

Fuzzy Inference System based on Boolean and Kleenean relations with Convex Combiner

Sistema de Inferencia Difusa basado en Relaciones Booleanas y Kleeneanas con Combinador Convexo

Juan D. Chacón¹, Jairo Soriano¹, Omar Salazar^{*,1}

¹Universidad Distrital Francisco José de Caldas

*Correspondence: osalazarm@correo.udistrital.edu.co

Recibido: 19-12-2016. Modificado: 21-03-2017. Aceptado: 02-05-2017.

Abstract

Context: In the design process of Fuzzy Inference Systems based on Boolean and Kleenean Relations (FIS-BKR) there is a dilemma choosing the regular kleenean extensions of a given boolean function. The set of possible kleenean extensions of a boolean function has a lattice structure under the usual partial order of functions. The fuzzy convex combination proposed by Zadeh guarantees some properties related to this order.

Method: The addition of a convex combiner just before the defuzzifier offers a solution to the above situation. The ISE (Integral Squared Error) and ITSE (Integral Time-weighted Squared Error) performance indexes were used on an application for tuning a liquid level control system.

Results: The tuning process carried out on the FIS-BKR controller with fuzzy convex combiner using constant coefficients, implied an improvement of the controlled system up to 1.427% for ISE index and up to 21.99% for ITSE with respect to the extreme extensions.

Conclusions: New evidence of convenient characteristics of FIS-BKR controllers with fuzzy convex combiner was presented when the performance indexes ISE and ITSE were evaluated. On the other hand, although in this work parameter tuning for convex combination was done by grid search (brute force), it would be interesting to study more effective optimization methods for this purpose.

Keywords: Fuzzy convex combiner, Level control, Fuzzy control, ISE performance index, ITSE performance index.

Acknowledgments: The authors would like to thank to the Laboratory for Automation and Computational Intelligence (LAMIC) for all its contributions to this paper.

Language: Spanish.

Open access



Cite this work as J. D. Chacón, J. Soriano, O. Salazar, "Fuzzy Inference System based on Boolean and Kleenean Relations with Convex Combine", Ingeniería, vol. 23, no.1, pp. 7-30, 2018.

© The authors; reproduction right holder Universidad Distrital Francisco José de Caldas.

DOI: <https://doi.org/10.14483/23448393.11138>

Resumen

Contexto: En el proceso de diseño de sistemas de control difuso basados en relaciones booleanas o kleeneanas (FIS-BKR, por sus siglas en inglés) existe una clara disyuntiva a la hora de elegir las extensiones kleeneanas regulares de una función booleana dada. El conjunto de posibles extensiones kleeneanas de una función booleana tiene estructura reticular bajo el orden parcial usual de funciones. La combinación convexa difusa propuesta por Zadeh garantiza ciertas propiedades relacionadas con este orden.

Método: La adición de un combinador convexo antes del defusificador ofrece una solución a esta situación. Los índices de desempeño ISE (Integral Squared Error) e ITSE (Integral Time-weighted Squared Error) son usados en la sintonización de un sistema de control de nivel de líquido en un tanque cilíndrico.

Resultados: El proceso de sintonización llevado a cabo en el controlador de nivel tipo FIS-BKR con combinador convexo difuso usando coeficientes constantes implicó una mejora en la respuesta del sistema controlado de hasta el 1.427% para el ISE y de hasta el 21.99% para el ITSE con respecto al desempeño de las extensiones extremas.

Conclusiones: Se presentaron nuevas evidencias de las características favorables de controladores FIS-BKR con combinador convexo difuso cuando se evaluaron los índices de desempeño ISE e ITSE. Por otro lado, teniendo en cuenta que la sintonización de los parámetros de combinación convexa se realizó mediante búsqueda exhaustiva, como trabajo futuro valdría la pena explorar técnicas de optimización más eficientes.

Palabras clave: Combinador convexo difuso, Control de nivel, Control difuso, Criterio ISE, Criterio ITSE.

Agradecimientos: Los autores quieren agradecer al grupo de investigación LAMIC (Laboratorio de Automática e Inteligencia Computacional) por sus contribuciones en lo relacionado a este artículo.

Idioma: Español

1. Introducción¹

Los automatismos basados en lógica booleana (también llamada clásica o bivalente) han sido ampliamente utilizados para controlar distintos fenómenos, por ejemplo, temperatura, nivel, caudal y otros [2]. Estos cuentan con principios básicos de diseño (como tablas de verdad), y técnicas de simplificación (mapas de Karnaugh, por ejemplo) que ayudan a su rápido planteamiento, fácil aplicación y bajo costo computacional. Sin embargo, presentan serios inconvenientes a causa de la naturaleza discontinua de la acción de control sobre los fenómenos sobre los que operan.

Buscando mejorar este comportamiento Soriano *et. al.* [3] propusieron el reemplazo de la lógica booleana por la lógica difusa en el diseño de automatismos. Esto derivó en un nuevo esquema de inferencia difusa con claras diferencias sobre el esquema clásico propuesto por Mamdani y As-

¹Nota aclaratoria: el presente documento hace una extensión del artículo [1], el cual fue socializado por los mismos autores en el marco del Congreso Mundial en Inteligencia Computacional (WCCI 2016) de IEEE. El presente manuscrito exhibe una aplicación original de los controladores FIS-BKR con combinador convexo, mostrando nueva evidencia de la pertinencia de la técnica cuando se usan índices de desempeño ISE e ITSE. Por otro lado, también incluye una reflexión sobre la cardinalidad del conjunto de funciones extensión de una función booleana dada y la ventaja que ofrece la técnica en este sentido.

silian [4]. Desde ese entonces han aparecido varias aplicaciones: control de temperatura, control de nivel, control en procesos químicos, control de velocidad de un motor, control de actitud de un satélite, control en convertidores DC-DC, identificación de sistemas no lineales, identificación de series de tiempo, identificación de personas, mapas cognitivos difusos, y además estudios teóricos con el fin de consolidar las bases de este sistema de inferencia difusa. Entre estas contribuciones se destaca la realizada por Salazar *et. al.* [5], que presenta un método para extender funciones booleanas (o de lógica clásica) a funciones regulares² kleeneanas (o de lógica trivalente de Kleene). Tema con ciertas consecuencias teóricas que son precisamente el punto de partida de este trabajo.

Una de las ventajas de trabajar con funciones kleeneanas para el diseño de controladores difusos está en el hecho de que esta metodología permite proponer bases de reglas en forma de tablas de verdad finitas de tres elementos, que pueden ser representadas mediante fórmulas lógicas fáciles de computar, similar a como se hace en el diseño de automatismos booleanos. La metodología para el diseño y optimización de sistemas de control difuso basados en relaciones booleanas y kleeneanas (FIS-BKR) fue presentada en [7]. Dicha metodología hace amplio uso de la extensión de funciones booleanas a kleeneanas regulares. Sin embargo, la forma de extender una función booleana a una kleeneana regular no es única. Esto en ocasiones convierte el proceso de diseño en un reto debido a que la cantidad de extensiones puede ser enorme e inmanejable. La elección de una extensión regular adecuada para una aplicación particular requiere ciertos criterios de ingeniería.

La propuesta sobre la que se basa este artículo fue introducida en [1]. Se usa la estructura reticular de las funciones kleeneanas que son extensiones regulares de una función booleana dada y la combinación convexa difusa introducida por Zadeh [8]. Esta metodología de diseño cambia el panorama de la optimización de un problema discreto sobre un conjunto finito a uno continuo en un conjunto infinito.

En este artículo se presenta una aplicación al control de nivel de un sistema hidráulico compuesto por un tanque cilíndrico horizontal. Problema que fue trabajado en [9]–[12]. Los índices de desempeño que se trabajaron son el ISE (Integral Squared Error) e ITSE (Integral Time-weighted Squared Error) los cuales fueron minimizados en cierto espacio de parámetros relacionados con los coeficientes de la combinación convexa difusa. El control con dicha técnica implicó una mejora en la respuesta del sistema controlado de hasta el 1.427% para el ISE y de hasta 21.99% para el ITSE con respecto al desempeño de las extensiones extremas. Se evidenció la aparición de nuevos puntos de operación que ayudarán al diseñador, y creemos presentan importantes propiedades para futuras aplicaciones.

Consideramos que esta metodología, aparte de simplificar el proceso de diseño, capta la naturaleza de los sistemas de inferencia basados en relaciones algebraicas. Sin embargo, tareas relacionadas con la optimización de parámetros quedan abiertas, y serán clave a la hora de buscar un modelo más completo de este tipo de controladores.

²La regularidad de las funciones kleeneanas, estudiada por Mukaidono [6], es condición necesaria y suficiente para que estas funciones puedan ser expresadas por fórmulas lógicas.

Este documento está dividido en tres grandes partes. En la primera se presentan algunos preliminares sobre funciones lógicas así como de controladores FIS-BKR y su metodología de diseño. En la segunda se presenta la propuesta sobre el controlador FIS-BKR con combinador convexo. En la tercera parte se exhibe una aplicación de control de nivel en un tanque cilíndrico, donde se realiza un proceso de sintonización y se presentan los resultados. El documento concluye con una sección de conclusiones.

2. Preliminares

Definición 1 (Lógica trivalente de Kleene). Se denotará por \mathcal{K} el álgebra de valores de verdad $\langle K = \{0, 1/2, 1\}; \vee, \wedge, ', 0, 1 \rangle$, con el orden usual para sus elementos ($0 < 1/2 < 1$), donde las operaciones disyunción (\vee) y conjunción (\wedge) entre $a, b \in K$ están dadas por $a \vee b = \max \{a, b\}$ y $a \wedge b = \min \{a, b\}$. El complemento ($'$) de $a \in K$ es $a' = 1 - a$. También se identificará el elemento $1/2$ con la letra u . En la Tabla I(b) se muestran estas operaciones lógicas actuando sobre K .

Definición 2 (Lógica bivalente de Boole). La restricción de \mathcal{K} al conjunto $B = \{0, 1\}$ se conoce como lógica bivalente de Boole (lógica clásica o lógica booleana) y se denota con la letra \mathcal{B} . En la Tabla I(a) se muestran las operaciones lógicas actuando sobre B .

Tabla I. (a) Operaciones lógicas disyunción (\vee), conjunción (\wedge) y complemento ($'$) en lógica bivalente de Boole. (b) Operaciones lógicas disyunción (\vee), conjunción (\wedge) y complemento ($'$) en lógica trivalente de Kleene.

(a)			(b)																
\vee	0	1	\wedge	0	1	$'$	0	1	\vee	0	u	1	\wedge	0	u	1	$'$	0	1
0	0	1	0	0	0	0	1	0	0	u	1	0	0	0	0	0	0	1	
1	1	1	1	0	1	1	0	u	u	u	1	u	u	u	u	u	u	u	
								1	1	1	1	1	1	0	u	1	1	0	

Definición 3 (Lógica difusa estándar). Se denotará por \mathcal{I} el álgebra de valores de verdad conformada por $\langle I = [0, 1]; \vee, \wedge, ', 0, 1 \rangle$, con el orden usual para sus elementos, donde las operaciones disyunción (\vee) y conjunción (\wedge) entre $a, b \in I$ están dadas por $a \vee b = \max \{a, b\}$ y $a \wedge b = \min \{a, b\}$. El complemento ($'$) de $a \in I$ es $a' = 1 - a$.

Observación 1. Aunque \mathcal{K} e \mathcal{I} no están definidas sobre los mismos conjuntos de valores de verdad, \mathcal{K} e \mathcal{I} tienen la misma lógica proposicional [13], es decir, sus proposiciones lógicas son las mismas. Como consecuencia, \mathcal{K} permite la elaboración de tablas de verdad y obtención de fórmulas lógicas para \mathcal{I} , lo cual sería imposible para esta última debido a su carácter infinito.

Definición 4 (Funciones de lógica bivalente, trivalente y multivalente). Una función f de n variables se dice *función de lógica bivalente* (FLB) (o función booleana) si es una aplicación de B^n a B , se dice *función de lógica trivalente* (FLT) (o función kleeneana) si es una aplicación de K^n a K , y se dice *función de lógica multivalente* (FLM) si es una aplicación de I^n a I .

Observación 2. Una caracterización de funciones kleenanas desde un punto de vista más general se encuentra en [14]. Para el propósito de este artículo, una función kleeneana es una aplicación de K^n a K .

Definición 5 (Fórmulas bien formadas). Sea V un conjunto finito de variables. Dadas las constantes $0, u$ y 1 , y los símbolos \wedge, \vee y $'$, el conjunto de fórmulas bien formadas F está definido recursivamente como:

1. $0, 1, u \in F$ y si $x \in V$ entonces $x \in F$.
2. Si $f \in F$ y $g \in F$ entonces $f \wedge g \in F, f \vee g \in F$ y $f' \in F$.

Los elementos de F son llamados *fórmulas bien formadas*, *fórmulas lógicas* o simplemente *fórmulas*.

Es conocido que toda FLB es representable a través de una fórmula que no involucra la constante u en ella. Esto es consecuencia del teorema de expansión de Boole. Sin embargo, esto no siempre es cierto para las FLT. Una clase importante de FLT, llamadas *regulares*, son aquellas que pueden ser representadas por fórmulas.

Definición 6. Se define sobre K el siguiente orden parcial³: $0 \preceq_a u, 1 \preceq_a u$, y $a \preceq_a a$ para todo $a \in K$. La relación \preceq_a no compara 0 y 1 . Esta relación es extendida a K^n como sigue: Sean $\mathbf{a} = (a_1, \dots, a_n), \mathbf{b} = (b_1, \dots, b_n) \in K^n$, entonces $\mathbf{a} \preceq_a \mathbf{b}$ si, y solo si, $a_i \preceq_a b_i$ para todo $i = 1, 2, \dots, n$.

Definición 7 (Regularidad). Sea $f : K^n \rightarrow K$ una FLT. f se dice *regular* si, y solo si, $f(\mathbf{a}) \in B = \{0, 1\}$ implica $f(\mathbf{a}) = f(\mathbf{b})$ para todo $\mathbf{b} \preceq_a \mathbf{a}$.

Mukaidono [6] demostró que la regularidad es condición necesaria y suficiente para que una FLT sea representable a través de una fórmula. En esencia, este autor mostró que el conjunto F y el conjunto de las FLT regulares (FLTR) es el mismo. Dado que una FLB (o FLTR) puede tener más de una fórmula, entonces cada FLB (o FLTR) se identificará con su fórmula *canónica*, también llamada *forma normal*⁴.

Ahora, dada una tabla de verdad asociada a una FLB (o FLTR) es de interés conocer su *forma normal* (conjuntiva y/o disyuntiva). El siguiente procedimiento tomado y adaptado de [13] se usa para este fin. En [13], [15]–[17] se probó la existencia y unicidad de las formas normales.

- Método 1** (Forma normal disyuntiva [13]).
1. *Para las filas que tienen valor 1 en la columna de f que debe ser puesta en su forma normal disyuntiva:* Formar la *conjunción* de las variables que tienen valor 1 con los complementos de las variables que tienen valor 0 (omitir las variables que tienen valor u).
 2. *Para las filas que tienen valor u en la columna de f que debe ser puesta en su forma normal disyuntiva:* Formar la *conjunción* de las variables que tienen valor 1 con los complementos de las variables que tienen valor 0 y la conjunción tanto de la variable como de su complemento para las que tienen valor u .

³Mukaidono [6] llama *ambigüedad* a la relación \preceq_a .

⁴En el conjunto F dos fórmulas son equivalentes si, y solo si, representan la misma función. Esto define una relación de equivalencia sobre F . Una *forma normal* (o *canónica*) es un representante de cada clase de equivalencia [13], [15]–[17].

3. Formar la *disyunción* de las *conjunciones* obtenidas en los dos pasos anteriores. En caso de no existir conjunciones se obtendrá una disyunción vacía que representa la forma normal disyuntiva para 0.

La *forma normal disyuntiva kleeneana* de f , denotada $\text{FNCK}(f)$, se obtiene al descartar conjunciones que tienen los mismos (o posiblemente más) literales que otras conjunciones.

Observación 3. La *forma normal disyuntiva booleana* de f , denotada $\text{FNDB}(f)$, se obtiene siguiendo los pasos 1 y 3 del Método 1, teniendo en cuenta el valor de f solo en la filas de la tabla de verdad donde aparecen unos y ceros exclusivamente.

Observación 4. Las correspondientes formas normales conjuntivas (kleeneana y booleana), denotadas $\text{FNCK}(f)$ y $\text{FNDB}(f)$ respectivamente, se obtienen del Método 1 intercambiando las palabras *conjunción* por *disyunción*, *disyunción* por *conjunción*, *ceros* por *unos*, y *unos* por *ceros*.

Una forma de obtener fórmulas en \mathcal{I} es por medio de \mathcal{K} (debido a la igualdad de sus lógicas proposicionales [13]). Dado que \mathcal{B} es una restricción de \mathcal{K} , una manera de lograr lo anterior es partir de una FLB f , luego extender f a una FLTR g , y posteriormente encontrar una fórmula para g . La fórmula encontrada para g en \mathcal{K} es la misma que en \mathcal{I} , sin embargo, esta fórmula en \mathcal{I} es una FLM. En [5] se propuso el Método 2 para extender una FLB a una (o más) FLTR. De acuerdo a [5, Teorema 3.1], dada f una FLB el Método 2 garantiza que siempre es posible encontrar a lo menos una FLTR g que es extensión de f , pero no necesariamente única.

Definición 8. Sea $\beta : K^n \rightarrow N : \mathbf{a} \rightarrow \beta(\mathbf{a})$, donde $N = \{0, 1, 2, \dots, n\}$, definida⁵ como el número total de u en $\mathbf{a} \in K^n$.

Método 2 (Extensión [5]). 1. Dada una FLB $f : B^n \rightarrow B$ de n variables x_1, \dots, x_n construir una tabla de verdad de 2^n filas y $n+1$ columnas (n columnas para x_1, \dots, x_n y 1 columna para f). Estas filas tendrán las 2^n combinaciones de valores para x_1, \dots, x_n y los correspondientes valores para f (los cuales son representados como $f_i \in B, i = 0, 1, \dots, 2^n - 1$) como se muestra en la Tabla II(a).

2. A la tabla elaborada en el numeral 1, agregar $3^n - 2^n$ nuevas filas. Primero aquellas $\binom{n}{1}2^{n-1}$ filas cuya combinación de valores de las variables tienen solamente una u (nivel de ambigüedad $\beta = 1$), enseguida aquellas $\binom{n}{2}2^{n-2}$ filas cuya combinación de valores de las variables tienen dos u (nivel de ambigüedad $\beta = 2$), y así sucesivamente. Al final, la tabla finaliza con aquella $\binom{n}{n}2^{n-n} = 1$ fila que tiene n valores u en la combinación de valores de sus variables (nivel de ambigüedad $\beta = n$), como se muestra en Tabla II(b). Cada grupo de filas agregado previamente es separado por una línea horizontal que representa la separación entre diferentes niveles de ambigüedad β . El nivel de ambigüedad es creciente.
3. Hacer $g_i = f_i$ para $i = 0, 1, 2, \dots, 2^n - 1$.
4. Para cada grupo con nivel de ambigüedad $\beta = 1, 2, \dots, n$ hacer lo siguiente:
 - (a) Para la j -ésima fila considerada dentro del grupo con nivel de ambigüedad β , con $j = 1, 2, \dots, \binom{n}{\beta}2^{n-\beta}$, encontrar aquellas 2β filas en el grupo inmediatamente previo con

⁵El valor $\beta(\mathbf{a}) \in N$ se conoce como el *nivel de ambigüedad de a* [5, Definición 2.6]. Por ejemplo, si $\mathbf{a}, \mathbf{b} \in K^4$ tal que $\mathbf{a} = (0, u, u, 1)$ y $\mathbf{b} = (0, 1, 0, 1)$, entonces $\beta(\mathbf{a}) = 2$ y $\beta(\mathbf{b}) = 0$, es decir, \mathbf{a} tiene dos u y \mathbf{b} tiene cero u .

Tabla II. (a) FLB inicial $f : B^n \rightarrow B$ dada como una tabla de verdad. (b) FsLTR $g : K^n \rightarrow K$ (extensiones de f) dadas como tabla de verdad.

	x_1	x_2	\dots	x_n	f
2^n filas	0	0	\dots	0	f_0
	0	0	\dots	1	f_1
	\vdots	\vdots	\ddots	\vdots	\vdots
	1	1	\dots	1	f_{2^n-1}

β	x_1	x_2	\dots	x_n	g
0	0	0	\dots	0	$g_0 = f_0$
	0	0	\dots	1	$g_1 = f_1$
	\vdots	\vdots	\ddots	\vdots	\vdots
	1	1	\dots	1	$g_{2^n-1} = f_{2^n-1}$
1	u	0	\dots	0	g_{2^n}
	u	0	\dots	1	g_{2^n+1}
	\vdots	\vdots	\ddots	\vdots	\vdots
	1	1	\dots	u	\vdots
2	u	u	\dots	0	\vdots
	u	u	\dots	1	\vdots
	\vdots	\vdots	\ddots	\vdots	\vdots
	1	1	\dots	u	\vdots
\dots	\vdots	\vdots	\ddots	\vdots	\vdots
	\vdots	\vdots	\ddots	\vdots	\vdots
n	u	u	\dots	u	g_{3^n-1}

nivel de ambigüedad $\beta - 1$ cambiando uno por uno cada u por 0 y 1 sobre la j -ésima fila. Estas 2^β filas son inmediatamente menos o igual de ambigüas que la j -ésima fila de acuerdo a la relación \preceq_a definida sobre K y extendida sobre K^n (Definición 6).

- (b) Para cada una de las 2^β filas encontradas en el numeral 4a, encontrar el conjunto de valores más o igual de ambigüos al respectivo valor g de esa fila. Escribir este conjunto en una columna separada.
- (c) El conjunto de valores que g puede tomar sobre la j -ésima fila considerada es la intersección de aquellos conjuntos encontrados en el numeral 4b. Si g puede tomar más de un valor, crear tantas nuevas columnas para nuevas funciones g como sea posible. Para cada g creada, repetir el numeral 4.

Como se mencionó anteriormente, la forma de extender una FLB a una FLTR no es única. Si f es una FLB y $E_f = \{f_i\}_{i=1,2,\dots,L}$ es el conjunto de todas las FLTR que son extensión de f , entonces E_f ordenado en el sentido de la Definición 9 forma una estructura de retículo [1] con mínimo y máximo. El conjunto E_f es finito dado que $L \leq 3^{3^n}$, donde n es el número de variables de f .

Definición 9. Sea f una FLB y sean g y h dos FLTR que son extensión de f . Se dirá que g es menor o igual que h , denotado $g \preceq h$, si y solo si, $g(x) \leq h(x)$ para todo x en el dominio de g y h .

Definición 10 (Funciones extremas). Sean f una FLB y $E_f = \{f_i\}_{i=1,2,\dots,L}$ el conjunto de todas las FLTR que son extensiones de f . Se denotará por f_M a la extensión máxima en E_f tal que $f_i \preceq f_M$ para toda $i = 1, 2, \dots, L$ y similarmente por f_m a la extensión mínima en E_f tal que $f_m \preceq f_i$ para $i = 1, 2, \dots, L$.

Por definición $f_M = \bigvee_{i=1}^L f_i$ y $f_m = \bigwedge_{i=1}^L f_i$. Si L es grande (y también desconocido a priori) las anteriores expresiones pueden ser poco útiles para su cálculo. La siguiente proposición, demostrada en [1], da expresiones simples para f_M y f_m que no requieren del conocimiento previo de la totalidad de extensiones.

Proposición 1. [1] Sea f una FLB. La forma normal conjuntiva booleana de f es igual a f_M y la forma normal disyuntiva booleana de f es igual a f_m .

Las fórmulas correspondientes para f_M y f_m sobre lógica difusa son FLM. Estas dos FLM pueden ser operadas a través de combinación convexa (Definición 11) para obtener otras FLM, en particular, todas las extensiones en E_f vía la Proposición 2. La siguiente es una definición general dada por Zadeh [8].

Definición 11 (Combinación convexa). [8] Sean $f : X \rightarrow I$, $g : X \rightarrow I$ y $\lambda : X \rightarrow I$ funciones definidas sobre un conjunto X no vacío. La combinación convexa de f y g con coeficiente λ , denotada por $\langle f, g; \lambda \rangle$, es una función $\langle f, g; \lambda \rangle : X \rightarrow I$ dada por

$$\langle f, g; \lambda \rangle (x) = \lambda(x)f(x) + (1 - \lambda(x))g(x), \quad x \in X.$$

Proposición 2. [18], [19] Sean $f : X \rightarrow I$, $g : X \rightarrow I$ y $h : X \rightarrow I$ funciones definidas sobre X . Entonces $f \wedge g \preceq h \preceq f \vee g$ si, y solo si, existe $\lambda : X \rightarrow I$ tal que $h = \langle f, g; \lambda \rangle$.

La Proposición 2 aplicada a f_M y f_m , las extensiones máxima y mínima del conjunto $E_f = \{f_i\}_{i=1,2,\dots,L}$, tomando $X = I^n$ lleva a $f_m \wedge f_M = f_m \preceq f_i \preceq f_M = f_m \vee f_M$ si, y solo si, existe $\lambda_i : X \rightarrow I$ tal que $f_i = \langle f_m, f_M; \lambda_i \rangle$, $i = 1, 2, \dots, L$. Entonces todas las extensiones regulares de una FLB pueden ser obtenidas vía combinación convexa. Además, con otras elecciones de λ también se pueden obtener otras FLM entre f_m y f_M que no están en E_f .

3. Sistema de inferencia difusa basado en relaciones booleanas y kleeneanas (FIS-BKR)

El sistema de inferencia difusa basado en relaciones⁶ booleanas y kleeneanas (Fuzzy Inference System based on Boolean and Kleenean Relations (FIS-BKR)) fue propuesto por primera vez en el año 2001 por Soriano *et. al.* [3]. La propuesta original ha sido ampliada considerablemente tras casi 16 años de investigación continua y varios nombres han aparecido con el tiempo. La Figura 1 muestra el diagrama de bloques actual. A continuación se presenta el método de diseño propuesto en [7].

Fusificador El fusificador usado es *unitario* (o *singleton* en inglés). Para un \mathbf{x}^* dado, el conjunto difuso unitario A (singleton) a la salida del fusificador está definido por la función de pertenencia:

$$\mu_A(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} = \mathbf{x}^*, \\ 0, & \mathbf{x} \neq \mathbf{x}^*, \end{cases}$$

donde $\mathbf{x} = (x_1, \dots, x_n) \in X$ es el vector de variables de entrada, $\mathbf{x}^* = (x_1^*, \dots, x_n^*) \in X$ es un valor particular tomado en la entrada del fusificador y $X = X_1 \times \dots \times X_n \subseteq \mathbb{R}^n$ es el universo de entrada.

⁶La palabra *relación* usada originalmente por Soriano *et. al.* [3] (y conservada hasta la actualidad) siempre ha sido usada como sinónimo de la palabra *fórmula*. Nunca ha tomado el significado clásico de las matemáticas como un subconjunto del producto cartesiano de conjuntos. Por razones históricas se ha conservado esta palabra queriendo decir que se trata de un modelo *basado en fórmulas*.

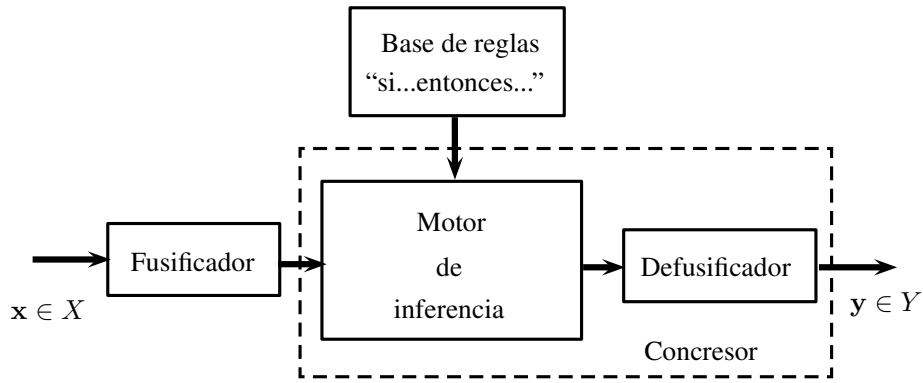


Figura 1. Sistema de inferencia difusa basado en relaciones booleanas y kleeneanas (FIS-BKR). Tomado de [7]

Base de reglas Para la j -ésima salida y_j ($j = 1, 2, \dots, m$) del vector de salidas $\mathbf{y} \in Y$, donde $\mathbf{y} = (y_1, \dots, y_m)$ y $Y = Y_1 \times \dots \times Y_m \subseteq \mathbb{R}^m$, una regla típica de la base de reglas luce como:

Si x_1 (no) es A_1 y/o \dots y/o x_n (no) es A_n entonces y_j (no) es B ,

donde A_i ($i = 1, 2, \dots, n$) es uno de los posibles conjuntos difusos formulados sobre el i -ésimo universo de entrada X_i y B es uno de los posibles conjuntos difusos formulados sobre el j -ésimo universo de salida Y_j .

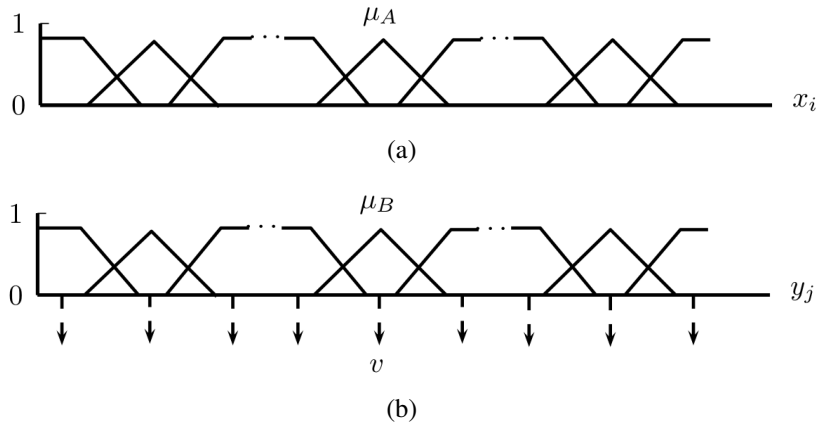


Figura 2. (a) Ejemplo de formulación de conjuntos difusos en el i -ésimo universo de entrada X_i ($i = 1, 2, \dots, n$). Se muestra la función de pertenencia de uno de los conjuntos difusos A . (b) Ejemplo de formulación de conjuntos difusos en el j -ésimo universo de salida Y_j ($j = 1, 2, \dots, m$). Se muestra la función de pertenencia de uno de los conjuntos difusos B con su valor característico $v \in Y_j$ llamado *actuador virtual*.

Las operaciones lógicas “y”, “o” y “no” son las dadas por la lógica difusa estándar⁷ y/o lógica booleana. La formulación de los conjuntos difusos sobre cada universo de entrada y salida

⁷Es conocido que cualquier operación lógica difusa (tipo-1) siempre se reduce a la correspondiente de la lógica booleana cuando el conjunto de valores de verdad I es cambiado por B . Además, por simplicidad en la explicación, en este artículo se usa la operación de complemento estándar $a' = 1 - a$, $a \in I$, en la lógica difusa. Sin embargo, esta operación puede ser cambiada por cualquiera que sea monótona decreciente e involutiva sin alterar significativamente lo que se dice en este documento. Ver [7] para más detalles.

puede seguir algún criterio (Figura 2(a) y Figura 2(b)), por ejemplo, puede obedecer a la definición de *partición difusa* [17, def. 7.4.1]. Además, cada conjunto difuso B formulado sobre el j -ésimo universo de salida $Y_j \subseteq \mathbb{R}$ debe tener un único valor característico $v \in Y_j$ llamado *actuador virtual*.

Motor de inferencia El diseño del motor de inferencia se realiza en dos etapas:

1. En la primera se hace una aproximación a través de lógica booleana,
2. y en la segunda se hace una extensión a la lógica difusa estándar a través de lógica trivalente.

A continuación se explican los pasos a seguir en cada etapa.

Etapas 1: Aproximación a través de lógica booleana

1. **Aproximación de conjuntos difusos mediante conjuntos bivalentes** Todos los conjuntos de los antecedentes de las reglas son aproximados mediante conjuntos bivalentes, esto es, si A es uno de los posibles conjuntos difusos formulados sobre el i -ésimo universo de entrada $X_i \subseteq \mathbb{R}$ ($i = 1, 2, \dots, n$), entonces este conjunto es aproximado mediante un conjunto bivalente \hat{A} con función de pertenencia bivalente (es decir, $\mu_{\hat{A}} \in \{0, 1\}$).
2. **Codificación bivalente de las regiones aproximadas de operación** Cada conjunto bivalente \hat{A} formulado en el paso anterior realiza una partición del universo de entrada X_i ($i = 1, 2, \dots, n$) en la siguiente forma:
 - (a) **Codificación con 0:** para aquellos $x_i \in X_i$ tales que $\mu_{\hat{A}}(x_i) = 0$ y
 - (b) **Codificación con 1:** para aquellos $x_i \in X_i$ tales que $\mu_{\hat{A}}(x_i) = 1$.

En este paso es posible realizar una codificación bivalente de todas las posibles regiones aproximadas de operación⁸ mediante una tabla de verdad de lógica bivalente (ceros y unos), basados en la partición descrita anteriormente. Cada codificación corresponderá a una fila de la tabla. Si después de codificar todas las posibles regiones aproximadas de operación aún existen códigos por asignar, estos códigos son tratados como condiciones *no importa*.

3. **Asignación de las acciones del sistema en las regiones aproximadas de operación** Sobre la tabla de verdad del paso anterior, crear una función booleana \hat{f} por cada actuador virtual $v \in Y_j$ ($j = 1, 2, \dots, m$) perteneciente a cada conjunto difuso B formulado sobre el j -ésimo universo de salida $Y_j \subseteq \mathbb{R}$. El valor de $\hat{f} \in \{0, 1\}$ en la respectiva codificación (fila de la tabla) dependerá de la acción que tomará el sistema en la respectiva región aproximada de operación. Cada función \hat{f} es llamada *salida de activación* (del respectivo actuador virtual v) y corresponderá con una columna de la tabla. Sobre las filas de la tabla consideradas como condiciones *no importa* el valor de \hat{f} es arbitrario y puede ser elegido con algún criterio dado por el diseñador.

⁸Una región de operación es un subconjunto del universo de entrada $X = X_1 \times \dots \times X_n \subseteq \mathbb{R}^n$.

4. **Obtención de fórmulas** Para cada función booleana \hat{f} creada en el paso anterior obtener su forma normal (disyuntiva o conjuntiva) booleana, donde las variables de dicha fórmula son las funciones de pertenencia de cada conjunto bivalente.
5. **Simplificación (optimización) de fórmulas** Para cada fórmula obtenida en el paso anterior aplicar la versión de lógica booleana del método de simplificación⁹ (optimización) de fórmulas propuesto en [20].

Etapa 2: Extensión a la lógica difusa mediante lógica trivalente de Kleene

1. **Retrosustitución de conjuntos difusos** Todos los conjuntos bivalentes \hat{A} usados como aproximación en la primera etapa son nuevamente sustituidos por los originales conjuntos difusos A . Las funciones de pertenencia de estos últimos son multivalentes (es decir, $\mu_A \in [0, 1]$).
2. **Codificación trivalente de las regiones de operación** Dado un $\varepsilon \in [0, 1]$ definido por el diseñador, tal que $0 < \varepsilon \leq \xi$, donde ξ es el punto fijo¹⁰ de la operación lógica “no” (o complemento (')), cada conjunto difuso A formulado en el paso anterior realiza una partición del universo de entrada X_i en la siguiente forma:
 - (a) **Codificación con 0:** para aquellos $x_i \in X_i$ tales que $\mu_A(x_i) \in [0, \varepsilon)$,
 - (b) **Codificación con u :** para aquellos $x_i \in X_i$ tales que $\mu_A(x_i) \in [\varepsilon, \varepsilon')$ y
 - (c) **Codificación con 1:** para aquellos $x_i \in X_i$ tales que $\mu_A(x_i) \in (\varepsilon', 1]$.

En este paso es posible realizar una codificación trivalente de todas las posibles regiones de operación mediante una tabla de verdad de lógica trivalente (0, u y 1), basados en la partición descrita anteriormente. Cada codificación corresponderá con una fila de la tabla. Esta codificación trivalente debe preservar en su totalidad la codificación bivalente obtenida en la primera etapa. De esta forma la tabla de verdad de lógica trivalente es una extensión de la correspondiente de lógica bivalente.

3. **Extensión a la lógica difusa (por medio de la lógica trivalente de Kleene)** Usar el método de extensión (Método 2) para extender cada función booleana \hat{f} (salidas de activación) obtenida en la primera etapa a una (o varias) funciones kleeneanas regulares f .
4. **Selección de las extensiones regulares adecuadas** Por cada función booleana \hat{f} obtenida en la primera etapa, seleccionar una función kleeneana regular f (de las varias existentes) encontradas en el paso anterior. Esta selección puede hacerse con algún criterio que depende de la aplicación.
5. **Obtención de fórmulas lógicas difusas** Para cada función kleeneana regular f seleccionada en el paso anterior obtener su forma normal (disyuntiva o conjuntiva) kleeneana.

⁹El método de simplificación de fórmulas propuesto en [20] no se explica en este documento debido a consideraciones de espacio.

¹⁰El punto fijo del complemento (') es un valor $\xi \in [0, 1]$ tal que $\xi' = \xi$. Este valor existe y es único para todo complemento monótono decreciente e involutivo. Por ejemplo, para el complemento $a' = 1 - a$ de la lógica difusa estándar usada en este documento se tiene $\xi = 1/2$.

6. Simplificación (optimización) de fórmulas lógicas difusas Para cada fórmula obtenida en el paso anterior aplicar la versión de lógica trivalente de Kleene del método de simplificación (optimización) de fórmulas propuesto en [20].

Defusificador El vector de salidas $\mathbf{y} \in Y$, donde $\mathbf{y} = (y_1, \dots, y_m)$ y $Y = Y_1 \times \dots \times Y_m \subseteq \mathbb{R}^m$, se calcula por medio de un defusificador basado en fórmulas lógicas (booleanas y/o kleeneanas). Estas fórmulas ponderan el valor del respectivo actuador.

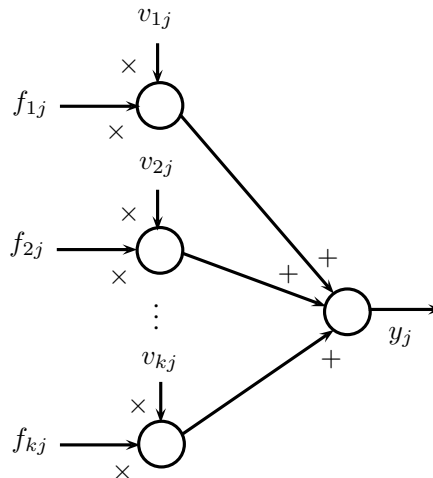


Figura 3. Diagrama de bloques del defusificador para la j -ésima salida y_j ($j = 1, 2, \dots, m$), basado en las fórmulas lógicas (booleanas y/o kleeneanas) f_{ij} ($i = 1, 2, \dots, k$) y actuadores virtuales v_{ij}

La j -ésima salida y_j ($j = 1, 2, \dots, m$), correspondiente al j -ésimo universo de salida Y_j donde se tienen k actuadores virtuales v_{ij} ($i = 1, 2, \dots, k$) que están siendo ponderados por k formulas lógicas f_{ij} dadas por el motor de inferencia, está dada por (Figura 3):

$$y_j = f_{1j}v_{1j} + f_{2j}v_{2j} + \dots + f_{kj}v_{kj} = \sum_{i=1}^k f_{ij}v_{ij}.$$

De acuerdo a Espitia y Soriano [9], [21], v_{ij} es llamado *actuador virtual*, f_{ij} es llamada *salida de activación* y el producto $f_{ij}v_{ij}$ es llamado *salida virtual*. Estos nombres corresponden al hecho que en lógica booleana $f_{ij} \in \{0, 1\}$ y en lógica difusa $f_{ij} \in [0, 1]$, por lo tanto f_{ij} (al ser adimensional y construido usando la respectiva lógica) actúa como un *activador/desactivador* del actuador v_{ij} . Además, v_{ij} al ser un elemento del respectivo universo de salida Y_i tiene las mismas dimensiones de este último, por lo que funciona como un *actuador*. El término *virtual* corresponde al hecho que v_{ij} (dependiendo de la aplicación) podría (o no) tener alguna correspondencia con algún elemento físico real donde el sistema difuso está actuando.

Concesor Históricamente se ha denominado *Concesor* a la unión entre el *Motor de Inferencia* y el *Defusificador* (Figura 1). El *Concesor* es aquel bloque que realiza la *concreción*¹¹ de la información difusa.

¹¹(Del lat. *concretio*, *-ōnis*). Acción y efecto de concretar [22].

4. FIS-BKR con combinador convexo

En algunas aplicaciones la selección de las extensiones regulares en el paso 4 de la segunda etapa del diseño del motor de inferencia puede ser una tarea difícil, principalmente por su gran número. La propuesta dada en [1] está en la adición de un bloque llamado *Combinador Convexo*¹² justo antes del defusificador y después del motor de inferencia. Esto permite integrar todas las extensiones regulares kleeneanas. En la Figura 4 se muestra el diagrama de bloques del combinador convexo.

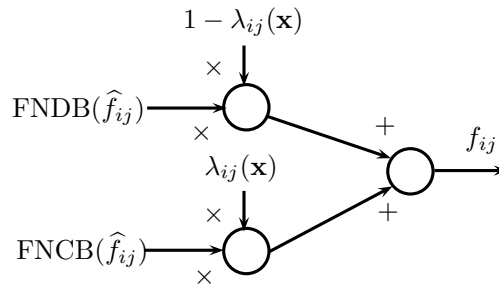


Figura 4. Diagrama de bloques del combinador convexo con coeficiente $\lambda_{ij}(\mathbf{x}) \in [0, 1]$ para la i -ésima función booleana \hat{f}_{ij} ($i = 1, 2, \dots, k$) formulada sobre el j -ésimo universo de salida ($j = 1, 2, \dots, m$). FNCB(\hat{f}_{ij}) y FNDB(\hat{f}_{ij}) corresponden a la forma normal conjuntiva y disyuntiva booleanas de \hat{f}_{ij} , respectivamente.

El proceso de diseño del FIS-BKR con combinador convexo es el mismo al presentado en la Sección 3 con las siguientes modificaciones en el diseño del motor de inferencia:

Etapa 1: Aproximación a través de lógica booleana

1. **Aproximación de conjuntos difusos mediante conjuntos bivalentes.** Igual al paso 1, etapa 1, Sección 3.
2. **Codificación bivalente de las regiones aproximadas de operación** Igual al paso 2, etapa 1, Sección 3.
3. **Asignación de las acciones del sistema en las regiones aproximadas de operación** Igual al paso 3, etapa 1, Sección 3.
4. **Obtención de fórmulas** Para cada función booleana \hat{f} creada en el paso anterior obtener su forma normal disyuntiva y conjuntiva booleana, donde las variables de dicha fórmula son las funciones de pertenencia de cada conjunto bivalente. La forma normal disyuntiva booleana de \hat{f} corresponde a \hat{f}_m (extensión regular mínima) y la forma normal conjuntiva booleana de \hat{f} corresponde a \hat{f}_M (extensión regular máxima).
5. **Simplificación (optimización) de fórmulas** Este paso no se realiza.

Etapa 2: Extensión a la lógica difusa mediante lógica trivalente de Kleene

1. **Retrosustitución de conjuntos difusos** Igual al paso 1, etapa 2, Sección 3.

¹²El uso de combinadores convexos para combinar dos o más respuestas es una idea con antecedentes. Por ejemplo, en [23] se usa sobre filtros adaptativos.

2. **Codificación trivalente de las regiones de operación** Este paso no se realiza.
3. **Extensión a la lógica difusa (por medio de la lógica trivalente de Kleene)** Este paso no se realiza.
4. **Selección de las extensiones regulares adecuadas** Este paso no se realiza.
5. **Obtención de fórmulas lógicas difusas** Este paso no se realiza.
6. **Simplificación (optimización) de fórmulas lógicas difusas** Este paso no se realiza.
7. **Obtención de combinaciones convexas** Para cada extensión mínima \hat{f}_m y máxima \hat{f}_M encontrada en la etapa 1 obtener su combinación convexa difusa con coeficiente λ definido por el diseñador. El coeficiente λ puede ser seleccionado con algún criterio dependiendo de la aplicación.

5. Aplicación

La siguiente aplicación muestra el uso del método de diseño de controladores FIS-BKR con combinador convexo difuso. Esta aplicación fue presentada por Espitia [9] y Salazar [20], y previamente por Lara [12], Hernández [11] y Tunyasirut [10] donde se propusieron otras soluciones de control empleando lógica difusa.

5.1. Modelo dinámico

Se desea mantener el nivel $h(t)$, siendo t la variable tiempo, de cierta cantidad de líquido en un tanque cilíndrico en posición horizontal donde el flujo de salida $\phi_s(t)$ varía a través de la válvula V_S . Existen dos válvulas de flujo de entrada. La primera (V_G) deja entrar un flujo grande $\phi_g(t)$ de líquido mientras que la segunda (V_P) deja entrar un flujo pequeño $\phi_p(t)$. El flujo total de entrada está dado por $\phi_e(t) = \phi_g(t) + \phi_p(t)$ como se muestra en la Figura 5(a). Las variables y constantes del sistema con sus unidades se muestran en la Tabla III.

Tabla III. Variables y constantes del sistema de llenado de un tanque cilíndrico en posición horizontal

Símbolo	Significado	Valor	Unidad
H	Altura del tanque	0.4	m
$R = H/2$	Radio del tanque	0.2	m
h	Altura del nivel del líquido		m
H_{ref}	Altura de referencia	0.3	m
L	Longitud del tanque	0.6	m
ρ	Densidad del líquido		kg/m ³
v	Volumen del líquido dentro del tanque		m ³
$m = \rho v$	Masa del líquido		kg
$\phi_e = \phi_g + \phi_p$	Flujo total de entrada		m ³ /s
ϕ_g	Flujo de entrada por la válvula V_G		m ³ /s
ϕ_p	Flujo de entrada por la válvula V_P		m ³ /s
$\phi_s = C_{V_S} \sqrt{h}$	Flujo de salida por la válvula V_S		m ³ /s
C_{V_S}	Constante de la válvula V_S	1.6×10^{-4}	m ^{5/2} /s
v_p	Actuador virtual asociado a la válvula V_P	0.5×10^{-4}	m ³ /s
v_g	Actuador virtual asociado a la válvula V_G	1.5×10^{-4}	m ³ /s

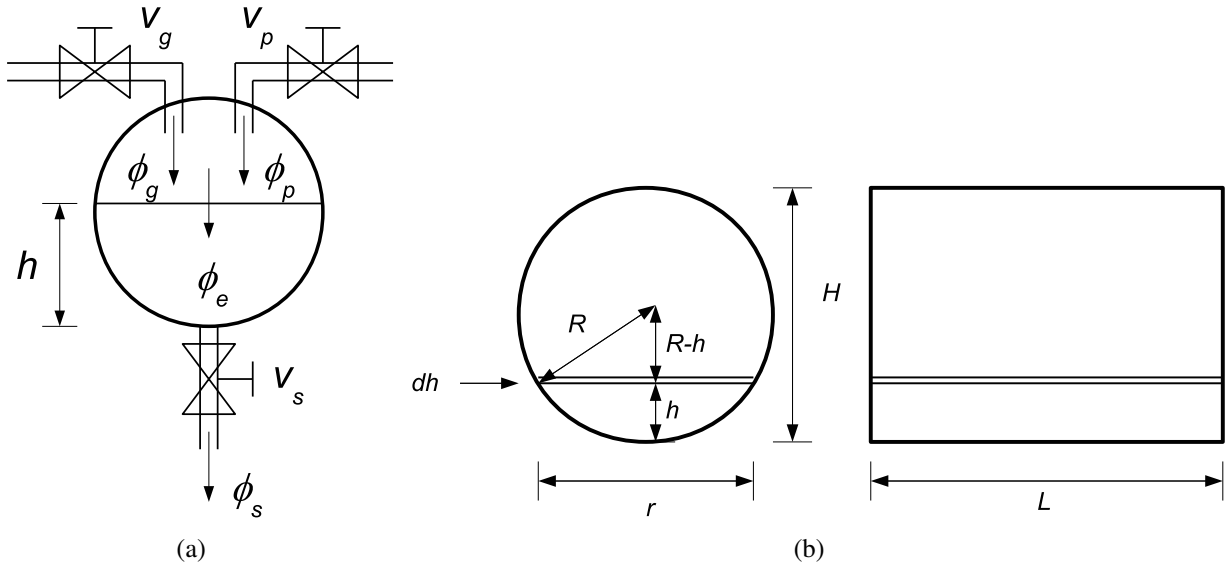


Figura 5. (a) Sistema de llenado de un tanque cilíndrico en posición horizontal. (b) Geometría del tanque cilíndrico en posición horizontal

La Figura 5(b) muestra la geometría del tanque donde $(r/2)^2 + (R - h)^2 = R^2$, por lo tanto al despejar r se tiene $r = 2\sqrt{2Rh - h^2}$. El diferencial de volumen del líquido en el tanque está dado por $dv = Lrdh = 2L\sqrt{Hh - h^2}dh$, siendo $H = 2R$. Para esta aplicación se supone que la densidad del líquido del tanque (ρ) es constante y la misma que la del líquido entrante y saliente. La ecuación de conservación de la masa para este sistema es: $dm/dt = dm_e/dt - dm_s/dt$ donde m es la masa de líquido del tanque, m_e es la masa entrante y m_s la masa saliente. Esta ecuación se puede simplificar a $dv/dt = \phi_e - \phi_s$ donde $\phi_e = dv_e/dt$ y $\phi_s = dv_s/dt$ son los flujos de entrada y salida, v_e y v_s son los volúmenes de líquido entrante y saliente, y usando las ecuaciones $m = \rho v$, $m_e = \rho v_e$ y $m_s = \rho v_s$. Sustituyendo el diferencial de volumen dv encontrado antes y despejando para dh/dt se obtiene: $dh/dt = (\phi_e - \phi_s)/(2L\sqrt{Hh - h^2})$. Dado que la válvula de salida se encuentra en la parte inferior del tanque y actúa por acción de la gravedad, el flujo de salida está dado por $\phi_s = C_{V_s}\sqrt{h}$, donde C_{V_s} es la constante de la válvula V_s . Se obtiene entonces la siguiente ecuación que modela el sistema:

$$\frac{dh}{dt} = \frac{\phi_e - C_{V_s}\sqrt{h}}{2L\sqrt{Hh - h^2}}, \quad 0 \leq h \leq H.$$

La Figura 6(a) presenta un diagrama de bloques de la anterior ecuación diferencial.

5.2. Diseño de un controlador FIS-BKR con combinador convexo

La Figura 6(b) muestra el diagrama de bloques de la estrategia de control elegida. Esta consiste en un esquema de retroalimentación de la variable nivel h , comparada contra una altura de referencia H_{ref} , para obtener el error $e = H_{ref} - h$, que junto a su derivada, serán las entradas del controlador FIS-BKR con combinador convexo tipo proporcional-derivativo (PD). La salida del controlador es el flujo de entrada al tanque ϕ_e .

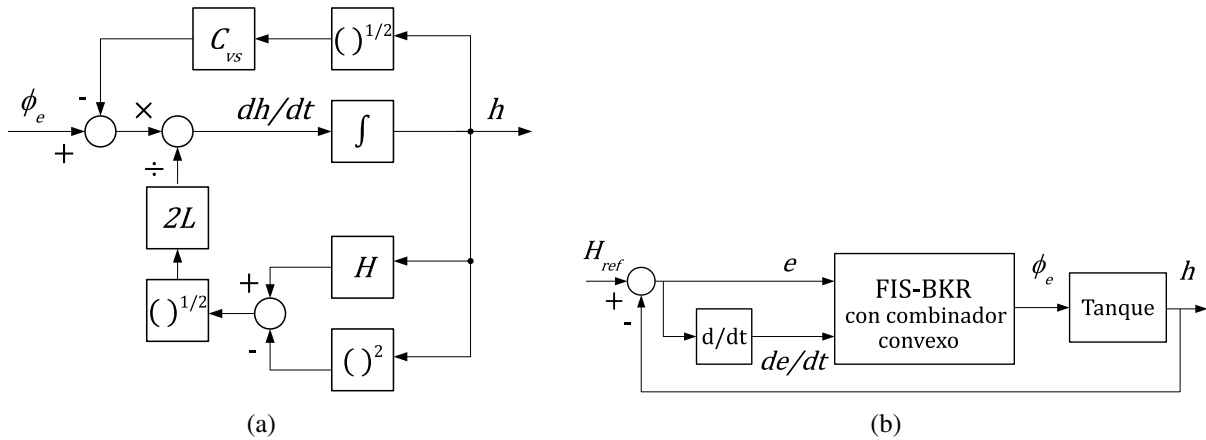


Figura 6. (a) Diagrama de bloques del sistema de llenado del tanque cilíndrico en posición horizontal. (b) Control usando FIS-BKR con combinador convexo.

Aproximación a través de lógica booleana Para el error de nivel e se definen dos conceptos: *error positivo* (E_P) y *error negativo* (E_N) con las siguientes funciones de pertenencia (Figura 7):

$$\mu_{E_P}(e) = \begin{cases} 1, & e \geq 0.02 \text{ m,} \\ 0, & \text{otro caso,} \end{cases} \quad \mu_{E_N}(e) = \begin{cases} 1, & e \leq -0.02 \text{ m,} \\ 0, & \text{otro caso.} \end{cases} \quad (1)$$

Para la variable de/dt se define un solo concepto: *derivada positiva* (D_P) con la siguiente función de pertenencia:

$$\mu_{D_P}(de/dt) = \begin{cases} 1, & de/dt \geq 0 \text{ m/s,} \\ 0, & \text{otro caso.} \end{cases} \quad (2)$$

Para la variable de flujo de entrada ϕ_e se definen cuatro conceptos: *flujo grande* (F_G), *flujo mediano* (F_M), *flujo pequeño* (F_P) y *flujo nulo* (F_N) con las siguientes funciones de pertenencia:

$$\mu_{F_G}(\phi_e) = \begin{cases} 1, & \phi_e = v_p + v_g = 2 \times 10^{-4} \text{ m}^3/\text{s,} \\ 0, & \text{otro caso,} \end{cases} \quad \mu_{F_M}(\phi_e) = \begin{cases} 1, & \phi_e = v_g = 1.5 \times 10^{-4} \text{ m}^3/\text{s,} \\ 0, & \text{otro caso,} \end{cases}$$

$$\mu_{F_P}(\phi_e) = \begin{cases} 1, & \phi_e = v_p = 0.5 \times 10^{-4} \text{ m}^3/\text{s,} \\ 0, & \text{otro caso,} \end{cases} \quad \mu_{F_N}(\phi_e) = \begin{cases} 1, & \phi_e = 0 \text{ m}^3/\text{s,} \\ 0, & \text{otro caso.} \end{cases}$$

La Tabla IV muestra las acciones de control, similares a las presentadas en [9], con funciones de activación \hat{f} y \hat{F} para las válvulas V_G y V_P para así obtener los valores de flujo grande, mediano, pequeño y nulo con la activación y desactivación de V_G y V_P .

Tabla IV. Tabla de verdad para las funciones de activación \hat{f} y \hat{F} de V_P y V_G respectivamente

μ_{E_P}	μ_{E_N}	μ_{D_P}	\hat{f}	\hat{F}	Regla
1	0	1	1	1	Si e es E_P y e no es E_N y de/dt es D_P entonces ϕ_e es F_G
1	0	0	1	0	Si e es E_P y e no es E_N y de/dt no es D_P entonces ϕ_e es F_P
0	1	1	0	1	Si e no es E_P y e es E_N y de/dt es D_P entonces ϕ_e es F_M
0	1	0	0	1	Si e no es E_P y e es E_N y de/dt no es D_P entonces ϕ_e es F_M
0	0	1	0	0	Si e no es E_P y e no es E_N y de/dt es D_P entonces ϕ_e es F_N
0	0	0	0	0	Si e no es E_P y e no es E_N y de/dt no es D_P entonces ϕ_e es F_N

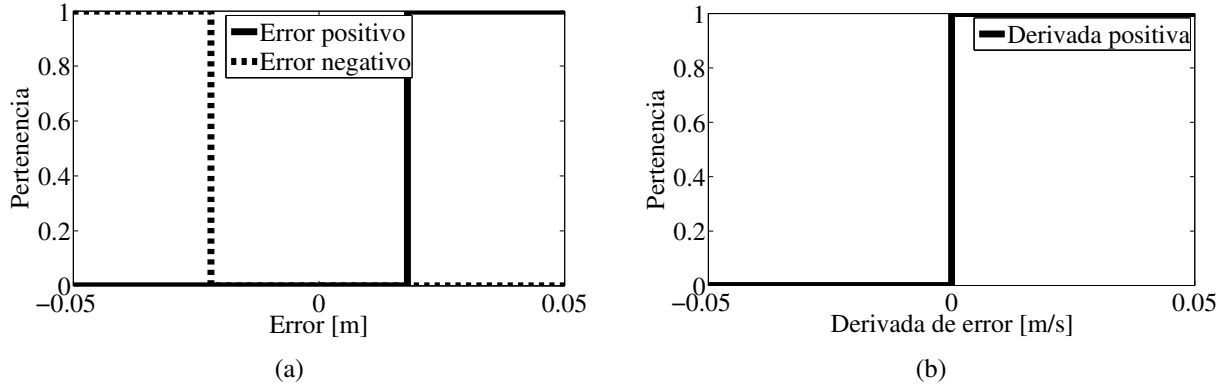


Figura 7. (a) Conjuntos asociados al error positivo y negativo. (b) Conjunto asociado a la derivada del error positiva

Extensión a la lógica difusa estándar y uso del combinador convexo El primer paso consiste en suavizar los conceptos booleanos cambiándolos por difusos. Los conjuntos difusos para los respectivos conceptos del antecedente son los siguientes:

$$\mu_{E_P}(e) = \begin{cases} 0, & e < 0 \text{ m}, \\ 25e, & 0 \text{ m} \leq e \leq 0.04 \text{ m}, \\ 1, & e > 0.04 \text{ m}, \end{cases} \quad \mu_{E_N}(e) = \begin{cases} 1, & e < -0.04 \text{ m}, \\ -25e, & -0.04 \text{ m} \leq e \leq 0 \text{ m}, \\ 0, & e > 0 \text{ m}, \end{cases}$$

$$\mu_{D_P}(de/dt) = \begin{cases} 0, & de/dt < -0.02 \text{ m/s}, \\ 25de/dt + 0.5, & -0.02 \text{ m/s} \leq de/dt \leq 0.02 \text{ m/s} \\ 1, & de/dt > 0.02 \text{ m/s}. \end{cases}$$

Las funciones de pertenencia para los conjuntos difusos del consecuente son los mismos que para el caso booleano.

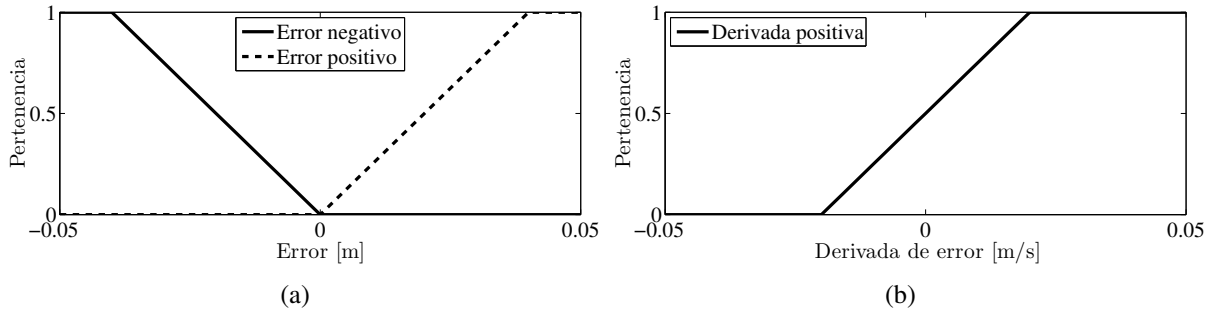


Figura 8. (a) Conjuntos difusos asociados al error positivo y negativo. (b) Conjunto difuso asociado a la derivada del error positiva

Las formas normales (disyuntiva y conjuntiva) booleanas para las funciones de activación \hat{f} y \hat{F} de la Tabla IV llevan a las extensiones kleeneanas extremas asociadas a \hat{f} y \hat{F} , las cuales son:

$$f_m = (\mu_{E_P} \wedge \mu'_{E_N} \wedge \mu_{D_P}) \vee (\mu_{E_P} \wedge \mu'_{E_N} \wedge \mu'_{D_P}),$$

$$f_M = (\mu_{E_P} \vee \mu'_{E_N} \vee \mu'_{D_P}) \wedge (\mu_{E_P} \vee \mu'_{E_N} \vee \mu_{D_P}) \wedge (\mu_{E_P} \vee \mu_{E_N} \vee \mu'_{D_P}) \wedge (\mu_{E_P} \vee \mu_{E_N} \vee \mu_{D_P}),$$

$$F_m = (\mu_{E_P} \wedge \mu'_{E_N} \wedge \mu_{D_P}) \vee (\mu'_{E_P} \wedge \mu_{E_N} \wedge \mu_{D_P}) \vee (\mu'_{E_P} \wedge \mu_{E_N} \wedge \mu'_{D_P}),$$

$$F_M = (\mu'_{E_P} \vee \mu_{E_N} \vee \mu_{D_P}) \wedge (\mu_{E_P} \vee \mu_{E_N} \vee \mu'_{D_P}) \wedge (\mu_{E_P} \vee \mu_{E_N} \vee \mu_{D_P}).$$

Estas fórmulas en lógica trivalente de Kleene pueden ser entendidas como fórmulas en lógica difusa estándar basados en la teoría expuesta en [13], [15]–[17].

Los coeficientes de la combinación convexa fueron tomados como conjuntos difusos con pertenencia constante. Esto significa que $\lambda_i(x) = \lambda_i \in [0, 1]$ para todo x , con $i = 1, 2$. Las expresiones para las funciones de activación f y F están dadas por:

$$f = \lambda_1 f_M + (1 - \lambda_1) f_m, \quad \lambda_1 \in [0, 1], \quad (3)$$

$$F = \lambda_2 F_M + (1 - \lambda_2) F_m, \quad \lambda_2 \in [0, 1]. \quad (4)$$

Para obtener la acción de control (flujo de entrada) ϕ_e se ponderan (3) y (4) con actuadores virtuales $v_p = 0.5 \times 10^{-4} \text{ m}^3/\text{s}$ y $v_g = 1.5 \times 10^{-4} \text{ m}^3/\text{s}$ como sigue:

$$\phi_e = v_p \cdot f + v_g \cdot F. \quad (5)$$

Notar que las unidades de ϕ_e son las mismas que las unidades de los actuadores virtuales v_p y v_g dado que las funciones de activación f y F son adimensionales. La Figura 9 muestra las superficies de control extremas para el flujo de entrada.

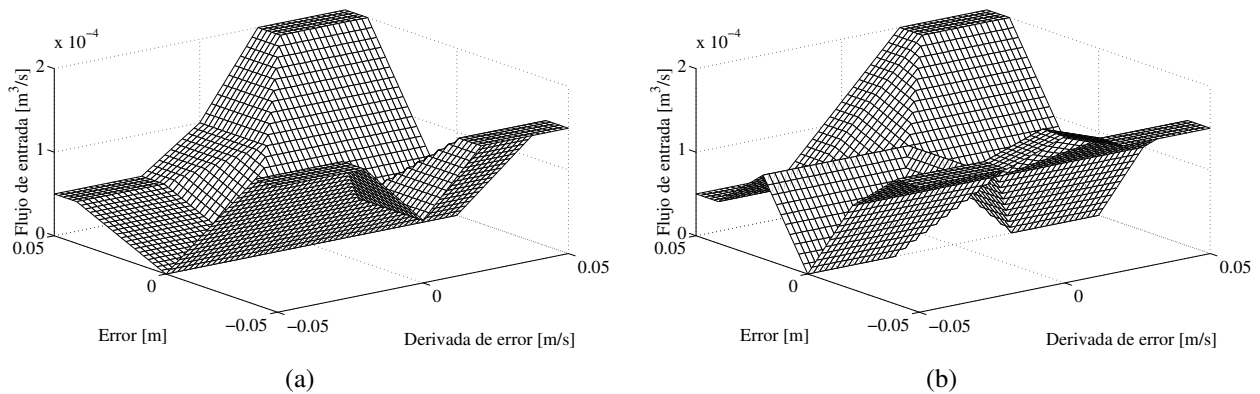


Figura 9. Superficies de control para el flujo de entrada ϕ_e cuando: (a) $\lambda_1 = 0$ y $\lambda_2 = 0$, (b) $\lambda_1 = 1$ y $\lambda_2 = 1$.

Cabe anotar que si las expresiones (3) y (4) son evaluadas para los conceptos de lógica booleana (1) y (2), las extensiones máxima y mínimas coinciden. Esto implica que $f = f_M = f_m$ y $F = F_M = F_m$, es decir, no hay sensibilidad a los parámetros de la combinación λ . En la Figura 10 se muestra el diagrama de bloques del controlador FIS-BKR diseñado.

5.3. Resultados

La implementación del sistema de control se realizó mediante la simulación del sistema dinámico en el entorno de programación visual Simulink de MATLAB¹³. Se usó un tiempo de simulación $T_s = 5000 \text{ s}$ y pasos de tiempo de $\Delta t = 5 \text{ s}$. Teniendo como flujo de entrada al tanque:

$$\phi_e = 0.5 \times 10^{-4} \cdot (\lambda_1 f_M + (1 - \lambda_1) f_m) + 1.5 \times 10^{-4} \cdot (\lambda_2 F_M + (1 - \lambda_2) F_m). \quad (6)$$

¹³Los archivos de simulación se pueden encontrar en el repositorio público [24].

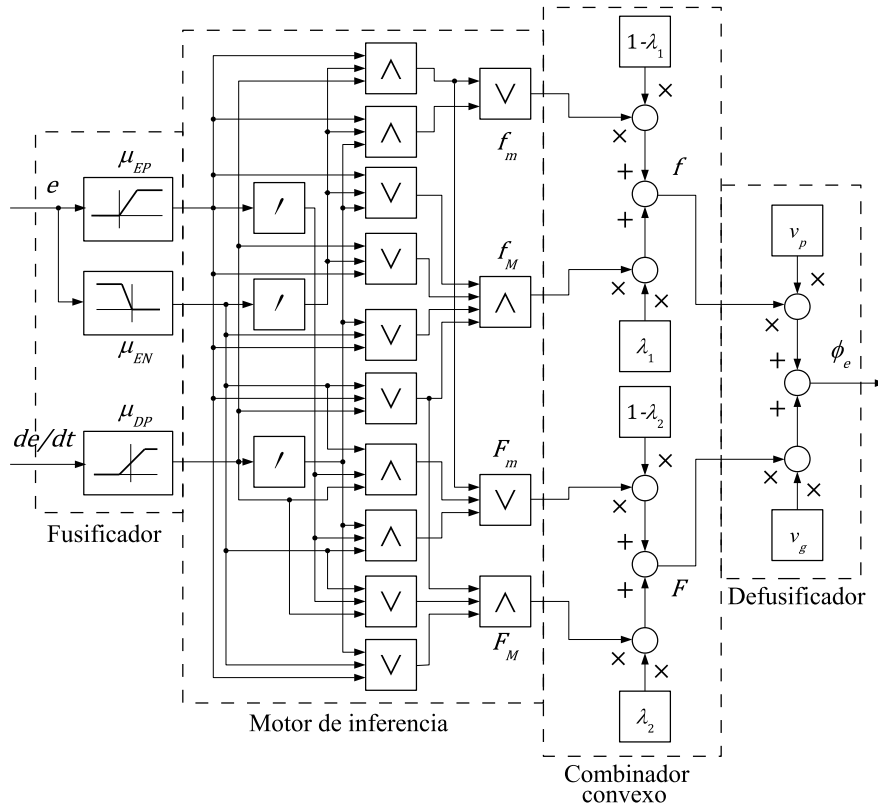


Figura 10. Diagrama de bloques del controlador FIS-BKR con combinador convexo diseñado

Al variar los parámetros λ_1 y λ_2 , el sistema controlado presenta distintas repuestas temporales. En la Figura 11 se presenta la respuesta del controlador booleano (sin suavización de los conceptos), al tiempo que se exhiben respuestas para algunos parámetros de combinación convexa. Se observa que los controladores presentan error en estado estacionario y diferentes tiempos de establecimiento. Llama la atención la respuesta del sistema para los parámetros $\lambda_1 = 0$ y $\lambda_2 = 0$ puesto que no hay establecimiento alguno del nivel. La razón de este comportamiento está asociada a la magnitud del flujo de entrada que en este caso es la acción de control mínima posible (ecuación (6)). Este comportamiento muestra clara evidencia de un hecho importante: no todo controlador FIS-BKR diseñado a partir de un controlador booleano que aproxima la referencia controla el sistema.

En la Figura 11 se observa que el control booleano presenta cierta variabilidad consecuencia de la discontinuidad de los conceptos, situación que no ocurre cuando estos son suavizados, como se propone en la metodología de diseño de controladores FIS-BKR.

Evaluación de índices de desempeño Es de nuestro interés evaluar para el sistema controlado los índices de desempeño ITSE e ISE. Para este caso se usó la definición discreta de tales índices donde se reemplaza la integral por una sumatoria y los valores de error son muestreados. Estos índices están definidos por:

$$ISE = \sqrt{\int_0^{T_s} e^2(t)dt}, \quad ITSE = \sqrt{\int_0^{T_s} te^2(t)dt},$$

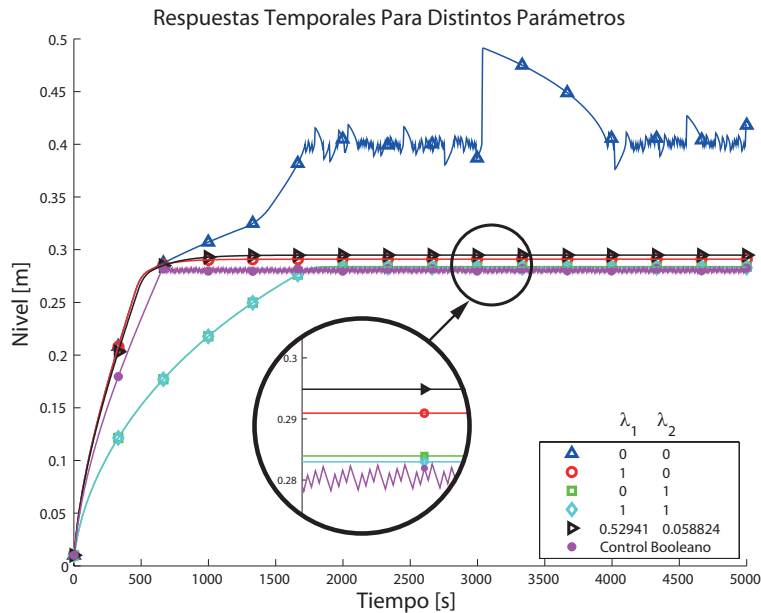


Figura 11. Respuesta temporal de la variable nivel para el sistema controlado para diferentes coeficientes de combinación convexa.

donde $e(t) = H_{ref} - h(t)$ es el error de nivel y T_s es el tiempo de simulación. Estos criterios de un solo número dan cuenta del cuadrado del error en el tiempo. El primero favorece sistemas con respuestas que se acercan a una referencia dada, mientras que el segundo, por tener en cuenta el tiempo usado para alcanzarla, es menor para sistemas que lo hacen rápidamente. La Figura 12 muestra cómo cambian los anteriores índices de desempeño con la variación de los parámetros λ_1 y λ_2 .

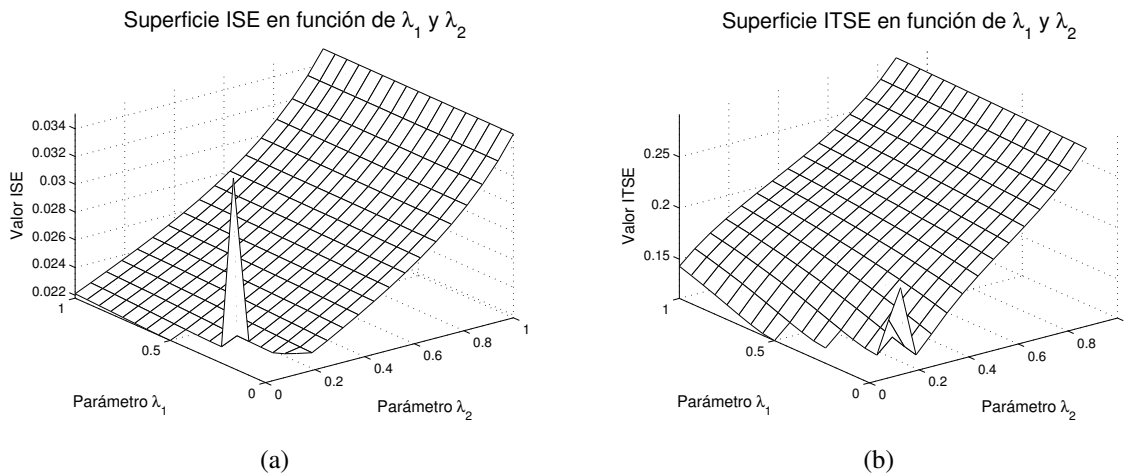


Figura 12. (a) Valores del índice ISE para diferentes valores de λ_1 y λ_2 . (b) Valores del índice ITSE para diferentes valores de λ_1 y λ_2 .

El proceso de optimización de los índices ISE e ITSE se realizó mediante la evaluación de una rejilla de 16×16 puntos en el dominio $[0, 1] \times [0, 1]$, el cual tomó un tiempo de 35 minutos y 30 segundos¹⁴. Se obtuvieron los parámetros óptimos $\lambda_1 = 0.52941$ y $\lambda_2 = 0.058824$ para ambos

¹⁴Proceso ejecutado en un equipo de computo con: procesador: AMD A8 2.90 GHz, memoria RAM: 4 GB, sistema

índices. La Tabla V resume los valores que optimizan los índices de estudio, además de presentar el desempeño del sistema de control booleano.

Tabla V. Índices ISE e ITSE para las extensiones extremas f_m, f_M, F_m, F_M y valores de los coeficientes λ_1 y λ_2 que minimizan ambos índices (negrilla).

λ_1	λ_2	ISE	ITSE
0	0	0.0496	1.1556
0	1	0.0350	0.1431
1	0	0.0221	0.3364
1	1	0.0351	0.3405
0.5294	0.0588	0.0224	0.1116
Booleano		0.0255	0.2389

En este caso el proceso de optimización llevó a una mejora de hasta el 1.427% para ISE y de hasta 21.99% para el ITSE con respecto al desempeño de las extensiones extremas. Sin embargo, cabe notar que no se puede concluir que este controlador óptimo sea mejor que cualquier extensión algebraica ya que esta aplicación se limitó exclusivamente a coeficientes con pertenencia constante. Por otro lado, la simple validación de si lo es o no resulta una tarea extensa como se verá a continuación.

Conteo de extensiones Por último, con el fin de mostrar al lector lo complejo que puede ser buscar la mejor extensión algebraica dado un índice de desempeño por el método tradicional (Sección 3), en la Tabla VI se muestran las extensiones posibles de la función \hat{f} de la Tabla IV obtenidas al aplicar el Método 2.

Tabla VI. Posibilidades para extender regularmente la función booleana \hat{f} de la Tabla IV.

(a)					(b)					
Cantidad de u 's (nivel de ambigüedad)	μ_{EP}	μ_{EN}	μ_{DP}	\hat{f}	Cantidad de u 's (nivel de ambigüedad)	μ_{EP}	μ_{EN}	μ_{DP}	\hat{f}	
0	1	0	1	1	1	0	u	1	$0, u$	
	1	0	0	1		0	u	0	$0, u$	
	0	1	1	0		1	1	u	-	
	0	1	0	0		1	0	u	$1, u$	
	0	0	1	0		0	1	u	$0, u$	
	0	0	0	0		0	0	u	$0, u$	
	1	1	1	-		u	u	1	u	
	1	1	0	-		u	u	0	u	
1	u	0	1	u	2	1	u	u	-	
	u	0	0	u		0	u	u	$0, u$	
	u	1	1	-		u	1	u	-	
	u	1	0	-		u	0	u	u	
	1	u	1	-		3	u	u	u	u
	1	u	0	-			u	u	u	u

Donde “-” representa una condición de *no importa* y pueden valer 0, 1 o u , teniendo en cuenta las condiciones de regularidad. Para conocer la cantidad de extensiones es necesario realizar el producto de las posibilidades de cada fila, lo cual, sin tener en cuenta las condiciones de *no importa*, es de a lo menos $2^6 = 64$. Esta sería la cantidad de funciones kleeneanas regulares que se tendrían

operativo: Windows 10 de 64 bit, Versión MATLAB: 7.12.0 (R2011a).

que evaluar como mínimo para elegir la mejor, no sin antes encontrar sus fórmulas lógicas lo cual de por sí es una tarea ardua.

6. Conclusiones y comentarios finales

Este artículo presentó nueva evidencia de las características favorables de controladores FIS-BKR con combinador convexo difuso cuando se evaluaron los índices de desempeño ISE e ITSE. Basados en hechos teóricos se puede asegurar que la estrategia de control FIS-BKR con combinador convexo difuso y su metodología de diseño generalizan el FIS-BKR actual, permitiendo así trabajar con un modelo más completo, como lo demuestra una mejora en el desempeño de 1.427% para el ISE y 21.99% en el caso del ITSE, comparados con las alternativas extremas del FIS-BKR y un controlador basado en lógica booleana.

Por el carácter global que tienen las funciones extremas, se puede decir que la información obtenida de la etapa de diseño con lógica booleana está siendo incorporada sin sesgo alguno y sin mayor complejidad; ya que se obtiene a partir de las alternativas extremas, sin necesidad de pasar por métodos computacionalmente costosos de deducción de fórmulas que requieren otras extensiones, solucionando en cierto modo algunas de las desventajas mencionadas por Salazar en [7].

Tanto el estudio de la simplificación de fórmulas representadas mediante formas normales, como la caracterización de las familias de coeficientes-parámetro para el proceso de sintonización serían importantes aportes en la fundamentación del uso del combinador convexo, que por su novedad, ha sido poco estudiada. Así mismo, consideramos que el estudio de otras técnicas de optimización en dominios compactos para el proceso de sintonización, diferentes al de “fuerza bruta” que fue utilizada, es indispensable cuando la cantidad de parámetros sea considerablemente grande, ya que hacerlo con una técnica poco sofisticada podría resultar inviable.

Referencias

- [1] J. Soriano and J. Chacón, “Concresor based on kleenean relations using convex combinator,” in *The 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Vancouver, BC, Canada, Jul. 2016, pp. 133–139. <https://doi.org/10.1109/FUZZ-IEEE.2016.7737679>. ↑ 8, 9, 13, 14, 19
- [2] G. Pangalos and G. Lichtenberg, “Approach to boolean controller design by algebraic relaxation for heating systems,” in *Proceedings of the 4th IFAC Conference on Analysis and Design of Hybrid Systems (ADHS 12)*, vol. 45, no. 9, Eindhoven, The Netherlands, Jun. 2012, pp. 210–215. <https://doi.org/10.3182/20120606-3-NL-3011.00070>. ↑ 8
- [3] J. J. Soriano, O. L. González Melo, F. V. Munar Fontecha, and A. A. Ramos Acosta, “Propuesta de concresor basado en relaciones booleanas,” *Ingeniería*, vol. 6, no. 2, pp. 42–50, 2001, Bogotá, Colombia. ↑ 8, 14
- [4] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller,” *International journal of man-machine studies*, vol. 7, no. 1, pp. 1–13, 1975. [https://doi.org/10.1016/S0020-7373\(75\)80002-2](https://doi.org/10.1016/S0020-7373(75)80002-2). ↑ 9
- [5] O. Salazar, J. Soriano, and H. Serrano, “From binary logic functions to fuzzy logic functions,” *Applied Mathematical Sciences*, vol. 7, no. 103, pp. 5129–5138, 2013, ISSN 1312-885X. ↑ 9, 12
- [6] M. Mukaidono, “Regular ternary logic functions—ternary logic functions suitable for treating ambiguity,” *IEEE Transactions on Computers*, vol. C-35, no. 2, pp. 179–183, Feb. 1986. <https://doi.org/10.1109/TC.1986.1676738>. ↑ 9, 11

- [7] O. Salazar, “Método de diseño y optimización de controladores difusos FIS-BBR cuasi-estándar por medio de lógicas clásica y trivalente de Kleene,” Trabajo de grado en Maestría en Ingeniería Industrial, Universidad Distrital Francisco José de Caldas. Facultad de Ingeniería, Bogotá D.C., Colombia, 2014. ↑ 9, 14, 15, 28
- [8] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965. [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X). ↑ 9, 14
- [9] H. E. Espitia, “Aplicación del congresor basado en relaciones booleanas para sistemas de lógica difusa tipo dos,” Trabajo de grado en Maestría en Ingeniería Industrial, Universidad Distrital Francisco José de Caldas. Facultad de Ingeniería, Bogotá D.C., Colombia, Jul. 2009. ↑ 9, 18, 20, 22
- [10] S. Tunyasrirut and S. Wangnipparnto, “Level control in horizontal tank by fuzzy-pid cascade controller,” *World Academy of Science, Engineering and Technology*, vol. 25, pp. 78–82, 2007. ↑ 9, 20
- [11] J. Hernández S. and C. Cuadra A., “Estrategias comparativas de control difuso y control con arreglo inverso Nyquist en sistemas multivariables a lineales,” *Revista Facultad de Ingeniería*, vol. 9, pp. 53–58, 2001. ↑ 9, 20
- [12] F. Lara-Rojo, E. N. Sánchez, and D. Zaldívar-Navarro, “Real-time fuzzy microcontroller for a didactic level system,” in *Electro 2001*, 2001, pp. 153–158. ↑ 9, 20
- [13] M. Gehrke, C. Walker, and E. Walker, “Normal forms and truth tables for fuzzy logics,” *Fuzzy Sets and Systems*, vol. 138, no. 1, pp. 25–51, Aug. 2003. [https://doi.org/10.1016/S0165-0114\(02\)00566-3](https://doi.org/10.1016/S0165-0114(02)00566-3). ↑ 10, 11, 12, 24
- [14] N. Takagi, K. Nakashima, H. Kikuchi, and M. Mukaidono, “A characterization of Kleenean functions,” in *Proceedings of the 25th International Symposium on Multiple-Valued Logic*, Bloomington, Indiana, May 1995, pp. 236–241. <https://doi.org/10.1109/ISMVL.1995.513537>. ↑ 10
- [15] M. Gehrke, C. Walker, and E. Walker, “A mathematical setting for fuzzy logic,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 5, no. 3, pp. 223–238, Jun. 1997. <https://doi.org/10.1142/S021848859700021X>. ↑ 11, 24
- [16] —, “Some comments on fuzzy normal forms,” in *Proceedings of the ninth IEEE International Conference on Fuzzy Systems*, vol. 2, San Antonio, Texas, May 2000, pp. 593–598. ↑ 11, 24
- [17] H. T. Nguyen and E. A. Walker, *A First Course in Fuzzy Logic*, 3rd ed. Boca Raton, Florida: Chapman & Hall/CRC, 2006. ↑ 11, 16, 24
- [18] O. Salazar and J. Soriano, “Convex combination and its application to fuzzy sets and interval-valued fuzzy sets I,” *Applied Mathematical Sciences*, vol. 9, no. 22, pp. 1061–1068, 2015, ISSN 1312-885X. ↑ 14
- [19] —, “Convex combination and its application to fuzzy sets and interval-valued fuzzy sets II,” *Applied Mathematical Sciences*, vol. 9, no. 22, pp. 1069–1076, 2015, ISSN 1312-885X. ↑ 14
- [20] O. Salazar, “Método de simplificación de fórmulas booleanas, borrosas cuasi-estándar y borrosas cuasi-estándar de intervalo por medio de álgebras finitas,” Trabajo de grado en Ingeniería Electrónica, Universidad Distrital Francisco José de Caldas. Facultad de Ingeniería, Bogotá D.C., Colombia, May 2011. ↑ 17, 18, 20
- [21] H. E. Espitia and J. J. Soriano, “Sistema de inferencia difusa basado en relaciones booleanas,” *Ingeniería*, vol. 15, no. 2, pp. 52–66, 2010, Bogotá. Colombia. ↑ 18
- [22] R. A. E., *Diccionario de la lengua española*, 22nd ed. Espasa Calpe, S.A., 2003, edición Electrónica. Versión 1.0. ↑ 18
- [23] J. Arenas-García, A. R. Figueiras-Vidal, and A. H. Sayed, “Mean-square performance of a convex combination of two adaptive filters,” *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 1078–1090, Mar. 2006. <https://doi.org/10.1109/TSP.2005.863126>. ↑ 19
- [24] J. D. Chacón. (2017). [Online]. Available: <https://github.com/juandados/ConvexCombinator>. ↑ 24

Juan David Chacón León

Ingeniero Electrónico, Universidad Distrital Francisco José de Caldas. Matemático, Universidad Nacional de Colombia, sede Bogotá. Magister en Ciencias Matemáticas, Universidad Nacional de Colombia, sede Bogotá. Correo electrónico: jdchaconl@correo.udistrital.edu.co

José Jairo Soriano Méndez

Ingeniero Electrónico, Universidad Distrital Francisco José de Caldas. Ingeniero Químico, Universidad Nacional de Colombia, sede Bogotá. Magister en Ingeniería Industrial, Universidad Distrital Francisco José de Caldas. Correo electrónico: jairosoriano@udistrital.edu.co

Omar Salazar Morales

Ingeniero Electrónico, Universidad Distrital Francisco José de Caldas. Magister en Ingeniería Industrial, Universidad Distrital Francisco José de Caldas. Magister en Ciencias Matemáticas, Universidad Nacional de Colombia, sede Bogotá. Estudiante de Doctorado en Ciencias Matemáticas, Universidad Nacional de Colombia, sede Bogotá. Correo electrónico: osalazarm@correo.udistrital.edu.co