

Implementación de dos estrategias de control para la velocidad de un motor DC

Implementing two DC motor speed control strategies

José Danilo Rairán Antolines¹, Yeni Paola Sierra Niño² y Néstor Iván Moreno Roballo⁵

RESUMEN

Si bien las técnicas de control sobre sistemas lineales ya han sido probadas con suficiencia, en la práctica los sistemas son no lineales. Esto obliga a que se hagan ajustes en la sintonización de los controladores, para que sean útiles en una configuración experimental. Este artículo muestra el proceso de sintonización y reajuste de un control PID y un control basado en un modelo de referencia, los cuales actúan sobre un motor DC de imán permanente. Al final, se evalúa el desempeño de cada algoritmo y se dan recomendaciones.

Palabras clave: controlador PID, control basado en un modelo de referencia, motor DC, identificación, control en tiempo real.

ABSTRACT

While linear control techniques for dynamic systems have been widely tested, systems are not linear in practice. This means that controllers must be re-tuned to make them useful in an experimental setup. This article presents the tuning and re-tuning process for two control strategies: a PID and an algorithm based on the choice of overall transfer function controlling a DC permanent magnet motor. The algorithms' performance is evaluated and some recommendations are made.

Keywords: PID controller, choice of overall transfer function, DC motor, identification, real-time workshop.

Recibido: julio 4 de 2008

Aceptado: mayo 29 de 2009

Introducción

La implementación de un controlador PID no es un problema resuelto en su totalidad aún. A medida que se incrementa el grado de detalle en el análisis de cualquier problema, surge más complejidad de la que se espera; es así como hasta la sintonización de un controlador clásico para el ajuste de la velocidad de un motor DC requiere experticia, además del uso de herramientas matemáticas que en principio parecen no ser necesarias. En particular en este artículo, incluso desde cuando se identifica la ganancia del sensor, es notorio que el comportamiento del tacómetro dista de ser ideal; esa aparente dicotomía entre la simulación y la implementación en tiempo real es el problema que se trata en él. Se presenta el diseño de dos controladores, en apariencia sencillos, y se demuestra que, aunque se requieren ajustes, el procedimiento de diseño que parte de una función de transferencia, tratamiento matemático y simulaciones es válido, pero solo como punto de partida en la selección de las ganancias del compensador real.

El motor DC es uno de los sistemas dinámicos más utilizados en la prueba de algoritmos de control debido a que su función de transferencia puede ser reducida a un sistema lineal de orden dos; por esto se ha convertido en el ejemplo obligado en los textos de control clásicos. Además es un dispositivo económico, tanto así que es popular en la generación de movimiento, rotacional o lineal; también en procesos más complejos, como en la robótica, la aviónica y la automatización de procesos en general, entre muchos más (Liu *et al.*, 2006; Serres *et al.*, 2007). Las razones ya expuestas sirven de motivación para trabajar con el motor DC de imán perma-

nente, con el objetivo de verificar las virtudes de su utilización; sin embargo, no pueden despreciarse las alinealidades, como la banda muerta, ganancias de avance y retroceso distintas, histéresis y saturación (Thirusakthimurugan y P. Dananjayan, 2007). Estas alinealidades restringen el uso de las constantes de sintonización de cualquier controlador, cuando estas son el resultado del diseño a partir de un modelo ideal.

En la literatura científica se encuentran aplicaciones de estrategias de control para el ajuste de la velocidad en DSP, FPGA, μ c y PC (Tipsuwan, Sribye y Kamonsantiroj, 2007; Akkaya, Kulaksiz y Aydogdu, 2007; Gupta, Mukhopadhyay y Moi, 2005). Una de las razones por las cuales en este artículo se utiliza el PC es porque se espera que este desarrollo se utilice en la enseñanza del control, y el PC resulta adecuado para el almacenamiento, la manipulación y la visualización de información. El programa sobre el cual se prueban los algoritmos es Matlab; en particular se usa la herramienta de trabajo en tiempo real, la cual permite la conexión con Simulink; así, el resultado de otras herramientas de Matlab es aprovechado de manera directa, por ejemplo, de la herramienta de diseño de filtros, de diseño de sistemas de control SISO o de la herramienta de identificación. Otra opción es programar en LabView; este también se manipula de manera gráfica, aunque con otra filosofía (Thepsantorn *et al.*, 2006). La interfaz entre el sistema a controlar y el computador es una tarjeta de adquisición de datos de National Instruments, la PCI-6024E, la cual cuenta con entradas y salidas analógicas y digitales.

Hasta este momento se ha justificado el trabajo con el motor DC y con el control PID; ahora se explica la configuración en lazo ce-

¹ Candidato a Ph.D., en Ingeniería de Sistemas y Computación. Profesor, Universidad Distrital Francisco José de Caldas, Colombia. drairan@udistrital.edu.co.

² Tecnóloga en electricidad. Ingeniera electricista, Universidad Distrital Francisco José de Caldas, Colombia.

³ Tecnólogo en electricidad. Ingeniero electrónica, Escuela Colombiana de Carreras Industriales, Colombia. gice@udistrital.edu.co

rado, la cual se presenta en la Figura 1. En ella se diferencian dos partes del sistema de control: dentro del PC (en ambiente Simulink), y fuera de él (el montaje de laboratorio). Son importantes los bloques de adquisición de datos, la etapa de potencia, el sensor, el motor y el algoritmo de control, porque corresponden a las secciones de este artículo.

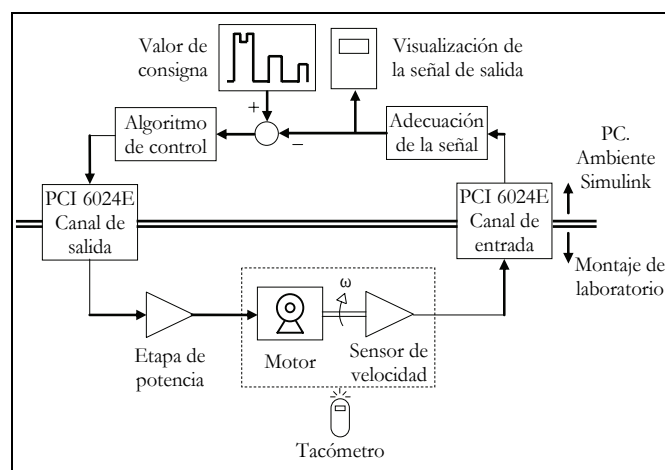


Figura 1. Montaje experimental. La conexión entre el PC y el motor la realiza una sola tarjeta de adquisición de datos, con un canal de salida y otro de entrada. El motor y el sensor están ligados por el eje y constituyen una unidad; además se utiliza un tacómetro digital externo para realizar algunas mediciones.

En el artículo se dedican las dos secciones iniciales a la configuración del experimento, de manera que pueda ser replicado por los investigadores interesados. Se enfatiza en la configuración de la tarjeta de adquisición de datos, en Matlab, y en la adecuación de las señales del tacómetro. Luego, en otra sección, se obtiene el modelo del motor, como una función de transferencia, la cual es indispensable para sintonizar los controladores. Se dedican dos secciones más a los algoritmos de control PID y al aquí denominado modelo de referencia; el resultado es la selección de las constantes de ajuste de cada controlador. En la sección siete se hace una discusión con respecto a la efectividad de las dos configuraciones, pero ya en el montaje experimental, por lo cual se requieren reajustes en las constantes de sintonización. Se finaliza con la sección de conclusiones y recomendaciones acerca de trabajo futuro.

Adquisición de datos y etapa de potencia

En la adquisición de datos se utiliza un canal de entrada de la tarjeta PCI-6024E de National Instruments, la cual toma datos de voltaje análogo del sensor, en un rango de -10 a 10 V, y los convierte a formato digital de doce bits, los cuales quedan listos para ser tratados o almacenados en el PC. Este proceso se realiza a una tasa de 1 ms; el tiempo de muestreo se define luego de una prueba que comienza en $0,1$ s y va hasta 1 ms. A esa velocidad se logra la mejor visualización de la señal del tacómetro, la cual es una onda senoidal con frecuencia y amplitud proporcionales a la velocidad del motor. La frecuencia angular máxima es 555 rad/s, y la amplitud máxima es $6,6$ V.

El primer paso en la configuración de la tarjeta de adquisición es instalar la utilidad *Real-Time Windows Target* de Matlab. Asociado a esta se debe contar con un compilador, para este artículo se utiliza *Microsoft Visual Studio C/C++*. Enseguida se inicia un archivo nuevo en Simulink y se le adiciona un bloque del menú *Real-Time Windows Target*, el bloque puede ser *Analog Input*; se accede al bloque con doble clic; se selecciona la tarjeta PCI-6024E

y se corre la prueba de la tarjeta; de esta manera se sabe si está instalada y bien configurada. En este momento se define el intervalo de muestra.

En el menú *Simulation* de Simulink se accede a la opción *Configuration parameters* y se ajusta el tipo de solución como "ode5", a pasos fijos de 1 ms. En la misma ventana se selecciona la opción *Real-Time Windows Target* y se elige el *target rtwin.tlc*, el cual debe estar disponible en la lista si los pasos del párrafo anterior fueron bien realizados. Además, en el menú *tools* de Simulink se va a la opción *External Mode Control Panel*, y en la configuración de la señal y el disparo se escoge una duración adecuada, en función del tiempo de simulación que se requiera, por ejemplo 1×10^6 muestras, que en este caso equivale a 1.000 s. Para iniciar la ejecución del programa, primero Simulink debe estar en modo externo, en seguida se compila, y luego se conecta (con el icono *Connect to Target*), finalmente se está listo para correr el programa.

En la construcción de la etapa de potencia se tiene en cuenta que se va a ajustar un único sentido de giro, esto implica que solo se necesitan voltajes con una polaridad, por lo tanto la etapa de potencia es un transistor configurado en modo emisor común. Además de esto se observan las restricciones que fija la electrónica: de una parte la tarjeta puede entregar hasta 5 mA y de otra el motor a voltaje nominal y en vacío requiere 100 mA. La resistencia de base se halla teniendo en cuenta que la salida máxima de la tarjeta es 10 V, entonces con una resistencia de 10 k Ω se tiene una corriente de base máxima cercana a 1 mA. Así se limita la corriente y se protege la tarjeta. Se utiliza el transistor 2N2222, por su costo bajo y porque opera dentro del rango de corriente y voltaje que se necesita.

Modelado del sensor

Si ya se ha configurado la tarjeta de adquisición de datos y pueden enviarse órdenes de Simulink al motor, enseguida se trabaja con el sensor. Este, además de entregar la señal de retroalimentación, define la ganancia del sistema en lazo cerrado, como el inverso de la ganancia del sensor; siempre y cuando el conjunto PID-planta tenga un polo en el origen, como puede probarse por álgebra de bloques. De otra parte, es fundamental garantizar que la constante de tiempo del sensor sea menor que la de la planta; así se asegura un seguimiento adecuado de la velocidad. En este caso la constante de tiempo del motor es cercana a 300 ms.

El sensor de velocidad es una dínamo tacométrica y forma un conjunto con el motor, porque viene acoplada al eje del motor, de fábrica. La señal de salida es una senoidal con frecuencia y amplitud variable en función de la velocidad angular del motor. No se utiliza la variación de frecuencia porque requiere más tiempo que leer la amplitud; sin embargo, la amplitud exige rectificar y filtrar la señal senoidal (Choi et al., 2007). La rectificación consiste en el cálculo del valor absoluto, el cual se hace desde Simulink, donde también se implementa el filtro. Este último es en una función de transferencia continua de orden uno, con ganancia unitaria. Luego de varias pruebas se determinó que la mejor constante de tiempo es 50 ms; tiempos superiores retrasan la señal y tiempos menores dejan rizado. Con estos ajustes puede considerarse al sensor funcional.

Ahora conviene hacer un barrido de voltaje por el rango de operación del motor, el cual se supone entre 0 y 12 V para el motor trabajado; este proceso sirve para detectar alinealidades del motor y del sensor (Saco, Mallo y Mazzone, 2003). En principio se realiza

con una señal de la forma $v(t) = 5 + 4\text{sen}(0,2\pi t)$, pero es visible que el rango de funcionamiento del motor es menor, y se pasa a $v(t) = 3,8 + 2\text{sen}(0,2\pi t)$; el resultado de esta prueba se observa en la Figura 2. La conclusión más importante es que la zona lineal del motor es menor de lo que se espera, y se reduce a voltajes de alimentación que van de 3,3 a 4,3 V. En adelante todos los cálculos, simulaciones y datos experimentales se refieren a esa zona de trabajo.

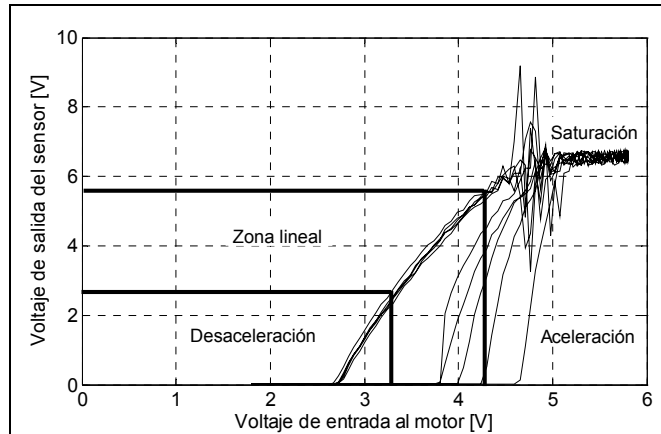


Figura 2. Barrido de voltaje de alimentación en el motor y respuesta del sensor. Con esta prueba es visible que el motor, cuando menos, posee banda muerta, saturación e histéresis. Además de esto hay que tener en cuenta que esta prueba es para un único sentido de giro, por lo cual el número de alinealidades puede aumentar.

La determinación de la ganancia del sensor, o patronamiento, requiere un experimento nuevo. En este se aplica un voltaje DC al motor, se espera hasta cuando la velocidad se estabiliza, se toma la lectura de velocidad angular con un tacómetro externo y se mide el voltaje de salida del sensor. El tacómetro externo usado fue el DL2026R de DeLorenzo. Enseguida se repite el mismo procedimiento, cada 0,1 V, de 3,3 a 4,3 V. Se construye la curva con eje X como la velocidad angular medida por el tacómetro; en el eje Y, el voltaje de salida del sensor. El resultado es una línea recta con pendiente 0,016, la cual es la ganancia del sensor $-K_s-$. El coeficiente de correlación, cuando los datos se ajustan a una línea recta, es 0,999; esto indica que puede confiarse en la linealidad del sensor en la región de interés.

Una prueba más consiste en implementar un compensador PI al sistema, para observar el comportamiento general del motor, aunque por ahora sin realizar el proceso de sintonización. Por ensayo y error la selección es $K_p = 1,5$ y $K_i = 2,5$; además se define el valor de consigna con una fuente que exige cambios tipo escalón y rampa, con el fin de observar la dinámica del sistema en lazo cerrado. El resultado es que la salida del sensor indica cambios que, por el tiempo de respuesta del motor, no pueden ser posibles; así que se interpretan como ruido en el sensor. Ese comportamiento puede verse en la parte superior de la figura 3, en ella las oscilaciones de frecuencia alta del sensor son visibles como una línea gruesa. La decisión es adicionar un filtro a la salida del filtro actual, aunque también es posible pensar en reemplazarlo.

En el diseño del filtro se utiliza la herramienta *toolbox Filter Desing & Analisis Tool* de Matlab. Las dos opciones que esta ofrece son los filtros tipo FIR e IIR, de los cuales, por los resultados, se selecciona IIR. Se probaron las configuraciones Butterworth, Chevyshev y elíptico, comenzando en orden 3 y hasta orden 5. El filtro seleccionado es un Butterworth de orden 5, configurado como pasa-

bajos, con frecuencia de corte en 20 Hz y frecuencia de muestreo de 250 Hz.

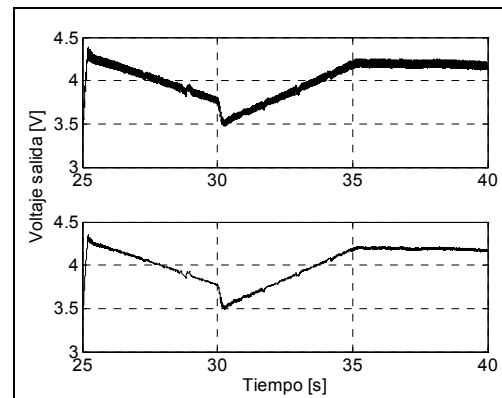


Figura 3. Salida del sensor, con filtro y sin él. La grafica muestra una fracción del comportamiento del sistema, la cual se considera adecuada para justificar la conexión del filtro nuevo.

Identificación del motor

Una vez que la señal del sensor es confiable y se conoce su ganancia, se encuentra un modelo matemático de la planta, porque este es indispensable en el proceso de sintonización de los controladores. Si para llegar a ese modelo se parte de los datos experimentales de la relación salida-entrada, al modelado se lo denomina identificación; este es el procedimiento seleccionado en este artículo. La planta con que se trabaja es un motor DC de imán permanente de la marca Bühler y referencia 1.13.78.xxx; las características principales de funcionamiento son: voltaje de 12 V, potencia de 4,3 W, velocidad de 4.060 rpm, par de 10 mNm. Se evita la determinación de las ecuaciones del motor, aunque es lo recomendado en los libros de texto, pues requieren la medición de constantes como la resistencia, inductancia, fricción e inercia, algunas de las cuales cambian con el punto de operación del motor o son costosas de medir.

La herramienta de Matlab que realiza la identificación se denomina *System Identification*. Ella necesita de una colección de datos de entrada (el voltaje de alimentación del motor), y uno de salida (la velocidad angular del motor). El conjunto de datos debe contener toda la dinámica del sistema a identificar, por lo cual es conveniente utilizar señales de entrada con cambios rápidos y lentos, relativos a la velocidad de respuesta del motor. En este trabajo se seleccionan voltajes en la ventana 3,3 a 4,3 V, con transiciones de tipo escalón, debido a que son las más difíciles de seguir por un sistema dinámico. Cada voltaje se mantiene por un tiempo aleatorio, el cual, luego de varias mediciones en el laboratorio, debe estar entre 0,5 s y 2,5 s.

Al importar los datos de entrada se informa a la aplicación que el tiempo inicial correspondiente a estos es $t = 0$ s y que el intervalo de muestreo es de 1 ms. Esta información es necesaria para que el programa reconstruya el eje tiempo, porque los datos que se importan son solo un vector con el voltaje o la velocidad angular. Una vez que se identifica es posible observar el grado de coincidencia de los datos importados con sistemas lineales de orden 1 en adelante, de manera que el usuario pueda seleccionar el más adecuado. Además de esto existe la opción de forzar al programa para que el modelo tenga un cero o un tiempo de retardo, si es necesario. La primera vez que se corrió la aplicación se alcanzó un grado de coincidencia de 30%, con un sistema de orden 1, luego

se pasó a orden 2 y se alcanzó el 60%; del orden 3 en adelante no se obtuvo incremento en el porcentaje.

Después de analizar los datos se concluye que es posible mejorar la identificación, si se tiene en cuenta el efecto de la banda muerta del motor, como en Kim y Murphy (2003). Mientras el modelo calculado por el programa es lineal, el motor real solo empieza a girar cuando el voltaje de alimentación supera los 2,7 V; por esto las ganancias del motor real y del modelo no pueden coincidir. La corrección consiste en restar 2,7 V a los datos de entrada, con lo cual se obtienen las curvas en la figura 4, cuyo resultado tiene un porcentaje de coincidencia de 83%. Este valor es suficiente para comenzar la sintonización de los compensadores, como se explica en la sección siguiente.

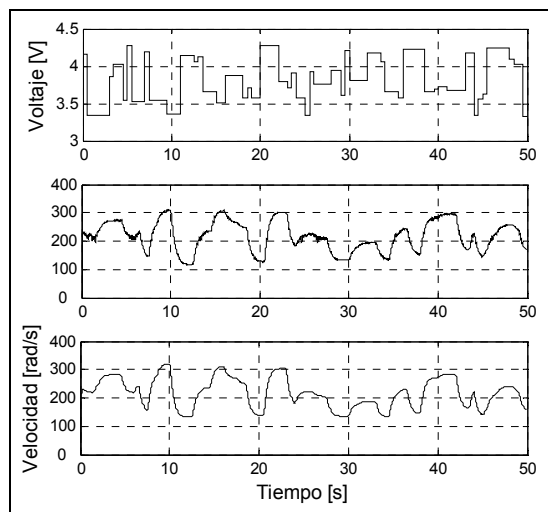


Figura 4. Relación salida-entrada en el motor. La curva superior corresponde al voltaje de alimentación del motor, la de en medio a su respuesta de velocidad, y la inferior a la salida del modelo identificado. Este último proviene del proceso de validación, el cual lo realiza la herramienta de Matlab con los mismos datos de entrada con que identifica.

El modelo final está en la ecuación 1. La función de transferencia $H(S)$ representa la relación de velocidad angular de salida $W(S)$ a voltaje de entrada del motor $V(S)$, en la cual S es la variable compleja. Debe mencionarse que en el modelo de este motor se incluye el efecto de amplificación del transistor o etapa de potencia, debido a que los datos de voltaje de entrada son los que están en el canal de salida de la tarjeta PCI y no en los terminales de alimentación del motor. Se encuentra que la constante de tiempo más lenta es de 370 ms y que la ganancia es 201 rad/s/V.

$$\frac{W(S)}{V(S)} = H(s) = \frac{201}{(0,374 \cdot S + 1)(0,072 \cdot S + 1)} \quad (1)$$

Para efectos de sintonización se utiliza la ecuación 1, porque es lineal, pero para emular el motor de manera más fiel en Simulink, al voltaje de entrada se le restan 2,7; además, solo se permite que alimenten a $H(S)$ voltajes superiores a cero, lo cual se realiza con el bloque denominado saturación.

Sintonización del controlador PID

En esta sección se presenta el diseño del controlador PID, el cual parte de un modelo idealizado del sistema, pasa por las combinaciones P, PI y PID, hasta llegar al reajuste de ganancias en un montaje experimental. El procedimiento de sintonización es el de ubicación de raíces, en todos los casos; este relaciona el valor de los

polos y los ceros del sistema con la respuesta en el tiempo (Emami, Watkins y O'Brien, 2007). De otra parte, la medición de la estabilidad se realiza teniendo en cuenta la ubicación de los polos en el semiplano izquierdo del plano S .

De la función de transferencia en la ecuación 1 se observa que la señal de velocidad tiene un tiempo de estabilización de 1,8 s y una ganancia de 201 rad/s/V. Estos dos parámetros delimitan el funcionamiento de la planta en lazo cerrado en cuanto al tiempo de estabilización, la ganancia y el sobrepico. El objetivo es que el tiempo de estabilización disminuya, por ejemplo, hasta la mitad; en relación con la ganancia, es conveniente mantenerla, así se evita subutilizar o sobrecargar al motor; con respecto al sobrepico, se busca reducirlo, pero no eliminarlo. La reducción en el tiempo de estabilización es importante porque está asociada con el incremento del ancho de banda del sistema, esto se traduce en más operaciones del motor en el mismo tiempo. Para mantener la ganancia se adiciona una constante a la salida del sensor, así este pasa de $K_s = 0,016$ a $K_s = 1/201$; sin embargo, en el laboratorio, el efecto de la saturación en el motor obliga a que solo se llegue a 1/73,5, valores inferiores hacen que el controlador no pueda generar la corrección suficiente para estabilizar la salida. La decisión con respecto a si se permite que la salida tenga sobrepico ante un escalón depende de la aplicación que se dé al motor, por eso no se hace énfasis en este criterio.

El primer algoritmo de control es el proporcional -P-, su salida es la señal de error multiplicada por una constante $-K_p-$, la cual es la ganancia a calcular (se define al error como la diferencia entre el valor de consigna y la señal que proviene del sensor (Figura 1). Este compensador es el más sencillo, sin embargo, su aplicación puede ser suficiente; en caso contrario, cuando menos brinda información acerca de la necesidad de las etapas integral o derivativa. En la búsqueda de la constante K_p se implementó un programa en Matlab, el cual utiliza en su núcleo las instrucciones *rlocus* y *rlocfind* de Matlab; posteriormente, con el propósito de utilizar una herramienta más de Matlab, se utilizó SisoTool; debe anotarse que se encontraron resultados idénticos. Para finalizar, la verificación se efectúa en Simulink.

Por medio del método de ubicación de las raíces se ajusta a K_p en 0,65, con lo cual se logra un tiempo de estabilización de 0,32 s, con cero de sobrepico. El inconveniente es que la salida posee un error de estado estacionario -e.e.e- de 36%. Entiéndase por este error a la diferencia porcentual entre la señal lograda ante un escalón, y la que se espera a la misma entrada. Esto se corrige en el párrafo siguiente incluyendo una componente más al controlador. El último paso es implementar el compensador P en laboratorio, con lo cual se consigue un tiempo de estabilización de 0,38 s y un e.e.e de 36%.

En este caso, dado que la planta no cuenta con un polo en el origen, el error de estado estacionario se elimina al anexar la componente integral al algoritmo de control; así ahora se trabaja con un compensador PI. Este adiciona un cero, ubicado en un punto que depende de las constantes K_p y K_i , y un polo en el origen; por lo cual ya no están solo los dos polos del motor, sino tres. La función de transferencia resultante está en la ecuación 2.

$$C(S) = 1,5 \frac{(1 + 0,33S)}{S} \quad (2)$$

De la ecuación 2 se obtienen las constantes del controlador, teniendo en cuenta que $C(S) = K_p + K_i/S$. Así $K_p = 0,5$ y $K_i = 1,5$. Una vez más, a este diseño se llega mediante un programa con código de los autores y mediante la herramienta SisoTool, comple-

mentado con Simulink. El resultado experimental se aleja de la simulación, por lo que es recomendable reajustar las ganancias, partiendo de las que resultaron del diseño en el PC. Se recomienda comenzar con Kp, para mejorar el tiempo de estabilización; luego cambiar Ki hasta acercarse tanto como sea posible a la respuesta en la simulación; finalmente volver a Kp, hasta lograr un resultado satisfactorio. De estos pasos se llegó a Kp = 1 y Ki = 3.

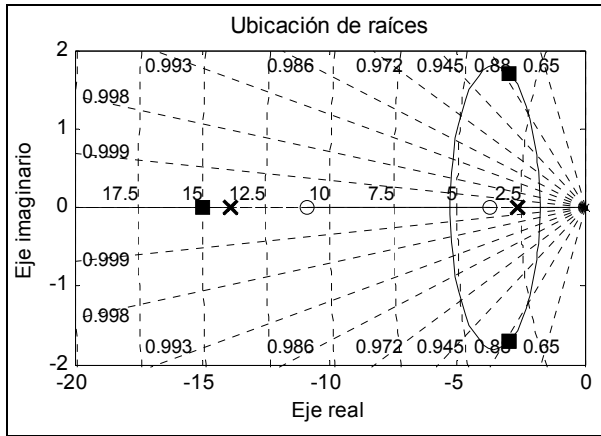


Figura 5. Ubicación de raíces con un compensador PID. La herramienta SisoTool muestra la ubicación de los polos al valor de K seleccionado como ideal (símbolo ■); además los polos del sistema a K = 0 (símbolo X), con el viaje que estos hacen, en este caso hacia dos ceros del controlador (símbolo O) y otro cero en infinito.

Los resultados con la combinación PD siguen presentando e.e.e, por eso no se consignan en este artículo. La última prueba se hace con el compensador PID. Se espera que este elimine el e.e.e, por la componente -I-; también, que reduzca el tiempo de respuesta, por -D-, y que actúe de forma proporcional al error -P- (Lin, Jan y Shieh, 2003). En la Figura 5 está el diagrama de ubicación de raíces de este controlador, cuando la constante de tiempo integral -Ti- es 0,36 s y el tiempo derivativo -Td- es 0,069 s. Aquí Ti = Kp/Ki y Td = Kd/Kp.

La función de transferencia del PID tiene una forma impropia, por lo cual no puede ser implementada como un bloque en Simulink. Su emulación se logra con bloques individuales P, I, D, sumados; sin embargo, el resultado no es el esperado, por los errores que se introducen en Matlab al calcular la derivada de una transición brusca. En la ecuación 3 está la función de transferencia del controlador, de la cual resultan ganancias Kp = 3, Kd = 0,12 y Ki = 3,5; con estas constantes se logra en laboratorio un tiempo de estabilización de 0,4 s y cero de sobrepico.

$$C(S) = 100 \cdot \frac{(1 + 1,23 \cdot S)(1 + 22,9 \cdot S)}{S} \quad (3)$$

Modelo de referencia

El proceso de sintonización de un PID consiste en ir de adentro hacia afuera. Desde adentro porque comienza con el ajuste de las ganancias internas -kp, ki, kd-; y hacia afuera porque busca llegar a un comportamiento general del sistema en lazo cerrado. Una estrategia opuesta es denominada en este artículo "modelo de referencia"; en ella se parte de la selección del comportamiento general del sistema, para finalizar en el diseño del controlador que convierte al sistema en el modelo deseado: el trabajo es de afuera hacia adentro. La ventaja más importante de este segundo punto de vista consiste en evitar el procedimiento de ensayo error que se requiere en la ubicación de los ceros del PID; en lugar de ello la

función de transferencia del compensador se obtiene de manera determinística, siendo una única por cada modelo de referencia.

En el control por medio del modelo de referencia se conserva la estructura de la Figura 1, y se define al compensador a partir de una función de transferencia general -H0(S)-; esta representa el comportamiento ideal en lazo cerrado (Chen, 1993). En este artículo H0(S) tiene el mismo número de polos del motor y no tiene ceros, con el fin de minimizar las operaciones matemáticas y los parámetros a calcular. Los polos se escriben como polinomios, de los cuales los más usados son los de tipo Bessel, diseñados para no presentar sobrepico, y los ITAE, que optimizan la energía consumida para llegar a un valor estable. El polinomio ITAE (de las siglas en inglés para "integral del tiempo multiplicado por el valor absoluto del error"), presenta sobrepico; no obstante, son rápidos en cuanto al tiempo de respuesta (Zhang y Xu, 2002). En la ecuación 4 está el polinomio ITAE, y en la 5, el tipo Bessel, de orden 2 (Franklin, Powell y Emani-Naeini, 1991). En ambas ecuaciones ωn es la frecuencia natural, en rad/s.

$$S^2 + \sqrt{2}\omega_n S + \omega_n^2 \quad (4)$$

$$S^2 + \sqrt{3}\omega_n S + \omega_n^2 \quad (5)$$

A continuación se determina la constante ωn, solo en el caso ITAE, porque es el mismo procedimiento para el tipo Bessel. Se comienza con la definición completa de H0(S), como está escrito en la ecuación 6, donde H0(S) tiene la estructura canónica de un sistema de orden dos.

$$H_0(S) = \frac{73,5\omega_n^2}{S^2 + \sqrt{2}\omega_n S + \omega_n^2} \quad (6)$$

Ahora se encuentra el algoritmo de control, en función de los parámetros siguientes: el sistema deseado -H0(S)-, el modelo del motor -H(S)- y la constante del sensor -Ks-. Esto se realiza por álgebra de bloques sobre la configuración en la figura 1; es así como se obtiene la ecuación 7.

$$C(S) = \omega_n^2 \frac{73,5}{201} \cdot \frac{(0,374S + 1)(0,072S + 1)}{S^2 + \sqrt{2}\omega_n S} \quad (7)$$

Si se aumenta ωn la velocidad del sistema general crece, tanto así que ωn puede relacionarse con el tiempo de estabilización -ts-, mediante la ecuación siguiente: ts ≈ 4,6/(0,707ωn); esta es una opción válida para calcular ωn (Rairán, 2007). Existe otro criterio, que tiene en cuenta el primer valor de corrección del controlador; esta es una medida del esfuerzo del controlador, por lo cual se prefiere en este trabajo. En la ecuación 8, y una vez más por álgebra de bloques, se determina la salida del controlador -U(S)-.

$$U(S) = \frac{1}{S} \cdot \frac{73,5}{201} \cdot \frac{\omega_n^2 \cdot (0,374S + 1)(0,072S + 1)}{S^2 + \sqrt{2}\omega_n S + \omega_n^2} \quad (8)$$

En seguida se define un valor aceptable para la salida inicial del controlador -u(0)-; en este caso se asume u(0) = 1, así el valor más alto que recibe el motor ante una transición tipo escalón es igual al valor de referencia. Luego se utiliza el teorema del valor inicial, como se presenta en la ecuación 9.

$$\lim_{t \rightarrow 0} u(t) = \lim_{S \rightarrow \infty} S \cdot U(S) = 1 \longrightarrow \omega_n \cong 10 \quad (9)$$

Una vez se encuentra el valor de ωn ya se tiene la función de transferencia del controlador. Resta implementarla y verificar su funcionamiento en laboratorio. En la sección siguiente se discute

acerca de los resultados experimentales con los compensadores de polinomio ITAE y Bessel, y se comparan con el algoritmo PID.

Medición de la efectividad de los controladores

En esta sección se comparan los resultados de la implementación de las dos estrategias, a saber: control PID y control basado en un modelo de referencia, cada una con sus variantes. Con este objetivo, en la Tabla 1 se recopila el tiempo de estabilización y el sobrepico de la señal de velocidad angular; así mismo, se evalúa la respuesta en frecuencia y la respuesta ante disturbios.

Los resultados en la Tabla 1 corresponden a la medición experimental del comportamiento del sistema en lazo cerrado, ante una señal escalón de entrada. Es necesario mencionar que el controlador P es el único con error de estado estacionario, de 36%, por lo cual no debe tenerse en cuenta en una aplicación industrial. De otra parte, al analizar las variantes PID puede verse que la opción PID es la mejor, porque no presenta sobrepico y su tiempo de estabilización es el mismo que el de las demás combinaciones. En el caso del modelo de referencia, sus combinaciones producen un tiempo de estabilización mayor que la opción PID; y además, distan de presentar los datos de las simulaciones. Por ejemplo, en el caso del polinomio Bessel debería no producirse sobrepico, pero su aplicación práctica sí lo hace. Esto se debe a la diferencia entre el modelo con que se sintoniza y la planta física; recuérdese que el modelo matemático corresponde a la idealización del funcionamiento en la región lineal, comprendida entre 3,3 y 4,3 V de alimentación del motor (Figura 2). Aunque todos los resultados son cercanos, puede afirmarse que la opción PID es la más adecuada en cuanto a los dos parámetros analizados.

Tabla 1. Comparación de resultados PID – Modelo de referencia

Tipo de Controlador	Tiempo de estabilización $-t_s$ [s]	Sobrepico $-M_p$ [%]
P	0,38	0
PI	0,4	8,2
PID	0,4	0
ITAE	0,9	9,6
Bessel	0,9	8,2

El tiempo de estabilización y el sobrepico son los parámetros para medir la efectividad del controlador ante transitorios tipo escalón. No se incluye la opción PD porque genera error de estado estacionario, al igual que el control P, por tanto no agrega información importante en la selección del controlador más adecuado.

El criterio más importante en la evaluación de un compensador es la estabilidad; en este artículo esta se evalúa por medio de la observación de la ubicación de los polos del sistema en lazo cerrado, los cuales deben estar en el semiplano izquierdo del plano S. Otros criterios de interés, además de la estabilidad, son la medición del ancho de banda y la respuesta ante disturbios, como se efectúa a continuación. La medida del ancho de banda permite estudiar la salida del sistema en régimen senoidal permanente. En esa prueba se mide la relación salida a entrada en decibelios $-db$.

La diferencia lazo abierto-lazo cerrado a frecuencias bajas, en la parte superior de la Figura 6, es cercana a 9 db. Esta se debe a las ganancias del motor y sensor; de una parte 201 y de otra 1/73,5. Otra característica, la cual justifica la aplicación del control en lazo cerrado, es la frecuencia de corte $-\omega_c$. En lazo abierto ω_c es cercana a 1 rad/s, y en lazo cerrado, 10 rad/s; dado que el ancho de banda del sistema aumenta, pueden hacerse más operaciones por unidad de tiempo. El comportamiento en frecuencia del control PID y el llamado modelo de referencia, a bajas frecuencias, es si-

mil; sin embargo, mientras la frecuencia de corte del primero es 10 rad/s, la del segundo es cercana a 6 rad/s, como se consigna en la Figura 6.

En el análisis de la respuesta ante disturbios se aplica una señal sobre la cual no se tenga control, luego se verifica el trabajo del compensador para llevar el sistema al valor estable anterior a la aparición del disturbio. En tal caso la señal de disturbio puede aparecer sobre la corriente, el par de carga, la posición angular, entre otros, es decir, cualquier señal diferente a la velocidad, que es el parámetro medido a través del sensor, y por lo tanto la única variable física controlada. En el caso de la Figura 6, parte inferior, se seleccionó emular el disturbio sobre la salida del sensor, o sea un voltaje, con una duración de 0,5 s y una amplitud aleatoria y máxima del 10% del valor de referencia. El resultado es que el sistema no tarda más de 0,3 s en retomar el valor estable.

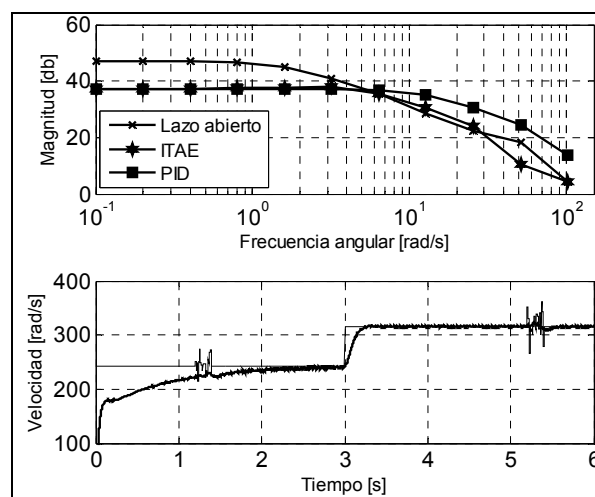


Figura 6. Respuesta en frecuencia y ante disturbios. La respuesta en frecuencia (arriba) muestra un aumento en la frecuencia de corte, además de una regulación de la ganancia, en este caso igual a 20 log(73,5). En cuanto a la respuesta a disturbios (abajo), el sistema recupera el comportamiento anterior al disturbio.

Conclusiones

La implementación de las dos estrategias de control, PID y modelo de referencia, diseñadas para sintonizarse sobre sistemas dinámicos lineales, son útiles en el motor DC, el cual cuenta con un número alto de alinealidades, como es la saturación y la banda muerta. Se comprobó que es conveniente comenzar con la definición de una zona de comportamiento lineal, de la cual se obtienen ganancias iniciales, para ajustarlas en el montaje experimental, con el fin de tener en cuenta las diferencias entre el modelo idealizado y el motor.

Se verificó que la identificación del sistema es una herramienta adecuada para obtener el modelo lineal del motor; esto debido a que se basa en el funcionamiento real y particular de la máquina, y evita el cálculo de las constantes que son necesarias al modelar a partir de las ecuaciones diferenciales. El grado de coincidencia entre los datos experimentales y el modelo final es de 83%, el cual no puede ser mejorado por el efecto de la banda muerta en el motor. Si bien ese grado puede considerarse bajo en algunas aplicaciones de ingeniería, no es así en control, donde el ajuste fino del controlador se da en el montaje de laboratorio.

Tanto en PID como en el modelo de referencia no es posible asignar como ganancia de lazo cerrado la misma de lazo abierto. Ello

se debe a la saturación que presenta la máquina, la cual hace que el motor no responda siempre de manera proporcional al voltaje de corrección, especialmente cuando esa corrección es superior a 4,5 V. El resultado es que el error no se puede eliminar, y debido al integrador del controlador, el voltaje de corrección $-u(t)$ aumenta hasta un máximo de 10 V. En este trabajo la ganancia pasa de 201 rad/s/V en lazo abierto a 73,5 rad/s/V en lazo cerrado.

Bibliografía

- Akkaya, R., Kulaksiz, A., Aydogdu, O., DSP implementation of a PV system with GA-MLP-NN based MPPT controller supplying BLDC motor drive., *ScientDirect Energy conversion and management*, Konya Turkey, Jun. 2007, pp. 210-218.
- Chen, C., Analog and digital control system design, editorial: Saunders Collage Publishing., Primera edición, impreso en Estados Unidos de América, 1993, pp. 339-344.
- Choi, G. H., Ra, W. S., Yoon, T. S., Park, J. B, Low-Cost Tachometer based on the Recursive Frequency Estimation for Automotive Applications., *SICE Kagawa University, Japón*, Sep, 2007, pp. 46-49.
- Emami, T., Watkins, J. M., O'Brien, R. T., A Unified Procedure for Continues-Time and Discrete-time Root-Locus and Bode Design., *American Control Conference, ACC '07*, Jul, 2007, pp 2509 – 2514.
- Franklin, G., Powell, D., Emani-Naeini, A., *Automatic Control Systems.*, editorial: Addison-Wesley Iberoamericana S.A., impreso en Estados Unidos de América, 1991, pp. 334-337.
- Gupta, G., Mukhopadhyay, S., Moi, C., A project based approach to teach mixed-signal embedded microcontroller for DC motor control., *IEEE Proceedings of the third IEEE International workshop on electronic design, Malaysia, Ene*, 2005, pp. 1-6.
- Kim, W-J., Murphy, B. C., Development of a Novel Direct-Drive Tubular Linear Brushless Permanent-Magnet Motor., *Industry Applications Conference, 2003, 38th IAS Annual Meeting*, Oct, 2003, pp.1664-1671.
- Lin, Ch-L., Jan, H-Y., Shieh, N-Ch., GA-Based Multiobjective PID Control for a Linear Brushless DC Motor., *Mechatronics, Transaction en IEEE/ASME*, Vol. 8, No. 1, Mar, 2003, pp 56 – 65.
- Liu, H., Meussel, P., Seitz, N., Willberg, B., Hirzinger, G., Jin, M. N., Liu, Y. W., Wei, R, Xie, Z. W., Grant M., The modular multisensory DLR-HIT-Hand., *ScienceDirect Mechanism and Machine Theory*, 42, Jun, 2006, pp. 612-625.
- Rairán, D., *Análisis de sistemas dinámicos y control PID.*, primera edición, Colombia: Universidad Distrital Francisco José de Caldas, 2007, pp. 348-351.
- Saco, R., Mallo, S., Mazzone, V., Construcción y diseño de controladores de un péndulo invertido rotante., *Proyecto de grado, Ingeniería Automatización y control industrial, Universidad Nacional de Quilmes, Argentina*, Feb, 2003.
- Serres, J., Dray, D., Ruffer, F., Franceschini, N., A vision-based autopilot for a miniature air vehicle: joint speed control and lateral obstacle avoidance., *Springer Auton robot, Dic*, 2007, pp. 103-122.
- Thepsantorn, P., Numsomran, A., Tipwanporn, V., Teanthong, T., DC motor speed control using fuzzy logic based on Labview., *IEEE International joint conference, Busan Korea*, Oct, 2006, pp. 3617-3620.
- Thirusakthimurugan, P., Dananjayan, P., A novel robust speed controller scheme for PMSBLDC motor., *ScienceDirect ISA Transactions*, Jun, 2007, pp. 471-477.
- Tipsuwan, Y., Sribye, J., Kamonsantiroj, S., An experiment study of Network – Based DC motor speed control using SANFIS., en *The 33rd Annual Conference of the IEEE Industrial Electronics Society, Taipei Taiwan*, Nov, 2007, pp. 426-432.
- Zhang, W., Xu, X., On the choice of overall transfer functions., *IEEE Proceedings of the 4th world congress on intelligent and automation, Shanghai China*, Jun, 2002, pp. 560-563.