

Proyecto de Ingeniería Aplicada

Controlador difuso para sistemas levitadores aerodinámicos usando Arduino Uno

Fuzzy controller for aerodynamic levitating systems using Arduino Uno

Elías Prudencio Chavez Jaldin 1. Edgar Roberto Ramos Silvestre 2.
Dorian Alexander Romero Alcocer 3. Mauricio Chirinos Caero 4.

1. Ingeniero Electrónico, Docente Dpto. de Electrónica y Telecomunicaciones, Universidad Privada del Valle, Cochabamba, elias_1335@hotmail.com
2. Ingeniero Electrónico, Maestría en Tecnología de Cómputo, Director Dpto. de Electrónica y Telecomunicaciones, Universidad Privada del Valle, Cochabamba, eramoss@univalle.edu
3. Estudiante de Ingeniería Biomédica. Universidad Privada del Valle Cochabamba, Cochabamba, alexdron99@gmail.com
4. Estudiante de Ingeniería Electrónica. Universidad Privada del Valle Cochabamba, Cochabamba. maurichc@hotmail.com

RESUMEN

La lógica booleana propuesta por von Neumann presenta problemas a la hora de abordar ciertas tareas del mundo real. Los sistemas borrosos o difusos inciden sobre la lógica booleana asignando valores intermedios entre 1 y 0, emulando el razonamiento del ser humano. Los sistemas levitadores aerodinámicos se caracterizan por su comportamiento no lineal y hacen del sistema difuso un candidato adecuado para su control.

Para el diseño del control difuso, primero se calcula el modelo matemático de los sistemas levitadores aerodinámicos de una esfera y una viga. Se explica los actuadores usados y sus propiedades aerodinámicas: ventilador y motor Brushless. Se realizó la simulación de las ecuaciones diferenciales calculadas en SIMULINK (herramienta usada para diseños basados en modelos) bajo entradas no estándar. Se ajustaron las constantes con los datos reales obtenidos usando MATLAB (laboratorio de matrices o Matrix Laboratory, por sus siglas en inglés). Se asignaron las variables lingüísticas de entrada y salida con sus respectivas particiones empleando MATLAB. Se simularon los diseños del control difuso en SIMULINK obteniendo resultados óptimos. Se implementó estos valores con ligeros cambios de

sintonía en el entorno de desarrollo integrado de la tarjeta de desarrollo electrónico Arduino y se guardaron los datos de respuesta real. Se realizó la comparación de los resultados de simulación obteniendo una coherencia con los resultados simulados y reales. El sistema de control difuso ayudó a realizar un control adecuado y sencillo para sistemas levitadores aerodinámicos.

Palabras clave: Control difuso. Levitación aerodinámica. Lógica difusa. Modelo matemático. Sistema no lineal.

ABSTRACT

The Boolean logic proposed by von Neumann presents problems when dealing with certain tasks in the real world. Fuzzy systems influence Boolean logic by assigning intermediate values between 1 and 0, emulating the reasoning of the human being. Aerodynamic levitating systems are characterized by their non-linear behavior and make the fuzzy system a suitable candidate for control.

For the design of the fuzzy control, the mathematical model of the aerodynamic levitating systems is first calculated from a sphere and a beam. The used actuators and their

aerodynamic properties are explained: Fan and Brushless motor. Simulation of the differential equations calculated in SIMULINK (tool used for model-based designs) was performed under non-standard inputs. The constants were adjusted with the actual data obtained using MATLAB (Matrix Laboratory). The input and output linguistic variables were assigned with their respective partitions using MATLAB. Fuzzy control designs were simulated in SIMULINK obtaining optimal results. These values were implemented with slight tuning changes in the integrated development environment of the Arduino electronic development board and the actual response data was saved. The simulation results were compared, obtaining consistency with the simulated and real results. Fuzzy control system helped make proper and easy control for aerodynamic levitating systems.

Keywords: Aerodynamic levitation. Diffuse logic Fuzzy control. Mathematical model. Nonlinear system.

INTRODUCCIÓN

El camino a la construcción de máquinas inteligentes se puede considerar desde el inicio de la Segunda Guerra Mundial, con el diseño de ordenadores analógicos para controlar cañones antiaéreos o para navegación. Luego, con el surgimiento de nuevas teorías sobre la realimentación, el avance de la electrónica con la invención del transistor, el estudio de sistemas nerviosos de seres vivos se construyeron máquinas capaces de responder y aprender como los animales. Estas máquinas denominadas ahora como ordenadores digitales, están basados en la separación de estructura (hardware) y función (software). El paso inicial conceptual para estos ordenadores se debe al matemático de origen húngaro John von Neumann. Está basado en la lógica digital que opera ejecutando en serie las instrucciones que componen el algoritmo que se codifica en forma de programa y se almacena en la memoria (Del Brío & Molina, 2002).

En 1960, John McCarthy acuña el término inteligencia artificial para definir métodos algorítmicos capaces de hacer pensar a los ordenadores. Después de diez años la inteligencia artificial presentó problemas a la hora de abordar ciertas tareas en el mundo real debido a su base en la lógica booleana de von Neumann. La teoría de sistemas expertos cobra mayor fuerza con sus complejos programas de computador codificando el conocimiento de expertos en formas de reglas de decisión. También surgieron otras series de paradigmas alternativos en los años ochenta como las redes neuronales, los sistemas borrosos o computación evolutiva. Los sistemas difusos o lógica difusa o sistemas borrosos es un tipo de lógica multivaluada, que emulan el lado del software (Del Brío & Molina, 2002).

La lógica difusa (Fuzzy logic en inglés) emula al cerebro en el razonamiento, permite tratar información imprecisa en términos de conjuntos difusos. Estos conjuntos difusos se combinan en reglas para definir acciones (Si-Entonces). Los sistemas de control basados en lógica difusa contienen variables de entrada que por medio de reglas difusas producen variables de salida. La aplicación más extendida de la lógica difusa son los controladores difusos, Fuzzy Logic Controllers (FLC), también llamados sistemas de inferencia difusa, Fuzzy Inference Systems (FIS) (Del Brío & Molina, 2002).

El diseño e implementación de controladores difusos en sistemas de levitación aerodinámica son el principal objetivo del presente artículo. La aerodinámica es una rama de la mecánica de fluidos que se ocupa de estudiar interacciones entre un sólido y el fluido en el que se halla inmerso cuando sus velocidades son distintas (de Saavedra & Acher, 2012). Los sistemas a controlar son de levitación aerodinámica de una esfera y una viga. Estos sistemas de levitación aerodinámica presentan modelos matemáticos no lineales y variantes en el tiempo. Estas dos situaciones dificultan el diseño de otros tipos de controladores digitales, y lo hace un buen candidato al controlador difuso.

METODOLOGÍA

1. Modelos matemáticos de sistemas levitadores aerodinámicos

1.1. Modelado del sistema de levitación aerodinámica de una esfera

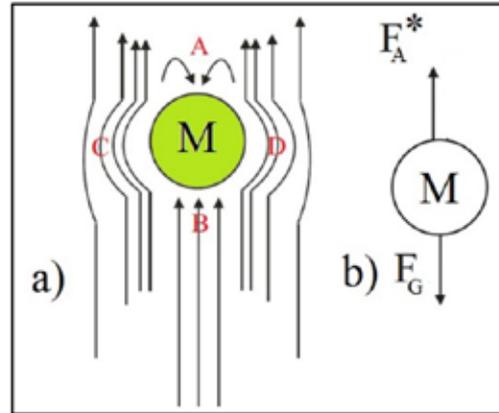
Uno de los problemas clásicos de la física se refiere a un objeto esférico que se mueve verticalmente a través de un medio resistivo. Por ejemplo, en una bala de cañón disparada en el aire, o en un grano de arena que se hunde lentamente en el fondo de un lago. La fuerza resistiva o de arrastre de aire (F_A) puede ser lineal o cuadrática a la velocidad, pero ambos dependen del número de Reynolds (Timmerman & Van der Weele, 1999). Su fórmula general es expresada por la ecuación (1).

$$F_A = \frac{1}{2} C_d \cdot \rho_A \cdot A \cdot v_A^2 \quad (1)$$

Donde, C_d coeficiente de arrastre, ρ_A densidad del aire, v_A la velocidad media de la corriente de aire y A es el área de la sección transversal que se enfrenta a la corriente de aire (Kuzhandairaj, 2018).

Este sistema puede ser modelado por un simple cuerpo o esfera que se desplaza en el seno de un gas. En los puntos A y B (Figura 1a), la velocidad de flujo de aire es cero, en C y D existe una velocidad mayor debido a la lejanía de la esfera. Hay una diferencia de presión entre las regiones alejadas (baja presión) y el centro de la corriente de aire (alta presión). Al desplazarse, la esfera experimenta presión debido al aire en reposo (punto B) que es mayor que el aire en movimiento (puntos C y D). La presión crea una fuerza neta sobre la esfera que lo mantiene en el centro de la corriente impidiendo que descienda (Escaño et al., 2004).

Figura 1. a) Líneas de flujo de aire sobre la esfera. b) Diagrama de cuerpo libre de la esfera.



Fuente: Elaboración propia basado en la dinámica de Fluidos. 2020.

La elevación aerodinámica se puede modelar utilizando una relación de fuerza de arrastre estándar, modificando la ecuación (1) (Jernigan et al., 2008), teniendo la ecuación (2):

$$F_A^* = \frac{1}{2} C_d \cdot \rho_A \cdot A \cdot (v_A - \frac{dy}{dt})^2 \quad (2)$$

Donde “ dy/dt ” es la velocidad absoluta de la esfera. La velocidad media del aire “ v_A ” está en función de la altura sobre la salida de la boquilla y la velocidad de rotación del ventilador (Jernigan et al., 2008). Con la segunda ley de Newton, se tiene la ecuación dinámica para el sistema de levitación de aire. Las únicas fuerzas que actúan sobre el objeto levitante (la esfera) son el efecto hacia arriba del flujo de aire, dado por el primer término de la ecuación (3), además de ser representado por la flecha que va hacia arriba en la Figura 1b. El efecto hacia abajo por la fuerza de la gravedad ($F_G=mg$); segundo término de la ecuación (3) y representado por la flecha que baja en la Figura 1b (Chacon et al., 2017). Por tanto, de lo expuesto se tiene la ecuación (3):

$$\sum F_y = m \frac{d^2y}{dt^2} \\ \frac{1}{2} C_d \cdot \rho_A \cdot A \cdot (v_A - \frac{dy}{dt})^2 - mg = m \frac{d^2y}{dt^2} \quad (3)$$

Donde " V_A " es la variable de control y la variable " y " representa la variable de salida o la altura de la esfera levitada. Como se observa la ecuación (3) es una ecuación diferencial no lineal. El coeficiente de arrastre C_d es un término que depende del número de Reynold, que, a su vez, depende de la velocidad relativa entre el objeto que se mueve dentro de un flujo y la velocidad de dicho flujo. Para velocidades pequeñas, se puede suponer que C_d es constante (Chacon et al., 2017). El movimiento de la esfera en el eje horizontal a medida que asciende es descontrolado. Si se desplaza a un lado aparecen fuerzas resistivas de fricción de diferente magnitud en los puntos C y D (Figura 1b), esta diferencia provoca un giro en la esfera de velocidad W tal como se muestra en la ecuación (4) (Escaño et al., 2004):

$$\sum M = I\omega$$

$$M_C - M_D = I\omega \quad (4)$$

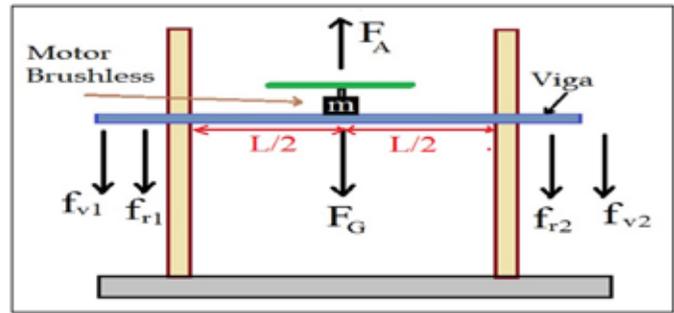
Donde I es el momento de inercia de la esfera, y W la velocidad angular. Al girar la esfera, parte de la energía potencial adquirida por el flujo de aire se convierte en energía cinética de rotación, y otra en calor de fricción, esto provoca un descenso de la pelota. El efecto Magnus (Cuando la rotación de un objeto afecta a la trayectoria del mismo a través de un fluido) aparece, por tanto, es difícil elevar la esfera con pequeños aumentos de velocidad. Se observa que el desplazamiento de la esfera es bastante complejo y su ecuación tiene muchas variables que hacen que un modelo exacto sea difícil de obtener (Escaño et al., 2004).

1.2 Modelo matemático del sistema de levitación aerodinámica de una viga

La viga levitada se muestra en la Figura 2. La fuerza de empuje (F_A) producido por el motor Brushless (motor eléctrico sin escobillas) está representada por la ecuación (2). El sistema presenta la fuerza de la gravedad producida por la masa del motor

(la masa de la viga es despreciable), las fricciones de Coulomb (f_{r1}, f_{r2}), y las fricciones viscosas (f_{v1}, f_{v2}) (Kuo, 2010) en las pilares de sujeción de la viga.

Figura N° 2. Sistema de levitación aerodinámica de una viga con diagrama de fuerzas



Fuente: Elaboración propia. 2020.

Si se aplica la segunda ley de Newton se tiene la ecuación (5):

$$\sum F_y = m \frac{d^2y}{dt^2}$$

$$F_A - F_G - f_{r1} - f_{r2} - f_{v1} - f_{v2} = m \frac{d^2y}{dt^2} \quad (5)$$

Se obtiene la ecuación (6):

$$\frac{1}{2} C_d \cdot \rho_A \cdot A \cdot (v_A - \frac{dy}{dt})^2 - mg - (F_{c1} + F_{c2}) \frac{dy}{|dy|} - (B_1 + B_2) \frac{dy}{dt} = m \frac{d^2y}{dt^2} \quad (6)$$

Donde:

F_{c1}, F_{c2} : Coeficiente de fricción de Coulomb.

B_1, B_2 : Coeficiente de fricción viscosa.

A : Área del disco formado por la rotación de las hélices del motor Brushless.

v_A : Velocidad del aire producido por el motor Brushless.

La variable de control de la ecuación (6) es la velocidad del aire " v_A ", y la variable de salida " y " es la altura de la viga. Se observa que la ecuación diferencial del sistema no es lineal, además la viga puede presentar una inclinación al levitar, ecuación (7):

$$\sum M = I\omega$$

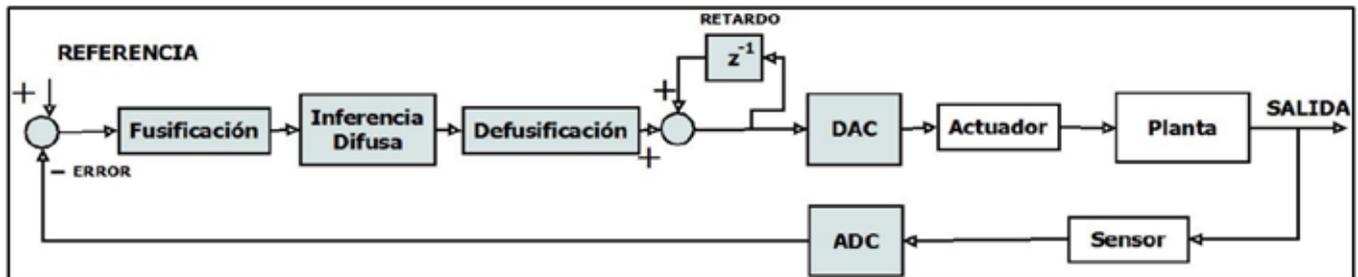
$$\frac{L}{2} \left[(B_1 - B_2) \frac{dy}{dt} + (F_{C1} - F_{C2}) \frac{\left(\frac{dy}{dt}\right)}{\left|\frac{dy}{dt}\right|} \right] = I\omega \quad (7)$$

Para evitar la inclinación en la levitación la ecuación (7) muestra que los coeficientes de fricción deben ser iguales en ambos pilares.

2. Sistema de control difuso

El sistema de control difuso empleado es el mostrado en la Figura 3. Los componentes del sistema de control se dividen en analógico y digital. El analógico está compuesto por el actuador, la planta y los sensores analógicos. En digital, se usa los DAC (Digital to Analogue Converter) y ADC (Analog to Digital Converter) como interfaces entre las dos divisiones y, propiamente, el control difuso.

Figura N° 3. Diagrama de bloques del sistema de control difuso usado



Fuente: Elaboración propia. 2020.

2.1 Actuador en el sistema de levitación aerodinámica de una esfera

El actuador usado para la generación de flujo de aire son dos ventiladores helicoidales/axiales con siete aspas mostrado en la Figura 4. Sus características se detallan en la Tabla 1, conectados uno sobre el otro para obtener mayor fuerza aerodinámica.

Figura N°4. Ventilador JF0825-1H-00



Fuente: Chipdip. 2020.

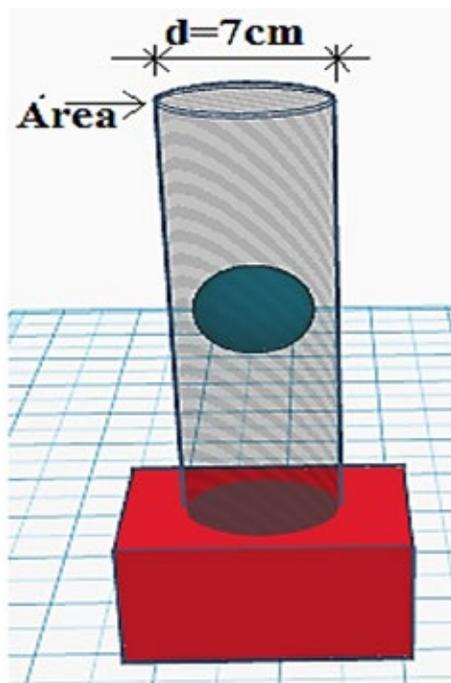
La velocidad de flujo de aire está dada por la ecuación (8):

$$q_0 = A \cdot v_A \quad (8)$$

Donde A es el área del tubo del sistema mostrado en la Figura 5, v_A es la velocidad de flujo de aire en el tubo del sistema. El diámetro del tubo del sistema es 7 [cm] (el área debe ser de un círculo $A = \pi/4 d^2$), el caudal según la Tabla 1 es 0.01828 [m³/s], entonces la velocidad máxima del sistema con un ventilador es:

$$v_{A1} = 4.75 \text{ [m/s]} \quad \text{o} \quad 17.1 \text{ [km/h]}$$

Figura N°5. Dimensiones del tubo del sistema de levitación aerodinámica



Fuente: Elaboración propia. 2020.

Tabla N°1. Características del ventilador

| | |
|--------------------------------------|--|
| Modelo | JF0825-1H-00 |
| Voltaje nominal | 12V |
| Corriente nominal | 0.19 A |
| Rango de voltaje de operación | 10.2V – 13.8V |
| Velocidad | 3000RPM |
| Velocidad de flujo de aire (q_0) | 38.74 CFM/ 0.01828 [m ³ /s] |
| Dimensiones | 80x80x25mm |
| Ruido | 31.0 dBA |

Fuente: Jamicon. 2020.

Para un sistema de dos ventiladores instalados uno encima del otro (Bastianon, 2008) se obtiene la ecuación (9):

$$v_A = \sqrt{(4.75)^2 + (4.75)^2}$$

$$v_{A \max} = 6.72 \text{ [m/s]} \quad \text{o} \quad 24.2 \text{ [km/h]} \quad (9)$$

2.2 Actuador del sistema de levitación aerodinámica de una viga

El actuador usado para generar el impulso de flujo de aire al sistema es un motor Brushless mostrado en la Figura 6. Sus características se muestran en la Tabla 2. La hélice (APC 10 x 5 E) usado tiene un diámetro de 25.4 cm.

Figura N°6. Motor Brushless



Fuente: HETPRO. 2020.

Tabla N°2. Características del motor Brushless

| | |
|-------------------------|--------------------|
| Modelo | A2212/13T |
| KV | 1000 RPM/V |
| Peso | 47g |
| Voltaje | 2 – 3s <u>LiPo</u> |
| Corriente máxima | 15 ^a |
| APC 10 x 5 E | 12.15V |

Fuente: HETPRO. 2020.

La relación de velocidades λ (Bastianon, 2008), se define como la relación de velocidad tangencial ($R \cdot \omega$) del extremo de las hélices y la velocidad de flujo de aire, ecuación (10):

$$\lambda = \frac{R \cdot \omega}{V_A} \quad (10)$$

La velocidad máxima del motor Brushless a 12V, ecuación (11), según la Tabla 2.

$$n_{max} = 12000 [RPM] \quad (11)$$

El resultado de la ecuación (11) debe ser expresado en [rad/s], ecuación (12):

$$\omega_{max} = 1257 [\text{rad/s}] \quad (12)$$

La ecuación (10) se modifica con estos resultados, considerando diámetro de la hélice (Tabla 2) de 25.4 [cm], ecuación (13):

$$V_{A \max} = \frac{159.6}{\lambda} [\text{m/s}] \quad \text{o} \quad \frac{574.6}{\lambda} [\text{km/h}] \quad (13)$$

Donde V_A es la velocidad de flujo de aire. La relación de velocidades λ de la hélice de doble pala está entre 4 a 7 (Bastianon, 2008), se asume en este sistema un valor intermedio de 5.5 (Bastianon, 2008), por tanto, de la ecuación (13) se deduce la ecuación (14):

$$V_{A \max} = 29 [\text{m/s}] \quad \text{o} \quad 104 [\text{km/h}] \quad (14)$$

2.3 Simulación del modelo matemático del sistema levitador aerodinámico de una esfera

Se parte de la ecuación (3) y se despeja la aceleración de la esfera, ecuación (15):

$$\frac{d^2y}{dt^2} = \frac{1}{2} \frac{C_d \cdot \rho_A \cdot A}{m} \cdot \left(v_A - \frac{dy}{dt} \right)^2 - g \quad (15)$$

Se realiza cambios de variable, ecuación (16):

$$\frac{d^2y}{dt^2} = k_2 \left(k_1 * DAC - \frac{dy}{dt} \right)^2 - g \quad (16)$$

Donde:

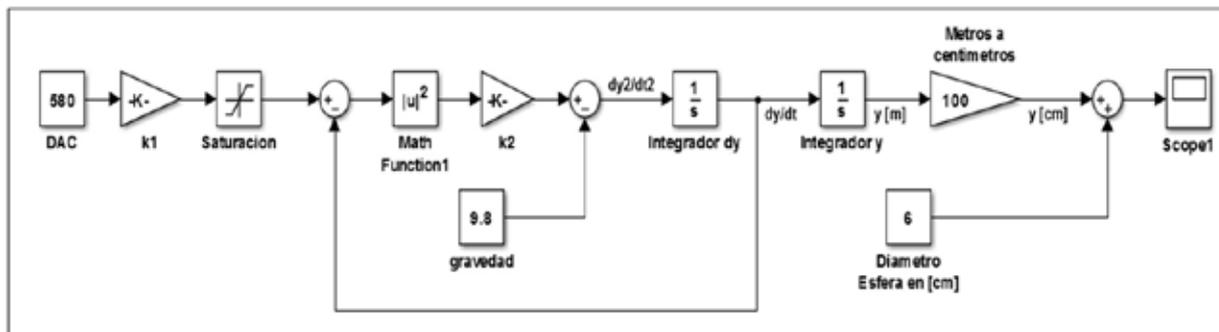
$$k_2 = \frac{1}{2} \frac{C_d \cdot \rho_A \cdot A}{m} \quad v_A = k_1 * DAC$$

DAC: Valor PWM de salida del controlador digital (por siglas en inglés de pulse-width modulation – modulación por ancho de pulso). k_1 Es el factor de conversión entre el valor DAC (PWM) y la velocidad real del empuje de aire (v_A)

Esta ecuación diferencial no lineal se implementa en SIMULINK, como se muestra en la Figura 7, tomando en cuenta la modulación PWM usada en la variable de control. Se usa la saturación como la velocidad máxima de la ecuación (9). El valor de la constante k_1 es dada por la ecuación (17):

$$k_1 = \frac{v_{A \max}}{PWM_{\max}} = \frac{1}{152.23} \quad (17)$$

Figura N°7. Diagrama de bloques de la ecuación diferencial del sistema de levitación aerodinámica de la esfera

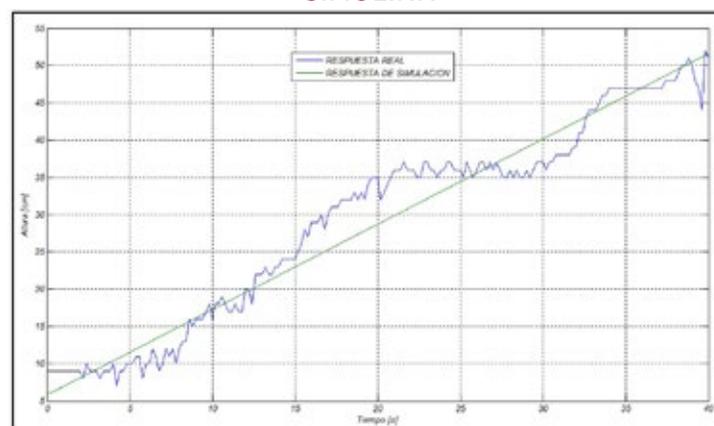


Fuente: Elaboración propia. 2020.

La constante k_2 : $C_d=0.4$ (Timmerman & Van der Weele, 1999), $\rho_A=1.2$ [kg/m³] (Jernigan et al., 2008), el área de la esfera de plastofomo ($\rho=10$ [kg/m³]) de diámetro de 6.5 [cm] es $A=3.32 \cdot 10^{-3}$ [m²], la masa de la esfera de plastofomo es $m=1.13$ [g], se reemplaza en la ecuación (16) $k_2=0.5533$, el valor usado en SIMULINK es un valor arbitrario de ajuste $k_2=0.6792$.

Se usa el ARDUINO UNO como tarjeta de adquisición de datos, como sensor es usado el de ultrasonido, HC-SR04, para la medición de la distancia. La respuesta real del sistema en lazo abierto y la respuesta de simulación en lazo abierto se muestran en la Figura 8. Se observa en la gráfica que existen puntos donde la esfera presenta una rotación sobre su eje explicado en la ecuación (4).

Figura N°8. Respuesta del sistema en lazo abierto real y el sistema del modelo matemático en SIMULINK



Fuente: Elaboración propia. 2020.

2.4 Simulación del modelo matemático del sistema levitador aerodinámico de una viga

Se parte de la ecuación (6) se despeja la aceleración de la viga y se realiza cambios de variable, ecuación (18):

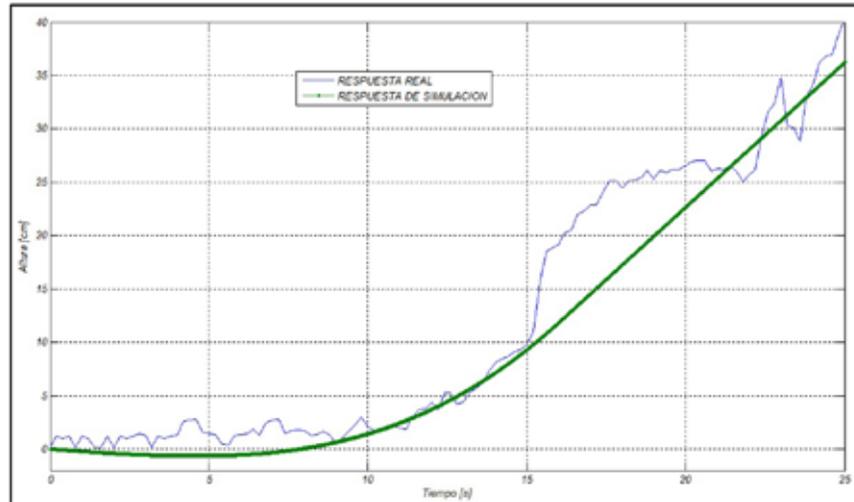
$$\frac{d^2y}{dt^2} = K_2 \cdot \left(v_A - \frac{dy}{dt}\right)^2 - g - (K_{FC1} + K_{FC2}) \frac{\left(\frac{dy}{dt}\right)}{\left|\frac{dy}{dt}\right|} - (K_{B1} + K_{B2}) \frac{dy}{dt} \quad (18)$$

Donde:

$$v_A = K_1 * DAC = \frac{29}{1023} DAC \quad K_2 = \frac{1}{2} \frac{C_d \cdot \rho_A \cdot A}{m} = 0.2323$$

Y las constantes tienen valores arbitrarios de ajuste realizados en SIMULINK; comparado con la respuesta real, ver Figura 9, en donde como tarjeta de adquisición de datos se usó el ARDUINO UNO, y como sensor infrarrojo SHARP GP2Y0A21.

Figura N°9. Respuesta del modelo matemático y la respuesta real del sistema



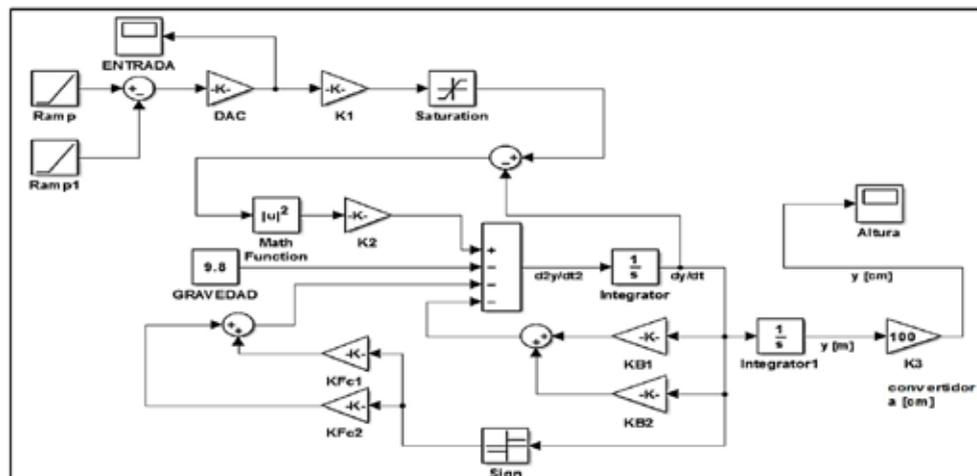
Fuente: Elaboración propia. 2020.

$$K_{FC1} = \frac{F_{C1}}{m} = 0.5 \quad K_{FC2} = \frac{F_{C2}}{m} = 0.5$$

$$K_{B1} = \frac{B_1}{m} = 2000 \quad K_{B2} = \frac{B_2}{m} = 2000$$

La ecuación diferencial no lineal de la ecuación (18) se muestra en la Figura 10.

Figura N°10. Diagrama de bloques de la ecuación diferencial del sistema de levitación aerodinámica de la viga

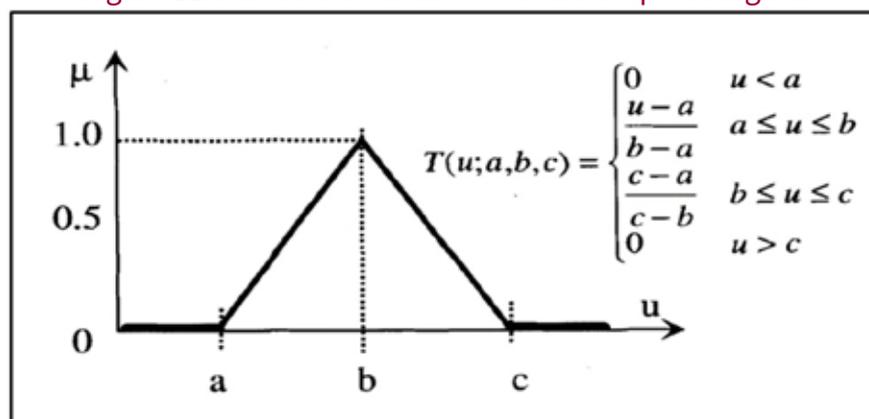


Fuente: Elaboración propia. 2020.

3. Diseño del control difuso usando MATLAB

La función de tipo T o triangular es usada en este diseño. La función triangular tiene la propiedad de modelar con un valor de inclusión distinto de cero para un rango de valores estrecho en torno a un punto “b” (Del Brío & Molina, 2002; Sivanandam et al., 2007) como se muestra en la Figura 11.

Figura 11. Función de membresía difusa tipo triangular



Fuente: Del Brío & Molina, 2002

3.1 Variable lingüística de entrada

La variable lingüística de entrada es el ERROR (valor de referencia menos el valor real), y sus particiones difusas son tres: NEGATIVO, CERO y POSITIVO. NEGATIVO es cuando el valor de referencia es menor al valor real del sistema. CERO cuando el valor de referencia y el valor real del sistema son iguales. POSITIVO cuando el valor de referencia es mayor al valor real del sistema. El conjunto difuso de entrada se puede representar mediante la ecuación (19):

$$\text{ERROR} = \{\text{NEGATIVO}, \text{CERO}, \text{POSITIVO}\} \quad (19)$$

Las particiones difusas dependen del tipo de sistema y la velocidad de respuesta del actuador, son mostradas en la Tabla 3. Las particiones se expresan mediante tres números: [a,b,c] siguiendo el criterio de la Figura 11, por ejemplo: NEGATIVO: [a=-60,b=-60,c=-15].

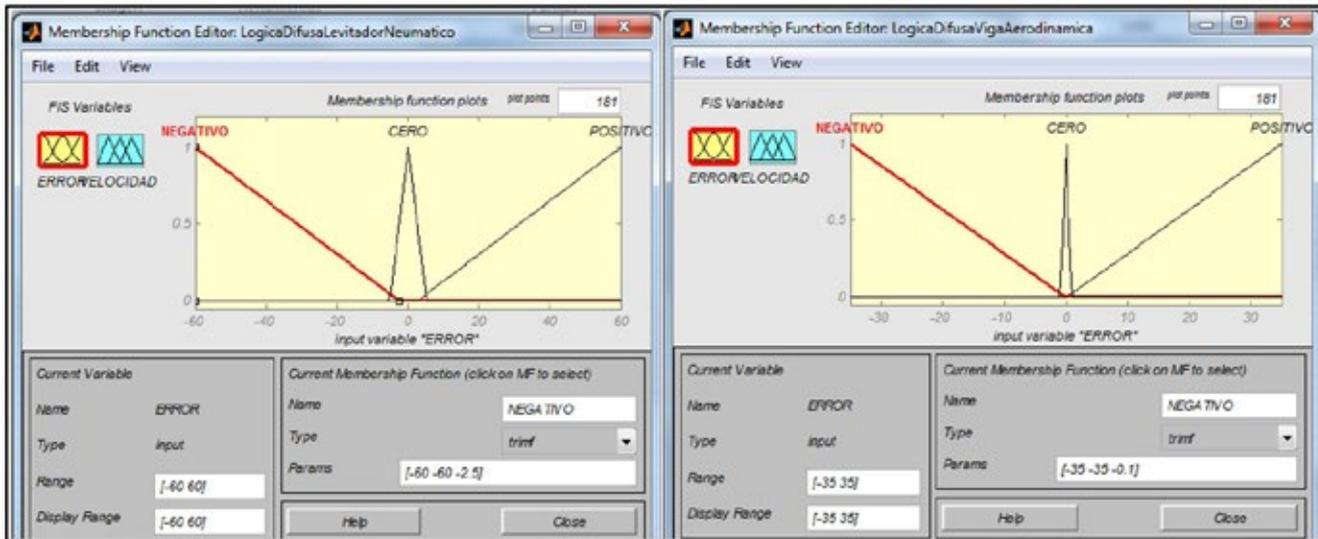
Tabla N°3. Valores de las particiones NEGATIVO, POSITIVO Y CERO de los sistemas de los aerodinámicos

| ERROR | Sistema de levitador neumático | Sistema viga aerodinámica |
|----------|--------------------------------|---------------------------|
| NEGATIVO | [-60 -60 -2.5] | [-35 -35 -0.1] |
| CERO | [-5 0 5] | [-1 0 1] |
| POSITIVO | [2.5 60 60] | [0.1 35 35] |

Fuente: Elaboración propia. 2020.

Para un error en estado estacionario pequeño, los valores de las particiones NEGATIVO y POSITIVO de los extremos derecho e izquierdo consecutivamente deben estar lo más cercano posible a cero como se muestra en la Figura 12.

Figura N°12. Valores de las particiones en el sistema de levitación aerodinámica de una esfera (izquierda) y de una viga (derecha) usando fuzzy de MATLAB de la variable lingüística de entrada



Fuente: Elaboración propia. 2020.

3.2 Variable lingüística de salida

La variable lingüística de salida es el grado de incremento o disminución de la VELOCIDAD. Esta variable define el tiempo de asentamiento del sistema. Las particiones dependen del conocimiento de funcionamiento del sistema y la velocidad de respuesta de sus actuadores (Del Brío & Molina, 2002). El conjunto difuso de salida se representa mediante la ecuación (20):

$$VELOCIDAD = \{DISMINUIR, MANTENER, INCREMENTAR\} \quad (20)$$

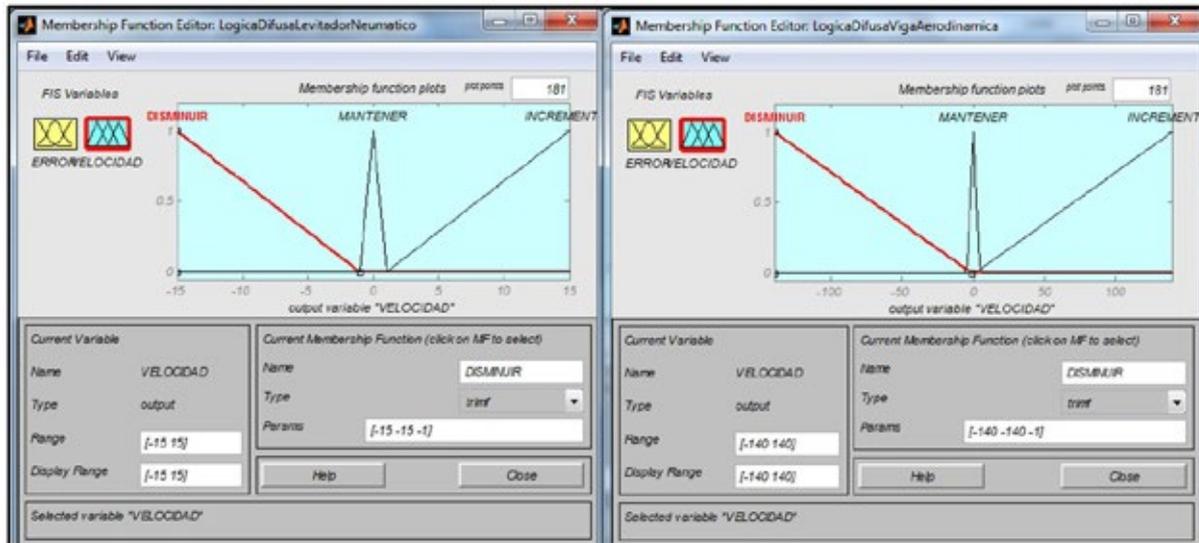
Las particiones difusas: DISMINUIR reduce la velocidad, INCREMENTAR aumenta la velocidad, y MANTENER mantiene la velocidad se muestran en la Tabla 4 y en la Figura 13.

Tabla N°4. Valores de las particiones DISMINUIR, MANTENER, INCREMENTAR de los sistemas de los aerodinámicos

| VELOCIDAD | Sistema de levitador neumático | Sistema viga aerodinámica |
|-------------|--------------------------------|---------------------------|
| DISMINUIR | [-15 -15 -1] | [-140 -140 -1] |
| MANTENER | [-1 0 1] | [-5 0 5] |
| INCREMENTAR | [1 15 15] | [1 140 140] |

Fuente: Elaboración propia. 2020.

Figura 13. Valores de las particiones en el sistema de levitación aerodinámica de una esfera (izquierda) y de una viga (derecha) usando fuzzy de MATLAB de la variable lingüística de salida



Fuente: Elaboración propia. 2020.

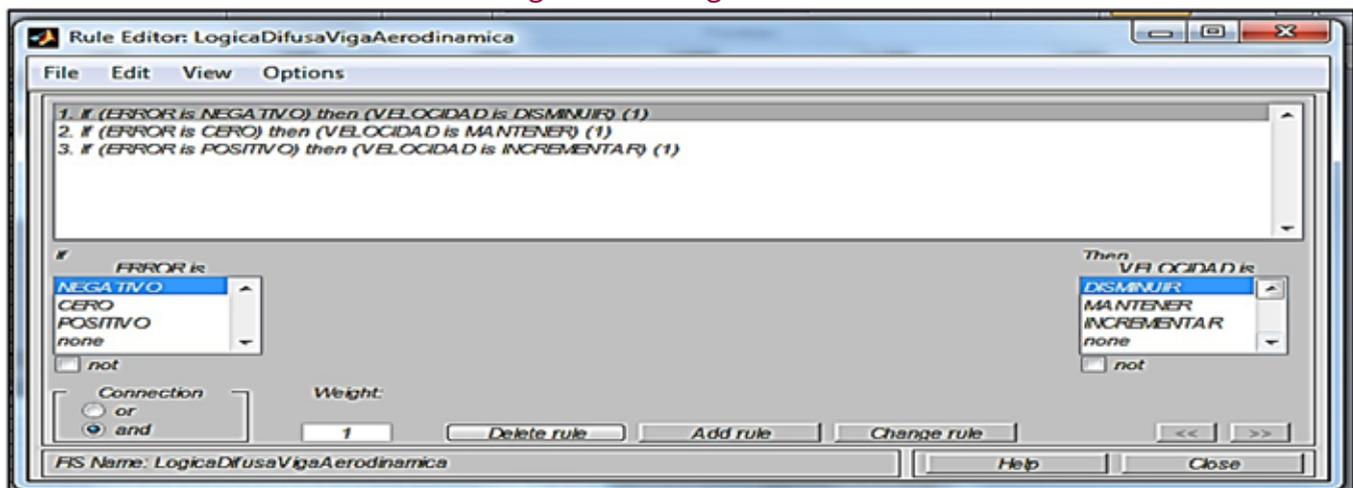
3.3 Reglas difusas

Las reglas difusas siguen el siguiente razonamiento:

- Si el error es NEGATIVO se debe DISMINUIR la velocidad.
- Si el error es POSITIVO se debe INCREMENTAR la velocidad.
- Si el error es CERO se debe MANTENER la velocidad.

Estas reglas se muestran en la Figura 14, son las mismas para ambos sistemas.

Figura N°14. Reglas difusas

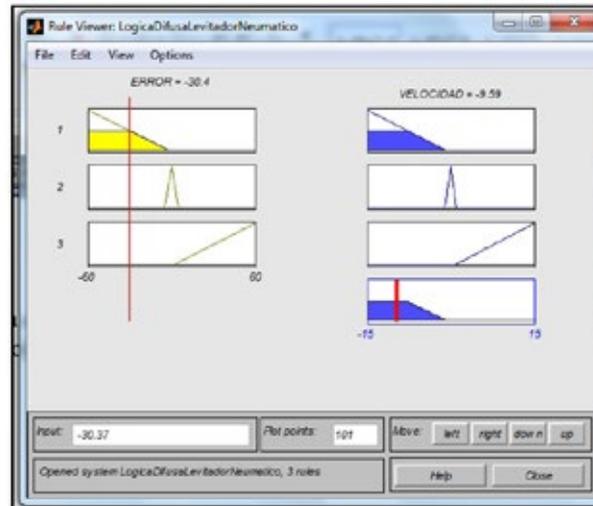


Fuente: Elaboración propia. 2020.

3.4 Fusificación, inferencia difusa y defusificación

La fusificación convierte los valores reales de entrada (ERROR=-30.4) en valores difusos (grado de pertenencia a un conjunto difuso). La inferencia aplica las reglas difusas definidas entre las variables lingüísticas de entrada (ERROR) y salida (VELOCIDAD) usando el método de Mamdani. La defusificación convierte los resultados obtenidos en las reglas difusas en valores adecuados (VELOCIDAD=-9.54) para el control usando el método de defusificación centroide (Del Brío & Molina, 2002) Todo este proceso se muestra en la Figura 15, realizado en MATLAB.

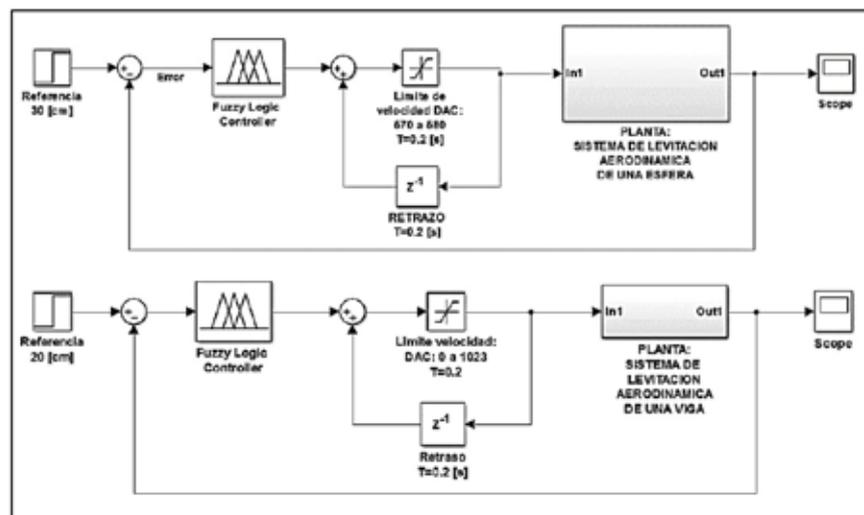
Figura N°15. Etapas de la lógica difusa en MATLAB



Fuente: Elaboración propia. 2020.

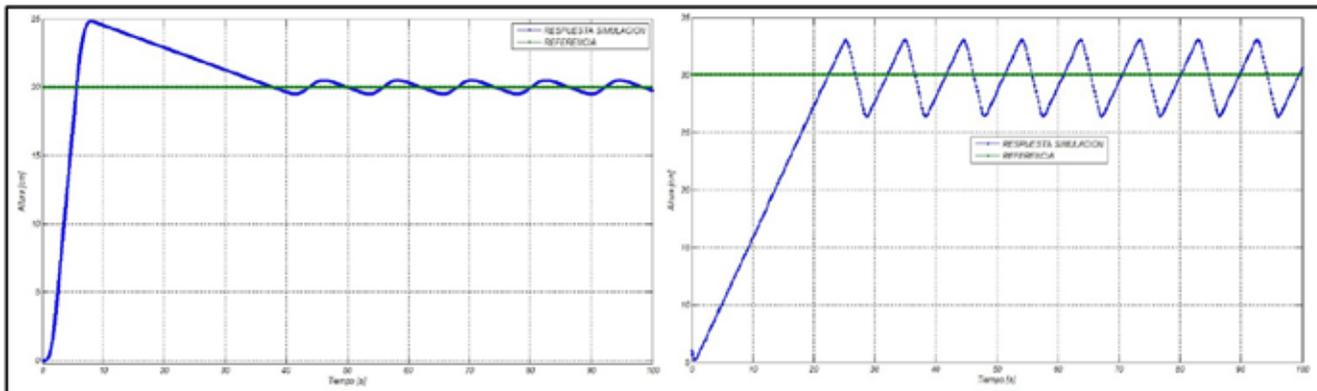
La implementación del control difuso en MATLAB se muestra en la Figura 16 para los dos sistemas. Se incluye el modelo encontrado en s ecuaciones (16) y (18). Sus respuestas transitoria y permanente se muestran en la Figura 17 para los dos sistemas de levitación aerodinámica de viga y esfera. El tiempo de estabilidad aproximado del sistema de levitación aerodinámica de una viga es 40 segundos y de la esfera es 25 segundos.

Figura N°16. Sistema de control difuso en SIMULINK del sistema levitación aerodinámica de la esfera y viga



Fuente: Elaboración propia. 2020.

Figura N°17. Respuesta del sistema de levitación aerodinámica de una viga (izquierda) y una esfera (derecha)

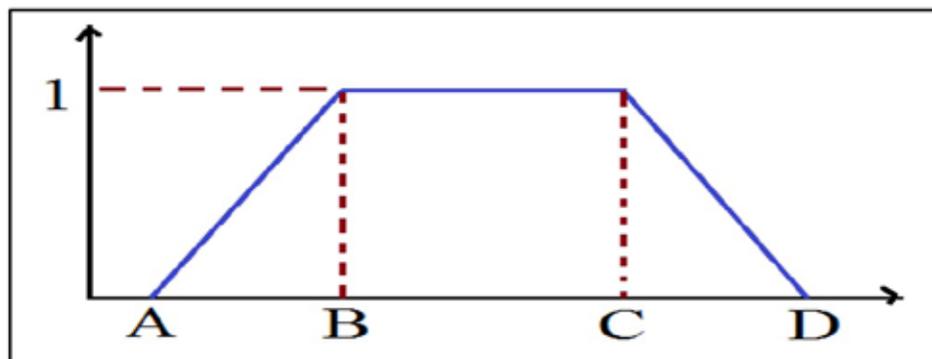


Fuente: Elaboración propia. 2020.

4. Implementación del control difuso en ARDUINO

Se usa la biblioteca prediseñada para ARDUINO de lógica difusa: “Fuzzy.h” y debe ser incluida desde su plataforma de inclusión (Artero, 2013). Para la asignación de las particiones de cada conjunto difuso, se sigue el criterio mostrado en la Figura 18. Las particiones corresponden a la función de pertenencia trapezoidal, para obtener una función de pertenencia triangular simplemente los valores “B” y “C” deben tener el mismo valor.

Figura N°18. Función de pertenencia trapezoidal



Fuente: Elaboración propia. 2020.

Los valores de las particiones empleados y sintonizados figuran en las Tablas 5 y 6. Existen ligeros ajustes en las particiones de la variable lingüística de ERROR de la Tabla 5, debido a la pérdida en energía cinética de la esfera en la práctica. Para el sistema de levitación aerodinámica de una viga, el controlador del motor Brushless necesita responder a un incremento PWM de unidades.

Tabla N°5. Valores de las particiones en el sistema de levitación aerodinámica de una esfera

| Variable lingüística | Sistema de levitación aerodinámica de una esfera | |
|------------------------|--|----------------------------|
| | Partición usada en SIMULINK | Partición usada en ARDUINO |
| ERROR: NEGATIVO | [-60 -60 -2.5] | [-60 -60 -5] |
| ERROR: CERO | [-5 0 5] | [-5 0 5] |
| ERROR: POSITIVO | [2.5 60 60] | [5 60 60] |
| VELOCIDAD: DISMINUIR | [-15 -15 -1] | [-15 -15 -1] |
| VELOCIDAD: MANTENER | [-1 0 1] | [-1 0 1] |
| VELOCIDAD: INCREMENTAR | [1 15 15] | [1 15 15] |

Fuente: Elaboración propia. 2020.

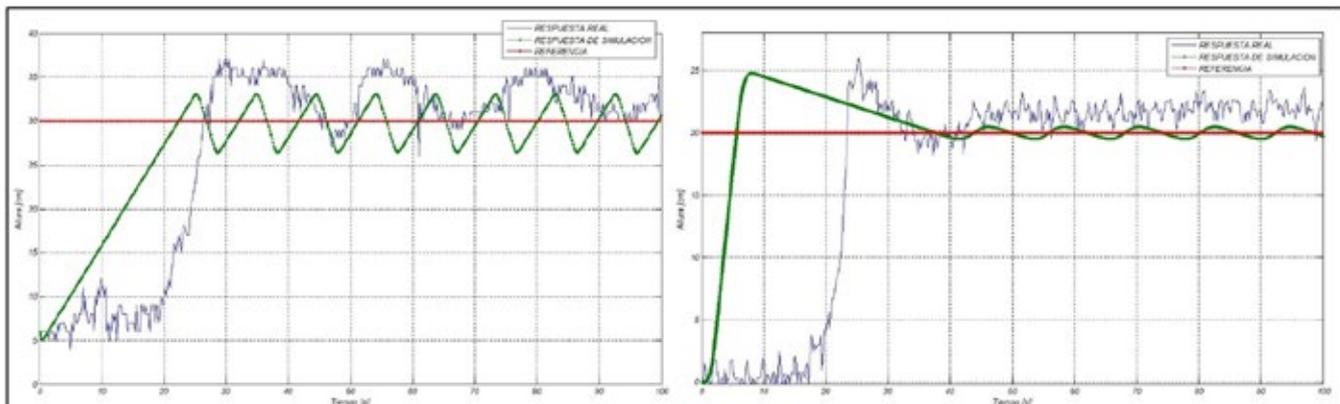
Tabla N°6. Valores de las particiones en el sistema de levitación aerodinámica de una viga

| Variable lingüística | Sistema de levitación aerodinámica de una viga | |
|------------------------|--|----------------------------|
| | Partición usada en SIMULINK | Partición usada en ARDUINO |
| ERROR: NEGATIVO | [-35 -35 -0.1] | [-35 -35 -5] |
| ERROR: CERO | [-1 0 1] | [-5 0 5] |
| ERROR: POSITIVO | [0.1 35 35] | [5 35 35] |
| VELOCIDAD: DISMINUIR | [-140 -140 -1] | [-10 -10 -1] |
| VELOCIDAD: MANTENER | [-5 0 5] | [-5 0 5] |
| VELOCIDAD: INCREMENTAR | [1 140 140] | [1 10 10] |

Fuente: Elaboración propia. 2020.

El incremento de decenas provoca fallas en el controlador y evita el funcionamiento correcto del motor Brushless. Por este motivo se cambia en la implementación los límites inferior y superior en los valores de las particiones de la variable lingüística VELOCIDAD como se muestra en la Tabla 6.

Figura N°19. Respuesta de simulación y real del sistema de levitación aerodinámica de una esfera (izquierda) y una viga (derecha)



Fuente: Elaboración propia. 2020.

El cambio de los límites a valores de unidades causará un incremento en el tiempo de crecimiento del sistema. El tiempo de estabilidad se mantendrá debido a que sus límites vecinos no se cambiaron. Los límites vecinos de las particiones de la variable lingüística ERROR también fueron sintonizados, ver Tabla 6. La asignación de variables lingüísticas de entrada y salida es lo mismo para cada sistema, se muestra en la Figura 20.

Figura N°20. Asignación de variables lingüísticas de entrada y salida

| | |
|--|---|
| <pre> /// PRIMERA ENTRADA: ERROR DE ENTRADA FuzzyInput *ERROR_ENTRADA = new FuzzyInput(1); //// primera variable error negativo FuzzySet *NEGATIVO = new FuzzySet(P_A1,P_B1,P_C1,P_D1); ERROR_ENTRADA->addFuzzySet(NEGATIVO); //// segunda variable error cero FuzzySet *CERO = new FuzzySet(P_A2,P_B2,P_C2,P_D2); ERROR_ENTRADA->addFuzzySet(CERO); //// tercera variable error positivo FuzzySet *POSITIVO = new FuzzySet(P_A3,P_B3,P_C3,P_D3); ERROR_ENTRADA->addFuzzySet(POSITIVO); //guardando en la variable lingüística fuzzy->addFuzzyInput(ERROR_ENTRADA); </pre> | <pre> //////////////////////////////////// // SALIDA DEL SISTEMA: VALOR DE VELOCIDAD FuzzyOutput *VELOCIDAD = new FuzzyOutput(1); //// primera condicion de salida FuzzySet *DISMINUIR = new FuzzySet(P_A4,P_B4,P_C4,P_D4); VELOCIDAD ->addFuzzySet(DISMINUIR); //// segunda condicion de salida FuzzySet *MANTENER = new FuzzySet(P_A5,P_B5,P_C5,P_D5); VELOCIDAD ->addFuzzySet(MANTENER); //// tercera condicion de salida FuzzySet *INCREMENTAR = new FuzzySet(P_A6,P_B6,P_C6,P_D6); VELOCIDAD ->addFuzzySet(INCREMENTAR); //guardando en la variable lingüística fuzzy->addFuzzyOutput(VELOCIDAD); </pre> |
|--|---|

Fuente: Elaboración propia. 2020.

La regla de fusificación se sigue conforme a lo mostrado en la Figura 14. El código equivalente en ARDUINO también debe cumplir las tres reglas como se muestra en la Figura 21.

Figura N°21. Reglas de fusificación en ARDUINO

| | |
|--|---|
| <pre> //// PRIMERA REGLA: FuzzyRuleAntecedent *ifErrorNegativo = new FuzzyRuleAntecedent(); ifErrorNegativo->joinSingle(NEGATIVO); FuzzyRuleConsequent *thenDecrementar = new FuzzyRuleConsequent(); thenDecrementar->addOutput(DISMINUIR); FuzzyRule *REGLA01 = new FuzzyRule(1, ifErrorNegativo, thenDecrementar); fuzzy->addFuzzyRule(REGLA01); </pre> | <pre> //// SEGUNDA REGLA: FuzzyRuleAntecedent *ifCero = new FuzzyRuleAntecedent(); ifCero->joinSingle(CERO); FuzzyRuleConsequent *thenMantener = new FuzzyRuleConsequent(); thenMantener->addOutput(MANTENER); FuzzyRule *REGLA02 = new FuzzyRule(1, ifCero, thenMantener); fuzzy->addFuzzyRule(REGLA02); </pre> |
| <pre> //// TERCERA REGLA: FuzzyRuleAntecedent *ifPositivo = new FuzzyRuleAntecedent(); ifPositivo->joinSingle(POSITIVO); FuzzyRuleConsequent *thenIncrementar = new FuzzyRuleConsequent(); thenIncrementar->addOutput(INCREMENTAR); FuzzyRule *REGLA03 = new FuzzyRule(1, ifPositivo, thenIncrementar); fuzzy->addFuzzyRule(REGLA03); </pre> | |

Fuente: Elaboración propia. 2020.

La fusificación, la inferencia difusa y la defusificación se realizan en ARDUINO cada 200[ms] manteniendo un tiempo de muestreo constante. Se calcula primero el error midiendo la distancia, se envía este valor para ser fusificado, luego a la inferencia difusa para terminar en la defusificación. Lo explicado se muestra en la Figura 22.

Figura N°22. Fusificación, inferencia difusa y defusificación.

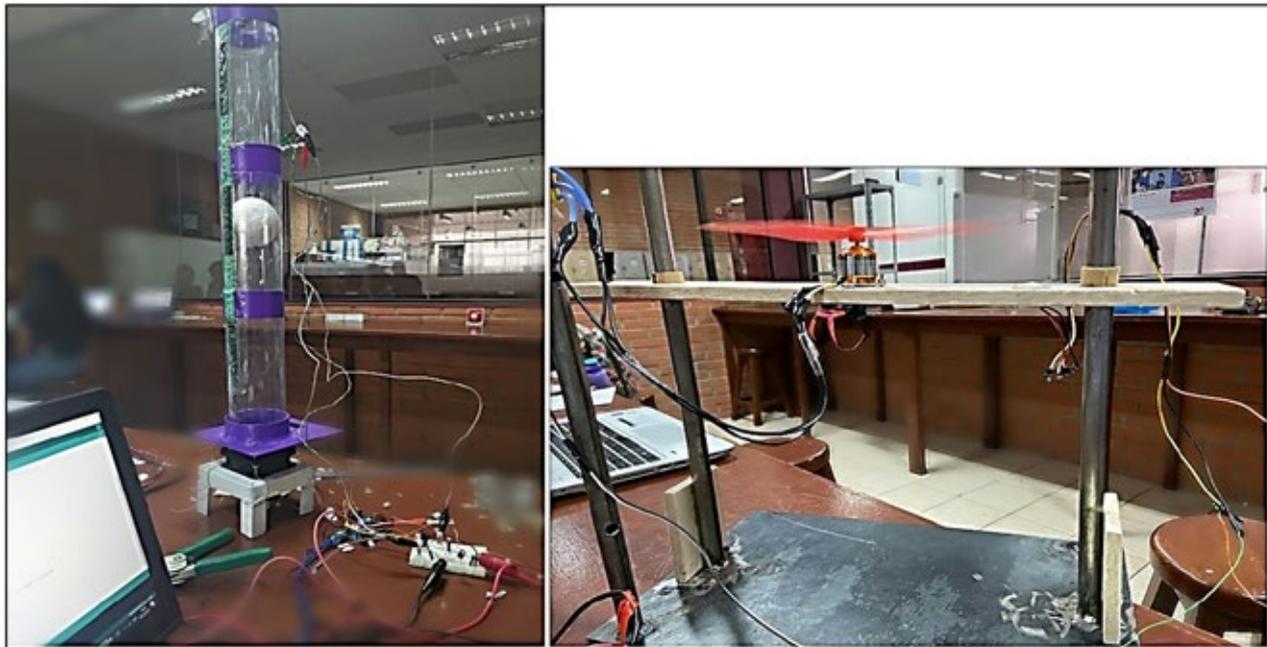
```
fuzzy->setInput(1, Error);  
// Se realiza la Fusificacion  
fuzzy->fuzzify();  
// Resuelve con la Defusificacion  
float salida = fuzzy->defuzzify(1);
```

Fuente: Elaboración propia. 2020.

RESULTADOS DEL SISTEMA DE CONTROL DIFUSO

Los resultados del sistema de levitación aerodinámica de una esfera y una viga, simulados y reales, se muestran en la Figura 19. Los sistemas físicos se muestran en la Figura 23.

Figura N°23. Sistema de levitación aerodinámica de una esfera (izquierda) y una viga (derecha)



Fuente: Elaboración propia. 2020.

CONCLUSIONES

En el presente artículo se llegaron a las siguientes conclusiones:

- El cálculo de los modelos matemáticos de cada sistema de levitadores aerodinámicos estudiados ayuda a predecir el comportamiento real del sistema. La ecuación diferencial no lineal de los sistemas levitadores aerodinámicos son fáciles de modelar en SIMULINK de MATLAB, con previo ajuste de sus constantes comparando con datos reales en lazo abierto.
- Para lograr determinar los valores adecuados de las particiones difusas de cada variable lingüística, se usa el modelo matemático de cada sistema aerodinámico implementado en SIMULINK de MATLAB. Su respuesta transitoria y en estado estacionario en simulación facilita el diseño de controladores difusos.
- La implementación de un control difuso facilita el control de un sistema levitador aerodinámico, la defusificación se encarga de realizar el cambio de estado de acuerdo a los parámetros elegidos. Para realizar la fusificación en MATLAB se utiliza una función triangular, es decir, se implementan tres particiones difusas tanto para la variable lingüística de entrada y salida. Por otro lado, la fusificación en ARDUINO presenta una función trapezoidal y para poder asemejar a la función de MATLAB se utilizan los mismos valores en la parte central de la función new FuzzySet.
- El sistema de levitación aerodinámica de la esfera en lazo abierto presenta demasiada inestabilidad en la práctica para una entrada escalón, tal como lo predice su modelo matemático expresado en la ecuación (3). En el actuador fue necesario usar dos ventiladores para generar el flujo adecuado para la levitación de la esfera. El tiempo de estabilidad de todo el sistema es aproximadamente de 30 segundos. Como se esperaba en la respuesta del modelo matemático de la Figura 17 presenta ligeras oscilaciones, así también el sistema real presenta ligeras oscilaciones.
- Los beneficios de la implementación de un controlador difuso en el control del sistema de levitación aerodinámica de una viga son su facilidad en la programación, flexibilidad en la sintonización, robustez en su respuesta ante perturbaciones interna y externa del sistema. En la Figura 19 se observa que el tiempo de crecimiento en la respuesta real es más extenso que la respuesta de simulación, esto por el cambio de límites en las particiones. Sin embargo, el tiempo de estabilidad es de 40 segundos igual a lo esperado en simulación mostrado en la Figura 17.

Las nuevas tarjetas de desarrollo electrónicos que surgen en la actualidad facilitan la implementación de estos controladores a bajo costo. La placa de desarrollo electrónico ARDUINO UNO es suficiente para la implementación del control de un sistema de simple entrada y salida, como el presentado en este artículo. Con el avance de nuevas tecnologías en tarjetas de desarrollo en el campo de la electrónica se podrá llegar a implementar controladores difusos de múltiples entradas y salidas, incluso llegar a implementar redes neuronales a bajo costo.

AGRADECIMIENTOS

Un agradecimiento a estudiantes que cursaron la asignatura de SISTEMAS DE CONTROL: Dorian Alexander Romero Alcocer, Mauricio Chirinos Caero, Manuel Alejandro Aguirre Iturre, Kevin Brian Alvarado Calle, Ivonne Andrea Anave Zenteno, Rodriguez Arnez Iriarte, Maria Isabel Serna Pereira, Rosse Mary Cabellos Escalera, Diego Siles Guardia, Diego Weimar Sandoval Tarifa.

REFERENCIAS BIBLIOGRÁFICAS

- Artero, Ó. T. (2013). ARDUINO. Curso práctico de formación: RC Libros.
- Bastianon, R. A. (2008). Cálculo y diseño de la hélice óptima para turbinas eólicas. Buenos Aires, 1-48.
- Chacon, J., Saenz, J., Torre, L., Diaz, J., & Esquembre, F. (2017). Design of a low-cost air levitation system for teaching control engineering. *Sensors*, 17(10), 2321. <https://doi.org/10.3390/s17102321>
- Chipdip. (2020). JF0825B1H-R, ventilador de 12V, 80x80x25mm, rodamiento de 3500 rpm. 2020, from <https://www.chipdip.ru/product/jf0825b1h>
- de Saavedra, J. M. G. A., & Acher, G. R. (2012). Introducción a la aerodinámica potencial: Editorial Paraninfo.
- Del Brío, B. M., & Molina, A. S. (2002). Redes neuronales y sistemas difusos Bonifacio Martín del BRío, Alfredo Sanz Molina (2 ed.): Alfaomega Ra-Ma.
- Escaño, J. M., Algarín, D., & Ortega, M. (2004). IDENTIFICACION Y CONTROL DE POSICION DE UN SISTEMA DE LEVITACION NEUMATICA. 1-7. <http://intranet.ceautomatica.es/old/actividades/jornadas/XXV/documentos/39-anessiuore.pdf>
- HETPRO. (2020). MOTOR BRUSHLESS A2212/13T 1000KV. Retrieved 2020, from <https://hetpro-store.com/motor-brushless-a2212-13t-1000-kv/>
- Jamicon. (2020). DC FAN 80x80x25 mm. 2020, from <https://static.chipdip.ru/lib/245/DOC000245581.pdf>
- Jernigan, S. R., Fahmy, Y., & Buckner, G. D. (2008). Implementing a remote laboratory experience into a joint engineering degree program: Aerodynamic levitation of a beach ball. *IEEE Transactions on Education*, 52(2), 205-213. <https://doi.org/10.1109/TE.2008.924217>
- Kuo, B. C. (2010). Automatic control systems (pp. 149-150). USA: John Wiley & Sons, Inc. <https://doi.org/10.1109/TAC.2010.2069430>
<https://doi.org/10.1007/s12555-010-0410-6>
<https://doi.org/10.1007/s12555-010-0426-y>
- Kuzhandairaj, J. C. (2018). Development, control and testing of an air levitation system for educational purpose. (Corso di Laurea Magistrale in Ingegneria dell'Automazione), POLITECNICO DI MILANO 1863, ITALIA.
- Sivanandam, S., Sumathi, S., & Deepa, S. (2007). Introduction to fuzzy logic using MATLAB (Vol. 1): Springer. <https://doi.org/10.1007/978-3-540-35781-0>
- Timmerman, P., & Van der Weele, J. P. (1999). On the rise and fall of a ball with linear or quadratic drag. *American Journal of Physics*, 67(6), 538-546. <https://doi.org/10.1119/1.19320>

Derechos de Autor (c) 2020 Elías Prudencio Chavez Jaldin; Edgar Roberto Ramos Silvestre; Dorian Alexander Romero Alcocer, Mauricio Chirinos Caero.



Este texto está protegido por una licencia [Creative Commons 4.0](https://creativecommons.org/licenses/by/4.0/).

Usted es libre para Compartir —copiar y redistribuir el material en cualquier medio o formato— y Adaptar el documento —remezclar, transformar y crear a partir del material— para cualquier propósito, incluso para fines comerciales, siempre que cumpla la condición de:

Atribución: Usted debe dar crédito a la obra original de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace de la obra.

[Resumen de licencia](#) - [Texto completo de la licencia](#)