

From Theory to HIL Testing: Practical Implementation of Centralized TITO PID Controllers

Anna Paula V. de A. Aguiar

Federal University of Campina Grande - UFCG

Pericles Rezende Barros

Federal University of Campina Grande - UFCG

George Acioli Junior

`georgeacioli@dee.ufcg.edu.br`

Federal University of Campina Grande - UFCG

Julian Gutierrez

National University of the South - UNS

Research Article

Keywords: Centralized controller, Practical implementation, Simplified decoupler, Decentralized controller, PID control

Posted Date: October 14th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11142650/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

From Theory to HIL Testing: Practical Implementation of Centralized TITO PID Controllers

Anna Paula V. de A. Aguiar¹, Péricles Rezende Barros¹,
George Acioli Júnior¹, Julián Gutiérrez²

^{1*}Department of Electrical Engineering, Federal University of Campina
Grande - UFCG, R. Aprígio Veloso, Campina Grande, 58429-900,
Paraíba, Brazil.

²Department, National University of the South - UNS, Av. Alem, Bahía
Blanca, B8000, Provincia de Buenos Aires, Argentina.

Contributing authors: anna.aguiar@ee.ufcg.edu.br;
prbarros@dee.ufcg.edu.br; georgeacioli@dee.ufcg.edu.br;
julian.gutierrez@eq.ufcg.edu.br;

Abstract

The paper shows how centralized proportional-integral-derivative (PID) controllers for two-input two-output (TITO) processes can be implemented in practice. This is achieved by changing the centralized structure to a decentralized controller plus simplified decoupler. This reformulation, whose mathematical equivalence is demonstrated, aims to overcome the implementation complexity of centralized controller in industrial systems (such as PLCs) without compromising performance or robustness. The simulated confirm that the alternative structure achieves performance and robustness almost identical to the traditional centralized controller. Furthermore, experimental results were obtained in a separation vessel using a hardware-in-the-loop environment, making this implementation recommended for practical application.

Keywords: Centralized controller, Practical implementation, Simplified decoupler, Decentralized controller, PID control

1 Introduction

Multivariable (MIMO) processes are characterized by coupling between variables. This coupling affects the performance of the control loop. There are three proportional-integral-derivative (PID) control structures for these processes: centralized control, decentralized controller, and decentralized controller plus decoupler.

The centralized controller is based on a full control matrix in which each element corresponds to a PID controller. This structure enables the proper handling of interactions between multiple variables. Although there are many centralized controller design techniques [1–3] it should only be implemented when justified. Some disadvantages: more complex design, greater sensitivity to modeling errors and uncertainties, and higher implementation complexity [4]. Furthermore, closed-loop stability depends on all loops remaining continuously closed. However, in industrial environments, it is common for some loops to be kept in manual mode or deactivated for maintenance, while others continue in normal operation.

In decentralized, the overall process is decomposed into several single-input single-output (SISO) subsystems, each one designed independently using classical SISO control design techniques. This structure simplifies the design and implementation stages. However, when strong coupling is present, neglecting cross-interactions can lead to degraded performance or even closed-loop instability. To mitigate these effects, several strategies can be adopted, including the application of advanced tuning techniques, the use of sequential loop-closure methodologies [5], and the incorporation of the concept of effective open-loop interactions, as proposed by [4].

One way to combine the benefits of a centralized and decentralized controller is by using a decentralized controller plus decoupler. In this case, a decoupling block is placed between the decentralized controller and the process to compensate for interactions among the variables. This transforms the MIMO system into SISO loops. A review of the properties, advantages, and application domains of different decouplers is presented in [6]. In [7], the frequency comparison between static and dynamic (ideal, simplified, and inverted) decouplers is presented. Loops employing simplified decoupling are nominally stable, in contrast to those using ideal or inverted decoupling. The latter may become unstable—particularly when right-half-plane (RHP) zeros arise in the closed-loop transfer function.

In [8], decentralized controller with a simplified decoupler is employed to regulate the concentration and temperature of a continuously stirred tank reactor (CSTR). Similarly, in [9], the simplified decoupler is used to attenuate process interactions in coupled-tank systems, and a decentralized controller design technique is proposed. In both works, the inclusion of the decoupler yielded stable and robust closed-loop behavior.

Despite the existence of various data-driven PID control design techniques, decouplers are still designed using the process model [10, 11]. For processes with high-order dynamics, it is necessary to approximate the process model [12] to ensure that the decoupler transfer function is realizable and easily implementable within the control system. However, these approximations and model uncertainties affect the decoupler performance.

Thus, it is concluded that centralized controllers achieve superior performance for multivariable processes, with the primary challenges being process model identification and the implementation of this structure in industrial controllers. The first issue is addressed by data-driven design techniques found in the literature; however, the second issue remains unresolved.

In this context, this article presents a methodology for implementing a centralized controller on industrial programmable logic controllers (PLCs), using the controllers native function blocks. More specifically, the Rockwell Automation ControlLogix 5000 was used in the example. This is achieved using a decentralized structure complemented by a simplified decoupler. The proposed methodology is applied to two-input two-output (TITO) processes. In the examples provided, a controller design based on the process frequency response [1] is used to design the centralized controller.

This paper is organized as follows. In section 2, the problem statement is presented. In section 3, the practical implementation of the centralized PID controller is presented. A controller design technique is reviewed in section 4. In sections 5 and 6, the simulation example and hardware-in-the-loop results are discussed. Finally, the conclusion is presented in section 7.

2 The problem statement

Consider a closed-loop TITO process as presented in Fig. 1, in which $\mathbf{G}(s)$ is a stable linear time invariant process:

$$\mathbf{G}(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix}, \quad (1)$$

$\mathbf{C}_C(s)$ is a centralized PID TITO controller:

$$\mathbf{C}_C(s) = \begin{bmatrix} C_{C11}(s) & C_{C12}(s) \\ C_{C21}(s) & C_{C22}(s) \end{bmatrix}, \quad (2)$$

$\mathbf{u}(t)$, $\mathbf{y}(t)$, $\mathbf{e}(t)$, and $\mathbf{r}(t)$ are the input, output, error, and reference signal vectors, respectively. Each element of matrix (2) is a PID controller:

$$C_{Cij}(s) = K_{Pij} + \frac{K_{Iij}}{s} + K_{Dij}s, \quad (3)$$

and the proportional, integral, and derivative gains are given by K_P , K_I , and K_D , respectively.

The closed-loop transfer function, or complementary sensitivity function, of the system in Fig. 1 is given by:

$$\mathbf{T}(s) = \mathbf{G}(s)\mathbf{C}(s)(\mathbf{I} + \mathbf{G}(s)\mathbf{C}(s))^{-1} \quad (4)$$

The problem consists of adapting the centralized PI/PID controller $\mathbf{C}_C$ in a manner that enables its practical implementation in a real industrial control system without compromising performance or robustness.

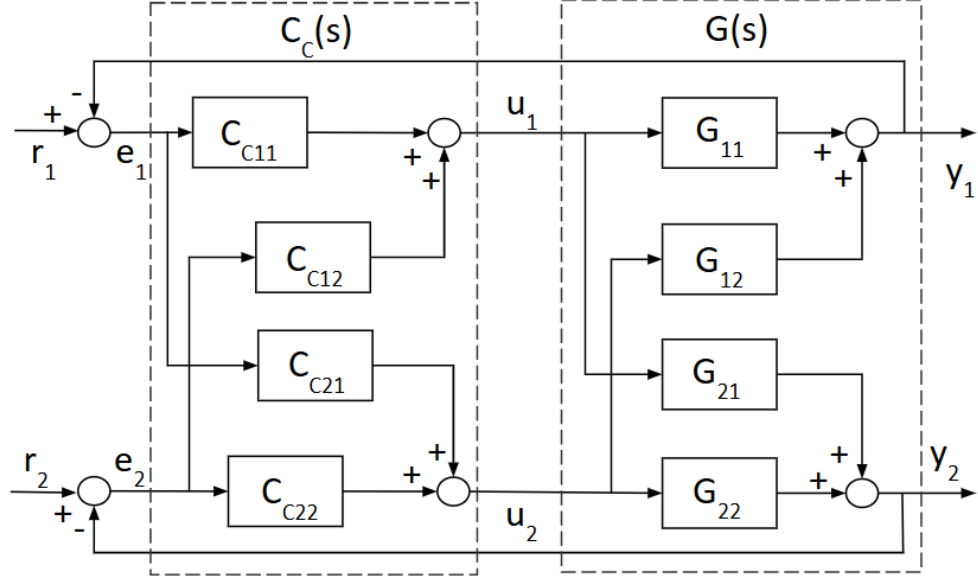


Fig. 1 TITO system with centralized controller

3 Centralized controller practical implementation

Consider a closed-loop TITO process with a centralized controller, as shown in Fig. 1. Consider also the same closed-loop process as shown in Fig. 2, where $\mathbf{C}_D(s)$ is the decentralized controller

$$\mathbf{C}_D(s) = \begin{bmatrix} C_{D1}(s) & 0 \\ 0 & C_{D2}(s) \end{bmatrix}, \quad (5)$$

$\mathbf{D}(s)$ is the simplified decoupler

$$\mathbf{D}(s) = \begin{bmatrix} 1 & D_{12}(s) \\ D_{21}(s) & 1 \end{bmatrix} \quad (6)$$

$\mathbf{u}' = [u'_1 \ u'_2]^T$ and $\mathbf{u} = [u_1 \ u_2]^T$ are the controller output and process input signal vector, respectively. The decentralized controller plus simplified decoupler is given by

$$\mathbf{F}(s) = \mathbf{D}(s)\mathbf{C}_D(s). \quad (7)$$

Analyzing Figs. 1 and 2, it can be stated that the centralized controller is equivalent to the decentralized controller plus simplified decoupler.

$$\mathbf{C}_C = \mathbf{F} = \mathbf{D}\mathbf{C}_D, \quad (8)$$

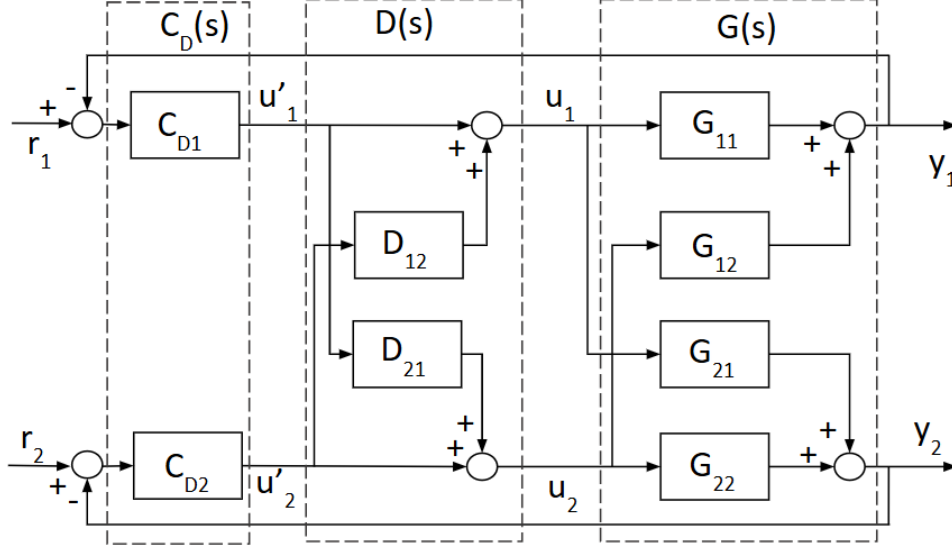


Fig. 2 TITO system with simplified decoupler

Substituting (2), (5) and (6) into (8):

$$\begin{bmatrix} C_{C11} & C_{C12} \\ C_{C21} & C_{C22} \end{bmatrix} = \begin{bmatrix} C_{D1} & C_{D2}D_{12} \\ C_{D1}D_{21} & C_{D2} \end{bmatrix}. \quad (9)$$

From (9), C_{D1} and C_{D2} are given by:

$$C_{D1} = C_{C11}, \quad (10)$$

$$C_{D2} = C_{C22}, \quad (11)$$

By equating the off-diagonal terms of the matrices in (9):

$$C_{C12} = C_{D2}D_{12}, \quad (12)$$

$$C_{C21} = C_{D1}D_{21}. \quad (13)$$

From (12):

$$D_{12} = C_{C12}/C_{D2}. \quad (14)$$

Substituting (11) into (14):

$$D_{12} = C_{C12}/C_{C22}. \quad (15)$$

Similarly, $D_{21} = C_{C21}/C_{C11}$.

Thus, a centralized PID controller $\mathbf{C}_C(s)$ can be written as a product of the decentralized controller $\mathbf{C}_D(s)$ and the simplified decoupler $\mathbf{D}(s)$, where:

$$C_{D1}(s) = C_{C11}(s), \quad (16)$$

$$C_{D2}(s) = C_{C22}(s), \quad (17)$$

$$D_{12}(s) = \frac{C_{C12}(s)}{C_{C22}(s)}, \quad (18)$$

$$D_{21}(s) = \frac{C_{C21}(s)}{C_{C11}(s)}. \quad (19)$$

To achieve better performance in the multivariable control loop, it is recommended to use techniques that do not rely on a parametric process model, that is, data-driven techniques such as the one presented in the following section.

4 Centralized controller design

The technique used to design the centralized controllers in this paper was presented in [1]. The centralized PID controller is designed by solving an optimization problem with constraints. The problem is given by:

$$\begin{aligned} & \min_{\theta} \lambda \\ & \text{subject to:} \\ & \begin{bmatrix} \mathbf{I} & (\mathbf{G}(j\omega)\mathbf{C}(j\omega, \theta) - \mathbf{L}_d(j\omega))^* \\ \mathbf{G}(j\omega)\mathbf{C}(j\omega, \theta) - \mathbf{L}_d(j\omega) & \lambda\mathbf{I} \end{bmatrix} \geq \mathbf{0}, \\ & \begin{bmatrix} \mathbf{I} & \Psi(j\omega, \theta)^* \\ \Psi(j\omega, \theta) & \mathbf{I} \end{bmatrix} \geq \mathbf{0}, \forall \omega \in \Omega. \end{aligned} \quad (20)$$

for $\Omega = [\omega_1, \omega_2, \dots, \omega_N]$ a finite set of frequencies and $\omega_1 > 0$, where $\Psi(j\omega, \theta) = (\mathbf{L}(j\omega, \theta) - \mathbf{L}_d(j\omega))\mathbf{S}_d(j\omega)$, $\mathbf{S}_d(s) = (\mathbf{I} + \mathbf{L}_d(s))^{-1}$, $\theta = [\mathbf{K}_P \ \mathbf{K}_I \ \mathbf{K}_D]$, $\mathbf{C}(s, \theta)$ is a centralized or decentralized controller, $\mathbf{L}(s, \theta)$ is a loop gain transfer function, and $\mathbf{L}_d(s, \theta)$ is a desired loop gain transfer function.

To solve the problem, the reference model of the loop gain transfer function and the process frequency response are required. The desired loop gain transfer function ($\mathbf{L}_d(j\omega)$) is defined from the closed-loop transfer function ($\mathbf{T}_d(j\omega)$):

$$\mathbf{L}_d(j\omega) = \mathbf{T}_d(j\omega)(\mathbf{I} - \mathbf{T}_d(j\omega))^{-1}. \quad (21)$$

The frequency response of the process ($\mathbf{G}(j\omega)$) can be obtained from the process model or from the input ($\mathbf{u}$) and output ($\mathbf{y}$) data by applying the Fourier transform. This technique can be used to design centralized and decentralized PID controllers.

5 Simulation example

The first step is to design the centralized controller using the technique described in section 4. The optimization problem (20) is solved using the CVX framework [13]. Next, the presented procedure was used.

To better analyze the results, in addition to the standard centralized controller and the proposed implementation, a decentralized controller was also implemented. The latter was designed considering the frequency response of the decoupled process ($\mathbf{G}(s)\mathbf{D}(s)$), with the decoupler parameters derived from the estimated process model.

This decoupler was designed based on the approximate first-order plus time delay (FOPTD) model of the process:

$$D_{12}(s) = -\frac{\hat{G}_{12}(s)}{\hat{G}_{11}(s)} = k_{d12} \frac{\tau_{n12}s+1}{\tau_{d12}s+1}, \quad (22)$$

$$D_{21}(s) = -\frac{\hat{G}_{21}(s)}{\hat{G}_{22}(s)} = k_{d21} \frac{\tau_{n21}s+1}{\tau_{d21}s+1}, \quad (23)$$

where k_{dij} is a gain, τ_{nij} is a lead time in seconds (numerator time constant), and τ_{dij} is a lag time in seconds (denominator time constant).

To analyze the performance of the implemented structures, the performance indices used are mean absolute error (MAE), mean total value of the control signal (MTV) and mean absolute error with respect to the reference model output (M_d), respectively, are given by:

$$\text{MAE} = \frac{1}{N} \sum_{k=1}^N |\mathbf{r}(k) - \mathbf{y}(k)|, \quad (24)$$

$$\text{MTV} = \frac{1}{N} \sum_{k=1}^N |\mathbf{u}(k+1) - \mathbf{u}(k)|, \quad (25)$$

$$M_d = \frac{1}{N} \sum_{k=1}^N |\mathbf{y}_d(k) - \mathbf{y}(k)|. \quad (26)$$

where $\mathbf{y}_d(k)$ is the reference model output.

Furthermore, in the simulation results, the sensitivity function maximum singular value curve is used to show the designed closed-loop robustness. Lower peak values M_S of this curve indicate greater robustness.

5.1 Simulated process

Consider the Wood-Berry binary distillation column process [14]:

$$\mathbf{G}_1(s) = \begin{bmatrix} \frac{12.8}{16.7s+1}e^{-s} & \frac{-18.9}{21.0s+1}e^{-3s} \\ \frac{6.6}{10.9s+1}e^{-7s} & \frac{-19.4}{14.4s+1}e^{-3s} \end{bmatrix}. \quad (27)$$

The closed-loop transfer function ($\mathbf{T}_d$) considered is a diagonal matrix; the non-zero elements are a FOPTD function:

$$\mathbf{T}_d(s) = \begin{bmatrix} \frac{e^{-1.01s}}{19.28s + 1} & 0 \\ 0 & \frac{e^{-2.99s}}{16.00s + 1} \end{bmatrix}. \quad (28)$$

The process frequency response was obtained from the Fourier transform of the input and output data resulting from applying a pseudorandom binary sequence (PRBS) signal to the input of the open-loop process, Fig. 3. The frequency interval $[0.017-3.04]$ with 1800 linearly spaced points was used.

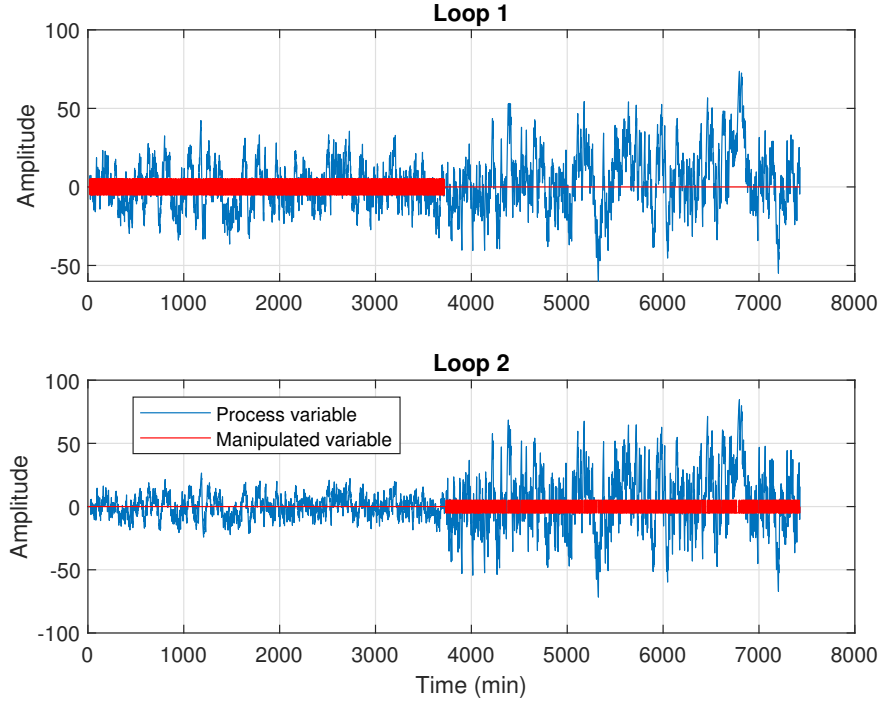


Fig. 3 Open-loop PRBS response - Simulation results

5.2 PI controller

5.2.1 Centralized control

Considering the reference model (28) and the process frequency response, the centralized PI controller designed by solving the optimization problem (20) is given by:

$$\mathbf{C}_C^{PI}(s) = \begin{bmatrix} 0.0775 + \frac{0.0078}{s} & -0.0156 - \frac{0.0080}{s} \\ 0.0222 + \frac{0.0027}{s} & -0.0396 - \frac{0.0054}{s} \end{bmatrix}, \quad (29)$$

It can be written as a decentralized PI controller plus simplified decoupler. The decentralized controller ((16) and (17)) is given by:

$$\mathbf{C}_{CD}^{PI}(s) = \begin{bmatrix} 0.0775 + \frac{0.0078}{s} & 0 \\ 0 & -0.0396 - \frac{0.0054}{s} \end{bmatrix}, \quad (30)$$

Using (18) and (19), the simplified decoupler is:

$$\mathbf{D}_C^{PI}(s) = \begin{bmatrix} 1 & 1.4748 \frac{1.9461s + 1}{7.2793s + 1} \\ 0.3414 \frac{8.3405s + 1}{9.9531s + 1} & 1 \end{bmatrix}. \quad (31)$$

5.2.2 Decentralized controller plus simplified decoupler estimated from the process model

Considering the reference model (28) and the decoupled process with the decoupler computed from the estimated process model in [15]:

$$\mathbf{D}_e(s) = \begin{bmatrix} 1 & 1.49 \frac{18.28s + 1}{18.95s + 1} e^{-1.98s} \\ 0.34 \frac{13.01s + 1}{11.69s + 1} e^{-4.08s} & 1 \end{bmatrix}, \quad (32)$$

the decentralized controller was designed by solving the optimization problem (20). In this case, the problem (20) was solved using the frequency response of the decoupled process. The obtained controller is given by:

$$\mathbf{C}_{DD}^{PI}(s) = \begin{bmatrix} 0.0587 + \frac{0.0076}{s} & 0 \\ 0 & -0.0628 - \frac{0.0055}{s} \end{bmatrix}. \quad (33)$$

5.2.3 Decentralized control

Considering the reference model (28) and the process frequency response, the decentralized controller was designed to solve the optimization problem (20):

$$\mathbf{C}_D^{PI}(s) = \begin{bmatrix} 0.0626 + \frac{0.0052}{s} & 0 \\ 0 & -0.0227 - \frac{0.0018}{s} \end{bmatrix}. \quad (34)$$

5.2.4 Results

In Figs. 4 and 5, closed-loop step responses and the control signals are presented, respectively. The performance indices are presented in the table 1. The closed-loop with the standard decentralized controller (29) and with an alternative implementation ((31) and (30)) resulted in a close response, as observed in Fig. 4 and in the MAE and M_d indices in table 1, as expected. However, due to the change in structure, the control signal was modified, becoming more smoothly variable (see MTV in Table 1). With the decentralized controller with decoupler ((32) and (33)) and decentralized controller without decoupler (34), the values obtained for MTV and M_d indices were higher compared to those proposed. Furthermore, with decentralized controller without a decoupler, the MAE was higher due to the strong coupling of the process.

Table 1 Performance indices - Simulation results - PI

	$\mathbf{C}_{CD}^{PI}$	$\mathbf{C}_C^{PI}$	$\mathbf{C}_{DD}^{PI}$	$\mathbf{C}_D^{PI}$
M_S	1.2308	1.2308	1.3872	1.2134
Loop 1				
MAE	0.1095	0.1095	0.1203	0.1986
MTV	0.1014	0.2367	0.1260	0.1582
M_d	0.0102	0.0102	0.0211	0.1021
Loop 2				
MAE	0.1106	0.1107	0.1051	0.2791
MTV	0.1260	0.1417	0.1286	0.1098
M_d	0.0170	0.0170	0.0192	0.1881

The maximum singular value curve of the sensitivity function is shown in Fig. 6. Analyzing Fig. 6 and the M_S index, it is observed that changing the implementation of the centralized controller maintained the robustness of the closed-loop. In terms of robustness, the applied structures are considered robust.

To conclude, the best structure to be implemented must take into account the performance and practical implementation in control systems. Considering the values presented in table 1, the proposed implementation for the centralized controller $\mathbf{C}_{CD}^{PI}$ plus $\mathbf{D}_C^{PI}$ is recommended in this case.

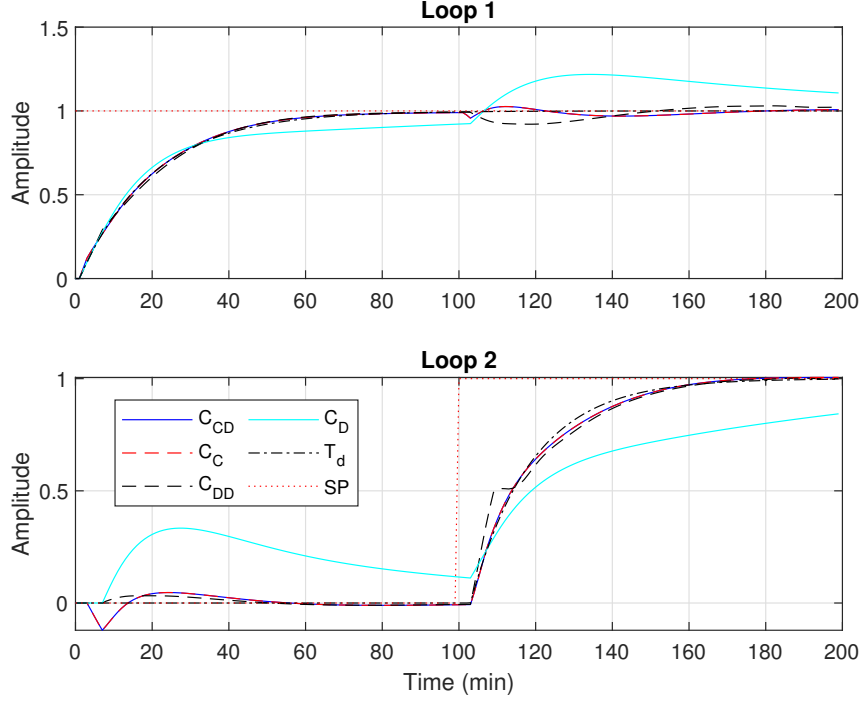


Fig. 4 Closed-loop step response - Simulation results - PI

5.3 PID controller

When a PID controller is used, a derivative filter must be incorporated into the controller to ensure causality and avoid the amplification of measurement noise. Since the filter is not considered in the controller design, the filter's time constant (τ_{fij}) is calculated based on the proportional and derivative gains [16]:

$$\tau_{fij} = 0.1 \frac{K_{Dij}}{K_{Pij}}. \quad (35)$$

This calculation (35) was necessary only for the controllers resulting from the optimization problem. For the proposed implementation, the filter value was the result of the transformation—that is, of (16) and (17).

5.3.1 Centralized control

Considering the reference model (28), we solve the optimization problem (20) to obtain the centralized PID controller (36). Although a PID controller was specified, the project resulted in PI controllers for the terms (2,1) and (2,2) of the $\mathbf{C}_{C_1}^{PID}$ controller matrix.

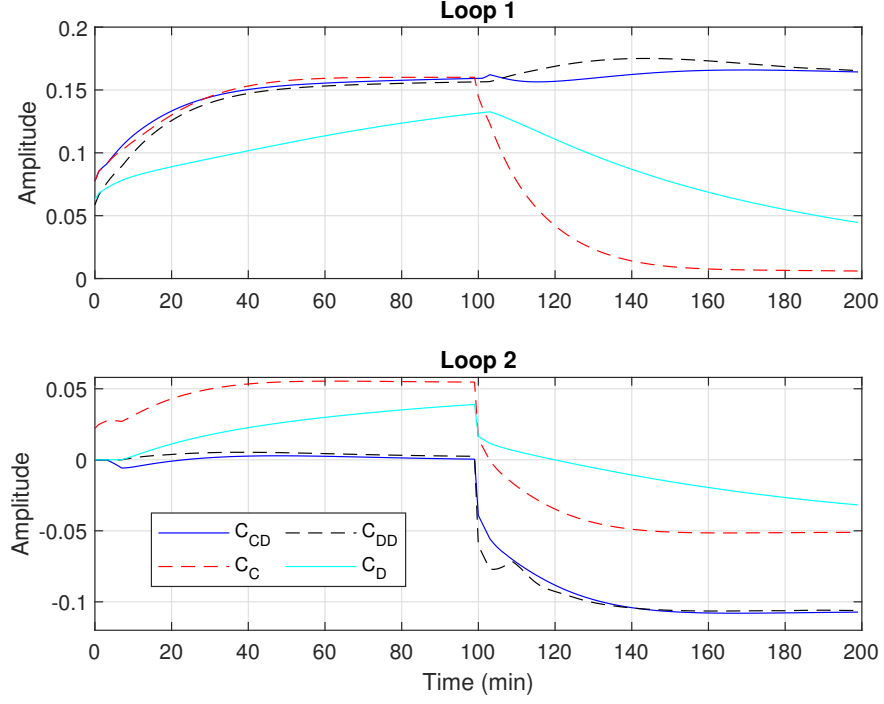


Fig. 5 Control signals - Simulation results - PI

$$\mathbf{C}_{C_1}^{PID}(s) = \begin{bmatrix} 0.073 + \frac{0.008}{s} + \frac{0.072s}{0.099s + 1} & -0.012 - \frac{0.008}{s} - \frac{0.044s}{0.355s + 1} \\ 0.0193 + \frac{0.0027}{s} & -0.0394 - \frac{0.0054}{s} \end{bmatrix}. \quad (36)$$

The controller (36) has been rewritten in the form of a decentralized controller plus simplified decoupler given respectively by (37) and (39).

$$\mathbf{C}_{CD_1}^{PID}(s) = \begin{bmatrix} 0.073 + \frac{0.008}{s} + \frac{0.072s}{0.099s + 1} & 0 \\ 0 & -0.039 - \frac{0.005}{s} \end{bmatrix}, \quad (37)$$

$$\mathbf{D}_{C_1}^{PID}(s) = \begin{bmatrix} 1 & \frac{3.46s^2 + 1.09s + 0.57}{s^2 + 2.95s + 0.39} \\ \frac{0.02s^2 + 0.25s + 0.03}{s^2 + 0.93s + 0.1} & 1 \end{bmatrix}. \quad (38)$$

Note that the resulting decoupler does not have the lead-lag form, which is the usual form for decoupler implementation. Therefore, to adjust the decoupler function, the Routh stability criterion [17, 18] was used to perform the approximation, resulting in:

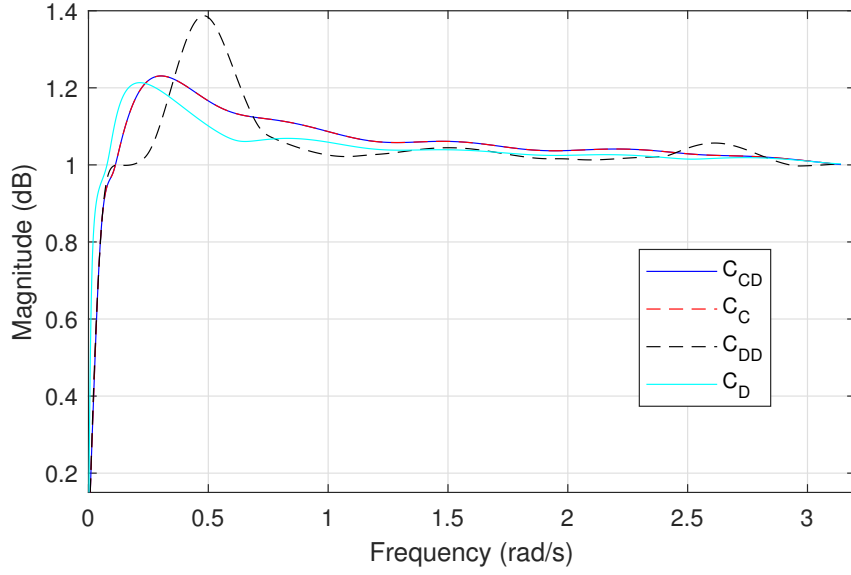


Fig. 6 Maximum singular value of the sensitivity function - Simulation results - PI

$$\mathbf{D}_{C_1}^{PID}(s) = \begin{bmatrix} 1 & 1.4745 \frac{1.8999s + 1}{7.6043s + 1} \\ 0.3409 \frac{7.3837s + 1}{9.4877s + 1} & 1 \end{bmatrix}. \quad (39)$$

5.3.2 Decentralized controller plus simplified decoupler estimated from the process model

Considering the reference model (28) and the model-based decoupler (32), the decentralized controller was designed by solving the optimization problem (20). The obtained controller is given by (40).

$$\mathbf{C}_{DD_1}^{PID}(s) = \begin{bmatrix} 0.0587 + \frac{0.008}{s} + \frac{0.004s}{0.007s + 1} & 0 \\ 0 & -0.063 - \frac{0.006}{s} + \frac{0.005s}{0.007s + 1} \end{bmatrix}. \quad (40)$$

5.3.3 Decentralized control

Considering the reference model (28) and the process frequency response, the decentralized controller was designed by solving the optimization problem (20):

$$\mathbf{C}_{D_1}^{PID}(s) = \begin{bmatrix} 0.063 + \frac{0.005}{s} + \frac{0.012s}{0.019s + 1} & 0 \\ 0 & -0.023 - \frac{0.002}{s} + \frac{0.002s}{0.011s + 1} \end{bmatrix}. \quad (41)$$

5.3.4 Results

The closed-loop step responses and the control signals are presented in Figs. 7 and 8, respectively. The performance indices were shown in table 2.

Note that even with the reduction of the decoupler model, the proposed implementation resulted in a closed-loop response close to the centralized controller, with less variation in the control signal.

The decentralized controller plus decoupler (40) resulted in a higher M_d . The centralized controller (36) resulted in a lower M_d , although it was close to the value obtained with the (37) plus (39) for loop 1. The lowest performance was obtained with the decentralized controller without decoupling (41).

The maximum singular value curve of the sensitivity function is shown in Fig. 9. The lowest value was achieved with the centralized controller ($\mathbf{C}_{C_1}^{PID}$). However, all structures resulted in robust closed-loops.

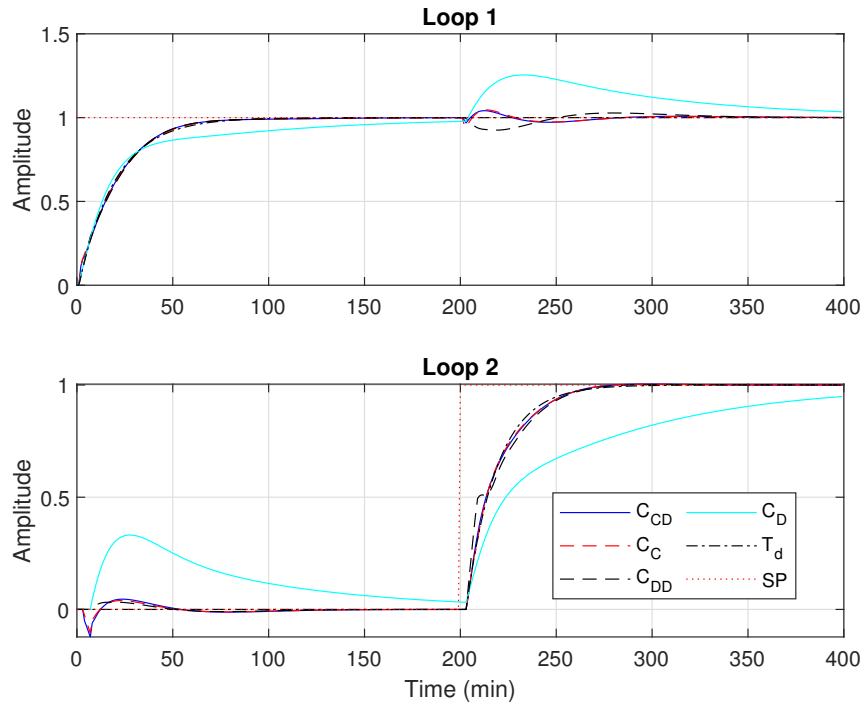


Fig. 7 Closed-loop step response - Simulation results - PID

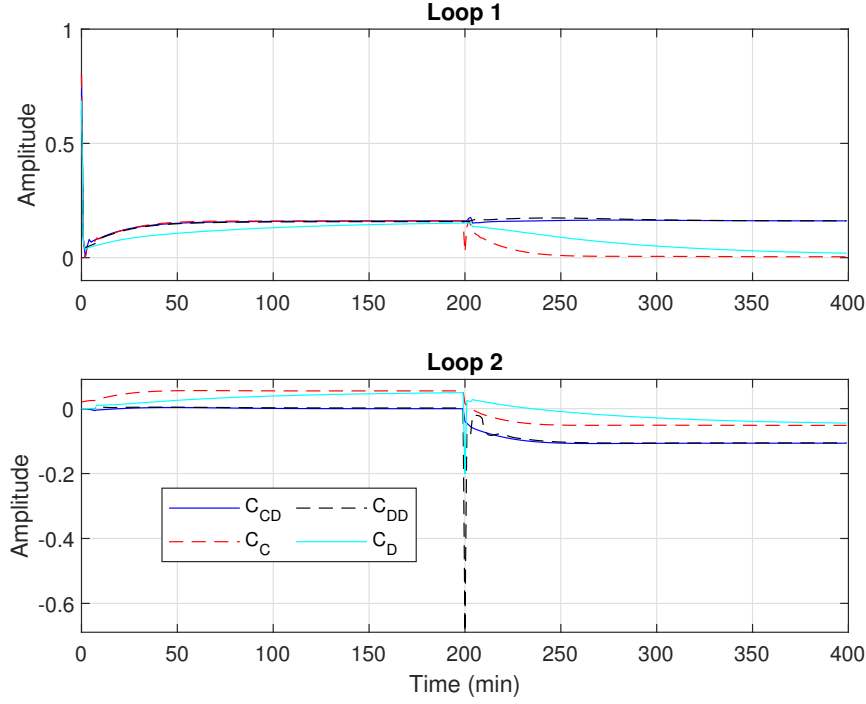


Fig. 8 Control signals - Simulation results - PID

6 Hardware-in-the-loop results

Consider a hardware-in-the-loop (HIL) architecture, where the process is simulated in Aveva Dynamic Simulation (DynSim), while the control is executed on the industrial controller (PLC), Rockwell Automation ControlLogix 5000; see Fig. 10.

The OPC UA communication protocol [19] is utilized for data exchange between the process simulator and the PLC in a client-server communication model. For this HIL architecture, it was necessary to use an OPC UA client interface, given that both DynSim and the ControlLogix are configured only as OPC UA servers.

6.1 Simulated process

The simulated process is the separation process in a separation tank or flash vessel. This process, which involves contact between the liquid and vapor phases, aims to separate two or more components based on the difference in their volatilities.

The feed to the flash vessel is a mixture composed of mole fractions z_1 through z_n . The tank is at a constant temperature T and pressure P , where only two equilibrium phases exist: a liquid phase with mole fractions x_1 through x_n , and a vapor phase with mole fractions y_1 through y_n . The molar balance for the component is:

Table 2 Performance indices - Simulation results - PID

	$C_{CD_1}^{PID}$	$C_{C_1}^{PID}$	$C_{DD_1}^{PID}$	$C_{D_1}^{PID}$
M_S	1.2204	1.2183	1.3788	1.2009
Loop 1				
MAE	0.0571	0.0572	0.0620	0.1345
MTV	1.0422	1.4110	0.7502	0.9053
M_d	0.0077	0.0078	0.0123	0.0858
Loop 2				
MAE	0.0554	0.0544	0.0529	0.1935
MTV	0.1277	0.1445	1.4953	0.6099
M_d	0.0089	0.0075	0.0103	0.1455

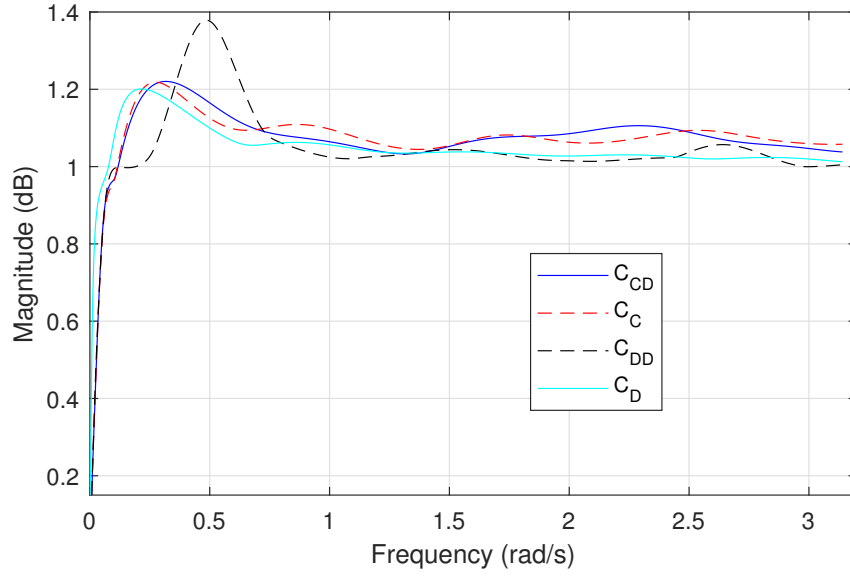


Fig. 9 Maximum singular value of the sensitivity function - Simulation results - PID

$$z_i = x_i + y_i(1 - L) \quad (42)$$

$$y_i = k_i x_i, \quad (43)$$

where L represents the net liquid stream downstream of the flash vessel and k_i is the ratio of the compositions in the vapor and liquid phases. The parameters used in the mathematical modeling are presented in Table 3.

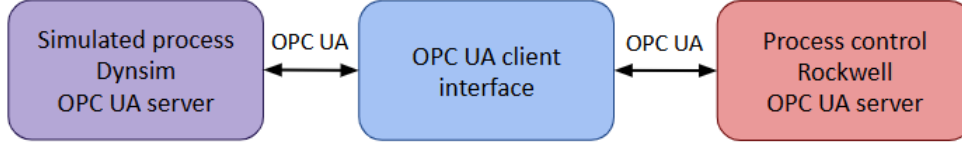


Fig. 10 Hardware-in-the-loop architecture

Table 3 Mathematical modeling parameters for the flash vessel, adapted from [20]

		Composition (kmol)
Components	Methane	1
	Ethane	0.5
	Propane	0.3
	n-butane	0.1
Pressure (kPag)		Temperature (K)
Inlet Conditions	2000	286
Height (m)		Diameter (m)
Flash Vessel	3	1.5

At constant temperature and pressure, the vessel has two phases: liquid and vapor. The inlet pressure, level, temperature, and flow rate of the vessel are control variables. In this example, let's consider control loop 1 as pressure and loop 2 as temperature. The pressure range is 0 to 10000 kPa and the temperature range is 273°C to 400°C. The range of the control signal is 0 to 1.

6.2 Controller design

The closed-loop transfer function ($\mathbf{T}_d$) considered is a diagonal matrix; the non-zero elements are a FOPTD transfer function:

$$\mathbf{T}_d(s) = \begin{bmatrix} \frac{1}{23.07s + 1} & 0 \\ 0 & \frac{1}{37.76s + 1} \end{bmatrix}, \quad (44)$$

For a frequency response, a pseudorandom binary sequence (PRBS) signal was applied to the input of the open-loop process (Fig. 11).

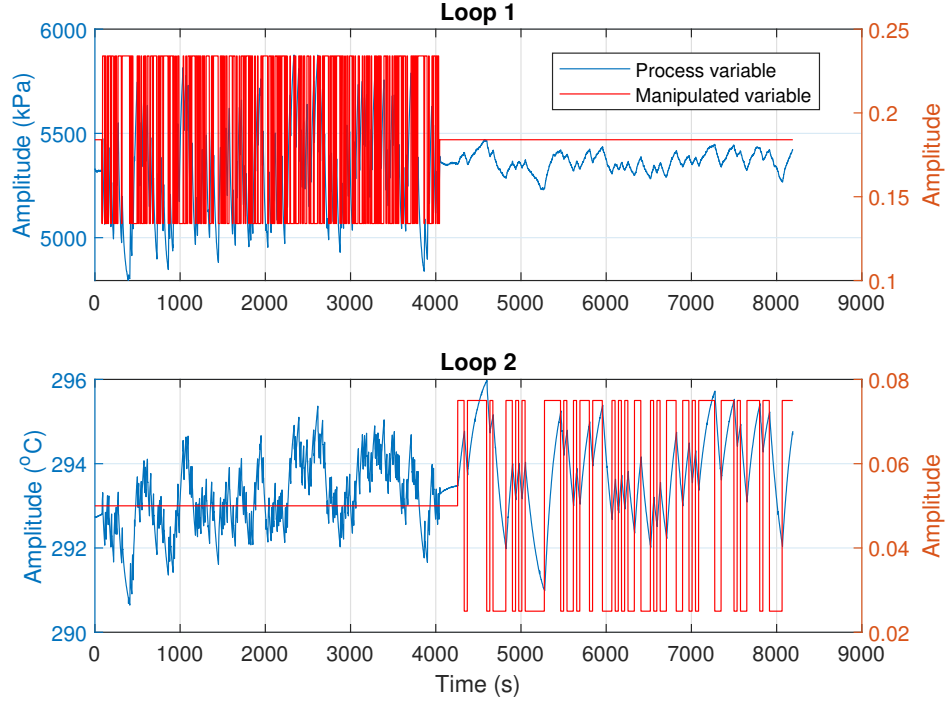


Fig. 11 Open-loop PRBS response - Hardware-in-the-loop results

6.2.1 Centralized control

From the reference model (44) and the frequency response of the open-loop process, the optimization problem (20) was solved and the centralized PI controller was obtained:

$$\mathbf{C}_C^{PI}(s) = \begin{bmatrix} -1.0210 - \frac{0.0492}{s} & 0.3621 + \frac{0.0164}{s} \\ -1.6250 - \frac{0.0471}{s} & 4.0870 + \frac{0.0467}{s} \end{bmatrix}, \quad (45)$$

It can be written as a decentralized PI controller plus simplified decoupler. The decentralized controller ((16) and (17)) is given by:

$$\mathbf{C}_{CD}^{PI}(s) = \begin{bmatrix} -1.0210 - \frac{0.0492}{s} & 0 \\ 0 & 4.0870 + \frac{0.0467}{s} \end{bmatrix}, \quad (46)$$

and the simplified decoupler ((18) and (19)) is:

$$\mathbf{D}_C^{PI}(s) = \begin{bmatrix} 1 & 0.3512 \frac{22.0793s + 1}{87.5000s + 1} \\ 0.9573 \frac{34.5011s + 1}{20.7516s + 1} & 1 \end{bmatrix}. \quad (47)$$

6.2.2 Decentralized control plus simplified decoupler estimated from the process model

In this case, the first step is to identify a first-order model with time delay to design the decoupler. Thus, an open-loop step experiment was performed. The input and output data collected during this experiment were used to estimate the model:

$$\mathbf{G}(s) = \begin{bmatrix} \frac{-1.0945}{23.074s + 1} & \frac{0.4073}{95.939s + 1} \\ \frac{-0.5252}{50.124s + 1} & \frac{0.8526}{113.27s + 1} \end{bmatrix}. \quad (48)$$

The decoupler was designed from (48):

$$\mathbf{D}_e(s) = \begin{bmatrix} 1 & 0.3721 \frac{23.074s + 1}{95.939s + 1} \\ 0.6160 \frac{113.27s + 1}{50.124s + 1} & 1 \end{bmatrix}, \quad (49)$$

Decentralized control was designed considering the frequency response of the decoupled process ($\mathbf{G}(s)\mathbf{D}(s)$). When solving the optimization problem, the resulting controller was:

$$\mathbf{C}_D^{PI}(s) = \begin{bmatrix} -1.0160 - \frac{0.0495}{s} & 0 \\ 0 & 4.0700 + \frac{0.0470}{s} \end{bmatrix}. \quad (50)$$

6.3 Controller implementation

ControlLogix PLC programming begins with the creation of the program organization unit (POU), with function blocks being the chosen PLC programming language. This choice was necessary because the decoupler is implemented using the lead-lag block, which is only available for function block designs. Next, the PIDE, LDLG, and ADD blocks were selected to build the control logic.

The following centralized control, as shown in Fig. 1, was implemented; see Fig.12. The function block used to implement PI was PIDE with devirative gain equal to zero, with PIDE_06, PIDE_07, PIDE_08, and PIDE_05 representing the C_{C11} , C_{C12} , C_{C21} and C_{C22} , respectively.

The modified controller parameters were the gains K_p and K_i , and the input and output ranges according to the corresponding loop. It was necessary to adjust the

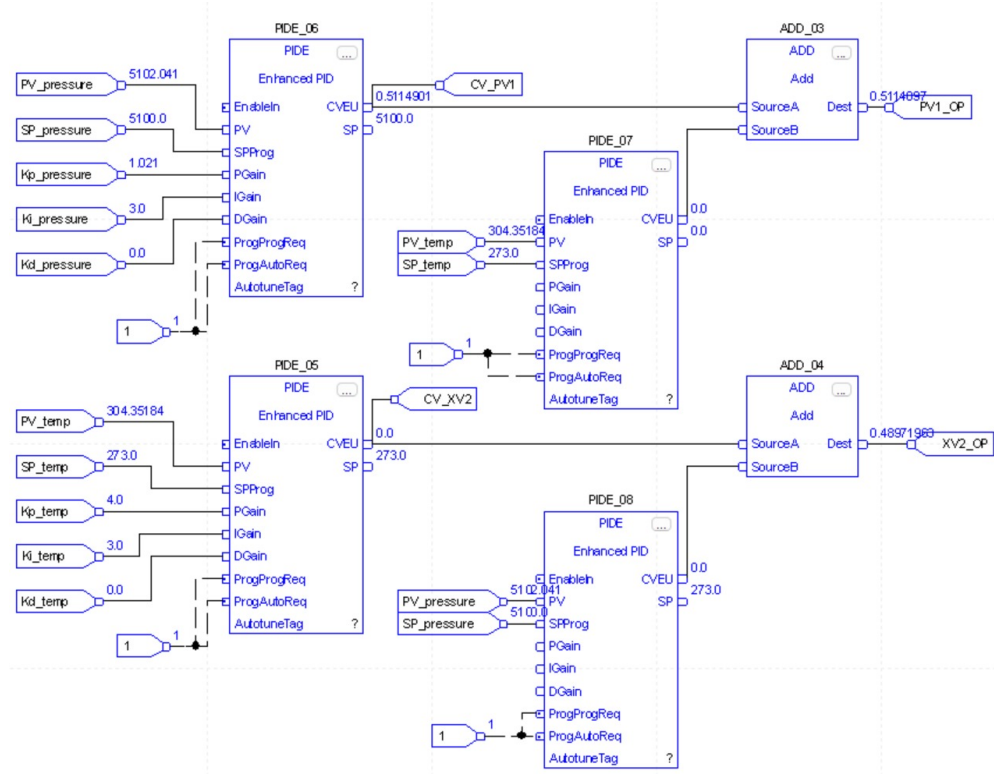


Fig. 12 Centralized controller implementation in ControlLogix - Hardware-in-the-loop results

projected integral gain because the engineering unit was $1/s$, while in the PIDE block, this unit is $1/min$. This was done by multiplying the calculated K_i values by 60. For blocks PIDE_06 and PIDE_05, a variable corresponding to the controller gain was created, however for blocks PIDE_07 and PIDE_08, these parameters were entered directly from the block configuration screen, showing two ways to enter this data.

Furthermore, the control direction was adjusted. For the pressure loop that has reverse gain, i.e., negative gain, the PIDE must be adjusted to calculate the error as the process variable (PV) minus the setpoint (SP): $e = PV - SP$. For the temperature loop, the option should be selected $e = SP - PV$. Since the setpoint signal was connected to the SPProg input of the PIDE block, it was necessary to enable the ProgProgReq and ProgAutReq inputs.

The sum of two different PIDE codes is calculated in the ADD block. For this block, no configuration is required. The output signal from the ADD block is sent to the valve. The outputs of blocks ADD_03 and ADD_04 are the control signals for loop 1, and loop 2, respectively.

After performing an experiment with the centralized controller shown in Fig. 12, this controller was implemented as a decentralized plus simplified decoupler Fig. 2. The implementation in ControlLogix is shown in Fig. 13.

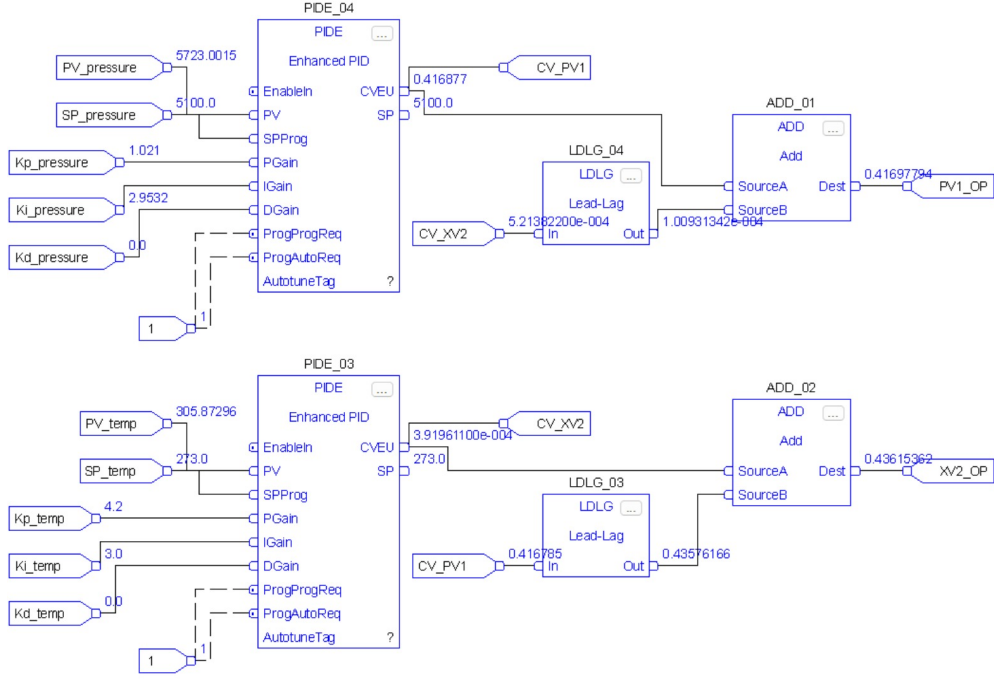


Fig. 13 Proposed centralized controller implementation in ControlLogix - Hardware-in-the-loop results

The same changes were made to the PIDE blocks, but now PIDE_04 corresponds to term (1,1) of C_D and PIDE_03 to (2,2). Additionally, the ADD_01 block calculates the sum of the output of term (1,1) of controller to decoupler D_{12} , and ADD_02 calculates the sum of the output of (2,2) term of controller to decoupler D_{21} , the outputs of the ADD_01 and ADD_02 blocks being the control signals for loops 1 and 2, respectively.

The decoupling block was implemented using the LDLG block available in the control system. For this block, the parameters are the calculated values of gain, lead (time constant numerator) and lag (time constant denominator). The LDLG_04 and LDLG_03 are the D_{12} and D_{21} decoupler terms, respectively.

This same structure was used to implement the decentralized controller designed directly using the frequency response of the decoupled process.

6.4 Results

The response to applying a step to loops 1 and 2 is shown in Figs. 14 and 16 and the respective control signals in Figs. 15 and 17, respectively.

In this case, we can observe that when the pressure step was applied, the loop with the standard centralized controller (45) (C_C) was faster, with a rise time of approximately 271 s. When implemented as a decentralized controller plus decoupling (46) plus (47) (C_{CD}) and in the loop with decentralized controller (50) (C_D), the rise times were 310 s and 363 s, respectively. Furthermore, the alternative form of

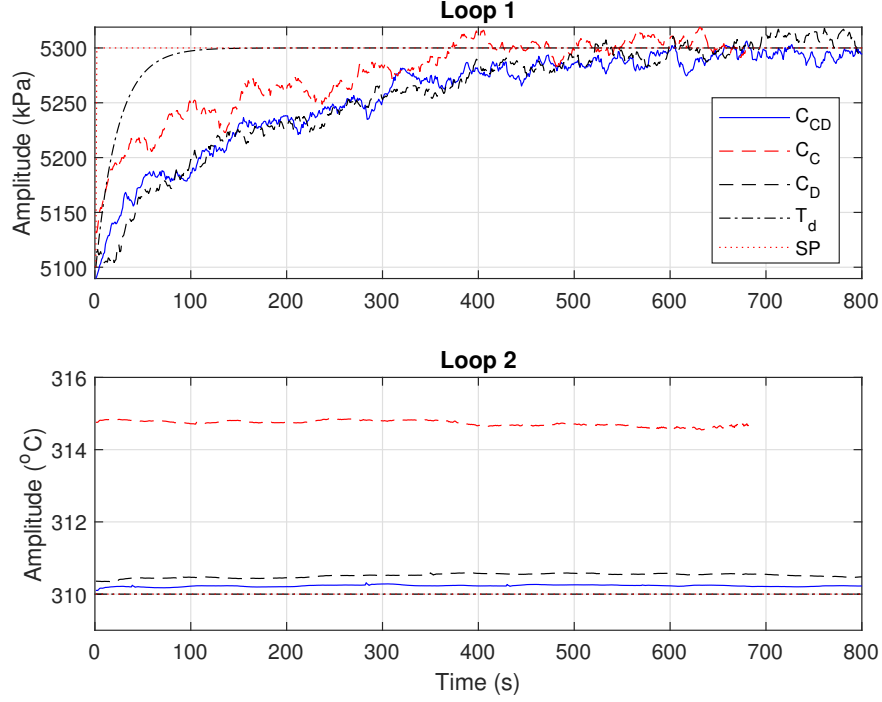


Fig. 14 Closed-loop step response - step in loop 1 - Hardware-in-the-loop results

the centralized controller reduces interaction with the temperature loop. The largest jump of the control signal occurred with the decentralized controller (50) (C_D), at approximately 20 s.

When the step was applied to the temperature loop, similar performance to the previous one could be observed. In this case, the rise time were 54 s, 67 s, and 118 s for the loops with the C_C , C_{CD} , and C_D controllers, respectively. However, the main operating regime error occurred when using the centralized controller (45) (C_C), and this error was reduced when the implementation was changed (C_{CD} - (46) and (47)). The control signal showed no significant jump in this case; however, the largest change was with the (C_{CD}) controller.

7 Conclusion

This paper discusses a way to implement centralized PI/PID controllers in Rockwell Automation ControlLogix PLC. A design technique based on the process frequency response, formulated as a convex optimization problem, was used to design the controller. In addition to implementing the centralized controller in the traditional way, an alternative approach was also discussed, rewriting this controller as a decentralized controller with a simplified decoupler.

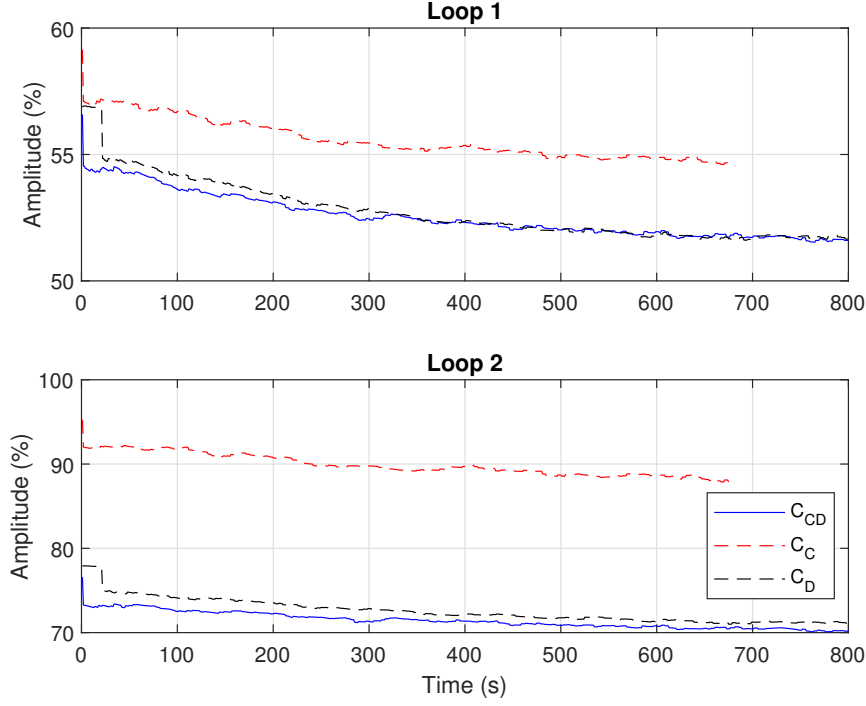


Fig. 15 Control signals - step in loop 1 - Hardware-in-the-loop results

For comparison analysis, the optimization problem was also used to design the decentralized controller. However, in this case, the algorithm input was the frequency response of the decoupled process. The parameters of this decoupler were calculated from a FOPTD model of the process.

One point to consider in the controller design is the sign of the designed parameters. It is common for centralized control design techniques to result in parameters with different signs, as happens in [21], [22], [23], [24] and [25]. This occurs mainly when a PID controller is used. An example of this is the parameters K_p and K_i having positive values, while K_d has a negative value. However, controllers with different signals are not implemented in industrial controllers.

In the implementation of the centralized controller in the form of a decentralized controller plus a decoupler, this sign difference can result in unstable poles in both the controller and the decoupler. To solve this problem, it is necessary to add a sign constraint to the parameters in the optimization problem. Furthermore, in the transformation of the centralized control, it may be necessary to reduce the order of the transfer functions obtained for the decoupler, so that it takes the form of lead-lag.

The results obtained show that it is possible to implement centralized control as a more decoupled decentralized control, without losing performance and robustness. The next steps in the research are to use the inverted decoupler.

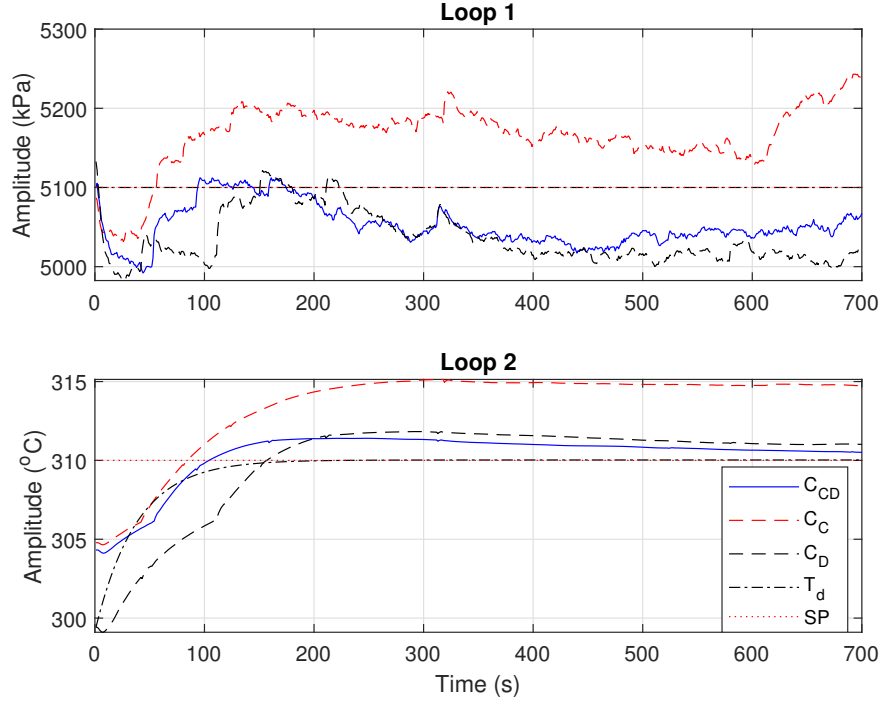


Fig. 16 Closed-loop step response - step in loop 2 - Hardware-in-the-loop results

Funding. This work has been fully funded by Petrobras through Cooperation Agreement No. 0050.0126772.24.9 and by the project Desenvolvimento de Soluções em Plataformas Abertas de Automação de Processos O-PAS supported by CENTRO DE COMPETÊNCIA EMBRAPPII VIRTUS EM HARDWARE INTELIGENTE PARA INDÚSTRIA - VIRTUS-CC, with financial resources from the PPI HardwareBR of the MCTI grant number 055/2023, signed with EMBRAPPII and by CAPES (Call for Proposals No. 38/2022) in partnership with FAPESQ.

Declarations

- **Conflict of interest** The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.
- **Data availability** Not applicable
- **Code availability** Not applicable
- **Author contribution**
 - Conceptualization: Anna Paula V de A Aguiar, Péricles Rezende Barros
 - Methodology: Anna Paula V. de A. Aguiar, Péricles Rezende Barros, George Acioli Júnior

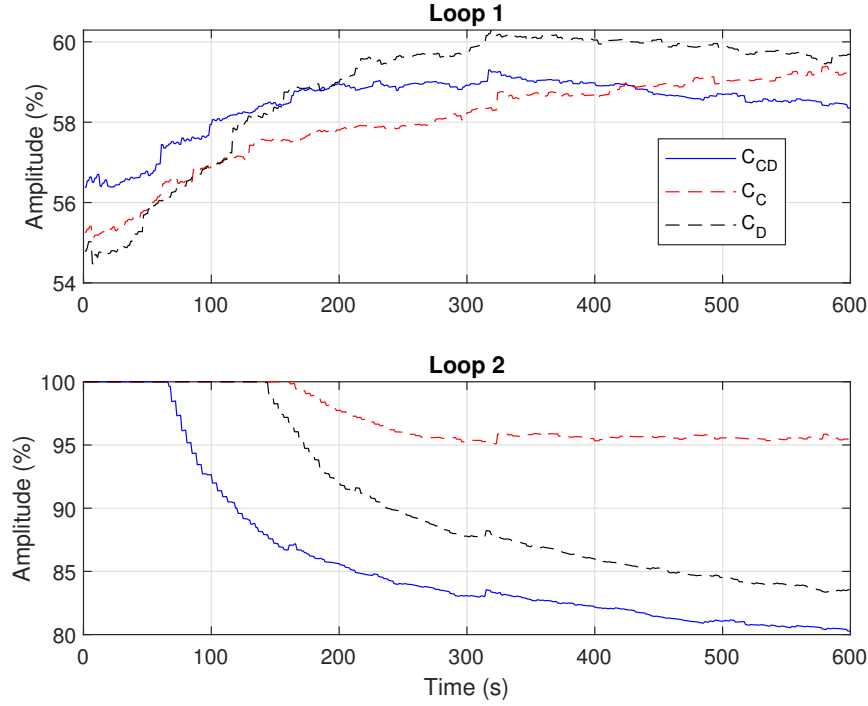


Fig. 17 Control signals - step in loop 2 - Hardware-in-the-loop results

- Writing - Anna Paula V. de A. Aguiar;
- Writing - review: Péricles Rezende Barros, George Acioli Júnior;
- Implementation and coding: Anna Paula V. de A. Aguiar, Julián Gutiérrez;
- Aquisição de financiamento: Péricles Rezende Barros, George Acioli Júnior;
- Supervision: Péricles Rezende Barros.

References

- [1] Aguiar, A.P.V.d.A., Acioli Junior, G., Barros, P.R.: Frequency-based multivariable control design with stability margin constraints: A linear matrix inequality approach. *Journal of Process Control* **132**, 103–115 (2023) <https://doi.org/10.1016/j.jprocont.2023.103115>
- [2] Karimi, A., Kammer, C.: A data-driven approach to robust control of multivariable systems by convex optimization. *Automatica* **85**, 227–233 (2017) <https://doi.org/10.1016/j.automatica.2017.07.063>
- [3] Huff, D.D., Campestrini, L., Silva, G.R., Bazanella, A.S.: Data-driven control design by prediction error identification for multivariable systems. *Journal of*

- [4] Garrido, J., Ruz, M.L., Morilla, F., Vázquez, F.: Iterative method for tuning multiloop PID controllers based on single loop robustness specifications in the frequency domain. *Processes* **9**(1), 140 (2021)
- [5] Mayne, D.Q.: Sequential design of linear multivariable systems. *Proceedings of the Institution of Electrical Engineers* **126**, 568–5724 (1979) <https://doi.org/10.1049/piee.1979.0135>
- [6] Liu, L., Tian, S., Xue, D., Zhang, T., Chen, Y., Zhang, S.: A review of industrial MIMO decoupling control. *International Journal of Control, Automation and Systems* **17**(5), 1246–1254 (2019)
- [7] Govind, K.A., Mahapatra, S., Mahapatro, S.R.: A comparative analysis of various decoupling techniques using frequency domain specifications. In: 2023 3rd International Conference on Artificial Intelligence and Signal Processing (AISP), pp. 1–6 (2023). IEEE
- [8] Achu Govind, K., Mahapatra, S., Mahapatro, S.R.: Nonlinear constraint optimization based robust decentralized PID controller for a benchmark CSTR system using kharitonov-hurwitz stability analysis. *Arabian Journal for Science and Engineering* **48**(11), 15377–15402 (2023)
- [9] KR, A.G., Mahapatra, S., Ranjan Mahapatro, S.: Design of a decentralized control law for variable area coupled tank systems using H_∞ complimentary sensitivity function. *Asian Journal of Control* **26**(3), 1540–1552 (2024)
- [10] Mahapatro, S.R., Subudhi, B.: A robust stability region-based decentralized PI controller for a multivariable liquid level system. *IEEE Systems Journal* **16**(1), 124–131 (2021)
- [11] Chuong, V.L., Vu, T.N.L., Truong, N.T.N., Jung, J.H.: An analytical design of simplified decoupling smith predictors for multivariable processes. *Applied Sciences* **9**(12), 2487 (2019)
- [12] Garrido, J., Vázquez, F., Morilla, F.: Centralized multivariable control by simplified decoupling. *Journal of process control* **22**(6), 1044–1062 (2012)
- [13] Grant, M.C., Boyd, S.P.: Graph implementations for nonsmooth convex programs. In: *Recent Advances in Learning and Control*, pp. 95–110. Springer, ??? (2008)
- [14] Wood, R., Berry, M.: Terminal composition control of a binary distillation column. *Chemical Engineering Science* **28**(9), 1707–1717 (1973)
- [15] Aguiar, A.P.V.d.A., Acioli Jr, G., Barros, P.R.: Evaluation and redesign of the

- inverted decoupler: open and closed-loop approaches. *International Journal of Control, Automation and Systems* **18**(6), 1435–1444 (2020)
- [16] Wang, L.: *PID Control System Design and Automatic Tuning Using MATLAB/Simulink*. John Wiley & Sons, England (2020)
 - [17] Krishnamurthy, V., Seshadri, V.: Model reduction using the routh stability criterion. *IEEE Transactions on Automatic control* **23**(4), 729–731 (1978)
 - [18] Singh, A., Yadav, S., Patel, S.: Model predictive-based pid controller design and order reduction of higher order system. *International Energy Journal* **25** (2025)
 - [19] OPC Foundation: OPC Foundation. <https://opcfoundation.org/> Accessed 2025-11-26
 - [20] Seader, J.D., Henley, E.J.: *Separation Process Principles*. Wiley New York, United States of America (1998)
 - [21] Garrido, J., Garrido-Jurado, S., Vázquez, F., Arrieta, O.: Design of multivariable PID control using iterative linear programming and decoupling. *Electronics* **13**(4), 698 (2024)
 - [22] Konkov, A.E., Mitrishkin, Y.V.: Synthesis methodology for discrete MIMO PID controller with loop shaping on LTV plant model via iterated LMI restrictions. *Mathematics* **12**(6), 810 (2024)
 - [23] Boyd, S., Hast, M., Åström, K.J.: MIMO PID tuning via iterated LMI restriction. *International Journal of Robust and Nonlinear Control* **26**, 1718–1731 (2016) <https://doi.org/10.1002/rnc.3376>
 - [24] Feng, Z.-Y., Guo, H., She, J., Xu, L.: Weighted sensitivity design of multivariable pid controllers via a new iterative lmi approach. *Journal of Process Control* **110**, 24–34 (2022)
 - [25] Galdos, G., Karimi, A., Longchamp, R.: H_∞ controller design for spectral MIMO models by convex optimization. *Journal of Process Control* **20**(10), 1175–1182 (2010)