ISSN 1870-4069

Sobre la algebrización de la lógica paraconsiente CG´3

Miguel Pérez Gaspar, Everardo Bárcenas

Universidad Nacional Autónoma de México,
Facultad de Ingeniería,
México

miguel.perez@fib.unam.mx, ebarcenas@unam.mx

Resumen. Las lógicas paraconsistentes son una familia de lenguajes formales con variadas aplicaciones en las ciencias computacionales, particularmente, en el campo de la inteligencia artificial, las lógicas paraconsistentes han sido exitosamente aplicadas en programación lógica, razonamiento difuso, y hasta en la construcción de redes neuronales paraconsistentes. G'3 es una lógica 3-valuada con un único valor de verdad representado por 1. CG'3 es una lógica paraconsistente, 3-valuada que extiende a G'3 con dos valores de verdad representados por 1 y 2. El estado del arte de CG'3 comprende una semántica de tipo Kripke y una axiomatización de tipo Hilbert inspirada en la técnica Lindenbaum-Łos. En este trabajo, se demuestra que existe un algebrización de la lógica CG'3 en el sentido de Blok y Pigozzi. Este resultado fundamenta el desarrollo de sistemas de razonamiento paraconsistentes.

Keywords: Lógicas Paraconsistentes, Algebrización Blok-Pigozzi, Lógica CG'3.

On the algebrization of paraconscious logic CG`3

Abstract. Paraconsistent logics are a family of formal languages with varied applications in computer science, particularly in the field of artificial intelligence, paraconsistent logics have been successfully applied in programming, fuzzy reasoning, and even in the construction of paraconsistent neural networks. G'3 is a 3-valued logic with a single truth value represented by 1. CG'3 is a paraconsistent, 3-valued logic that extends G'3 with two truth values represented by 1 and 2. The state of the CG'3 art comprises a Kripke-type semantics and a Hilbert-type axiomatization inspired by the Lindenbaum-Łos technique. In this work, it is shown that there is an algebrization of the CG'3 logic in the sense of Blok and Pigozzi. This result supports the development of paraconsistent reasoning systems.

Keywords: Paraconsistent Logics, Blok-Pigozzi Algebrization, CG'3 Logic.

1. Introducción

Es bien conocido el origen filosófico, y su aplicación en los fundamentos de las matemáticas, de los lenguajes lógicos, o simplemente lógicas. En el campo de las ciencias computacionales, diversas lógicas son utilizadas en la especificación de lenguajes de programación, esto es, los programas pueden ser caracterizados como demostraciones en sistemas de inferencia lógicos (isomorfismo Curry- Howard) [10]. Esto hace posible la participación de variadas lógicas en la Inteligencia Artificial (IA): desde usos en el proceso de implementación con información analítica, hasta en la demostración de correctitud de algoritmos. Existen incluso casos en los que ciertas teorías lógicas han servido para el desarrollo de paradigmas de programación, tales como la programación lógica [23].

Las lógicas paraconsistentes forman una familia de lenguajes diseñados para el análisis y razonamiento a partir de inconsistencias desde el punto de vista de la lógica clásica), como a menudo es requerido en diversas problemáticas en la IA, tales como el procesamiento de señales e imágenes y sistemas expertos [25]. Dentro de las familiaS de lógicas paraconsistentes, las lógicas anotadas, que abarcan la teoría de conjuntos difusos, son las que más ampliamente han sido aplicadas en la IA [2]. Otro campo de aplicación de las lógicas paraconsistentes es el de razonamiento no monótono, concepto fundamental en el desarrollo de sistemas inteligentes. En [3,4] se puede encontrar una semántica estándar para el razonamiento no monótono para lógicas anotadas y programas lógicos anotados.

Las lógicas multivaluadas son lógicas no clásicas [5]. Al igual que en la lógica clásica, las lógicas multivaluadas también disfrutan del principio de funcionalidad de verdad, a saber, el valor de verdad de una oración compuesta es determinada a través de los valores de verdad de sus oraciones componentes, y permanece igual cuando una de sus oraciones componentes se reemplaza por otra oración con el mismo valor de verdad. Sin embargo, en contraste con el caso clásico, las lógicas multivaluadas no restringen el número de valores de verdad a solo dos, un conjunto más grande de grados de verdad es la característica distintiva en el contexto de muchos valores.

En [14], podemos encontrar un resumen detallado de lógicas multivaluadas. Algunos sistemas de lógicas multivaluados se presentan como familias de sistemas de valores finitos e infinitos uniformemente definidos, por ejemplo, la lógica de Lukasiewicz, la lógica de Gödel, los sistemas basados en la norma t, los sistemas 3-valuados, el sistema de 4 valores de Dunn-Belnap, Sistemas de productos. Los principales tipos de cálculo lógico para sistemas de lógicas multivaluadas son el cálculo de tipo Hilbert, el cálculo de secuentes de tipo Gentzen o Tableaux [14]. Una amplia clase de lógicas infinitamente valoradas presentadas por [24].

La lógica clásica, así como la lógica intuicionista sufren un inconveniente a la hora de razonar con información inconsistente. Según el principio de explosión, también conocido como “ex contradictione sequitur quodlibet”, toda teoría o base de conocimiento inconsistente es totalmente trivial, lo que hace que estas lógicas sean inútiles para razonar con inconsistencias. Como resultado, las alternativas a la lógica clásica que no tienen este inconveniente tienen evolucionado, llamados enfoques “paraconsistentes”. En 1954 F. Asenjo, en su disertación doctoral, propone por primera vez utilizar lógicas multivaluadas para generar lógicas paraconsistentes (lógicas cuya relación de consecuencia lógica semántica o teoría de prueba no es explosiva [21]).

El enfoque de muchos valores es abandonar esta suposición clásica y permitir más de dos valores de verdad. La estrategia más común es usar tres valores de verdad: verdadero, falso y ambos (verdadero y falso) para las evaluaciones de fórmulas.

George Boole introdujo el álgebra de la lógica o lógica algebraica en [7], como un sistema algebraico explícito que muestra la estructura matemática subyacente de la lógica. La metodología iniciada por Boole fue continuada en el siglo XIX por el trabajo de A. De Morgan, W. S. Jevons, C. S. Peirce y E. Schröder. Un resumen de estos trabajos se puede encontrar en [8]. La relación entre lógica y álgebra desde la perspectiva contemporánea se remonta a las ideas de Lindenbaum y Tarski, en las cuales las fórmulas de una lógica dada son interpretadas por álgebras con operaciones asociadas con los conectivos logicos. En [6], Blok y Pigozzi propusieron una generalización de las técnicas de algebrización originales para abarcar un rango más amplio de lógica. Después de esto, se sugirieron varias generalizaciones del método de Blok y Pigozzi en la literatura [11, 12, 13].

La lógica C1 de da Costa fue introducida con la intención recuperar el razonamiento clásico. Mortensen en su trabajo Every quotient algebra for c1 is trivial (1980), probó que no se puede algebrizar con el método de Lindenbaum-Tarski. Lewin et al., en su trabajo C1 is not algebraizable (1991), probaron que no es algebrizable Blok-Pigozzi. Es sabido que la lógica CG'3 es una extensión de C1. En este trabajo mostramos que CG'3 es algebrizable en el sentido Blok-Pigozzi, lo que permitiría decir que CG'3 es una lógica con los requerimientos que da Costa pedía para C1 pero que es algebrizable.

Este artículo está organizado de la siguiente manera: En la sección 2, presentamos algunas definiciones conocidas y resultados acorde a la ambientación del presente manuscrito con el fin de facilitar la lectura del texto. Estudiamos la lógica CG'3 que se define en términos de cuatro conectivos $\wedge$, $\vee$, $\rightarrow$ y $\neg$ donde la implicación es una implicación deductiva. En la sección 3, establecemos que CG'3 es una lógica algebrizable con el método de Blok-Pigozzi. Finalmente, en la última sección, presentamos una lista de problemas abiertos para ser estudiados en el futuro.

2. Antecedentes

Primero presentamos la sintaxis de las fórmulas lógicas consideradas en este artículo. Seguimos la notación estándar y las definiciones básicas como W. Carnielli y M. Coniglio en [9].

Definición 1 (Conjunto proposicional). Un conjunto proposicional es un conjunto Θ de símbolos llamados conectivos, junto con la información relativa a la aridad de cada conectivo.

Los siguientes símbolos se usarán como conectivos lógicos: $\wedge$ (conjunción, binario); $\vee$ (disyunción, binario); $\rightarrow$ (implicación, binario); $\neg$ (negación débil, unario); (operador de inconsistencia, unario); $\sim$ (negación fuerte, unario); $\perp$ (fórmula bottom, 0-ary).

Definición 2 (Lenguaje proposicional). Sea Var ={p1, p2,…} un conjunto numerable de variables proposicionales, y sea Θ cualquier conjunto proposicional. El lenguaje proposicional generado por Θ de Var será denotado por LΘ.

Definición 3 (Lógica estándar). Una lógica L definida sobre un lenguaje L que tiene una relación de consecuencia ⊢, es Tarskian si cumple las siguientes propiedades, para cada $\Gamma \cup \Delta \cup \{\alpha\} \subseteq L$:

i. si $\alpha \in \Gamma$ entonces $\Gamma \vdash \alpha$;
ii. si $\Gamma \vdash \alpha$ y $\Gamma \subseteq \Delta$ entonces $\Delta \vdash \alpha$;
iii. si $\Delta \vdash \alpha$ y $\Gamma \vdash \beta$ para cada $\beta \in \Delta$ entonces $\Gamma \vdash \alpha$.

Una lógica que satisface el inciso (ii) es llamada monótona Una lógica L se dice que es finitaria si cumple lo siguiente:

iv. si $\Gamma \vdash \alpha$ entonces existe un subconjunto finito Γ0 de Γ tal que $\Gamma 0 \vdash \alpha$. Una lógica L definida sobre un lenguaje proposicional L generado por un conjunto de variables proposicionales se llama estructural, si cumple con la siguiente propiedad:
v. si $\Gamma \vdash \alpha$ entonces $\sigma\,[\Gamma] \vdash \sigma[\alpha]$, para cada sustitución σ de fórmulas para variables.

Una lógica proposicional es estándar, si es tarskiana, finitaria y estructural. De ahora en adelante, una lógica L será representada por un par $L = \langle L, \vdash \rangle$, donde L y ⊢ denota el lenguaje y la relación de consecuencia de L, respectivamente. L es generada por un lenguaje proposicional Θ de V ar, esto es, L = LΘ entonces escribiremos $L = \langle \Theta, \vdash \rangle$.

Sea $L = \langle L, \vdash \rangle$ una lógica. Sea α una fórmula en L y sean X1… Xn una secuencia finita (para $n \geq 1$) tal que para cada Xi es o bien un conjunto de fórmulas en L o una fórmula en L. Entonces, como es usual, X1,… , Xn ⊢ α representará $X'1 \cup \cdots \cup X'n \vdash \alpha$, donde, para cada i, X'i es Xi, si Xi es un conjunto de fórmulas, o X'i es {Xi}, si Xi es una fórmula.

Definición 4 (Lógica paraconsistente). Una lógica Tarskiana L es paraconsistente si tiene una negación (primitiva o definida) ¬ tal que $\alpha, \neg\alpha \nvdash L \;\; \beta$ para algunas fórmulas α y β en el lenguaje de L.

Observación 1. Si L tiene una implicación deductiva →, en el sentido que se satisface el meta-teorema de la Deducción MTD, entonces L es paraconsistente si y sólo si la fórmula $\varphi \rightarrow (\neg\varphi \rightarrow \psi)$ no es válida, es decir, la ley de explosión no es válida L con respecto a la negación ¬. Esto es, la negación ¬ no es explosiva.

Ahora, presentamos la noción de lógica de formal inconsistencia.

Definición 5 (Lógica de Formal Inconsistencia). Sea $L = \langle \Theta, \vdash \rangle$ una lógica estándar. Supongamos que Θ de L contiene una negación ¬, y sea ○(p) un conjunto de fórmulas no vacío que depende exactamente de la variable proposicional p. En consecuencia, L es una lógica de Formal Inconsistencia, (LFI), con respecto a ¬ y ○(p) si lo siguiente se tiene:

i. $\varphi, \neg\varphi \nvdash \psi$ para algunas φ y ψ;
ii. hay dos fórmulas α y β tal que;
    a. $\circ(\alpha), \alpha \nvdash \beta$;
    b. $\circ(\alpha), \neg\alpha \nvdash \beta$;
iii. $\circ(\varphi), \varphi, \neg\varphi \vdash \psi$ para cada φ y ψ. Observación 2.

- Cuando ○ es un singular, sus elementos se denotan por ∘p, donde ∘ es el operador de consistencia.
- Una lógica que satisface la propiedad (iii) se llama suavemente explosiva.

Finalmente, definimos una noción más fuerte de LFI para más referencia, vea [9].

Definición 6 (Lógica de Formal Inconsistencia Fuerte). SeaL = $\langle\Theta, \vdash\rangle$ una lógica estándar. Supongamos que Θ de L contiene una negación ¬, y sea

○(p) un conjunto de fórmulas no vacío que depende exactamente de la variable proposicional p. Entonces L una LFI fuerte con respecto a ¬ y ○(p) si lo siguiente se tiene:

i. hay dos fórmulas α y β tal que:
   a. $\alpha, \neg\alpha \nvdash \beta$;
   b. $\circ(\alpha), \alpha \nvdash \beta$;
   c. $\circ(\alpha), \neg\alpha \nvdash \beta$; y

ii. $\circ(\varphi), \varphi, \neg\varphi \vdash \psi$ para cada φ y ψ. Observación 3.

– Cualquier LFI fuerte es una LFI.

– Si L es una lógica proposicional entonces L es una LFI fuerte siempre que se cumpla lo siguiente:

i. hay dos variables proposicionales p y q tal que:
   a. $p, \neg p \nvdash q$;
   b. $\circ(p), p \nvdash q$;
   c. $\circ(p), \neg p \nvdash q$; y

ii. $\circ(\varphi), \varphi, \neg\varphi \vdash \psi$ para cada φ y ψ.

Definición 7 (Algebrización Blok-Pigozzi). Sea Θ un conjunto proposicional, y sea L una lógica proposicional estándar definida sobre el lenguaje LΘ, con una relación de consecuencia ⊢L. Entonces L es algebrizable en el sentido de Blok y Pigozzi si existe un conjunto no vacío $\Delta(p1, p2) \subseteq L\Theta$ de fórmulas dependiendo de las variables p1 y p2, y un conjunto no vacío $E(p1) \subseteq L\Theta \times L\Theta$ de pares de fórmulas dependiendo de la variable p1 satisfaciendo las siguientes propiedades:

i. $\vdash_L \delta(p1, p1)$, para cada $\delta(p1, p2) \in \Delta(p1, p2)$;

ii. $\Delta(p1, p2) \vdash_L \delta(p2, p1)$, para cada $\delta(p1, p2) \in \Delta(p1, p2)$;

iii. $\Delta(p1, p2), \Delta(p2, p3) \vdash_L \delta(p1, p3)$, para cada $\delta(p1, p2) \in \Delta(p1, p2)$;

iv. $\Delta(p1, pn+1),\ldots, \Delta(pn, p2n) \vdash_L \delta(\#(p1,\ldots, pn), \#(pn+1,\ldots ,p2n))$, para cada $\delta(p1, p2) \in \Delta(p1, p2)$, cada conectivo n-ario, # de Θ y cada $n \geq 1$;

   a. (v) $p1 \vdash_L \delta(\gamma(p1), s(p1))$, para cada $\delta(p1, p2) \in \Delta(p1, p2)$ y cada $\langle\gamma(p1), \epsilon(p1)\rangle \in E(p1)$;

v. $\{\delta(\gamma(p1), \epsilon(p1)) : \delta(p1, p2) \in \Delta(p1, p2), \langle\gamma(p1), \epsilon(p1)\rangle \in E(p1)\} \vdash_L p1$.

Los conjuntos Δ(p1, p2) y E(p1) denominan formulas de sistema de equivalencia de fórmulas y sistema de ecuaciones definidas, respectivamente.

Definición 8 (Relación). Sea Θ un conjunto proposicional, y sea $\Theta \subseteq L\Theta \times L\Theta$ una relación definida sobre el álgebra de fórmulas LΘ si cumple las siguientes propiedades:

i. $\alpha\theta\alpha$ para cada $\alpha \in L\Theta$ (reflexividad).
ii. $\alpha\theta\beta$ implica que $\beta\theta\alpha$ para cada $\alpha, \beta \in L\Theta$ (simetría).
iii. $\alpha\theta\beta$ y $\beta\theta\gamma$ implica que $\alpha\theta\gamma$ para cada $\alpha, \beta\gamma \in L\Theta$ (transitividad).
iv. Dados αi y βi en LΘ (para $1 \leq i \leq n$) tal que $\alpha 1\theta\beta 1, \ldots, \alpha n\theta\beta n$, entonces $\#(\alpha 1, \ldots, \alpha n)\theta\#(\beta 1, \ldots, \beta n)$ para cada conectivo n-ario # de Θ y cada $n \geq 1$.
v. Una congruencia θ en LΘ es trivial si ya sea $\theta = \{(\alpha, \alpha) : \alpha \in L\Theta\}$ o $\theta = L\Theta \times L\Theta$.

Definición 9 (Congruencia lógica). Sea L una lógica estándar definida sobre el lenguaje LΘ.

i. Una congruencia θ en LΘ es compatible con una teoría $\Gamma \subseteq L\Theta$ si se satisface lo siguiente:

ii. $\alpha\theta\beta$ y $\Gamma \vdash L\ \alpha$ implica que $\Gamma \vdash L\ \beta$.

i. Una congruencia θ en LΘ es una congruencia lógica en L si θ es compatible con cada teoría Γ. Equivalentemente, θ es una congruencia lógica en L si, para cada α y β:

iii. $\alpha\theta\beta$ implica que $\alpha \vdash L\ \beta$ y $\beta \vdash L\ \alpha$.

La manera usual de definir la semántica multivaluada de una lógica es a través de una matriz. Introducimos la definición de la matriz determinista, también conocida como matriz lógica o simplemente como matriz. En [16], podemos encontrar una discusión exhaustiva sobre lógica multivaluada y algunos ejemplos.

Definición 10 (Matriz). Dada una lógica L en el lenguaje L, la matriz de L es una estructura $M = (D, D*, F)$, donde:

i. D es un conjunto no vacío de valores de verdad (dominio).
ii. $D*$ es un subconjunto de D (conjunto de valores designados).
iii. $F = \{fc \mid c \in C\}$ es un conjunto de funciones de verdad, con una función por cada conectivo lógico c de L.

Definición 11 (Interpretación). Dada una lógica L en el lenguaje L, una interpretación t, es una función t: Var → D que asigna variables proposicionales a elementos en el dominio.

Cualquier interpretación t se puede extender a una función a todas las fórmulas en LΣ como de costumbre, es decir, aplicando recursivamente las funciones de verdad de los conectivos lógicos en F. Si t es una interpretación en la lógica L, diremos que t es una interpretación L. Las interpretaciones nos permiten definir la noción de validez en este tipo de semántica de la siguiente manera:

Definición 12 (Fórmula válida). Dada una fórmula φ y una interpretación t en una lógica L, decimos que la fórmula φ es válida bajo t en L, si $t(\varphi) \in D*$, y lo denotamos como $t \vDash L\ \varphi$.

Tengamos en cuenta que la validez depende de la interpretación, pero si queremos hablar de "verdades l ógicas" en el sistema, entonces la validez debe ser absoluta, como se indica en la siguiente definición:

Tabla 1. Funciones de verdad para los conectivos ∨, ∧, →, y ¬ en CG'3.

$f\vee$	**0**	**1**	**2**	$f\wedge$	**0**	**1**	**2**	$f\rightarrow$	**0**	**1**	**2**		$f\neg$
0	0	1	2	**0**	0	0	0	**0**	2	2	2	**0**	2
1	1	1	2	**1**	0	1	1	**1**	0	2	2	**1**	2
2	2	2	2	**2**	0	1	2	**2**	0	1	2	**2**	0

Definición 13 (Tautología). Dada una fórmula φ en el lenguaje de una lógica L, decimos que φ es una tautología en L, si para cada posible interpretación, la fórmula φ Es válida, y la denotamos como $\vDash$L φ.

Si φes una tautología en la lógica L, decimos que φ es una L -tautología.

2.1. La lógica CG'3

La lógica CG'3 fue introducida en [17]. Los autores, la definieron como una lógica 3-valuada en donde la matriz esta dada por la estructura M = ⟨ D, D∗, F ⟩, donde D = { 0, 1, 2 }, el conjunto D∗ de valores designados { 1, 2 }, y F es el conjunto de funciones de verdad definidas en la Tabla 1.

Observación 4.

1 Note que → es una implicación deductiva: Γ, α $\vDash$CG'3 β si y sólo si Γ $\vDash$CG'3:

2 α → β.

1 Considerando el orden natural 0 ≤ 1 ≤ 2 en D, ∨ corresponde al supremo:

∧ corresponde al ínfimo y → es el residuo de ∧:

z ∧ x ≤ y si y sólo si z ≤ x → y, para cada x, y, z ∈ D.

2 La lógica CG'3 fue axiomatizada en [19], aplicando el método de Lindenbaum-Łos. Además los autores definen tres conectivos: $\sim\varphi{:=}\varphi\rightarrow\bot$ (Negación fuerte), $\bullet\varphi{:=}\sim\sim\varphi\wedge\neg\varphi$ (Operador de insoncistencia) $\circ\varphi{:=}\neg\bullet\varphi$ (Operador de consistencia).

3. La algebrización de CG'3

En [6], Blok y Pigozzi dieron un concepto matemático de lógica algebrizable. La idea principal de esta definición es la siguiente: Una lógica es algebrizable si existe una clase de álgebras asociadas a la lógica de la misma manera que la clase de álgebras booleanas está relacionada con la lógica proposicional clásica.

Proposición 14 La lógica CG'3 es una LFI fuerte con un operador de consistencia ∘ como se definió anteriormente.

Demostración. Suponga que p y q son dos variables proposicionales diferentes. Al considerar la interpretación v1 tal que v1(p) = 1, v1(¬p) = 2, y v1(q) = 0, resulta que p, ¬p $\vDash$CG'3 q y la cláusula (i.a) de la Observación 3 se verifica. Considere la interpretación v2 tal que v2(p) = 2, v2(∘p) = 2, y v2(q) = 0, se tiene que p, ∘p ⊨CG'3 q y la cláusula (i.b) de la Observación 3 se cumple.

Ahora, considerando la interpretación v3 tal que v3(¬p) = 1, v3(∘p) = 2, y v3(q) = 0, resulta que p, p ⊨CG'3 q y la cláusula (i.c) de la Observación 3 se verifica. Finalmente, no hay una interpretación que haga a φ, $\neg\varphi$, y $\circ\varphi$ simultáneamente verdaderas. Por lo tanto, el inciso (ii) de la Observación 3 se satisface. Por lo tanto, CG'3 es una LFI fuerte con respecto a ¬ y ∘.

Proposición 15 Sea h una interpretación para CG'3. Entonces:

i. h (p1 → p2) ∈ D∗ si y sólo si h(p1) = 0 o h(p2) ∈ D∗;

ii. h (p1 ∧ p2) ∈ D∗ si y sólo si h(p1) ∈ D∗ y h(p2) ∈ D∗;

iii. h (p1 ∨ p2) ∈ D∗ si y sólo si h(p1) ∈ D∗ o h(p2) ∈ D∗;

iv. h (p1 ↔ p2) ∈ D∗ si y sólo si o bien h(p1) ∈ D∗ y h(p2) ∈ D∗, o h(p1) = h(p2) = 0.

Demostración. Inmediato de las tablas de verdad.

Definición 16 Sea δ (p1, p2) la siguiente fórmula de LΣ:
δ (p1, p2) = (p1 ↔ p2) ∧ (∘p1 ↔ ∘p2)

Proposición 17 Sean α y β fórmulas en LΣ. Entonces para cada interpretación h para CG'3, se tiene que:

1. h (δ (α, β)) ∈ D∗ si y sólo si tenemos:
   a. o bien h(α) ∈ D∗ y h(β) ∈ D∗, o h(α) = h(β) = 0; y
   b. h (∘α) = h(∘β)
2. Así, h (δ (α, β)) ∈ D∗
3. H (•α → α) ∈ D∗.
4. si y sólo si h(α) = h(β).
5. h(∘α) = h ((•α → α)).
6. h (δ (α, •α → α)) ∈ D∗ si y sólo si h(α) ∈ D∗.

Demostración

1. Parte "Sólo si". H (δ(α, β)) ∈ D∗ si y sólo si de acuerdo con la Proposición 15(ii), h(α ↔ β) ∈ D∗ y h(α ↔ β) ∈ D∗. Por la Proposición 15(iv), h(α ↔β) ∈D∗ es equivalente a o bien h(α)∈D∗ y h(β)∈D∗, o h(α) = h(β) = 0, mientras h(∘α ↔∘ β) D∗ si y sólo si o bien h(∘ α) ∈ D∗ y h(∘ β) ∈ D∗, o h(∘α) = h(∘β) = 0. De la definición de ∘, h(∘α ↔∘ β) ∈ D∗ es equivalente a h(∘α) = h(∘β). Ahora, supongamos que h(δ(α, β))∈D∗. Si h(α) = 2 y h(β) = 1, entonces h(∘α) = 2 y h(∘β) = 0, violando que h(∘α) = h(∘β).
2. Análogamente, es imposible tener que h(α) = 1 y h(β) = 2. Esto muestra que h(α) = h(β). Parte "Si" es obvia, a la luz de las cláusulas.
3. La prueba es directa.

4. Si $h(\circ\alpha) = 2$ entonces $h(\alpha) \in \{0, 2\}$ y $h(\bullet\alpha) = 0$ y así $h(\bullet\alpha \rightarrow \alpha) = 2$; por lo tanto, $h(\circ(\bullet\alpha \rightarrow \alpha)) = 2$. Si $h(\circ\alpha) = 0$ entonces $h(\alpha) = 1$ y así $h(\bullet\alpha) = 2$; por lo tanto, $h(\bullet\alpha \rightarrow \alpha) = 1$; así, $h(\circ(\bullet\alpha \rightarrow \alpha)) = 0$.
5. Parte "Sólo si". Por el inciso 1, $h(\delta(\alpha, \bullet\alpha \rightarrow \alpha)) \in D*$ implica que $h(\alpha) =$

$h(\bullet\alpha \rightarrow \alpha)$. Entonces, por el inciso 2, $h(\alpha) \in D*$. Parte "Si". Supongamos que $h(\alpha) \in D$. Por el inciso 2, $h(\bullet\alpha \rightarrow \alpha) \in D*$ y, por el inciso 3, $h(\circ\alpha) = h(\circ(\bullet\alpha \rightarrow \alpha))$. Finalmente, aplicando inciso 1, $h(\delta(\alpha, \bullet\alpha \rightarrow \alpha)) \in D*$.

Teorema 18 La lógica CG'3 es algebrizable en el sentido de Blok y Pigozzi con un sistema de fórmulas equivalentes dado por $\Delta(p1, p2) = \{\delta(p1, p2)\}$ y un sistema de ecuaciones definidas dado por $E(p1) = \{p1, \bullet p1 \rightarrow p1\}$.

Demostración. Es fácil demostrar que el sistema Δ (p1, p2) cumple las condiciones (i)-(iv) según el inciso 1 de la Proposición 17. Por el inciso 4 de la misma proposición, las condiciones (v)-(vi) se siguen fácilmente.

4. Conclusiones y trabajo futuro

CG'3 se define por su semántica multivaluada, la matriz de la lógica CG'3 está dada por M= (D, D∗, F); donde el dominio es D= {0, 1, 2} y el conjunto de valores designados es D∗ = {1, 2}. Esta lógica es paraconsistente y puede verse como una extensión de la lógica G'3 también introducida por Osorio en 2008, [18]. En este artículo, ampliamos los estudios sobre esta lógica al presentar algunos resultados relacionados con lógica algebraica. El resultado principal del trabajo es la algebrización de CG'3 usando la técnica de Blok y Pigozzi. Este resultado nos abre las puertas para el desarrollo, implementación y aplicación de sistemas de inferencia paraconsistente robustos [20]. Entre las aplicaciones, es de nuestra particular atención la verificación de sistemas [15].

Otra pregunta de investigación de nuestro interés es con respecto a la relación de CG'3 y la familia de lógicas paraconsistentes anotadas, definidas por Subrahmanian en [22]. Hoy en d ía se conocen muchas aplicaciones de lógicas paraconsistentes en diversos campos de las ciencias computacionales, como: circuitos eléctricos, razonamiento no monótono, sistemas de control, automatización y robótica, por mencionar algunos [1].

References

1. Abe, J.M., Nakamatsu, K., da Silva Filho, J.I.: Three decades of paraconsistent annotated logics: a review paper on some applications. Procedia Computer Science, 159, pp. 1175–1181 (2019)
2. Abe, J.M.: Fundamentos da lógica anotada. Foundations of Annotated Logics. Ph. D. Thesis, University of São Paulo (1992)
3. Abe, J.M.: Paraconsistent intelligent-based systems: New trends in the applications of paraconsistency. 94, Springer (2015)
4. Abe, J.M., Akama, S., Nakamatsu, K.: Introduction to annotated logics: foundations for paracomplete and paraconsistent reasoning. 88, Springer (2015)

5. Blass, A.: 2003 annual meeting of the association for symbolic logic. Bulletin of Symbolic Logic 10(1), 120–145 (2004). https://doi.org/10.1017/S1079898600004248
6. Blok, W.J., Pigozzi, D.: Algebraizable logics, vol. 77. American Mathematical Soc. (1989)
7. Boole, R.: The mathematical analysis of logic, 1847 (1847)
8. Burris, S., Legris, J.: The algebra of logic tradition. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, fall 2018 edn. (2018)
9. Carnielli, W.A., Coniglio, M.E.: Paraconsistent logic: Consistency, contradiction and negation. Springer (2016)
10. Curry, H.B.: Functionality in combinatory logic. In: Proceedings of the National Academy of Sciences 20(11), 584–590 (1934)
11. Czelakowski, J.: Protoalgebraic logics. In: Protoalgebraic Logics, pp. 69–122. Springer (2001)
12. Font, J.M., Jansana, R.: A general algebraic semantics for sentential logics. Vol. 7. Cambridge University Press (2017)
13. Font, J.M., Jansana, R., Pigozzi, D.: A survey of abstract algebraic logic. Studia Logica 74(1-2), 13–97 (2003)
14. Gottwald, S.: Many-valued logic. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, winter 2017 edn. (2017)
15. Limón, Y., Bárcenas, E., Benítez-Guerrero, E., Molero, G.: On the consistency of context-aware systems. J. Intell. Fuzzy Syst. 34(5), 3373–3383 (2018)
16. Malinowski, G.: Many-Valued Logics. Oxford University Press (1993)
17. Osorio, M., Carballido, J.L., Zepeda, C., et al.: Revisiting Z. Notre Dame Journal of 4 Formal Logic 55(1), 129–155 (2014)
18. Osorio Galindo, M., Carballido Carranza, J.L.: Brief study of g'3 logic. Journal of Applied Non-Classical Logics 18(4), 475–499 (2008)
19. Pérez-Gaspar, M., Barcenas, E.: Completeness for the paraconsistent logic CG (2019)
20. Pérez-Gaspar, M., Hernandez-Tello, A., Arrazola Ramírez, J., Osorio Galindo, M.: An axiomatic approach to CG'3 logic. Logic Journal of the IGPL (2020)
21. Priest, G., Tanaka, K., Weber, Z.: Paraconsistent logic. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, summer 2018 edn. (2018)
22. Subrahmanian, V.: On the semantics of quantitative logic programs. In: SLP, pp. 173–182 (1987)
23. Thomason, R.: Logic and artificial intelligence. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy, Metaphysics Research Lab, Stanford University, summer 2020 edn. (2020)
24. Wieckowski, B., Metcalfe, G., Olivetti, N., Gabbay, D.: Proof theory for fuzzy logics. Applied logic series, vol. 36, Bulletin of Symbolic Logic 16(3), 415–419 (2010)
25. Zamansky, A.: On recent applications of paraconsistent logic: an exploratory literature review. Journal of Applied Non-Classical Logics 29(4), 382–391 (2019)