



Universidad
Zaragoza

Trabajo Fin de Grado

Estrategia de estimación de parámetros de
máquina síncrona de imanes permanentes

Autor/es

Iván Manrique Las

Director/es

Estanislao Oyarbide Usabiaga

Escuela de Ingeniería y Arquitectura (EINA)
2019/2020

Estrategia de estimación de parámetros de máquina síncrona de imanes permanentes

Resumen

En el Control de los motores síncronos de imanes permanentes (PMSM) es fundamental conocer los parámetros del mismo motor, los cuales cambian, bien sea por la corriente de los devanados o por la temperatura de estos. En la actualidad el método utilizado para saber estos parámetros es el uso de Look-Up- tables (LUT). Aunque este método es un método eficaz tiene varios problemas. Como por ejemplo la gran necesidad de memoria necesaria para almacenar estas tablas o la poca precisión que puedan tener estas, pues para ser mas precisas se necesita una mayor memoria. Es por eso por lo que surge la necesidad de la estimación de los parámetros. Estos parámetros de los que estamos hablando son:

- Resistencia de fase del devanado del motor (R)
- Flujo de los imanes permanentes del motor (ϕ)
- Inductancia equivalente en el eje d (L_d)
- Inductancia equivalente en el eje q (L_q)

En el presente TFG se ha desarrollado el estudio de una estrategia de estimación de parámetros para ser utilizada posteriormente en el control de motores síncronos de imanes permanentes.

Tabla de contenido

NOMENCLATURA.....	3
1. INTRODUCCIÓN	4
2. OBJETIVOS	5
2.1. Fases y Temporalización	5
3. ESTADO DE LA TÉCNICA	7
3.1. Métodos con inyección de señal	7
3.2. Métodos sin inyección de señal.....	7
3.2.1. Genetic Algorithm.....	8
3.2.2. Kalman Filter (KF)	9
3.2.3. Model Reference Adaptive System (MRAS)	9
3.2.4. Affine Projection.....	10
4. ESTRATEGIA DE ESTIMACIÓN ELEGIDA	11
4.1. Explicación del algoritmo	11
4.2. Ecuaciones de Flujo y Resistencia.....	13
4.3. Ecuaciones de Inductancias	15
5. SIMULACIÓN.....	18
5.1. Estimación R.....	18
5.1.1. Valor nominal	18
5.1.2. Rampa.....	19
5.2. Estimación Flujo	20
5.2.1. Valor nominal	20
5.2.2. Rampa.....	20
5.3. Estimación R y Flujo	21
5.3.1. Valor nominal R y Flujo.....	21
5.3.2. Valor nominal Flujo y rampa en R	24
5.3.3. Valor nominal R y rampa en Flujo	24
5.3.4. Rampa en R y Flujo	25
5.4. Estimación L_d	26
5.4.1. Valor nominal	27

5.4.2.	Escalones	27
5.5.	Estimación L_q	28
5.5.1.	Valor nominal	28
5.5.2.	Escalones	29
5.6.	Estimación L_d y L_q	30
5.6.1.	Valor nominal L_d y L_q	30
5.6.2.	Valor nominal L_q y escalón en L_d	31
5.6.3.	Valor nominal L_d y escalón en L_q	32
5.6.4.	Escalones en L_d y L_q	34
6.	CONCLUSIONES	36
7.	REFERENCIAS.....	37

NOMENCLATURA

L_d	Inductancia en el eje d
L_q	Inductancia en el eje q
R	Resistencia del devanado estator
ϕ	Flujo de los imanes permanentes
p	Número de pares de polos
i_s	Amplitud de la corriente de fase
ω	Frecuencia de la intensidad de fase del estator
PMSM	Permanent Magnet Synchronous Motor. (Motor síncrono de imanes permanentes)
MTPA	Maximum Torque Per Ampere. (Máximo par por amperio)
MTPV	Maximum Torque Per Volt. (Máximo par por voltio)
V_{bus}	Tensión de alimentación del convertidor

1. INTRODUCCIÓN

En el control de los motores síncronos de imanes permanentes (PMSM) es fundamental conocer los parámetros de estos, como son R , ϕ , L_d y L_q . Actualmente estos valores se almacenan en Look-Up-tables y ocupan una gran cantidad de espacio en memoria. Además, para cada motor necesitamos sus correspondientes tablas. Son inconvenientes que podemos solucionar utilizando un estimador con el cual es posible controlar cualquier máquina de manera óptima, calculando sus parámetros online. Por estos motivos, surge la necesidad de usar estimadores, para poder ahorrar memoria en nuestras placas de control y para que el variador pueda funcionar con cualquier motor sin necesidad de reconfigurarlo. Una consideración que hay que tener en cuenta es que L_d y L_q deben estimarse simultáneamente para ser utilizadas en un control, debido a que la diferencia se emplea en métodos de control como el MTPA (máximo par por amperio) o MPTV (máximo par por voltio). Por lo cual, en el caso de no poder estimar los cuatro parámetros simultáneamente, deberemos priorizar emparejarlos, de tal forma que L_d y L_q se estimen a la vez, mientras que R y ϕ no es necesario que sean estimadas conjuntamente.

2. OBJETIVOS

Los objetivos concretos de este TFG son los siguientes:

- Desarrollar un estimador de las variables de un PMSM que tenga las siguientes características:
 - Basado en expresiones analíticas.
 - Que sea flexible ante cambios de corrientes y velocidad.
 - Que cuente con 5 entradas: Corrientes y tensiones en los ejes dq y velocidad.
 - Que permita operar en un rango amplio de velocidades.
- Desarrollar unas expresiones analíticas:
 - Que permitan el cálculo directo sin necesidad de aplicar correcciones.
 - Que no necesiten realizar barridos de cálculos para obtener el resultado.
 - Que sean lo más exactas posibles.
- Verificar el correcto funcionamiento del estimador

Se pretende lograr un bloque de estimación de las variables tomando como entradas las corrientes en los ejes d y q, las tensiones en los ejes d y q y la velocidad del motor. Y que a su vez sea flexible a los cambios en estas variables como se ve en la Fig. 1.

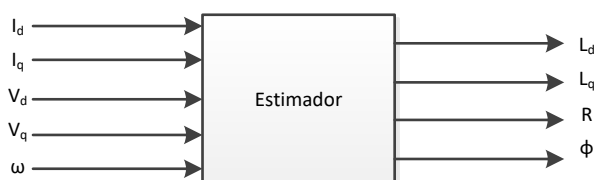


Fig. 1 Bloque estimador

2.1. Fases y Temporalización

Este trabajo está dividido en diferentes partes que requieren seguir un orden específico. A continuación, serán enumeradas con el tiempo dedicado a cada una de ellas. La duración total del proyecto ha sido de cinco meses (septiembre 2019-enero 2020)

1. Estudio previo (quince días): lectura de publicaciones centradas en distintas estrategias de estimación de parámetros.
2. Estudio analítico de las distintas situaciones de funcionamiento (un mes):

- a. Determinación de las ecuaciones que definen todas las situaciones de funcionamiento posibles.
 - b. Creación del modelo del motor en Matlab mediante Simulink.
3. Definición del algoritmo estimador (un mes).
4. Implementación del algoritmo Sfunction en Matlab Simulink y comprobación del correcto funcionamiento de este mediante la generación de gráficas a partir de los datos obtenidos (dos meses).
5. Redacción de la memoria (quince días).

3. ESTADO DE LA TÉCNICA

En este apartado veremos las diferentes opciones existentes para estimar los parámetros de un PMSM, pudiéndolos diferenciar en dos categorías principales: estimación mediante inyección de señal y estimación sin inyección de señal.

3.1. Métodos con inyección de señal

Este es el método de inyección de un rizado de corriente [1]. Gracias a esta inyección de corriente y a las ecuaciones de un motor PMSM (1), podremos obtener los parámetros del motor.

$$\begin{aligned} V_d &= R i_d + L_d \frac{d i_d}{d t} - \omega L_q i_q \\ V_q &= R i_q + L_q \frac{d i_q}{d t} + \omega (\phi_m + L_d i_d) \end{aligned} \tag{1}$$

Debido a que solo tenemos dos ecuaciones, este método, al igual que los analizados posteriormente, nos permitirá estimar dos de los cuatro parámetros del motor: o resistencia y flujo o bien las inductancias L_d y L_q . Estas combinaciones se deben a que L_d y L_q se deben estimar de modo conjunto ya que para los controles MTPA y MTPV se necesita saber su diferencia, por lo cual la estimación por separado de estas sería más bien contraproducente. La principal ventaja de este método es su rápida convergencia, que, al ser una medida objetiva, el tiempo de obtención de los parámetros es cuasi nulo. Pero tiene desventajas, por ejemplo, la necesidad de utilizar un hardware adicional, que aumentaría los costes y reduciría la fiabilidad. Y, al medir rizados de corriente, también es necesaria una gran robustez del sistema ante ruidos externos al mismo. Por no hablar de la complicación que eso supone, puesto que estos rizados deben de ser pequeños para que no afecten al control.

3.2. Métodos sin inyección de señal

Como su nombre indica, no se inyecta ningún tipo de señal al sistema, sino que los parámetros se estiman mediante la medida de las señales del motor. La principal ventaja de estos métodos es que son mucho más baratos ya que no hay que añadir hardware adicional, por lo que normalmente se pueden implementar en plataformas existentes. A continuación, veremos algunos de los métodos utilizados como estimadores y finalmente haremos una comparativa y selección del método a usar.

3.2.1. Genetic Algorithm

Este método [2] es un método estocástico, lo cual quiere decir que los cambios en las variables son aleatorios, pero con ciertas condiciones. Como se ve en el esquema, Fig. 2, primero en este método se inicializan las variables a un valor, y se mandan al control. Posteriormente, se ejecuta el control y se evalúan los resultados de este. Por último, se valora si los resultados obtenidos son los óptimos o no. En caso de que sean los valores óptimos, se finalizará el algoritmo y serán los valores utilizados. En el caso de que no lo sean, se compararán los valores con los anteriores y se almacenarán los mejores valores. Una vez obtenidos los mejores valores, se realizará una variación aleatoria de los mismos y se ejecutará otra vez la prueba yendo al punto de enviar las variables de nuevo al control.

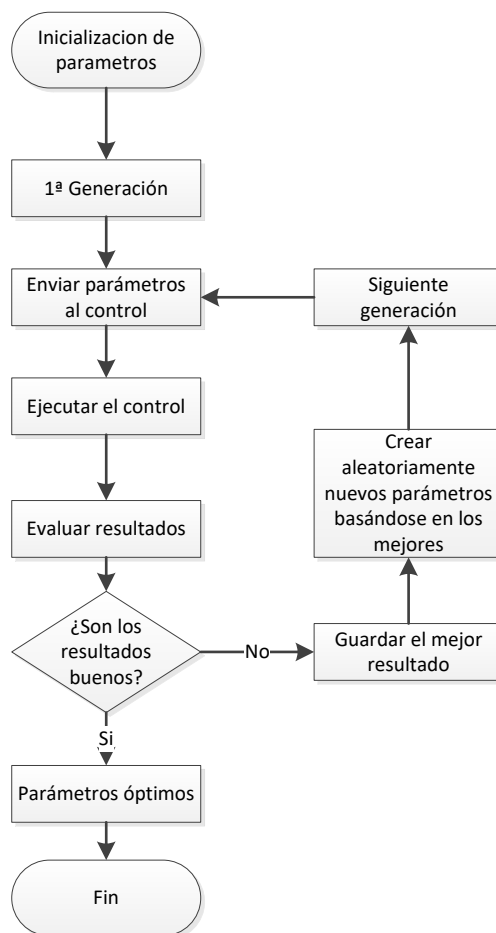


Fig. 2 Genetic Algorithm

Este método tiene un inconveniente para nuestras necesidades y es que no funciona en un sistema en tiempo real al tener que realizar pruebas constantemente y no estimar en tiempo real. La aplicación correcta de este método sería para sintonizar los parámetros de un controlador. Por este motivo, este método queda descartado.

3.2.2. Kalman Filter (KF)

El funcionamiento de este método es el siguiente [3]: aplicamos una misma entrada a nuestro sistema real y al modelo en el cual tenemos las estimaciones. Después comparamos las salidas de estos sistemas y, en el caso de no ser las salidas iguales, aplicaremos una corrección a los parámetros que deseamos estimar de nuestro modelo, Fig. 3.

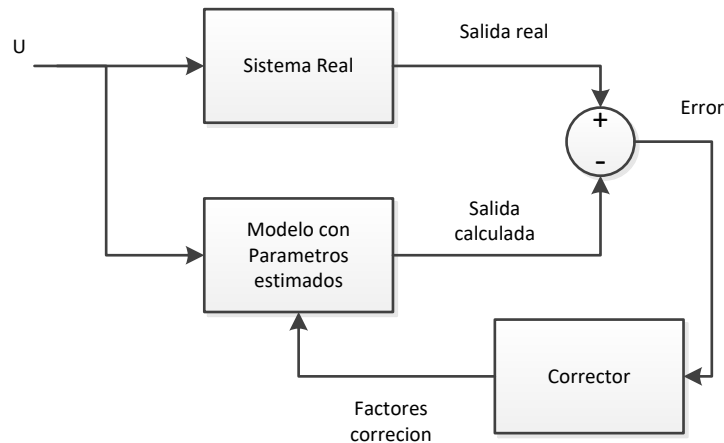


Fig. 3 Filtro de Kalman

Este método conlleva un gran coste computacional [4], además de tener un tiempo de convergencia similar a otros métodos. Por lo tanto, en caso de no existir otro método que pueda solventar estos inconvenientes, sería el método de elección.

3.2.3. Model Reference Adaptive System (MRAS)

Este método [5] se basa en construir dos sistemas de ecuaciones diferentes cuyo resultado, teóricamente, debe ser igual. Las variables por estimar aparecerán sólo en uno de los dos sistemas de ecuaciones, por lo que un error en la comparación será atribuible a una mala estimación, Fig. 4. En el sistema de referencia usaremos la ecuación o sistema de ecuaciones en las que no aparezcan las variables a estimar. Mientras que en el otro sistema o modelo ajustable usaremos la ecuación o sistema de ecuaciones en las que aparezcan los parámetros a estimar. Posteriormente restaremos las salidas de ambas para obtener un error, el cual pasaremos por un corrector y ajustaremos así los parámetros a estimar hasta que el error se minimice.

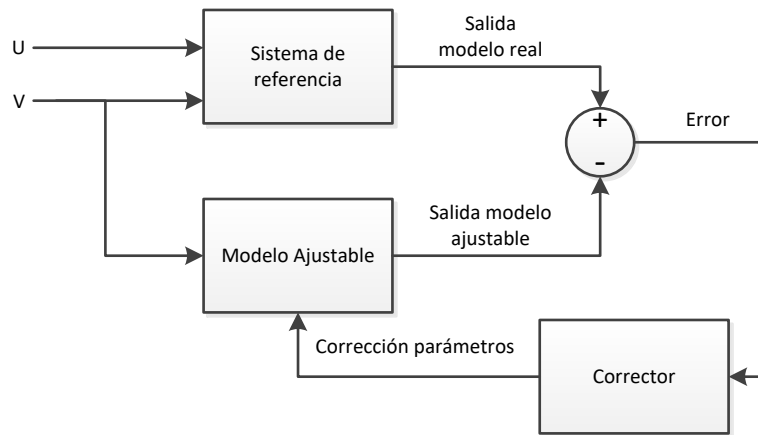


Fig. 4 MRAS

Para nuestra aplicación este método sería válido debido a que se pueden extraer las ecuaciones desde dos puntos.

3.2.4. Affine Projection

Este método [6] es muy similar al anterior, de hecho, su estructura, Fig. 5, es idéntica al método MRAS. Lo que los diferencia es que, en este, en vez de aplicar un corrector cualquiera para el error, se usa el método Affine projection, el cual se basa en la corrección mediante la minimización del error cuadrático.

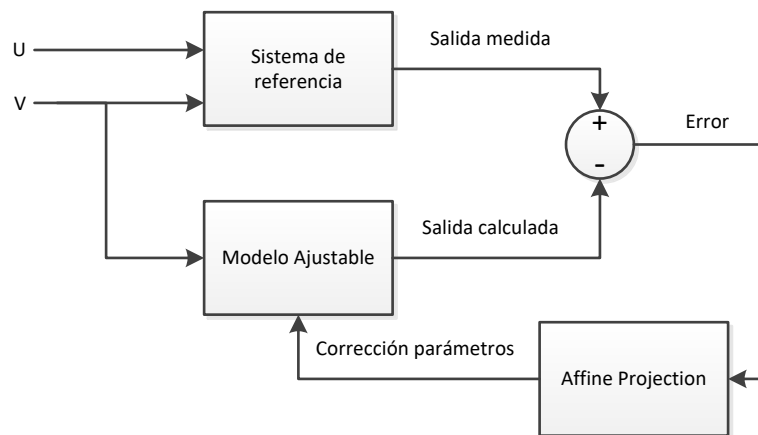


Fig. 5 Affine Projection

Este método nos resulta idóneo, pues su coste computacional es muy bajo. Además, el tiempo de convergencia es reducido, siendo el más efectivo para nuestra aplicación y cualquier aplicación en tiempo real.

4. ESTRATEGIA DE ESTIMACIÓN ELEGIDA

4.1. Explicación del algoritmo

El algoritmo en el que basamos nuestro estimador es el “Affine projection” [7] .Este está basado en un método de mínimos cuadrados. A continuación, vamos a desarrollar el funcionamiento de este partiendo del esquema, Fig. 6,hasta obtener las ecuaciones.

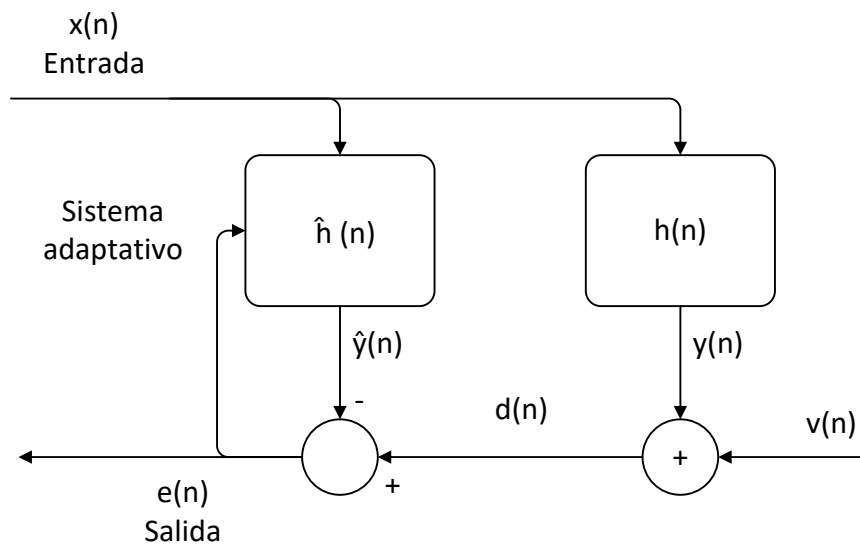


Fig. 6 Affine Projection

Este esquema es en el que se basa nuestro estimador. Para nuestra aplicación requeriríamos de dos estimadores como este, uno para las inductancias con una convergencia más rápida, debido a que las inductancias cambian con las corrientes, y uno más lento para la resistencia y el flujo, debido a que estos cambian con la temperatura, que tiene una dinámica más lenta.

A continuación, explicaremos el funcionamiento del “Affine projection”. Previamente, analizaremos cómo funciona el método de mínimos cuadrados.

En el método de mínimos cuadrados, o LMS, lo que queremos conseguir es optimizar la función de costes que está basada en el error cuadrático (2).

$$C(n) = E\{|e(n)|^2\} \quad (2)$$

Donde $C(n)$ es la función de costes, $e(n)$ es el error y $E\{\cdot\}$ es la función de costes del error cuadrático. Para minimizar la función de costes realizamos una derivada parcial respecto al parámetro a optimizar del filtro (5).

$$\nabla_{\hat{h}} C(n) = \nabla_{\hat{h}} E\{|e(n)|^2\} = 2E\{e(n)\nabla_{\hat{h}}(e(n))\} \quad (3)$$

$$\nabla_{\hat{h}} e(n) = \nabla_{\hat{h}} (d(n) - x(n)\hat{h}(n)) = -x(n) \quad (4)$$

$$\nabla_{\hat{h}} C(n) = -2E\{x(n)e(n)\} \quad (5)$$

Una vez tenemos la función de costes lista para encontrar el mínimo, en el siguiente paso de cómputo tenemos que aumentar o disminuir en la dirección contraria a la derivada $\nabla C(n)$ (6), lo que expresamos de la siguiente forma, siendo $\frac{\mu}{2}$ una constante de adaptación del paso.

$$\hat{h}(n+1) = \hat{h}(n) - \frac{\mu}{2} \nabla_{\hat{h}} C(n) = \hat{h} + \frac{\mu}{2} 2E\{x(n)e(n)\} = \hat{h}(n) + \mu E\{x(n)e(n)\} \quad (6)$$

Este sería el algoritmo LMS, pero para que sea funcional, necesitamos aclarar la función de costes $E\{\cdot\}$. Con esta finalidad, vamos a considerar una función típica (7), donde la N es el número de muestras tomadas para realizar la estimación.

$$E\{x(n)e(n)\} = \frac{1}{N} \sum_{i=0}^{N-1} x(n-i)e(n-i) \quad (7)$$

En nuestro caso usaremos una muestra, por lo tanto, la función quedara de la siguiente forma (8).

$$E\{x(n)e(n)\} = x(n)e(n) \quad (8)$$

Quedando así el algoritmo LMS de una forma mucho más simple (9).

$$\hat{h}(n+1) = \hat{h}(n) + \mu x(n)e(n) \quad (9)$$

Este es el método de estimación en el que nos basamos para desarrollar el nuestro, al cual le añadiremos varias mejoras. La primera de ellas es la normalización de la entrada, hecho que lo convierte en válido para cambios del rango de trabajo. Con esta mejora el algoritmo queda de la siguiente forma (10).

$$\hat{h}(n + 1) = \hat{h}(n) + \frac{\mu x(n)e(n)}{x^T(n)x(n)} \quad (10)$$

Por último, para obtener ya nuestro método "Affine projection" añadiremos una sensibilidad a la parte de la normalización que permite ajustarla según queramos que afecte más o menos (11).

$$\hat{h}(n + 1) = \hat{h}(n) + \frac{\mu x(n)e(n)}{\lambda + x^T(n)x(n)} \quad (11)$$

4.2. Ecuaciones de Flujo y Resistencia

Una vez tenemos el algoritmo, primero trataremos el caso de resistencia y flujo. Para este caso, las ecuaciones a utilizar y de las que extraeremos los parámetros son las ecuaciones de la tensión de un motor de imanes permanentes (12).

$$\begin{aligned} V_d &= Ri_d + L_d \frac{di_d}{dt} - \omega L_q i_q \\ V_q &= Ri_q + L_q \frac{di_q}{dt} + \omega(\phi_m + L_d i_d) \end{aligned} \quad (12)$$

Como vamos a trabajar en un sistema discreto, primero adaptaremos las ecuaciones (13) siendo T periodo de ejecución.

$$\begin{aligned} V_d &= Ri_d + L_d \frac{\Delta i_d}{T} - \omega L_q i_q \\ V_q &= Ri_q + L_q \frac{\Delta i_q}{T} + \omega(\phi_m + L_d i_d) \end{aligned} \quad (13)$$

De estas ecuaciones separamos los parámetros a estimar (14)

$$\begin{aligned}
-Ri_d &= L_d \frac{\Delta i_d}{T} - \omega L_q i_q - V_d \\
-Ri_q - \omega \phi_m &= +L_q \frac{\Delta i_q}{T} + \omega L_d i_d - V_q
\end{aligned} \tag{14}$$

Combinando las dos ecuaciones extraemos los dos sistemas requeridos (15) (16) (17) para nuestro algoritmo

$$\hat{y} = \begin{bmatrix} -\hat{R}i_d \\ -\hat{R}i_q - \omega \hat{\phi}_m \end{bmatrix} \tag{15}$$

$$\hat{h} = \begin{bmatrix} \hat{R} \\ \hat{\phi}_m \end{bmatrix} \tag{16}$$

$$h = \begin{bmatrix} R \\ \phi_m \end{bmatrix} \tag{17}$$

Por lo tanto, para cumplir con que $x(n)\hat{h} = \hat{y}$ la matriz $x(n)$ (18) tiene que ser:

$$x(n) = \begin{bmatrix} -i_d & 0 \\ -i_q & -\omega(n) \end{bmatrix} \tag{18}$$

Comprobamos que las ecuaciones concuerdan (19)

$$x(n)\hat{h} = \begin{bmatrix} -i_d & 0 \\ -i_q & -\omega(n) \end{bmatrix} \begin{bmatrix} \hat{R} \\ \hat{\phi}_m \end{bmatrix} = \begin{bmatrix} -\hat{R}i_d \\ -\hat{R}i_q - \omega \hat{\phi}_m \end{bmatrix} = \hat{y} \tag{19}$$

Ya tenemos el sistema de los parámetros a estimar, pero como en este caso h es desconocida, no podemos usar este sistema para calcular $y(n)$ y $d(n)$. Por este motivo, utilizaremos el otro lado de la ecuación de la tensión de un motor PMSM (20), quedándonos así el siguiente vector.

$$d(n) = \begin{bmatrix} L_d \frac{\Delta i_d}{T} - V_d(n) - L_q \omega(n) i_q(n) \\ L_q \frac{\Delta i_q}{T} - V_q(n) + L_d \omega(n) i_d(n) \end{bmatrix} \tag{20}$$

Por último, nos queda calcular el error $e(n)$ (21)

$$\begin{aligned}
e(n) = d(n) - \hat{y}(n) &= \begin{bmatrix} L_d \frac{\Delta i_d}{T} - V_d(n) - L_q \omega(n) i_q(n) \\ L_q \frac{\Delta i_q}{T} - V_q(n) + L_d \omega(n) i_d(n) \end{bmatrix} - \begin{bmatrix} -\hat{R} i_d \\ -\hat{R} i_d - \omega \hat{\phi}_m \end{bmatrix} = \\
&= \begin{bmatrix} L_d \frac{\Delta i_d}{T} - V_d(n) - L_q \omega(n) i_q(n) + \hat{R} i_d \\ L_q \frac{\Delta i_q}{T} - V_q(n) + L_d \omega(n) i_d(n) + \hat{R} i_d + \omega \hat{\phi}_m \end{bmatrix}
\end{aligned} \tag{21}$$

Dicho error tiene la misma forma que las ecuaciones de la tensión del motor si las reordenamos (22)

$$\begin{aligned}
0 &= R i_d + L_d \frac{d i_d}{dt} - \omega L_q i_q - V_d \\
0 &= R i_q + L_q \frac{d i_q}{dt} + \omega (\phi_m + L_d i_d) - V_q
\end{aligned} \tag{22}$$

Con estos cálculos ya hemos conseguido las ecuaciones necesarias para estimar tanto resistencia como flujo.

4.3. Ecuaciones de Inductancias

Teniendo ya el algoritmo y las ecuaciones de Resistencia y Flujo, ya solo nos quedan por hallar las ecuaciones que estiman las inductancias L_d y L_q . Para ello, al igual que en el caso anterior, partiremos de las ecuaciones de la tensión en un motor PMSM (23).

$$\begin{aligned}
V_d &= R i_d + L_d \frac{d i_d}{dt} - \omega L_q i_q \\
V_q &= R i_q + L_q \frac{d i_q}{dt} + \omega (\phi_m + L_d i_d)
\end{aligned} \tag{23}$$

Al igual que en el caso anterior, como el sistema es discreto, pasaremos la ecuación a una ecuación discreta con periodo T (24).

$$\begin{aligned}
V_d &= R i_d + L_d \frac{\Delta i_d}{T} - \omega L_q i_q \\
V_q &= R i_q + L_q \frac{\Delta i_q}{T} + \omega (\phi_m + L_d i_d)
\end{aligned} \tag{24}$$

A continuación, despejamos de cada ecuación los parámetros a estimar L_d y L_q (25).

$$\begin{aligned} -L_d \frac{\Delta i_d}{T} + \omega L_q i_q &= R i_d - V_d \\ -L_q \frac{\Delta i_q}{T} - \omega L_d i_d &= R i_q + \omega \phi_m - V_q \end{aligned} \quad (25)$$

De esta ecuación podemos extraer \hat{y} (26) y \hat{h} (27).

$$\hat{y} = \begin{bmatrix} -L_d \frac{\Delta i_d}{T} + \omega L_q i_q \\ -L_q \frac{\Delta i_q}{T} - \omega L_d i_d \end{bmatrix} \quad (26)$$

$$\hat{h} = \begin{bmatrix} \widehat{L}_d \\ \widehat{L}_q \end{bmatrix} \quad (27)$$

Con estas matrices para cumplir que $x(n)\hat{h} = \hat{y}$ la matriz $x(n)$ tiene que ser (28):

$$x(n) = \begin{bmatrix} -\frac{\Delta i_d}{T} & \omega i_q \\ -\omega i_d & -\frac{\Delta i_q}{T} \end{bmatrix} \quad (28)$$

Comprobamos que la ecuación sea correcta (29).

$$x(n)\hat{h} = \begin{bmatrix} -\frac{\Delta i_d}{T} & \omega i_q \\ -\omega i_d & -\frac{\Delta i_q}{T} \end{bmatrix} \begin{bmatrix} \widehat{L}_d \\ \widehat{L}_q \end{bmatrix} = \begin{bmatrix} -\widehat{L}_d \frac{\Delta i_d}{T} + \omega \widehat{L}_q i_q \\ -\widehat{L}_q \frac{\Delta i_q}{T} - \omega \widehat{L}_d i_d \end{bmatrix} = \hat{y} \quad (29)$$

Con estas matrices ya tenemos resuelto el lado izquierdo de la ecuación. A continuación, nos queda el lado derecho, que refleja los parámetros que podemos medir o conocemos de nuestro sistema, que a su vez viene dado por la matriz $d(n)$ (30).

$$d(n) = \begin{bmatrix} R i_d - V_d \\ R i_q + \omega \phi_m - V_q \end{bmatrix} \quad (30)$$

Una vez tenemos estas matrices y vectores, ya podemos calcular el error (31).

$$\begin{aligned}
e(n) = d(n) - \hat{y}(n) &= \begin{bmatrix} Ri_d - V_d \\ Ri_q + \omega\phi_m - V_q \end{bmatrix} - \begin{bmatrix} -\widehat{L}_d \frac{\Delta i_d}{T} + \omega\widehat{L}_q i_q \\ -\widehat{L}_q \frac{\Delta i_q}{T} - \omega\widehat{L}_d i_d \end{bmatrix} = \\
&= \begin{bmatrix} Ri_d - V_d + \widehat{L}_d \frac{\Delta i_d}{T} - \omega\widehat{L}_q i_q \\ Ri_q + \omega\phi_m - V_q + \widehat{L}_q \frac{\Delta i_q}{T} + \omega\widehat{L}_d i_d \end{bmatrix}
\end{aligned} \tag{31}$$

Y al igual que en el caso anterior, si nos fijamos, la ecuación del error es idéntica a la de la tensión del motor (32).

$$\begin{aligned}
Ri_d - V_d + \widehat{L}_d \frac{\Delta i_d}{T} - \omega\widehat{L}_q i_q &= 0 \\
Ri_q + \omega\phi_m - V_q + \widehat{L}_q \frac{\Delta i_q}{T} + \omega\widehat{L}_d i_d &= 0
\end{aligned} \tag{32}$$

Con estos cálculos, ya tenemos también todo lo necesario para estimar las inductancias.

5. SIMULACIÓN

Para comprobar que nuestro estimador funciona, lo simularemos en Matlab, más concretamente en un modelo de motor PMSM implementado en Simulink. Los estimadores estarán implementados en una sFunction. Para realizar esta simulación, utilizaremos los datos de un motor EPS (Electronic Power Steering) utilizado en dirección asistida de vehículos. Los datos nominales de este motor y los usados para estos experimentos son los siguientes, Tabla 1:

R	50.25 mΩ
ϕ	4.7mWb
L_d	60 μH
L_q	96 μH
p	4
Par_{max}	3 Nm
I_{smax}	$40\sqrt{2}$ A =56.56A
V_{bus}	12V
f_{sw}	20kHz
ω_{max}	15000 r.p.m.
$\omega_{experimentos}$	1000 r.p.m.

Tabla 1 Parámetros de simulación

Una vez definidos los parámetros más relevantes empezaremos a probar los estimadores de manera individual. Posteriormente, una vez estos funcionen, realizaremos la prueba combinándolos.

5.1. Estimación R

La primera variable que estimaremos será la R. Para comprobar el correcto funcionamiento, primero usaremos una R del motor fija y posteriormente una R en rampa, debido a que el comportamiento que tiene la R es dependiente de la temperatura. Para ambos experimentos utilizaremos la misma sintonía $\mu=50$ y $\lambda=0.05$.

5.1.1. Valor nominal

La primera prueba tiene como objetivo que nuestro estimador consiga estimar el valor de la resistencia manteniendo esta su valor nominal. Tras la simulación, los resultados obtenidos nos permiten observar que el estimador funciona de manera correcta, Fig. 7. consiguiendo un error menor al 1% y un rizado en la estimación despreciable.

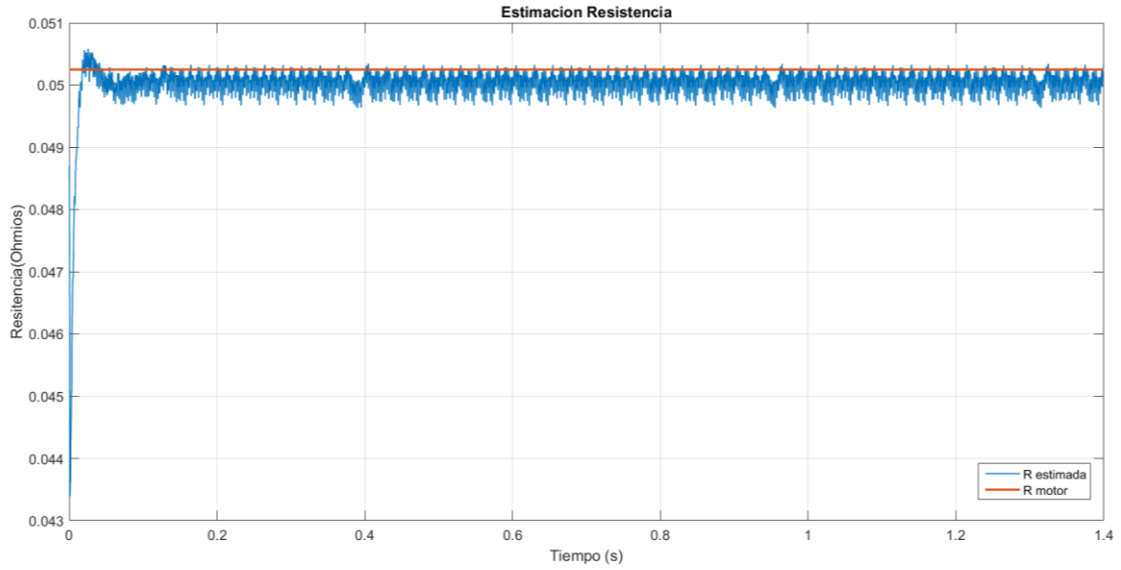


Fig. 7 Estimación de R a valor nominal

5.1.2. Rampa

Manteniendo la misma sintonía, ahora simularemos la tasa de cambio que tendría la R en la realidad. Como esta depende linealmente de la temperatura de manera ascendente, realizaremos el experimento de tal forma que la R sea una rampa. Una vez simulado, observamos, Fig. 8, que el estimador sigue a la perfección la referencia, obteniendo un error casi nulo $<1\%$ y un rizado en la estimación despreciable.

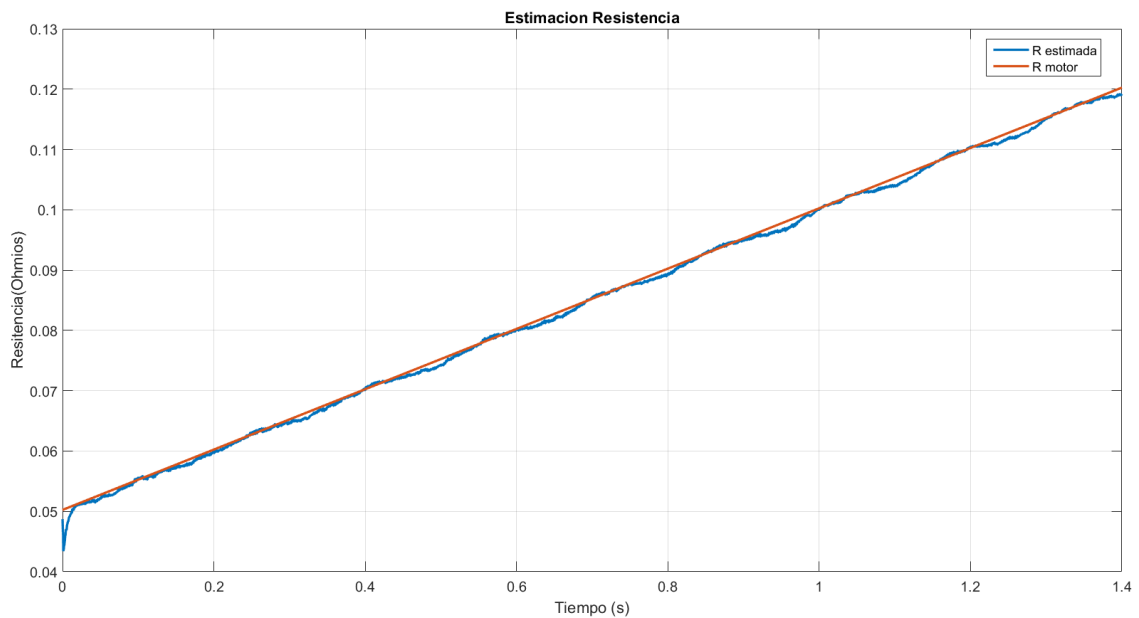


Fig. 8 Estimación de R con tasa de variación

5.2. Estimación Flujo

El siguiente parámetro por estimar es el flujo, ya que así, una vez en funcionamiento, lo probaremos en combinación con la R. Al igual que en la resistencia, consideramos que el flujo varía con la temperatura. Así que realizaremos los experimentos fijos a valor nominal y con una tasa de variación lineal. La sintonía de este estimador es $\mu=30$ y $\lambda=30$.

5.2.1. Valor nominal

El primer experimento que realizaremos para la estimación del flujo es con un valor constante de flujo, valor nominal. Una vez obtenidos los resultados, Fig. 9, observamos que hay error de sobrepasamiento inferior al 2%, pero, pese a ello, es un valor aceptable de error. Además, el rizado en la estimación es casi nulo, siendo estos resultados unos valores más que aceptables.

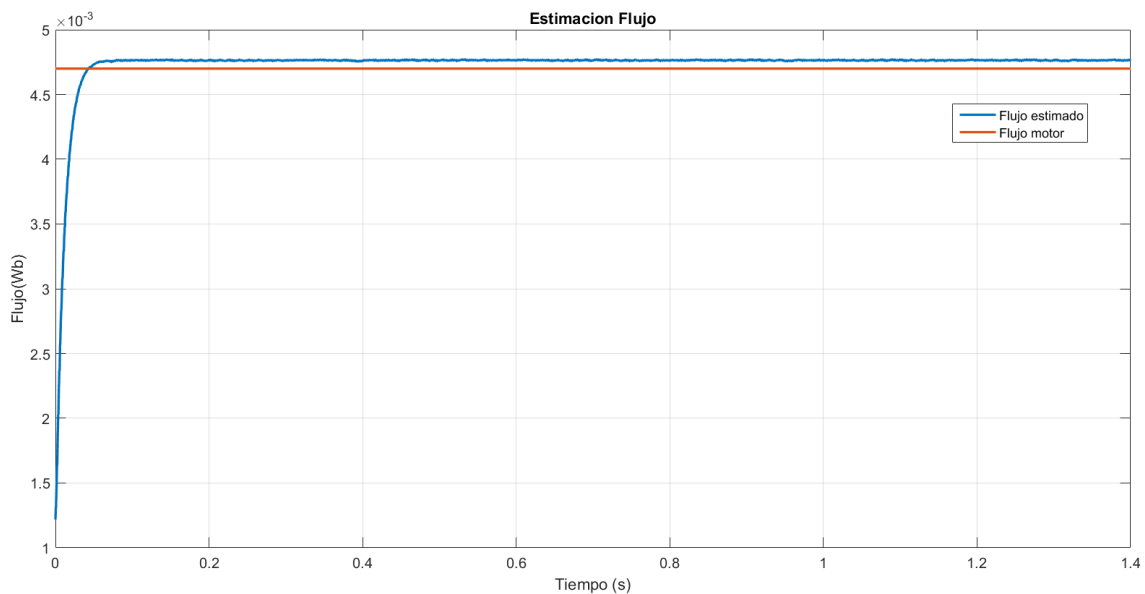


Fig. 9 Estimación de Flujo con valor nominal

5.2.2. Rampa

Una vez hemos comprobado que el estimador funciona y va al valor nominal fijo, probaremos a ver si sigue una referencia cambiante. En este caso, como suponemos que el flujo varía con la temperatura y que la variación con la corriente es despreciable, este aumentará de forma directamente proporcional a la temperatura. Esto nos condiciona a hacer el experimento de tal forma que la tasa de variación del flujo sea lineal. Una vez obtenidos los resultados, Fig. 10, podemos observar que el estimador sigue a la perfección la referencia con un error casi nulo y un rizado despreciable.

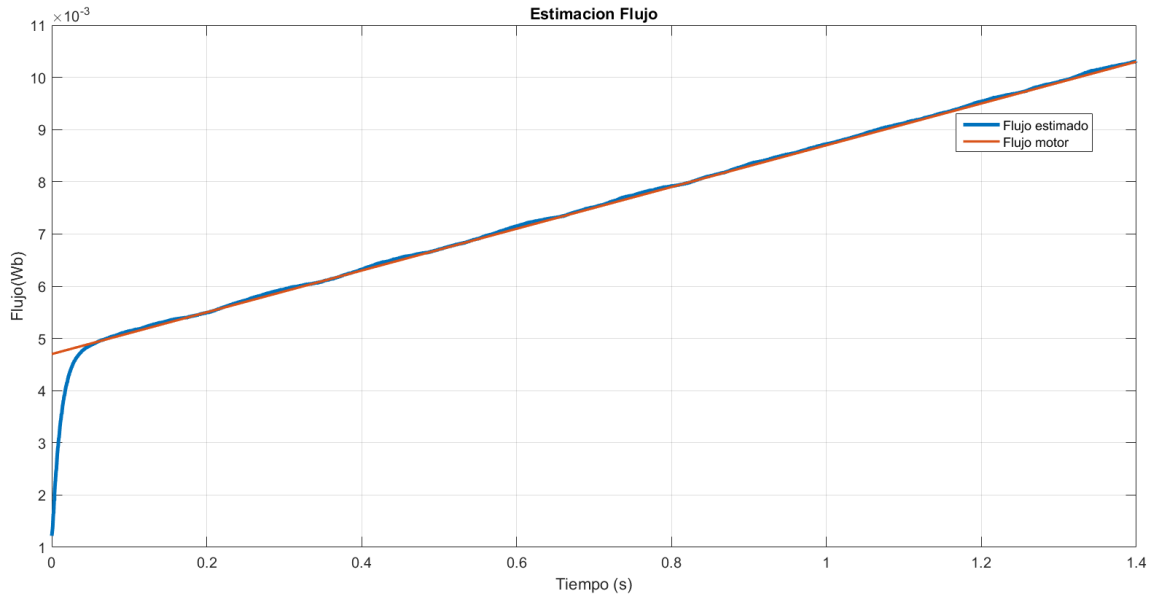


Fig. 10 Estimación de Flujo con tasa de variación

5.3. Estimación R y Flujo

Una vez tenemos los dos estimadores en funcionamiento, procedemos a probarlos en cruzado. Es decir, funcionando los dos a la vez.

5.3.1. Valor nominal R y Flujo

Primero, al igual que cuando los hemos probado por separado, haremos que los parámetros del motor sean fijos a su valor nominal. Para este experimento se ha mantenido la misma sintonía que en los apartados anteriores $\mu_R=50$, $\lambda_R=0.05$, $\mu_\phi=30$ y $\lambda_\phi=30$. Una vez realizado el experimento, Fig. 11 y Fig. 12, observamos que los errores han aumentado. Esto se puede deber a que cuando se estiman individualmente, en el caso del flujo solo afecta la ecuación del eje q teniendo solo una ecuación del sistema (33), mientras que en el estimador de la resistencia, la R aparece en dos ecuaciones, consiguiendo igualmente una única solución (34). El problema al combinarlos es que podemos obtener un error positivo en un eje y uno negativo en el otro, compensándose los términos marcados en (35) y produciéndose así un error de estimación. Otra causa que puede provocar este error en la estimación es el error causado por la discretización a la hora de calcular las derivadas de corriente. En este caso los errores son del 10% para ambos estimadores. Para solucionar esto lo que podríamos hacer es aumentar la frecuencia de muestreo del sistema, reduciendo así el periodo de conmutación y haciendo más preciso el término $\frac{\Delta i_d}{T}$ y $\frac{\Delta i_q}{T}$. Al doblar la frecuencia lo que hemos conseguido es reducir el error a la mitad, Fig. 13 y Fig. 14, esto puede deberse a la influencia del periodo de conmutación el cálculo de las derivadas de corriente $\frac{\Delta i_d}{T}$ y $\frac{\Delta i_q}{T}$ haciéndolas mucho más precisas, y en consecuencia en la estimación. En cuanto al rizado, podemos observar que al igual que antes es despreciable.

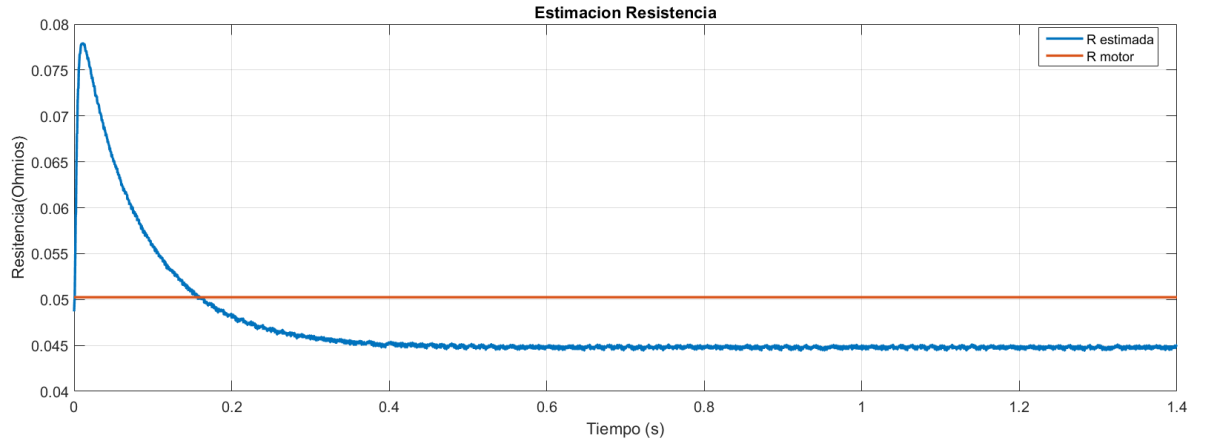


Fig. 11 Estimación de R con R y Flujo del motor constantes

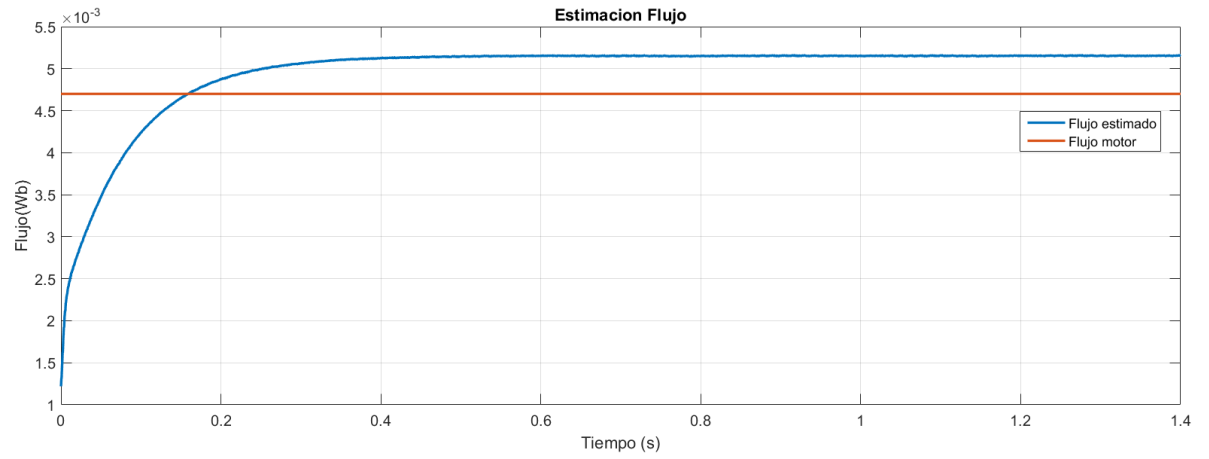


Fig. 12 Estimación de Flujo con R y Flujo del motor constantes

$$e(n) = L_q \frac{\Delta i_q}{T} - V_q(n) + L_d \omega(n) i_d(n) + R i_q + \omega \widehat{\Phi}_m \quad (33)$$

$$e(n) = \begin{bmatrix} L_d \frac{\Delta i_d}{T} - V_d(n) - L_q \omega(n) i_q(n) + \widehat{R} i_d \\ L_q \frac{\Delta i_q}{T} - V_q(n) + L_d \omega(n) i_d(n) + \widehat{R} i_q + \omega \widehat{\Phi}_m \end{bmatrix} \quad (34)$$

$$e(n) = \begin{bmatrix} L_d \frac{\Delta i_d}{T} - V_d(n) - L_q \omega(n) i_q(n) + \widehat{R} i_d \\ L_q \frac{\Delta i_q}{T} - V_q(n) + L_d \omega(n) i_d(n) + \widehat{R} i_q + \omega \widehat{\Phi}_m \end{bmatrix} \quad (35)$$

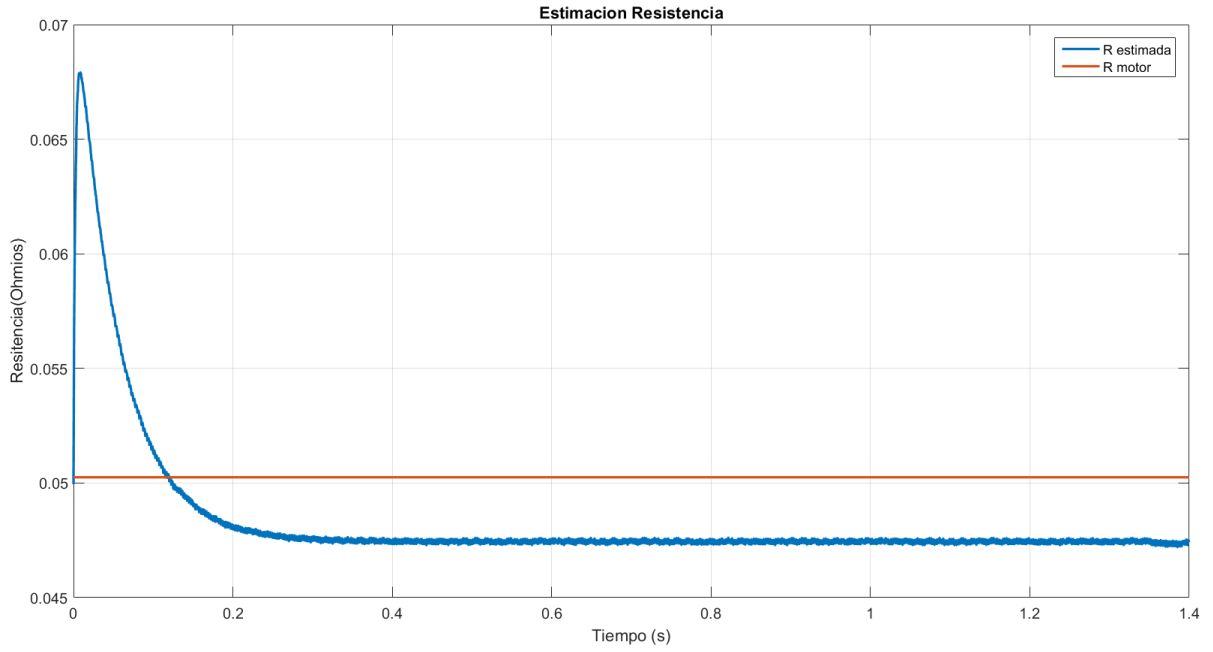


Fig. 13 Estimación de R con R y Flujo del motor constantes y $f_{sw}=40\text{KHz}$

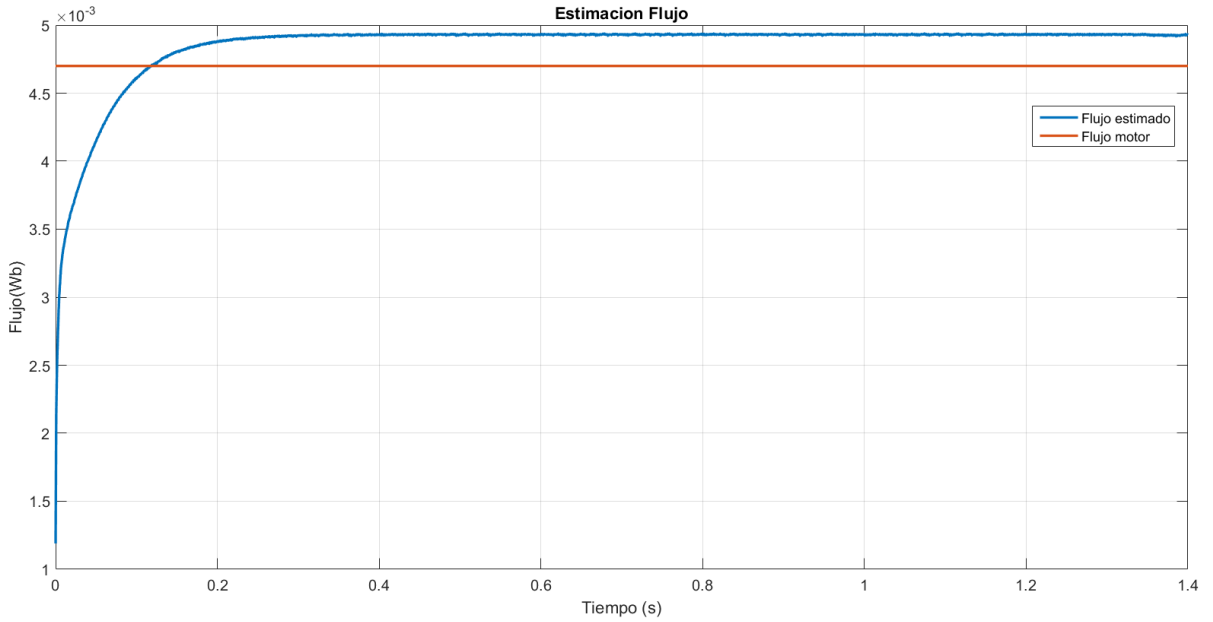


Fig. 14 Estimación de Flujo con R y Flujo del motor constantes y $f_{sw}=40\text{KHz}$

5.3.2. Valor nominal Flujo y rampa en R

Una vez visto que con los valores fijos van a la referencia, comenzaremos a variarlos. Para ello, empezaremos dejando el valor del flujo fijo. Y, como en los apartados anteriores, la tasa de variación de la R será lineal. Una vez realizada la simulación, podemos observar, Fig. 15 y Fig. 16, que, al igual que en el caso anterior, sigue a la señal, pero con un error de estimación del 10%. Las razones son las mismas que en el caso anterior. Sin embargo, el rizado sigue siendo despreciable.

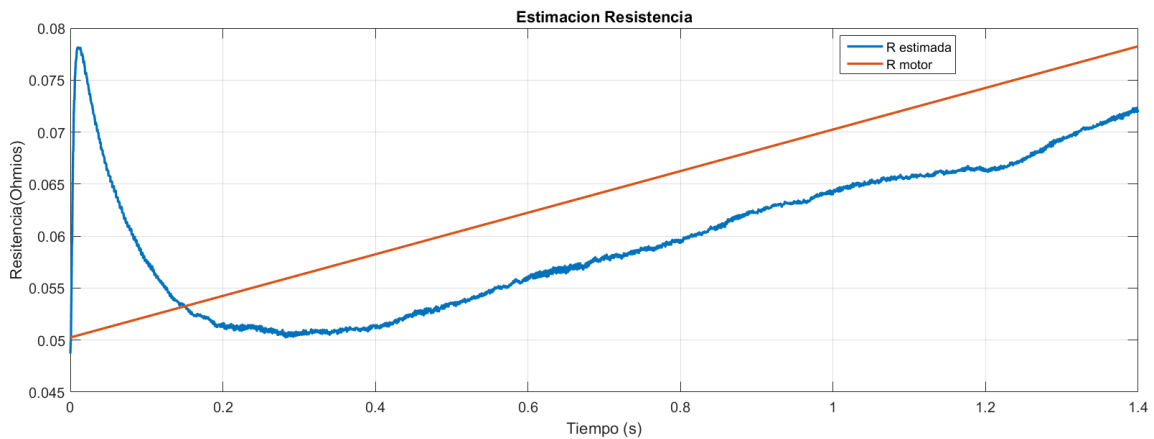


Fig. 15 Estimación de R con R en rampa y Flujo del motor constante

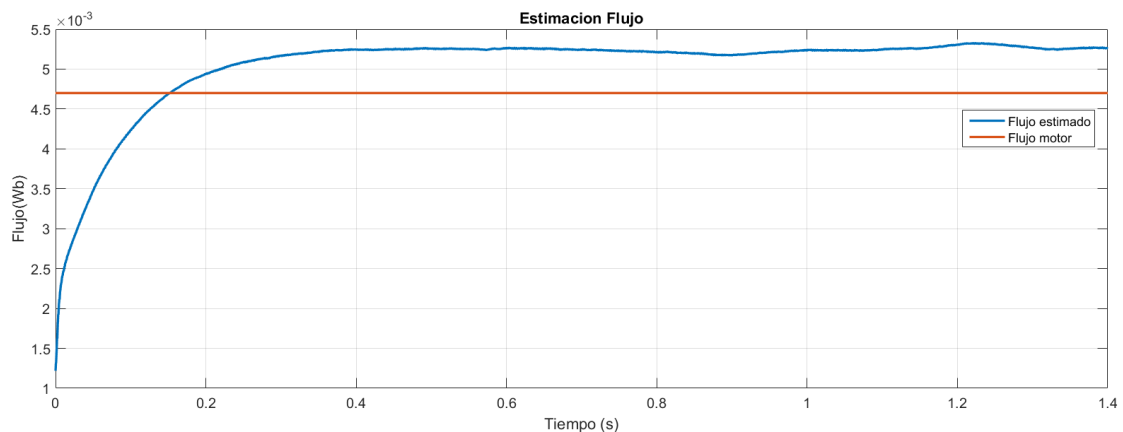


Fig. 16 Estimación de Flujo con R en rampa y Flujo del motor constante

5.3.3. Valor nominal R y rampa en Flujo

Ahora realizaremos la combinación al revés que en el caso anterior. La R será fija mientras que el flujo, como en apartados anteriores, será lineal. Una vez realizado el experimento observamos, Fig. 17 y Fig. 18, que, al igual que en los casos anteriores, existe un error de seguimiento inferior al 10%, por las mismas razones que antes y con las mismas soluciones. En cuanto al rizado de estimación, es despreciable.

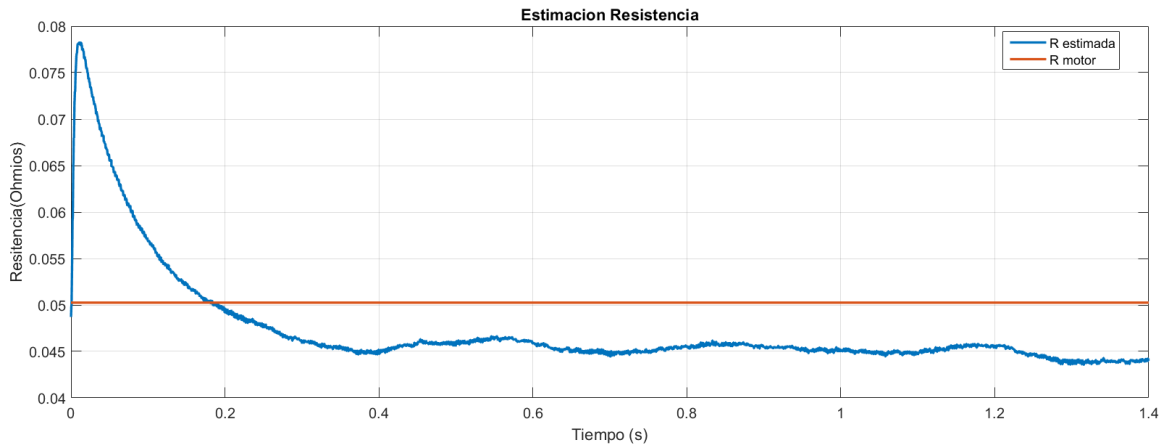


Fig. 17 Estimación de R con Flujo en rampa y R del motor constante

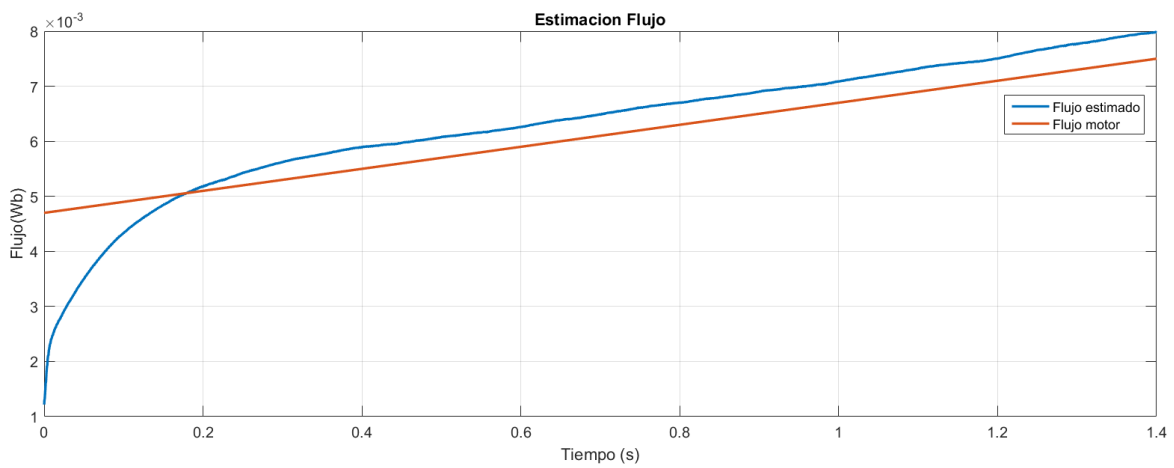


Fig. 18 Estimación de Flujo con Flujo en rampa y R del motor constante

5.3.4. Rampa en R y Flujo

El último experimento de la R y el flujo es contemplar el supuesto de que ambos sigan una referencia con tasa de cambio lineal. Tras realizar el experimento, podemos ver, Fig. 19 y Fig. 20, que ambos estimadores siguen bien a las referencias, con un error del 10%, por las mismas razones y con las mismas soluciones que antes. En cuanto al rizado del estimador, observamos que es despreciable.

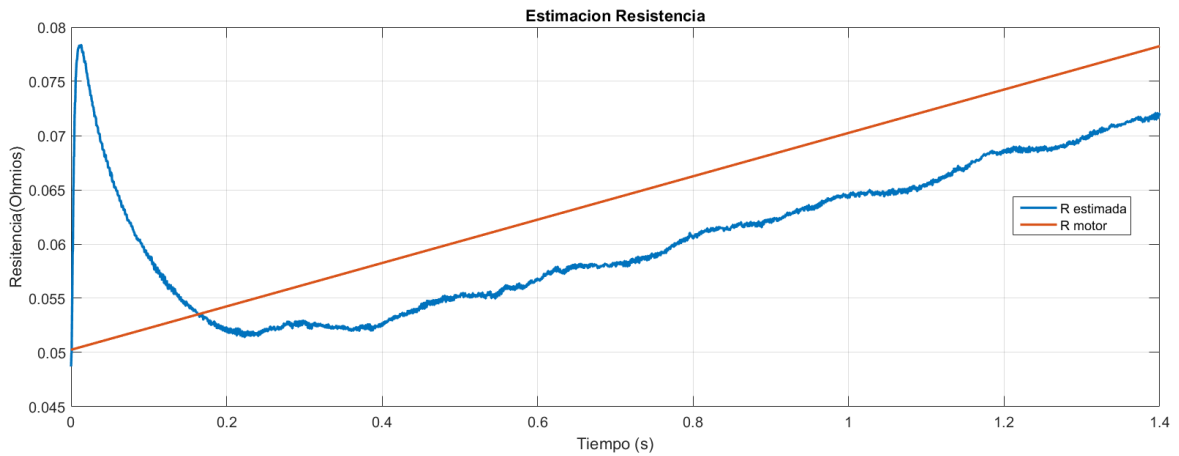


Fig. 19 Estimación de R con R y Flujo en rampa

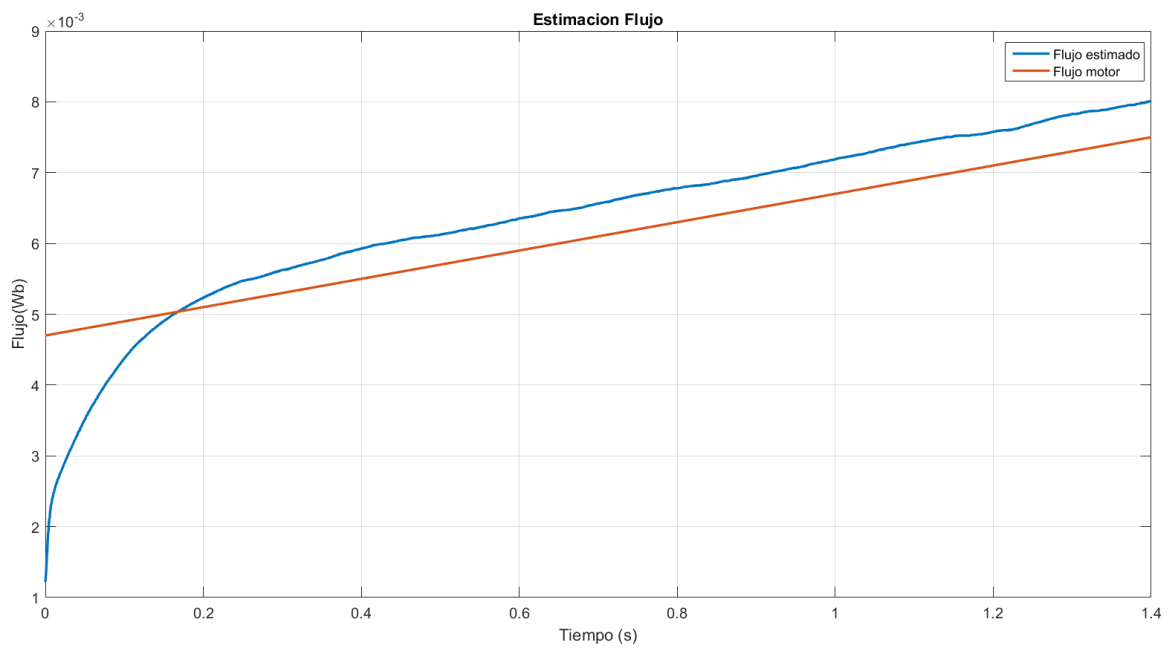


Fig. 20 Estimación de Flujo con R y Flujo en rampa

5.4. Estimación L_d

En este apartado estimaremos de manera individual la inductancia en el eje d. Para simularla, primero usaremos como parámetro su valor nominal constante y posteriormente usaremos una serie de escalones para simular su comportamiento, ya que las inductancias varían con las corrientes y estas tienen unas dinámicas más rápidas, que simularemos como escalones. La sintonía para estos experimentos es de $\mu=0.25$ y $\lambda=10$.

5.4.1. Valor nominal

La primera simulación la realizaremos con una L_d del motor constante a valor nominal. Una vez realizada la simulación, podemos observar, Fig. 21, que el estimador va al valor real, pero tiene un error del 10%. Esto se debe al cálculo de las derivadas de corriente en discreto. Una posible solución sería aumentar la frecuencia del convertidor. En cuanto al rizado de la estimación, observamos que es despreciable <1%.

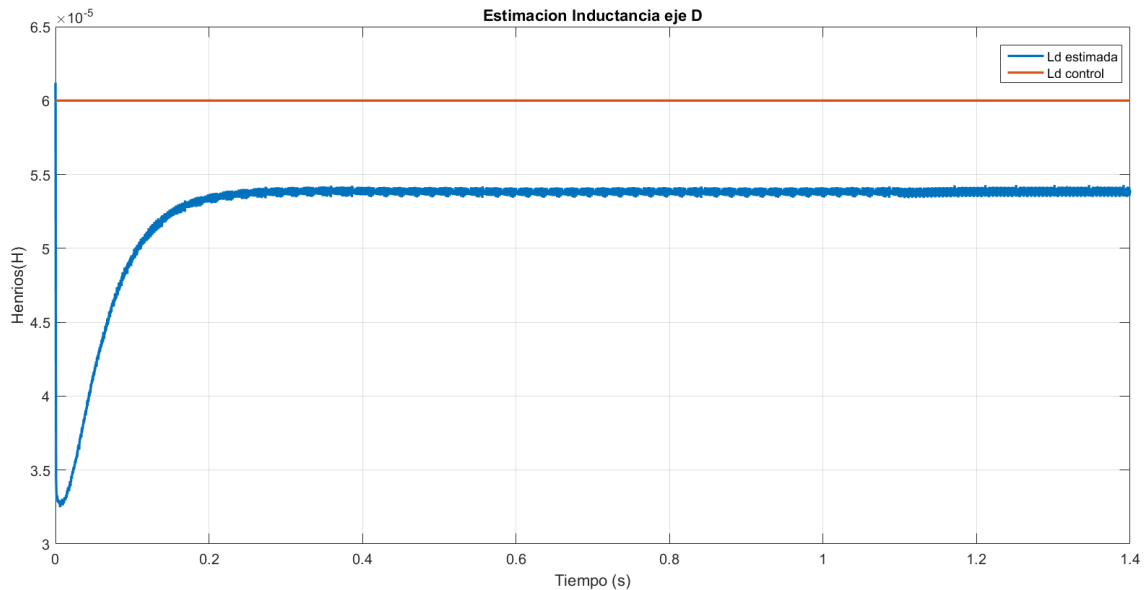


Fig. 21 Estimación de L_d con valor nominal

5.4.2. Escalones

Para comprobar el correcto funcionamiento, pasaremos a observar cómo se comporta ante la variación de los parámetros. Como en un motor las inductancias varían con la corriente y esta tiene una dinámica muy rápida que puede cambiar de forma brusca, simularemos estos cambios con escalones entre los valores en los que podría variar. Tras realizar el experimento observamos, Fig. 22, que, al igual que con el valor nominal, el estimador sigue a la referencia, aunque sigue existiendo ese error del 10%, cuyas causas y soluciones son las mismas que en el caso anterior. En cuanto al rizado de estimación, sigue siendo un rizado despreciable <1%.

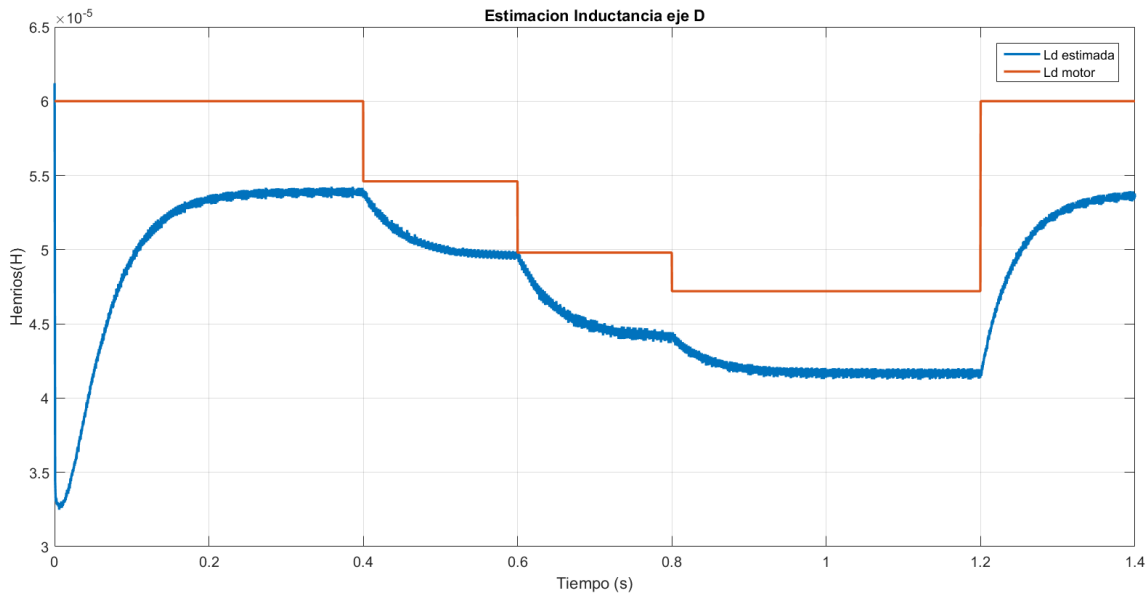


Fig. 22 Estimación de L_d con tasa de variación

5.5. Estimación L_q

A continuación, estudiaremos la estimación de la inductancia en el eje q . Para esta inductancia, igual que para la anterior, primero haremos una simulación con un valor del motor constante y nominal. Posteriormente simularemos con unos escalones, debido a que la inductancia depende de la corriente y esta puede cambiar de forma brusca. La sintonía para este estimador es $\mu=0.25$ y $\lambda=15$.

5.5.1. Valor nominal

Primero comenzaremos por simular el estimador con un valor fijo de inductancia, en este caso el nominal. Tras simularlo, observamos, Fig. 23, cómo sigue a la referencia con un error menor al 5%, siendo este un error más que aceptable. En cuanto al rizado, este es inferior al 1%, por lo que es despreciable

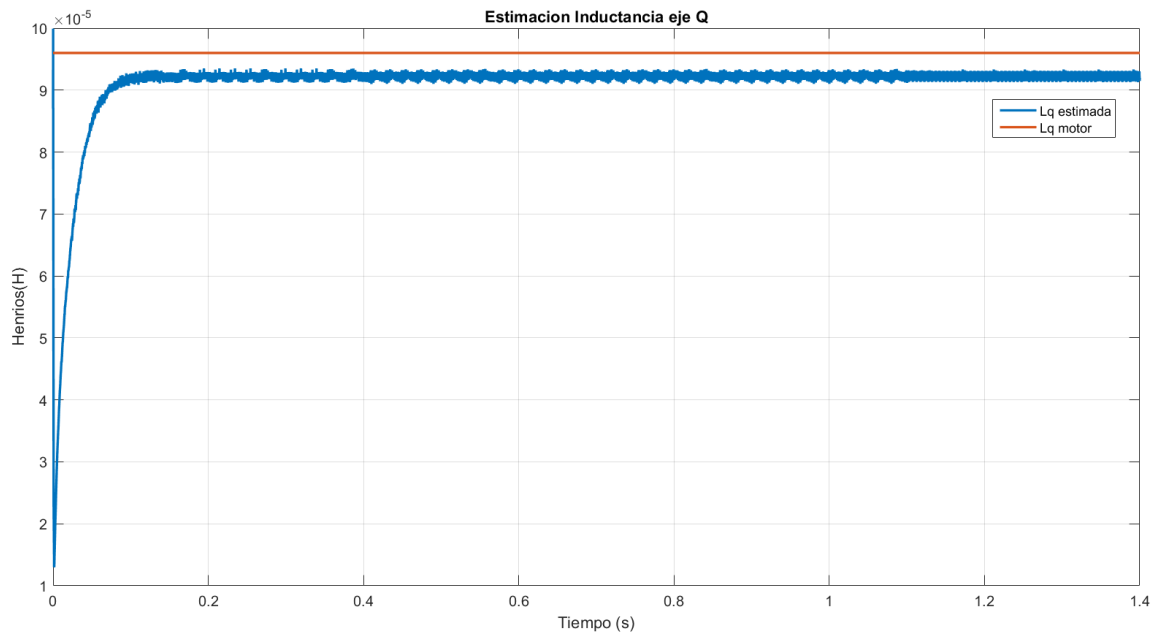


Fig. 23 Estimación de L_q con valor nominal constante

5.5.2. Escalones

Una vez hemos comprobado que el estimador sigue una referencia fija, probaremos a ver cómo se comporta ante variaciones. Para ello, al igual que en el caso anterior, utilizaremos escalones, puesto que la corriente, y en consecuencia la inductancia, pueden cambiar de forma brusca. Una vez simulado, podemos observar, Fig. 24, cómo sigue a los escalones con un error del 5%, igual que en el caso anterior. Y con un rizado de estimación menor al 1%.

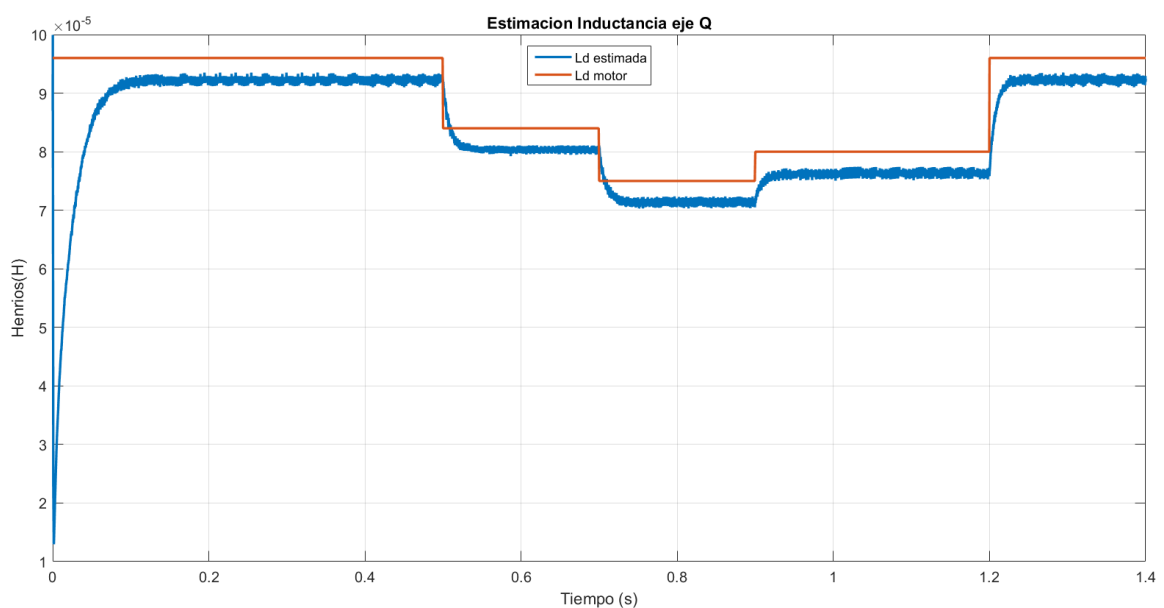


Fig. 24 Estimación de L_q con tasa de variación

5.6. Estimación L_d y L_q

Por último, para acabar de comprobar que los estimadores de inductancias funcionan, los probaremos en cruzado, es decir, funcionando los dos a la vez. Para estas simulaciones, utilizaremos la misma sintonía que en los casos anteriores, $\mu_q=0.25$, $\lambda_q=15$, es $\mu_d=0.25$ y $\lambda_d=10$.

5.6.1. Valor nominal L_d y L_q

Primero, para comprobar que funcionan, empezaremos el experimento con valores nominales y constantes de ambas inductancias. Una vez simulado, podemos observar, Fig. 25 y Fig. 26, cómo ambos estimadores siguen la referencia igual que cuando trabajaban por separado, manteniendo los mismos errores.

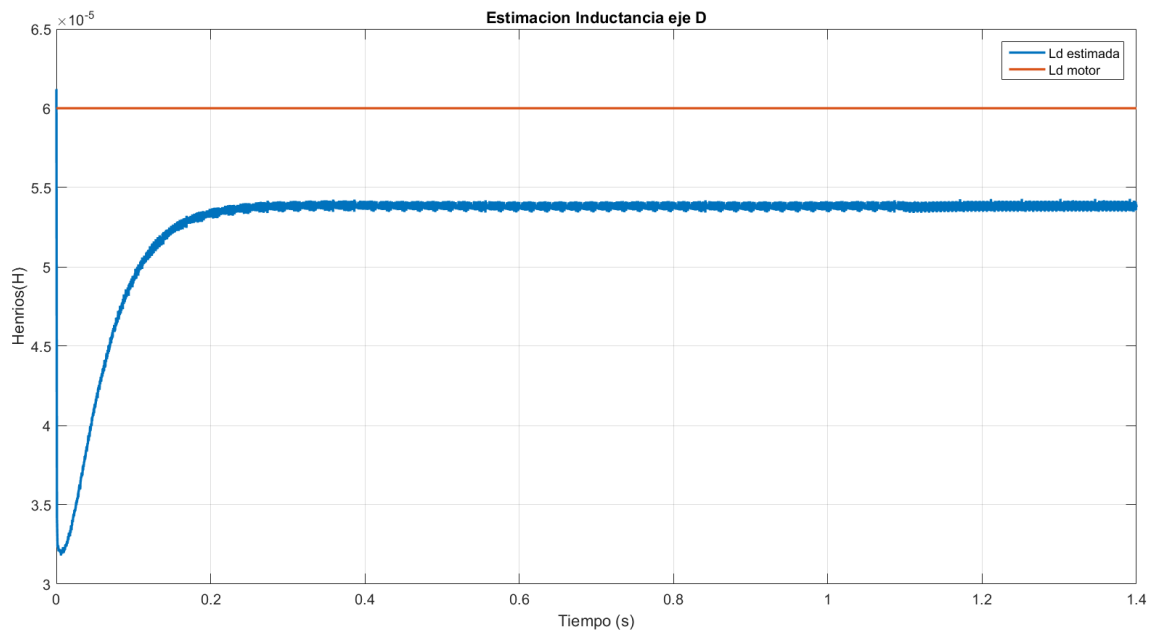


Fig. 25 Estimación de L_d con L_d y L_q con valor nominal constante

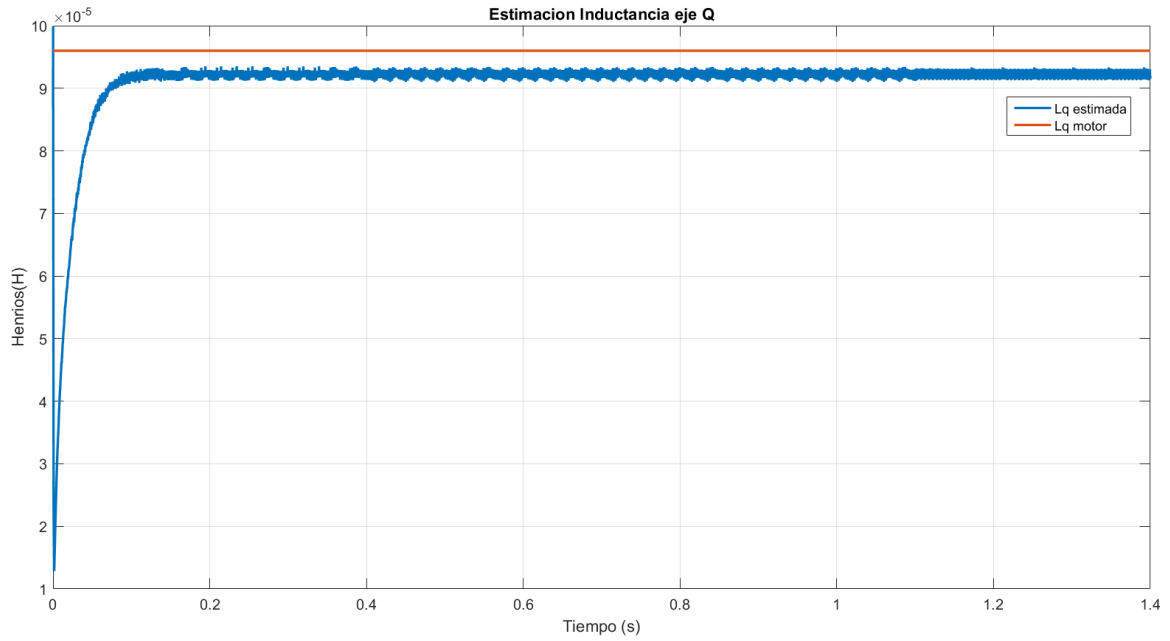


Fig. 26 Estimación de L_q con L_d y L_q con valor nominal constante

5.6.2. Valor nominal L_q y escalón en L_d

Una vez hemos comprobado que los estimadores se comportan de la forma deseada con valores constantes de ambas inductancias, procederemos a introducir variaciones en una de ellas. En este caso, empezaremos con escalones en la inductancia del eje d. Una vez realizada la simulación, podemos ver, Fig. 27 y Fig. 28, cómo ambos estimadores actúan de forma correcta, manteniendo el mismo error que en casos anteriores, 5% para el eje q y 12% para el eje d, además de tener un rizado despreciable <1%.

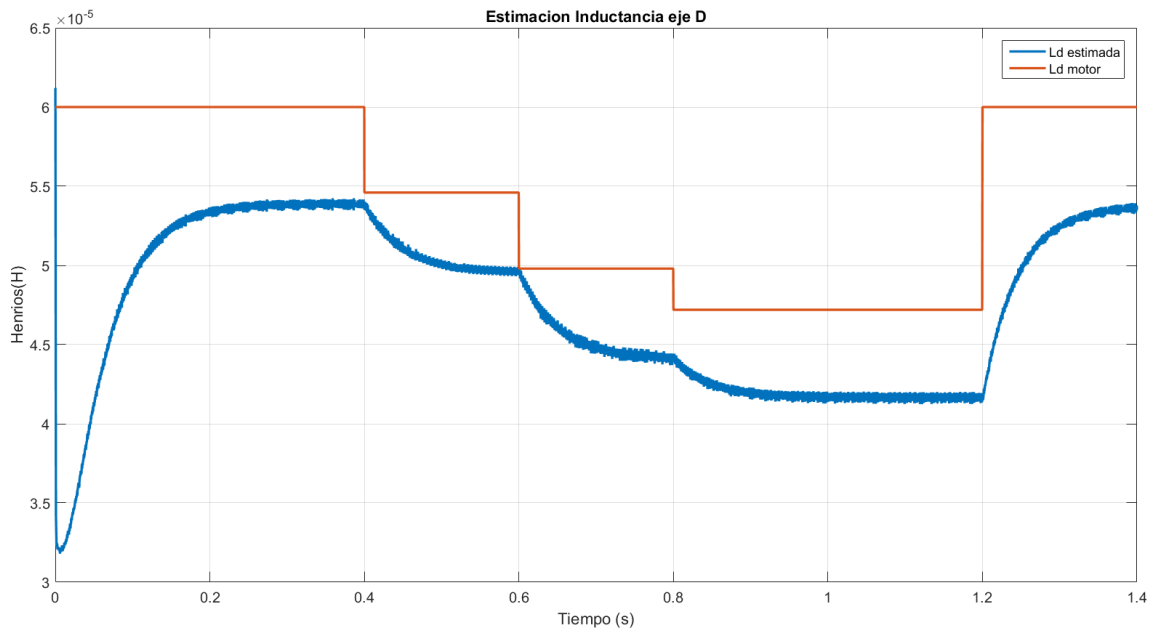


Fig. 27 Estimación de L_d con L_d con escalones y L_q con valor nominal constante

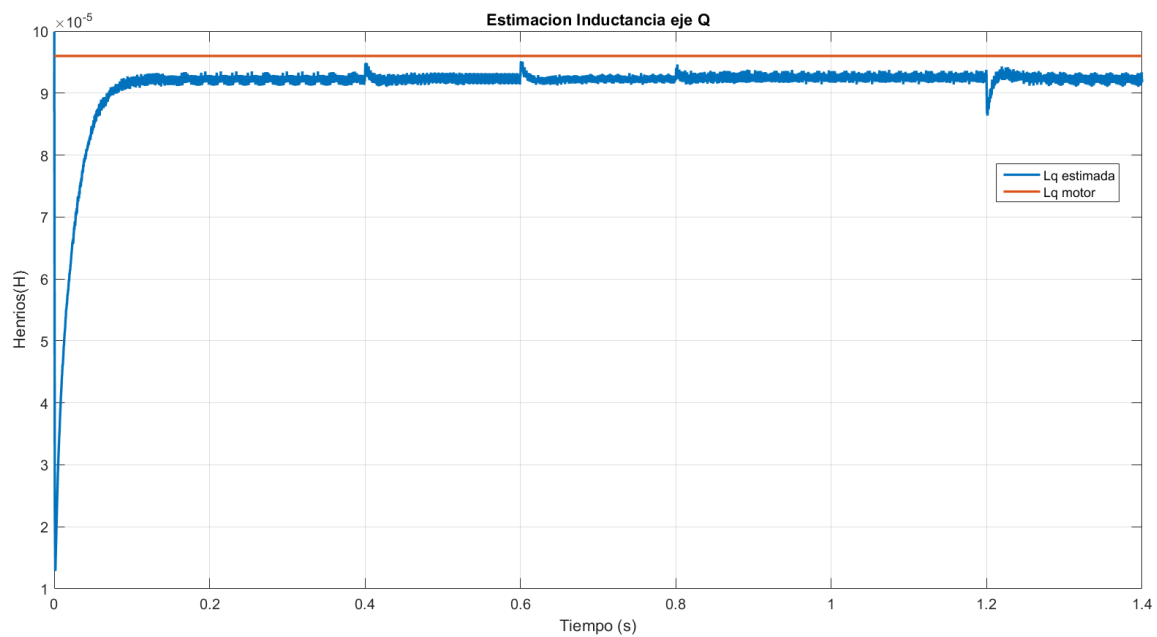


Fig. 28 Estimación de L_q con L_d con escalones y L_q con valor nominal constante

5.6.3. Valor nominal L_d y escalón en L_q

A continuación, realizaremos el mismo experimento, pero esta vez la inductancia del eje d será constante y la del eje q tendrá escalones. Una vez simulada, observamos, Fig. 29 y Fig. 30, que en este caso las variaciones en el eje q afectan más a la estimación del eje d, pero aun así, consideramos los resultados válidos porque este error sigue estando en torno al 10%. En cuanto a la estimación del eje q, no se ve afectada por la variación de

la estimación en el eje d. En cuanto al rizado de la estimación, este sigue siendo despreciable <1%.

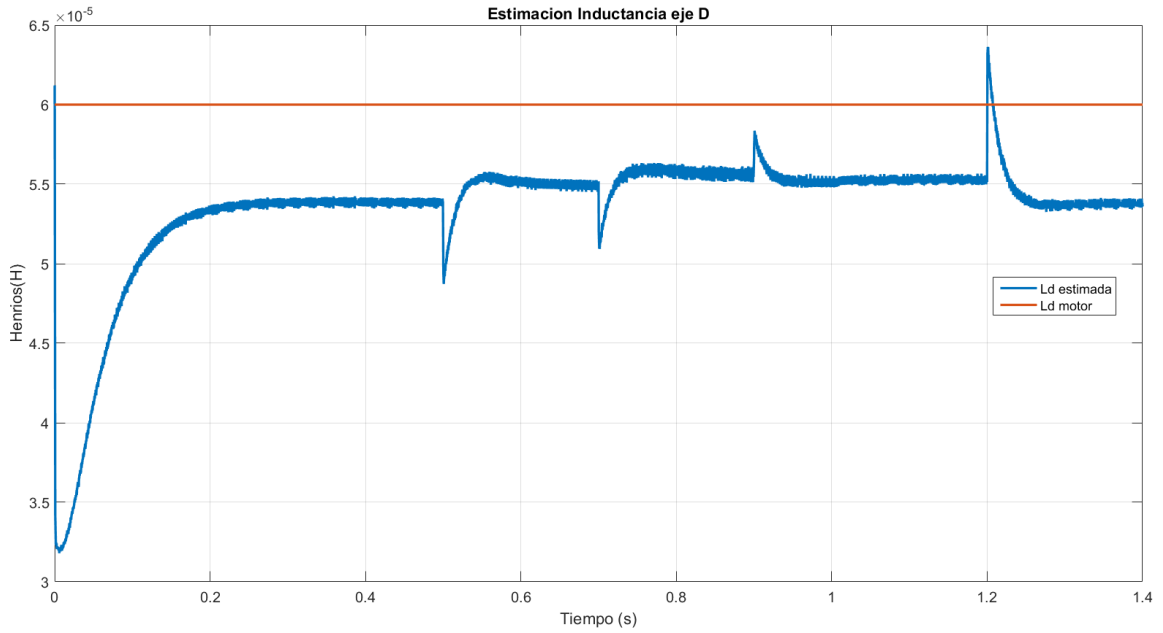


Fig. 29 Estimación de L_d con L_q con variaciones y L_d con valor nominal constante

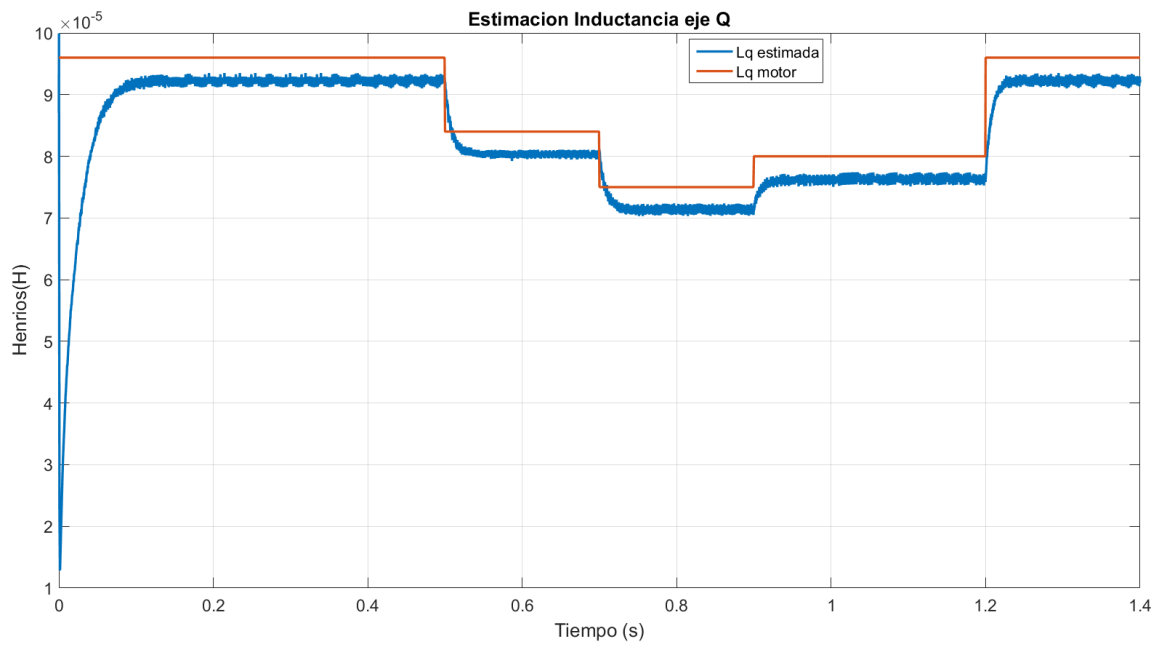


Fig. 30 Estimación de L_q con L_q con variaciones y L_d con valor nominal constante

5.6.4. Escalones en L_d y L_q

Por último, probaremos el funcionamiento teniendo variaciones en forma de escalón en ambas inductancias. Tras realizar el experimento, podemos ver, Fig. 31 y Fig. 32, cómo los dos estimadores llegan al valor deseado con los mismos errores que cuando funcionaban por separado, 10% en el eje d y 5% en el eje q. Así pues, queda demostrado definitivamente que el estimador funciona estimando las dos variables a la vez.

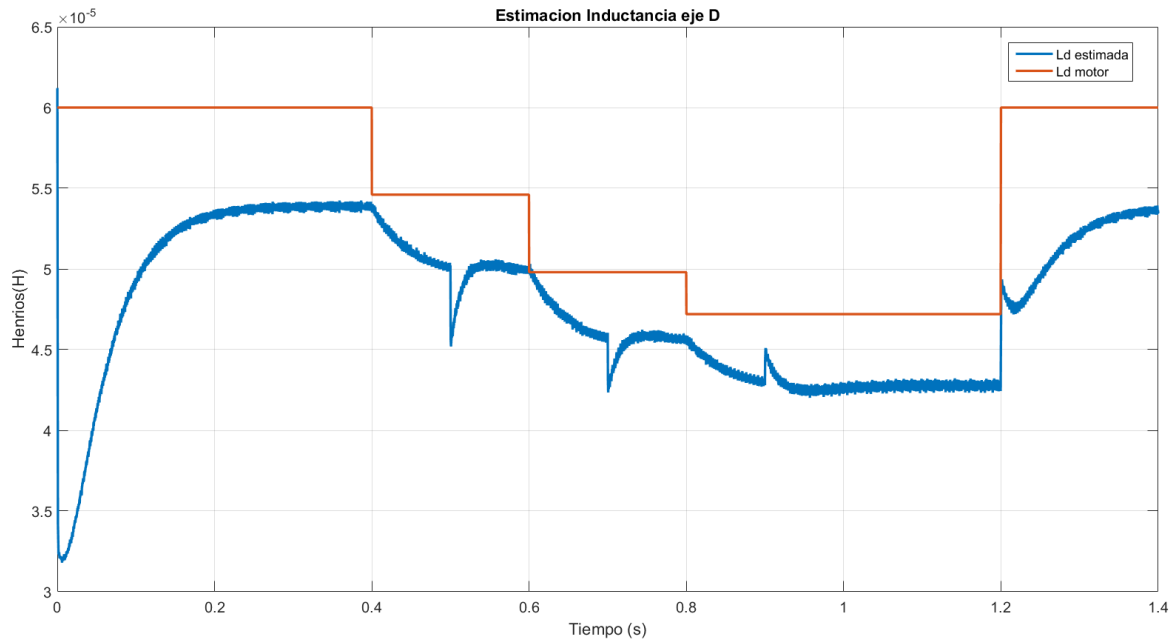


Fig. 31 Estimación de L_d con L_q y L_d con variaciones

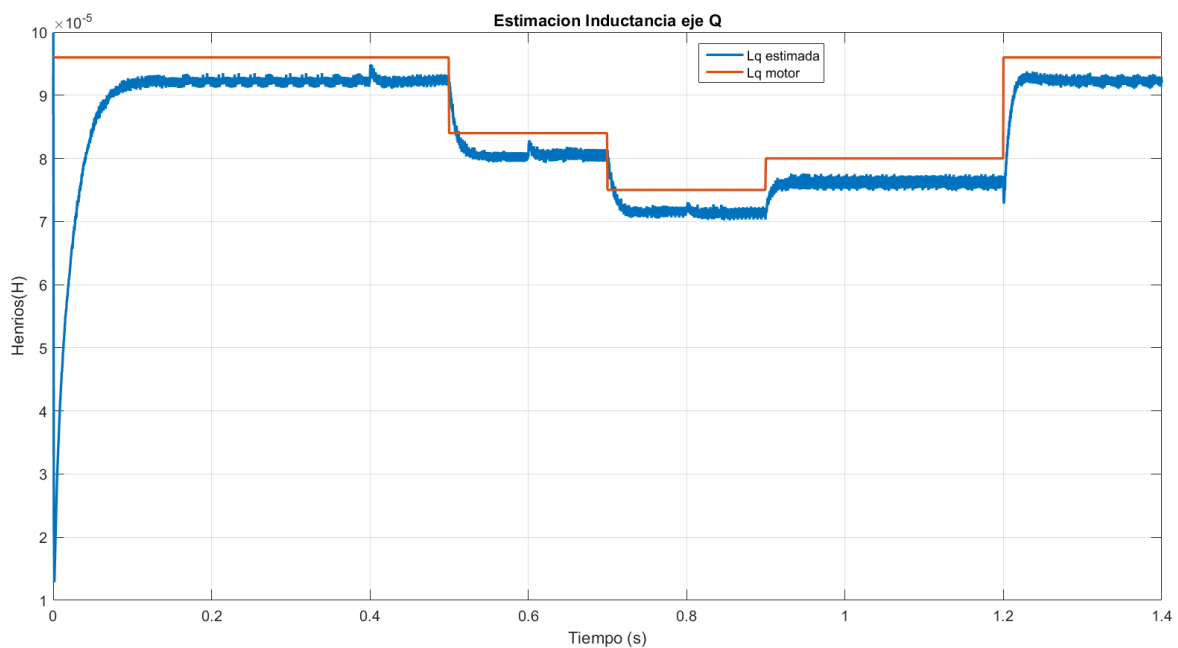


Fig. 32 Estimación de L_q con L_q y L_d con variaciones

6. CONCLUSIONES

En este TFG se ha propuesto una solución para la estimación de los parámetros de un motor PMSM, permitiendo así operar con cualquier máquina sin la necesidad de actualizar ningún dato del controlador.

La solución propuesta funciona parcialmente, debido a que existe un error de estimación, y es flexible a pequeñas variaciones de los parámetros o fallos de estimación.

Se han probado las combinaciones de estimación de parámetros más relevantes y el algoritmo converge a un valor relativamente cercano al deseado, existiendo un error de estimación, salvo en el caso de intentar estimar las cuatro variables a la vez. Para hacerlo funcionar habría que buscar algún otro método, debido a que tenemos dos ecuaciones y cuatro incógnitas siendo imposible estimar las cuatro a la vez.

Se proponen las siguientes líneas de trabajo para investigaciones futuras:

- Conseguir estimar las cuatro variables al mismo tiempo
- Estudiar el comportamiento en distintos puntos de operación.
- Reducir el error de estimación a valores aceptables.
- Calcular el tiempo de cálculo, para saber el tiempo que costaría procesarlo al controlador.
- Probar los estimadores en una plataforma real.

7. REFERENCIAS

- [1] Sanz, A., & Oyarbide, E. (2018). Método sensorless para PMSM basado en la derivada de la corriente para muy bajas velocidades. *Jornada de Jóvenes Investigadores Del I3A*, 6(2), 3–5. <https://doi.org/10.26754/jji-i3a.201802822>
- [2] Ren, Y., & Zhao, C. (2010). Optimal PID controller design in PMSM based on Improved Genetic Algorithm. *ICIMA 2010 - 2010 2nd International Conference on Industrial Mechatronics and Automation*, 2, 123–126. <https://doi.org/10.1109/ICINDMA.2010.5538354>
- [3] Filtro de Kalman. (2019, 11 de septiembre). Wikipedia, La enciclopedia libre. Fecha de consulta: 14:44, enero 14, 2020 desde https://es.wikipedia.org/w/index.php?title=Filtro_de_Kalman&oldid=119180733.
- [4] Shah, R., & Gajjar, R. (2018). A comparative study of various methods for parameter estimation of PMSM. *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing, ICECDS 2017*, 1712–1715. <https://doi.org/10.1109/ICECDS.2017.8389742>
- [5] Ouvang, Y., & Dou, Y. (2018). Speed Sensorless Control of PMSM Based on MRAS Parameter Identification. *ICEMS 2018 - 2018 21st International Conference on Electrical Machines and Systems*, 1618–1622. <https://doi.org/10.23919/ICEMS.2018.8549314>
- [6] Dang, D. Q., Rifaq, M. S., Choi, H. H., & Jung, J.-W. (2016). Online Parameter Estimation Technique for Adaptive Control Applications of Interior PM Synchronous Motor Drives. *IEEE Transactions on Industrial Electronics*, 63(3), 1438–1449. <https://doi.org/10.1109/TIE.2015.2494534>
- [7] Wikipedia contributors. (2019, November 6). Least mean squares filter. In Wikipedia, The Free Encyclopedia. Retrieved 17:59, January 14, 2020, from https://en.wikipedia.org/w/index.php?title=Least_mean_squares_filter&oldid=924881077