

Avances en el control predictivo para controladores lógicos programables

Rogelio E. Rivero-Contreras^{a,b,*}, Jesús M. Zamarreño^{a,b}, Fernando Tadeo^{a,b}

^aDepartamento de Ingeniería de Sistemas y Automática, Universidad de Valladolid, Paseo Prado de la Magdalena, 3-5, 47011 Valladolid, España.

^bInstituto de Procesos Sostenibles, Universidad de Valladolid, C/Dr. Mergelina, S/N, 47011, Valladolid, España.

To cite this article: Rivero-Contreras, R.E., Zamarreño, J.M., Tadeo, F. 2026. Advances in model predictive control for programmable logic controllers. Revista Iberoamericana de Automática e Informática Industrial 23, 13-24. <https://doi.org/10.4995/riai.2025.22466>

Resumen

Este artículo resume diversas iniciativas para implementar el control predictivo (MPC) en controladores lógicos programables (PLCs), a partir de la experiencia acumulada, formulaciones de MPC utilizadas, algoritmos de optimización, sistemas de proceso considerados y los estándares de programación y marcas comerciales de dispositivos PLCs empleadas. Los estudios demuestran la viabilidad de implementar formulaciones de MPC clásicos junto con métodos de optimización en forma embebida. Destacan, en particular, los algoritmos de optimización como el método de Hildreth y de conjuntos activos, que han demostrado ser eficientes y codificables según el estándar IEC 61131-3. Además, los estudios resaltan la necesidad de reducir los requerimientos de memoria y cálculos computacionales, y que esto permita escalar estos algoritmos desde simulaciones hardware-in-the-loop (HiL) y procesos a escala de laboratorio hacia plantas industriales. Las tendencias actuales se orientan hacia la simplificación del uso de recursos computacionales, la mejora en la codificación de las formulaciones de MPC y los algoritmos de optimización, y la integración de estos en dispositivos modernos basados en internet de las cosas (PLCs-IoT).

Palabras clave: Controladores lógicos programables, control predictivo, optimización, control de procesos avanzado.

Advances in model predictive control for programmable logic controllers

Abstract

This article presents a synthesis of the various initiatives undertaken to implement predictive control (MPC) in programmable logic controllers (PLCs). It is based on an analysis of the accumulated experience, MPC formulations used, optimisation algorithms, process systems considered and the programming standards and commercial brands of PLC devices employed. The studies indicate the feasibility of implementing classical MPC formulations in conjunction with optimisation methods in an embedded form. It is particularly noteworthy that optimisation algorithms such as the Hildreth and active sets method have been demonstrated to be both efficient and codable in accordance with the IEC 61131-3 standard. Furthermore, studies emphasise the necessity to reduce memory and computational calculation requirements, which enables the scaling of these algorithms from hardware-in-the-loop (HiL) simulations and laboratory-scale processes to industrial plants. Current trends are moving towards simplifying the use of computational resources, improving MPC formulations and optimization algorithms, and integrating these into modern Internet of Things devices (PLCs-IoT).

Keywords: Programmable logic controllers, model predictive control, optimisation, advanced process control.

1. Introducción

La automatización de los procesos industriales ha experimentado un avance significativo desde sus inicios, a mediados

del siglo XX, hasta la actualidad (Samad et al., 2007). En principio, la automatización se centró en el uso de relés electro-mecánicos y controladores simples para realizar tareas repetitivas y minimizar la intervención humana. Sin embargo, con la

*Autor para correspondencia: rogelioemilio.rivero@uva.es

Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

llegada de los controladores lógicos programables (PLCs) en la década de 1960, la industria dio un paso significativo hacia la flexibilidad y la fiabilidad en el control de procesos. Los PLCs permitieron la programación y reconfiguración de las secuencias de control de una manera fácil, lo que permitió incrementar la eficiencia y la adaptabilidad de los sistemas industriales (Alphonsus and Abdullah, 2016). En las décadas siguientes, el desarrollo de los sistemas de control distribuido (DCS), y los avances en instrumentación y tecnologías de medición mejoraron la precisión y el control de los procesos industriales. Estos sistemas permitieron una supervisión y control más integrados y centralizados, optimizando la producción y las gestión de los recursos (Mohammadi, 2023).

Por otro lado, el control predictivo (MPC) ha emergido como una de las estrategias clave en la mejora de la producción industrial. Desde hace varias décadas, el MPC ha sido ampliamente estudiado tanto en el ámbito académico como en la industria, consolidándose como una técnica avanzada de control (García et al., 1989; Lee, 2011; Mayne, 2014). Esta estrategia de control se fundamenta en la resolución de un problema de optimización, cuyo objetivo es determinar las acciones óptimas para las variables manipuladas con el fin de llevar o mantener las variables controladas en sus valores de consigna. Para lograr esto, el MPC se basa en predicciones futuras del comportamiento del proceso (Camacho and Bordons, 2007a). Es de destacar que el MPC, a diferencia de otras técnicas de control, considera restricciones explícitas del sistema de proceso en el problema de optimización. No obstante, la implementación del MPC conlleva una carga computacional considerable debido a la complejidad de los algoritmos de optimización empleados (Saluk et al., 2018). Por esta razón, la mayoría de las aplicaciones de MPC y sus correspondientes métodos de optimización han sido desarrolladas en ordenadores tipo PC (Qin and Badgwell, 2003; Forbes et al., 2015). Esto se debe a la alta capacidad de procesamiento y almacenamiento de información que poseen este tipo de ordenadores, en comparación con otros dispositivos existentes en el mercado, y su facilidad de integración con los DCS y otras arquitecturas de control disponibles en la industria (Asadipooya and Safavi, 2016).

Las arquitecturas de control basadas en PLCs, DCS y PACs (controladores autómatas programables) se encuentran integradas en el estándar IEC 62264-1 (International Electrotechnical Commission, 2013b), el cual propone una estructura jerarquizada de los diversos elementos que participan en un sistema de gestión empresarial y control industrial. Estas arquitecturas se definen en el nivel 2, debido a que se considera que ejecutan estrategias de control continuo, discreto y batch. Las estrategias de MPC se incluyen en el nivel 3 del estándar IEC 62264-1, las cuales pueden ser ejecutadas en aquellos dispositivos adecuados para su correcta implementación, sin discernir que no puedan ser implementadas en los dispositivos mencionados en otros niveles que define el estándar.

El éxito del MPC se ha basado, principalmente, en la maximización de los rendimientos de los productos en los procesos productivos en los que se han implementado, relacionado de forma directa con la reducción del consumo energético y de las materias primas necesarias (Cutler and Ramaker, 1980; Rawlings and Amrit, 2009), aplicado a sistemas multivariables. Desde el punto de vista económico, la adquisición de una so-

lución comercial de este tipo de estrategias de control avanzado supone un coste elevado, que solo puede ser adoptado por empresas con altos niveles de rentabilidad. Esto ilustra la relevancia de poder adaptar esta estrategia de control avanzado en dispositivos existentes y proporcionar una implementación de bajo coste económico. De esta forma, algunas instituciones e investigadores han estudiado la posibilidad de implementar el MPC en los PLCs comerciales disponibles.

Esta revisión va a resumir las diversas iniciativas de investigación y sus avances hasta el momento. El artículo se estructura de la siguiente manera: la sección 2 introduce las principales definiciones y conceptos requeridos para comprender la temática abordada, incluyendo definiciones matemáticas, requisitos computacionales y de programación, y la configuración de los PLCs; la sección 3 presenta los estudios realizados en los diversos dispositivos comerciales existentes; la sección 4 resume las patentes generadas y registradas relacionadas con el tema y las aplicaciones comerciales disponibles; por último, la sección 5 resume en las conclusiones las principales contribuciones obtenidas y las posibles tendencias y futuros trabajos de investigación sobre el tema.

2. Principales definiciones del MPC y PLCs

2.1. Control Predictivo (MPC)

El control predictivo (MPC) se considera un método de control óptimo, que toma acciones basadas en la optimización de un criterio. Esta optimización se lleva a cabo en un intervalo de tiempo discreto futuro denominado *horizonte de predicción* N_2 . Además, el comportamiento futuro del sistema depende de las acciones obtenidas durante un *horizonte de control futuro* N_u en cada *periodo de muestreo* j , aplicando la técnica del *horizonte móvil* (Figura 1). Como resultado, en cada instante de muestreo se obtienen los valores de las acciones de las *variables manipuladas* u y comportamientos de las *variables controladas* y durante un *horizonte de predicción* N_2 , resolviendo un nuevo problema de optimización. Este intervalo en el que se obtienen las nuevas acciones de las variables manipuladas se denomina *horizonte de control* N_u (Camacho and Bordons, 2007a).

EL MPC requiere de un *modelo del proceso*, el cual es un modelo matemático que describe el comportamiento del sistema y que puede obtenerse a partir de ecuaciones de primeros principios o mediante técnicas de identificación de modelos aplicadas a datos experimentales obtenidos (Ljung, 2010). Dado que los problemas de optimización se resuelven mediante dispositivos electrónicos, se prefieren modelos matemáticos discretos a los continuos, siendo algunos de ellos: variables en espacio de estados (SS) (Rao and Arun, 1992), respuesta al escalón (Liu et al., 2013) o funciones de transferencias en retardo unitario ($Fcn(q^{-1})$) (Unbehauen and Rao, 1998), entre otros. Por otro lado, la *función de coste* indica el criterio a optimizar y, generalmente, toma la forma de una función cuadrática que considera las ponderaciones de las diferencias entre la salida prevista y la salida deseada de las variables controladas (*trayectoria de referencia* w), durante el *horizonte de coincidencia* (*desde* N_1 *hasta* N_2) y la ponderación de los cambios de las variables manipuladas Δu en el *horizonte de control* N_u . En algunos casos se considera el uso de *restricciones*, las cuales fijan los límites dentro los cuales debe operar el sistema, impuesto

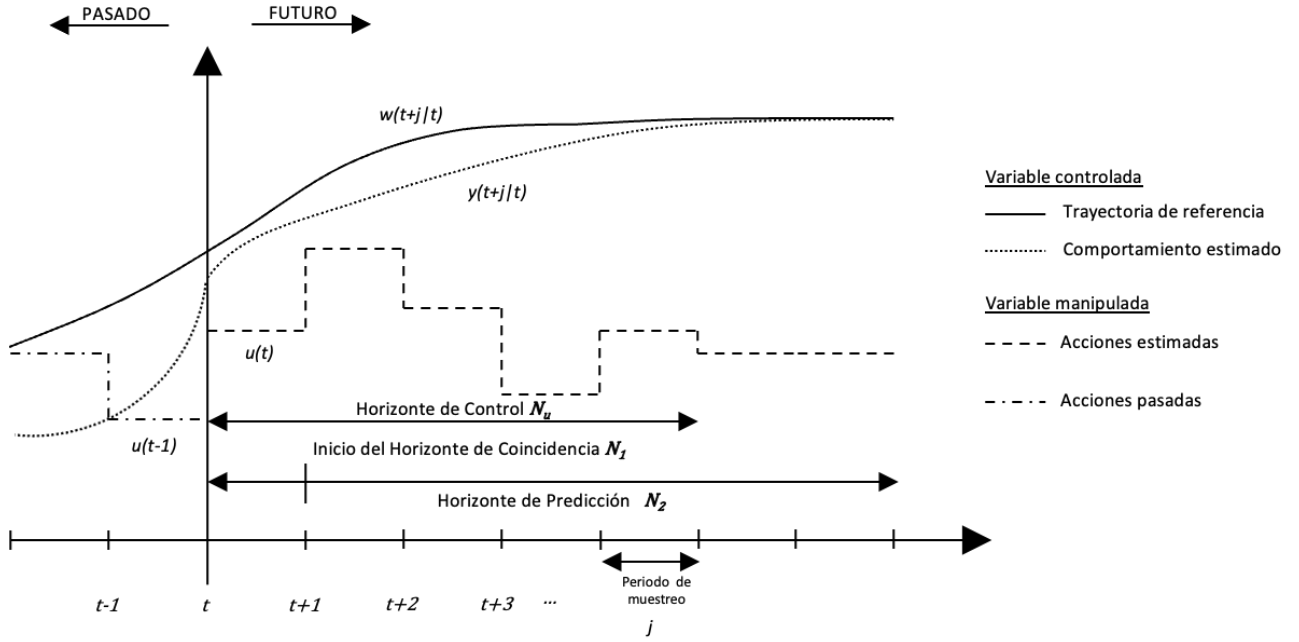


Figura 1: Principales definiciones del MPC.

por razones económicas, operativas o de seguridad; en el MPC estas restricciones se definen de manera genérica en los límites de operación de las variables controladas y , manipuladas u , y en los cambios de las acciones de control Δu . Por último, la *optimización* que se ejecuta en el control predictivo busca resolver un problema de optimización minimizando la función de coste sujeta a las restricciones establecidas, resultando, de manera general, en un problema de programación cuadrática (QP) para un sistema multivariable (Camacho and Bordons, 2007a) (Ecuaciones (1) y (2)). Aunque existen en la literatura formulaciones MPC de carácter más general, por motivos de simplicidad y una exposición más clara y divulgativa, se ha optado por expresar el problema de optimización del MPC a través de las Ecuaciones (1) y (2) en base a la formulación de (Camacho and Bordons, 2007a).

$$\min_{\Delta u_K(t+j|t)} \sum_{k=1}^N \sum_{j=N_{1k}}^{N_{2k}} \gamma_k (\hat{y}_k(t+j|t) - w_k(t+j|t))^2 + \sum_{K=1}^M \sum_{j=0}^{N_{uK}-1} \beta_K (\Delta u_K(t+j|t))^2 \quad (1)$$

$$\begin{aligned} \Delta u_K(t+j|t) &= 0, j \geq N_{uK}; K: 1, \dots, M \\ \Delta u_{Kmin} &\leq \Delta u_K(t+j|t) \leq \Delta u_{Kmax}, j: 0, \dots, N_{uK}-1 \\ u_{Kmin} &\leq u_K(t+j-1|t) \leq u_{Kmax}, j: 0, \dots, N_{uK}-1 \\ y_{kmin} &\leq \hat{y}_k(t+j|t) \leq y_{kmax}, j: N_{1k}, \dots, N_{2k}; k: 1, \dots, N \end{aligned} \quad (2)$$

donde N es el número total de variables controladas; k es un subíndice que representa cada una de las variables controladas; N_{1k} y N_{2k} son el horizonte inicial y horizonte final de predicción para cada variable controlada, respectivamente; $0 \leq \gamma_k$ es el peso ponderado de la variable controlada k en la función a minimizar; $\hat{y}_k(t+j|t)$ es el valor de la variable controlada predicho en

el periodo de muestreo $t+j$ con las medidas disponibles hasta el instante de tiempo actual t , en todo el horizonte de coincidencia, usando algún tipo de modelo matemático del comportamiento del sistema (modelo de respuesta salto, modelo en función de transferencias, etc.); $w_k(t+j|t)$ es el valor de la trayectoria de referencia deseada en el periodo de muestreo $t+j$ generada en el periodo de muestreo t ; M es el número total de variables manipuladas; K es un subíndice que representa cada una de las variables manipuladas; N_{uK} es el horizonte de control para cada variable manipulada; $0 \leq \beta_K$ es el peso ponderado de la variable manipulada K en el horizonte de control; $\Delta u_K(t+j|t)$ es el incremento de la acción de control para la variable manipulada K para el periodo de muestreo j calculado en el instante actual t ; Δu_{Kmin} y Δu_{Kmax} ($\Delta u_{Kmin} < \Delta u_{Kmax}$) son los incrementos mínimo y máximo para los valores de los incrementos de cada variable manipulada; u_{Kmin} y u_{Kmax} ($u_{Kmin} < u_{Kmax}$) son los límites inferior y superior para cada variable manipulada; y_{kmin} y y_{kmax} ($y_{kmin} < y_{kmax}$) son los límites inferior y superior para cada variable controlada.

Una vez resuelto el problema de optimización, de las N_{uK} soluciones $\Delta u_K(t+j|t)$ para cada variable manipulada K , aplicamos al proceso la primera de ellas ($u_K(t|t) = u_K(t-1|t) + \Delta u_K(t|t)$), descartando el resto y volvemos a repetir la resolución del problema de optimización en el periodo de muestreo siguiente (estrategia de horizonte móvil).

2.2. Formulaciones de MPC

Desde los inicios del MPC, las formulaciones desarrolladas para esta estrategia de control han evolucionado de manera significativa, generándose diversas variantes que se han podido adaptar a algunas aplicaciones industriales y de investigación. De manera general, entre las formulaciones de MPC más relevantes se encuentran: el MPC clásico, que utiliza un modelo lineal del sistema y resuelve un problema de optimización cuadrática (Muske and Rawlings, 1993); el MPC No Lineal

(NMPC) empleado cuando el modelo matemático del proceso es no lineal (Camacho and Bordons, 2007b); el MPC robusto, el cual garantiza el cumplimiento de las restricciones impuestas en presencia de perturbaciones (Bemporad and Morari, 2007); el MPC distribuido, el cual divide el problema de optimización en subproblemas más pequeños ejecutados por controladores comunicados entre sí, siendo útil para sistemas complejos (Christofides et al., 2013); el MPC adaptativo que ajusta el modelo del sistema en tiempo real, basándose en la identificación continua del sistema para manejar posibles variaciones en la dinámica del proceso (Rodellar and Sánchez, 1996); el MPC económico, que optimiza indicadores del sistema como el costo de operación o el consumo de energía, además de considerar la estabilidad y seguimiento de las referencias de las variables controladas (Ellis et al., 2014); el MPC estocástico, el cual maneja las incertidumbres en el modelo y las mediciones considerando probabilidades y distribuciones de las variables (Mesbah, 2016); el MPC basado en redes neuronales y aprendizaje automático, el cual usa estas técnicas para modelar el sistema para predecir su comportamiento, lo cual permite controlar sistemas altamente no lineales y complejos (Ren et al., 2022), entre otros.

A partir de la clasificación descrita previamente, se han desarrollado formulaciones específicas en base al tipo de modelo que se usa para predecir el comportamiento de la variable controlada. En el caso de los estudios desarrollados en los PLCs se pueden mencionar las formulaciones de control de matriz dinámica (DMC) el cual se basa en la utilización de una respuesta al escalón para modelar el comportamiento dinámico del proceso (Cutler and Ramaker, 1980); el control predictivo generalizado (GPC) el cual considera como modelo del proceso la media móvil regresiva integrada (ARIMA) (Clarke et al., 1987); el control predictivo funcional (PFC) que utiliza modelos simplificados obtenidos por diversos métodos utilizando el principio de descomposición y define la función de costo en términos de las variables controladas, simplificando el proceso de optimización (Richalet and O'Donovan, 2009); el MPC considerando funciones de Laguerre (LOMPC) para modelar y predecir el comportamiento del sistema, en el cual mediante una serie ortogonal se representan dinámicas complejas con un número reducido de parámetros, con el objetivo de mejorar la eficiencia computacional (Wang, 2004); el MPC óptimo (OMPC) en el cual el comportamiento óptimo sin restricciones del sistema se incorpora en las predicciones, y las restricciones se manejan como perturbaciones alrededor del valor óptimo (Rossiter et al., 1998); el MPC basado en el modelo Takagi-Sugeno-Kang modificado (MTSK) el cual permite modelar las reglas difusas SI-LUEGO sin aproximaciones, manteniendo el comportamiento del sistema (P. G. Mendes et al., 2022), entre otros.

Por otro lado, en el ámbito del MPC se distinguen dos enfoques principales: el MCP explícito y el MPC implícito. El MPC *explícito* divide el espacio de estados en regiones, en las cuales se define una función por cada región que determina la acción de control óptima; al ejecutarse, el controlador identifica la región en la que se encuentra el estado actual del sistema y aplica la función definida para esta, lo que reduce el tiempo de cálculo. No obstante, cuando los sistemas se vuelven complejos, debido a la cantidad de dimensiones definidas o restric-

ciones consideradas, se necesita un almacenamiento mayor, lo que limita la aplicación de este enfoque (Ju and Senlin, 2012). En contraste, el MPC *implícito* es la forma tradicional, en la cual se resuelve el problema de optimización en cada periodo de muestreo, permitiendo una gran flexibilidad y la capacidad de manejar sistemas complejos y no lineales, requiriendo un mayor coste computacional (Lee, 2011).

2.3. Algoritmos de Optimización

Los algoritmos de optimización permiten encontrar la solución óptima a un problema determinado, maximizando o minimizando una función de coste sujeta a una serie de restricciones. Estos algoritmos se pueden clasificar según el modelo del proceso utilizado, la estructura de la función de coste y los métodos matemáticos utilizados para resolverlos (Ali et al., 2015). Hoy en día, se han desarrollado cientos de algoritmos de optimización, los cuales se encuentran disponibles en artículos de investigación, y algunos de ellos se han implementado en aplicaciones informáticas comerciales o libres (Ploskas and Sahinidis, 2022).

En el caso de los estudios enfocados a los PLCs, se pueden mencionar el método de Hildreth el cual es un algoritmo utilizado para resolver problemas de optimización cuadrática sujeto a restricciones lineales (Hildreth, 1957); el algoritmo de programación cuadrática multiparamétrica (mp-QP) donde se definen ciertos parámetros, los cuales varían y van modificando la solución óptima, reduciendo el número de iteraciones y ahorrando recursos computacionales (Bemporad et al., 2002); el método de conjuntos activos que puede ser usado cuando se tienen restricciones lineales y no lineales, y es capaz de adaptarse a los cambios de los parámetros del problema de una iteración a otra, incluido en el solver qpOASES (Ferreau et al., 2014); los métodos primal-dual de primer orden incluidos en el solver FiOrdOs (Ullmann and Richter, 2011); el algoritmo FISTA diseñado para resolver problema de minimización de funciones compuestas, siendo una extensión del algoritmo iterativo de reducción y umbralización (ISTA), e introduce una mejora para acelerar la convergencia del proceso iterativo (Beck and Teboulle, 2009); el algoritmo ADMM que permite descomponer un problema de optimización complejo en partes sencillas, mediante descomposición de variables (Boyd and Vandenberghe, 2004); el algoritmo PCDM que permite resolver problemas con altas dimensiones mediante el enfoque de descenso de coordenadas en paralelo, siendo útil para problemas donde la función objetivo es separable Beck and Teboulle (2009); el algoritmo de árbol de búsqueda binaria (BST) en el cual se puede organizar y buscar información de una manera eficiente, a partir de soluciones precalculadas previamente (Ur Rehman et al., 2009); los algoritmos basados en estrategias evolutivas, los cuales son métodos estocásticos basados en iteraciones sucesivas (Kramer, 2010), dentro de los que destacan los algoritmos genéticos (Lambora et al., 2019); los algoritmos basados en ecuaciones diferenciales ordinarias (ODE) los cuales modelan el problema como un sistema dinámico que evoluciona hacia la solución óptima (Lin, 1991); el método de gradiente rápido de Nesterov (NFG) el cual se basa en un esquema de aceleración que combina iteraciones actuales y pasadas para predecir la mejor convergencia del gradiente y obtener una convergencia más rápida del problema de

optimización (Nesterov, 2012); el método basado en el gradiente proyectado en el cual el gradiente comienza moviéndose en la dirección opuesta al gradiente de la función objetivo, para tratar de encontrar un punto que minimice la función (Newton et al., 2018); los framework y aplicaciones CVXGEN (Mattingley and Boyd, 2011), FORCES (Domahidi et al., 2012), MINPACK (NETLIB, 2017), MIDACO (MIDACO-SOLVER, 2024) los cuales permiten ejecutar algoritmos de optimización que pueden ser personalizados por el usuario y ser compilados para ejecutarse en dispositivos externos (Mattingley and Boyd, 2011), entre otros.

2.4. Controladores lógicos programables (PLCs)

Los PLCs son dispositivos electrónicos utilizados en la automatización industrial para ejecutar estrategias de control en máquinas, procesos y sistemas de manufactura. Estos dispositivos están diseñados para ser robustos y duraderos, siendo capaces de operar en entornos industriales hostiles y ejecutar múltiples funciones de control en tiempo real (Koondhar et al., 2023).

Un PLC típico consta de varias partes fundamentales (Figura 2): la unidad central de procesamiento (CPU) que ejecuta el programa codificado, procesa la información y establece la comunicación con otros dispositivos; la memoria, que almacena el código de programa configurado, datos de usuarios y variables de proceso, pudiendo ser de diversos tipos como RAM, ROM, EEPROM y flash; los módulos de entradas y salidas (E/S) que reciben las señales asociadas a la instrumentación del proceso y las envían a los actuadores, teniendo señales de tipo digitales y/o analógicas en intensidad o voltaje; la fuente de alimentación que proporciona la energía necesaria para el funcionamiento del dispositivo y los módulos que lo integran, los módulos de comunicación que permiten la interacción con otros dispositivos y redes industriales existentes, entre otros (Bolton, 2015). El estándar IEC 61131 (International Electrotechnical Commission, 2025) en sus partes define las características, modelos e interfases de comunicación, lenguajes de programación, entre otros que deben seguir los dispositivos PLCs. No obstante, se espera que estos dispositivos evolucionen y vayan adoptando las especificaciones descritas en el estándar IEC 61499 (International Electrotechnical Commission, 2012), el cual describe una comunicación basada en eventos y la definición de bloques funcionales a ser utilizados en las arquitecturas de control industrial distribuidas.

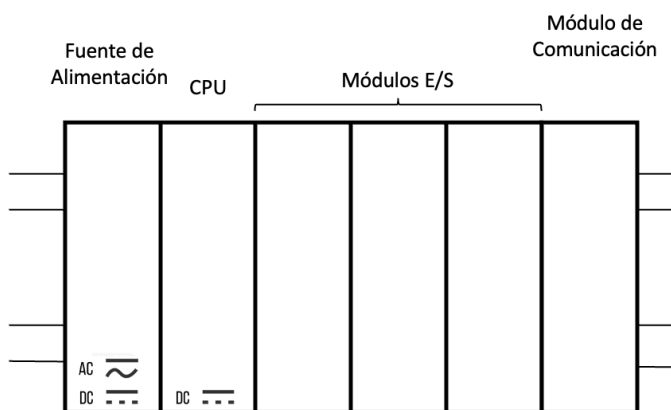


Figura 2: Componentes básicos de un PLC.

Los sistemas de control basados en PLCs permiten configurar diversas arquitecturas, en función de la ubicación y configuración de sus componentes. En la arquitectura centralizada, todos los componentes del sistema de control están ubicados en un solo lugar o chasis, lo que facilita la instalación y mantenimiento, reduciendo la latencia y disminuyendo los costos de comunicación, siendo ideal para sistemas pequeños o medianos donde los puntos de control se encuentran cercanos. En contraste, la arquitectura distribuida dispersa los componentes del sistema de control, conectados mediante una red de comunicación industrial, lo que permite una escalabilidad fácil, mejora la redundancia y fiabilidad del sistema (Bolton, 2015).

Hoy en día, existen numerosos fabricantes de PLCs en el mercado, ofreciendo una amplia gama de marcas y modelos adaptados a las diversas necesidades de la industria (Alphonso and Abdullah, 2016). En los últimos años, estos dispositivos han evolucionado de manera significativa, aumentando su capacidad de almacenamiento y procesamiento de la información, lo que ha permitido la incorporación de características adicionales y nuevas tecnologías asociadas a la industria 4.0. Esta evolución incluye la integración de capacidades de comunicación avanzadas, además de ethernet y las redes inalámbricas, de protocolos basados en OPC UA (International Electrotechnical Commission, 2020) e IoT (Mellado and Núñez, 2022), facilitando una mayor conectividad y control remoto. Esto ha permitido que los dispositivos PLCs puedan gestionar e integrar grandes volúmenes de datos en tiempo real, lo que los considera dispositivos relevantes para la implementación de soluciones inteligentes en la industria, mejorando la eficiencia y productividad en un entorno cada vez más competitivo. Además, los PLCs modernos están diseñados para ser más seguros, ofreciendo protección contra ciberataques y fallos de los sistemas de control, lo que garantiza la continuidad y confiabilidad de las operaciones industriales (Ramírez et al., 2022). Por otro lado, los simuladores de PLCs disponibles han evolucionado de tal manera que están comenzando a ofrecerse como dispositivos comerciales virtuales (Siemens, 2024), ofreciendo las mismas capacidades y desempeño de un dispositivo físico, y que puede ser utilizado desde entornos virtuales (Rivero-Contreras et al., 2025).

Los PLCs permiten ejecutar lógicas secuenciales, temporizadores, contadores, operaciones aritméticas, controladores PID, entre otros, mediante librerías existentes en las aplicaciones de programación y configuración. Inicialmente, la programación de los PLCs se realizó mediante el uso de diagramas de contacto o escalera, conocidos como Ladder, con el objetivo de replicar y facilitar la transición de los sistemas de control analógicos basados en relés a dispositivos digitales. Este enfoque permitió a los ingenieros y técnicos familiarizados con los sistemas de control mecánicos adaptarse a la nueva tecnología digital. En la actualidad, los PLCs comerciales han adoptado el estándar IEC 61131-3 (International Electrotechnical Commission, 2013a) para su programación. Este estándar define cuatro lenguajes de programación basados en textos y gráficos: lista de instrucciones (IL), un lenguaje de bajo nivel similar al ensamblador que permite una programación detallada y eficiente; texto estructurado (ST), un lenguaje de alto nivel similar a Pascal que facilita la escritura de programas complejos; diagramas Ladder (LD), basado en las lógicas de relés; diagrama de bloques funcionales (FBD), que utiliza bloques funcionales para

representar funciones y operaciones, ideal para aplicaciones de control continuo y procesamiento de señales. Esto ha permitido la interoperabilidad entre diferentes marcas y modelos de PLCs, facilitando la integración y expansión de los sistemas de control en diversas industrias (Molina et al., 2007).

Es de destacar que la mayoría de los algoritmos y las estrategias de control se han desarrollado, de manera tradicional, en entornos de simulación basados en ordenadores tipo PC. Estos entornos ofrecen la flexibilidad y potencia necesaria para el diseño y ajuste fino de algoritmos complejos. Sin embargo, la implementación de estos algoritmos y estrategias de control en los PLCs en sus lenguajes de programación nativos es poco intuitiva. Es por ello que se han desarrollado herramientas como el PLC Coder de MATLAB (MathWorks, 2024), la cual permite generar códigos, de manera automática, en el estándar IEC 61131-3 a partir de Simulink, facilitando la traducción directa de los algoritmos desarrollados en MATLAB; y CODESYS (CODESYS Group, 2025) el cual es un entorno de desarrollo integrado que permite programar los dispositivos PLCs, permitiendo ejecutar librerías implementadas en C, Matlab y otros lenguajes.

Al codificar cualquier estrategia de control en los PLCs, es necesario comprender la importancia que tiene la memoria de almacenamiento y el ciclo de scan del dispositivo. El espacio que ocupan las variables requeridas en el dispositivo depende del tipo de variable que se declare al momento de definirla y el arreglo de los datos que se considere; además, las líneas de códigos de funciones definidas por el usuario o bloques de librería de software que se deseen usar, reservan espacio de la capacidad máxima de almacenamiento y procesamiento de la información que tienen estos dispositivos. En dispositivos antiguos, la capacidad máxima de una base de datos de memoria, pudiendo definirse varias de estas, es de 64 kB, mientras que en los modelos modernos puede llegar a varios megabytes. Algunos bloques de memoria pueden ser retentivos, lo que significa que retienen su valor después de un reinicio del sistema. Por otro lado, el ciclo de scan es el proceso repetitivo en el cual un dispositivo PLC verifica la correcta configuración del mismo, lee las entradas, ejecuta el programa definido y actualiza las salidas, realizando esta secuencia de manera continua, con la finalidad de asegurar una respuesta en tiempo real a los cambios en el proceso controlado (Bolton, 2015).

3. Estudios realizados

La Tabla 1 proporciona un resumen de los diversos estudios de investigación relacionados con las estrategias de control MPC llevados a cabo en dispositivos PLCs comerciales (desde 2004 hasta 2025). Estos estudios se han realizado mediante simulaciones hardware-in-the-loop (HiL) y en plantas reales a escala de laboratorio usando dispositivos PLCs físicos, lo que permite evaluar resultados en diversos escenarios, antes de un posible escalado a plantas piloto y a nivel industrial. Es importante destacar que en la Tabla 1 solo se consideran implementaciones realizadas en PLCs, excluyendo a los PACs, al momento de publicarse los estudios realizados; no obstante, algunos modelos de dispositivos PLCs han evolucionado por lo que hoy en día son considerados PACs. Estos últimos combinan las capacidades de los PLCs tradicionales con una mayor flexibilidad

y capacidad en la programación y almacenamiento de información, logrando un desempeño similar a los DCS en cuanto a las tareas y aplicaciones que pueden ejecutar (Chen and Luyang, 2018).

Algunos estudios realizados optaron por implementar el MPC en su forma explícita, debido a las limitaciones de capacidad de almacenamiento y procesamiento de la información de los dispositivos utilizados. Estas restricciones técnicas plantearon la necesidad de desarrollar estrategias que permitieran ejecutar las pruebas sin interrupciones y evitar la realización de cálculos intensivos en tiempo real, para no comprometer el rendimiento del sistema. El objetivo principal de estos estudios fue garantizar la continuidad y la eficiencia de las pruebas, asegurando que las operaciones de control se llevaran a cabo de manera fluida, evitando sobrecargar los recursos limitados de los dispositivos. Para lograr estos objetivos, los investigadores consideraron una variedad de formulaciones de MPC y algoritmos de optimización que permiten obtener soluciones óptimas a partir de conjuntos de soluciones precargadas en los dispositivos PLCs. Dentro de estos estudios se encuentran los de (Valencia-Palomo and Rossiter, 2010) y (Valencia-Palomo and Rossiter, 2012), quienes implementaron la formulación de MPC conocida como LOMPC. De manera similar, (Valencia-Palomo and Rossiter, 2011a) utilizó una variante basada en OMPC, mientras que (P. G. Mendes et al., 2022) optó por la formulación de MPC MTSK. Todos estos estudios hicieron uso del optimizador mp-QP, demostrando su capacidad para gestionar las complejidades de la optimización en tiempo real dentro de las limitaciones del hardware disponible. Además, se realizaron implementaciones que utilizaron el optimizador BST en conjunto con la formulación de MPC clásico, como se observó en los estudios de (Rauová et al., 2011) y (Velagić and Šabić, 2014). A partir de estas investigaciones, se considera que las implementaciones de MPC explícito en PLCs son factibles y efectivas.

En un enfoque diferente, (Tarnawski et al., 2022) implementaron una estrategia innovadora que involucraba el uso de un servidor externo para resolver el problema de optimización, lo cual representó un avance significativo en la gestión de procesos complejos. Este servidor, con una capacidad de procesamiento significativamente mayor, realizaba los cálculos intensivos necesarios, liberando al PLC de esta carga computacional. Una vez obtenidos estos resultados, se transmitían al PLC a través de una red de comunicación. Esta red aseguraba una rápida transferencia de datos, permitiendo que el dispositivo PLC realizara ajustes en tiempo real, garantizando un funcionamiento óptimo en todo momento. La configuración desarrollada integraba de manera eficiente el servidor y el PLC, asegurando que el sistema funcionara de manera adecuada.

Los estudios restantes incluidos en la Tabla 1 se centraron en la implementación de MPC en su forma implícita, con el objetivo de que los algoritmos de optimización seleccionados se ejecuten en línea. Este enfoque es especialmente relevante dado que los PLCs se encuentran instalados en entornos industriales donde la adquisición de una solución de control avanzando basada en MPC no es viable por razones económicas o por limitaciones de acceso o conectividad. En este contexto, los estudios de (Laszczyk, 2004), (Metzger and Laszczyk, 2004), (Bolzoni, 2017), (Laszczyk, 2006) y (Kreutz et al., 2014) se destacaron por implementar formulaciones de MPC basadas en

Tabla 1: Implementaciones de MPC en PLCs

Autor(es)	PLC		Tipo de Prueba		Formulación MPC	Algoritmo de Optimización
	Marca	Serie y Modelo	HiL	Laboratorio		
(Laszczyk, 2004)	Allen-Bradley	ControlLogix 5550	X		PFC	Definido por PFC
(Metzger and Laszczyk, 2004)	Allen-Bradley	ControlLogix 5550	X		PFC	Definido por PFC
(Valencia-Palomo et al., 2009)	Allen-Bradley	SCL500	X		GPC	Definido por GPC
(Valencia-Palomo and Rossiter, 2010)	Allen-Bradley	SCL500	X		LOMPC	mp-QP
(Valencia-Palomo and Rossiter, 2011a)	Allen-Bradley	SCL500	X		OMPC	mp-QP
(Valencia-Palomo and Rossiter, 2011b)	Allen-Bradley	SCL500	X		GPC	Optimización por interpolación (definido en referencia)
(Valencia-Palomo and Rossiter, 2012)	Allen-Bradley	SCL500	X		LOMPC	mp-QP
(P. G. Mendes et al., 2022)	Allen-Bradley	Micro850	X		MTSK MPC	mp-QP
(Bolzoni, 2017)	Siemens	S7-300	X		PFC	Definido por PFC
(Huyck et al., 2012a)	Siemens	S7-300	X		DMC	Hildreth
			X		MPC clásico	Hildreth
						Método de conjuntos activos en qpOASES simplificado
(Huyck et al., 2014)	Siemens	S7-300	X		MPC clásico	Hildreth
			X			Método de conjuntos activos en qpOASES simplificado
(Huyck et al., 2012b)	Siemens	S7-300	X		MPC clásico	Hildreth
			X			Método de conjuntos activos en qpOASES simplificado
(Kreutz et al., 2014)	Siemens	S7-300	X		PFC	Definido por PFC
(Laszczyk, 2006)	Siemens	S7-300	X		PFC	Definido por PFC
(Mrosko and Miklovičová, 2012)	Siemens	S7-200	X		GPC	Definido por GPC
(Necoara and Clipici, 2013)	Siemens	S7-1200	X		MPC clásico	PCDM
(Rauová et al., 2011)	Siemens	S7-200	X		MPC clásico	BST
(Zielonacki and Tarnawski, 2024)	Siemens	S7-1200	X		MPC clásico	Estrategias evolutivas
(Rivero-Contreras et al., 2023b)	Siemens	S7-1200	X		MPC clásico	Algoritmos genéticos
(Rivero-Contreras et al., 2023a)	Siemens	S7-1500	X		MPC clásico	Algoritmos genéticos
(Gracia et al., 2025)	Siemens	S7-1500	X		MPC clásico	ADMM
(Yuwono and Schwung, 2023)	Siemens	ET200SP	X		MPC adaptativo	Optimización distribuida
(Binder et al., 2015)	ABB	AC500	X		MPC clásico	Método de conjuntos activos en qpOASES
			X			Métodos primal-dual en FiOrdOS
(Binder et al., 2014)	ABB	AC500	X		MPC clásico	Métodos primal-dual en FiOrdOS
(Kufolalor et al., 2017)	ABB	AC500	X		MPC clásico	Algoritmo 1 (definido en referencia)
(Kufolalor et al., 2014)	ABB	AC500	X		MPC clásico	Métodos primal-dual en FiOrdOS
(Purohit and Buch, 2015)	ABB	AC500	X		MPC clásico	Método de conjuntos activos en qpOASES
			X			Optimización convexa en CVXGEN
			X			Optimización distribuida en MIDACO
			X			Método de mínimos cuadrados en MINPACK SQP
(Torrissi et al., 2017)	ABB	SC800	X		NMPC	
(Seman et al., 2013)	B&R Automation	X20	X		OMPC	Método de conjuntos activos en qpOASES
(Carvalho Henriques et al., 2023)	B&R Automation	X20		X	DMC	No especificado
(Serkes and Gorla, 2021)	B&R Automation	X20		X	MPC clásico	No especificado
(Bonne et al., 2017)	Schneider Electric	TSX		X	MPC clásico	Método de conjuntos activos en qpOASES
			X			Optimizador ODE
(Krupa et al., 2018)	Schneider Electric	Modicon M340	X	X	MPC clásico	FISTA
(Krupa et al., 2021)	Schneider Electric	Modicon M340	X		MPC clásico	FISTA
			X			ADMM
(Krupa et al., 2020)	Schneider Electric	Modicon M340	X		MPC clásico	Método de conjuntos activos
(Pereira et al., 2015)	Schneider Electric	Modicon M340	X		MPC clásico	NFG
(Velagić and Šabić, 2014)	Schneider Electric	Modicon M340		X	MPC clásico	BST
(Plamowski, 2020)	Mitsubishi	FX5		X	DMC	No especificado
(Syaichu-Rohman and Sirius, 2011)	Mitsubishi	MELSEC-Q		X	MPC clásico	Algoritmo 2 definido en referencia
(Wojtulewicz and Ławryńczuk, 2018)	Mitsubishi	iQ-R		X	DMC	No especificado
(Käpernick and Graichen, 2014)	Festo	CECX-X-C1		X	NMPC	Gradiente descendente proyectado
(Hyl and Wagnerova, 2016)	Bachman	MC210		X	MPC clásico	Método de conjuntos activos en qpOASES
(Aslam et al., 2017)	Delta Group	DVP20	X		GPC	Definido por GPC
(Tange et al., 2020)	Fuji Group	MICREX SX		X	MPC Edge (definido en referencia)	No especificado
(Tarnawski et al., 2022)	GE Fanuc	RX3i	X		MPC adaptativo	Gradiente normalizado (definido en referencia)

PFC. Mientras tanto, (Valencia-Palomo et al., 2009), (Valencia-Palomo and Rossiter, 2011b), (Mrosko and Miklovičová, 2012) y (Hyl and Wagnerova, 2016) consideraron formulaciones de MPC basados en GPC. Estas formulaciones de MPC requieren menos cálculos computacionales debido a que simplifican la manera en que se resuelve la función de coste. Además, algunos de estos estudios optaron por no considerar restricciones en sus modelos, lo que simplificó aún más el proceso de implementación y permitió una mayor eficiencia en la ejecución en tiempo real.

El resto de las implementaciones de MPC implícito consideraron algoritmos de optimización en línea. La mayoría de las formulaciones de MPC se basaron en el MPC clásico, con algunas implementaciones en DMC (Bolzoni, 2017; Wojtulewicz and Ławryńczuk, 2018; Plamowski, 2020; Carvalho Henriques et al., 2023), OMPC (Seman et al., 2013), MPC SQP (Torrissi et al., 2017) y NMPC (Käpernick and Graichen, 2014). Res-

pecto a los algoritmos de optimización considerados, se puede indicar que la mayoría de los MPC clásicos implementaron en mayor medida los algoritmos de Hildreth, métodos de conjuntos activos en el solver qpOASES y FISTA, con algunas excepciones que tomaron en cuenta los optimizadores NFG, ADMM y algoritmos propios definidos en las referencias.

La codificación de estas formulaciones de MPC y algoritmos de optimización se realizó en su mayoría en el estándar IEC 61131-3 de forma nativa. No obstante, algunos investigadores (Huyck et al., 2012a,b, 2014) utilizaron la herramienta PLC Coder disponible en MATLAB, para convertir el algoritmo de métodos de conjuntos activos basados en el solver qpOASES, incluido en MATLAB, al estándar IEC 61131-3. Sin embargo, el uso de esta herramienta no garantiza que la conversión al estándar de los PLCs se realice de manera óptima, y en algunos casos se optó por versiones simplificadas de los algoritmos para poder ejecutarlos. Es importante destacar que el PLC Co-

der realiza la conversión de cualquier simulación de proceso, formulación de MPC y algoritmo de optimización, pudiendo ejecutarse o no de forma satisfactoria en los dispositivos PLCs. No obstante, el PLC Coder no considera el tamaño de la memoria de almacenamiento y de trabajo que tienen estos dispositivos, lo que en algunos casos impide incluso cargar el resultado del PLC Coder a los dispositivos PLCs considerados. Respecto a los dispositivos que se codifican mediante el entorno de desarrollo CODESYS, estos permitieron ejecutar sin inconvenientes los algoritmos de optimización que se incluyen en los solvers qpOASES, CVXGEN, FORCES, MINPACK, MIDACO y FiOrdOS, debido a su fácil integración que proporciona esta herramienta en los PLCs.

Algunos investigadores han adoptado un enfoque diferente para mejorar la implementación de los algoritmos de optimización en línea, considerando las limitaciones de memoria de almacenamiento de los dispositivos PLCs. Así, (Krupa et al., 2020, 2021) desarrollaron un módulo que se ejecuta en un ordenador PC y simplifica las ecuaciones de la formulación de MPC y algoritmo de optimización que deben cargarse al dispositivo PLC para que ejecute la optimización en línea. Por otro lado, (Rivero-Contreras et al., 2023b,a) definieron una metodología de programación, en la cual la gestión del almacenamiento de los datos y las funciones de programación que realizan los cálculos necesarios para ejecutar las formulaciones de MPC y los algoritmos de optimización se segmentan de tal manera que permiten la ejecución de cualquier algoritmo seleccionado para cualquier sistema de proceso, garantizando que no se supere las limitaciones de memoria de trabajo que tienen los dispositivos PLCs; además, consideran que el uso de la memoria retentiva permite ejecutar los cálculos requeridos sin la limitación que el ciclo de scan de los PLCs tiene por defecto.

En el ámbito del MPC, los sistemas estudiados se dividen en dos categorías: sistemas SISO (Entrada Simple, Salida Simple) y sistemas MIMO (Entradas Múltiples, Salidas Múltiples). Los sistemas SISO son más sencillos en la identificación del modelo, configuración y control. No obstante, los estudios recientes han puesto un mayor énfasis en los sistemas MIMO, y representan el 57 % de los estudios resumidos en la Tabla 1, debido a su relevancia en aplicaciones industriales donde es necesario optimizar los rendimientos manejando múltiples variables de manera simultánea. La complejidad de los sistemas MIMO radica en la interacción entre varias entradas y salidas, lo que requiere un enfoque de control más sofisticado. En estos sistemas donde el MPC ha demostrado ser especialmente útil, ejecutado en ordenadores PC, se pueden gestionar estas interacciones de manera efectiva, asegurando un rendimiento óptimo del proceso.

Los sistemas de proceso SISO considerados en los estudios mostrados en la Tabla 1 se resumen en la Tabla 2. Dentro de este grupo, se destaca que una gran proporción de los estudios se centran en el control de la temperatura. Esto se debe a que la temperatura, en comparación con otras variables de proceso como la presión o el caudal, tiene un comportamiento dinámico más lento, lo que permite que las estrategias de MPC pueden ser aplicadas con mayor eficacia, debido al tiempo que conlleva que los algoritmos de optimización generen una solución óptima para el comportamiento que debe tener la variable manipulada.

Tabla 2: Estudios realizados en sistemas de procesos SISO

Proceso	Cantidad
Control de temperatura en un intercambiador de calor	2
Control de temperatura en un sistema de calefacción con soplador centrífugo y resistencia calefactora	5
Control de temperatura en un sistema de ventilación HVAC	5
Control de temperatura en una unidad de incubación	5
Control de velocidad de un motor	6
Control de posición de un péndulo invertido	1
Control de caudal en un reactor químico	2
Control de presión en un compresor	1
Control de nivel en un tanque	2
Total	29

Por otro lado, la Tabla 3 proporciona un resumen de los sistemas MIMO. Estos sistemas son de particular interés debido a la complejidad inherente a la interacción entre las variables. En los estudios considerados, se consideran cantidades bajas de variables manipuladas y controladas, respectivamente, cuando se tienen plantas reales a escala de laboratorio. En cambio, cuando se tienen simulaciones HiL, los estudios consideran un mayor número de variables en función del sistema de proceso y la capacidad de almacenamiento del PLC seleccionado.

Tabla 3: Estudios realizados en sistemas de procesos MIMO

Proceso	Cantidad
Reactor químico	4
Planta química compuesta de reactores y separadores	1
Columna de destilación	1
Sistema de cuatro tanques interconectados	3
Sistema de distribución de agua potable	1
Unidad de fabricación de manufactura inteligente	1
Unidad de calentamiento y envasado de manufactura	1
Sistema de producción de petróleo submarino	1
Bomba sumergible eléctrica en un pozo petrolero	1
Separador compacto submarino	1
Tren de compresión en paralelo	1
Sistema de compresión de aire centrífugo	1
Sistema de acoplamiento elástico, compuesto por dos motores eléctricos	1
Grúa de laboratorio	1
Refrigerador criogénico	1
Planta simulada sin detalles	1
Sistema de calefacción en edificio	1
Sistema acondicionador de temperatura	2
Sistema de iluminación de laboratorio	1
Modelo de laboratorio con ventiladores y bulbos de voltaje	1
Total	26

En los estudios realizados, se ha utilizado una variedad de dispositivos PLCs comerciales, tal como se muestra en la Tabla 1, lo que refleja la diversidad de opciones disponibles por parte de los principales proveedores de este tipo de tecnología. Esta variedad no solo permite una evaluación comparativa de las capacidades de cada dispositivo en relación con la formulación de MPC y el algoritmo de optimización implementado, sino que facilita la identificación de las mejores opciones para un posible escalado a nivel industrial. Al considerar dispositivos de diferentes proveedores, se pueden obtener datos relevantes sobre el

rendimiento y las limitaciones específicas de cada PLC, lo que es crucial al momento de decidir cuál va a ser el más adecuado para aplicaciones industriales más complejas y exigentes. Además, se verifica el uso de códigos estandarizados, lo que permite la interoperabilidad e integración de los algoritmos desarrollados en los diversos PLCs comerciales.

Por último, los estudios se enfocaron en verificar el correcto funcionamiento de las formulaciones de MPC y los algoritmos de optimización seleccionados, evaluando rigurosamente el uso de la memoria de almacenamiento de los dispositivos PLCs y el tiempo necesario para que los algoritmos implementados obtuvieran las soluciones y convergencias de iteraciones necesarias para generar las mejores acciones de control. En general, estos estudios han demostrado que, a medida que aumenta la complejidad del sistema – ya sea por el mayor número de variables manipuladas, controladas y restricciones o por la extensión de los horizontes de predicción y control definidos en los parámetros del MPC– se incrementa el tiempo de cálculo y el requerimiento de almacenamiento para ejecutar de manera correcta las operaciones matemáticas requeridas. Esto es relevante, ya que sugiere la necesidad de optimizar tanto los algoritmos, criterios de programación, configuración y la infraestructura de hardware que deberían tener los dispositivos para que las soluciones generadas cumplan con los requisitos de rendimiento en aplicaciones industriales.

4. Patentes relacionadas y Aplicaciones Comerciales

4.1. Patentes generadas

La patente (Ji, 2014) describe un sistema y método para implementar MPC en un PLC. Este método permite realizar cálculos de control óptimos directamente en un bloque de función MPC en tiempo real, considerando las limitaciones computacionales inherentes a los PLCs tradicionales. Esta patente define una aplicación de ingeniería que permite generar modelos de predicción mediante una herramienta de identificación y ajusta los parámetros de la formulación de MPC y algoritmo de optimización, generando bloques predefinidos que se pueden descargar al PLC y ejecutarse en el dispositivo.

La formulación de MPC que es utilizada en la patente es el DMC con un modelo de respuesta salto, aunque sugiere que puede adaptarse a otros modelos como espacios de estados o funciones de transferencias en retardo unitario, en función de la formulación de MPC que se desee implementar. Además, la optimización se va a ejecutar en línea y considera una función de coste que incluye coeficientes de ponderación, lo que permite ajustar la optimización en función de las limitaciones y objetivos del proceso. El autor de esta patente parece estar relacionado con SIEMENS, lo que sugiere que puede estar optimizado para funcionar en el ecosistema de hardware y software de este proveedor.

4.2. Aplicaciones comerciales y librerías disponibles

Además de las herramientas PLC Coder de MATLAB y el entorno de desarrollo CODESYS, así como los desarrollos propios de investigadores en sus estudios, se han encontrados pocos componentes o librerías en software comerciales que indiquen un desarrollo avanzando del tema. Entre ellas, destaca la

librería de control predictivo (Schneider Electric, 2021) disponible en el software EcoStruxure Control Expert, compatible con dispositivos de Schneider Electric.

Por otro lado, Spiro Control ha presentado una solución denominada SPIRO-MPC (Spiro Control Ltd, 2022), en la cual el proveedor indica que se puede ejecutar control predictivo en dispositivos PLCs de manera nativa y mediante dispositivos edge de bajo costo. Esta solución indica que ejecuta los algoritmos de control mediante formulaciones de espacios de estados, y han sido codificadas utilizando CODESYS y ejecutándose en los dispositivos mediante el estándar IEC-61131-3.

5. Conclusiones

5.1. Principales contribuciones

Los estudios realizados han demostrado que la implementación de MPC en PLCs es viable, aunque presentan desafíos relacionados con la capacidad de almacenamiento y procesamiento de estos dispositivos. Por otro lado, se han identificado diversas formulaciones de MPC y algoritmos de optimización que pueden ejecutarse en estos dispositivos, considerando las limitaciones que presentan. En particular, las formulaciones de MPC como el DMC, GPC y PFC se han implementado con éxito en múltiples estudios, utilizando algoritmos de optimización como el método de Hildreth, los métodos de conjuntos activos, FISTA, ADMM, entre otros. Además, la mayoría de los estudios han codificado las formulaciones de MPC y algoritmos de optimización de forma nativa en el estándar IEC-61131-3, aunque en algunos casos, y debido a la complejidad que este presenta, se ha recurrido a herramientas como PLC Coder y CODESYS para facilitar la conversión del código.

Es de destacar que al implementar el MPC en sistemas MIMO, el incremento en el número de variables manipuladas y controladas, junto con la definición de horizontes de predicción y control más amplios, genera un aumento en los tiempos de cómputo y en los requerimientos de memoria. Por esta razón, algunos estudios han optado por enfoques como el MPC explícito, para reducir la carga computacional en tiempo real.

Finalmente, la diversidad de estudios y dispositivos utilizados indica que la implementación del MPC en PLCs se encuentra en constante evolución y desarrollo. Aunque los dispositivos PLCs modernos han mejorado en términos de capacidad de procesamiento y almacenamiento, sigue existiendo la necesidad de optimizar tanto los algoritmos de optimización como las estrategias de programación para garantizar implementaciones eficientes.

5.2. Tendencias

Las tendencias en el desarrollo del MPC en PLCs se orientan hacia la optimización de los recursos computacionales y la integración con tecnologías emergentes. En particular, se espera un avance en la implementación de algoritmos de optimización más eficientes, capaces de reducir los tiempos de cálculo y minimizar los requerimientos de memoria, sin comprometer la calidad del control. Asimismo, se espera que la adopción del estándar IEC 61499 en los dispositivos de control, basado en una arquitectura orientada a eventos y no de ciclos de tiempo, va a permitir superar las limitaciones de tiempos de cálculos requeridos que poseen los dispositivos en la actualidad.

Por otro lado, se hace necesario considerar el escalado industrial de estas implementaciones para verificar su eficacia en sistemas de procesos reales, lo que va a proporcionar información relevante sobre su rendimiento en condiciones operativas. Finalmente, la convergencia del MPC con tecnologías de industria 4.0 va a generar nuevas oportunidades. El uso de dispositivos PLCs habilitados para IoT, junto con la integración de soluciones edge computing va a facilitar la ejecución de formulaciones de MPC y algoritmos de optimización en dispositivos de bajo costo, ofreciendo un equilibrio entre rendimiento, costos y accesibilidad a la tecnología.

Agradecimientos

Este trabajo ha sido realizado parcialmente gracias al apoyo de la Conserjería de Educación de la Junta de Castilla y León y la UE-FEDER (CL-EI-2021-07, UIC 233), y el Ministerio de Ciencia e Innovación de España con el proyecto PID2021-123654OB-C31 (a-CIDit) de MCIN/AEI/10.13039/501100011033/UE-FEDER. El primer autor ha recibido soporte financiero de la convocatoria de contratos predoctorales 2019 de la Universidad de Valladolid y el Banco Santander.

Referencias

- Ali, M. M., Adewumi, A. O., Blamah, N., Falowo, O., 2015. Mathematical modeling and optimization of industrial problems. *Journal of Applied Mathematics* 2015, Article ID 438471, 3 pages, DOI: 10.1155/2015/438471.
- Alphonsus, E. R., Abdullah, M. O., 2016. A review on the applications of programmable logic controllers (PLCs). *Renewable and Sustainable Energy Reviews* 60, 1185–1205, DOI: 10.1016/j.rser.2016.01.025.
- Asadipooya, H., Safavi, A. A., 2016. Enhancement of model predictive control implementation on a DCS PCS7. In: 2016 24th Iranian Conference on Electrical Engineering (ICEE). IEEE, DOI: 10.1109/iranianee.2016.7585717.
- Aslam, S., Hannan, S., Sajjad, M. U., Zafar, M. W., 2017. PLC based model predictive control for industrial process control. *International Journal of ADVANCED AND APPLIED SCIENCES* 4 (6), 63–71, DOI: 10.21833/ijaas.2017.06.009.
- Beck, A., Teboulle, M., 2009. A Fast Iterative Shrinkage-Thresholding Algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences* 2 (1), 183–202, DOI: 10.1137/080716542.
- Bemporad, A., Morari, M., 2007. Robust model predictive control: A survey. Springer London, pp. 207–226, DOI: 10.1007/bfb0109870.
- Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E. N., 2002. The explicit linear quadratic regulator for constrained systems. *Automatica* 38 (1), 3–20, DOI: 10.1016/s0005-1098(01)00174-1.
- Binder, B. J. T., Kufoalor, D. K. M., Johansen, T. A., 2015. Scalability of QP solvers for embedded model predictive control applied to a subsea petroleum production system. In: 2015 IEEE Conference on Control Applications (CCA). IEEE, DOI: 10.1109/CCA.2015.7320771.
- Binder, B. J. T., Kufoalor, D. K. M., Pavlov, A., Johansen, T. A., 2014. Embedded model predictive control for an electric submersible pump on a programmable logic controller. In: 2014 IEEE Conference on Control Applications (CCA). IEEE, DOI: 10.1109/CCA.2014.6981402.
- Bolton, W., 2015. Programmable Logic Controllers. Elsevier, URL: <https://educate.elsevier.com/book/details/9780128029299>.
- Bolzoni, L. D. F., 2017. Implementación de controladores predictivos en dispositivos programables industriales. Trabajo de fin de máster, Escuela de Ingenierías Industriales. Universidad de Valladolid, URL: <http://uvadoc.uva.es/handle/10324/26101>.
- Bonne, F., Alamir, M., Bonnay, P., may 2017. Experimental investigation of control updating period monitoring in industrial PLC-based fast MPC: Application to the constrained control of a cryogenic refrigerator. *Control Theory and Technology* 15 (2), 92–108, DOI: 10.1007/s11768-017-6109-y.
- Boyd, S., Vandenberghe, L., Mar. 2004. Convex Optimization. Cambridge University Press. DOI: 10.1017/cbo9780511804441
- Camacho, E. F., Bordons, C., 2007a. Model Predictive control. Springer London, DOI: 10.1007/978-0-85729-398-5.
- Camacho, E. F., Bordons, C., 2007b. Nonlinear Model Predictive Control: An Introductory Review. Springer Berlin Heidelberg, pp. 1–16, DOI: 10.1007/978-3-540-72699-9_1.
- Carvalho Henriques, J. a. P., Neto, E. R., Magalhães de Paula Paiva, J. a. P., Souza, L. C., Lugli, A. B., Carvalho Rocha, T. V., Teixeira Rip, M. L., 2023. Model predictive control based on dynamics matrix embedded in a programmable logic controller. In: 2023 11th International Conference on Control, Mechatronics and Automation (ICCA). IEEE, DOI: 10.1109/iccma59762.2023.10374801.
- Chen, H., Luyang, X., 2018. Software architecture and framework for programmable automation controller: A systematic literature review and a case study. Thesis of master, Department of Software Engineering. Blekinge Institute of Technology.
- Christofides, P. D., Scattolini, R., de la Peña, D. M., Liu, J., 2013. Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering* 51, 21–41, DOI: 10.1016/j.compchemeng.2012.05.011.
- Clarke, D., Mohtadi, C., Tuffs, P., 1987. Generalized predictive control - Part I. the basic algorithm. *Automatica* 23 (2), 137–148, DOI: 10.1016/0005-1098(87)90087-2.
- CODESYS Group, 2025. CODESYS. Software. URL: <https://www.codesys.com/products/codesys-engineering.html>.
- Cutler, C. R., Ramaker, B. L., 1980. Dynamic matrix control. a computer control algorithm. In: 1980 Joint Automatic Control Conference. Vol. 1.
- Domahidi, A., Zraggen, A. U., Zeilinger, M. N., 2012. Efficient interior point methods for multistage problems arising in receding horizon control. In: 51st Conference on Decision and Control 2012 IEEE 51st IEEE Conference on Decision and Control (CDC). ETH Zurich, pp. 668–674, URL: <https://infoscience.epfl.ch/handle/20.500.14299/86360>.
- Ellis, M., Durand, H., Christofides, P. D., 2014. A tutorial review of economic model predictive control methods. *Journal of Process Control* 24 (8), 1156–1178, DOI: 10.1016/j.jprocont.2014.03.010.
- Ferreau, H. J., Kirches, C., Potschka, A., Bock, H. G., Diehl, M., 2014. qpOASES: a parametric active-set algorithm for Quadratic programming. *Mathematical Programming Computation* 6 (4), 327–363, DOI: 10.1007/s12532-014-0071-1.
- Forbes, M. G., Patwardhan, R. S., Hamadah, H., Gopaluni, R. B., 2015. Model predictive control in industry: Challenges and opportunities. *IFAC-PapersOnLine* 48 (8), 531–538, DOI: 10.1016/j.ifacol.2015.09.022.
- García, C. E., Prett, D. M., Morari, M., may 1989. Model predictive control: Theory and practice—a survey. *Automatica* 25 (3), 335–348, DOI: 10.1016/0005-1098(89)90002-2.
- Gracia, V., Krupa, P., Limon, D., Alamo, T., 2025. Practical MPC formulation implemented on a PLC: soft constraints and online-adaptable model. In: I Simposio CEA de los GT: Ingeniería de Control - Modelado, Simulación y Optimización - Educación en Automática. pp. 1–5.
- Hildreth, C., 1957. A quadratic programming procedure. *Naval Research Logistics Quarterly* 4, 79–85, DOI: 10.1002/nav.3800040113.
- Huyck, B., Callebaut, L., Logist, F., Ferreau, H. J., Diehl, M., De Brabanter, J., Van Impe, J., De Moor, B., 2012a. Implementation and experimental validation of classic MPC on programmable logic controllers. In: 2012 20th Mediterranean Conference on Control & Automation (MED). IEEE, pp. 679–684.
- Huyck, B., De Brabanter, J., De Moor, B., Van Impe, J. F., Logist, F., 2014. Online model predictive control of industrial processes using low level control hardware: A pilot-scale distillation column case study. *Control Engineering Practice* 28, 34–48, DOI: 10.1016/j.conengprac.2014.02.016.
- Huyck, B., Ferreau, H. J., Diehl, M., De Brabanter, J., Van Impe, J. F. M., De Moor, B., Logist, F., 2012b. Towards online model predictive control on a programmable logic controller: Practical considerations. *Mathematical Problems in Engineering* 2012, Article ID 912603, 20 pages, DOI: 10.1155/2012/912603.
- Hýl, R., Wagnerova, R., 2016. Design and realization of embedded model predictive controller with software support. In: 2016 17th International Carpathian Control Conference (ICCC). IEEE, DOI: 10.1109/carpathiancc.2016.7501105.
- International Electrotechnical Commission, 2012. IEC 61149-1. Function blocks - Part 1: Architecture. Standard. URL: <https://webstore.iec.ch/publication/5506>.
- International Electrotechnical Commission, 2013a. IEC 61131-3. Programmable controllers - Part 3: Programming languages. Standard. URL: <https://webstore.iec.ch/publication/4552>.
- International Electrotechnical Commission, 2013b. IEC 62264-1. Enterprise-

- control system integration – Part 1: Models and terminology. Standard. URL: <https://webstore.iec.ch/publication/6675>.
- International Electrotechnical Commission, 2020. IEC 62541. OPC Unified Architecture. Standard. URL: <https://webstore.iec.ch/publication/61114>.
- International Electrotechnical Commission, 2025. IEC 61131. Programmable controllers – all parts. Standard. URL: <https://webstore.iec.ch/publication/62427>.
- Ji, K., 2014. US20140297002A1 – system and method for implementing model predictive control in PLC. URL: <https://patents.google.com/patent/US20140297002A1/en>. Filed: 02/04/2013, Granted: 06/14/2016, Adjuncted Expiration: 11/22/2034.
- Ju, Z., Senlin, W., 2012. A review of explicit model predictive control. In: 2012 31st Chinese Control Conference CCC. pp. 4233–4238.
- Käpernick, B., Graichen, K., 2014. PLC implementation of a nonlinear model predictive controller. IFAC Proceedings Volumes 47 (3), 1892–1897, DOI: 10.3182/20140824-6-za-1003.00911.
- Koondhar, M. A., Kaloi, G. S., Junejo, A. K., Soomro, A. H., Chandio, S., Ali, M., 2023. The role of PLC in automation, industry and education purpose: A review. Pakistan Journal of Engineering, Technology & Science 11 (2), 22–31, DOI: 10.22555/pjets.v11i1.975.
- Kramer, O., 2010. A review of constraint-handling techniques for evolution strategies. Applied Computational Intelligence and Soft Computing 2010, Article ID 185063, 11 pages, DOI: 10.1155/2010/185063.
- Kreutz, M., Richalet, J., Mocha, K., Haber, R., 2014. Implementation of PFC (Predictive Functional Control) in a PLC (Programmable Logic Controller) for a HVAC (Heating, Ventilation and Air Conditioning) system. Journal of Physics: Conference Series 570 (3), Article ID 032003, 12 pages, DOI: 10.1088/1742-6596/570/3/032003.
- Krupa, P., Limon, D., Alamo, T., 2018. Implementation of model predictive controllers in programmable logic controllers using IEC 61131-3 standard. In: 2018 European Control Conference (ECC). IEEE, DOI: 10.23919/ecc.2018.8550126.
- Krupa, P., Limon, D., Alamo, T., 2021. Implementation of model predictive control in programmable logic controllers. IEEE Transactions on Control Systems Technology 29 (3), 1117–1130, DOI: 10.1109/tcst.2020.2992959.
- Krupa, P., Saraf, N., Limon, D., Bemporad, A., 2020. PLC implementation of a real-time embedded MPC algorithm based on linear input/output models. IFAC-PapersOnLine 53 (2), 6987–6992, DOI: 10.1016/j.ifacol.2020.12.424.
- Kufoalor, D. K. M., Richter, S., Imsland, L., Johansen, T. A., 2017. Structure exploitation of practical MPC formulations for speeding up first-order methods. In: 2017 IEEE 56th Annual Conference on Decision and Control (CDC). IEEE, DOI: 10.1109/cdc.2017.8263929.
- Kufoalor, D. K. M., Richter, S., Imsland, L., Johansen, T. A., Morari, M., Eikrem, G. O., 2014. Embedded model predictive control on a PLC using a primal-dual first-order method for a subsea separation process. In: 22nd Mediterranean Conference on Control and Automation. IEEE, DOI: 10.1109/med.2014.6961399.
- Lambora, A., Gupta, K., Chopra, K., 2019. Genetic algorithm – a literature review. In: 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon). IEEE, DOI: 10.1109/commitcon.2019.8862255.
- Laszczyk, P., 2004. A18: Advanced predictive control implementation on programmable logic-type controllers. IFAC Proceedings Volumes 37 (20), 105–110, DOI: 10.1016/s1474-6670(17)30579-7.
- Laszczyk, P., 2006. Predictive control libraryblock for SIMATIC-S7 PLC. IFAC Proceedings Volumes 39 (21), 208–213, DOI: 10.1016/s1474-6670(17)30185-4.
- Lee, J. H., 2011. Model predictive control: Review of the three decades of development. International Journal of Control, Automation and Systems 9 (3), 415–424, DOI: 10.1007/s12555-011-0300-6.
- Lin, T.-W., 1991. Solving optimization problems using ordinary differential equations. Journal of the Chinese Institute of Engineers 14 (4), 391–395, DOI: 10.1080/02533839.1991.9677350.
- Liu, T., Wang, Q.-G., 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.procont.2013.08.003.
- Ljung, L., 2010. Perspectives on system identification. Annual Reviews in Control 34 (1), 1–12, DOI: 10.1016/j.arcontrol.2009.12.001.
- MathWorks, 2024. Simulink PLC Coder. URL: <https://es.mathworks.com/products/simulink-plc-coder.html>. Visitado en agosto de 2024.
- Mattingley, J., Boyd, S., 2011. CVXGEN: a code generator for embedded convex optimization. Optimization and Engineering 13, 1–27, DOI: 10.1007/s11081-011-9176-9.
- Mayne, D. Q., 2014. Model predictive control: Recent developments and future promise. Automatica 50 (12), 2967–2986, DOI: 10.1016/j.automatica.2014.10.128.
- Mellado, J., Núñez, F., 2022. Design of an IoT-PLC: a containerized programmable logical controller for the industry 4.0. Journal of Industrial Information Integration 25, Article ID 100250, DOI: 10.1016/j.jii.2021.100250.
- Mesbah, A., 2016. Stochastic model predictive control: An overview and perspectives for future research. IEEE Control Systems 36 (6), 30–44, DOI: 10.1109/mcs.2016.2602087.
- Metzger, M., Laszczyk, P., 2004. Implementation of predictive model based control on programmable logic controller. IFAC Proceedings Volumes 37 (13), 1341–1346, DOI: 10.1016/s1474-6670(17)31414-3.
- MIDACO-SOLVER, 2024. MIDACO-SOLVER – numerical high-performance optimization software. URL: <https://www.midaco-solver.com>. Visitado en agosto de 2024.
- Mohammadi, S., 2023. Investigating the performance of DCS wide control systems. Journal of Engineering in Industrial Research 4 (1), 44–60, DOI: 10.48309/jeires.2023.1.5.
- Molina, F., Barbancho, J., León, C., Molina, A., Gómez, A., 2007. Using industrial standards on PLC programming learning. In: 2007 Mediterranean Conference on Control & Automation. IEEE, DOI: 10.1109/med.2007.4433652.
- Mrosko, M., Miklovičová, E., 2012. Real-time implementation of predictive control using programmable logic controllers. INTERNATIONAL JOURNAL OF SYSTEMS APPLICATIONS, ENGINEERING & DEVELOPMENT 6 (1), 106–113.
- Muske, K. R., Rawlings, J. B., 1993. Model predictive control with linear models. AIChE Journal 39 (2), 262–287, DOI: 10.1002/aic.690390208.
- Necoara, I., Clipici, D. N., 2013. A computationally efficient parallel coordinate descent algorithm for MPC: Implementation on a PLC. In: 2013 European Control Conference (ECC). IEEE, pp. 3596–3601, DOI: 10.23919/ECC.2013.6669390.
- Nesterov, Y., 2012. Gradient methods for minimizing composite functions. Mathematical Programming 140, 125–161, DOI: 10.1007/s10107-012-0629-5.
- NETLIB, 2017. MINPACK – routines for solving systems of non-linear equations. URL: <https://hpcc.umd.edu/hpcc/help/OLDSWsoftware/minpack.html>. Visitado en agosto de 2024.
- Newton, D., Yousefian, F., Pasupathy, R., 2018. Stochastic Gradient Descent: Recent Trends. INFORMS, pp. 193–220, DOI: 10.1287/educ.2018.0191.
- P. G. Mendes, T., Schnitman, L., dos Reis Nogueira, I. B., Almeida Peixoto Ribeiro, A. M., Rodrigues, A. E., Loureiro, J. M., Martins, M. A., 2022. A new Takagi-Sugeno-Kang model-based stabilizing explicit MPC formulation: An experimental case study with implementation embedded in a PLC. Expert Systems with Applications 210, Article ID 118369, DOI: 10.1016/j.eswa.2022.118369.
- Pereira, M., Limon, D., Muñoz de la Peña, D., Alamo, T., 2015. MPC implementation in a PLC based on Nesterov's fast gradient method. In: 2015 23rd Mediterranean Conference on Control and Automation (MED). IEEE, DOI: 10.1109/med.2015.7158777.
- Plamowski, S., 2020. Model predictive control of a dynamic system with fast and slow dynamics: Implementation using PLC. In: Advanced, Contemporary Control. Advances in Intelligent Systems and Computing. Vol. 1196. Springer International Publishing, pp. 1104–1115, DOI: 10.1007/978-3-030-50936-1_92.
- Ploskas, N., Sahinidis, N. V., 2022. Review and comparison of algorithms and software for mixed-integer derivative-free optimization. Journal of Global Optimization 82, 433–462, DOI: 10.1007/s10898-021-01085-0.
- Purohit, A., Buch, J., 2015. Evaluation of optimization solvers on programmable logic controller. In: 2015 IEEE Conference on Control Applications (CCA). IEEE, DOI: 10.1109/cca.2015.7320684.
- Qin, S., Badgwell, T. A., 2003. A survey of industrial model predictive control technology. Control Engineering Practice 11 (7), 733–764, DOI: 10.1016/s0967-0661(02)00186-7.
- Ramirez, R., Chang, C.-K., Liang, S.-H., 2022. PLC cyber-security challenges in industrial networks. In: 2022 18th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA). IEEE, DOI: 10.1109/mesa55290.2022.10004463.
- Rao, B., Arun, K., 1992. Model based processing of signals: a state space approach. Proceedings of the IEEE 80 (2), 283–309, DOI: 10.1109/5.123298.
- Rauová, I., Valo, R., Kvasnica, M., Fikar, M., 2011. Real-time model predictive control of a fan heater via PLC.
- Rawlings, J. B., Amrit, R., 2009. Nonlinear Model Predictive Control Towards New Challenging Application. Springer Berlin Heidelberg, DOI:

- 10.1007/978-3-642-01094-1-10.
- Ren, Y. M., Alhajeri, M. S., Luo, J., Chen, S., Abdullah, F., Wu, Z., Christofides, P. D., 2022. A tutorial review of neural network modeling approaches for model predictive control. *Computers & Chemical Engineering* 165, Article ID 107956, DOI: 10.1016/j.compchemeng.2022.107956.
- Richalet, J., O'Donovan, D., 2009. *Predictive Functional Control*. Springer London, DOI: 10.1007/978-1-84882-493-5.
- Rivero-Contreras, R., Merino, A., Zamarreño, J. M., Vila, C., Tadeo, F., 2025. A virtual environment with programmable logic controllers: an application in the evaluation of control strategies. *Revista Iberoamericana de Automática e Informática Industrial* 22 (1), DOI: 10.4995/riai.2024.20881.
- Rivero-Contreras, R., Zamarreño, J. M., Tadeo, F., 2023a. Implementation of multivariable predictive control strategies (MPC) in programmable logic controllers (PLCs). In: *Workshop on Simulation and Optimization for Sustainable Engineering*. pp. 33–34, DOI: 10.22429/euc2023.026.
- Rivero-Contreras, R., Zamarreño, J. M., Tadeo, F., 2023b. Multivariable model predictive control (MPC) strategies in programmable logic controllers (PLCs). In: *WCCE11 - 11th World Congress of Chemical Engineering*. p. 320, URL: <https://www.wcce11.org/wc/template/Proceedings-Abstracts.-WCCE11.pdf>
- Rodellar, J., Sánchez, J. M., 1996. *Adaptive Predictive Control*. Prentice-Hall International Series in Systems and Control Engineering, NJ: Prentice Hall.
- Rossiter, J., Kouvaritakis, B., Rice, M., 1998. A numerically robust state-space approach to stable-predictive control strategies. *Automatica* 34 (1), 65–73, DOI: 10.1016/S0005-1098(97)00171-4.
- Saltik, M. B., Özkan, L., Ludlage, J. H., Weiland, S., den Hof, P. M. V., 2018. An outlook on robust model predictive control algorithms: Reflections on performance and computational aspects. *Journal of Process Control* 61, 77–102, DOI: 10.1016/j.jprocont.2017.10.006.
- Samad, T., McLaughlin, P., Lu, J., 2007. System architecture for process automation: Review and trends. *Journal of Process Control* 17 (3), 191–201, DOI: 10.1016/j.jprocont.2006.10.010.
- Schneider Electric, 2021. *Predictive Control Library*. URL: https://www.se.com/id/en/download/document/Predictive_Control_Library/. Visitado en febrero de 2025.
- Seman, P., Juhas, M., Tkac, L., Honek, M., 2013. New possibilities of industrial programming software. In: *2013 International Conference on Process Control (PC)*. IEEE, DOI: 10.1109/pc.2013.6581456.
- Serkies, P., Gorla, A., 2021. Implementation of PI and MPC-Based speed controllers for a drive with elastic coupling on a PLC controller. *Electronics* 10 (24), Article ID 3139, DOI: 10.3390/electronics10243139.
- Siemens, 2024. *SIMATIC S7-1500V familiar functionalities, completely virtual*. URL: <https://www.siemens.com/global/en/products/automation/systems/industrial/plc/simatic-s7-1500/virtual-plc.html>. Visitado en febrero de 2024.
- Spiro Control Ltd, 2022. *SPIRO-MPC*. URL: <https://spirocontrol.com/home/products-and-services/spiro-mpc-2/>. Visitado en febrero de 2025.
- Syaichu-Rohman, A., Sirius, R., 2011. Model predictive control implementation on a programmable logic controller for DC motor speed control. In: *2011 International Conference on Electrical Engineering and Informatics*. IEEE, DOI: 10.1109/iceei.2011.6021841.
- Tange, Y., Kiryu, S., Matsui, T., 2020. Implementation of edge-type model predictive control on generic programmable logic controller compliant to IEC61131-3 standard. *IEEE Transactions on Electronics, Information and Systems* 140 (9), 1057–1063, DOI: 10.1541/ieej.iss.140.1057.
- Tarnawski, J., Kudelka, P., Korzeniowski, M., 2022. Advanced control with PLC-Code generator for aMPC controller implementation and cooperation with external computational server for dealing with multidimensionality, constraints and LMI based robustness. *IEEE Access* 10, 10597–10617, DOI: 10.1109/access.2022.3142054.
- Torresi, G., Grammatico, S., Cortinovis, A., Mercangöz, M., Morari, M., Smith, R. S., 2017. Model predictive approaches for active surge control in centrifugal compressors. *IEEE Transactions on Control Systems Technology* 25 (6), 1947–1960, DOI: 10.1109/tcst.2016.2636027.
- Ullmann, F., Richter, S., 2011. *FiOrdOs: A Matlab toolbox for C-Code generation for first order methods*. URL: <http://hdl.handle.net/20.500.11850/60549>.
- Unbehauen, H., Rao, G., 1998. A review of identification in continuous-time systems. *Annual Reviews in Control* 22, 145–171, DOI: 10.1016/S1367-5788(98)00015-7.
- Ur Rehman, I., Khan, S. U. R., Khiyal, M. S. H., 2009. A survey on maintaining binary search tree in optimal shape. In: *2009 International Conference on Information Management and Engineering*. IEEE, DOI: 10.1109/icime.2009.128.
- Valencia-Palomo, G., Hilton, K., Rossiter, J., 2009. Predictive control implementation in a PLC using the IEC 1131.3 programming standard. In: *2009 European Control Conference (ECC)*. IEEE, DOI: 10.23919/ecc.2009.7074588.
- Valencia-Palomo, G., Rossiter, J., 2010. PLC implementation of a predictive controller using Laguerre functions and multi-parametric solutions. In: *UKACC International Conference on Control 2010. Institution of Engineering and Technology*, DOI: 10.1049/ic.2010.0440.
- Valencia-Palomo, G., Rossiter, J., 2011a. Efficient suboptimal parametric solutions to predictive control for PLC applications. *Control Engineering Practice* 19 (7), 732–743, DOI: 10.1016/j.conengprac.2011.04.001.
- Valencia-Palomo, G., Rossiter, J., 2011b. Programmable logic controller implementation of an auto-tuned predictive control based on minimal plant information. *ISA Transactions* 50 (1), 92–100, DOI: 10.1016/j.isatra.2010.10.002.
- Valencia-Palomo, G., Rossiter, J., 2012. Novel programmable logic controller implementation of a predictive controller based on Laguerre functions and multiparametric solutions. *IET Control Theory & Applications* 6 (8), 1003–1014, DOI: 10.1049/iet-cta.2010.0430.
- Velagić, J., Šabić, B., 2014. Design, implementation and experimental validation of explicit MPC in programmable logic controller. In: *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*. IEEE, DOI: 10.1109/isie.2014.6864592.
- Wang, L., 2004. Discrete model predictive controller design using Laguerre functions. *Journal of Process Control* 14 (2), 131–142, DOI: 10.1016/S0959-1524(03)00028-3.
- Wojtulewicz, A., Ławryńczuk, M., 2018. Computationally efficient implementation of dynamic matrix control algorithm for very fast processes using programmable logic controller. In: *2018 23rd International Conference on Methods & Models in Automation & Robotics (MMAR)*. IEEE, DOI: 10.1109/mmar.2018.8486132.
- Yuwono, S., Schwung, A., 2023. Model predictive control with adaptive PLC-based policy on low dimensional state representation for industrial applications. In: *2023 31st Mediterranean Conference on Control and Automation (MED)*. IEEE, DOI: 10.1109/med59994.2023.10185716.
- Zielonacki, K., Tarnawski, J., 2024. PLC-based implementation of stochastic optimization method in the form of evolutionary strategies for PID, LQR, and MPC control. *International Journal of Control, Automation and Systems* 22 (6), 1846–1855, DOI: 10.1007/s12555-023-0869-6.