (iii) ensuring the feasibility of the control problem through an implementable control zone scheme, (iv) offset-free compensation of unmeasured disturbance, and (v) tracking targets for the production choke valve opening-like aiming to improve economic performance. For the sake of simplicity, only the results referring to the HiL stage were discussed in detail. Nevertheless, specific aspects of the other methodological steps were highlighted when necessary.

##### 4.1. Control Performance under Setpoint Tracking and Disturbance Compensation

The first test scenario was characterized by three setpoint changes, mapping a wide operating region of the system, and severe variations in manifold pressure ( $p_m$ ). The economic target ( $\mathbf{u}_{tg,k}$ ) was turned off, just by setting  $\mathbf{Q}_u = \text{diag}([0 \ 0])$ , while the other tuning parameters were the same as in Table 1.

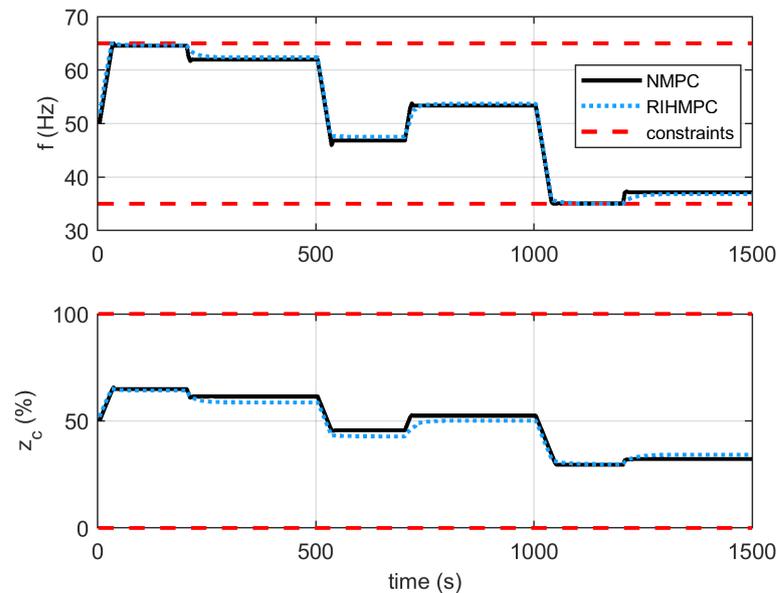
The manifold pressure (unmeasured disturbance) was systematically changed from 20 bar to 10 bar (at 200 s), then to 30 bar (at 700 s) and 35 bar (at 1200 s), while the intake pressure setpoints were defined at 38 bar (0 to 500 s), 60 bar (500 to 1000 s) and 85 bar (1000 to 1500 s). Furthermore, the head setpoint was defined by the optimization problems of the controllers in such a way that it remains within the range delimited by the operational envelope (zone control). The ESP-lifted oil-well system starts at a steady state corresponding to  $\mathbf{u}(0) = [50 \text{ Hz}, 50 \text{ \%}]$  and  $\mathbf{y}(0) = [60 \text{ bar}, 592 \text{ m}]$ . Please note that this scenario represents a case of plant–model mismatch due to the inclusion of unmeasured persistent disturbances.

Figure 4 shows the performance of the NMPC and RIHMPC controllers in the assigned task. Similar dynamic responses were observed during transient periods, and both controllers met the expected requirements related to intake pressure tracking and disturbance compensation.

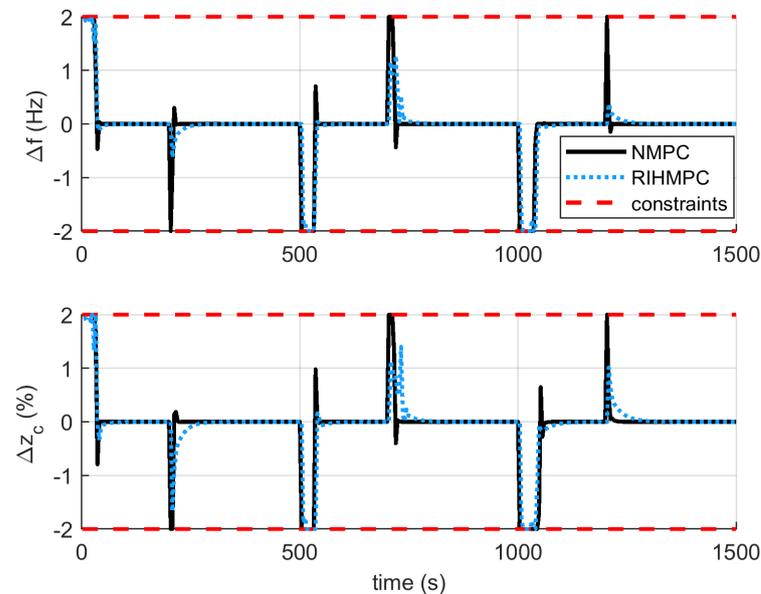


**Figure 4.** Dynamic of the intake pressure under the zone NMPC and RIHMPC controllers.

It is worth noting that considerable effort was devoted to the tuning stage of the controllers to obtain such similar control performance. Even so, the NMPC had a less damped response and drove the intake pressure back to the setpoint faster than the RIHMPC after disturbances, mainly due to the difference in its weights in the movement suppression matrices ( $\mathbf{R}$ ). Regarding the manipulated variables, Figures 5 and 6 show that the control actions did not violate the physical constraints of the actuators detailed in Table 1. The most aggressive actions of the NMPC due to its tuning are demonstrated in Figure 6, where variations of the MVs easily reached saturation during transients.



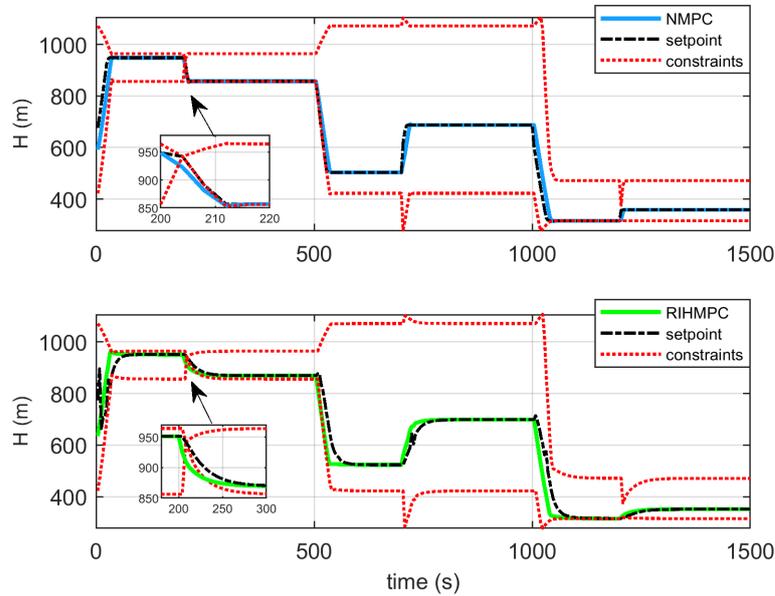
**Figure 5.** Manipulated variables and the input constraints for both controllers.



**Figure 6.** Manipulated variables increments and the respective constraints for both controllers.

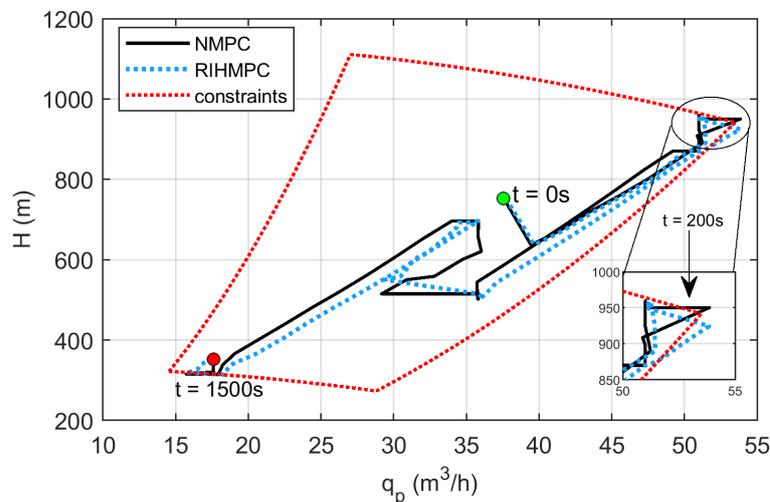
The zone-control scheme is clearly shown in Figure 7, in which the setpoint is a decision variable constrained by the zone. This mechanism softens the optimization problems and contributes to its feasibility. This technique was strategically employed to accommodate the time-varying constraints of the ESP operational envelope since it gives additional degrees of freedom for the system to accommodate disturbances. Evidence of this property can

be seen in the highlighted zoom of Figure 7, where after the disturbance in the manifold pressure at 200 s enters the plant, this was forced to operate momentarily outside the envelope. However, the setpoint remained within the zone in both controllers. After this short time, the control actions compensated for the disturbance and drove the process back to its desired zone.



**Figure 7.** Head zone from the ESP operational envelope and artificial setpoints of the NMPC and RIHMPC controllers.

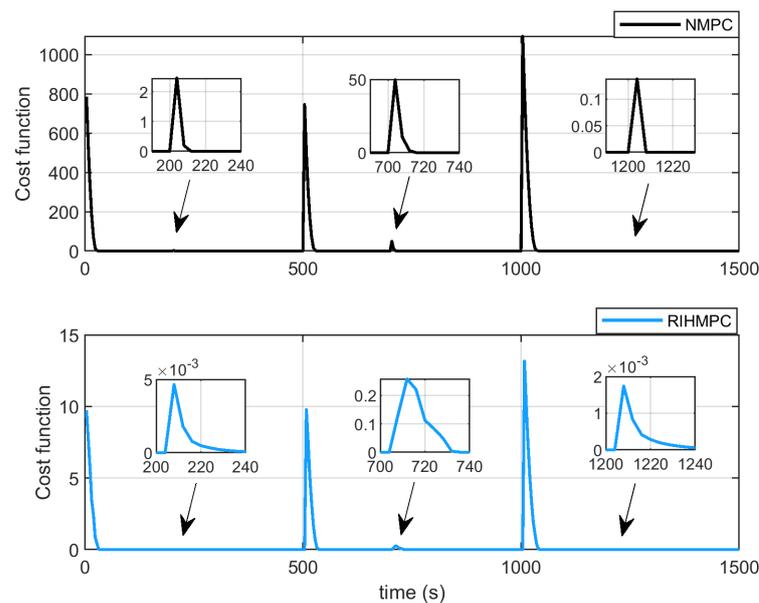
Figure 8 shows the ESP operational envelope, highlighting the constraint violation from another perspective. Despite this occurrence, the system was kept operating within the safe region most of the time, which is crucial to maintain the lifetime of the equipment. In this scenario, the residence time outside the envelope was approximately 8 s for both controllers investigated. In ESP-equipped oilfield installations, keeping the operation close to the up-thrust region is often desirable in order, for example, to maximize production. In such cases, constraining the limits of the operational envelope to a safety margin may be good practice. Nonetheless, the system was pushed to the limits to evaluate the feasibility of these embedded control techniques.



**Figure 8.** Trajectories over the ESP operational envelope for the NMPC and RIHMPC controllers.

Regarding the cost functions of the controllers depicted in Figure 9, their asymptotic behavior toward zero was observed after changing setpoints and the occurrence of disturbances. After a time, such functions reached their origin, which confirms that there was no offset error in the controllers. This property was achieved by the incremental model in the RIHMPC strategy and by the error correction included in the NMPC cost function.

Despite this asymptotic behavior, it is important to highlight that the NMPC formulation does not guarantee nominal or robust stability. On the other hand, the RIHMPC has a robustly stabilizing strategy only for the multi-linear models-based description of uncertainty. Although the proof cannot be taken in this plant-model mismatch scenario, the RIHMPC strategy provided adequate control actions, performing similarly to the NMPC strategy, which encompasses the phenomenological model.



**Figure 9.** Cost function of both controllers.

The computational time performed throughout the HiL simulation by both control strategies on the host hardware Teensy 4.1 is shown in Figure 10. This is one validation to assess whether the controllers can operate in real time or if they need to be redesigned. The NMPC had an average time of 0.1680 s and a maximum time of 0.9505 s throughout this simulation, while the RIHMPC performed an average time of 0.2080 s and a maximum time of 1.3304 s. Naturally, the moments of more significant computational burden occurred during transient periods, in which the SQP solver performed more iterations to converge the NMPC and RIHMPC optimization problems, as shown in Figure 11. This indicates that both projected control laws met the real-time operation requirements with the ESP system, whose sampling time was 4 s. To demonstrate the repeatability of these results, a total of 10 simulations were performed with the same scenario, obtaining the mean ( $\mu$ ) and standard deviation ( $\sigma$ ) of the average and maximum computational times of both controllers, which are shown in Table 2.

Figure 11 shows the number of iterations that the SQP solver performed to converge the NMPC and RIHMPC optimization problems. Only NMPC reached the maximum iteration limit during the transitory period. However, as shown in Figure 12, its number of objective function evaluations was smaller than the ones needed for the RIHMPC. From this result, it is clear that the RIHMPC control law required a computational effort slightly greater than the NMPC to be solved. This was also confirmed by the data in Table 2, in which the average and maximum times were greater for the RIHMPC.

However, it is important to point out that the RIHMPC has robust stability, with the introduction of the cost-contraction constraint (18), which naturally makes the optimization

problem more complex. Still, warm starting the primal variables, the SQP solver turned out to be efficient for real-time operation, handling nonlinearities and converging to the optimal solution. In the aspect of optimality, another important property can be emphasized, the convex control law of the RIHMPC delivers a globally optimal solution, while in the NMPC strategy, there is no such guarantee.

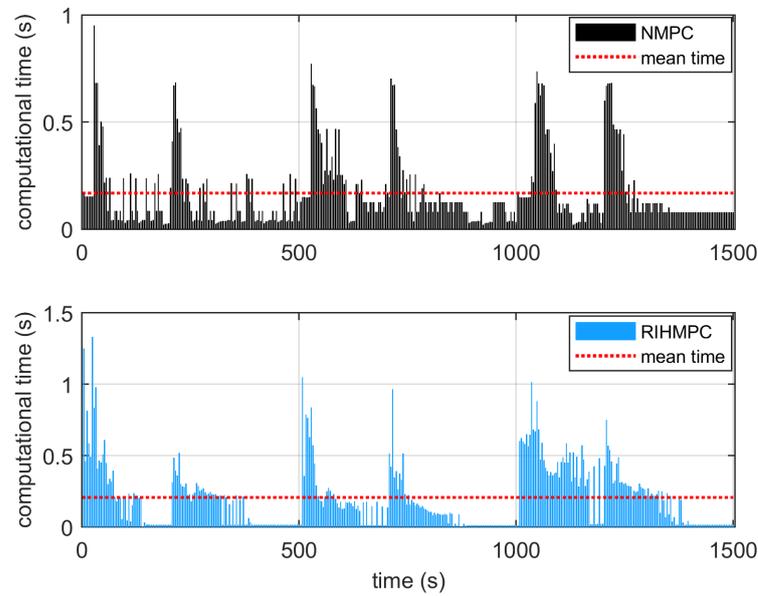


Figure 10. Computational performance between the NMPC and RIHMPC optimization problems.

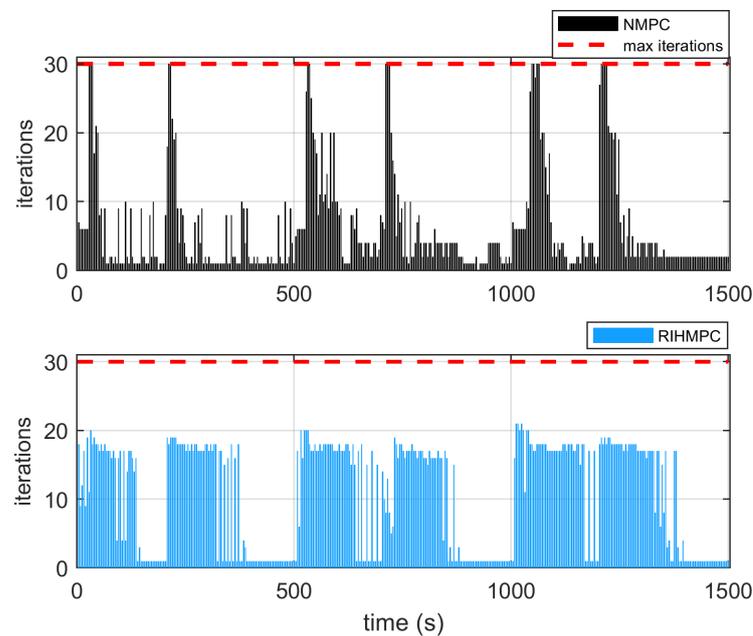
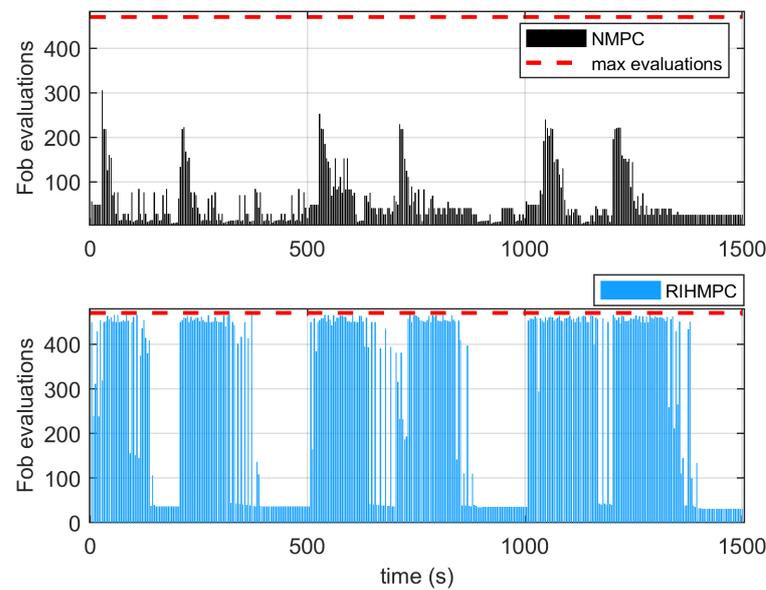


Figure 11. Number of iterations of the NMPC and RIHMPC optimization problems during the simulated scenario.

Table 2. Computational performance obtained for 10 simulations.

Index	NMPC	RIHMPC
avg. time (s)	$\mu = 0.1680$ and $\sigma = 0.0000$	$\mu = 0.2081$ and $\sigma = 0.0002$
max. time (s)	$\mu = 0.9505$ and $\sigma = 0.0000$	$\mu = 1.3303$ and $\sigma = 0.0001$



**Figure 12.** Objective function evaluations in the NMPC and RIHMPC optimization problems execution.

Although the NMPC and RIHMPC strategies presented similar control performance, e.g., computational time, they have very different demands in terms of hardware memory, which can be a determining factor for the feasibility or choice of applications according to available computational resources. The total flash and RAM used in each implementation is summarized in Table 3. The embedded RIHMPC application required about 80% more flash memory and 60% more RAM than the embedded NMPC application. This was mainly due to the need to store and process multidimensional matrices in the RIHMPC implementation, for example, the model matrices for each of the three controller models. In addition, code generation for the SQP solver via MATLAB Coder imposes the use of double-type variables, which also increased the storage memory requirement. As computational resources are not always compatible with such demand, it may be necessary to adopt reduced numerical precision strategies to deploy this controller on more limited hardware, adapting the methodology presented here. The NMPC, in turn, had a leaner coding and required few variables in storage, with the SQP solver and the numerical integration of the nonlinear model (Runge–Kutta 4th-order algorithm) being its main computational burden.

**Table 3.** Amount of memory used in the implementation of NMPC and RIHMPC controllers on Teensy 4.1.

Memory Type	NMPC	RIHMPC
Flash (bytes)	81,916	147,452
RAM (bytes)	149,888	246,432

#### 4.2. Realistic Setpoint/Target Tracking Scenario under Measurement Noises

The following scenario imposed the controllers to a scenario more consistent with the operation of a real ESP-lifted oil well: the control aimed to track setpoints and economic targets in the manipulated variables of interest, in addition to the presence of unmeasured noises in the monitored variables. Therefore, the setpoint changes for the intake pressure were defined at 70 bar (0 to 500 s) and 38 bar (500 to 1000 s), in addition to an opening target of 90% for the choke valve active throughout the HiL simulation. Additionally, the unmeasured noises in the controlled variables had Gaussian distribution  $\mathcal{N}(\mathbf{0}, \mathbf{W})$ , with  $\mathbf{W} = \text{diag}([1.90 \text{ bar}^2, 23.81 \text{ m}^2])$ . Finally, to re-enforce the plant–model mismatch condition on both controllers, at instant 300 s, a systematic disturbance in the manifold

pressure was introduced with an increase of 40% (from 20 to 28 bar) that lasted until the end of the test.

Figure 13 shows the intake pressure dynamics obtained by the controllers in the noisy scenario, where a considerable difference in performance was noted in the transitory period of the first setpoint change. Both NMPC and RIHMPC controllers stabilized the plant, but the NMPC was faster at driving the intake pressure to the desired setpoint with a transient of approximately 24 s, while the RIHMPC took around 140 s. This can be easily explained by the fact that the NMPC encompassed a nonlinear model that captured the plant dynamics very well. In contrast, the most probable RIHMPC model ( $\Theta_1$ ) is suitable for high-gain regions, and the activation of the economic target forced the controller to operate the plant with a low gain (large valve opening choke), as shown in Figure 14. However, it can be emphasized that this deliberate choice was intended to show that the controller remains stable within the multi-model-based uncertainty description.

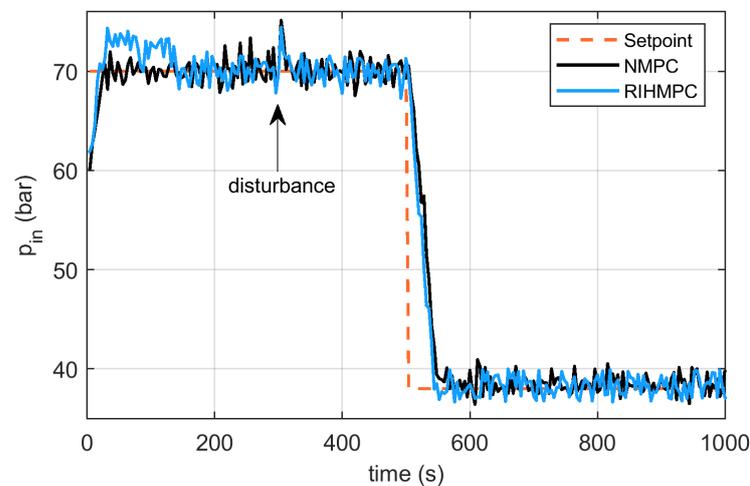


Figure 13. Intake pressure of both controllers under a realistic scenario.

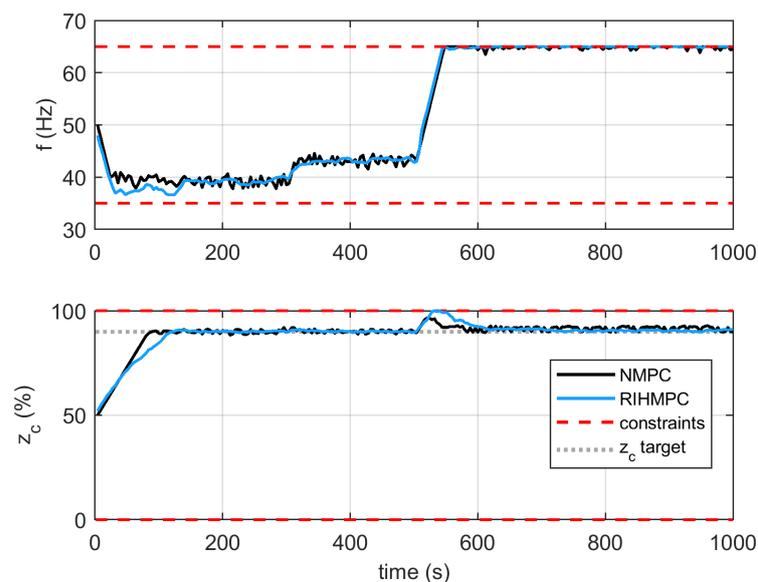
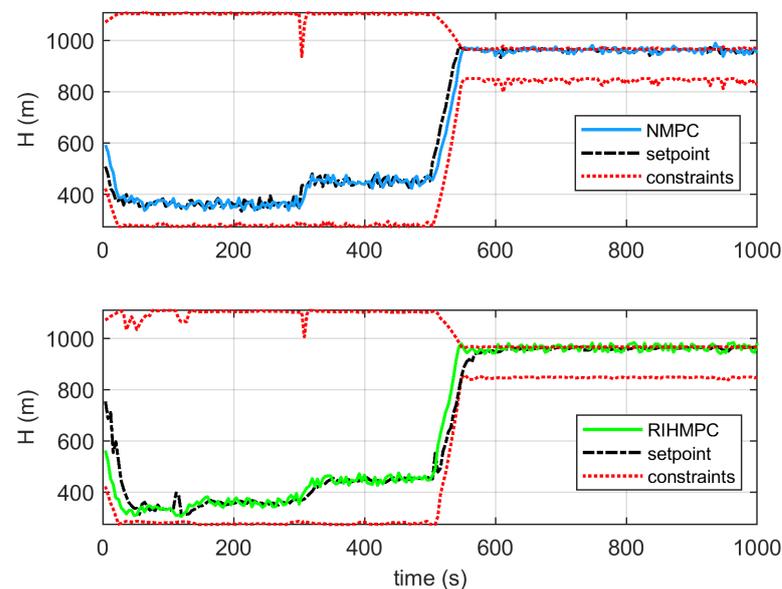


Figure 14. Manipulated variables, the input constraints, and economic target for both controllers under a realistic scenario.

With regard to manipulated variables, Figure 14 shows that the economic target in the choke valve was tracked whenever possible, with necessary deviations during the transitory period, since priority was given to tracking the controlled variables, according to

Table 1. Please note that at 300 s, the system had a degree of freedom to compensate for the disturbance solely by changing the frequency and maintaining the choke valve opening at the target of 90%. In the second setpoint change, a frequency saturation occurred, and the controllers moved the choke valve to stabilize the system but were able to drive it back to its target. In this way, both controllers sought to keep the valve as close as possible to the 90% opening and thus reduce energy losses, whose effectiveness has already been shown in [17].

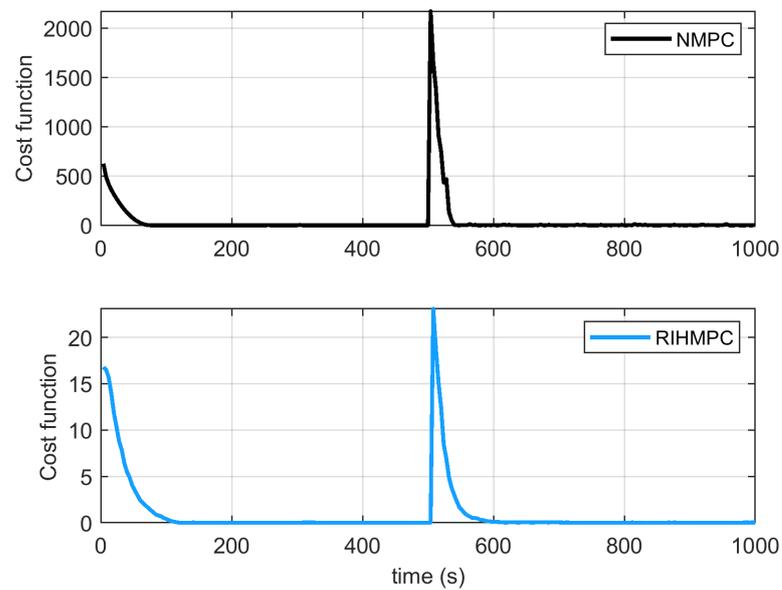
Figure 15 shows the effectiveness of the zone-control strategy employed in both controllers, keeping the ESP head within its operational envelope constraints. After the last setpoint change, the system was operated in the down-thrust limit region with temporary envelope violations due to the noises that affect the process. This emphasizes that using a safety margin in the operational envelope limits can be a good practice to avoid safety violations, especially in noisy scenarios that portray real situations.



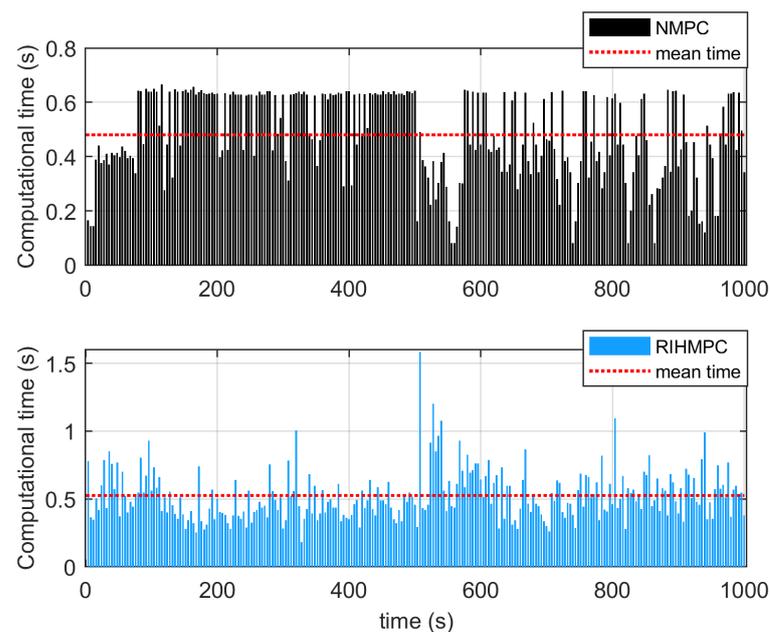
**Figure 15.** Head zone from the ESP operational envelope and artificial setpoints of the NMPC and RIHMPC controllers under a realistic scenario.

The cost functions of the controllers shown in Figure 16 indicated that the NMPC converged faster than the RIHMPC controller in all transient periods. Despite this, the latter delivered a performance very close to the former, which opens a question about the applicability of each one. In this regard, reflecting on obtaining the necessary models for each control law is essential. As is well known, obtaining linear models is generally less expensive and can follow several paths through identification techniques. On the other hand, nonlinear or phenomenological models are more laborious and require knowledge of the physical phenomena that describe the process, which may not be a viable choice for some applications.

Figure 17 shows the computational times obtained on the Teensy 4.1 hardware in this scenario. The NMPC had an average time of 0.4798 s and a maximum time of 0.6658 s throughout this simulation, while the RIHMPC performed an average time of 0.5243 s and a maximum time of 1.5848 s. In this respect, both met the real-time operation requirements with a sampling time of 4 s. Special attention must be given to the RIHMPC controller during transients because it reached the maximum registered value to comply with the predefined tolerances. This analysis was crucial to know if the computational performance of the hardware was compatible with the sampling intervals of the application. To verify the reliability of the results and that the embedded controllers presented repetitive behavior, a total of 10 simulations were performed with the same scenario, obtaining the mean ( $\mu$ ) and standard deviation ( $\sigma$ ) of the average and maximum computational times of both controllers, which are shown in Table 4.



**Figure 16.** Cost function of both NMPC and RIHMPC controllers under a realistic scenario.



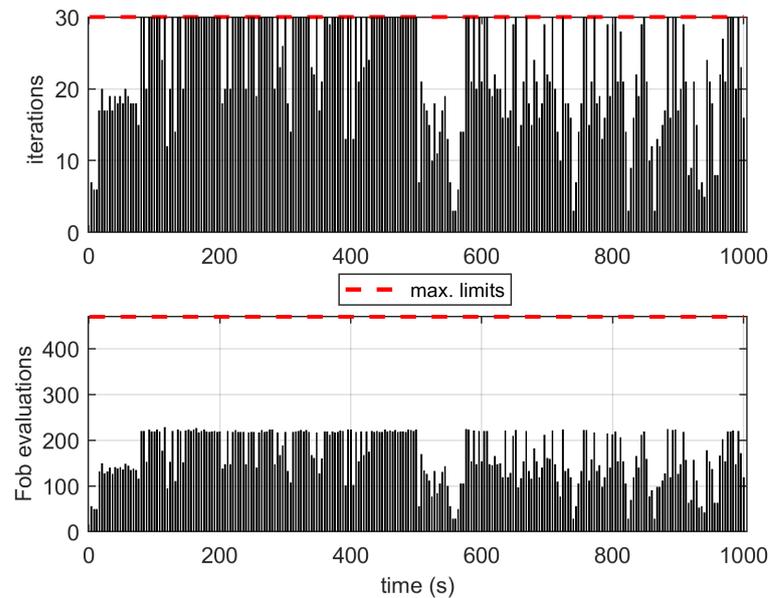
**Figure 17.** Solver computational time of both NMPC and RIHMPC controllers under a realistic scenario.

**Table 4.** Computational performance obtained for a total of 10 simulations in the noisy setpoint/target tracking scenario.

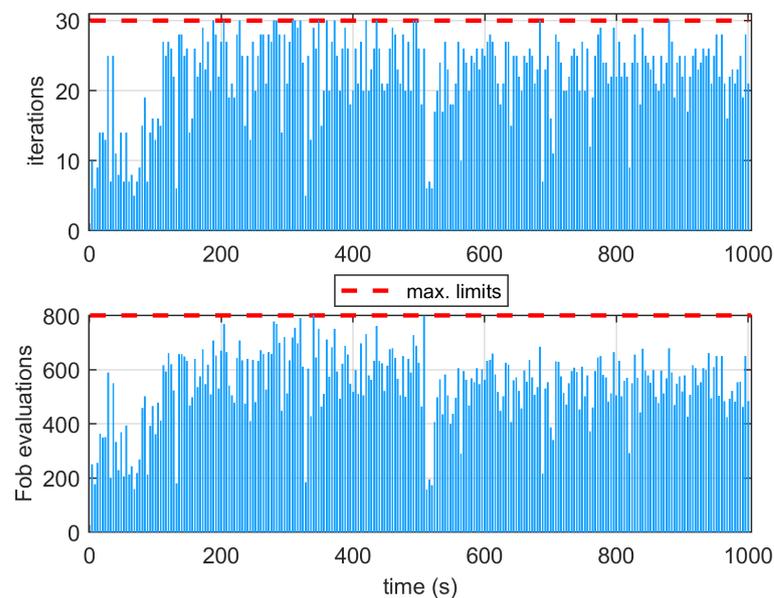
Index	NMPC	RIHMPC
avg. time (s)	$\mu = 0.4679$ and $\sigma = 0.0105$	$\mu = 0.5323$ and $\sigma = 0.0163$
max. time (s)	$\mu = 0.6720$ and $\sigma = 0.0190$	$\mu = 1.5815$ and $\sigma = 0.0097$

Finally, Figures 18 and 19 show the performance of the SQP solver to converge the optimization problems focusing on the number of iterations and objective function evaluations of the NMPC and RIHMPC, respectively. These were the main factors related to the computational times shown in Figure 17. To solve the NMPC control law, the limits

of 30 iterations and 470 objective function evaluations were enough without infeasibility. Similarly, to solve the RIHMPC control law, the limit of 30 iterations was also sufficient. However, increasing the limit of objective function evaluations to 800 turned out to be necessary to avoid infeasibility. Thus, naturally, the computational times recorded in this scenario were greater than those recorded in the previous scenario, but they safely met the real-time requirements of the application.



**Figure 18.** NMPC solver performance under the noisy scenario.



**Figure 19.** RIHMPC solver performance under the noisy scenario.

## 5. Conclusions

This paper dealt with the HiL application of robust and nonlinear MPC strategies embedded in a Teesy 4.1 microcontroller to control an ESP-lifted oil-well system. The novelty of this paper focused on the robustness and computational performance analysis of these MPC controllers with feasibility guarantee and their successful implementation in the limited hardware, which constitute a significant practical contribution to the field of control in E&P.

From the simulated scenarios, it was possible to show the effectiveness of the zone-control scheme employed in both controllers. It successfully accommodates the ESP operational envelope constraints and ensures the feasibility of the optimization problems. The embedded controllers were able to stabilize the intake pressure over a wide operating region, showing that the robust formulation, whose models are more easily obtained, can closely approximate the performance of the nonlinear formulation, even in a noisy and plant–model mismatch scenario.

The computational burden required to solve the control laws met real-time operation requirements. However, the robust formulation required a slightly higher computational time to converge, mainly due to the complexity imposed by the cost-contraction constraint. Regarding hardware memory, the RIHMPC controller used up to 80% more memory than the NMPC controller implementation due to its multi-model structure. Since these resources are not always abundant in industrial hardware, this is a factor that can limit its application. Even so, the RIHMPC formulation proved to be very promising for real-time operation in ESP-lifted oil-well systems, mainly when it is not feasible to obtain an accurate phenomenological model and when the necessary computational resources are available.

**Author Contributions:** B.A.S.: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing—original draft. V.S.M.: Conceptualization, Methodology, Visualization, Writing—original draft. D.D.S.: Conceptualization, Writing—original draft, Writing—review and editing. M.A.F.M.: Conceptualization, Funding acquisition, Project administration, Resources, Supervision, Writing—review and editing. All authors have read and agreed to the published version of the the manuscript.

**Funding:** This work was funded by ANP (Brazilian Agency of Oil, Natural Gas and Biofuels) in the scope of PRH 35.1. (Human Resources Training Program).

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

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors thank the ANP (Brazilian Agency of Oil, Natural Gas and Biofuels) in the scope of PRH 35.1 and CAPES under Finance Code 001, for their financial support. Márcio Martins would also like to thank CNPq under grant 408339/2021-7.

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

## Abbreviations

The following abbreviations are used in this manuscript:

ESP	Electric submersible pump
MPC	Model Predictive Controller
IHMPC	Infinite-Horizon Model Predictive Controller
RIHMPC	Robust Infinite-Horizon Model Predictive Controller
NMPC	Nonlinear Model Predictive Controller
HiL	Hardware-in-the-loop
PLC	Programmable logic controller
QP	Quadratic programming
SQP	Sequential quadratic programming
E&P	Exploration and production
EKF	Extended Kalman Filter
MV	Manipulated variables
I/O	Input output

## References

1. Liang, X.; He, J.; Du, L. Electrical Submersible Pump System Grounding: Current Practice and Future Trend. *IEEE Trans. Ind. Appl.* **2015**, *51*, 5030–5037. [[CrossRef](#)]
2. Zhu, J.; Zhang, H.Q. A Review of Experiments and Modeling of Gas-Liquid Flow in Electrical Submersible Pumps. *Energies* **2018**, *11*, 180. [[CrossRef](#)]
3. Takacs, G. *Electrical Submersible Pumps Manual*, 1st ed.; Gulf Professional Publishing: Houston, TX, USA, 2009.
4. Krishnamoorthy, D.; Bergheim, E.M.; Pavlov, A.; Fredriksen, M.; Fjalestad, K. Modelling and Robustness Analysis of Model Predictive Control for Electrical Submersible Pump Lifted Heavy Oil Wells. *IFAC-PapersOnLine* **2016**, *49*, 544–549. [[CrossRef](#)]
5. Pavlov, A.; Krishnamoorthy, D.; Fjalestad, K.; Aske, E.; Fredriksen, M. Modelling and model predictive control of oil wells with Electric Submersible Pumps. In Proceedings of the 2014 IEEE Conference on Control Applications (CCA), Nice, France, 8–10 October 2014; pp. 586–592. [[CrossRef](#)]
6. Patel, K.; Bakhurji, A.; Salloum, H.; Kim, H.; Winarno, M.; Mubarak, S. Use of advanced process control for automating conventional oilfield operations. In Proceedings of the SPE Kingdom of Saudi Arabia Annual Technical Symposium and Exhibition 2018 (SATS 2018), Dammam, Saudi Arabia, 23–26 April 2018; p. 2020. [[CrossRef](#)]
7. Binder, B.J.; Kufoalor, D.K.; Pavlov, A.; Johansen, T.A. Embedded model predictive control for an electric submersible pump on a programmable logic controller. In Proceedings of the 2014 IEEE Conference on Control Applications (CCA 2014), Nice, France, 8–10 October 2014. [[CrossRef](#)]
8. Santana, B.A.; Fontes, R.M.; Martins, M.A. Controlador Preditivo de Horizonte Infinito Embarcado: Aplicação Hardware-in-the-loop a um Sistema de Bombeio Centrífugo Submerso. In Proceedings of the 15th Simpósio Brasileiro de Automação Inteligente, Online, 17–21 October 2021. [[CrossRef](#)]
9. Matos, V.S.; Santana, B.A.; Chagas, T.P.; Martins, M.A. Embedded predictive controller based on fuzzy linear parameter-varying model: A hardware-in-the-loop application to an ESP-lifted oil well system. *Digit. Chem. Eng.* **2022**, *5*, 100054. [[CrossRef](#)]
10. Binder, B.J.; Johansen, T.A.; Imsland, L. Improved predictions from measured disturbances in linear model predictive control. *J. Process. Control* **2019**, *75*, 86–106. [[CrossRef](#)]
11. Fontes, R.; Costa, E.; Abreu, O.L.; Martins, M.; Schnitman, L. On application of a zone IHMPC to an ESP-lifted oil well system. In Proceedings of the Anais do Congresso Brasileiro de Automática 2020, Online, 23–26 November 2020. [[CrossRef](#)]
12. Delou, P.d.A.; de Azevedo, J.P.; Krishnamoorthy, D.; de Souza, M.B.; Secchi, A.R. Model Predictive Control with Adaptive Strategy Applied to an Electric Submersible Pump in a Subsea Environment. *IFAC-PapersOnLine* **2019**, *52*, 784–789. [[CrossRef](#)]
13. Delou, P.; Souza, M.; Secchi, A.R. A Robust Adaptive MPC coupled with Kalman Filter for an Electric Submersible Pump System: A multi-model approach. In Proceedings of the I Brazilian Congress on Process Systems Engineering (PSE-BR 2019), Rio de Janeiro, Brazil, 20–22 May 2019; pp. 435–438.
14. Binder, B.J.; Pavlov, A.; Johansen, T.A. Estimation of flow rate and viscosity in a well with an electric submersible pump using moving horizon estimation. *IFAC-PapersOnLine* **2015**, *28*, 140–146. [[CrossRef](#)]
15. Santana, B.A.; Fontes, R.M.; Schnitman, L.; Martins, M.A. An Adaptive Infinite Horizon Model Predictive Control Strategy Applied to an ESP-lifted Oil Well System. *IFAC-PapersOnLine* **2021**, *54*, 176–181. [[CrossRef](#)]
16. Jordanou, J.P.; Osnes, I.; Hernes, S.B.; Camponogara, E.; Antonelo, E.A.; Imsland, L. Nonlinear Model Predictive Control of Electrical Submersible Pumps based on Echo State Networks. *Adv. Eng. Informat.* **2022**, *52*, 101553. [[CrossRef](#)]
17. Santana, B.A.; Fontes, R.M.; Martins, M.A. An implementable zone NMPC applied to an ESP-lifted oil well system: Handling the lack of measurements with nonlinear state estimator coupling. *J. Pet. Sci. Eng.* **2022**, *216*, 110816. [[CrossRef](#)]
18. Allamaa, J.P.; Listov, P.; Van Der Auweraer, H.; Jones, C.; Son, T.D. Real-time Nonlinear MPC Strategy with Full Vehicle Validation for Autonomous Driving. In Proceedings of the 2022 American Control Conference (ACC), Atlanta, GA, USA, 8–10 June 2022; pp. 1982–1987. [[CrossRef](#)]
19. Dini, P.; Saponara, S. Processor-in-the-Loop Validation of a Gradient Descent-Based Model Predictive Control for Assisted Driving and Obstacles Avoidance Applications. *IEEE Access* **2022**, *10*, 67958–67975. [[CrossRef](#)]
20. Adhau, S.; Patil, S.; Ingole, D.; Sonawane, D. Implementation and Analysis of Nonlinear Model Predictive Controller on Embedded Systems for Real-Time Applications. In Proceedings of the 2019 18th European Control Conference (ECC), Naples, Italy, 25–28 June 2019; pp. 3359–3364. [[CrossRef](#)]
21. Andersson, J.; Gillis, J.; Horn, G.; Rawlings, J.; Diehl, M. CasADi: A software framework for nonlinear optimization and optimal control. *Math. Program. Comput.* **2019**, *11*, 1–36. [[CrossRef](#)]
22. Wächter, A.; Biegler, L. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program. Comput.* **2006**, *106*, 25–27. [[CrossRef](#)]
23. Odloak, D. Extended robust model predictive control. *AIChE J.* **2004**, *50*, 1824–1836. [[CrossRef](#)]
24. González, A.; Marchetti, J.; Odloak, D. Robust model predictive control with zone control. *IET Control Theory Appl.* **2009**, *3*, 121–135. [[CrossRef](#)]

25. Martins, M.A.; Odloak, D. A robustly stabilizing model predictive control strategy of stable and unstable processes. *Automatica* **2016**, *67*, 132–143. [[CrossRef](#)]
26. Forrai, A. *Embedded Control System Design: A Model Based Approach*; Springer: Berlin/Heidelberg, Germany, 2012.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.