



**Universidad Nacional del Sur**

TESIS DE DOCTOR EN CIENCIAS DE LA COMPUTACIÓN

*Un Marco Argumentativo Abstracto Dinámico*

Nicolás D. Rotstein

BAHÍA BLANCA, ARGENTINA

2010



## PREFACIO

Esta Tesis se presenta como parte de los requisitos para optar al grado Académico de Doctor en Ciencias de la Computación, de la Universidad Nacional del Sur y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otra. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el ámbito del Departamento de Ciencias e Ingeniería de la Computación durante el período comprendido entre el 2 de agosto de 2005 y el 12 de abril de 2010, bajo la dirección del Doctor Guillermo Ricardo Simari.

Nicolás D. Rotstein

e-mail: [ndr@cs.uns.edu.ar](mailto:ndr@cs.uns.edu.ar)

Departamento de Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur, Bahía Blanca, Argentina



UNIVERSIDAD NACIONAL DEL SUR  
Secretaría General de Posgrado y Educación Continua

La presente tesis ha sido aprobada el .../.../..., mereciendo la calificación de .....(.....)



## AGRADECIMIENTOS

Quiero agradecer a mis directores, tanto de hecho como de derecho, Guillermo Simari y Alejandro García, quienes me ofrecieron la oportunidad de realizar mi doctorado en este laboratorio: el LIDIA. Desde mi comienzo en *RoboCup*, su aporte en mi corta carrera en investigación fue imprescindible y valioso. Sus perfiles, diferentes en cuanto a la investigación, me han dado la oportunidad de combinar diferentes visiones, enriqueciendo mi formación.

A mis compañeros de la *salita de becarios*, con quienes compartir el espacio de trabajo ha sido un inmenso placer que, por momentos, ha llegado a atentar contra el trabajo mismo. Siempre consideraré este hecho como algo feliz y (contradictoriamente) productivo. Con muchos de ellos he trabajado directamente, escribiendo artículos o programando, y he constatado que además de grandes amigos son excelentes profesionales. Los mates y discusiones en la salita, así como las reuniones y festejos fuera de ella quedarán intercalados en mi memoria. Dentro de este mismo ámbito incluyo a varios asistentes y profesores. Con ellos, cientos de comentarios supuestos a ser breves indefectiblemente se tornaron en largas charlas –algunas de tono laboral y otras no tanto.

Quiero agradecer a mi familia por su participación, tan imprescindible como indirecta, en esta tesis. A mis padres, que tanto a mi como a mis hermanos “nos han inculcado el valor de lo intelectual antes que lo material” (en palabras de mi madre), y nos han dado la libertad sufi-

ciente durante nuestra infancia para que desarrollemos la creatividad y persigamos nuestros intereses con total independencia. A mis hermanos: Horacio, Andrea, Agustina, Flavia, Mónica, Andrés y Claudio. Crecer y estar junto a ellos ha sido y sigue siendo un privilegio invaluable.

También le doy las gracias a aquellos amigos que no están relacionados con mi entorno profesional, quienes en ciertos casos son una extensión de mi familia, y cuya presencia ha sido sumamente importante. En especial me refiero a aquellos con quienes he compartido innumerables veraneos en Pehuén-Có; las vacaciones son la infancia del año.

Finalmente, quiero agradecer al Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) por el apoyo económico brindado durante los 5 años del doctorado. También considero oportuno destacar la posibilidad que me brindó mi país para acceder a una educación pública y gratuita de calidad internacional.

## RESUMEN

El trabajo realizado en esta tesis pertenece al área de *argumentación* en inteligencia artificial. La representación de conocimiento en un formalismo basado en argumentación se realiza a través de la especificación de *argumentos*, cada uno en favor de una *conclusión* a partir de ciertas *premisas*. Dado que estas conclusiones pueden estar en contradicción, se producen *ataques* entre los argumentos. Luego, la evaluación de toda la información presente podrá dar preponderancia a algunos argumentos por sobre aquellos que los contradicen, produciendo un conjunto de conclusiones que se considerarán *garantizadas*.

El objetivo principal de esta tesis es la definición de un nuevo *marco argumentativo* capaz de manejar *dinámica de conocimiento*. En este sentido, se da una representación no sólo a los argumentos, sino que también se introduce la noción de *evidencia* como entidades especiales dentro del sistema. En cada instante, el conjunto de evidencia se corresponde con la situación actual, dándole contexto al marco argumentativo. La plausibilidad de los argumentos en un instante dado depende exclusivamente de la evidencia disponible. Cuando la evidencia es suficiente para dar soporte a un argumento, éste se denominará *activo*. También se considera la posibilidad de que algunos argumentos se encuentren activos aún sin encontrar soporte directamente desde la evidencia, ya que podrían hacerlo a través de las conclusiones de otros argumentos activos. Estas conexiones entre argumentos dan lugar a lo que en esta tesis se denomina

*estructura argumental*, proveyendo una visión un tanto más compleja que la usual en cuanto a la representación de conocimiento argumentativo.

Los resultados obtenidos en esta tesis permitirán estudiar la dinámica de conocimiento en sistemas argumentativos. En la actualidad, ya se han publicado artículos que presentan un formalismo que combina argumentación y la teoría clásica de revisión de creencias. En esta línea de investigación se definen operadores de cambio que se aplican sobre el marco argumentativo abstracto dinámico y tienen como objetivo alcanzar cierto estado del sistema; por ejemplo, garantizar un argumento determinado. Por otra parte, este marco también permitirá estudiar métodos para acelerar el cómputo de garantía a partir del proceso de razonamiento realizado en estados anteriores.

**PALABRAS CLAVE:** Inteligencia Artificial, Representación de Conocimiento, Razonamiento No Monótono, Argumentación Computacional, Argumentación Abstracta, Dinámica de Conocimiento



## ABSTRACT

This thesis falls within the area of *argumentation* in artificial intelligence. Knowledge representation in an argumentation-based formalism is performed through the specification of *arguments*, each of which supports a *claim* from certain *premises*. Provided that these claims might be in contradiction, *attacks* may arise among arguments. The evaluation of all the information at stake could make some arguments prevail on top of those which contradict them, yielding a set of conclusions considered *warranted*.

The main objective of this thesis is the definition of a new *argumentation framework* capable of dealing with *knowledge dynamics*. In this sense, representation is not constrained not only to arguments, but the notion of *evidence* is also introduced, as a special entity within the system. At each moment, the set of evidence will correspond to the current situation, providing context to the argumentation framework. The plausibility of arguments at a given moment depends exclusively on the availability of evidence. Whenever evidence proves to be enough to support an argument, the latter will be referred as *active*. Some arguments could be active even when they do not find enough support directly from evidence, but from other active arguments' claims. These connections among arguments give place to what we call *argumental structure*, providing a more complex representation than what is usually found in the literature.

The results obtained in this thesis will allow the study of knowledge dynamics in argumentation systems. Currently, several articles have been published introducing a formalism that combines argumentation and classic belief revision theory. This research line is devoted to the definition of change operators applied over the dynamic abstract argumentation framework with the goal of reaching a certain state of the system; for instance, to seek warrant for a given argument. Another research line that benefits from this framework studies methods that look for a speed-up of the warrant computation procedure, based on the reasoning procedure performed in previous states.

**KEYWORDS:** Artificial Intelligence, Knowledge Representation, Non-monotonic Reasoning, Computational Argumentation, Abstract Argumentation, Knowledge Dynamics

## Índice general

1.. <i>Introducción</i> . . . . .	1
1.1. Argumentación Abstracta . . . . .	2
1.2. Representación de Conocimiento y Razonamiento . . . . .	3
1.3. Marcos Argumentativos Abstractos . . . . .	3
1.3.1. Argumentos y Ataques . . . . .	4
1.3.2. Semánticas Argumentativas . . . . .	4
1.4. La Consideración de Subargumentos . . . . .	7
1.5. Dinámica en Argumentación . . . . .	8
1.6. Argumentación Estática vs. Dinámica . . . . .	9
1.7. Teoría del Cambio en Argumentación . . . . .	10
1.8. El DAF y sus Aplicaciones . . . . .	11
1.9. Contribuciones . . . . .	12
2.. <i>El Marco Argumentativo Abstracto Dinámico (DAF)</i> . . . . .	15
2.1. Argumentos y el Rol de la Evidencia . . . . .	15
2.1.1. ¿Qué es evidencia? . . . . .	15
2.1.2. Conflicto . . . . .	16
2.1.3. Una Interfaz para Argumentos . . . . .	17
2.1.4. Principios de Coherencia . . . . .	19
2.2. Definiendo el DAF . . . . .	20

---

2.2.1. Evidencia, Premisas y Conclusiones . . . . .	21
2.2.2. Argumentos Activos e Inactivos . . . . .	21
2.2.3. Estructuras Argumentales . . . . .	23
2.2.4. El DAF . . . . .	33
2.3. La Instancia Activa . . . . .	34
2.4. Modificando un DAF . . . . .	43
2.5. Semánticas Argumentativas y el DAF . . . . .	51
2.5.1. Argumentación y Árboles de Dialéctica . . . . .	52
2.5.2. Grafos de Argumentos . . . . .	58
3.. <i>DAF vs. Sistemas Basados en Reglas</i> . . . . .	59
3.1. Programación en Lógica Rebatible . . . . .	59
3.2. Programas Lógicos Rebatibles en Términos del DAF . . . . .	65
3.3. El Sistema ASPIC . . . . .	75
3.4. Teorías Rebatibles en Términos del DAF . . . . .	80
4.. <i>Teoría del Cambio en Argumentación</i> . . . . .	85
4.1. Contracción Garantizante de Estruct. Argumentales . . . . .	86
4.2. Expansión Garantizante de Estructuras Argumentales . . . . .	101
4.3. Un Enfoque Híbrido a través de Expansiones y Contrac- ciones de Estructuras Argumentales . . . . .	102
4.4. Revisión del DAF por la Relación de Derrota . . . . .	103
5.. <i>Refinando Operaciones de Cambio</i> . . . . .	105
5.1. Relevancia del Cambio . . . . .	105
5.2. Bonsai de Dialéctica . . . . .	110

---

5.2.1. Cómo Calcular la Fuerza de una Estructura Argumental . . . . .	113
5.2.2. Construcción de un Bonsai de Dialéctica . . . . .	115
5.2.3. Bonsai <i>Fast-prune</i> . . . . .	120
5.2.4. Resultados Experimentales . . . . .	122
5.2.5. Discusión Final . . . . .	125
6.. <i>Aplicaciones del DAF</i> . . . . .	129
6.1. Una Arquitectura de Agentes . . . . .	129
6.1.1. La Arquitectura . . . . .	130
6.1.2. Creencias . . . . .	132
6.1.3. Deseos . . . . .	134
6.1.4. Intenciones . . . . .	139
6.1.5. Dominio de Aplicación: Fútbol de Robots . . . . .	142
6.1.6. Dominio de Aplicación: Sistema de Seguridad . . . . .	144
6.2. Análisis de Casos Legales . . . . .	151
7.. <i>Conclusiones</i> . . . . .	157



## Índice de figuras

1.1. (a) $\mathcal{A}$ aceptable con respecto a $S$ (b) Conjunto admisible $S$	6
1.2. Dos marcos argumentativos para el Ejemplo 1.1 . . . . .	6
1.3. Dos marcos argumentativos para el Ejemplo 1.2 . . . . .	7
1.4. Dos marcos argumentativos para el Ejemplo 1.3 . . . . .	7
2.1. Estructuras argumentales mal formadas . . . . .	27
2.2. Estructuras argumentales bien formadas . . . . .	28
2.3. Estructuras argumentales del Ejemplo 2.3 . . . . .	32
2.4. Estructuras argumentales del Ejemplo 2.4 . . . . .	35
2.5. Estructuras argumentales con mismo argumento tope . .	38
2.6. DAF del ejemplo 2.5 . . . . .	41
2.7. Instancias activas del Ej. 2.5 de acuerdo a distintos con- juntos de evidencia . . . . .	42
2.8. Dos DAFs con sus respectivos conjuntos de evidencia . .	50
2.9. Resultado de la mezcla de los DAFs en la Figura 2.8 . .	51
2.10. Árbol potencial del Ejemplo 2.7 . . . . .	55
2.11. Árboles activos del árbol potencial de la Figura 2.10 . . .	56
3.1. Árbol de dialéctica del Ejemplo 3.4 . . . . .	65
3.2. Las tres posibles formas de inclusión de los nodos $S$ e $I$ .	73
4.1. Árbol de dialéctica del Ejemplo 4.1 . . . . .	88

---

4.2.	Árbol de dialéctica de la Figura 4.1 sin las líneas de ataque	88
4.3.	Esquema del principio de preservación . . . . .	91
4.4.	Árbol hipotético del Ejemplo 4.4 . . . . .	94
4.5.	(a) Árbol de dialéctica del Ejemplo 4.5 (b) Árbol hipotético	96
4.6.	Árbol producto de la revisión del Ejemplo 4.6 . . . . .	100
5.1.	Árbol de dialéctica potencial del Ejemplo 2.7 . . . . .	107
5.2.	Árbol potencial considerando la relación de conflicto $\bowtie_{5,1}$	107
5.3.	Árbol con valores de fuerza del Ejemplo 5.3 . . . . .	115
5.4.	Comparación entre fuerza potencial y actual . . . . .	116
5.5.	Árbol de dialéctica activo y poda <i>fast-prune</i> del Ejemplo 5.4	121
5.6.	Árboles (a) potencial, (b) activo y (c) bonsai del Ejemplo 5.5	123
6.1.	Arquitectura BDI de agentes basada en el DAF . . . . .	131
6.2.	Digrafo de estructuras argumentales para el Ejemplo 6.2	136
6.3.	Un escenario para el agente futbolista . . . . .	142
6.4.	Un escenario para el agente futbolista . . . . .	143
6.5.	DAF de filtrado del agente de seguridad . . . . .	146
6.6.	Instancia activa del agente de seguridad según $\mathcal{B}_1$ . . . . .	149
6.7.	Árbol de dialéctica con los alegatos iniciales . . . . .	153
6.8.	Garantía de la inocencia de Jack, luego de la revisión . . .	154
6.9.	Árbol de dialéctica derrotando el argumento de la defensa	156
6.10.	(a) Activación del derrotador $\mathcal{D}$ para $\mathcal{B}_1$ (b) Resultado de la revisión . . . . .	156



## 1. INTRODUCCIÓN

El trabajo realizado en esta tesis pertenece al área de *argumentación* en inteligencia artificial [CML00, PV02, BCD07, RS09]. Esta disciplina es relativamente nueva y forma parte del área de razonamiento no monótono. La representación de conocimiento en argumentación se realiza a través de la especificación de *argumentos*, cada uno de los cuales da soporte a una *conclusión* a partir de un conjunto de *premisas*. Un formalismo argumentativo brinda la posibilidad de representar conocimiento en contradicción, dado que la presencia de un argumento puede resultar conflictiva para otros. Lo que resulta una ventaja es la tolerancia a la inconsistencia. Es decir, es posible obtener conclusiones válidas aún cuando el conocimiento tenga porciones conflictivas; el proceso argumentativo decidirá qué prevalece dada una contradicción.

En argumentación, el razonamiento es realizado mediante la evaluación y confrontación de argumentos en conflicto. Se dice que un argumento  $\mathcal{A}$  *derrota* a un argumento  $\mathcal{B}$  si ambos están en conflicto y  $\mathcal{A}$  es preferido a  $\mathcal{B}$ . En enfoques teóricos, esta relación de ataque suele definirse como un conjunto de pares ordenados de argumentos, donde la primera componente ataca a la segunda. Un ataque doble usualmente se entiende como una situación de *bloqueo*; es decir, ninguno de los dos argumentos involucrados es considerado como un derrotador efectivo del otro.

La teoría de argumentación puede ser dividida en dos subáreas: *argumentación abstracta*, y *argumentación basada en reglas*. En la visión

abstracta, los argumentos son considerados como cajas negras y se evita toda referencia a su composición interna. Por otra parte, en el enfoque basado en reglas, un argumento es definido como un conjunto de reglas que, bajo ciertas condiciones (minimalidad, consistencia, *etc.*), deriva su conclusión. En este trabajo, el foco se sitúa sobre la visión abstracta de argumentación, dado que resulta un enfoque más apropiado para estudiar propiedades de alto nivel. Aún así, los resultados obtenidos pueden ser traducidos a formalismos basados en reglas, tal como se mostrará en el Capítulo 3.

### 1.1. Argumentación Abstracta

La primera propuesta introduciendo un marco argumentativo abstracto fue dada por Dung [Dun95], en un artículo cuyos resultados aún son objeto de análisis en la literatura. Varios autores han tomado esta propuesta básica para extenderla en diversos sentidos [BCD07, WBC07], logrando formalismos con una capacidad expresiva de un espectro realmente amplio.

La especificación de un sistema argumentativo abstracto implica definir dos conjuntos: el de argumentos y el de ataques entre ellos. Esto permite representar un marco argumentativo como un grafo dirigido, donde los nodos son los argumentos y los arcos, los ataques, apuntando al argumento siendo derrotado. Este grafo suele resultar una herramienta útil para visualizar el sistema y analizar propiedades. La finalidad de representar conocimiento en un marco argumentativo es tener la capacidad de obtener conclusiones del sistema a pesar de la potencial inconsistencia del conocimiento. Una *semántica* aplicada al marco argumentativo determinará el criterio por el cual ciertos argumentos serán *garantizados* (*i.e.*, aceptados a pesar del conocimiento en contradicción) y otros,

rechazados. El conjunto de argumentos garantizados contendrá aquellos argumentos en los cuales se puede creer luego de haber considerado todo el conocimiento relacionado.

## 1.2. Representación de Conocimiento y Razonamiento

Dadas sus características, el enfoque argumentativo está preparado para representar conocimiento inherentemente inconsistente, lo cual es de utilidad para la toma de decisiones y/o determinación de creencias considerando información en oposición. En lo que resta de este capítulo quedará claro que, en argumentación, el carácter no monótono del razonamiento surge de la pugna entre argumentos: la aparición de argumentos nuevos en el sistema puede llevar a la obtención de un subconjunto de las conclusiones previas.

La comunidad de argumentación usualmente se enfoca en la propuesta de nuevos elementos teóricos para aumentar la capacidad expresiva de los formalismos. Otras extensiones se dedican a manejar diferentes aspectos del proceso argumentativo y su utilización: su capacidad para manejar diálogos [DDBC05], protocolos de negociación [SJNP98, ADM07], y estrategias [RLT09, RL08]. Complementariamente, algunos enfoques han estudiado la idoneidad de la argumentación dentro de distintos contextos de aplicación, tales como sistemas de recomendación [CM04], sistemas multi-agente [RA06, RGS08] y la web [RZR07b, RZR07a].

## 1.3. Marcos Argumentativos Abstractos

Esta sección está dedicada a la descripción del concepto de *marco argumentativo abstracto* (*abstract argumentation framework*, de aquí en adelante *AF*) en la forma en que se suele presentar en la literatura,

basándose en las definiciones iniciales en [Dun95].

### 1.3.1. Argumentos y Ataques

Como fue establecido en la Sección 1.2, los AFs están compuestos de argumentos y ataques entre ellos. En *argumentación abstracta*, los argumentos son tratados como entidades abstractas e indivisibles, sin hacer referencia sobre su estructura interna. La relación de ataque se define como un conjunto de pares de argumentos, y cada par representa un ataque de la primera componente sobre la segunda.

**Definición 1.1 (Marco Argumentativo [Dun95])** *Un marco argumentativo (AF) es una tupla  $\langle W, \hookrightarrow \rangle$ , donde  $W$  es un conjunto finito de argumentos, y  $\hookrightarrow \subseteq (W \times W)$ , es la relación de ataque sobre  $W$ .*

Los AFs son esencialmente un grafo donde los nodos son argumentos y los arcos, ataques. Este tipo de marcos representan el máximo grado de abstracción para un formalismo argumentativo. A partir de esta visión pueden definirse nociones centrales, y el foco suele posarse sobre las *semánticas argumentativas*. Muchas extensiones han sido definidas en la literatura sobre el marco de Dung: *value-based frameworks* (marcos basados en valores) [BC02], *bipolar frameworks* (marcos bipolares) [ACLSL08], *labelled frameworks* (marcos etiquetados) [CS06], e incluso incorporando la noción de subargumento [MGS07]. En el capítulo siguiente se explicará en detalle el aporte de esta tesis, que puede ser visto como una extensión o un refinamiento del AF clásico.

### 1.3.2. Semánticas Argumentativas

Al representar conocimiento a través de un AF se debe proveer un mecanismo para obtener las conclusiones inferidas por el sistema; esto es, el

AF debe poseer una *semántica* asociada. La aplicación de una semántica a un AF retornará un conjunto de extensiones, y cada extensión será un conjunto de argumentos. El requerimiento mínimo para una extensión es ser *libre de conflictos*. Las nociones explicadas a continuación fueron tomadas de [Dun95]. Dado un AF  $\phi = \langle W, \hookrightarrow \rangle$  y un subconjunto  $S \subseteq W$ :

- **Libertad de conflicto:**  $S$  es libre de conflicto con respecto a “ $\hookrightarrow$ ” sssi ningún argumento  $\mathcal{A} \in S$  es tal que  $\mathcal{A} \hookrightarrow \mathcal{B}$ , para todo  $\mathcal{B} \in S$ .
- **Argumento aceptable:** Un argumento  $\mathcal{A} \in W$  es aceptable con respecto a  $S$  y “ $\hookrightarrow$ ” sssi para cada argumento  $\mathcal{B} \in W$  se cumple que si  $\mathcal{B} \hookrightarrow \mathcal{A}$  entonces existe un argumento  $\mathcal{C} \in S$  tal que  $\mathcal{C} \hookrightarrow \mathcal{B}$ . Este concepto se ilustra en la Figura 1.1(a).
- **Admisibilidad:**  $S$  es admisible con respecto a “ $\hookrightarrow$ ” sssi  $S$  es libre de conflicto con respecto a “ $\hookrightarrow$ ” y cada argumento en  $S$  es aceptable con respecto a  $S$  y “ $\hookrightarrow$ ”. Este concepto se ilustra en la Figura 1.1(b).
- **Extensión preferida:** Una extensión preferida de  $\phi$  es un conjunto maximal (con respecto a inclusión de conjuntos)  $S \subseteq W$  tal que  $S$  es admisible con respecto a “ $\hookrightarrow$ ”.
- **Función característica:** La función característica de  $\phi$  es:
  - $F_\phi : 2^W \rightarrow 2^W$
  - $F_\phi(S) = \{\mathcal{A} \mid \mathcal{A} \text{ es aceptable con respecto a } S \text{ y “}\hookrightarrow\text{”}\}$
- **Extensión grounded:** La extensión *grounded* de  $\phi$  es el mínimo punto fijo de la función característica  $F_\phi$ .
- **Extensión completa:** Si  $S$  es un conjunto admisible,  $S$  es una extensión completa de  $\phi$  sssi cada argumento en  $S$  es aceptable con respecto a  $S$  y “ $\hookrightarrow$ ”.

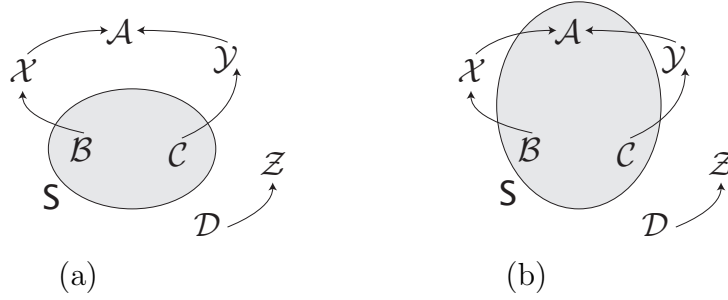


Fig. 1.1: (a)  $\mathcal{A}$  aceptable con respecto a  $S$  (b) Conjunto admisible  $S$

El conjunto de *extensiones preferidas* de un AF constituye su semántica *crédula*, mientras que la semántica *escéptica* está definida por la extensión *grounded*. La anterior permite múltiples extensiones, mientras que la última retorna sólo una, que puede ser vacía. La extensión completa puede verse como un enlace entre las preferidas y la *grounded*.

**Teorema 1.1** ([Dun95]) *Se verifican las siguientes propiedades:*

1. *Cada extensión preferida es completa, pero no vice versa.*
2. *La extensión grounded es la menor (según  $\subseteq$ ) extensión completa.*

**Ejemplo 1.1** *En la Figura 1.2 pueden verse dos AFs. En el AF de la figura (a) el conjunto de extensiones completas es:  $\{\emptyset, \{A\}, \{B\}\}$ , mientras que en el de la figura (b) es:  $\{\emptyset, \{A, C\}, \{B\}\}$*

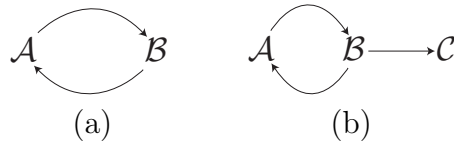


Fig. 1.2: Dos marcos argumentativos para el Ejemplo 1.1

**Ejemplo 1.2** *En los AFs de la Figura 1.3 se muestra cómo la existencia de ciclos afecta a las extensiones preferidas. En el AF de la figura (a) el*

conjunto de extensiones preferidas es:  $\{\{\mathcal{A}_1, \mathcal{C}\}, \{\mathcal{A}_2, \mathcal{C}\}\}$ , mientras que en el de la figura (b) es:  $\{\{\mathcal{A}, \mathcal{C}\}, \{\mathcal{A}, \mathcal{D}\}, \{\mathcal{B}, \mathcal{D}\}\}$ .

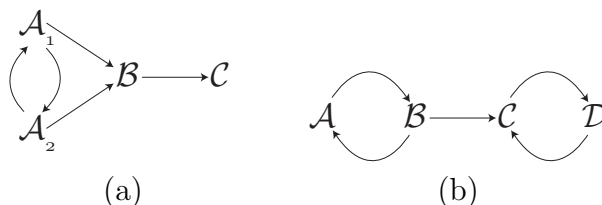


Fig. 1.3: Dos marcos argumentativos para el Ejemplo 1.2

**Ejemplo 1.3** En los AFs de los Ejemplos 1.1 y 1.2 la extensión grounded es el conjunto vacío, dado que la función característica no puede encontrar argumentos aceptables con respecto al conjunto vacío, i.e., no derrotados por ningún argumento. Consideremos una variante de los AFs del Ejemplo 1.2, en los que la extensión grounded es no vacía. Estos AFs están ilustrados en la Figura 1.4. En el AF de la figura (a) la extensión grounded es:  $\{\mathcal{A}_2, \mathcal{C}\}$ , mientras que en el de la figura (b) es:  $\{\mathcal{A}\}$ .

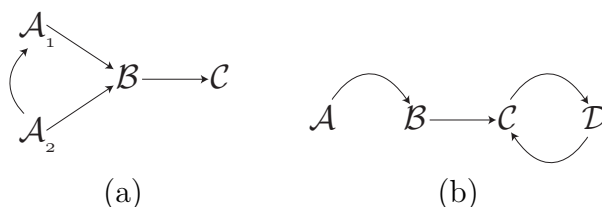


Fig. 1.4: Dos marcos argumentativos para el Ejemplo 1.3

## 1.4. La Consideración de Subargumentos

Muchas extensiones han sido definidas sobre AFs para ampliar las capacidades de representación de conocimiento. Una de las adiciones más significativas que afecta directamente al objeto de esta tesis es la introducción de la noción de *subargumento*. La intuición detrás del concepto

de subargumento puede describirse como “una porción de un argumento que es un argumento por sí misma”. Los subargumentos son de utilidad para descomponer argumentos en piezas más pequeñas. Dado que estas piezas son, a su vez, argumentos válidos, éstas pueden ser consideradas separadamente; por ejemplo, pueden ser atacadas. La consideración de subargumentos provee medios para una representación de conocimiento más poderosa, al costo de un incremento en la complejidad: todo análisis llevado a cabo sobre argumentos se torna inherentemente recursivo.

En esta tesis el concepto de subargumento será incorporado al formalismo, y tendrá repercusión en la complejidad de algunos métodos que operan sobre el marco argumentativo dinámico, tal como las operaciones de cambio descritas en el Capítulo 4. Por otra parte, al permitir tal granularidad en la representación, en el Capítulo 3 mostraremos que es posible expresar dos formalismos basados en reglas en términos del marco dinámico. La consideración de subargumentos suele implicar una visión abstracta más próxima a los enfoques prácticos de la argumentación.

### 1.5. *Dinámica en Argumentación*

La tesis se centra en la visión abstracta de la argumentación aplicada a la inteligencia artificial. El objetivo principal es la definición de un nuevo marco argumentativo capaz de manejar *dinámica de conocimiento*. Este *marco argumentativo abstracto dinámico* se denominará *DAF*, sigla basada en *dynamic abstract argumentation framework*. En este marco se da una representación no sólo a los argumentos, sino que también se introduce la noción de *evidencia*. En cada instante, el conjunto de evidencia se corresponderá con la situación actual, dándole contexto al marco argumentativo. Adicionalmente, se dota a los argumentos de ciertos atributos: una *interfaz* proveyendo conclusión y premisas. De esta



manera, la validez de los argumentos en un instante dado puede verificarse a través de la evidencia disponible. Cuando esto se dé, el argumento en cuestión se denominará *activo*. También se considera la posibilidad de que algunos argumentos se encuentren activos aún sin encontrar soporte por evidencia, ya que podrían hacerlo a través de las conclusiones de otros argumentos activos. Estas conexiones entre argumentos dan lugar a lo que en esta tesis se denomina *estructura argumental*, que da lugar a una visión más compleja que la usual en cuanto a la representación de conocimiento vía argumentos.

El DAF contiene argumentos, ataques y evidencia. En cada instante, algunos argumentos estarán activos, lo cual determinará que sólo ciertos ataques también lo estarán. Los argumentos y ataques restantes permanecerán inactivos, representando aquel conocimiento que es válido, pero que no se corresponde con la situación actual. En el Capítulo 6 se mostrará cómo este conocimiento inactivo resulta de utilidad al realizar razonamiento hipotético.

Por otra parte, es importante establecer una relación concreta entre el marco dinámico propuesto en esta tesis y la noción usual de marco argumentativo. Esto permitirá no sólo utilizar los resultados obtenidos en la literatura, sino también la extensión de ciertos conceptos. Principalmente, resulta interesante estudiar cómo las semánticas definidas para el sistema de Dung pueden aplicarse al marco aquí propuesto. Esta relación estará dada por la *instancia activa* y se define en el Capítulo 2.

## 1.6. Argumentación Estática vs. Dinámica

En esta tesis denominamos “argumentación estática” al proceso de razonamiento realizado por aquellos sistemas que siguen la definición para marcos abstractos dada por Dung. La razón detrás de esto es que, cuando

se trabaja en este tipo de marcos, no se hace referencia a una situación posiblemente cambiante ni se formaliza cómo considerar la evolución de los mismos. Aquí estamos interesados en analizar la dinámica dentro de un formalismo argumentativo para proveer herramientas de meta-nivel que permitan analizar situaciones posibles y dictar la evolución del sistema. Por lo tanto, no sólo reconoceremos y reflejaremos situaciones cambiantes dentro del marco, sino que también proveeremos los medios para razonar acerca de cómo podría evolucionar el marco de una manera conveniente. Por ejemplo, si la meta es alcanzar garantía para un conjunto de argumentos, sería interesante contar con una herramienta que nos indique qué pasos de evolución debería seguir el marco para alcanzar el objetivo planteado. Esto puede considerarse al utilizar este formalismo en una arquitectura de agentes, donde un estado meta podría especificarse como un conjunto de argumentos garantizados.

### 1.7. *Teoría del Cambio en Argumentación*

El hecho de considerar dinámica de conocimiento en la teoría argumentativa posibilita la aplicación de conceptos de revisión de creencias. El Capítulo 4 se dedicará a la presentación de un operador de revisión sobre el DAF. Este operador agregará un argumento al marco buscando su garantía de manera tal que las modificaciones que hayan sido necesarias provoquen mínimo cambio, de acuerdo a algún criterio. En ese sentido, el DAF resultante representa el estado más cercano al actual en el que el argumento en cuestión estaría garantizado. Es decir, la revisión no cambia el entorno (o “mundo”), sino la teoría.

La revisión puede seguir cuatro enfoques, según se modifique el marco a través de desactivación de argumentos, activación, ambos, o modificación de la relación de derrota. En el Capítulo 4 se describirá con detalle la

---

primera versión, y luego sólo se discutirán las complicaciones subyacentes a las tres restantes. Cabe mencionar que esta línea de investigación constituye trabajo ya publicado y en curso, realizado en conjunto con Martín Moguillansky [RMF<sup>+</sup>08a, RMF<sup>+</sup>08b], y su descripción completa desviaría el foco de esta tesis.

Las nociones de cambio aplicadas al conjunto de argumentos implicarán una de las siguientes alternativas:

- variación del conjunto de evidencia,
- modificación del conjunto de trabajo de argumentos,
- cambio a nivel de derrota.

Cada tipo de cambio tiene un significado diferente: el de evidencia puede interpretarse como una meta, un estado del mundo al cual se desea llegar. Por ejemplo, un agente podría concluir que debe moverse a una determinada locación y realizar cierta acción para llegar al estado objetivo. Por su parte, el cambio en los argumentos considerados como válidos reflejaría una nueva especificación de conocimiento, o una corrección del mismo. Por último, el cambio a nivel de la relación de derrota podría indicar un cambio en las preferencias.

## 1.8. El DAF y sus Aplicaciones

La formalización de un marco argumentativo capaz de manejar dinámica facilita la aplicación de técnicas de argumentación en diversas áreas. En el Capítulo 6 se presentarán dos aplicaciones del DAF:

1. Una arquitectura BDI de agentes cuyos módulos están definidos en base a DAFs; estos agentes se denominarán *BDI-DAF*. Esta generalización explora el uso de argumentación para especificar agentes de

manera tal que sirva de plantilla para la implementación utilizando cualquier formalismo argumentativo.

2. Análisis de casos legales. La relación entre argumentación y leyes dentro del área de inteligencia artificial ha sido motivo de estudio en los últimos años, demostrando ser un gran campo de aplicación, donde la formalización de herramientas específicas surge naturalmente. La presentación se enfocará en la utilización de operadores de cambio en marcos que representan litigios, mostrando cómo podría haberse alterado el resultado final en un caso hipotético; es decir, bajo qué condiciones el resultado habría sido distinto.

### 1.9. *Contribuciones*

La contribución principal de esta tesis es la presentación de un nuevo marco argumentativo abstracto, el DAF, capaz de lidiar con dinámica de conocimiento. La definición completa del sistema es dada en el Capítulo 2. El DAF presenta elementos que no han sido estudiados en la literatura, tal como la consideración de un conjunto distinguido de evidencia, el cual es tomado como base para establecer qué argumentos pueden utilizarse para efectuar razonamiento. El Capítulo 3 se dedica a la comparación entre el DAF y dos sistemas basados en reglas, definiendo una conversión de los elementos de estos últimos en términos del primero.

A partir de este marco dinámico se abre la posibilidad de estudiar nociones de cambio dentro de un formalismo argumentativo. La incorporación de conceptos de revisión de creencias para definir operaciones de cambio (en este caso, de revisión) dentro del DAF se explica en detalle en el Capítulo 4, y también se proponen definiciones alternativas del operador, cada una enfocada en la modificación de un aspecto diferente

---

del sistema. Lo presentado en este capítulo es sólo la parte relevante a esta tesis de lo que actualmente se está desarrollando en esta línea de investigación.

A raíz del estudio del cambio en el marco argumentativo, surge un aporte complementario, dedicado a explorar la correlación entre la dinámica de la base de conocimiento y el estado de garantía de los argumentos. Esto se introduce en el Capítulo 5, donde no sólo se definen las condiciones bajo las cuales un cambio en el conocimiento no implica un cambio en el estado de garantía de cierto argumento, sino que también se define un mecanismo de poda para aquellos casos en los que la única opción es recalcular garantía. Esta técnica se vale de la dinámica en cuanto a que considera un estado inicial cuya información es utilizada para generar valores heurísticos que asisten en la reducción del espacio de búsqueda al evaluar el estado de un argumento.

Las aplicaciones estudiadas en el Capítulo 6 representan una parte importante de la contribución del trabajo hecho en esta tesis, ya que la utilidad de un formalismo de representación de conocimiento y razonamiento sólo puede probarse mediante su uso en aplicaciones concretas. La definición de la arquitectura de agente BDI-DAF demuestra cómo puede utilizarse el DAF en líneas de investigación preexistentes, ofreciendo un alto grado de abstracción para definir nuevos elementos teóricos basados en argumentación; mientras que la utilización de operaciones de cambio sobre un DAF representando un litigio sugiere una aplicación natural del formalismo argumentativo que puede resultar de utilidad como herramienta de análisis. En resumen, con respecto a las aplicaciones, el formalismo definido en esta tesis provee un marco de trabajo que, si bien se mantiene dentro de una abstracción teórica, procura mantenerse cercano a una implementación.

Existen otras líneas de investigación cuyos resultados se verán enriquecidos a partir de la definición del DAF. Entre otras, la especificación de un marco formal para dar soporte a diálogos entre dos o más agentes, lo cual requeriría tomar consideraciones diferentes al caso de la argumentación monológica. Principalmente, la base de conocimiento común (producto del diálogo entre los agentes) será un DAF conformado a través de sucesivas operaciones de cambio, como las definidas en la Sección 2.4. En particular, resulta de interés resolver la fusión del conocimiento de los agentes que se encuentra en desacuerdo; por ejemplo, la preferencia al momento de decidir una derrota entre un par de argumentos. Por último, en el Capítulo 7, se presentan las conclusiones finales de esta tesis, junto a una discusión del trabajo futuro en las diversas líneas de investigación que se basan en ella, las cuales conforman trabajo en curso.

## 2. EL MARCO ARGUMENTATIVO ABSTRACTO DINÁMICO (DAF)

El *marco argumentativo abstracto dinámico* (DAF, por su sigla en inglés) es el principal aporte de esta tesis, y los capítulos subsiguientes se enfocarán en el uso de este formalismo. En el presente capítulo se definirán todos los elementos que componen al DAF, permitiendo la codificación de conocimiento en forma de razones a favor y en contra de ciertas sentencias, a partir de ciertas premisas. También se describirá cómo estas razones se podrán componer para formar *estructuras argumentales*, a los fines de soportar efectivamente la conclusión final a partir del conjunto de evidencia disponible. El objetivo general es extender la teoría clásica de argumentación abstracta [CML00, PV02] para definir un marco capaz de representar y manipular la dinámica del conocimiento.

### 2.1. Argumentos y el Rol de la Evidencia

#### 2.1.1. ¿Qué es evidencia?

En esta tesis se considera a la *evidencia* como una entidad separada, perteneciente a un lenguaje lógico cuyas sentencias representarán hechos. Este dominio de sentencias lógicas será denotado como “ $\mathcal{E}$ ”, y también será el lenguaje base de *premisas* y *conclusiones*. El conjunto de evidencia compone la base de conocimiento a partir de la cual algunos argumentos pueden ser considerados *activos*. Los detalles concernientes a la activa-

ción de argumentos serán dados más adelante. La consideración de estas características de los argumentos, *i.e.*, premisas, inferencia y conclusión, ha formado parte de la literatura en lógica y argumentación desde los comienzos del área [Tou59, Wal96], y más recientemente en [CMM<sup>+</sup>06].

La evidencia es considerada separadamente a los argumentos, dado que no representa una razón para una dada sentencia, sino que simplemente es un medio para declarar que esa sentencia se verifica (indiscutiblemente) en el instante de tiempo actual. Si bien podría definirse un caso particular de argumento “vacío” en favor de una pieza evidencia, esto sugeriría que la misma puede ser puesta en discusión. Incluso requeriría una verificación de consistencia para este tipo de argumentos, de modo que no existan dos de ellos soportando evidencias complementarias. Este tratamiento especial le quitaría homogeneidad al formalismo, por lo cual se decidió que los argumentos serán utilizados sólo para representar aquello que es discutible, mientras que todo el conocimiento factual será codificado vía piezas de evidencia.

### 2.1.2. Conflicto

La noción de conflicto es una componente central en cualquier sistema argumentativo, por ello el DAF estará provisto de un conjunto conteniendo todos los conflictos entre argumentos. Paralelamente, dado que el dominio  $\mathfrak{L}$  podría expresar sentencias positivas y negativas, existirán conflictos que necesariamente deberán pertenecer a dicho conjunto, tal como se formalizará en la Definición 2.3. A lo largo de esta tesis, utilizaremos la notación complemento para expresar que una sentencia es la negación de otra:  $\bar{\alpha} = \neg\alpha$ ,  $\overline{\bar{\alpha}} = \alpha$ . Más allá de aquellos argumentos con conclusiones complementarias, la relación de conflicto permitirá establecer pares de argumentos en conflicto que no cumplen esa propiedad.



Por ejemplo, *ir\_izquierda* e *ir\_derecha* no son sentencias sintácticamente complementarias, y resultaría artificial especificar mediante un argumento que cada una de ellas contradice a la otra; simplemente debe poderse establecer que ambas no pueden verificarse simultáneamente, *i.e.*, están en conflicto.

### 2.1.3. Una Interfaz para Argumentos

Consideraremos a los argumentos como los pasos de razonamiento más pequeños a ser representados en el DAF; es decir, el nivel más fino de granularidad en cuanto a representación de conocimiento.

**Definición 2.1 (Argumento)** *Dado un lenguaje  $\mathcal{L}$ , un argumento  $\mathcal{A}$  es un paso de razonamiento en favor de una conclusión en  $\mathcal{L}$  a partir de un conjunto de premisas en  $\mathcal{L}$ .*

Un argumento puede representar cualquier modo de inferencia, sea de tipo deductivo o no. A pesar de ser abstractos, los argumentos estarán provistos de una *interfaz*: tienen un conjunto de premisas asociado, así como una única conclusión. Esta interfaz debería ser común a cualquier implementación de marcos argumentativos, en particular, aquellos descritos en el Capítulo 3. Para facilitar el manejo de estas características, se asumen dos funciones:

- $\text{pr} : \text{Args} \rightarrow 2^{\mathcal{L}}$ ; dado un argumento  $\mathcal{A}$ ,  $\text{pr}(\mathcal{A})$  retorna el conjunto de premisas de  $\mathcal{A}$ ,
- $\text{cl} : \text{Args} \rightarrow \mathcal{L}$ ; dado un argumento  $\mathcal{A}$ ,  $\text{cl}(\mathcal{A})$  retorna la conclusión de  $\mathcal{A}$ ,

donde  $\text{Args}$  es un conjunto de argumentos y  $\mathcal{L}$  es un lenguaje de sentencias basado en alguna lógica. Por lo tanto  $\text{Args} \in 2^{2^{\mathcal{L}} \times \mathcal{L}}$ .

En este trabajo no se brindará la posibilidad de que un argumento incluya como premisa a la conclusión. La razón detrás de esto es que tal representación no agrega expresividad; aquella información que no sea considerada evidencia ni esté soportada por premisa alguna puede representarse mediante argumentos con un conjunto vacío de premisas. A diferencia de la evidencia, este tipo de conocimiento puede estar sujeto a discusión.

**Definición 2.2 (Interfaz de Argumento)** *Dado un argumento  $\mathcal{A}$  tal que su conjunto de premisas es  $\text{pr}(\mathcal{A}) = \{\beta_1, \dots, \beta_m\}$  y su conclusión es  $\text{cl}(\mathcal{A}) = \alpha$ , la interfaz de  $\mathcal{A}$  es  $\text{int}(\mathcal{A}) = \langle \{\beta_1, \dots, \beta_m\}, \alpha \rangle$ .*

La notación para la interfaz es, por lo tanto, un par; por ejemplo, un argumento  $\mathcal{A}$  con premisas  $a$ ,  $b$  y  $c$ , y conclusión  $s$  tiene una interfaz denotada como  $\text{int}(\mathcal{A}) = \langle \{a, b, c\}, s \rangle$ . En ocasiones utilizaremos sólo la interfaz para referirnos a un argumento. En el caso de  $\mathcal{A}$ , podremos referirnos a él simplemente como  $\langle \{a, b, c\}, s \rangle$ .

Estos atributos de un argumento son similares a los que se encuentran en el sistema argumentativo propuesto por Besnard y Hunter [BH08, BH09]. En este sistema, los argumentos están definidos sobre un lenguaje proposicional. Sin embargo, dado que cada argumento incluye las fórmulas necesarias para derivar la conclusión, una comparación más certera se realizará con las *estructuras argumentales*, en la Sección 2.2.3.

Dado que en el marco de esta tesis los argumentos poseen elementos adicionales a lo tratado en la literatura de argumentación abstracta, la definición de la relación de derrota entre ellos debe ser adaptada en consecuencia. En el DAF, la relación de derrota se definirá sobre pares de argumentos en conflicto, y se utilizará una función de preferencia para decidir cuál derrota al otro. Como fue mencionado previamente, el len-

guaje para expresar conclusiones, premisas y evidencia acepta sentencias complementarias. Dos argumentos  $\mathcal{A}$  y  $\mathcal{B}$  con conclusiones complementarias están automáticamente en contradicción y la relación de conflicto debe incluir al menos este tipo de pares.

**Definición 2.3 (Relación de Conflicto entre Argumentos)**

Dado un conjunto de argumentos  $Args$ , el conjunto  $\bowtie \subseteq Args \times Args$  denota una **relación de conflicto** sobre  $Args$ , verificando  $\bowtie \supseteq \{(\mathcal{A}, \mathcal{B}) \mid \{\mathcal{A}, \mathcal{B}\} \subseteq Args, \text{cl}(\mathcal{A}) = \overline{\text{cl}(\mathcal{B})}\}$ .

Los pares ordenados representando conflictos entre argumentos modelan una relación simétrica, *i.e.*,  $(\mathcal{A}, \mathcal{B}) = (\mathcal{B}, \mathcal{A})$ ; en los ejemplos se indicará sólo uno de estos pares. Según la Definición 2.3 no sólo se considerarán aquellos pares sintácticamente en conflicto, sino que también se permitirá la incorporación de conflictos adicionales, estipulados por el ingeniero de conocimiento. En todo formalismo argumentativo debe definirse un mecanismo para decidir, ante cada conflicto, qué argumento prevalece. Esto se explicará más adelante, en base a la noción de estructura argumental, ya que para decidir qué argumento es más fuerte debe tomarse en cuenta toda la información que lo soporta.

2.1.4. Principios de Coherencia

Ciertas restricciones deben considerarse para que un argumento suponga un paso de razonamiento *coherente*. En particular, el hecho de considerar las piezas de evidencia para evaluar si un dado argumento está activo o no sugiere que un argumento no debería proveer una razón a favor ni en contra de un elemento de evidencia existente, ya que la evidencia no necesita razones que la soporten y es indisputable, respectivamente. Adicionalmente, un argumento coherente propagará estas nociones de

consistencia y no redundancia en relación a su conclusión con respecto a sus propias premisas. Estos cuatro principios se definen formalmente a continuación, dando lugar al concepto de *argumento coherente*.

**Definición 2.4 (Argumento Coherente)** *Un argumento  $\mathcal{A}$  es **coherente** con respecto a un conjunto de evidencia  $\mathbf{E}$  sssi  $\mathcal{A}$  verifica:*

- (**Consistencia Interna**)  $\overline{\text{cl}(\mathcal{A})} \notin \text{pr}(\mathcal{A})$ ;
- (**Consistencia con respecto a  $\mathbf{E}$** )  $\overline{\text{cl}(\mathcal{A})} \notin \mathbf{E}$ ;
- (**No Redundancia Interna**)  $\text{cl}(\mathcal{A}) \notin \text{pr}(\mathcal{A})$ ;
- (**No Redundancia con respecto a  $\mathbf{E}$** )  $\text{cl}(\mathcal{A}) \notin \mathbf{E}$ .

En este marco, impondremos como requerimiento que los argumentos sean coherentes; de esta manera, nos aseguraremos que la evidencia es usada de forma apropiada, y que la relación entre premisas y conclusión es sintácticamente sensata. Al asumir esto, descartaremos aquellos argumentos que no brindan información útil y que, incluso, podrían llevar a la construcción de falacias componiendo cadenas de razonamiento incluyendo conclusiones contrarias a la evidencia o a sus propias premisas. Estas restricciones tienen como principal objetivo detener la ramificación producto de considerar argumentos no coherentes como parte de estructuras argumentales que, de ser aceptadas en principio, deberían ser descartadas más tarde.

## 2.2. Definiendo el DAF

En esta sección presentaremos los elementos que conforman el DAF: la agregación de argumentos en estructuras argumentales, cuáles de ellas se consideran válidas, la noción de conocimiento activo e inactivo, y finalmente la definición del DAF en sí mismo.

### 2.2.1. Evidencia, Premisas y Conclusiones

Al situar un argumento en el contexto de la evidencia disponible, es posible encontrar que sus premisas forman parte de ese conjunto. Cuando esto ocurre, diremos que las premisas están trivialmente satisfechas y el argumento es capaz de llegar a su conclusión directamente desde la evidencia. De cualquier manera, cuando esto no ocurra, el argumento igualmente debería poder recurrir a las conclusiones de otros argumentos para soportar aquellas premisas que no son parte de la evidencia actual. Esto es, recurriría a dichas conclusiones como si se tratase de evidencia. Esta relación entre argumentos será llamada *de soporte*, lo cual se asemeja a lo definido en *marcos de argumentación bipolares* [ACLSL08], aunque en nuestro caso el soporte se da de una manera más explícita, y resulta imprescindible. En un marco bipolar la relación de soporte se utiliza para defender a un argumento de un ataque; en nuestro caso el soporte brinda el conocimiento faltante para que el argumento en cuestión llegue a su conclusión.

**Definición 2.5 (Argumento de Soporte)** *Un argumento  $\mathcal{B}$  es un **argumento de soporte** de un argumento  $\mathcal{A}$  ssi  $\text{cl}(\mathcal{B}) \in \text{pr}(\mathcal{A})$ . Si  $\text{cl}(\mathcal{B}) = \beta$ , entonces se dice que  $\mathcal{B}$  soporta a  $\mathcal{A}$  a través de  $\beta$ .*

### 2.2.2. Argumentos Activos e Inactivos

Intuitivamente, un argumento es considerado *activo* cuando, a la luz de la evidencia disponible, es sensato pensar que respalda completamente su conclusión. En nuestro formalismo, un argumento estará activo siempre que todas sus premisas estén *satisfechas* en función de la evidencia actual. Dado un conjunto de evidencia  $\mathbf{E}$ , una premisa de un argumento está satisfecha si o bien pertenece a  $\mathbf{E}$ , o si es la conclusión de un

argumento activo según  $\mathbf{E}$ . Como fue dicho antes, aquellos argumentos que no encuentren suficiente evidencia para soportar sus premisas pueden recurrir a otros argumentos para tal fin. De todas maneras, es importante notar que, cada vez que un argumento recurre a otro para satisfacer una premisa, un nivel adicional de rebatibilidad es introducido (*i.e.*, la premisa en cuestión puede ponerse en duda), el cual no existiría si la premisa fuese directamente evidencia. Resulta claro que siempre se intentará minimizar esta debilitación al exponer argumentos, discusión que se retomará más adelante. El concepto formal de argumento activo se define a continuación.

**Definición 2.6 (Argumento Activo)** *Dado un conjunto de argumentos  $\text{Args}$  y un conjunto de evidencia  $\mathbf{E}$ , un argumento coherente  $\mathcal{A} \in \text{Args}$  está **activo** con respecto a  $\mathbf{E}$  sssi para cada  $\beta \in \text{pr}(\mathcal{A})$  se cumple una de las siguientes condiciones:*

- $\beta \in \mathbf{E}$ , o
- hay un argumento activo  $\mathcal{B} \in \text{Args}$  con respecto a  $\mathbf{E}$  que soporta a  $\mathcal{A}$  a través de  $\beta$ .

Como será discutido a continuación, esta interacción entre argumentos da lugar a una nueva entidad que comprende todos los argumentos utilizados para la determinación de un argumento como activo. Aquellos argumentos que, dada la evidencia, resultan inactivos representan razones que, aunque válidas, no deben ser tomadas en cuenta debido al contexto actual. Por ejemplo, la presentación de una apelación fuera del período aceptable de tiempo podría tornar inactivo ese argumento. A pesar de que la apelación constituye una razón válida, es incompatible con el estado actual del mundo. En este ejemplo particular la inactividad se da por una condición temporal, pero otras podrían ser consideradas. Un

concepto similar al de argumento inactivo fue dado en [WBC08], bajo el nombre de *inadmissible arguments* (argumentos inadmisibles).

### 2.2.3. Estructuras Argumentales

Cuando un argumento necesita de otros argumentos para poder ser considerado activo, resulta de mucha utilidad reconocer esta agregación de argumentos como una entidad distinguida que emerge del sistema. Más aún, esto nos dará control sobre el tipo de agregaciones que consideraremos *bien formadas*. Nos referiremos a estas entidades como *estructuras argumentales*, y serán árboles vinculando argumentos de conclusiones hacia premisas vía soporte (ver Definición 2.5). Luego definiremos cómo extraer un subárbol para obtener *subestructuras argumentales*, que es un concepto similar al que puede encontrarse en la literatura como *subargumento*. De esta manera, contaremos con una representación de conocimiento que, aún permitiendo un alto grado de complejidad, se mantiene abstracta.

**Definición 2.7 (Estructura Argumental)** *Dado un conjunto de argumentos  $Args$ , una **estructura argumental** para una conclusión  $\alpha$  a partir de  $Args$  es un árbol de argumentos  $\Sigma$  verificando las siguientes propiedades:*

1. *El argumento raíz  $\mathcal{A}_t \in Args$  (denominado **argumento tope**) es tal que  $cl(\mathcal{A}_t) = \alpha$  y se denota como  $top(\Sigma)$ ;*
2. *Un nodo interno es un argumento  $\mathcal{A}_i \in Args$  tal que por cada premisa  $\beta \in pr(\mathcal{A}_i)$  existe a lo sumo un argumento hijo  $\mathcal{A}_k \in Args$  soportando a  $\mathcal{A}_i$  a través de  $\beta$ .*
3. *Una hoja es un argumento  $\mathcal{A}_h \in Args$  tal que no existe argumento en  $Args$  soportándolo.*

Con respecto a la notación para una estructura argumental  $\Sigma$ :

- El conjunto de **premisas** de  $\Sigma$  se define como:  

$$\text{pr}(\Sigma) = \bigcup_{\mathcal{A} \in \text{args}(\Sigma)} (\text{pr}(\mathcal{A})) \setminus \bigcup_{\mathcal{A} \in \text{args}(\Sigma)} (\text{cl}(\mathcal{A})).$$
- La **conclusión** de  $\Sigma$  se denota como  $\text{cl}(\Sigma) = \alpha$ .
- El **conjunto de argumentos** perteneciente a  $\Sigma$  se denota  $\text{args}(\Sigma)$ .

Debe notarse que las funciones  $\text{pr}(\cdot)$  y  $\text{cl}(\cdot)$  están sobrecargadas: ahora se aplican a estructuras argumentales. Esto no resultará confuso, dado que por su uso se podrá determinar unívocamente a qué entidad se está aplicando. Cuando el conjunto de argumentos de una estructura es unitario diremos que es una *estructura argumental primitiva*. A partir de ahora, cuando esté lo suficientemente claro, nos referiremos a estructuras argumentales simplemente como “estructuras”.

Como fue mencionado antes, la noción de estructura argumental tiene cierta semejanza con los argumentos del sistema argumentativo propuesto por Besnard y Hunter [BH08, BH09]. Este sistema se define sobre una base de conocimiento proposicional, y los argumentos son pares  $\langle \Phi, \alpha \rangle$ , donde  $\Phi$  es un conjunto de fórmulas minimal y consistente que deriva la sentencia  $\alpha$ . El conjunto  $\Phi$  es similar al conjunto de argumentos de una estructura en el DAF, y la derivación es análoga al árbol de argumentos donde cada premisa está soportada por otros argumentos. Las propiedades de consistencia y minimalidad se satisfarán con la Definición 2.9 de *estructura argumental bien formada* y el Lema 2.1 de *minimalidad* sobre estructuras argumentales activas. Con respecto a este último aspecto, para poder hablar apropiadamente de *minimalidad* en términos de estructuras argumentales, debemos recurrir a la noción de *estructura argumental activa* (Definición 2.16). De otra manera, una estructura



minimal sería siempre una estructura primitiva (*i.e.*, conteniendo sólo el argumento tope), ya que nada le exigiría incorporar más argumentos. La única razón por la cual un argumento es soportado por otro/s es la no disponibilidad de evidencia suficiente.

La definición de estructura no es suficiente para asegurar una representación de conocimiento sensata. Por ejemplo, permite la existencia de conclusiones contradictorias en argumentos pertenecientes a la misma estructura. Por esto, debe definirse qué se considera una *estructura argumental bien formada*, lo cual requiere la previa formalización del siguiente concepto.

**Definición 2.8 (Soporte Transitivo)** *Dada una secuencia de argumentos  $[\mathcal{A}_1, \dots, \mathcal{A}_n]$ , donde  $\text{cl}(\mathcal{A}_k) \in \text{pr}(\mathcal{A}_{k+1})$  ( $1 \leq k \leq n - 1$ ), se dice que  $\mathcal{A}_1$  soporta transitivamente a  $\mathcal{A}_n$ .*

**Definición 2.9 (Estructura Argumental Bien Formada)**

*Dado un conjunto de argumentos  $\text{Args}$  y una relación de conflicto  $\bowtie \subseteq \text{Args} \times \text{Args}$ , una estructura argumental  $\Sigma$  de  $\text{Args}$  está **bien formada** según “ $\bowtie$ ” sssi  $\Sigma$  verifica las siguientes propiedades:*

- **(Consistencia)** *Para cada argumento  $\mathcal{A} \in \text{args}(\Sigma)$  no existe  $\mathcal{B} \in \text{args}(\Sigma)$  tal que  $\mathcal{A} \bowtie \mathcal{B}$ ;*
- **(No Circularidad)** *Ningún argumento  $\mathcal{A} \in \text{args}(\Sigma)$  soporta transitivamente a un argumento  $\mathcal{B} \in \text{args}(\Sigma)$  cuando  $\text{cl}(\mathcal{B}) \in \text{pr}(\mathcal{A})$ .*
- **(Uniformidad)** *Si  $\mathcal{A} \in \text{args}(\Sigma)$  soporta a  $\mathcal{B} \in \text{args}(\Sigma)$  a través de  $\beta$ , entonces para todo  $\mathcal{B}_i \in \text{args}(\Sigma)$  que tiene a  $\beta$  como premisa,  $\mathcal{A}$  soporta a  $\mathcal{B}_i$  a través de  $\beta$ .*

*El dominio de estructuras argumentales bien formadas a partir de  $\text{Args}$  y “ $\bowtie$ ” se denota  $\text{str}_{(\text{Args}, \bowtie)}$ .*

La propiedad de *consistencia* invalida aquellas estructuras que son inherentemente contradictorias. El requerimiento de *no circularidad* evita tomar en consideración estructuras que podrían constituir una cadena de razonamiento falaz, donde un argumento termina formando parte del soporte de una premisa propia. Finalmente, la restricción de *uniformidad* evita validar aquellas estructuras con soporte heterogéneo de premisas: cada aparición de una misma premisa debe estar soportada por el mismo argumento cada vez. La diversidad de soporte para una misma premisa restaría sensatez al argumento, al mismo tiempo que agregaría puntos extra de ataque. En definitiva, se está incorporando información innecesaria para soportar la conclusión final. Por otra parte, la no uniformidad atenta contra la minimalidad, ya que se utilizan argumentos adicionales para soportar premisas que podrían ser satisfechas sin ellos.

Las restricciones para una estructura bien formada están definidas para poder confiar en ella como una cadena de razonamiento sensata, independientemente del conjunto de evidencia. La consideración de la evidencia formará parte de la noción de estructura argumental activa.

**Ejemplo 2.1** *Los conjuntos de argumentos ilustrados en la Figura 2.1 son estructuras argumentales, pero no están bien formadas<sup>1</sup>:*

- $\Sigma_4$  viola la propiedad de consistencia, dado que  $\mathcal{A}_1$  y  $\mathcal{A}_3$  alcanzan conclusiones contradictorias.
- $\Sigma_5$  viola la propiedad de no circularidad, ya que  $\text{int}(\mathcal{A}_1) = \langle \{a\}, b \rangle$  soporta transitivamente a  $\text{int}(\mathcal{A}_3) = \langle \{c\}, a \rangle$ .
- $\Sigma_6$  y  $\Sigma_7$  violan la propiedad de uniformidad:

<sup>1</sup> En este y otros ejemplos asumiremos un lenguaje proposicional para  $\mathfrak{L}$ , donde la contradicción se establece mediante negación clásica.

- en  $\Sigma_6$ , la premisa  $b$  tiene dos apariciones, pero es soportada por  $\mathcal{A}_1$  en un caso, y es dejada sin soporte alguno en el otro.
- en  $\Sigma_7$ , la premisa  $b$  es soportada por dos argumentos diferentes.

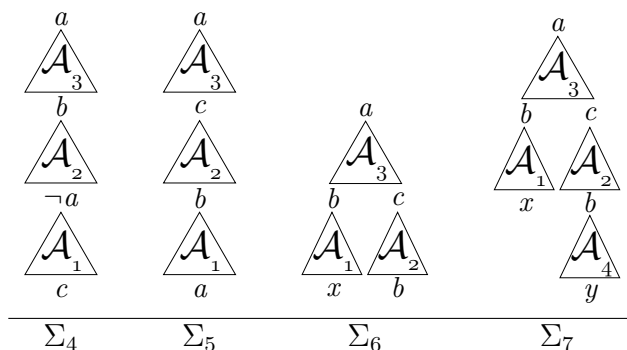


Fig. 2.1: Estructuras argumentales mal formadas

**Ejemplo 2.2** Los conjuntos de argumentos mostrados en la Figura 2.2 componen estructuras argumentales bien formadas:

- La estructura  $\Sigma_8$  es un ejemplo trivial: no hay circularidad, elementos en contradicción ni premisas repetidas.
- La estructura  $\Sigma_9$  muestra un caso similar al de  $\Sigma_7$ . Sin embargo, la premisa  $b$  está soportada siempre por el mismo argumento ( $\mathcal{A}_1$ ) y así verifica la propiedad de uniformidad.

De ahora en adelante, al hacer referencia a una estructura, será asumido que la misma está bien formada, a menos que se establezca lo contrario.

Desde el punto de vista de representación de conocimiento, el objetivo de la estructura argumental es comparable al de cada uno de los

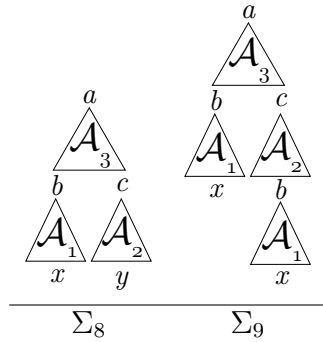


Fig. 2.2: Estructuras argumentales bien formadas

argumentos que la componen: ambos soportan una conclusión a partir de un conjunto de premisas. La diferencia es que un argumento no puede ser descompuesto en fracciones más pequeñas. Si bien una estructura argumental puede descomponerse en partes donde cada una sea uno de los argumentos que forman parte de ella, hay otras agregaciones de argumentos (dentro de la estructura) que resultan de mayor interés. Nos referimos a aquellas porciones de una estructura que conforman una estructura por sí mismas, a las que denominaremos *subestructura argumental*; como lo indica esta noción intuitiva, este concepto es similar al de subargumento mencionado en [MGS07].

**Definición 2.10 (Subestructura Argumental)** *Dadas dos estructuras argumentales  $\Sigma$  y  $\Sigma_i$  construidas a partir de un conjunto de argumentos  $Args$ ,  $\Sigma_i$  es una **subestructura argumental** de  $\Sigma$  (notado  $\Sigma_i \sqsubseteq \Sigma$ ) sssi  $args(\Sigma_i) \subseteq args(\Sigma)$ . Si  $args(\Sigma_i) \subsetneq args(\Sigma)$  entonces  $\Sigma_i$  es una **subestructura argumental propia** de  $\Sigma$ , notado  $\Sigma_i \sqsubset \Sigma$ .*

Debe notarse que, por definición, las subestructuras argumentales también son estructuras: no cualquier subconjunto del conjunto de argumentos de una estructura es una subestructura argumental de ella. Por otra parte, toda estructura es una subestructura (no propia) de sí mis-

ma. En adelante nos referiremos a subestructuras argumentales sólo como “subestructuras”.

**Proposición 2.1** *Todo subárbol de una estructura argumental  $\Sigma$  es una subestructura argumental de  $\Sigma$ .*

DEMOSTRACIÓN: *Trivialmente, por Definición 2.7, un subárbol de  $\Sigma$  respeta las condiciones para ser una estructura argumental.  $\square$*

Toda estructura es un árbol, y en todo árbol existe un subárbol por cada nodo en el mismo, por lo cual se verifica el siguiente corolario.

**Corolario 2.1** *Dada una estructura argumental  $\Sigma$  se verifica:  $|\text{args}(\Sigma)| = |\bigcup_i(\Sigma_i)|$ , donde  $\Sigma_i$  es una subestructura de  $\Sigma$ .*

**Proposición 2.2** *Dados un conjunto de argumentos  $\text{Args}$ , una relación de conflicto  $\bowtie \subseteq \text{Args} \times \text{Args}$ , y una estructura argumental  $\Sigma$  de  $\text{Args}$  bien formada según “ $\bowtie$ ”, toda subestructura argumental  $\Sigma_i \sqsubseteq \Sigma$  es una estructura argumental bien formada según “ $\bowtie$ ”.*

DEMOSTRACIÓN: *Dado que el caso de una subestructura no propia resulta trivial, nos enfocaremos en demostrar que esta propiedad se verifica para una subestructura propia  $\Sigma_i \sqsubset \Sigma$ . Las tres condiciones para que una estructura argumental se considere bien formada son:*

1. *Consistencia: si no había conflictos entre ningún par de argumentos en  $\Sigma$ , tampoco los habrá en  $\Sigma_i$ , ya que los argumentos en  $\Sigma_i$  son un subconjunto de los de  $\Sigma$ ;*
2. *No circularidad: las secuencias de soporte transitivo en  $\Sigma_i$  son subsecuencias de las de  $\Sigma$ . Si en estas últimas no se verifica la condición de circularidad, tampoco se verificará en  $\Sigma_i$ ;*

3. *Uniformidad:* por la Proposición 2.1,  $\Sigma_i$  es un subárbol de la estructura  $\Sigma$ . La propiedad de uniformidad exige que todo subárbol enraizado en una dada premisa sea igual, para todas las apariciones de esa premisa. Supongamos por el absurdo que una premisa  $\beta$  aparece dos veces en  $\Sigma_i$  y los subárboles (i.e., subestructuras)  $T$  y  $T'$  soportando  $\beta$  son diferentes en cada ocasión. Dado que  $\Sigma_i$  es un subárbol de  $\Sigma$ , tanto  $T$  como  $T'$  son subestructuras de  $\Sigma$ , y violan la propiedad de homogeneidad. Esto es absurdo, dado que  $\Sigma$  está bien formada, por hipótesis. El absurdo surgió de suponer que  $\Sigma_i$  no verifica homogeneidad.

Dado que  $\Sigma_i$  es una estructura argumental de  $Args$  que verifica consistencia, no circularidad y homogeneidad,  $\Sigma_i$  está bien formada.  $\square$

En la literatura suele asumirse una relación de derrota estableciendo pares ordenados de argumentos donde la primer componente derrota a la segunda. En el caso del DAF, esta relación de derrota se obtendrá al aplicar una función de preferencia sobre pares de estructuras en conflicto. Como fue mencionado anteriormente, asumiremos que los argumentos componen el nivel de granularidad más fina a nivel de representación de conocimiento para efectuar razonamiento; por lo tanto, resulta sensato definir la relación de conflicto a nivel de argumentos (ver Definición 2.3), y luego propagarla al nivel de estructuras. Esto da lugar a la noción de *conflicto entre estructuras*. Una vez determinados estos conflictos, una función de preferencia se encargará de decidir qué argumento (soportado por la correspondiente estructura argumental) prevalece.

### **Definición 2.11 (Conflicto Entre Estructuras Argumentales)**

Dado un conjunto de argumentos  $Args$ , una relación de conflicto  $\bowtie \subseteq Args \times Args$ , y dos estructuras argumentales  $\Sigma_1, \Sigma_2 \in \mathbf{str}_{(Args, \bowtie)}$ , la es-

estructura  $\Sigma_1$  está en **conflicto** con  $\Sigma_2$  sssi  $\text{top}(\Sigma_1) \bowtie \text{top}(\Sigma_2)$ . El conflicto entre estructuras argumentales se denota como “ $\succ$ ”.

Esta propiedad suele denominarse “herencia de conflicto” y fue mencionada en [MGS07]. Debe notarse que de esta forma se caracteriza cómo se propaga el conflicto desde el nivel básico de argumentos hacia el nivel de estructuras argumentales.

**Definición 2.12 (Función de Preferencia)** Dado un conjunto de argumentos  $Args$  y una relación de conflicto  $\bowtie \subseteq Args \times Args$ , y dos estructuras argumentales  $\Sigma_1, \Sigma_2 \in \mathbf{str}_{(Args, \bowtie)}$ , la **función de preferencia**  $\mathbf{pref} : \mathbf{str}_{(Args, \bowtie)} \times \mathbf{str}_{(Args, \bowtie)} \rightarrow \mathbf{str}_{(Args, \bowtie)} \cup \{\epsilon\}$  sobre estructuras argumentales es tal que  $\mathbf{pref}(\Sigma_1, \Sigma_2) = [\Sigma_1 | \Sigma_2 | \epsilon]$  determina cuál de las dos estructuras se prefiere, de acuerdo a algún criterio; de no preferirse ninguna, se retornará  $\epsilon$ .

La función de preferencia se define sobre estructuras argumentales y no sobre los argumentos en conflicto, ya que, para decidir correctamente cuál de ellas prevalece, debe tomarse en consideración todo el conocimiento que soporta al argumento en cuestión. Dado que el carácter de activo de un argumento se define de acuerdo a la evidencia, en distintos escenarios un mismo argumento puede estar soportado por diferentes estructuras activas. Si la preferencia se estableciera sólo a nivel de argumentos, este cambio en el soporte real del argumento sería invisible al formalismo.

**Ejemplo 2.3** Consideremos las estructuras argumentales ilustradas en la Figura 2.3. Asumamos que la relación de conflicto sólo incluye el par  $(\mathcal{A}_1, \mathcal{A}_3)$  y que la función de preferencia establece que:

$$\mathbf{pref}(\Sigma_1, \Sigma_3) = \Sigma_1 \text{ y } \mathbf{pref}(\Sigma_2, \Sigma_3) = \Sigma_3$$

Si el conjunto de evidencia es  $\{a, b, e\}$ ,  $\Sigma_1$  y  $\Sigma_3$  están activas y  $\Sigma_1$  prevalece. Si la evidencia cambia a  $\{c, e\}$ ,  $\Sigma_1$  estaría inactiva porque  $\mathcal{A}_2$  es redundante según la evidencia,  $\Sigma_2$  se activaría, y  $\Sigma_3$  permanecería activa. En este caso,  $\Sigma_3$  prevalece por sobre  $\Sigma_2$ . Si la relación de preferencia hubiese sido definida a nivel de argumentos, esta situación sería imposible de representar, ya que no podría modelarse que  $\mathcal{A}_1$  es preferido a  $\mathcal{A}_3$  en un momento, y que en otro la relación se invierte.

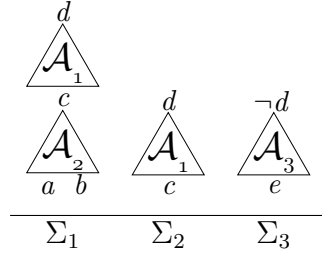


Fig. 2.3: Estructuras argumentales del Ejemplo 2.3

### Definición 2.13 (Derrota Entre Estructuras Argumentales)

Dado un conjunto de argumentos  $Args$  y una relación de conflicto  $\bowtie \subseteq Args \times Args$ , una estructura argumental  $\Sigma_1 \in \mathbf{str}_{(Args, \bowtie)}$  **derrota** a  $\Sigma_2 \in \mathbf{str}_{(Args, \bowtie)}$  sssi existe  $\Sigma_i \sqsubseteq \Sigma_2$  tal que  $\Sigma_1 \succ \Sigma_i$  y  $\mathbf{pref}(\Sigma_1, \Sigma_i) = \Sigma_1$ . La relación de derrota entre estructuras se denota como “ $\Rightarrow$ ”.

Por definición, cuando una estructura derrota a otra, el ataque proviene de la conclusión de la primera y puede estar dirigido a cualquier conclusión de una subestructura de la última. El ataque no apunta hacia un argumento, sino a una subestructura, que es la porción real de la estructura que está bajo ataque.



## 2.2.4. El DAF

El marco argumentativo dinámico nos permitirá distinguir, dentro de un *conjunto de trabajo* de argumentos, aquellos que se encuentran activos. De esta forma, también podremos derivar qué estructuras se encuentran activas y, de la misma forma, sabremos qué conflictos están actualmente vigentes, tanto a nivel de argumentos como de estructuras, para luego decidir estos ataques en términos de la función de preferencia.

**Definición 2.14 (Marco Argumentativo Abstracto Dinámico)**

Un *marco argumentativo abstracto dinámico (DAF)* es una tupla  $\langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , donde  $\mathbf{E}$  es un conjunto de evidencia,  $W$  es el conjunto de trabajo de argumentos,  $\bowtie \subseteq W \times W$  es la relación de conflicto entre argumentos, y  $\text{pref}$  es la función de preferencia definida sobre  $\text{str}_{(W, \bowtie)}$ .

En lo que sigue de este trabajo, para referirnos al conjunto de estructuras bien formadas que pueden construirse a partir de un DAF  $F$  utilizaremos la notación “ $\text{str}_F$ ”, por una cuestión de simplicidad.

Diferentes instancias del conjunto de evidencia determinarán diferentes instancias del DAF, cada una relacionada a una situación particular. De esta forma, cuando se “restringe” un DAF a su conjunto de evidencia  $\mathbf{E}$  asociado, puede obtenerse un AF en el sentido clásico, *i.e.*, un par en el cual todos los argumentos forman parte del proceso de razonamiento, y la derrota se define sobre ellos. Esta “restricción” será denominada *instancia activa* y será definida en la sección siguiente. Dicho esto, el DAF puede verse como una plantilla a partir de la cual surge un gran número de AFs codificando el mismo conocimiento, pero aplicado a situaciones diferentes. La cantidad de situaciones diferentes es  $2^{|\mathfrak{L}|}$ , donde  $\mathfrak{L}$  es el lenguaje utilizado para representar evidencia, premisas y conclusiones.

En la definición del DAF se asume un *conjunto de trabajo de argumentos* que contiene todos aquellos argumentos que se consideran válidos en el sistema. Luego, de acuerdo al conjunto de evidencia efectivamente disponible, algunos argumentos estarán activos o no, sin dejar de constituir razones válidas en soporte de una conclusión. También se permitirá realizar cambios sobre este conjunto de trabajo (ver Sección 2.4), de modo que el marco pueda adaptarse y así representar cambios en la especificación del conocimiento.

Un subconjunto del conjunto de trabajo de argumentos será considerado el *conjunto de argumentos activos* según la evidencia presente. Este conjunto contendrá aquellos argumentos que son tomados en cuenta al realizar razonamiento en la situación actual.

### Definición 2.15 (Conjunto de Argumentos Activos)

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , el *conjunto de argumentos activos* en  $F$  según  $\mathbf{E}$  es  $\mathbb{A} = \{ \mathcal{A} \in W \mid \mathcal{A} \text{ está activo según } \mathbf{E} \}$ .

Una vez calculado el conjunto de argumentos activos, es sencillo obtener aquellos que están inactivos:  $W \setminus \mathbb{A}$ . Este conjunto contiene información de utilidad cuando se desea razonar sobre mundos hipotéticos, *i.e.*, mundos en los que los argumentos ahora inactivos pasan a estar activos. Dado que la activación de nuevos argumentos posiblemente cambie el estado de garantía de otros argumentos, sería deseable contar con herramientas para poder regular la evolución del sistema hasta un estado deseado. Esta discusión será retomada más adelante, en la Sección 5.1.

## 2.3. La Instancia Activa

La instancia activa de un DAF debe contener sólo estructuras activas, concepto definido a continuación.

**Definición 2.16 (Estructura Argumental Activa)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , una estructura argumental bien formada  $\Sigma$  en  $F$  está **activa** según  $\mathbf{E}$  sssi  $\text{pr}(\Sigma) \subseteq \mathbf{E}$  y cada argumento en  $\text{args}(\Sigma)$  es coherente según  $\mathbf{E}$ .

El conjunto de estructuras argumentales activas de un DAF se define análogamente al conjunto de argumentos activos, y se denota como “ $\mathcal{S}$ ”. Esta definición establece una propiedad importante: el soporte de una estructura activa está compuesto por evidencia. Esto acerca las nociones de estructura activa y argumento activo, lo cual demuestra que las estructuras argumentales, con cierto grado de abstracción, se comportan igual que los argumentos. También debe tenerse en cuenta que una definición de estructura activa podría haber sido “toda estructura compuesta por argumentos activos”. Sin embargo, esta noción no captura el sentido que le queremos dar, como se muestra en el siguiente ejemplo.

**Ejemplo 2.4** Consideremos la evidencia  $\mathbf{E}_{2,4} = \{a, b\}$ , y dos estructuras  $\Sigma_1$  y  $\Sigma_2$  tal que  $\text{args}(\Sigma_1) = \{\mathcal{A}_1, \mathcal{A}_2\}$ ,  $\text{args}(\Sigma_2) = \{\mathcal{A}_1\}$ , donde  $\text{int}(\mathcal{A}_1) = \langle \{c\}, d \rangle$ ,  $\text{int}(\mathcal{A}_2) = \langle \{a, b\}, c \rangle$ , como puede verse en la Figura 2.4. Según  $\mathbf{E}_{2,4}$ ,  $\Sigma_1$  está activa, pero  $\Sigma_2$  no. Claramente, ambas estructuras contienen argumentos activos, pero esta condición no asegura que las estructuras lo estén.

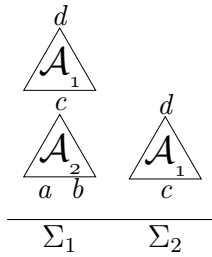


Fig. 2.4: Estructuras argumentales del Ejemplo 2.4

Esto es, el tipo de estructura argumental que queremos que sea considerada como activa debe incluir todos los argumentos que fueron necesarios para su construcción. Toda otra estructura “incompleta” no nos brindaría más información que la ofrecida por cada argumento activo. Por otra parte, cabe destacar que una estructura activa está compuesta sólo por argumentos activos. La relación inversa no siempre se verifica, tal como se acaba de explicar.

**Proposición 2.3** *Si  $\Sigma$  es una estructura argumental activa según un conjunto de evidencia  $\mathbf{E}$ , entonces todo argumento en  $\text{args}(\Sigma)$  está activo según  $\mathbf{E}$ .*

DEMOSTRACIÓN: *Las hipótesis son:  $\text{pr}(\Sigma) \subseteq \mathbf{E}$  y todo argumento en  $\text{args}(\Sigma)$  es coherente.*

*Probaremos este resultado por inducción sobre la longitud  $n$  de la secuencia de soporte entre argumentos. Llamaremos secuencia de soporte de un argumento  $\mathcal{A}_k \in \text{args}(\Sigma)$  a la secuencia de argumentos  $[\mathcal{A}_1, \dots, \mathcal{A}_k]$ , donde  $\mathcal{A}_{1 \leq i \leq k} \in \text{args}(\Sigma_k)$  y  $\Sigma_k \sqsubseteq \Sigma$  es la subestructura argumental cuyo argumento tope es  $\mathcal{A}_k$ .*

Caso base ( $n = 1$ ) *Los argumentos que son hojas del árbol subyacente a  $\Sigma$  no están soportados por ningún argumento y están trivialmente activos, ya que sus premisas están satisfechas por  $\mathbf{E}$ : si  $\mathcal{A}$  es hoja entonces  $\text{cl}(\mathcal{B}) \notin \text{pr}(\mathcal{A})$ , para cualquier  $\mathcal{B} \in \text{args}(\Sigma)$ , y por lo tanto  $\text{pr}(\mathcal{A}) \subseteq \text{pr}(\Sigma) \subseteq \mathbf{E}$ , por definición de conjunto de premisas de una estructura argumental.*

Hipótesis inductiva ( $n \leq k$ ) *Dada la secuencia de soporte  $[\mathcal{A}_1, \dots, \mathcal{A}_k]$ , donde cada premisa de  $\mathcal{A}_i$  está soportada o bien por  $\mathcal{A}_j$  ( $1 \leq j < i \leq k$ ), o por evidencia en  $\mathbf{E}$ . Asumamos que  $\mathcal{A}_k$  está activo.*

Paso inductivo ( $n = k + 1$ ) *Dada la secuencia de  $[\mathcal{A}_1, \dots, \mathcal{A}_{k+1}]$ , cada premisa de  $\mathcal{A}_{k+1}$  está soportada o bien por evidencia en  $\mathbf{E}$ , o por  $\mathcal{A}_j$  ( $j \leq$*

$k$ ). Por hipótesis inductiva tenemos que  $\mathcal{A}_j$  está activo, por lo cual  $\mathcal{A}_{k+1}$  tiene sus premisas satisfechas por evidencia o por argumentos activos; es decir  $\mathcal{A}_{k+1}$  está activo.

Por último, por hipótesis todo argumento  $\Sigma$  es coherente; luego, todo argumento en  $\text{args}(\Sigma)$  está activo según  $\mathbf{E}$ .  $\square$

Más aún, todo argumento activo está respaldado por una estructura argumental, en la cual el argumento en cuestión es el tope de la estructura.

**Proposición 2.4** *Dado un conjunto de evidencia  $\mathbf{E}$ , un argumento  $\mathcal{A}$  está activo según  $\mathbf{E}$  sssi existe al menos una estructura argumental activa  $\Sigma$  según  $\mathbf{E}$  tal que  $\text{top}(\Sigma) = \mathcal{A}$ .*

DEMOSTRACIÓN:

( $\Rightarrow$ ) Si  $\mathcal{A}$  está activo según  $\mathbf{E}$ , cada premisa que no esté en  $\mathbf{E}$  está soportada por un argumento activo. Esto responde a la definición de estructura argumental. Dado que todo argumento activo es coherente, y las premisas no alcanzadas por otro argumento en esta estructura argumental  $\Sigma$  serán evidencia en  $\mathbf{E}$ , las premisas de  $\Sigma$  (la diferencia entre los conjuntos de premisas y conclusiones de todos sus argumentos) estarán incluidas en  $\mathbf{E}$ .

( $\Leftarrow$ ) Por la Proposición 2.3 tenemos que en cada estructura argumental activa, todos sus argumentos están activos, en particular, el argumento tope.  $\square$

Esta proposición establece que para cada argumento activo no necesariamente hay una única estructura argumental activa asociada. Por ejemplo, consideremos las estructuras argumentales ilustradas en la Figura 2.5, y un conjunto de evidencia  $\{a, x\}$ ; es claro que ambas estructuras están activas. Nótese que ambas tienen el mismo argumento tope, pero

el argumento que utilizan para lograr su activación es distinto en cada caso.

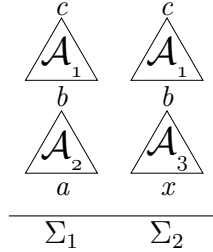


Fig. 2.5: Estructuras argumentales con mismo argumento tope

Por último, las estructuras activas en un DAF involucran a todos los argumentos activos en ese DAF.

**Proposición 2.5** *Dado un DAF  $F$ , el conjunto de argumentos activos  $\mathbb{A}$  en  $F$  según  $\mathbf{E}$  y el conjunto de estructuras argumentales activas  $\mathbb{S}$  en  $F$  según  $\mathbf{E}$ , entonces  $\bigcup_{\Sigma \in \mathbb{S}}(\text{args}(\Sigma)) = \mathbb{A}$ .*

DEMOSTRACIÓN: *Por la Proposición 2.4 se tiene que todo argumento activo es el argumento tope de al menos una estructura argumental activa. Por esto, con sólo unir estos argumentos tope ya tendríamos todo el conjunto de argumentos activos.  $\square$*

Las estructuras argumentales activas son minimales en el sentido que incluyen todos los argumentos necesarios para determinar que el argumento tope está activo, pero ninguno más. Esta propiedad es importante para establecer el carácter no superfluo de los elementos que componen una razón válida para una cierta conclusión. Por otra parte, la incorporación de más argumentos que los necesarios debilitaría innecesariamente la estructura argumental. Dado que la definición para una estructura activa no indica cómo se realiza su construcción, el siguiente resultado demuestra esta afirmación.

**Lema 2.1 (Minimalidad de una Estruct. Argumental Activa)**

Dada una estructura argumental  $\Sigma$  activa según un conjunto de evidencia  $\mathbf{E}$ , no existe otra estructura argumental  $\Sigma_i$  activa según  $\mathbf{E}$  tal que  $\text{args}(\Sigma_i) \subsetneq \text{args}(\Sigma)$ .

DEMOSTRACIÓN: Supongamos por el absurdo que existe una estructura  $\Sigma_i$  activa tal que  $\text{args}(\Sigma_i) \subsetneq \text{args}(\Sigma)$ , i.e.,  $\Sigma_i$  está bien formada, todo argumento en  $\Sigma_i$  es coherente, y  $\text{pr}(\Sigma_i) \subseteq \mathbf{E}$ . Sin pérdida de generalidad, se puede asumir que existe un argumento  $\mathcal{A}$  tal que  $\text{args}(\Sigma) \setminus \text{args}(\Sigma_i) = \{\mathcal{A}\}$ . Dado que  $\Sigma_i$  está activa se verifica que, o bien:

- $\text{cl}(\mathcal{A}) \in \text{pr}(\Sigma_i) \subseteq \mathbf{E}$ , i.e.,  $\text{cl}(\mathcal{A}) \in \mathbf{E}$ . Esto significa que  $\mathcal{A}$  es redundante con respecto a la evidencia,  $\mathcal{A}$  no es un argumento coherente y  $\Sigma$  no puede ser una estructura activa, ya que éstas contienen sólo argumentos coherentes (ver Definición 2.16), o
- existe  $\mathcal{B} \in \text{args}(\Sigma)$  tal que  $\text{cl}(\mathcal{B}) = \text{cl}(\mathcal{A})$ . Esto significa que la inclusión de  $\mathcal{A}$  en  $\Sigma$  con respecto a  $\Sigma_i$  viola la propiedad de uniformidad. Luego,  $\Sigma$  no es una estructura bien formada y no puede estar activa.

Dado que  $\Sigma$  está activa por hipótesis, el absurdo provino de suponer la existencia de  $\Sigma_i$ . Luego,  $\Sigma$  es minimal con respecto a la inclusión de conjuntos sobre  $\text{args}(\Sigma)$ .  $\square$

Para entender la siguiente definición debe recordarse que la derrota entre dos estructuras argumentales se calcula a partir de la relación de conflicto y la función de preferencia, y se denota “ $\Rightarrow$ ” (ver Definición 2.13).

**Definición 2.17 (Derrotas Activas entre Estructuras Argum.)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  y el conjunto  $\mathbb{S}$  de estructuras argu-

mentales activas en  $F$  según  $\mathbf{E}$ , la **relación activa de derrota entre estructuras argumentales** en  $F$  es  $\Rightarrow = \{(\Sigma_1, \Sigma_2) \in \Rightarrow \mid \Sigma_1, \Sigma_2 \in \mathbb{S}\}$ .

**Definición 2.18 (Instancia Activa)** La **instancia activa** de un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  es el AF  $(\mathbb{S}, \Rightarrow)$ , donde  $\mathbb{S}$  es el conjunto de estructuras argumentales activas a partir de  $W$  según  $\mathbf{E}$ , y  $\Rightarrow$  es la relación activa de derrota entre estructuras argumentales en  $F$ .

Un DAF tiene, en todo momento, una instancia activa asociada, la cual es un AF. De esta forma, los resultados existentes en la literatura acerca de aceptabilidad de argumentos y semánticas sobre marcos argumentativos son aplicables al DAF. Más aún, dado que el DAF representa argumentos con mayor nivel de detalle (*i.e.*, premisas y conclusiones), se puede ir un paso más allá y considerar justificación de conclusiones, ya sea a partir de una semántica escéptica u osada.

Como fue mencionado antes, los DAFs pueden ser vistos como una plantilla para generar múltiples AFs representando el mismo conocimiento, pero aplicado a diferentes situaciones. Puede obtenerse una gran cantidad de instancias activas a partir de un DAF: dado que cada subconjunto de evidencia conforma una instancia activa diferente, entonces la cantidad de instancias de un DAF está en el orden de  $2^{|\mathfrak{L}|}$ , donde  $\mathfrak{L}$  es el lenguaje para representar evidencia, premisas y conclusiones.

**Lema 2.2** Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , la instancia activa  $(\mathbb{S}, \Rightarrow)$  de  $F$  es un marco argumentativo abstracto [Dun95].

DEMOSTRACIÓN: Dado que cada estructura argumental representa una razón para una conclusión en base a un conjunto de premisas, éstas siguen la misma noción que los argumentos en un AF. Los ataques entre estructuras siguen el mismo fundamento que los ataques entre argumentos de un AF: denotar una relación de derrota de una estructura



sobre otra. Finalmente, la instancia activa sólo contiene estructuras argumentales y ataques que están activos, i.e., pueden ser utilizados para el razonamiento. Luego, una instancia activa de un DAF es capaz de representar, al menos, el mismo conocimiento que un AF.  $\square$

**Ejemplo 2.5** Consideremos el DAF  $\langle \mathbf{E}_{2,5}, W_{2,5}, \bowtie_{2,5}, \text{pref}_{2,5} \rangle$ , donde:

$$W_{2,5} = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3\};$$

$$\bowtie_{2,5} = \{(\mathcal{A}_2, \mathcal{A}_1), (\mathcal{B}_1, \mathcal{A}_1), (\mathcal{B}_2, \mathcal{B}_1)\};$$

$$\text{pref}_{2,5}(\mathcal{A}_2, \mathcal{A}_1) = \mathcal{A}_2; \text{pref}_{2,5}(\mathcal{B}_1, \mathcal{A}_1) = \mathcal{B}_1; \text{pref}_{2,5}(\mathcal{B}_2, \mathcal{B}_1) = \mathcal{B}_2.$$

Las interfaces de los argumentos y las estructuras argumentales se ilustran en la Figura 2.6. La dirección de las flechas indica la preferencia ante un conflicto.

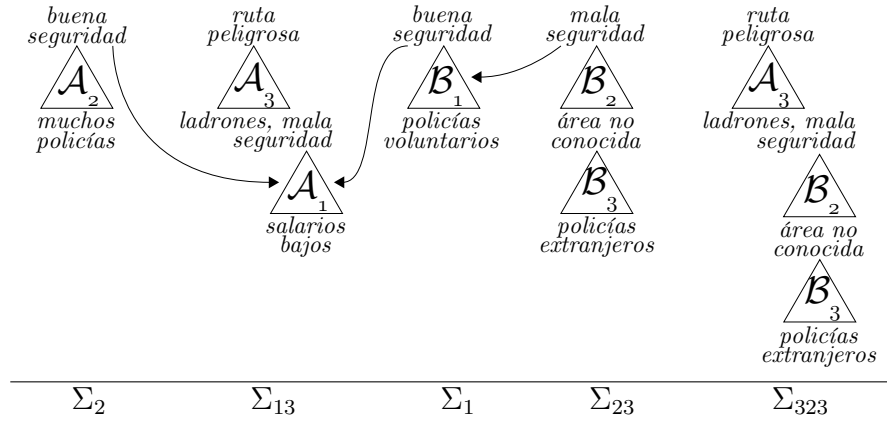


Fig. 2.6: DAF del ejemplo 2.5

La estructura  $\Sigma_2$  especifica que la presencia de muchos policías en un lugar suele ser una señal de buena seguridad. La estructura  $\Sigma_{13}$  establece que una ruta es peligrosa porque hay ladrones allí y la seguridad es mala debido a que la policía del lugar recibe salarios bajos. La estructura  $\Sigma_1$  provee una razón a favor de la buena seguridad en una locación si es que

los policías allí son voluntarios. La estructura  $\Sigma_{23}$  dice que la presencia de policías extranjeros en un lugar lleva a pensar que la seguridad es mala, dado que no conocen el área. La estructura  $\Sigma_{323}$  tiene el mismo argumento tope que  $\Sigma_{13}$ , pero justifica la mala seguridad en un lugar por la razón dada por  $\Sigma_{323}$ .

Dados los conjuntos de evidencia:

$$\mathbf{E}_{2,5,1} = \{\text{muchos policías, ladrones, salarios bajos}\}$$

$$\mathbf{E}_{2,5,2} = \{\text{muchos policías, ladrones, salarios bajos, policías voluntarios}\}$$

$$\mathbf{E}_{2,5,3} = \{\text{ladrones, salarios bajos, policías voluntarios, policías extranj.}\}$$

$$\mathbf{E}_{2,5,4} = \{\text{ladrones, salarios bajos, policías voluntarios}\}$$

tenemos las instancias activas ilustradas en la Figura 2.7, donde los triángulos con vértices redondeados representan estructuras argumentales<sup>2</sup>.

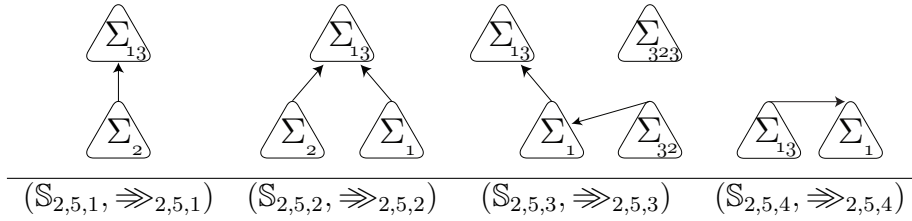


Fig. 2.7: Instancias activas del Ej. 2.5 de acuerdo a distintos conjuntos de evidencia

Si consideramos la semántica grounded, las estructuras argumentales garantizadas en cada situación serían:

- $\Sigma_2$  en  $(\mathbb{S}_{2,5,1}, \Rightarrow_{2,5,1})$ ;
- $\Sigma_2$  y  $\Sigma_1$  en  $(\mathbb{S}_{2,5,2}, \Rightarrow_{2,5,2})$ ;

<sup>2</sup> Esta convención se mantendrá a lo largo de esta tesis.

- $\Sigma_{13}$ ,  $\Sigma_{32}$  y  $\Sigma_{323}$  en  $(\mathbb{S}_{2,5,3}, \Rightarrow_{2,5,3})$ ; y
- $\Sigma_{13}$  en  $(\mathbb{S}_{2,5,4}, \Rightarrow_{2,5,4})$ .

## 2.4. Modificando un DAF

El DAF permite reflejar los cambios que suceden en el mundo dentro de la representación de conocimiento y razonamiento. Esto obliga a proveer herramientas para modificar tanto el conjunto de evidencia, así como el conjunto de trabajo de argumentos y la relación de derrota. De esta forma, la especificación del mundo puede ser cambiada, respondiendo a diversas causas: cambio de preferencias, en las reglas de juego, *etc.*

### Definición 2.19 (Agregado Múltiple de Evidencia)

Dado un DAF  $\langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  definido sobre un dominio  $\mathcal{L}$  y un conjunto de evidencia  $\mathbf{E}_1 \subseteq \mathcal{L}$  tal que todo  $\beta \in \mathbf{E}_1$ ,  $\bar{\beta} \notin \mathbf{E}$ , el resultado de un **agregado múltiple de evidencia** es el DAF  $\langle \mathbf{E} \cup \mathbf{E}_1, W, \bowtie, \text{pref} \rangle$ .

### Definición 2.20 (Borrado Múltiple de Evidencia)

Dado un DAF  $\langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  definido sobre un dominio  $\mathcal{L}$  y un conjunto de evidencia  $\mathbf{E}_1 \subseteq \mathcal{L}$ , el resultado de un **borrado múltiple de evidencia** es el DAF  $\langle \mathbf{E} \setminus \mathbf{E}_1, W, \bowtie, \text{pref} \rangle$ .

El agregado y borrado de evidencia (inspirado de las definiciones originales en [KM91]) cambia la instancia del DAF, haciendo que varíe el conjunto de argumentos activos. De esta forma, el cambio en el conjunto de evidencia podría considerarse una forma de revisión [AGM85]. De todas formas, este tipo de operación de cambio no realiza ni tiene como objetivo la realización de una revisión formal de la teoría. Por otra parte, la actualización de la evidencia no refleja una modificación en la especi-

cación del conocimiento acerca del mundo, sino que es disparada por un cambio en la percepción.

La evolución de un DAF podría ser disparada por un mecanismo externo, tal como un operador de cambio. Un operador de revisión sobre un DAF fue definido en [RMF<sup>+</sup>08b, RMF<sup>+</sup>08a], el cual será explicado en el Capítulo 4. A continuación introduciremos operaciones básicas para modificar un DAF por un conjunto de argumentos y luego la definición análoga para la relación de conflicto.

**Definición 2.21 (Expansión Priorizada por un Conj. de Args.)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , el resultado de la *expansión de F por un conjunto de argumentos*  $\Gamma$  es el DAF  $F + \Gamma = \langle \mathbf{E}, W \cup \Gamma, \bowtie_{\Gamma}, \text{pref}_{\Gamma} \rangle$ , donde la relación de conflicto es:

$$\bowtie_{\Gamma} = \bowtie \cup \{(\mathcal{A}, \mathcal{B}) \mid \{\mathcal{A}, \mathcal{B}\} \cap \Gamma \neq \emptyset, \text{cl}(\mathcal{A}) = \overline{\text{cl}(\mathcal{B})}\},$$

y la función de preferencia  $\text{pref}_{\Gamma} : \text{st}_{F+\Gamma} \times \text{st}_{F+\Gamma} \rightarrow \text{st}_{F+\Gamma} \cup \{\epsilon\}$  es:

$$\text{pref}_{\Gamma}(\Sigma_1, \Sigma_2) = \begin{cases} X & \text{args}(\Sigma_1), \text{args}(\Sigma_2) \in W \text{ y } \text{pref}(\Sigma_1, \Sigma_2) = X \\ \Sigma_1 & \text{args}(\Sigma_1) \in \Gamma \text{ y } \Sigma_2 \in W \\ \Sigma_2 & \text{args}(\Sigma_2) \in \Gamma \text{ y } \Sigma_1 \in W \end{cases}$$

Esta definición de expansión es *priorizada*, es decir, cuando surja un conflicto entre un argumento nuevo (perteneciente a  $\Gamma$ ) y uno de  $W$ , se dará preferencia al nuevo.

**Definición 2.22 (Contracción por un Conjunto de Argumentos)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , el resultado de la *contracción de F por un conjunto de argumentos*  $\Gamma$  es el DAF  $F - \Gamma = \langle \mathbf{E}, W \setminus \Gamma, \bowtie^{-}, \text{pref} \rangle$ , donde  $\bowtie^{-} = \{(\mathcal{B}, \mathcal{C}) \in \bowtie \mid \mathcal{B} \notin \Gamma \text{ y } \mathcal{C} \notin \Gamma\}$ .

La contracción de un DAF por un conjunto de argumentos implica que algunos conflictos ya no serán válidos; más aún, remover los conflictos que involucran al argumento por el cual el DAF está siendo contraído es un requerimiento para que el resultado esté bien definido (recordar que  $\bowtie \subseteq W \times W$ ).

Para completar la definición de operadores básicos de cambio para el DAF, definiremos operadores para modificar la relación de conflicto, lo cual repercutirá en la relación de derrota entre estructuras argumentales.

**Definición 2.23 (Expansión por un Conjunto de Conflictos)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , el resultado de la *expansión de  $F$  por un conjunto de conflictos*  $\bowtie^+$  es el DAF  $F \stackrel{\bowtie^+}{\dashv} \bowtie^+ = \langle \mathbf{E}, W^+, \bowtie \cup \bowtie^+, \text{pref} \rangle$ , donde  $W^+ = W \cup \{ \mathcal{A}, \mathcal{B} \mid (\mathcal{A}, \mathcal{B}) \in \bowtie^+ \}$ .

**Definición 2.24 (Contracción por un Conjunto de Conflictos)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , el resultado de la *contracción de  $F$  por un conjunto de conflictos*  $\bowtie^-$ , donde  $\forall (\mathcal{A}, \mathcal{B}) \in \bowtie^-, \text{cl}(\mathcal{A}) \neq \overline{\text{cl}(\mathcal{B})}$ , es el DAF  $F \stackrel{\bowtie^-}{\dashv} \bowtie^- = \langle \mathbf{E}, W, \bowtie \setminus \bowtie^-, \text{pref} \rangle$ .

Como en el caso de la contracción de argumentos, la expansión por un conflicto requiere corregir la relación de conflicto; en este caso, deben incorporarse a  $W$  los argumentos involucrados en el nuevo conflicto, ya que podrían no estar previamente en  $W$ . También se protegen los pares sintácticamente complementarios de forma tal que la contracción por conflictos no viole la restricción establecida por la Definición 2.3. De esta forma, la expansión por un único conflicto entre dos argumentos puede involucrar una expansión por hasta dos argumentos.

**Definición 2.25 (Contracción por una Conclusión)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  definido sobre un dominio  $\mathcal{L}$ , el resultado de la *con-*

**tracción de  $F$  por una conclusión**  $\alpha \in \mathcal{L}$  es el DAF  $\langle \mathbf{E}, W \setminus \Gamma, \bowtie \setminus \bowtie^{\neg}, \text{pref} \rangle$ , donde  $\Gamma = \{\mathcal{A} \mid \text{cl}(\mathcal{A}) = \alpha\}$  y  $\bowtie^{\neg} = \{(\mathcal{A}, \mathcal{B}) \mid \mathcal{A} \in \Gamma \text{ o } \mathcal{B} \in \Gamma\}$ .

Esta operación es un caso particular de la contracción por un conjunto de argumentos que remueve todo argumento para una conclusión particular; esto podría representar que una conclusión es considerada inválida.

Las operaciones arriba definidas permiten cambiar la especificación del conocimiento codificado en un DAF. La expansión por argumentos permite incorporar razones para una cierta conclusión en respuesta a nuevo conocimiento. Análogamente, la contracción por argumentos permite quitar aquellas razones que ya no son consideradas válidas. La modificación del conjunto de argumentos puede responder a un proceso de aprendizaje o de adaptación a un contexto cambiante.

Es importante remarcar que estos cambios son totalmente independientes de la disponibilidad de evidencia; el conocimiento que se modifica se refiere a la racionalidad detrás de cada argumento. Cuando un argumento deja de ser racional puede ser quitado, ya que su posible activación haría que parte del conocimiento representado pierda sentido. De la misma manera, al agregar un argumento se está expandiendo el conocimiento concerniente a una conclusión en particular, y esto es independiente de que el argumento que se incorpora resulte inmediatamente activo o inactivo: la razón es válida siempre, y en ocasiones la situación actual hará que, además, esté activa. Un ejemplo de cambio en la validez de ciertas razones es un cambio estratégico en un equipo de fútbol de robots cuya arquitectura de agentes esté basada en el DAF; si se opta por otorgarle al equipo un perfil más ofensivo, deberían desecharse algunos argumentos para mantener una formación defensiva, mientras que se incorporarían nuevos para maximizar las opciones de patear al arco.

---

La modificación de la relación de conflicto persigue la misma iniciativa: permitir que la evolución del DAF se adapte al contexto adecuadamente. La aparición de nuevos conflictos y la desaparición de otros (no sintácticos) que por alguna razón pierdan su justificación tiene que estar contemplada. Notar que esto también es independiente de qué conflictos están actualmente activos. En resumen, estas operaciones no implican un cambio en el entorno o mundo, sino que responden a cambios en la conceptualización del mismo, o incluso a un estado de razonamiento hipotético, donde la alteración del DAF es una herramienta para trazar nuevos objetivos.

Estas operaciones básicas de cambio pueden ser compuestas en operaciones más complejas. Por ejemplo, puede definirse la expansión de un DAF por otro DAF a través de la combinación de la expansión por argumentos y conflictos; esto resultaría interesante cuando se desee incorporar argumentos junto con los conflictos. En forma más compleja pueden definirse operaciones de revisión que combinen la modificación de los tres elementos del DAF. Por supuesto, cualquier operación definida debe tener un propósito relevante y concreto. En el Capítulo 4 se profundizará sobre este tema y se definirá un operador de revisión que incorpora un argumento a un DAF realizando las modificaciones necesarias para que el nuevo argumento resulte garantizado.

Por último, definiremos la operación de mezcla de DAFs, que realiza la unión de dos DAFs manteniendo la consistencia del conjunto de evidencia. Una operación similar se define en [CMDK<sup>+</sup>07] donde la mezcla se realiza sobre marcos de Dung, cada uno de los cuales se asocia a un agente. El resultado de la mezcla contiene todos los argumentos propuestos por los agentes, y la relación de derrota depende del tipo de mezcla: puede contener sólo las derrotas aceptadas por todos los agentes,

o simplemente aquellas que los agentes no rechazan. En el operador que definiremos a continuación, las divergencias entre preferencias se resolverán por el carácter priorizado de la operación. Por otra parte, nuestra mezcla es binaria y la definida en [CMDK<sup>+</sup>07] es  $n$ -aria.

En la *mezcla priorizada* de DAFs la prioridad es dada al primer operando: toda evidencia en contradicción resultará en una conversión de evidencia a argumento sobre la pieza de evidencia del segundo operando, y todo conflicto que surja entre un argumento del primer operando con un argumento del segundo será resuelto por la función de preferencia en favor del primero.

La debilitación del conocimiento en conflicto cuando éste representa información indiscutible está inspirado en la operación de mezcla definida en [FKIS02], donde se define una revisión no priorizada para insertar (parcial o totalmente) una dada explicación (una prueba minimal de conocimiento indiscutible) para una sentencia de la base de conocimiento. Si la explicación resulta ser aceptada parcialmente, esto significa que parte del conocimiento indiscutible fue debilitado y transformado a rebatible, y como tal se lo reconoce como un argumento, *i.e.*, una prueba minimal de conocimiento rebatible.

### Definición 2.26 (Mezcla Priorizada de DAFs)

Dados dos DAFs  $F_1 = \langle \mathbf{E}_1, W_1, \bowtie_1, \text{pref}_1 \rangle$  y  $F_2 = \langle \mathbf{E}_2, W_2, \bowtie_2, \text{pref}_2 \rangle$  definidos sobre un dominio  $\mathcal{L}$ , la *mezcla* de  $F_1$  y  $F_2$  se define como  $F_1 \ominus F_2 = \langle \mathbf{E}_\ominus, W_\ominus, \bowtie_\ominus, \text{pref}_\ominus \rangle$ , donde:

- $\mathbf{E}_\ominus = \mathbf{E}_1 \cup \{\beta \in \mathbf{E}_2 \mid \bar{\beta} \notin \mathbf{E}_1\}$ ;
- $W_\ominus = W_1 \cup W_2 \cup W_m$ , donde  $W_m = \{\langle \{\}, \beta \rangle \mid \beta \in \mathbf{E}_2, \bar{\beta} \in \mathbf{E}_1\}$ ;
- $\bowtie_\ominus = \bowtie_1 \cup \bowtie_2 \cup \{(\mathcal{A}, \mathcal{B}) \in W_\ominus \times W_\ominus \mid \text{cl}(\mathcal{A}) = \overline{\text{cl}(\mathcal{B})}\}$ ;



- La función de preferencia es tal que:

$$\text{pref}_{\ominus}(\Sigma_1, \Sigma_2) = \begin{cases} X & \text{args}(\Sigma_1), \text{args}(\Sigma_2) \in W_k, k = 1, 2, \\ & \text{pref}_k(\Sigma_1, \Sigma_2) = X \\ \Sigma_1 & \text{args}(\Sigma_1) \in W_1 \text{ y } \text{args}(\Sigma_2) \in W_2, \\ & \text{o } \text{args}(\Sigma_1) \in W_m \\ \Sigma_2 & \text{args}(\Sigma_2) \in W_1 \text{ y } \text{args}(\Sigma_1) \in W_2, \\ & \text{o } \text{args}(\Sigma_2) \in W_m \\ \epsilon & \text{en otro caso} \end{cases}$$

Al efectuar una mezcla  $F_1 \ominus F_2$ , toda evidencia  $\beta$  en  $\mathbf{E}_2$  que esté en contradicción con evidencia  $\bar{\beta}$  en  $\mathbf{E}_1$  es transformada en un argumento sin premisas y con conclusión  $\beta$ . De esta manera, el resultado de una mezcla asegura un conjunto de evidencia consistente. En cuanto a los conflictos, dado que ambos DAFs manejan el mismo lenguaje para premisas y conclusiones, es posible que surjan nuevos conflictos por la unión de argumentos y la incorporación de piezas de evidencia convertidas en argumentos. En la mezcla, estos nuevos conflictos se unen directamente.

En el caso de aquellos pares de estructuras pertenecientes al mismo marco, la función de preferencia responde como las funciones del DAF correspondiente. En otro caso, las estructuras pertenecientes a  $F_1$  serán preferidas a cualquiera de  $F_2$ , ya que en esta operación de mezcla en particular se le da prioridad al primer operando. Asimismo, las estructuras (primitivas) compuestas por uno de los argumentos en  $W_m$  siempre serán preferidas a cualquier otra estructura que esté en conflicto con ellas (ya que los argumentos en  $W_m$  eran evidencia). Nótese que nunca se dará el caso de dos estructuras de  $W_m$  en conflicto, dado que el conjunto de evi-

dencia en  $F_2$  debe ser consistente. Por último, las estructuras “mixtas”, *i.e.*, con argumentos de distintos conjuntos universales, no reportarán preferencia alguna.

Existe la alternativa de definir una mezcla sin prioridad alguna: las piezas de evidencia en contradicción se convertirían en argumentos, y la función de preferencia retornaría  $\epsilon$ , indicando que ninguno de estos argumentos derrota al otro.

Para finalizar, cabe aclarar que aquellos nuevos argumentos surgidos de la evidencia inconsistente estarán inactivos al momento de la mezcla, ya que no serán coherentes. Estos argumentos sólo formarán parte de la instancia activa de un DAF cuando dejen de ser inconsistentes con respecto a la evidencia.

**Ejemplo 2.6** Consideremos los DAFs ilustrados en la Figura 2.8, debajo de los cuales se muestran los respectivos conjuntos de evidencia, con fondo gris.

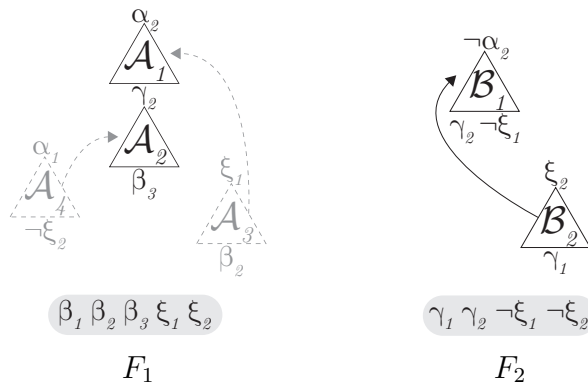


Fig. 2.8: Dos DAFs con sus respectivos conjuntos de evidencia

En  $F_1$ , los argumentos  $\mathcal{A}_1$  y  $\mathcal{A}_2$  están activos, ya que  $\mathcal{A}_3$  es redundante con respecto a la evidencia y  $\mathcal{A}_4$  no tiene suficiente evidencia para estar activo. En  $F_2$ , tanto  $\mathcal{B}_1$  como  $\mathcal{B}_2$  están activos. Si realizamos la mezcla

$F_1 \ominus F_2$  obtenemos el DAF ilustrado en la Figura 2.9.

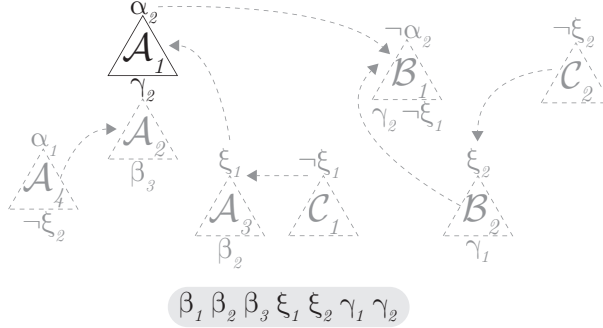


Fig. 2.9: Resultado de la mezcla de los DAFs en la Figura 2.8

En la mezcla,  $A_2$  queda inactivo porque es redundante con respecto a  $\gamma_2$ , el caso de  $B_2$  es análogo con respecto a  $\xi_2$ , y  $B_1$  ahora carece de evidencia suficiente para estar activo. Los dos argumentos nuevos ( $C_1$  y  $C_2$ ) que se generan a partir de la evidencia descartada quedan instantáneamente inactivos por contradecir la evidencia. Nótese cómo se genera un nuevo conflicto entre argumentos en ambos DAFs, y cómo la función de preferencia decide en favor de las estructuras argumentales en  $F_1$  por sobre las de  $F_2$  ( $A_1$  derrota a  $B_1$ ), así como da preferencia a  $C_1$  y  $C_2$ , ya que son estructuras argumentales compuestas por evidencia convertida en argumento.

## 2.5. Semánticas Argumentativas y el DAF

Como fue explicado en la Sección 1.3, la noción de semántica en marcos argumentativos abstractos ha sido estudiada con profundidad en la literatura, principalmente por Baroni y Giacomin [BG07]. Se han definido varios criterios para determinar bajo qué condiciones un argumento debe considerarse aceptado (*i.e.*, garantizado), incluyendo variantes crédulas (como la semántica preferida) y escépticas (como la *grounded*). Por lo

tanto, en esta sección haremos un estudio de semánticas sobre el DAF relacionando argumentos, estructuras y subestructuras. No sólo abordaremos este enfoque involucrando el cómputo de la instancia activa, sino que también definiremos herramientas para calcular semántica directamente en el DAF. Primeramente abordaremos el enfoque de árboles de dialéctica para determinar garantía, y luego estudiaremos la utilización de grafos de argumentos.

### 2.5.1. Argumentación y Árboles de Dialéctica

El enfoque argumentativo basado en árboles de dialéctica responde a la intuición de una discusión en torno a un tema. Es decir, la estructura argumental en la raíz del árbol está planteando no sólo una razón a favor de una sentencia  $\alpha$ , sino también un tema de discusión. Debido a esto, se discutirá acerca de  $\alpha$ , de su negación  $\neg\alpha$  y, adicionalmente, de aquellas temáticas traídas a colación por las estructuras argumentales soportando estas sentencias, ya que éstas pueden ser, a su vez, atacadas en una conclusión interna (ver Definición 2.13). A continuación se describen todas las nociones básicas de este intercambio de argumentos para dar lugar a la definición de *árbol de dialéctica*.

La confrontación de razones en conflicto en un sistema argumentativo como el DAF se asemeja a una discusión entre dos partes. En la literatura esto suele definirse como una *línea de argumentación*.

**Definición 2.27 (Línea de Argumentación)** *Dado un DAF  $F$ , una línea de argumentación es una secuencia  $[\Sigma_1, \dots, \Sigma_n]$ , donde cada  $\Sigma_i$  ( $1 \leq i \leq n$ ) es una estructura argumental bien formada de  $F$  que ataca a su antecesor. La **raíz** de  $\lambda$  es  $\Sigma_1$  y la **hoja** de  $\lambda$  es  $\Sigma_n$ .*

El siguiente concepto distingue, en una línea de argumentación, el

camino desde la raíz hasta un argumento. Esto será de utilidad para la definición de *árbol de dialéctica* (Definición 2.32).

**Definición 2.28 (Segmento Superior)** *Dada una línea de argumentación  $\lambda = [\Sigma_1, \dots, \Sigma_n]$ , el **segmento superior** de  $\Sigma_i$  ( $1 < i \leq n$ ) en  $\lambda$  es  $[\Sigma_1, \dots, \Sigma_i]$  y se nota  $\lambda^\uparrow(\Sigma_i)$ . El **segmento superior propio** de  $\Sigma_i$  en  $\lambda$  es  $[\Sigma_1, \dots, \Sigma_{i-1}]$  y se nota  $\lambda^\uparrow[\Sigma_i]$ .*

Como se mencionó antes, el intercambio de argumentos se asemeja a una discusión entre pares. Como tal, es sensato suponer que la introducción de un argumento por parte de cada lado debería ser consistente con los argumentos previamente presentados. Más aún, también resulta deseable que ninguna de las partes repita un argumento ya presentado. Llamaremos *pro* y *con*, respectivamente, a cada una de las partes que realiza la argumentación. Las definiciones dadas a continuación preceden a la formalización de las intuiciones referentes a una *línea de argumentación aceptable*.

**Definición 2.29 (Conjunto Argumental Consistente)**

*Dado un conjunto  $S$  de estructuras y una relación de conflicto  $\bowtie \subseteq S \times S$ , el conjunto  $S$  se dice **consistente** con respecto a  $\bowtie$  sssi no hay  $\{\Sigma_1, \Sigma_2\} \subseteq S$  tal que  $\Sigma_1 \bowtie \Sigma_2$ .*

**Definición 2.30 (Conjunto de Estruct. Argumentales *pro/con*)**

*Dada una línea de argumentación  $\lambda = [\Sigma_1, \dots, \Sigma_n]$ , el **conjunto de estructuras argumentales** *pro* (respectivamente, *con*) está compuesto por todas las  $\Sigma_i$  ( $1 \leq i \leq n$ ), con  $i$  impar (respectivamente, par). El conjunto de estructuras *con* en  $\lambda$  se denota  $\lambda^-$ .*

**Definición 2.31 (Línea de Argumentación Aceptable)**

*Dada una línea de argumentación  $\lambda$  en el contexto de un DAF  $F =$*

$\langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ ,  $\lambda$  es **aceptable** en  $F$  sssi se verifican:

- (**No circularidad**) No hay repetición de estructuras en  $\lambda$ , y
- (**Concordancia**) Los conjuntos pro y con son consistentes con respecto a  $\bowtie$ .

Las propiedades de *no circularidad* y *concordancia* se denominan **restricciones dialécticas**. Cabe mencionar que estas restricciones podrían ser diferentes, e incluso no existir. Ésta es una de las definiciones posibles para que una línea de argumentación sea considerada aceptable.

Una línea de argumentación aceptable se dice *exhaustiva* si no es posible insertar más estructuras argumentales en la secuencia.

**Definición 2.32 (Árbol de Dialéctica)** Dado un DAF  $F$  y un conjunto  $S$  de líneas exhaustivas de  $F$  enraizadas en  $\Sigma_1$  tal que  $S$  es maximal con respecto a la inclusión de conjuntos, un **árbol de dialéctica** para una estructura argumental  $\Sigma_1$  es un árbol  $\mathcal{T}_F(\Sigma_1)$  verificando:

- $\Sigma_1$  es la **raíz**;
- una estructura argumental  $\Sigma_{i \neq 1}$  en una línea  $\lambda_i \in S$  es un **nodo interno** sssi tiene como hijos todos aquellos  $\Sigma_j$  en líneas  $\lambda_j \in S$  tal que  $\Sigma_j \Rightarrow \Sigma_i$  y  $\lambda_i^\uparrow[\Sigma_i] = \lambda_j^\uparrow(\Sigma_j)$ ;
- las **hojas** son las hojas de las líneas en  $S$ .

Debe notarse que los árboles de dialéctica están definidos sobre el conjunto de trabajo de argumentos, pudiendo contener estructuras argumentales tanto activas como inactivas. Un árbol de dialéctica conteniendo sólo estructuras activas se denomina **árbol de dialéctica activo**, y se denotará  $\mathbb{T}_F(\Sigma)$ . En este sentido, para enfatizar esta diferencia, un árbol que contenga estructuras inactivas será denominado *árbol de dialéctica*

*potencial*. Este árbol potencial es importante para vislumbrar todas aquellas estructuras que podrían formar parte de un eventual árbol activo.

**Ejemplo 2.7** *Supongamos un DAF  $\langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , donde  $\bowtie = \{(\mathcal{A}, \mathcal{X})\}$ ,  $\text{pref}(\mathcal{A}, \mathcal{X}) = \mathcal{A}$  y  $W = \{\mathcal{X}, \mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}\}$ . Las interfaces de los argumentos son:  $\text{int}(\mathcal{A}) = \langle \{b, c\}, \neg a \rangle$ ,  $\text{int}(\mathcal{B}) = \langle \{d, e\}, b \rangle$ ,  $\text{int}(\mathcal{C}) = \langle \{f\}, c \rangle$ ,  $\text{int}(\mathcal{D}) = \langle \{g, h\}, d \rangle$ ,  $\text{int}(\mathcal{X}) = \langle \{y, z\}, a \rangle$ . El árbol potencial para la estructura primitiva compuesta por  $\mathcal{X}$  es el que se muestra en la Figura 2.10.*

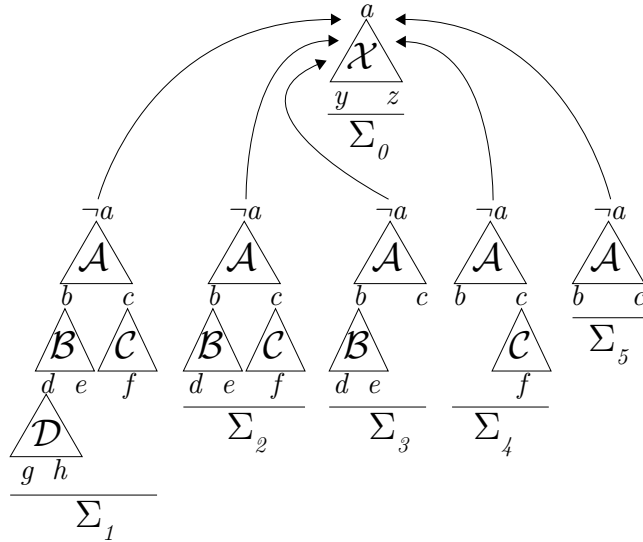


Fig. 2.10: Árbol potencial del Ejemplo 2.7

*Notar que todas las posibles estructuras para  $\neg a$  son construidas. Sin embargo, a la luz de cierto conjunto de evidencia, algunas de estas estructuras estarán inactivas, ya sea por falta de evidencia necesaria, o por ser redundantes o inconsistentes con respecto a la evidencia.*

*Si  $\mathbf{E} = \{g, h, f, y, z\}$ , sólo  $\Sigma_0$  y  $\Sigma_1$  están activas, y el árbol activo es de la Figura 2.11(a). Por otra parte, si el conjunto de evidencia fuera  $\mathbf{E} = \{g, h, f, d, e, y, z\}$ , el argumento  $\mathcal{D}$  resultaría redundante con respecto*

a la pieza de evidencia  $d$ ; por lo tanto,  $\Sigma_1$  no estaría activa, y sí lo estaría  $\Sigma_2$ . El árbol activo estaría compuesto por  $\Sigma_0$  y  $\Sigma_2$ , tal como se muestra en la Figura 2.11(b). Una situación similar surge con  $\Sigma_4$  y  $\Sigma_5$  si  $\mathbf{E} = \{b, c, f, y, z\}$ , en cuyo caso sólo  $\Sigma_0$  y  $\Sigma_5$  estarían activas y el árbol activo es el de la Figura 2.11(c).

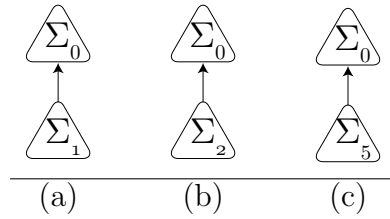


Fig. 2.11: Árboles activos del árbol potencial de la Figura 2.10

Una vez que el árbol para una estructura fue construido, un *criterio de marcado* determinará qué estructuras dentro del árbol están derrotadas y cuáles no. Este criterio se especifica mediante una *función de marcado*. Aunque podrían asignarse otros valores para marcar los nodos del árbol, en esta tesis asumiremos sólo estos dos; es decir, una estructura estará marcada como “derrotada” o “no derrotada”. En [RMS09] se dan postulados para que una función de marcado arbitraria sea considerada sensata; asimismo, se discuten ciertas opciones que difieren del marcado aquí propuesto, tal como asignar un *grado de derrota* a cada argumento, *i.e.*, un número real entre 0 y 1.

**Definición 2.33 (Función de Marcado)** Dado un DAF  $F$  y una estructura argumental  $\Sigma_i$  en una línea  $\lambda_i$  en un árbol de dialéctica  $\mathcal{T}_\Sigma(F)$ , una **función de marcado** es tal que  $\mathbf{m}(\Sigma_i, \lambda_i, \mathcal{T}_\Sigma(F)) = [\mathbf{U}|\mathbf{D}]$ , donde  $\mathbf{U}$  representa “no derrotada” y  $\mathbf{D}$ , “derrotada”.

Una elección sensata para esta función de marcado es tomar la definida en DELP, la cual hace una evaluación de carácter escéptico, *i.e.*, una



estructura argumental es considerada como no derrotada sólo cuando todos sus derrotadores hayan sido derrotados.

**Definición 2.34 (Función de Marcado Escéptica)**

Dado un árbol de dialéctica  $\mathcal{T}_F(\Sigma)$ , la función de marcado escéptica  $\mathbf{m}_e$  se define de la siguiente manera:

$$\mathbf{m}_e(\Sigma_i, \lambda_i, \mathcal{T}_F(\Sigma)) = \text{D} \text{ sssi } (\exists \Sigma_j) \mathbf{m}_e(\Sigma_j, \lambda_j, \mathcal{T}_F(\Sigma)) = \text{U},$$

donde  $\Sigma_j$  es un hijo de  $\Sigma_i$  en  $\mathcal{T}_F(\Sigma)$

Una vez que la función de marcado está definida, se puede determinar el *estado de garantía* de la estructura raíz del árbol de dialéctica; formalmente:

**Definición 2.35 (Garantía)** Dado un DAF  $F$  y una función de marcado  $\mathbf{m}$ , una estructura argumental  $\Sigma$  de  $F$  está **garantizada** en  $F$  sssi  $\mathbf{m}(\Sigma, \lambda, \mathbb{T}_F(\Sigma)) = \text{U}$ , donde  $\lambda$  es cualquier línea de  $\mathbb{T}_F(\Sigma)$ . La conclusión  $\text{cl}(\Sigma)$  se dice **justificada** por  $\Sigma$ .

En el caso de considerar una función de marcado diferente, la definición de garantía cambiará consecuentemente. Por ejemplo, si consideramos un grado de derrota para cada estructura y la marca asignada a cada una está entre 0 y 1, la función de garantía podría establecer que una estructura está garantizada cuando es la raíz de un árbol activo de dialéctica y su marca sobrepasa un umbral, (e.g., 0,7). Las estructuras argumentales garantizadas (en la literatura también se suele utilizar el término “aceptadas”) serán determinadas por esta definición de garantía, dando lugar a la noción de semántica sobre el marco argumentativo dinámico. Aquellas estructuras que no estén garantizadas se denominarán *rechazadas*. Debe notarse que la noción de garantía está definida sobre árboles activos, ya que todo razonamiento debe ser realizado

sobre argumentos activos. Por último, el argumento tope de una estructura argumental garantizada también se lo denomina *garantizado*.

Es importante considerar el concepto de consistencia sobre los argumentos aceptados, ya que la finalidad de un marco argumentativo es justamente resolver la inconsistencia. En cuanto al conjunto de estructuras rechazadas, en este sentido no es necesario imponer restricción alguna.

**Definición 2.36 (Garantía Consistente)**

*Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , una función de marcado  $\mathbf{m}$  provee una **garantía consistente** sssi no existen dos estructuras argumentales  $\Sigma_1, \Sigma_2 \in \text{str}_F$  garantizadas de acuerdo a  $\mathbf{m}$  tal que  $\Sigma_1 \bowtie \Sigma_2$ .*

### 2.5.2. Grafos de Argumentos

Dada una semántica de Dung y un DAF, se puede computar la instancia activa asociada a él (equivalente a un AF) y entonces calcular el conjunto de argumentos aceptados de acuerdo a esa semántica. Luego, provisto que los argumentos en la instancia activa se traducen en estructuras argumentales activas en el DAF, podremos saber qué estructuras están garantizadas. Mientras la semántica utilizada mantenga la libertad de conflicto (ver Sección 1.3), la garantía será consistente.

En lo que sigue, a modo de simplificación, se asumirán dos funciones  $\text{skep}(\cdot)$  y  $\text{cred}(\cdot)$  que retornarán las sentencias garantizadas siguiendo una semántica escéptica y crédula, respectivamente. En el primer caso, se retornará un conjunto de estructuras representando una única extensión (que puede ser vacía), mientras que en el segundo se retornará un conjunto de conjuntos de estructuras representando cada extensión crédula del marco. En ambos casos diremos que las funciones retornan la semántica escéptica o crédula del DAF en cuestión.

### 3. DAF VS. SISTEMAS BASADOS EN REGLAS

En este capítulo presentaremos dos marcos argumentativos basados en reglas: Programación en Lógica Rebatible [GS04] (DELP, la sigla de *defeasible logic programming*) y el sistema ASPIC [Pro04], y los compararemos con el enfoque abstracto dado por el DAF. El objetivo es mostrar hasta qué punto el DAF es una generalización de este tipo de sistemas, e incluso qué características deberían agregársele para lograr una abstracción más completa.

En los sistemas basados en reglas los argumentos se expresan mediante conjuntos de reglas satisfaciendo ciertas propiedades, tales como minimalidad y consistencia. El conflicto entre ellos, por su parte, se determina en base a los literales derivados por las reglas dentro de los argumentos. En particular, los dos sistemas que repasaremos en esta sección codifican dos tipos de conocimiento: estricto y rebatible. El primer tipo de conocimiento representa conocimiento irrefutable, mientras que el segundo sólo se utiliza para establecer las condiciones que brindan una razón para creer en cierta sentencia. Otros sistemas usan todo el cuerpo de conocimiento para construir argumentos, sin hacer distinción alguna, por lo cual puede considerarse que todo el conocimiento es rebatible. Un ejemplo de este tipo de sistemas es el de Besnard y Hunter [BH08].

#### 3.1. Programación en Lógica Rebatible

DELP combina resultados de *programación en lógica y argumenta-*

*ción rebatible*<sup>1</sup>. Este formalismo ha sido extendido para utilizarse como servidor de razonamiento en [GRTS07], y se le han añadido capacidades de representación de explicaciones [GRS07]. Un *programa lógico rebatible* (*de.l.p.*)  $\mathbf{p}$  consiste de un hechos, reglas estrictas y rebatibles. Los hechos son literales fijos representando información atómica (o su negación). Las *reglas estrictas* representan información indiscutible y se denotan como  $\alpha \leftarrow \beta_1, \dots, \beta_n$ , donde  $\alpha$  es un literal fijo y cada  $\beta_{1 \leq i \leq n}$  también. Una regla estricta se asume con cuerpo no vacío; la consideración de un cuerpo vacío se cubre con los hechos. Las *reglas rebatibles* representan información tentativa y se denotan como  $\alpha \multimap \beta_1, \dots, \beta_n$ . En este caso, las reglas rebatibles pueden tener cuerpo vacío, en cuyo caso se las denomina *pre-suposiciones*.

Los *de.l.p.* se denotan como  $(\Pi, \Delta)$ , donde  $\Pi$  es el conjunto de hechos y reglas estrictas, y  $\Delta$  es el conjunto de reglas rebatibles (ver Ejemplo 3.1). A partir de un programa se pueden derivar literales contradictorios usando ambos tipos de reglas. De todas maneras, el conjunto  $\Pi$  debe mantenerse consistente para dotar de coherencia a la representación de conocimiento. En DELP se pueden derivar literales a partir de reglas, tal como en programación en lógica, siendo una *derivación rebatible* (denotada como “ $\sim$ ”) una que utiliza, al menos, una regla rebatible. Por último, una *consulta* es un literal presentado ante un *de.l.p.* para averiguar si está o no garantizado.

**Ejemplo 3.1** Consideremos el *de.l.p.*  $\mathbf{p}_{3,1} = (\Pi_{3,1}, \Delta_{3,1})$ :

$$\Pi_{3,1} = \left\{ \begin{array}{l} t, z, \\ (p \leftarrow t) \end{array} \right\} \Delta_{3,1} = \left\{ \begin{array}{l} (\neg a \multimap y), (y \multimap x), (x \multimap z), \\ (y \multimap p), (a \multimap w), (w \multimap y), \\ (\neg w \multimap t), (\neg x \multimap t), (x \multimap p) \end{array} \right\}$$

<sup>1</sup> El sistema está totalmente implementado y está disponible en Internet [LID07].

A partir de un programa es posible derivar literales contradictorios (*e.g.*, los literales  $\neg a$  y  $a$  en el Ejemplo 3.1). DELP incorpora un formalismo de argumentación rebatible para el tratamiento de conocimiento contradictorio. Este formalismo permite identificar estas piezas de conocimiento, y un *proceso dialéctico* se utiliza para decidir qué información prevalece como garantizada. Este proceso dialéctico involucra la construcción y evaluación de argumentos que soportan o interfieren con la consulta bajo análisis.

**Definición 3.1 (Estructura de Argumento)**

Sea  $\mathfrak{p} = (\Pi, \Delta)$  un de.l.p.,  $\langle A, \alpha \rangle$  es una estructura de argumento para el literal  $\alpha$  a partir de  $\mathfrak{p}$ , si  $A$  es el conjunto minimal (según  $\subseteq$ ) de reglas rebatibles de  $\Delta$  tal que:

1.  $\Pi \cup A \vdash \alpha$ , y
2. el conjunto  $\Pi \cup A$  no es contradictorio.

Por simplicidad, usualmente nos referiremos a un argumento  $\langle A, \alpha \rangle$  simplemente como  $A$ , aunque ciertamente un mismo conjunto de reglas rebatibles puede dar soporte a distintas conclusiones, en virtud del uso de reglas estrictas.

**Ejemplo 3.2** *A partir del programa del Ejemplo 3.1, se pueden construir los siguientes argumentos:*

$$\langle B_1, \neg a \rangle = \langle \{\neg a \rightarrow y, y \rightarrow x, x \rightarrow z\}, \neg a \rangle$$

$$\langle B_2, \neg a \rangle = \langle \{\neg a \rightarrow y, y \rightarrow p\}, \neg a \rangle$$

$$\langle B_3, a \rangle = \langle \{a \rightarrow w, w \rightarrow y, y \rightarrow p\}, a \rangle$$

$$\langle B_4, \neg w \rangle = \langle \{\neg w \rightarrow t\}, \neg w \rangle$$

$$\langle B_5, \neg x \rangle = \langle \{\neg x \rightarrow t\}, \neg x \rangle$$

$$\langle B_6, x \rangle = \langle \{x \rightarrow p\}, x \rangle$$

Un literal  $\alpha$  está garantizado si existe un argumento  $A$  para  $\alpha$  que se considera no derrotado. Para establecer si  $\langle A, \alpha \rangle$  es un argumento no derrotado, se consideran los *derrotadores* para  $\langle A, \alpha \rangle$ , *i.e.*, contra-argumentos que por algún criterio se prefieren a  $\langle A, \alpha \rangle$ . Un argumento  $A_1$  es un contra-argumento para  $A_2$  sssi  $A_1 \cup A_2 \cup \Pi$  resulta contradictorio. Cuando  $A_1$  sea más fuerte (según el criterio elegido) que  $A_2$ , el derrotador se dice *propio*; cuando el criterio no puede decidir, ambos argumentos son derrotadores *por bloqueo* entre sí. En DELP, el criterio de comparación suele ser *especificidad generalizada* [SGCS03, DS01], pero en los ejemplos dados en esta sección nos abstraeremos del criterio para no introducir complicaciones innecesarias. En su lugar, enumeraremos explícitamente las derrotas entre argumentos.

Al momento de establecer los derrotadores de un argumento, estos pueden estar, a su vez, derrotados, y así siguiendo, hasta formar cadenas de derrotadores. Estas cadenas se denominarán *líneas de argumentación*. Las definiciones dadas a continuación para formalizar el concepto de árbol de dialéctica son similares a las dadas para definir el mismo concepto en base al DAF en la Sección 2.5.1.

**Definición 3.2 (Línea de Argumentación)** *Dado un de.l.p.  $\mathfrak{p}$ , y los argumentos  $B_1, \dots, B_n$  en  $\mathfrak{p}$ , una **línea de argumentación**  $\lambda$  es cualquier secuencia finita  $[B_1, \dots, B_n]$  tal que  $B_i$  es un derrotador para  $B_{i-1}$ , con  $1 < i \leq n$ . Diremos que  $\lambda$  está **enraizada** en  $B_1$ , y que  $B_n$  es la **hoja** of  $\lambda$ .*

Como las líneas de argumentación son un intercambio de argumentos en oposición, podríamos pensar en ellas como representando una discusión entre dos partes, que llamaremos *pro* y *con*.

**Definición 3.3 (Conjunto de Argumentos *pro/con*)** Dada una línea de argumentación  $\lambda$ , el conjunto de argumentos **con** (resp., **pro**)  $\lambda^-$  (resp.,  $\lambda^+$ ) de  $\lambda$  es el conjunto conteniendo todos los argumentos ubicados en posiciones **impares** (resp., **pares**) en  $\lambda$ .

Para evitar secuencias de argumentos indeseables que pueden representar cadenas de razonamiento falaces, en DELP una línea de argumentación tiene que ser **aceptable**: tiene que ser finita, un argumento no puede aparecer dos veces, se prohíbe que haya dos derrotas por bloqueo consecutivamente, y los conjuntos *pro* y *con* tienen que ser internamente consistentes. Una línea aceptable de un programa  $\mathfrak{p}$  se denomina *exhaustiva* si es imposible agregar más argumentos de  $\mathfrak{p}$  a la secuencia. Para más detalles, el lector deberá referirse a [GS04].

**Ejemplo 3.3** Considerando los argumentos del Ejemplo 3.2, y las siguientes relaciones de derrota:  $B_3$  es un derrotador propio de  $B_2$ , y  $B_4$  es un derrotador propio de  $B_3$ . Podemos construir la secuencia  $[B_2, B_3, B_4]$ , que es una línea de argumentación aceptable: no es circular, es finita, no contiene dos bloqueos consecutivos, y  $B_2 \cup B_4 \cup \Pi_{3,1}$  es no contradictorio.

Muchas líneas de argumentación podrían aparecer para un argumento, dando lugar a una estructura de árbol. En un *árbol de dialéctica*, cada nodo (excepto la raíz) representa un derrotador de su padre, y las hojas se corresponden a argumentos no derrotados.

**Definición 3.4 (Árbol de Dialéctica)** Dado un de.l.p.  $\mathfrak{p}$ , un **árbol de dialéctica**  $\mathcal{T}_{\mathfrak{p}}(A)$  enraizado en un argumento  $A$  está determinado por el conjunto  $X$  de todas las líneas de argumentación exhaustivas de  $\mathfrak{p}$  enraizadas en  $A$ . Un argumento  $D \in \lambda'$ , con  $\lambda' \in X$ , es un **hijo** de  $B \in \lambda$  en  $\mathcal{T}_{\mathfrak{p}}(A)$  sssi  $D$  derrota a  $B \in \lambda'$  y las secuencias de  $A$  a  $B$  en  $\lambda'$  y  $\lambda$  son iguales. Las **hojas** de  $\mathcal{T}_{\mathfrak{p}}(A)$  son las hojas de cada línea en  $X$ .

El objetivo de un árbol de dialéctica es la evaluación de toda la información que podría determinar el estado de garantía del argumento raíz. Adicionalmente, las líneas de argumentación incluidas en el árbol deben conformar las condiciones de aceptabilidad para asegurar que el intercambio de argumentos se realiza de una manera sensata.

Dado un literal  $\alpha$  y un argumento  $\langle A, \alpha \rangle$  en un programa  $\mathfrak{p}$ , para decidir si  $\alpha$  está garantizado, en DELP se utiliza una función de marcado que se aplica sobre los árboles de dialéctica. Esta función es la dada en la Definición 2.34: cada nodo (*i.e.*, argumento) será marcado como D (*derrotado*) o U (*no derrotado*), conduciendo a un árbol marcado:

1. las hojas en  $\mathcal{T}_{\mathfrak{p}}(A)$  se marcan como U; y
2. cada nodo interno  $B$  de  $\mathcal{T}_{\mathfrak{p}}(A)$  se marca como U sssi cada hijo de  $B$  se marca como D.

De esta manera, un nodo interno  $B$  será marcado como D sssi tiene al menos un hijo marcado como U. Por último, si el nodo raíz  $A$  está marcado como U diremos que el árbol  $\mathcal{T}_{\mathfrak{p}}(A)$  *garantiza*  $\alpha$ , y que  $\alpha$  está *garantizado* por  $\mathfrak{p}$ . En ocasiones también se dirá que  $A$  está garantizado.

**Ejemplo 3.4** *A partir del de.l.p. del Ejemplo 3.1 podemos considerar un nuevo programa  $\mathfrak{p}_{3,4} = (\Pi_{3,1}, \Delta_{3,1} \cup \{a \multimap x\})$ , a partir del cual se pueden construir los argumentos del Ejemplo 3.2 y adicionalmente el argumento  $\langle A, a \rangle = \langle \{(a \multimap x), (x \multimap z)\}, a \rangle$ . Las relaciones de derrota son:  $B_1, B_2$  y  $B_5$  derrotan a  $A$ ;  $B_3$  derrota a  $B_2$ ;  $B_4$  derrota a  $B_3$ ; y  $B_6$  derrota a  $B_5$ . El árbol de dialéctica es el ilustrado en la Figura 3.1, donde los argumentos se denotan con triángulos, los arcos representan la relación de derrota, y el color gris indica que un argumento está marcado como derrotado. En*



este caso, el argumento raíz está derrotado y por lo tanto el árbol  $\mathcal{T}_{\mathfrak{p}_{3,4}}(A)$  no lo garantiza.

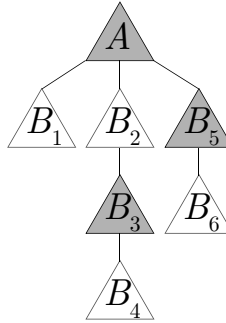


Fig. 3.1: Árbol de dialéctica del Ejemplo 3.4

Por simplicidad, aquellos argumentos que pueden ser contruidos a partir de  $\mathfrak{p}_{3,4}$  pero que no aparecen en el árbol  $\mathcal{T}_{\mathfrak{p}_{3,4}}(A)$  se asumen derrotados por aquellos argumentos que sí aparecen. Por ejemplo, se asume que  $A$  y  $B_3$  son preferidos sobre  $\langle \{\neg a \multimap y, y \multimap x, x \multimap p\}, \neg a \rangle$ , el cual, en consecuencia, no es un derrotador para  $A$  ni para  $B_3$ .

### 3.2. Programas Lógicos Rebatibles en Términos del DAF

La traducción de un *de.l.p.* a un DAF puede realizarse de forma relativamente directa a nivel de reglas y hechos. La dificultad radica en la construcción de las estructuras argumentales y en la definición de las derrotas. Cada hecho en un *de.l.p.* se traduce al DAF como una pieza de evidencia, y cada regla, sea estricta o rebatible, se traduce como un argumento. Dado un lenguaje  $\mathcal{L}$ , cada regla pertenece al dominio  $\mathcal{L} \times 2^{\mathcal{L}}$ ; sobre el mismo lenguaje, un argumento pertenece al mismo dominio, ya que también cuenta con un conjunto de premisas en  $2^{\mathcal{L}}$  y una conclusión en  $\mathcal{L}$ .

**Definición 3.5 (Conversión de Reglas)** Dado un conjunto de reglas

(estrictas y/o rebatibles)  $\Gamma \subseteq 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  definidas sobre un lenguaje  $\mathcal{L}$ , la **función de conversión de reglas** es  $\Psi_r : 2^{(\mathcal{L} \times 2^{\mathcal{L}})} \rightarrow 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  tal que:

$$\Psi_r(\Gamma) = \{ \langle \{\beta_1, \dots, \beta_n\}, \alpha \rangle \mid \alpha \rightarrow \beta_1, \dots, \beta_n \in \Gamma \text{ o } \alpha \leftarrow \beta_1, \dots, \beta_n \in \Gamma \}$$

Al traducir reglas en argumentos se pierde la cualidad de estricta o rebatible de cada una de ellas. En esta instancia surge una diferencia entre ambos formalismos, la cual será discutida más adelante en esta sección.

La traducción de los hechos se realiza en forma separada a las reglas, ya que estos representan el conjunto de evidencia del sistema. Al momento de realizar la conversión, la evidencia disponible quedará conformada de acuerdo a los hechos presentes en el programa en ese momento.

**Definición 3.6 (Conversión de Hechos)** Dado un conjunto de hechos  $\Omega \subset \mathcal{L}$  definidos sobre un lenguaje  $\mathcal{L}$ , la **función de conversión de hechos** es  $\Psi_f : 2^{\mathcal{L}} \rightarrow 2^{\mathcal{L}}$  tal que:

$$\Psi_f(\Omega) = \{ \omega \mid (\omega \leftarrow true) \in \Omega \}$$

Cada par de reglas con conclusiones complementarias en DELP se traduce al DAF como un par dentro de la relación de conflicto; esto incluye a los conflictos que involucran reglas estrictas.

**Definición 3.7 (Generación de la Relación de Conflicto)** Dado un conjunto de reglas (estrictas y/o rebatibles)  $\Gamma \subseteq 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  definidas sobre un lenguaje  $\mathcal{L}$ , la **relación de conflicto** subyacente a  $\Gamma$  es  $\Psi_c : 2^{(\mathcal{L} \times 2^{\mathcal{L}})} \rightarrow 2^{(\mathcal{L} \times 2^{\mathcal{L}})} \times 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  tal que:

$$\Psi_c(\Gamma) = \{ (\mathcal{A}, \mathcal{B}) \mid A, B \in \Gamma, \Psi_r(\{A\}) = \{A\}, \Psi_r(\{B\}) = \{B\}, \text{cl}(\mathcal{A}) = \overline{\text{cl}(\mathcal{B})} \}$$

Aquellas estructuras de argumento que emergen en un programa DELP se traducirán en estructuras argumentales, de forma relativamente directa, a través de las conversiones definidas arriba. La diferencia entre ambas entidades es que las estructuras de argumento en DELP contienen sólo reglas rebatibles, mientras que, en la conversión, las estructuras argumentales tienen argumentos que representan reglas de ambos tipos. En definitiva, las estructuras de argumento serán representadas por el DAF de manera *extendida*, codificando en ellas no sólo las reglas rebatibles, sino también las estrictas. Una formalización de DELP utilizando este tipo de argumentos se presenta en [VTS09].

**Lema 3.1** *Dado un de.l.p.  $\mathfrak{p} = (\Pi, \Delta)$ , si  $\langle A, \alpha \rangle$  es una estructura de argumento de  $\mathfrak{p}$ , entonces existe una estructura argumental  $\Sigma$  tal que  $\text{args}(\Sigma) = \Psi_r(A \cup \Pi_r)$  y  $\text{cl}(\Sigma) = \alpha$ , donde  $\Pi_r$  es el subconjunto de las reglas en  $\Pi' \subseteq \Pi$ , que es un subconjunto minimal de  $\Pi$  que verifica  $A \cup \Pi' \vdash \alpha$ .*

DEMOSTRACIÓN: *Trivial por Definición 2.7, ya que el árbol subyacente a  $\Sigma$  contiene las conversiones a argumento de las reglas en  $A$  junto con las reglas estrictas de  $\Pi$  (en el conjunto  $\Pi_r$ ) que forman parte de la derivación rebatible de  $\alpha$ .  $\square$*

### Definición 3.8 (Estructura Argumental Equivalente)

Sea  $\langle A, \alpha \rangle$  una estructura de argumento en un de.l.p.  $(\Pi, \Delta)$ , la **estructura argumental equivalente** a  $\langle A, \alpha \rangle$  es  $\Psi_s : 2^{(\mathcal{L} \times 2^{\mathcal{L}})} \times \mathcal{L} \rightarrow 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  tal que:

$$\Psi_s(\langle A, \alpha \rangle) = \Sigma, \text{args}(\Sigma) = \Psi_r(A \cup \Pi_r), \text{cl}(\Sigma) = \alpha,$$

donde  $\Pi_r$  es el subconjunto de las reglas en  $\Pi'$ , que es el subconjunto minimal de  $\Pi$  verificando  $A \cup \Pi' \vdash \alpha$ .

Debe destacarse que, por definición de estructura de argumento, no habrá ninguna que sea inconsistente con respecto a  $\Pi$ . Dado que se traduce de estructuras de argumento a estructuras argumentales, no se obtendrán estructuras argumentales violando esta propiedad.

**Lema 3.2** *Sea  $\langle A, \alpha \rangle$  una estructura de argumento en un de.l.p.  $(\Pi, \Delta)$ , la estructura argumental  $\Psi_s(\langle A, \alpha \rangle)$  está activa según el conjunto de evidencia  $\Psi_f(\Pi_f)$ , donde  $\Pi_f = \{(a \leftarrow true) \mid (a \leftarrow true) \in \Pi\}$ .*

**DEMOSTRACIÓN:** Sean  $\mathbf{p} = (\Pi, \Delta)$ ,  $\Sigma = \Psi_s(\langle A, \alpha \rangle)$ , y  $\mathbf{E} = \Psi_f(\Pi_f)$ . Supongamos por el absurdo que  $\Sigma$  no está activa según  $\mathbf{E}$ , i.e., las premisas de  $\Sigma$  no están satisfechas por evidencia y/o  $\Sigma$  no está bien formada. Supongamos por el absurdo que ambas no se verifican:

1.  $\text{pr}(\Sigma) \not\subseteq \mathbf{E}$ . Consideremos una premisa  $\beta$  de  $\Sigma$  tal que  $\beta \notin \mathbf{E}$ . Por la Definición 3.8, los argumentos en  $\Sigma$  provienen de la conversión de las reglas (estrictas y rebatibles) en  $\mathbf{p}$  que derivan rebatiblemente  $\alpha$ . Sea  $\Pi' \subseteq \Pi$  el subconjunto minimal tal que  $A \cup \Pi' \vdash \alpha$ . Consideremos una regla  $r \in A \cup \Pi'$  con una premisa  $\gamma$  en su cuerpo que no está derivada por otra regla en  $A \cup \Pi'$ ; luego,  $\gamma$  es una premisa en el argumento  $\Psi_r(r)$ , que es una hoja en el árbol subyacente a  $\Sigma$ . Asumamos que  $\Psi_f(\{\gamma\}) = \{\beta\}$ . Como  $\gamma$  no está satisfecha por ninguna regla en  $A \cup \Pi'$ , entonces  $\gamma \in \Pi_f$ . Eso significa que  $\beta \in \Psi_f(\Pi_f)$ , y  $\Psi_f(\Pi_f) = \mathbf{E}$ , lo cual es absurdo, el cual provino de suponer que  $\beta \notin \mathbf{E}$ . Luego,  $\text{pr}(\Sigma) \subseteq \mathbf{E}$ .

2.  $\Sigma$  no está bien formada:

- *Consistencia:* Sea  $\Pi' \subseteq \Pi$  el subconjunto minimal tal que  $A \cup \Pi' \vdash \alpha$ . Por la Definición 3.1,  $A \cup \Pi$  es consistente, por lo cual  $A \cup \Pi'$  también lo será. Los argumentos en  $\Sigma$  son la conversión de las reglas en  $A \cup \Pi'$  y los conflictos se generan a partir de

*conclusiones complementarias en el conjunto de reglas. Como  $A \cup \Pi'$  es consistente, su conversión, i.e.,  $\text{args}(\Sigma)$ , estará libre de conflictos.*

- *No circularidad: La estructura de argumento soporta  $\alpha$  a través de una derivación rebatible, que involucra una secuencia finita de pasos, y un conjunto minimal de reglas. Luego, una estructura de argumento no puede incluir una derivación circular. Para más detalles, referirse a [GS04].*
- *Uniformidad: La no uniformidad supondría que una misma premisa se deriva a partir de dos (o más) reglas distintas en la misma estructura de argumento. Si esto ocurriera, la estructura de argumento no sería minimal. Luego, el conjunto de reglas asociado a una estructura de argumento verifica uniformidad.*

*Por lo tanto, es absurdo suponer que la conversión de una estructura de argumento en DELP es una estructura argumental no activa.  $\square$*

La función de preferencia en DELP suele asumirse como especificidad generalizada, cuyo equivalente en términos de un DAF  $F$  será implementado por la función  $\text{pref}_e$ , que dadas dos estructuras argumentales de  $\text{str}_F$  retorna la más concisa o de mayor contenido informacional (para más detalle, referirse a [GS04]; en otro caso retorna  $\epsilon$  (ver Definición 2.12). La traducción de este principio de preferencia en términos del DAF no será formalizada aquí para no introducir complejidad innecesaria.

### **Definición 3.9 (Función de Preferencia DELP)**

*Dados un de.l.p.  $(\Pi, \Delta)$ , la función  $\text{pref}_e$  de especificidad generalizada sobre estructuras argumentales, y dos estructuras argumentales  $\Sigma_1, \Sigma_2 \in$*

$\Psi_r(\Pi \cup \Delta)$ , la **función de preferencia** DELP es:

$$\mathbf{pref}_{delp}(\Sigma_1, \Sigma_2) = \begin{cases} \Sigma_k & \Gamma \subseteq \Pi, \Psi_r(\Gamma) = \mathbf{args}(\Sigma_{k=1,2}) \\ \mathbf{pref}_e(\Sigma_1, \Sigma_2) & \text{de otra manera} \end{cases}$$

De esta forma, se respeta la preponderancia de las derivaciones estrictas por sobre cualquier argumento en oposición a ellas. Así se simula el chequeo de consistencia con respecto a  $\Pi$  de la definición de argumento, sin modificar la definición de estructura argumental bien formada, y conservando las derivaciones estrictas. Esto quedará claro en el Ejemplo 3.5.

**Definición 3.10 (DAF Equivalente)** Sea  $(\Pi, \Delta)$  un de.l.p., el **DAF equivalente** se define como:

$$\Psi_{delp}((\Pi, \Delta)) = \langle \Psi_f(\Pi_f), \Psi_r(\Pi_r \cup \Delta), \Psi_c(\Pi_r \cup \Delta), \mathbf{pref}_{delp} \rangle,$$

donde  $\Pi = \Pi_f \cup \Pi_r$ ,  $\Pi_f \cap \Pi_r = \emptyset$ ,  $\Pi_f = \{(a \leftarrow true) \mid (a \leftarrow true) \in \Pi\}$ .

**Ejemplo 3.5** Consideremos el de.l.p.:

$$(\{(\neg a \leftarrow true), (b \leftarrow true)\}, \{a \multimap b\})$$

En DELP,  $A = \langle \{a \multimap b\}, a \rangle$  no respeta la definición de estructura de argumento. Si consideramos el DAF equivalente  $\langle \{\neg a, b\}, \langle \{b\}, a \rangle, \emptyset, \mathbf{pref}_{delp} \rangle$ , el argumento  $\langle \{b\}, a \rangle$  es inconsistente con respecto a la evidencia y no será activo.

Si pasamos a considerar el de.l.p.:

$$(\{(\neg a \leftarrow c), (c \leftarrow true), (b \leftarrow true)\}, \{a \multimap b\}),$$

En este caso,  $A$  tampoco respeta la definición de estructura de argumento. Sin embargo, en el DAF equivalente, existirán los argumentos

$\mathcal{B} = \langle \{b\}, a \rangle$  y  $\mathcal{C} = \langle \{c\}, \neg a \rangle$ , que se determinarán en conflicto (Definición 3.7). Aquí entra en juego la función de preferencia. Las estructuras argumentales primitivas serán, respectivamente,  $\Sigma_b$  y  $\Sigma_c$ , y por Definición 3.9  $\Sigma_c$  será preferida a  $\Sigma_b$ . En términos de garantía, ambos formalismos se comportan igual. En DELP, al consultar por  $\neg a$  se encontrará que se deriva estrictamente y la garantía provendrá de un argumento con un conjunto vacío de reglas rebatibles. En el DAF equivalente, la garantía de  $\Sigma_c$  provendrá de la definición de la función  $\mathbf{pref}_{delp}$ , que da preponderancia a aquellas estructuras conteniendo conversiones de reglas estrictas.

Sin embargo, el hecho de que en el DAF existan estructuras argumentales representando derivaciones estrictas (que en  $\mathbf{p}$  son argumentos vacíos) reviste una dificultad: en DELP las derivaciones rebatibles inconsistentes con respecto a la parte estricta del programa no generan una estructura de argumento, *i.e.*, no aparecen en los árboles de dialéctica. En cambio, en el DAF equivalente, estas estructuras argumentales sí formarán parte del árbol de dialéctica, así como las que representan esa porción de conocimiento puramente estricto, que las derrotan. Este punto resulta clave, ya que la definición de  $\mathbf{pref}_{delp}$  asegura que esto siempre se verifica. Es decir, siempre que una estructura argumental  $I$ , inconsistente con el conocimiento estricto, se incorpora a un árbol, también se incorpora la estructura argumental de conocimiento estricto  $S$  que derrota a  $I$ . La estructura  $S$  es una hoja, ya que ninguna otra estructura la derrota, por definición de  $\mathbf{pref}_{delp}$ . Como plantea el siguiente lema, la inclusión de cada estructura  $I$  no cambia el marcado del árbol de dialéctica, ya que cada una de ellas siempre estará derrotada por la correspondiente estructura  $S$ , y la presencia de ambas en el árbol es irrelevante al marcado, independientemente de la posición que ocupen.

**Lema 3.3** *Sea  $T$  un árbol de dialéctica de acuerdo a las restricciones dialécticas de DELP con un conjunto  $N$  de nodos y un conjunto  $A$  de arcos, marcado según el criterio escéptico de DELP. Un árbol  $T'$  con un conjunto de nodos  $N' = N \cup \{S, I\}$  y un conjunto de arcos  $A' = A \cup \{(S, I)\} \cup A''$ , donde  $A'' \subseteq \{(I, X) \mid X \in N\}$ , marcado bajo el mismo criterio, conserva la marca de los nodos en  $N$ .*

DEMOSTRACIÓN: *Las tres posibles inclusiones de los arcos  $(S, I)$  e  $(I, X)$  se ilustran en la Figura 3.2. El color gris indica un nodo marcado como derrotado, y el blanco significa que el nodo está no derrotado.*

- a) *Al incorporar el arco  $(I, X)$  sobre un nodo  $X$  que era hoja el marcado de  $X$  no cambia, pues  $I$  se encuentra siempre marcado como derrotado, y  $X$  tendría a todos sus hijos derrotados. Ver Figura 3.2(a).*
- b) *Si el nodo  $X$  tenía al menos un hijo no derrotado,  $X$  está derrotado, independientemente de la marca del resto de sus hijos. Ver Figura 3.2(b).*
- c) *Si el nodo  $X$  estaba no derrotado porque todos sus hijos estaban derrotados, la incorporación de  $(I, X)$  no altera esta situación, ya que  $I$  está derrotado. Ver Figura 3.2(c).*

*Sólo resta considerar si la inclusión de los arcos  $(S, I)$  e  $(I, X)$  respetan las restricciones dialécticas de DELP:*

- *No circularidad: dado que  $S, I \notin N$  y  $S$  es un nodo hoja, es imposible que ocurra una circularidad.*
- *Concordancia: si  $I$  no respeta concordancia, no se incorpora al árbol, i.e., no lo altera. Supongamos por el absurdo que  $I$  respeta concordancia y  $S$  no, entonces  $S$  derrota a (i.e., es hijo de) un nodo  $Z$  ancestro de  $I$ , lo cual significa que  $Z$  es inconsistente con*



el conocimiento estricto, pero  $Z$  es un nodo del árbol  $T$ , el cual contiene estructuras de argumento, y éstas deben ser consistentes con el conocimiento estricto. El absurdo provino de asumir que  $S$  no respeta concordancia.

- *No doble bloqueo: el arco  $(S, I)$  denota una derrota propia, por lo cual si  $(I, X)$  no resultó en un doble bloqueo, ambos arcos se incorporan al árbol. Si  $(I, X)$  introdujera un doble bloqueo, directamente no sería incorporado al árbol.*

Luego, la incorporación de los arcos  $(S, I)$  e  $(I, X)$  a  $T$  no cambia su marcado.  $\square$

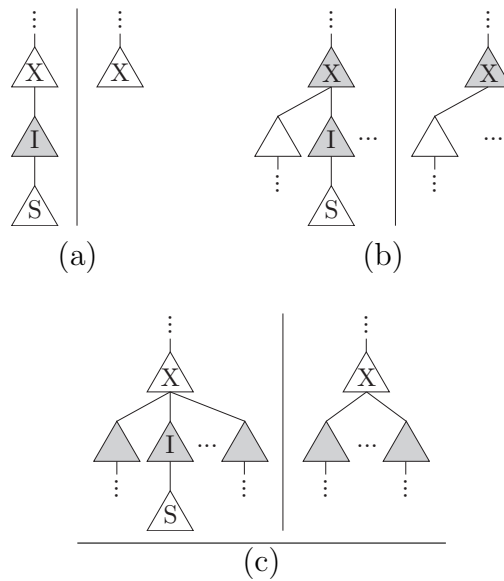


Fig. 3.2: Las tres posibles formas de inclusión de los nodos  $S$  e  $I$

**Teorema 3.1** *Si  $\langle A, \alpha \rangle$  es una estructura de argumento garantizada por un de.l.p.  $\mathbf{p}$ , entonces la estructura argumental equivalente  $\Psi_s(\langle A, \alpha \rangle)$  está garantizada a partir del DAF equivalente  $\Psi_{del\mathbf{p}}(\mathbf{p})$ , de acuerdo a la*

semántica de árboles de dialéctica con restricciones dialécticas de no circularidad, concordancia, y no doble bloqueo, utilizando una función de marcado escéptica.

DEMOSTRACIÓN: Debemos demostrar que la marca de cada nodo en el árbol de dialéctica  $T$  para  $\langle A, \alpha \rangle$  en  $\mathfrak{p}$  es igual a la marca del nodo equivalente en el árbol de dialéctica activo  $\mathbb{T}_\Sigma(F)$  para  $\Sigma = \Psi_s(\langle A, \alpha \rangle)$  en  $F = \Psi_{del\mathfrak{p}}(\mathfrak{p})$ :

1. Por el Lema 3.2, la conversión de cada estructura de argumento en  $\mathfrak{p}$  retorna una estructura argumental activa según la conversión de los hechos en  $\mathfrak{p}$ . Luego, cada nodo de  $T$  es un nodo de  $\mathbb{T}_\Sigma(F)$ .
2. Por la Definición 3.7, cada par de reglas (estrictas y/o rebatibles) de  $\mathfrak{p}$  con conclusiones complementarias genera un conflicto entre argumentos de  $F$ . Por la Definición 3.9, se le da preferencia a aquellas estructuras argumentales enteramente compuestas por conversiones de reglas estrictas; en su defecto, se le da más preferencia a las más específicas. Este último caso es trivial: el comportamiento en  $F$  es igual que en  $\mathfrak{p}$ , porque la función de preferencia es la misma, aplicada sobre conocimiento equivalente. En el caso de las estructuras con conocimiento sólo estricto, por el Lema 3.3, no son relevantes al marcado del árbol. Luego, cada arco en  $T$  es un arco en  $\mathbb{T}_\Sigma(F)$ , y los arcos y nodos adicionales no cambian el marcado de la raíz de  $\mathbb{T}_\Sigma(F)$ .

Por lo tanto, todo nodo en  $T$  tiene su equivalente en  $\mathbb{T}_\Sigma(F)$ , y los arcos y nodos adicionales en  $\mathbb{T}_\Sigma(F)$  no cambian el marcado de su raíz. Luego, si  $\langle A, \alpha \rangle$  está garantizada en  $\mathfrak{p}$ ,  $\Sigma$  está garantizada en  $F$ , aplicando la semántica de árboles de dialéctica.  $\square$

### 3.3. El Sistema ASPIC

El sistema ASPIC (la sigla de *Argumentation Service Platform with Integrated Components*) [Pro04] fue desarrollado en el contexto del proyecto de ese mismo nombre. Este proyecto, comenzado en el año 2004 y finalizado en el año 2006, planteó el marco argumentativo que describiremos en esta sección, junto a la exploración de las propiedades deseables que debería tener, así como diversos dominios de aplicación. Para más detalles se recomienda al lector dirigirse a [www.argumentation.org](http://www.argumentation.org).

La explicación de este sistema resultará más somera que el caso de DELP, ya que varios elementos coinciden en su intuición. El sistema ASPIC está definido en base a un lenguaje  $\mathcal{L}$  de literales, capaz de representar negación; y se asume la existencia de una función complemento “ $\neg$ ”, aplicable a elementos de  $\mathcal{L}$ , tal que  $\neg\neg\psi = \psi$  y  $\neg\neg\neg\psi = \neg\psi$ .

Una *regla estricta* es una expresión de la forma  $\phi_1, \dots, \phi_n \longrightarrow \psi$ , indicando que si se verifican  $\phi_1, \dots, \phi_n$ , entonces sin excepción se verifica  $\psi$ . Una *regla rebatible* se denota  $\phi_1, \dots, \phi_n \Longrightarrow \psi$ , indicando que si se verifican  $\phi_1, \dots, \phi_n$ , entonces usualmente se verifica  $\psi$ . Cada  $\phi_{1 \leq i \leq n}$  y  $\psi$  son elementos de  $\mathcal{L}$ . A diferencia de DELP, los hechos son considerados un caso particular de regla estricta con cuerpo vacío.

**Definición 3.11 (Teoría Rebatible)** *Una teoría rebatible  $\mathcal{T}$  es un par  $\langle \mathcal{S}, \mathcal{D} \rangle$  donde  $\mathcal{S}$  es un conjunto de reglas estrictas, y  $\mathcal{D}$  es un conjunto de reglas rebatibles.*

Los argumentos se construirán a partir de una dada teoría rebatible. Previo a la definición de argumento se introducen las siguientes funciones:

- **Conc** retorna la conclusión final de un argumento;

- Sub retorna todos los subargumentos de un argumento;
- `StrictRules` y `DefRules` retornan, respectivamente, el conjunto de reglas estrictas y rebatibles utilizadas en un argumento.

**Definición 3.12 (Argumento)** *Dada una teoría rebatible  $\langle \mathcal{S}, \mathcal{D} \rangle$ , un argumento  $A$  es<sup>2</sup>:*

- $A_1, \dots, A_n \rightarrow \psi$  si  $A_1, \dots, A_n$  son argumentos tal que existe una regla estricta  $\text{Conc}(A_1), \dots, \text{Conc}(A_n) \rightarrow \psi$ ; el argumento  $A$  es tal que:
  - $\text{Conc}(A) = \psi$ ;
  - $\text{Sub}(A) = \text{Sub}(A_1) \cup \dots \cup \text{Sub}(A_n) \cup \{A\}$ ;
  - $\text{StrictRules}(A) = \text{StrictRules}(A_1) \cup \dots \cup \text{StrictRules}(A_n) \cup \{\text{Conc}(A_1), \dots, \text{Conc}(A_n) \rightarrow \psi\}$ ;
  - $\text{DefRules}(A) = \text{DefRules}(A_1) \cup \dots \cup \text{DefRules}(A_n)$ .
- $A_1, \dots, A_n \Rightarrow \psi$  si  $A_1, \dots, A_n$  son argumentos tal que existe una regla rebatible  $\text{Conc}(A_1), \dots, \text{Conc}(A_n) \Rightarrow \psi$ ; el argumento  $A$  es tal que:
  - $\text{Conc}(A) = \psi$ ;
  - $\text{Sub}(A) = \text{Sub}(A_1) \cup \dots \cup \text{Sub}(A_n) \cup \{A\}$ ;
  - $\text{StrictRules}(A) = \text{StrictRules}(A_1) \cup \dots \cup \text{StrictRules}(A_n)$ ;
  - $\text{DefRules}(A) = \text{DefRules}(A_1) \cup \dots \cup \text{DefRules}(A_n) \cup \{\text{Conc}(A_1), \dots, \text{Conc}(A_n) \Rightarrow \psi\}$ .

**Definición 3.13 (Argumentos Estrictos y Rebatibles)** *Un argumento  $A$  es **estricto** sssi  $\text{DefRules}(A) = \emptyset$ ; de otra manera, es **rebatible**.*

<sup>2</sup> Las flechas que denotan argumentos son más cortas que las que denotan reglas.

**Ejemplo 3.6** Dados  $\mathcal{S}_{3,6} = \{\rightarrow a, \rightarrow d\}$  y  $\mathcal{D}_{3,6} = \{a \implies b, d \implies \neg b\}$ , se pueden construir los siguientes argumentos:

$$A_1 : [\rightarrow a]$$

$$A_2 : [\rightarrow d]$$

$$A_3 : [A_1 \rightarrow b]$$

$$A_4 : [A_2 \rightarrow \neg b]$$

$A_1$  y  $A_2$  son argumentos estrictos, mientras que  $A_3$  y  $A_4$  son argumentos rebatibles. Además, estos dos argumentos tienen como subargumentos a  $A_1$  y  $A_2$ , respectivamente.

Los argumentos pueden estar en conflicto con otros de diferentes maneras. Uno de estos tipos de conflicto está asociado a las conclusiones de los argumentos: dos argumentos estarán en conflicto cuando den soporte a conclusiones contradictorias.

**Definición 3.14 (Refutación)** Dados dos argumentos  $A$  y  $B$ ,  $A$  **refuta** a  $B$  sssi  $\exists A' \in \text{Sub}(A)$  donde  $\text{Conc}(A') = \phi$  y  $\exists B' \in \text{Sub}(B)$  donde  $B'$  es un argumento rebatible y  $\text{Conc}(B') = -\phi$ .

Esta definición da prevalencia a los argumentos estrictos por sobre los rebatibles, en el sentido en que un argumento estricto puede refutar a uno rebatible, pero la inversa no se verifica. Más aún, puede verse que los argumentos pueden estar en desacuerdo no sólo en sus conclusiones finales, sino también en las intermedias.

**Ejemplo 3.7** Dados  $\mathcal{S} = \{\rightarrow a, \rightarrow t, a \rightarrow b\}$  y  $\mathcal{D} = \{b \implies c, t \implies \neg b, \neg b \implies d\}$ , el argumento  $[[[\rightarrow a] \rightarrow b] \Rightarrow c]$  refuta a  $[[[\rightarrow t] \Rightarrow \neg b] \Rightarrow d]$ . La inversa no se verifica.

Existe otro tipo de conflicto que puede capturarse en un formalismo argumentativo: un argumento podría objetar la aplicabilidad de una regla utilizada por otro argumento. En la definición que sigue se utiliza el operador de objetivación “[·]” [Pol95], que convierte una expresión de meta-nivel (en este caso, una regla rebatible) en una expresión a nivel de objeto (en este caso, un literal). Esto es requerido por el formato de los argumentos, donde la conclusión puede ser únicamente un literal, mientras que para realizar un *undercut* (en español, “socavamiento”), es necesario expresar la no aplicabilidad de una regla.

**Definición 3.15 (Socavamiento)** *Dados dos argumentos  $A$  y  $B$ ,  $A$  socava a  $B$  sssi  $\exists B' \in \text{Sub}(B)$  tal que  $B' : B'_1, \dots, B'_n \Rightarrow \psi$  y  $\exists A' \in \text{Sub}(A)$  donde  $\text{Conc}(A') = \neg[\text{Conc}(B'_1), \dots, \text{Conc}(B'_n) \Longrightarrow \psi]$ .*

Un clásico ejemplo [Pol95] que ilustra la diferencia entre refutación y socavamiento es el siguiente. Consideremos un argumento ( $A$ ) “*El objeto es rojo porque Mauro dice que se ve rojo.*” Una refutación para  $A$  podría ser ( $B_1$ ) “*El objeto no es rojo porque Martín dice que se ve azul.*” Un socavamiento para  $A$  podría ser ( $B_2$ ) “*El objeto está iluminado por una luz roja.*” Si bien esto no es una razón para que el objeto no sea rojo, sí expone que el hecho de que se vea rojo no es una razón para que realmente lo sea. Por último, para introducir la noción de *derrota*, se asume un factor de certeza asociado a cada regla. Siguiendo el principio afirmando que “una cadena es tan fuerte como el más débil de sus eslabones”, el factor de certeza de un argumento es el mínimo de los factores de certeza de las reglas incluidas en él. En P-DELP [ACGS08] se introdujo un criterio similar para decidir la derrota entre argumentos conflictivos. Cabe aclarar que este criterio para comparar argumentos es el que se utiliza en el sistema implementado, pero los autores aclaran que podría haber otros

criterios, *e.g.*, prioridades entre reglas.

**Definición 3.16 (Factor de Certeza)** *Dada una teoría rebatible  $\langle \mathcal{S}, \mathcal{D} \rangle$ , el factor de certeza de una regla  $r \in \mathcal{S} \cup \mathcal{D}$  es un número real y se denota  $\text{fc}(r, \langle \mathcal{S}, \mathcal{D} \rangle)$ . El factor de certeza de un argumento es  $\text{fc}(A, \langle \mathcal{S}, \mathcal{D} \rangle) = \min(\bigcup_{r \in R} (\text{fc}(r, \langle \mathcal{S}, \mathcal{D} \rangle)))$ , donde  $R = \text{DefRules}(A) \cup \text{StrictRules}(A)$  y la función  $\min(\cdot)$  retorna el mínimo valor de un conjunto de números reales.*

**Definición 3.17 (Derrota)** *Dados dos argumentos  $A$  y  $B$  de una teoría rebatible  $\mathcal{T}$ ,  $A$  **derrota** a  $B$  si lo refuta o socava y  $\text{fc}(A, \mathcal{T}) > \text{fc}(B, \mathcal{T})$ .*

Una teoría rebatible en el sistema ASPIC se define como un par de conjuntos de reglas  $\mathcal{T} = \langle \mathcal{S}, \mathcal{D} \rangle$ ; sin embargo, también puede verse como un par  $\langle W, \leftrightarrow \rangle$ , donde  $W$  es el conjunto de todos los argumentos que pueden construirse a partir de  $\mathcal{T}$ , y “ $\leftrightarrow$ ” se define como la relación que incluye todas las refutaciones y socavamientos. Para determinar qué argumentos están aceptados, puede utilizarse cualquiera de las semánticas definidas sobre el AF de Dung, dando lugar a un conjunto de extensiones.

En este sistema se distinguen aquellas conclusiones que pueden determinarse como aceptadas, y se denominan *conclusiones justificadas*, englobadas dentro del conjunto **Output**.

**Definición 3.18 (Conclusiones Justificadas)** *Dado un AF  $\langle W, \leftrightarrow \rangle$  y su conjunto de extensiones bajo alguna semántica de Dung  $\{E_1, \dots, E_n\}$ :*

- $\text{Concs}(E_i) = \{\text{Conc}(A) \mid A \in E_{1 \leq i \leq n}\}$
- $\text{Output} = \bigcap_{1 \leq i \leq n} (\text{Concs}(E_i))$

La definición de **Output** sigue una actitud escéptica, dado que una postura crédula podría resultar en la inclusión de conclusiones contradictorias, incluso cuando cada extensión sea individualmente consistente.

**Ejemplo 3.8** *Dados los conjuntos  $\mathcal{S}_{3,6}$  y  $\mathcal{D}_{3,6}$ , se pueden construir los argumentos del Ejemplo 3.6. Se tiene que los argumentos  $\mathcal{A}_3$  y  $\mathcal{A}_4$  se derrotan mutuamente, y  $\mathcal{A}_1$  y  $\mathcal{A}_2$  no tienen derrotadores, tenemos las extensiones  $E_1 = \{\mathcal{A}_3, \mathcal{A}_1, \mathcal{A}_2\}$  y  $E_2 = \{\mathcal{A}_4, \mathcal{A}_1, \mathcal{A}_2\}$ , y el conjunto de conclusiones justificadas es  $\text{Output} = \{a, d\}$ .*

### 3.4. Teorías Rebatibles en Términos del DAF

En el caso del sistema ASPIC la traducción será más sencilla, ya que no se consideran excepcionalmente los hechos, y toda regla es a su vez un argumento, inclusive las estrictas.

**Definición 3.19 (Conversión de Reglas)** *Dado un conjunto de reglas (estrictas y/o rebatibles)  $\Gamma \subseteq 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  definidas sobre un lenguaje  $\mathcal{L}$ , la **función de conversión de reglas** es  $\Psi_R : 2^{(\mathcal{L} \times 2^{\mathcal{L}})} \rightarrow 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  tal que:*

$$\Psi_R(\Gamma) = \{ \{ \{ \beta_1, \dots, \beta_n \}, \alpha \} \mid \{ \beta_1, \dots, \beta_n \} \rightarrow \alpha \in \Gamma \text{ o } \beta_1, \dots, \beta_n \implies \alpha \in \Gamma \}$$

**Definición 3.20 (Generación de la Relación de Conflicto)**

*Dado un conjunto  $\Gamma$  de reglas (estrictas y/o rebatibles), la **relación de conflicto** subyacente a  $\Gamma$  es  $\Psi_C : 2^{(\mathcal{L} \times 2^{\mathcal{L}})} \rightarrow 2^{(\mathcal{L} \times 2^{\mathcal{L}})} \times 2^{(\mathcal{L} \times 2^{\mathcal{L}})}$  tal que:*

$$\Psi_C(\Gamma) = \{ (\mathcal{A}, \mathcal{B}) \mid A, B \in \Gamma, \text{Conc}(A) = -\text{Conc}(B), \Psi_R(A) = \mathcal{A}, \Psi_R(B) = \mathcal{B} \}$$

En el caso del sistema ASPIC, el DAF no es capaz de representar todas las formas de derrota, ya que no puede modelar los socavamientos. La utilización de un operador de objetivización podría lograr este efecto, tal como en ASPIC, pero complicaría la representación de conocimiento.



La codificación de argumentos con el objetivo de socavar la forma de razonamiento de una determinada estructura argumental requeriría mucho esfuerzo por parte del ingeniero de conocimiento. Un enfoque actualmente en curso consiste en realizar una clasificación de *tipos de inferencia*, para luego etiquetar cada argumento indicando bajo qué forma de razonamiento las premisas soportan a la conclusión. La inclusión de estas etiquetas en el lenguaje objeto permitirá la existencia de argumentos que especifiquen bajo qué condiciones (premisas) determinado tipo de inferencia no es aceptable (conclusión). A diferencia de la codificación manual de este tipo de conflictos, los conflictos podrían generarse automáticamente.

**Definición 3.21 (Función de Preferencia ASPIC)** *Dados una teoría rebatible  $\mathcal{T}$ , un factor de certeza asociado a cada regla en  $\mathcal{T}$ , y dos estructuras argumentales  $\Sigma_1, \Sigma_2 \in \Psi_R(\mathcal{S} \cup \mathcal{D})$ , la **función de preferencia ASPIC** es:*

$$\mathbf{pref}_{aspic}(\Sigma_A, \Sigma_B) = \begin{cases} \Sigma_A & \text{fc}(A, \mathcal{T}) > \text{fc}(B, \mathcal{T}) \\ \Sigma_B & \text{fc}(A, \mathcal{T}) < \text{fc}(B, \mathcal{T}) \\ \epsilon & \text{de otra manera} \end{cases}$$

donde  $\Psi_S(A) = \Sigma_A, \Psi_S(B) = \Sigma_B$ .

Finalmente, a partir de una teoría rebatible podemos obtener el DAF equivalente, de la siguiente manera.

**Definición 3.22 (DAF Equivalente)** *Dada una teoría rebatible sin socavamientos  $\langle \mathcal{S}, \mathcal{D} \rangle$ , el **DAF equivalente** se define como:*

$$\Psi_{aspic}(\langle \mathcal{S}, \mathcal{D} \rangle) = \langle \emptyset, \Psi_R(\mathcal{S} \cup \mathcal{D}), \Psi_C(\mathcal{S} \cup \mathcal{D}), \mathbf{pref}_{aspic} \rangle$$

Nótese que, a pesar de que el conjunto de evidencia es vacío, esto no evita la existencia de argumentos activos ni la construcción de estructuras activas, ya que todo argumento con un conjunto vacío de premisas estará activo, y a partir de este tipo de argumentos se podrán construir las estructuras activas. Por lo tanto, toda estructura activa en un DAF equivalente a una teoría rebatible tiene un conjunto vacío de premisas.

Un argumento en ASPIC (ver Definición 3.12) es básicamente un árbol de derivación que puede incluir tanto reglas estrictas como rebatibles. Dado que ambos tipos de reglas se convierten como argumentos en términos del DAF, un argumento ASPIC es automáticamente una estructura argumental del DAF. Estas estructuras no necesariamente tienen que estar bien formadas, puesto que en ASPIC no se impone restricción alguna en la definición de argumento.

**Lema 3.4** *Dado una teoría rebatible  $\mathcal{T} = \langle \mathcal{S}, \mathcal{D} \rangle$ , si  $A$  es un argumento en  $\mathcal{T}$ , entonces existe una estructura argumental  $\Sigma$  tal que  $\text{args}(\Sigma) = \Psi_R(\mathcal{S}' \cup \mathcal{D}')$  y  $\text{cl}(\Sigma) = \text{Conc}(A)$ , donde  $\mathcal{S}'$  (resp.,  $\mathcal{D}'$ ) es un subconjunto de las reglas en  $\mathcal{S}$  (resp.,  $\mathcal{D}$ ) tal que  $\mathcal{S}' \cup \mathcal{D}'$  deriva  $\alpha$ .*

DEMOSTRACIÓN: *Trivial por Definición 3.12, ya que el árbol subyacente a  $\Sigma$  contiene las conversiones a argumento de las reglas en  $\mathcal{S}' \cup \mathcal{D}'$  necesarias para derivar  $\text{Conc}(A)$ .  $\square$*

**Definición 3.23 (Estructura Argumental Equivalente)** *Dado un argumento  $A$ , la **estructura argumental equivalente** a  $A$  es  $\Psi_S : 2^{\mathcal{L} \times 2^{\mathcal{L}}} \rightarrow 2^{\mathcal{L} \times 2^{\mathcal{L}}}$  tal que:*

$$\Psi_S(A) = \Sigma, \text{args}(\Sigma) = \Psi_R(\text{DefRules}(A) \cup \text{StrictRules}(A))$$

**Teorema 3.2** *Sean  $S$  una semántica de Dung y  $\mathcal{T}$ , una teoría rebatible que no incluye socavamientos. Si  $A$  es un argumento garantizado por*

$\mathcal{T}$  de acuerdo a  $S$ , entonces la estructura argumental equivalente  $\Psi_S(A)$  está garantizada a partir del DAF equivalente  $\Psi_{aspic}(\mathcal{T})$  de acuerdo a  $S$ .

DEMOSTRACIÓN: Por el Lema 3.4 la estructura argumental  $\Sigma = \Psi_S(A)$  existe y representa el mismo conocimiento que  $A$ . Sea  $F = \Psi_{aspic}(\mathcal{T})$ ,  $(\mathbb{S}, \Rightarrow)$ , la instancia activa de  $F$ , y  $\langle W, \leftrightarrow \rangle$ , un AF tal que  $W$  es el conjunto de todos los argumentos que pueden construirse a partir de  $\mathcal{T}$ , y “ $\leftrightarrow$ ” es la relación de derrota sobre  $W$  incluyendo todas las refutaciones. Dado que todo argumento en  $W$  es una estructura activa en  $\mathbb{S}$ , y toda derrota en “ $\leftrightarrow$ ” se mapea como un conflicto en “ $\Rightarrow$ ”, si  $A \in W$  está garantizado al aplicar la semántica  $S$  sobre  $\langle W, \leftrightarrow \rangle$ ,  $\Sigma$  estará garantizada al aplicar  $S$  sobre  $(\mathbb{S}, \Rightarrow)$ .  $\square$



## 4. TEORÍA DEL CAMBIO EN ARGUMENTACIÓN

El conjunto de evidencia en el DAF representa el estado actual del mundo, el cual debería cambiar de manera acorde. Esto se realiza mediante las operaciones de borrado y agregado múltiple de evidencia presentadas en 2.4. El cambio en el conjunto de evidencia tendrá una correlación directa sobre el conjunto de argumentos activos. La representación de conocimiento también es pasible de ser modificada, por lo cual el conjunto de trabajo de argumentos podría cambiar, tal como fue explicado en 2.4. De nuevo, esta variación en la disponibilidad de argumentos afectará al subconjunto de argumentos activos.

En este capítulo se definirá un operador de revisión a nivel de argumentos enmarcado en el formalismo Argument Theory Change (ATC). Este formalismo toma como inspiración elementos de la teoría clásica de cambio de creencias y los aplica al DAF. Para definir esta revisión consideraremos tres enfoques: el primero (el único que será explicado detalladamente) alcanza la garantía del argumento en cuestión a través de desactivación de argumentos; el segundo, mediante activación; y el tercero es un método híbrido que combina los dos anteriores. Complementariamente, se discutirá un enfoque que define la revisión en términos de cambio sobre la relación de ataque.

Es importante destacar que el cambio al cual se hace referencia aquí no se aplica directamente sobre el mundo, sino que es un *cambio en la teoría*. Es decir, el resultado de una revisión por un argumento indicará cuál

es el “mundo” que garantiza dicho argumento. Más aún, este mundo será el más cercano al actual en términos del principio de mínimo cambio adoptado.

#### 4.1. Contracción Garantizante de Estruct. Argumentales

La idea general del operador de contracción por garantía de un argumento es retirar todo argumento que compromete la garantía del argumento en cuestión. La teoría abstracta de este método proviene de [RMF<sup>+</sup>08b], mientras que una reificación de la misma a *programación en lógica rebatible* se describió en [RMF<sup>+</sup>08a].

En ATC se busca alcanzar la garantía de una estructura argumental en particular; por esto, el estado de garantía de dicha estructura se verifica mediante su correspondiente árbol de dialéctica activo. Cuando un árbol de dialéctica no garantiza a la estructura raíz, es posible identificar qué líneas de argumentación son las “responsables” de esta situación. Para ello se define el *conjunto de ataque*.

**Definición 4.1 (Conjunto de Ataque)** *Dado un DAF  $F$  y un árbol de dialéctica activo  $\mathbb{T}_F(\Sigma)$  con un conjunto  $S$  de líneas de argumentación, el **conjunto de ataque**  $\text{Att}(\mathbb{T}_F(\Sigma))$  es el subconjunto minimal de  $S$  tal que el árbol construido a partir de  $S \setminus \text{Att}(\mathbb{T}_F(\Sigma))$  garantiza  $\Sigma$ , o  $\text{Att}(\mathbb{T}_F(\Sigma)) = S$ .*

En ciertos escenarios, el conjunto de ataque será la totalidad de las líneas del árbol ( $\text{Att}(\mathbb{T}_F(\Sigma)) = S$ ); esto ocurre cuando todas las líneas deben ser alteradas para alcanzar la garantía de la estructura raíz. En este caso, la diferencia entre el conjunto de líneas del árbol y el conjunto de ataque sería el conjunto vacío.

Aquellas líneas en el conjunto de ataque se denominarán *líneas de*

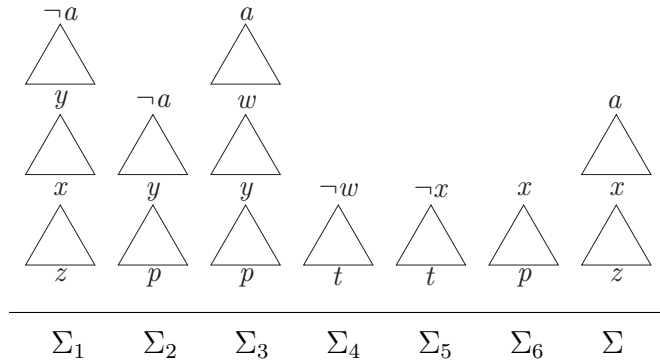
ataque. Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \mathbf{pref} \rangle$ , denotaremos como  $\mathbf{lines}_F$  al conjunto de todas las líneas de argumentación que se pueden construir a partir de las estructuras bien formadas en  $\mathbf{str}_F$ .

**Ejemplo 4.1** Consideremos el DAF  $F = \langle \mathbf{E}, W, \bowtie, \mathbf{pref} \rangle$ , donde:

$$\mathbf{E} = \{p, z, t\}$$

$$W = \left\{ \begin{array}{lll} \langle \{z\}, x \rangle & \langle \{x\}, y \rangle & \langle \{y\}, \neg a \rangle \\ \langle \{p\}, y \rangle & \langle \{y\}, w \rangle & \langle \{w\}, a \rangle \\ \langle \{t\}, \neg w \rangle & \langle \{t\}, \neg x \rangle & \langle \{p\}, x \rangle \\ \langle \{x\}, a \rangle & & \end{array} \right\}$$

Tenemos las siguientes estructuras argumentales activas:



Ahora consideremos que la relación de conflicto “ $\bowtie$ ” y la función de preferencia  $\mathbf{pref}$  determinan el árbol de dialéctica de la Figura 4.1, enraizado en  $\Sigma$  y marcado según la Definición 2.34; el color gris indica una estructura derrotada, mientras que el blanco significa que la estructura está no derrotada.

El conjunto de ataque del árbol de la Figura 4.1 deberá contener, al menos, las dos líneas de la izquierda, ya que los derrotadores directos de la raíz están marcados como no derrotados, provocando que la

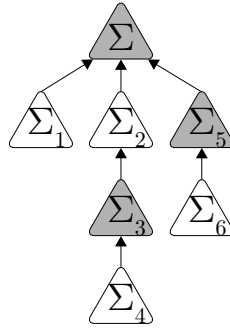


Fig. 4.1: Árbol de dialéctica del Ejemplo 4.1

raíz esté derrotada. Quitando estas dos líneas obtenemos el árbol de la Figura 4.2, que garantiza a  $\Sigma$ . Por lo tanto, el conjunto de ataque es  $\{[\Sigma, \Sigma_1], [\Sigma, \Sigma_2, \Sigma_3, \Sigma_4]\}$ .

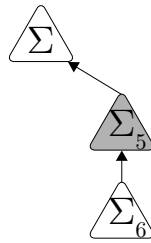


Fig. 4.2: Árbol de dialéctica de la Figura 4.1 sin las líneas de ataque

Análogamente a las operaciones de revisión descritas en el área de revisión de creencias, el paso inicial consiste en realizar una *selección* de aquello que será modificado para lograr la revisión efectiva. En este caso, se seleccionan las estructuras argumentales que interfieren con la garantía de la estructura raíz, para luego desactivarlas. Dado que puede haber más de una estructura candidata a ser desactivada, asumiremos un criterio de comparación entre conjuntos de argumentos que determina cuál provocaría menos cambio al ser desactivado. Dados dos conjuntos  $S_1$  y  $S_2$ , para significar que la desactivación de  $S_1$  provoca menos cambio



que la de  $S_2$ , la notación será:  $S_1 \prec S_2$ . Para comparar estructuras argumentales, utilizaremos el conjunto de argumentos que las componen. Si la desactivación de  $\Sigma_1$  provoca menos cambio que la desactivación de  $\Sigma_2$ :  $\text{args}(\Sigma_1) \prec \text{args}(\Sigma_2)$ . En esta presentación nos abstraeremos de este criterio de comparación. El lector interesado en este aspecto puede referirse al artículo [RMF<sup>+</sup>08a], donde se presentan dos criterios concretos para evaluar la cantidad de cambio: uno que minimiza el impacto sobre el árbol de dialéctica (útil cuando el árbol sea utilizado como una explicación de la garantía), y otro que minimiza el impacto en la base de conocimiento.

A continuación se define la *función de selección*, la cual se aplica sobre una línea de argumentación y retorna la estructura *con* cuya desactivación provoca menos cambio que la desactivación de cualquier otra estructura *con* en la línea.

**Definición 4.2 (Función de Selección)** *Dado un DAF  $F$ , la **función de selección de estructuras argumentales**  $\gamma : \text{lines}_F \rightarrow \text{st}_F$  es tal que  $\gamma(\lambda) = \Sigma$ , siendo  $\lambda \in \text{Att}(\mathbb{T}_F(\Sigma_1))$ , donde  $\Sigma_1$  es la raíz de  $\lambda$ ,  $\Sigma \in \lambda^-$ , y  $(\nexists \Sigma' \in \lambda^-) \text{args}(\Sigma') \prec \text{args}(\Sigma)$ .*

**Ejemplo 4.2** *A partir del árbol de dialéctica del Ejemplo 4.1 deben seleccionarse estructuras en cada una de las dos líneas de ataque:*

- En  $[\Sigma, \Sigma_1]$ , la selección es la única estructura *con*, i.e.,  $\Sigma_1$ ;
- En  $[\Sigma, \Sigma_2, \Sigma_3, \Sigma_4]$ , la selección puede ser tanto  $\Sigma_2$  como  $\Sigma_4$ .

La eliminación de una estructura *con* seleccionada deja un segmento superior que termina en una estructura *pro*. El siguiente lema establece que esto constituye una alteración válida de la línea.

**Lema 4.1** *Una línea de argumentación cuya hoja es una estructura argumental *pro* no es una línea de ataque.*

**DEMOSTRACIÓN:** Una línea  $\lambda$  con hoja pro tiene paridad impar y tomada individualmente respeta la secuencia de marcado  $U - D - U - \dots - U$ , y la raíz se encuentra garantizada. Por lo tanto, para que la raíz quede no garantizada, es estrictamente necesario considerar al menos una línea de argumentación adicional  $\lambda'$  tal que una estructura pro de  $\lambda$  quede derrotada por una estructura con de de  $\lambda'$  y la secuencia de marcado desde allí hasta la raíz (en ambas líneas) invierta sus valores. Luego, por minimalidad del conjunto de ataque,  $\lambda$  no es una línea de ataque.  $\square$

Una vez que se selecciona una estructura para ser desactivada, y análogamente con la teoría clásica de revisión, se requiere una función de incisión que determine qué porción de la estructura argumental será efectivamente desactivada, desactivando así a la estructura que la contiene.

**Definición 4.3 (Función de Incisión)** Dado un DAF  $F$ , la **función de incisión sobre estructuras argumentales**  $\sigma : \text{str}_F \rightarrow \mathbb{A}$  es tal que  $\sigma(\Sigma_1) = \mathcal{A}$ , donde  $\mathcal{A} \in \text{args}(\Sigma_1)$  tal que  $(\nexists \mathcal{A}' \in \text{args}(\Sigma_1)) \{\mathcal{A}'\} \prec \{\mathcal{A}\}$ .

Para desactivar una estructura, se utiliza una función de incisión que retorna el argumento activo perteneciente a esa estructura más apropiado a ser desactivado. Sin embargo, varias estructuras pueden tener argumentos en común dentro de un mismo árbol, por lo cual la desactivación de un argumento puede tener efectos colaterales, provocando la consecuente desactivación de otras estructuras.

**Definición 4.4 (Incisión Colateral)** Una incisión  $\sigma(\Sigma_1) = \mathcal{A}$  provoca una **incisión colateral** sobre una estructura  $\Sigma_2$  sssi  $\mathcal{A} \in \text{args}(\Sigma_2)$ . Si  $\mathcal{A} \notin \text{args}(\Sigma)$ , para toda  $\Sigma$  en  $\lambda^\uparrow(\Sigma_2)$ , entonces la incisión colateral es **superior** y se denota  $\sigma(\Sigma_1)^{\Sigma_2}$ .

En general sólo las incisiones colaterales superiores serán de impor-

tancia, dado que la desactivación colateral de una estructura hace que todo el subárbol enraizado en ella desaparezca del árbol original, incluyendo las incisiones colaterales “inferiores”. Por ende, toda estructura que esté por debajo dejará de influir en el estado de garantía de la estructura raíz del árbol. Las incisiones colaterales podrían comprometer la correctitud del resultado de la revisión. Para controlar esto, a continuación definiremos el principio de *preservación*. Este principio se ilustra en la Figura 4.3.

**(Preservación)** Si  $\sigma(\gamma(\lambda_1))^{\Sigma_2}$  entonces  $\gamma(\lambda_2)$  está en  $\lambda_2^\uparrow[\Sigma_2]$ ,  
 con  $\lambda_1, \lambda_2$  en  $\mathbb{T}_F(\Sigma)$  y  $\Sigma_2$  en  $\lambda_2$ .

El principio de preservación asegura que las incisiones colaterales no invalidarán la revisión. Para cada incisión colateral sobre una estructura  $\Sigma_2$  en una línea  $\lambda_2$ , este principio restringe la imagen de la función de selección en  $\lambda_2$  al segmento superior de  $\Sigma_2$ . De esta manera, si este segmento superior resultara dentro del conjunto de líneas a alterar por la revisión, la selección se comportaría de manera correcta.

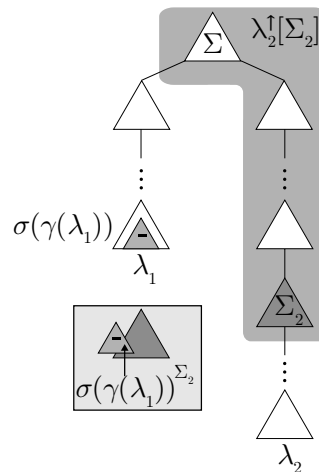


Fig. 4.3: Esquema del principio de preservación

De esta manera, todas las selecciones estarán en el segmento superior de las incisiones colaterales. El operador de revisión se encargará de “recolectar” todas las incisiones (sean colaterales o no) para modificar la teoría de acuerdo a la garantía que se desea alcanzar. El conjunto de líneas de argumentación a modificar se denominará *conjunto de alteración* (ver Definición 4.7).

Una cuestión que resulta elemental es el no producir incisiones colaterales sobre la raíz, ya que de esta manera sería imposible garantizarla. Para este aspecto, introducimos el principio de *preservación de la raíz*.

**(Preservación de la raíz)**  $\sigma(\gamma(\lambda_1))^{\Sigma}$  no se verifica,  
para toda línea  $\lambda_1$  perteneciente a  $\mathbb{T}_F(\Sigma)$

**Proposición 4.1** *Si una función de incisión verifica preservación, entonces verifica preservación de la raíz.*

**DEMOSTRACIÓN:** *Trivial, dado que el segmento superior de la raíz de un árbol de dialéctica deja un conjunto de argumentos con vacío y por lo tanto el consecuente de preservación no se verifica:  $\lambda^{\uparrow}[\Sigma] = [\Sigma]$ ,  $\lambda^{-} = \emptyset$ , y  $\gamma(\lambda) \in \lambda^{-}$ . Luego, el antecedente tampoco se cumple (por hipótesis toda la fórmula debe ser verdadera) y no hay incisión colateral sobre la raíz.*  
□

Una propiedad interesante que no siempre puede cumplirse es la de evitar toda ocurrencia de incisiones colaterales. Esto está caracterizado por el principio de *preservación estricta*.

**(Preservación estricta)**  $\sigma(\gamma(\lambda_1))^{\Sigma_1}$  no se verifica,  
para cualquier línea  $\lambda_1$  y estructura y  $\Sigma_1$  en  $\mathbb{T}_F(\Sigma)$

**Ejemplo 4.3** *Considerando la línea de ataque  $[\Sigma, \Sigma_2, \Sigma_3, \Sigma_4]$  del Ejemplo 4.2, si la selección fuese  $\Sigma_2$ , cualquiera sea el argumento a desactivar*

habrá una incisión colateral, ya sea sobre  $\Sigma_1$  o sobre  $\Sigma_3$ . En cambio, si la selección fuera  $\Sigma_4$ , la desactivación del argumento en esta estructura no provocaría incisión colateral alguna, respetando preservación estricta.

**Proposición 4.2** *Si una función de incisión verifica preservación estricta, entonces verifica preservación.*

DEMOSTRACIÓN: *Trivial, ya que la no ocurrencia de incisiones colaterales hace que el antecedente de la fórmula de preservación sea falso, tornando verdadera la fórmula completa.  $\square$*

**Definición 4.5 (Función de Incisión Garantizante)** *Una función de incisión “ $\sigma$ ” se dice **garantizante** sssi cumple con el principio de preservación.*

Al buscar garantía para una estructura  $\Sigma$  debe alterarse el árbol de dialéctica enraizado en ella. Esto es, algunas líneas de argumentación deben ser incididas para que el árbol resultado sea garantizante. La ocurrencia de incisiones colaterales es la única problemática en este proceso, cuyo análisis requiere considerar un *árbol hipotético*, para poder llevar cuenta de las incisiones hipotéticas que podarían el árbol original. A continuación se define este árbol hipotético, y luego se lo utilizará para poder definir el *conjunto de alteración* de líneas de un árbol dado.

**Definición 4.6 (Árbol Hipotético)** *Dado un DAF  $F$ , un conjunto de argumentos  $\Omega \subseteq \mathbb{A}$ , y un árbol de dialéctica activo  $\mathbb{T}_F(\Sigma)$  con un conjunto  $S$  de líneas de argumentación, el **árbol hipotético**  $\text{htree}(\mathbb{T}_F(\Sigma), \Omega)$  es un árbol construido a partir del siguiente conjunto de líneas:*

$$\{\lambda^\dagger(\Sigma_1) \mid \lambda \in S, \Omega \cap \text{args}(\Sigma_1) \neq \emptyset, (\nexists \Sigma_2 \text{ en } \lambda^\dagger(\Sigma_1)) \Omega \cap \text{args}(\Sigma_2) \neq \emptyset\}$$

En palabras, para calcular el árbol hipotético de un árbol  $\mathcal{T}$  dado un

conjunto de argumentos  $\Omega$ , se asume que estos argumentos serán desactivados y se poda  $\mathcal{T}$  de acuerdo a las estructuras que se ven afectadas, quedando los segmentos superiores de las mismas.

**Ejemplo 4.4** Considerando el árbol de dialéctica del Ejemplo 4.1 y el conjunto de argumentos  $\Omega = \{\langle\{p\}, y\rangle, \langle\{p\}, x\rangle\}$ , las estructuras argumentales afectadas son  $\Sigma_2$ ,  $\Sigma_3$  y  $\Sigma_6$ , por lo cual sólo sus segmentos superiores (en las respectivas líneas) permanecen en el árbol hipotético. Esto está ilustrado en la Figura 4.4(a). Debe observarse que  $\Sigma_2$  y  $\Sigma_3$  pertenecen a la misma línea, pero el segmento superior que queda en el árbol hipotético es el de  $\Sigma_2$ ; por último,  $\Sigma_4$  no forma parte del árbol hipotético. En la Figura 4.4(b) puede verse el árbol hipotético marcado luego de considerar la posibilidad de desactivar los argumentos en  $\Omega$ .

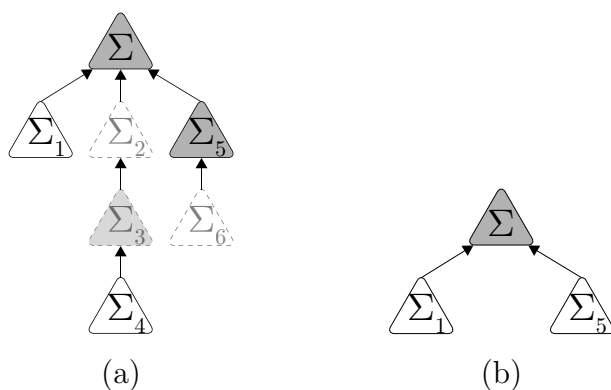


Fig. 4.4: Árbol hipotético del Ejemplo 4.4

A continuación se define el conjunto de líneas que serán efectivamente alteradas por el mecanismo de revisión. Este conjunto resulta sencillo de definir a partir del concepto de árbol hipotético, ya que las incisiones colaterales pueden provocar nuevas líneas de ataque, y éstas sólo serán visibles en el árbol hipotético.

**Definición 4.7 (Conjunto de Alteración)** *Dado un DAF  $F$  y un árbol de dialéctica  $\mathbb{T}_F(\Sigma)$  con un conjunto  $S$  de líneas de argumentación, el **conjunto de alteración**  $\Lambda_F(\Sigma)$  de  $\mathbb{T}_F(\Sigma)$  es el menor punto fijo del operador  $\ell_F(\Sigma)$ :*

$$\ell_F(\Sigma)^0 = \text{Att}(\mathbb{T}_F(\Sigma))$$

$$\ell_F(\Sigma)^k = \ell_F(\Sigma)^{k-1} \cup$$

$$\{\lambda \in S \mid (\exists \Sigma_1) \lambda^\uparrow(\Sigma_1) \in \text{Att}(\text{htree}(\mathbb{T}_F(\Sigma), \bigcup_{\lambda \in \ell_F(\Sigma)^{k-1}} \sigma(\gamma(\lambda))))\}$$

El operador de punto fijo comienza con aquellas líneas que fueron inicialmente identificadas como de ataque. Luego, en cada iteración, se agregan aquellas líneas conteniendo algún segmento superior perteneciente al conjunto de ataque del árbol hipotético producto de todas las incisiones hechas hasta la iteración anterior. De esta manera, se incluyen aquellas líneas de argumentación afectadas por incisiones colaterales cuyo segmento superior remanente compromete la garantía de la estructura en la raíz del árbol.

**Ejemplo 4.5** *Consideremos el árbol de dialéctica activo  $\mathbb{T}_F(\Sigma)$  en el contexto de un DAF  $F$ , ilustrado en la Figura 4.5(a). La primer iteración del operador  $\ell_F(\Sigma)$  incluye la única línea de ataque:  $[\Sigma, \Sigma_2, \Sigma_5, \Sigma_7]$ . Luego la segunda iteración considera cada línea del árbol hipotético producto de incidir las selecciones de la iteración anterior. Suponiendo que la estructura seleccionada es  $\Sigma_7$  y la incisión se realiza sobre el argumento  $\mathcal{B}$  en ella, el árbol hipotético será  $\text{htree}(\mathbb{T}_F(\Sigma), \{\mathcal{B}\})$ , ilustrado en la Figura 4.5(b).*

*Dado que el segmento superior propio de  $\Sigma_6$  es  $[\Sigma, \Sigma_3]$  y éste pertenece al conjunto de ataque del árbol hipotético (notar que su remoción logra garantizar a  $\Sigma$ ), la segunda iteración del operador incorpora la línea completa, i.e.,  $[\Sigma, \Sigma_3, \Sigma_6]$  al conjunto de alteración. Se considera la línea*

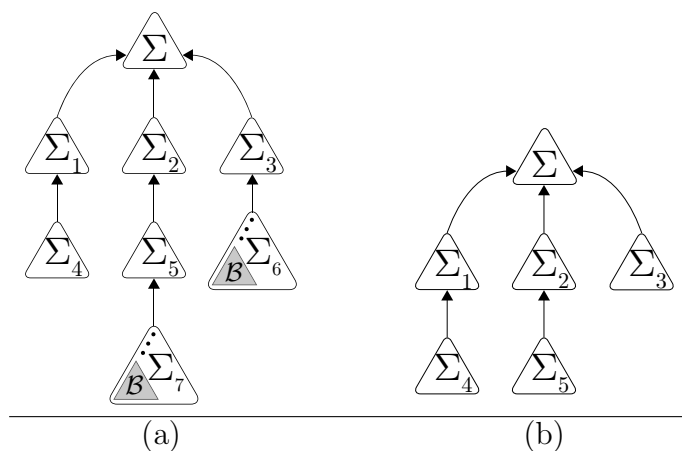
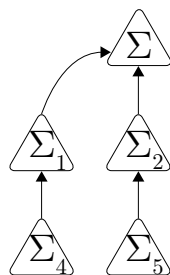


Fig. 4.5: (a) Árbol de dialéctica del Ejemplo 4.5 (b) Árbol hipotético

completa porque el conjunto de alteración calculado se refiere al árbol de dialéctica original; cada iteración se adelanta a los efectos colaterales que ocurrirán, incorporando líneas de este árbol hasta que no haya nuevas.

La tercer iteración evaluará la incisión sobre la selección en  $[\Sigma, \Sigma_3, \Sigma_6]$ . La única posibilidad es realizar una incisión sobre  $\Sigma_3$ ; asumiendo que esto no provoca ninguna incisión colateral, el segmento superior de  $\Sigma_3$  es  $[\Sigma]$  y no hay líneas de ataque en el árbol hipotético:



Por ende, el operador encontró un punto fijo, y el conjunto de alteración de  $\mathbb{T}_F(\Sigma)$  es  $\Lambda_F(\Sigma) = \{[\Sigma, \Sigma_2, \Sigma_5, \Sigma_7], [\Sigma, \Sigma_3, \Sigma_6]\}$  y ambas líneas serán contempladas cuando el operador de revisión haga efectivas las incisiones.

Al momento de activar una estructura argumental  $\Sigma$  la única alter-



nativa viable es incorporar las premisas de  $\Sigma$  al conjunto de evidencia del DAF, ya que la adición de otros argumentos en soporte de las premisas insatisfechas de  $\Sigma$  podría no lograr su activación, requiriendo la añadidura de la evidencia necesaria para estos argumentos.

**Definición 4.8 (Activación de una Estructura Argumental)**

Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$ , la **activación de una estructura argumental**  $\Sigma \in \text{str}_F$  se logra mediante el agregado múltiple del conjunto de evidencia  $\text{pr}(\Sigma)$ , obteniendo el DAF  $F \oplus \Sigma = \langle \mathbf{E} \cup \text{pr}(\Sigma), W, \bowtie, \text{pref} \rangle$ .

Existen dos formas de desactivar un argumento: (1) retirar parte de la evidencia que lo soporta, (2) simplemente quitar el argumento del conjunto de trabajo. Dado que un argumento puede estar activo a partir de otros argumentos activos, una función que remueva la evidencia que lo soporta debería buscarla dentro de la estructura que tiene al argumento como tope; incluso puede darse el caso de que ninguna de las premisas del argumento esté soportada por evidencia, y acceder a la estructura asociada se torne imprescindible. Para lograr esto, debería definirse una función de incisión a nivel de evidencia. En esta tesis nos volcaremos por la segunda opción, y la desactivación de argumentos se llevará a cabo a través de la remoción de argumentos del conjunto de trabajo.

**Definición 4.9 (Contracción Garantizante)** El **operador de contracción garantizante** “ $-\omega$ ” de un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  por una estructura argumental  $\Sigma$  de  $F$  se define como:

$$F -\omega \Sigma = \langle \mathbf{E}, W \setminus \bigcup_{\lambda \in \Lambda_F(\Sigma)} (\sigma(\gamma(\lambda))), \bowtie, \text{pref} \rangle,$$

donde “ $\sigma$ ” es una función de incisión garantizante.

**Definición 4.10 (Revisión Garantizante)** *El operador de revisión garantizante “ $*^\omega$ ” de un DAF  $F$  por una estructura argumental  $\Sigma$  de  $F$  se define como:*

$$F *^\omega \Sigma = (F \oplus \Sigma) -^\omega \Sigma,$$

donde “ $\oplus$ ” es el operador de activación de una estructura argumental.

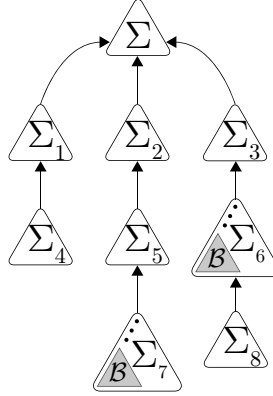
**Lema 4.2** *Si  $F' = F *^\omega \Sigma$  entonces  $\text{Att}(\mathbb{T}_{F'}(\Sigma)) = \emptyset$ .*

DEMOSTRACIÓN: *Supongamos por el absurdo que  $\text{Att}(\mathbb{T}_{F'}(\Sigma)) \neq \emptyset$ . Sin pérdida de generalidad podemos asumir que  $\text{Att}(\mathbb{T}_{F'}(\Sigma)) = \{\lambda\}$ . Sea  $F_1 = F \oplus \Sigma$ , entonces  $\lambda = \lambda_1^\uparrow[\Sigma_1]$ , para alguna estructura  $\Sigma_1$  en  $\lambda_1$ . (En particular, podría darse el caso  $\lambda = \lambda_1$ .) Por definición de conjunto de alteración, toda línea cuya alteración (i.e., el segmento superior luego de una selección e incisión) resulte en una línea de ataque en el árbol hipotético formará parte de  $\Lambda_{F_1}(\Sigma)$ ; en particular,  $\Lambda_{F_1}(\Sigma)$  incluirá a  $\lambda_1$ , ya que se asume que  $\lambda$  es una línea de ataque luego de la revisión. Luego, por definición de contracción garantizante, todas las líneas en  $\Lambda_{F_1}(\Sigma)$  serán afectadas por una incisión garantizante. La incisión sobre una estructura en  $\lambda_1$  resultará en que su segmento superior  $\lambda$  pertenezca al árbol producto de la revisión  $\mathbb{T}_{F'}(\Sigma)$ , pero por el Lema 4.1 la incisión garantizante de una selección en una línea produce segmentos superiores que no son líneas de ataque, i.e.,  $\lambda$  no es una línea de ataque. El absurdo provino de suponer que lo era.  $\square$*

**Teorema 4.1** *Dado un DAF  $F$  y un operador de revisión “ $*^\omega$ ”, una estructura argumental  $\Sigma$  está garantizada en el DAF  $F *^\omega \Sigma$ .*

DEMOSTRACIÓN: *Trivial por Definición 4.1 con  $\text{Att}(\mathbb{T}_F(\Sigma)) = \emptyset$  y Lema 4.2.  $\square$*

**Ejemplo 4.6** Consideremos una revisión garantizante efectuada sobre un DAF  $F' = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  por una estructura  $\Sigma$ . El siguiente árbol de dialéctica  $\mathbb{T}_F(\Sigma)$  está construido en el contexto del DAF  $F = F' \oplus \Sigma = \langle \mathbf{E} \cup \text{pr}(\Sigma), W, \bowtie, \text{pref} \rangle$ :



Las líneas se denominarán:

- $\lambda_1 = [\Sigma, \Sigma_1, \Sigma_4]$ ;
- $\lambda_2 = [\Sigma, \Sigma_2, \Sigma_5, \Sigma_7]$ ;
- $\lambda_3 = [\Sigma, \Sigma_3, \Sigma_6, \Sigma_8]$ .

El conjunto de ataque es  $\text{Att}(\mathbb{T}_F(\Sigma)) = \{\lambda_2, \lambda_3\}$ . Supongamos que inicialmente las selecciones son:

- $\gamma(\lambda_1) = \Sigma_1$ , ya que no hay otras estructuras con en  $\lambda_1$ ;
- $\gamma(\lambda_2) = \Sigma_7$ , ya que  $\Sigma_7 \prec \Sigma_2$ ;
- $\gamma(\lambda_3) = \Sigma_8$ , ya que  $\Sigma_8 \prec \Sigma_3$ .

Asumiendo la incisión  $\sigma(\Sigma_7) = \mathcal{B}$ , hay una incisión colateral  $\sigma(\Sigma_7)^{\Sigma_6}$ , por lo cual el principio de preservación restringe la imagen de la selección en  $\lambda_3$  al segmento superior  $[\Sigma, \Sigma_3, \Sigma_6]$ . Esto provoca que deba realizarse

una nueva selección para satisfacer dicho principio, ya que la revisión se define a través del uso de incisiones garantizantes (ver Definiciones 4.5 y 4.10). La nueva selección en  $\lambda_3$  puede ser sólo  $\Sigma_3$ .

El conjunto de alteración en este caso es igual al conjunto de ataque, debido a que la incisión colateral se provocó sobre una línea que ya era de ataque, y ninguna línea nueva se incorpora en la segunda iteración del operador. Si consideramos que  $\sigma(\Sigma_3) = \mathcal{C}, \mathcal{C} \notin \text{args}(\Sigma_{1 \leq i \leq 7})$ , la revisión concluye luego de efectuar dicha incisión, junto a la incisión sobre  $\Sigma_7$ :

$$F_R = F' *^\omega \Sigma = (F' \oplus \Sigma) -^\omega \Sigma = \langle \mathbf{E}, W \cup \text{args}(\Sigma) \setminus \{\mathcal{B}, \mathcal{C}\}, \boxtimes, \text{pref} \rangle$$

El árbol de dialéctica  $\mathbb{T}_{F_R}(\Sigma)$  es el ilustrado en la Figura 4.6.

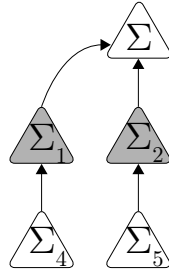


Fig. 4.6: Árbol producto de la revisión del Ejemplo 4.6

En cuanto a trabajo relacionado al formalismo de ATC, en el artículo [CdLS08] los autores estudian propiedades desde el punto de vista del cambio en el conjunto de extensiones en un sistema argumentativo, partiendo de la adición de un argumento al mismo. En este aspecto imponen una fuerte restricción: el argumento añadido debe tener *a lo sumo* una interacción (vía ataque) con un argumento preexistente en el sistema. Esta restricción (la cual no es requerida en nuestro trabajo), simplifica enormemente el problema de la revisión, ya que la dificultad reside en las múltiples interacciones con el sistema original que un argumento nue-

vo podría tener. Más aún, debido a que ATC está basado en el DAF, debe considerarse la dificultad adicional de aquellos argumentos que pertenecen a más de una estructura argumental. De esta forma, la simple activación de un argumento para alcanzar la garantía (*i.e.*, el objetivo de la revisión) puede hacer que múltiples nuevas estructuras argumentales emerjan e interactúen con el sistema original. Otra diferencia entre ambos enfoques reside en el modo de evaluación de la garantía: mientras que en ATC la misma se computa sobre un argumento a partir de árboles de dialéctica, en [CdLS08] el cambio se estudia desde el punto de vista de las extensiones (*i.e.*, conjuntos de argumentos garantizados) en un grafo de argumentos.

#### 4.2. Expansión Garantizante de Estructuras Argumentales

Complementariamente al enfoque explicado previamente, puede definirse un operador que alcance la garantía de una estructura argumental sólo mediante expansiones. En este caso, la garantía se logra a través de activación de derrotadores en posiciones específicas del árbol de dialéctica enraizado en la estructura en cuestión. A continuación describiremos someramente la definición de un operador de revisión sobre DAFs respondiendo a estas características. Estas nociones fueron dadas más detalladamente en [MRF<sup>+</sup>09].

Si bien este enfoque se define en forma análoga a su versión por desactivación de argumentos, existen algunas diferencias que resultan significativas. La diferencia más notoria es que, en la revisión por desactivación, el éxito puede asegurarse, ya que, en última instancia, las incisiones colaterales llevan a la poda total del árbol, dejando sólo a la raíz, y garantizándola. Al contrario de esto, en la revisión por activación existe la posibilidad de que simplemente no haya un derrotador disponible para

ser activado.

La mecánica de la revisión por expansión incluye la detección de las líneas de ataque, y en ellas la selección de un argumento *con* al cual se le activará un derrotador. La activación de estructuras argumentales se realiza simétricamente: se detecta qué argumentos deberían activarse, y se expande el conjunto de evidencia en concordancia con ello. Esto se denomina *emparchado* de estructuras argumentales y provoca el efecto contrario a la incisión.

El procedimiento es simple, salvo por la ocurrencia de efectos colaterales. Al contrario de lo que ocurre al desactivar argumentos, la activación de derrotadores implica el posible brote de nuevos subárboles: aquellos enraizados en estos nuevos derrotadores, y los que surjan de la activación colateral de otras estructuras argumentales. Esto presenta una nueva dificultad; mientras que en el enfoque por desactivación los árboles de dialéctica hipotéticos presentaban igual o menos información que el árbol original, en este enfoque debe contemplarse la aparición de nuevas estructuras argumentales a lo largo de todo el árbol. Para lidiar con esto se utiliza una versión adaptada del principio de *preservación*. Adicionalmente, dado que las activaciones pueden encadenarse, se introduce el principio de *no acumulación*, para evitar que una misma línea sea extendida más de una vez por sucesivas activaciones colaterales. La definición de este operador de revisión da lugar a la definición del operador híbrido.

### 4.3. Un Enfoque Híbrido a través de Expansiones y Contracciones de Estructuras Argumentales

Existen casos en los cuales alcanzar garantía a través de un único método resulta demasiado restrictivo, provocando así más cambio que el

que resultaría de su combinación. Sin embargo, la complicación reside en decidir cuál es la combinación correcta de métodos tal que el cambio se minimice. Un operador híbrido haría uso de las operaciones descritas anteriormente para lograr tal objetivo. En términos generales, este operador analiza cada línea de ataque, y decide qué tipo de selección hacer, comparando los enfoques por activación y desactivación en cuanto a la cantidad de cambio que provocarían. Luego, en función de esto, se realiza una incisión o un emparchado. La combinación de enfoques conlleva un problema que anteriormente no estaba presente: la aplicación de uno de los métodos en una línea puede anular colateralmente lo logrado por el método complementario en otra línea, *i.e.*, una incisión podría resultar emparchada, o *vice versa*. Esta situación escapa a lo que puede ser controlado por preservación, por lo cual un nuevo principio es necesario. A tales efectos, se define *progreso*; este principio controla que la activación de una estructura argumental no sea deshecha por una desactivación colateral, y *vice versa*.

#### 4.4. Revisión del DAF por la Relación de Derrota

La variación del conjunto de trabajo de argumentos y de la evidencia disponible repercuten en el conjunto de estructuras argumentales disponibles para efectuar razonamiento. Adicionalmente a esto, también puede concebirse la modificación de la relación de derrota entre argumentos. Esto involucraría:

- Modificar la noción de conflicto entre argumentos: dos argumentos que en un momento dado no eran conflictivos podrían comenzar a estar en conflicto en respuesta a la situación actual.
- Reflejar un cambio en las preferencias: una relación de derrota entre

dos argumentos podría invertirse, o convertir una situación de no preferencia podría pasar a tener una resolución.

La modificación de la relación de conflicto debería realizarse cuidadosamente, ya que está directamente relacionada al lenguaje de premisas y conclusiones. En aquellos casos en los que la contradicción esté dada por conclusiones complementarias, el conflicto no puede simplemente “desaparecer”. Por ende, la modificación de esta relación actuaría por fuera de este conjunto.

En cuanto al cambio en la función de preferencia, la revisión tiene como objetivo de reflejar el cambio global en las preferencias que logra garantizar cierto argumento. Este tipo de revisión podrá invertir la preferencia o generar un bloqueo, pero no eliminar el conflicto.

Utilizando este último tipo de operación es posible emular los tres tipos de revisión vistos anteriormente. Para cada línea de ataque se hará una selección. Si el método elegido es la desactivación, se invertirá la preferencia en el ataque de forma tal que la estructura argumental seleccionada pase a estar derrotada y quede fuera de la línea. Por otra parte, si se siguiera el método por activación, se buscará una estructura que, por estar derrotada por la estructura seleccionada, actualmente esté fuera de la línea de argumentación, y se invertirá la preferencia para que este nuevo derrotador forme parte de la misma.

En ambos casos pueden darse efectos colaterales. A diferencia de los métodos por contracción y expansión de argumentos, cualquier inversión en preferencias puede provocar colateralidades de ambos tipos: pueden podarse algunas ramas, así como pueden brotar nuevos subárboles. En este sentido, este enfoque presenta mayor dificultad, ya que debe delinearse un principio de preservación que capture ambos efectos.



## 5. REFINANDO OPERACIONES DE CAMBIO

Al contar con un sistema argumentativo dinámico, es deseable contar con mecanismos que amortigüen el impacto que los cambios en el mundo producen en el conjunto de garantías del marco. En este capítulo los cambios en el mundo se procesarán en dos niveles: primeramente, en la Sección 5.1 se verificará qué garantías deberían recomputarse, estableciendo “dependencias” entre los estados de garantía de las estructuras argumentales con respecto al cambio realizado en el conocimiento; en segundo término, en la Sección 5.2 el recómputo de garantías se realizará con la asistencia de un mecanismo de poda. Para facilitar el análisis, nos restringiremos a estudiar estos aspectos bajo la semántica de árboles de dialéctica con marcado escéptico.

### 5.1. *Relevancia del Cambio*

Puesto que algunas piezas de evidencia no están relacionadas a ciertos argumentos, esta asociación puede utilizarse para estar al tanto del subconjunto relevante de evidencia para cada estructura argumental. El reconocer esta situación representa una oportunidad para mejorar la computación de garantía, no sólo reduciendo el conjunto de argumentos a considerar, sino también para reconocer cuándo un cambio no afecta al estado de garantía de ciertas estructuras. Debe notarse que la asociación se realiza a partir del conjunto de trabajo de argumentos, y luego se restringe a los activos y a la evidencia disponible.

El árbol de dialéctica potencial agrupa todos aquellos argumentos que (directa o indirectamente) están relacionados con la raíz vía ataque, cualquiera sea el rol que cumplan (*pro* o *con*). El carácter de in/activos de los nodos de un árbol potencial tendrá incidencia directa sobre el estado de garantía de la raíz. Para simplificar las definiciones que siguen, asumiremos una función  $\text{inTree} : \mathbf{trees}_F \rightarrow 2^{\text{st}_F}$  que retorna el conjunto de todas las estructuras argumentales de un árbol de dialéctica en el contexto de un DAF  $F$ , donde  $\mathbf{trees}_F$  es el conjunto de todos los árboles de dialéctica que se pueden construir a partir de las líneas (exhaustivas) en  $\mathbf{lines}_F$ .

**Definición 5.1 (Nube de Premisas)**

Dado un árbol de dialéctica  $\mathcal{T}_F(\Sigma)$  en el contexto de un DAF  $F$ , la **nube de premisas** de  $\Sigma$  es  $\text{cloud}(\Sigma, F) = \{\text{pr}(\Sigma_i) \mid \Sigma_i \in \text{inTree}(\mathcal{T}_F(\Sigma))\}$ .

La nube de premisas de una estructura argumental contiene toda posible pieza de evidencia relacionada con el estado de garantía de esa estructura. Debe notarse que las premisas intermedias de las estructuras argumentales también son consideradas, por el carácter exhaustivo del árbol potencial (ver Ejemplo 5.1). Más aún, dado que los árboles de dialéctica sólo contienen estructuras argumentales bien formadas, se considerarán únicamente aquellas premisas intermedias que podrían ser parte de una estructura activa bien formada.

**Ejemplo 5.1** A partir del árbol potencial para la estructura argumental  $\Sigma_0$  en el contexto del DAF  $\langle \mathbf{E}_{2,7}, W_{2,7}, \bowtie_{2,7}, \mathbf{pref}_{2,7} \rangle$  del Ejemplo 2.7 (ver Figura 5.1), tenemos que la nube de premisas de  $\Sigma_0$  es el conjunto  $\{y, z, b, c, d, e, f, g, h\}$ .

Si modificamos la relación de conflicto:  $\bowtie_{5,1} = \{(\mathcal{B}, \mathcal{C})\} \cup \bowtie_{2,7}$  y asumimos  $\mathbf{pref}_{5,1}(\mathcal{B}, \mathcal{C}) = \mathcal{B}$ , entonces el nuevo árbol potencial es el de la

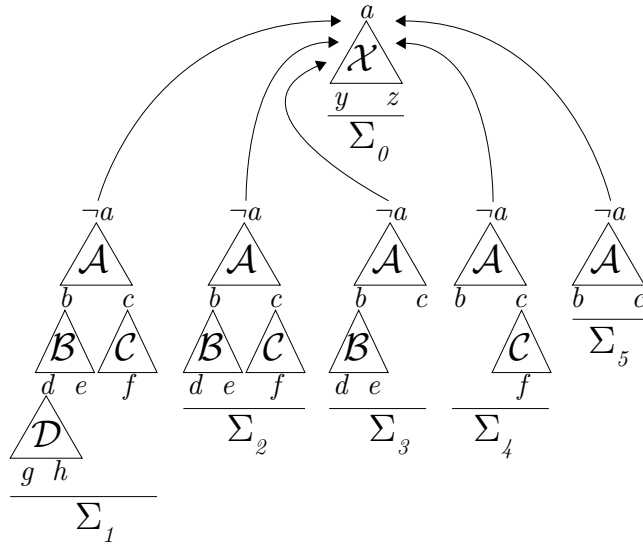


Fig. 5.1: Árbol de dialéctica potencial del Ejemplo 2.7

Figura 5.2, y la nueva nube de premisas de  $\Sigma_0$  es  $\{y, z, b, c, f, d, e\}$ , dejando de lado aquellas premisas que forman parte de argumentos que ya no pertenecen a estructuras bien formadas.

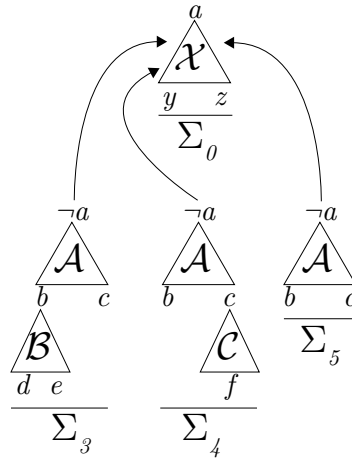


Fig. 5.2: Árbol potencial considerando la relación de conflicto  $\bowtie_{5,1}$

**Definición 5.2 (Independencia de las Premisas)** Dado un DAF  $F$

y dos estructuras argumentales  $\Sigma_1$  y  $\Sigma_2$  en  $F$  tal que  $\text{cloud}(\Sigma_1, F) \cap \text{cloud}(\Sigma_2, F) = \emptyset$ , se dice que  $\Sigma_1$  y  $\Sigma_2$  son **independientes con respecto a las premisas**.

La noción de independencia de las premisas es útil para poder establecer dependencias entre los estados de garantía de las distintas estructuras argumentales que emergen de un marco argumentativo. Más aún, la nube de premisas de una estructura argumental podría utilizarse como una herramienta de depurado para detectar dependencias no deseadas. De la misma manera podría utilizarse el árbol de dialéctica potencial.

**Proposición 5.1** *Dado un DAF  $F$ , si  $\Sigma_1$  y  $\Sigma_2$  son dos estructuras argumentales en  $F$  independientes con respecto a las premisas, entonces  $\text{inTree}(\mathbb{T}_F(\Sigma_1)) \cap \text{inTree}(\mathbb{T}_F(\Sigma_2)) = \emptyset$ , cualquiera sea el conjunto de evidencia de  $F$ .*

DEMOSTRACIÓN: *Por hipótesis,  $\text{cloud}(\Sigma_1, F) \cap \text{cloud}(\Sigma_2, F) = \emptyset$ . Supongamos por el absurdo que existe una estructura argumental  $\Sigma$  tal que pertenece a ambos árboles potenciales, i.e.,  $\text{inTree}(\mathbb{T}_F(\Sigma_1)) \cap \text{inTree}(\mathbb{T}_F(\Sigma_2)) \supseteq \{\Sigma\}$ . Por definición de nube de premisas,  $\text{pr}(\Sigma) \in \text{cloud}(\Sigma_1, F)$  y  $\text{pr}(\Sigma) \in \text{cloud}(\Sigma_2, F)$ . Lo cual significa que  $\text{cloud}(\Sigma_1, F) \cap \text{cloud}(\Sigma_2, F) \supseteq \{\Sigma\}$ , violando la hipótesis. El absurdo provino de suponer que existe una estructura argumental que está en ambos árboles. Dado que la nube de premisas está definida sobre árboles potenciales y el árbol activo asociado a cada árbol potencial tiene un subconjunto de sus nodos y arcos, la propiedad se cumple para todo árbol activo, i.e., es independiente del conjunto de evidencia de  $F$ .  $\square$*

Debe notarse que la inversa de esta proposición no se verifica, ya que dos estructuras podrían no compartir argumentos en sus árboles potenciales, y aún así tener nubes de evidencia que se intersecten.

Un cambio aplicado a un DAF que no involucre a una estructura argumental en el árbol de dialéctica potencial para una estructura  $\Sigma$  no afecta el estado de garantía de  $\Sigma$ .

**Lema 5.1** *Dado un DAF  $F = \langle \mathbf{E}, W, \bowtie, \text{pref} \rangle$  definido sobre un lenguaje  $\mathcal{L}$ , una estructura argumental  $\Sigma$  garantizada en  $F$  de acuerdo a una semántica  $S$ , si se efectúa la operación de cambio  $F' = F \text{ op } \Gamma$ , donde:*

1.  $\Gamma \subseteq \mathcal{L}$  es un conjunto de evidencia tal que  $\Gamma \cap \text{cloud}(\Sigma, F) = \emptyset$ , y “op” es la operación de borrado/agregado múltiple de evidencia, o
2.  $\Gamma \subseteq W$  es un conjunto de argumentos de  $F$  tal que  $\Gamma \cap \text{inTree}(\mathbb{T}_F(\Sigma)) = \emptyset$  y  $\text{op} = [+|-]$  es la expansión priorizada o la contracción por un conjunto de argumentos, o
3.  $\Gamma \subseteq W \times W$  es un conjunto de conflictos sobre pares de argumentos en  $F$  tal que  $(\forall (\mathcal{A}_1, \mathcal{A}_2) \in \Gamma) \mathcal{A}_{i=1,2} \notin \text{inTree}(\mathcal{T}_F(\Sigma))$  y  $\text{op} = [\overset{\bowtie}{+} | \overset{\bowtie}{-}]$  es la expansión o la contracción por un conjunto de conflictos,

entonces  $\Sigma$  está garantizada en  $F'$  bajo  $S$ , cualquiera sea el conjunto de evidencia de  $F$ .

DEMOSTRACIÓN: *La garantía de  $\Sigma$  proviene del marcado de su árbol activo  $\mathbb{T}_F(\Sigma)$ , el cual contiene sólo estructuras activas, determinadas por la evidencia actual  $\mathbf{E}_1 \subseteq \text{cloud}(\Sigma, F)$ .*

1. *Al considerar el borrado múltiple de evidencia por  $\Gamma$ , tenemos que  $\text{cloud}(\Sigma, F) \setminus \Gamma = \text{cloud}(\Sigma, F)$ , luego  $\mathbf{E}_1 \subseteq \text{cloud}(\Sigma, F) \setminus \Gamma$  y  $\Sigma$  está garantizado en  $F'$ . Si consideramos el agregado múltiple de evidencia, tenemos que  $\text{cloud}(\Sigma, F) \cup \Gamma \supseteq \text{cloud}(\Sigma, F)$ , luego  $\mathbf{E}_1 \subseteq \text{cloud}(\Sigma, F) \cup \Gamma$  y  $\Sigma$  está garantizada en  $F'$ .*

2. Al considerar la contracción, tenemos que  $\text{inTree}(\mathbb{T}_F(\Sigma)) \setminus \Gamma = \text{inTree}(\mathbb{T}_F(\Sigma))$ , luego el árbol activo para  $\Sigma$  no se ve modificado:  $\mathbb{T}_{F'}(\Sigma) = \mathbb{T}_F(\Sigma)$  y  $\Sigma$  está garantizada en  $F'$ . Si consideramos la expansión, tenemos que  $\text{inTree}(\mathbb{T}_F(\Sigma)) \cup \Gamma \supseteq \text{inTree}(\mathbb{T}_F(\Sigma))$ , pero  $(\text{inTree}(\mathbb{T}_F(\Sigma)) \cup \Gamma) \cap \mathcal{T}_F(\Sigma) = \text{inTree}(\mathbb{T}_F(\Sigma))$ , ya que los argumentos que se agregan están fuera del árbol potencial  $\mathcal{T}_F(\Sigma)$ . Luego,  $\Sigma$  está garantizada en  $F'$ .
3. Análogamente al caso anterior, los arcos que agrega o remueve la expansión o contracción por un conjunto de conflictos no afectarán al árbol potencial para  $\Sigma$  en  $F$ , por lo cual el árbol potencial en  $F'$  se mantendrá sin cambios. Luego, a la luz de la evidencia,  $\mathbb{T}_F(\Sigma) = \mathbb{T}_{F'}(\Sigma)$  y  $\Sigma$  está garantizada en  $F'$ .

Por lo tanto, al aplicar alguna de esas operaciones bajo las condiciones establecidas para cada una de ellas,  $\Sigma$  estará garantizada en  $F'$ .  $\square$

Al momento aplicar una operación argumental de cambio, el uso de estos resultados será fundamental para simplificar el problema, ya que podría evitar la recomputación de garantía, similar a la propuesta dada en [CdLS08], la cual se describe someramente en la Sección 4.1.

## 5.2. Bonsai de Dialéctica

Gran parte de la literatura acerca de argumentación en inteligencia artificial se ha dedicado a extender el marco argumentativo de Dung, proveyéndole una mayor capacidad expresiva. Sin embargo, muchas de las dificultades subyacentes en el costado práctico de la teoría argumentativa no han sido muy estudiadas. La implementación de marcos argumentativos aún no ha alcanzado la madurez, y los pocos sistemas disponibles

aún no han sido probados con grandes bases de conocimiento. De todas formas, esto es entendible para una disciplina nueva.

En esta sección se propone una metodología para lidiar con grandes bases de argumentos, utilizando árboles de dialéctica para calcular garantía. Esta metodología constituye trabajo en curso y aún no publicado, realizado con Sebastián Gottifredi, y su implementación se encuentra en etapa experimental; los resultados obtenidos hasta el momento corresponden a un prototipo en desarrollo.

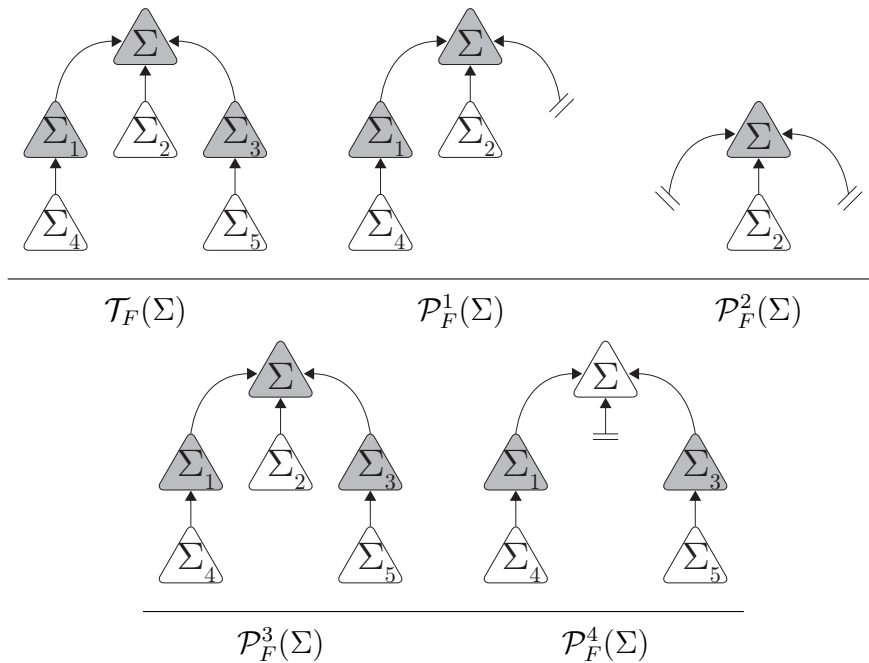
Un enfoque directo para acelerar este cálculo es construir los árboles más pequeños posibles tal que permitan establecer el estado de la raíz. Esto puede lograrse a través de una *técnica de poda*. Para maximizar las oportunidades de poda esta técnica se basará en la utilización de una medida de *fuerza* de las estructuras argumentales para guiar la evaluación del árbol. Al resultado de la poda lo llamaremos *bonsai de dialéctica*; esta metáfora reside en que intentamos lograr árboles pequeños que retienen las propiedades del árbol original.

Dadas las características de la función de marcado, utilizaremos una técnica de poda *and-or* como la que utiliza DELP [CSG00], la cual denominaremos *poda 1U* (el concepto formal es introducido en la Definición 5.5): *Cuando un derrotador para una estructura argumental  $\Sigma$  es marcado como **no derrotado**,  $\Sigma$  puede ser marcado directamente como **derrotado**, ignorando el resto de los derrotadores.*

Aunque esta técnica de poda no resulta muy restrictiva e incluso con un recorrido a ciegas podría ser utilizada frecuentemente dentro de un mismo árbol, encontrar todas las oportunidades de poda sin guía alguna resulta una cuestión de azar, ya que los derrotadores no derrotados deben ser encontrados en primera instancia, antes que sus hermanos (ver Ejemplo 5.2). Generalmente en la literatura, la selección de derrotadores

al construir árboles o grafos no sigue un criterio determinado; esto se da no sólo en los enfoques prácticos, sino en la teoría también. En las implementaciones basadas en reglas, éstas se encuentran ubicadas arbitrariamente y los contra-argumentos se construyen recorriendo las reglas de arriba hacia abajo. En los enfoques teóricos, simplemente se cuenta con un conjunto de argumentos del cual se eligen los contra-argumentos arbitrariamente. En resumen, no se ha propuesto aún una metodología que aumente la capacidad de poda de un sistema argumentativo. Esta información debería ser provista por un mecanismo externo; en nuestro caso, el valor de fuerza de argumentos.

**Ejemplo 5.2** Consideremos el árbol de dialéctica  $\mathcal{T}_F(\Sigma)$  ilustrado a continuación, y cuatro posibles podas  $\mathcal{P}_F^i(\Sigma)$ ,  $1 \leq i \leq 4$ :



De acuerdo a cómo se vaya construyendo el árbol, surgen distintas oportunidades de poda  $1U$ , según el orden en el que se expandan los derrotadores.

- $\mathcal{P}_F^1(\Sigma)$ : se selecciona  $\Sigma_1$ , luego  $\Sigma_2$  (no derrotada) y  $\Sigma_3$  se corta;



- $\mathcal{P}_F^2(\Sigma)$ :  $\Sigma_2$  se selecciona primero, sus hermanos son podados;
- $\mathcal{P}_F^3(\Sigma)$ :  $\Sigma_2$  se selecciona al final, no hay poda.

La poda  $\mathcal{P}_F^4(\Sigma)$  es el caso de una poda que no verifica el criterio que asumimos. Puede verse que  $\Sigma_2$  se poda sin razón aparente, y se llega a un árbol cuya marca en la raíz difiere de la del original. Este tipo de podas no calificaría como un bonsai.

### 5.2.1. Cómo Calcular la Fuerza de una Estructura Argumental

La noción de fuerza explorada en esta sección es similar a la propuesta en [MT08], y está basada en la siguiente afirmación: “una estructura argumental es tan fuerte como débiles son sus derrotadores”. De esta manera, la cantidad de derrotadores no es el único parámetro que afecta la fuerza de una estructura, sino que el enfoque también tiene una componente cualitativa, ya que es susceptible a la topología del árbol. Por ejemplo, dada una estructura *pro*  $P$ , un subárbol enraizado en  $P$  conteniendo muchas estructuras *pro* debería darle a  $P$  un valor alto de fortaleza. En cierta manera, ese valor de fuerza codifica la fuerza de todo el subárbol, *i.e.*, el valor de fuerza de cada estructura en el árbol actúa como un representante del subárbol debajo de ella. En este sentido, una estructura argumental con un alto valor de fuerza podría ser considerado directamente una hoja, lo cual mejoraría el resultado de la poda aceptando el riesgo de perder sensatez.

Como se verá más adelante, los valores de fuerza de las estructuras serán usados como un valor heurístico para guiar la construcción de los árboles de dialéctica, buscando maximizar las posibilidades de poda. La idea básica detrás de estas medidas de fuerza es codificar cuán proclive es una estructura a terminar derrotada en el árbol.

A continuación proponemos una fórmula para calcular la fuerza de una dada estructura en un árbol. La misma responde a las intuiciones dadas en los párrafos anteriores. Dado que la fuerza se calcula en base a los derrotadores, se utilizará el árbol de dialéctica potencial para propagar los valores de fuerza desde las hojas hasta la raíz.

**Definición 5.3 (Fuerza de una Estructura Argumental)** *Dado un DAF  $F$ , una estructura argumental  $\Sigma_i$  en una línea de argumentación  $\lambda_i$ , un árbol potencial  $\mathcal{T} \in \mathbf{trees}_F$ , y una estructura argumental  $\Sigma_j$  en una línea de argumentación  $\lambda_j$  tal que  $\Sigma_j$  es un hijo de  $\Sigma_i$  en  $\mathcal{T}$ , la **fuerza** de  $\Sigma_i$  es:*

$$\mu(\Sigma_i, \lambda_i, \mathcal{T}) = \frac{1}{1 + \sum_j (\mu(\Sigma_j, \lambda_j, \mathcal{T}))}$$

Luego, la fuerza de una estructura argumental  $\Sigma$  en el contexto de un DAF  $F$  se calcula como  $\mu(\Sigma, \cdot, \mathcal{T}_F(\Sigma))$ . Cuando sea conveniente y claro, se utilizará la notación abreviada  $\mu(\Sigma)$ .

El valor de fuerza de una estructura es calculado mediante la construcción del árbol potencial enraizado a ella. Aunque cada estructura distinta de la raíz tiene una fuerza asociada, ésta es *local*; este valor no es la fuerza *real* que cada estructura tendría en el árbol potencial enraizado en ella, sino una medida parcial que permite el cálculo de la fuerza de la estructura raíz. Incluso una estructura ubicada en dos lugares distintos en el mismo árbol exhibiría dos valores diferentes de fuerza local. De esta manera, los valores locales codifican con precisión la “derrotabilidad” del nodo con respecto al subárbol enraizado en él. En este enfoque utilizaremos la fuerza real para la construcción de los bonsai, almacenando un mapeo que asocia a una estructura su valor de fuerza.

**Ejemplo 5.3** *Consideremos el DAF  $F$ , a partir del cual pueden cons-*

truirse las estructuras activas  $\Sigma$ ,  $\Sigma_1$ ,  $\Sigma_2$  y  $\Sigma_3$ , donde los conflictos y la preferencia se ilustran en la Figura 5.3. Las estructuras  $\Sigma_1$  y  $\Sigma_3$  no reciben ataques, así que la suma de la fuerza de sus derrotadores es 0 y  $\mu(\Sigma_1, \lambda_1, \mathcal{T}_F(\Sigma)) = 1/(1+0) = 1$ , al igual que la fuerza de  $\Sigma_3$ . La estructura  $\Sigma_2$  está derrotada sólo por  $\Sigma_3$ , por lo cual  $\mu(\Sigma_2, \lambda_2, \mathcal{T}_F(\Sigma)) = 1/(1+1) = 0,5$ . Por último,  $\Sigma$  está derrotada por  $\Sigma_1$  y  $\Sigma_2$  y su valor de fuerza es  $\mu(\Sigma) = 1/(1+1+0,5) = 0,4$ , como se muestra en la Figura 5.3.

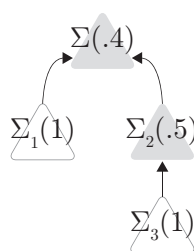


Fig. 5.3: Árbol con valores de fuerza del Ejemplo 5.3

### 5.2.2. Construcción de un Bonsai de Dialéctica

Una vez que cada estructura argumental tiene un valor de fuerza asociado, éste puede utilizarse como un valor heurístico para la construcción de los *bonsai de dialéctica*. Como se mencionó antes, la fuerza de una estructura se calcula de acuerdo al árbol potencial asociado a ella. Esto significa que, en un momento en particular, la fuerza *real actual* de la estructura posiblemente difiera con la fuerza *real potencial*, *i.e.*, la calculada en base al árbol potencial. De hecho, el valor de fuerza que correspondería debería ser calculado sobre el árbol activo enraizado en la estructura. Esto significa que los valores deberían ser recalculados cada vez que se produce un cambio en la base de conocimiento, lo cual no es deseable.

Nuestro enfoque es pragmático: los valores de fuerza potencial se cal-

culan una única vez, y se utilizan para cualquier situación emergente. Salvo en aquellos escenarios en los que el árbol activo coincida con el potencial, los valores serán aproximados, como se muestra en la Figura 5.4; de todas maneras, como se verá en la Sección 5.2.4, los resultados experimentales muestran un gran ahorro en términos de poda. La Figura 5.4(a) muestra un árbol potencial para una estructura, asignándole el valor de fuerza 0,33. La Figura 5.4(b) muestra un posible árbol activo del árbol potencial en (a), donde los valores de fuerza no fueron recalculados. La Figura 5.4(c), en cambio, muestra el valor de fuerza actual de la estructura raíz, es decir, el calculado sobre el árbol activo. Nótese que, a pesar de que la morfología del árbol cambió considerablemente, el valor de fuerza de la raíz no presenta un cambio proporcional.

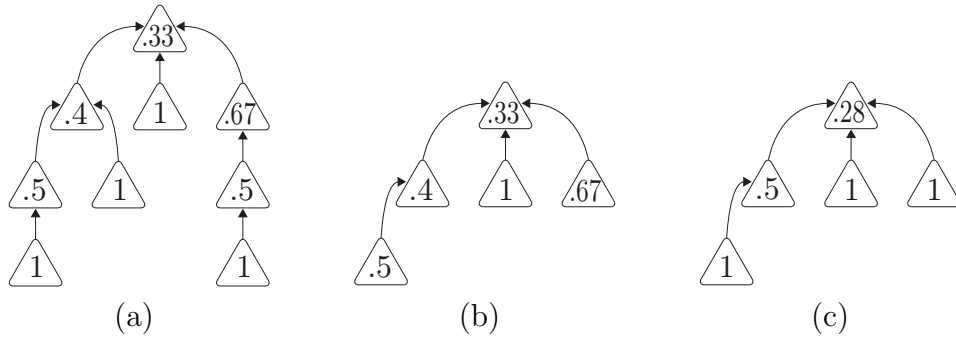


Fig. 5.4: Comparación entre fuerza potencial y actual

**Definición 5.4 (Poda)** Dado un árbol de dialéctica  $\mathcal{T}_F(\Sigma)$  con un conjunto de arcos  $A_{\mathcal{T}}$ , una **poda**  $\mathcal{P}_F(\Sigma)$  para  $\mathcal{T}_F(\Sigma)$  es un árbol enraizado en  $\Sigma$  con un conjunto de arcos  $A_{\mathcal{P}}$  verificando:

- $A_{\mathcal{P}} \subseteq A_{\mathcal{T}}$
- $\text{inTree}(\mathcal{P}_F(\Sigma)) \subseteq \text{inTree}(\mathcal{T}_F(\Sigma))$

Una **poda activa** es una poda para un árbol de dialéctica activo.

Debe notarse que un árbol activo es una poda del árbol potencial asociado, donde toda aparición de una estructura argumental inactiva es eliminada. Sin embargo, la Definición 5.4 valida todo tipo de podas, incluyendo una en la cual una estructura se elimina de una rama mientras sigue formando parte de otra, lo cual sería imposible si la poda fuese un árbol activo. Más adelante se explicará por qué no cualquier poda califica como un bonsai de dialéctica. A continuación formalizamos la noción de poda 1U, cuya intuición fue introducida previamente.

**Definición 5.5 (Poda 1U)** *Dada una poda activa  $\mathcal{P}_F(\Sigma)$  para un árbol de dialéctica  $\mathcal{T}_F(\Sigma)$  en el contexto de un DAF  $F$ , sea  $\Sigma_i$  una estructura argumental interna en  $\mathcal{P}_F(\Sigma)$  cuyo conjunto de derrotadores en  $\mathcal{P}_F(\Sigma)$  es  $\{\Sigma_{i1}, \dots, \Sigma_{in}\}$ . La poda  $\mathcal{P}_F(\Sigma)$  es una **poda 1U** sssi la existencia de una estructura  $\Sigma_{ix}$  ( $1 \leq x \leq n$ ) en una línea de argumentación  $\lambda_x$  tal que  $\mathbf{m}(\Sigma_{ix}, \lambda_x, \mathcal{T}_F(\Sigma)) = \mathbf{U}$  implica que hay a lo sumo una estructura  $\Sigma_{ik}$  ( $1 \leq k \leq n$ ) en una línea de argumentación  $\lambda_i$  tal que  $\mathbf{m}(\Sigma_{ik}, \lambda_i, \mathcal{P}_F(\Sigma)) = \mathbf{U}$ .*

Potencialmente habrá múltiples podas 1U para cada árbol de dialéctica, pero sólo un subconjunto de ellas conducirá al árbol más pequeño posible, en términos de cardinalidad del conjunto de estructuras en el árbol. Las *podas 1U optimales* son aquellas que expanden nodos no derrotados en primera instancia. Si en un conjunto de derrotadores hay más de un nodo no derrotado, es posible que haya múltiples podas optimales. Intuitivamente, una poda 1U optimal será una poda en la cual cada conjunto de derrotadores de una estructura contiene o bien una estructura no derrotada, o  $n$  estructuras derrotadas.

**Definición 5.6 (Poda 1U Optimal)** *Una poda 1U  $\mathcal{P}_F(\Sigma)$  es una **poda 1U optimal** sssi para cada estructura argumental interna  $\Sigma_i$  en una línea de argumentación  $\lambda_i$  en  $\mathcal{P}_F(\Sigma)$  cuyo conjunto de derrotadores es  $\Gamma$*

se verifica una de las siguientes condiciones:

1.  $\mathbf{m}(\Sigma_k, \lambda_k, \mathcal{P}_F(\Sigma)) = \mathbf{D}, \forall \Sigma_k \in \Gamma, \lambda_k^\uparrow(\Sigma_k) = \lambda_i^\uparrow[\Sigma_i];$
2.  $\mathbf{m}(\Sigma_j, \lambda_j, \mathcal{P}_F(\Sigma)) = \mathbf{U}, \Gamma = \{\Sigma_j\}, \lambda_j^\uparrow(\Sigma_j) = \lambda_i^\uparrow[\Sigma_i].$

El objetivo del cálculo de los bonsai de dialéctica es obtener podas lo más similares a la optimal. Es decir, si los valores heurísticos indican con un buen grado de certeza qué estructuras del árbol terminarán no derrotadas, el bonsai tenderá a encontrar las estructuras no derrotadas primero, cortando el resto de los derrotadores pendientes de análisis. A continuación se da la definición de *bonsai de dialéctica* como un caso particular de poda en el cual el único requerimiento es que se mantenga la información concerniente al estado de garantía de la raíz con respecto al árbol completo. Como las podas se realizan al momento de calcular garantía, el bonsai se define sobre árboles activos.

**Definición 5.7 (Bonsai de Dialéctica)** Sea  $\mathbb{T}_F(\Sigma)$  un árbol de dialéctica activo en el contexto de un DAF  $F$ , un **bonsai de dialéctica**  $\mathcal{B}_F(\Sigma)$  es una poda de  $\mathcal{T}_F(\Sigma)$  verificando:

$$\begin{aligned} \mathbf{m}(\Sigma_i, \lambda_k, \mathcal{T}_F(\Sigma)) &= \mathbf{m}(\Sigma_i, \lambda_j, \mathcal{B}_F(\Sigma)), \\ \forall \Sigma_i \in \text{inTree}(\mathcal{B}_F(\Sigma)), \lambda_k^\uparrow(\Sigma_i) &= \lambda_j^\uparrow(\Sigma_i) \end{aligned}$$

**Proposición 5.2** Una poda  $1U$  para un árbol de dialéctica  $\mathcal{T}$  es un bonsai de dialéctica para  $\mathcal{T}$ .

DEMOSTRACIÓN: Sea  $\mathcal{B}$  una poda  $1U$  de  $\mathcal{T}$ . Supongamos por el absurdo que  $\mathcal{B}$  no es un bonsai. Es decir, o bien una estructura derrotada en  $\mathcal{T}$  pasa a estar no derrotada en  $\mathcal{B}$ , o vice versa:

- $\mathbf{m}(\Sigma_i, \lambda_k, \mathcal{T}) = \mathbf{U}$  y  $\mathbf{m}(\Sigma_i, \lambda_j, \mathcal{B}) = \mathbf{D}$ . Sea  $\Gamma$  el conjunto de derrotadores de  $\Sigma_i$  en  $\mathcal{T}$ , por definición de poda el conjunto de derrotadores

de  $\Sigma_i$  en  $\mathcal{B}$  es  $\Gamma' \subseteq \Gamma$ . Por definición de marcado escéptico, como  $\Sigma_i$  está no derrotada, todas las estructuras en  $\Gamma$  están derrotadas. Luego, todas las estructuras en  $\Gamma'$  están derrotadas, y  $\Sigma_i$  en  $\mathcal{B}$  no puede estar derrotada.

- $\mathbf{m}(\Sigma_i, \lambda_k, \mathcal{T}) = \mathbf{D}$  y  $\mathbf{m}(\Sigma_i, \lambda_j, \mathcal{B}) = \mathbf{U}$ . Sea  $\Gamma$  el conjunto de derrotadores de  $\Sigma_i$  en  $\mathcal{T}$ , por definición de poda el conjunto de derrotadores de  $\Sigma_i$  en  $\mathcal{B}$  es  $\Gamma' \subseteq \Gamma$ . Por definición de marcado escéptico, como  $\Sigma_i$  está derrotada, existe  $\Sigma_u \in \Gamma$  tal que está no derrotada. Por definición de poda, si hay al menos un derrotador en  $\Gamma$  no derrotado, entonces hay a lo sumo un derrotador en  $\Gamma'$  no derrotado. Luego,  $\Sigma_i$  en  $\mathcal{B}$  no puede estar no derrotada.

Por lo tanto, las marcas de las estructuras en  $\mathcal{B}$  se preservan con respecto a las marcas en  $\mathcal{T}$ , y  $\mathcal{B}$  es un bonsai de  $\mathcal{T}$ .  $\square$

Por definición un bonsai de dialéctica no es una poda arbitraria del árbol activo asociado, sino una que mantiene la misma marca para toda estructura común a ambos; en particular, la marca de la raíz. Una versión más relajada de esta definición podría requerir que se comparta al menos la marca de la raíz. De todas maneras, en este primer enfoque resulta más simple no dar tal libertad, dado que introduciría una complejidad innecesaria. La propiedad más importante que un bonsai de dialéctica debería satisfacer es garantizar exactamente las mismas estructuras argumentales que los árboles no podados.

**Lema 5.2 (Sensatez & Completitud)** *Dado un DAF  $F$  y un bonsai de dialéctica  $\mathcal{B}_F(\Sigma)$  para un árbol de dialéctica  $\mathcal{T}_F(\Sigma)$ ,  $\Sigma$  está garantizada a partir de  $\mathcal{T}_F(\Sigma)$  sssi  $\Sigma$  está garantizada a partir de  $\mathcal{B}_F(\Sigma)$ .*

DEMOSTRACIÓN: *Trivialmente, por Definición 5.7, la marca de toda estructura raíz en  $\mathcal{B}_F(\Sigma)$  es igual a la marca de la raíz en  $\mathcal{T}_F(\Sigma)$ .  $\square$*

Una versión diferente de la definición de bonsai de dialéctica podría implicar un procedimiento bastante más complicado para determinar el estado de la estructura raíz. Esto, sin embargo, no debería interferir con la satisfacción de las metapropiedades de sensatez y completitud, aunque ciertamente su demostración se complicaría proporcionalmente.

### 5.2.3. Bonsai Fast-prune

Cuando se construye un árbol de dialéctica, el orden en el cual se visitan los derrotadores de un nodo interno es muy relevante para la poda. En este enfoque, cada estructura argumental tiene un valor de fuerza asociado; luego, una vez que todos los vecinos de un nodo interno fueron expandidos, se los visita en profundidad del más fuerte al más débil. La medida de fuerza codifica cuán proclive es una estructura a ser derrotada: que una estructura sea “fuerte” (*i.e.*, con un valor de fuerza cercano a 1) significa que, en general, estará marcada como no derrotada. Se comienza expandiendo los derrotadores del más fuerte, intentando encontrar una estructura no derrotada lo más rápidamente posible, de forma tal que se poden sus hermanos. Esta estrategia se denomina *poda rápida* (en inglés, *fast-prune*).

**Definición 5.8 (Bonsai *Fast-prune*)** Dado un DAF  $F$  y un bonsai de dialéctica  $\mathcal{B}_F(\Sigma)$  para un árbol de dialéctica activo  $\mathbb{T}_F(\Sigma)$ , sea  $\Sigma_i \in \text{inTree}(\mathcal{B}_F(\Sigma))$  con un conjunto de derrotadores  $\Gamma$  en  $\mathbb{T}_F(\Sigma)$  tal que el conjunto de derrotadores en  $\mathcal{B}_F(\Sigma)$  es  $\Gamma' \subseteq \Gamma$ . Dadas una función de fuerza  $\mu(\cdot)$  y una función de marcado  $\mathbf{m}(\cdot, \cdot, \cdot)$ :

$\mathcal{B}_F(\Sigma)$  es un **bonsai fast-prune** sssi

$\Sigma_k \in \Gamma', \mathbf{m}(\Sigma_k, \lambda_k, \mathcal{B}_F(\Sigma)) = \mathbf{U}$  implica que

$\forall \Sigma_{j \neq k} \in \Gamma', \mu(\Sigma_k) \leq \mu(\Sigma_j), \mathbf{m}(\Sigma_j, \lambda_j, \mathcal{B}_F(\Sigma)) = \mathbf{D}$  y

$\mu(\Sigma_x) \leq \mu(\Sigma_k), \forall \Sigma_x \in \Gamma \setminus \Gamma'$ .



**Proposición 5.3** *Un bonsai fast-prune para un árbol de dialéctica  $\mathcal{T}$  es una poda 1U para  $\mathcal{T}$ .*

DEMOSTRACIÓN: *Trivialmente, por Definición 5.8, el bonsai fast-prune permite a lo sumo un derrotador no derrotado por nodo interno del árbol, verificando la Definición 5.5.  $\square$*

**Ejemplo 5.4** *Consideremos el árbol de dialéctica activo ilustrado en la Figura 5.5, en el cual cada estructura argumental tiene anotado su correspondiente valor de fuerza. A la derecha del árbol puede verse el respectivo bonsai fast-prune, el cual surge al elegir los derrotadores más fuertes primero:  $\Sigma_2$  antes que  $\Sigma_1$ , luego  $\Sigma_6$  antes que  $\Sigma_5$ , y finalmente  $\Sigma_9$ , que es el único derrotador para  $\Sigma_6$ , y similarmente  $\Sigma_8$  para  $\Sigma_5$ .*

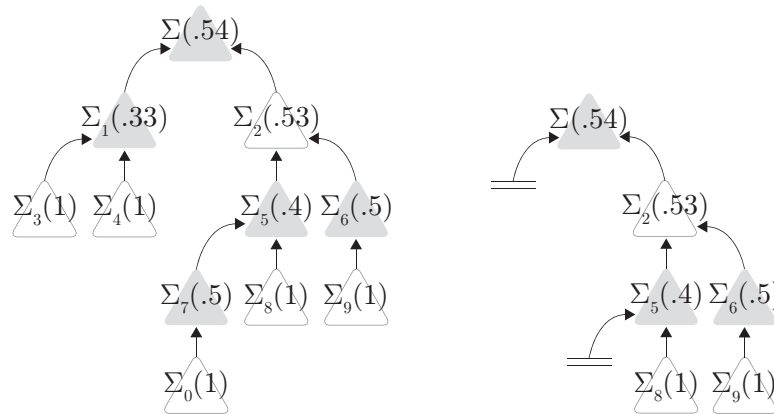


Fig. 5.5: Árbol de dialéctica activo y poda *fast-prune* del Ejemplo 5.4

En el escenario descrito en el Ejemplo 5.4, todos los valores de fuerza fueron calculados a partir de ese árbol activo. Como fue explicado antes, mantener estos valores actualizados recalculándolos cada vez que la situación cambia puede ser computacionalmente prohibitivo; nuestro enfoque consiste en calcularlos una única vez, sobre el árbol potencial, para luego utilizarlos en cualquier instancia activa. Por esto, cuanto más

similar sean el árbol potencial y el activo, más precisos serán los valores de fuerza. Sin embargo, los resultados experimentales han demostrado que estos valores precalculados funcionan bastante bien, aún en casos donde los árboles potencial y activo son muy diferentes.

**Ejemplo 5.5** *Consideremos el árbol potencial de la Figura 5.6(a). Asumiendo que  $\Sigma_{11}$ ,  $\Sigma_7$ ,  $\Sigma_8$  y  $\Sigma_6$  no están activos, tenemos el árbol activo de la Figura 5.6(b). Aplicando la técnica de poda 1U guiada por los valores de fuerza de cada estructura en el árbol potencial, obtenemos el bonsai fast-prune de la Figura 5.6(c). Debe notarse que  $\Sigma$  figura como derrotado en el árbol potencial, a causa de que  $\Sigma_1$  está no derrotado. En el árbol activo, en cambio, si bien  $\Sigma$  también está marcado como derrotado, lo está a raíz de  $\Sigma_2$  y  $\Sigma_3$ . Por último, es importante resaltar que, si los árboles activo y potencial hubiesen más similares, la heurística habría sido más acertada, explorando  $\Sigma_3$  primero, hallando así la poda optimal.*

#### 5.2.4. Resultados Experimentales

En esta sección medimos empíricamente la *performance* del método de bonsai *fast-prune* para construir el árbol activo. Este análisis estadístico utiliza como punto de comparación un método que construye el árbol activo a ciegas, imitando la estrategia típica, capaz de realizar podas, pero sin contemplar un orden específico al momento de explorar los derrotados de una estructura interna. Ésta es la forma en que, por ejemplo, se comporta DELP.

El análisis empírico consiste de 6 pruebas, cada una de las cuales incluye una serie de 1000 corridas. Cada corrida contiene 3 etapas:

1. generación al azar de un árbol potencial;
2. desactivación al azar de un cierto número de estructuras en el árbol

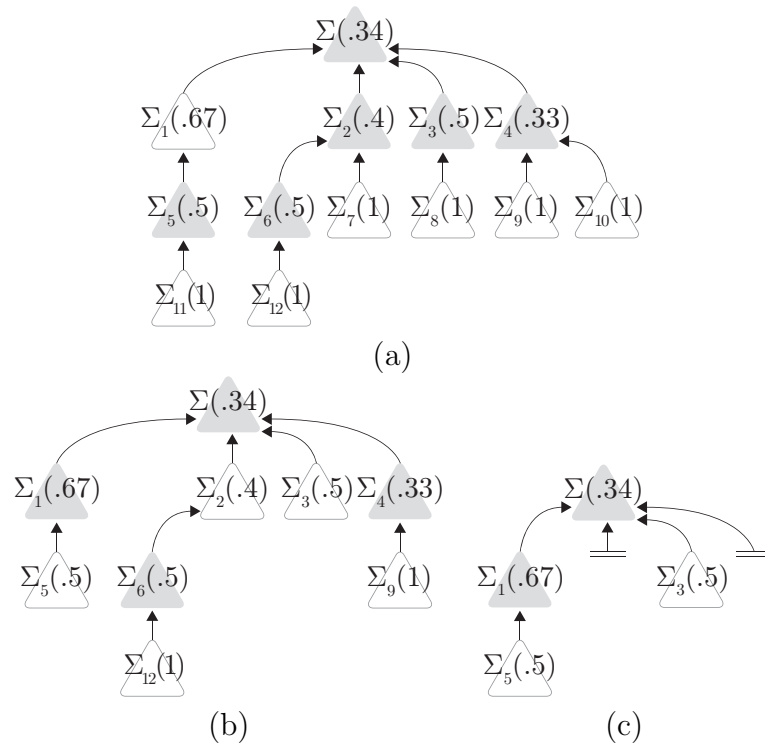


Fig. 5.6: Árboles (a) potencial, (b) activo y (c) bonsai del Ejemplo 5.5

potencial para obtener el árbol activo sin podas;

3. construcción del bonsai:

- bonsai *fast-prune* para obtener el árbol activo podado;
- bonsai *blind* (a ciegas) para obtener el árbol activo podado.

A grandes rasgos, la diferencia entre ambos tipos de bonsai dejará en evidencia el ahorro provocado por la utilización de la técnica *fast-prune*. En la Tabla 5.1 se muestran los resultados finales de dicha experimentación. La primera columna (“Potencial”) muestra la cantidad de nodos en el árbol potencial generado, junto al porcentaje que fue desactivado. Dado que cada estructura desactivada provoca que un subárbol completo desaparezca del árbol potencial, en la segunda columna (“Activo”) se da

la cantidad promedio de nodos en los árboles activos completos (*i.e.*, sin podas) construidos en cada corrida. Las columnas tercera y cuarta (“FP” y “C”) contienen el número promedio de nodos en los bonsai *fast-prune* y *blind*. La quinta columna (“Mejor”) brinda el porcentaje de corridas en las cuales el bonsai *fast-prune* resultó mejor que el *blind*. Por último, la sexta columna (“Prop. Mejor”) muestra el tamaño del bonsai *fast-prune* en proporción al bonsai no guiado. Por ejemplo, la anteúltima fila establece que, en cada corrida, se generó un árbol potencial de 10000 nodos, desactivando 10 % de sus nodos al azar, conduciendo a un árbol activo completo de un promedio de 4872 nodos, y luego de 1000 corridas el bonsai *fast-prune* promedió un tamaño de 58 nodos mientras que el bonsai *blind* logró 188. A partir de un análisis estadístico de confianza se obtuvo que el *fast-prune* fue más pequeño que el ciego en el 87,5 % de los casos, y en esos casos el tamaño del primero fue entre un 5,8 % y un 7,4 % del tamaño del último.

Potencial	Activo	FP	C	Mejor	Prop. Mejor
2k-10 %	1130	28	73	87 %	[14,2 %, 16,9 %]
2k-20 %	599	22	59	84,5 %	[12,6 %, 15,1 %]
5k-10 %	2568	38	119	88,4 %	[9,3 %, 11,4 %]
5k-20 %	1259	28	76	85,1 %	[10,1 %, 12,4 %]
10k-10 %	4872	58	188	87,5 %	[5,8 %, 7,4 %]
10k-20 %	2157	37	110	86,6 %	[6,7 %, 8,3 %]

Tab. 5.1: Tabla de datos experimentales

Resulta evidente que la poda *fast-prune* se comporta mejor cuanto más grande es el árbol potencial asociado. En cuanto a los resultados arrojados por la Tabla 5.1, hay dos aspectos que deben resaltarse:

1. Incluso cuando el tamaño promedio de los árboles activos completos indican una diferencia significativa con sus respectivos árboles

potenciales en relación a la cantidad de nodos, la heurística provista por los valores de fuerza precalculados funcionó bien, posibilitando la obtención de bonsai muy pequeños.

2. El tamaño promedio de los bonsai *fast-prune* está entre el 1,2 % y el 3,7 % del tamaño de los árboles activos completos, y en casi el 90 % de los casos representa cerca del 10 % del tamaño del bonsai a ciegas.

### 5.2.5. *Discusión Final*

En esta sección propusimos un método para acelerar el cómputo de garantía utilizando una técnica de poda sobre árboles de dialéctica en el contexto de un DAF. Los valores heurísticos que posibilitan tales podas son precalculados en un estado inicial, representado por árboles potenciales, y luego utilizados en cualquier instancia activa del DAF. De esto se desprende que el enfoque aquí presentado sólo tiene sentido en un contexto donde la dinámica es tomada en cuenta, ya que el objetivo es amortiguar el impacto de los cambios producidos en el marco argumentativo con respecto a su repercusión en el estado de garantía de las estructuras argumentales en él. En resumen, el método toma en cuenta el conocimiento previo para acelerar el proceso de cómputo de garantías.

Existe un alto grado de incertidumbre acerca de cómo la fluctuación de estructuras activas afecta al estado de garantía de otras estructuras. Sin embargo, los resultados empíricos indican que la información codificada a partir de los árboles potenciales puede utilizarse exitosamente para aproximar cuán proclive es un argumento a estar derrotado en cualquier árbol de dialéctica. Incluso esos valores podrían utilizarse para aproximar garantía, lo cual no ha sido estudiado aquí. La reducción en el espacio de búsqueda es especialmente útil en escenarios donde la argumentación

se lleva a cabo sobre dominios grandes, como la *World Wide Web*. En estos casos, adicionalmente, la búsqueda de contra-argumentos suele ser una tarea costosa. En [Hun04] la problemática del tamaño de los árboles de dialéctica se encara desde un punto de vista y una intención muy diferentes. El autor concluye que algunos árboles pueden ser simplemente “muy grandes”, influyendo negativamente en el impacto de la argumentación en relación con la “audiencia” a la que se dirige. En función de este impacto se decide no presentar ciertos argumentos, para maximizar el efecto que la argumentación quiere provocar en la audiencia. El resultado es un árbol que presenta ciertas podas, cuya naturaleza es diferente a las presentadas aquí: los bonsai de dialéctica que hemos definido exhiben exactamente el mismo comportamiento que los árboles completos.

Si bien la construcción de los árboles potenciales para cada estructura se lleva a cabo una única vez (en el estado inicial) la complejidad computacional podría dificultar la implementación del método de poda. Sin embargo, esta complejidad puede ser disminuida mediante la utilización de técnicas de programación dinámica, reutilizando subárboles de árboles ya calculados. Notar que la aplicación de restricciones dialécticas (*e.g.*, no circularidad) logra que, al considerar un subárbol  $S$  de  $T$  para construir un nuevo árbol  $T'$ , algunas estructuras argumentales en el camino entre la raíz de  $S$  y la raíz de  $T$  pueden formar parte de  $T'$ . Por esto no puede tomarse a  $S$  directamente como un árbol de dialéctica. Sin embargo, los nodos de  $S$  estarán pre-calculados en  $T'$ , y eso puede significar un gran ahorro.

Adicionalmente, puede considerarse el almacenamiento de los árboles potenciales, para evitar su reconstrucción parcial, tal como se propone en [CCS05]. Este enfoque puede ser combinado con la técnica de poda presentada en esta sección, logrando un método de poda sobre los árbo-

les previamente almacenados. Cabe aclarar que ya no utilizaríamos los valores de fuerza reales de los argumentos, sino los locales. Es decir, los árboles potenciales almacenados tendrán cada nodo etiquetado con su correspondiente valor de fuerza local. La desventaja de este método es el gran espacio de almacenamiento necesario para llevarlo a cabo: para cada estructura debe almacenarse su árbol potencial etiquetado.

En cuanto a trabajo relacionado, en [BH06] se presentó un enfoque que estudia cómo acelerar la construcción de árboles de dialéctica. El objetivo de este trabajo y el nuestro están relacionados: ambos desarrollan técnicas para superar el desafío computacional que constituye la construcción de árboles de dialéctica. Similarmente a nuestra utilización de un valor de fortaleza de argumentos, ellos usan información de la base de conocimiento para acelerar el proceso argumentativo. En su sistema, el conocimiento está basado en un lenguaje proposicional, a partir del cual se pueden distinguir subconjuntos inconsistentes. En concreto, la propuesta consiste en una compilación de la base de conocimiento, produciendo un hipergrafo de subconjuntos inconsistentes. Esta estructura de datos, mediante un algoritmo especializado, se utiliza para acelerar la construcción de los árboles de dialéctica. La diferencia principal con nuestro enfoque reside en la etapa de la construcción de los árboles en la cual se centra el esfuerzo: la compilación hecha en [BH06] busca ahorrar tiempo al detectar ataques, sin hacer referencia al tamaño del espacio de búsqueda, mientras que en nuestro trabajo procuramos obtener los árboles más pequeños que arrojen un resultado conclusivo sobre el estado de garantía de la raíz. Los resultados de ambos trabajos parecen seguir caminos paralelos, lo cual sugiere que podrían ser combinados.





## 6. APLICACIONES DEL DAF

En este capítulo presentaremos dos aplicaciones del DAF: primeramente, se definirá una arquitectura BDI de agentes basada en argumentación, y luego se describirá la utilización de dos de las operaciones de revisión vistas en el Capítulo 4 para efectuar razonamiento hipotético sobre litigios.

### 6.1. Una Arquitectura de Agentes

La arquitectura presentada en esta sección está basada en la definida en el artículo [RGS07], el cual fue extendido en [RGS08]. En dichos artículos se implementa una arquitectura BDI utilizando DELP para representar conocimiento y razonamiento. En este caso, traduciremos las componentes de la arquitectura para ser representadas mediante el DAF. Además de presentar la arquitectura, nos enfocaremos en dos aplicaciones: un sistema de seguridad y fútbol de robots. Éstas fueron elegidas porque representan dos tipos de agente significativamente diferentes.

En la arquitectura se provee un mecanismo de razonamiento rebatible para filtrar los *deseos* de un agente para luego obtener un conjunto que denominaremos de *deseos actuales*, *i.e.*, aquellos alcanzables en la situación actual. Un conjunto de *reglas de filtrado*, representadas mediante un DAF, serán utilizadas para representar razones a favor y en contra de la adopción de deseos. Por ejemplo, el argumento con interfaz  $\langle \{humo(R)\}, llamar(bomberos) \rangle$  significa “si hay humo en la habi-

tación  $R$  entonces hay una razón para llamar a los bomberos”, donde  $llamar(bomberos)$  está dentro de los deseos disponibles. El agente controlando el sistema de seguridad será provisto de un conjunto de reglas de filtrado que, analizadas en conjunto, indicarán qué deseos son plausibles para ser perseguidos.

Dado que este enfoque permite la definición de diferentes tipos de agente, en el caso del sistema de seguridad implementaremos un agente *cauto*, lo cual se traduce en un agente que sólo considera como actuales aquellos deseos que están garantizados por una semántica argumentativa escéptica. Luego de calculado el conjunto de deseos actuales, el agente estará en condiciones de elegir su próxima intención. Sobre el final de esta sección veremos que una de las diferencias principales entre los dominios de aplicación elegidos es la cardinalidad del conjunto de intenciones: mientras que el agente de seguridad puede tener varias intenciones para satisfacer (una por cada situación de peligro en las habitaciones que él controla), el agente controlando un robot futbolista puede perseguir sólo una intención a la vez, ya que le es imposible, por ejemplo, patear al arco y hacer un pase simultáneamente.

### 6.1.1. La Arquitectura

El esquema general de la arquitectura BDI basada en el DAF es el ilustrado en la Figura 6.1. En resumen, el ciclo percepción-acción de esta arquitectura comienza con un conjunto de evidencia  $\phi$  representando la *percepción*. Esta información forma parte del *DAF de creencias* que, luego de un proceso argumentativo, conduce al conjunto  $B$  de *creencias*. Por ejemplo, supongamos que un agente percibe que *está marcado* y que *un compañero de equipo tiene la pelota*, entonces puede garantizar la creencia significando que *no es candidato para recibir un pase*.

Como se ve en la figura, el *DAF de filtrado* se mezcla con las *creencias* y, junto al *conjunto de deseos* y la *función de filtrado*, son la entrada de un proceso argumentativo que deriva el *conjunto de deseos actuales*. Continuando con el ejemplo, el agente podría tener una regla de filtrado dictando que *si no puede recibir un pase entonces hay una razón para moverse a un lugar diferente*. La etapa final del ciclo de comportamiento del agente incluye la utilización de un conjunto de *reglas de intención* y una *política de intención* que determinará la regla preferida y por ende la *intención seleccionada*.

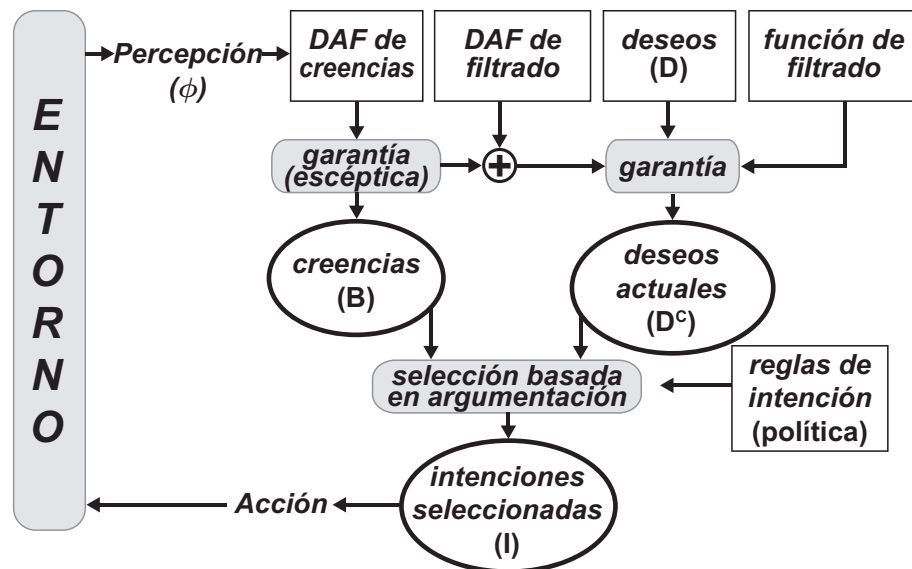


Fig. 6.1: Arquitectura BDI de agentes basada en el DAF

En la Figura 6.1 puede verse que hay tres procesos principales. Cada uno de ellos se vale de la argumentación rebatible vía DAF para realizar inferencias. La dinámica proviene de la percepción del entorno, que se reevalúa en cada ciclo.

### 6.1.2. Creencias

Las creencias de un agente se corresponden con una semántica escéptica (*e.g.*, *grounded*) calculada sobre el DAF de creencias  $F_{\mathbf{B}}$ . A lo largo de esta sección se utilizará la semántica de DELP (ver Definición 2.34), calculando garantía a través de árboles de dialéctica con marcado escéptico.

El lenguaje de sentencias lógicas para premisas y conclusiones utilizado a lo largo de toda la arquitectura se denotará  $\mathfrak{L}_{bdi}$ . Del conjunto de evidencia en  $F_{\mathbf{B}} = \langle \mathbf{E}_{\mathbf{B}}, W_{\mathbf{B}}, \bowtie_{\mathbf{B}}, \mathbf{pref}_{\mathbf{B}} \rangle$  se distinguen dos subconjuntos: el conjunto  $\phi$  de *creencias percibidas* y  $\varphi$ , el de *creencias estáticas*. A partir de la evidencia, el agente puede utilizar los argumentos en el conjunto  $W_{\mathbf{B}}$  de *creencias rebatibles* para obtener garantías para sus *creencias derivadas*. Las creencias percibidas y estáticas, junto con las derivadas componen el conjunto  $\mathbf{B}$  de creencias. Dado que el conjunto de evidencia debe ser consistente, se utiliza la mezcla de DAFs (ver Definición 2.26) para incorporar la percepción a las creencias.

**Definición 6.1 (Creencias)** *Dado un conjunto de creencias percibidas  $\phi$  y un DAF de creencias estáticas  $\langle \varphi, W, \bowtie, \mathbf{pref} \rangle$ , el conjunto de **creencias** es  $\mathbf{B} = \mathit{skep}(F_{\mathbf{B}})$ , donde  $F_{\mathbf{B}}$  es el DAF de creencias:*

$$F_{\mathbf{B}} = \langle \mathbf{E}_{\mathbf{B}}, W_{\mathbf{B}}, \bowtie_{\mathbf{B}}, \mathbf{pref}_{\mathbf{B}} \rangle = \langle \phi, \emptyset, \emptyset, \mathbf{pref}_{nil} \rangle \ominus \langle \varphi, W, \bowtie, \mathbf{pref} \rangle,$$

tal que  $\mathbf{pref}_{nil}(\cdot, \cdot) = \epsilon$ .

El DAF de creencias se actualiza continuamente de acuerdo a la percepción, y todas aquellas creencias estáticas en conflicto con las percibidas serán convertidas en argumentos. Por simplicidad, los ejemplos en esta sección no considerarán conjuntos de creencias percibidas  $\phi$  y estáticas  $\varphi$  en contradicción, y  $\mathbf{E}_{\mathbf{B}} = \phi \cup \varphi$ .

**Ejemplo 6.1** Consideremos un agente de fútbol robótico cuyo razonamiento está regido por el DAF  $\langle \mathbf{E}_B, W_B, \bowtie_B, \mathbf{pref}_B \rangle$ , donde en  $\mathbf{E}_B$  se distinguen los conjuntos  $\phi = \{tienePelota(c1), marcado(c1)\}$  de hechos percibidos representando “el jugador  $c1$  tiene la pelota” y “el compañero  $c1$  está marcado”, y el conjunto  $\varphi = \{companhero(c1), oponente(o1)\}$  de creencias estáticas. El conjunto de creencias rebatibles es  $W_B = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3\}$ , donde:

- $\text{int}(\mathcal{A}_1) = \langle \{companhero(X), tienePelota(X)\}, recibir(yo) \rangle$ ,
- $\text{int}(\mathcal{A}_2) = \langle \{marcado(yo)\}, \neg recibir(yo) \rangle$ ,
- $\text{int}(\mathcal{A}_3) = \langle \{oponente(X), tienePelota(X)\}, \neg recibir(yo) \rangle$

A partir de  $\langle \mathbf{E}_B, W_B, \bowtie_B, \mathbf{pref}_B \rangle$  el agente puede garantizar recibir( $yo$ ), ya que la estructura argumental compuesta por  $\mathcal{A}_1$  no tiene derrotadores. De esta manera, el agente incorpora a su conjunto  $\mathbf{B}$  la creencia dictando que es un candidato para recibir un pase.

El conjunto de evidencia del DAF de creencias está compuesto no sólo por la percepción, sino también por aquella evidencia en forma de hechos representando características del agente, roles, *etc.* Estos hechos, en la medida en que no contradigan a la percepción, son persistentes en el sentido de que no cambian en función de la situación actual, como  $miRol(defensor)$  o  $companhero(c1)$ .

Se asume una función de percepción que provee al agente con información acerca de su entorno. Esta función será invocada por el agente para actualizar sus creencias percibidas. Cuando esto ocurre, la información nueva sobrescribe la vieja mediante un borrado múltiple de evidencia (Definición 2.20) y un posterior agregado múltiple (Definición 2.19). De esta forma, no hay problemas de consistencia en cuanto a la percepción, y

luego la mezcla asegura la unión consistente entre percepción y creencias estáticas.

### 6.1.3. Deseos

Los deseos de los agentes estarán representados a través de un conjunto  $D$  de sentencias lógicas en  $\mathcal{L}_{bdi}$ . Si bien el lenguaje de premisas y conclusiones es el mismo que el utilizado en el DAF de creencias, se impone como restricción que ninguna creencia puede ser un deseo, y *vice versa*:  $B \cap D = \emptyset$ , en todo momento. Esta restricción resulta razonable para un formalismo de representación de conocimiento, ya que una sentencia representando un deseo no será percibida ni derivada como una creencia. En ciertos contextos, los deseos se definen como configuraciones (tal vez parciales) del mundo a alcanzar. Estas metas, una vez alcanzadas, pasan a formar parte de las creencias [vRDM09]. Este no es el caso de la arquitectura definida en esta sección, en la cual los deseos representan una acción (de alto nivel) a ejecutar. El conjunto  $D$  puede ser contradictorio, ya que sólo representa los deseos que el agente podría querer alcanzar, y no las metas que se ha propuesto.

De acuerdo a la situación actual, algunos deseos puede ser imposibles de ser llevados a cabo. Por ejemplo, si el agente futbolista no tiene posesión de la pelota y la misma se encuentra en un lugar  $p$ , entonces el deseo *patearAlArco* sería inviable, mientras que *irA(p)* sería una opción plausible. Los agentes deberían ser capaces de razonar acerca de sus deseos para seleccionar aquellos que pueden ser efectivamente alcanzados. Siguiendo el espíritu del modelo BDI, una vez que los deseos más apropiados son detectados, el agente deberá comprometerse a una intención, y a partir de allí realizar aquellas acciones (de bajo nivel) que la cumplen, o que al menos acercan al agente a su cumplimiento (ver Figura 6.1).

Para realizar la selección de deseos, el agente utiliza sus creencias en combinación con reglas de filtrado dentro de un DAF. Las reglas de filtrado están implícitas dentro de este DAF en forma de argumentos a favor y en contra de la adopción de deseos. Una vez establecido qué deseos son alcanzables, el agente puede adoptar uno o más de ellos como intención.

**Definición 6.2 (Regla de Filtrado)** *Dado un conjunto  $D$  de deseos, una **regla de filtrado** es un argumento  $\langle \{\beta_1, \dots, \beta_n\}, \delta \rangle$ , donde  $\delta \in D$ .*

Debe notarse que una regla de filtrado es un argumento y como tal puede ser derrotado. Las premisas de estos argumentos pueden ser tanto creencias como deseos, es decir, conclusiones de otras reglas de filtrado. Como restricción de diseño, se decidió no representar el filtrado vía evidencia, ya que se busca que la selección de deseos sea el resultado de una deliberación. En todo caso, un argumento lo suficientemente fuerte determinará que un deseo quede frecuentemente elegido como intención. Las reglas de filtrado son potencialmente conflictivas, y como tales, se organizan dentro del *DAF de filtrado*  $\langle \emptyset, W_F, \bowtie_F, \mathbf{pref}_F \rangle$ . Este DAF, en conjunto con las creencias, forman la *base de conocimiento* del agente.

**Definición 6.3 (Base de Conocimiento)** *Dado un conjunto de creencias  $B$  y un DAF de filtrado  $\langle \emptyset, W_F, \bowtie_F, \mathbf{pref}_F \rangle$ , la **base de conocimiento** de un agente es el DAF  $K_{bdi} = \langle B, W_F, \bowtie_F, \mathbf{pref}_F \rangle$ .*

**Ejemplo 6.2** *Supongamos un agente futbolista con deseos  $D = \{patear, llevar, pase, mover\}$  y el DAF de filtrado  $\langle B, W_F, \bowtie_F, \mathbf{pref}_F \rangle$ <sup>1</sup>, donde:*

<sup>1</sup> Los argumentos son dados explícitamente con su interfaz para facilitar la lectura.

$$W_F = \left\{ \begin{array}{l} \text{int}(\mathcal{B}_1) = \langle \{\neg\text{pelota}\}, \neg\text{llevar} \rangle \\ \text{int}(\mathcal{B}_2) = \langle \{\neg\text{pelota}\}, \neg\text{patear} \rangle \\ \text{int}(\mathcal{B}_3) = \langle \{\neg\text{pelota}\}, \neg\text{pase} \rangle \\ \text{int}(\mathcal{B}_4) = \langle \{\text{nadieAdelante}\}, \text{llevar} \rangle \\ \text{int}(\mathcal{B}_5) = \langle \{\text{arqueroMal}\}, \text{patear} \rangle \\ \text{int}(\mathcal{B}_6) = \langle \{\text{compaLibre}\}, \text{pase} \rangle \\ \text{int}(\mathcal{B}_7) = \langle \{\text{patear}\}, \neg\text{llevar} \rangle \\ \text{int}(\mathcal{B}_8) = \langle \{\text{arcoLejos}\}, \neg\text{patear} \rangle \\ \text{int}(\mathcal{B}_9) = \langle \{\neg\text{pelota}\}, \text{mover} \rangle \end{array} \right\}, \quad \bowtie_F = \left\{ \begin{array}{l} (\mathcal{B}_1, \mathcal{B}_4), \\ (\mathcal{B}_2, \mathcal{B}_5), \\ (\mathcal{B}_3, \mathcal{B}_6), \\ (\mathcal{B}_4, \mathcal{B}_7), \\ (\mathcal{B}_5, \mathcal{B}_8), \\ (\mathcal{B}_8, \mathcal{B}_5) \end{array} \right\}$$

El digrafo subyacente a este DAF puede verse en la Figura 6.2, donde la dirección de las flechas indica la preferencia entre estructuras argumentales.

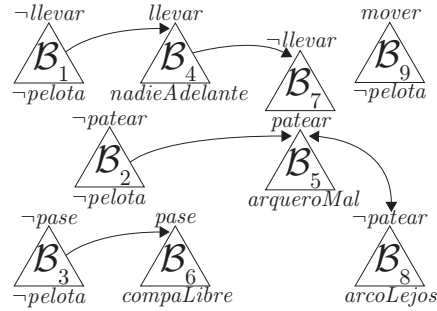


Fig. 6.2: Digrafo de estructuras argumentales para el Ejemplo 6.2

En una situación en particular en la que el agente no tiene la pelota el conjunto de creencias es  $\mathbf{B} = \{\neg\text{pelota}\}$ . A partir de la base de conocimiento  $\langle \mathbf{B}, W_F, \bowtie_F, \text{pref}_F \rangle$  hay garantías para  $\neg\text{llevar}$ ,  $\neg\text{pase}$  y  $\neg\text{patear}$ , ya que sólo  $\mathcal{B}_1$ ,  $\mathcal{B}_2$  y  $\mathcal{B}_3$  están activos.

Considerando otra situación, asumamos que  $\mathbf{B} = \{\text{pelota}, \text{arqueroMal}, \text{arcoLejos}\}$ ; esto es, una situación en la que el agente tiene la pelota, el



arquero contrario está mal ubicado, y el arco contrario está lejos de la posición del agente. En este caso hay sendas estructuras argumentales para *patear* y  $\neg$ *patear*, compuestas por los argumentos  $\mathcal{B}_5$  y  $\mathcal{B}_8$ , respectivamente, no hay preferencia entre ellas, y no hay garantía para ninguno de ellos. También debe notarse que  $\mathcal{B}_5$  y  $\mathcal{B}_7$  componen una estructura argumental no derrotada (i.e., garantizada) para  $\neg$ *llevar*.

La siguiente definición introduce un mecanismo para filtrar el conjunto  $D$  y así obtener sólo aquellos deseos que son alcanzables en la situación actual. Se permite la definición de distintos *tipos de agente*, cada uno de los cuales especificará un proceso de filtrado diferente.

**Definición 6.4 (Deseos Actuales)** *Dados una base de conocimiento  $K_{bdi}$ , un conjunto de deseos  $D$ , y una función booleana  $T$  representando un criterio de selección, el conjunto  $D^c$  de **deseos actuales** es:*

$$D^c = \{\delta \in D \mid T(\delta, K_{bdi}) = true\}$$

Diferentes tipos de agente pueden ser definidos, de acuerdo al dominio de aplicación donde se encuentre el agente. Las siguientes son alternativas interesantes:

- AGENTE CAUTO: la función  $T(\delta, K_{bdi})$  es verdadera cuando hay una garantía para  $\delta$  a partir de  $K_{bdi}$ ;
- AGENTE OSADO: la función  $T(\delta, K_{bdi})$  es verdadera cuando no hay una garantía para  $\bar{\delta}$  a partir de  $K_{bdi}$ .

Cabe remarcar que es posible que un agente osado incluya deseos complementarios en su conjunto de deseos actuales. Esto ocurrirá cuando un deseo y su complemento no puedan ser garantizados a partir de  $K_{bdi}$ .

Otra opción plausible para un agente osado es incluir aquellos deseos que están garantizados crédulamente.

**Ejemplo 6.3** *A partir del DAF de filtrado y el conjunto de deseos del Ejemplo 6.2, consideremos el conjunto de creencias:*

$$\mathcal{B}_1 = \{\text{arcoLejos}, \text{nadieAdelante}, \text{pelota}\},$$

por lo cual el conjunto de argumentos activos es:

$$\mathbb{A} = \{\mathcal{B}_4, \mathcal{B}_8\}$$

Un agente osado generaría el siguiente conjunto de deseos actuales:

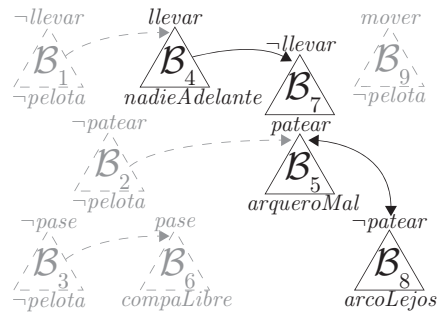
$$\mathcal{D}^c = \{\text{llevar}, \text{pase}, \text{mover}\},$$

ya que la estructura  $\mathcal{B}_4$  para llevar está garantizada y no hay garantía para  $\neg\text{pase}$  ni para  $\neg\text{mover}$ . Nótese que un agente cauto sólo tendría el deseo llevar en  $\mathcal{D}^c$ .

Ahora consideremos el conjunto de creencias:

$$\mathcal{B}_2 = \{\text{arqueroMal}, \text{arcoLejos}, \text{nadieAdelante}, \text{pelota}\},$$

siendo la instancia activa:



El deseo llevar se encuentra garantizado, ya que derrota a la estructura para  $\neg\text{llevar}$ ; el deseo patear está en derrota mutua con  $\neg\text{patear}$  (argumentos  $\mathcal{B}_5$  y  $\mathcal{B}_8$ ); y los deseos pase y mover no tienen razones a favor ni en contra. Un agente osado tendría el conjunto de deseos actuales:

$$D^c = \{\text{llevar}, \text{patear}, \text{pase}, \text{mover}\},$$

mientras que un agente cauto, nuevamente, sólo consideraría llevar como deseo actual.

#### 6.1.4. Intenciones

En este enfoque, una intención es un deseo actual que el agente se comprometerá a realizar. Para especificar bajo qué condiciones la intención puede ser alcanzada, el agente será provisto con un conjunto de *reglas de intención*. A continuación se formalizan estos conceptos, y la noción de *regla de intención aplicable*.

**Definición 6.5 (Regla de Intención)** Una *regla de intención* se denota  $(d \Leftarrow \{p_1, \dots, p_n\}, \{c_1, \dots, c_m\})$ , donde  $d$  es un literal representando un deseo que podría ser elegido como una intención,  $p_1, \dots, p_n$  (resp.,  $c_1, \dots, c_m$ ) son sentencias de  $\mathfrak{L}_{bdi}$  representando precondiciones (resp., restricciones).

**Ejemplo 6.4** El agente futbolista podría tener el siguiente conjunto de reglas de intención:

$$IR_1 : (\text{llevar} \Leftarrow \{\text{pelota}\}, \{\})$$

$$IR_2 : (\text{pase} \Leftarrow \{\text{pelota}\}, \{\text{patear}\})$$

$$IR_3 : (\text{patear} \Leftarrow \{\text{pelota}\}, \{\text{marcado}\})$$

$$IR_4 : (\text{llevar} \Leftarrow \{\text{ganando}\}, \{\})$$

$$IR_5 : (\text{mover} \Leftarrow \{\}, \{\})$$

**Definición 6.6 (Regla de Intención Aplicable)** Dado un conjunto de deseos actuales  $D^c$  y una base de conocimiento  $K_{bdi}$ , una regla de intención  $(d \Leftarrow \{p_1, \dots, p_n\}, \{c_1, \dots, c_m\})$  es **aplicable** sssi

1.  $d \in D^c$ ,

2.  $p_i \in (\mathbf{B} \cup \mathbf{D}^c), (1 \leq i \leq n),$

3.  $c_j \notin (\mathbf{B} \cup \mathbf{D}^c), (1 \leq j \leq m).$

El objetivo de las reglas de intención es seleccionar el conjunto final de intenciones. En general, esta selección entre deseos actuales no puede ser realizada simplemente con las reglas de filtrado. En toda regla de intención aplicable se verifica:

1. la cabeza  $d$  es un deseo actual del agente,
2. toda precondition  $p_i$  que es una creencia está garantizada escépticamente en  $\mathbf{K}_{bdi},$
3. toda precondition  $p_i$  que es un deseo pertenece a  $\mathbf{D}^c,$
4. toda restricción  $c_j$  que es una creencia no está garantizada escépticamente en  $\mathbf{K}_{bdi},$
5. toda restricción  $c_j$  que es un deseo no pertenece a  $\mathbf{D}^c.$

**Ejemplo 6.5** Consideremos un agente osado con la base de conocimiento  $\langle \mathbf{B}_1, W_F, \bowtie_F, \mathbf{pref}_F \rangle$  del Ejemplo 6.3, y las reglas de intención del Ejemplo 6.4. La regla  $IR_1$  es aplicable porque llevar  $\in \mathbf{D}^c$  y pelota  $\in \mathbf{B}_1$ ; la regla  $IR_2$  es aplicable porque pase  $\in \mathbf{D}^c$  y patear  $\notin \mathbf{D}^c$ ; la regla  $IR_3$  no es aplicable porque patear  $\notin \mathbf{D}^c$ ; la regla  $IR_4$  no es aplicable porque la precondition ganando no está en  $\mathbf{B}_1$  ni en  $\mathbf{D}^c$  (incluso podría no pertenecer a  $\mathbf{K}_{bdi}$ ); la regla  $IR_5$  es aplicable porque mover  $\in \mathbf{D}^c$ . El conjunto de reglas de intención aplicables es  $\{IR_1, IR_2, IR_5\}.$

En las reglas de intención aplicables las cabezas representan *intenciones aplicables*, que son alcanzables en la situación actual. Dependiendo del dominio de aplicación se determinará una política de selección de intenciones. Por ejemplo, algunos agentes podrían ser capaces de perseguir

varias intenciones simultáneamente, mientras que otros deberán comprometerse a alcanzar sólo una. Por ejemplo, si el agente debe seleccionar una única intención, y hay dos deseos actuales como cabeza de dos reglas de intención aplicables, ¿cómo podría elegir uno entre ellos? Existe la necesidad de un mecanismo externo para realizar esa decisión. Más aún, dado que la deliberación sobre deseos pudo no haber terminado de individualizar el más apropiado, las creencias podrían jugar un rol en esta selección.

**Definición 6.7 (Conjunto de Intenciones)** *Dado un conjunto  $IR$  de reglas de intención, sea  $App$  el conjunto de todas las reglas aplicables de  $IR$ , y  $p : IR \rightarrow 2^D$ , una política de selección, el **conjunto de intenciones** es  $I = p(App)$ .*

Un ejemplo simple de política podría ser “retornar todas las cabezas de regla en  $App$ ”. Una versión más restrictiva podría adoptarse en el dominio de fútbol robótico, donde los agentes deben seleccionar sólo una intención aplicable a la vez. Una posibilidad es dar un orden de preferencia entre las reglas y retornar la cabeza de la primera regla aplicable según este orden.

**Ejemplo 6.6** *Continuando con el Ejemplo 6.5, el conjunto de reglas de intención aplicables es  $App = \{IR_1, IR_2, IR_5\}$ . Si la política de selección es  $p(App) = \{\delta \mid IR_i : (\delta \Leftarrow \{\cdot\}, \{\cdot\}) \text{ es aplicable y } \nexists IR_{j < i} \text{ aplicable}\}$ , el conjunto de intenciones es la cabeza de  $IR_1$ , i.e.,  $p(App) = \{\text{llevar}\}$ .*

Ahora que se han introducido todos los elementos que conforman a un agente BDI basado en el DAF, se puede definir la estructura formal de un agente *BDI-DAF*.

**Definición 6.8 (Agente BDI-DAF)** *Un agente **BDI-DAF** es una*

tupla  $\langle D, \langle \mathbf{E}_B, W_B, \bowtie_B, \mathbf{pref}_B \rangle, \langle \emptyset, W_F, \bowtie_F, \mathbf{pref}_F \rangle, T, IR, p \rangle$ , con un conjunto  $D$  de deseos, un DAF de creencias  $\langle \mathbf{E}_B, W_B, \bowtie_B, \mathbf{pref}_B \rangle$ , un DAF de filtrado  $\langle \emptyset, W_F, \bowtie_F, \mathbf{pref}_F \rangle$ , un tipo de agente  $T$ , un conjunto de reglas de intención  $IR$ , y una política de selección de intenciones  $p(\cdot)$ .

### 6.1.5. Dominio de Aplicación: Fútbol de Robots

El agente futbolista se define como:

$$A_f = \langle D, \langle \mathbf{E}_B, W_B, \bowtie_B, \mathbf{pref}_B \rangle, \langle \emptyset, W_F, \bowtie_F, \mathbf{pref}_F \rangle, T, IR, p \rangle,$$

donde el conjunto  $D$  y  $\langle \mathbf{E}_B, W_B, \bowtie_B, \mathbf{pref}_B \rangle$  provienen del Ejemplo 6.2, el conjunto  $IR$  es el definido en el Ejemplo 6.4 y la política será la provista en el Ejemplo 6.6. En cuanto al tipo  $T$  de agente, en los ejemplos dados a continuación se evaluarán como alternativas los tipos osado y cauto.

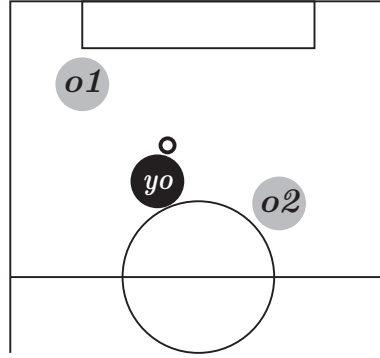
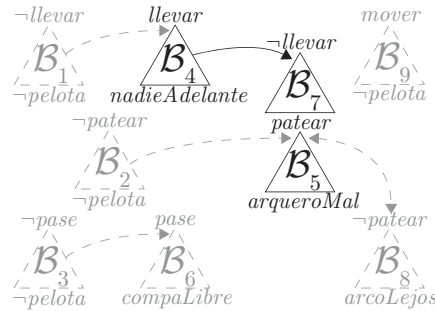


Fig. 6.3: Un escenario para el agente futbolista

Consideremos al agente  $A_f$  en el escenario ilustrado en la Figura 6.3, donde  $o1$  y  $o2$  representan las posiciones de dos oponentes, siendo  $o1$  el arquero, y  $yo$  es la posición de  $A_f$ , quien tiene la pelota. La percepción de  $A_f$  es  $\phi_1 = \{pelota, nadieAdelante, arqueroMal\}$ , y el DAF de creencias es  $\langle \phi_1, \emptyset, \emptyset, \mathbf{pref}_{nil} \rangle$ , por lo tanto,  $\mathbf{B}_1 = \phi_1$ . En esta situación, la instancia

activa es:



Puede verse que *patear* y *llevar* están garantizados, mientras que *pase*,  $\neg$ *pase*, *mover* y  $\neg$ *mover* no lo están. De acuerdo al tipo de agente<sup>2</sup>, el conjunto de deseos actuales será diferente y también lo será la intención seleccionada:

- AGENTE CAUTO:  $D_{C1}^c = \{patear, \neg llevar\}$ , la regla de intención  $IR_3$  es aplicable e  $I_{C1} = \{patear\}$ ;
- AGENTE OSADO:  $D_{O1}^c = \{patear, \neg llevar, pase, mover\}$ , las reglas  $IR_1$  e  $IR_3$  son aplicables e  $I_{O1} = \{llevar\}$ .

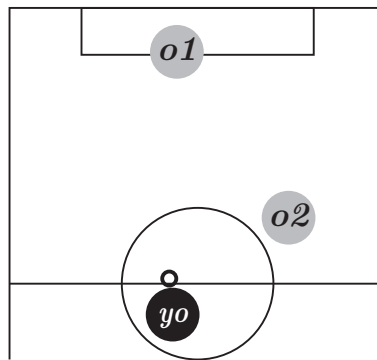
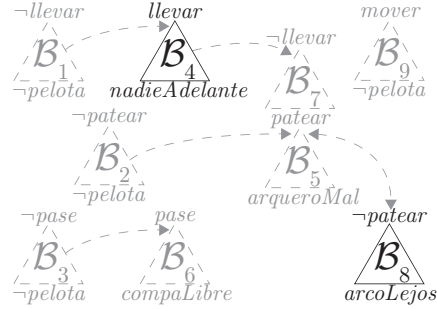


Fig. 6.4: Un escenario para el agente futbolista

Consideremos el mismo agente  $A_f$  pero en el escenario ilustrado en la Figura 6.4. La percepción del agente es  $\phi_2 = \{pelota, nadieAdelante,$

<sup>2</sup> Los conjuntos  $D$  e  $I$  de un agente estarán subindicados denotando el tipo.

$\text{arcoLejos}\} = \mathcal{B}_2$ . En esta situación, la instancia activa del DAF correspondiente a la base de conocimientos del agente es:



- AGENTE CAUTO:  $D_{C_2}^c = \{\text{llevar}\}$ , la regla de intención  $IR_1$  es aplicable e  $I_{C_2} = \{\text{llevar}\}$ ;
- AGENTE OSADO:  $D_{O_2}^c = \{\text{llevar}, \text{pase}\}$ , las reglas  $IR_1$  e  $IR_2$  son aplicables e  $I_{O_2} = \{\text{llevar}\}$ .

En esta situación particular, ambos tipos de agente consideran la misma intención final. Como puede verse en la Figura 6.4, la opción de llevar la pelota es claramente preferible a patear al arco, hacer un pase, o moverse a un lugar.

### 6.1.6. Dominio de Aplicación: Sistema de Seguridad

A continuación presentaremos un ejemplo de implementación de un agente que controla un sistema de seguridad. El *agente de seguridad* controla los sensores ubicados en habitaciones de un edificio. Hay cuatro tipos de sensores: temperatura, humo, movimiento, y cámaras de video. Cuando se dispare un sensor de temperatura y/o de humo, el agente tendrá una razón para llamar a los bomberos; análogamente, cuando un sensor de movimiento y/o una cámara indique/n que un intruso pudo haber entrado a una habitación, la policía debería ser llamada. Estas instrucciones no representan un acto reflejo por parte del agente, sino que



existe una componente deliberativa, ya que los sensores podrían arrojar información contradictoria. Justamente, la idea detrás del emparejamiento de sensores es que actúen como respaldo mutuo: los sensores de humo con los de temperatura, y los sensores de movimiento con las cámaras.

Aunque las parejas de sensores proveen cierta robustez, también traen ciertas complicaciones, *e.g.*, un sensor de movimiento podría detectar un intruso, mientras que la cámara asociada no registra cambio en la imagen. Esto puede ocurrir por diversas razones, desde un mal funcionamiento hasta una fotografía pegada a la cámara. Lo mismo puede darse con el otro par de sensores; por ejemplo, tal vez alguien esté fumando en una habitación, disparando el sensor de humo pero no el de temperatura.

Toda inconsistencia de sensores que surja en una habitación es manejada por el agente enviando un guardia a esa habitación. Si el guardia confirma que un intruso ingresó a la habitación y/o la habitación está en llamas, accionará la alarma manual correspondiente, proveyendo una razón para hacer un llamado a la policía y/o los bomberos.

El agente de seguridad se define como:

$$A_s = \langle D, \langle \mathbf{E}_B, W_B, \bowtie_B, \mathbf{pref}_B \rangle, \langle \emptyset, W_F, \bowtie_F, \mathbf{pref}_F \rangle, T, IR, p \rangle$$

El conjunto de deseos es:

$$D = \{guard(R), llamar(b, R), llamar(p, R)\},$$

donde  $guard(R)$  significa que un guardia debe ser enviado a la habitación  $R$ , y los deseos  $llamar(b, R)$  y  $llamar(p, R)$  representan un llamado a los bomberos y la policía para verificar la habitación  $R$ , respectivamente.

Una diferencia fundamental con el dominio de fútbol robótico es que el agente de seguridad no posee el requerimiento de elegir sólo una intención.

Este agente tendrá la posibilidad de seleccionar una cantidad arbitraria de intenciones, e incluso podría no elegir ninguna. Por ejemplo, el agente podría enviar varios guardias a ciertas habitaciones mientras se hacen llamadas a la policía y a los bomberos para resolver situaciones en otras habitaciones. Para simplificar el ejemplo, asumiremos que el DAF de creencias del agente está compuesto sólo por la percepción, *i.e.*,  $\mathbf{B} = \phi$ . El DAF de filtrado  $\langle \emptyset, W_F, \bowtie_F, \mathbf{pref}_F \rangle$  de  $A_s$  es el ilustrado en la Figura 6.5.

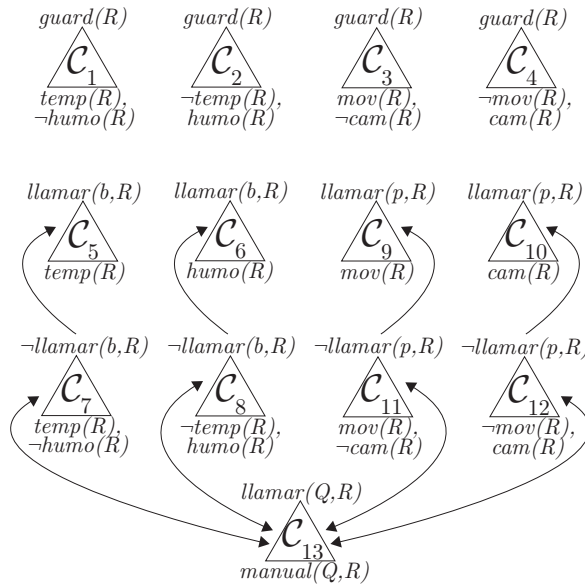


Fig. 6.5: DAF de filtrado del agente de seguridad

El agente de seguridad es de tipo cauto, *i.e.*,  $A_s$  considerará como actuales sólo aquellos deseos que se encuentren garantizados a partir de su base de conocimiento  $\langle \mathbf{B}, W_F, \bowtie_F, \mathbf{pref}_F \rangle$ . Debe notarse que todas las estructuras argumentales son primitivas, *i.e.*, compuestas por un único argumento, por lo cual nos referiremos a cada una de ellas directamente a través del argumento que la compone. También cabe aclarar que las letras mayúsculas indican variables que deben ser instanciadas. De alguna manera, cada argumento en el DAF de filtrado de la Figura 6.5

representa un esquema, y cada posible instanciación generará un nuevo argumento, refiriéndose a un nuevo cuarto –en particular, el argumento  $\mathcal{C}_{13}$  se verá instanciado con la fuerza de seguridad a llamar. Dado que la argumentación se realizará sobre argumentos instanciados, cada deliberación se referirá separadamente a cada habitación. Estos esquemas se introdujeron por simplicidad.

El conjunto  $IR$  de reglas de intención de  $A_s$  es:

$$IR_1 : (guard(R) \Leftarrow \{\}, \{manual(Q, R)\})$$

$$IR_2 : (llamar(b, R) \Leftarrow \{\}, \{\})$$

$$IR_3 : (llamar(p, R) \Leftarrow \{\}, \{\})$$

La regla  $IR_1$  impone una restricción: no se enviará un guardia a una habitación  $R$  en la cual se disparó una alarma manual, dado que el guardia que la disparó ya está en  $R$ . Debe resaltarse que implementar este comportamiento en las reglas de filtrado complicaría la representación, obligando a agregar la restricción de la alarma manual en las premisas de  $\mathcal{A}_1$ ,  $\mathcal{A}_2$ ,  $\mathcal{A}_3$  y  $\mathcal{A}_4$ . Una buena codificación de reglas de intención nos permite simplificar la especificación del DAF de filtrado.

La política  $p$  de selección de intenciones tomará todas las reglas de intención aplicables y retornará el conjunto de sus cabezas:

$$p(App) = \{h \mid (h \Leftarrow \{\cdot\}, \{\cdot\}) \in App\}$$

A continuación se dará una serie de conjuntos de creencias (en este caso, directamente percepciones), describiendo diferentes escenarios. Para cada uno de ellos, se calculará el conjunto de intenciones seleccionadas. Asumiremos que el agente de seguridad está controlando un edificio con

dos habitaciones:  $r1$  y  $r2$ . El conjunto inicial de creencias es:

$$B_0 = \left\{ \begin{array}{ll} temp(r1), & smoke(r1), \\ \neg cam(r1), & \neg mov(r1), \\ \neg temp(r2), & \neg smoke(r2), \\ \neg cam(r2), & \neg mov(r2) \end{array} \right\}$$

Las sentencias positivas indican que un sensor se disparó, y siguiendo la suposición del mundo abierto, lo contrario se indica mediante sentencias negativas. Se asume que el cálculo del conjunto de deseos actuales se realiza sólo cuando el conjunto de creencias cambia; de otra manera se enviarían guardias y se harían llamadas continuamente. La percepción en  $B_0$ , junto con las reglas de filtrado, nos dan dos estructuras argumentales sin derrotadores:  $C_5$  y  $C_6$ . El conjunto de deseos actuales es  $D_0^c = \{llamar(b, r1)\}$ , por lo cual la única regla de intención aplicable es  $IR_2$  y el conjunto de intenciones es  $I_0 = \{llamar(b, r1)\}$ .

Supongamos que el entorno cambia, y el sensor de movimiento en  $r2$  se dispara. El nuevo conjunto de creencias es:

$$B_1 = \left\{ \begin{array}{ll} temp(r1), & smoke(r1), \\ \neg cam(r1), & \neg mov(r1), \\ \neg temp(r2), & \neg smoke(r2), \\ \neg cam(r2), & mov(r2) \end{array} \right\}$$

La instancia activa de la base de conocimiento del agente de seguridad de acuerdo a  $B_1$  es la ilustrada en la Figura 6.6. Nuevamente, el deseo  $llamar(b, r1)$  está soportado por dos argumentos garantizados y, adicionalmente, se activa el argumento  $C_3$  para enviar un guardia a  $r2$ , y aunque se activa el argumento  $C_9$  para llamar a la policía, éste es derrotado por  $C_{11}$ . Finalmente, el conjunto de deseos actuales es  $D_1^c =$

$\{llamar(b, r1), guard(r2)\}$ . La regla de intención  $IR_1$  es aplicable, ya que la precondition  $guard(r2) \in D_1^c$  y la restriccion  $manual(Q, r2) \notin B_1$ . La regla  $IR_2$  también es aplicable, porque  $llamar(b, r1) \in D_1^c$ . Luego,  $I_1 = \{guard(r2), llamar(b, r1)\}$ .

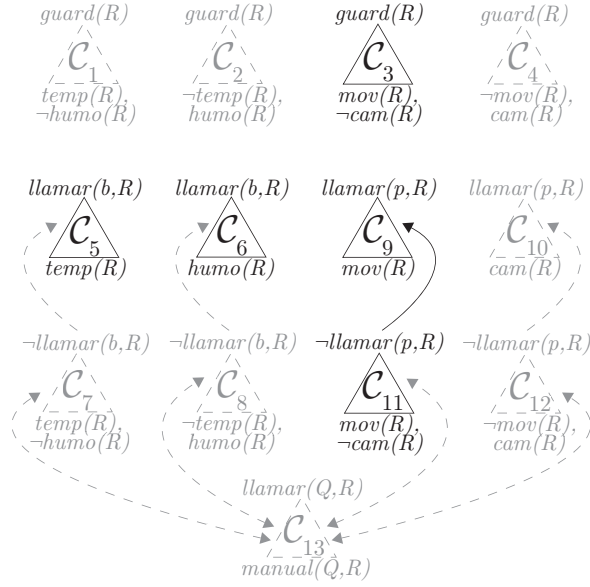
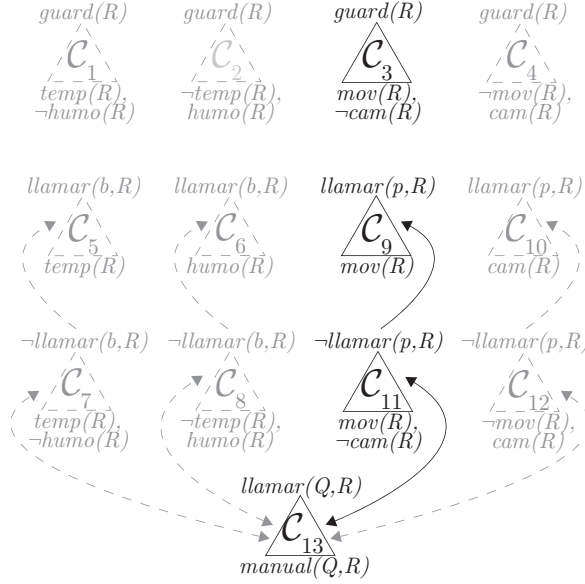


Fig. 6.6: Instancia activa del agente de seguridad según  $B_1$

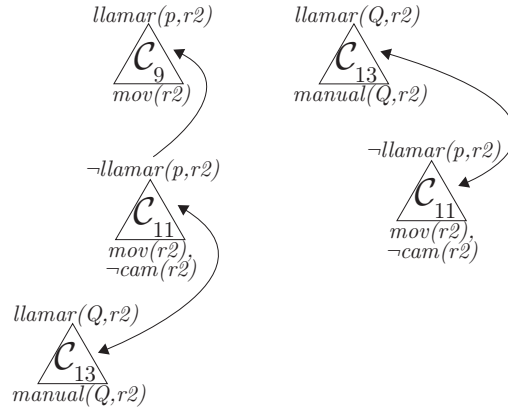
Ahora supongamos un nuevo escenario, en el cual los sensores de la habitación  $r1$  no demuestran actividad, pero como sólo el sensor de movimiento de  $r2$  se había disparado, un guardia fue enviado a verificar la existencia de un intruso. Asumamos que el guardia encontró un ladrón en  $r2$  y consecuentemente acciona la alarma manual, cambiando el conjunto de creencias de  $A_s$  a  $B_2$ :

$$B_2 = \left\{ \begin{array}{ll} \neg temp(r1), & \neg smoke(r1), \\ \neg cam(r1), & \neg mov(r1), \\ \neg temp(r2), & \neg smoke(r2), \\ \neg cam(r2), & mov(r2) \\ manual(p, r2) \end{array} \right\}$$

La instancia activa de la base de conocimiento de  $A_s$  es:



Si bien la estructura argumental  $C_{13}$  para  $llamar(p, r2)$  no está garantizada,  $C_9$  sí lo está. Los dos árboles de dialéctica para  $llamar(p, r2)$  son:



El deseo  $guard(r2)$  también está garantizado, por lo cual el conjunto de deseos actuales es  $D_2^c = \{llamar(p, r2), guard(r2)\}$ . Ahora la regla de intención  $IR_1$  no es aplicable porque  $manual(p, r2) \in B_2$ ; la única regla aplicable es  $IR_3$  y el conjunto de intenciones seleccionadas es  $I_2 = \{llamar(p, r2)\}$ . Cabe destacar que resulta sensato no enviar un guardia

a una habitación en la que ya hay un guardia y que, además, va a ser controlada por la policía.

Finalmente, consideremos un último caso que no registra actividad alguna en los sensores:

$$B_3 = \left\{ \begin{array}{ll} \neg temp(r1), & \neg smoke(r1), \\ \neg cam(r1), & \neg mov(r1), \\ \neg temp(r2), & \neg smoke(r2), \\ \neg cam(r2), & \neg mov(r2) \end{array} \right\}$$

A partir de estas creencias, no hay argumento alguno para los deseos a partir de la instancia activa de la base de conocimiento de  $A_s$ . Por lo tanto, el conjunto de deseos actuales es vacío, así como el de intenciones seleccionadas. El sistema permanecerá en este estado hasta que algún sensor se dispare.

La elección de un agente cauto en lugar de uno osado resulta clara al analizar este último escenario: un agente osado determinaría que todos los deseos son actuales, ya que no hay argumentos a favor ni en contra de ninguno y, por lo tanto, la negación de cada deseo no está garantizada. Luego, todas las reglas de intención estarían activas, provocando que se envíen guardias a ambas habitaciones y se realicen llamados a la policía y los bomberos para que las verifiquen. Este comportamiento no sería deseable para el agente de seguridad.

## 6.2. Análisis de Casos Legales

En algunos artículos sobre *argument theory change* (por ejemplo, en [MRF<sup>+</sup>09]) se tomó como ejemplo de aplicación el dominio de casos legales. En esta sección describiremos cómo utilizar la revisión a nivel de argumentos en este dominio para realizar razonamiento hipotético.

La utilización de formalismos basados en argumentación en relación a cuestiones legales constituye un campo bien establecido, que ya ha sido estudiado en la literatura [PS96, Ver03, WBC08].

El objetivo es utilizar el formalismo para posibilitar el estudio de un veredicto y así determinar la configuración del mundo que garantiza cierto argumento. El estado de garantía se computará a partir de árboles de dialéctica, y el argumento a garantizar será la presunción de inocencia. El resto de los nodos del árbol serán los alegatos presentados en el juicio. Primeramente mostraremos cómo utilizar el enfoque por contracción, indicando qué información debería haber estado ausente (u objetada) para que la presunción de inocencia se vea garantizada. Luego describiremos el caso simétrico, *i.e.*, el enfoque por expansión, indicando qué argumento/s debería/n haber estado activos para establecer la inocencia del acusado.

El caso que estudiaremos está basado en un hecho real, en el cual un hombre (Jack) es acusado de matar a su novia (Lucy). Este caso es particularmente interesante debido al gran número de alegatos presentados en el juicio; de todas formas en este ejemplo no se intentará imitar el curso real de los eventos. El resumen del caso es el siguiente: el fiscal reclama que el acusado es culpable porque asesinó y quemó a la chica en un fogón, mientras que la defensa alega que la chica murió por una sobredosis de cocaína y que luego fue quemada por el hombre en estado de desesperación.

Para que la exposición de este ejemplo resulte práctica, utilizaremos un subconjunto de los alegatos efectivamente presentados en el caso real. Cada alegato será un argumento, y se organizarán en el árbol de dialéctica que discute la inocencia de Jack. La raíz del árbol será el argumento inicial expuesto por la defensa, el cual estará derrotado, y luego será necesario aplicar la maquinaria lógica de ATC para verificar qué mundos



cercanos (en el sentido de mínimo cambio) habrían garantizado la inocencia del acusado. En este tipo de escenarios ATC demuestra que puede ser una herramienta útil para efectuar razonamiento hipotético. El árbol de dialéctica inicial se ilustra en la Figura 6.7.

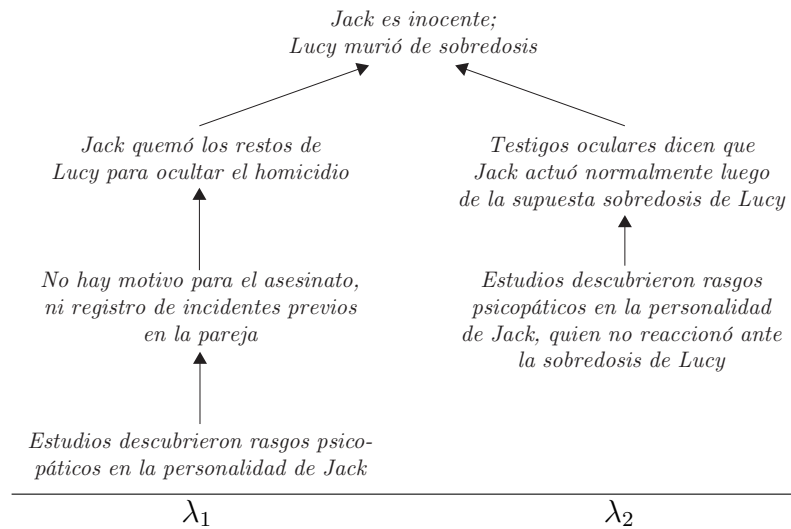


Fig. 6.7: Árbol de dialéctica con los alegatos iniciales

El conjunto de ataque en este árbol está compuesto por la línea de argumentación  $\lambda_1$  (ver Definición 4.1). Supongamos que la selección en esa línea es la estructura argumental hoja: “*Estudios descubrieron rasgos psicopáticos en la personalidad de Jack*”. La única opción para hacer una incisión en esta estructura es el argumento que la compone. Sin embargo, esto produce una incisión colateral sobre la hoja de la línea de argumentación  $\lambda_2$ : “*Estudios descubrieron rasgos psicopáticos en la personalidad de Jack, quien no reaccionó ante la sobredosis de Lucy*”. El segmento superior de esta estructura pertenece al conjunto de ataque del árbol hipotético producto de la incisión y su correspondiente incisión colateral; es decir, el segmento superior pertenece al conjunto de alteración. Siguiendo el principio de preservación, la selección en  $\lambda_2$  debe ubicarse por

encima de la incisión colateral. Dado que la única estructura argumental con es “*Testigos oculares dicen que Jack actuó normalmente luego de la supuesta sobredosis de Lucy*”, se realiza una incisión sobre ella, sin efectos colaterales. El árbol resultante, ilustrado en la Figura 6.8, garantiza la inocencia de Jack.

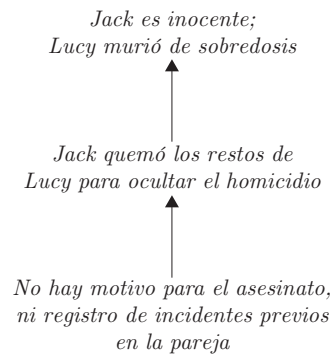


Fig. 6.8: Garantía de la inocencia de Jack, luego de la revisión

Ahora supongamos otra variante del mismo caso, con un número mayor de alegatos, y el objetivo de alcanzar la garantía de la inocencia de Jack a través de la activación de estructuras argumentales. El escenario resultante determinará qué evidencia debería estar disponible para llegar a tal veredicto. Asumamos un DAF cuyo conjunto de trabajo de argumento contiene los siguientes argumentos:

$\mathcal{A}$ : *Jack es inocente; Lucy murió de sobredosis.*

$\mathcal{B}_1$ : *Jack quemó los restos de Lucy para ocultar el homicidio.*

$\mathcal{B}_2$ : *Según amigos, Lucy nunca había consumido drogas y les temía.*

$\mathcal{B}_3$ : *No se encontraron rastros de violencia en la escena del crimen.*

$\mathcal{B}_4$ : *No hay motivo para el asesinato ni registro de incidentes previos en la pareja.*

$\mathcal{B}_5$ : *Una sobredosis puede ocurrir la primera vez que se consume.*

$\mathcal{B}_6$ : *Cabellos de Lucy fueron encontrados cerca de la escena del crimen, lo cual implica violencia.*

$\mathcal{B}_7$ : *Estudios descubrieron rasgos psicopáticos en la personalidad de Jack.*

$\mathcal{D}_1$ : *No se pudo determinar la causa de muerte a partir de la autopsia.*

$\mathcal{D}_2$ : *Si la causa de muerte no puede determinarse, no hay prueba de asesinato.*

$\mathcal{D}_3$ : *Si la causa de muerte no puede determinarse, no hay prueba de sobredosis.*

$\mathcal{D}_4$ : *Llamadas telefónicas grabadas sugieren que algunos amigos de Lucy fueron amenazados para declarar a favor de ella.*

Asumiremos que todos los argumentos están activos, con excepción de  $\mathcal{D}_1$  y  $\mathcal{D}_4$ . A partir de estos argumentos podrán construirse dos estructuras argumentales  $\mathcal{D}$  y  $\mathcal{D}'$ , tal que:  $\{\mathcal{D}_1, \mathcal{D}_2\} \in \text{args}(\mathcal{D})$  y  $\{\mathcal{D}_1, \mathcal{D}_3\} \in \text{args}(\mathcal{D}')$ . Tal como en el dominio de aplicación de fútbol de robots, en este caso las estructuras argumentales primitivas serán identificadas a través de su único argumento. Conflictos y preferencias se ilustran en el árbol de dialéctica activo de la Figura 6.9. Adicionalmente, se verifican los conflictos:  $\mathcal{D} \bowtie \mathcal{B}_1$ ,  $\mathcal{D}' \bowtie \mathcal{B}_5$ ,  $\mathcal{D}_4 \bowtie \mathcal{B}_2$ ; y la función de preferencia es tal que:  $\text{pref}(\mathcal{D}, \mathcal{B}_1) = \mathcal{B}_1$ ,  $\text{pref}(\mathcal{D}', \mathcal{B}_5) = \mathcal{D}'$ ,  $\text{pref}(\mathcal{D}_4, \mathcal{B}_2) = \mathcal{D}_4$ .

Suponiendo que la Figura 6.9 refleja el estado actual del juicio, supongamos que la defensa quiere saber cómo proceder para cambiar el resultado, *i.e.*, qué alegatos debería presentar para garantizar la inocencia de Jack. El conjunto de ataque es  $\{\lambda_1, \lambda_2\}$ , donde  $\lambda_1 = [\mathcal{A}, \mathcal{B}_1, \mathcal{B}_3, \mathcal{B}_6]$  y  $\lambda_2 = [\mathcal{A}, \mathcal{B}_1, \mathcal{B}_4, \mathcal{B}_7]$ .

En  $\lambda_1$ , asumiendo que  $\mathcal{B}_1 \prec \mathcal{B}_6$ , la selección sería la estructura argumental  $\mathcal{B}_1$ , y un derrotador para ella debe ser activado, por ejemplo,  $\mathcal{D}$ . La activación de esta estructura requiere la activación de  $\mathcal{D}_1$  ( $\mathcal{D}_2$  ya lo está). Notar que esta activación resulta en una *activación colateral* sobre



## 7. CONCLUSIONES

En esta tesis se propuso un nuevo marco argumentativo abstracto, denominado DAF, capaz de manejar dinámica de conocimiento. Tal como en los marcos abstractos propuestos en la literatura, en el DAF los argumentos se representan como entidades abstractas representando un paso de inferencia no necesariamente atado a la deducción clásica. Sin embargo, la abstracción se ve enriquecida por la noción intuitiva de argumento: cada uno de ellos es provisto de una interfaz que consiste de un conjunto de premisas y una conclusión. A partir de la inclusión de estos atributos, pueden surgir ciertas configuraciones que pueden resultar indeseables (*e.g.*, una conclusión contradiciendo una premisa), y para capturar esto se definió el concepto de argumento coherente, requiriendo las condiciones mínimas para que un argumento sea considerado como un paso de inferencia válido.

En los marcos abstractos, la relación de derrota entre argumentos suele asumirse como dada. En el DAF, en cambio, la relación de derrota surge a partir de la especificación de conflictos, y la utilización de una función de preferencia. Esto resulta en una versión extendida de esta relación, verificando una condición mínima de normalidad: la relación de conflicto debe contener aquellos pares de argumentos cuyas conclusiones son complementarias con respecto al lenguaje utilizado. Aparte de estos pares, pueden agregarse otros, denotando conflictos que no pueden detectarse sintácticamente. Esto podría verse como un atajo, ya que la

representación de conocimiento original podría extenderse para abarcar estos conflictos. Sin embargo, este tipo de artificios oscurecería la definición del marco. Luego, sobre cada conflicto, la función de preferencia determinará la derrota, considerando toda la información relevante a la activación de los argumentos en cuestión, *i.e.*, se evalúan las estructuras argumentales subyacentes.

La noción de estructura argumental se introdujo en la Sección 2.2.3, como un árbol de argumentos conectados de premisas a conclusiones. Esta entidad que emerge del sistema revela toda la información requerida por un dado argumento (el argumento tope de la estructura en cuestión) para efectivamente dar soporte a su conclusión. Ante ciertos escenarios las estructuras argumentales podrían presentar una configuración que no resulta aceptable como modo de inferencia (*e.g.*, contener dos argumentos en conflicto), por lo cual se definió el control que determina si una estructura argumental está o no bien formada. Este control es adicional al de coherencia de argumentos, y captura configuraciones anormales que no serían detectables por este último.

El aspecto que más diferencia al DAF de otros marcos argumentativos abstractos es la consideración especial de un conjunto de evidencia, cuyos elementos se representan según el lenguaje de premisas y conclusiones. La parte central del carácter dinámico del conocimiento se origina a partir de la variación de este conjunto, que representa el estado actual del entorno. Los argumentos activos son aquellos que encuentran sus premisas satisfechas o bien por la evidencia disponible, o por las conclusiones de argumentos activos. De esta manera, dado que el razonamiento se lleva a cabo mediante argumentos activos, la variación del conjunto de evidencia repercute en el conjunto de conclusiones garantizadas por el DAF. En la Sección 2.3 se definen las condiciones a partir de las cuales una estruc-

tura argumental bien formada se considera activa, y en la Sección 5.1 se estudia bajo qué condiciones un cambio en el conocimiento no incide sobre el estado de garantía de una dada estructura argumental.

La posibilidad de representar conocimiento válido pero inactivo de acuerdo a la situación actual permite realizar razonamiento hipotético y también formalizar nociones de la teoría del cambio aplicadas a argumentación. Estos tópicos fueron desarrollados en los capítulos 4 y 5, así como en la Sección 6.2. En particular, la técnica de poda explicada en la Sección 5.2 no sería posible sin la consideración del estado inicial provisto por los árboles de dialéctica potenciales, el cual incluye estructuras argumentales inactivas. La técnica depende de esta información para determinar los valores heurísticos asociados a las estructuras argumentales. Por último, la arquitectura BDI de agentes vista en la Sección 6.1 se vale de la dinámica del conocimiento para la actualización de la percepción, cuya repercusión no se acota al conjunto de creencias, sino que también determina la selección de deseos e intenciones.

En el Capítulo 3 se mostró cómo traducir los formalismos DELP y ASPIC en términos del DAF. De esta manera puede verse cómo el nivel de abstracción del DAF se adapta naturalmente a sistemas ya implementados, a diferencia del marco argumentativo de Dung, que sólo es capaz de brindar abstracciones de muy alto nivel. Por otra parte, mediante esta comparación es posible notar que el DAF resulta un formalismo útil para explorar la definición de propiedades a nivel teórico, para luego llevarlas a la implementación.

Parte del trabajo futuro de esta tesis consistirá en la implementación de los métodos de cambio y optimización vistos en los capítulos 4 y 5 en un formalismo práctico, como DELP. En particular, resulta muy interesante el estudio y explotación de técnicas de optimización en el cálculo de

garantía para evitar la recomputación al enfrentarse a una variación del conjunto de argumentos activos. Este estudio apuntaría a la definición de estructuras de datos (*e.g.*, arcos adicionales representando derrotas transitivas) para dar soporte al cálculo de garantía, estableciendo distintos tipos de dependencias entre los argumentos. A medida que el DAF vaya atravesando distintos estados, aumentará la calidad del conocimiento acerca de cómo ciertas configuraciones evidencia-argumentos afectan al estado de garantía de los argumentos, lo cual debería usufructuarse mediante técnicas de aprendizaje automatizado.

Por último, resulta necesario completar la teoría del formalismo ATC del Capítulo 4; los enfoques descritos en las secciones 4.2, 4.3 y 4.4 constituyen trabajo actualmente en curso. Este formalismo, incluso, podría combinarse con los resultados de optimización del cálculo de garantía, dado que ATC brinda información específica acerca de las distintas configuraciones de un DAF que dan garantía a un argumento.



## Bibliografía

- [ACGS08] Teresa Alsinet, Carlos Iván Chesñevar, Lluís Godo, and Guillermo Ricardo Simari. A logic programming framework for possibilistic argumentation: Formalization and logical properties. *Fuzzy Sets and Systems*, 159(10):1208–1228, 2008.
- [ACLSL08] Leila Amgoud, Claudette Cayrol, Marie Christine Lagasque-Schiex, and Paul Livet. On Bipolarity in Argumentation Frameworks. *International Journal of Intelligent Systems*, 23:1062–1093, 2008.
- [ADM07] Leila Amgoud, Yannis Dimopoulos, and Pavlos Moraitis. A general framework for argumentation-based negotiation. In *4th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS)*, pages 1–17, 2007.
- [AGM85] Carlos Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50:510–530, 1985.
- [BC02] Trevor Bench-Capon. Value-based argumentation frameworks. In *9th International Workshop on Non-Monotonic Reasoning (NMR)*, pages 443–454, 2002.

- [BCD07] Trevor Bench-Capon and Paul Dunne. Argumentation in artificial intelligence. *Artificial Intelligence*, 171(10-15):619–641, 2007.
- [BG07] Pietro Baroni and Massimiliano Giacomin. On Principle-Based Evaluation of Extension-Based Argumentation Semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.
- [BH06] Philippe Besnard and Anthony Hunter. Knowledgebase compilation for efficient logical argumentation. In *10th International Conference on the Principles of Knowledge Representation and Reasoning (KR)*, pages 123–133, 2006.
- [BH08] Philippe Besnard and Anthony Hunter. *Elements of Argumentation*. The MIT Press, 2008.
- [BH09] Philippe Besnard and Anthony Hunter. Argumentation based on classical logic. *Argumentation in Artificial Intelligence*, pages 133–152, 2009.
- [CCS05] Marcela Capobianco, Carlos I. Chesñevar, and Guillermo R. Simari. Argumentation and the dynamics of warranted beliefs in changing environments. *Journal of Autonomous Agents and Multi-Agent Systems*, 11:127–151, 2005.
- [CdLS08] Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasque-Schiex. Revision of an argumentation system. In *11th International Conference on the Principles of Knowledge Representation and Reasoning (KR)*, pages 124–134, 2008.
- [CM04] C. Chesñevar and A. Maguitman. ARGUENET: An Argument-Based Recommender System for Solving Web

- 
- Search Queries. In *2nd IEEE International IS-2004 Conference. Varna, Bulgaria*, pages 282–287, June 2004.
- [CMDK<sup>+</sup>07] Sylvie Coste-Marquis, Caroline Devred, Sébastien Konieczny, Marie-Christine Lagasquie-Schiex, and Pierre Marquis. On the merging of dung’s argumentation systems. *Artificial Intelligence*, 171(10-15):730–753, 2007.
- [CML00] Carlos I. Chesñevar, Ana G. Maguitman, and Ronald P. Loui. Logical Models of Argument. *ACM Computing Surveys*, 32(4):337–383, December 2000.
- [CMM<sup>+</sup>06] Carlos Chesñevar, Jarred McGinnis, Sanjay Modgil, Iyad Rahwan, Chris Reed, Guillermo Simari, Matthew South, Gerard Vreeswijk, and Steven Willmott. Towards an argument interchange format. *Knowledge Engineering Review*, 21(4):293–316, 2006.
- [CS06] C. Chesñevar and G. Simari. Modelling inference in argumentation through labelled deduction: Formalization and logical properties. *International Journal of Intelligent Systems (IJIS)*, 1(1):(in press), 2006.
- [CSG00] Carlos I. Chesñevar, Guillermo R. Simari, and Alejandro J. García. Pruning search space in defeasible argumentation. In *Workshop on Advances and Trends in AI*, pages 46–55, 2000.
- [DDBC05] P. Dunne, S. Doutre, and T. J. M. Bench-Capon. Discovering inconsistency through examination dialogues. In *19th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1680–1681, 2005.

- [DS01] Phan Minh Dung and Tran Cao Son. An argument-based approach to reasoning with specificity. *Artificial Intelligence*, 133(1–2):35–85, 2001.
- [Dun95] Phan Minh 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.
- [FKIS02] Marcelo A. Falappa, Gabriele Kern-Isberner, and Guillermo R. Simari. Belief revision, explanations and defeasible reasoning. *Artificial Intelligence Journal*, 141:1–28, 2002.
- [GRS07] Alejandro J. García, Nicolás D. Rotstein, and Guillermo R. Simari. Dialectical explanations in defeasible argumentation. In *European Conferences on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, pages 295–307, 2007.
- [GRTS07] Alejandro J. García, Nicolás D. Rotstein, Mariano Tucac, and Guillermo R. Simari. An argumentative reasoning service for deliberative agents. In *2nd International Conference on Knowledge Science, Engineering and Management (KSEM)*, pages 128–139, 2007.
- [GS04] Alejandro J. García and Guillermo R. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4(1-2):95–138, 2004.
- [Hun04] Anthony Hunter. Towards Higher Impact Argumentation. In *19th AAAI Conference on Artificial Intelligence*, pages 275–280. MIT Press, 2004.

- 
- [KM91] Hirofumi Katsuno and Alberto O. Mendelzon. On the difference between updating a knowledge base and revising it. In *2nd International Conference on the Principles of Knowledge Representation and Reasoning (KR)*, pages 387–394, 1991.
- [LID07] LIDIA. DeLP Web Client and Visualization Tool. [http://lidia.cs.uns.edu.ar/delp\\_client](http://lidia.cs.uns.edu.ar/delp_client), 2007.
- [MGS07] Diego C. Martínez, Alejandro J. García, and Guillermo R. Simari. Modelling Well-Structured Argumentation Lines. In *20th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 465–470, 2007.
- [MRF<sup>+</sup>09] Martín O. Moguillansky, Nicolás D. Rotstein, Marcelo A. Falappa, Alejandro J. García, and Guillermo R. Simari. Argument Theory Change Through Defeater Activation. In *9th Workshop on Computational Models of Natural Argument (CMNA)*, pages 24–33, 2009.
- [MT08] Paul-Amaury Matt and Francesca Toni. A game-theoretic measure of argument strength for abstract argumentation. In *JELIA*, pages 285–297, 2008.
- [Pol95] John Pollock. *Cognitive Carpentry: A Blueprint for How to Build a Person*. MIT Press, 1995.
- [Pro04] ASPIC Project. Theoretical framework for argumentation. *Deliverable D2.1 – [www.argumentation.org](http://www.argumentation.org)*, 2004.
- [PS96] Henry Prakken and Giovanni Sartor. A dialectical model of assessing conflicting arguments in legal reasoning. *Artificial Intelligence and Law*, 4(3-4):331–368, 1996.

- [PV02] Henry Prakken and Gerard Vreeswijk. Logical systems for defeasible argumentation. In D.Gabbay, editor, *Handbook of Philosophical Logic, 2nd ed.* Kluwer Academic Pub., 2002.
- [RA06] I. Rahwan and L. Amgoud. An argumentation-based approach for practical reasoning. In *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2006.
- [RGS07] Nicolás D. Rotstein, Alejandro J. García, and Guillermo R. Simari. Reasoning from Desires to Intentions: A Dialectical Framework. In *22nd AAAI Conference on Artificial Intelligence*, pages 136–141, 2007.
- [RGS08] Nicolás D. Rotstein, Alejandro J. García, and Guillermo R. Simari. Defeasible Argumentation Support for an Extended BDI Architecture. In *ArgMAS Post-proceedings*, pages 145–163, 2008.
- [RL08] Iyad Rahwan and Kate Larson. Mechanism design for abstract argumentation. In *6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1031–1038, 2008.
- [RLT09] Iyad Rahwan, Kate Larson, and Fernando A. Tohmé. A characterisation of strategy-proofness for grounded argumentation semantics. In *21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 251–256, 2009.
- [RMF<sup>+</sup>08a] Nicolás D. Rotstein, Martín O. Moguillansky, Marcelo A. Falappa, Alejandro J. García, and Guillermo R. Simari. Argument Theory Change Applied to Defeasible Logic Pro-

- 
- gramming. In *23rd AAAI Conference on Artificial Intelligence*, pages 132–137, 2008.
- [RMF<sup>+</sup>08b] Nicolás D. Rotstein, Martín O. Moguillansky, Marcelo A. Falappa, Alejandro J. García, and Guillermo R. Simari. Argument Theory Change: Revision Upon Warrant. In *2nd International Conference on Computational Models of Argument (COMMA)*, pages 336–347, 2008.
- [RMS09] Nicolás D. Rotstein, Martín O. Moguillansky, and Guillermo R. Simari. Dialectical abstract argumentation: A characterization of the marking criterion. In *21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 898–903, 2009.
- [RS09] Iyad Rahwan and Guillermo Simari. *Argumentation in Artificial Intelligence*. Springer, 2009.
- [RZR07a] Iyad Rahwan, Fouad Zablith, and Chris Reed. Laying the foundations for a world wide argument web. *Artificial Intelligence*, 171(10-15):897–921, 2007.
- [RZR07b] Iyad Rahwan, Fouad Zablith, and Chris Reed. Towards large scale argumentation support on the semantic web. In *22nd AAAI Conference on Artificial Intelligence*, pages 1446–1451, 2007.
- [SGCS03] Frieder Stolzenburg, Alejandro J. García, Carlos I. Chesñevar, and Guillermo R. Simari. Computing generalized specificity. *Journal of Applied Non-Classical Logics*, 13(1):87–113, January 2003.

- [SJNP98] Carles Sierra, Nick R. Jennings, Pablo Noriega, and Simon Parsons. A framework for argumentation-based negotiation. In Munindar P. Singh, Anand Rao, and Michael J. Wooldridge, editors, *4th International Workshop on Agent Theories, Architectures, and Languages (ATAL)*, volume 1365 of *LNAI*, pages 177–192, Berlin, July 24–26 1998. Springer.
- [Tou59] Stephen Toulmin. *The Uses of Argument*. Cambridge University Press, 1959.
- [Ver03] Bart Verheij. Dialectical argumentation with argumentation schemes: An approach to legal logic. *Artificial Intelligence and Law*, 11(2-3):167–195, 2003.
- [vRDM09] M. Birna van Riemsdijk, Mehdi Dastani, and John-Jules Ch. Meyer. Goals in conflict: semantic foundations of goals in agent programming. *Autonomous Agents and Multi-Agent Systems*, 18(3):471–500, 2009.
- [VTS09] I. Viglizzo, F. A. Tohmé, and G. R. Simari. A redefinition of arguments in defeasible logic programming. In T. Bench-Capon, S. Parsons, and H. Prakken, editors, *The Uses of Computational Argumentation*, pages 70–75, 2009.
- [Wal96] Douglas Walton. *Argument Structure: A Pragmatic Theory (Toronto Studies in Philosophy)*. University of Toronto Press, August 1996.
- [WBC07] Adam Wyner and Trevor Bench-Capon. Towards an extensible argumentation system. In *European Conferences on*



---

*Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, pages 283–294, 2007.

- [WBC08] Adam Wyner and Trevor Bench-Capon. Modelling Judicial Context in Argumentation Frameworks. In *2nd International Conference on Computational Models of Argument (COMMA)*, pages 417–428, 2008.