Preprints, 4th IFAC Conference on
Advances in Proportional-Integral-Derivative Control
Almería, Spain | June 12-14, 2024

A PID Control Architecture Based on IEC 61499 ⋆

Guzmán González-Mateos * **Miguel A. Prada** *
Antonio Morán * **Raúl González-Herbón** *
Manuel Domínguez *

* *SUPPRESS research group, Escuela de Ingenierías, Universidad de León, Campus de Vegazana, León, 24007, Spain*
https://suppress.unileon.es, ggonzm@unileon.es

Abstract: This work presents an approach for PID control and optimization in a distributed system, using the IEC 61499 control standard. This standard enables communication among different PLCs, which are used to develop a three-layered event-driven control of a SISO loop. The lowest layer is in charge of cyclical data acquisition. The second layer carries out an event-based PID control. The highest layer runs a control optimization algorithm, specifically a simple tuning approach based on Ziegler-Nichols that is used to determine the PID parameters for different operating points. The proposed approach is assessed in a tank level SISO control problem, whose behavior can be modeled as a first-order plus dead time system in each operating point. For that purpose, it has been implemented using two PLCs and a software PLC running on an industrial computer. The experimental results on the SISO level control loop show the feasibility of the proposed approach for event-driven control in a distributed system and open interesting research questions in the intersection of controller design and distributed automation.

Keywords: IEC61499, PID, Distributed Control System, PID Tuning, Industrial application, Applications of PID control.

1. INTRODUCTION

In recent years, a new digitalization trend, known as Industry 4.0, has emerged in the industrial sector. This new paradigm involves a digital interconnection that leads to the concept of cyber-physical systems (CPS), which integrate computational and communication capabilities with physical processes (Lu, 2017).

Following this paradigm, the standard IEC 61499 has arisen for the development of modern control automation. This standard proposes a more modular, scalable and flexible approach. The main idea behind it is to achieve a decentralized control structure in which the control system is distributed through a communication network. In order to achieve that, the standard makes use of an event-driven execution system (Vyatkin, 2011; Lyu and Brennan, 2021). Thus, in IEC 61499, control is not based on a cyclic execution as in traditional control, but rather on events that are triggered by the system in response to: signal changes, the completion of a task, an error occurrence or any other predefined condition.

In order to manage the events, programmable logic controllers (PLCs) following the standard make use of Function Blocks, which act as modular, reusable components that can receive events, process data, and generate new events or outputs. These blocks respond to events by executing algorithms inside a state machine that are programmed using Structured Text, a language defined in IEC 61131. Once the block finishes, it triggers an output event in order to perform another task. The data handled within IEC 61499 resides within the function blocks and flows between them. This data encapsulation ensures that each block is self-contained, with its own input and output data. Data exchange occurs through well-defined interfaces, allowing blocks to share information without exposing their internal operation. Figure 1 shows the structure of a Function Block using the IEC 61499.

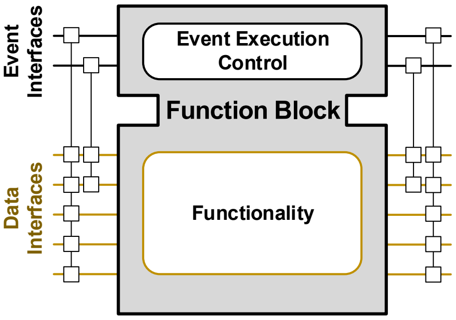

Fig. 1. Function Block in IEC 61499

From the point of view of process control, this standard opens new opportunities for controller design and implementation through the use of distributed applications. In this sense, Garcia et al. (2018) presented a model predictive control of an oil pipeline system implemented using

---

⋆ This work was supported by the Spanish State Research Agency, MCIN/ AEI/ 10.13039/ 501100011033 under Grant PID2020-117890RB-I00 and Grant PID2020-115401GB-I00. The work of Guzmán González-Mateos was supported by a grant from the *2022 Edition of Research Programme of the University of León.*

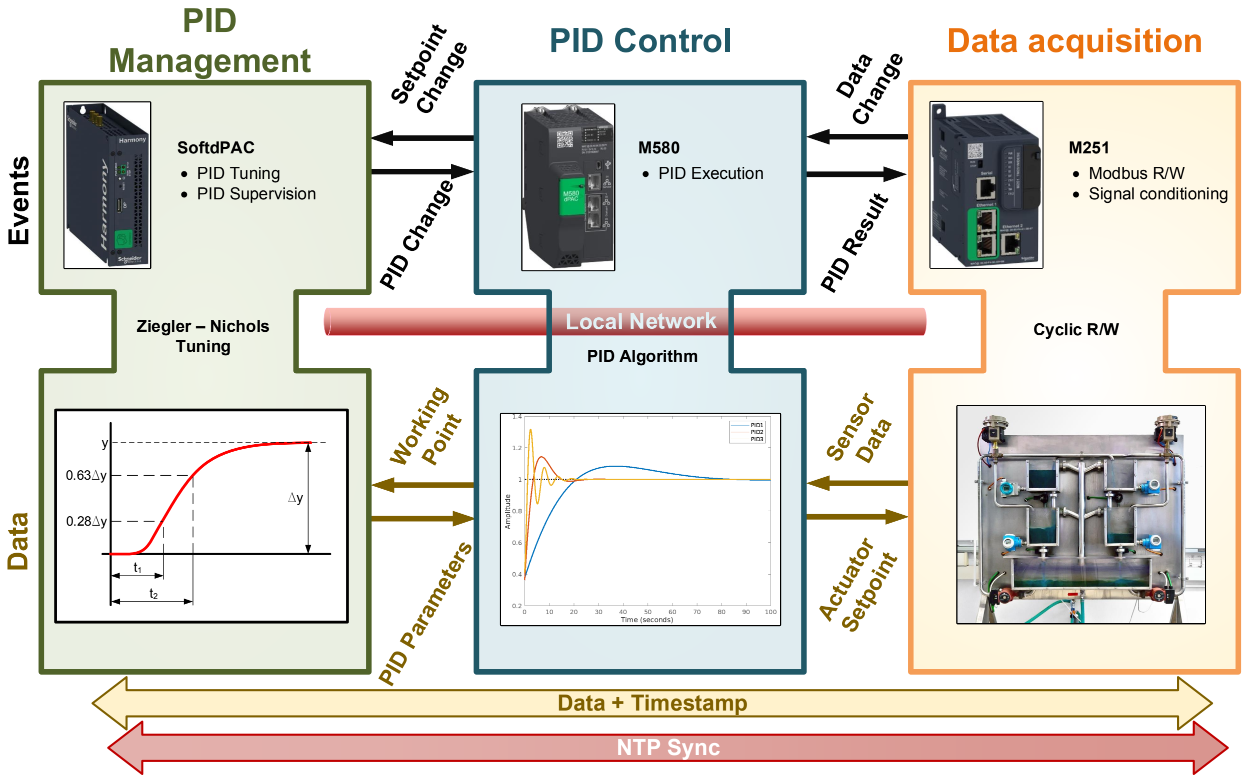

Fig. 2. Summary of the proposed control architecture.

IEC 61499 function blocks, whereas the implementation of event-based PID in IEC 61499 function blocks was first discussed in Miguel-Escrig and Romero-Pérez (2018). Indeed, in this context, the use of event-based control (Dormido et al., 2008) appears as a natural choice since system execution is no longer necessarily time-triggered.

It is necessary to consider different scenarios where this standard could be interesting for controller tuning and implementation. On the one hand, because the expected increasing adoption of this standard in the industry will require an educational effort, since control education needs to address the new technological challenges that appear as a result of industrial digitalization (Muñoz de la Peña et al., 2022). On the other hand, from a research perspective, it is interesting to study the use of event-triggered controllers or the integration of controller optimization procedures in IEC-61499 applications that are distributed in a network systems.

For that reason, in this paper, a first step is taken towards that direction through the proposal and assessment of a PID control architecture implementable on IEC 61499. This event-driven architecture uses three layers that are executed in different devices to perform data acquisition, event-based PID control and control optimization.

The rest of the paper is structured as follows. Section 2 explains the proposed method for the control architecture, including data acquisition, PID control and parameterization. Section 3 presents the physical system that will be used for the experiments, as well as the specific automation that is used for the implementation of the proposed approach. The experiments and results are described in Section 4. Finally, conclusions are drawn in Section 5

2. CONTROL ARCHITECTURE

Using these automation systems and within the modular idea of the standard's control architecture, a three-layer structure is proposed in order to achieve a control configuration that allows better control of a system in a distributed manner. As shown in Figure 2, each of these layers has a purpose.

2.1 Layer 1. Data acquisition

The acquisition layer allows reading and writing the physical variables of the system that is controlled. The objective of this layer is to make the cyclical reading of its variables and to carry out the necessary operations to condition the physical signals. In order to achieve greater modularity with older equipment, the proposed architecture makes use of a remote Input/Output device.

In this case, the I/O reading/writing Function Block runs continuously in an internal task of the PLC. When a substantial change in the measured value is detected, an event is triggered so that the value can be transmitted to the block that executes the PID controller algorithm, a block that runs on another programmable controller and corresponds to a different layer within the application architecture. Control outputs will also be received by means of events triggered by the PID controller. Data exchange will always be associated to the specific timestamp of the event generation. The transmission of that timestamp lets control know the time elapsed between readings, writings and transmissions.

2.2 Layer 2. PID Control

This layer is the one where the controller runs and it is event-triggered. An event-based control requires recording the time elapsed between each event that triggers the execution of the PID algorithm, which in this case are the updates of the values read by layer 1. This is unlike traditional PLC implementations in which the time elapsed between two PID executions is known. However, as a safety measure to guarantee a maximum time elapsed between executions regardless of the generated events, a cyclic execution task has also been configured. This task will act in case no event has occurred during a parameterized time lapse.

The proposed controller implementation uses the improvements to Årzén's event-based PID (Årzén, 1999) proposed by Durand and Marchand (2009). Therefore, a similar approach to the one proposed in Miguel-Escrig and Romero-Pérez (2018) is followed, where an event-based PID was also developed in IEC 61499. The main difference with this implementation is that their method required an external cyclic event in order to execute the PID and check whether the maximum time without computing the control action has elapsed, whereas our proposal includes an internal cyclic event to trigger PID execution when the maximum time has elapsed, avoiding unnecessary data transmission. In case the PID is executed by an external event, this internal counter is reset.

Once a new value of the control output has been calculated by the PID, an event will be triggered towards layer 1, so that the value can be written to the physical system. Every time the control system determines that there is a change in the setpoint, an event is generated towards layer 3 so that this layer computes new parameters optimized for the current operating point. As data are transmitted with the timestamp of its acquisition, the delay between the emission of the event from layer 1 and its reception in this layer can be measured. This measurements might be interesting if networked control systems (NCS) techniques (Zhang et al., 2020) are applied.

2.3 Layer 3. PID Management

This last layer is responsible for carrying out the optimization and adjustment tasks of the PID parameters. In order to be able to adapt, it requires a more advanced optimization methodology that is not easily implemented in a PLC but that can be deployed on a low-cost PC. Within that device, optimization algorithms need to be implemented. The developed functionality receives the values necessary to execute the selected algorithm and returns necessary control parameters.

In order to assess the architecture, it is proposed to use a simple tuning approach based on the Ziegler-Nichols method (Ziegler and Nichols, 1942). The system can be approximated to a first-order model with a delay, so in order to obtain the PID parameters, the Ziegler Nichols method is applied using a two-point algorithm based on the calculation of the times elapsed by this system model to reach 28.3% and 63.2% of the final value (Liu et al., 2013).

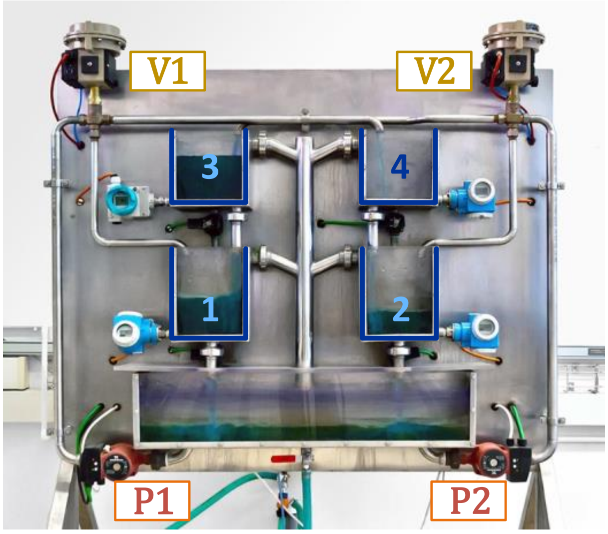

Fig. 3. Scale-model of the quadruple-tank process used to test the architecture.

The task is triggered when there is a setpoint change in the control system, and so layer 2 launches an event sharing the conditions of the operating point that are necessary to determine the new linearized transfer function, which is used to compute the parameters K_p, T_i and T_d using the aforementioned tuning procedure. Once those parameters are computed, another event is generated to transmit them to layer 2, where the running PID parameterization is updated.

3. APPLICATION

3.1 Physical system

The experimental platform for our proposed experiments is based on a pilot plant equipped with industrial instrumentation. Its setup is based on Karl Henrik Johansson's *quadruple-tank process* (Johansson, 2000). The industrial plant, as shown in Figure 3, was developed at the Remote Laboratory of Automatic Control at the University of León (Fuertes et al., 2008).

The configuration of this system involves four water tanks organized in pairs, vertically aligned so that upper tanks directly feed into the lower ones. To regulate the water flow, two symmetrical pumps with variable speed drives supply the required flow from a lower water tank. This flow is then distributed among the tanks using two three-way valves. Notably, the distribution occurs in a crossed manner: the left-hand pump and valve combination control the lower left and upper right tanks, while the right-hand combination manages the lower right and upper left tanks. Each tank is outfitted with a pressure sensor, enabling measurement of liquid level. This facilitates control of these variables by using the pumps and valves within the system as outputs. Additionally, a valve situated at the base of each tank allows for individual modification of the drained flow rate to intentionally introduce disturbances affecting the control of the corresponding tank level.

For clarity, Figure 4 offers a schematic representation of Johansson's model, whereas Table 1 outlines the main variables and constants of the quadruple-tank process.

Table 1. Variables and constants of the quadruple-tank process

Variable	**Units**	**Description**
h_i	cm	water level in tank i
h_i^0	cm	steady-state of tank i
x_i	cm	level deviations $x_i = h_i$-h_i^0
q_i	cm^3/s	flows of the pumps to tanks
υ_j	0-100	ratio of the pumps
υ_j^0	0-100	steady-state pumps
u_j	0-100	deviations of pumps $u_i = \upsilon_j$-υ_j^0
$q_{pump,j}$	cm^3/s	total flows of the pumps
γ_j	0-1	ratio of the valves
Constant	**Units**	**Description**
A_i	cm^2	cross-section of the tanks
a_i	cm^2	cross-section of the lower outlets
g	cm^2/s	acceleration due to gravity
k_j	cm^3/s	pump flow constants
k_c	cm	lower tank constants

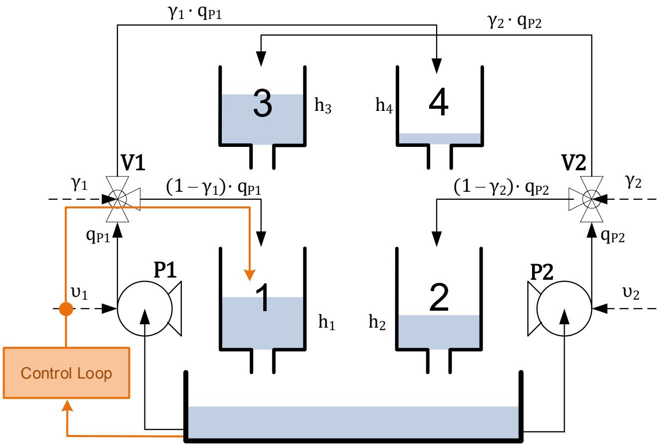

Fig. 4. Schematic diagram of the quadruple-tank process.

The mathematical representation of the system can be formulated by combining the corresponding equations applied to the tanks and pumps: Bernoulli's law, $q_{out} = a_i\sqrt{2gh_i}$, and mass balances, $A\dot{h} = q_{in} - q_{out}$. In this paper, since the purpose is to test the proposed architecture, only a SISO system is considered, creating a loop control with pump 1 and the level of tank 1. For this scenario, Valve 1 and Valve 2 are always set to 0 and pump 2 is turned off. Figure 4 shows this control loop.

Taking this into account, the model of the SISO system of tank 1 can be described by the following differential equation:

$$\frac{dh_1}{dt} = -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{k_1}{A_1}\upsilon_1 \tag{1}$$

If this differential equation is linearized around an operating point using the Taylor series expansion, a transfer equation of the control loop for tank 1 can be obtained:

$$G(s) = \frac{(1-\gamma_1)c_1}{1+sT_1} \tag{2}$$

where

$$T_1 = \frac{A_1}{a_1}\sqrt{\frac{2h_1^0}{g}} \qquad c_1 = \frac{T_1 k_1 k_c}{A_1} \tag{3}$$

Fig. 5. Control cabinet with IEC 61499 PLCs

As it can be seen in the transfer function, the system is a first order whose gain and time constant depend on the initial level of the tank. Therefore, depending on the operating point, the dynamics of the system changes. In addition to that, the system presents a delay in the response due to the pump response and the tube length, resulting in a first-order plus dead time (FOPDT) model, as follows:

$$G(s) = e^{-\theta s}\frac{(1-\gamma_1)c_1}{1+sT_1} \tag{4}$$

3.2 Implementation of the control architecture

As a result of its interest in education and research on Industry 4.0 (Fuertes et al., 2021), the SUPPRESS group at the University of León has industrial-level devices based on the IEC 61499 standard. These devices, belonging to the Schneider Electric dPAC range, are among the first commercially available IEC 61499-enabled PLCs developed by industrial manufacturers. For the implementation of the proposed architecture, they have been installed in one of the control cabinets available in the ULE-Schneider Electric IoT Classroom (Domínguez et al., 2022), as can be seen in Figure 5.

The data acquisition layer in the proposed control architecture is implemented with a low-end programmable logic controller, the Schneider Electric dPAC TM251. To carry out the communication, variables are read through Modbus TCP requests cyclically every 100ms and sent to the control layer only when the corresponding event is triggered (sensor_event in Fig 6).

The control layer is implemented instead in a Schneider Electric dPAC M580 PLC, which has enough processing

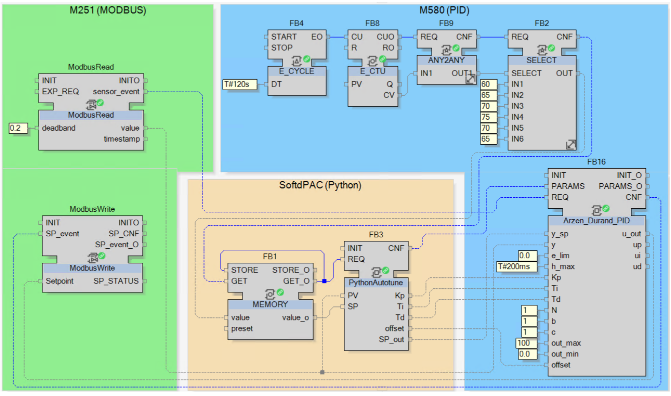

Fig. 6. Implementation of the control architecture

power to run all the monitoring and control routines, including the execution of the PID. In order to achieve an event-triggered behavior in the PID controller, the default PID function block implementation in the M580 cannot be used, since it implements a typical time-triggered PID oriented to be run in a cyclic task. For that reason, a new function block has been programmed in Structured Text to implement the event-triggered PID controller algorithm that was discussed in section 2.

The PID management layer runs on an Industrial PC (Magelis IoT Box), with a Linux operating system. A virtualized system known as Soft dPAC runs on this system in order to add IEC 61499 functionality to a PC. This technology makes possible that any type of device can become another control element within the distributed architecture. The optimization algorithms are implemented using Python programming language and encapsulated into a Function Block for its integration with the rest of the program.

Since these devices are synchronized using NTP, the network delay is known and does not need to be modeled. Moreover, in this work the experiments are carried out in a local network, so this delay will be negligible and it was found unnecessary to implement networked control techniques. Figure 6 shows the program of the architecture. The advantage of using the IEC 61499 for distributed control is that the whole architecture is programmed in the same layout but then each program block is assigned to its corresponding device.

4. EXPERIMENTAL RESULTS

To assess the performance of the proposed architecture, a series of experiments have been carried out in the system described in the previous section. Tank level control is carried out with small changes in the control setpoint in order to work with linearized models at different operating points. In order to compare the system operation with simpler configurations, the execution of the control system has been tested with three alternatives:

- A PID running with cyclic events triggered every 80 ms. The PID parameters, tuned for a setpoint of 50%, are fixed during the whole experiment (Fixed Cyclic).
- A PID also running with cyclic events but the PID parameters are tuned for each setpoint change (Autotune Cyclic).

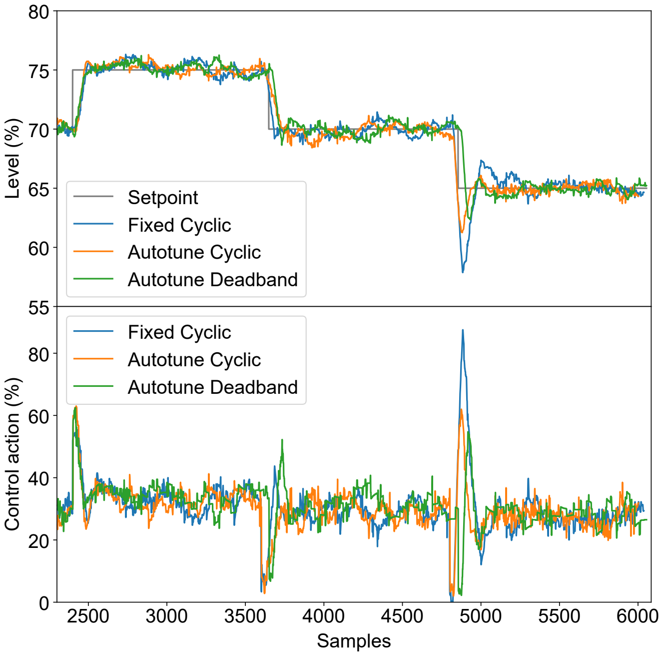

Fig. 7. Detail of the tank level and the action control for the different PID controllers.

- An event-triggered PID that takes full advantage the proposed architecture, using a deadband of 0.2% and tuning triggered after each setpoint change (Autotune Deadband). If the controlled value does not change more than 0.2% respect to the last sent value, no new event will be sent to the control layer. However, if this situation extends over 200ms, a new PID execution will be performed using the last sent value.

The experiment performs five small step changes to the setpoint starting from an initial operating point (60%). For the comparison among the resulting system responses, the values of different performance indices (Mousakazemi, 2021) are computed for each setup of the control loop. The selected indices are the integral of the absolute error (IAE), the integral of time multiplied Absolute Error Criterion (ITAE) and the integral absolute variation of control signal (IAVU). Table 2 shows the resulting performance indices. It can be seen that IAE and ITAE results are lower for the implementations where PID parameters are tuned, showing the advantage that a PID optimization layer such as the one proposed might provide. Regarding IAVU, the autotuned event-driven PID provides clearly superior results. These results are consistent with the conclusions achieved in Årzén (1999). It can also be seen that the proposed architecture allows to take advantage of an event-based approach without major control penalties, thanks to the autotuning process that is carried out in the PID Management layer in this case.

Furthermore, Figure 7 shows the detail of fragment of the experiment, so that the performance of the controllers can also be visually compared. Both the table and the figure show how performance is not degraded by the distributed implementation of the proposed approach. Instead, the potential usefulness of the continuous optimization and event-driven execution of the controller is highlighted.

Table 2. Comparison between the obtained control performance indices

Control	IAE	ITAE	IAVU
Fixed cyclic	8.06	2785.1	3272.34
Autotune cyclic	6.99	2522.71	3534.17
Autotune deadband	6.78	2676.31	2309.38

5. CONCLUSIONS

This work presents an architecture proposal for PID control and optimization in a distributed system, based on the recent IEC 61499 control standard. Taking advantage of its ability to communicate the different PLCs using a system based on event-driven data transmission, the control of a physical system is distributed into three layers. At the lowest layer, the tasks of acquiring and conditioning physical signals are carried out cyclically, but they are transmitted to the control algorithm through non-periodic events. At the second layer, the execution of an event-triggered PID is carried out. This PID is executed based on the events received, so it is necessary to consider the elapsed times between consecutive executions. For this reason, data are always transmitted with the timestamp associated with the event that generated them. Finally, a third layer is used for controller optimization. In this case, the Durand modification of the Årzén's event-based PID algorithm has been used in the second layer. Furthermore, Ziegler-Nichols has been used to tune the PID parameters for a first-order model with delay in the third layer. The viability of the proposed architecture has been successfully assessed through its implementation in commercially available industrial devices and its application to the SISO control of a tank level loop in a pilot plant.

Future lines of this work would be to implement other more advanced optimization algorithms at the third layer in order to improve the control parameters. Furthermore, the search for these control algorithms does not have to be restricted to the use of traditional PID tuning algorithms, but methods based on model predictive control could also be of interest. As a future interesting line, the implementation of methods from networked control theory can be highlighted, which might be needed to improve the response at large delays. In the case of this work, these considerations were neglected because delays are quite low since all the control devices are located in the same network.

REFERENCES

Domínguez, M., Pérez, D., Morán, A., Alonso, S., Prada, M.A., and Fuertes, J.J. (2022). Remote training in cybersecurity for industrial control systems. In *IFAC-PapersOnLine*, 17, 320–325. Elsevier B.V. doi: 10.1016/J.IFACOL.2022.09.299.

Dormido, S., Sánchez, J., and Kofman, E. (2008). Muestreo, control y comunicación basados en eventos. *Revista Iberoamericana de Automática e Informática industrial*, 5(1), 5–26. doi:10.1016/S1697-7912(08)70120-1.

Durand, S. and Marchand, N. (2009). Further results on event-based pid controller. In *2009 European control conference (ECC)*, 1979–1984. IEEE.

Fuertes, J., Prada, M.A., Domínguez, M., Reguera, P., Díaz, I., and Díez, A. (2008). Visualization of dynamic parameters of a multivariable system using self-organizing maps. *IFAC Proceedings Volumes*, 41(2), 11001–11006.

Fuertes, J.J., Prada, M.A., Rodríguez-Ossorio, J.R., González-Herbón, R., Pérez, D., and Domínguez, M. (2021). Environment for education on industry 4.0. *IEEE Access*, 144395–144405. doi: 10.1109/ACCESS.2021.3120517.

Garcia, C.A., Castellanos, E.X., Buele, J., Espinoza, J., Lanas, D., Beltran, C., and Garcia, M.V. (2018). MPC under IEC-61499 using low-cost devices for oil pipeline system. In *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, 659–664. IEEE.

Johansson, K.H. (2000). The Quadruple–Tank Process: A Multivariable Laboratory Process with an Adjustable Zero. *IEEE Transaction on Control Systems Technology*, 8(3), 456–465.

Liu, T., Wang, Q.G., and Huang, H.P. (2013). A tutorial review on process identification from step or relay feedback test. *Journal of Process Control*, 23(10), 1597–1623. doi:10.1016/j.jprocont.2013.08.003.

Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. *Journal of Industrial Information Integration*, 6, 1–10. doi: https://doi.org/10.1016/j.jii.2017.04.005.

Lyu, G. and Brennan, R.W. (2021). Towards IEC 61499-based distributed intelligent automation: A literature review. *IEEE Transactions on Industrial Informatics*, 17(4), 2295–2306. doi:10.1109/TII.2020.3016990.

Miguel-Escrig, O. and Romero-Pérez, J.A. (2018). Implementation and evaluation of event-based PID in the IEC-61499 standard. In *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, volume 1, 574–580. doi: 10.1109/ETFA.2018.8502574.

Mousakazemi, S.M.H. (2021). Comparison of the error-integral performance indexes in a GA-tuned PID controlling system of a PWR-type nuclear reactor point-kinetics model. *Progress in Nuclear Energy*, 132, 103604.

Muñoz de la Peña, D., Domínguez, M., Gomez-Estern, F., Reinoso, O., Torres, F., and Dormido, S. (2022). State of the art of control education. *Revista Iberoamericana de Automática e Informática industrial*, 117–131. doi: 10.4995/riai.2022.16989.

Vyatkin, V. (2011). IEC 61499 as enabler of distributed and intelligent automation: State-of-the-art review. *IEEE Transactions on Industrial Informatics*, 7(4), 768–781. doi:10.1109/TII.2011.2166785.

Zhang, X.M., Han, Q.L., Ge, X., Ding, D., Ding, L., Yue, D., and Peng, C. (2020). Networked control systems: a survey of trends and techniques. *IEEE/CAA Journal of Automatica Sinica*, 7(1), 1–17. doi:10.1109/JAS.2019.1911651.

Ziegler, B.J.G. and Nichols, N.B. (1942). Optimum settings for automatic controllers. *Journal of Fluids Engineering*. doi:10.1115/1.2899060.

Årzén, K.E. (1999). A simple event-based PID controller. *IFAC Proceedings Volumes*, 32(2), 8687–8692. doi:10.1016/S1474-6670(17)57482-0. 14th IFAC World Congress 1999, Beijing, Chia, 5-9 July.