

Control Predictivo Basado en Datos

J.M. Manzano*, D. Limon*, T. Alamo*, J.P. Callies**

*Universidad de Sevilla. josmancre@alum.us.es, dlim@us.es, talamo@us.es

**University of Cambridge. jpc73@cam.ac.uk

Resumen

Este artículo presenta el diseño estabilizante de un controlador predictivo a partir únicamente de datos de entrada-salida de un sistema a controlar. El modelo que incluye este controlador es una función no lineal estimada usando una técnica de aprendizaje automático no paramétrica conocida como Kinky Inference. Como primeras pruebas en esta nueva línea de investigación, se diseñan y se prueban controladores en un reactor continuamente agitado, considerando las cuestiones necesarias para una correcta implementación práctica.

Palabras clave: Control predictivo; Control basado en datos; Aprendizaje automático; Sistemas no lineales con restricciones.

1. Introducción

El control predictivo basado en modelo (MPC, por sus siglas en inglés) es una técnica de control no lineal capaz de controlar un sistema garantizando la satisfacción de restricciones duras [6], tanto en las entradas manipulables como en las variables controladas. Para ello el controlador cuenta con un modelo dinámico del sistema, que se usa para predecir el comportamiento esperado del mismo ante una secuencia de acciones futuras prevista. Las predicciones obtenidas permiten estimar el coste asociado a la trayectoria prevista así como el grado de satisfacción de las restricciones impuestas. De esta forma, es posible calcular la secuencia de acciones futuras que logren un menor coste de operación satisfaciendo las restricciones, mediante la resolución de un problema de optimización estática. Con el fin de dotar de realimentación al sistema y obtener mejores predicciones, se aplica tan sólo la acción calculada correspondiente al instante actual, descartando el resto. De esta forma, en cada tiempo de muestreo se actualiza la información necesaria para realizar las predicciones, actuando ante posibles errores de modelado. Esta es la llamada estrategia de horizonte deslizante.

El control predictivo ha logrado un notable éxito en la práctica gracias a su carácter óptimo y a su

capacidad de manejar las restricciones. Este éxito se ha dado principalmente en las aplicaciones en las que se dispone de modelos de predicción fiables, típicamente de primeros principios.

Al ser un controlador basado en modelo, los errores de modelado inducen errores en las predicciones y por lo tanto en la toma de decisiones. Por ello, la disponibilidad de buenos modelos de predicción es clave en el diseño de controladores predictivos. Cuando los modelos de primeros principios no resultan adecuados para realizar predicciones, se pueden inferir modelos de predicción a partir de datos históricos disponibles. Típicamente esto se lleva a cabo mediante técnicas de identificación de sistemas dinámicos, especialmente adecuadas para modelos lineales en torno a un punto de operación [3].

El objetivo de este artículo es estudiar el desarrollo de controladores predictivos que sean capaces de controlar la planta a partir de datos históricos disponibles del sistema a controlar. Mediante técnicas de aprendizaje automático, se van a inferir procedimientos para calcular las predicciones a partir de las cuales se determinará un controlador predictivo, de forma que el controlador resultante se basará directamente en los datos de entrada-salida de la planta a controlar. En [1] se propone usar técnicas de aprendizaje automático para estimar las incertidumbres globales del sistema con el fin de mejorar las predicciones. En [7] se propone un controlador predictivo no lineal basado en datos usando técnicas de interpolación de Lipschitz asumiendo que se conoce de antemano la constante de Lipschitz del sistema. En este trabajo nos vamos a centrar en técnicas de inferencia basadas en la interpolación de Lipschitz (extendidas a funciones con continuidad de Hölder) que no requieren un conocimiento previo de las constantes de Lipschitz [4]. Se van a presentar los trabajos preliminares que demuestran que estas técnicas son adecuadas para el desarrollo de controladores predictivos basados únicamente en datos.

Notación Sean dos espacios, \mathcal{X} e \mathcal{Y} , se definen las pseudo-métricas $\mathfrak{d}_{\mathcal{X}}(\cdot, \cdot) : \mathcal{X}^2 \rightarrow \mathbb{R}_{\geq 0}$ y $\mathfrak{d}_{\mathcal{Y}}(\cdot, \cdot) : \mathcal{Y}^2 \rightarrow \mathbb{R}_{\geq 0}$ como simétricas, positivas, y que cumplen la desigualdad triangular.

2. Kinky Inference: el predictor

Como se ha comentado, el objetivo de esta investigación es el desarrollo de controladores predictivos cuyo modelo está únicamente basado en datos. Este modelo será no explícito, generado por un método de aprendizaje automático propuesto por [5] y conocido como Lazily Adapted Constant Kinky Inference (LACKI). Este método se va a usar para aprender un cierto mapa $f : \mathcal{X} \rightarrow \mathcal{Y}$ definido entre espacios y pseudo-métricas $(\mathcal{X}, \mathfrak{d}_{\mathcal{X}})$ e $(\mathcal{Y}, \mathfrak{d}_{\mathcal{Y}})$, que se asume es continuo en el sentido de Hölder con una constante de Hölder L^* y exponente α , por lo que para todo $\forall x_1, x_2 \in \mathcal{X}$ cumple la siguiente condición:

$$\mathfrak{d}_{\mathcal{Y}}(f(x_1), f(x_2)) \leq L^* \mathfrak{d}_{\mathcal{X}}(x_1, x_2)^\alpha \quad (1)$$

De dicha función objetivo $f : \mathcal{X} \rightarrow \mathcal{Y}$ se conocen ciertas parejas de entradas y salidas (posiblemente ruidosas) que conforman el dataset $\mathcal{D}_n := \{(s_i, \tilde{f}_i) | i = 1, \dots, N_n\}$, siendo \tilde{f} la medida con ruido de f . Aunque se desconoce la función real f , se asume que se conoce la cota del error observacional, dada por $\bar{\epsilon}$, de forma que $\forall x \in \mathcal{X} : \mathfrak{d}_{\mathcal{Y}}(\tilde{f}(x), f(x)) \leq \bar{\epsilon} \in \mathbb{R}_{\geq 0}$.

Dado que tan sólo se dispone de los datos de entrada-salida medidos, resulta necesario estimar una cierta constante que se use como parámetro en el aprendizaje del sistema. En [5] se describen varios métodos para realizarlo. El que consideramos en este artículo incluye un parámetro de regularización, γ , que suaviza la predicción conociendo la cota de ruido observacional $\bar{\epsilon}$. El término *Lazily Adapted Constant* de LACKI hace referencia a la forma de obtener la constante, pues será la mínima que sea válida conforme a los datos \mathcal{D} . Se calcula de la siguiente forma:

$$L(\mathcal{D}_n, \gamma) := \max_{(s, s') \in \mathcal{U}_n} \frac{\mathfrak{d}_{\mathcal{Y}}(\tilde{f}(s), \tilde{f}(s')) - \gamma}{\mathfrak{d}_{\mathcal{X}}^\alpha(s, s')} \quad (2)$$

con $\gamma \geq 2\bar{\epsilon}$. De ahora en adelante se expresará $L(\mathcal{D}_n, \gamma)$ como $L_{\mathcal{D}}$, para simplificar. Esta estimación es siempre menor o igual que la mínima constante de Hölder de la función, $L_{\mathcal{D}} \leq L^*$, y tiende a ésta a medida que aumenta la densidad del conjunto de datos.

A partir únicamente del conjunto de puntos observados, y de la constante $L_{\mathcal{D}}$ (estimada a partir de ese conjunto), se puede realizar la interpolación de Lipschitz para cualquier otro punto $x \notin \mathcal{D}_n$. El estimador \hat{f}_n se construye de la siguiente forma para la componente j -ésima de la función:

$$\hat{f}_j(x; \mathcal{D}_n) = \frac{1}{2} \min_{i=1, \dots, N_n} \tilde{f}_{i,j} + L(\mathcal{D}_n) \mathfrak{d}_{\mathcal{X}}^\alpha(x, s_i)$$

$$+ \frac{1}{2} \max_{i=1, \dots, N_n} \tilde{f}_{i,j} - L(\mathcal{D}_n) \mathfrak{d}_{\mathcal{X}}^\alpha(x, s_i) \quad (3)$$

Es importante destacar que el error de estimación del predictor de KI, definido como $\mu = \mathfrak{d}_{\mathcal{Y}}(f, \hat{f})$, tiende a cero a medida que aumenta la densidad de los puntos de entrada, siendo capaz de estimar el mapa verdadero $f(\cdot)$ con un error arbitrariamente pequeño a partir de un conjunto de datos suficientemente denso, siempre que el mapa real sea continuo en el sentido de Hölder.

3. Control predictivo basado en datos

Se desea desarrollar un controlador predictivo que permita controlar un sistema del cual tan sólo se dispone de las medidas de las variables a controlar $y(k)$, definidas en el espacio $\mathcal{Y} = \mathbb{R}^n$, y de las variables manipulables $u(k)$, definidas en el espacio $\mathcal{U} \subseteq \mathbb{R}^m$. Realizando una hipótesis no muy restrictiva, se asume que el sistema se puede describir mediante un modelo ARX, con las características descritas en [9], de forma que

$$y(k+1) = f(x(k), u(k)) + e(k) \quad (4)$$

donde $x(k) = [y(k), \dots, y(k - n_a), u(k - 1), \dots, u(k - n_b)] \in \mathcal{X} := \mathcal{Y}^{(n_a+1)} \times \mathcal{U}^{n_b} \subset \mathbb{R}^{n_x}$ con $n_x = (n_a + 1)n + n_b m$, para cierto horizonte de memoria $n_a, n_b \in \mathbb{N}$; y $e(k)$ describe posibles ruidos en las medidas. Nótese que $x(k)$ está definido para $n_b \geq 1$; sin embargo, se contempla la posibilidad de tener $n_b = 0$ para indicar que el estado no depende de las acciones de control pasadas, $x_k \neq x_k(u)$.

En esta sección se desarrollan controladores predictivos para controlar este sistema, del cual no se conoce el modelo $f(\cdot, \cdot)$, sino un conjunto de datos históricos de entrada-salida $\mathcal{D}_n = \{(u(j), y(j)), j = 1, \dots, N_n\}$, obtenido a partir de ensayos previos. El modelo de predicción será el estimador derivado usando la técnica LACKI para aprender el modelo regresivo del sistema. Este predictor permitirá estimar la evolución del sistema para el instante $k+1$ según cierta señal de control u_k dada, y a partir de un conjunto de variables pasadas (el regresor x_k) y el dataset \mathcal{D}_n , que se considera fijo y con el que se estima la $L_{\mathcal{D}}$:

$$\hat{y}(k+1) = \hat{f}(x(k), u(k); \mathcal{D}_n, L_{\mathcal{D}}) \quad (5)$$

Así, el modelo de predicción en variables de estado, en el instante k , para una secuencia de acciones futuras (\hat{u}_0, \dots) , vendrá dado por:

$$\hat{x}(j+1|k) = \hat{F}(\hat{x}(j|k), \hat{u}(j)) \quad (6)$$

$$\hat{y}(k|j) = M\hat{x}(j|k) \quad (7)$$

siendo

$$\hat{x}(j|k) = (\hat{y}(j|k), \dots, \hat{y}(1|k), y(k), \dots, y(k+j-n_a), \hat{u}(j-1), \dots, \hat{u}(j-n_b))$$

y la función del modelo $\hat{F}(\hat{x}(j|k), u(k+j))$ es

$$\begin{aligned} & (\hat{f}(\hat{x}(j|k), u(k+j); \mathcal{D}_n, L_{\mathcal{D}}), \\ & \hat{y}(j|k), \dots, y(k), \dots, \\ & y(k+j-n_a+1), \dots, \\ & u(k+j), \dots, u(k+j-n_b+1)) \end{aligned}$$

y $M = [I_n, 0, \dots, 0]$.

Para derivar el controlador predictivo se debe definir una función de coste de etapa, que en este caso se ha elegido como una función cuadrática que penalice el error de seguimiento de las entradas y salidas respecto al punto de referencia en el que se quiere estabilizar el sistema, dado por (y_r, u_r) . Se obtiene así $x_r = (y_r, \dots, y_r, u_r, \dots, u_r) \in \mathbb{R}^{n_a+1+n_b}$. Dicho coste de etapa se define a continuación:

$$\ell(y, u) = (y - y_r)^T Q (y - y_r) + (u - u_r)^T R (u - u_r) \quad (8)$$

Por todo ello el problema de optimización $P_{N_p, N_c}(x(k), \mathcal{D}_n)$ a resolver es el siguiente [13]:

$$\begin{aligned} \underset{\hat{\mathbf{u}}}{\text{mín}} \quad & V_{N_p, N_c}(x(k), \hat{\mathbf{u}}) = \sum_{i=0}^{N_c-1} \ell(\hat{y}(i|k), \hat{u}(i)) \\ & + \sum_{i=N_c}^{N_p-1} \ell(\hat{y}(i|k), \kappa_f(\hat{x}(i|k))) + \lambda V_f(\hat{x}(N_p|k)) \end{aligned}$$

$$\begin{aligned} \text{s.a.} \quad & \hat{x}(0|k) = x(k) \\ & \hat{x}(j+1|k) = \hat{F}(\hat{x}(j|k), \hat{u}(j)), \quad j \in \mathbb{I}_{0, N-1} \\ & \hat{y}(j|k) = M\hat{x}(j|k) \\ & \hat{u}(j) \in \mathcal{U} \end{aligned} \quad (9)$$

siendo λ una constante mayor que 1 que es un parámetro del controlador. N_p es el horizonte de predicción, N_c el de control; y κ_f la ley de control terminal. La ley de control será $u(k) = \kappa_{MPC}(x(k); \mathcal{D}_n) = \hat{u}^*(0)$.

Para garantizar la estabilidad nominal por diseño, se ha añadido una función de coste terminal cuadrática $V_f(x) = (x - x_r)^T P (x - x_r)$ y una ley de control terminal lineal $\kappa_f(x) = K(x - x_r) + u_r$, siendo K la solución del LQR. Estos términos se pueden diseñar de la forma estándar [10] a partir del modelo linealizado obtenido numéricamente de la función de predicción. Como se demuestra en [11], esta ley control garantiza la estabilidad para cualquier $\lambda \geq 1$ con un dominio de atracción que crece al aumentar λ . Además, gracias a la continuidad de Hölder de los ingredientes del problema

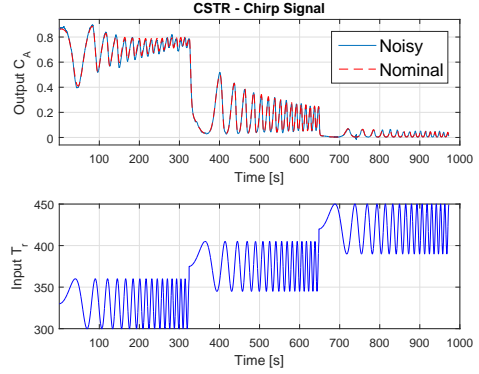


Figura 1: Ensayo para la obtención de datos

de optimización, se puede demostrar que el sistema controlado es estable entrada-estado respecto a los errores de estimación del predictor [12].

4. Caso de estudio

En las secciones anteriores se ha expuesto la teoría necesaria para el desarrollo de un controlador predictivo, estabilizante por diseño, basado íntegramente en datos. El objetivo de este artículo es aplicar dichas metodologías a un caso de estudio que motive la investigación llevada a cabo. Se pretende validar dichas ideas con casos realistas, mediante ensayos realmente implementables.

Como primer trabajo dentro del proyecto de investigación, se está desarrollando un controlador para un **reactor continuamente agitado** (CSTR, por sus siglas en inglés), presentando en [15]. Esta sección analiza el diseño de los experimentos, desde la obtención de los datos y la construcción del predictor hasta el control del sistema bajo determinadas condiciones.

4.1. Obtención de los datos

Se parte de un modelo de caja negra del reactor, que recibe como entrada la temperatura del refrigerante T_r [K] y proporciona como salida la concentración del reactivo a controlar, CA . Las restricciones en la entrada son $300 \leq T_r \leq 450$, y no se consideran restricciones en los estados o la salida. Se simula además con un ruido en las señales de hasta el 3,5%. Es nuestro objetivo realizar un ensayo en el reactor que proporcione un conjunto de datos con información suficiente para poder diseñar posteriormente el controlador.

La primera parte del problema consiste entonces en aplicar técnicas basadas en la **identificación de sistemas**. Se estudia la característica estática del sistema y sus dinámicas principales, para diseñar posteriormente un ensayo de acuerdo con las

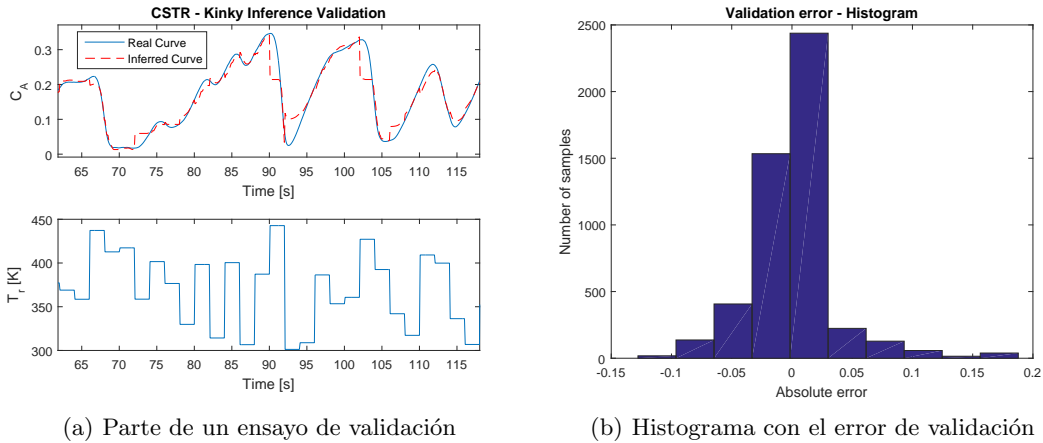


Figura 2: Muestra del funcionamiento del predictor. En (a) se muestra la respuesta del sistema y la del predictor frente a una misma señal de entrada aleatoria. En (b) se realiza un histograma del error de predicción: $\mu = \mathfrak{d}_Y(\tilde{f}, \hat{f})$

metodologías seguidas en [14]. Este ensayo estará compuesto por la superposición de varias señales chirp como entrada, con frecuencias variando entre 0.1 mHz y 0.1 Hz y amplitud suficiente con cierto solape entre ellas para cubrir todo el espacio de interés. El resultado es el representado en la Figura 1.

Se obtienen además varios ensayos generados por señales de entrada aleatorias, que serán usados como ensayos de validación. Los siguientes pasos consisten en obtener un dataset \mathcal{D} válido para la interpolación, y una estimación de la constante de Lipschitz del sistema. Para ello es necesario cierto procesamiento de los datos obtenidos con el ensayo mostrado en la Figura 1. Esto incluye el escalado de las señales y el esquilmo de los datos, pues el ensayo de la señal chirp contenía cerca de 10000 puntos. Se aplica un filtro para obtener dos datasets menores: D_1 con 297 puntos, y D_2 con 895 puntos.

En primer lugar debe construirse el modelo dinámico del sistema. Haciendo uso del modelo autorregresivo presentado en (4), podemos expresar la concentración en cierto instante como una función de los valores pasados de temperaturas (u), y concentraciones (y):

$$y_{k+1} = F(x_k, u_k) \quad (10)$$

siendo F la función que se desea aprender (obtendremos el estimador \hat{F}), u_k la señal de control en ese mismo instante y x_k el vector de regresión en el instante k , dado por:

$$x_k := [y_k, y_{k-1}, \dots, y_{k-n_a}, u_{k-1}, \dots, u_{k-n_b}] \quad (11)$$

El siguiente paso por tanto es la construcción de

este regresor con los valores indicados en (11) para todos los puntos del dataset \mathcal{D} obtenido en el ensayo mostrado en la Figura 1. Cada punto está por tanto compuesto por $\{\tilde{f}_k, s_k\}$, donde $\tilde{f}_k = y_{k+1}$ y $s_k = (x_k, u_k)$.

Puede así estimarse la constante de Lipschitz del sistema aplicando la ecuación (2), si fueran dados los parámetros n_a y n_b ; y obtener por consiguiente el predictor, definido en la ecuación (3). Puede comprobarse su bondad mediante ensayos de validación, usando los datasets aleatorios obtenidos previamente (ver Figura 2).

Estos ensayos de validación sirven, entre otras funciones, para determinar los valores óptimos de los parámetros n_a y n_b . Para $n_a = 2$ y $n_b = 0$ se minimiza el error de validación (diferencia entre la concentración real del sistema y la estimada por el predictor, ante la misma señal de entrada). Con estos valores y con $\gamma = 2\bar{\epsilon}$ (recuérdese que $\bar{\epsilon}$ se asume conocida) se obtiene una $L_{\mathcal{D}} = 1,563$. Nótese que según se ha definido el modelo autorregresivo en (4), es posible el valor $n_b = 0$, pues la señal de control actual, u_k , está definida fuera del regresor x_k .

4.2. Control del reactor

Se aborda entonces el problema de control presentado en la Sección 3. Se ha implementado un controlador predictivo en MatLab usando la función de optimización *fmincon*. Las ecuaciones que gobiernan este MPC son las expresadas en (9).

Para el ensayo se fija el estado inicial en el punto de equilibrio dado por $y = 0,7509$. La referencia se fija en otro punto de equilibrio dado por $y = 0,25$, aunque se construye un SSTO por si esta referencia no fuera alcanzable. Se considerará un

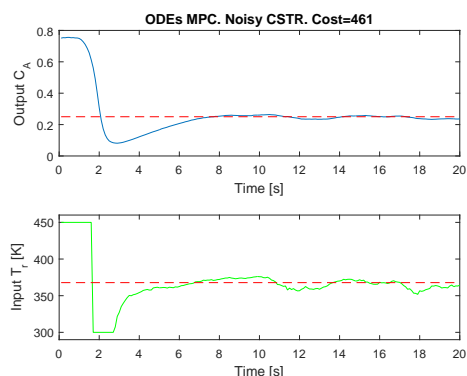


Figura 3: MPC ideal en el que se conociese el modelo del sistema, para ilustrar el control objetivo

coste de etapa cuadrático, con $Q = 100$ y $R = 5$; y se definen las restricciones en la entrada y el coste terminal como se indica en la sección 3.

Para dar una idea al lector del objetivo del control; se realiza un primer experimento en el que el modelo usado por el MPC es la propia planta (definida por su conjunto de ecuaciones diferenciales u ODEs) a la que luego se aplica la señal de control. Nótese que este modelo de la planta, para el resto del artículo, es desconocido; es decir, sólo se ha utilizado aquí para obtener una idea de cómo sería un controlador óptimo que conociese el sistema. Este resultado se muestra en la Figura 3. Sin embargo, como se ha indicado previamente, este modelo de ODEs es desconocido, de caja negra, por lo que experimentalmente sólo hemos podido obtener un conjunto de datos a partir de señales de entrada.

Basándose en los datos se ha obtenido el predictor como se indica en las secciones 2 y 3. Los resultados de este controlador aplicado sobre el CSTR se muestran en la Figura 4. Como se puede observar, los resultados no son satisfactorios. Esto se debe a problemas de implementación que se analizan en la siguiente sección.

4.3. Mejoras en la implementación

Como parte experimental de la teoría presentada en [13], ciertos aspectos prácticos surgen a la hora de implementar los controladores allí definidos. En este apartado se analizan posibles modificaciones para reducir la carga computacional y para mejorar la actuación del optimizador.

4.3.1. Partición en árboles

El propósito de esta línea de investigación es el manejo de una cantidad de datos suficiente no sólo

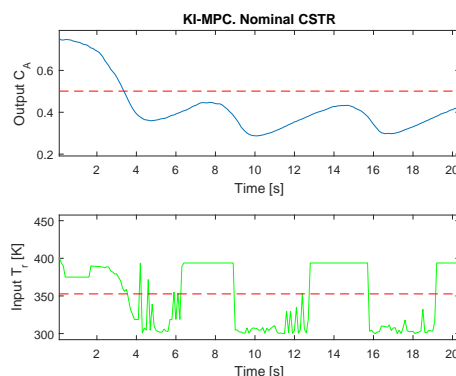


Figura 4: Primeros ensayos realizados usando el predictor KI, con problemas en la resolución del optimizador

para explorar toda la región de trabajo, sino también para proporcionar una predicción veraz y con error acotado, que permita manejar sistemas no lineales con restricciones. Los modelos de predicción presentados, basados en datos, requieren un dataset que a menudo no es pequeño, y que puede contar con numerosas dimensiones (tanto en entrada como en salida) dado el carácter dinámico y multivariable de los sistemas considerados.

Por todo ello, creemos necesario desarrollar metodologías que reduzcan la carga computacional de los algoritmos, y poder así implementar un control en tiempo real. Una de estas metodologías consiste en dividir el espacio de trabajo con técnicas de aprendizaje automático como *clustering* o *nearest neighbors*. Cierta bibliografía respecto a este tema puede encontrarse en [2], [4] y [8].

En este trabajo se han realizado particiones iguales en el espacio de entrada $\mathcal{X} \in \mathbb{R}^3$ del dataset \mathcal{D}_n , de forma que un nuevo punto a interpolar $x \notin \mathcal{D}_n$ se situará primero en la división que le corresponde, y se realizará la predicción utilizando solamente un subconjunto de \mathcal{D}_n alrededor de x .

Dividiendo en 10 particiones por cada dimensión del regresor, el tiempo de cálculo se reduce hasta un 90%, sin aumentar el error de predicción, $\mu = \mathfrak{d}_Y(\tilde{f}, \hat{f})$.

4.3.2. Filtrado del predictor

El término *Kinky Inference* hace referencia al resultado del predictor descrito en la ecuación (3). Debido al tipo de interpolación que se hace basada en los datos y la continuidad de Lipschitz, la función resultante \hat{f} no es diferenciable, sino que presenta una forma angulosa, que propicia el término *kinky* en inglés.

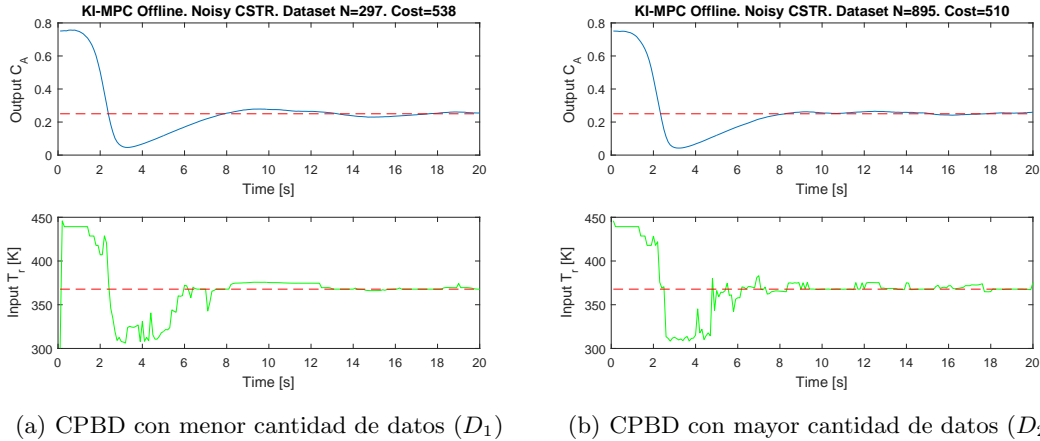


Figura 5: Control predictivo basado en datos para un reactor CSTR, proporcionando al modelo datasets con diferente tamaño

Esta función forma parte del problema de optimización, y el hecho de ser no diferenciable perjudica gravemente los métodos de optimización gradenciales como la función *fmincon*. Es nuestro objetivo, por tanto, suavizar o filtrar de cierta forma esta \hat{f} para convertirla en diferenciable, manteniendo la misma $L_{\mathcal{D}}$.

En [4] se propone una nueva versión del predictor conocida como *smoothed kinky inference* (SKI), que consiste en realizar una combinación convexa de ciertos puntos alrededor del punto x a interpolar, como se indica a continuación:

$$f_n^*(x) = \frac{1}{\sum_{i=-q}^q w_i} \sum_{i=-q}^q w_i \hat{f}_n(x + i\sigma_n) \quad (12)$$

para q puntos alrededor de x con pesos w_i y siendo σ_n un vector en \mathbb{R}^{n_x} que indica el incremento en cada componente dimensional.

En la Figura 5 se muestra el resultado de los experimentos tras aplicar las mejoras citadas en esta sección. Se realiza además con diferentes datasets, con el objetivo de demostrar que con más datos se mejora la actuación, y motivando la teoría presentada en [13].

5. Conclusiones y futuras líneas

Se han presentado las primeras líneas para el proyecto de investigación de controladores predictivos basados en datos. A partir de los problemas desarrollados en [13], se ha considerado un reactor continuamente agitado como modelo de caja negra, por lo que ha sido necesario diseñar el método para obtener el predictor, mediante técnicas de identificación. Se ha construido el modelo NARX,

y estimando la constante de Lipschitz se ha construido el predictor de KI. Posteriormente se ha diseñado un MPC usando ese modelo, y se han realizado ensayos de control. Se proponen algunas de mejoras y se muestran los resultados obtenidos, aplicados al CSTR.

Como se comentaba, este artículo representa una primera aproximación al control basado en datos. Experimentalmente, algunos de los próximos pasos a seguir en esta línea comprenden:

- La adaptación de los algoritmos al control robusto, sin pérdida de generalidad ni de estabilidad.
- La versión *online*, o en línea, del controlador. Es nuestro objetivo tener en cuenta nuevos puntos que vayan surgiendo durante la operación del sistema, y que permitan mejorar nuestro controlador. Destacamos tres usos:
 - Para actualizar la constante de Lipschitz del sistema. Como se comentaba en la sección 2, la estimación de $L_{\mathcal{D}}$ a partir de los datos es menor o igual que la real, por lo que nuevos datos permitirán un cálculo recursivo para mejorar dicha estimación.
 - Para añadir puntos al dataset con los que mejorar la interpolación. Se deberá llegar a un compromiso entre reducir el error de predicción y aumentar la carga computacional por el incremento del dataset.
 - Para explorar regiones del espacio \mathcal{X} no contempladas en \mathcal{D}_n , si bien deberá hacerse sin incrementar el error de estimación.
- Seguir implementando técnicas para reducir la carga computacional de los algoritmos,

pues al realizar la interpolación entre puntos el tiempo de cálculo dista del deseado. Una técnica a estudiar será la paralelización de estos algoritmos.

Agradecimientos

A los autores les gustaría agradecer al MINECO y a los fondos FEDER (contratos DPI2013-48243-C2-2-R y DPI2016-76493-C3-1-R) por financiar este trabajo.

Referencias

- [1] Aswani, A., Gonzalez, H., Sastry, S.S., & Tomlin, C. (2013). *Provably safe and robust learning-based model predictive control*. *Automatica*, 49(5), 1216–1226.
- [2] Beliakov, G. (2006). *Interpolation of Lipschitz functions*. *Journal of Computational and Applied Mathematics*, 196(1), 20–44.
- [3] Bravo, J.M. et al. (2016). *A General Framework for Predictors based on Bounding Techniques and Local Approximation*. *IEEE Transactions on Automatic Control*.
- [4] Calliess, J.P. (2014). *Conservative decision-making and inference in uncertain dynamical systems*. Ph.D. Thesis, University of Oxford.
- [5] Calliess, J.P. (2016). *Lazily Adapted Constant Kinky Inference for Nonparametric Regression and Model-Reference Adaptive Control*. arXiv preprint arXiv:1701.00178.
- [6] Camacho, E. F., & Bordons, C. (2013). *Model predictive control*. Springer Science & Business Media.
- [7] Canale, M., Fagiano, L., & Signorile, M.C. (2014). *Nonlinear model predictive control from data: a set membership approach*. *International Journal of Robust and Nonlinear Control*, 24(1), 123–139.
- [8] Kvasov, D. & Sergeyev, Y. (2012). *Lipschitz gradients for global optimization in a one-point-based partitioning scheme*. *Journal of Computational and Applied Mathematics*, 236(16), 4042–4054.
- [9] Leontaritis, I.J & Billings, S.A. (1985). *Input-output parametric models for nonlinear systems. Part I: deterministic nonlinear systems*. *International Journal of Control* 41:303-328.
- [10] Limon, D. (2002). *Control predictivo de sistemas no lineales con restricciones: estabilidad y robustez*. Tesis doctoral, Universidad de Sevilla.
- [11] Limon, D., Alamo, T., Salas, F., & Camacho, E.F. (2006). *On the stability of MPC without terminal constraint*. *IEEE Transactions on Automatic Control*, 51(5), 832–836.
- [12] Limon, D., Alamo, T., et al. (2009). *Input-to-state stability: a unifying framework for robust model predictive control*. *International Workshop on Assessment and Future Direction of Nonlinear Model Predictive Control*, L. Magni, D. M. Raimondo, & F. Allgöwer, Eds. Springer, 2009, pp. 1–26.
- [13] Limon, D., Callies, J.P., Maciejowski, J.M. (2017). *Learning-based Nonlinear Model Predictive Control*. IFAC World Congress, Toulouse, Francia.
- [14] Rivera, D.E. & Kyoung S. (2000) *An integrated identification and control design methodology for multivariable process system applications*. *IEEE Control Systems*, 20(3), 25–37.
- [15] Seborg, D. E., Edgar, T. & Mellichamp, D. (1989). *Process Dynamics and Control*, Wiley, 3ª Edición, Capítulo 2.4.6.