Control no lineal adaptativo con identificación dispersa

Roger Domingo-Enrich
roger.domingo.enrich@estudiantat.upc.edu
Andreu Cecilia*
andreu.cecilia@upc.edu
Ramon Costa Castelló*
ramon.costa@upc.edu
*Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028 Barcelona, Spain

Resumen

En este trabajo se propone un controlador con estimación de parámetros enfocada a sistemas no lineales, desconocidos y cambiantes en el tiempo. Se presenta un algoritmo de aprendizaje en línea para estimar las ecuaciones del sistema y un controlador que usa esta información para estabilizar la planta. Para ello, se han combinado un observador no lineal y un estimador de parámetros con técnicas de optimización dispersa. En el algoritmo hay dos fases diferenciadas: la fase de aprendizaje, donde se aprenden en línea las funciones desconocidas, y la fase de ejecución, donde se controla la planta con unos objetivos preestablecidos. El algoritmo propuesto es capaz de decidir de forma autónoma en qué fase se encuentra, mediante una métrica de aprendizaje. Este algoritmo se ha validado por simulación numérica en una planta roto-magnet.

Palabras clave: Observador de alta ganancia extendido, estimación de parámetros dispersa, métrica de aprendizaje.

1. Introducción

Actualmente, más del 90 % de los controladores de la industria son del tipo PID [1]. Su popularidad es resultado de su simplicidad, funcionalidad clara y aplicabilidad. Sin embargo, encuestas recientes [2] muestran que las prestaciones de esta tecnología no son suficientes para atajar los problemas o las tendencias actuales y es necesario desarrollar nuevos algoritmos que sustituyan o mejoren el PID.

Una de las limitaciones del PID es su dificultad de sintonía en presencia de no linealidades. En la práctica, los factores no lineales se consideran como perturbaciones a dominar por el controlador. Esta estrategia conlleva diseñar controladores demasiado conservadores con bajas prestaciones durante el transitorio. Actualmente, este hecho no es un factor limitante, ya que la mayoría de controladores tienen como objetivo mantener una salida cerca de un punto de equilibrio. En este contexto, las no linealidades tienen un peso muy bajo en la dinámica del sistema y el PID es una opción más que suficiente. Sin embargo, cada vez se requiere de algoritmos que sean capaces de seguir trayectorias complejas y operar lejos del punto de equilibrio del sistema. Es en esta situación donde la linealidad del PID limita las prestaciones del sistema y es necesario desarrollar algoritmos que trabajen con sistemas no lineales.

El control de sistemas no lineales es un campo que se lleva estudiando desde hace décadas [3]. Pese a los numerosos resultados de aplicaciones prácticas, existen diversos factores que limitan su aplicación directa en la industria. Primero, muchos algoritmos asumen un conocimiento exacto de todos los estados del sistema [4]. Este supuesto no es realista, ya que en la práctica sólo se dispone de la información de unos pocos sensores. Segundo, muchos algoritmos suponen un conocimiento exacto del modelo del sistema [5]. Este supuesto incrementa el coste económico del controlador, ya que requiere una identificación previa del sistema.

Este trabajo presenta un algoritmo de control de sistemas no lineales que pretende relajar las restricciones mencionadas. En primer lugar, el algoritmo incorpora un observador que estima los estados del sistema y la incertidumbre del modelo. En teoría, este elemento soluciona las limitaciones aludidas del control no lineal. Sin embargo, se muestra que el observador ofrece bajas prestaciones en presencia de mucha incertidumbre.

Por eso, en segundo lugar, se complementa el observador con un algoritmo que estima el modelo matemático del sistema en línea. Este paso se realiza comúnmente asumiendo que el modelo tiene forma de regresión lineal y aplicando un algoritmo de mínimos cuadrados recursivos [6]. Sin embargo, este tipo de algoritmo es frágil ante la incertidumbre en el regresor [8]. Es decir, si alguna función del regresor es desconocida, el algoritmo presentará muy baja precisión. En consecuencia, este trabajo propone un método para estimar las funciones del regresor. La idea es considerar un vector de "funciones candidatas"que pueden o no aparecer en el sistema y estimar los parámetros de este vector mediante regresión lineal dispersa.

Finalmente, a diferencia de muchas técnicas de aprendizaje [9], el algoritmo propuesto incorpora una métrica que le permite cuantificar la precisión de los parámetros estimados. Mediante esta métrica se podrá discernir entre una fase de aprendizaje, en la que el controlador está aprendiendo el sistema, y una fase de ejecución, en la que el controlador sigue la referencia deseada. Además, el controlador se complementa con la información de la métrica para robustecer el algoritmo ante la diferencia entre el modelo aprendido y la realidad; así se puede garantizar la estabilidad durante la fase de aprendizaje.

Este trabajo está organizado de la siguiente forma. En la sección 2, se formula el problema matemático a resolver. En la sección 3, se presenta el algoritmo de control. En la sección 4, se valida numéricamente el algoritmo en una planta rotomagnet. Finalmente, en la sección 5 se presentan las conclusiones del trabajo.

2. Formulación del problema

Este trabajo considera sistemas de orden n no lineales con una entrada y una salida representadas a través de la siguiente forma canónica:

$$\begin{aligned} \dot{x}_i &= x_{i+1}, \qquad i = 1, \dots, n-1 \\ \dot{x}_n &= \phi(\boldsymbol{x}) + u \\ y &= x_1, \end{aligned} \tag{1}$$

donde $\boldsymbol{x} = [x_1, \dots, x_n]^\top \in \mathbb{R}^n$ son los estados, $u \in \mathbb{R}$ es la entrada, $y \in \mathbb{R}$ es la salida mesurable y $\phi(\boldsymbol{x})$ es una función posiblemente no lineal.

Nota 2.1. *Si se cumplen ciertas condiciones, cualquier sistema de grado relativo n se puede transformar a la forma (1) a través del cambio de coordenadas adecuado [5].*

El objetivo principal del trabajo es diseñar un controlador que a través de la señal de control, u, haga el origen del sistema (1) estable, es decir, tal que

$$\lim_{t \to \infty} \boldsymbol{x} = 0.$$

Nota 2.2. *Todos los resultados que se proponen en este trabajo siguen siendo válidos si, en lugar de estabilizar la planta en el origen, se desea que la salida converja a una trayectoria y_{ref} predefinida y posiblemente no constante. Sólo es necesario realizar un cambio de coordenadas [10]. Para más detalles, consultar la sección 4.*

Este problema se ha estudiado en profundidad en la literatura de control no lineal [3]. Una solución relativamente sencilla consiste en cancelar las no linealidades del sistema mediante la acción de control, u. De esta manera, el sistema resultante es completamente lineal y se puede controlar con algoritmos de control lineales. En concreto, si se introduce la siguiente señal de control

$$u = -\phi(\boldsymbol{x}) + v, \tag{2}$$

el sistema (1) se transforma en

$$\begin{aligned} \dot{x}_i &= x_{i+1}, \qquad i = 1, \dots, n-1 \\ \dot{x}_n &= v \\ y &= x_1, \end{aligned} \tag{3}$$

el cual es completamente lineal de v a y y se puede controlar con un controlador de realimentación lineal de estados.

La diferencia entre este trabajo y los resultados existentes en la literatura aparece al considerar el siguiente escenario. En lugar de suponer que se conoce exactamente la función $\phi(\boldsymbol{x})$ [11], que es un supuesto muy restrictivo, se asume que pertenece a una librería de funciones **sobre-completa** conocida, $\boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{\theta}$, la cual cumple las siguientes características:

- $\boldsymbol{\psi}(\boldsymbol{x}) \in \mathbb{R}^{1 \times q}$ es un vector de funciones conocido y $\boldsymbol{\theta} \in \mathbb{R}^{q \times 1}$ es un vector de parámetros desconocido y a estimar por el algoritmo.
- La base de funciones cumple $\boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{\theta} = \phi(\boldsymbol{x})\ \forall \boldsymbol{x}$.
- El vector θ es **disperso**. Es decir, tiene muchos elementos iguales a cero.

Por ejemplo, $\boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{\theta}$ podría ser una librería de funciones candidatas que podrían (o no) aparecer en la planta real.

Al considerar este escenario, aparecen diversas limitaciones que impiden implementar muchas de las soluciones de la literatura. Primero, los estados $\boldsymbol{x}$ no son conocidos: por tanto, se tienen que estimar a través de un observador. Segundo, el vector $\boldsymbol{\theta}$ no es conocido. En consecuencia, se tienen que estimar estos parámetros en línea para poder implementar el controlador (2). Finalmente, el punto más importante es que, como $\boldsymbol{\theta}$ es disperso, el vector $\boldsymbol{\psi}(\boldsymbol{x})$ nunca presentará la excitación suficiente para poder estimar los parámetros $\boldsymbol{\theta}$ [6]. Por eso, algoritmos de estimación de parámetros, como los mínimos cuadrados recursivos, no se pueden utilizar en este escenario. Este punto se discutirá con más detalle en la subsección 3.2.

El objetivo de este trabajo es presentar un algoritmo de control que sea capaz de operar con un vector de parámetros $\boldsymbol{\theta}$ disperso y sólo midiendo la salida del sistema.

3. Propuesta de solución

La solución propuesta se compone de tres partes. Primero, un observador no lineal que estima los estados del sistema. Segundo, un estimador de parámetros basado en regresión lineal dispersa, que a partir de las estimaciones del observador puede estimar los regresores y los parámetros de la planta real. Finalmente, una métrica que indica al sistema cuándo aprender los parámetros y cuándo ejecutar el control deseado.

3.1. Observador de alta ganancia extendido

El observador no lineal utilizado en el trabajo es el observador de alta ganancia extendido [10]. A diferencia de otros observadores, tiene como ventaja que, además del vector de estados, también estima la incertidumbre del modelo. La incertidumbre, σ, es la diferencia entre la función desconocida y el modelo matemático:

$$\sigma = \phi(\boldsymbol{x}) - \boldsymbol{\psi}(\boldsymbol{x})\hat{\boldsymbol{\theta}}, \tag{4}$$

donde $\hat{\boldsymbol{\theta}}$ es una estimación de los parámetros $\boldsymbol{\theta}$.

Con la ecuación (4) el sistema (1) se puede reescribir como:

$$\begin{aligned} \dot{x}_i &= x_{i+1}, \qquad i = 1, \ldots, n-1 \\ \dot{x}_n &= \boldsymbol{\psi}(\boldsymbol{x})\hat{\boldsymbol{\theta}} + \sigma + u \\ y &= x_1. \end{aligned} \tag{5}$$

En este tipo de sistemas, un observador de alta ganancia extendido es una copia del sistema más un factor proporcional al error entre la salida del sistema, x_1, y la salida del sensor, y:

$$\begin{aligned} \dot{\hat{x}}_i &= \hat{x}_{i+1} + \frac{\alpha_i}{\varepsilon^i}(y - \hat{x}_1), \qquad i = 1, \ldots, n-1 \\ \dot{\hat{x}}_n &= \boldsymbol{\psi}(\hat{\boldsymbol{x}})\hat{\boldsymbol{\theta}} + \hat{\sigma} + u + \frac{\alpha_n}{\varepsilon^n}(y - \hat{x}_1) \\ \dot{\hat{\sigma}} &= \frac{\alpha_{n+1}}{\varepsilon^{n+1}}(y - \hat{x}_1) \end{aligned} \tag{6}$$

donde $\boldsymbol{\psi}(\hat{\boldsymbol{x}})\hat{\boldsymbol{\theta}}$ es una estimación de la función desconocida $\boldsymbol{\phi}(\boldsymbol{x})$ que se desarrollará en la subsección 3.2.

El observador incluye un estado extra, $\hat{\sigma}$, el cual es una estimación de la incertidumbre del modelo, σ.

Los parámetros a especificar para el observador son las constantes α_i y la constante ε. Por un lado, las constantes α_i determinan el comportamiento transitorio del error de estimación, i.e. $\tilde{x} = x - \hat{x}$. Para que este error converja a cero hay que diseñar las constantes α_i de tal forma que todas las raíces del siguiente polinomio presenten parte real negativa:

$$s^n + \alpha_1 s^{n-1} + \cdots + \alpha_{n-1} s + \alpha_n. \tag{7}$$

Por otro lado, el parámetro ε es una constante positiva que determina las prestaciones del observador. Si escogemos un valor de ε demasiado pequeño, amplificamos el efecto del ruido y el fenómeno de pico [12][13]. Por el contrario, si escogemos un valor demasiado grande, amplificamos el efecto de la incertidumbre y ralentizamos la dinámica del observador, por lo que el error de estimación va a converger más lentamente. No hay ningún método directo para determinar el valor óptimo del parámetro; en la práctica, se busca un equilibrio de las prestaciones del observador.

3.2. Estimador de parámetros disperso

Una parte esencial del observador y el controlador es la necesidad de estimar los parámetros $\hat{\boldsymbol{\theta}}$. Este no es un problema sencillo, ya que sólo se dispone de la información del sensor y, y los parámetros, $\boldsymbol{\theta}$, no están directamente relacionados con esta señal. Sin embargo, la estructura del observador propuesto (6) se presta a simplificar este conflicto. En concreto, a través de la definición de σ (4) se puede deducir la siguiente relación:

$$Y \triangleq \sigma + \boldsymbol{\psi}(\hat{\boldsymbol{x}})\hat{\boldsymbol{\theta}} = \boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{\theta}. \tag{8}$$

Nótese que (8) es una regresión lineal en los parámetros. Sin embargo, el valor de σ no es conocido: sólo se dispone de la estimación $\hat{\sigma}$ realizada por el observador. En consecuencia, sólo se tiene una aproximación de la regresión anterior:

$$\hat{Y} \triangleq \hat{\sigma} + \boldsymbol{\psi}(\hat{\boldsymbol{x}})\hat{\boldsymbol{\theta}} = \boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{\theta} - [\sigma - \hat{\sigma}]. \tag{9}$$

Para estudiar la validez de la regresión (9) es importante remarcar dos puntos. Primero, si la estimación de la incertidumbre es precisa, i.e. $\sigma - \hat{\sigma} \approx 0$, la regresión (9) converge a la regresión ideal (8). Segundo, cuanto mayor sea la precisión de la estimación de los parámetros, $\hat{\boldsymbol{\theta}}$, mayor será la precisión de la estimación de la incertidumbre $\hat{\sigma}$. En consecuencia, al resolver iterativamente la regresión lineal (9) e inyectar iterativamente los resultados en el observador (6), se puede conseguir una estimación $\hat{\boldsymbol{x}}, \hat{\boldsymbol{\theta}}$ y $\hat{\sigma}$ que converja al valor real.

El único punto que queda por clarificar es como estimar los parámetros, $\hat{\boldsymbol{\theta}}$, de la regresión (9). En la estructura de (9) es común implementar un algoritmo de mínimos cuadrados recursivos [6][7]. Es decir, implementar un algoritmo que resuelva la siguiente optimización

$$\min_{\theta} \|\hat{Y} - \boldsymbol{\psi}(\hat{\boldsymbol{x}})\boldsymbol{\theta}\|_2, \tag{10}$$

donde $\|\cdot\|_2$ es la norma 2.

Es bien sabido que el algoritmo de mínimos cuadrados converge al valor real si el vector $\boldsymbol{\psi}(\hat{\boldsymbol{x}})$ cumple con cierta condición de excitación [6]. Esta condición establece que las trayectorias del sistema son "suficientemente ricas", de forma que sólo exista una única solución de parámetros, $\boldsymbol{\theta}$, que explique la trayectoria.

Sin embargo, en este trabajo se consideran vectores $\boldsymbol{\theta}$ dispersos. Por tanto, la función $\boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{\theta}$ contiene muchas más funciones de las necesarias para explicar la trayectoria del sistema. En este contexto, aunque la base de funciones mínimas pueda cumplir la condición de excitación [6], la base sobre-completa $\psi(\boldsymbol{x})$, en general, nunca cumplirá esta condición.

Por eso, es necesario estimar los parámetros de la regresión lineal (9) con un algoritmo distinto a los mínimos cuadrados recursivos. La idea es incorporar la propiedad de dispersión de $\boldsymbol{\theta}$ en la estimación de parámetros. Por eso, se incorpora un factor de regularización que penaliza la norma-1 de los parámetros. En concreto, se resuelve el siguiente problema:

$$\min_{\theta} \|\hat{Y} - \boldsymbol{\psi}(\hat{\boldsymbol{x}})\boldsymbol{\theta}\|_2 + \lambda|\boldsymbol{\theta}|_1, \tag{11}$$

donde λ es un parámetro positivo a sintonizar y $\|\cdot\|_1$ es la norma 1.

Para más detalles sobre este tipo de optimización, se remite el lector a la referencia [14].

3.3. Métrica de aprendizaje

El algoritmo divide el control en línea en dos fases diferentes: una fase de aprendizaje en la que se estima la función desconocida del sistema, y una fase de ejecución en la que se controla el sistema para conseguir unos objetivos preestablecidos. No se pueden estimar parámetros constantemente, sino que es necesario dividir el algoritmo en dos fases, ya que en el aprendizaje es preciso que la trayectoria de las variables de estado x cumpla la condición de excitación permanente. En algunas ocasiones se tiene que excitar externamente el sistema en esta fase para conseguir una buena estimación de los parámetros.

Se ha diseñado una métrica de aprendizaje, M, para que el algoritmo distinga de forma autónoma en qué fase del control en línea debe estar el sistema. El cálculo de la métrica se basa en la estimación de la incertidumbre, $\hat{\sigma}$, que nos proporciona el observador (6). La métrica nos permite cuantificar la incertidumbre del sistema en un espacio de tiempo T y se calcula de la siguiente forma:

$$M = \frac{\int_t^{t+T} |\hat{\sigma}_{filtrado}| \cdot dt}{T}. \tag{12}$$

Si el valor la métrica de aprendizaje es alto implica que hay bastante incertidumbre en el sistema y que no se tiene una buena estimación de los parámetros. El algoritmo calcula la métrica cada T segundos, y evalúa si se encuentra por debajo del umbral de lo que consideramos una buena estimación. Si es así, el sistema pasa a la fase de ejecución; si no, es que ya se encontraba en ella. Por lo contrario, el sistema salta a la fase de aprendizaje; si no, es que ya se encontraba en ella.

En la fase de aprendizaje se realizan iteraciones que duran un tiempo T, en las que se estiman los parámetros $\hat{\boldsymbol{\theta}}$. Al final de cada iteración se calcula la métrica de aprendizaje M para evaluar si se tiene que seguir aprendiendo el sistema; si es así, se actualizan las estimaciones de parámetros en el observador (6) y en el controlador (13). En la fase de ejecución también se calcula el parámetro de aprendizaje cada cierto período de tiempo T para detectar si la planta real ha cambiado.

3.4. Controlador final

El controlador final realimenta a la entrada del sistema (5) la siguiente señal de control:

$$u = v - \boldsymbol{\psi}(\hat{\boldsymbol{x}})\hat{\boldsymbol{\theta}} - \hat{\sigma}, \tag{13}$$

donde la señal v corresponde a un controlador de realimentación lineal de estados, y determina la evolución temporal de la salida del sistema linealizado. La estimación de la incertidumbre, $\hat{\sigma}$, y la estimación de los estados usada en v y en el vector de funciones, $\boldsymbol{\psi}(\boldsymbol{x})$, provienen del observador de alta ganancia extendido. Estas estimaciones se van actualizando constantemente para seguir la trayectoria de los estados reales. Por último, la estimación de los parámetros, $\hat{\boldsymbol{\theta}}$, proviene del estimador. La estimación de parámetros se actualiza al final de cada iteración de la fase de aprendizaje; si el sistema está en la fase en ejecución, no se actualiza porque la estimación ya se considera buena.

Con esta señal de control (13), por un lado se consigue controlar el comportamiento del sistema linealizado con la señal v de realimentación de los estados. Por otro lado, se cancelan exactamente las no linealidades y la incertidumbre del sistema, haciéndolo así un controlador más robusto que un PID tradicional.

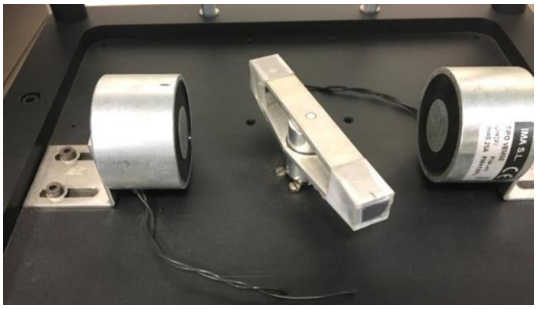

Figura 1: Estructura de la planta roto-magnet

4. Validación numérica

El algoritmo propuesto se ha validado con la planta roto-magnet [16]. En esta sección, se va a explicar cómo es y qué características tiene, y posteriormente se va a mostrar un ejemplo de simulación aplicando el algoritmo propuesto.

4.1. Explicación de la planta

La planta roto-magnet está diseñada para ser controlada con fines educativos. Esta planta es un sistema mecatrónico que está compuesto por: un motor DC, una barra giratoria, un imán permanente y dos electroimanes fijados. Más concretamente, el imán permanente con dos polos magnéticos opuestos está montado encima de la barra giratoria. Los dos electroimanes, con campos magnéticos diferentes, están fijados cerca de la barra, como se ve en la Fig. 1. El pequeño motor DC, situado debajo de la mesa experimental, está unido a la parte inferior de la barra.

Durante el funcionamiento del motor DC, la interacción entre el imán permanente y los electroimanes fijados crea un campo magnético. Este campo magnético causa un par pulsante Γ_p en el movimiento de la barra, que depende de la posición angular relativa Θ entre la barra y los electroimanes.

Para tener la misma nomenclatura que en los sistemas anteriores, se considera que el ángulo Θ es el estado x_1, y que la velocidad angular es el estado x_2. La salida de la planta es el ángulo relativo entre los imanes, que se mide con un sensor de posición angular.

El modelo de esta planta puede describirse como [15]:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\alpha_1 x_2 + \Gamma_p(x_1) + \alpha_2 u \\ y &= x_1, \end{aligned} \tag{14}$$

donde α_1 y α_2 son dos constates relacionadas con la fricción, y u es la acción de control que representa el par del motor. En algunos trabajos [15] el par pulsante puede ser definido como la suma de un seno y un coseno dependientes del ángulo $\Gamma_p(x_1) = \delta \sin(kx_1) + \gamma \cos(kx_1)$.

Esta planta presenta dos problemas. El primero es que los parámetros $\alpha_1, \delta_k, \gamma_k$ y la frecuencia k de las sinusoidales son difíciles de estimar. El segundo es que, cuando los electroimanes fijos se desactivan y no generan un campo magnético, el par pulsante, Γ_p, es igual a cero. En este trabajo se asume que el controlador no tiene ninguna información sobre la activación o desactivación del campo electromagnético. Entonces, el algoritmo tiene que, por un lado, ser capaz de calcular los parámetros desconocidos, y por otro, detectar si el campo magnético está activo e incluir (o no) las funciones sinusoidales de frecuencia desconocida en el modelo de la planta.

Este problema se puede formular en el contexto de optimización dispersa planteada en este trabajo. Este hecho se presenta en detalle en la siguiente sección.

4.2. Aplicación del algoritmo

A continuación, vamos a ver los diferentes pasos a seguir para aplicar el algoritmo en un ejemplo para conseguir que el ángulo x_1 de la barra giratoria del roto-magnet siga una trayectoria angular predefinida y no constante. El algoritmo se divide en dos partes.

La primera parte consiste en diseñar los diferentes elementos del sistema: el observador, el controlador y la librería de funciones.

Paso 1. Se define la librería de posibles funciones. Las únicas informaciones de las que se dispone es que hay fricción en el sistema y que el par pulsante es una suma de sinusoidales que dependen del ángulo. Por esto se incluye el ángulo, la velocidad angular y sinusoidales de diferentes frecuencias en la librería de posibles funciones:

$$\begin{aligned} \boldsymbol{\psi}(\boldsymbol{x}) = [&x_1, x_2, \sin(x_1), \sin(\frac{x_1}{2}) \\ &\sin(\frac{x_1}{3}), \cos(x_1), \cos(\frac{x_1}{2}), \cos(\frac{x_1}{3})]. \end{aligned} \tag{15}$$

Nótese que en esta librería sólo tres factores aparecen realmente en la planta. Estos son: $x_2, \sin(x_1)$ y $\cos(x_1)$.

Paso 2. Se diseña el observador de alta ganancia extendido (6), asignando los valores a las constantes: $\alpha_1 = 15, \alpha_2 = 75, \alpha_3 = 125$ y $\varepsilon = 0{,}55$.

Paso 3. Se diseña el controlador (13), donde la acción proporcional es $v = -[3 \quad 3{,}5]\hat{\boldsymbol{x}}$, que estabiliza los polos del sistema linealizado en -1 y $-1{,}5$.

Paso 4. Como se desea que la salida la planta converja a una trayectoria no constante, se realiza un cambio de coordenadas en las variables de estado de los componentes definidos en los pasos anteriores:

$$\begin{aligned}\bar{x}_1 &= x_1 - y_{ref}\\ \bar{x}_2 &= x_2 - \dot{y}_{ref},\end{aligned} \tag{16}$$

donde la trayectoria a seguir por el ángulo x_1 del roto-magnet es:

$$y_{ref}(t) = \sin(t) + 2\sin\left(\frac{t}{2}\right) + \frac{3}{2}\cos(t). \tag{17}$$

La segunda parte del algoritmo consiste en determinar cómo funcionan las fases de aprendizaje y de ejecución del control en línea. La fase de aprendizaje consiste en hacer iteraciones en las cuales se estiman los parámetros. En cada iteración, primero se aplica la optimización dispersa (11) para recuperar la base de funciones que realmente aparece en la planta, y luego la optimización de los mínimos cuadrados (10) para mejorar la estimación de los parámetros.

Paso 5. Se tiene el modelo de regresión:

$$\hat{Y} = \boldsymbol{\psi}(\hat{\boldsymbol{x}})\hat{\boldsymbol{\theta}}. \tag{18}$$

En cada iteración se toman m medidas en línea de las variables $\hat{Y}$ y de las funciones de la librería (15), a partir de las estimaciones del observador. Estas medidas conforman las matrices $\hat{\boldsymbol{Y}}$ y $\hat{\boldsymbol{\Psi}}$ de tamaño mx1 y mx8, respectivamente, con las que se resuelve el problema de optimización (11). Se sintoniza el parámetro λ que minimiza el error de regresión y el numero de funciones regresoras. Como resultado de la optimización se obtiene una estimación de los parámetros $\boldsymbol{\theta}$ dispersa, ya que se minimiza el número de funciones de la librería (15) que aparecen en el modelo de regresión (18).

Paso 6. Se aplica la optimización de mínimos cuadrados (10), sólo considerando las columnas de la matiz $\hat{\boldsymbol{\Psi}}$ que corresponden a los regresores que han obtenido una estimación de su parámetro diferente de cero en el *paso 5*. Como resultado se obtiene una mejor estimación de los parámetros, ya que en los mínimos cuadrados sólo se minimiza el error de regresión. Así se consigue un modelo, $\boldsymbol{\psi}(\boldsymbol{x})\boldsymbol{\theta}$, de la función desconocida, $\boldsymbol{\phi}(\boldsymbol{x})$, con las funciones regresoras más importantes. Este modelo se actualiza en el observador y en el controlador al final de cada iteración de la fase de aprendizaje.

Paso 7. Se tiene que definir el umbral de la métrica de aprendizaje que indique al sistema cuándo aprender la función desconocida y cuándo ejecutar el control preconcebido. Se calcula la métrica M cada 25 segundos, que en el caso de estar en la

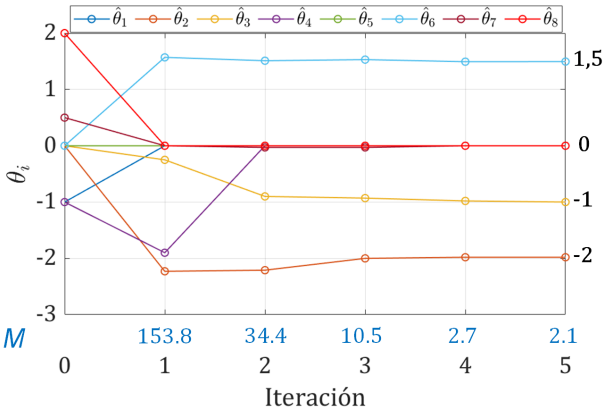

Figura 2: Evolución de las estimaciones $\hat{\boldsymbol{\theta}}$ y de la métrica M entre los segundos 125 y 150

fase de aprendizaje coincide con la duración de las iteraciones. Esta ventana de tiempo de 25 segundos se ha escogido teniendo en cuenta la dinámica de la referencia a seguir (17). Si el valor de la métrica está por encima de 2,5, se tiene que estar en la fase de aprendizaje porque hay incertidumbre en el sistema. En cambio, si la métrica está por debajo de 2,5, se ejecuta el control preestablecido sin estimar los parámetros. Se ha elegido el umbral 2,5 porque, por debajo de él, tenemos una buena estimación $\boldsymbol{\psi}(\hat{\boldsymbol{x}})\hat{\boldsymbol{\theta}}$ de la función desconocida $\boldsymbol{\phi}(\boldsymbol{x})$.

4.3. Resultados

Para hacer la simulación se considera que los parámetros desconocidos de la planta (14) son: $\alpha_1 = -2$, $\alpha_2 = 1$, $\delta = -1$, $\gamma = 1{,}5$ y $k = 1$. Se asignan los parámetros de la planta real para realizar la simulación, pero el algoritmo no los conoce. Se simulará que los electroimanes generan un campo magnético los primeros 275 segundos y que luego se desactivan. El controlador desconoce el tiempo de desactivación.

En el inicio de la simulación, se considera el siguiente modelo inicial de la función desconocida, que se usará en el observador y el controlador:

$$\boldsymbol{\psi}(\boldsymbol{x})\hat{\boldsymbol{\theta}} = -x_1 - \sin(\frac{x_1}{2}) + 0{,}5\cos(\frac{x_1}{2}) + 2\cos(\frac{x_1}{3}).$$

Nótese que el modelo inicial no incluye las mismas funciones que el modelo real (14). El controlador deberá ser capaz de notar esta discrepancia y, de forma autónoma, aprender las funciones que realmente aparecen en el sistema.

Del segundo 0 al 125, el algoritmo realiza 5 iteraciones en la fase de aprendizaje. Se puede ver la evolución de la estimación de los parámetros y de la métrica en la Fig. 2. En la quinta iteración el sistema pasa a la fase de ejecución ya que el valor de la métrica de aprendizaje es menor que 2,5. La

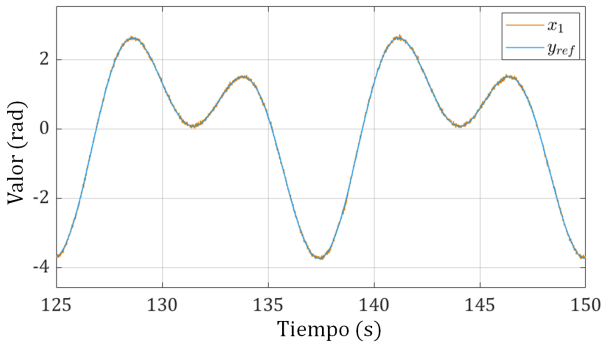

Figura 3: Evolución de x_1 y y_{ref} entre los segundos 125 y 150

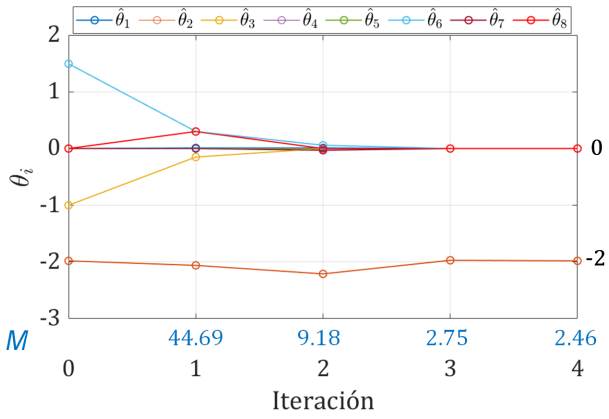

Figura 4: Evolución de las estimaciones $\hat{\boldsymbol{\theta}}$ y de la métrica M entre los segundos 275 y 375.

estimación final de la función desconocida que se usa en el observador y el controlador es:

$$\boldsymbol{\psi}(\boldsymbol{x})\hat{\boldsymbol{\theta}} = -1,99x_2 - 1,01\sin(x_1) + 1{,}50\cos(x_1).$$

En el segundo 125 empieza la fase de ejecución, donde el control del sistema funciona correctamente, como se puede apreciar en la Fig. 3.

El sistema se encuentra en la fase de ejecución hasta el segundo 300. Entre los segundos 275 y 300, la métrica M se dispara a un valor de 44,7, lo que significa que se han desactivado los electroimanes y que la planta real ha cambiado. En consecuencia, el algoritmo vuelve a entrar en la fase de aprendizaje, y se hacen 4 iteraciones en esta fase. Se puede ver la evolución de la estimación de los parámetros y de la métrica en la Fig. 4.

En el segundo 375 la métrica pasa a valer 2,46, y se considera que se han aprendido las ecuaciones del roto-magnet sin las sinusoidales correspondientes al par pulsante (14). La estimación de la función desconocida es:

$$\boldsymbol{\psi}(\boldsymbol{x})\hat{\boldsymbol{\theta}} = -1{,}99x_2.$$

El sistema vuelve a la fase de ejecución entre los segundos 375 y 500, que acaba la simulación. El sistema se controla correctamente ya que en este periodo de tiempo el seguimiento de la referencia es similar al de la Fig. 3.

5. Conclusiones

En este trabajo se ha propuesto un controlador para sistemas no lineales, con planta desconocida y cambiantes en el tiempo. Se asume que las funciones de la planta son desconocidas y sólo se dispone de una librería de posibles "funciones candidatas". Se propone un algoritmo que combina un observador de alta ganancia extendido con una optimización dispersa para así recuperar las funciones reales de la planta a la vez que se garantiza la estabilidad de la planta durante el aprendizaje. Además, se diseña una métrica que nos permite cuantificar la precisión del modelo. El algoritmo se ha validado mediante simulación numérica en una planta roto-magnet.

Agradecimientos

Este trabajo se ha realizado con el apoyo de la Agencia Estatal de Investigación española a través del sello de excelencia de María de Maeztu, IRI (MDM-2016-0656) y a través del proyecto DOVELAR (ref. RTI2018-096001-B-C32).

English summary

Nonlinear adaptive control with sparse identification

Abstract

This work presents a controller with parameter estimation focused on non-linear, unknown and time-variant systems. The proposal consists of an on-line learning algorithm to estimate the system equations, and a controller that uses this information to stabilize the plant. Our approach is to combine a non-linear observer and a parameter estimator using sparse optimization techniques. The resulting algorithm operates in two stages: the learning stage, where unknown functions are learned on-line, and the execution stage, where the plant is controlled with pre-established objectives. The proposed algorithm can autonomously decide what is its current stage by using a learning metric. We have validated our algorithm using numerical simulation in a roto-magnet plant.

Keywords: extended high gain observer, sparse parameter estimation, learning metric.

Referencias

[1] K. H. Ang, G. Chong y Y. Li. PID control system analysis, design, and technology, en IEEE Transactions on Control Systems Technology, vol. 13, no. 4, pp. 559-576, July 2005, doi: 10.1109/TCST.2005.847331.

[2] F. Lamnabhi-Lagarrigue, A. Annaswamy, S. Engell, A. Isaksson, P. Khargonekar, R. M. Murray, H. Nijmeijer, T. Samad, D. Tilbury, P. Van den Hof. Systems & control for the future of humanity, research agenda: Current and future roles, impact and grand challenges, Annual Reviews in Control, vol. 43, pp. 1-64, 2017, doi:/10.1016/j.arcontrol.2017.04.001.

[3] A. Isidori, E. D. Sontag y M. Thoma. Nonlinear control systems. Vol. 3. London: Springer, 1995.

[4] A. Isidori, A. Krener, C. Gori-Giorgi y S. Monaco. Nonlinear decoupling via feedback: A differential geometric approach, en IEEE Transactions on Automatic Control, vol. 26, no. 2, pp. 331-345, April 1981, doi: 10.1109/TAC.1981.1102604.

[5] C. I. Byrnes y A. Isidori. New results and examples in nonlinear feedback stabilization, Systems & Control Letters, vol. 12, no. 5, pp. 437-442, 1989, doi:10.1016/0167-6911(89)90080-7.

[6] L. Ljung y T. Söderström, Theory and Practice of Recursive Identification. Cambridge, 1983.

[7] A. Cecilia, M. Serra y R. Costa-Castelló. Nonlinear adaptive observation of the liquid water saturation in polymer electrolyte membrane fuel cells. Journal of Power Sources 492 (2021): 229641.

[8] A. Astolfi, D. Karagiannis, y R. Ortega. Nonlinear and adaptive control with applications. Springer Science & Business Media, 2007.

[9] R. Sun, M. L. Greene, D. M. Le, Z. I. Bell, G. Chowdhary y W. E. Dixon. Lyapunov-Based Real-Time and Iterative Adjustment of Deep Neural Networks, en IEEE Control Systems Letters, doi: 10.1109/LCSYS.2021.3055454.

[10] H. K. Khalil. High-gain observers in nonlinear feedback control. Society for Industrial and Applied Mathematics, 2017.

[11] P. A. Ioannou y J. Sun. Robust adaptive control. Courier Corporation, 2012.

[12] H. J. Sussmann y P. V. Kokotovic. The peaking phenomenon and the global stabilization of nonlinear systems. IEEE Transactions on automatic control 36.4 (1991): 424-440.

[13] A. Cecilia y R. Costa-Castelló. High gain observer with dynamic deadzone to estimate liquid water saturation in pem fuel cells. Revista Iberoamericana de Automática e Informática industrial 17.2 (2020): 169-180.

[14] D. L. Donoho, Compressed sensing, en IEEE Transactions on Information Theory, vol. 52, no. 4, pp. 1289-1306, April 2006, doi: 10.1109/TIT.2006.871582.

[15] J. Na, Y. Xing y R. Costa-Castelló, Adaptive Estimation of Time-Varying Parameters With Application to Roto-Magnet Plant, en IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, no. 2, pp. 731-741, Feb. 2021, doi: 10.1109/TSMC.2018.2882844.

[16] R. Costa-Castelló, J. Nebot y R. Grino, Demonstration of the internal model principle by digital repetitive control of an educational laboratory plant, en IEEE Transactions on Education, vol. 48, no. 1, pp. 73-80, Feb. 2005, doi: 10.1109/TE.2004.832873.

© 2021 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution CC BY-NC-SA 4.0 license (https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es).