

Diseño de un control difuso tipo Mamdani para regular la temperatura de una Acuario tipo tropical

Designing a Mamdani type fuzzy control to regulate the temperature of a tropical Aquarium

DE LA CRUZ-ALEJO, Jesús†* & LOYA-ESCALANTE, María Teresa

Tecnológico de Estudios Superiores de Ecatepec

ID 1^{er} Autor: *Jesús, De La Cruz-Alejo* / ORC ID: 0000-0001-5072-3985, CVU CONACYT ID: 173289

ID 1^{er} Coautor: *María Teresa, Loya-Escalante* / ORC ID: 0000-0003-1515-4312, CVU CONACYT ID: 635849

DOI: 10.35429/JID.2019.8.3.1.9

Recibido 09 Julio, 2019; Aceptado 03 de Septiembre, 2019

Resumen

En el presente trabajo de investigación, se muestra una técnica de control automático basada en lógica difusa para regular las condiciones de temperatura en un acuario tropical con capacidad para 100 litros. Para analizar la efectividad de la técnica se realizó el diseño y la implementación de un controlador de lógica difusa tipo Mamdani. Posteriormente el controlador difuso se integró a un sistema eléctrico utilizando una resistencia de 100 watts. La regulación se realiza a partir de la adquisición de la temperatura del agua, variando así la corriente de la resistencia, se utilizó un sensor de temperatura, un calentador como actuadores. La funcionalidad del sistema se verifica a través de la plataforma Artix-7 FPGA, empleando como lenguaje de descripción de hardware VHDL. Se desarrolló el diseño del controlador mediante la herramienta de fuzzy logic en Matlab para poder comparar resultados experimentales de la FPGA, obteniéndose un error de 2.94% de exactitud entre ambos diseños.

Lógica difusa, Vhdl, Temperatura

Abstract

In the present research work, an automatic control technique based on fuzzy logic is shown to regulate the temperature conditions in a tropical aquarium with a capacity of 100 liters. To analyze the effectiveness of the technique, the design and implementation of a fuzzy logic controller type Mamdani was carried out. Subsequently, the diffuse controller was integrated into an electrical system using a 100-watt resistor. The regulation is carried out from the acquisition of the water temperature, thus varying the current of the resistance, a temperature sensor, and a heater as actuators were used. The functionality of the system is verified through the Artix-7 FPGA platform, using the VHDL hardware description language. The controller design was developed using the fuzzy logic tool in Matlab in order to compare experimental results of the FPGA, obtaining an error of 2.94% accuracy between both designs.

Fuzzy logic, Vhdl, Temperature

Citación: DE LA CRUZ-ALEJO, Jesús & LOYA-ESCALANTE, Maria Teresa. Diseño de un control difuso tipo Mamdani para regular la temperatura de una Acuario tipo tropical. Revista del Diseño Innovativo. 2019 3-8: 1-9

* Correspondencia del Autor (Correo electrónico: jdelaacruz@tese.edu.mx)

† Investigador contribuyendo como primer autor.

Introducción

Los acuarios son objetos de cristal donde se reproducen los ambientes subacuáticos de agua dulce, salada entre otros, en los que se trata de mantener las condiciones del ecosistema con base al tipo de acuario que se quiera reproducir. Por lo general, se utilizan en hogares, empresas, hoteles con fines ornamentales. Sin embargo, el hecho de tener un acuario requiere llevar a cabo un cuidadoso mantenimiento para que esté en condiciones ideales. Las dimensiones del acuario, tipo de peces que se introducirán dentro del acuario, niveles de oxigenación, temperatura e iluminación son algunas de las condiciones que se deben de tener en cuenta para mantener un acuario en condiciones ambientales óptimas. La temperatura juega un papel importante dentro de las condiciones del ecosistema para los peces que se introducirán, ya que los cambios de temperatura suelen ser críticos en la reproducción de ciertas especies, así también, algunas enfermedades comunes en peces dependen también de la temperatura. [I] [II]

Un factor fisiológico vital para los peces es el metabolismo, y el ritmo del metabolismo es determinado ampliamente por la temperatura del agua ya que la temperatura interna del pez es igual a la temperatura externa del agua. El agua más caliente acelera el ritmo de la digestión y otros procesos biológicos. Esto ocasiona que el pez llegue a la adultez más rápido, pero también acorta su periodo de vida [III] [IV]. Otros procesos biológicos como la eclosión de huevos y desarrollo de los alevines también se ven alterados por una temperatura del agua más elevada.

La lógica difusa no es la respuesta para todos los problemas técnicos, pero sí para aquellos problemas de control donde la simplicidad y la rapidez en la implementación son importantes. Algunas de las aplicaciones donde se ha utilizado la lógica difusa es en el control de ambientes, equipos para el hogar, equipos de electrónica y sistemas de automóviles [V] [VI]. Así la lógica difusa tiene una ventaja sobre un sistema de control convencional, el cual, es capaz de incorporar la experiencia de un experto en el área de conocimiento. Esta experiencia es trasladada hacia el diseño de las reglas que definirán las variables lingüísticas que permitan definir las acciones de control dentro del diseño.

El objetivo de este trabajo es diseñar e implementar un sistema de control utilizando lógica difusa tipo Mamdani para regular la temperatura de un acuario bajo un dispositivo lógico programable FPGA.

La aportación del presente trabajo de investigación presenta una alternativa de solución para el control de este tipo de acuarios que permitan disminuir costos, complejidad, eficiencia y bajo consumo de potencia, así como poder generar una base de conocimiento que se puede ir ampliando para proyectos más grandes o donde el control de temperatura sea un factor importante dentro del proceso. El presente trabajo está organizado en las siguientes secciones. Sección 2, se describe el control lógico difuso, sección 3, diseño e implementación del FLC, sección 4, Se realiza la caracterización del sistema, mientras que en la sección 5, se presentan los resultados experimentales del FLC, sección 6, conclusiones son presentadas.

Controlador Lógico Difuso (CLD)

El diagrama a bloques del controlador propuesto para el control de la temperatura, se presenta en la figura 1, en donde se muestra las etapas del controlador difuso tipo Mamdani. La etapa de pre procesamiento (entrada) se encarga de acondicionar las señales de entrada, puesto que la tarjeta cuenta con entradas analógicas-digitales. En la etapa de Fusificación se convierte la magnitud de la señal en una cantidad difusa, obteniendo el valor de pertenencia que tiene en cada uno de los valores lingüísticos en el universo de discusión.

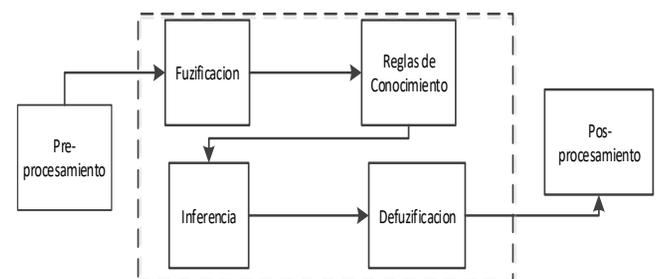


Figura 1 Diagrama del Controlador tipo Mamdani.

Fuente: Elaboración Propia

El diseño del controlador propuesto se describe en la tabla 1, donde se puede observar que contiene 2 entradas y una salida, cada entrada cuenta con 5 conjuntos difusos, con una resolución de 10 bits para cada universo de discusión.

La propuesta se define de esta manera, al utilizar un mayor número de conjuntos, se tiene una mayor resolución, sin embargo se observó a través de simulaciones realizadas que al utilizar más de 7 conjuntos difusos no aumenta significativamente la eficiencia del controlador y el error de aproximación es casi el mismo, por lo que se concluye que después de 7 conjuntos difusos, el error de aproximación es prácticamente cero y por otro lado, se aumenta la complejidad y el consumo de recursos del sistema por lo que esto no es recomendable. De esta manera, se proponen 5 conjuntos difusos tipo triangular para cada variable de entrada y una resolución de 8 bits para la función de membresía.

Variabes	Nombre	Resolución	Función de Membresía	Conjuntos	Tipo de controlador
Entradas	Error, derivada del error	10 bits (0 -1023)	Triangular	5	Mamdani
Salidas	Voltaje	8 bits (0 - 255)	Triangular	5	

Tabla 1 Diseño del controlador

Fuente: Elaboración Propia

Para la implementación del controlador, las variables de entrada son el error (1) en la temperatura del agua del acuario y la derivada de este error (2). La primera variable de entrada permite conocer qué tan alejado se encuentra la temperatura del valor de referencia, que para este caso, se ha establecido en 25°C ya que es un acuario tropical. La segunda variable de entrada permite conocer la variación de este error, esto es:

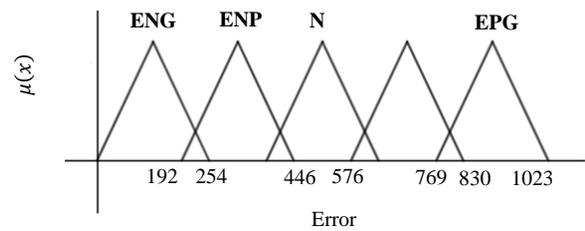
$$error(x) = Temp. de referencia - Temperatura actual \quad (1)$$

$$derivada del error (y) = error actual - error anterior \quad (2)$$

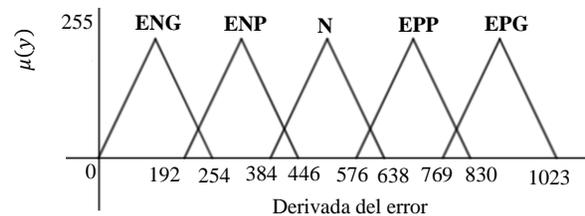
Variabes de entrada- salida del CLD

Las variables de entrada del controlador son las que permiten monitorear el sistema con una frecuencia determinada. En la figura 2, se muestra el universo de discurso de las variables que se proponen con base a los parámetros que se deben mantener dentro de un acuario de tipo tropical. Para la primera de las variables de entrada figura 2(a), se muestran los conjuntos difusos, donde las variables lingüísticas propuestas son: Error Negativo Grande, Error Negativo Pequeño, Nulo, Error Positivo Pequeño, Error Positivo Grande, contenidas en el rango de -11°C a 11°C.

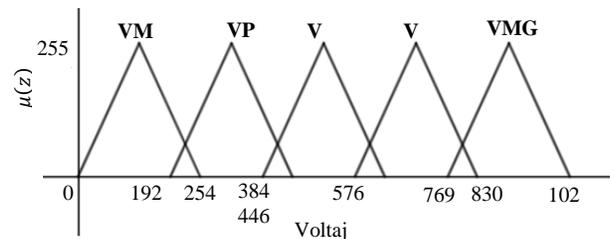
La segunda entrada al controlador es la derivada del error, contenidas en el rango de -6°C a 6°C grados, cuyas variables lingüísticas se muestran en la figura 2 (b) y son: Error Positivo Grande, Error positivo Pequeño, Nulo, Error Negativo Pequeño, Error Negativo Grand. La salida del controlador manipula el voltaje del actuador. En la figura 2(c), se muestran las variables lingüísticas que se utilizaron para la salida: Voltaje Muy Pequeño, Voltaje Pequeño, Voltaje Medio, voltaje Grande, Voltaje Muy Grande.



(a) Conjuntos difusos de entrada del error



(b) Conjuntos difusos segunda entrada derivada del error



C) Conjuntos de salida Voltaje

Figura 2 Conjuntos difusos del controlador

Fuente: Elaboración Propia

Diseño del CLD

Fuzificación

El primer paso del diseño del controlador es la Fuzificación, se propone como primer diseño el uso de funciones de tipo triangular debido a que proporciona cálculos más rápidos dentro de la configuración de hardware de la FPGA. Para obtener la función triangular de cada conjunto propuesto, se generan las tablas de búsqueda, empleando funciones triangulares, las cuales con base a (3), representan una función para cada uno de los conjuntos difusos

$$\mu(x) = \begin{cases} \frac{a(b-x)}{b-c}, & b \leq x \leq c \\ \frac{a(d-x)}{d-c}, & c \leq x \leq d \end{cases} \quad (3)$$

Estas contienen los valores de membresía de todas las funciones triangulares del universo de discurso. El algoritmo que se implemento es con base a (4).

$$\text{if } x < d \text{ then} \quad (4)$$

$$\mu(x) = \frac{[a(b-x)]}{(b-c)} \quad \text{else}$$

$$\mu(x) = \frac{[a(d-x)]}{(d-c)}$$

En la tabla 2, se muestra el rango de cada junto, donde se puede observar que son conjuntos simétricos, y el máximo traslape entre ellos es de dos conjuntos sin superar el punto medio de cada conjunto.

Conjunto	Intervalo	Error	Derivada del error	Punto medio
ENG	0-254	-11 a -6	-6 a -3	127
ENP	192-446	-8 a -2	-4 a -1	319
N	384-638	-3 a 3	-1.5 a 1.5	511
EPP	576-830	2 a 8	1 a 4	703
EPG	769-1023	6 a 11	3 a 6	897

Tabla 2 Tabla de los conjuntos
Fuente: Elaboración Propia

En la figura 3, se presentan los resultados de simulación obtenidos con el software dentro de la simulación del ISE donde x, y , representan la variable de entrada del error, y la derivada del error, respectivamente. Mientras que $mA... mE$ representan a las funciones de membresía para x , y nA, \dots , las funciones de membresía para y . En la figura 3, se observa que para valores distintos de x y y , se activan los valores de membresías correspondientes a cada conjunto activado. Así por ejemplo, si $x=380$ y $y=150$, se obtiene un valor de membresía $mb=132$ y $na=208$.

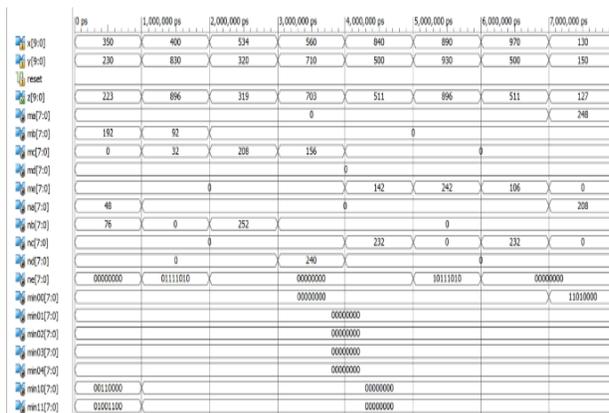


Figura 3 Simulación de la Fuzificación
Fuente: Elaboración Propia

Inferencia

Se plantean las reglas de conocimiento que requiere el sistema, estas reglas son del tipo modus ponens (si- entonces) con base a (5).

$$\text{If (antecedente) and (antecedente) then (consecuente)} \quad (5)$$

Se genera, una matriz que contiene todas las comparaciones entre los valores de membresía generados por ambas entradas, como se muestra en la tabla 3, obtenidas con base a (6).

$$\begin{aligned} \text{if } x \text{ es } A_1 \text{ and } y \text{ es } B_1 \text{ then } z &= (1,1) \\ \text{if } x \text{ es } A_1 \text{ and } y \text{ es } B_2 \text{ then } z &= (1,2) \\ \text{if } x \text{ es } A_5 \text{ and } y \text{ es } B_5 \text{ then } z &= (5,5) \end{aligned} \quad (6)$$

		Entrada Y				
		B ₁	B ₂	B ₃	B ₄	B ₅
Entrada X	A ₁	R ₁ = min(1,1)	R ₂ = min(1,2)	R ₃ = min(1,3)	R ₄ = min(1,4)	R ₅ = min(1,5)
	A ₂	R ₆ = min(2,1)	R ₇ = min(2,2)	R ₈ = min(2,3)	R ₉ = min(2,4)	R ₁₀ = min(2,5)
	A ₃	R ₁₁ = min(3,1)	R ₁₂ = min(3,2)	R ₁₃ = min(3,3)	R ₁₄ = min(3,4)	R ₁₅ = min(3,5)
	A ₄	R ₁₆ = min(4,1)	R ₁₇ = min(4,2)	R ₁₈ = min(4,3)	R ₁₉ = min(4,4)	R ₂₀ = min(4,5)
	A ₅	R ₂₁ = min(5,1)	R ₂₂ = min(5,2)	R ₂₃ = min(5,3)	R ₂₄ = min(5,4)	R ₂₅ = min(5,5)

Tabla 3 Matriz de inferencia

En la tabla 4, se muestra la Matriz de Asociación Difusa “FAM”, en la cual se establecen las reglas de conocimiento que nos permitirán obtener una acción de control adecuada. Se puede observar que para un error positivo pequeño y una derivada del error positivo grande se obtiene una acción de control de voltaje grande.

		Error				
		ENG	ENP	N	EPP	EPG
Derivada del error	ENG	VMP	VMP	VMP	VMP	VMP
	ENP	VMP	VMP	VMP	VMP	VMP
	N	VMP	VMP	VMP	VMP	VMP
	EPP	VMG	VG	VM	VG	VMG
	EPG	VMG	VG	VM	VG	VMG

Tabla 4 Matriz de Asociación Difusa (FAM)
Fuente: Elaboración Propia

Agregación

Para llevar a cabo la etapa de agregación, se utiliza la implicación $máx$, que representa la unión de las reglas difusas activadas y los conjuntos difusos que tiene un valor distinto de cero. El proceso se lleva a cabo a partir de la matriz obtenida en la etapa de inferencia.

Para la etapa de agregación, las variables de entrada son los 25 mínimos generados en la etapa anterior ($min00, \dots, min44$). Como variables de salida, se obtiene un vector de agregación con los valores máximos de cada una de las columnas que representan los conjuntos difusos de la variable de salida (voltaje).

Posteriormente, se selecciona la primera columna de la matriz, la cual representa el primer conjunto difuso de la variable de salida etiquetada como voltaje muy pequeño (VMP), la cual, consta de los mínimos $min00, min10, min20, min30, min40$, obteniéndose el máximo valor de ella, con base a (7).

$$Max_C1 = \max((\min00, \min10), \max(\min20, \min30), 40) \quad (7)$$

El resultado final es un vector que contiene los máximos de cada una de las columnas, el cual se obtiene mediante (8):

$$\mu(z_i) = \max[R_{1,i} \dots R_{5,i}] \quad (8)$$

Defuzificación

El método propuesto en este trabajo es el de *niveles alfa* obtenido mediante (9). Este método convierte la acción de control difusa de salida a sus correspondientes valores reales. En la figura 4, se muestra el código desarrollado en VHDL, para el cual se utilizó el ciclo if-then.

```
--columna 0
if c_max0(0) < c_max0(1) then
  max1:=c_max0(1);
  alfa_inicio_max1:=alfa_inicio_10;
  alfa_final_max1:=alfa_final_10;
else
  max1:=c_max0(0);
  alfa_inicio_max1:=alfa_inicio_00;
  alfa_final_max1:=alfa_final_00;
end if;

if c_max0(2) < c_max0(3) then
  max2:=c_max0(3);
  alfa_inicio_max2:=alfa_inicio_30;
  alfa_final_max2:=alfa_final_30;
else
  max2:=c_max0(2);
  alfa_inicio_max2:=alfa_inicio_20;
  alfa_final_max2:=alfa_final_20;
end if;

if max1 < max2 then
  max3:=max2;
  alfa_inicio_max3:=alfa_inicio_max2;
  alfa_final_max3:=alfa_final_max2;
else
  max3:=max1;
  alfa_inicio_max3:=alfa_inicio_max1;
  alfa_final_max3:=alfa_final_max1;
end if;
if max3 < c_max0(4) then
```

```
max_0:=c_max0(4);
alfa_inicio_max_0:=alfa_inicio_40;
alfa_final_max_0:=alfa_final_40;
else
max_0:=max3;
alfa_inicio_max_0:=alfa_inicio_max3;
alfa_final_max_0:=alfa_final_max3;
```

end if;

Figura 4 Agregación

$$niveles\ alfa = \frac{\sum_{k=1}^N \left(\frac{x_f^{\alpha_k} - x_i^{\alpha_k}}{2} \right) + x_i^{\alpha_k}}{N} \quad (9)$$

Donde $x_f^{\alpha_k}$ es nivel alfa final, $x_i^{\alpha_k}$ es nivel alfa inicial, del valor de membresía que se obtiene después de haber llevado a cabo la etapa de agregación, N es el número de conjuntos difusos de salida que tienen un grado de pertenencia distinto de cero.

Caracterización del sistema

Para llevar a cabo la acción de control y poder activar los actuadores (calentador y ventilador), se hace de un condicional *If-then* dentro de la programación en VHDL. En primera instancia, se realiza un análisis con base a las reglas de control propuestas y se propone controlar cada actuador partiendo del punto de equilibrio que en nuestro caso es un valor digital de 512 que representa un voltaje de 1.43V, que es el punto de equilibrio de la señal PWM.

Con base a los conjuntos del voltaje de salida (VMP, VP, VM), si la acción de control está dentro del rango de 0 a 512, se acciona el ventilador, ya que con base al análisis realizado se determina que la temperatura del acuario es alta, esta acción de control se modula a través de la señal PWM en un rango de 32 posiciones.

Por otro lado, si la acción de control es mayor a 512, en los conjuntos de la variable de salida voltaje (VM, VG, VMG), se acciona el calentador, ya que con base al análisis realizado se determina que la temperatura del acuario es baja, por lo que se requiere controlar la temperatura del agua en el acuario. En la figura 5, se muestra el algoritmo de la acción de control.

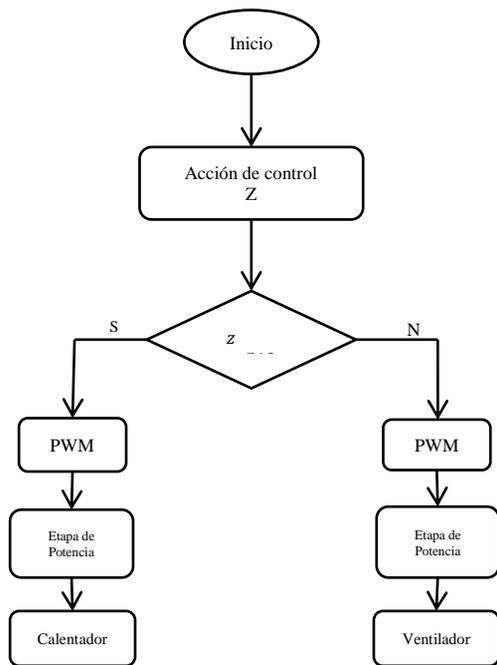


Figura 5 Caracterización de la variable de salida
Fuente: Elaboración Propia

Resultados de simulación y experimentales

Para realizar la validación de los resultados, se tomó una muestra de 50 lecturas, los resultados experimentales del controlador se muestran en la tabla 5, donde se puede observar que la diferencia entre los resultados obtenidos de la implementación en la FPGA utilizando el método de niveles alfa y los obtenidos usando Matlab, es mínima.

Temperatura °c		Valores en Digital		Niveles alfa FPGA	COG MATLA B
X	Y	X	Y		
19,0537634	19,16129032	5	15	127	128
19,1075269	20,07526882	10	100	127	128
19,2258065	19,95698925	21	89	127	127
19,3225806	28,67741935	30	900	896	895
19,3870968	19,83870968	36	78	127	127
19,483871	19,53763441	45	50	127	127
19,5376344	27,60215054	50	800	799	801
19,7204301	20,34408602	67	125	127	127
19,9677419	19,48387097	90	45	127	127
20,1827957	20,93548387	110	180	127	127
20,3225806	21,51612903	123	234	223	254
20,3978495	20,61290323	130	150	127	127
20,827957	25,37634409	170	593	607	566
20,9354839	22,76344086	180	350	319	319
21,1505376	22,22580645	200	300	319	319
21,2580645	23,83870968	210	450	511	511
21,3655914	27,70967742	220	810	799	822
21,4193548	25,93548387	225	645	703	703

21,516129	25,09677419	234	567	511	511
22,2258065	26,95698925	300	740	703	703
22,7096774	20,61290323	345	150	127	127
22,7634409	21,47311828	350	230	223	242
22,8387097	27,84946237	357	823	799	868
23,3010753	27,92473118	400	830	896	843
23,3010753	25,55913978	400	610	607	615
23,6344086	27,38709677	431	780	799	745
23,6344086	26,07526882	431	658	703	703
23,7311828	25,74193548	440	627	607	664
23,8924731	23,64516129	455	432	415	463
24,6989247	25,34408602	530	590	607	557
24,7419355	22,44086022	534	320	319	319
24,8387097	29,61290323	543	987	896	703
25,0215054	26,6344086	560	710	703	703
25,0967742	23,90322581	567	456	511	511
25,1182796	29,04301075	569	934	896	703
25,2365591	22,95698925	580	368	319	319
25,9784946	22,70967742	649	345	319	319
26,2903226	27,22580645	678	765	703	703
26,5268817	23,83870968	700	450	511	511
27,3870968	29,43010753	780	970	896	895
28,0322581	24,37634409	840	500	511	511
28,5698925	29	890	930	896	895
28,5698925	26,29032258	890	678	896	703
28,6451613	25,89247312	897	641	896	703
29,4301075	24,37634409	970	500	511	511
29,4516129	25,93548387	972	645	896	703
29,5376344	20,61290323	980	150	127	127
29,6451613	29,75268817	990	1000	896	895
29,7311828	29,98924731	998	1022	896	895
29,7634409	19,30107527	1001	28	127	127
30	29,64516129	1023	990	0	512

Tabla 5 Resultados experimentales del controlador
Fuente: Elaboración Propia

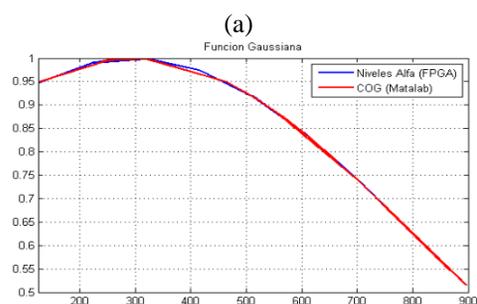
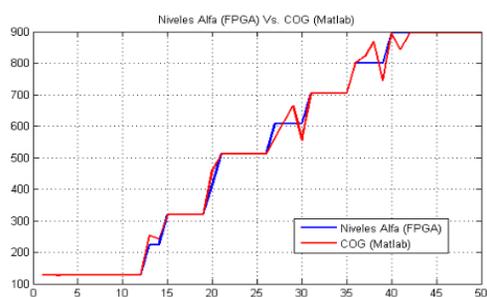
En el gráfico 1(a), se muestra la comparación de los resultados experimentales en color azul y los obtenidos en Matlab en color rojo. Se puede observar estadísticamente que la media de Niveles alfa es de 510.62 y la obtenida por Matlab es de 510.82, obteniendo un error de aproximación de 2.94%. Este porcentaje es aceptable, el cual, se obtuvo con base a (10).

$$Error = \frac{\delta_{niveles\ alfa} - \delta_{cog}}{resolución} * 100 \tag{10}$$

$$Error = \frac{302.43 - 272.32}{1023} * 100 = 2.94\%$$

En 1(b) se muestran la desviación de los resultados obtenidos usando una campana de gauss.

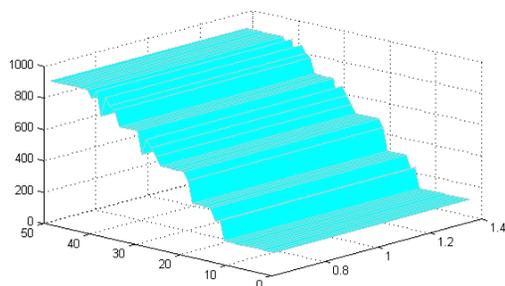
Otro resultado que se puede obtener a partir de las reglas de conocimiento propuesta es la superficie de las reglas de control que se muestra en el gráfico 2(a), la cual, presenta los resultados obtenidos en Matlab, mientras que en (b), la superficie obtenida a partir de los resultados obtenidos por el controlador implementado en la FPGA.



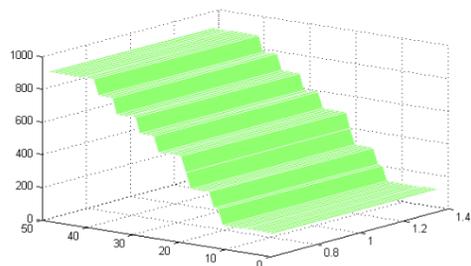
(b)

Gráfico 1 Análisis estadístico COG vs Niveles Alfa
Fuente: Elaboración Propia

La similitud entre ambas, nos muestra la precisión de los resultados, obteniéndose una función monótona ya que su acción se realiza suavemente. En la figura 6, se muestran los diferentes ciclos de trabajo que se implementaron en la tarjeta FPGA.



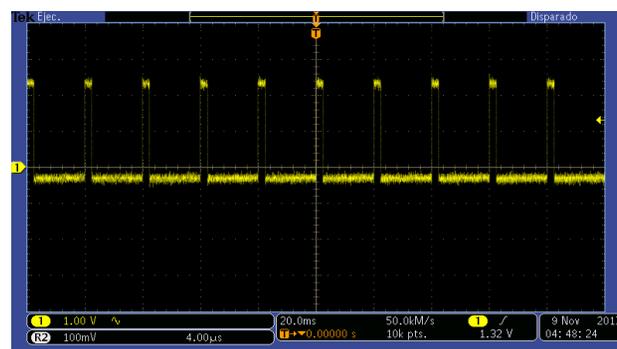
(a) COG Matlab



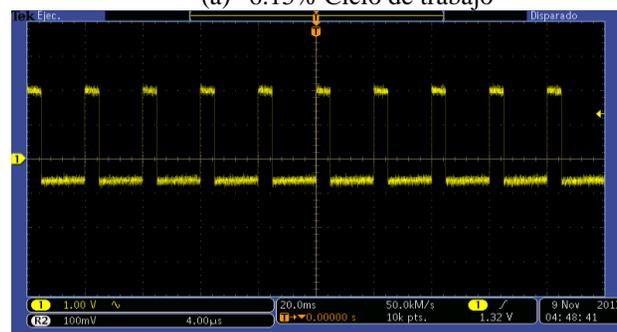
(b) Niveles alfa FPGA

Gráfico 2 Superficies de control

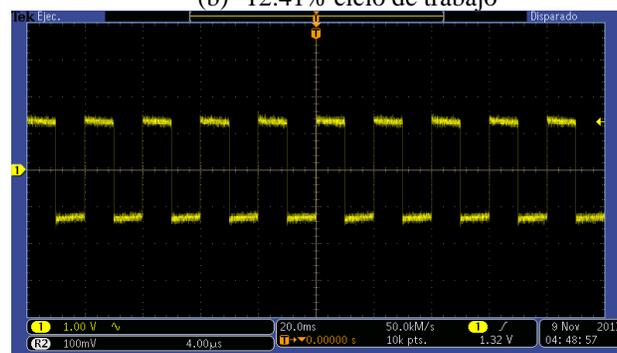
Como se observa, conforme aumenta el porcentaje del ciclo de trabajo, aumenta el tiempo que se mantiene encendido, variándose con esta la potencia de la resistencia y la velocidad del ventilador.



(a) 6.15% Ciclo de trabajo



(b) 12.41% ciclo de trabajo



(c) 24.92% ciclo de trabajo

Figura 6 Señal PWM

Fuente: Elaboración Propia

En el gráfico 3, se muestra la respuesta del calentador ante los cambios que se presentan por la acción de control, donde se observa que la corriente del calentador responde de manera lineal con respecto a los cambios generados por la señal PWM.



Gráfico 3 Respuesta del calentador

En el gráfico 4, se muestra la respuesta del ventilador ante los cambios que se presentan por la acción de control, donde se observa que el voltaje del ventilador responde de manera lineal con respecto a los cambios generados por la señal de PWM. Después de haber diseñado el PWM y acoplarlo a los actuadores a través de una etapa de potencia se obtienen los resultados mostrados en la tabla 6, donde se puede ver la acción de control sobre cada uno de los actuadores e ir variando de acuerdo al ciclo de trabajo establecido, por el controlador.



Gráfico 4 Respuesta del Ventilador

Fuente: *Elaboración Propia*

La respuesta del sistema con base al controlador, presenta resultados favorables que se demuestran estadísticamente, obteniéndose un error del 2.94% muy aceptable. También, se comprueba la funcionalidad del método propuesto con respecto al controlador diseñado en Matlab, así también se comprueba físicamente la respuesta del calentador hacia la repuesta de la señal PWM.

Estados	Salida PWM (V)	V _{out}	I _{calentador}	Ciclo de Trabajo (%)
0	0	0	0	0
1	0.9	0.34	0.35	3,125
2	0.180	0.86	0.38	6,25
3	0.270	0.146	0.35	9,375
4	0.360	0.209	0.32	12,5
5	0.450	0.273	0.46	15,625
6	0.540	0.340	0.47	18,75
7	0.629	0.407	0.49	21,875
8	0.720	1.54	0.51	25
9	0.809	1.68	0.51	28,125
10	0.900	1.81	0.52	31,25
11	0.990	1.92	0.54	34,375
12	1.08	2.00	0.56	37,5
13	1.17	2.08	0.58	40,625
14	1.25	2.14	0.59	43,75
15	1.35	2.20	0.62	46,875
16	1.43	2.25	0.63	50
17	1.52	2.29	0.65	53,125

18	1.61	2.32	0.67	56,25
19	1.70	2.36	0.69	59,375
20	1.79	2.39	0.72	62,5
21	1.88	2.41	0.76	65,625
22	1.97	2.44	0.78	68,75
23	2.06	2.46	0.80	71,875
24	2.15	2.48	0.83	75
25	2.242	2.50	0.85	78,125
26	2.33	2.51	0.87	81,25
27	2.42	2.53	0.89	84,375
28	2.51	2.55	0.90	87,5
29	2.60	2.56	0.91	90,625
30	2.69	2.57	0.91	93,75
31	2.78	2.59	0.91	96,875

Tabla 6 Respuesta del sistema

Fuente: *Elaboración Propia*

Conclusiones

El uso de dispositivos FPGA permite mayor capacidad de procesamiento de información, además de cumplir con la función de acoplar los elementos del sistema referenciado en la lógica del controlador difuso desarrollado. También, se pudo observar que la dependencia o variación de la potencia del calentador por la cantidad de agua en el acuario se mantiene, y se controla su potencia a través de la acción de control mediante la señal PWM, por lo que no habría problemas si se colocara un calentador con mayor potencia que se tiene en función de los litros de agua, debido a que la referencia de la temperatura del agua del acuario, se regula a través de la potencia del calentador y de esta manera, se mejora la vida de los peces tropicales a través del control difuso de la temperatura aplicado a el ecosistema de una acuario, evitando totalmente el descuido por el control manual. Los resultados de simulación y experimentales confirman las predicciones teóricas.

Referencias

- I. M. Mariani, El gran libro de los peces, Planeta
- II. M. A. Olmos Castillo, «Tutorial de Introduccion de Lógica Borrosa,» 2008. [En línea]. Available: http://www.dma.fi.upm.es/recursos/aplicaciones/logica_borrosa/web/tutorial_fuzzy/introduccion2.html.
- III. Atlas ilustrado del acuario, Susaeta.

- IV. Guia de Acuariofilia. Manual de un hobby fascinante, EHEIM
- V. J. Lloret Pineda, «Diseño, instalación y mantenimiento de un acuario de exposición de agua dulce.,» Cádiz, 2006.
- VI. D. Guzman y V. Castaño, «La Logica Difusa en Ingenieria: principios, Aplicaciones y Futuro,» *Ciencia y tecnologia: Investigación*, 2006.
- VII. F. Rodriguez, *Acuariofilia Rentable. Manual teorico-practico para la cria de peces.*
- VIII. H. A. Caicedo Analuisa y D. S. Obando Panchi , «Diseño e implementación de un sistema automatizado para el control del acuario del museo de Ciencias de la escuela politecnica Nacional,» Quito, Ecuador, 2011.
- IX. L. A. Sala Carillosama y A. P. Arcos Lara, «Construccion de un prototipo de deteccion y control electronico de temperatura e iluminacion para el ecosistema de un acuario de peces tropicales en un clima templado.,» Peru, 2014.
- X. H. Nelson Acosta y J. Garrido Salas, *Diseño de controladores dedicados a la logica difusa*, Madrid, España, 2006.
- XI. P. Ponce Cruz, *Inteligencia Artificial con aplicaciones a la Ingeniera*, Mexico: Alfaomega, 2010
- XII. W. Banks y G. Hayward, *fuzzy Logic in Embedded Microcomputer and control System*, Byte Craft, 2002
- XIII. D. G. Maxinez y J. Alcalá, *VHDL, el arte de programar sistemas digitales*, Monterrey, México: CECSA, 2002
- XIV. J. C. Garcia Montalva, J. De la Cruz Alejo y J. Diaz Salgado, «Fuzzy Logic Control on FPGA Using LabVIEW,» *Multibody, Mechatronic Systems*, pp. 261-270, 2014
- XV. R. Antonio Mendez, J. De la Cruz Alejo y O. Peñaloza Mejia, «Fuzzy Logic Control on FPGA for Solar Tracking System,» *Multibody Mechatronic System*, pp. 11-21, 2015
- XVI. Diseño, Automatizacion y Control, al alcance para todos,» [En línea]. Available: <http://softwareparaingenieros.com/xilinx-ise-14-7/>. [Último acceso: noviembre 2017].
- XVII. J. De la Cruz Alejo, R. Antonio Méndez y M. Salazar Pereyra, «Fuzzy Logic Control on FPGA for two axes solar tracking,» *The Natural Computing Application*, 2017.