Sobre la relación entre la Semántica $\mathcal{GS}$ y el Razonamiento Rebatible

Laura A. Cecchi

Depto. de Cs. de la Computación - Fa.E.A.
Universidad Nacional del Comahue
Buenos Aires 1400 - 8300 Neuquén - Argentina
Tel/Fax (54) (299) 4490312/313
e-mail:lcecchi@uncoma.edu.ar

Guillermo R. Simari

Depto. de Cs. e Ing. de la Computación
Universidad Nacional del Sur
Av. Alem 1253 - 8000 Bahía Blanca - Argentina
Tel/Fax (54) (291) 4595135/5136
e-mail:grs@cs.uns.edu.ar

Palabras Claves: Sistemas Argumentativos - Razonamiento Rebatible - Programación en Lógica - Semántica basada en Juegos

Resumen

En los últimos años, las teorías de juegos y de diálogos para la argumentación han recibido gran interés en varios campos de la Inteligencia Artificial, por sus aplicaciones en los sistemas multiagentes y en los sistemas legales, entre otras.

Una herramienta de representación de conocimiento cuyo sistema de razonamiento está basado en la argumentación es la Programación en Lógica Rebatible Básica. Si bien, la escuela de la argumentación rebatible ha desestimado tradicionalmente los enfoques semánticos declarativos, en los últimos años, la semántica operacional ha sido estudiada desde un punto de vista declarativo, con el objeto de determinar el significado preciso de un programa lógico sin recurrir al control.

El propósito de este trabajo es presentar la semántica declarativa desarrollada $\mathcal{GS}$ para el sistema de razonamiento no monótono de la Programación en Lógica Rebatible, junto con los resultados de sensatez y completitud. Dicha semántica vincula dos teorías con fundamentos matemáticos rigurosos: la teoría de juegos y la teoría de modelos. $\mathcal{GS}$ refleja las relaciones sintácticas entre los argumentos de un modo natural, por la similitud existente entre un árbol dialéctico y un juego de dos participantes.

1. Introducción

En los últimos años, las teorías de juegos y de diálogos para la argumentación han recibido gran interés en varios campos de la Inteligencia Artificial, por sus aplicaciones en los sistemas multiagentes[ABCMB04, PSJ98] y en los sistemas legales [OS99], entre otras.

Una de las principales razones de esta tendencia es que el razonamiento no monotónico o rebatible basado en la argumentación involucra aspectos dinámicos que no pueden ser modelados en forma natural a través de la lógica.

La Programación en Lógica Rebatible Básica (DeLP)[CS00, GS99, Gar00] es una herramienta de representación de conocimiento, cuyo sistema de razonamiento está basado en la argumentación. El cálculo de su semántica operacional involucra la construcción de árboles dialécticos los que tienen una gran similitud con un juego de dos participantes.

Los juegos permiten modelar al razonamiento rebatible como una disputa entre dos participantes: el proponente y el oponente. El proponente comienza con un argumento para un literal y luego cada jugador debe atacar al argumento jugado anteriormente con un contraargumento que derrote al anterior. El literal es consecuencia del sistema rebatible si el proponente tiene una estrategia ganadora, i.e., si el proponente deja sin respuesta al oponente.

Si bien, la escuela de la argumentación rebatible ha desestimado tradicionalmente los enfoques semánticos declarativos [Vre93], en los últimos años, la semántica operacional ha sido estudiada desde un punto de vista declarativo[Dun95, KT99, Cn01], con el objeto de determinar el significado preciso de un programa lógico sin recurrir al control.

El propósito de este trabajo es presentar la semántica declarativa desarrollada para el sistema de razonamiento no monótono de la DeLPjunto con los resultados de completitud. En la sección 2, se describen brevemente la sintaxis de la DeLP y su semántica operacional. En la siguiente sección, se presenta la semántica declarativa desarrollada, sustentada en una propuesta de semánticas basadas en juegos introducida por Abramsky en [Abr97]. Se formaliza la noción de modelo basado en juego, G-modelo, justificado en los conceptos de juego, estrategias y criterios de triunfo. Estos cimientos permiten por último, introducir la definición más importante: *semántica* $\mathcal{GS}$. Dicha semántica vincula dos teorías rigurosas. Por un lado, la teoría de juegos y por el otro, la teoría de modelos, ya que su definición utiliza el enfoque de modelo minimal. En la sección 4, presentamos, el principal resultado de este trabajo. Introducimos una transformación de secuencias de un juego en un árbol dialéctico. A partir de esta transformación probamos que la semántica declarativa basada en juegos desarrollada es sensata y completa con respecto a semántica operacional de la DeLP. Finalmente, en la sección 5, se presentan las conclusiones y los trabajos futuros.

2. Programación en Lógica Rebatible Básica

En la mayoría de los dominios del conocimiento humano, las creencias no son categóricas, por lo que las conclusiones que se obtienen a partir de ellas son potencialmente contradictorias. Precisamente, por esta razón, es que se requieren de técnicas de representación de conocimiento que resuelvan este problema. La DeLP es un sistema de representación de conocimiento y razonamiento que permite manipular tanto información certera como tentativa a través de dos clases diferentes de reglas: las reglas estrictas y las reglas rebatibles.

El lenguaje de la DeLP está formado por todos los posibles literales fijos(ground) que se puedan obtener a partir de la signatura del programa y lo notamos *Lit*. En otras palabras, L

pertenece a Lit si L es un átomo fijo A o un átomo fijo negado $\sim A$, siendo $\sim$ la representación de la negación fuerte.

Definición 2.1. [GS04, CS00]**(Hechos. Regla Estricta Básica. Regla Rebatible Básica)**
Un *hecho* es un literal. Una *Regla Estricta Básica* es un par ordenado $L \leftarrow L_1, \ldots, L_n$ donde $L \in Lit$ y $\{L_1, \ldots, L_n\}$ es un subconjunto de Lit, con $n > 0$. Una *Regla Rebatible Básica* es un par ordenado $L \prec L_1, \ldots L_m$ donde $L \in Lit$ y $\{L_1, \ldots, L_m\}$ es un subconjunto de Lit, con $m > 0$. ■

Las reglas rebatibles son reglas que pueden ser derrotadas en presencia de evidencia contraria. Las reglas estrictas representan conocimiento seguro y libre de excepciones.

Definición 2.2. [GS04]**(Programa Lógico Rebatible Básico)**
Un *programa lógico rebatible básico* (de ahora en más *de.l.p.*) es un conjunto, posiblemente infinito, de hechos, reglas estrictas básicas y rebatibles básicas. Si $\mathcal{P}$ es un *de.l.p.*, distinguiremos el subconjunto Π de hechos y reglas estrictas básicas en $\mathcal{P}$, y el subconjunto Δ de reglas rebatibles básicas en $\mathcal{P}$. Cuando se requiera denotaremos $\mathcal{P}$ como $\langle \Pi, \Delta \rangle$. ■

La semántica operacional de la DeLP se basa en los *sistemas de argumentación*, que constituyen una formalización del razonamiento no monótono. Con el objeto de explicar una creencia, un agente realiza un análisis dialéctico teniendo en cuenta las pruebas tentativas que pueda construir para tal creencia. Las explicaciones que soportan a la creencia se denominan *argumentos* y son el cimiento del razonamiento rebatible.

Definición 2.3. [Gar00]**(Estructura de Argumento: enfoque procedural)**
Sea $\mathcal{P} = \langle \Pi, \Delta \rangle$ un programa lógico rebatible. Una *estructura de argumento* para un literal fijo h en el contexto Π, que denotaremos $\langle \mathcal{A}, h \rangle_\Pi$ o simplemente $\langle \mathcal{A}, h \rangle$, si $\mathcal{A}$ es un subconjunto de Δ que cumple las siguientes condiciones:

1. h se deriva rebatiblemente a partir de $\Pi \cup \mathcal{A}$,
2. $\Pi \cup \mathcal{A}$ no es contradictorio, y
3. $\mathcal{A}$ es minimal con respecto a la inclusión de conjuntos, i.e., no existe $\mathcal{A}' \subset \mathcal{A}$ tal que $\mathcal{A}'$ satisface las dos primeras condiciones.

Una estructura de argumento $\langle \mathcal{B}, h' \rangle$ es un *subargumento* de $\langle \mathcal{A}, h \rangle$ si $\mathcal{B} \subseteq \mathcal{A}$.
Dos estructuras de argumento $\langle \mathcal{B}, h' \rangle$ y $\langle \mathcal{A}, h \rangle$ están en *desacuerdo* cuando $\Pi \cup \{h, h'\}$ sea contradictorio. $\langle \mathcal{B}', q \rangle$ se denomina *contraargumento* de $\langle \mathcal{A}, h \rangle$ si existe un subargumento $\langle \mathcal{A}', h' \rangle$ de $\langle \mathcal{A}, h \rangle$ y $\langle \mathcal{A}', h' \rangle$ y $\langle \mathcal{B}', q \rangle$ están en desacuerdo. ■

Nos referiremos a la estructura de argumento $\langle \mathcal{A}, h \rangle$ como *argumento*, siempre que no sea ambiguo.

De todos los contraargumentos que puedan existir para un argumento, nosotros estaremos interesados solamente en aquellos que lo derroten. Con el objeto de poder decidir entre contraargumentos, se define una relación de preferencia entre argumentos. Esta relación puede ser especificada de diferentes maneras, como por ejemplo, prioridad entre reglas [PV00] y especificidad[Gar00].

El análisis dialéctico, que puede ser visto gráficamente como un árbol, se construye comenzando con un argumento para la consulta, considerando, luego, todos los contraargumentos que lo derroten. Ya que los contraargumentos son argumentos, se deberá considerar los contraargumentos de los contraargumentos y así sucesivamente.

Un literal está garantizado si en el árbol que representa su análisis dialéctico resulta que cada contraargumento que podría derrotar al argumento en la raíz es a su vez derrotado, teniendo en cuenta que las hojas del árbol son argumentos no derrotados. En caso de que un literal h esté garantizado diremos Bh, i.e., el agente cree en h. Las posibles respuestas a una consulta h son[GS04]:

SI:	si Bh (cree en h)
NO:	si $B\overline{h}$ (cree en $\overline{h}$)
INDECISO:	si $\neg Bh$ y $\neg B\overline{h}$ (no cree en h ni en $\overline{h}$)
DESCONOCIDO:	si h no pertenece a la signatura del programa

La formalización de estas ideas pueden encontrarse en detalle en [Gar00, GS04]. En la siguiente sección se analiza la semántica operacional desde un punto de vista declarativo.

3. Semántica declarativa basada en juegos

El análisis dialéctico, en el que se basa la argumentación, se asemeja a un juego en el que de manera alternada dos jugadores compiten. Un jugador, el *Proponente*, propone un argumento y luego trata de defenderlo contra cualquier ataque que provenga del otro participante, el *Oponente*. En este juego los jugadores hacen sucesivos *movimientos* de acuerdo a un conjunto de *reglas*, introduciendo argumentos. Al definir el juego, consideraremos, como es natural, aquellas posiciones que han sido defendidas en forma exitosa y que determinarán el criterio vencedor.

3.1. Jugadas Legales

Las instrucciones para jugar un juego comienzan con la definición de la arena la que se desarrollará el juego y con los instrumentos que necesitaremos para participar. En nuestro caso, nuestros elementos de juego serán los *argumentos*. El juego comienza con un argumento y, en forma sucesiva y alternada, se jugarán diferentes argumentos hasta llegar a una situación en la que el juego termine y se pueda determinar qué jugador ha vencido.

Definición 3.1. **(**$Arg(h,\mathcal{P})$**)**
Sean h un literal y $\mathcal{P} = \langle \Pi, \Delta \rangle$ un *de.l.p.*. Definimos a $Arg(h,\mathcal{P})$ como el conjunto de todos los argumentos y contraargumentos que pueden participar en un análisis dialéctico que comience con alguna estructura de argumento para h. ■

Observación 3.1 *Si $\langle \varnothing, h \rangle \in Arg(h,\mathcal{P})$, luego $Arg(h,\mathcal{P})$ es unitario. $\langle \varnothing, h \rangle$ no tiene contraargumentos[GS04], y tampoco pueden existir otros argumentos a favor de h ya que violarían la condición de minimalidad. Luego $\langle \varnothing, h \rangle$ no tiene puntos de ataque y por lo tanto, es el único argumento que forma parte de $Arg(h,\mathcal{P})$.*

Observación 3.2 *Si $Arg(h,\mathcal{P}) \neq \varnothing$ y $Arg(\overline{h},\mathcal{P}) \neq \varnothing$, luego $Arg(h,\mathcal{P}) = Arg(\overline{h},\mathcal{P})$. Si ninguno de los conjuntos es vacío, luego los argumentos de $Arg(h,\mathcal{P})$ son contraargumentos de $Arg(\overline{h},\mathcal{P})$ y viceversa. Así estos argumentos deberán estar en ambos conjuntos y todos los argumentos que sean puntos de ataque a estos últimos, también estarán en $Arg(h,\mathcal{P})$ y en $Arg(\overline{h},\mathcal{P})$.*

A partir de $Arg(h, \mathcal{P})$ especificaremos cuáles pueden ser posibles movimientos en un juego.

Definición 3.2. [CS04]**(Movimientos del Juego)**
Sea $\langle \mathcal{A}, h \rangle$ una estructura de argumento bajo un *de.l.p.* $\mathcal{P}$. El conjunto de todos los movimientos permitidos en un juego cuya primer jugada es $\langle \mathcal{A}, h \rangle$ y que notaremos $M_{G(\langle \mathcal{A}, h \rangle, \mathcal{P})}$ es un subconjunto de $Arg(h, \mathcal{P})$. Notaremos $M^*_{G(\langle \mathcal{A}, h \rangle, \mathcal{P})}$ al conjunto de todas las secuencias que se puedan obtener a partir de $M_{G(\langle \mathcal{A}, h \rangle, \mathcal{P})}$, $|s|$ a la longitud de la secuencia s, $s^{\mathcal{A}}$ a la proyección del primer elemento de s, $\mathcal{A}$, y $s^h = h$ a la proyección del segundo elemento de s. ■

Serán las reglas que gobiernan el juego las encargadas de determinar cuáles movimientos están permitidos. No siempre las discusiones entre dos agentes son correctas. En el proceso dialéctico es posible encontrar fallas que invalidan la discusión. Por lo tanto, es necesario detectar falacias en el proceso dialéctico como por ejemplo, pares de argumentos que se contradicen y derrotan entre ellos y participantes que se contradicen a sí mismos, entre otras[MG99, Gar00].

Determinar si la dialéctica es correcta involucra el análisis de los criterios de preferencia entre los contraargumentos. En [CS02a], se estudiaron la especificidad generalizada [SGCnS02] y las relaciones basadas en prioridades entre reglas[Vre92, AMB00], caracterizándolas a través de un conjunto de propiedades. Dadas dos estructuras de argumento $\langle \mathcal{A}_1, h_1 \rangle$ y $\langle \mathcal{A}_2, h_2 \rangle$ y una relación binaria entre argumentos $\mathcal{R}$, pueden presentarse las siguientes situaciones:

(a) $\langle \mathcal{A}_1, h_1 \rangle \mathcal{R} \langle \mathcal{A}_2, h_2 \rangle$ y $\langle \mathcal{A}_2, h_2 \rangle \not\mathcal{R} \langle \mathcal{A}_1, h_1 \rangle$, en cuyo caso diremos que$\langle \mathcal{A}_1, h_1 \rangle$ es estrictamente mejor que $\langle \mathcal{A}_2, h_2 \rangle$, y lo notaremos $\langle \mathcal{A}_1, h_1 \rangle \succ \langle \mathcal{A}_2, h_2 \rangle$

(b) $\langle \mathcal{A}_2, h_2 \rangle \mathcal{R} \langle \mathcal{A}_1, h_1 \rangle$ y $\langle \mathcal{A}_1, h_1 \rangle \not\mathcal{R} \langle \mathcal{A}_2, h_2 \rangle$, en cuyo caso diremos que$\langle \mathcal{A}_2, h_2 \rangle$ es estrictamente mejor que $\langle \mathcal{A}_1, h_1 \rangle$, y lo notaremos $\langle \mathcal{A}_2, h_2 \rangle \succ \langle \mathcal{A}_1, h_1 \rangle$

(c) $\langle \mathcal{A}_1, h_1 \rangle$ y $\langle \mathcal{A}_2, h_2 \rangle$ son incomparables bajo $\mathcal{R}$,i.e., $\langle \mathcal{A}_1, h_1 \rangle \not\mathcal{R} \langle \mathcal{A}_2, h_2 \rangle$ y $\langle \mathcal{A}_2, h_2 \rangle \not\mathcal{R} \langle \mathcal{A}_1, h_1 \rangle$, que lo notaremos $\langle \mathcal{A}_1, h_1 \rangle \approx \langle \mathcal{A}_2, h_2 \rangle$, o bien

(d) $\langle \mathcal{A}_1, h_1 \rangle \mathcal{R} \langle \mathcal{A}_2, h_2 \rangle$ y $\langle \mathcal{A}_2, h_2 \rangle \mathcal{R} \langle \mathcal{A}_1, h_1 \rangle$, que lo notaremos $\langle \mathcal{A}_1, h_1 \rangle \approx \langle \mathcal{A}_2, h_2 \rangle$.

Definición 3.3. [CS04]**(Derrota. Derrotador Propio. Derrotador de Bloqueo)**
Sean $\mathcal{P} = (\Pi, \Delta)$ un *de.l.p.*, h un literal, $\langle \mathcal{A}, h \rangle$ una estructura de argumento para h bajo $\mathcal{P}$ y s una secuencia de $M^*_{G(\langle \mathcal{A}, h \rangle, \mathcal{P})}$. s_j derrota a s_i, $1 \leq j, i \leq |s|$, si existe s' tal que

1. $s'^{\mathcal{A}} \subseteq {s_i}^{\mathcal{A}}$ y $\{s'^h, {s_j}^h\} \cup \Pi$ es inconsistente, y
2. Se cumple que
   a) $s_j \succ s'$, en cuyo caso diremos que s_j es un *derrotador propio* para s'.
   b) $s_j \approx s'$en cuyo caso diremos que s_j es un *derrotador de bloqueo* para s'.

■

Definición 3.4. **(Secuencia legal)**
Sea $\langle \Pi, \Delta \rangle$ un *de.l.p.*. Una secuencia finita $s \in M^*_{G(\langle \mathcal{A}, h \rangle, \mathcal{P})}$ es legal si

(a)

$$\Pi \cup \bigcup_{\substack{impar(i) \\ 1 \leq i \leq |s|}} s_i^{\mathcal{A}} \not\models \bot \text{ y } \Pi \cup \bigcup_{\substack{par(i) \\ 1 \leq i \leq |s|}} s_i^{\mathcal{A}} \not\models \bot$$

(b) no existe s_k tal que $s_k^{\mathcal{A}} \subseteq s_i^{\mathcal{A}},\ i < k \leq |s|$.

(c) Si $s_i \approx s_{i+1},\ 1 \leq i < |s|$, luego si existe s_{i+2} entonces $s_{i+2} \succ s_{i+1}$.

■

Una secuencia legal es equivalente a una línea de argumentación aceptable[GS04], ya que el punto (a) de la definición anterior no permitirá que un jugador contradiga sus propios dichos en el debate. El punto (b) no permite reintroducir en el debate subargumentos de argumentos ya argüidos. Y por último, el punto (c) contempla la posibilidad de que exista en la secuencia un derrotador de bloqueo. En esta situación, exigiremos que el siguiente argumento en la línea sea un derrotador propio, al igual que en el caso de una línea aceptable.

Lema 3.1. Sean $\mathcal{P} = \langle \Pi, \Delta \rangle$ un *de.l.p.* y s una secuencia de estructuras de argumentos bajo $\mathcal{P}$. s es legal si y sólo si s es una línea de argumentación aceptable. ■

Definición 3.5. [CS04]**(Juego)**
Sea $\mathcal{P} = (\Pi, \Delta)$ un *de.l.p.*, h un literal y $\langle \mathcal{A}, h \rangle$ una estructura de argumento para h. Un *juego* para $\langle \mathcal{A}, h \rangle$ con respecto a $\mathcal{P}$, que notaremos $G(\langle \mathcal{A}, h \rangle, \mathcal{P})$, es una estructura

$$(M_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}, \lambda_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}, P_{G(\langle \mathcal{A},h \rangle, \mathcal{P})})$$

donde

- $M_{G(\langle \mathcal{A},h \rangle, \mathcal{P})} \subseteq Arg(h, \mathcal{P})$.
- $\lambda_{G(\langle \mathcal{A},h \rangle, \mathcal{P})} :\ M_{G(\langle \mathcal{A},h \rangle, \mathcal{P})} \times I \rightarrow \{P, O\}$ donde I es un índice enumerable;
- $P_{G(\langle \mathcal{A},h \rangle, \mathcal{P})} \subseteq M^*_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}$, donde$P_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}$ es no vacío y cerrado con respecto a los prefijos de sus secuencias.

  Cada secuencia s de $P_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}$ satisface:

  1. $s = [\langle \mathcal{A}, h \rangle]s'$, con s' posiblemente vacía.
  2. Para todo $i,\ 1 \leq i \leq |s|$

  $$\lambda_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}(s_i, i) \ = \ \begin{cases} O & \text{si } i \text{ es par} \\ P & \text{si } i \text{ es impar} \end{cases}$$

  3. Si $s = s'[\langle \mathcal{A}_1, h_1 \rangle] \ \in \ P_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}$ donde s' es una secuencia posiblemente vacía, entonces para cada estructura de argumento $\langle \mathcal{A}_2, h_2 \rangle$ que derrota a $\langle \mathcal{A}_1, h_1 \rangle$, existe una secuencia $t \in P_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}$ tal que $t = s[\langle \mathcal{A}_2, h_2 \rangle] = s'[\langle \mathcal{A}_1, h_1 \rangle, \langle \mathcal{A}_2, h_2 \rangle]$, siempre y cuando t sea legal.
  4. Ninguna otra secuencia pertenece a $P_{G(\langle \mathcal{A},h \rangle, \mathcal{P})}$.

■

Dependiendo del jugador, cada movida captura un argumento de soporte o de interferencia con respecto al argumento inicial. De todas las posibles secuencias, nosotros estamos interesados solamente en secuencias de movimientos de cierta clase: aquellas que caracterizan la construcción del árbol dialéctico. Por esta razón se han impuesto tres condiciones en la definición 3.5. El punto (1) indica que el juego siempre comienza con la estructura de argumento $\langle \mathcal{A}, h \rangle$. El siguiente inciso restringe las secuencias a aquellas en las que los participantes juegan en forma alternada, comenzando el Proponente. El tercer inciso condiciona los movimientos a argumentos que se ataquen, i.e., cada movimiento es un contraargumento que derrota al argumento de la movida precedente. Asimismo, se exige que cada juego contemple todos los derrotadores de cada argumento jugado. Por último, se circunscribe el conjunto de secuencias que pertenecen a $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$ a aquellas y sólo aquellas que cumplan las tres condiciones.

Dado un literal h, el proponente podría jugar como movimiento inicial, cualquiera de los argumentos que soporta a h. De este modo, tendríamos para cada literal un juego por cada argumento que lo soporte.

Definición 3.6. [CS02b]**(Familia de Juegos)**
Sean $\mathcal{P}$ un *de.l.p.*, h un literal bajo la signatura de $\mathcal{P}$, $\langle\mathcal{A}_1, h\rangle, \ldots, \langle\mathcal{A}_n, h\rangle$ todos las estructuras de argumento de h bajo $\mathcal{P}$ y

$$G(\langle\mathcal{A}_1, h\rangle, \mathcal{P}), G(\langle\mathcal{A}_2, h\rangle, \mathcal{P}), \ldots, G(\langle\mathcal{A}_n, h\rangle, \mathcal{P})$$

los juegos correspondientes a todos los argumentos de h. Diremos que

$$\{G(\langle\mathcal{A}_1, h\rangle, \mathcal{P}), G(\langle\mathcal{A}_2, h\rangle, \mathcal{P}), \ldots, G(\langle\mathcal{A}_n, h\rangle, \mathcal{P})\}$$

es la familia de juegos de h y la notaremos $\mathcal{F}(h, \mathcal{P})$. ■

3.2. Vencedores

Una vez definido el juego con sus reglas, necesitaremos determinar quién es el vencedor. Con este objetivo en mente, daremos algunas definiciones preliminares, entre la que se encuentra la idea de estrategia para el juego.

Definición 3.7. [CS99] **(Secuencia Completa - Secuencia Preferida)**
Sea a el primer movimiento del proponente en el juego. Una secuencia s es completa si $s = [a]s_1$, con s_1 posiblemente vacía, entonces no existe ningún movimiento $b \in M_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$ tal que $[a]s_1[b] \in P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$. Una secuencia s' se dice preferida si cada movimiento del oponente tiene una respuesta del proponente. En otras palabras, una secuencia s' es preferida si $|s'|$ es impar. ■

Una secuencia completa es una línea de argumentación [Gar00], i.e. una camino desde el primer movimiento hasta un movimiento que nos permita alcanzar una hoja. Una secuencia preferida captura la idea de que dicha secuencia termina con un movimiento del proponente.

Definición 3.8. [CS04]**(Estrategia)**
Una estrategia sobre un juego G es un conjunto de secuencias S, tal que para toda secuencia $s \in S$:

- s es preferida; o
- existe otra secuencia $s' \in S$, tal que s' es preferida y si $s = [s_1, s_2, \ldots, s_n, s_{n+1}, \ldots, s_k]$ entonces $s' = [s_1, s_2, \ldots, s_n, s'_{n+1}, \ldots, s'_r]$, $s_{n+1} \neq s'_{n+1}$ y n es par, i.e., s y s' difieren a partir de una posición impar.

■

Un conjunto de secuencias completas que formen una estrategia caracterizará un árbol dialéctico cuya raíz no haya sido derrotada. Para esto, analizaremos todas las secuencias legales que conforman el juego. Si dichas secuencias forman una estrategia entonces diremos que el juego fue ganado por el proponente.

Definición 3.9. [CS04]**(Criterio de Triunfo)**
Sea $\mathcal{P}$ un *de.l.p.*, $h \in Lit$ y $G(\langle \mathcal{A}, h \rangle, \mathcal{P}) \in \mathcal{F}(h, \mathcal{P})$. Diremos que *P gana el juego* $G(\langle \mathcal{A}, h \rangle, \mathcal{P})$ o que $G(\langle \mathcal{A}, h \rangle, \mathcal{P})$ *es ganado por P*, si el conjunto de las secuencias completas de $P_{G(\langle \mathcal{A}, h \rangle, \mathcal{P})}$ es una estrategia. De otro modo, diremos que *O gana el juego* o que $G(\langle \mathcal{A}, h \rangle, \mathcal{P})$ *es ganado por O*. ■

3.3. G-Modelo

La semántica basada en juegos está definida a partir del concepto de modelo minimal.

Definición 3.10. [CS04] **(G-Interpretación)**
Sea $\mathcal{P}$ un *de.l.p.*. Una *interpretación basada en juegos para* $\mathcal{P}$, que llamaremos G-Interpretación para $\mathcal{P}$, es una tupla $\langle V, F \rangle$, tal que V y F son subconjunto de átomos bajo la signatura de $\mathcal{P}$ y $V \cap F = \varnothing$. ■

Asumimos que el conjunto de átomos INDECISOS está definido como $Lit^+ - \{V \cup F\}$, siendo Lit^+ el conjunto de todos los átomos de Lit.

Definición 3.11. [CS04] **(G-Modelo)**
Sean $\mathcal{P}$ un *plr*, h un átomo bajo la signatura de $\mathcal{P}$, $\mathcal{F}(h, \mathcal{P})$ la familia de juegos del literal h y $\mathcal{F}(\overline{h}, \mathcal{P})$ la familia de juegos del literal $\overline{h}$ para el *plr* $\mathcal{P}$. Un *modelo basado en juegos para* $\mathcal{P}$, que llamaremos G-Modelo de $\mathcal{P}$, es una G-interpretación $\langle V, F \rangle$ tal que:

- Si existe un juego $G(\langle \mathcal{A}, h \rangle, \mathcal{P})$ en la familia $\mathcal{F}(h, \mathcal{P})$ ganado por P, entonces h pertenece a V.
- Si existe un juego $G(\langle \mathcal{A}, \overline{h} \rangle, \mathcal{P})$ en la familia $\mathcal{F}_{\overline{h}}$ ganado por P, entonces h pertenece a F. ■

Nótese que la definición anterior no contempla la respuesta DESCONOCIDO, ya que sólo consideramos literales bajo la signatura del *de.l.p.*.

El conjunto ordenado $\{$G-Modelos$, \subseteq\}$ forma un reticulado inferiormente con primer elemento, cuyo ínfimo es la intersección de cada una de las componentes del G-Modelo. El primer elemento de dicho reticulado inferior es el modelo minimal. Asimismo, se puede probar que $\{$G-Modelos$, \subseteq\}$ es un filtro.

Definición 3.12. [CS04] **(Semántica $\mathcal{GS}$)**
Sea $\mathcal{P}$ un *de.l.p.*. La *semántica basada en juegos para* $\mathcal{P}$, que notaremos $\mathcal{GS}$, es el G-modelo minimal. ■

La semántica $\mathcal{GS}$ vincula la teoría de los juegos y la teoría de modelos reflejando de manera natural, las relaciones sintácticas entre argumentos, por la similitud existente entre un árbol dialéctico y un juego de dos participantes. Asimismo, $\mathcal{GS}$ pone en evidencia el rol de las restricciones dialécticas para evitar falacias, al utilizar la noción de secuencia legal.

4. Resultados de Completitud

En esta sección se presentan los lemas y teoremas que prueban que la semántica desarrollada es sensata y completa con respecto a la semántica operacional de la DeLP. Para esto, definiremos en primer lugar una transformación que nos permita obtener un árbol dialéctico a partir de las secuencias de $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$. Se omiten las demostraciones por razones de espacio.

4.1. Transformación de $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$ a un árbol dialéctico

Como paso intermedio en las demostraciones de los resultados de la subsección siguiente, necesitamos construir con las secuencias de $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$ un árbol y comprobar que dicho árbol es dialéctico.

Definición 4.1. **(Transformación Φ)**
Sea S un conjunto de las secuencias completas con primer elemento idéntico. El árbol generado a partir de S, que notaremos $\Phi(S)$, se construye del siguiente modo:

- La raíz del árbol $\Phi(S)$ es el primer elemento de todas las secuencias en S, que notaremos a.
- Sean $X_1,\ldots,X_n$ conjuntos de secuencias generados a partir de S, con el siguiente criterio:

  $$\text{si } s=[a,x_i]s', \text{ entonces } [x_i]s'\in X_i$$

  Para todo $i\neq j: 1\leq i,j\leq n$, se verifica que $X_i\cap X_j=\varnothing$.

  Sean $\Phi(X_1),\ldots,\Phi(X_n)$ los árboles generados a partir de $X_1,\ldots,X_n$. a tiene un hijo por cada subárbol $\Phi(X_i)$ generado. Si tales árboles no existieran a es un nodo hoja. ∎

Existe un único árbol dialéctico con raíz $\langle\mathcal{A},h\rangle$, excepto por el orden en que se agregan los nodos vecinos. Así la transformación no es una función, ya que para un conjunto de secuencias S, $\Phi(S)$ devuelve árboles dialécticos diferentes aunque equivalentes.

Notaremos al conjunto de todas las secuencias completas de $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$ como $P^{Comp}_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$.

Lema 4.1. Sean $G(\langle\mathcal{A},h\rangle,\mathcal{P})$ un juego y $\Phi(P^{Comp}_{G(\langle\mathcal{A},h\rangle,\mathcal{P})})$ un árbol generado a partir de las secuencias completas de $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$. $\Phi(P^{Comp}_{G(\langle\mathcal{A},h\rangle,\mathcal{P})})$ es un árbol dialéctico con raíz $\langle\mathcal{A},h\rangle$ bajo $\mathcal{P}$. ∎

Lema 4.2. Sea T un árbol dialéctico con raíz $\langle\mathcal{A},h\rangle$ bajo $\mathcal{P}$. Luego existe un juego $G(\langle\mathcal{A},h\rangle,\mathcal{P})$ tal que $\Phi(P^{Comp}_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}) = T$. ∎

Los lemas anteriores muestran que existe una equivalencia entre las secuencias de $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$ y un árbol dialéctico bajo $\mathcal{P}$ con raíz $\langle\mathcal{A},h\rangle$.

4.2. Sensatez y Completitud

El siguiente lema es necesario para probar los teoremas 4.4 y 4.5 que son el principal resultado de este trabajo. En dicho lema mostramos que si el conjunto de secuencias completas de $P_{G(\langle\mathcal{A},h\rangle,\mathcal{P})}$ es una estrategia, luego el juego ha sido ganado por el Proponente, lo que se refleja en el árbol dialéctico a través del etiquetamiento de la raíz con "U".

Lema 4.3. Sea $\mathcal{P}$ un *de.l.p.* y $\langle\mathcal{A}, h\rangle$ una estructura de argumento bajo $\mathcal{P}$. El conjunto de las secuencias completas de $P_{G(\langle\mathcal{A}, h\rangle, \mathcal{P})}$ es una estrategia si y sólo si un árbol dialéctico para $\langle\mathcal{A}, h\rangle$ bajo $\mathcal{P}$ está marcado con U. ■

A partir de los lemas demostrados anteriormente estamos en condiciones de probar las nociones de sensatez y completitud del concepto de garantía con respecto a la caracterización semántica en términos del G-modelo minimal. Los teoremas muestran que si la respuesta para un literal bajo la semántica operativa es SI, luego ese literal pertenece al conjunto V de la semántica $\mathcal{GS}$ y si la respuesta para dicho literal es NO, luego el literal pertenece al conjunto F de la semántica $\mathcal{GS}$.

Teorema 4.4. Sean $\mathcal{P}$ un *de.l.p.* y h un átomo. h está garantizado si y sólo si h pertenece al conjunto V del G-modelo minimal bajo la semántica $\mathcal{GS}$. ■

Demostración: h está garantizado si y sólo si existe un árbol dialéctico T con raíz $\langle\mathcal{A}, h\rangle$ etiquetado con "U", por definición, si y sólo si existe un juego $G(\langle\mathcal{A}, h\rangle, \mathcal{P})$ tal que $\Phi(P^{Comp}_{G(\langle\mathcal{A}, h\rangle, \mathcal{P})}) = T$, por lema 4.2 si y sólo si $P^{Comp}_{G(\langle\mathcal{A}, h\rangle, \mathcal{P})}$ es una estrategia, por el lema 4.3 si y sólo si $G(\langle\mathcal{A}, h\rangle, \mathcal{P})$ es ganado por el Proponente, por definición si y sólo si h pertenece al conjunto V de todo G-modelo, por definición de G-modelo si y sólo si h pertenece al conjunto V del G-modelo minimal. ■

Teorema 4.5. Sean $\mathcal{P}$ un *de.l.p.* y h un átomo. $\overline{h}$ está garantizado si y sólo si h pertenece al conjunto F del G-modelo minimal bajo la semántica $\mathcal{GS}$. ■

Demostración: $\overline{h}$ está garantizado si y sólo si existe un árbol dialéctico T con raíz $\langle A, \overline{h}\rangle$ etiquetado con "U", por definición, si y sólo si existe un juego $G(\langle\mathcal{A}, \overline{h}\rangle, \mathcal{P})$ tal que $\Phi(P^{Comp}_{G(\langle\mathcal{A}, \overline{h}\rangle, \mathcal{P})}) = T$, por lema 4.2 si y sólo si $P^{Comp}_{G(\langle\mathcal{A}, \overline{h}\rangle, \mathcal{P})}$ es una estrategia, por el lema 4.3 si y sólo si $G(\langle\mathcal{A}, \overline{h}\rangle, \mathcal{P})$ es ganado por el Proponente, por definición si y sólo si h pertenece al conjunto F de todo G-modelo, por definición de G-modelo si y sólo si h pertenece al conjunto F del G-modelo minimal. ■

Hemos probado la equivalencia entre las nociones sintáctica y semántica de la DeLP. Esto permite la caracterización del sistema de razonamiento argumentativo sin recurrir a la operación. Asimismo, posibilita la comparación con otros sistemas no monotónicos.

5. Conclusiones y Trabajos Futuros

Se han presentado los principales lemas y teoremas que muestran que la semántica declarativa $\mathcal{GS}$ es sensata y completa con respecto al sistema de razonamiento procedural de la DeLP. $\mathcal{GS}$ vincula la teoría de juegos basada en el formalismo propuesto por Abramsky en [Abr97] y en la teoría de modelos. Dicha semántica es definida a través del G-modelo minimal para un *de.l.p.* y captura tres clases de respuestas posibles para un literal de la signatura del *de.l.p.* que se está analizando.

Los resultados de completitud a los que se han arribado motivan a continuar el estudio del sistema de razonamiento argumentativo, caracterizándolo a través de la semántica declarativa $\mathcal{GS}$, y a compararlo con otros sistemas no monotónico.

Si bien en esta etapa no se ha instanciado el criterio de preferencia entre argumentos, nuestro interés reside en el estudio de la especificidad generalizada.

Entre nuestros trabajos futuros, se encuentra la comparación con otras semánticas declarativas. Asimismo, se espera extender la semántica a juegos de más de dos participantes, con el objetivo de modelar un sistema multiagentes.

6. Agradecimientos

Este trabajo ha sido parcialmente financiado por la Secretaría General de Ciencia y Tecnología de la Universidad Nacional del Sur, por la Universidad Nacional del Comahue (Proyecto de Investigación 04/E046) y por la Agencia Nacional de Promoción Científica y Tecnológica (PICT 2002 No. 13096).

Referencias

[ABCMB04] Katie Atkinson, Trevor Bench-Capon, and Peter Mc Burney. A dialogue game protocol for multi-agent argument over proposals for action. Technical Report ULCS-04-007, Department of Computer Science, University of Liverpool, Liverpool, U.K., 2004.

[Abr97] Samson Abramsky. Semantics of Interaction. In A.Pitts and P. Dibyer, editors, *Semantics and Logic Computation*. Cambridge, 1997.

[AMB00] Grigoris Antoniou, Michael Maher, and David Billington. Defeasible Logic versus logic programming without negation as failure. *Journal of Logic Programming*, (42):47–57, 2000.

[Cn01] Carlos Iván Chesñevar. *Formalización de los Procesos de Argumentación Rebatible como Sistemas Deductivos Etiquetados*. PhD thesis, Universidad Nacional del Sur, 2001.

[CS99] Laura A. Cecchi and Guillermo R. Simari. Game-based approach for modeling dialectical analysis: Preliminary Report. In *Proceedings of V CACiC*, 1999.

[CS00] Laura A. Cecchi and Guillermo R. Simari. Sobre la Relación entre la Definición Declarativa y Procedural de Argumento. In *VI CACiC*, Ushuaia, 2000.

[CS02a] Laura A. Cecchi and Guillermo R. Simari. Sobre la Relación de Preferencia entre Argumentos. In *VIII CACiC*, Buenos Aires, 2002.

[CS02b] Laura A. Cecchi and Guillermo R. Simari. Un enfoque declarativo basado en juegos del razonamiento rebatible. In *Jornadas Chilenas en Computación 2002. III Workshop on Advances and Trends in Artificial Intelligence for Problem Solving (ATAI)*, Copiapó - Chile, 2002.

[CS04] Laura A. Cecchi and Guillermo R. Simari. Semántica declarativa basada en juegos para el razonamiento rebatible. 2004. Bajo consideración para su publicación en las Jornadas Chilenas en Computación 2004.

[Dun95] Phan M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning and logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.

[Gar00] Alejandro J. García. *Programación en Lógica Rebatible: Lenguaje, Semántica Operacional y Paralelismo*. PhD thesis, Universidad Nacional del Sur, 2000.

[GS99] A. García and G. R. Simari. Strong and Default Negation in Defeasible Logic Programming. In *4th Dutch/German Workshop on Nonmonotonic Reasoning Techniques and their applications*, Amsterdam, 25 - 27, Marzo 1999.

[GS04] A. García and G. Simari. Defeasible Logic Programming: An Argumentative Approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.

[KT99] A. Kakas and F. Toni. Computing argumentation in logic programming. *Journal of Logic and Computation*, (4):515–562, 1999.

[MG99] Diego Martínez and Alejandro García. Significancia de las falacias en los sistemas argumentativos. In *In Proceedings of V Congreso Argentino en Ciencias de la Computación*, Tandil, 1999.

[OS99] James Osborn and Leon Sterling. JUSTICE:A Judicial Search Tool Using Intelligent Concept Extraction. In ACM, editor, *Proceedings of the Conference Artificial Intelligence and Law*, pages 173–181, 1999.

[PSJ98] Simon Parsons, Carles Sierra, and Nick Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292, 1998.

[PV00] Henry Prakken and Gerard Vreeswijk. Logics for defeasible argumentation. In D. Gabbay, editor, *Handbook of Philosophical Logic*. Kluwer Academic, second edition, 2000.

[SGCnS02] Frieder Stolzenburg, Alejandro J. García, Carlos I. Chesñevar, and Guillermo R. Simari. Computing Generalized Specificity. *Journal of Applied Non-Classical Logics*, 12:1–27, 2002.

[Vre92] Gerard Vreeswijk. Abstract Argumentation Systems. *Artificial Intelligence*, 2(3):259–310, June 1992.

[Vre93] G. A. Vreeswijk. *Studies in Defeasible Argumentation*. PhD thesis, Vrije University, Amsterdam, Holanda, 1993.