

IMPLEMENTACIÓN DE UN ALGORITMO DE CONTROL PREDICTIVO EN ESPACIO DE ESTADOS SOBRE UNA PLATAFORMA DE SIMULACIÓN DESARROLLADA EN MATLAB[®]

Julio C. Zambrano^{1,*} y Ana I. González²

Resumen

El control predictivo basado en modelo (MPC) es una metodología que se enmarca dentro de la ingeniería de control moderno y que desde sus inicios ha venido evolucionando como una herramienta capaz de mitigar mayormente los problemas que enfrenta la industria desde el enfoque de la teoría de control. En su esencia la estrategia de control predictivo hace uso de un modelo matemático interno y de una estrategia de optimización para predecir las salidas del sistema dentro de un intervalo de tiempo al que se le denomina horizonte de predicción. La formulación que sustenta el algoritmo demanda un coste computacional elevado y es por ello que desde sus inicios esta estrategia ha tenido su nicho de implementación en ordenadores que agilicen y soporten un cálculo matricial considerable. En el presente artículo se detalla el análisis matemático y el procedimiento para la implementación de un algoritmo de control predictivo bajo MATLAB[®], además, se presenta los resultados de la aplicación a una planta de simple entrada y simple salida (SISO) y a una planta de múltiples entradas y múltiples salidas (MIMO). En ambos casos se incorporan restricciones a las variables del proceso, atendiendo a una de las características más atractivas de esta estrategia.

Palabras clave: Espacio de estados, integrador embebido, optimización, restricciones.

Abstract

Model Predictive Control (MPC) is a methodology in the field of modern control engineering. Since its beginnings it has progressively evolved to become a tool capable of largely mitigating the problems facing industry from the control theory point of view. Essentially, the predictive control strategy uses an internal mathematical model and an optimization strategy to predict system outputs within a time period known as the prediction horizon. The formulation supporting the algorithm has a high computational cost. This is why this strategy has found its niche since the start in computers which speed up and support substantial matrix calculations. This paper details the mathematical analysis and procedure for implementation of a predictive control algorithm under MATLAB[®]. Also presented are the results of its application to a single-input single-output (SISO) plant and a multiple-input multiple-output (MIMO) plant. In both cases, restrictions are incorporated in the process variables, regarded as one of the most attractive features of this strategy.

Keywords: State space, embedded integrator, optimization, restrictions.

^{1,*}Ingeniero electrónico, egresado del programa de Maestría de Automatización y Control Industrial, ESPOL, Guayaquil, Ecuador; docente de la Universidad Politécnica Salesiana Sede Cuenca. Autor para correspondencia ✉: jzambrano@ups.edu.ec

²Ingeniera en Máquinas Computadoras, Máster en Automática, Doctora en Tecnologías de la Información de la Facultad de Ciencias de la Universidad de Valladolid, España, profesora titular del Departamento de Automática y Computación, Facultad Eléctrica, ISPJAE, Cuba, vicedecana docente Facultad Eléctrica.

Recibido: 20-05-2013, Aprobado tras revisión: 10-06-2013.

Forma sugerida de citación: Zambrano J. y González, A. (2013). "Implementación de un algoritmo de control predictivo en espacio de estados sobre una plataforma de simulación desarrollada en Matlab[®]". INGENIUS. N.º9, (Enero-Junio). pp. 5-14. ISSN: 1390-650X.

1. Introducción

El control predictivo basado en modelo (MPC) se describe como una estrategia alrededor de ciertas ideas comunes que constituyen el fundamento de dicha técnica. Esta estrategia de control avanzado utiliza un modelo para predecir las salidas del proceso y calcular la acción de control futura a través de la minimización de una función de coste. La idea genérica de MPC fue introducida a través de la industria petroquímica a finales de los 70 y presentada como control predictivo heurístico basado en modelo MPHC [1]. Desde sus inicios ha revolucionado la ingeniería de control dando solución a procesos con comportamiento dinámico complejo y es por esta razón que en los últimos años su aplicación se ha extendido hacia otros sectores de la industria. MPC presenta grandes ventajas con respecto al control tradicional y a otras técnicas de control moderno. Entre sus principales ventajas se puede citar la gran naturalidad para tratar con sistemas multivariados y la posibilidad de incorporar restricciones. Por otra parte, la desventaja más considerable es la complejidad del algoritmo lo cual requiere un coste computacional elevado. Por la razón anterior su nacimiento se ha dado en procesos con dinámica lenta; sin embargo, el avance de la ciencia y la tecnología está dejando de lado este inconveniente y así lo afirma [2] quien manifiesta que: *“en los últimos diez años el incremento de la velocidad de hardware conjuntamente con las mejoras de algoritmos de optimización ha incrementado en un factor 10^6 el tiempo de solución del problema de optimización”*.

La evolución de MPC se ve reflejada en las variantes que oferta (ver [3]). Por la facilidad de implementar el algoritmo generalmente estas variantes utilizan modelos de predicción lineales (ver [4], [5] y [6]). No obstante, en los últimos años se ha dado una gran aceptación a algoritmos que utilizan modelos no lineales. En [7], [8] y [9] se puede encontrar abundante literatura en cuanto al tema. Todas estas variantes se han establecido en base al tipo de modelo de predicción utilizado. Por ejemplo DMC (control con matriz dinámica) utiliza como modelo de predicción una matriz de coeficientes de la respuesta al escalón del sistema.

Esta técnica de control apareció a finales de los setenta y se atribuye la idea a Cluter y Ramarker, [10] trabajadores de la Shell Oil Co. En [4] se puede ver en detalle la formulación del algoritmo; por otra parte [11] se presenta la implementación de un DMC en una caldera.

Otra variante de MPC constituye el control predictivo generalizado (GPC) propuesto por Clarke et al., en 1987 [12] y hoy en día es una de las variantes más populares dentro del control predictivo. GPC utiliza un modelo CARIMA (Controller Autoregressive and Integrated Moving-Average) para representar el

comportamiento de la planta. Este modelo es óptimo cuando las perturbaciones del sistema no son estacionarias. En [13] se puede ver la incorporación de restricciones de entrada y salida en la formulación del GPC. Por otra parte en [14] se puede ver los resultados de aplicaciones en un molino de cemento y en una torre de secado por pulverización.

Existen variantes de MPC que son utilizadas en plataformas patentadas que funcionan bajo un nombre comercial. Por ejemplo DMCplus[®] es una plataforma desarrollada por ASPENTECH para implementar controladores predictivos [15]. Esta herramienta utiliza modelos en espacio de estados y brinda la posibilidad de realizar identificación mediante la técnica de subespacios de estado. Según [16] DMCplus[®] integra un sistema *off-line* compatible con Microsoft[®] para diseño y simulación de sistemas. La plataforma de ejecución *on-line* es compatible con plataformas DEC[®], IBM[®] y UNIX[®]. DMCplus[®] está siendo ampliamente utilizada en la industria del petróleo. En [17] y [18] se dejan ver breves reportes de su implementación. Otra variante comercial de MPC lo constituye Profit[®] Controller, plataforma desarrollada por Honeywell [19] que brinda la posibilidad de implementar controladores predictivos para sistemas MIMO altamente interactivos. Se basa en la tecnología de control predictivo multivariable robusto RMPCT (Robust Multivariable Predictive Control Technology[®]) patentada por la misma Honeywell. Más detalles sobre esta plataforma se pueden encontrar en [20]. Por otra parte en [21] y [22] se muestra el resultado de haber utilizado Profit[®] Controller sobre plantas industriales.

CpmPlus Expert Optimizer[®] es otra variante comercial de MPC. Esta plataforma desarrollada por ABB [23] soporta muchas tecnologías de control, incluyendo el control predictivo basado en modelo. Utiliza una programación fundamentada en diagramas de bloques e interconexiones físicas. La conexión entre la planta y la plataforma se realiza mediante OPC (OLE para Control de Procesos). Adicionalmente, brinda la opción de estimar estados del proceso mediante la técnica de filtro de Kalman [24]. Esta plataforma es más comúnmente utilizada en procesos de fabricación de cemento [25] y plantas de flotación [26].

2. Fundamento del control predictivo basado en modelo

2.1. Basamento teórico

El objetivo de MPC consiste en el hallazgo de una trayectoria futura de la variable manipulada u . En teoría el fundamento de la estrategia se detalla a continuación.

Se fija un horizonte de predicción N para el cual se van a predecir en cada instante k_i las salidas futuras. Esto se logra utilizando un modelo de predicción que describe el comportamiento de las variables del proceso a controlar y el cual reside en el controlador. Las predicciones dependen de los valores conocidos hasta el instante k_i y de las señales de control futuras. Esto se puede ver claramente en las ecuaciones 8, 9 y 10. Por otra parte para calcular las señales de control futuras se utiliza una función de coste que en el caso estándar es cuadrática y penaliza los errores entre la salida predicha y la trayectoria de referencia. Generalmente, se suele incluir el esfuerzo de control en la función de coste cuadrática. El problema puede ser resuelto analíticamente en ausencia de restricciones pero en caso contrario se debe utilizar un algoritmo iterativo de optimización. Para cualquiera de las dos alternativas la solución al problema devuelve un vector que contiene las acciones de control futuras cuya dimensión depende del horizonte de control, sin embargo, solamente el primer elemento de este vector debe ser enviado a la planta. Este proceso se vuelve a repetir para cada instante k_i . La estrategia MPC y la idea de algunos de los términos involucrados se ilustran en la Figura 1.

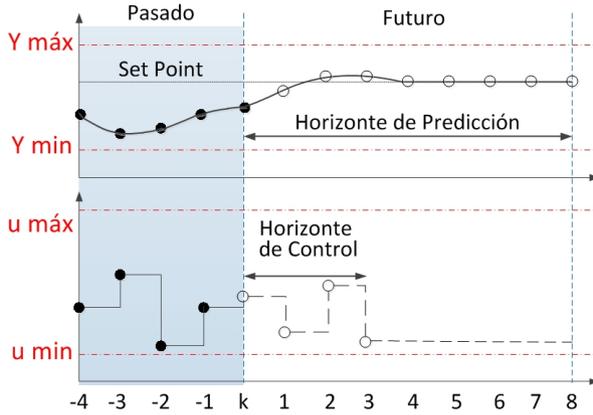


Figura 1. Idea de control predictivo.

2.2. Formulación matemática utilizando espacio de estados

La formulación del MPC en espacio de estados (véase las ecuaciones 1 y 2) debe partir de un modelo con integrador embebido para garantizar un error en estado estacionario nulo. Este modelo se ve representado a través de las ecuaciones 3 y 4. Para mayores detalles se puede consultar [5] y [6]. En estas ecuaciones A_e , B_e y C_e son las matrices del modelo aumentado, o_m es una matriz de ceros con longitud igual al orden del sistema.

$$x(k+1) = Ax(k) + Bu(k) \quad (1)$$

$$y(k) = Cx(k) + Du(k) \quad (2)$$

$$\begin{bmatrix} x(k+1) \\ \Delta x(k+1) \\ y(k) \end{bmatrix} = \begin{bmatrix} A_e \\ A & o_m^T \\ CA & 1 \end{bmatrix} \begin{bmatrix} x(k) \\ \Delta x(k) \\ y(k) \end{bmatrix} + \dots \\ \dots + \begin{bmatrix} B_e \\ B \\ C.B \end{bmatrix} \Delta u(k) \quad (3)$$

$$y(k) = \begin{bmatrix} C_e \\ o_m & 1 \end{bmatrix} \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix} \quad (4)$$

A partir del modelo aumentado se calcula las predicciones de estado y de salida para el instante presente k_i . El conjunto de predicciones de estado se muestran en las ecuaciones 5, 6 y 7.

$$x(k_i + 1 | k_i) = A_e x(k_i) + B_e \Delta u(k_i) \quad (5)$$

$$x(k_i + 2 | k_i) = A_e^2 x(k_i) + A_e B_e \Delta u(k_i) + \dots \\ \dots + B_e \Delta u(k_i + 1) \quad (6)$$

$$x(k_i + N | k_i) = A_e^N x(k_i) + A_e^{N-1} B_e \Delta u(k_i) + \dots \\ \dots + A_e^{N-M} B_e \Delta u(k_i + M - 1) \quad (7)$$

Donde M y N son los horizontes de control y predicción respectivamente. El mismo criterio puede ser utilizado para obtener las predicciones de salida.

$$y(k_i + 1 | k_i) = C_e A_e x(k_i) + C_e B_e \Delta u(k_i) \quad (8)$$

$$y(k_i + 2 | k_i) = C_e A_e^2 x(k_i) + C_e A_e B_e \Delta u(k_i) + \dots \\ \dots + B_e \Delta u(k_i + 1) \quad (9)$$

$$y(k_i + N | k_i) = C_e A_e^N x(k_i) + C_e A_e^{N-1} B_e \Delta u(k_i) + \dots \\ \dots + C_e A_e^{N-2} B_e \Delta u(k_i + 1) + \dots \\ \dots + C_e A_e^{N-M} B_e \Delta u(k_i + M - 1) \quad (10)$$

El conjunto de predicciones de salida puede ser representado por la ecuación 11. Donde las matrices F y Φ (véase las ecuaciones 12 y 13) son constantes que se formulan en base a las matrices del modelo aumentado.

$$Y = Fx(k_i) + \Phi \Delta U \quad (11)$$

$$F = \begin{bmatrix} C_e A_e \\ C_e A_e^2 \\ C_e A_e^3 \\ \vdots \\ C_e A_e^N \end{bmatrix} \quad (12)$$

$$\Phi = \begin{bmatrix} C_e B_e & 0 & \dots & 0 \\ C_e A_e B_e & C_e B_e & \dots & 0 \\ C_e A_e^2 B_e & C_e A_e B_e & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ C_e A_e^{N-1} B_e & C_e A_e^{N-2} B_e & \dots & C_e A_e^{N-M} B_e \end{bmatrix} \quad (13)$$

2.3. Optimización

Con las matrices F y Φ se puede abordar el problema de optimización utilizando una función de costo J (véase la ecuación 14).

$$J(y, u) = (Rs - Y)^T (Rs - Y) + \Delta U^T R \Delta U \quad (14)$$

Esta función minimiza el error entre la salida predicha y el punto de ajuste Rs teniendo en consideración los incrementos de la señal ΔU . R es un parámetro de sintonía que da atención a los incrementos de control ΔU . Cuando no existen restricciones la función de costo puede ser derivada e igualada a cero para despejar ΔU . El resultado de este procedimiento se muestra en la Ecuación 15.

$$\Delta U = H\Phi^T [Rs.r(k_i) - Fx(k_i)] \quad (15)$$

Donde $H = (\Phi^T \Phi + R)^{-1}$ se la conoce como matriz hessiana. La dimensión de ΔU depende del horizonte de control y únicamente el primer término del vector debe ser enviado a la planta. Para el caso en el que se impongan restricciones estas deberán ser expresadas en el formato $M \cdot \Delta U = \gamma$ (véase la ecuación 16).

$$\begin{bmatrix} -I \\ I \\ -\rho_2 \\ \rho_2 \\ -\Phi \\ \Phi \end{bmatrix} \Delta U \leq \begin{bmatrix} -\Delta U^{min} \\ \Delta U^{max} \\ \rho_1 u(k_i - 1) - U^{min} \\ \rho_1 u(k_i - 1) + U^{max} \\ Fx(k_i) - Y^{min} \\ Fx(k_i) - Y^{max} \end{bmatrix} \quad (16)$$

Donde ΔU^{min} , ΔU^{max} , U^{min} , U^{max} , Y^{min} y Y^{max} son las restricciones mínimas y máximas impuestas a la tasa de cambio de la variable de control, amplitud de la variable de control y amplitud de la variable de salida respectivamente. I es una matriz identidad, ρ_1 es un vector unidad y ρ_2 es una matriz diagonal inferior. La dimensión de estos tres últimos parámetros depende del número de puntos sobre los cuales se aplican las restricciones. Para mayor detalle se puede consultar [6].

3. Procedimiento para implementar un controlador predictivo basado en modelo

La implementación del MPC se puede dividir en dos etapas: fase de inicialización y proceso iterativo o fase de ejecución. En la fase de inicialización se calculan las matrices constantes A_e , B_e , C_e , F , Φ , H , Φ^T . Estas matrices son calculadas una sola vez en el controlador sin la necesidad de que haya conexión con la planta. Durante la fase de ejecución el algoritmo predictivo interactúa con la planta, en el caso de que no existan restricciones el proceso iterativo queda determinado

por la ecuación 15, este proceso también se puede representar a través del diagrama de bloques mostrado en la Figura 2.

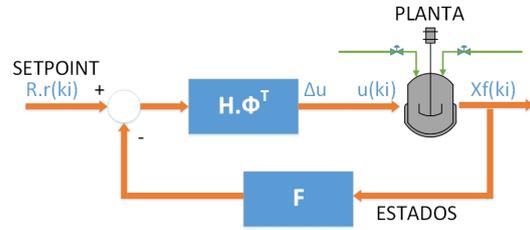


Figura 2. Interacción entre el controlador MPC y la planta en el caso de que no existan restricciones.

En el caso de que existan restricciones la fase de inicialización se extiende con la formulación de la matriz constante M y la matriz inicial γ . Bajo esta modalidad el proceso iterativo puede tomar caminos diferentes en cada período de iteración. Atendiendo a la variable a la que se impuso las restricciones y a las características de las mismas se puede tener los siguientes casos:

Caso 1: Cuando las restricciones son aplicadas únicamente ΔU . En este caso las matrices M y γ son constantes durante todo el proceso iterativo.

Caso 2: Cuando las restricciones son aplicadas a la amplitud de U y/o amplitud de Y independientemente de si se impone o no restricciones a ΔU . En este caso parte de la matriz de restricción γ es variable por lo tanto, para cada iteración se formula una nueva matriz.

Caso 3: Cuando las restricciones son inactivas. En este caso en la iteración k_i se cumple que $M \cdot \Delta U \leq \gamma$, entonces se extrae el primer valor de ΔU y se envía a la planta.

Caso 4: Cuando las restricciones son activas. En este caso en la iteración k_i no se cumple que $M \cdot \Delta U \leq \gamma$, por lo tanto, es necesario calcular un nuevo ΔU utilizando un algoritmo de optimización QP como por ejemplo el propuesto por Hildreth's (véase [6]).

Los casos 1 y 2 hacen referencia a la formulación de las restricciones y, los casos 3 y 4 a la forma en la que se llega al cumplimiento de estas. Por lo tanto, para cada iteración puede haber combinaciones del caso 1 o 2 con el caso 3 o 4 respectivamente. La Figura 3 muestra un diagrama de flujo que sintetiza la implementación de un MPC con los tres tipos de restricciones (véase la ecuación 16). En el caso de que las restricciones sean impuestas únicamente a ΔU el diagrama de la Figura 3 aplica sin la actualización de la matriz γ .

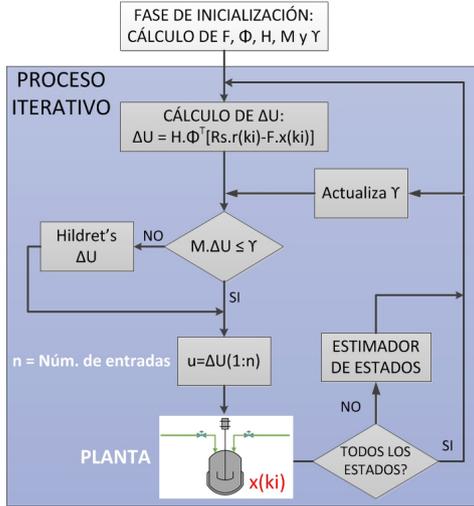


Figura 3. Interacción entre el controlador MPC y la planta en el caso de que se impongan restricciones.

4. Plataforma de simulación

La plataforma de simulación implementada en GUIDE de MATLAB[®] orienta al usuario para que especifique la planta a controlar y el tipo de modelo de predicción a utilizar, la ventana principal permite el ingreso de los parámetros de sintonía, el tiempo de simulación, las consignas, las condiciones iniciales de la planta y las restricciones en caso de optar por esta modalidad, además la ejecución del controlador deja ver la respuesta del sistema y la acción de control para llevar a la salida la consigna establecida.

Al inicializar el controlador la plataforma lee los parámetros de ingreso e invoca a varios ficheros .m que calculan las matrices del modelo embebido, las matrices F , Φ y la matriz hessiana, respectivamente.

En el caso de no haber restricciones el algoritmo iterativo implementado en un fichero .m calcula la acción de control y envía a la planta, la cual actualiza sus estados para que sean leídos por el controlador siempre y cuando la opción de estados disponibles esté habilitada, caso contrario el algoritmo iterativo hará uso de un observador de estados para estimarlos y ponerlos a disposición del controlador. En el caso de que se haya habilitado la opción de restricciones estas deberán ser especificadas para cada una de las variables a través de sus valores mínimos y máximos. La Figura 4 muestra una vista panorámica de la ventana principal de la plataforma de simulación.

5. Presentación y análisis de resultados

El algoritmo MPC se ejecuta en un entorno de simulación lo cual quiere decir que la planta se representa por un conjunto de ecuaciones diferenciales las cuales

actualizan los estados $x(t)$ en función de la señal de control u . Para el caso SISO se ha utilizado un servomecanismo de posición descrito por las ecuaciones 17 y 18 y cuya estructura del sistema se muestra en la Figura 5. Este modelo de planta se ha utilizado en [27] para la implementación de un controlador PID de orden fraccional. En este sistema: $K_\theta = 1280.2$, $K_T = 10$, $J_M = 0.5$, $J_L = 25$, $\rho = 20$, $B_M = 0.1$, $B_L = 0.25$, $R = 20$ | $U < 220V$ y $K_r = -(\beta_M + K_T^2/R)/J_M$. La Figura 6 muestra el resultado de la ejecución del algoritmo para 2 conjuntos diferentes de los parámetros de sintonía N y M . Para los dos casos se ha asumido un factor de peso $R = 9e^{-9}$ y una consigna del sistema $\theta_L = 1$ rad.

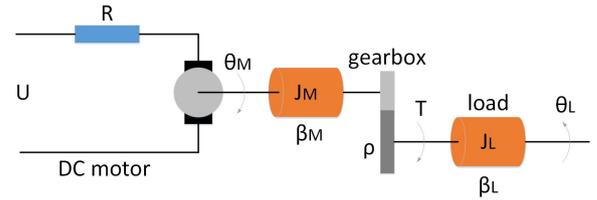


Figura 5. Servomecanismo de posición para prueba SISO del algoritmo MPC [27].

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_\theta}{J_L} & -\frac{\beta_\theta}{J_L} & \frac{k_\theta}{\rho J_L} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_\theta}{\rho J_M} & 0 & \frac{k_\theta}{\rho^2 J_M} & k_R \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_T}{R J_L M} \end{bmatrix} u \quad (17)$$

Mediante las gráficas de la Figura 6 se puede notar la efectividad del algoritmo para llevar al sistema a la consigna. Para el caso en el que $N = 9$ y $M = 4$ se logra la sintonía del controlador pero el transitorio es muy prolongado y la amplitud de la señal de control sobrepasa la tensión nominal del motor. Al mover los parámetros de sintonía a $N = 3$ y $M = 1$ el transitorio mejora notablemente y la amplitud de la señal de control disminuye favorablemente. La Figura 7 muestra los efectos del factor de peso en la sintonía del controlador. Se han considerado dos casos con valores diferentes de R mientras que N y M se ha mantenido en 12 y 9 respectivamente para ambas ejecuciones.

Una de las ventajas más atractivas de la estrategia MPC es la incorporación de restricciones. De esta manera se puede medir los incrementos y decrementos de la variable de control, así como las amplitudes mínimas y máximas de las variables de salida y control respectivamente. En la Figura 8 se muestra el resultado de haber incorporado de manera satisfactoria el conjunto de restricciones: $(0 \leq u \leq 60)$ y $(-15 \leq \Delta u \leq 15)$.

Para el caso MIMO se ha utilizado una planta que consiste de un conjunto turbina-caldera tipo tambor (véase Figura 9). Esta planta es representada dentro del ordenador por un sistema no lineal de ecuaciones



Figura 4. Plataforma de simulación para la sintonía de controladores predictivos.

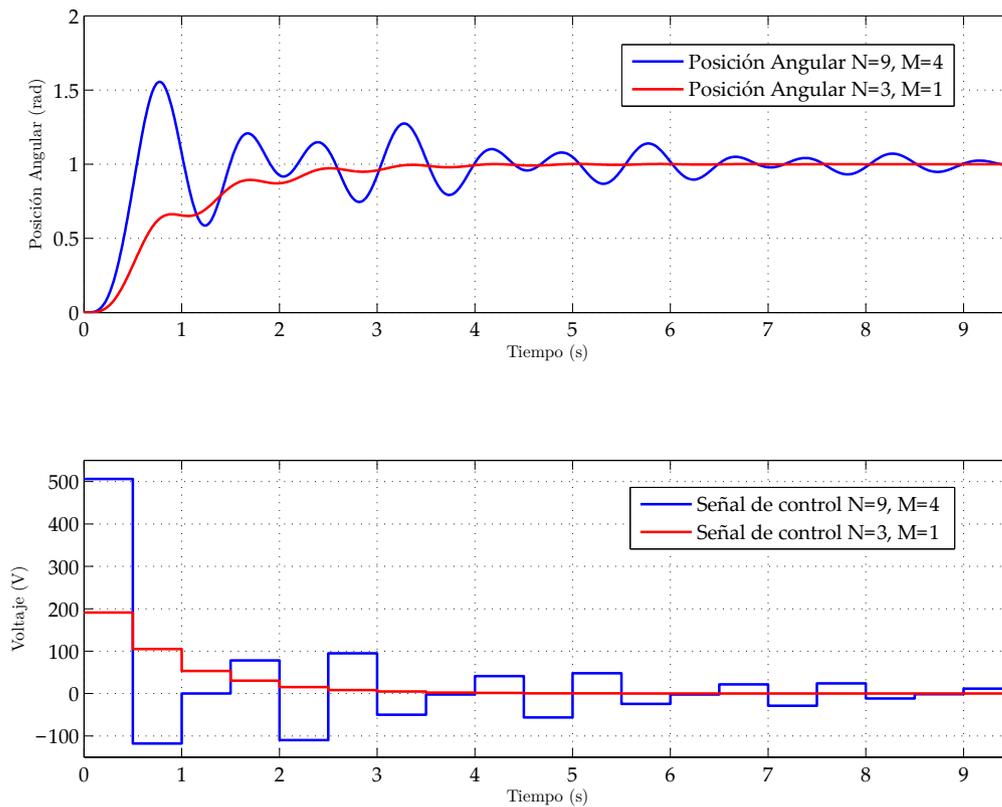


Figura 6. Ejecución del algoritmo MPC sobre planta SISO para dos conjuntos diferentes de horizonte de control y predicción.

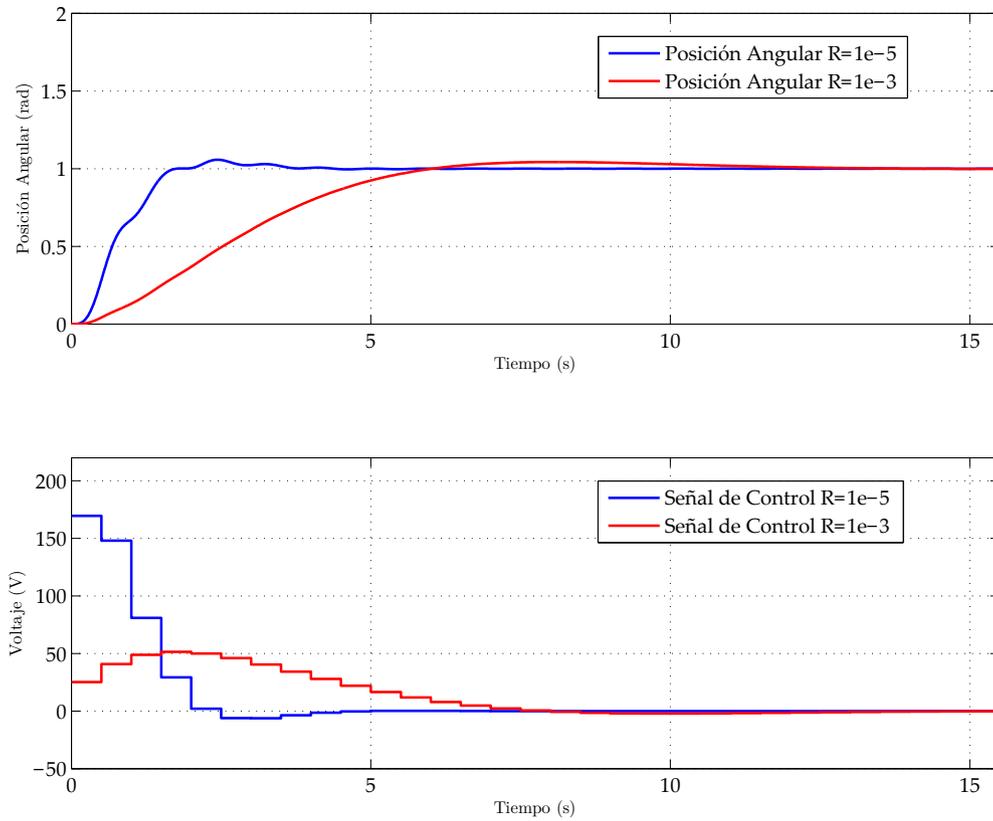


Figura 7. Ejecución del algoritmo MPC sobre planta SISO para dos conjuntos diferentes del factor de peso R .

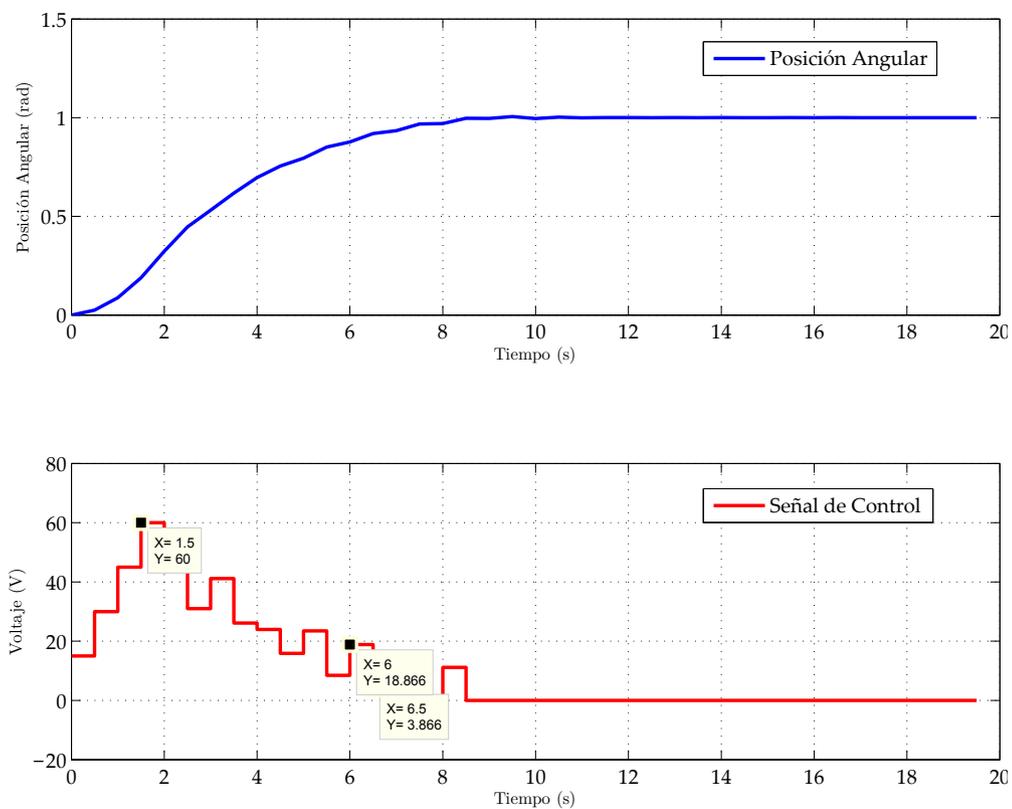


Figura 8. Ejecución del algoritmo MPC sobre planta SISO para dos conjuntos diferentes del factor de peso R .

diferenciales y el modelo de predicción ha sido linealizado y transformado a espacio de estados discreto. Este modelo de planta propuesto por [11] se ha utilizado para la implementación de un controlador predictivo DMC. Las expresiones 19 a 24 muestran las ecuaciones dinámicas que gobiernan el sistema no lineal, el modelo de predicción se ha linealizado con $T_s = 5$ s en los puntos de trabajo $y_1 = 115 \text{ kg/cm}^2$, $y_2 = 85 \text{ MW}$ y $y_3 = 0$ para ser utilizado por el controlador como modelo de predicción.

$$\dot{x}_1 = -0,0018u_2x_1^{9/8} + 0,9u_1 - 0,15u_3 \quad (18)$$

$$\dot{x}_2 = \frac{[(0,73u_2 - 0,16)x_1^{9/8} - x_2]}{10} \quad (19)$$

$$\dot{x}_3 = \frac{[114u_3 - (1,1u_2 - 0,19)x_1]}{85} \quad (20)$$

$$y_1 = x_1 \quad y_2 = x_2 \quad (21)$$

$$y_3 = 0,05 \left(0,13x_3 + 100a_{cs} + \frac{q_e}{9} - 67,97 \right) \quad (22)$$

$$a_{cs} = \frac{(1 - 0,001538x_3)(0,81x_1 - 25,6)}{x_3(1,0394 - 0,0012304x_1)} \quad (23)$$

$$q_e = (0,854u_2 - 0,147x_1) + 45,59u_1 - \dots \\ \dots - 2,51u_3 + 2,096 \quad (24)$$

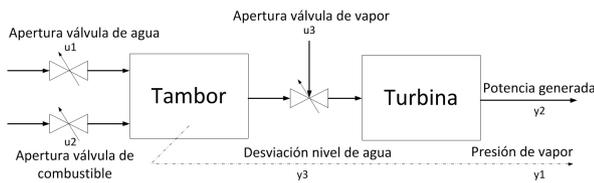


Figura 9. Turbina-caldera tipo tambor para prueba MIMO del algoritmo MPC [11].

En la Figura 10 se puede ver las salidas del sistema para la ejecución del algoritmo MPC sobre el sistema multivariable y por otra parte, en la Figura 11 se ve las acciones de control para lograr las consignas o puntos de referencia (Presión de vapor = 112 kg/cm^2 , Potencia generada = 93 MW , desviación nivel de agua = 0). Claramente se puede comprobar la efectividad del control predictivo multivariable tanto en alcanzar las consignas como en las restricciones impuestas. Los parámetros de sintonía para la ejecución del algoritmo son $N = 8$, $M = 3$ y $R = 70$. Este se ha probado bajo las siguientes restricciones:

1. $-0,07 \leq \Delta u_1 \leq 0,07$
2. $-0,05 \leq \Delta u_1 \leq 0,05$
3. $-0,05 \leq \Delta u_1 \leq 0,05$
4. $0 \leq u_1 \leq 1$
5. $0 \leq u_2 \leq 1$
6. $0 \leq u_3 \leq 1$
7. $0 \leq y_1 \leq 160$
8. $0 \leq y_2 \leq 120$
9. $0 \leq y_3 \leq 4$

6. Conclusiones

De la revisión bibliográfica realizada se pudo observar que las plataformas comerciales de control predictivo basado en modelo tratan de dar al usuario las comodidades necesarias para hacer uso de la técnica sin la necesidad de acudir a otras plataformas. Por lo general, están vinculadas con otras herramientas como de identificación de sistemas, estimación de estados, linealización, reducción de orden, que en mayor o menor grado pueden ser requeridas por el algoritmo de control.

Con el trabajo realizado se pudo comprobar la fácil adaptación del algoritmo del caso SISO al caso multivariable, únicamente se debe prestar atención a las dimensiones de las matrices y al orden en el que se formulan las restricciones.

La implementación del algoritmo de control predictivo dejó ver que su estructura se basa en operaciones matriciales las cuales demandan un coste computacional elevado. Por ejemplo, para la implementación en el caso multivariable con restricciones, el tiempo de cálculo de la acción de control alcanzó hasta los 0.4 segundos, lo cual debe ser analizado antes de decidir utilizar este controlador en procesos con dinámica rápida.

De las simulaciones ejecutadas se pudo comprobar la eficacia de esta estrategia, pudiendo controlar las características del transitorio a través de los parámetros de sintonía.

El algoritmo de control al ser formulado en espacio de estados requiere de manera indispensable los estados de la planta, esto obliga a la estrategia a contar con observadores para disponer de los mismos.

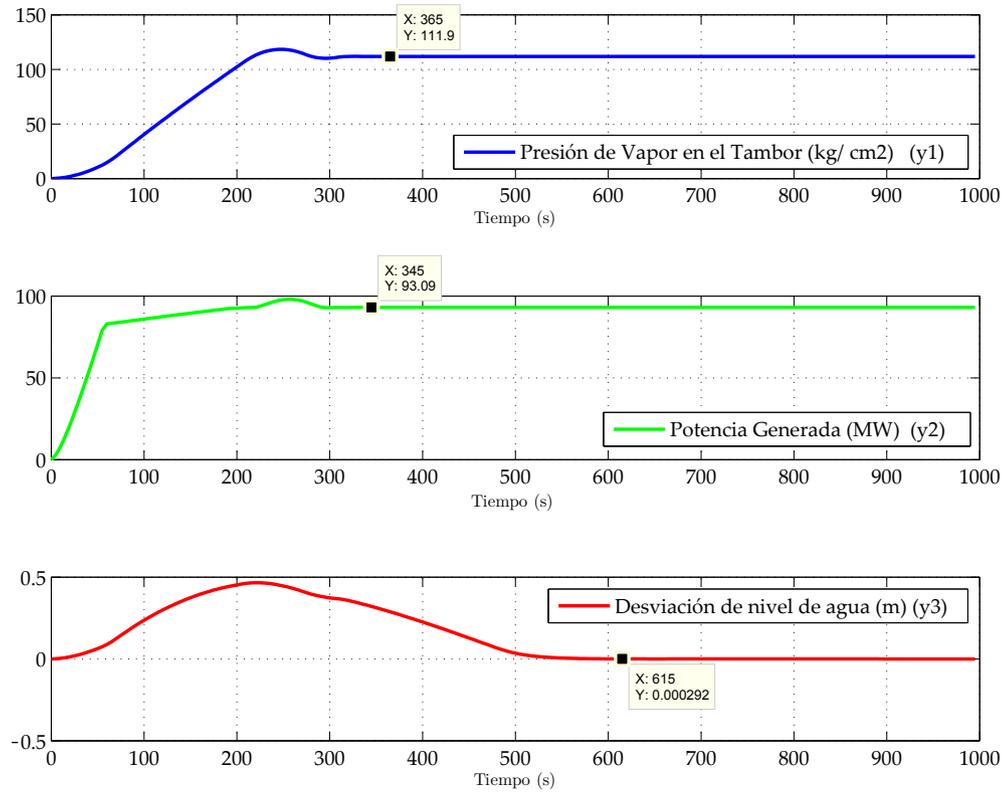


Figura 10. Implementación del algoritmo MPC sobre el sistema multivariable turbina-caldera (salidas del sistema).

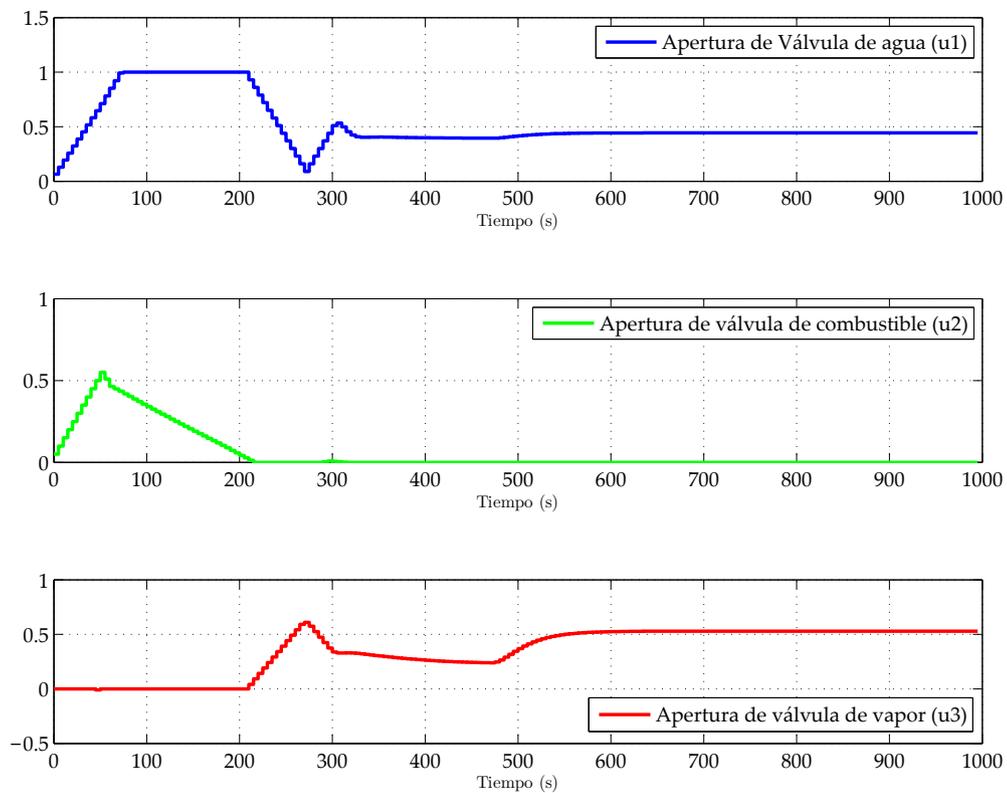


Figura 11. Implementación del algoritmo MPC sobre el sistema multivariable turbina-caldera (variables de control).

Referencias

- [1] J. Richalet, A. Rault, J. Testud, and J. Papon, "Model predictive heuristic control: Applications to industrial processes," *Automatica*, vol. 14, no. 5, pp. 413–428, 1978.
- [2] C. Roos, T. Terlaky, and J.-P. Vial, *Theory and algorithms for linear optimization: an interior point approach*. Wiley Chichester, 1997.
- [3] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control engineering practice*, vol. 11, no. 7, pp. 733–764, 2003.
- [4] E. F. Camacho, C. Bordons, E. F. Camacho, and C. Bordons, *Model predictive control*. Springer London, 2004, vol. 2.
- [5] J. M. Maciejowski, *Predictive Control: With Constraint*. Pearson education, 2002.
- [6] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB®*. Springer Verlag, 2009.
- [7] A. Grancharova and T. A. Johansen, *Explicit nonlinear model predictive control: Theory and applications*. Springer Verlag, 2012, vol. 429.
- [8] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*. Springer, 2011.
- [9] B. Kouvaritakis and M. Cannon, *Non-Linear Predictive Control: Theory and Practice*, ser. IEE Monographs. Institution of Engineering and Technology, 2001.
- [10] C. R. Cutler and B. Ramaker, "Dynamic matrix control—a computer control algorithm," in *Proceedings of the joint automatic control conference*, vol. 1. University of Michigan Ann Arbor, MI, 1980, pp. Wp5–B.
- [11] U.-C. Moon and K. Y. Lee, "Step-response model development for dynamic matrix control of a drum-type boiler–turbine system," *IEEE Transactions on Energy Conversion*, vol. 24, no. 2, pp. 423–430, 2009.
- [12] D. W. Clarke, C. Mohtadi, and P. Tuffs, "Generalized predictive control—part i. the basic algorithm," *Automatica*, vol. 23, no. 2, pp. 137–148, 1987.
- [13] E. Camacho, "Constrained generalized predictive control," *IEEE Transactions on Energy Conversion*, vol. 38, no. 2, pp. 327–332, 1993.
- [14] D. W. Clarke, "Application of generalized predictive control to industrial processes," *IEEE Control Systems Magazine*, vol. 8, no. 2, pp. 49–55, 1988.
- [15] ASPENTECH, "DMCplus: A powerful, proven multivariable controller for maintaining processes at their optimal operating point," 2012. [Online]. Available: <http://www.aspentech.com/products/aspndmcplus.aspx>
- [16] —, "DMCplus TM," 2012. [Online]. Available: <http://www.aspentech.com>
- [17] S. Goodhart, "Dmcpplus model predictive control applications," in *Model Predictive Control: Techniques and Applications - Day 2 (Ref. No. 1999/096)*, IEE Two-Day Workshop on, 1999, pp. 6/1–6/6.
- [18] ASPENTECH and Valero Energy Corporation, "Valero Implements APC Solution to Manage Stock Switches, Reducing Product Downgrades by 400Mper Year," 2010.
- [19] Honeywell International Inc., "Control Performance Solutions: Profit Controller," 2012. [Online]. Available: www.honeywellprocess.com/en-us/explore/products/
- [20] —, "Profit Controller Multivariable Control and Local Optimization Technology," 2012.
- [21] S. Yoo, T. Kim, B. Choi, S. Ahn, J. Yoon, S. Seo, and H. Kim, "Optimized model predictive control of commercial btx plant using profit® controller," in *SICE-ICASE, 2006. International Joint Conference*. IEEE, 2006, pp. 1341–1344.
- [22] N. Moldoványi and J. Abonyi, "Control of a continuous vacuum crystalliser in an industrial environment: A feasibility study—comparison pid to model predictive control solution," *Hungarian Journal of Industrial Chemistry*, 2008.
- [23] ABB, "cpmPlus Expert Optimizer: Overview," 2012. [Online]. Available: www.abb.com/product/db0003db004001/aeb109afc47df071852576170049499e.aspx
- [24] G. Welch and G. Bishop, "An introduction to the kalman filter," 1995. [Online]. Available: http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf
- [25] D. Castagnoli, M. Kiener, and E. Gallestey, "Rentabilidad del cemento: hacia la optimización total de las plantas industriales de minerales y cemento," *Revista ABB*, no. 4, pp. 59–62, 2006.
- [26] M. Lundh and S. Gaulocher, "Model predictive control for flotations plants," 2008.
- [27] D. Xue, C. Zhao, and Y. Chen, "Fractional order pid control of a dc-motor with elastic shaft: a case study," in *American Control Conference*. IEEE, 2006, p. 6.