Discretización de Controladores Análogos y Control clásico en tiempo discreto

Control Digital Avanzado

Camilo Andres Vera Ruiz

[caverar@unal.edu.co](mailto:caverar@unal.edu.co)

1. Características de la Planta

Se trabaja con la misma planta utilizada en la prtactica anterior, con lo cual se tiene que la funcion de transferencia del modelo obtenido, es la siguiente:

Adicionalmente se tiene en cuenta que el periodo de muestreo sigue siendo el mismo, es decir 5ms.

1. Diseño de Controlador

Se requiere un controlador PI con cerro error de estado estacionario y un tiempo de subida menor al 70% del tiempo de subida de la planta sin realimentar, es decir 45.3ms, y teniendo en cuenta que el componente integral del controlador debería eliminar el error de estado estacionario, solo es necesario preocuparse por el tiempo de subida y la saturación del actuador.

El diseño se realizó en tiempo continuo por medio de la herramienta PID Tuner de MATLAB, la cual calcula las constantes del controlador a partir de dos parámetros que controlan el tiempo de respuesta y la robustes. En la figura 3 se muestra como se ajustó el parámetro de tiempo de respuesta con un valor muy cercano a los 45.3ms, en este caso se selecciono el valor de 43.6ms, luego se modificó el segundo parámetro buscando reducir el esfuerzo de control al menor posible, con el fin reducir el riesgo de saturar el actuador, en este caso buscando una amplitud en la señal de control, menor a 1, teniendo en cuenta que la señal de PWM va desde -1 a 1. Finalmente se obtiene un controlador con un tiempo de subida de 41.6ms el cual es menor al límite máximo requerido, así como un repuesta paso prácticamente de primer orden, con un sobre pico muy pequeño de 1.09% y un tiempo de estabilización de 62.3 ms. En la figura 3 se muestra una captura de la herramienta PID Tuner con gráficas de la respuesta al paso y el esfuerzo de control, así como los parámetros de desempeño y las constantes obtenidas para el componente proporcional y el componente integral del controlador, utilizando una topología de control PI paralelo.

Interfaz de usuario gráfica

Descripción generada automáticamente

Figura 3: Diseño de PI

El controlador puede ser expresado mediante la siguiente expresión de función de transferencia:

1. Discretizaciones

Como lo especifica la guía de laboratorio, se implementaron 6 métodos de discretización: invarianza del impulso, invarianza del paso (ZOH), aproximación Backward, aproximación Forward, aproximación Tustin, y mapeo de polos y ceros. 4 de estas se realizaron mediante la función de Matlab “c2d”, mientras que las aproximaciones Forward y Backward se realizaron con ayuda del toolbox de matemática simbólica de MATLAB, realizando un cambio de variable para la aproximación Forward y un cambio de variable para la aproximación Backward. El periodo de muestreo de todas las discretizaciones, fue de 5ms.

En la tabla 1 se presentan las funciones de transferencia resultantes para cada método de discretización, adicionalmente se coloca el ancho de banda como método de verificación de que el periodo de muestreo es lo suficientemente pequeño para que el controlador funcione apropiadamente. En esta caso dado que la frecuencia de muestreo es de 200Hz, y por lo tanto la frecuencia Nyquist es de 100Hz, y además se requiere que esta última sea un orden de magnitud mayor al ancho de banda de los controladores, se concluye que los controladores no deben superar un ancho de banda de 10Hz, cumpliendo con el criterio para todas las discretizaciones.

|  |  |  |
| --- | --- | --- |
| Tabla 1: Funciones de Transferencia de las discretizaciones | | |
| Técnica de discretización | Función de Transferencia | Ancho de banda |
| Invarianza del Impulso |  | 9.2564 Hz |
| Invarianza del paso |  | 8.9244 Hz |
| Aproximación Forward |  | 8.9244 Hz |
| Aproximación Backward |  | 9.305 Hz |
| Aproximación Tustin |  | 9.0849 Hz |
| Mapeo de polos y ceros |  | 9.0916 Hz |

En la figura 4 se muestra una captura con la respuesta al paso de todas las discretizaciones, donde se evidencia que todas tienen un comportamiento muy similar al controlador continuo, con prácticamente el mismo tiempo subida, a excepción de la discretización obtenida mediante el método de la invarianza del impulso, donde el comportamiento cambia drásticamente a un sistema de segundo orden, con un sobre pico cercano al 30% y un tiempo de establecimiento considerablemente mas grande, de alrededor de 250ms, en contraposición a los 62.3ms del controlador continuo, que prevalece aproximadamente en el resto de discretizaciones.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Figura 4: Respuesta al paso de cada método de discretización.

1. Simulación y Resultados de la implementación

Dado que se requiere implementar algunos de estos controladores en un microcontrolador, se tomaron las funciones de transferencia y se convirtieron en su forma de potencias negativas mediante MATLAB, de esta forma cualquier función de transferencia puede ser implementada en el microcontrolador de la siguiente forma, donde representa el valor de la señal de entrada n muestras atrás, y representa el valor de la señal de salida, m muestras atrás.

Las 6 discretizaciones fueron simuladas mediante Simulink, usando el diagrama de bloques que se muestra en la figura 5, donde se busca reflejar de la forma mas precisa posible el sistema real, para ello se utiliza la planta continua, junto con retenedores de orden cero en la señal de realimentación, y en la salida del controlador, simulando el efecto del muestreo del encoder y la aplicación del voltaje al motor mediante PWM, también se incluye un bloque saturación para reflejar los límites físicos de la señal de control que alimenta el motor, y un retenedor de orden cero adicional para reflejar el comportamiento discreto de la señal de referencia. El controlador fue implementado como una función de transferencia en z.

Diagrama

Descripción generada automáticamente

Figura 5: Diagrama de bloques de la simulación

De la Figura 6 a la figura 12 se muestra la respuesta a un señal de referencia cuadrada de 0.25Hz, que varía entre 0.8 y 1.3 rpm para el controlador continuo y para todas las discretizaciones, usando Simulink con el diagrama de bloques que se muestra en la figura 5, graficando además la señal de control, con el fin de verificar que no halla saturación. La señal de control se muestra en color azul, la referencia en amarillo, y la salida en naranja.

Gráfico

Descripción generada automáticamente con confianza baja

Figura 6: Respuesta del controlador continuo.

Diagrama

Descripción generada automáticamente

Figura 7: Respuesta de la discretización por invarianza del impulso.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Figura 8: Respuesta de la discretización por invarianza del paso (ZOH).

Gráfico

Descripción generada automáticamente

Figura 9: Respuesta de la discretización por aproximación Backward.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Figura 10: Respuesta de la discretización por aproximación Forward.

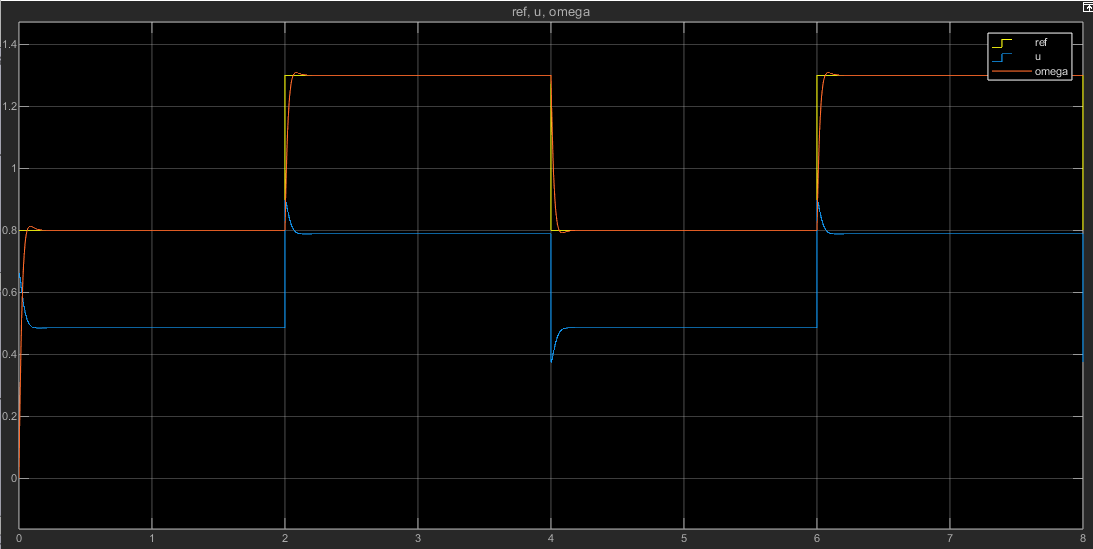


Figura 11: Respuesta de la discretización por aproximación Tustin.

Gráfico

Descripción generada automáticamente

Figura 12: Respuesta de la discretización por mapeo de polos y ceros.

En las anteriores figuras, se evidencia que nunca se da la saturación, es decir la señal de control nunca alcanza el límite de 1, por lo cual se procede a realizar la implementación de los controladores.

1. Implementación y Resultados

Se seleccionaron 3 discretizaciones para implementar en el microcontrolador: invarianza del impulso, invarianza del paso y aproximación Tustin. Para comparar los resultados de la implementación con simulaciones, se seleccionó una señal cuadrada de 0.25Hz que va desde 0.8 a 1.3 rpm, se capturaron los datos de Simulink y de los controlador implementado en el microcontrolador, para cada discretización, luego se exportaron los datos a MATLAB, donde se alinearon de tal forma que las señales de referencia coincidiesen en el tiempo, para finalmente imprimir los datos en una única grafica para cada discretización, incluyendo las señales de control. Los resultados se muestran en las Figura 13, 14 y 15.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Figura 13: Resultados del controlador de invarianza del impulso

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Figura 14: Resultados del controlador de invarianza del paso.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Figura 14: Resultados del controlador por aproximación Tustin.

1. Análisis y Conclusiones

Al analizar los resultados es posible concluir que:

* El modelo del motor de primer orden obtenido mediante MATLAB, modela muy bien la dinámica real del sistema, dado que el comportamiento entre los controladores implementados y las simulaciones es bastante similar, sin embargo el modelo no es lo suficientemente bueno, como para modelar apropiadamente el comportamiento decreciente del sistema, cuya dinámica en la realidad, presenta mayores sobre picos a los esperados por la simulación.
* Tal y como se esperaba, existe un error estacionario en la señales de control, entre el comportamiento real y el simulado, dado que el modelo hallado tuvo un error estacionario de alrededor del 12.5%, sin embargo esto no supone un problema ya que la componente integral del controlador, es capaz de lograr una respuesta al paso con un error de estado estacionario nulo.
* La aproximación por invariancia al impulso es sin lugar a duda, la que tuvo un peor desempeño en aproximar la respuesta del controlador continuo al tiempo discreto, de hecho esta incumple con los requerimientos especificados para el diseño. Por otro lado las aproximaciones por invariancia del impulso y la aproximación Tustin, si que tuvieron un comportamiento muy similar al controlador continuo, tanto para la simulación, como para la implementación real del sistema, logrando incluso un comportamiento más rápido en la implementación real, es decir que se cumple satisfactoriamente con los requerimientos especificados,