



Universidad Nacional del Sur

TESIS DE DOCTORADO EN CIENCIAS DE LA COMPUTACIÓN

*Diálogos colaborativo-argumentativos para compartir
conocimiento en sistemas multi-agente*

Ramiro Andrés Agis

BAHÍA BLANCA

ARGENTINA

2019

Prefacio

Esta Tesis se presenta como parte de los requisitos para optar al grado Académico de Doctorado 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 1 de Abril de 2016 y el 17 de Diciembre de 2019, bajo la dirección del Dr. Alejandro Javier García, Profesor Asociado del Departamento de Ciencias e Ingeniería de la Computación y del Dr. Sebastian Gottifredi, Profesor Adjunto del Departamento de Ciencias e Ingeniería de la Computación.

.....
Ramiro Andrés Agis

ramiro.agis@cs.uns.edu.ar

Departamento de Ciencias e Ingeniería de la Computación

Universidad Nacional del Sur

Bahía Blanca, 17 de Diciembre de 2019



UNIVERSIDAD NACIONAL DEL SUR
Secretaría General de Posgrado y Educación Continua

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

Resumen

En los sistemas multi-agente, los agentes necesitan poder dialogar para resolver conflictos de información, solucionar problemas, encontrar pruebas formales, o simplemente informarse mutuamente de los hechos pertinentes. En esta tesis se propondrán dos protocolos y estrategias de diálogo para dos tipos de diálogo colaborativo-argumentativos en sistemas multi-agente: *consulta a experto* e *indagación*.

En el diálogo de consulta a experto un agente es un experto en cierto dominio de conocimiento, y otro agente le consulta sobre un tema de ese dominio—el cual desconoce o conoce parcialmente—con el fin de adquirir información. Actualmente, no existen otras estrategias de diálogo de consulta a experto, y aquellas semejantes no consideran que el agente consultante podría tener información estricta en contradicción con las creencias del agente experto, ni que podrían tener diferentes preferencias entre argumentos.

Por otra parte, en el diálogo de indagación un grupo de agentes busca llegar a un acuerdo general sobre la respuesta a una pregunta, siendo imparciales y contribuyendo toda la información relevante que tengan. Si bien en la literatura existen otras estrategias de diálogo de indagación, éstas no consideran que los agentes podrían tener información estricta en contradicción con las creencias de los demás. Además, las pocas que no se limitan a sólo dos agentes no permiten que el conjunto de participantes varíe dinámicamente durante del diálogo.

En los dos protocolos y estrategias de diálogo que se definirán en esta tesis se confrontarán dichas limitaciones con el fin de proveer una mayor versatilidad que la de las propuestas ya existentes.

Abstract

In multi-agent systems, agents need to be able to dialogue in order to resolve information conflicts, solve problems, find formal proofs, or simply inform each other of the pertinent facts. In this thesis, two dialogue protocols and strategies will be proposed for two types of collaborative-argumentative dialogues in multi-agent systems: *expert consultation* and *inquiry*.

In the expert consultation dialogue, an agent is an expert in a certain field of knowledge, and another agent consults the expert on a subject of that domain—which does not know or partially knows about—in order to acquire information. Currently, there are no other expert consultation dialogue strategies, and those that are similar do not consider that the consulting agent may have strict information in contradiction with the expert agent’s beliefs, nor that they may have different preferences among arguments.

On the other hand, in the inquiry dialogue, a group of agents seeks to reach a general agreement on the answer to a question while being impartial and contributing all the relevant information they have. Although there are other inquiry dialogue strategies in the literature, these do not consider that agents may have strict information in contradiction with the beliefs of others. Moreover, the few that are not limited to only two agents do not allow the set of participants to vary dynamically during the dialogue.

In the two protocols and dialogue strategies that will be defined in this thesis, these limitations will be confronted in order to provide greater versatility than that of the existing proposals.

Índice general

1. Introducción	1
1.1. Contribuciones	3
1.2. Publicaciones	5
1.3. Organización de la tesis	5
2. Conceptos preliminares	7
2.1. Argumentación en sistemas multi-agente	7
2.2. Diálogos en sistemas multi-agente	9
2.3. Programación en Lógica Rebatible (DeLP)	12
2.3.1. Conceptos preliminares de DeLP	13
2.3.2. Programas lógicos rebatibles	15
2.3.3. Construcción de argumentos	17
2.3.4. Conflictos entre argumentos	19
2.3.5. Resolución de conflictos entre argumentos	19
2.3.6. Aceptabilidad	20
2.4. Agentes en DeLP	28
2.5. Sistemas de transiciones	29

3. Protocolo y estrategia de diálogo de consulta a experto	31
3.1. Introducción y motivación	32
3.2. Sesión de consulta a experto	36
3.2.1. Envío de justificación y adopción de argumentos	39
3.2.2. Introspección: ¿es necesario continuar la sesión?	45
3.2.3. Preguntas de preferencia: la sesión debe continuar	47
3.2.4. Preguntas de negador: requiriendo más argumentos	55
3.2.5. Ejemplo integrador	60
3.3. Resultados formales	64
3.4. Minimización o reducción de la sesión	65
3.5. Extensión: rechazo de la opinión del experto	72
3.6. Trabajo relacionado	73
3.7. Conclusiones	79
4. Protocolo y estrategia de diálogo de indagación	81
4.1. Introducción y motivación	82
4.2. Sesión de indagación	87
4.2.1. Adquirir información	90
4.2.2. Unirse y abandonar una sesión	91
4.2.3. Movimientos en el árbol de indagación	94
4.2.4. Fin de una ronda: ¿es necesario continuar la sesión?	106
4.2.5. Ejemplo integrador	107
4.3. Resultados formales	110
4.4. Extensiones	113
4.4.1. Indagación sobre un literal	113
4.4.2. Construcción conjunta de argumentos	114
4.4.3. Determinismo	115

4.5. Comparación empírica de criterios de selección de movimientos	116
4.6. Trabajo relacionado	120
4.7. Conclusiones	126
5. Conclusiones y trabajo a futuro	129
5.1. Trabajo a futuro	131
A. Pruebas formales	133
A.1. Apéndice A. Pruebas formales	133

Capítulo 1

Introducción

Disponer de herramientas adecuadas para compartir conocimiento resulta crucial en dominios de aplicación donde un grupo de agentes debe interactuar para resolver problemas complejos. Esto se acentúa aún más en aquellas aplicaciones donde es imprescindible que los agentes se desenvuelvan con autonomía y no es posible disponer de ningún tipo de control centralizado.

Afortunadamente, en los últimos años se han obtenido resultados significativos en este área, los cuales fueron usados como punto de partida para el desarrollo de esta tesis. En particular, en [WK95] se han propuesto y categorizado diferentes tipos de diálogos entre agentes inteligentes, los cuales permiten describir con precisión diferentes formas de compartir y adquirir conocimiento en una comunidad de agentes. Además, se han desarrollado nuevas herramientas de representación de conocimiento y razonamiento como la *argumentación* [PV02, SR09], la cual permite manejar de manera más natural información potencialmente contradictoria que proviene de diferentes fuentes o agentes. La argumentación consiste en la obtención de conclusiones a través de razonamiento lógico y es un componente importante de la inteligencia humana. La habilidad de sintetizar ideas en *argumentos* para razonar, expresar pensamientos y entender afirmaciones complejas, es un factor clave de las interacciones inteligentes entre cualquier ser en una sociedad. Por esta razón, diversas técnicas de argumentación han sido aplicadas durante años en el área de inteligencia artificial y sistemas multi-agente.

En esta tesis se proponen y describen formalmente dos protocolos y estrategias de diálogo para que agentes de un sistema multi-agente puedan compartir y adquirir conocimiento de sus pares mediante diálogos en los que las afirmaciones son sustentadas mediante

argumentos. En particular, se trabajó sobre dos tipos de interacciones muy importantes para el desarrollo de aplicaciones con sistemas multi-agente: cómo adquirir conocimiento de un agente experto y cómo un grupo de agentes puede resolver colaborativamente una cuestión. Para ello, se desarrolló un protocolo y estrategia de diálogo de *consulta a experto* y un protocolo y estrategia de diálogo de *indagación*.

En [WK95] se define *diálogo* como un framework normativo que comprende el intercambio de argumentos entre agentes, los cuales pueden tener objetivos individuales pero razonan juntos para lograr una meta colectiva. Uno de los tipos de diálogo colaborativos de la clasificación propuesta en dicho trabajo es el diálogo de *consulta a experto* (en inglés, *expert consultation*). En este tipo de diálogo, uno de los agentes es un experto en cierto dominio de conocimiento y el otro le consulta sobre un tema de ese dominio, el cual desconoce o conoce parcialmente. En particular, el objetivo es que el consultante obtenga la opinión calificada del experto sobre el tema en cuestión. Un diálogo de consulta a experto es considerado exitoso si, una vez finalizada la interacción, el consultante está de acuerdo con la opinión del experto.

Otro de los tipos de diálogo colaborativos definidos por [WK95] es el diálogo de *indagación* (en inglés, *inquiry*). En este tipo de diálogo un grupo de agentes busca llegar a un acuerdo general sobre la respuesta a una pregunta. Esto se realiza mediante un proceso acumulativo de argumentos que finaliza cuando los participantes llegan a una conclusión que, desde su punto de vista, es indudable. Es necesario que los participantes sean imparciales y contribuyan todas las conclusiones intermedias y premisas correspondientes. Un diálogo de indagación es considerado exitoso si se logra llegar a un acuerdo general sobre el tema en discusión.

Una solución *naïve* para ambos tipos de diálogo sería que los agentes unan sus bases de creencias. Sin embargo, esta no es una solución sensata ni factible porque podría implicar la violación de la privacidad de los agentes, podría ser computacionalmente impracticable, y probablemente conllevaría conflictos y contradicciones sin relación con el tema que motiva la interacción. Por esta razón, a partir de las definiciones propuestas por [WK95], surgieron varios interrogantes que motivaron y orientaron el desarrollo de la línea de investigación de esta tesis: ¿Cómo pueden modelarse formalmente ambos tipos de diálogo? ¿Qué argumentos deben compartir los agentes para que se cumplan los objetivos en cada tipo diálogo? ¿Qué pueden hacer los agentes cuando surge un conflicto de información? ¿Cuándo puede determinarse formalmente que un diálogo terminó exitosamente? Los

capítulos de esta tesis que desarrollan cada uno de los protocolos y estrategias propuestos dan respuesta a estos interrogantes.

Hay una diferencia sutil entre un *protocolo* de diálogo y una *estrategia* de diálogo [TG10, MTB⁺13, Thi14]: un protocolo permite construir diálogos coherentes, mientras que una estrategia permite construir diálogos con metas específicas. Precisamente, un protocolo de diálogo define un conjunto de *movimientos* o *locuciones* que los agentes pueden realizar, y determina cuándo tiene sentido realizarlos para que el diálogo resultante sea coherente. Por ejemplo, sólo tendría sentido que un agente rechace un argumento si éste fue expuesto previamente en el diálogo. Por otro lado, una estrategia de diálogo es la que define su tipo dado que determina el comportamiento que deberían tener los agentes para poder lograr sus metas. Por ejemplo, en un diálogo de consulta a experto, el agente consultante podría lograr su meta de estar de acuerdo con la opinión del experto pidiéndole más información si los argumentos que éste le envió no son suficientes.

En la literatura se han definido diversos protocolos de diálogo [AMP00, PWA02, AMP02, BH07, TGKIS08, BH09, Thi09, TG10, FT11, Thi14], algunos de los cuales proveen estrategias de ciertos tipos. Como se verá a continuación, las contribuciones de esta tesis son un protocolo y estrategia de diálogo de consulta a experto y un protocolo y estrategia de diálogo indagación, los cuales presentan mejoras en varios aspectos respecto a los trabajos relacionados existentes.

1.1. Contribuciones

Protocolo y estrategia de diálogo de consulta a experto

La primer contribución de esta tesis es la formalización de un protocolo y estrategia de diálogo de consulta a experto en el cual los agentes utilizan argumentación estructurada para la representación de conocimiento y razonamiento. El diálogo comienza con un *agente consultante* que realiza una consulta con el objetivo de adquirir información sobre un tema que desconoce o conoce parcialmente. En respuesta, un *agente experto* en el dominio de conocimiento de la consulta le envía un argumento que justifica su opinión. Dado que el consultante considera al otro agente un experto en el tema, su meta es aceptar la justificación que recibió en respuesta a su consulta. Sin embargo, como el consultante puede tener creencias previas, podría construir argumentos que estén en conflicto con la

opinión del experto. Por lo tanto, si el consultante está comprometido a lograr su meta de aceptar la justificación, tal vez necesite actualizar sus creencias y realizar preguntas al experto para adquirir más información.

El protocolo y estrategia de diálogo de consulta a experto que se propone en esta tesis incluye dos mejoras respecto a los trabajos similares presentes en la literatura [PWA02, FT12, FT15]. Primero, se considera que el agente consultante podría tener información estricta en contradicción con las creencias del agente experto sobre el tema consultado. Segundo, se considera que ambos agentes podrían tener diferentes preferencias entre argumentos. Por lo tanto, si el consultante está comprometido a lograr su meta de aceptar la justificación, tal vez necesite actualizar sus preferencias y eliminar creencias de su base que—desde la perspectiva del experto—son inválidas.

El protocolo y estrategia de diálogo de consulta a experto se definirá formalmente utilizando un sistema de transiciones. Éste mostrará cómo el diálogo debe evolucionar desde el estado inicial en el que el consultante todavía no consultó al experto hasta un estado final en el que el consultante logró aceptar la justificación. Se probará formalmente que todo diálogo desarrollado mediante este protocolo y estrategia termina exitosamente.

Protocolo y estrategia de diálogo de indagación

La segunda contribución de esta tesis es la formalización de un protocolo y estrategia de diálogo de indagación en el que los agentes utilizan argumentación estructurada para la representación de conocimiento y razonamiento. El diálogo comienza cuando se propone un argumento sobre el cual múltiples agentes desean llegar a un acuerdo general y finaliza cuando ninguno tiene más para contribuir. Los agentes pueden contribuir al diálogo dando contra-argumentos, contra-argumentos para esos contra-argumentos, y así sucesivamente, rechazando argumentos que consideran inválidos (dando una razón adecuada) u objetando información en contradicción con sus creencias. La meta de los participantes no es ganar una discusión o convencer a los otros de algo, sino contribuir imparcialmente todas sus creencias relevantes hasta llegar a un acuerdo general sobre el argumento de indagación.

El protocolo y estrategia de diálogo de indagación que se propone en esta tesis incluye cuatro mejoras respecto a los trabajos similares presentes en la literatura [PWA02, BH07, TGKIS08, Thi09, BH09, FT15]. Primero, se considera que los agentes podrían tener información estricta en contradicción con las creencias de los demás. Segundo, el diálogo no se limita a sólo dos agentes. Tercero, los participantes pueden optar

por abandonar el diálogo si no están dispuestos a seguir el protocolo. Por último, nuevos agentes pueden unirse libremente al diálogo incluso cuando ya ha comenzado.

El protocolo y estrategia de diálogo de indagación se definirá formalmente utilizando un sistema de transiciones. Éste mostrará cómo el diálogo debe evolucionar desde el estado inicial en el que se propone un argumento de indagación hasta un estado final en el que los participantes lograron llegar a un acuerdo. Se probará formalmente que todo diálogo desarrollado mediante este protocolo y estrategia termina exitosamente.

1.2. Publicaciones

Parte de los resultados obtenidos durante el desarrollo de esta tesis fueron publicados en revistas de la especialidad. A continuación se detallan dichos trabajos:

- En el trabajo “*Acquiring knowledge from expert agents in a structured argumentation setting*” [AGG19b] publicado en la revista *Argument & Computation* vol. 10 (2019) se definió el protocolo y estrategia de diálogo de consulta a experto que será desarrollado en el Capítulo 3 de esta tesis.
- En el trabajo “*An approach for distributed discussion and collaborative knowledge sharing: Theoretical and empirical analysis*” [AGG19a] publicado en la revista *Expert Systems with Applications* vol. 116 (2019) se definió el protocolo y estrategia de diálogo de indagación que será desarrollado en el Capítulo 4 de esta tesis.

1.3. Organización de la tesis

El resto de esta tesis se encuentra organizado de la siguiente manera:

- En el Capítulo 2 se presentarán los conceptos necesarios para el desarrollo de esta tesis: el formalismo de representación de conocimiento y razonamiento DeLP, agentes y sistemas de transiciones.
- En el Capítulo 3 se presentará un protocolo y estrategia de diálogo de consulta a experto para dos agentes que utilizan DeLP como formalismo de representación de conocimiento y razonamiento.

- En el Capítulo 4 se presentará un protocolo y estrategia de diálogo de indagación para múltiples agentes que utilizan DeLP como formalismo de representación de conocimiento y razonamiento.
- En el Capítulo 5 se presentarán las conclusiones de esta tesis y el trabajo a futuro.
- En el Apéndice A se incluyen todas las propiedades introducidas a lo largo de esta tesis junto a sus correspondientes demostraciones.

Capítulo 2

Conceptos preliminares

En este capítulo se presentarán los conceptos necesarios para el desarrollo de esta tesis. En primer lugar, en la Sección 2.1, se hablará sobre argumentación en sistemas multi-agente y sus aplicaciones. Luego, en la Sección 2.2, se hablará sobre diálogos en sistemas multi-agente, su clasificación de tipos, y protocolos y estrategias de diálogo. En la Sección 2.3 se presentará el formalismo de representación de conocimiento y razonamiento que será utilizado por los agentes. Después, en la Sección 2.4, se formalizará el concepto de agente utilizado en esta tesis. Finalmente, en la Sección 2.5, se presentará la herramienta que será utilizada para definir formalmente los protocolos y estrategias de diálogo.

2.1. Argumentación en sistemas multi-agente

La argumentación es una forma de razonamiento en la cual, para una determinada afirmación, se consideran los argumentos presentados y se resuelven los posibles conflictos entre ellos. En particular, una afirmación es aceptada o rechazada en función del análisis de los argumentos a su favor y en su contra. La forma en que los argumentos para una afirmación son considerados permite definir un mecanismo de razonamiento automático que contempla la posibilidad de contar con información contradictoria, incompleta e incierta [PV02, SR09].

El estudio de la argumentación ha sido abordado desde diferentes enfoques: a nivel lógico puede verse como una forma de modelar inferencia rebatible, y a nivel dialógico puede verse como una forma de interacción entre agentes inteligentes. La argumentación

ha sido ampliamente estudiada en la filosofía [Tou58, Pol87], y desde los años '70 es posible encontrar estudios en las ciencias de la computación que contribuyeron notablemente a la noción de argumento. A mediados de los años '80 comenzaron los desarrollos desde un punto de vista computacional, en los que los argumentos son explícitamente construidos y comparados como medios para resolver problemas.

Existen diferentes aproximaciones para modelar argumentación, de las cuales se destacan los sistemas de argumentación abstracta [Dun95, Pra09] y los sistemas argumentativos estructurados o basados en reglas [PS97, GS04, DKT06, AK07, MP14, Ton14, GS14]. Además, la argumentación ha sido empleada exitosamente en diversos dominios y aplicaciones dentro del área de inteligencia artificial como conceptualización del razonamiento de sentido común, razonamiento legal, tomas de decisiones, sistemas multi-agente (SMA), entre muchos otros.

En los SMA, los agentes pueden tener diferentes objetivos y es normal asumir que su comportamiento no tiene un control centralizado. De hecho, una de las ventajas de los SMA es que la información es descentralizada. Por lo tanto, los agentes deben interactuar para obtener la información que necesitan, o para compartir sus creencias. En términos generales, los agentes en los SMA suelen interactuar para tomar decisiones colectivas o llegar a acuerdos. Dado que esas interacciones pueden ser modeladas utilizando técnicas de argumentación, su aplicación en SMA ha sido durante décadas un campo de investigación muy activo [CI15].

La aplicación de técnicas de argumentación en los SMA ha sido utilizada para resolver diversos problemas concretos de la vida real en dominios diferentes. En particular, ha sido utilizada como mecanismo de negociación o clasificación en el campo del *e-commerce*, por ejemplo, para encontrar productos potencialmente interesantes para el usuario [HL10], para hacer tratos con proveedores y clientes [ZJH10], y para negociar estrategias de suministro [WWW10], entre otros. En el campo de las organizaciones virtuales ha sido utilizada, por ejemplo, para razonar con información incompleta y contradictoria [JH12], para analizar factores emocionales [MRN05], para decidir si una persona puede solicitar un beneficio particular con un conjunto de restricciones [WCBC12], para asignar crédito [PTB14], para construir modelos de reputación [HGAB10], para la resolución de conflictos en cadenas de suministro [HGAB06], y para construir sistemas de inteligencia ambiental [MS07, FO12]. En el campo de la administración industrial ha sido utilizada en sistemas de gestión para múltiples tareas, por ejemplo, para gestionar las descargas de aguas residua-

les [ATT⁺12], para la toma de decisiones en un proceso de carga automático [CSCC13], y para mejorar el rendimiento de los sistemas de transporte en áreas rurales [VRO⁺12]. En el campo de la ingeniería del conocimiento ha sido utilizada, por ejemplo, para alinear automáticamente ontologías o bases de conocimiento [MSC11, MS14], y para construir un sistema de soporte a decisiones para la especificación y promulgación de protocolos de tratamiento médicos [GGB12]. Estos son unos pocos ejemplos de los trabajos de la literatura que han aplicado técnicas de argumentación en los SMA para resolver diversos problemas concretos de la vida real, lo cual muestra su idoneidad para entornos complejos y heterogéneos.

Esta exitosa combinación se debe principalmente a la inmensa y diversa cantidad de avances teóricos en el área, la cual incluye—entre otros—modelos BDI argumentativos [RGS02, RA06, RGS07, SFEA10, Ber14, dOGAP⁺18], el uso de argumentación en confianza y reputación multi-agente [BV14, GTGS18], y diálogos argumentativos en SMA.

2.2. Diálogos en sistemas multi-agente

Al construir sistemas multi-agente, damos por sentado el hecho de que los agentes que componen el sistema deberán comunicarse. Los agentes necesitan comunicarse para resolver conflictos de información o de intereses, solucionar problemas, encontrar pruebas, o simplemente informarse mutuamente de los hechos pertinentes. Muchos de estos requisitos de comunicación no pueden cumplirse mediante el intercambio de mensajes individuales, sino que los agentes en cuestión deben poder intercambiar una secuencia de mensajes con coherencia y cohesión. En otras palabras, necesitan la capacidad de dialogar.

En [WK95] se define *diálogo* como un framework normativo que comprende el intercambio de argumentos entre agentes, los cuales pueden tener objetivos individuales pero razonan juntos para lograr una meta colectiva. Los participantes de un diálogo típicamente hacen afirmaciones y las sustentan con razones. Como suelen estar en desacuerdo con las posiciones de los demás, la argumentación consiste en explorar y tal vez resolver dichos conflictos.

A pesar de que el desacuerdo es parte de la naturaleza de un diálogo argumentativo, esto no quiere decir que los agentes no puedan llevar a cabo un *diálogo colaborativo*. La diferencia radica en que, en un diálogo colaborativo, no todos los participantes persiguen

un objetivo individual que involucra un beneficio personal. Por ejemplo, un grupo de agentes podrían dar imparcialmente argumentos a favor y en contra para evaluar la validez de cierta pieza de información. Incluso podrían combinar sus creencias para construir argumentos que no serían posibles de construir con las creencias individuales de cada uno. En este escenario los participantes del diálogo se benefician de una conclusión obtenida a partir de un razonamiento más informado y refinado. Otro ejemplo: un agente podría enseñarle a otro sobre cierto tema dándole argumentos que sustenten la validez de sus afirmaciones, y el agente que adquiere información podría contra-argumentar en caso de estar en desacuerdo con su posición. En este escenario uno de los participantes del diálogo no obtiene beneficio alguno; sólo comparte información al otro.

Dentro de la clasificación de diálogos propuesta por [WK95], la cual ha sido recurrentemente utilizada en el área de argumentación, aquellos diálogos que son considerados colaborativos son el diálogo de *búsqueda de información*, el diálogo de *indagación* y el diálogo de *deliberación*:

- *Búsqueda de información* (en inglés, *information-seeking*): Uno de los participantes tiene cierto conocimiento o está en posición de saber algo, y el otro carece y necesita esa información. El objetivo es la difusión de dicha información de un agente al otro. Ver *e. g.*, [PWA02, FT12, FT15]. A su vez se subclasifica en:
 - *Consulta a experto* (en inglés, *expert consultation*): Uno de los participantes es un experto en cierto dominio de conocimiento y el otro le consulta sobre un tema de ese dominio, el cual desconoce o conoce parcialmente. El objetivo es que el consultante obtenga la opinión calificada del experto sobre el tema en cuestión. El diálogo se considera exitoso si, una vez finalizado, el consultante está de acuerdo con la opinión del experto. Ver *e. g.*, [AGG19b].
- *Indagación* (en inglés, *inquiry*): Un grupo de agentes busca llegar a un acuerdo general sobre la respuesta a una pregunta. Esto se hace mediante un proceso acumulativo de argumentos que finaliza cuando los participantes llegan a una conclusión que, desde su punto de vista, es indudable. Es necesario que los participantes sean imparciales, y que contribuyan todas las conclusiones intermedias y premisas correspondientes. El diálogo es considerado exitoso si se logra llegar a un acuerdo general sobre el tema en discusión. Ver *e. g.*, [PWA02, BH07, TGKIS08, Thi09, BH09, BA09, FT15, AGG19a].

- *Deliberación* (en inglés, *deliberation*): Consiste en la decisión sobre cómo actuar ante una circunstancia o problema y la formación de un plan adecuado. Los participantes tienen que considerar diferentes opciones y los posibles resultados dependiendo del estado actual del mundo. Ver *e. g.*, [KM03, Amg05, MS07, AP09, ATT⁺12, JH12, VRO⁺12, GGB12, CSCC13, FO13, FCS⁺13, FTMW14, GGL14, TCT15].

Por otro parte, los diálogos que no son considerados colaborativos son el diálogo de *persuasión*, el diálogo de *negociación*, y el diálogo *erístico*:

- *Persuasión* (en inglés, *persuasion*): Consiste en la resolución de un conflicto de puntos de vista, la cual implica el cambio de la opinión de uno de los participantes. Cada agente busca hacer prevalecer su propio punto de vista y persuadir al otro participante de aceptarlo. Ver *e. g.*, [AP01, Ben03, Pra06, BA09, DR09, HL10, RAPB11, vdWDM⁺11, HJBJ13a, HJBJ13b, RK16].
- *Negociación* (en inglés, *negotiation*): Consiste en hacer un trato en el cual cada agente busca maximizar su beneficio. El trato es llevado a cabo mediante un proceso de negociación de interés propio, donde la estrategia consiste en llegar a un acuerdo que sea atractivo para ambas partes. [Kra97, PSJ98, APM00, MVEPA03, RRJ⁺03, MB04, Amg05, BDC08, ZJH10, YYH⁺11, WWW10, MA11, ESM12, MS14, PMVB15].
- *Erístico* (en inglés, *eristic*): Sirven como sustituto de una pelea. Ver *e. g.*, [Wis03].

Dado que [WK95] no presenta definiciones formales para cada tipo de diálogo, en los últimos años se han desarrollado protocolos y estrategias muy diversas, cada uno con sus particularidades, ventajas y desventajas.

Hay una diferencia sutil entre un *protocolo* de diálogo y una *estrategia* de diálogo [TG10, MTB⁺13, Thi14]: un protocolo permite construir diálogos coherentes, mientras que una estrategia permite construir diálogos con metas específicas. Precisamente, un protocolo de diálogo define un conjunto de *movimientos* o *locuciones* que los agentes pueden realizar, y determina cuándo tiene sentido realizarlos para que el diálogo resultante sea coherente. Por ejemplo, sólo tendría sentido que un agente rechace un argumento si éste fue expuesto previamente en el diálogo. Por otro lado, una estrategia de diálogo es la que define su tipo dado que determina el comportamiento que deberían tener los agentes para poder lograr sus metas. Por ejemplo, en un diálogo de consulta a experto,

el agente consultante podría lograr su meta de creer en la opinión del experto pidiéndole más información si los argumentos que éste le envió no son suficientes.

2.3. Programación en Lógica Rebatible (DeLP)

En esta sección se presentará el formalismo de representación de conocimiento y razonamiento *Programación en Lógica Rebatible* (DeLP, por su nombre en inglés *Defeasible Logic Programming*) [GS04, GS14]. Este formalismo combina resultados de programación en lógica y argumentación rebatible, y ha sido aplicado exitosamente en diferentes dominios de aplicación.

DeLP adopta como lenguaje lógico de representación una extensión de la programación en lógica que incorpora la negación fuerte y permite representar información estricta e información rebatible. Esta información es modelada a través de programas lógicos rebatibles, los cuales constituyen una base de conocimiento. Luego, a partir de estos programas es posible construir argumentos, identificar conflictos entre ellos (mediante el uso de negación fuerte), determinar las derrotas entre ellos (mediante el uso de un criterio de comparación) y utilizar procedimientos de prueba dialécticos para determinar cuáles son los argumentos aceptados del sistema.

En esta tesis se utilizará la versión de DeLP en la que la información se representa únicamente con hechos y reglas rebatibles¹. El motivo de esta elección radica en que esta versión de DeLP ha resultado adecuada para entornos dinámicos [CCS05, RGS07, GGS09, GGS10, AGG19a, AGG19b] como los que son tratados en esta tesis, en especial, en agentes inteligentes.

Un *programa lógico rebatible* está formado por un conjunto de *hechos*, los cuales representan información estricta, y *reglas rebatibles*, las cuales representan información tentativa y potencialmente contradictoria. Toda conclusión del programa deberá estar sustentada por algún *argumento* que pueda construirse utilizando los hechos y las reglas rebatibles del programa. Cuando se utilicen reglas rebatibles para derivar una conclusión c , esta conclusión será tentativa y podrá ser refutada por información que la contradiga. De esta manera, un argumento que sustenta una conclusión c podrá ser atacado por otros *contra-argumentos* que lo contradigan y que posiblemente resulten ser *derrotadores* para

¹En el trabajo original de DeLP [GS04, GS14] también se utilizan reglas estrictas.

el mismo. De manera similar, estos derrotadores podrán a su vez ser atacados, dando así lugar a nuevos derrotadores. Entonces, para decidir cuándo una conclusión c puede aceptarse a partir de un programa lógico rebatible, se realizará un análisis dialéctico considerando los argumentos a su favor y en su contra. Esto conducirá a la construcción de una estructura arbórea de derrotadores conocida como *árbol de dialéctica*. Finalmente, una conclusión c estará *garantizada* si existe un argumento para c que sobreviva a todas las derrotas que recibe en su árbol de dialéctica asociado.

A continuación, se introducirán algunos conceptos preliminares de la programación en lógica que son utilizados por DeLP. Luego, se presentará la sintaxis del lenguaje de representación de los programas lógicos rebatibles, y se indicará cómo se efectúa la construcción de argumentos, la identificación de conflictos entre argumentos y su resolución para obtener las derrotas correspondientes. Finalmente, se presentará un procedimiento de prueba dialéctico que permite inferir las conclusiones garantizadas a partir de un programa lógico rebatible.

2.3.1. Conceptos preliminares de DeLP

En esta sección se introducirán conceptos y terminología estándar de la lógica y la programación en lógica que resultan necesarios para la formalización de DeLP.

Definición 2.1 (Signatura) Una *signatura* es una tupla $\sigma = \langle \mathcal{V}, Func, Pred \rangle$, donde \mathcal{V} , $Func$, y $Pred$ son conjuntos finitos que representan variables, funciones y predicados respectivamente. ■

Dada una signatura σ , la función ARIDAD le asigna a cada elemento de $Func$ y $Pred$ un número entero positivo. Si $f \in Func$ y $ARIDAD(f) = 0$ entonces f se denomina *constante*. Por otra parte, si $p \in Pred$, y $ARIDAD(p) = 0$ entonces p se denomina *proposición*.

Definición 2.2 (Alfabeto) El *alfabeto* generado a partir de una signatura σ consiste del conjunto de elementos miembros de la signatura, el símbolo de negación fuerte ‘ \sim ’, y los símbolos de puntuación ‘(, ’’, y ‘,’. ■

Definición 2.3 (Término) Sea $\sigma = \langle \mathcal{V}, Func, Pred \rangle$ una signatura, un *término* t y sus *componentes* $COMP(t)$ se definen inductivamente como sigue:

- Toda variable $v \in \mathcal{V}$ es un término, y $\text{COMP}(v) = \{v\}$.
- Toda constante $c \in \text{Func}$ (i. e., $\text{ARIDAD}(c) = 0$) es un término, y $\text{COMP}(c) = \{c\}$.
- Si $f \in \text{Func}$ tal que $\text{ARIDAD}(f) = n$ ($n \geq 1$) y t_1, \dots, t_n son términos, entonces $f(t_1, \dots, t_n)$ es un término, y $\text{COMP}(f(t_1, \dots, t_n)) = \{f\} \cup \left(\bigcup_{1 \leq i \leq n} \text{COMP}(f_i)\right)$. ■

En DeLP, todos los elementos de un programa están compuestos por *literales fijos* (negativos o positivos), los cuales a su vez están compuestos por un predicado y un conjunto de *términos fijos*, conceptos que son definidos a continuación.

Definición 2.4 (Término fijo) Sea $\sigma = \langle \mathcal{V}, \text{Func}, \text{Pred} \rangle$ una signatura y t un término, t es un *término fijo* si $\text{COMP}(t) \cap \mathcal{V} = \emptyset$, i. e., no contiene variables. ■

Definición 2.5 (Átomo) Sea $\sigma = \langle \mathcal{V}, \text{Func}, \text{Pred} \rangle$ una signatura y t_1, \dots, t_n términos, si $p \in \text{Pred}$ tal que $\text{ARIDAD}(p) = n$ entonces $p(t_1, \dots, t_n)$ es un *átomo*. Un *átomo fijo* es un átomo $p(t_1, \dots, t_n)$ tal que todos sus términos t_1, \dots, t_n son fijos. ■

Definición 2.6 (Literal) Un *literal* es un átomo a o un átomo negado $\sim a$. Un literal es *negativo* si es un átomo negado, o *positivo* en caso contrario. Un literal es *fijo* si es un átomo fijo o un átomo fijo negado. ■

La definición anterior establece que los literales podrán ser átomos negados utilizando el símbolo ‘ \sim ’ de la *negación fuerte* (en inglés, *strong negation*)². Por una parte, este tipo de negación difiere de la negación utilizada en la lógica clásica ya que, por ejemplo, no permite anidamiento (i. e., ‘ $\sim \sim a$ ’ no es un literal válido). Por otra parte, la negación fuerte también difiere de la *negación por falla* utilizada en la programación en lógica, la cual es usualmente denotada con el símbolo ‘*not*’.

Definición 2.7 (Complemento de un literal) Sea l un literal y a un átomo, el *complemento* de l con respecto a la negación fuerte, denotado \bar{l} , se define como:

$$\bar{l} = \begin{cases} a, & \text{si } l = \sim a \\ \sim a, & \text{si } l = a \end{cases} \quad \blacksquare$$

²La definición y propiedades de la negación fuerte pueden encontrarse en [AP93, PA93].

2.3.2. Programas lógicos rebatibles

Un programa lógico rebatible está compuesto por *hechos* y *reglas rebatibles*, cuya sintaxis se define a continuación.

Definición 2.8 (Hecho) Un *hecho* es un literal fijo l . ■

Definición 2.9 (Regla rebatible) Una *regla rebatible* es un par ordenado, denotado $Cabeza \prec Cuerpo$, donde $Cabeza$ es un literal fijo y $Cuerpo$ es un conjunto finito de literales fijos. Una regla rebatible con cabeza l_0 y cuerpo $\{l_1, \dots, l_n\}$ ($n \geq 0$) se denota $l_0 \prec l_1, \dots, l_n$. ■

Pragmáticamente, los hechos se emplean para representar conocimiento seguro y libre de excepciones, mientras que las reglas rebatibles son reglas de inferencia que se utilizan para representar información tentativa, la cual puede ser cuestionada. En general, una regla rebatible $Cabeza \prec Cuerpo$ se lee como “ $Cuerpo$ da una razón para inferir $Cabeza$ ”.

Definición 2.10 (Programa lógico rebatible) Un *programa lógico rebatible* \mathcal{P} es un conjunto de hechos no contradictorios y reglas rebatibles. Se identifica con Π al subconjunto de hechos y con Δ al subconjunto de reglas rebatibles, y se denota a \mathcal{P} mediante el par $\mathcal{P} = (\Pi, \Delta)$. ■

Ejemplo 2.1 El programa lógico rebatible $\mathcal{P} = (\Pi, \Delta)$ representa información sobre el mercado bursátil:

$$\Pi = \left\{ \begin{array}{l} \text{buen_precio(umbrella)} \\ \text{en_fusión(umbrella)} \\ \sim \text{banca_rota(umbrella)} \end{array} \right\} \quad \Delta = \left\{ \begin{array}{l} \text{comprar_acciones}(X) \prec \text{buen_precio}(X) \\ \sim \text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \sim \text{comprar_acciones}(X) \prec \text{banca_rota}(X) \\ \text{riegosa}(X) \prec \text{en_fusión}(X) \\ \text{riegosa}(X) \prec \text{con_deuda}(X) \end{array} \right\}$$

Note que los literales en las reglas rebatibles pueden tener variables. No obstante, DeLP utiliza instancias fijas de estas reglas para determinar los literales que deriva. Una *instancia fija* para una regla rebatible es una versión de la regla tal que las variables son reemplazadas por términos fijos, donde se asume que las variables con el mismo nombre

dentro de la regla representan el mismo elemento. Cada regla rebatible con variables constituye entonces un esquema que representa al conjunto de sus instancias fijas.

Definición 2.11 (Instancia fija de una regla rebatible) Sea r una regla rebatible, una *instancia fija* $r\sigma$ de r se obtiene reemplazando cada una de las variables de r por un término fijo de forma tal que las variables de r con el mismo nombre sean reemplazadas consistentemente. ■

La siguiente definición determina qué literales pueden ser *derivados* a partir de un programa lógico rebatible.

Definición 2.12 (Derivación de un literal) Sea $\mathcal{P} = (\Pi, \Delta)$ un programa lógico rebatible y l un literal, una *derivación* para l a partir de \mathcal{P} es una secuencia finita de literales fijos $l_1, l_2, \dots, l_n = l$ tal que l_i ($1 \leq i \leq n$) pertenece a la secuencia porque:

- l_i es un hecho en Π , o
- existe en \mathcal{P} una instancia fija $r_i\sigma$ de una regla rebatible r_i con cabeza l_i y cuerpo c_1, \dots, c_k , donde todo literal c_j del cuerpo ($0 \leq j \leq k$) es un elemento de la secuencia que precede a l_i . ■

Ejemplo 2.2 En el programa lógico rebatible $\mathcal{P} = (\Pi, \Delta)$ del Ejemplo 2.1 existen las siguientes derivaciones:

La secuencia $\text{buen_precio}(\text{umbrella}), \text{comprar_acciones}(\text{umbrella})$ es una derivación para $\text{comprar_acciones}(\text{umbrella})$.

La secuencia $\text{en_fusión}(\text{umbrella}), \text{riesgosa}(\text{umbrella}), \sim\text{comprar_acciones}(\text{umbrella})$ es una derivación para $\sim\text{comprar_acciones}(\text{umbrella})$.

La secuencia $\text{en_fusión}(\text{umbrella}), \text{riesgosa}(\text{umbrella})$ es una derivación para $\text{riesgosa}(\text{umbrella})$.

La secuencia $\text{buen_precio}(\text{umbrella})$ es una derivación para $\text{buen_precio}(\text{umbrella})$.

La secuencia $\text{en_fusión}(\text{umbrella})$ es una derivación para $\text{en_fusión}(\text{umbrella})$.

La secuencia $\sim\text{banca_rota}(\text{umbrella})$ es una derivación para $\sim\text{banca_rota}(\text{umbrella})$ ■

Note que a partir de un programa lógico rebatible es posible derivar literales complementarios y que todo hecho tiene una derivación. Además, podrían existir múltiples derivaciones para un mismo literal.

Los reglas rebatibles permiten expresar información potencialmente contradictoria, con las cuales es posible obtener derivaciones para un literal y su complemento. Esto sugiere que la noción de derivación no resulta conveniente como procedimiento de prueba para un literal. Por lo tanto, para decidir si un literal l está garantizado a partir de un programa lógico rebatible se realiza un análisis global más profundo considerando toda la información relevante del mismo.

En particular, DeLP define un procedimiento de prueba para decidir qué literales están garantizados a partir de un programa lógico rebatible. Este procedimiento de prueba consiste en la construcción de argumentos para un literal, la construcción de contraargumentos para esos argumentos, y la determinación de las derrotas entre dichos argumentos. Finalmente, un análisis dialéctico permite determinar cuándo un literal de un programa lógico rebatible está garantizado.

2.3.3. Construcción de argumentos

Un argumento para un literal c construido a partir de un programa lógico rebatible $\mathcal{P} = (\Pi, \Delta)$ consiste de un subconjunto minimal y no contradictorio de reglas rebatibles de Δ y un subconjunto minimal de hechos de Π que permiten derivar c .

Definición 2.13 (Argumento) Sea $\mathcal{P} = (\Pi, \Delta)$ un programa lógico rebatible y c un literal, un *argumento* \mathcal{A} para c es una tripla $\mathcal{A} = \langle c, R, H \rangle$ tal que:

1. $R \subseteq \Delta$, $H \subseteq \Pi$, y
2. existe una derivación para c a partir de $R \cup H$, y
3. $\Pi \cup R$ es no contradictorio (*i. e.*, no deriva literales complementarios), y
4. R y H son minimales (*i. e.*, no existe $R' \subset R$ ni $H' \subset H$ que satisfaga 2 y 3). ■

Dado un argumento $\mathcal{A} = \langle c, R, H \rangle$, el literal c se denomina la *conclusión* de \mathcal{A} , la cual es denotada $\text{CONCLUSIÓN}(\mathcal{A})$.

Ejemplo 2.3 A partir del programa lógico rebatible del Ejemplo 2.1 pueden construirse los siguientes argumentos:

$$\mathcal{A}_1 = \left\langle \begin{array}{l} \text{comprar_acciones}(\text{umbrella}), \\ \text{comprar_acciones}(X) \prec \text{buen_precio}(X) \end{array} \right\rangle,$$

$$\begin{aligned}
& \left\{ \text{buen_precio}(\text{umbrella}) \right\} \rangle \\
\mathcal{A}_2 = & \left\langle \begin{array}{l} \sim\text{comprar_acciones}(\text{umbrella}), \\ \left\{ \begin{array}{l} \sim\text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riegosa}(X) \prec \text{en_fusión}(X) \end{array} \right\}, \\ \left\{ \text{en_fusión}(\text{umbrella}) \right\} \end{array} \right\rangle \\
\mathcal{A}_3 = & \left\langle \begin{array}{l} \text{riegosa}(\text{umbrella}), \\ \left\{ \text{riegosa}(X) \prec \text{en_fusión}(X) \right\}, \\ \left\{ \text{en_fusión}(\text{umbrella}) \right\} \end{array} \right\rangle \\
\mathcal{A}_4 = & \left\langle \text{buen_precio}(\text{umbrella}), \emptyset, \{\text{buen_precio}(\text{umbrella})\} \right\rangle \\
\mathcal{A}_5 = & \left\langle \text{en_fusión}(\text{umbrella}), \emptyset, \{\text{en_fusión}(\text{umbrella})\} \right\rangle \\
\mathcal{A}_6 = & \left\langle \sim\text{banca_rota}(\text{umbrella}), \emptyset, \{\sim\text{banca_rota}(\text{umbrella})\} \right\rangle \quad \blacksquare
\end{aligned}$$

Observe que es posible construir argumentos cuyo conjunto de reglas rebatibles es vacío. Estos argumentos se denominan *argumentos estrictos* y corresponden a hechos del programa a partir del cual se construyen. De esta manera, dado un programa $\mathcal{P} = (\Pi, \Delta)$, existe un argumento estricto $\langle h, \emptyset, \{h\} \rangle$ para cada hecho $h \in \Pi$ y no existe ningún argumento para \bar{h} (dado que Π es no contradictorio).

Teniendo en cuenta que un argumento constituye un conjunto de reglas rebatibles y un conjunto de hechos, es posible definir la noción de *subargumento*. Básicamente, un argumento no estricto \mathcal{S} es subargumento de otro argumento \mathcal{A} si el conjunto de reglas rebatibles de \mathcal{S} está contenido en las de \mathcal{A} , mientras que un argumento estricto \mathcal{S} es subargumento de \mathcal{A} si la conclusión de \mathcal{S} está contenida en el conjunto de hechos de \mathcal{A} . Por lo tanto, todo argumento es subargumento de sí mismo.

Definición 2.14 (Subargumento) Sea \mathcal{P} un programa lógico rebatible y $\mathcal{S} = \langle s, R_s, H_s \rangle$ y $\mathcal{A} = \langle a, R_a, H_a \rangle$ dos argumentos construidos a partir de \mathcal{P} , el argumento \mathcal{S} es un *subargumento* de \mathcal{A} si y sólo si $\emptyset \subset R_s \subseteq R_a$, o si $R_s = \emptyset$ y $s \in H_a$. ■

Ejemplo 2.4 Considere los argumentos del Ejemplo 2.3.

\mathcal{A}_3 es subargumento de \mathcal{A}_2 .

\mathcal{A}_4 es subargumento de \mathcal{A}_1 .

\mathcal{A}_5 es subargumento de \mathcal{A}_2 y de \mathcal{A}_3 . ■

2.3.4. Conflictos entre argumentos

En DeLP, un argumento \mathcal{A} *ataca* a otro argumento \mathcal{B} , *i. e.*, \mathcal{A} es un *contra-argumento* para \mathcal{B} , si \mathcal{A} sustenta una conclusión complementaria³ a la de algún subargumento de \mathcal{B} .

Definición 2.15 (Ataque / Contra-argumento) Sea \mathcal{P} un programa lógico rebatible y $\mathcal{A} = \langle a, R_a, H_a \rangle$ y \mathcal{B} dos argumentos contruidos a partir de \mathcal{P} , el argumento \mathcal{A} *ataca* a \mathcal{B} , *i. e.*, \mathcal{A} es un *contra-argumento* para \mathcal{B} , si y sólo si existe un subargumento $\mathcal{S} = \langle \bar{a}, R_{\bar{a}}, H_{\bar{a}} \rangle$ de \mathcal{B} . El argumento \mathcal{S} se denomina *subargumento en conflicto* con \mathcal{A} . ■

Como todo argumento es subargumento de sí mismo, siempre que un argumento \mathcal{A} ataca a otro argumento \mathcal{B} existe un subargumento \mathcal{S} de \mathcal{B} que ataca a \mathcal{A} . Además, note que los argumentos estrictos no pueden tener contra-argumentos. Esto se debe a que, dado $\langle h, \emptyset, \{h\} \rangle$, no es posible construir un argumento para \bar{h} ya que éste estaría en contradicción con la parte estricta Π del programa lógico rebatible.

Ejemplo 2.5 Considere los argumentos del Ejemplo 2.3. El argumento \mathcal{A}_1 ataca al argumento \mathcal{A}_2 , siendo \mathcal{A}_2 el subargumento en conflicto con \mathcal{A}_1 . Recíprocamente, el argumento \mathcal{A}_2 ataca al argumento \mathcal{A}_1 , siendo \mathcal{A}_1 el subargumento en conflicto con \mathcal{A}_2 . ■

La noción de ataque no es suficiente para determinar cuál argumento prevalece ante un conflicto. Para esto, es necesario un criterio de comparación de argumentos que establezca cuándo un argumento es preferido sobre otro.

2.3.5. Resolución de conflictos entre argumentos

Como se mencionó anteriormente, siempre que un argumento \mathcal{A} ataca a otro argumento \mathcal{B} existe un subargumento \mathcal{S} de \mathcal{B} que ataca a \mathcal{A} . Dada esta simetría, la noción de ataque no es suficiente para determinar cuándo un argumento derrota al otro. Por este motivo, es necesario un *criterio de comparación de argumentos* que establezca cuándo un argumento es preferido sobre otro, y así poder determinar cuál de ellos prevalece ante un conflicto.

DeLP utiliza un criterio de comparación de argumentos modular, lo cual permite utilizar el más adecuado para el dominio de aplicación. Por lo tanto, como la comparación

³En la versión original de DeLP los ataques ocurren a través de literales *en desacuerdo*, los cuales consideran las reglas estrictas del programa lógico rebatible.

entre argumentos puede definirse de diversas formas, en esta tesis se abstraerá de la especificación de este criterio suponiendo que se trata de una relación de orden parcial entre argumentos denotada ‘ $>$ ’. De esta manera, $\mathcal{A} > \mathcal{B}$ denota que \mathcal{A} es preferido sobre \mathcal{B} .

Si \mathcal{A} ataca a \mathcal{B} , es necesario analizar si es lo suficientemente fuerte como para derrotarlo. En consecuencia, es necesario efectuar una comparación entre el argumento atacante y el subargumento atacado. Concretamente, si \mathcal{A} ataca a \mathcal{B} , y \mathcal{S} es el subargumento de \mathcal{B} en conflicto con \mathcal{A} , se deben comparar \mathcal{A} y \mathcal{S} .

Definición 2.16 (Derrota) Sea \mathcal{P} un programa lógico rebatible, \mathcal{A} y \mathcal{B} dos argumentos contruidos a partir de \mathcal{P} , y $>$ una relación de orden parcial entre argumentos, \mathcal{A} *derrota* a \mathcal{B} si y sólo si \mathcal{A} ataca a \mathcal{B} , siendo \mathcal{S} el subargumento de \mathcal{B} en conflicto con \mathcal{A} , y se cumple que:

- $\mathcal{A} > \mathcal{S}$, *i. e.*, \mathcal{A} es un *derrotador propio* para \mathcal{B} , o
- $\mathcal{A} \not> \mathcal{S}$ y $\mathcal{S} \not> \mathcal{A}$, *i. e.*, \mathcal{A} es un *derrotador por bloqueo* para \mathcal{B} . ■

La definición anterior identifica dos situaciones en las que un ataque entre argumentos resulta exitoso. En una derrota propia el argumento atacante \mathcal{A} es preferido sobre el subargumento atacado \mathcal{S} . En contraste, en una derrota por bloqueo el argumento atacante \mathcal{A} y el subargumento atacado \mathcal{S} son incomparables según la relación de preferencia entre argumentos.

Ejemplo 2.6 Considere los argumentos del Ejemplo 2.3 y los ataques del Ejemplo 2.5, y supongamos que $\mathcal{A}_2 > \mathcal{A}_1$. El argumento \mathcal{A}_2 es un derrotador propio para \mathcal{A}_1 . ■

A continuación se presenta la representación gráfica que se utilizará en esta tesis para los argumentos y las derrotas. Cada argumento será representado mediante un triángulo cuyo vértice superior corresponde a su conclusión. Además, una flecha desde un argumento \mathcal{A} hacia otro argumento \mathcal{B} representa que \mathcal{A} derrota a \mathcal{B} . En particular, la Figura 2.1 ilustra la derrota entre los argumentos del Ejemplo 2.6.

2.3.6. Aceptabilidad

El objetivo de todo sistema argumentativo es identificar los argumentos aceptados, para lo cual es necesario considerar las derrotas entre los argumentos. En el caso de

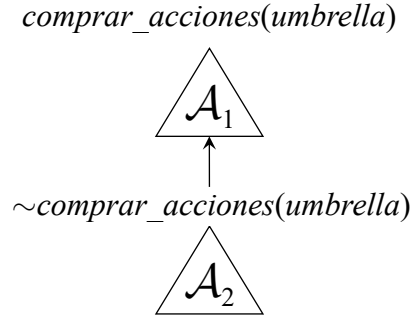


Figura 2.1: Representación gráfica de una derrota entre dos argumentos.

DeLP, los argumentos aceptados determinarán qué literales están *garantizados*, *i. e.*, son inferibles a partir de un programa lógico rebatible. Por lo tanto, para determinar si es posible inferir un literal c a partir de un programa lógico rebatible \mathcal{P} , es necesario analizar todos los argumentos a favor y en contra de c que pueden construirse a partir de \mathcal{P} .

Por ejemplo, si a partir de un programa lógico rebatible \mathcal{P} se construye un argumento $\mathcal{A}_1 = \langle c, R, H \rangle$ que no tiene derrotadores, entonces un agente que disponga de \mathcal{P} para razonar podrá concluir el literal c . Sin embargo, también podría ocurrir que a partir de \mathcal{P} se construya un argumento \mathcal{A}_2 que derrote a \mathcal{A}_1 , con lo cual existiría una razón para no concluir c . Sin embargo, también podría ocurrir que el argumento \mathcal{A}_2 tenga a su vez un derrotador \mathcal{A}_3 , el cual *defiende* a \mathcal{A}_1 y reinstaura la conclusión c del agente. Esta secuencia de derrotas podría continuar con otro derrotador \mathcal{A}_4 para \mathcal{A}_3 que invalida nuevamente la conclusión c , y así sucesivamente, dando origen a una secuencia de argumentos $[\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \dots]$ llamada *línea de argumentación*, donde cada elemento de la secuencia es un derrotador de su predecesor.

Definición 2.17 (Línea de argumentación) Sea \mathcal{P} un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, y \mathcal{A}_1 un argumento construido a partir de \mathcal{P} , una *línea de argumentación* para \mathcal{A}_1 es una secuencia de argumentos de \mathcal{P} , denotada $\Lambda = [\mathcal{A}_1, \dots, \mathcal{A}_n]$, donde cada argumento es un derrotador para su predecesor. ■

Dada una línea de argumentación $\Lambda = [\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \dots]$ para un argumento \mathcal{A}_1 , los argumentos en posiciones impares conforman la secuencia de *argumentos a favor*, denotada $\Lambda_F = [\mathcal{A}_1, \mathcal{A}_3, \dots]$, mientras que los argumentos en posiciones pares componen la secuencia de *argumentos en contra*, denotada $\Lambda_C = [\mathcal{A}_2, \mathcal{A}_4, \dots]$.

Para evitar secuencias de argumentos que representen cadenas de razonamiento falaz, en DeLP las líneas de argumentación deben ser *acceptables*. Resumidamente, una línea de argumentación es aceptable si es una secuencia finita, no contiene argumentos o subargumentos repetidos, no posee dos derrotas por bloqueo consecutivas, y tanto Λ_F como Λ_C no tienen pares de argumentos que se ataquen entre sí.

El requisito de que una línea de argumentación sea finita está directamente relacionado con el requisito de que no contenga argumentos o subargumentos repetidos ya que ambos evitan que ocurra un proceso argumentativo falaz y posiblemente infinito llamado *argumentación circular* [GS04]. El requisito de que no tenga dos derrotas por bloqueo consecutivas impide que un literal pertenezca a las inferencias de un programa simplemente por tener más argumentos a favor que argumentos en contra. Por último, el requisito de que tanto Λ_F como Λ_C no tengan pares de argumentos que se ataquen entre sí evita la situación falaz en la que un argumento es defendido por otro argumento que lo ataca.

Teniendo en cuenta todos estos requisitos y considerando que el operador \in^* corresponde a la pertenencia de un elemento en una secuencia, la siguiente definición formaliza el concepto de línea de argumentación aceptable.

Definición 2.18 (Línea de argumentación aceptable) Sea $\mathcal{P} = (\Pi, \Delta)$ un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, y $\Lambda = [\mathcal{A}_1, \dots, \mathcal{A}_n]$ una línea de argumentación para \mathcal{A}_1 obtenida a partir de \mathcal{P} y de $>$, Λ es *acceptable* si y sólo si satisface las siguientes condiciones:

- Λ es una secuencia finita,
- si $\mathcal{A}_i \in^* \Lambda$ y $\mathcal{A}_j \in^* \Lambda$ tal que $i < j$, entonces \mathcal{A}_j no es subargumento de \mathcal{A}_i ,
- para todo $\mathcal{A}_i \in^* \Lambda$ tal que \mathcal{A}_i es un derrotador por bloqueo para $\mathcal{A}_{i-1} \in^* \Lambda$, si $\mathcal{A}_{i+1} \in^* \Lambda$, entonces \mathcal{A}_{i+1} es un derrotador propio para \mathcal{A}_i , y
- $\Pi \cup \{\mathbf{R} \mid \langle c, \mathbf{R}, \mathbf{H} \rangle \in^* \Lambda_F\}$ y $\Pi \cup \{\mathbf{R} \mid \langle c, \mathbf{R}, \mathbf{H} \rangle \in^* \Lambda_C\}$ son no contradictorios. ■

Note que una línea de argumentación aceptable podría no ser exhaustiva. Es decir, podrían no considerarse argumentos que influyen en la evaluación del argumento en cuestión. Por lo tanto, al momento de determinar cuáles son las conclusiones garantizadas a partir de un programa lógico rebatible sólo se considerarán *líneas de argumentación aceptables exhaustivas*.

Definición 2.19 (Línea de argumentación aceptable exhaustiva) Sea \mathcal{P} un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, y $\Lambda = [\mathcal{A}_1, \dots, \mathcal{A}_n]$ una línea de argumentación aceptable para \mathcal{A}_1 obtenida a partir de \mathcal{P} y de $>$, Λ es *exhaustiva* si a partir de \mathcal{P} no puede construirse un argumento \mathcal{A}_{n+1} tal que $\Lambda' = [\mathcal{A}_1, \dots, \mathcal{A}_n, \mathcal{A}_{n+1}]$ sea una línea de argumentación aceptable para \mathcal{A}_1 . ■

Como se mencionó anteriormente, en un programa lógico rebatible los argumentos pueden tener múltiples derrotadores. Por lo tanto, dado un argumento \mathcal{A} puede obtenerse una ramificación de líneas de argumentación para \mathcal{A} , la cual da origen a un árbol de derrotadores denominado *árbol de dialéctica*. En un árbol de dialéctica \mathcal{T} , cada nodo \aleph está etiquetado con un argumento—denotado $\text{ETIQUETA}(\aleph) = \mathcal{A}$ o $\aleph_{\mathcal{A}}$ —y cada rama desde la raíz hasta una hoja es una línea de argumentación aceptable exhaustiva.

Definición 2.20 (Árbol de dialéctica) Sea \mathcal{P} un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, y \mathcal{A}_1 un argumento construido a partir de \mathcal{P} , un *árbol de dialéctica* $\mathcal{T} = \langle \text{Nodos}, \text{Arcos} \rangle$ para \mathcal{A}_1 es un árbol tal que cada nodo $\aleph \in \text{Nodos}$ está etiquetado con un argumento construido a partir de \mathcal{P} , cada arco $\sphericalangle \in \text{Arcos}$ desde un nodo $\aleph_{\mathcal{B}}$ hacia otro nodo $\aleph_{\mathcal{C}}$ representa que \mathcal{B} derrota a \mathcal{C} , y cumple las siguientes condiciones:

- si $\Lambda = [\mathcal{A}_1, \dots, \mathcal{A}_n]$ es una línea de argumentación aceptable exhaustiva para \mathcal{A}_1 en \mathcal{P} , entonces existe en \mathcal{T} una rama compuesta de una secuencia de nodos $[\aleph_1, \dots, \aleph_n]$ tal que $\text{ETIQUETA}(\aleph_i) = \mathcal{A}_i$ ($1 \leq i \leq n$), y
- si $\aleph_{\mathcal{D}} \in \text{HIJOS}(\aleph_{\mathcal{A}})$, entonces no existe $\aleph'_{\mathcal{D}} \neq \aleph_{\mathcal{D}}$ tal que $\aleph'_{\mathcal{D}} \in \text{HIJOS}(\aleph)$. ■

Ejemplo 2.7 Considere el programa lógico rebatible $\mathcal{P} = (\Pi, \Delta)$:

$$\Pi = \left\{ \begin{array}{cc} c & g \\ k & e \\ j & i \end{array} \right\} \quad \Delta = \left\{ \begin{array}{cccc} a \prec b & b \prec c & \sim b \prec c, d & d \prec g \\ \sim d \prec k & \sim b \prec e & \sim b \prec c, f & f \prec g \\ \sim f \prec g, \sim h & \sim h \prec j & \sim f \prec i & h \prec k \end{array} \right\}$$

A partir de \mathcal{P} pueden construirse los argumentos:

$$\mathcal{A} = \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle$$

$$\mathcal{B}_1 = \langle b, \{b \prec c\}, \{c\} \rangle, \text{ el cual es subargumento de } \mathcal{A} \text{ y ataca a } \mathcal{B}_2, \mathcal{B}_3, \text{ y } \mathcal{B}_4.$$

$$\mathcal{B}_2 = \langle \sim b, \{\sim b \prec c, d; d \prec g\}, \{c, g\} \rangle, \text{ el cual ataca a } \mathcal{B}_1 \text{ y } \mathcal{A}.$$

- $\mathcal{D}_1 = \langle d, \{d \prec g\}, \{g\} \rangle$, el cual es subargumento de \mathcal{B}_2 y ataca a \mathcal{D}_2 .
 $\mathcal{D}_2 = \langle \sim d, \{\sim d \prec k\}, \{k\} \rangle$, el cual ataca a \mathcal{D}_1 y \mathcal{B}_2 .
 $\mathcal{B}_3 = \langle \sim b, \{\sim b \prec e\}, \{e\} \rangle$, el cual ataca a \mathcal{B}_1 y \mathcal{A} .
 $\mathcal{B}_4 = \langle \sim b, \{\sim b \prec c, f; f \prec g\}, \{c, g\} \rangle$, el cual ataca a \mathcal{B}_1 y \mathcal{A} .
 $\mathcal{F}_1 = \langle f, \{f \prec g\}, \{g\} \rangle$, el cual es subargumento de \mathcal{B}_4 y ataca a \mathcal{F}_2 y \mathcal{F}_3 .
 $\mathcal{F}_2 = \langle \sim f, \{\sim f \prec g, \sim h; \sim h \prec j\}, \{g, j\} \rangle$, el cual ataca a \mathcal{F}_1 y \mathcal{B}_4 .
 $\mathcal{H}_1 = \langle \sim h, \{\sim h \prec j\}, \{j\} \rangle$, el cual es subargumento de \mathcal{F}_2 y ataca a \mathcal{H}_2 .
 $\mathcal{F}_3 = \langle \sim f, \{\sim f \prec i\}, \{i\} \rangle$, el cual ataca a \mathcal{F}_1 y \mathcal{B}_4 .
 $\mathcal{H}_2 = \langle h, \{h \prec k\}, \{k\} \rangle$, el cual ataca a \mathcal{H}_1 y \mathcal{F}_2 .

Supongamos que $\mathcal{B}_2 > \mathcal{B}_1$, que $\mathcal{B}_4 > \mathcal{B}_1$ y que $\mathcal{F}_2 > \mathcal{F}_1$. Al momento de generar el árbol de dialéctica para \mathcal{A} , son relevantes las siguientes derrotas:

- \mathcal{B}_2 y \mathcal{B}_4 son derrotadores propios para \mathcal{A} .
 \mathcal{B}_3 es un derrotador por bloqueo para \mathcal{A} .
 \mathcal{D}_2 es un derrotador por bloqueo para \mathcal{B}_2 .
 \mathcal{F}_2 es un derrotador propio para \mathcal{B}_4 .
 \mathcal{F}_3 es un derrotador por bloqueo para \mathcal{B}_4 .
 \mathcal{H}_2 es un derrotador por bloqueo para \mathcal{F}_2 .

Por lo tanto, existen cuatro líneas de argumentación aceptables exhaustivas para \mathcal{A} :

- $\Lambda_1 = [\mathcal{A}, \mathcal{B}_2, \mathcal{D}_2]$
 $\Lambda_2 = [\mathcal{A}, \mathcal{B}_3]$
 $\Lambda_3 = [\mathcal{A}, \mathcal{B}_4, \mathcal{F}_2, \mathcal{H}_2]$
 $\Lambda_4 = [\mathcal{A}, \mathcal{B}_4, \mathcal{F}_3]$

Teniendo en cuenta estas líneas de argumentación, se puede construir el árbol de dialéctica para \mathcal{A} ilustrado en la Figura 2.2. ■

Teniendo en cuenta la Definición 2.20, una línea de argumentación Λ de un árbol de dialéctica también puede verse como la secuencia de nodos $\Lambda = [\aleph_1, \dots, \aleph_n]$ cuyas etiquetas son los argumentos que la componen. Considerando esto, una subsecuencia inicial de nodos de una línea de argumentación se define a través de su *segmento superior*.

Definición 2.21 (Segmento superior) Dada una línea de argumentación

$\Lambda = [\aleph_1, \dots, \aleph_n]$ y un nodo $\aleph_i \in^* \Lambda$, el *segmento superior* de Λ hasta \aleph_i se define como $\text{SEGMENTOSUPERIOR}(\Lambda, \aleph_i) = [\aleph_1, \dots, \aleph_i]$. ■

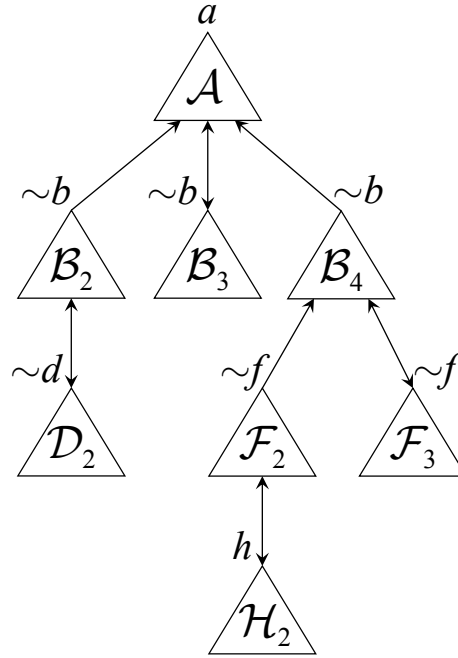


Figura 2.2: Árbol de dialéctica.

Como puede observarse en el Ejemplo 2.7, las etiquetas de los nodos hoja del árbol de dialéctica corresponden a argumentos no derrotados. Por otro lado, las etiquetas de los nodos internos que tienen al menos un hijo etiquetado con un argumento no derrotado corresponden a argumentos derrotados. En contraste, las etiquetas de los nodos internos cuyos hijos están todos etiquetados con un argumento derrotado corresponden a argumentos no derrotados. Siguiendo con este análisis desde las hojas hacia la raíz, los nodos de un árbol de dialéctica pueden marcarse como *derrotados* o *no derrotados*, respectivamente denotados $\textcircled{\text{D}}$ y $\textcircled{\text{U}}$ (por sus nombres en inglés, *defeated* y *undefeated*).

Definición 2.22 (Marca de un nodo) Sea \mathcal{P} un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, y $\mathcal{T} = \langle \text{Nodos}, \text{Arcos} \rangle$ un árbol de dialéctica para un argumento \mathcal{A} construido a partir de \mathcal{P} y de $>$, la *marca* de un nodo $\aleph \in \text{Nodos}$ se define como:

$$\text{MARCA}(\aleph, \mathcal{T}) = \begin{cases} \textcircled{\text{U}}, & \text{si para todo } \aleph' \in \text{HIJOS}(\aleph) \text{ se cumple que } \text{MARCA}(\aleph', \mathcal{T}) = \textcircled{\text{D}}. \\ \textcircled{\text{D}}, & \text{si } \aleph' \in \text{HIJOS}(\aleph) \text{ tal que } \text{MARCA}(\aleph', \mathcal{T}) = \textcircled{\text{U}}. \end{cases} \quad \blacksquare$$

La Figura 2.3 ilustra el árbol de dialéctica de la Figura 2.2 con sus nodos marcados. Observe que el nodo etiquetado con \mathcal{B}_3 , el cual está marcado como $\textcircled{\text{U}}$, es la causa por la

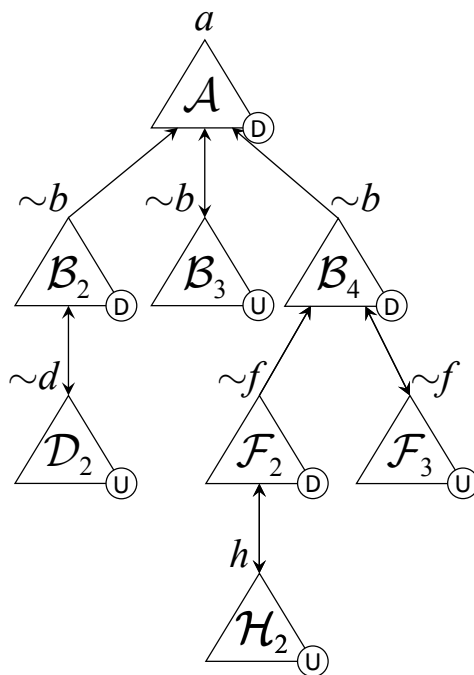


Figura 2.3: Árbol de dialéctica marcado.

cual el nodo raíz está marcado como \textcircled{D} . De la misma manera, el nodo etiquetado con \mathcal{F}_3 es la causa por la cual el nodo etiquetado con \mathcal{B}_4 está marcado como \textcircled{U} . Un argumento que impide que un nodo esté marcado como \textcircled{U} se denomina *negador*.

Definición 2.23 (Negador) Sea $\mathcal{T} = \langle \text{Nodos}, \text{Arcos} \rangle$ un árbol de dialéctica y $\aleph_{\mathcal{A}} \in \text{Nodos}$ y $\aleph_{\mathcal{N}} \in \text{HIJOS}(\aleph_{\mathcal{A}})$ dos nodos tales que $\text{MARCA}(\aleph_{\mathcal{A}}) = \textcircled{D}$, el argumento \mathcal{N} es un *negador* para el argumento \mathcal{A} si y sólo si $\text{MARCA}(\aleph_{\mathcal{N}}) = \textcircled{U}$. ■

Ejemplo 2.8 Considere el árbol de dialéctica marcado de la Figura 2.3.

\mathcal{D}_2 es un negador para \mathcal{B}_2 .

\mathcal{B}_3 es un negador para \mathcal{A} .

\mathcal{H}_2 es un negador para \mathcal{F}_2 .

\mathcal{F}_3 es un negador para \mathcal{B}_4 . ■

Un árbol de dialéctica para un argumento \mathcal{A} representa el análisis dialéctico en el cual todos los argumentos relevantes de un programa lógico rebatible son considerados para determinar el estado de aceptabilidad de \mathcal{A} . Considerando esto, un argumento \mathcal{A} está *aceptado* si la raíz del árbol de dialéctica \mathcal{T} —denotada $\text{RAÍZ}(\mathcal{T})$ —para \mathcal{A} está marcada como \textcircled{U} .

Definición 2.24 (Argumento aceptado) Sea \mathcal{P} un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, \mathcal{A} un argumento construido a partir de \mathcal{P} , y \mathcal{T} el árbol de dialéctica para \mathcal{A} construido a partir de \mathcal{P} y de $>$, el argumento \mathcal{A} está *aceptado* si y sólo si $\text{MARCA}(\text{RAÍZ}(\mathcal{T})) = \textcircled{U}$. ■

Ejemplo 2.9 Considere los árboles de dialéctica marcados de la Figura 2.4. Los argumentos \mathcal{A}_1 y \mathcal{A}_2 están aceptados. ■

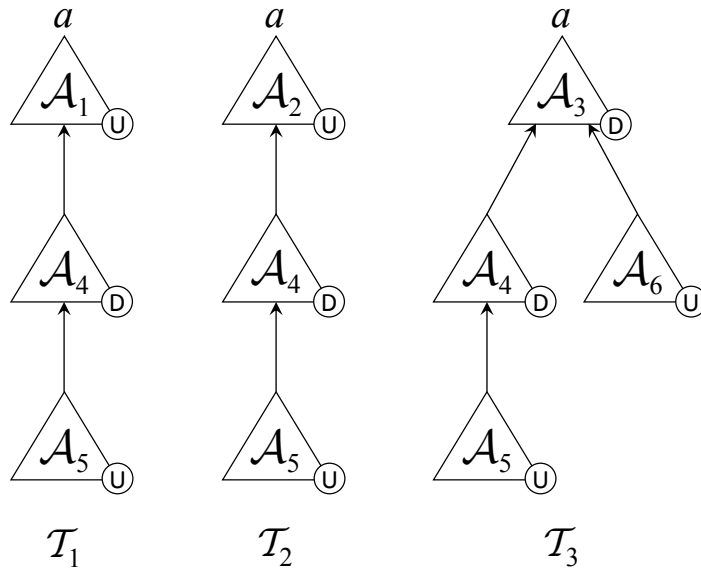


Figura 2.4: Árboles de dialéctica marcados.

Claramente, si hay un argumento aceptado, esto implica que hay una garantía para inferir su conclusión. Siguiendo este razonamiento, las conclusiones de los argumentos aceptados que pueden ser construidos a partir de un programa lógico rebatible son *literales garantizados*.

Definición 2.25 (Literal garantizado) Sea \mathcal{P} un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, y l un literal, l está *garantizado* si y sólo si existe un argumento aceptado $\mathcal{A} = \langle l, R, H \rangle$ construido a partir de \mathcal{P} y de $>$. ■

Ejemplo 2.10 Considere los árboles de dialéctica marcados de la Figura 2.4. El literal a está garantizado. ■

Dado que en un mismo programa lógico rebatible $\mathcal{P} = (\Pi, \Delta)$ pueden existir múltiples argumentos para un mismo literal l , éste podría estar garantizado por más de un argumento aceptado. Por lo tanto, el conjunto de árboles de dialéctica enraizados con un argumento aceptado que concluye l , denominados *árboles de dialéctica garantes*, representan todos los procesos de razonamiento diferentes que garantizan una conclusión.

Definición 2.26 (Árboles de dialéctica garantes) Sea \mathcal{P} un programa lógico rebatible, $>$ una relación de orden parcial entre argumentos, y l un literal, el conjunto de *árboles de dialéctica garantes* de l se define como $\text{ADG}(l, \Pi, \Delta, >) = \{\mathcal{T} \mid \mathcal{T} \text{ es construido a partir de } \mathcal{P} \text{ y de } > \text{ tal que } \text{RAÍZ}(\mathcal{T}) = \mathcal{A} = \langle l, \text{R}, \text{H} \rangle \text{ y } \mathcal{A} \text{ está aceptado}\}$. ■

Ejemplo 2.11 Considere los árboles de dialéctica marcados de la Figura 2.4. El conjunto de árboles de dialéctica garantes para el literal a es $\text{ADG}(a, \Pi, \Delta, >) = \{\mathcal{T}_1, \mathcal{T}_2\}$. ■

2.4. Agentes en DeLP

En esta sección se formalizará el concepto de agente que será utilizado en esta tesis. En particular, los agentes utilizarán DeLP como formalismo de representación de conocimiento y razonamiento.

Definición 2.27 Un *agente* es una 4-tupla $Z = (ID, \Pi, \Delta, >)$ donde ID es un identificador, (Π, Δ) es un programa lógico rebatible, y $>$ es una relación de orden parcial entre argumentos. ■

El identificador ID le permite al agente ser identificado unívocamente. En cambio, Π , Δ y $>$ representan la *base de creencias* del agente, *i. e.*, las creencias que tiene sobre el entorno en un determinado momento. En otras palabras, un agente *Cree* en todo hecho $h \in \Pi$, *Cree* en toda regla rebatible $r \in \Delta$ y *Cree* que \mathcal{A} es preferido sobre \mathcal{B} para todo $(\mathcal{A}, \mathcal{B}) \in >$.

Dado un agente $Z = (ID, \Pi, \Delta, >)$, teniendo en cuenta las Definiciones 2.13, 2.20, 2.25 y 2.26 y considerando que (Π, Δ) es un programa lógico rebatible, se dirá que...

- Z puede construir un argumento \mathcal{A} si y sólo si \mathcal{A} puede ser construido a partir de (Π, Δ) .

- Z cree que un argumento \mathcal{D} derrota a otro argumento \mathcal{A} si y sólo si \mathcal{D} derrota a \mathcal{A} a partir de $>$.
- Z puede construir un árbol de dialéctica \mathcal{T} si y sólo si \mathcal{T} puede ser construido a partir de (Π, Δ) y de $>$.
- Z concluye un literal l si y sólo si l está garantizado a partir de (Π, Δ) y de $>$.
- Z acepta un argumento \mathcal{A} si y sólo si \mathcal{A} es un argumento aceptable a partir de (Π, Δ) y de $>$.
- el conjunto de árboles de dialéctica de Z que garantizan un literal l es $\text{ADG}(l, \Pi, \Delta, >)$.

Como se verá en los Capítulos 3 y 4, los agentes pueden modificar sus bases de creencias mediante la interacción con otros agentes durante un diálogo. Por ejemplo, un agente podría adquirir nuevos hechos y reglas rebatibles que fueron compartidos por otros y potencialmente construir nuevos argumentos que antes no podía. También podría dejar de creer en hechos que fueron considerados inválidos mediante un consenso y potencialmente dejar de construir argumentos que antes podía. Incluso podría modificar sus preferencias entre argumentos y cambiar cómo sus argumentos se derrotan entre sí. Considerando esto, los árboles de dialéctica que pueden construir los agentes—y por ende, sus conclusiones—podrían cambiar dinámicamente durante un diálogo.

2.5. Sistemas de transiciones

En esta sección se introducirá el concepto de *sistema de transiciones*, el cual será utilizado para definir formalmente los protocolos y estrategias de diálogo de esta tesis y luego probar sus propiedades.

Un sistema de transiciones [Pl081] puede utilizarse para describir el comportamiento de un sistema discreto. Éste consiste de un conjunto de *reglas de transición* que determinan cómo el sistema pasa de un estado a otro. Una regla de transición está compuesta por un estado de origen, un estado de destino, y una condición bajo la cual la regla es aplicable. Ésta se denota

$$t : \frac{\text{condición}}{\text{estado_origen} \rightarrow \text{estado_destino}}$$

y especifica que si el sistema está en el *estado_origen* y se cumple la *condición* entonces el sistema evoluciona al *estado_destino*.

En esta tesis se utilizará un sistema de transiciones para definir el protocolo y estrategia de diálogo de consulta a experto del Capítulo 3 y otro para definir el protocolo y estrategia de diálogo de indagación del Capítulo 4. Muchas de las propiedades formales de estos sistemas de transiciones serán definidas en términos de *estados alcanzables*. Un estado e_a es alcanzable desde otro estado e_0 si existe una secuencia finita de reglas de transición que, aplicadas en ese orden, hacen que e_0 evolucione a e_a .

Definición 2.28 (Estado alcanzable) Dado un sistema de transiciones Θ y dos estados e_0 y e_a , se dice que e_a es *alcanzable* desde e_0 mediante Θ si y sólo si existe una secuencia finita de estados e_0, \dots, e_a tal que $e_i \xrightarrow{rt} e_{i+1}$ con $0 \leq i < a$ y $rt \in \Theta$. ■

En cada protocolo y estrategia de diálogo se distinguirán dos estados especiales: el *estado inicial* y los *estados finales*. Cuando se hable de estado alcanzable y no se mencione cuál es el estado e_0 implícitamente se hará referencia al estado inicial correspondiente a dicho protocolo y estrategia.

Por último, cuando existe una secuencia finita de reglas de transición que aplicadas en ese orden hacen que un estado inicial e_0 evolucione a un estado final e_f , la secuencia de estados comprendidos entre e_0 y e_f se denomina *ejecución del sistema de transiciones*.

Definición 2.29 (Ejecución de un sistema de transiciones) Dado un sistema de transiciones Θ , un estado inicial e_0 , y un estado final e_f alcanzable desde e_0 , la *ejecución* de Θ desde e_0 es una secuencia finita de estados e_0, \dots, e_f tal que $e_i \xrightarrow{rt} e_{i+1}$ ($0 \leq i < f$) con $rt \in \Theta$. ■

Capítulo 3

Protocolo y estrategia de diálogo de consulta a experto

En [WK95] se define *diálogo* como un framework normativo que comprende el intercambio de argumentos entre agentes, los cuales pueden tener objetivos individuales pero razonan juntos para lograr una meta colectiva. Uno de los tipos de diálogo colaborativos de la clasificación propuesta en dicho trabajo es el diálogo de *consulta a experto* (en inglés, *expert consultation*). En este tipo de diálogo, uno de los agentes es un experto en cierto dominio de conocimiento, y el otro le consulta sobre un tema de ese dominio, el cual desconoce o conoce parcialmente. En particular, el objetivo es que el consultante obtenga la opinión calificada del experto sobre el tema en cuestión. Según [WK95], un diálogo de consulta a experto es exitoso si, una vez finalizada la interacción, el consultante está de acuerdo con la opinión del experto.

En este capítulo se presentará un protocolo y estrategia de diálogo de consulta a experto en el cual los agentes utilizan como formalismo de representación de conocimiento y razonamiento el lenguaje DeLP presentado en el Capítulo 2. El diálogo comienza con un *agente consultante* que realiza una consulta. En respuesta, un *agente experto* en dicho dominio de conocimiento envía un argumento que justifica su opinión.

Dado que el consultante considera al otro agente un experto en el tema, su meta es aceptar la justificación que recibió en respuesta a su consulta. Sin embargo, como el consultante puede tener creencias previas, podría construir argumentos que estén en conflicto con la opinión del experto. Por lo tanto, si el consultante está comprometido

a lograr su meta de aceptar la justificación, tal vez necesite actualizar sus creencias y realizar preguntas al experto para adquirir más información.

El protocolo y estrategia de diálogo de consulta a experto se definirá formalmente utilizando un sistema de transiciones. Éste mostrará cómo el diálogo debe evolucionar desde el estado inicial en el que el consultante todavía no consultó al experto hasta un estado final en el que el consultante logró aceptar la justificación. Al final de este capítulo se probará formalmente que todo diálogo desarrollado mediante este protocolo y estrategia termina exitosamente.

3.1. Introducción y motivación

El protocolo y estrategia de diálogo de consulta a experto que se propone en este capítulo incluye dos mejoras respecto a los trabajos similares presentes en la literatura. Primero, se considera que el agente consultante podría tener información estricta en contradicción con las creencias del agente experto sobre el tema consultado. Segundo, se considera que ambos agentes podrían tener diferentes preferencias entre argumentos. Por lo tanto, si el consultante está comprometido a lograr su meta de aceptar la justificación, tal vez necesite actualizar sus preferencias y eliminar creencias de su base que—desde la perspectiva del experto—son inválidas. En la Sección 3.6 se realizará un análisis comparativo entre esta propuesta y los trabajos similares presentes en la literatura.

Antes de explicar el protocolo y estrategia de diálogo de consulta a experto, es importante comentar sobre algunas soluciones *naïve* al problema de resolver una consulta que no son aplicables o que implican consecuencias no deseables. Por ejemplo, una solución posible sería que el experto le envíe toda su base de creencias al consultante. Aunque parezca simple, esta solución es extrema, insensata e infactible por varias razones. En primer lugar, dependiendo del dominio de aplicación, el experto podría tener información privada que es sensible o confidencial y no debería ser compartida. En segundo lugar, la base de creencias del experto podría ser muy grande y unirla con la del consultante podría ser computacionalmente impracticable. En tercer lugar, la completa unión de las bases de creencias de ambos agentes probablemente causaría muchas contradicciones completamente fuera del dominio de aplicación de la consulta realizada. Ignorar estas contradicciones daría lugar a resultados y conclusiones indeseadas, pero resolverlas sería una pérdida innecesaria de tiempo de computo.

Por lo tanto, parece más adecuado que el experto envíe solamente la información necesaria para responder la consulta. Aún en este caso, también debe tenerse en cuenta cómo proceder ante las posibles contradicciones que surjan con la información previa que tenga el consultante. Considerando esto, una solución posible sería que el consultante elimine toda información previa que tenga sobre el tema—o al menos toda la información previa que esté en conflicto con la recibida—y así lograr aceptar la justificación del experto en un único paso. Sin embargo, como se mostrará en el ejemplo a continuación, esto podría implicar la eliminación innecesaria de piezas de información que, desde la perspectiva del experto, son válidas.

Ejemplo 3.1 Considere un agente que es experto en el mercado bursátil, y otro agente que es inversionista y consulta al experto si es conveniente comprar acciones de la empresa *Umbrella Corporation*. Las creencias del agente experto $\mathbf{E} = (\mathbf{exp}, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$ son:

$$\Pi_{\mathbf{E}} = \left\{ \begin{array}{ll} \text{nuevo_producto}(\text{umbrella}) & \text{en_fusión}(\text{umbrella}, \text{aperture}) \\ \text{fuerte}(\text{aperture}) & \text{problemas_legales}(\text{pollos_hermanos}) \\ \text{nuevo_dueño}(\text{umbrella}) & \end{array} \right\}$$

$$\Delta_{\mathbf{E}} = \left\{ \begin{array}{l} \text{comprar_acciones}(X) \prec \text{valor_aumentará}(X) \\ \text{valor_aumentará}(X) \prec \text{nuevo_producto}(X) \\ \sim \text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riesgosa}(X) \prec \text{en_fusión}(X, Y) \\ \sim \text{riesgosa}(X) \prec \text{en_fusión}(X, Y), \text{fuerte}(Y) \\ \text{vender_acciones}(X) \prec \text{problemas_legales}(X) \end{array} \right\}$$

Por otro lado, las creencias iniciales del agente consultante $\mathbf{I} = (\mathbf{inv}, \Pi_{\mathbf{I}}, \Delta_{\mathbf{I}}, >_{\mathbf{I}})$ son:

$$\Pi_{\mathbf{I}} = \left\{ \text{en_fusión}(\text{umbrella}, \text{aperture}) \right\}$$

$$\Delta_{\mathbf{I}} = \left\{ \begin{array}{l} \sim \text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riesgosa}(X) \prec \text{en_fusión}(X, Y) \end{array} \right\}$$

El experto está a favor de comprar acciones de Umbrella Corporation y justifica su respuesta con el siguiente argumento: “*Umbrella Corporation recientemente anunció un nuevo producto, lo cual da una razón para creer que el valor de sus acciones aumentará; considerando esto, hay una razón para comprar acciones de esta empresa*”.

$$\mathcal{M}_{\mathbf{I}} = \left\langle \text{comprar_acciones}(\text{umbrella}), \right.$$

$$\left\{ \begin{array}{l} \left\{ \begin{array}{l} \text{comprar_acciones}(X) \prec \text{valor_aumentará}(X) \\ \text{valor_aumentará}(X) \prec \text{nuevo_producto}(X) \end{array} \right\}, \\ \left\{ \text{nuevo_producto}(\text{umbrella}) \right\} \end{array} \right\}$$

Como el inversionista consultante confía en la experticia del otro agente, integra la información de este argumento con sus creencias previas para poder inferir la misma conclusión. En el caso de que el inversionista no tuviera información previa sobre el tema consultado (o al menos no tenga información en conflicto) lograría aceptar la justificación del experto inmediatamente. Sin embargo, el consultante tiene el siguiente contra-argumento: “*Umbrella Corporation está en fusión con Aperture Science, y generalmente una empresa que está en fusión con otra es riesgosa; no debería comprar acciones de una empresa riesgosa*”.

$$\mathcal{M}_2 = \left\{ \begin{array}{l} \sim \text{comprar_acciones}(\text{umbrella}), \\ \left\{ \begin{array}{l} \sim \text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riesgosa}(X) \prec \text{en_fusión}(X, Y) \end{array} \right\}, \\ \left\{ \text{en_fusión}(\text{umbrella}, \text{aperture}) \right\} \end{array} \right\}$$

Claramente, este argumento está en conflicto con la conclusión de la justificación del experto. Una solución naïve para resolver este conflicto y lograr aceptar la opinión calificada del experto sería que el consultante elimine de sus creencias todas las piezas de información conflictivas sin siquiera verificar su veracidad. En cambio, el consultante podría seguir una estrategia diferente: continuar la interacción con el experto dándole su argumento en conflicto para que éste le devuelva su opinión. Ahora supongamos que el experto considera que el argumento del consultante es válido, pero le responde con el siguiente contra-argumento: “*Aperture Science es una empresa fuerte, y las empresas en fusión con empresas fuertes también son fuertes; esto da una razón para creer que Umbrella Corporation no es riesgosa*”.

$$\mathcal{M}_3 = \left\{ \begin{array}{l} \sim \text{riesgosa}(\text{umbrella}), \\ \left\{ \begin{array}{l} \sim \text{riesgosa}(X) \prec \text{en_fusión}(X, Y), \text{fuerte}(Y) \\ \text{en_fusión}(\text{umbrella}, \text{aperture}) \\ \text{fuerte}(\text{aperture}) \end{array} \right\} \end{array} \right\}$$

El agente consultante integra la información de este argumento con sus creencias previas y, como logra aceptar la justificación, el diálogo de consulta a experto finaliza de

manera exitosa. El consultante adquirió información valiosa del experto sobre el tema sin pérdidas innecesarias de información. ■

Note que, en el ejemplo anterior, el agente experto tenía otras creencias que no fueron enviadas al agente consultante. En el protocolo y estrategia de diálogo que se definirá en la siguiente sección, el experto enviará al consultante sólo información relevante y necesaria hasta que logre aceptar la justificación que recibe en respuesta a su consulta. Como se verá durante el desarrollo de la propuesta, dado que el experto no puede predecir las creencias del consultante—y por ende, tampoco qué información necesita enviarle de ante mano—se entablará un diálogo que iterará hasta finalizar de manera exitosa. En la Sección 3.2.5 se mostrará el desarrollo completo del ejemplo anterior utilizando el protocolo propuesto.

Como se explicó anteriormente, según [WK95] un diálogo de consulta a experto es exitoso si el consultante logra aceptar la respuesta del experto a su consulta. Considerando esto, se probará que todo diálogo desarrollado mediante el protocolo y estrategia definido en este capítulo es exitoso en el contexto del formalismo de representación de conocimiento y razonamiento utilizado por los agentes: el agente consultante logrará aceptar la justificación que recibe del agente experto en respuesta a su consulta. El resto del capítulo está organizado de la siguiente manera:

- En la Sección 3.2 se definirá formalmente el protocolo y estrategia de diálogo de consulta a experto utilizando un sistema de transiciones.
- En la Sección 3.3 se demostrará que todo diálogo de consulta a experto desarrollado mediante este protocolo y estrategia finaliza de manera exitosa.
- En la Sección 3.4 se definirán operadores que permiten al experto minimizar o reducir la cantidad de interacciones necesarias para que el consultante cumpla su meta.
- En la Sección 3.5 se extenderá el sistema de transiciones para permitir al consultante rechazar la opinión del experto.
- En la Sección 3.6 se presentará el trabajo relacionado y se lo comparará con la propuesta de este capítulo.
- En la Sección 3.7 se presentarán las conclusiones.

Como se mencionó anteriormente, la propuesta y los resultados formales que se presentarán en este capítulo fueron publicados en la revista *Argument & Computation* vol. 10 [AGG19b].

3.2. Sesión de consulta a experto

En el protocolo y estrategia de diálogo de consulta a experto que se presentará en esta sección, los agentes serán representados mediante la 4-tupla $(ID, \Pi, \Delta, >)$ definida en el Capítulo 2. La interacción entre un agente consultante $C = (ID, \Pi_C, \Delta_C, >_C)$ y un agente experto $E = (ID, \Pi_E, \Delta_E, >_E)$ será llamada *sesión de consulta a experto*. Una sesión comienza con una consulta representada por un literal q y finaliza cuando el agente consultante logra aceptar la justificación que recibió del agente experto en respuesta a q .

Siempre que el agente C consulta al agente E por un literal q , ocurre uno y sólo uno de los siguientes tres escenarios¹:

1. E concluye q (*i. e.*, q está garantizado a partir de (Π_E, Δ_E) y $>_E$). En este escenario el experto puede construir al menos un árbol de dialéctica cuya raíz está marcada como \textcircled{U} y tiene q como conclusión.
2. E concluye \bar{q} (*i. e.*, \bar{q} está garantizado a partir de (Π_E, Δ_E) y $>_E$). En este escenario el experto no puede garantizar q , pero puede construir al menos un árbol de dialéctica cuya raíz está marcada como \textcircled{U} y tiene \bar{q} como conclusión.
3. E no concluye q ni \bar{q} (*i. e.*, ni q ni \bar{q} están garantizados a partir de (Π_E, Δ_E) y $>_E$). En este escenario o bien el experto no puede construir argumentos para q o \bar{q} , o bien puede construirlos pero todos terminan derrotados.

Dada una consulta q , una sesión comenzará y progresará sólo en los escenarios 1 y 2, *i. e.*, sólo si el agente experto tiene suficientes creencias para garantizar q o \bar{q} . Cuando esto ocurre, el experto enviará como respuesta un argumento para q o un argumento para \bar{q} , según corresponda. Este argumento se llamará *justificación* y será denotado \mathcal{J} . Al final de esta sección se probará que, una vez finalizada la sesión, el agente consultante habrá logrado su meta de aceptar \mathcal{J} . En otras palabras, habrá logrado marcar \mathcal{J} como \textcircled{U} en su árbol de dialéctica enraizado con \mathcal{J} . Si ocurre el tercer escenario, entonces el experto no puede darle una respuesta garantizada al consultante y no se realizará ninguna sesión.

Antes de comenzar a explicar cómo se desarrolla una sesión, se deben tener en cuenta las siguientes suposiciones:

1. Durante una sesión las creencias del agente consultante sólo cambian a través de la interacción con el agente experto.

¹En DeLP está probado que estos escenarios no pueden ocurrir simultáneamente.

2. Durante una sesión las creencias del agente experto no cambian.
3. Si el agente consultante tiene información personal que representa un contexto para su consulta y que podría afectar la respuesta del agente experto, ésta es enviada antes de que la sesión comience y temporalmente formará parte del programa lógico rebatible el experto utiliza para responder la consulta.

Aunque las suposiciones anteriores en principio parecen un poco fuertes, es importante tener en cuenta que son inherentes al modelo de interacción planteado. La suposición 1 implica que el consultante no puede estar en más de una sesión simultáneamente. Sin embargo, no impide que el experto esté en múltiples sesiones con diferentes consultantes al mismo tiempo. Esta suposición no es una grave restricción ya que si un agente debe consultar a varios expertos simplemente tiene que realizar sus consultas secuencialmente, en algún orden. La suposición 2 tampoco representa una gran restricción ya que, en general, es lo que ocurre al consultar a un experto (*e. g.*, consulta a un médico). La suposición 3 considera una situación fuera del foco de esta tesis ya que puede resolverse utilizando el concepto de *consulta contextual* definido en [GS14]. De esta forma, la información contextual del consultante podría ser temporalmente considerada por el experto sin la necesidad de cambiar su base de creencias permanentemente.

Una vez que una sesión ha comenzado, su estado actual es determinado por un *estado de sesión*. Intuitivamente, un estado de sesión es una estructura que almacena los elementos que cambian durante la sesión (*e. g.*, las creencias del agente consultante) o que determinan el flujo de ejecución (*e. g.*, los argumentos enviados por el experto y las preguntas pendientes del consultante). Por lo tanto, dado que $\mathbf{E} = (ID, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$ no cambia durante la sesión, el agente experto no formará parte de los estados.

Definición 3.1 (Estado de sesión de consulta a experto) Un *estado de sesión de consulta a experto* es una 4-tupla $(\mathbf{C}, \mathbb{F}, \mathbb{P}, \mathbb{N})$ donde $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$ es un agente consultante, \mathbb{F} es una secuencia de argumentos, \mathbb{P} es un conjunto de preguntas de preferencia, y \mathbb{N} es un conjunto de preguntas de negador. ■

El primer componente de la 4-tupla es el agente $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$ que hizo una consulta representada por un literal q . Aunque la dinámica de los otros tres componentes serán explicados en detalle durante esta sección, serán presentados intuitivamente a continuación: $\mathbb{F} = [\mathcal{F}_1, \dots, \mathcal{F}_n]$ es una secuencia (potencialmente vacía) de argumentos a favor

de la conclusión de la justificación (ya sea q o \bar{q}) enviados por el experto (*argumentos a favor*, para abreviar), de los cuales \mathcal{F}_1 es la justificación \mathcal{J} . \mathbb{P} es un conjunto de *preguntas de preferencia* pendientes, las cuales son pares de argumentos $(\mathcal{A}, \mathcal{B})$ que representan la pregunta “¿Cual es tu preferencia entre los argumentos \mathcal{A} y \mathcal{B} ?”. Por último, \mathbb{N} es un conjunto de *preguntas de negador* pendientes, que son argumentos para los cuales el consultante necesita un derrotador del experto.

En particular, serán distinguidos dos estados de sesión especiales: el *estado inicial* y los *estados finales*. El primer componente del estado inicial consiste en el agente consultante $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$ antes de realizar la consulta y adquirir información del agente experto, mientras que el resto de sus componentes están vacíos. Este estado es denotado

$$e_0 = (\mathbf{C}, [], \emptyset, \emptyset).$$

Por otra parte, el primer componente de un estado final consiste en el agente consultante $\mathbf{C}' = (ID, \Pi_{\mathbf{C}'}, \Delta_{\mathbf{C}'}, >_{\mathbf{C}'})$ luego de haber adquirido creencias del agente experto durante toda la sesión. Su segundo componente contiene al menos un argumento (la justificación $\mathcal{J} = \mathcal{F}_1$) y su último elemento es un terminador fin. Por último, el tercer y el cuarto componente están vacíos, *i. e.*, \mathbf{C}' no tiene preguntas pendientes. Este estado es denotado

$$e_f = (\mathbf{C}', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset) \text{ con } n \geq 1.$$

A lo largo de esta sección se hará referencia a la Figura 3.1, la cual ilustra un grafo dirigido que bosqueja el flujo de ejecución de una sesión. Los nodos—identificados con letras griegas—representan diferentes etapas de la interacción, mientras que los arcos dirigidos representan reglas de transición.

Con el objetivo de definir formalmente el protocolo y estrategia de diálogo y luego probar sus propiedades, la semántica operacional de la interacción entre los agentes será definida mediante un *sistema de transiciones* (ver Sección 2.5). Un sistema de transiciones es un conjunto de reglas de transición que describen el comportamiento de un sistema discreto determinando cómo el sistema pasa de un estado a otro. En particular, el protocolo y estrategia de diálogo de consulta a experto será definido mediante el sistema de transiciones $\Theta_{\mathbf{C}} = \{rc_1, rc_2, rc_3, rc_4, rc_5, rc_6\}$. Por ejemplo, la regla de transición rc_1 especificará cómo la sesión evoluciona del estado inicial en el que el agente consultante \mathbf{C} todavía no consultó por q , a otro estado en el que \mathbf{C} adoptó la justificación \mathcal{J} enviada por el agente experto \mathbf{E} . Dado que se asume un escenario colaborativo, los agentes estarán comprometidos a seguir el protocolo.

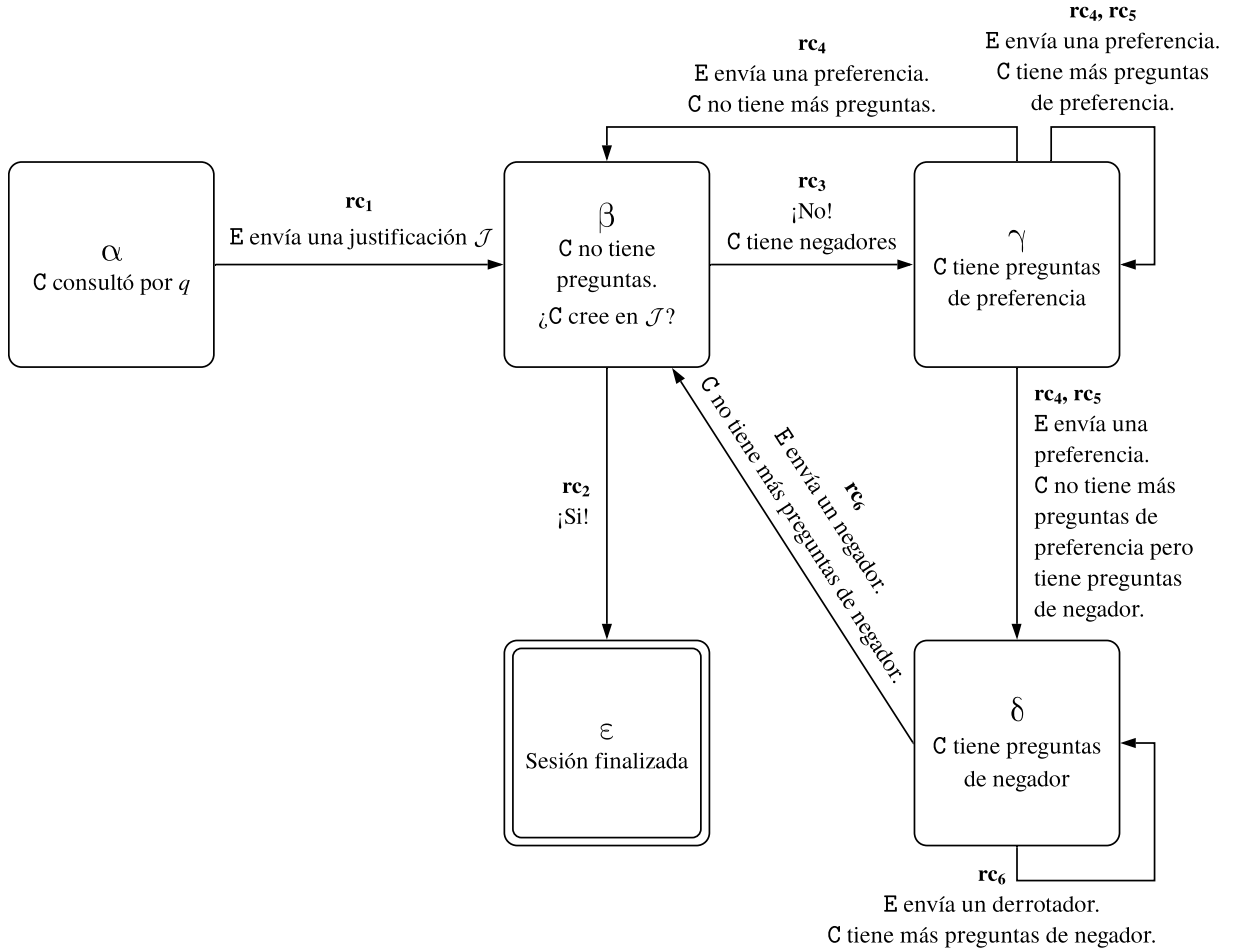


Figura 3.1: Bosquejo de la evolución de una sesión de consulta a experto.

3.2.1. Envío de justificación y adopción de argumentos

Una vez que una sesión ha comenzado, el agente experto $E = (ID, \Pi_E, \Delta_E, >_E)$ debe enviar al agente consultante una justificación para q o \bar{q} , dependiendo de cuál de los literales puede garantizar. La justificación es la raíz de un árbol de dialéctica marcada como \textcircled{U} . Dado que una sesión comienza sólo si E garantiza q o \bar{q} , según la Definición 2.25 este argumento debe existir. Sin embargo, como se mostrará a continuación, el experto podría tener más de un argumento para enviar como justificación en respuesta a la consulta. Formalmente, el conjunto $\text{ADG}(q, \Pi_E, \Delta_E, >_E)$ (o $\text{ADG}(\bar{q}, \Pi_E, \Delta_E, >_E)$, según corresponda) podría tener más de un elemento (ver Definición 2.26).

Ejemplo 3.2 Considere la base de creencias del agente experto $E = (\text{exp}, \Pi_E, \Delta_E, >_E)$:

$$\Pi_E = \left\{ \begin{array}{cc} c & w \\ z & e \\ f & x \end{array} \right\} \quad \Delta_E = \left\{ \begin{array}{cccc} a \prec b & b \prec c & b \prec w, z & a \prec r \\ r \prec c, d & d \prec e & \sim a \prec d & \sim r \prec w \\ \sim d \prec p & p \prec f & \sim p \prec x & \end{array} \right\}$$

A partir de (Π_E, Δ_E) se pueden construir los siguientes argumentos:

$$\begin{aligned} \mathcal{A}_1 &= \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle & \mathcal{A}_2 &= \langle a, \{a \prec b; b \prec w, z\}, \{w, z\} \rangle \\ \mathcal{A}_3 &= \langle a, \{a \prec r; r \prec c, d\}, \{c, d\} \rangle & \mathcal{A}_4 &= \langle \sim a, \{\sim a \prec d; d \prec e\}, \{e\} \rangle \\ \mathcal{A}_5 &= \langle \sim d, \{\sim d \prec p; p \prec f\}, \{f\} \rangle & \mathcal{A}_6 &= \langle \sim r, \{\sim r \prec w\}, w \rangle \\ \mathcal{A}_7 &= \langle \sim p, \{\sim p \prec x\}, x \rangle \end{aligned}$$

Además, considere las siguientes preferencias entre argumentos:

$$\mathcal{A}_4 >_E \mathcal{A}_1, \quad \mathcal{A}_4 >_E \mathcal{A}_2, \quad \mathcal{A}_4 >_E \mathcal{A}_3, \quad \mathcal{A}_5 >_E \mathcal{A}_4, \quad \mathcal{A}_5 >_E \mathcal{A}_7, \quad \mathcal{A}_6 >_E \mathcal{A}_3.$$

La Figura 3.2 ilustra los árboles de dialéctica que el experto construiría para las consultas a y $\sim a$. Note que el experto concluye a y que $\text{ADG}(a, \Pi_E, \Delta_E, >_E) = \{\mathcal{T}_1, \mathcal{T}_2\}$, *i. e.*, existen dos árboles de dialéctica cuya raíz concluye a y está marcada como \textcircled{U} . Por lo tanto, el experto tiene dos posibles justificaciones para la consulta a o $\sim a$: \mathcal{A}_1 y \mathcal{A}_2 . ■

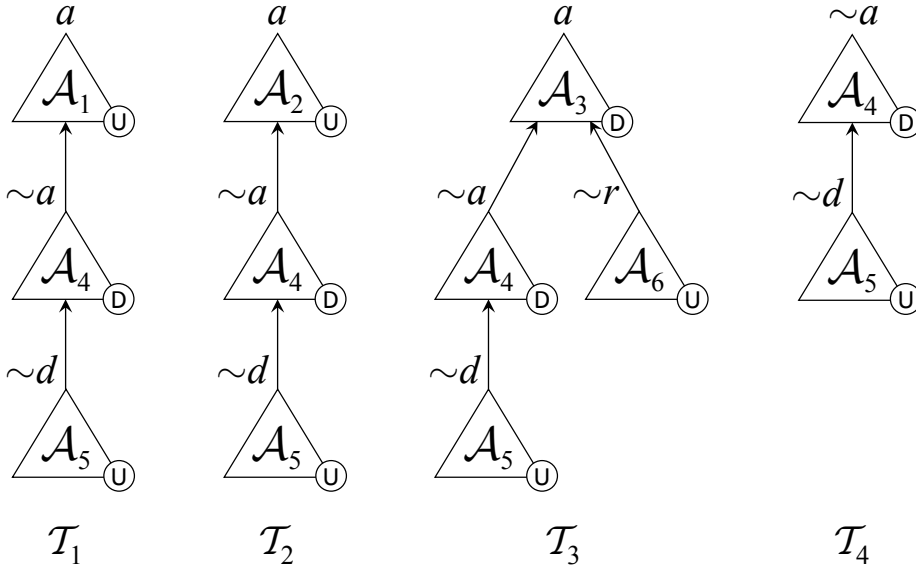


Figura 3.2: Árboles de dialéctica generados por el agente experto en el Ejemplo 3.2.

Dado que pueden existir diferentes estrategias para seleccionar una justificación, esta parte del protocolo y estrategia de diálogo será definida de manera modular y se supondrá

la existencia de un operador SELECCIÓNJUSTIFICACIÓN que selecciona una justificación a partir de un conjunto de árboles de dialéctica. Esto permite utilizar la implementación del operador más adecuada para el dominio de aplicación. Por ejemplo, el experto podría seleccionar el mejor argumento respecto a su relación de preferencia entre argumentos $>_{\mathbf{E}}$. En la Sección 3.4 se definirán dos variantes para el operador SELECCIÓNJUSTIFICACIÓN, con las cuales el experto puede minimizar o reducir la cantidad de interacciones necesarias para que el consultante logre su meta de aceptar la justificación. La respuesta del agente experto a una consulta se define mediante el operador JUSTIFICACIÓN.

Definición 3.2 (Justificación del experto) Dado un literal q y un agente experto $\mathbf{E} = (ID, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$, la *justificación* de \mathbf{E} en respuesta a q se define como:

$$\text{JUSTIFICACIÓN}(q) = \begin{cases} \text{SELECCIÓNJUSTIFICACIÓN}(\text{ADG}(q, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})), & \text{si } \mathbf{E} \text{ concluye } q \\ \text{SELECCIÓNJUSTIFICACIÓN}(\text{ADG}(\bar{q}, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})), & \text{si } \mathbf{E} \text{ concluye } \bar{q} \end{cases}$$

■

Como se mencionó anteriormente, la meta del agente consultante $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$ es marcar la justificación \mathcal{J} enviada por el agente experto como $\textcircled{\mathbf{U}}$ en su árbol de dialéctica enraizado con \mathcal{J} . Si está comprometido a lograr su meta, el consultante deberá primero *adoptar* \mathcal{J} . Adoptar un argumento implica realizar los cambios necesarios en $(\Pi_{\mathbf{C}}, \Delta_{\mathbf{C}})$ para poder construirlo. Por lo tanto, el consultante deberá añadir a su base de creencias todos los hechos y reglas rebatibles del argumento, y eliminar de $\Pi_{\mathbf{C}}$ algunos hechos que son inválidos desde la perspectiva del experto. La adopción de un argumento se define mediante el operador ADPARGUMENTO.

Definición 3.3 (Adopción de argumento) Dado un agente consultante $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$ y un argumento $\mathcal{A} = \langle a, \mathbf{R}, \mathbf{H} \rangle$, la *adopción* de \mathcal{A} por \mathbf{C} se define como $\text{ADPARGUMENTO}(\mathbf{C}, \mathcal{A}) = \mathbf{C}' = (ID, \Pi_{\mathbf{C}'}, \Delta_{\mathbf{C}'}, >_{\mathbf{C}'})$, donde:

$$\Pi_{\mathbf{C}'} = (\Pi_{\mathbf{C}} \cup \mathbf{H}) \setminus (X \cup Y \cup Z), \text{ donde:}$$

$$X = \{x : \bar{x} \in \mathbf{H}\}$$

$$Y = \{y : \bar{y} \prec p_1, \dots, p_n \in \mathbf{R}\}$$

$$Z = \{z : z \prec p_1, \dots, p_n \in \mathbf{R}\}$$

$$\Delta_{\mathbf{C}'} = \Delta_{\mathbf{C}} \cup \mathbf{R}$$

■

La definición anterior tiene en cuenta las cuatro condiciones necesarias para que un conjunto de reglas rebatibles y un conjuntos de hechos sea un argumento (ver Definición 2.13). Dado un agente consultante $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$ que adopta un argumento $\mathcal{A} = \langle a, \mathbf{R}, \mathbf{H} \rangle$, todos los hechos de \mathbf{H} son añadidos a $\Pi_{\mathbf{C}}$ y todas las reglas rebatibles de \mathbf{R} son añadidos a $\Delta_{\mathbf{C}}$, asegurando que se cumplan las condiciones 1 y 2 de la Definición 2.13. Además todos los hechos de $\Pi_{\mathbf{C}}$ que contradicen a algún hecho de \mathbf{H} (conjunto X) o a la cabeza de alguna regla rebatible de \mathbf{R} (conjunto Y) son eliminados, asegurando que la base de creencias de \mathbf{C} no quede inconsistente y que se cumpla la condición 3 de la Definición 2.13. Por último, todos los hechos que se corresponden con la cabeza de alguna regla rebatible de \mathbf{R} son eliminados (conjunto Z), asegurando que \mathcal{A} sea minimal para \mathbf{C} y que se cumpla la condición 4 de la Definición 2.13.

Dado $\text{ADPARGUMENTO}(\mathbf{C}, \mathcal{A}) = \mathbf{C}'$, note que \mathbf{C} y \mathbf{C}' tienen el mismo identificador. Por lo tanto, \mathbf{C}' no es un agente diferente sino el mismo con una base de creencias modificada. En otras palabras, \mathbf{C} *evoluciona* a \mathbf{C}' . Con el objetivo de llevar registro de la evolución de las bases de creencias de los consultantes se añadirán superíndices a las variables de agente. Por ejemplo, $\mathbf{C}^2 = (\text{curly}, \Pi_{\mathbf{C}^2}, \Delta_{\mathbf{C}^2}, >_{\mathbf{C}^2})$ representa el agente `curly` en cierto punto de la sesión, y $\mathbf{C}^3 = (\text{curly}, \Pi_{\mathbf{C}^3}, \Delta_{\mathbf{C}^3}, >_{\mathbf{C}^3})$ representa ese mismo agente luego de que su base de creencias fue modificada. Los siguientes dos ejemplos mostrarán como el operador ADPARGUMENTO se comporta escenarios diferentes.

Ejemplo 3.3 Continuando con el Ejemplo 3.2, considere el agente $\mathbf{L}^1 = (\text{larry}, \Pi_{\mathbf{L}^1}, \Delta_{\mathbf{L}^1}, >_{\mathbf{L}^1})$ con $>_{\mathbf{L}^1} = \emptyset$ (ver Figura 3.3) que consulta al agente experto $\mathbf{E} = (\text{exp}, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$ por a . Dado que el experto concluye a y que $\text{ADG}(a, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}}) = \{\mathcal{T}_1, \mathcal{T}_2\}$, suponemos que $\text{SELECCIÓNJUSTIFICACIÓN}(\{\mathcal{T}_1, \mathcal{T}_2\}) = \mathcal{A}_1$. Por lo tanto el experto envía $\text{JUSTIFICACIÓN}(a) = \mathcal{A}_1 = \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle$ como justificación a `larry`. Con el fin de adoptar \mathcal{A}_1 , `larry` añade c a $\Pi_{\mathbf{L}^1}$ y las reglas rebatibles $a \prec b$ y $b \prec c$ a $\Delta_{\mathbf{L}^1}$ sin necesidad de eliminar hechos de $\Pi_{\mathbf{L}^1}$. La Figura 3.3 ilustra el resultado de la adopción del argumento: $\text{ADPARGUMENTO}(\mathbf{L}^1, \mathcal{A}_1) = \mathbf{L}^2 = (\text{larry}, \Pi_{\mathbf{L}^2}, \Delta_{\mathbf{L}^2}, >_{\mathbf{L}^1})$. En este caso, los conjuntos X , Y y Z de la Definición 3.3 son todos vacíos. ■

$$\begin{array}{l} \Pi_{\mathbf{L}^1} = \left\{ e \quad x \right\} \quad \Delta_{\mathbf{L}^1} = \left\{ \sim a \prec d \quad d \prec e \quad \sim p \prec x \right\} \\ \Pi_{\mathbf{L}^2} = \left\{ e \quad x \quad c \right\} \quad \Delta_{\mathbf{L}^2} = \left\{ \sim a \prec d \quad d \prec e \quad \sim p \prec x \quad a \prec b \quad b \prec c \right\} \end{array}$$

Figura 3.3: El agente `larry` antes y después de adoptar \mathcal{A}_1 en el Ejemplo 3.3.

Ejemplo 3.4 Continuando nuevamente con el Ejemplo 3.2, considere ahora el agente $M^1 = (\text{moe}, \Pi_{M^1}, \Delta_{M^1}, >_{M^1})$ con $>_{M^1} = \emptyset$ (ver Figura 3.4) que, a diferencia del agente Larry, consulta al experto $E = (\text{exp}, \Pi_E, \Delta_E, >_E)$ por $\sim a$. Dado que el experto concluye a (i. e., el complemento de la consulta) y que $\text{ADG}(a, \Pi_E, \Delta_E, >_E) = \{\mathcal{T}_1, \mathcal{T}_2\}$, supongamos que $\text{SELECCIÓNJUSTIFICACIÓN}(\{\mathcal{T}_1, \mathcal{T}_2\}) = \mathcal{A}_1$. Por lo tanto, el agente experto envía $\text{JUSTIFICACIÓN}(\sim a) = \mathcal{A}_1 = \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle$ como justificación a moe. Al adoptar \mathcal{A}_1 , moe añade el hecho c a Π_{M^1} y las reglas rebatibles $a \prec b$ y $b \prec c$ a Δ_{M^1} . Sin embargo, moe también necesita eliminar de Π_{M^1} dos hechos que le impiden construir \mathcal{A}_1 : $\sim b$, dado que es el complemento de la cabeza de la regla $b \prec c$, y $\sim c$, ya que está en contradicción con el hecho c . La Figura 3.4 ilustra el resultado de la adopción: $\text{ADPARGUMENTO}(M^1, \mathcal{A}_1) = M^2 = (\text{moe}, \Pi_{M^2}, \Delta_{M^2}, >_{M^1})$. En este caso, los conjuntos X, Y y Z de la Definición 3.3 son $X = \{\sim c\}$, $Y = \{\sim b\}$ y $Z = \{\}$. ■

$$\begin{array}{l} \Pi_{M^1} = \left\{ \begin{array}{l} \sim b \quad \sim c \quad f \\ f \quad c \end{array} \right\} \\ \Pi_{M^2} = \left\{ \begin{array}{l} \sim b \quad \sim c \quad f \\ f \quad c \end{array} \right\} \end{array} \quad \begin{array}{l} \Delta_{M^1} = \left\{ \begin{array}{l} g \prec f \\ g \prec f \quad a \prec b \quad b \prec c \end{array} \right\} \\ \Delta_{M^2} = \left\{ \begin{array}{l} g \prec f \\ g \prec f \quad a \prec b \quad b \prec c \end{array} \right\} \end{array}$$

Figura 3.4: El agente moe antes y después de adoptar \mathcal{A}_1 en el Ejemplo 3.4.

La siguiente proposición garantiza que siempre que el agente consultante adopta un argumento del agente experto, el consultante podrá construir dicho argumento.

Proposición 3.1 Sea E un agente experto, C^1 un agente consultante, y \mathcal{A} un argumento construido por E , si E envía \mathcal{A} a C^1 y $\text{ADPARGUMENTO}(C^1, \mathcal{A}) = C^2$ entonces \mathcal{A} puede ser construido por C^2 .

Demostración: ver Apéndice [pág. 133]. □

Como se explicará más adelante, la justificación enviada por el agente experto podría no ser el único argumento que el agente consultante necesitará adoptar para poder lograr su meta. Otra propiedad importante es que, luego de adoptar un nuevo argumento del experto, el consultante todavía será capaz de construir aquellos que adoptó previamente.

Proposición 3.2 Sea E un agente experto, C^1 un agente consultante, \mathcal{A} un argumento construido por E , y \mathcal{B} un argumento construido por el E y por C , si E envía \mathcal{A} a C^1 y $\text{ADPARGUMENTO}(C^1, \mathcal{A}) = C^2$ entonces \mathcal{B} puede ser construido por C^2 .

Demostración: ver Apéndice [pág. 134]. \square

Como se mencionó anteriormente, el protocolo y estrategia de diálogo de consulta a experto será definido mediante el sistema de transiciones $\Theta_C = \{rc_1, rc_2, rc_3, rc_4, rc_5, rc_6\}$. A continuación se presentará la regla de transición rc_1 , la cual especifica cómo evoluciona la sesión luego de comenzar (ver α a β en la Figura 3.1).

$$rc_1 : \frac{\text{JUSTIFICACIÓN}(q) = \mathcal{J} \wedge \text{ADPARGUMENTO}(\mathbf{C}, \mathcal{J}) = \mathbf{C}'}{(\mathbf{C}, [\], \emptyset, \emptyset) \rightarrow (\mathbf{C}', [\mathcal{J}], \emptyset, \emptyset)}$$

La regla de transición rc_1 especifica cómo la sesión evoluciona de un estado inicial a otro estado en el cual el agente experto envió la justificación \mathcal{J} en respuesta a la consulta q , y \mathcal{J} fue adoptada por el agente consultante \mathbf{C} , el cual evolucionó a \mathbf{C}' . Observe que el primer componente del estado de sesión es reemplazado por \mathbf{C}' , y su segundo componente—la secuencia de argumentos a favor enviados por el experto—es inicializado con \mathcal{J} .

Ejemplo 3.5 En el Ejemplo 3.3 la sesión comienza en el estado inicial $e_0 = (\mathbf{L}^1, [\], \emptyset, \emptyset)$. La regla de transición rc_1 es aplicada y la sesión evoluciona al estado $e_1 = (\mathbf{L}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ luego de que el experto envió la justificación \mathcal{A}_1 y **larry** la adoptó. Análogamente, en el Ejemplo 3.4, la sesión comienza en el estado inicial $e_0 = (\mathbf{M}^1, [\], \emptyset, \emptyset)$. La regla de transición rc_1 es aplicada y la sesión evoluciona al estado $e_1 = (\mathbf{M}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ luego de que el experto envió la justificación \mathcal{A}_1 y **moe** la adoptó. Observe que, aunque **larry** y **moe** hicieron consultas opuestas, el experto envió la misma justificación. \blacksquare

Considere el estado $e_1 = (\mathbf{M}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ del Ejemplo 3.5, luego de que **moe** adoptó la justificación \mathcal{A}_1 enviada por el experto. Según la Figura 3.4, el único argumento que **moe** puede construir es $\mathcal{A}_1 = \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle$. Por lo tanto, su árbol de dialéctica enraizado con \mathcal{A}_1 es el ilustrado en la Figura 3.5 (izquierda). En este escenario, la sesión podría terminar ya que **moe** logró aceptar la justificación enviada por el experto, *i. e.*, logró marcar \mathcal{A}_1 como $\textcircled{\mathbf{U}}$ en su árbol de dialéctica enraizado con \mathcal{A}_1 .

Por el contrario, considere el estado $e_1 = (\mathbf{L}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ luego de que **larry** adoptó la justificación \mathcal{A}_1 enviada por el experto. Según la Figura 3.3, los argumentos construidos por **larry** son $\mathcal{A}_1 = \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle$ y $\mathcal{A}_4 = \langle \sim a, \{\sim a \prec d; d \prec e\}, \{e\} \rangle$. Dado que $\succ_{\mathbf{L}^2} = \emptyset$, sus árboles de dialéctica son los ilustrados en la Figura 3.5 (derecha). En este escenario, \mathcal{A}_1 tiene el negador \mathcal{A}_4 (*i. e.*, derrotador marcado como $\textcircled{\mathbf{U}}$) y, por lo tanto, **larry** todavía no aceptó la justificación enviada por el experto. En la subsección siguiente se definirá un operador para determinar si es necesario continuar la sesión.

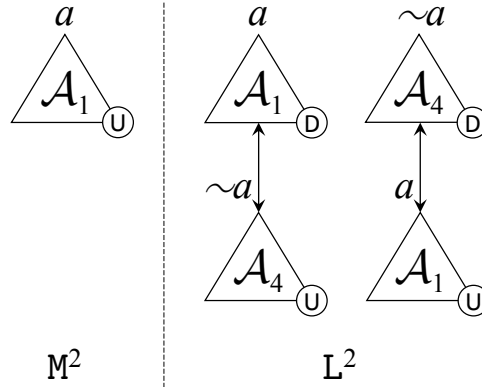


Figura 3.5: Árboles de dialéctica generados por los agentes M^2 y L^2 .

3.2.2. Introspección: ¿es necesario continuar la sesión?

Luego de que el consultante adopta la justificación \mathcal{J} enviada en respuesta a su consulta, la sesión evoluciona a un estado $(\mathcal{C}, [\mathcal{J}], \emptyset, \emptyset)$ en el que no hay preguntas pendientes para hacerle al experto. En ese punto el consultante debe realizar una *introspección* analizando el argumento recibido en conjunto con sus creencias para verificar si ya logró su meta de aceptar \mathcal{J} o, en caso contrario, si es necesario continuar la sesión. Para esto, el agente consultante analizará la marca de \mathcal{J} en su árbol de dialéctica enraizado con \mathcal{J} , y existen dos escenarios posibles:

1. \mathcal{J} está marcado como \textcircled{U} porque no tiene negadores.
2. \mathcal{J} está marcado como \textcircled{D} porque tiene al menos un negador.

En el escenario 1 el consultante logró su meta y la sesión puede finalizar. En cambio, en el escenario 2, el consultante necesitará realizar preguntas al experto sobre los negadores de \mathcal{J} y, por lo tanto, la sesión debe continuar.

Como se explicará más adelante, el agente consultante no sólo necesitará realizar introspecciones sobre la justificación \mathcal{J} sino también sobre otros argumentos a favor de \mathcal{J} enviados por el agente experto. En ambos casos, se utilizará el operador INTROSPECCIÓN definido a continuación.

Definición 3.4 (Introspección) Dado un agente consultante \mathcal{C} y dos argumentos \mathcal{R} y \mathcal{A} , y sea \mathcal{T} el árbol de dialéctica construido por \mathcal{C} tal que $\text{ETIQUETA}(\text{RAÍZ}(\mathcal{T})) = \mathcal{R}$, la *introspección* de \mathcal{C} sobre \mathcal{A} se define como:

$$\text{INTROSPECCIÓN}(\mathcal{C}, \mathcal{R}, \mathcal{A}) = \begin{cases} \textcircled{\mathbf{U}}, & \text{si } \text{MARCA}(\aleph_{\mathcal{A}}, \mathcal{T}) = \textcircled{\mathbf{U}} \text{ para todo nodo } \aleph_{\mathcal{A}} \text{ en } \mathcal{T} \\ \textcircled{\mathbf{D}}, & \text{si } \text{MARCA}(\aleph_{\mathcal{A}}, \mathcal{T}) = \textcircled{\mathbf{D}} \text{ para algún nodo } \aleph_{\mathcal{A}} \text{ en } \mathcal{T} \\ \textcircled{\mathbf{X}}, & \text{si } \mathcal{A} \text{ no está en } \mathcal{T} \end{cases} \quad \blacksquare$$

Recuerde que el subíndice de una variable de nodo corresponde a la variable de argumento que lo etiqueta, *i. e.*, $\text{ETIQUETA}(\aleph_{\mathcal{A}}) = \mathcal{A}$. Además, recuerde que un argumento \mathcal{A} podría derrotar a varios argumentos en un árbol de dialéctica \mathcal{T} , con la restricción de que \mathcal{A} no aparezca más de una vez en la misma línea de argumentación. Por lo tanto, \mathcal{A} podría ser la etiqueta de más de un nodo \aleph en \mathcal{T} . Este no es el caso de la justificación, ya que está en toda línea de argumentación como primer elemento.

Como se mencionó anteriormente, el agente consultante no sólo necesitará realizar introspecciones sobre la justificación \mathcal{J} sino también sobre otros argumentos a favor enviados por el experto. Por lo tanto, dado un estado de sesión $(\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \emptyset, \emptyset)$, el argumento \mathcal{A} de la Definición 3.4 algunas veces se corresponderá con \mathcal{J} , y otras con alguno de los argumentos $\mathcal{F}_2, \dots, \mathcal{F}_n$. En cambio, el argumento \mathcal{R} de la Definición 3.4 siempre se corresponderá con \mathcal{J} . Esto se debe a que el árbol de dialéctica enraizado con \mathcal{J} es el único de interés para el consultante. Por lo tanto, siempre que se hable de la marca de un argumento, implícitamente se hará referencia a la marca de dicho argumento en el árbol de dialéctica (del consultante o del experto, según corresponda) enraizado con \mathcal{J} . Como también se explicará más adelante, un argumento para el cual el consultante realiza una introspección podría no estar en su árbol de dialéctica enraizado con \mathcal{J} , en cuyo caso el operador devolverá $\textcircled{\mathbf{X}}$. Esto podía suceder, por ejemplo, si el consultante actualizó sus creencias y un argumento que antes estaba en el árbol de dialéctica fue desvinculado.

A continuación se presentará la regla de transición rc_2 , la cual especifica el escenario en el que el consultante logró su meta y la sesión puede finalizar (ver β a ε en la Figura 3.1).

$$rc_2 : \frac{\text{INTROSPECCIÓN}(\mathcal{C}, \mathcal{J}, \mathcal{J}) = \textcircled{\mathbf{U}}}{(\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \emptyset, \emptyset) \rightarrow (\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset)} \quad n \geq 1$$

Para facilitar la lectura de los estados y las reglas de transición, cuando sea conveniente el primer elemento de la secuencia de argumentos a favor enviados por el experto será notado como \mathcal{J} en lugar de \mathcal{F}_1 . Si la regla de transición rc_2 es aplicada, la sesión evoluciona a un estado final dado que el consultante logró su meta de aceptar la justificación.

El siguiente ejemplo mostrará un escenario en el cual el agente consultante no tiene negadores para la justificación. Por lo tanto, puede lograr su meta sin necesidad de continuar la sesión.

Ejemplo 3.6 Continuando con el Ejemplo 3.4, considere el estado $e_1 = (\mathbb{M}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ luego de que *moe* adoptó la justificación \mathcal{A}_1 enviada por el experto. Como en este punto *moe* no tiene preguntas pendientes, realiza una introspección para verificar la marca de \mathcal{A}_1 . Según la Figura 3.4, el único argumento que *moe* puede construir es $\mathcal{A}_1 = \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle$ y por lo tanto, su único árbol de dialéctica enraizado es el ilustrado en la Figura 3.5 (izquierda). Luego, $\text{INTROSPECCIÓN}(\mathbb{M}^2, \mathcal{A}_1, \mathcal{A}_1) = \textcircled{\mathbf{U}}$ dado que la justificación \mathcal{A}_1 no tiene negadores. En consecuencia, la regla de transición rc_2 puede ser aplicada y la sesión evoluciona al estado final $e_f = (\mathbb{M}^2, [\mathcal{A}_1, \text{fin}], \emptyset, \emptyset)$. ■

En cambio, el Ejemplo 3.7 mostrará un escenario en el cual el agente consultante tiene un negador para la justificación. Por lo tanto, la sesión deberá continuar.

Ejemplo 3.7 Continuando con el Ejemplo 3.3, considere el estado $e_1 = (\mathbb{L}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ luego de que *larry* adoptó la justificación \mathcal{A}_1 enviada por el experto. Como en este punto *larry* no tiene preguntas pendientes, realiza una introspección para verificar la marca de \mathcal{A}_1 . Según la Figura 3.3, los argumentos construidos por *larry* son $\mathcal{A}_1 = \langle a, \{a \prec b; b \prec c\}, \{c\} \rangle$ y $\mathcal{A}_4 = \langle \sim a, \{\sim a \prec d; d \prec e\}, \{e\} \rangle$. Dado que $\succ_{\mathbb{L}^2} = \emptyset$, sus árboles de dialéctica son los ilustrados en la Figura 3.5 (derecha). Luego, $\text{INTROSPECCIÓN}(\mathbb{L}^2, \mathcal{A}_1, \mathcal{A}_1) = \textcircled{\mathbf{D}}$ dado que la justificación \mathcal{A}_1 tiene el negador \mathcal{A}_4 . Por lo tanto, la regla de transición rc_2 no puede ser aplicada en el estado actual y la sesión todavía no puede evolucionar a un estado final. ■

3.2.3. Preguntas de preferencia: la sesión debe continuar

En el contexto de una sesión, un negador \mathcal{N} para el agente consultante (*e. g.*, \mathcal{A}_4 en el Ejemplo 3.7) es un argumento que causa que otro argumento \mathcal{A} esté marcado como $\textcircled{\mathbf{D}}$ cuando en realidad, desde la perspectiva del agente experto, \mathcal{A} debería estar marcado como $\textcircled{\mathbf{U}}$. Esto puede deberse a tres razones diferentes:

1. El consultante tiene creencias previas en oposición a las creencias del experto—que desde el punto de vista del experto son inválidas—que le permiten construir un negador \mathcal{N} para \mathcal{A} que el experto no puede construir.

2. La relación de preferencia entre argumentos de ambos agentes es diferente.
3. El experto tiene algún derrotador marcado como \textcircled{U} para \mathcal{N} (el cual hace que \mathcal{N} esté marcado como \textcircled{D} y no sea un negador) que el consultante no puede construir por falta de información.

Como ya se explicó anteriormente, una solución naïve para lidiar con los negadores que surgen por las razones 1 y 2 sería eliminarlos de la base de creencias del agente consultante. Sin embargo, esta solución podría implicar la eliminación innecesaria de creencias que—desde la perspectiva del experto—son válidas. Por esta razón, el consultante buscará resolver el conflicto adquiriendo más información relevante del experto a través de preguntas específicas. En particular, el consultante primero actualizará sus preferencias entre argumentos preguntando *preguntas de preferencia* al experto, que lo ayudarán a no considerar ciertos argumentos como negadores.

Una pregunta de preferencia es un par $(\mathcal{N}, \mathcal{A})$ tal que \mathcal{A} es un argumento a favor previamente enviado por el experto y \mathcal{N} es un negador para \mathcal{A} . Como se explicará más adelante, el experto responderá cada pregunta de preferencia con una *preferencia* del conjunto $>_{\mathbf{E}}$, la cual el consultante *adoptará* para actualizar $>_{\mathbf{C}}$ de manera acorde. El conjunto de preguntas de preferencia para un argumento se define mediante el operador `PREGUNTASREFERENCIA`.

Definición 3.5 (Preguntas de preferencia) Dado un agente consultante \mathbf{C} y dos argumentos \mathcal{R} y \mathcal{A} , y sea \mathcal{T} el árbol de dialéctica construido por \mathbf{C} tal que $\text{ETIQUETA}(\text{RAÍZ}(\mathcal{T})) = \mathcal{R}$, el conjunto de *preguntas de preferencia* de \mathbf{C} para \mathcal{A} se define como $\text{PREGUNTASREFERENCIA}(\mathbf{C}, \mathcal{R}, \mathcal{A}) = \{(\mathcal{N}, \mathcal{A}) \mid \mathfrak{N}_{\mathcal{N}} \text{ es un nodo en } \mathcal{T} \text{ tal que } \text{MARCA}(\mathfrak{N}_{\mathcal{N}}, \mathcal{T}) = \textcircled{U} \text{ y } \mathcal{N} \text{ es un derrotador para } \mathcal{A}\}$. ■

De manera similar a las introspecciones, el conjunto de preguntas de preferencia para un argumento es obtenido a partir del árbol de dialéctica enraizado con la justificación \mathcal{J} enviada por el experto. Por lo tanto, el argumento \mathcal{R} de la Definición 3.4 siempre se corresponderá con \mathcal{J} .

Ejemplo 3.8 Continuando con el Ejemplo 3.7, considere el estado $e_1 = (\mathbf{L}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ y recuerde que $\text{INTROSPECCIÓN}(\mathbf{L}^2, \mathcal{A}_1, \mathcal{A}_1) = \textcircled{D}$ ya que la justificación \mathcal{A}_1 tiene el negador \mathcal{A}_4 (ver Figura 3.5, derecha). Por lo tanto, $\text{PREGUNTASREFERENCIA}(\mathbf{L}^2, \mathcal{A}_1, \mathcal{A}_1) = \{(\mathcal{A}_4, \mathcal{A}_1)\}$. ■

A continuación se incluye otro ejemplo en el cual *curly*, un nuevo agente consultante, tiene que lidiar con más de un negador para la justificación.

Ejemplo 3.9 Considere una sesión diferente cuyo estado actual es $e_1 = (\mathcal{C}^2, [\mathcal{B}_1], \emptyset, \emptyset)$, luego de que un agente $\mathcal{C}^1 = (\text{curly}, \Pi_{\mathcal{C}^1}, \Delta_{\mathcal{C}^1}, >_{\mathcal{C}^1})$ consultó a un agente experto por un cierto literal, adoptó la justificación \mathcal{B}_1 , y evolucionó a \mathcal{C}^2 . Supongamos que \mathcal{C}^2 puede construir los argumentos $\mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_5$ y \mathcal{B}_6 , los cuales derrotan a la justificación \mathcal{B}_1 , y \mathcal{B}_7 , el cual derrota a \mathcal{B}_6 . Además, supongamos que las preferencias de *curly* son $\mathcal{B}_2 >_{\mathcal{C}^2} \mathcal{B}_1$, $\mathcal{B}_3 >_{\mathcal{C}^2} \mathcal{B}_1$, $\mathcal{B}_4 >_{\mathcal{C}^2} \mathcal{B}_1$ y $\mathcal{B}_6 >_{\mathcal{C}^2} \mathcal{B}_1$. La Figura 3.6 ilustra su árbol de dialéctica enraizado con \mathcal{B}_1 , en el cual hay cuatro negadores: $\mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4$ (derrotadores propios), y \mathcal{B}_5 (derrotador por bloqueo). Dado que \mathcal{B}_1 es el último argumento a favor enviado por el experto e $\text{INTROSPECCIÓN}(\mathcal{C}^2, \mathcal{B}_1, \mathcal{B}_1) = \textcircled{D}$, *curly* genera el conjunto de preguntas de preferencia $\text{PREGUNTAS PREFERENCIA}(\mathcal{C}^2, \mathcal{B}_1, \mathcal{B}_1) = \{(\mathcal{B}_2, \mathcal{B}_1), (\mathcal{B}_3, \mathcal{B}_1), (\mathcal{B}_4, \mathcal{B}_1), (\mathcal{B}_5, \mathcal{B}_1)\}$. ■

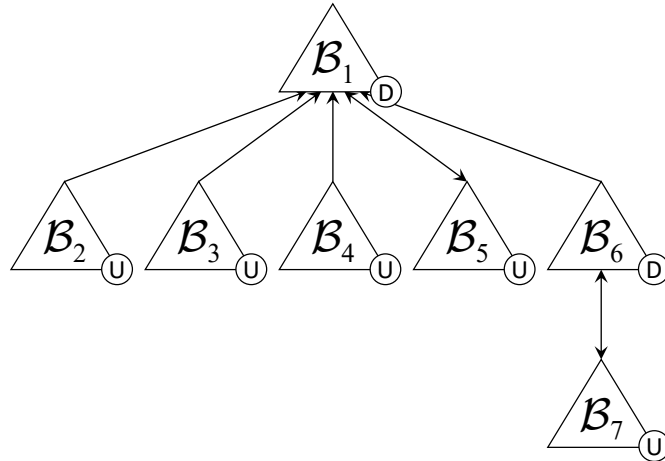


Figura 3.6: Árbol de dialéctica del generado por el agente *curly* en el Ejemplo 3.9.

Como se explicará en detalle más adelante, las preguntas de preferencia tal vez no sean suficientes para lidiar con los negadores. En estos casos, para que el agente consultante pueda cumplir su meta será necesario que el agente experto le envíe otros argumentos a favor. Estos argumentos son almacenados en el segundo componente del estado de sesión y derrotan a los negadores del consultante. Sin embargo, el consultante podría tener negadores para esos derrotadores, negadores para los nuevos derrotadores que envíe el experto, y así sucesivamente, creando un proceso iterativo de preguntas y respuestas que refinan sus creencias sobre el tema de la consulta y también hacen crecer su árbol

de dialéctica enraizado con la justificación. Por lo tanto, si la sesión evolucionó a un estado en el que el consultante no tiene preguntas pendientes para hacerle al experto (*i. e.*, $(\mathbb{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \emptyset, \emptyset)$), el consultante no necesariamente le preguntará sobre \mathcal{J} sino sobre el último argumento a favor enviado por el experto (*i. e.*, el argumento más a la derecha en $[\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n]$) que esté marcado como $\textcircled{\text{D}}$. En otras palabras, el consultante lidiará primero con los negadores de los argumentos a favor que se encuentran más lejos de la raíz. En consecuencia, considerando el procedimiento de marcado de DeLP (ver Definición 2.22), cuando un argumento a favor pase de estar marcado como $\textcircled{\text{D}}$ a estar marcado como $\textcircled{\text{U}}$, el próximo argumento a favor en esa línea de argumentación (arriba) tal vez pase a estar marcado como $\textcircled{\text{U}}$ (si no tiene otros negadores), y así sucesivamente, como un efecto en cadena.

A continuación se presentará la regla de transición rc_3 , la cual especifica cómo evoluciona la sesión cuando el consultante no tiene preguntas pendientes pero tiene negadores (ver β a γ en la Figura 3.1). Recuerde que \in^* corresponde a la pertenencia de un elemento en una secuencia. Por conveniencia, se supondrá que los subíndices de los argumentos en la secuencia $[\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_n]$ con $\mathcal{F}_1 = \mathcal{J}$ están ordenados de menor a mayor.

$$rc_3 : \frac{\begin{array}{l} \mathbb{F} = [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_i, \dots, \mathcal{F}_n] \wedge \text{INTROSPECCIÓN}(\mathbb{C}, \mathcal{J}, \mathcal{F}_i) = \textcircled{\text{D}} \wedge \\ \text{PREGUNTAS PREFERENCIA}(\mathbb{C}, \mathcal{J}, \mathcal{F}_i) = \mathbb{P} \wedge \\ \text{no existe } \mathcal{F}_j \in^* \mathbb{F} (j > i) \text{ tal que } \text{INTROSPECCIÓN}(\mathbb{C}, \mathcal{J}, \mathcal{F}_j) = \textcircled{\text{D}} \end{array}}{(\mathbb{C}, \mathbb{F}, \emptyset, \emptyset) \rightarrow (\mathbb{C}, \mathbb{F}, \mathbb{P}, \emptyset)} \quad n \geq i \geq 1$$

Dado un estado en el que el consultante no tiene preguntas pendientes, la regla de transición rc_3 especifica cómo dicho estado evoluciona a uno nuevo en el que se generó un conjunto de preguntas de preferencia \mathbb{P} para \mathcal{F}_i , el argumento más a la derecha en \mathbb{F} (*i. e.*, el último argumento a favor enviado por el experto) marcado como $\textcircled{\text{D}}$. Observe que el tercer componente del estado de sesión es inicializado con \mathbb{P} .

Ejemplo 3.10 Continuando con el Ejemplo 3.8, considere el estado $e_1 = (\mathbb{L}^2, [\mathcal{A}_1], \emptyset, \emptyset)$ en el que $\text{INTROSPECCIÓN}(\mathbb{L}^2, \mathcal{A}_1, \mathcal{A}_1) = \textcircled{\text{D}}$ y $\text{PREGUNTAS PREFERENCIA}(\mathbb{L}^2, \mathcal{A}_1, \mathcal{A}_1) = \{(\mathcal{A}_4, \mathcal{A}_1)\}$. Por lo tanto, la regla de transición rc_3 es aplicada y la sesión evoluciona al estado $e_2 = (\mathbb{L}^2, [\mathcal{A}_1], \{(\mathcal{A}_4, \mathcal{A}_1)\}, \emptyset)$ en el que larry tiene una pregunta de preferencia para hacerle al experto. ■

Ejemplo 3.11 Continuando con el Ejemplo 3.9, considere el estado $e_1 = (\mathcal{C}^2, [\mathcal{B}_1], \emptyset, \emptyset)$ en el que $\text{INTROSPECCIÓN}(\mathcal{C}^2, \mathcal{B}_1, \mathcal{B}_1) = \textcircled{\text{D}}$ y $\text{PREGUNTAS PREFERENCIA}(\mathcal{C}^2, \mathcal{B}_1, \mathcal{B}_1) = \{(\mathcal{B}_2, \mathcal{B}_1), (\mathcal{B}_3, \mathcal{B}_1), (\mathcal{B}_4, \mathcal{B}_1), (\mathcal{B}_5, \mathcal{B}_1)\}$. Por lo tanto, la regla de transición rc_3 es aplicada y la sesión evoluciona al estado $e_2 = (\mathcal{C}^2, [\mathcal{B}_1], \{(\mathcal{B}_2, \mathcal{B}_1), (\mathcal{B}_3, \mathcal{B}_1), (\mathcal{B}_4, \mathcal{B}_1), (\mathcal{B}_5, \mathcal{B}_1)\}, \emptyset)$ en el que *curly* tiene cuatro preguntas de preferencia para hacerle al experto. ■

Siempre que el consultante genere un conjunto de preguntas de preferencia, el experto las responderá una a una enviando su propia *preferencia* entre el correspondiente par de argumentos. Dada una pregunta de preferencia $(\mathcal{N}, \mathcal{A})$, existen tres posibilidades:

- El experto cree que \mathcal{N} es mejor que \mathcal{A} (prefiere a \mathcal{N} sobre \mathcal{A}).
- El experto cree que \mathcal{N} es peor que \mathcal{A} (prefiere a \mathcal{A} sobre \mathcal{N}).
- El experto cree que \mathcal{N} y \mathcal{A} están *no-relacionados* (no prefiere ni a \mathcal{N} ni a \mathcal{A}).

El experto $\mathbf{E} = (ID, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$ siempre podrá construir \mathcal{A} ya que es uno de sus argumentos previamente adoptados por el consultante. Sin embargo, el experto podría no construir \mathcal{N} si—desde su perspectiva—es un argumento inválido. Esto podría suceder si \mathcal{N} utiliza hechos que no están en $\Pi_{\mathbf{E}}$, utiliza reglas rebatibles que no están en $\Delta_{\mathbf{E}}$, utiliza hechos en contradicción con $\Pi_{\mathbf{E}}$, o no es minimal respecto a otro argumento construido por el experto. En estos casos, el experto siempre creará que un argumento que no puede construir es peor que un argumento que sí puede construir. La respuesta del agente experto a una pregunta de preferencia se define mediante el operador *PREFERENCIA*. Recuerde que en DeLP todo argumento es subargumento de sí mismo.

Definición 3.6 (Preferencia del experto) Dado un agente experto $\mathbf{E} = (ID, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$ y dos argumentos \mathcal{N} y \mathcal{A} , sea \mathcal{S} el subargumento de \mathcal{A} en conflicto con \mathcal{N} , la *preferencia* de \mathbf{E} entre \mathcal{N} y \mathcal{A} se define como:

$$\text{PREFERENCIA}(\mathcal{N}, \mathcal{A}) = \begin{cases} \text{mejor, si } \mathcal{N} >_{\mathbf{E}} \mathcal{S} \\ \text{peor, si } \mathcal{S} >_{\mathbf{E}} \mathcal{N} \text{ o } \mathbf{E} \text{ no puede construir } \mathcal{N} \\ \text{no-relacionados, si } \mathcal{N} \not>_{\mathbf{E}} \mathcal{S} \text{ y } \mathcal{S} \not>_{\mathbf{E}} \mathcal{N} \text{ y } \mathbf{E} \text{ puede construir } \mathcal{N} \end{cases} \blacksquare$$

Si el consultante $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$ está comprometido a lograr su meta, al recibir una preferencia del experto deberá actualizar $>_{\mathbf{C}}$ de manera acorde. Esto podría implicar la eliminación de alguna preferencia que, desde la perspectiva del experto, es inválida. Tenga

en cuenta que si \mathbf{C} hizo la pregunta de preferencia $(\mathcal{N}, \mathcal{A})$, entonces \mathcal{N} es un negador para \mathcal{A} . Luego, existen dos posibilidades: \mathcal{N} es un derrotador propio porque \mathbf{C} cree que \mathcal{N} es mejor que \mathcal{A} (*i. e.*, $\mathcal{N} >_{\mathbf{C}} \mathcal{S}$); o \mathcal{N} es un derrotador por bloqueo porque \mathbf{C} cree que \mathcal{N} y \mathcal{A} están no-relacionados (*i. e.*, $\mathcal{N} \not>_{\mathbf{C}} \mathcal{S}$ y $\mathcal{S} \not>_{\mathbf{C}} \mathcal{N}$). La adopción de una preferencia se define mediante el operador ADPPREFERENCIA.

Definición 3.7 (Adopción de preferencia) Dado un agente consultante $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$, dos argumentos \mathcal{N} y \mathcal{A} , y una preferencia $Pref \in \{\text{mejor, peor, no-relacionados}\}$, y sea \mathcal{S} el subargumento de \mathcal{A} en conflicto con \mathcal{N} , la adopción de $Pref$ para $(\mathcal{N}, \mathcal{A})$ por \mathbf{C} se define como $\text{ADPPREFERENCIA}(\mathbf{C}, \mathcal{N}, \mathcal{A}, Pref) = \mathbf{C}' = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}'})$, donde:

$$>_{\mathbf{C}'} = \begin{cases} >_{\mathbf{C}} \cup \{(\mathcal{N}, \mathcal{S})\}, & \text{si } Pref = \text{mejor} \\ (>_{\mathbf{C}} \setminus \{(\mathcal{N}, \mathcal{S})\}) \cup \{(\mathcal{S}, \mathcal{N})\}, & \text{si } Pref = \text{peor} \\ >_{\mathbf{C}} \setminus \{(\mathcal{N}, \mathcal{S})\}, & \text{si } Pref = \text{no-relacionados} \end{cases} \quad \blacksquare$$

Teniendo en cuenta la Definición 2.16 y las tres posibles respuestas del agente experto, los efectos en el árbol de dialéctica del agente consultante luego de la adopción de una preferencia son los siguientes:

- Si el experto respondió **mejor**, entonces \mathcal{N} continuará siendo un negador para \mathcal{A} . Si \mathcal{N} era un derrotador por bloqueo para \mathcal{A} , pasará a ser un derrotador propio.
- Si el experto respondió **peor**, entonces \mathcal{N} no estará más en el árbol de dialéctica dado que no será más un derrotador para \mathcal{A} .
- Si el experto respondió **no-relacionados**, entonces existen dos posibilidades: si \mathcal{N} era un derrotador por bloqueo para \mathcal{A} , no habrá cambio alguno; en cambio, si \mathcal{N} era un derrotador propio para \mathcal{A} , o bien pasará a ser un derrotador por bloqueo, o bien no estará más en el árbol de dialéctica. Esto depende de qué tipo de derrotador es \mathcal{A} para el siguiente argumento en esa línea de argumentación, dado que en DeLP no pueden haber dos derrotadores por bloqueo consecutivos.

A continuación se presentarán las reglas de transición rc_4 y rc_5 , las cuales especifican cómo evoluciona la sesión cuando el agente consultante adopta una preferencia enviada por el agente experto (ver γ a β , γ a γ , y γ a δ en la Figura 3.1). En particular, rc_4 puede ser aplicada sólo si \mathcal{N} deja de ser un negador, *i. e.*, pasa a estar marcado como $\textcircled{\mathbf{D}}$ o deja de estar en el árbol de dialéctica. En cambio, rc_5 puede ser aplicada sólo si la preferencia adoptada no es suficiente para lidiar con \mathcal{N} . Recuerde que, según la Definición 3.4, el resultado de una introspección sobre un argumento que no está en el árbol es $\textcircled{\mathbf{X}}$.

$$rc_4 : \frac{(\mathcal{N}, \mathcal{A}) \in \mathbb{P} \wedge \text{PREFERENCIA}(\mathcal{N}, \mathcal{A}) = \text{Pref} \wedge \text{ADPPREFERENCIA}(\mathcal{C}, \mathcal{N}, \mathcal{A}, \text{Pref}) = \mathcal{C}' \wedge \text{INTROSPECCIÓN}(\mathcal{C}', \mathcal{J}, \mathcal{N}) \neq \textcircled{\cup}}{(\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P}, \mathbb{N}) \rightarrow (\mathcal{C}', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P} \setminus \{(\mathcal{N}, \mathcal{A})\}, \mathbb{N})} \quad n \geq 1$$

$$rc_5 : \frac{(\mathcal{N}, \mathcal{A}) \in \mathbb{P} \wedge \text{PREFERENCIA}(\mathcal{N}, \mathcal{A}) = \text{Pref} \wedge \text{ADPPREFERENCIA}(\mathcal{C}, \mathcal{N}, \mathcal{A}, \text{Pref}) = \mathcal{C}' \wedge \text{INTROSPECCIÓN}(\mathcal{C}', \mathcal{J}, \mathcal{N}) = \textcircled{\cup}}{(\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P}, \mathbb{N}) \rightarrow (\mathcal{C}', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P} \setminus \{(\mathcal{N}, \mathcal{A})\}, \mathbb{N} \cup \{\mathcal{N}\})} \quad n \geq 1$$

Dado un estado en el que hay al menos una pregunta de preferencia $(\mathcal{N}, \mathcal{A})$ pendiente, las reglas de transición rc_4 y rc_5 especifican cómo dicho estado evoluciona en uno nuevo en el que el agente experto envió su preferencia $\text{Pref} \in \{\text{mejor}, \text{peor}, \text{no-relacionados}\}$ entre \mathcal{N} y \mathcal{A} , y Pref fue adoptada por el agente consultante \mathcal{C} , el cual evolucionó a \mathcal{C}' . Observe que $(\mathcal{N}, \mathcal{A})$ es eliminado del conjunto de preguntas de preferencia pendientes \mathbb{P} . Por lo tanto, las reglas de transición rc_4 y rc_5 serán aplicadas hasta que \mathbb{P} esté vacío. A diferencia de rc_4 , cuando rc_5 es aplicada \mathcal{N} es añadido al conjunto de *preguntas de negador* pendientes \mathbb{N} dado que el consultante necesita del experto un derrotador para este argumento.

Ejemplo 3.12 Continuando con el Ejemplo 3.11, considere el estado $e_2 = (\mathcal{C}^2, [\mathcal{B}_1], \{(\mathcal{B}_2, \mathcal{B}_1), (\mathcal{B}_3, \mathcal{B}_1), (\mathcal{B}_4, \mathcal{B}_1), (\mathcal{B}_5, \mathcal{B}_1)\}, \emptyset)$ luego de que el agente consultante *curly* generó el conjunto de preguntas de preferencia para \mathcal{B}_1 . Recuerde que sus preferencias entre argumentos son $\mathcal{B}_2 >_{\mathcal{C}^2} \mathcal{B}_1$, $\mathcal{B}_3 >_{\mathcal{C}^2} \mathcal{B}_1$, $\mathcal{B}_4 >_{\mathcal{C}^2} \mathcal{B}_1$ y $\mathcal{B}_6 >_{\mathcal{C}^2} \mathcal{B}_1$, y supongamos que las del agente experto son $\mathcal{B}_5 >_{\mathbf{E}} \mathcal{B}_1$, $\mathcal{B}_8 >_{\mathbf{E}} \mathcal{B}_2$, $\mathcal{B}_9 >_{\mathbf{E}} \mathcal{B}_5$, $\mathcal{B}_{10} >_{\mathbf{E}} \mathcal{B}_5$, $\mathcal{B}_{11} >_{\mathbf{E}} \mathcal{B}_{10}$ y $\mathcal{B}_1 >_{\mathbf{E}} \mathcal{B}_4$. La Figura 3.7 ilustra los árboles de dialéctica (relevantes a la sesión) de cada agente.

Primero, el agente *curly* hace la pregunta de preferencia $(\mathcal{B}_2, \mathcal{B}_1)$ y el agente experto responde $\text{PREFERENCIA}(\mathcal{B}_2, \mathcal{B}_1) = \text{no-relacionados}$. En consecuencia, \mathcal{C}^2 evoluciona a $\text{ADPPREFERENCIA}(\mathcal{C}^2, \mathcal{B}_2, \mathcal{B}_1, \text{no-relacionados}) = \mathcal{C}^3$ (*i. e.*, elimina $\mathcal{B}_2 >_{\mathcal{C}^2} \mathcal{B}_1$) y \mathcal{B}_2 pasa de ser un derrotador propio a ser un derrotador por bloqueo (ver [1](#) en la Figura 3.7). Como $\text{INTROSPECCIÓN}(\mathcal{C}^3, \mathcal{B}_2, \mathcal{B}_1) = \textcircled{\cup}$ (*i. e.*, \mathcal{B}_2 continúa siendo un negador) la regla de transición rc_5 es aplicada, añadiendo \mathcal{B}_2 al conjunto de preguntas de negador pendientes. Luego, la sesión evoluciona al estado $e_3 = (\mathcal{C}^3, [\mathcal{B}_1], \{(\mathcal{B}_3, \mathcal{B}_1), (\mathcal{B}_4, \mathcal{B}_1), (\mathcal{B}_5, \mathcal{B}_1)\}, \{\mathcal{B}_2\})$.

Después, *curly* hace la pregunta de preferencia $(\mathcal{B}_3, \mathcal{B}_1)$ y el agente experto responde $\text{PREFERENCIA}(\mathcal{B}_3, \mathcal{B}_1) = \text{peor}$ dado que no puede construir el argumento \mathcal{B}_3 . Por lo tanto, \mathcal{C}^3 evoluciona a $\text{ADPPREFERENCIA}(\mathcal{C}^3, \mathcal{B}_3, \mathcal{B}_1, \text{peor}) = \mathcal{C}^4$ (*i. e.*, elimina

$\mathcal{B}_3 >_{\mathcal{C}^3} \mathcal{B}_1$ y añade $\mathcal{B}_1 >_{\mathcal{C}^3} \mathcal{B}_3$), y \mathcal{B}_3 deja de ser un derrotador para \mathcal{B}_1 (ver [2]). Como $\text{INTROSPECCIÓN}(\mathcal{C}^4, \mathcal{B}_3, \mathcal{B}_1) = \otimes$ (*i. e.*, \mathcal{B}_3 dejó de ser un negador) la regla de transición rc_4 es aplicada y la sesión evoluciona al estado $e_4 = (\mathcal{C}^4, [\mathcal{B}_1], \{(\mathcal{B}_4, \mathcal{B}_1), (\mathcal{B}_5, \mathcal{B}_1)\}, \{\mathcal{B}_2\})$.

A continuación, el agente *curly* hace la pregunta de preferencia $(\mathcal{B}_4, \mathcal{B}_1)$ y el agente experto responde $\text{PREFERENCIA}(\mathcal{B}_4, \mathcal{B}_1) = \text{peor}$. En consecuencia, \mathcal{C}^4 evoluciona a $\text{ADPPREFERENCIA}(\mathcal{C}^4, \mathcal{B}_4, \mathcal{B}_1, \text{peor}) = \mathcal{C}^5$ (*i. e.*, elimina la preferencia $\mathcal{B}_4 >_{\mathcal{C}^4} \mathcal{B}_1$ y añade $\mathcal{B}_1 >_{\mathcal{C}^4} \mathcal{B}_4$), y \mathcal{B}_4 deja de ser un derrotador para \mathcal{B}_1 (ver [3]). Como $\text{INTROSPECCIÓN}(\mathcal{C}^5, \mathcal{B}_4, \mathcal{B}_1) = \otimes$ (*i. e.*, \mathcal{B}_4 dejó de ser un negador) la regla de transición rc_4 es aplicada y la sesión evoluciona al estado $e_5 = (\mathcal{C}^5, [\mathcal{B}_1], \{(\mathcal{B}_5, \mathcal{B}_1)\}, \{\mathcal{B}_2\})$.

Por último, el agente *curly* hace la pregunta de preferencia $(\mathcal{B}_5, \mathcal{B}_1)$ y el agente experto responde $\text{PREFERENCIA}(\mathcal{B}_5, \mathcal{B}_1) = \text{mejor}$. Por lo tanto, \mathcal{C}^5 evoluciona a $\text{ADPPREFERENCIA}(\mathcal{C}^5, \mathcal{B}_5, \mathcal{B}_1, \text{mejor}) = \mathcal{C}^6$, *i. e.*, añade $\mathcal{B}_5 >_{\mathcal{C}^5} \mathcal{B}_1$, y \mathcal{B}_5 pasa de ser un derrotador por bloqueo a ser un derrotador propio (ver [4]). Como $\text{INTROSPECCIÓN}(\mathcal{C}^6, \mathcal{B}_5, \mathcal{B}_1) = \oplus$ (*i. e.*, \mathcal{B}_5 continúa siendo un negador) la regla de transición rc_5 es aplicada, añadiendo \mathcal{B}_5 al conjunto de preguntas de negador pendientes. Por último, la sesión evoluciona al estado $e_6 = (\mathcal{C}^6, [\mathcal{B}_1], \emptyset, \{\mathcal{B}_2, \mathcal{B}_5\})$. ■

El ejemplo anterior muestra que las preguntas de preferencia pueden no ser suficientes para lidiar con los negadores de un argumento. En particular, los argumentos \mathcal{B}_2 y \mathcal{B}_5 del ejemplo continúan siendo negadores de la raíz del árbol de dialéctica en cuestión. Cuando esto sucede, el agente consultante le pedirá al agente experto que le envíe argumentos que derroten a los negadores pendientes.

Como se verá en la siguiente subsección, todos los argumentos que el experto envía (*i. e.*, los argumentos a favor que se almacenan en el segundo componente de los estados de sesión) están marcados como \oplus en su árbol de dialéctica. Esto implica que, según la regla de transición rc_3 , el argumento \mathcal{A} de toda pregunta de preferencia $(\mathcal{N}, \mathcal{A})$ siempre está marcado como \oplus en el árbol de dialéctica del experto. Por lo tanto, si un negador \mathcal{N} del consultante continúa siéndolo luego de actualizar sus preferencias (*i. e.*, \mathcal{N} es efectivamente un derrotador para \mathcal{A}), \mathcal{N} está marcado como \oplus en el árbol de dialéctica del experto; de lo contrario, \mathcal{A} no estaría marcado como \oplus . En consecuencia, el experto debe tener al menos un derrotador marcado como \oplus para \mathcal{N} . Como el agente consultante necesita dicho derrotador para lograr su meta, podrá preguntárselo al agente experto a través de una *pregunta de negador*.

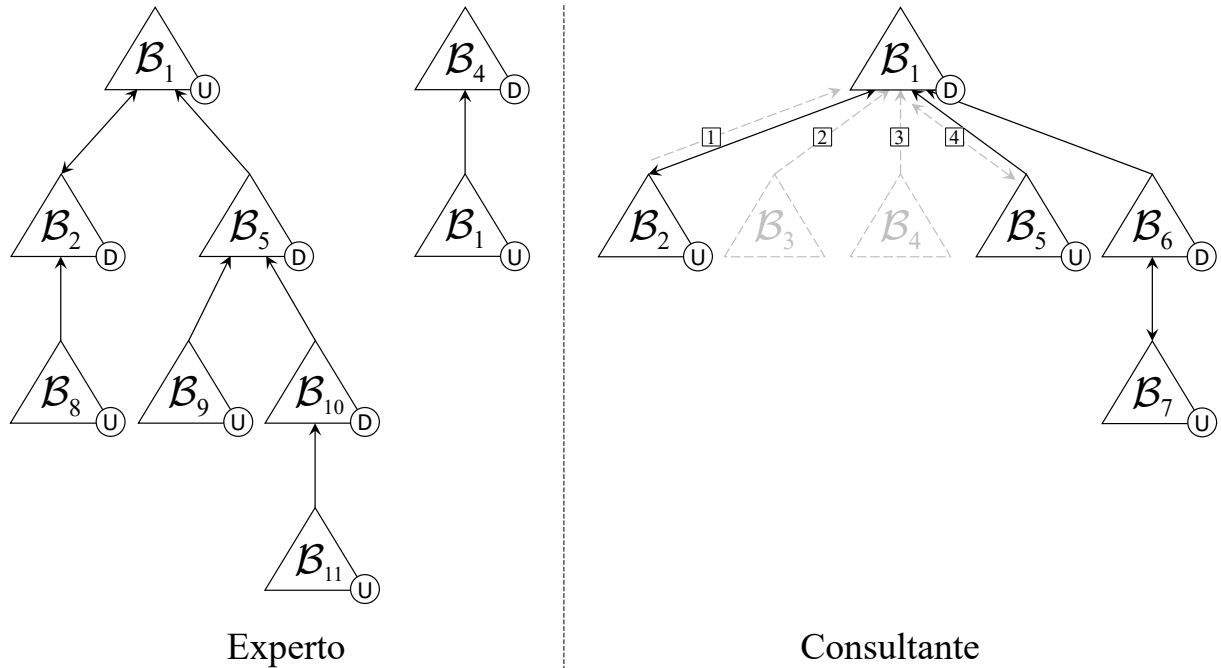


Figura 3.7: Árboles de dialéctica generados por el experto y `curly` en el Ejemplo 3.12. Los argumentos grises son aquellos negadores que dejaron de ser parte del árbol de dialéctica.

3.2.4. Preguntas de negador: requiriendo más argumentos

Como se explicó en la subsección anterior, la regla de transición rc_5 es aplicada cuando una pregunta de preferencia no fue suficiente para lidiar con un negador particular. En esos casos, el negador en cuestión pasa a ser una pregunta de negador pendiente.

Cuando el consultante tenga preguntas de negador pendientes, el experto las responderá una a una enviando un derrotador marcado como \textcircled{U} para el argumento en cuestión. Al igual que con la selección del árbol de dialéctica para la justificación, el experto podría tener múltiples derrotadores marcados como \textcircled{U} para un negador. Dado que pueden existir diferentes estrategias para seleccionar un derrotador, esta parte del protocolo y estrategia de diálogo será definida de manera modular y se supondrá la existencia de un operador `SELECCIÓNDERROTADOR` que selecciona un derrotador a partir de un conjunto de argumentos. Esto permite utilizar la implementación del operador más adecuada para el dominio de aplicación. Por ejemplo, el experto podría seleccionar el mejor argumento respecto a su relación de preferencia entre argumentos $>_E$. En la Sección 3.4 se definirán dos variantes para el operador `SELECCIÓNDERROTADOR`, con las cuales el agente exper-

to puede minimizar o reducir la cantidad de interacciones necesarias para que el agente consultante logre aceptar la justificación.

Dada una pregunta de negador, el experto también enviará su preferencia entre el derrotador y el negador (**mejor**, **peor** o **no-relacionados**) para que el consultante actualice su conjunto de preferencias y capture el mismo tipo de derrota entre los argumentos (propia o por bloqueo). La respuesta del agente experto a una pregunta de negador se define mediante el operador DERROTADOR.

Definición 3.8 (Derrotador del experto) Dado un experto $E = (ID, \Pi_E, \Delta_E, >_E)$, dos argumentos \mathcal{R} y \mathcal{N} , y \mathbb{F} una secuencia de argumentos recibidos por el consultante. Sea \mathcal{T} el árbol de dialéctica construido por E tal que $ETIQUETA(RAÍZ(\mathcal{T})) = \mathcal{R}$, sea $Todos = \{\aleph \mid \aleph \in HIJOS(\aleph_{\mathcal{N}}) \text{ con } MARCA(\aleph, \mathcal{T}) = \textcircled{U} \text{ para todo } \aleph_{\mathcal{N}} \text{ en } \mathcal{T}\}$, y sea $NoEnviados = \{\aleph \mid \aleph \in Todos \text{ tal que } ETIQUETA(\aleph) \notin \mathbb{F}\}$, el *derrotador* para \mathcal{N} de E se define como $DERROTADOR(\mathcal{R}, \mathcal{N}, \mathbb{F}) = (\mathcal{D}, Pref)$, donde

$$\mathcal{D} = \begin{cases} SELECCIÓNDERROTADOR(NoEnviados), & \text{si } NoEnviados \neq \emptyset \\ SELECCIÓNDERROTADOR(Todos), & \text{en caso contrario} \end{cases}$$

y $Pref = PREFERENCIA(\mathcal{D}, \mathcal{N})$. ■

Recuerde que un argumento podría aparecer más de una vez en el mismo árbol de dialéctica (en diferentes líneas de argumentación). Además, por las restricciones de las líneas de argumentación, podría suceder que un argumento \mathcal{D} que derrota a otro argumento \mathcal{N} en una línea no figure como derrotador para \mathcal{N} en otra. Seleccionar y enviar un derrotador a partir de un conjunto de no enviados es razonable para contemplar este tipo de situaciones donde las restricciones de las líneas de argumentación impiden el uso de ciertos derrotadores. Sin embargo, como también podría darse el caso que múltiples preguntas de negador generadas secuencialmente por la regla de transición rc_5 tengan un único derrotador en común como respuesta, en algunos casos el agente experto podría necesitar enviar un derrotador más de una vez.

Si el consultante está comprometido a lograr su meta, deberá adoptar el derrotador y la preferencia del experto. Dicho derrotador es un nuevo argumento a favor ya que defiende a otro argumento enviado por el experto de la derrota de un negador.

Definición 3.9 (Adopción de derrotador) Dado un agente consultante $\mathbf{C} = (ID, \Pi_{\mathbf{C}}, \Delta_{\mathbf{C}}, >_{\mathbf{C}})$, dos argumentos \mathcal{D} y \mathcal{N} , y $Pref \in \{\text{mejor, peor, no-relacionados}\}$, y sea \mathcal{S} el subargumento de \mathcal{N} en conflicto con \mathcal{D} , la *adopción* de \mathcal{D} como derrotador para \mathcal{N} por \mathbf{C} se define como $\text{ADPDERROTADOR}(\mathbf{C}, \mathcal{D}, \mathcal{N}, Pref) = \text{ADPPREFERENCIA}(\mathbf{C}', \mathcal{D}, \mathcal{N}, Pref)$, donde $\mathbf{C}' = \text{ADPARGUMENTO}(\mathbf{C}, \mathcal{D})$. ■

A continuación se presentará la regla de transición rc_6 , la cual especifica cómo evoluciona la sesión cuando el agente consultante adopta un derrotador enviado por el agente experto (ver δ a δ , y δ a β en la Figura 3.1).

$$rc_6 : \frac{\mathbb{F} = [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n] \wedge \mathbb{N} \in \mathbb{N} \wedge \text{DERROTADOR}(\mathcal{J}, \mathcal{N}, \mathbb{F}) = (\mathcal{D}, Pref) \wedge \text{ADPDERROTADOR}(\mathbf{C}, \mathcal{D}, \mathcal{N}, Pref) = \mathbf{C}'}{(\mathbf{C}, \mathbb{F}, \emptyset, \mathbb{N}) \rightarrow (\mathbf{C}', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \mathcal{D}], \emptyset, \mathbb{N} \setminus \{\mathcal{N}\})} \quad n \geq 1$$

Dado un estado en el que hay al menos una pregunta de negador \mathcal{N} pendiente y no hay preguntas de preferencia pendientes, la regla de transición rc_6 especifica cómo dicho estado evoluciona en uno nuevo en el que el agente experto envió un derrotador \mathcal{D} para \mathcal{N} y su preferencia $Pref \in \{\text{mejor, peor, no-relacionados}\}$ entre \mathcal{D} y \mathcal{N} , y tanto \mathcal{D} como $Pref$ fueron adoptados por el agente consultante \mathbf{C} , el cual evolucionó a \mathbf{C}' . Como \mathcal{D} es añadido al final de la secuencia de argumentos a favor \mathbb{F} , la próxima vez que la sesión evolucione a un estado $(\mathbf{C}'', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \mathcal{D}], \emptyset, \emptyset)$, el consultante procederá a realizar las preguntas de preferencia y de negador para el argumento \mathcal{D} . Por último, \mathcal{N} es eliminado del conjunto de preguntas de negador pendientes \mathbb{N} , lo cual implica que la regla de transición rc_6 será aplicada hasta que \mathbb{N} esté vacío.

Ejemplo 3.13 Continuando con el Ejemplo 3.12, considere el estado $e_6 = (\mathbf{C}^6, [\mathcal{B}_1], \emptyset, \{\mathcal{B}_2, \mathcal{B}_5\})$ en el que el agente *curly* tiene dos preguntas de negador pendientes (\mathcal{B}_2 y \mathcal{B}_5), y ya no tiene preguntas de preferencia por hacer. Recuerde que la Figura 3.7 (izquierda) ilustra el árbol de dialéctica para \mathcal{B}_1 del experto, y que sus preferencias son $\mathcal{B}_5 >_{\mathbf{E}} \mathcal{B}_1$, $\mathcal{B}_8 >_{\mathbf{E}} \mathcal{B}_2$, $\mathcal{B}_9 >_{\mathbf{E}} \mathcal{B}_5$, $\mathcal{B}_{10} >_{\mathbf{E}} \mathcal{B}_5$, $\mathcal{B}_{11} >_{\mathbf{E}} \mathcal{B}_{10}$ y $\mathcal{B}_1 >_{\mathbf{E}} \mathcal{B}_4$. Primero, *curly* hace la pregunta de negador \mathcal{B}_2 y el experto responde $\text{DERROTADOR}(\mathcal{B}_1, \mathcal{B}_2) = (\mathcal{B}_8, \text{mejor})$ dado que \mathcal{B}_8 es el único derrotador para \mathcal{B}_2 marcado como $\textcircled{\mathbf{U}}$. En consecuencia, \mathbf{C}^6 evoluciona a $\text{ADPDERROTADOR}(\mathbf{C}^6, \mathcal{B}_8, \mathcal{B}_2, \text{mejor}) = \mathbf{C}^7$ después de adoptar el argumento \mathcal{B}_8 y la preferencia $\mathcal{B}_8 >_{\mathbf{C}^6} \mathcal{B}_2$, y \mathcal{B}_8 se convierte en un derrotador propio para \mathcal{B}_2 . Luego, la regla

de transición rc_6 es aplicada y la sesión evoluciona al estado $e_7 = (C_7, [\mathcal{B}_1, \mathcal{B}_8], \emptyset, \{\mathcal{B}_5\})$. Después, el agente *curly* hace la pregunta de negador \mathcal{B}_5 y el experto responde $\text{DERROTADOR}(\mathcal{B}_1, \mathcal{B}_5) = (\mathcal{B}_9, \text{mejor})$ dado que \mathcal{B}_9 es el único derrotador para \mathcal{B}_5 marcado como \textcircled{U} . En consecuencia, C^7 evoluciona a $\text{ADPDERROTADOR}(C^7, \mathcal{B}_9, \mathcal{B}_5, \text{mejor}) = C^8$ después de adoptar el argumento \mathcal{B}_9 y la preferencia $\mathcal{B}_9 >_{C^7} \mathcal{B}_2$, y \mathcal{B}_9 se convierte en un derrotador propio para \mathcal{B}_5 . Luego, la regla de transición rc_6 es aplicada y la sesión evoluciona al estado $e_8 = (C_8, [\mathcal{B}_1, \mathcal{B}_8, \mathcal{B}_9], \emptyset, \emptyset)$. La Figura 3.8 ilustra el nuevo árbol de dialéctica enraizado con \mathcal{B}_1 del agente *curly*. Finalmente, como *curly* no tiene preguntas pendientes e $\text{INTROSPECCIÓN}(C^8, \mathcal{B}_1, \mathcal{B}_1) = \textcircled{U}$, la regla de transición rc_2 es aplicada y la sesión evoluciona al estado final $e_f = (C_8, [\mathcal{B}_1, \mathcal{B}_8, \mathcal{B}_9, \text{fin}], \emptyset, \emptyset)$ en el cual logró su meta de aceptar \mathcal{B}_1 . ■

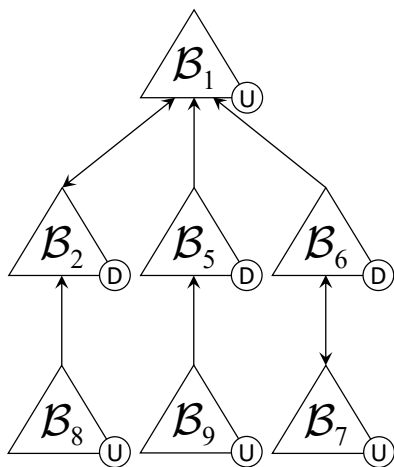


Figura 3.8: Árbol de dialéctica generado por el agente *curly* en el Ejemplo 3.13.

El ejemplo anterior muestra un escenario en el que el agente consultante logra su meta de marcar la justificación como \textcircled{U} después de adoptar los derrotadores necesarios (nuevos argumentos a favor) para dos negadores. Sin embargo, como se mostrará en el Ejemplo 3.14, el consultante podría tener negadores para los derrotadores recién adoptados, lo cual implicaría que la justificación sigue marcada como \textcircled{D} . Además, como el consultante adquiere nuevas creencias durante la sesión, podría suceder que construya un nuevo negador que causa que un argumento a favor previamente marcado como \textcircled{U} pase a estar marcado como \textcircled{D} nuevamente.

En estos casos, el consultante procederá a realizar las preguntas de preferencia y de negador correspondientes al último argumento a favor marcado como \textcircled{D} enviado por el

experto. Cuando este argumento pase a estar marcado como \textcircled{U} más adelante en la sesión, el próximo argumento a favor en esa línea de argumentación (arriba) tal vez pase a estar marcado como \textcircled{U} (si no tiene otros negadores), y así sucesivamente, como un efecto en cadena. Como las creencias de los agentes son finitas, el consultante eventualmente no podrá construir nuevos negadores para los derrotadores que envía el experto, y podrá lograr su meta de marcar la justificación como \textcircled{U} .

Ejemplo 3.14 Considere nuevamente la sesión entre el agente $\mathbf{E} = (\mathbf{exp}, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$ del Ejemplo 3.2 y el agente $\mathbf{L}^1 = (\mathbf{larry}, \Pi_{\mathbf{L}^1}, \Delta_{\mathbf{L}^1}, >_{\mathbf{L}^1})$ del Ejemplo 3.3. Recuerde que las preferencias del experto son $\mathcal{A}_4 >_{\mathbf{E}} \mathcal{A}_1$, $\mathcal{A}_4 >_{\mathbf{E}} \mathcal{A}_2$, $\mathcal{A}_4 >_{\mathbf{E}} \mathcal{A}_3$, $\mathcal{A}_5 >_{\mathbf{E}} \mathcal{A}_4$, $\mathcal{A}_5 >_{\mathbf{E}} \mathcal{A}_7$, $\mathcal{A}_6 >_{\mathbf{E}} \mathcal{A}_3$. La sesión comenzó en el estado inicial $e_0 = (\mathbf{L}^1, [], \emptyset, \emptyset)$, por la regla de transición rc_1 evolucionó a $e_1 = (\mathbf{L}^2, [\mathcal{A}_1], \emptyset, \emptyset)$, por rc_3 evolucionó a $e_2 = (\mathbf{L}^2, [\mathcal{A}_1], \{(\mathcal{A}_4, \mathcal{A}_1)\}, \emptyset)$, por rc_5 evolucionó a $e_3 = (\mathbf{L}^2, [\mathcal{A}_1], \emptyset, \{\mathcal{A}_4\})$, y por rc_6 evolucionó a $e_4 = (\mathbf{L}^3, [\mathcal{A}_1, \mathcal{A}_5], \emptyset, \emptyset)$. Hasta este punto, \mathbf{larry} adoptó la justificación \mathcal{A}_1 y un derrotador \mathcal{A}_5 y evolucionó a $\mathbf{L}^3 = (\mathbf{larry}, \Pi_{\mathbf{L}^3}, \Delta_{\mathbf{L}^3}, >_{\mathbf{L}^3})$ con $\Pi_{\mathbf{L}^3} = \{e, x, c, f\}$, $\Delta_{\mathbf{L}^3} = \{\sim a \prec d; d \prec e; \sim p \prec x; a \prec b; b \prec c; \sim d \prec p; p \prec f\}$, y $>_{\mathbf{L}^3} = \{(\mathcal{A}_4, \mathcal{A}_1), (\mathcal{A}_5, \mathcal{A}_4)\}$. Aunque \mathbf{larry} adoptó el argumento \mathcal{A}_5 que es un derrotador propio para el negador \mathcal{A}_4 , puede construir el argumento $\mathcal{A}_7 = \langle \sim p, \{\sim p \prec x\}, \{x\} \rangle$ que derrota por bloqueo a \mathcal{A}_5 . El argumento \mathcal{A}_7 también es un negador dado que está marcado como \textcircled{U} y está causando que los argumentos a favor \mathcal{A}_5 y \mathcal{A}_1 estén marcados como \textcircled{D} . Por lo tanto, la regla de transición rc_3 es aplicada nuevamente y la sesión evoluciona al estado $e_5 = (\mathbf{L}^3, [\mathcal{A}_1, \mathcal{A}_5], \{(\mathcal{A}_7, \mathcal{A}_5)\}, \emptyset)$ después de que fue generado el conjunto de preguntas de preferencia para \mathcal{A}_5 , *i. e.*, el último argumento a favor enviado por el experto que está marcado como \textcircled{D} . Luego, la regla de transición rc_4 es aplicada y la sesión evoluciona al estado $e_6 = (\mathbf{L}^4, [\mathcal{A}_1, \mathcal{A}_5], \emptyset, \emptyset)$ después de que el experto envió la preferencia **peor** y \mathbf{larry} evolucionó a $\mathbf{L}^4 = (\mathbf{larry}, \Pi_{\mathbf{L}^4}, \Delta_{\mathbf{L}^4}, >_{\mathbf{L}^4})$ con $\Pi_{\mathbf{L}^4} = \Pi_{\mathbf{L}^3}$, $\Delta_{\mathbf{L}^4} = \Delta_{\mathbf{L}^3}$, y $>_{\mathbf{L}^4} = \{(\mathcal{A}_4, \mathcal{A}_1), (\mathcal{A}_5, \mathcal{A}_4), (\mathcal{A}_5, \mathcal{A}_7)\}$. Como \mathcal{A}_7 ya no derrota al argumento a favor \mathcal{A}_5 , \mathcal{A}_5 ahora está marcado como \textcircled{U} y defiende a la justificación \mathcal{A}_1 de la derrota de \mathcal{A}_4 , *i. e.*, $\text{INTROSPECCIÓN}(\mathbf{L}^4, \mathcal{A}_1, \mathcal{A}_1) = \textcircled{U}$. Por lo tanto, la regla de transición rc_2 es aplicada y la sesión evoluciona al estado final $e_f = (\mathbf{L}^4, [\mathcal{A}_1, \mathcal{A}_5, \text{fin}], \emptyset, \emptyset)$ en el cual el agente \mathbf{larry} logró su meta de aceptar \mathcal{A}_1 . ■

3.2.5. Ejemplo integrador

A continuación se presentará un ejemplo integrador que formaliza la interacción entre los agentes del Ejemplo 3.1: un experto en el mercado bursátil y un inversionista que consulta si es conveniente comprar acciones de la empresa *Umbrella Corporation*.

Ejemplo 3.15 Considere la base de creencias del agente experto $\mathbf{E} = (\mathbf{exp}, \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}})$:

$$\Pi_{\mathbf{E}} = \left\{ \begin{array}{ll} \text{nuevo_producto}(\text{umbrella}) & \text{en_fusión}(\text{umbrella}, \text{aperture}) \\ \text{fuerte}(\text{aperture}) & \text{valor_estuvo_bajando}(\text{umbrella}) \\ \text{nuevo_dueño}(\text{umbrella}) & \text{problemas_legales}(\text{pollos_hermanos}) \\ \sim \text{en_fusión}(\text{cluckin!_bell}) & \sim \text{valor_estable}(\text{nuka_cola}) \end{array} \right\}$$

$$\Delta_{\mathbf{E}} = \left\{ \begin{array}{l} \text{comprar_acciones}(X) \prec \text{valor_aumentará}(X) \\ \text{valor_aumentará}(X) \prec \text{nuevo_producto}(X) \\ \sim \text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riegosa}(X) \prec \text{en_fusión}(X, Y) \\ \sim \text{riesgosa}(X) \prec \text{en_fusión}(X, Y), \text{fuerte}(Y) \\ \sim \text{valor_aumentará}(X) \prec \text{valor_estuvo_bajando}(X) \\ \text{vender_acciones}(X) \prec \text{problemas_legales}(X) \\ \sim \text{vender_acciones}(X) \prec \text{nueva}(X) \end{array} \right\}$$

Algunos de los argumentos que \mathbf{E} puede construir son:

$$\mathcal{M}_1 = \left\langle \begin{array}{l} \text{comprar_acciones}(\text{umbrella}), \\ \left\{ \begin{array}{l} \text{comprar_acciones}(X) \prec \text{valor_aumentará}(X) \\ \text{valor_aumentará}(X) \prec \text{nuevo_producto}(X) \end{array} \right\}, \\ \left\{ \text{nuevo_producto}(\text{umbrella}) \right\} \end{array} \right\rangle$$

$$\mathcal{M}_2 = \left\langle \begin{array}{l} \text{valor_aumentará}(\text{umbrella}), \\ \left\{ \text{valor_aumentará}(X) \prec \text{nuevo_producto}(X) \right\}, \\ \left\{ \text{nuevo_producto}(\text{umbrella}) \right\} \end{array} \right\rangle$$

$$\mathcal{M}_3 = \left\langle \begin{array}{l} \sim \text{comprar_acciones}(\text{umbrella}), \\ \left\{ \begin{array}{l} \sim \text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riegosa}(X) \prec \text{en_fusión}(X, Y) \end{array} \right\}, \\ \left\{ \text{en_fusión}(\text{umbrella}, \text{aperture}) \right\} \end{array} \right\rangle$$

$$\begin{aligned}
\mathcal{M}_4 &= \left\langle \begin{array}{l} \text{riegosa}(\text{umbrella}), \\ \left\{ \text{riegosa}(X) \prec \text{en_fusión}(X, Y) \right\}, \\ \left\{ \text{en_fusión}(\text{umbrella}, \text{aperture}) \right\} \end{array} \right\rangle \\
\mathcal{M}_5 &= \left\langle \begin{array}{l} \sim\text{riegosa}(\text{umbrella}), \\ \left\{ \sim\text{riegosa}(X) \prec \text{en_fusión}(X, Y), \text{fuerte}(Y) \right\}, \\ \left\{ \text{en_fusión}(\text{umbrella}, \text{aperture}) \right\} \\ \left\{ \text{fuerte}(\text{aperture}) \right\} \end{array} \right\rangle \\
\mathcal{M}_6 &= \left\langle \begin{array}{l} \sim\text{valor_aumentará}(\text{umbrella}), \\ \left\{ \sim\text{valor_aumentará}(X) \prec \text{valor_estuvo_bajando}(X) \right\}, \\ \left\{ \text{valor_estuvo_bajando}(\text{umbrella}) \right\} \end{array} \right\rangle
\end{aligned}$$

Note que \mathcal{M}_2 es subargumento de \mathcal{M}_1 y que \mathcal{M}_4 es subargumento de \mathcal{M}_3 . Considerando que $\mathcal{M}_2 >_{\mathbf{E}} \mathcal{M}_6$ y que $\mathcal{M}_5 >_{\mathbf{E}} \mathcal{M}_4$, la Figura 3.9 (izquierda) ilustra $\mathcal{T}_{\mathbf{E}}$, su árbol de dialéctica enraizado con \mathcal{M}_1 . Por otro lado, considere la base de creencias inicial del agente consultante $\mathbf{I}^1 = (\text{inv}, \Pi_{\mathbf{I}^1}, \Delta_{\mathbf{I}^1}, >_{\mathbf{I}^1})$:

$$\begin{aligned}
\Pi_{\mathbf{I}} &= \left\{ \text{en_fusión}(\text{umbrella}, \text{aperture}) \right\} \\
\Delta_{\mathbf{I}} &= \left\{ \begin{array}{l} \sim\text{comprar_acciones}(X) \prec \text{riegosa}(X) \\ \text{riegosa}(X) \prec \text{en_fusión}(X, Y) \end{array} \right\}
\end{aligned}$$

Note que los argumentos que el agente \mathbf{I}^1 puede construir son \mathcal{M}_3 y su subargumento \mathcal{M}_4 . Además, considere que $>_{\mathbf{I}^1} = \emptyset$. La sesión comienza en el estado inicial $e_0 = (\mathbf{I}^1, [], \emptyset, \emptyset)$ cuando \mathbf{I}^1 consulta $\text{comprar_acciones}(\text{umbrella})$ al agente \mathbf{E} . Dado que \mathbf{E} concluye $\text{comprar_acciones}(\text{umbrella})$ (\mathcal{M}_1 está marcado como \mathbf{U}) y que $\text{ADG}(\text{comprar_acciones}(\text{umbrella}), \Pi_{\mathbf{E}}, \Delta_{\mathbf{E}}, >_{\mathbf{E}}) = \{\mathcal{T}_{\mathbf{E}}\}$, el argumento que selecciona como justificación es $\text{SELECCIÓNJUSTIFICACIÓN}(\{\mathcal{T}_{\mathbf{E}}\}) = \mathcal{M}_1$ y por lo tanto $\text{JUSTIFICACIÓN}(\text{comprar_acciones}) = \mathcal{M}_1$. En consecuencia, la regla de transición rc_1 es aplicada, \mathbf{I}^1 evoluciona a $\text{ADPARGUMENTO}(\mathbf{I}^1, \mathcal{M}_1) = \mathbf{I}^2 = (\text{inv}, \Pi_{\mathbf{I}^2}, \Delta_{\mathbf{I}^2}, \emptyset)$ con

$$\Pi_{\mathbf{I}^2} = \left\{ \begin{array}{l} \text{en_fusión}(\text{umbrella}, \text{aperture}) \\ \text{nuevo_producto}(\text{umbrella}) \end{array} \right\}$$

$$\Delta_{I^2} = \left\{ \begin{array}{l} \sim\text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riesgosa}(X) \prec \text{en_fusión}(X, Y) \\ \text{comprar_acciones}(X) \prec \text{valor_aumentará}(X) \\ \text{valor_aumentará}(X) \prec \text{nuevo_producto}(X) \end{array} \right\}$$

y la sesión evoluciona al estado $e_1 = (I^2, [\mathcal{M}_1], \emptyset, \emptyset)$. El agente I^2 ahora puede construir los argumentos \mathcal{M}_1 , \mathcal{M}_3 y \mathcal{M}_4 , y su árbol de dialéctica enraizado con \mathcal{M}_1 está ilustrado en la Figura 3.9 (centro). Como la marca de la justificación no es \textcircled{U} a causa del negador \mathcal{M}_3 , la sesión con el agente experto debe continuar.

Dado que I^2 no tiene preguntas pendientes, que \mathcal{M}_1 es el último argumento a favor enviado por E, y que $\text{INTROSPECCIÓN}(I^2, \mathcal{M}_1, \mathcal{M}_1) = \textcircled{D}$, el agente I^2 procede a generar el conjunto de preguntas de preferencia $\text{PREGUNTASPREFERENCIA}(I^2, \mathcal{M}_1, \mathcal{M}_1) = \{(\mathcal{M}_3, \mathcal{M}_1)\}$. En consecuencia, la regla de transición rc_3 es aplicada y la sesión evoluciona al estado $e_2 = (I^2, [\mathcal{M}_1], \{(\mathcal{M}_3, \mathcal{M}_1)\}, \emptyset)$.

A continuación, el agente I^2 hace la pregunta de preferencia $(\mathcal{M}_3, \mathcal{M}_1)$ y el agente experto responde $\text{PREFERENCIA}(\mathcal{M}_3, \mathcal{M}_1) = \text{no-relacionados}$. Por lo tanto, I^2 evoluciona a $\text{ADPPREFERENCIA}(I^2, \mathcal{M}_3, \mathcal{M}_1, \text{no-relacionados}) = I^2$ dado que su conjunto de preferencias $>_{I^2} = \emptyset$ no cambia. Como $\text{INTROSPECCIÓN}(I^2, \mathcal{M}_1, \mathcal{M}_3) = \textcircled{U}$ (i. e., \mathcal{M}_3 continúa siendo un negador) la regla de transición rc_5 es aplicada, añadiendo \mathcal{M}_3 al conjunto de preguntas de negador pendientes. Luego, la sesión evoluciona al estado $e_3 = (I^2, [\mathcal{M}_1], \emptyset, \{\mathcal{M}_3\})$.

El agente I^2 procede con la pregunta de negador \mathcal{M}_3 a la que el experto responde $\text{DERROTADOR}(\mathcal{M}_1, \mathcal{M}_3) = (\mathcal{M}_5, \text{mejor})$ dado que \mathcal{M}_5 es el único derrotador para \mathcal{M}_3 marcado como \textcircled{U} . En consecuencia, al adoptar $(\mathcal{M}_5, \text{mejor})$, I^2 evoluciona a $\text{ADPDERROTADOR}(I^2, \mathcal{M}_5, \mathcal{M}_3, \text{mejor}) = I^3 = (\text{inv}, \Pi_{I^3}, \Delta_{I^3}, >_{I^3})$ con

$$\Pi_{I^3} = \left\{ \begin{array}{l} \text{en_fusión}(\text{umbrella}, \text{aperture}) \\ \text{nuevo_producto}(\text{umbrella}) \\ \text{fuerte}(\text{aperture}) \end{array} \right\}$$

$$\Delta_{I^3} = \left\{ \begin{array}{l} \sim\text{comprar_acciones}(X) \prec \text{riesgosa}(X) \\ \text{riesgosa}(X) \prec \text{en_fusión}(X, Y) \\ \text{comprar_acciones}(X) \prec \text{valor_aumentará}(X) \\ \text{valor_aumentará}(X) \prec \text{nuevo_producto}(X) \\ \sim\text{riesgosa}(X) \prec \text{en_fusión}(X, Y), \text{fuerte}(Y) \end{array} \right\}$$

y $>_{I^3} = \{\mathcal{M}_5 >_{I^3} \mathcal{M}_4\}$. Note que I^3 adoptó la preferencia $\mathcal{M}_5 >_{I^3} \mathcal{M}_4$ porque \mathcal{M}_4 es el subargumento de \mathcal{M}_3 en conflicto con \mathcal{M}_5 . Luego, la regla de transición rc_6 es aplicada y la sesión evoluciona al estado $e_4 = (I^3, [\mathcal{M}_1, \mathcal{M}_5], \emptyset, \emptyset)$. La Figura 3.9 (derecha) ilustra el árbol de dialéctica enraizado con \mathcal{M}_1 de I^3 .

Como I^3 no tiene preguntas pendientes, realiza una introspección para verificar la marca de la justificación, la cual resulta en $\text{INTROSPECCIÓN}(I^3, \mathcal{M}_1, \mathcal{M}_1) = \textcircled{U}$ ya que \mathcal{M}_1 no tiene más negadores. Finalmente, la regla de transición rc_2 es aplicada y la sesión evoluciona al estado final $e_5 = (I^3, [\mathcal{M}_1, \mathcal{M}_5, \text{fin}], \emptyset, \emptyset)$ en el cual el agente consultante logró su meta y concluye $\text{comprar_acciones}(\text{umbrella})$, la conclusión de \mathcal{M}_1 . ■

En esta sección se presentó el núcleo de la contribución de este capítulo, cuyos resultados formales serán mostrados en la sección siguiente. Más adelante, en la Sección 3.4, se definirán dos variantes para los operadores SELECCIÓNJUSTIFICACIÓN y SELECCIÓNDERROTADOR. Finalmente, en la Sección 3.5, se extenderá el sistema de transiciones para permitir al consultante rechazar la opinión del experto.

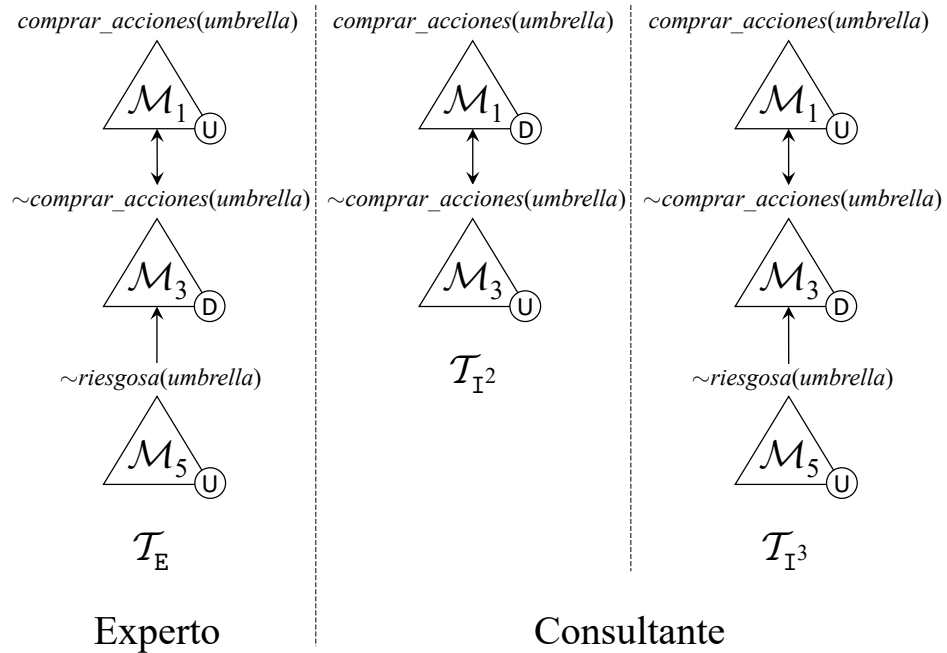


Figura 3.9: Árboles de dialéctica generados por los agentes del Ejemplo 3.15.

3.3. Resultados formales

En esta sección, se presentarán resultados formales que garantizan que el protocolo y estrategia de diálogo de consulta a experto definido por el sistema de transiciones $\Theta_C = \{rc_1, rc_2, rc_3, rc_4, rc_5, rc_6\}$ se comporta de manera sensata.

Recuerde que, de acuerdo a la Definición 2.28, un estado e_a es alcanzable desde otro estado e_0 mediante un sistema de transiciones Θ si existe una secuencia finita de reglas de transición de Θ que, aplicadas en ese orden, hacen que e_0 evolucione a e_a . En esta sección, cuando se hable de un estado de sesión alcanzable y no se mencione cuál es el estado e_0 , implícitamente se hará referencia al estado de sesión inicial $e_0 = (\mathcal{C}, [], \emptyset, \emptyset)$. Además, implícitamente se hará referencia al sistema de transiciones Θ_C .

Una de las propiedades más importantes del sistema de transiciones Θ_C es que toda sesión finaliza. Sin embargo, antes de probar esto se mostrará un resultado auxiliar. El Lema 3.1 garantiza que desde todo estado de sesión alcanzable (excepto un estado final) siempre hay una única regla de transición aplicable. Esto implica que la ejecución de una sesión se detendrá sólo si alcanza un estado final.

Lema 3.1 Sea $e_a \neq (\mathcal{C}, [\mathcal{F}_1, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset)$ un estado de sesión alcanzable, existe una y sólo una regla de transición de Θ_C aplicable desde e_a .

Demostración: ver Apéndice [pág. 135]. □

A continuación se presentará el antes mencionado Teorema 3.1, el cual garantiza que toda sesión eventualmente evoluciona a un estado final.

Teorema 3.1 Sea e_f un estado final, e_f es alcanzable.

Demostración: ver Apéndice [pág. 135]. □

El siguiente corolario es una consecuencia directa del Teorema 3.1 y garantiza que el sistemas de transiciones Θ_C es determinista.

Corolario 3.1 Sea e_f un estado final alcanzable, la secuencia de reglas de transición que hace evolucionar la sesión desde el estado inicial hasta e_f es única.

Demostración: ver Apéndice [pág. 136]. \square

Por último, recuerde que según [WK95] un diálogo de consulta a experto es exitoso si el consultante logra aceptar la respuesta del experto a su consulta. El Teorema 3.2 muestra que toda sesión desarrollada mediante el protocolo y estrategia de diálogo de consulta a experto definido en este capítulo es exitosa en el contexto de DeLP: una vez finalizada la sesión, el agente habrá aceptado la justificación que recibió del agente experto.

Teorema 3.2 Sea $e_f = (\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset)$ un estado de sesión final alcanzable, el agente consultante acepta \mathcal{J} .

Demostración: ver Apéndice [pág. 136]. \square

3.4. Minimización o reducción de la sesión

En esta sección se definirán dos variantes para los operadores SELECCIÓNJUSTIFICACIÓN y SELECCIÓNDERROTADOR utilizados en las Definiciones 3.2 y 3.8, respectivamente. Recuerde que el operador SELECCIÓNJUSTIFICACIÓN es utilizado por el agente experto para seleccionar una justificación \mathcal{J} a partir de un conjunto de árboles de dialéctica $\{\mathcal{T}_1, \dots, \mathcal{T}_n\}$, cuyas raíces son argumentos adecuados para enviar al agente consultante como justificación en respuesta a su consulta. Una vez que \mathcal{J} es enviada, el consultante comienza a hacer preguntas de preferencia y preguntas de negador hasta que logra aceptar \mathcal{J} . Siempre que el consultante realiza una pregunta de negador \mathcal{N} el operador SELECCIÓNDERROTADOR es utilizado por el experto para seleccionar un derrotador \mathcal{D} para \mathcal{N} marcado como $\textcircled{\mathbf{U}}$ en su árbol de dialéctica \mathcal{T} enraizado con \mathcal{J} .

Los operadores SELECCIÓNJUSTIFICACIÓN y SELECCIÓNDERROTADOR serán definidos con dos enfoques diferentes. En primer lugar, suponiendo el peor caso respecto a las creencias del consultante, serán definidos con el objetivo de que el experto pueda minimizar el tamaño del *árbol de dialéctica resultante del consultante* (ADRC), *i. e.*, el árbol de dialéctica que el consultante deberá construir hasta lograr aceptar la justificación. Intuitivamente, reducir el tamaño del ADRC también reduce la duración de una sesión en términos de la cantidad de preguntas de preferencia y preguntas de negador necesarias para que se cumpla la condición de finalización. Luego, suponiendo un escenario más realista, los operadores de selección serán definidos con el objetivo de ayudar al experto

a reducir el tamaño del ADRC considerando el contexto de la consulta y las creencias previas que el consultante expuso durante la sesión.

El tamaño del ADRC está ligado a cuáles *argumentos en contra* de \mathcal{J} en \mathcal{T} (el árbol de dialéctica del cual el experto seleccionó la justificación) puede construir el consultante. Esto es así porque aquellos negadores que no son realmente argumentos en contra (desde la perspectiva del experto) no estarán más en el árbol de dialéctica del consultante inmediatamente después de que el experto envíe *peor* en respuesta a la pregunta de preferencia correspondiente (ver Definición 3.6). Por el contrario, aquellos negadores que son realmente negadores permanecerán como tal, en el mejor caso, hasta que el consultante realice la pregunta de negador correspondiente y el experto envíe un derrotador \mathcal{D} (ver Definición 3.8). Luego, si el consultante puede construir un derrotador para \mathcal{D} , no sólo el negador permanecerá como tal sino que también el derrotador se convertirá en un nuevo negador con el que deberá lidiar.

En el peor caso, el consultante tendrá suficientes creencias para poder construir todos los argumentos en contra de $\{\mathcal{T}_1, \dots, \mathcal{T}_n\}$, *i. e.*, todos los árboles de dialéctica que el experto generará al buscar una justificación para la consulta. No obstante, aún si eso ocurre, el consultante no hará preguntas de preferencia y de negador para todo argumento en contra. En realidad, el consultante sólo lo hará para aquellos argumentos en contra que derroten a los argumentos a favor enviados por el experto. Por lo tanto, considerando que el experto debe enviar un único derrotador para cada pregunta de negador, seleccionarlos con perspicacia es la clave para minimizar el tamaño del ADRC.

Intuitivamente, si el agente experto quiere *minimizar* el tamaño del ADRC, el operador SELECCIÓNDERROTADOR siempre tiene que seleccionar el derrotador (argumento a favor) que minimizará la cantidad de argumentos en contra—construidos por él—con los que el agente consultante deberá lidiar al “explorar” las líneas de argumentación. Análogamente, el operador SELECCIÓNJUSTIFICACIÓN tiene que seleccionar la justificación del árbol de dialéctica que, considerando que SELECCIÓNDERROTADOR es optimal, minimizará la cantidad de argumentos en contra con los que el consultante deberá lidiar. Asumiendo el peor caso, optimizar es factible ya que el experto “sabe” todos los negadores que el consultante podría tener.

La Definición 3.10 detalla cómo asignar un *valor de selección del peor caso* (VSPC) a un nodo de un árbol de dialéctica. En el caso de un nodo etiquetado con un argumento a favor \mathcal{A} , este valor representa la cantidad de argumentos con los que el consultante tendrá

que lidia, suponiendo el peor caso, si el experto selecciona \mathcal{A} y continúa seleccionando los derrotadores con el menor VSPC.

Definición 3.10 (Valor de selección del peor caso) Dado un árbol de dialéctica \mathcal{T} y un nodo $\aleph_{\mathcal{A}}$ en \mathcal{T} , el *valor de selección del peor caso* de $\aleph_{\mathcal{A}}$ en \mathcal{T} se define como:

$$\text{VSPC}(\aleph_{\mathcal{A}}, \mathcal{T}) = \begin{cases} 0, & \text{si } \mathcal{A} \text{ es un argumento a favor en } \mathcal{T} \text{ e } \text{HIJOS}(\aleph_{\mathcal{A}}) = \emptyset \\ 1, & \text{si } \mathcal{A} \text{ es un argumento en contra en } \mathcal{T} \text{ e } \text{HIJOS}(\aleph_{\mathcal{A}}) = \emptyset \\ 1 + \text{MIN}(\{\text{VSPC}(\aleph', \mathcal{T}) \mid \aleph' \in \text{HIJOS}(\aleph_{\mathcal{A}})\}), & \text{si } \mathcal{A} \text{ es un argumento} \\ & \text{en contra en } \mathcal{T} \text{ e } \text{HIJOS}(\aleph_{\mathcal{A}}) \neq \emptyset \\ \sum_{\aleph' \in \text{HIJOS}(\aleph_{\mathcal{A}})} \text{VSPC}(\aleph', \mathcal{T}), & \text{si } \mathcal{A} \text{ es un argumento a favor en } \mathcal{T} \text{ e} \\ & \text{HIJOS}(\aleph_{\mathcal{A}}) \neq \emptyset \end{cases} \quad \blacksquare$$

El primer y segundo caso del operador asignan el VSPC correspondiente a las hojas del árbol de dialéctica. El tercer caso representa el hecho de que, siempre que el agente consultante hace una pregunta de negador, el agente experto puede enviar el derrotador que minimiza la cantidad de argumentos en contra que se conviertan en negadores para el consultante. Por último, el cuarto caso representa el hecho de que, suponiendo el peor escenario, el consultante podrá construir todo argumento en contra debajo del argumento a favor en cuestión. La Figura 3.10 ilustra dos ejemplos de árboles de dialéctica y el VSPC de cada uno de sus nodos.

A continuación, se definirán los operadores SELECCIÓNJUSTIFICACIÓN y SELECCIÓNDERROTADOR para permitir minimizar la cantidad de negadores con los que el agente consultante tendrá que lidiar suponiendo el peor caso respecto a sus creencias previas. En consecuencia, utilizar estos operadores implica una minimización de la duración de la sesión en términos de la cantidad de preguntas de preferencia y preguntas de negador que el consultante necesita realizar para lograr aceptar la justificación.

Definición 3.11 (Selección de justificación con criterio del peor caso) Dado un conjunto de árboles de dialéctica $\mathfrak{T} = \{\mathcal{T}_1, \dots, \mathcal{T}_n\} \neq \emptyset$, la *selección de la justificación* a partir de \mathfrak{T} se define como $\text{SELECCIÓNJUSTIFICACIÓN}(\mathfrak{T}) = \text{ETIQUETA}(\aleph)$ donde $\aleph = \text{RAÍZ}(\mathcal{T})$ tal que $\mathcal{T} \in \mathfrak{T}$ y no existe $\mathcal{T}' \in \mathfrak{T}$ tal que $\text{VSPC}(\text{RAÍZ}(\mathcal{T}), \mathcal{T}) > \text{VSPC}(\text{RAÍZ}(\mathcal{T}'), \mathcal{T}')$. \blacksquare

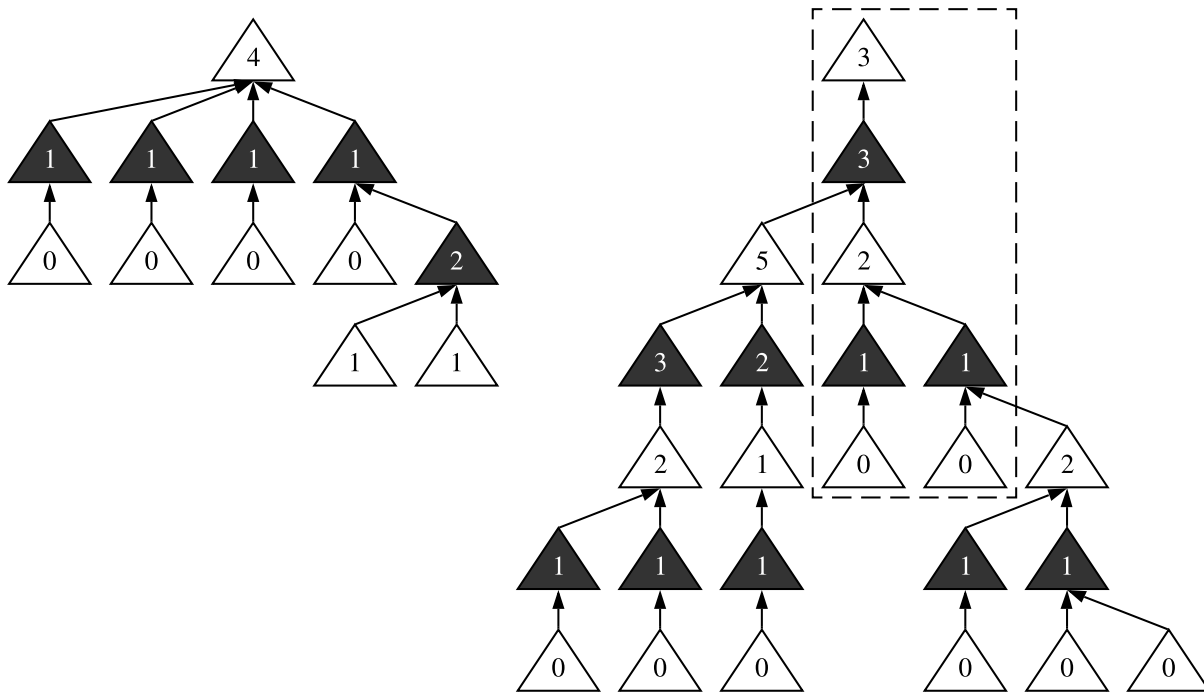


Figura 3.10: Dos árboles de dialéctica cuyos nodos contienen su VSPC. Los argumentos marcados como \textcircled{U} están ilustrados en blanco, y los marcados como \textcircled{D} en negro.

Ejemplo 3.16 Considere un agente experto que, dada una consulta, puede construir los árboles de dialéctica ilustrados en la Figura 3.10. En este caso, el operador SELECCIÓNJUSTIFICACIÓN seleccionaría la raíz del árbol de la derecha dado que tiene el menor VSPC. ■

Definición 3.12 (Selección de derrotador con criterio del peor caso) Dado un árbol de dialéctica \mathcal{T} y un conjunto de nodos $\mathfrak{N} = \{\mathfrak{N}_1, \dots, \mathfrak{N}_n\} \neq \emptyset$ en \mathcal{T} , la *selección del derrotador* a partir de \mathfrak{N} se define como $\text{SELECCIÓNDERROTADOR}(\mathfrak{N}) = \text{ETIQUETA}(\mathfrak{N})$ donde $\mathfrak{N} \in \mathfrak{N}$ y no existe $\mathfrak{N}' \in \mathfrak{N}$ tal que $\text{VSPC}(\mathfrak{N}, \mathcal{T}) > \text{VSPC}(\mathfrak{N}', \mathcal{T})$. ■

Ejemplo 3.17 Considere el árbol de dialéctica a la derecha de la Figura 3.10. El experto inicialmente selecciona y envía como justificación al consultante el argumento raíz ilustrado en blanco con un VSPC de 3 (*i. e.*, [blanco; 3]). Luego, si el consultante puede construir el argumento en contra [negro; 3] y hace la pregunta de negador correspondiente, el experto tendrá dos derrotadores para elegir: [blanco; 5] y [blanco; 2]. En este caso, el operador

SELECCIÓNDERROTADOR seleccionará y enviará [blanco; 2] (el de menor VSPC). Siguiendo el mismo criterio, si el consultante puede construir cualquiera de los dos argumentos en contra [negro; 1] y eventualmente hace la pregunta de negador correspondiente, el experto seleccionará y enviará el derrotador [blanco; 0]. En el peor caso, el ADRC será el subárbol de dialéctica dentro de las líneas punteadas. ■

Suponer que el consultante tiene suficientes creencias para construir todo argumento en contra del experto podría no ser realista dependiendo del dominio de aplicación. Sin embargo, está claro que al menos que el consultante envíe por adelantado todas sus creencias, es imposible que el experto prediga cuáles de sus argumentos en contra podrá construir el consultante. Por lo tanto, considerando que minimizar el tamaño del ADRC sin realizar suposiciones no es factible, también se definirán los operadores de selección para ayudar al experto a *reducir* el tamaño del ADRC considerando el *contexto* de la consulta y las creencias previas que el consultante expuso durante la sesión.

Como se mencionó al comienzo de la Sección 3.2, antes de que la sesión comience el consultante envía al experto toda información personal que representa un contexto para su consulta y que podría afectar la respuesta del experto. Como fue propuesto en [GS14], dicha información contextual puede ser temporalmente considerada por el experto al momento de generar los árboles de dialéctica para la consulta sin la necesidad de cambiar su base de creencias permanentemente. Una vez que la sesión finaliza, el contexto recibido desaparecerá de la base de base de creencias del experto y no será utilizada para responder las consultas de otros agentes. Referimos al lector interesado a [GS14], donde se definen diferentes operadores para integrar creencias temporalmente.

Siguiendo el propuesta de [GS14], el contexto de la consulta será representado por un programa lógico rebatible (Π_P, Δ_P) . Después de que los árboles de dialéctica para la consulta son generados, este programa será denominado *programa de creencias conocidas por el consultante* y será reutilizado para almacenar todos los hechos y reglas rebatibles que el experto sabe que el consultante cree. Siempre que el experto reciba una pregunta de preferencia $(\mathcal{N}, \mathcal{A})$ tal que $\mathcal{N} = \langle n, R_n, H_n \rangle$ y $\mathcal{A} = \langle a, R_a, H_a \rangle$, añadirá $H_n \cup H_a$ a Π_P , y $R_n \cup R_a$ a Δ_P . Además, siempre que el experto envíe un derrotador $\mathcal{D} = \langle d, R_d, H_d \rangle$ al consultante en respuesta a una pregunta de negador, añadirá H_d a Π_P , y R_d a Δ_P . Note que recibir un contexto, utilizarlo para generar los árboles de dialéctica, y mantener el programa de creencias conocidas por el consultante actualizado puede ser formalmente

incorporado en el protocolo y estrategia de diálogo modificando ligeramente la regla de transición rc_1 y las Definiciones 3.2, 3.6 y 3.8.

A pesar de que es imposible predecir con exactitud cuáles argumentos en contra el consultante podrá construir, el experto puede utilizar el programa de creencias conocidas por el consultante para seleccionar la justificación y los derrotadores basado en aquellos argumentos en contra que es *más probable* que el consultante pueda construir. Dado un nodo \aleph etiquetado con un argumento en contra \mathcal{A} , la probabilidad de que el consultante lo construya se puede estimar calculando su *razón de constructibilidad*. Esta razón representa la cantidad de elementos (hechos y reglas rebatibles) de \mathcal{A} que el consultante tendrá indudablemente en su base de creencias si la sesión llegase al punto en el que el experto le envió el argumento a favor que etiqueta al nodo padre de \aleph , dividido la cantidad total de elementos de \mathcal{A} . Por lo tanto, no sólo se debe considerar el programa de creencias conocidas por el consultante, sino también los *elementos ancestrales* de \aleph , *i. e.*, todos los hechos y reglas rebatibles que componen a los argumentos que etiquetan a los nodos ancestros de \aleph en la línea de argumentación correspondiente, como se define a continuación:

Definición 3.13 (Elementos ancestrales) Dado un árbol de dialéctica \mathcal{T} y un nodo \aleph_i en \mathcal{T} tal que $\text{ETIQUETA}(\aleph_i) = \langle c_i, R_i, H_i \rangle$, y sea Λ la línea de argumentación en \mathcal{T} tal que $\Lambda = [\aleph_1, \dots, \aleph_i, \dots, \aleph_n]$ con $1 \leq i \leq n$, los *elementos ancestrales* de \aleph_i se definen como $\text{ELEMENTOSANCESTRALES}(\aleph_i, \mathcal{T}) = (R_i \cup H_i) \cap (\{\bigcup R \cup H \mid \aleph \in^* [\aleph_1, \dots, \aleph_{i-1}] \text{ y } \text{ETIQUETA}(\aleph) = \langle c, R, H \rangle\})$. ■

Definición 3.14 (Razón de constructibilidad) Dado un árbol de dialéctica \mathcal{T} y un nodo \aleph en \mathcal{T} tal que $\text{ETIQUETA}(\aleph) = \langle c, R, H \rangle$, y sea (Π_P, Δ_P) el programa de creencias conocidas por el consultante, la *razón de constructibilidad* de \aleph se define como:

$$\text{RC}(\aleph, \mathcal{T}) = \frac{|(H \cap \Pi_P) \cup (R \cap \Delta_P) \cup \text{ELEMENTOSANCESTRALES}(\aleph, \mathcal{T})|}{|H \cup R|} \quad \blacksquare$$

Siguiendo una estrategia similar a el VSPC, la razón de constructibilidad puede ser utilizada para asignar un *valor de selección de constructibilidad* (VSC) a cada nodo de un árbol de dialéctica. En el caso de un nodo etiquetado con un argumento a favor \mathcal{A} , este valor representa la suma de las razones de constructibilidad de los argumentos en contra con los que el consultante tendrá que lidiar si el experto selecciona \mathcal{A} y continúa seleccionando los derrotadores con el menor VSC.

Definición 3.15 (Valor de selección de constructibilidad) Dado un árbol de dialéctica \mathcal{T} y un nodo $\aleph_{\mathcal{A}}$ en \mathcal{T} , el *valor de selección de constructibilidad* de $\aleph_{\mathcal{A}}$ en \mathcal{T} se define como:

$$\text{VSC}(\aleph_{\mathcal{A}}, \mathcal{T}) = \begin{cases} 0, & \text{si } \mathcal{A} \text{ es un argumento a favor en } \mathcal{T} \text{ e } \text{HIJOS}(\aleph_{\mathcal{A}}) = \emptyset \\ \text{RC}(\aleph_{\mathcal{A}}, \mathcal{T}), & \text{si } \mathcal{A} \text{ es un argumento en contra en } \mathcal{T} \text{ e } \text{HIJOS}(\aleph_{\mathcal{A}}) = \emptyset \\ \text{RC}(\aleph_{\mathcal{A}}, \mathcal{T}) + \text{MIN}(\{\text{VSC}(\aleph', \mathcal{T}) \mid \aleph' \in \text{HIJOS}(\aleph_{\mathcal{A}})\}), & \text{si } \mathcal{A} \text{ es un} \\ & \text{argumento en contra en } \mathcal{T} \text{ e } \text{HIJOS}(\aleph_{\mathcal{A}}) \neq \emptyset \\ \sum_{\aleph' \in \text{HIJOS}(\aleph_{\mathcal{A}})} \text{VSC}(\aleph', \mathcal{T}), & \text{si } \mathcal{A} \text{ es un argumento a favor en } \mathcal{T} \text{ e} \\ & \text{HIJOS}(\aleph_{\mathcal{A}}) \neq \emptyset \end{cases} \quad \blacksquare$$

Note que, a diferencia del operador VSPC, los argumentos en contra no incrementan el VSC total en 1 sino en el valor correspondiente a su razón de constructibilidad. Esta sección concluye con otra variante para los operadores SELECCIÓNJUSTIFICACIÓN y SELECCIÓNDERROTADOR, los cuales proveen guías de selección que contribuyen a reducir la cantidad de negadores con los que el consultante deberá lidiar durante la sesión, considerando el contexto de su consulta y las creencias previas que expuso durante la sesión. En general, utilizar estos operadores reducen la cantidad de preguntas de preferencia y preguntas de negador que el consultante necesita realizar para lograr su meta.

Definición 3.16 (Selección de justificación con criterio de constructibilidad)

Dado un conjunto de árboles de dialéctica $\mathfrak{T} = \{\mathcal{T}_1, \dots, \mathcal{T}_n\} \neq \emptyset$, la *selección de la justificación* a partir de \mathfrak{T} se define como $\text{SELECCIÓNJUSTIFICACIÓN}(\mathfrak{T}) = \text{ETIQUETA}(\aleph)$ donde $\aleph = \text{RAÍZ}(\mathcal{T})$ tal que $\mathcal{T} \in \mathfrak{T}$ y no existe $\mathcal{T}' \in \mathfrak{T}$ tal que $\text{VSC}(\text{RAÍZ}(\mathcal{T}), \mathcal{T}) > \text{VSPC}(\text{RAÍZ}(\mathcal{T}'), \mathcal{T}')$. \blacksquare

Definición 3.17 (Selección de derrotador con criterio de constructibilidad)

Dado un árbol de dialéctica \mathcal{T} y un conjunto de nodos $\aleph = \{\aleph_1, \dots, \aleph_n\} \neq \emptyset$ en \mathcal{T} , la *selección del derrotador* a partir de \aleph se define como $\text{SELECCIÓNDERROTADOR}(\aleph) = \text{ETIQUETA}(\aleph)$ donde $\aleph \in \aleph$ y no existe $\aleph' \in \aleph$ tal que $\text{VSC}(\aleph, \mathcal{T}) > \text{VSPC}(\aleph', \mathcal{T})$. \blacksquare

Dado que las Definiciones 3.16 y 3.17 utilizan el operador VSC, el cual se basa en lo que el experto sabe que el consultante cree, no hay garantía alguna de que haya una reducción efectiva en el tamaño del ADRC. A pesar de esto, claramente proveen una ventaja por sobre utilizar un método basado en selecciones aleatorias.

3.5. Extensión: rechazo de la opinión del experto

En el protocolo y estrategia de diálogo definido en la Sección 3.2 se supuso que el agente consultante está comprometido a lograr su meta de aceptar la justificación enviada por el agente experto. Para lograr su meta, además de incorporar todos los hechos y reglas rebatibles de los argumentos enviados por el experto, el consultante tal vez necesite actualizar sus preferencias y eliminar creencias de su base que son inválidas desde la perspectiva del experto. Aunque este protocolo y estrategia de diálogo fue pensado para dominios de aplicación en los que comprometerse a aceptar la opinión del experto es la mejor alternativa, puede haber situaciones en las que es mejor no hacerlo. En la Sección 3.6 se discutirá sobre la propuesta de [Wal10] para guiar la decisión sobre la aceptación de la opinión del experto.

Si la suposición de compromiso fuera relajada, el consultante podría discutir (con sí mismo u otros agentes) si aceptar o no la opinión del experto. A continuación se presentarán tres reglas de transición que pueden ser incorporadas en el sistema de transiciones. Con estas reglas adicionales el consultante puede optar por rechazar un argumento o una preferencia enviada por el experto, causando la terminación inmediata de la sesión.

$$rc_7 : \frac{\text{JUSTIFICACIÓN}(q) = \mathcal{J} \wedge \text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{J})}{(\mathcal{C}, [], \emptyset, \emptyset) \rightarrow (\mathcal{C}, [\mathcal{J}, \text{fin}], \emptyset, \emptyset)}$$

Con $\text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{J})$ se denota que el agente consultante $\mathcal{C} = (ID, \Pi_{\mathcal{C}}, \Delta_{\mathcal{C}}, >_{\mathcal{C}})$ decide rechazar el argumento $\mathcal{J} = \langle j, R, H \rangle$ enviado por el agente experto. Esto podría ocurrir, por ejemplo, si $h \in \Pi_{\mathcal{C}}$, $\bar{h} \in H$ y \mathcal{C} se rehúsa a eliminar h de $\Pi_{\mathcal{C}}$ para poder adoptar \mathcal{J} . La regla de transición rc_7 puede ser aplicada desde el mismo estado actual que rc_1 (el estado inicial), permitiendo a \mathcal{C} no adoptar la justificación.

$$rc_8 : \frac{(\mathcal{N}, \mathcal{A}) \in \mathbb{P} \wedge \text{PREFERENCIA}(\mathcal{N}, \mathcal{A}) = \text{Pref} \wedge \text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{N}, \mathcal{A}, \text{Pref})}{(\mathcal{C}, [\mathcal{F}_1, \dots, \mathcal{F}_n], \mathbb{P}, \mathbb{N}) \rightarrow (\mathcal{C}, [\mathcal{F}_1, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset)} \quad n \geq 1$$

Con $\text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{N}, \mathcal{A}, \text{Pref})$ se denota que \mathcal{C} decide rechazar la preferencia $\text{Pref} \in \{\text{mejor}, \text{peor}, \text{no-relacionados}\}$ entre \mathcal{N} y \mathcal{A} enviada por el experto. Esto podría ocurrir, por ejemplo, si \mathcal{C} se rehúsa a actualizar su conjunto de preferencias entre argumentos $>_{\mathcal{C}}$ en base a lo que el experto cree. La regla de transición rc_8 puede ser aplicada desde el mismo estado actual que rc_4 y rc_5 , permitiendo a \mathcal{C} no adoptar Pref .

$$rc_9 : \frac{\mathcal{N} \in \mathbb{N} \wedge \text{DERROTADOR}(\mathcal{J}, \mathcal{N}) = (\mathcal{D}, Pref) \wedge \text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{D}, \mathcal{N}, Pref)}{(\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \emptyset, \mathbb{N}) \rightarrow (\mathcal{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \mathcal{D}, \text{fin}], \emptyset, \emptyset)} \quad n \geq 1$$

Con $\text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{D}, \mathcal{N}, Pref)$ se denota que \mathcal{C} decide rechazar el derrotador \mathcal{D} o la preferencia $Pref \in \{\text{mejor, peor, no-relacionados}\}$ entre \mathcal{D} y \mathcal{N} . La regla de transición rc_9 puede ser aplicada desde el mismo estado actual que rc_6 , permitiendo a \mathcal{C} no adoptar \mathcal{D} como derrotador para el negador \mathcal{N} .

Note que después de que las reglas de transición rc_7 , rc_8 , o rc_9 son aplicadas, la sesión evoluciona a un estado final. Esto se debe a que si el agente consultante rechaza un argumento o una preferencia del agente experto ya no puede garantizarse que logrará su meta de marcar la justificación como \textcircled{U} . Por lo tanto, si cualquiera de estas tres reglas de transición es incorporada al sistema de transiciones $\Theta_C = \{rc_1, rc_2, rc_3, rc_4, rc_5, rc_6\}$ el Teorema 3.2 ya no es válido. Además, para poder ser incorporadas rc_7 , rc_8 , y rc_9 a Θ_C , las reglas de transición rc_1 , rc_4 , rc_5 y rc_6 deben ser modificadas como se indica a continuación:

- $\wedge \text{no}(\text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{J}))$ debe ser añadido a la condición de rc_1 .
- $\wedge \text{no}(\text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{N}, \mathcal{A}, Pref))$ debe ser añadido a la condición de rc_4 y a la condición de rc_5 .
- $\wedge \text{no}(\text{RECHAZOOPINIÓN}(\mathcal{C}, \mathcal{D}, \mathcal{N}, Pref))$ debe ser añadido a la condición de rc_6 .

3.6. Trabajo relacionado

Uno de los tipos de diálogos definidos por [WK95] es el *diálogo de búsqueda de información* (en inglés, *information-seeking*), en el cual uno de los participantes tiene cierto conocimiento o está en posición de saber algo, y el otro carece y necesita esa información. El objetivo del diálogo es la difusión de dicha información de un agente al otro.

A su vez, el diálogo de búsqueda de información se subclasifica en el *diálogo de consulta a experto* (en inglés, *expert consultation*), en el cual uno de los participantes es un experto en cierto dominio de conocimiento y el otro le consulta sobre un tema de ese campo del cual desconoce o conoce parcialmente. El objetivo de este subtipo diálogo es que el consultante obtenga la opinión calificada del experto sobre el tema en cuestión. Una consulta a experto se considera exitosa si, una vez finalizado el diálogo, el consultante está de acuerdo con la

opinión del experto. Otros subtipos del diálogo de búsqueda de información son el *diálogo didáctico* (en inglés, *didactic dialogue*), en el que uno de los agentes es un experto y busca transformar al otro agente en otro experto, y la *interrogación* (en inglés, *interrogation*), en el que un agente consultante busca conseguir del otro cierta evidencia sobre una asunto específico, la cual podría estar intentando ocultar.

Como se mencionó en el Capítulo 1, hay una diferencia sutil entre un *protocolo* de diálogo y una *estrategia* de diálogo: un protocolo permite construir diálogos coherentes, mientras que una estrategia permite construir diálogos con metas específicas [TG10, MTB⁺13, Thi14]. Precisamente, un protocolo de diálogo define un conjunto de *movimientos* o *locuciones* que los agentes pueden realizar, y determina cuándo tiene sentido realizarlos para que el diálogo resultante sea coherente. Por otro lado, una estrategia de diálogo es la que define su tipo, dado que determina el comportamiento que deberían tener los agentes para poder lograr su meta individual o la del diálogo.

A diferencia de los trabajos de la literatura en los que se define un protocolo de diálogo y una estrategia modular, el formalismo definido en este capítulo es un *protocolo y estrategia* de diálogo dado que ambos aspectos están fusionados en un conjunto de reglas de transición que determinan tanto la coherencia de los diálogos como el comportamiento de los agentes. Aunque es posible modificar el sistema de transiciones (como se mostró en la Sección 3.5), no es posible separar ambos aspectos sin una reestructuración completa. Si bien a simple vista esto parece una desventaja, definir un protocolo y estrategia específico para un tipo de diálogo permite concentrarse en los aspectos positivos que se desean lograr sin que la estrategia se vea limitada por el protocolo.

Si bien algunos de los protocolos de diálogo definidos en la literatura proveen una estrategia de búsqueda de información [PWA02, FT12, FT15], no se han desarrollado estrategias específicas de diálogo de consulta a experto como la que fue definida en este capítulo. La diferencia principal radica en que, al concentrarse en un tipo de diálogo más general, las estrategias de diálogo de búsqueda de información son más flexibles porque no buscan que el agente consultante esté de acuerdo con la información que recibe. Esto se debe a que, como [WK95] no define las condiciones bajo las cuales un diálogo de búsqueda de información es exitoso, el agente consultante no debe lograr ninguna meta en particular; puede sólo consultar y escuchar lo que el otro agente tiene para decir.

En la primera parte de esta sección se presentarán los protocolos y estrategias de diálogo de búsqueda de información de la literatura. Luego, se realizará un análisis com-

parativo entre ellos y el protocolo y estrategia de diálogo de consulta a experto definido en este capítulo (el cual será denotado [CAP3]). Todas las características comparadas serán sintetizadas en la tabla de la Figura 3.11

En [PWA02] se define un protocolo de diálogo para dos agentes que representan su conocimiento con fórmulas de la lógica proposicional y razonan utilizando el sistema argumentativo propuesto en [AC98]. El protocolo de diálogo está determinado por un conjunto de movimientos que los agentes pueden realizar (*e. g.*, *afirmar*, *aceptar* y *desafiar* (en inglés, *assert*, *accept* y *challenge*, respectivamente)), cada una con sus precondiciones y sus efectos. Los autores también definen una estrategia de diálogo de búsqueda de información mediante el cual un agente A pregunta sobre la validez de una proposición p y otro agente B busca darle una respuesta. Las proposiciones que un agente puede afirmar están determinadas por su *actitud de afirmación* (en inglés, *assertion attitude*):

- *Confiado* (en inglés, *confident*): Puede afirmar cualquier proposición p para la cual puede construir un argumento.
- *Cuidadoso* (en inglés, *thoughtful*): Puede afirmar cualquier proposición p para la cual puede construir un argumento aceptable.

Por otro lado, las proposiciones que un agente puede aceptar están determinadas por su *actitud de aceptación* (en inglés, *acceptance attitude*):

- *Crédulo* (en inglés, *credulous*): Puede aceptar una proposición p si está respaldada por un argumento.
- *Cauto* (en inglés, *cautious*): Puede aceptar una proposición p si está respaldada por un argumento y no es capaz de construir un argumento más fuerte para $\neg p$.
- *Escéptico* (en inglés, *skeptical*): Puede aceptar una proposición p si está respaldada por un argumento aceptable.

El diálogo de búsqueda de información comienza con una pregunta de A sobre p , para la cual B afirma p , $\neg p$ o \mathcal{U} dependiendo del contenido de su base de creencias y de su actitud de afirmación (confiado o cuidadoso). \mathcal{U} implica que por alguna razón el agente no puede dar una respuesta, terminando así el diálogo. Luego, A acepta la respuesta de B (si su actitud de aceptación se lo permite) o la desafía para que B muestre explícitamente el argumento que concluye esa proposición. Este proceso se repite para cada fórmula proposicional de dicho argumento.

Si bien los autores prueban que todo diálogo termina independientemente de las actitudes de los agentes, si el consultante es escéptico o cauto podría no lograr aceptar la proposición afirmada por el otro agente sin importar lo que diga.

En [FT11] se define un protocolo de diálogo para dos agentes que utilizan *ABA* (del inglés, *Assumption-Based Argumentation*) [DKT09, Ton14] como formalismo de representación de conocimiento y razonamiento, *i. e.*, cada agente está representado por un *framework ABA*. El protocolo de diálogo está determinado por un conjunto de *funciones de movimiento legal* (en inglés, *legal-move functions*), las cuales permiten a los agentes compartir reglas, suposiciones, y sus contrarios a través de *declaraciones* (en inglés, *utterances*) con el objetivo de construir conjuntamente frameworks ABA, y determinar la aceptabilidad de las suposiciones exploradas durante el diálogo.

En [FT12, FT15] se definen dos estrategias de diálogo de búsqueda de información para el protocolo de diálogo de [FT11]. En estas estrategias, un agente a_1 propone un tema χ y otro agente a_2 declara información de relevancia sobre χ . Se supone que a_1 no contribuye información, y que a_2 está interesado en dar información a favor de χ , pero no en contra. Además, se definen dos subtipos de diálogos de búsqueda de información, cada uno con su propia estrategia:

- *IS-Type I*: El agente a_2 puede construir al menos un argumento para χ que a_1 no puede construir, y el objetivo es generar un diálogo cuyo framework resultante contenga todo argumento para χ construido por a_2 .
- *IS-Type II*: El agente a_2 puede construir al menos un argumento para χ , el agente a_1 no puede construir ningún argumento para χ , y el objetivo es generar un diálogo cuyo framework resultante contenga sólo un argumento para χ construido por a_2 .

El comportamiento de los agentes es especificado a través de *funciones de movimiento de estrategia* (en inglés, *strategy-move functions*). Los autores prueban que los diálogos de tipo IS-Type I son sensatos sólo si a_1 utiliza una función de movimiento de estrategia ϕ_p *de sólo pasar* (en inglés, *pass*), la cual implica que a_1 puede iniciar el diálogo pero no puede declarar ningún tipo de información. Además, prueban que los diálogos de tipo IS-Type I son completos sólo si a_2 utiliza una función de movimiento de estrategia ϕ_{nh} *directa y sin ataques* (en inglés, *non-attack-thorough*), la cual implica que a_2 haga todas las declaraciones legales que pueda que no involucren contrarios. Por último, prueban que los diálogos de tipo IS-Type II son completos sólo si a_1 utiliza ϕ_p y a_2 utiliza ϕ_{nh} . Nada afirman sobre la sensatez de los diálogos de tipo IS-Type II.

La tabla de la Figura 3.11 compara las características del protocolo y estrategia de diálogo de consulta a experto propuesto en este capítulo ([CAP3]) con aquellas de los protocolos y estrategias de diálogo de búsqueda de información de [PWA02] y [FT12, FT15]. En particular, las primeras seis filas son características deseables, mientras que las últimas dos podrían ser beneficiosas o no dependiendo del dominio de aplicación.

	[PWA02]	[FT12] [FT15]	[CAP3] (Capítulo 3)
Éxito garantizado (consulta a experto)	N/A	N/A	✓
Hechos	×	×	✓
Reglas estrictas	×	×	×
Preferencias entre argumentos	✓	×	✓
Preferencias entre argumentos propias	×	N/A	✓
Preferencias entre argumentos cambian	×	N/A	✓
Consultante interactúa con experto	✓	×	✓
Efectiviza adquisición de creencias	×	×	✓

Figura 3.11: Comparación de las características de los protocolos y estrategias de diálogo de búsqueda de información presentes en la literatura con aquellas del protocolo y estrategia de diálogo de consulta a experto definido de este capítulo.

La característica *éxito garantizado* no aplica a [PWA02] ni a [FT12, FT15] dado que, como se mencionó anteriormente, [WK95] no define la noción de éxito para los diálogos de búsqueda de información. Si bien en [FT12, FT15] se realizan pruebas formales sobre la sensatez y completitud de la información expuesta por el agente consultado, el agente consultante no tiene la necesidad de aceptarla. Algo similar sucede en [PWA02], ya que los autores prueban que si el agente consultante tiene una actitud de aceptación escéptica o cauta, podría no lograr aceptar la proposición afirmada por el agente consultado sin importar lo que diga.

Respecto a la representación de información estricta, ni [PWA02] ni [FT12, FT15] permiten a los agentes tener hechos o reglas estrictas en sus bases de creencias, lo cual es

una consecuencia directa de los formalismos de representación de conocimiento utilizados. Por lo tanto, el consultante no puede tener información estricta en contradicción con las creencias del experto sobre el tema consultado.

Si bien en [CAP3] se permite a los agentes tener sus propios hechos, se optó por utilizar DeLP sin reglas estrictas. El tratamiento de reglas estrictas requiere de un análisis similar al de revisión de creencias en lógica proposicional, donde para actualizar Π incorporando una creencia α deben eliminarse todas las pruebas estrictas de $\bar{\alpha}$. Este tema ya fue estudiado y analizado en [FKIS02] y está fuera del foco de esta tesis..

Respecto a las preferencias entre argumentos, [PWA02] también las modela mediante una relación de orden pero, a diferencia de [CAP3], dicha relación es global a los agentes e inmodificable. Por otro lado, [FT12, FT15] no considera preferencias entre argumentos.

Por último, es importante comparar dos aspectos esenciales del funcionamiento de las estrategias. Primero, a diferencia de las demás, en [FT12, FT15] el agente consultante no interactúa con el agente consultado una vez comenzado el diálogo. Segundo, ni en [PWA02] ni en [FT12, FT15] el protocolo efectiviza la adquisición de creencias por parte del agente consultante. En otras palabras, la información comunicada por el agente consultado queda almacenada en una estructura auxiliar: el *commitment store* en el caso de [PWA02] y el framework ABA \mathcal{F}_δ en el caso de [FT12, FT15]. Dichos protocolos no especifican qué es lo que hace el agente consultante con esa información una vez finalizado el diálogo.

Antes de finalizar esta sección se discutirá sobre las diferencias entre [CAP3], revisión de creencias y *frameworks de argumentación* (AFs, por su nombre en inglés *abstract frameworks*). Recuerde que, siempre que el agente consultante recibe un argumento del agente experto, el operador ADPARGUMENTO elimina los hechos de la base de creencias del consultante que estén en contradicción con alguno de los hechos utilizados por el argumento (ver Definición 3.3) de manera análoga a lo que ocurre en revisión de creencias priorizada. Además, como DeLP permite la derivación de literales contradictorios, a diferencia de la revisión de creencias tradicional el operador ADPARGUMENTO añade toda regla rebatible del argumento a la base de creencias del consultante. Esto permite al consultante no sólo construir el argumento recibido, sino también combinar sus creencias previas con la información recién adquirida para construir nuevos argumentos. Dado que el consultante podría tener negadores para la justificación, la cual debe marcar como \textcircled{U} para lograr su meta, las técnicas de revisión de creencias no son suficientes. Después de

adoptar la justificación, el consultante también podría necesitar actualizar sus preferencias entre argumentos o incluso adoptar más argumentos del experto.

Considerando que incorporar argumentos nuevos o cambiar sus preferencias altera cómo sus argumentos se derrotan entre sí, podría verse una semejanza entre [CAP3] y los trabajos de revisión de frameworks de argumentación presentes en la literatura [MRF⁺10, CMKMM14, DHP14, SR16, DHL⁺18]. Estos trabajos se centran en la dinámica de un AF inicial al añadir o eliminar un argumento o una relación de ataque, devolviendo un nuevo AF o un conjunto de AFs como resultado. Sin embargo, [CAP3] difiere tanto conceptual como metodológicamente. No es deseable que el consultante logre su meta de aceptar la justificación en un único paso revisando sus creencias previas sin realizar preguntas adicionales al experto ya que esto podría implicar la eliminación innecesaria de piezas de información que, desde la perspectiva del experto, son válidas. En su lugar, el consultante mantendrá la comunicación con el experto haciendo preguntas adicionales para realizar cambios informados (considerando su opinión calificada) hasta lograr su meta.

3.7. Conclusiones

En este capítulo se presentó un protocolo y estrategia de diálogo de consulta a experto, el cual involucra dos agentes que utilizan DeLP como formalismo de representación de conocimiento y razonamiento: un agente experto en un cierto dominio de conocimiento y otro agente consultante que carece de esa cualidad. La interacción entre ambos agentes ocurre durante una sesión, la cual comienza cuando el agente consultante realiza una consulta al agente experto con el objetivo de adquirir información sobre un tema de dicho dominio, el cual desconoce o conoce parcialmente. En respuesta, si el agente experto tiene suficientes creencias para concluir el literal de la consulta o su complemento, elige un argumento que garantiza el literal correspondiente para enviar al otro agente como justificación a su consulta.

El consultante considera al otro agente un experto en el tema y, por lo tanto, su meta es aceptar la justificación que recibió en respuesta a su consulta. Por lo tanto, como podría tener argumentos que derrotan a la justificación, tal vez necesite realizar preguntas de preferencia y preguntas de negador al experto para que éste le envíe sus preferencias entre los argumentos involucrados o argumentos a favor de la justificación, respectivamente. Sin embargo, el consultante podría tener derrotadores para dichos argumentos a favor,

derrotadores para nuevos argumentos a favor, y así sucesivamente, creando un proceso iterativo de preguntas y respuestas que refinan sus creencias sobre la consulta adoptando información relevante del experto. Durante este proceso, el experto envía sólo la información necesaria para lograr la meta y el consultante evita la eliminación innecesaria de creencias que son válidas desde la perspectiva del experto.

El protocolo y estrategia de diálogo de consulta a experto que se definió en este capítulo incluye dos mejoras respecto a los trabajos similares presentes en la literatura. Primero, se consideró que el consultante podría tener hechos en contradicción con las creencias del experto sobre el tema consultado. Segundo, se consideró que ambos agentes podrían tener diferentes preferencias entre argumentos. Por lo tanto, si el consultante está comprometido a lograr su meta de aceptar la justificación, tal vez necesite actualizar sus preferencias y eliminar hechos de su base que son inválidas desde la perspectiva del experto.

Este protocolo y estrategia de diálogo fue definido utilizando un sistema de transiciones con el objetivo formalizar la interacción entre los agentes y luego probar sus propiedades. En particular, considerando la noción de éxito provista por [WK95] para los diálogos de consulta a experto, se probó que todo diálogo desarrollado mediante el protocolo y estrategia definido en este capítulo es exitoso: una vez finalizada una sesión, el agente consultante habrá logrado su meta de aceptar la justificación que recibió del agente experto en respuesta a su consulta.

Luego, se definieron dos variantes para los operadores SELECCIÓNJUSTIFICACIÓN y SELECCIÓNDERROTADOR. La primera supone el peor caso respecto a las creencias del agente consultante y permite minimizar la duración de la sesión en términos de la cantidad de preguntas de preferencia y preguntas de negador necesarias para que se cumpla la condición de finalización. La segunda supone un escenario más realista respecto a sus creencias y ayuda a reducir la duración de la sesión considerando el contexto de la consulta y las creencias previas que expuso el consultante.

Finalmente, aunque este protocolo y estrategia de diálogo fue pensado para dominios de aplicación en los que comprometerse a aceptar la opinión del agente experto es la mejor alternativa, se extendió el sistema de transiciones para permitir al agente consultante rechazarla, relajando la suposición de que está comprometido a lograr su meta.

Capítulo 4

Protocolo y estrategia de diálogo de indagación

Como se mencionó anteriormente, en [WK95] se define *diálogo* como un framework normativo que comprende el intercambio de argumentos entre agentes, los cuales pueden tener objetivos individuales pero razonan juntos para lograr una meta colectiva. Otro de los tipos de diálogo colaborativos de la clasificación propuesta en dicho trabajo es el diálogo de *indagación* (en inglés, *inquiry*). En este tipo de diálogo, un grupo de agentes busca llegar a un acuerdo general sobre la respuesta a una pregunta. Esto se realiza mediante un proceso acumulativo de argumentos que finaliza cuando los participantes llegan a una conclusión que, desde su punto de vista, es indudable. Es necesario que los participantes sean imparciales, y que contribuyan todas las conclusiones intermedias y premisas correspondientes. Según [WK95], un diálogo de indagación es exitoso si se logra llegar a un acuerdo general sobre el tema en discusión.

En este capítulo se presentará un protocolo y estrategia de diálogo de indagación en el que, al igual que en el Capítulo 3, los agentes utilizan DeLP como formalismo de representación de conocimiento y razonamiento. El diálogo comienza cuando se propone un argumento sobre el cual múltiples agentes desean llegar a un acuerdo general, llamado *argumento de indagación*, y finaliza cuando ninguno tiene algo más para contribuir. A diferencia del diálogo de consulta a experto tratado en el Capítulo 3, en un diálogo de indagación no hay participantes más calificados que otros en el dominio de conocimiento en cuestión.

Dado que los participantes del diálogo están en un ambiente colaborativo, deberán contribuir toda la información que dispongan que sea relevante. Los agentes pueden contribuir al diálogo derrotando al argumento de indagación con contra-argumentos, derrotando a esos contra-argumentos con otros contra-argumentos, y así sucesivamente. Siempre que un argumento es compartido, los participantes adquieren dicha información y la combinan con sus creencias previas, permitiéndoles potencialmente construir nuevos argumentos para contribuir. Además, los agentes pueden rechazar argumentos que consideran inválidos (dando una razón adecuada) u objetar hechos en contradicción con sus creencias. La meta de los participantes no es ganar una discusión o convencer a los otros de algo, sino contribuir imparcialmente todas sus creencias relevantes hasta que se llegue a un acuerdo general sobre el argumento de indagación.

El protocolo y estrategia de diálogo de indagación se definirá formalmente utilizando un sistema de transiciones. Éste mostrará cómo el diálogo debe evolucionar desde el estado inicial en el que se propone un argumento de indagación hasta un estado final en el que los participantes lograron llegar a un acuerdo. Al final de este capítulo se probará formalmente que todo diálogo desarrollado mediante este protocolo y estrategia termina exitosamente.

4.1. Introducción y motivación

El protocolo y estrategia de diálogo de indagación que se propone en esta tesis incluye cuatro mejoras respecto a los trabajos similares presentes en la literatura. Primero, se considera que los agentes podrían tener información estricta en contradicción con las creencias de los demás. Por lo tanto, cuando la información estricta entra en conflicto, los participantes deben llegar a un consenso sobre cuál pieza de información prevalece y deben actualizar sus creencias de acuerdo al consenso. Segundo, el diálogo no se limita a sólo dos agentes. Tercero, los participantes pueden optar por abandonar el diálogo si no están dispuestos a seguir el protocolo, *e. g.*, cuando no quieren actualizar sus creencias luego de un consenso o cuando no quieren compartir cierta pieza de información. Por último, nuevos agentes pueden unirse libremente al diálogo incluso cuando ya ha comenzado.

Una solución *naïve* al problema de modelar un protocolo y estrategia de diálogo de indagación sería que todos los participantes unan sus bases de creencias. Sin embargo, como se explicó en la Sección 3.1 del capítulo anterior, esta no es una solución sensata ni factible porque podría implicar la violación de la privacidad de los agentes, podría ser

computacionalmente impracticable, y la unión de las bases probablemente tendría muchas contradicciones cuya relevancia está fuera del dominio de conocimiento del argumento sobre el cual los participantes están discutiendo. Ignorar estas contradicciones daría lugar a resultados y conclusiones indeseadas, pero resolverlas sería una pérdida innecesaria de tiempo de computo. Otra solución sería que todos los participantes unan sólo parte de sus bases de creencias. Sin embargo, es imposible predecir con exactitud cuáles piezas de información serán útiles en la indagación sin realizar un análisis exhaustivo previo de todas las creencias de los participantes, lo cual sería contraproducente en términos de costo computacional y violación de la privacidad. Por lo tanto, para modelar un protocolo y estrategia de diálogo de indagación es necesaria una solución más adecuada. El siguiente ejemplo presenta la intuición de la contribución de este capítulo:

Ejemplo 4.1 Considere un videojuego en el que tres agentes caracterizan un equipo de mineros—*peter*, *egon* y *raymond*—e indagan sobre un argumento para darle mineral al jugador, quien necesita dicho recurso. La Figura 4.1 ilustra una árbol—estructuralmente idéntico a un árbol de dialéctica—que evolucionará a lo largo de este ejemplo en base al diálogo entre los agentes. Las creencias iniciales del agente $P = (\text{peter}, \Pi_P, \Delta_P, >)$ son:

$$\Pi_P = \left\{ \begin{array}{l} \text{ubicación}(\text{mineral}, \text{cavernas}) \\ \text{peligro}(\text{derrumbe}, \text{cavernas}) \\ \sim \text{difícil_encontrar}(\text{mineral}) \end{array} \right\}$$

$$\Delta_P = \left\{ \begin{array}{l} \sim \text{dar}(R, A) \prec \text{necesita}(R, A), \text{valioso}(R) \\ \text{valioso}(R) \prec \text{ubicación}(R, U), \text{peligrosas}(U) \\ \text{peligrosas}(U) \prec \text{peligro}(P, U) \\ \sim \text{valioso}(R) \prec \text{mucho_en_inventario}(R) \end{array} \right\}$$

Las creencias iniciales del agente $E = (\text{egon}, \Pi_E, \Delta_E, >)$ son:

$$\Pi_E = \left\{ \begin{array}{l} \text{peligro}(\text{derrumbe}, \text{cavernas}) \\ \text{enemigo}(\text{jugador}) \\ \text{equipamiento_seguridad}(\text{derrumbe}) \end{array} \right\}$$

$$\Delta_E = \left\{ \begin{array}{l} \sim \text{dar}(R, A) \prec \text{necesita}(R, A), \text{enemigo}(A) \\ \sim \text{peligrosas}(U) \prec \text{peligro}(P, U), \text{equipamiento_seguridad}(P) \end{array} \right\}$$

Por último, las creencias iniciales del agente $R = (\text{raymond}, \Pi_R, \Delta_R, >)$ son:

$$\Pi_R = \left\{ \begin{array}{l} \text{ubicación}(\text{mineral}, \text{cavernas}) \\ \text{escasas}(\text{cavernas}) \\ \sim \text{enemigo}(\text{jugador}) \end{array} \right\}$$

$$\Delta_R = \left\{ \begin{array}{l} \text{valioso}(R) \prec \text{difícil_encontrar}(R) \\ \text{difícil_encontrar}(R) \prec \text{ubicación}(R, U), \text{ escasas}(U) \end{array} \right\}$$

El diálogo comienza con el argumento de indagación “*deberíamos darle mineral al jugador ya que lo necesita*”.

$$\mathcal{D}_1 = \left\langle \begin{array}{l} \text{dar}(\text{mineral}, \text{jugador}), \\ \left\{ \text{dar}(R, A) \prec \text{necesita}(R, A) \right\}, \\ \left\{ \text{necesita}(\text{mineral}, \text{jugador}) \right\} \end{array} \right\rangle$$

Las reglas rebatibles y hechos del argumento de indagación son adquiridos por los agentes, y **peter** contra-argumenta “*el mineral parece ser un recurso valioso ya que se encuentra en cavernas, las cuales son peligrosas por los derrumbes; basados en esto, no deberíamos darle mineral al jugador aunque lo necesite*” (ver 1 en la Figura 4.1).

$$\mathcal{D}_2 = \left\langle \begin{array}{l} \sim \text{dar}(\text{mineral}, \text{jugador}), \\ \left\{ \begin{array}{l} \sim \text{dar}(R, A) \prec \text{necesita}(R, A), \text{ valioso}(R) \\ \text{valioso}(R) \prec \text{ubicación}(R, U), \text{ peligrosas}(U) \\ \text{peligrosas}(U) \prec \text{peligro}(P, U) \end{array} \right\}, \\ \left\{ \text{peligro}(\text{derrumbe}, \text{cavernas}) \right\} \end{array} \right\rangle$$

egon y **raymond** proceden a adquirir la información de este argumento, y **egon** contra-argumenta “*las cavernas no suelen ser peligrosas para nosotros ya que tenemos el equipamiento necesario para evitar y protegernos de los derrumbes*” (ver 2 en la Figura 4.1).

$$\mathcal{P} = \left\langle \begin{array}{l} \sim \text{peligrosas}(\text{cavernas}), \\ \left\{ \begin{array}{l} \sim \text{peligrosas}(U) \prec \text{peligro}(P, U), \text{ equipamiento_seguridad}(P) \\ \text{peligro}(\text{derrumbe}, \text{cavernas}) \\ \text{equipamiento_seguridad}(\text{derrumbe}) \end{array} \right\} \end{array} \right\rangle$$

Al igual que antes, los demás agentes proceden a adquirir la información del argumento compartido. Luego, **raymond** da un contra-argumento diferente para el argumento de indagación: “*el mineral parece ser difícil de encontrar ya que sólo está en cavernas, las*

cuales son escasas; esto hace que el mineral sea un recurso valioso y, basados en esto, no deberíamos darle al jugador aunque lo necesite” (ver [3] en la Figura 4.1).

$$\mathcal{D}_3 = \left\langle \begin{array}{l} \sim dar(\text{mineral}, \text{jugador}), \\ \left\{ \begin{array}{l} \sim dar(R, A) \prec necesita(R, A), \text{valioso}(R) \\ \text{valioso}(R) \prec \text{difícil_encontrar}(R) \\ \text{difícil_encontrar}(R) \prec \text{ubicación}(R, U), \text{escasas}(U) \end{array} \right\}, \\ \left\{ \begin{array}{l} \text{ubicación}(\text{mineral}, \text{cavernas}) \\ \text{escasas}(\text{cavernas}) \end{array} \right\} \end{array} \right\rangle$$

Una vez más, las reglas rebatibles y hechos de este argumento son adquiridos por los demás agentes. A pesar de que **peter** previamente argumentó en contra de darle mineral al jugador, él considera que el argumento de **raymond** es inválido porque afirma estar seguro de que el mineral no es un recurso difícil de encontrar. Esto sucede porque uno de los pasos de inferencia intermedios del argumento \mathcal{D}_3 es el literal $\text{difícil_encontrar}(\text{mineral})$, el cual es derivado utilizando una regla rebatible, pero **peter** cree en el hecho $\sim \text{difícil_encontrar}(\text{mineral})$, el cual representa información estricta. Por lo tanto, el argumento de **raymond** es rechazado (ver [4] en la Figura 4.1) y los agentes adquieren el hecho $\sim \text{difícil_encontrar}(\text{mineral})$ de **peter**.

Continuando con el diálogo, **egon** da un tercer contra-argumento para el argumento de indagación: “no deberíamos darle mineral al jugador aunque lo necesite porque es un enemigo” (ver [5] en la Figura 4.1).

$$\mathcal{D}_4 = \left\langle \begin{array}{l} \sim dar(\text{mineral}, \text{jugador}), \\ \left\{ \begin{array}{l} \sim dar(R, A) \prec necesita(R, A), \text{enemigo}(A) \end{array} \right\}, \\ \left\{ \begin{array}{l} necesita(\text{mineral}, \text{jugador}) \\ \text{enemigo}(\text{jugador}) \end{array} \right\} \end{array} \right\rangle$$

Al momento de adquirir la información de este argumento, **raymond** no puede incorporar el hecho $\text{enemigo}(\text{jugador})$ porque él cree que $\sim \text{enemigo}(\text{jugador})$. Por lo tanto, **raymond** procede a objetar el hecho $\text{enemigo}(\text{jugador})$, causando que todos los agentes del equipo deban llegar a un consenso sobre cuál de los hechos en contradicción es aceptado y cuál es rechazado. El equipo decide colaborativamente que el jugador no es un enemigo y, en consecuencia, el argumento de **egon** es invalidado por sustentarse en una premisa que fue considerada incorrecta (ver [6] en Figura 4.1).

Como nadie tiene algo más para contribuir al diálogo, la indagación finaliza y todo el equipo termina teniendo razones para estar de acuerdo con darle mineral al jugador. ■

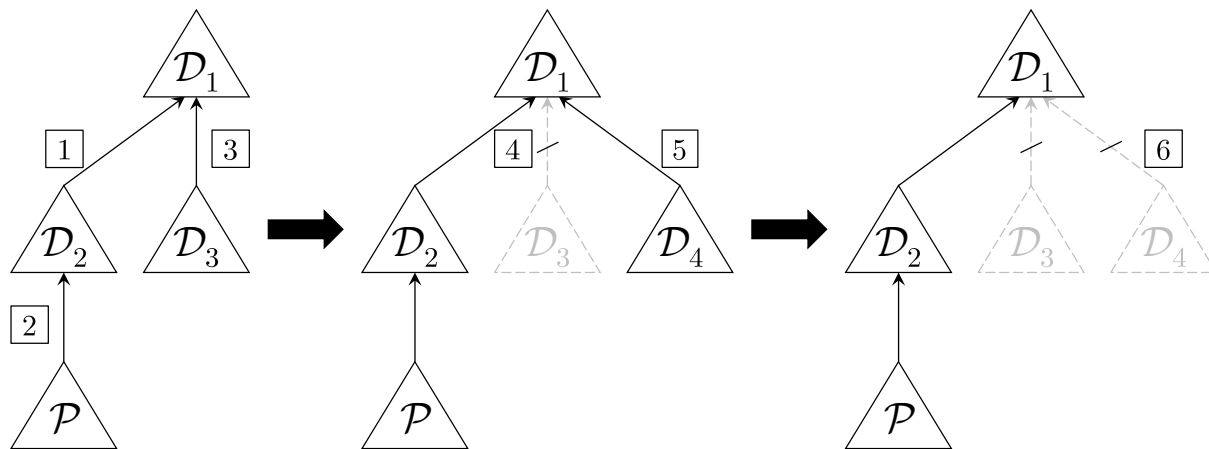


Figura 4.1: Evolución del diálogo de indagación del Ejemplo 4.1.

Según [WK95], un diálogo de indagación es exitoso si los participantes logran llegar a un acuerdo general sobre el tema en discusión. Considerando esto, se probará que todo diálogo desarrollado mediante el protocolo y estrategia definido en este capítulo es exitoso en el contexto del formalismo de representación de conocimiento y razonamiento utilizado por los agentes: una vez finalizado el diálogo, todos los participantes estarán de acuerdo sobre el estado de derrota del argumento de indagación. Este capítulo está organizado de la siguiente manera:

- En la Sección 4.2 se definirá formalmente el protocolo y estrategia de diálogo de indagación utilizando un sistema de transiciones.
- En la Sección 4.3 se demostrará que todo diálogo de indagación desarrollado mediante este protocolo y estrategia finaliza de manera exitosa.
- En la Sección 4.4 se discutirán posibles extensiones para el formalismo.
- En la Sección 4.5 se compararán empíricamente diferentes comportamientos de agente a la hora de realizar movimientos.
- En la Sección 4.6 se presentará el trabajo relacionado y se lo comparará con la propuesta de este capítulo.
- En la Sección 4.7 se presentarán las conclusiones.

Como se mencionó anteriormente, la propuesta y los resultados formales que se presentarán en este capítulo fueron publicados en la revista *Expert Systems with Applications* vol. 116 [AGG19a].

4.2. Sesión de indagación

Al igual que en el Capítulo 3, en el protocolo y estrategia de diálogo de indagación que se presentará en esta sección, los agentes serán representados mediante la 4-tupla $(ID, \Pi, \Delta, >)$ definida en el Capítulo 2. Toda la interacción entre los agentes será llamada *sesión de indagación*, la cual comienza cuando se propone un argumento sobre el cual múltiples agentes desean llegar a un acuerdo general, llamado *argumento de indagación*, y finaliza cuando ninguno tiene algo más para contribuir. Se supondrá que:

1. Durante una sesión las creencias de los participantes sólo cambian a través de la interacción con sus pares.
2. La relación de preferencias entre argumentos de todos los agentes es la misma.
3. El argumento de indagación ya fue propuesto antes de comenzar la sesión.

La suposición 1 implica que los participantes de una sesión no pueden estar en más de una sesión simultáneamente. Sin embargo, como se verá más adelante, nada impide que los agentes abandonen la sesión, adquieran información de fuentes externas, y vuelvan a unirse. La suposición 2 se debe a que, a diferencia del diálogo de consulta a experto tratado en el Capítulo 3, en un diálogo de indagación no hay agentes más calificados que otros en el dominio de conocimiento en cuestión. Respecto a la suposición 3, en la Sección 4.4 se discutirá sobre una extensión para que los agentes puedan indagar sobre un literal en lugar de un argumento y puedan proponer múltiples argumentos de indagación para ese literal durante la sesión.

Los agentes pueden contribuir a la sesión realizando distintos tipos de *movimientos*. En particular, pueden derrotar al argumento de indagación con contra-argumentos, derrotar esos contra-argumentos con otros contra-argumentos, rechazar los argumentos que consideran inválidos (dando una razón adecuada), y objetar hechos en contradicción con sus creencias. Los argumentos que son compartidos durante una sesión componen una estructura llamada *árbol de indagación*, la cual es conjuntamente actualizada cada vez que los agentes realizan movimientos.

Definición 4.1 (Árbol de indagación) Un *árbol de indagación* $\mathbb{I} = \langle \text{Nodos}, \text{Arcos} \rangle$ es un árbol tal que cada nodo $\aleph_{\mathcal{A}} \in \text{Nodos}$ está etiquetado con un argumento $\mathcal{A} = \langle c, \mathbf{R}, \mathbf{H} \rangle$, denotado $\text{ETIQUETA}(\aleph_{\mathcal{A}})$, cada arco $\varkappa \in \text{Arcos}$ desde un nodo $\aleph_{\mathcal{B}}$ hacia otro nodo $\aleph_{\mathcal{C}}$ representa que \mathcal{B} derrota a \mathcal{C} , y cumple las siguientes condiciones:

1. Si $\aleph_{\mathcal{B}} \in \text{HIJOS}(\aleph_{\mathcal{A}})$, entonces no existe $\aleph'_{\mathcal{B}} \neq \aleph_{\mathcal{B}}$ tal que $\aleph'_{\mathcal{B}} \in \text{HIJOS}(\aleph)$.
2. Si $h \in \mathbf{H}$, entonces no existe $\aleph' \in \text{Nodos}$ tal que $\text{ETIQUETA}(\aleph') = \langle c', \mathbf{R}', \mathbf{H}' \rangle$ y $\bar{h} \in \mathbf{H}$.
3. Toda rama en \mathbb{I} desde la raíz a un nodo hoja cumple con las condiciones de línea de argumentación aceptable (ver Definición 2.18).

El conjunto de hechos que forman parte de un árbol de indagación \mathbb{I} se define como $\text{EVIDENCIA}(\mathbb{I}) = \{ \bigcup \mathbf{H} \mid \text{ETIQUETA}(\aleph) = \langle c, \mathbf{R}, \mathbf{H} \rangle \text{ para todo } \aleph \in \text{Nodos} \}$. ■

Un árbol de indagación es estructuralmente igual a un árbol de dialéctica (ver Definición 2.20). La diferencia radica en que un árbol de dialéctica es construido exhaustivamente a partir de un único programa lógico rebatible, mientras que un árbol de indagación \mathbb{I} es construido progresiva y dinámicamente a partir de los argumentos que son compartidos por diferentes agentes (cada uno de los cuales tiene su propio programa lógico rebatible que no necesariamente compartirá en su totalidad). El árbol de indagación puede verse como una forma de construir un árbol de dialéctica de manera colaborativa y distribuida. Note que, de acuerdo a la condición 2, la evidencia de un árbol de indagación no puede ser contradictoria.

Al igual que en el Capítulo 3, una vez que una sesión ha comenzado, su estado actual es determinado por un *estado de sesión*.

Definición 4.2 (Estado de sesión de indagación) Un *estado de sesión de indagación* es una 4-tupla $(\mathbb{I}, \mathbb{Z}, \mathbb{R}, \mathbb{E})$ donde \mathbb{I} es un árbol de indagación, \mathbb{Z} es un conjunto de agentes, \mathbb{R} es una secuencia de secuencias de movimientos $\langle \text{ID}, \text{Movimiento} \rangle$ y marcas de espera ε , y $\mathbb{E} \subseteq \{ \text{ID} \mid (\text{ID}, \Pi_{\mathbf{z}}, \Delta_{\mathbf{z}}, >) \in \mathbb{Z} \}$ es un subconjunto de los identificadores de los agentes en \mathbb{Z} . ■

El segundo componente de un estado de sesión es el conjunto de agentes \mathbb{Z} que están indagando sobre la raíz de \mathbb{I} . Aunque los últimos dos componentes serán explicados en detalle más adelante, serán presentados intuitivamente a continuación: \mathbb{R} es una secuencia de *rondas* $[R_1, \dots, R_n]$ que ocurrieron durante la sesión, cuyo último elemento es llamado

ronda actual. Cada ronda R_i ($1 \leq i \leq n$) es una secuencia de *movimientos* de la forma $\langle ID, Movimiento \rangle$ que fueron realizados anteriormente, y de *marcas de espera* ε . Cada movimiento tiene un identificador ID asociado, el cual puede ser utilizado para identificar unívocamente al agente que lo realizó. Por último, \mathbb{E} es un conjunto de identificadores de agentes llamado *cola de espera*, la cual representa los agentes que están esperando para hacer un movimiento durante la ronda actual.

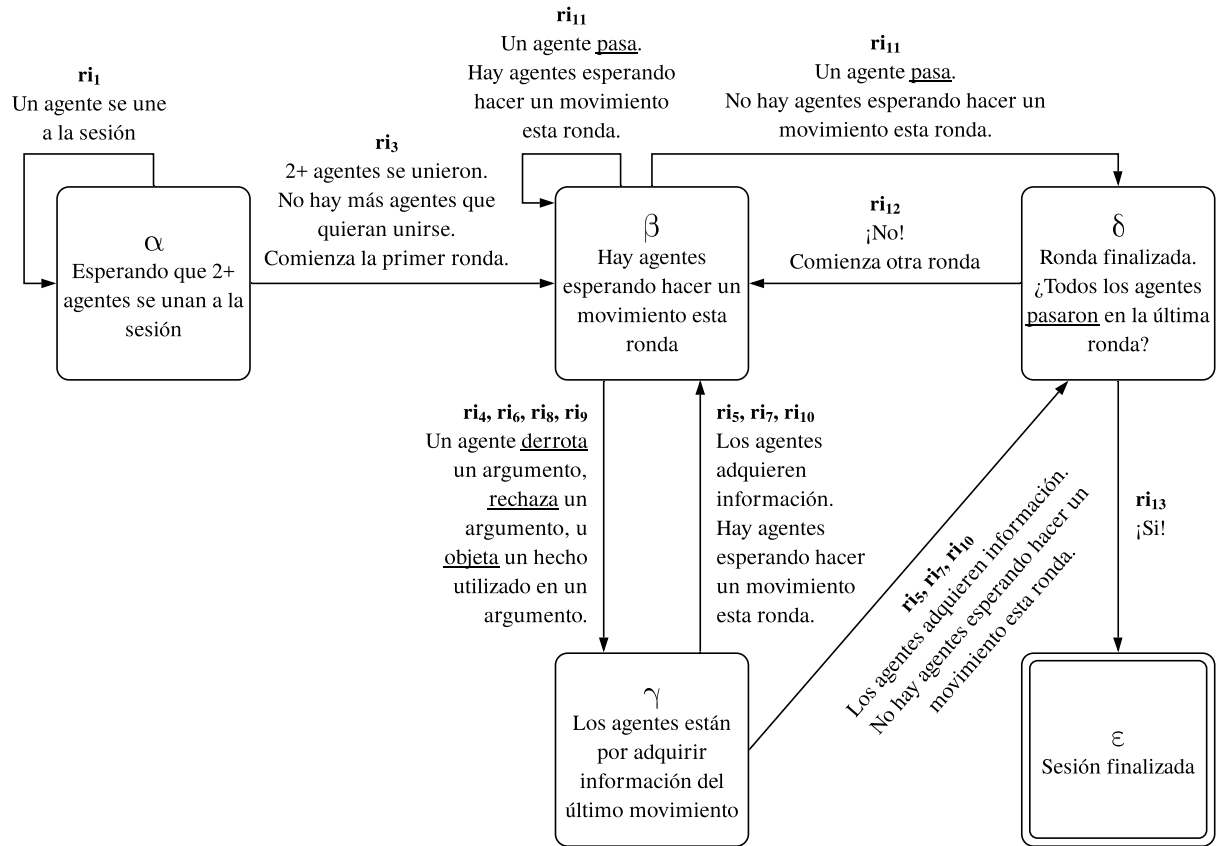


Figura 4.2: Bosquejo de la evolución de una sesión de indagación

En particular, serán distinguidos dos estados de sesión especiales: el *estado inicial* y los *estados finales*. El primer componente del estado inicial es un árbol de indagación \mathbb{I} cuya raíz (su único nodo) está etiquetada con el argumento de indagación, mientras que el resto de sus componentes están vacíos. Este estado es denotado

$$e_0 = (\mathbb{I}, \emptyset, [], \emptyset).$$

Por otro lado, el primer componente de un estado final es el árbol de indagación \mathbb{I}' resultante luego de todos los movimientos que realizaron los agentes. Su segundo componente

es el conjunto de agentes \mathbb{Z} que permaneció hasta el final de la sesión. Su tercer componente es la secuencia de rondas $[R_1, \dots, R_n, [\text{fin}]]$ donde R_i ($1 \leq i \leq n$) es la secuencia de movimientos y marcas de espera correspondientes la i -ésima ronda, y $[\text{fin}]$ es un terminador. Finalmente, su cuarto componente está vacío (*i. e.*, no hay agentes en la cola de espera). Este estado es denotado

$$e_f = (\mathbb{I}', \mathbb{Z}, [R_1, \dots, R_n, [\text{fin}]], \emptyset) \text{ con } 1 \leq n.$$

A lo largo de esta sección se hará referencia a la Figura 4.2, la cual ilustra un grafo dirigido que bosqueja el flujo de ejecución de una sesión. Los nodos—identificados con letras griegas—representan diferentes etapas de la interacción, mientras que los arcos dirigidos representan reglas de transición. Con el objetivo de probar las propiedades y de formalizar la interacción entre los agentes, el protocolo y estrategia de diálogo de indagación será definido mediante el sistema de transiciones $\Theta_I = \{ri_1, ri_2, ri_3, ri_4, ri_5, ri_6, ri_7, ri_8, ri_9, ri_{10}, ri_{11}, ri_{12}, ri_{13}\}$. Por ejemplo, la regla de transición ri_{13} especificará las condiciones necesarias para que una sesión alcance un estado final. En particular, las reglas de transición ri_1 y ri_2 son especiales porque pueden ser aplicadas en cualquier momento y siempre conducen a la misma etapa de la interacción. Dado que se asume un escenario colaborativo, los agentes estarán comprometidos a seguir el protocolo.

4.2.1. Adquirir información

Todo argumento compartido por un agente durante una sesión es *adquirido* por los demás participantes, permitiéndoles combinarlo con sus creencias previas y potencialmente construir nuevos argumentos para contribuir a la sesión. La adquisición de un argumento se define mediante el operador ADQARGUMENTO .

Definición 4.3 (Adquisición de argumento) Dado un agente $Z = (ID, \Pi_Z, \Delta_Z, >)$ y un argumento $\mathcal{A} = \langle c, R, H \rangle$, la *adquisición* de \mathcal{A} por parte de Z se define como $\text{ADQARGUMENTO}(Z, \mathcal{A}) = Z' = (ID, \Pi_{Z'}, \Delta_{Z'}, >)$, donde:

$$\Pi_{Z'} = \Pi_Z \cup \{h \mid h \in H \text{ y } \bar{h} \notin \Pi_Z\}$$

$$\Delta_{Z'} = \Delta_Z \cup R \quad \blacksquare$$

Dado $\text{ADQARGUMENTO}(Z, \mathcal{A}) = Z'$, note que Z y Z' tienen el mismo identificador. Por lo tanto, Z' no es un agente diferente sino el mismo con una base de creencias modificada. En otras palabras, Z *evoluciona* a Z' .

Ejemplo 4.2 Considere el agente $Z = (\text{ag}, \Pi_Z, \Delta_Z, >)$ con $\Pi_Z = \{y\}$ y $\Delta_Z = \{c \prec y; b \prec y, x\}$. Al adquirir el argumento $\mathcal{A} = \langle a, \{a \prec x\}, \{x\} \rangle$, el agente Z se convierte en $\text{ADQARGUMENTO}(Z, \mathcal{A}) = Z' = (\text{ag}, \Pi_{Z'}, \Delta_{Z'}, >)$ con $\Pi_{Z'} = \{y, x\}$ y $\Delta_{Z'} = \{c \prec y; b \prec y, x; a \prec x\}$. Note que Z' puede construir \mathcal{A} . Además, ahora puede construir el argumento $\mathcal{B} = \langle b, \{b \prec y, x\}, \{y, x\} \rangle$ por haber adquirido y combinado el hecho x con sus creencias previas. ■

Según la Definición 4.3, un agente que adquiere un argumento no incorpora los hechos que están en contradicción con los hechos de su base de creencia. Por lo tanto, es posible que no pueda construir dicho argumento inmediatamente después de adquirirlo, como se mostrará en el Ejemplo 4.3. Más adelante se explicará cómo proceden los agentes cuando esto ocurre en una sesión.

Ejemplo 4.3 Considere el agente $Z' = (\text{ag}, \Pi_{Z'}, \Delta_{Z'}, >)$ del Ejemplo 4.2. Al adquirir el argumento $\mathcal{D} = \langle d, \{d \prec \sim y, e; e \prec w\}, \{\sim y, w\} \rangle$ el agente Z' se convierte en $\text{ADQARGUMENTO}(Z', \mathcal{D}) = Z'' = (\text{ag}, \Pi_{Z''}, \Delta_{Z''}, >)$ con $\Pi_{Z''} = \{y, x, w\}$ y $\Delta_{Z''} = \{c \prec y; b \prec y, x; a \prec x; d \prec \sim y, e; e \prec w\}$. Note que Z'' no puede construir \mathcal{D} ya que, como cree en el hecho y , no adquirió $\sim y$. ■

4.2.2. Unirse y abandonar una sesión

Una vez que una sesión ha comenzado, los agentes son capaces de *unirse* libremente y en cualquier momento. Aquellos que se unan deberán adquirir todos los argumentos que forman parte del árbol de indagación para “ponerse al día” con lo que se ha acordado hasta el momento en la sesión. Tenga en cuenta que un agente que se une a la sesión antes de que comience la primer ronda sólo debe adquirir la raíz del árbol de indagación (*i. e.*, el argumento de indagación). La adquisición de todos los argumentos de un árbol de indagación se define mediante el operador recursivo ADQÁRBOL .

Definición 4.4 (Adquisición de árbol de indagación) Dado un agente Z y un árbol de indagación $\mathbb{I} = \langle \text{Nodos}, \text{Arcos} \rangle$, la *adquisición* de \mathbb{I} por Z se define como:

$$\text{ADQÁRBOL}(Z, \mathbb{I}) = \begin{cases} Z, & \text{si } \text{Nodos} = \emptyset \\ \text{ADQÁRBOL}(\text{ADQARGUMENTO}(Z, \mathcal{A}), \langle \text{Nodos}', \text{Arcos} \rangle) & \text{tal que} \\ \text{Nodos} = \mathfrak{R}_{\mathcal{A}} \cup \text{Nodos}', & \text{si } \text{Nodos} \neq \emptyset \end{cases}$$

■

Observe que, dado un agente Z y un árbol de indagación $\mathbb{I} = \langle \text{Nodos}, \text{Arcos} \rangle$, el operador ADQÁRBOL ejecuta el operador ADQARGUMENTO hasta que Z adquiera todos los argumentos que etiquetan los nodos en *Nodos*.

A continuación, se presentará la regla de transición ri_1 , la cual especifica cómo evoluciona la sesión cuando un agente quiere unirse a la sesión.

$$ri_1 : \frac{\text{existe } Z = (ID, \Pi_Z, \Delta_Z, >) \text{ tal que } \text{UNIRSESESION}(Z) \wedge \text{ADQÁRBOL}(Z, \mathbb{I}) = Z'}{(\mathbb{I}, Z, \mathbb{R}, \mathbb{E}) \rightarrow (\mathbb{I}, Z \cup \{Z'\}, \mathbb{R}, \mathbb{E} \cup \{ID\})}$$

Con $\text{UNIRSESESION}(Z)$ se denota que el agente Z quiere unirse a la sesión. Esta regla de transición no requiere que los componentes del estado actual satisfagan condicional alguna, permitiéndole a los agentes unirse a la sesión libremente y en cualquier momento. Dado un estado en el que existe un agente Z que quiere unirse a la sesión, la regla de transición ri_1 especifica cómo dicho estado evoluciona a uno nuevo en el que Z adquirió todos los argumentos en \mathbb{I} , evolucionó a Z' , y fue agregado al conjunto de participantes \mathbb{Z} . (e. g., ver α a α en la Figura 4.2). El identificador del agente ID es agregado a la cola de espera \mathbb{E} y, por lo tanto, podrá hacer un movimiento en la ronda actual.

Al igual que en el Capítulo 3, para llevar registro de la evolución de las bases de creencias de los agentes y del árbol de indagación se añadirán superíndices a las variables de agente y de árboles de indagación, respectivamente. Por ejemplo, $Z^5 = (\text{ag}, \Pi_{Z^5}, \Delta_{Z^5}, >)$ representa el agente **ag** en cierto punto de la sesión, y $Z^6 = (\text{ag}, \Pi_{Z^6}, \Delta_{Z^6}, >)$ representa ese mismo agente luego de que su base de creencias fue modificada.

Ejemplo 4.4 Considere el estado inicial de una sesión $e_0 = (\mathbb{I}^1, \emptyset, [], \emptyset)$ en el que el argumento de indagación es $\text{ETIQUETA}(\text{RAÍZ}(\mathbb{I}^1)) = \mathcal{A}_1 = \langle a, \{a \prec z\}, \{z\} \rangle$. Considere que los agentes $M^1 = (\text{moe}, \Pi_{M^1}, \Delta_{M^1}, >)$, $L^1 = (\text{larry}, \Pi_{L^1}, \Delta_{L^1}, >)$, $C^1 = (\text{curly}, \Pi_{C^1}, \Delta_{C^1}, >)$ y $S^1 = (\text{shemp}, \Pi_{S^1}, \Delta_{S^1}, >)$ quieren unirse a la sesión, en ese orden. La Figura 4.3 ilustra las bases de creencias de **moe**, **larry**, **curly** y **shemp** antes de unirse. La regla de transición ri_1 es aplicada y la sesión evoluciona al estado $e_1 = (\mathbb{I}^1, \{M^2\}, [], \{\text{moe}\})$ después de que M^1 se unió a la sesión, adquirió \mathbb{I}^1 (sólo \mathcal{A}_1), y evolucionó a M^2 . Luego, ri_1 es aplicada tres veces más y la sesión evoluciona al estado $e_4 = (\mathbb{I}^1, \{M^2, L^2, C^2, S^2\}, [], \{\text{moe}, \text{larry}, \text{curly}, \text{shemp}\})$. Note que los agentes podrían haberse unido en cualquier orden y el resultado sería el mismo. ■

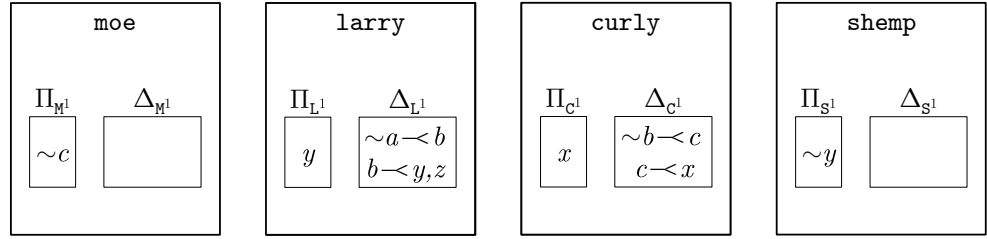
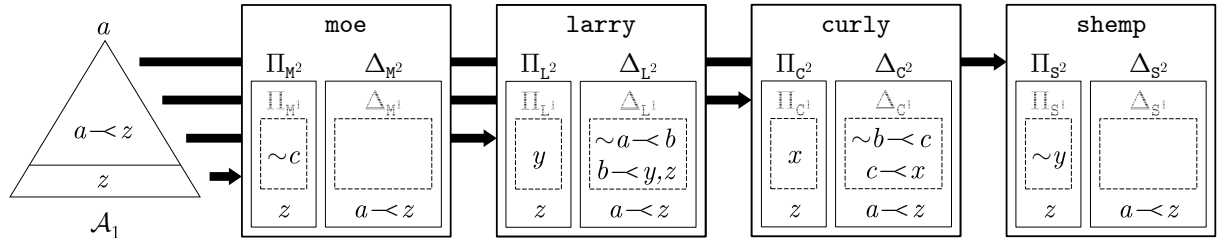


Figura 4.3: Bases de creencias de los agentes del Ejemplo 4.4 antes de unirse a la sesión.

Figura 4.4: Bases de creencias de los agentes del Ejemplo 4.4 después de unirse a la sesión y adquirir el argumento \mathcal{A}_1 .

Si un agente no quiere respetar el protocolo definido por las reglas de transición, puede abandonar la sesión en cualquier momento sin importar el estado actual. Por ejemplo, un agente podría negarse a incorporar o eliminar piezas de información de su base de creencias, o negarse a compartir alguna pieza de información considerada sensible y/o confidencial. A continuación, se presentará la regla de transición ri_2 , la cual especifica cómo evoluciona la sesión cuando un agente quiere abandonar la sesión.

$$ri_2 : \frac{\mathbf{Z} = (ID, \Pi_{\mathbf{Z}}, \Delta_{\mathbf{Z}}, >) \in \mathbb{Z} \wedge \text{ABANDONARSESION}(\mathbf{Z})}{(\mathbb{I}, \mathbb{Z}, \mathbb{R}, \mathbb{E}) \rightarrow (\mathbb{I}, \mathbb{Z} \setminus \{\mathbf{Z}\}, \mathbb{R}, \mathbb{E} \setminus \{ID\})}$$

Con $\text{ABANDONARSESION}(\mathbf{Z})$ se denota que el agente \mathbf{Z} quiere abandonar la sesión. Al igual que ri_1 , la regla de transición ri_2 no requiere que los componentes del estado actual satisfagan condición alguna. Esto permite a los agentes en \mathbb{Z} abandonar la sesión justo antes de tener que realizar algo a lo que se niegan. Un agente que abandona la sesión es eliminado de \mathbb{Z} y de la cola de espera \mathbb{E} .

En la Sección 4.3 se demostrará que toda sesión podrá evolucionar a un estado final incluso si todos los participantes la abandonan.

Una vez que al menos dos agentes se unieron a la sesión y no hay más agentes que quieran unirse en ese momento la primer ronda puede comenzar, como es especificado por la regla de transición ri_3 (ver α a β en la Figura 4.2).

$$ri_3 : \frac{|Z| \geq 2 \wedge \text{no existe } Z \text{ tal que UNIRSESESION}(Z)}{(\mathbb{I}, Z, [], \mathbb{E}) \rightarrow (\mathbb{I}, Z, [[\varepsilon]], \mathbb{E})}$$

Aunque la primera ronda sólo puede empezar si en ese momento no hay más agentes que quieran unirse a la sesión, la regla de transición ri_1 puede volver a aplicarse más tarde si es necesario. Por el contrario, la regla de transición ri_3 sólo se aplicará una vez ya que requiere que el tercer componente del estado actual esté vacío. Observe que una nueva secuencia para la primer ronda es creada e inicializada con una marca de espera ε . Siempre que el último elemento de la ronda actual sea una marca de espera ε y la cola de espera \mathbb{E} no esté vacía, será el turno de un agente en \mathbb{E} para realizar un *movimiento* en el árbol de indagación.

Ejemplo 4.5 Continuando con el Ejemplo 4.4, considere el estado de sesión $e_4 = (\mathbb{I}^1, \{\mathbb{M}^2, \mathbb{L}^2, \mathbb{C}^2, \mathbb{S}^2\}, [], \{\text{moe, larry, curly, shemp}\})$. Supongamos que ningún otro agente quiere unirse a la sesión. Luego, como hay al menos dos agentes en el segundo componente del estado de sesión, la primer ronda puede empezar. La regla de transición ri_3 es aplicada y la sesión evoluciona al estado $e_5 = (\mathbb{I}^1, \{\mathbb{M}^2, \mathbb{L}^2, \mathbb{C}^2, \mathbb{S}^2\}, [[\varepsilon]], \{\text{moe, larry, curly, shemp}\})$ en el que uno de los agentes en la cola de espera debe hacer un movimiento. ■

4.2.3. Movimientos en el árbol de indagación

Los posibles movimientos que puede realizar un agente durante su turno dependen tanto de su base de creencias como de la estructura del árbol de indagación. Un agente Z puede *derrotar* un argumento \mathcal{A} dando un argumento derrotador \mathcal{D} ; *rechazar* un argumento \mathcal{A} dando un argumento estricto \mathcal{C} que esté en contradicción a una regla rebatible de \mathcal{A} ; *rechazar* un argumento \mathcal{A} dando un argumento \mathcal{M} minimal respecto a \mathcal{A} ; *objetar* un hecho h que esté presente en la evidencia del árbol de indagación (si cree en \bar{h}); o *pasar* si no puede realizar otro movimiento. Un movimiento por un agente Z en un árbol de indagación \mathbb{I} se define mediante el operador MOVIMIENTO. Considere que el operador \curvearrowright

corresponde a la concatenación de secuencias y recuerde que, dada una línea de argumentación Λ y un nodo $\aleph \in^* \Lambda$, el operador `SEGMENTOSUPERIOR` devuelve la subsecuencia inicial de Λ que finaliza en \aleph .

Definición 4.5 (Movimiento) Dado un agente $Z = (ID, \Pi_Z, \Delta_Z, >)$ y un árbol de indagación $\mathbb{I} = \langle \text{Nodos}, \text{Arcos} \rangle$, un *movimiento* por Z en \mathbb{I} se define como:

$$\text{MOVIMIENTO}(\mathbb{I}, Z) = \left\{ \begin{array}{l} \text{derrotar}(\mathcal{A}, \mathcal{D}), \\ \text{rechazar}(\mathcal{A}, \mathcal{C}), \\ \text{rechazar}(\mathcal{A}, \mathcal{M}), \\ \text{objetar}(h, \bar{h}), \\ \text{pasar}, \end{array} \right. \left\{ \begin{array}{l} \text{si } Z \text{ puede construir un argumento } \mathcal{D} = \langle d, R, H \rangle \\ \text{que derrota a algún argumento } \mathcal{A} \text{ tal que} \\ \aleph_{\mathcal{A}} \in \text{Nodos}, \text{ y se cumple que } \aleph_{\mathcal{D}} \notin \text{HIJOS}(\aleph_{\mathcal{A}}), \\ \text{EVIDENCIA}(\mathbb{I}) \cup H \text{ no es contradictorio, y existe} \\ \text{una línea de argumentación } \Lambda \text{ en } \mathbb{I} \text{ con } \aleph_{\mathcal{A}} \in^* \Lambda \\ \text{tal que } \text{SEGMENTOSUPERIOR}(\Lambda, \aleph_{\mathcal{A}}) \frown [\aleph_{\mathcal{D}}] \text{ es} \\ \text{una línea de argumentación aceptable.} \\ \\ \text{si } Z \text{ puede construir un argumento estricto } \mathcal{C} = \langle h, \emptyset, \{h\} \rangle \\ \text{tal que } \bar{h} \prec p_1, \dots, p_n \in R \\ \text{para algún argumento } \mathcal{A} = \langle a, R, H \rangle \text{ tal que} \\ \aleph_{\mathcal{A}} \in \text{Nodos.} \\ \\ \text{si } Z \text{ puede construir un argumento} \\ \mathcal{M} = \langle a, R_2, H_2 \rangle \text{ tal que } R_2 \subset R \text{ para algún} \\ \text{argumento } \mathcal{A} = \langle a, R, H \rangle \text{ tal que } \aleph_{\mathcal{A}} \in \text{Nodos.} \\ \\ \text{si existe un hecho } h \in \text{EVIDENCIA}(\mathbb{I}) \text{ tal que} \\ \bar{h} \in \Pi_Z. \\ \\ \text{si no se cumple ninguna de las condiciones an-} \\ \text{teriores.} \quad \blacksquare \end{array} \right.$$

Observe que un argumento \mathcal{D} no puede **derrotar** a otro argumento \mathcal{A} si alguno de los hechos que contiene \mathcal{D} está en contradicción con la evidencia del árbol de indagación. Esto se debe a que, en un escenario argumentativo las conclusiones deberían deducirse a partir de premisas consistentes. Por lo tanto, si un agente quiere dar un argumento que contiene un hecho en contradicción con una pieza de evidencia del árbol de indagación, primero deberá **objetar** esa pieza de evidencia. De esta manera, el agente muestra la necesidad de llegar a un consenso sobre cuál de los hechos en contradicción debería ser aceptado y cuál

debería ser rechazado. Siempre que esto ocurra, el hecho del agente tiene la posibilidad de prevalecer sobre la pieza de evidencia si así lo deciden los participantes de la sesión.

Respecto a los *rechazos por contradicción*, un agente $Z = (ID, \Pi_Z, \Delta_Z, >)$ que hace un movimiento *rechazar*(\mathcal{A}, \mathcal{C}) con $\mathcal{A} = \langle a, R, H \rangle$ no podría construir \mathcal{A} incluso si cree en todos las reglas rebatibles y hechos necesarios (*i. e.*, $H \subseteq \Pi_Z$ y $R \subseteq \Delta_Z$). Esto se debe a que dos literales contradictorios serían derivados a partir de $\Pi_Z \cup R$ (los literales h y \bar{h} de la Definición 4.5) causando que la condición 3 de la Definición 2.13 no se cumpla.

Algo similar sucede con los *rechazos por no-minimalidad*. Si un agente puede construir un argumento \mathcal{M} minimal respecto a \mathcal{A} , entonces cree en al menos un hecho h que se corresponde con la cabeza de una regla rebatible $h \prec p_1, \dots, p_n$ de \mathcal{A} . Desde su perspectiva, $h \prec p_1, \dots, p_n$ es innecesaria, y h causa que la condición 4 de la Definición 2.13 no se cumpla y que, en consecuencia, \mathcal{A} no sea un argumento válido.

En ambos tipos de rechazo, el agente que realiza el movimiento comparte el *argumento invalidante* (*i. e.*, \mathcal{C} y \mathcal{M} , respectivamente) con el resto de los participantes de la sesión.

En las siguientes cuatro subsecciones, se explicarán en detalle las reglas de transiciones que especifican cómo evoluciona una sesión al realizar cada uno de los movimientos presentados en la definición anterior.

4.2.3.1. Movimiento para derrotar un argumento del árbol de indagación

Siempre que un agente derrote a un argumento \mathcal{A} del árbol de indagación mediante un movimiento *derrotar*, el argumento derrotador \mathcal{D} debe ser *vinculado* al árbol de indagación como nuevo hijo de \mathcal{A} . Recuerde que un argumento puede aparecer más de una vez en el mismo árbol de dialéctica (en diferentes líneas de argumentación), lo cual también podría suceder en un árbol de indagación. Por esta razón, un nodo etiquetado con \mathcal{D} será vinculado debajo de todo nodo etiquetado con \mathcal{A} siempre que la línea de argumentación resultante sea aceptable.

Definición 4.6 (Vinculación al árbol de indagación) Dado un árbol de indagación $\mathbb{I} = \langle \text{Nodos}, \text{Arcos} \rangle$ y dos argumentos \mathcal{A} y \mathcal{D} , la *vinculación* de \mathcal{D} debajo de \mathcal{A} en \mathbb{I} se define como $\text{VINCULACIÓN}(\mathbb{I}, \mathcal{A}, \mathcal{D}) = \mathbb{I}'$ donde \mathbb{I}' es un árbol de indagación con todos los nodos y arcos de \mathbb{I} y estos nodos adicionales: Para cada $\aleph_{\mathcal{A}} \in \text{Nodos}$ tal que $\aleph_{\mathcal{D}} \notin \text{HIJOS}(\aleph_{\mathcal{A}})$, sea Λ una línea de argumentación en \mathbb{I} con $\aleph_{\mathcal{A}} \in^* \Lambda$, si $\text{SEGMENTOSUPERIOR}(\Lambda, \aleph_{\mathcal{A}}) \cap [\aleph_{\mathcal{D}}]$ es una línea de argumentación aceptable, entonces $\aleph_{\mathcal{D}} \in \text{HIJOS}(\aleph_{\mathcal{A}})$. ■

A continuación, se presentará la regla de transición ri_4 , la cual especifica el efecto de realizar el movimiento **derrotar** (ver β a γ en la Figura 4.2).

$$ri_4 : \frac{ID \in \mathbb{E} \wedge \text{AGENTE}(ID, \mathbb{Z}) = \mathbf{Z} \wedge \text{MOVIMIENTO}(\mathbb{I}, \mathbb{Z}) = \text{derrotar}(\mathcal{A}, \mathcal{D}) \wedge \text{VINCULACIÓN}(\mathbb{I}, \mathcal{A}, \mathcal{D}) = \mathbb{I}'}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon], \mathbb{E}]) \rightarrow (\mathbb{I}', \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{derrotar}(\mathcal{A}, \mathcal{D}) \rangle]), \mathbb{E} \setminus \{ID\}}$$

Dado un conjunto de agentes \mathbb{Z} y un identificador ID , el operador **AGENTE** devuelve el agente de \mathbb{Z} cuyo identificador es ID . Recuerde que si el último elemento de la ronda actual es una marca de espera ε y la cola de espera \mathbb{E} no está vacía, un agente debe realizar un movimiento. Dado un estado en el que es el turno de un agente $\mathbf{Z} = (ID, \Pi_{\mathbf{Z}}, \Delta_{\mathbf{Z}}, >)$ con $ID \in \mathbb{E}$, la regla de transición ri_4 especifica cómo dicho estado evoluciona a uno nuevo en el que \mathbf{Z} dió un argumento \mathcal{D} como derrotador para otro argumento \mathcal{A} , y \mathcal{D} fue vinculado debajo de \mathcal{A} en el árbol de indagación \mathbb{I} , el cual evolucionó a \mathbb{I}' . El primer componente del estado de sesión es reemplazado por \mathbb{I}' , el movimiento $\langle ID, \text{derrotar}(\mathcal{A}, \mathcal{D}) \rangle$ es añadido luego de la marca de espera ε en la ronda actual, e ID es eliminado de la cola de espera \mathbb{E} .

Aunque \mathbb{E} es llamada “cola” de espera, la regla de transición ri_4 (y ri_6 , ri_9 , ri_8 , y ri_{11} , como se verá más adelante) seleccionan el agente que realizará el próximo movimiento de manera arbitraria. Esto quiere decir que el orden en el que los agentes realizan movimientos puede variar de ronda en ronda. Sin embargo, puede utilizarse la implementación del método de selección más adecuada para el dominio de aplicación, *e. g.*, una cola de prioridad. Por simplicidad, en los ejemplos del resto de la sección los agentes de \mathbb{E} serán seleccionados en un orden conveniente para la explicación que se está desarrollando.

Ejemplo 4.6 Continuando con el Ejemplo 4.5, considere el estado de sesión $e_5 = (\mathbb{I}^1, \{\mathbb{M}^2, \mathbb{L}^2, \mathbb{C}^2, \mathbb{S}^2\}, [[\varepsilon]], \{\text{moe}, \text{larry}, \text{curly}, \text{shemp}\})$. Como el último elemento de la ronda actual es una marca de espera ε y la cola de espera no está vacía, **moe**, **larry**, **curly** o **shemp** debe hacer un movimiento. En particular, $\mathbb{L}^2 = (\text{larry}, \Pi_{\mathbb{L}^2}, \Delta_{\mathbb{L}^2}, >)$ puede construir el argumento $\mathcal{A}_2 = \langle \sim a, \{\sim a \prec b; b \prec y, z\}, \{y, z\} \rangle$ (ver Figura 4.4) y supongamos que $\mathcal{A}_2 > \mathcal{A}_1$. Por lo tanto, el agente **larry** puede contribuir a la sesión el argumento \mathcal{A}_2 como un derrotador para el argumento de indagación $\mathcal{A}_1 = \langle a, \{a \prec z\}, \{z\} \rangle$. De acuerdo a la Definición 4.5, **larry** puede hacer el movimiento **derrotar**($\mathcal{A}_1, \mathcal{A}_2$). La regla de transición ri_4 es aplicada y la sesión evoluciona al estado $e_6 = (\mathbb{I}^2, \{\mathbb{M}^2, \mathbb{L}^2, \mathbb{C}^2, \mathbb{S}^2\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle]], \{\text{moe}, \text{curly}, \text{shemp}\})$ luego de que \mathcal{A}_2 fue vinculado debajo de \mathcal{A}_1 en \mathbb{I}^1 , el cual evolucionó a \mathbb{I}^2 . ■

Si en lugar de una marca de espera ε el último elemento de la ronda actual es $\langle ID, Movimiento \rangle$, los agentes deberán adquirir diferente información dependiendo del último movimiento que fue realizado. Si el último movimiento fue **derrotar**(\mathcal{A}, \mathcal{D}) los agentes procederán a adquirir el argumento derrotador \mathcal{D} , como es especificado por la regla de transición ri_5 (ver γ a β , y γ a δ en la Figura 4.2).

$$ri_5 : \frac{\mathbb{Z}' = \{Z' \mid Z \in \mathbb{Z} \wedge \text{ADQARGUMENTO}(Z, \mathcal{D}) = Z'\}}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{derrotar}(\mathcal{A}, \mathcal{D}) \rangle]], \mathbb{E}) \rightarrow (\mathbb{I}, \mathbb{Z}', [\dots, [\dots, \langle ID, \text{derrotar}(\mathcal{A}, \mathcal{D}) \rangle], \varepsilon], \mathbb{E})}$$

Dado un estado en el que el último elemento de la ronda actual es $\langle ID, \text{derrotar}(\mathcal{A}, \mathcal{D}) \rangle$, la regla de transición ri_5 especifica cómo dicho estado evoluciona en uno nuevo en el que los agentes en \mathbb{Z} adquirieron el argumento derrotador \mathcal{D} , resultando en un nuevo conjunto de agentes \mathbb{Z}' . Una marca de espera ε es añadida al final de la ronda actual y, por lo tanto, si la cola de espera \mathbb{E} no está vacía, será nuevamente el turno de un agente en \mathbb{E} .

Ejemplo 4.7 Continuando con el Ejemplo 4.6, considere el estado de sesión $e_6 = (\mathbb{I}^2, \{\mathbb{M}^2, \mathbb{L}^2, \mathbb{C}^2, \mathbb{S}^2\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle]], \{\text{moe}, \text{curly}, \text{shemp}\})$ en el que los agentes están por adquirir el argumento $\mathcal{A}_2 = \langle \sim a, \{\sim a \prec b; b \prec y, z\}, \{y, z\} \rangle$ a partir del movimiento **derrotar**($\mathcal{A}_1, \mathcal{A}_2$) hecho por **larry**. La regla de transición ri_5 es aplicada y la sesión evoluciona a $e_7 = (\mathbb{I}^2, \{\mathbb{M}^3, \mathbb{L}^2, \mathbb{C}^3, \mathbb{S}^3\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2), \varepsilon \rangle]], \{\text{moe}, \text{curly}, \text{shemp}\})$. La Figura 4.5 ilustra las bases de creencias de **moe**, **curly** y **shemp** luego de adquirir el argumento \mathcal{A}_2 . Observe que **shemp** no adquirió el hecho y porque cree en $\sim y$ (ver Definición 4.3) y, por lo tanto, no podrá construir \mathcal{A}_2 inmediatamente. Más adelante, en el Ejemplo 4.12, se explicará cómo **shemp** lidiará con esta situación. ■

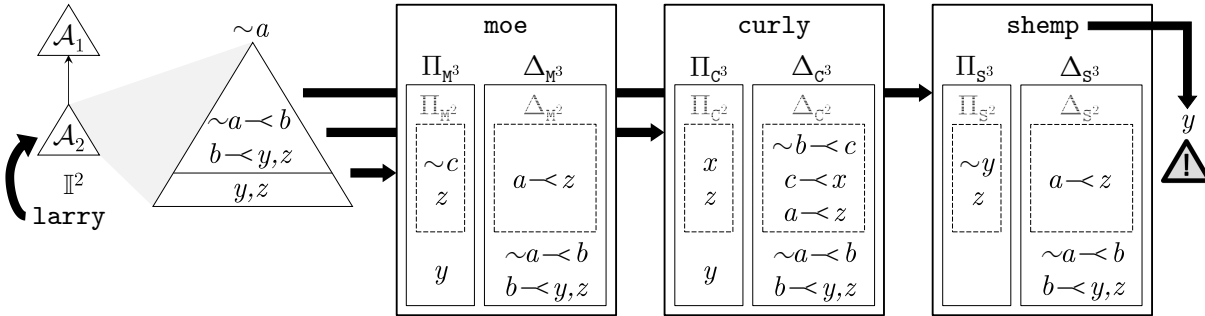


Figura 4.5: Bases de creencias de los agentes **moe**, **curly** y **shemp** en el Ejemplo 4.7 luego de adquirir el argumento \mathcal{A}_2 compartido por **larry** como derrotador de \mathcal{A}_1 .

Ejemplo 4.8 Continuando con el Ejemplo 4.7, considere el estado de sesión $e_7 = (\mathbb{I}^2, \{\mathbb{M}^3, \mathbb{L}^2, \mathbb{C}^3, \mathbb{S}^3\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2), \varepsilon \rangle], \{\text{moe}, \text{curly}, \text{shemp}\}])$ en el que moe, curly o shemp debe hacer un movimiento. En particular, $\mathbb{C}^2 = (\text{curly}, \Pi_{\mathbb{C}^2}, \Delta_{\mathbb{C}^2}, >)$ puede construir el argumento $\mathcal{B} = \langle \sim b, \{\sim b \prec c; c \prec x\}, \{x\} \rangle$ (ver Figura 4.5) y supongamos que $\mathcal{B} > \mathcal{A}_2$. Por lo tanto, curly puede hacer el movimiento $\text{derrotar}(\mathcal{A}_2, \mathcal{B})$. La regla de transición ri_4 es aplicada y la sesión evoluciona al estado $e_8 = (\mathbb{I}^3, \{\mathbb{M}^3, \mathbb{L}^2, \mathbb{C}^3, \mathbb{S}^3\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle], \{\text{moe}, \text{shemp}\}])$ luego de que \mathcal{B} fue vinculado debajo de \mathcal{A}_2 en \mathbb{I}^2 , el cual evolucionó a \mathbb{I}^3 . A continuación, la regla de transición ri_5 es aplicada y la sesión evoluciona al estado $e_9 = (\mathbb{I}^3, \{\mathbb{M}^4, \mathbb{L}^3, \mathbb{C}^3, \mathbb{S}^4\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon], \{\text{moe}, \text{shemp}\}])$ luego de que moe, larry y shemp adquirieron el argumento \mathcal{B} a partir del movimiento $\text{derrotar}(\mathcal{A}_2, \mathcal{B})$ hecho por curly (ver Figura 4.6). ■

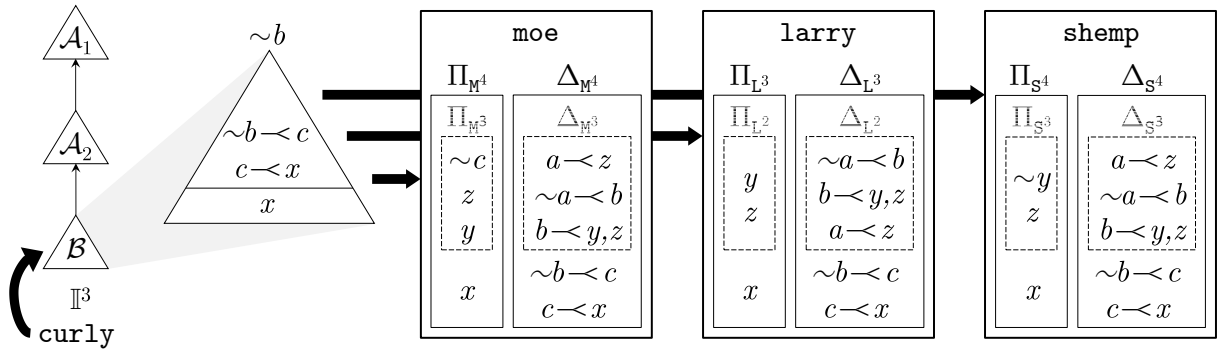


Figura 4.6: Bases de creencias de los agentes moe, larry y shemp en el Ejemplo 4.8 luego de adquirir el argumento \mathcal{B} compartido por curly como derrotador de \mathcal{A}_2 .

4.2.3.2. Movimiento para rechazar un argumento del árbol de indagación

Como se mencionó anteriormente, otro posible movimiento para los agentes es **rechazar** un argumento \mathcal{A} del árbol de indagación (por contradicción o no-minimalidad) dando un argumento invalidante \mathcal{I} . Siempre que esto ocurra, todo nodo etiquetado con un argumento que contenga a \mathcal{A} como subargumento¹ debe ser *desvinculado* del árbol de indagación junto con todos sus nodos descendientes. El árbol de indagación resultante \mathbb{I}' es un nuevo árbol que contiene el subconjunto maximal de nodos y arcos de \mathbb{I} tal que no hay nodos etiquetados con un argumento que contiene a \mathcal{A} como subargumento.

¹Recuerde que en DeLP todo argumento es subargumento de sí mismo.

Definición 4.7 (Desvinculación de árbol de indagación) Dado un árbol de indagación $\mathbb{I} = \langle \text{Nodos}, \text{Arcos} \rangle$ y un argumento \mathcal{A} , la *desvinculación* de \mathcal{A} de \mathbb{I} se define como $\text{DESVINCLACIÓN}(\mathbb{I}, \mathcal{A}) = \mathbb{I}'$ donde $\mathbb{I}' = \langle \text{Nodos}', \text{Arcos}' \rangle$ es un árbol de indagación tal que $\text{Nodos}' \subseteq \text{Nodos}$, $\text{Arcos}' \subseteq \text{Arcos}$, $\aleph_{\mathcal{S}} \notin \text{Nodos}'$ tal que \mathcal{A} es subargumento de \mathcal{S} , y Nodos' y Arcos' son maximales. ■

A continuación, se presentará la regla de transición ri_6 , la cual especifica el efecto de realizar el movimiento *rechazar* (ver β a γ en la Figura 4.2).

$$ri_6 : \frac{ID \in \mathbb{E} \wedge \text{AGENTE}(ID, \mathbb{Z}) = \mathbb{Z} \wedge \text{MOVIMIENTO}(\mathbb{I}, \mathbb{Z}) = \text{rechazar}(\mathcal{A}, \mathcal{I}) \wedge \text{DESVINCLACIÓN}(\mathbb{I}, \mathcal{A}) = \mathbb{I}'}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon]], \mathbb{E}) \rightarrow (\mathbb{I}', \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{I}) \rangle]), \mathbb{E} \setminus \{ID\}}$$

Dado un estado en el que es el turno de un agente $\mathbb{Z} = (ID, \Pi_{\mathbb{Z}}, \Delta_{\mathbb{Z}}, >)$ con $ID \in \mathbb{E}$, la regla de transición ri_6 especifica cómo dicho estado evoluciona a uno nuevo en el que \mathbb{Z} dio un argumento \mathcal{B} como razón para *rechazar* otro argumento \mathcal{A} , y \mathcal{A} fue desvinculado del árbol de indagación \mathbb{I} , el cual evolucionó a \mathbb{I}' .

Ejemplo 4.9 Continuando con el Ejemplo 4.8, considere el estado de sesión $e_9 = (\mathbb{I}^3, \{\mathbb{M}^4, \mathbb{L}^3, \mathbb{C}^3, \mathbb{S}^4\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon]], \{\text{moe}, \text{shemp}\})$. Dado que el último elemento de la ronda actual es una marca de espera ε , *moe* o *shemp* debe hacer un movimiento. En particular, $\mathbb{M}^4 = (\text{moe}, \Pi_{\mathbb{M}^4}, \Delta_{\mathbb{M}^4}, >)$ puede construir el argumento estricto $\mathcal{C} = \langle \sim c, \emptyset, \{\sim c\} \rangle$, el cual implica que \mathcal{B} no es un argumento válido ya que utiliza la regla rebatible $c \prec x$, *i. e.*, realiza una derivación en contradicción con una pieza de información estricta. Por lo tanto, según la Definición 4.5, *moe* puede realizar el movimiento $\text{rechazar}(\mathcal{B}, \mathcal{C})$. La regla de transición ri_6 es aplicada y la sesión evoluciona al estado $e_{10} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^3, \mathbb{C}^3, \mathbb{S}^4\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle]], \{\text{shemp}\})$ luego de que \mathcal{B} fue desvinculado de \mathbb{I}_3 , el cual evolucionó a \mathbb{I}_4 (ver Figura 4.7, izquierda). Observe que, aunque el argumento estricto \mathcal{C} es la razón para desvincular a \mathcal{B} de \mathbb{I}^3 , \mathcal{C} no es parte del nuevo árbol de indagación. ■

De manera similar a la regla de transición ri_5 , si el último elemento de la ronda actual es $\text{rechazar}(\mathcal{A}, \mathcal{I})$ los agentes procederán a adquirir el argumento invalidante \mathcal{I} , como es especificado por la regla de transición ri_7 (ver γ a β , y γ a δ en la Figura 4.2).

$$ri_7 : \frac{\mathbb{Z}' = \{Z' \mid Z \in \mathbb{Z} \wedge \text{ADQARGUMENTO}(Z, \mathcal{I}) = Z'\}}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{I}) \rangle], \mathbb{E}) \rightarrow (\mathbb{I}, \mathbb{Z}', [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{I}) \rangle], \varepsilon], \mathbb{E})}$$

Note que la condición de la regla de transición ri_7 es igual a la de ri_5 . En este caso el argumento adquirido siempre es un argumento estricto.

Ejemplo 4.10 Continuando con el Ejemplo 4.9, considere el estado de sesión $e_{10} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^3, \mathbb{C}^3, \mathbb{S}^4\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle], \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle], \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle], \{\text{shemp}\})$ en el que los agentes están por adquirir el argumento estricto $\mathcal{C} = \langle \sim c, \emptyset, \{\sim c\} \rangle$ a partir del movimiento $\text{rechazar}(\mathcal{B}, \mathcal{C})$ realizado por moe. La regla de transición ri_7 es aplicada y la sesión evoluciona al estado $e_{11} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^4, \mathbb{C}^4, \mathbb{S}^5\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle], \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle], \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle], \varepsilon], \{\text{shemp}\})$. La Figura 4.7 ilustra las bases de creencias de larry, curly y shemp luego de adquirir \mathcal{C} . Note que, según la Definición 2.13, los agentes ya no pueden construir el argumento \mathcal{B} porque ahora creen en el hecho $\sim c$. ■

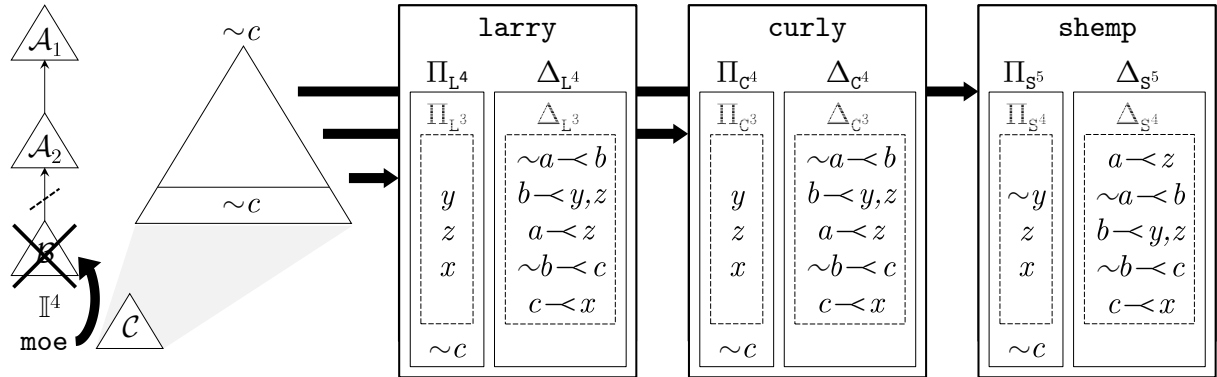


Figura 4.7: Bases de creencias de los agentes larry, curly y shemp en el Ejemplo 4.10 luego de adquirir el argumento estricto \mathcal{C} compartido por el agente moe al rechazar \mathcal{B} .

Ejemplo 4.11 Continuando con el Ejemplo 4.10, supongamos que el agente $J^1 = (\text{joe}, \Pi_{J^1}, \Delta_{J^1}, >)$ con $\Pi_{J^1} = \{\sim y\}$ y $\Delta_{J^1} = \emptyset$ quiere unirse a la sesión. La regla de transición ri_1 es aplicada y la sesión evoluciona al estado $e_{12} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^4, \mathbb{C}^4, \mathbb{S}^5, \mathbb{J}^2\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle], \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle], \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle], \varepsilon], \{\text{shemp}, \text{joe}\})$ luego de que J^1 se unió, adquirió \mathbb{I}^4 , y evolucionó a J^2 . La Figura 4.8 ilustra la

base de creencias de **joe** luego de adquirir los argumentos $\mathcal{A}_1 = \langle a, \{a \prec z\}, \{z\} \rangle$ y $\mathcal{A}_2 = \langle \sim a, \{\sim a \prec b; b \prec y, z\}, \{y, z\} \rangle$ del árbol de indagación. Observe que **joe**, al igual que **shemp** en el Ejemplo 4.7, no adquirió el hecho y porque cree en $\sim y$ (ver Definición 4.3). Por lo tanto, no podrá construir el argumento \mathcal{A}_2 inmediatamente. En el próximo ejemplo se explicará cómo **joe** y **shemp** lidiarán con esta situación. ■

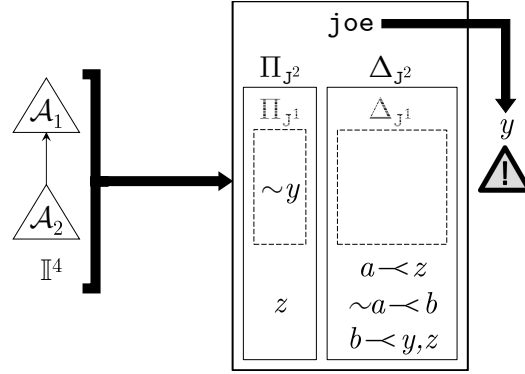


Figura 4.8: Base de creencias del agente **joe** en el Ejemplo 4.11 luego de unirse a la sesión y adquirir los argumentos del árbol de indagación \mathbb{I}^4 .

4.2.3.3. Movimiento para objetar un hecho del árbol de indagación

Un agente $Z = (ID, \Pi_Z, \Delta_Z, >)$ podría ser incapaz de construir un argumento $\mathcal{A} = \langle a, R, H \rangle$ luego de adquirirlo (recuerde los agentes **shemp** del Ejemplo 4.7 y **joe** del Ejemplo 4.11). Si esto ocurre, según las Definiciones 4.3 y 4.5, Z puede hacer el movimiento $\text{objetar}(h, \bar{h})$ ya que existe un hecho $h \in H \subseteq \text{EVIDENCIA}(\mathbb{I})$ tal que $\bar{h} \in \Pi_Z$. Por lo tanto, un agente que cree en un hecho en contradicción con una pieza de evidencia del árbol de indagación tiene un medio para comunicárselo a los demás participantes de la sesión.

El movimiento **objetar** muestra la necesidad de llegar a un consenso sobre cuál de los hechos en contradicción debería ser aceptado y cuál debería ser rechazado. Dado que en los sistemas multi-agente existen diferentes estrategias para lograr consensos (y este tema puntual está fuera del foco de esta tesis) se supondrá la existencia de un operador **CONSENSO** que encapsula de manera modular esta decisión. En particular, se supondrá que $\text{CONSENSO}(\mathbb{Z}, h, \bar{h})$ devuelve el literal que fue aceptado (h o \bar{h}) por el conjunto de agentes \mathbb{Z} . Esto permite utilizar la implementación del operador más adecuada para el dominio de aplicación, *e. g.*, *mayoría simple*. En la Sección 4.4 se discutirá cómo operadores

más sofisticados podrían ser implementados utilizando información adicional, tal como la credibilidad de los informantes [TGFS14, TGGS17, GTGS18].

A continuación, se presentarán las reglas de transición ri_8 y ri_9 , las cuales especifican los dos posibles efectos de realizar un movimiento $\text{objetar}(h, \bar{h})$ (ver β a γ en la Figura 4.2). Si el consenso entre los participantes de la sesión es aceptar \bar{h} y rechazar h (i. e., $\text{CONSENSO}(\mathbb{Z}, h, \bar{h}) = \bar{h}$), todo argumento que utiliza el hecho h como premisa debe ser desvinculado del árbol de indagación como lo especifica la regla de transición ri_8 . En caso contrario, si $\text{CONSENSO}(\mathbb{Z}, h, \bar{h}) = h$, el árbol quedará intacto como lo especifica la regla de transición ri_9 .

$$ri_8 : \frac{ID \in \mathbb{E} \wedge \text{AGENTE}(ID, \mathbb{Z}) = \mathbb{Z} \wedge \text{MOVIMIENTO}(\mathbb{I}, \mathbb{Z}) = \text{objetar}(h, \bar{h}) \wedge \text{CONSENSO}(\mathbb{Z}, h, \bar{h}) = \bar{h} \wedge \text{DESVINCULACIÓN}(\mathbb{I}, \langle h, \emptyset, \{h\} \rangle) = \mathbb{I}'}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon]], \mathbb{E}) \rightarrow (\mathbb{I}', \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{objetar}(h, \bar{h}, \bar{h}) \rangle]), \mathbb{E} \setminus \{ID\}}$$

$$ri_9 : \frac{ID \in \mathbb{E} \wedge \text{AGENTE}(ID, \mathbb{Z}) = \mathbb{Z} \wedge \text{MOVIMIENTO}(\mathbb{I}, \mathbb{Z}) = \text{objetar}(h, \bar{h}) \wedge \text{CONSENSO}(\mathbb{Z}, h, \bar{h}) = h}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon]], \mathbb{E}) \rightarrow (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{objetar}(h, \bar{h}, h) \rangle]), \mathbb{E} \setminus \{ID\}}$$

Dado un estado en el que es el turno de un agente $\mathbb{Z} = (ID, \Pi_{\mathbb{Z}}, \Delta_{\mathbb{Z}}, >)$ con $ID \in \mathbb{E}$, las reglas de transición ri_8 y ri_9 especifican cómo dicho estado evoluciona a uno nuevo en el que \mathbb{Z} objetó al hecho h porque cree en \bar{h} . Si el hecho h (el cual es parte de $\text{EVIDENCIA}(\mathbb{I})$) es el rechazado por el consenso, la regla de transición ri_8 es aplicada y todo argumento que usa h es desvinculado de \mathbb{I} . Recuerde que $\langle h, \emptyset, \{h\} \rangle$ es un subargumento de todo argumento $\mathcal{A}_2 = \langle a, \mathbf{R}, \mathbf{H} \rangle$ tal que $h \in \mathbf{H}$ y, por lo tanto, desvincular $\langle h, \emptyset, \{h\} \rangle$ de \mathbb{I} logra el efecto deseado y resulta en un nuevo árbol de indagación \mathbb{I}' . En cambio, si el hecho \bar{h} (el cual es parte de $\Pi_{\mathbb{Z}}$) es el rechazado por el consenso, la regla de transición ri_9 es aplicada y no hay argumentos que desvincular de \mathbb{I} . Respecto a lo añadido luego de la marca de espera ε , en ambas reglas de transición el movimiento objetar contiene un elemento adicional correspondiente al hecho que fue aceptado. Posteriormente los participantes utilizarán este hecho aceptado para actualizar sus bases de creencias como corresponde.

Ejemplo 4.12 Continuando con el Ejemplo 4.11, considere el estado de sesión $e_{12} = (\mathbb{I}^4, \{\mathbf{M}^4, \mathbf{L}^4, \mathbf{C}^4, \mathbf{S}^5, \mathbf{J}^2\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle], \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle], \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle], \varepsilon], \{\text{shemp}, \text{joe}\})$ en el que el agente shemp o larry debe realizar un

movimiento. Recuerde que, en el Ejemplo 4.7, **shemp** no adquirió el hecho y porque cree en $\sim y$. Luego, según la Definición 4.5 **shemp** puede realizar el movimiento $\text{objetar}(y, \sim y)$, lo cual implica que los agentes tendrán que llegar a un consenso sobre cuál de los hechos en contradicción es aceptado. Supongamos que $\text{CONSENSO}(\{\mathcal{M}^4, \mathcal{L}^4, \mathcal{C}^4, \mathcal{S}^5, \mathcal{J}^2\}, y, \sim y) = y$. En consecuencia, la regla de transición ri_9 es aplicada y la sesión evoluciona al estado $e_{13} = (\mathbb{I}^4, \{\mathcal{M}^4, \mathcal{L}^4, \mathcal{C}^4, \mathcal{S}^5, \mathcal{J}^2\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle, \varepsilon, \langle \text{shemp}, \text{objetar}(y, \sim y, y) \rangle]], \{\text{joe}\})$. ■

Si el último elemento de la ronda actual es $\text{objetar}(h, \bar{h}, a)$, todo agente que crea en \bar{a} deberá añadir a y eliminar \bar{a} de su base de creencias porque, de acuerdo al consenso, \bar{a} es un hecho inválido. Sin embargo, es razonable pensar que un agente podría rehusarse a actualizar sus creencias de manera acorde por no estar de acuerdo con el consenso. Dado que en este protocolo y estrategia de diálogo se supone un escenario colaborativo, dicho agente no puede seguir participando de la sesión porque el resto continuará razonando sobre el mismo conjunto de premisas consistentes. Por este motivo, la regla de transición ri_2 permite a dicho agente abandonar la sesión sin importar el estado actual.

A continuación, se presentará la regla de transición ri_{10} , la cual especifica cómo los participantes de una sesión actualizan sus creencias luego de que se realizó un movimiento $\text{objetar}(h, \bar{h}, a)$ (ver γ a β , y γ a δ en la Figura 4.2).

$$ri_{10} : \frac{\mathbb{Z}' = \{\mathbb{Z}' \mid (ID, \Pi_{\mathbb{Z}}, \Delta_{\mathbb{Z}}, >) \in \mathbb{Z} \wedge \mathbb{Z}' = (ID, (\Pi_{\mathbb{Z}} \setminus \{\bar{a}\}) \cup \{a\}, \Delta_{\mathbb{Z}}, >)\}}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle \text{OtroID}, \text{objetar}(h, \bar{h}, a) \rangle], \mathbb{E}) \rightarrow (\mathbb{I}, \mathbb{Z}', [\dots, [\dots, \langle \text{OtroID}, \text{objetar}(h, \bar{h}, a) \rangle, \varepsilon], \mathbb{E})}$$

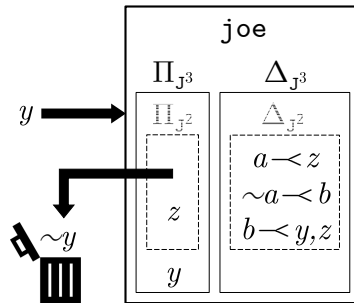


Figura 4.9: Base de creencias del agente joe en el Ejemplo 4.13.

Ejemplo 4.13 Continuando con el Ejemplo 4.12, considere el estado de sesión $e_{13} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^4, \mathbb{C}^4, \mathbb{S}^5, \mathbb{J}^2\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle, \varepsilon, \langle \text{shemp}, \text{objetar}(y, \sim y, y) \rangle]], \{\text{joe}\})$ en el que los agentes deben actualizar sus creencias en base al consenso sobre el movimiento $\text{objetar}(y, \sim y, y)$ realizado por **shemp**. Dado que **shemp** y **joe** creen en el hecho $\sim y$, si quieren continuar en la sesión deberán añadir y y eliminar $\sim y$ de sus bases de creencias. Sin embargo, **shemp** se rehúsa a dejar de creer en $\sim y$ y decide abandonar la sesión. En consecuencia, la regla de transición ri_2 es aplicada y la sesión evoluciona al estado $e_{14} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^4, \mathbb{C}^4, \mathbb{J}^2\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle, \varepsilon, \langle \text{shemp}, \text{objetar}(y, \sim y, y) \rangle]], \{\text{joe}\})$. Luego, la regla de transición ri_{10} es aplicada y la sesión evoluciona a $e_{15} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^4, \mathbb{C}^4, \mathbb{J}^3\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle, \varepsilon, \langle \text{shemp}, \text{objetar}(y, \sim y, y) \rangle, \varepsilon]], \{\text{joe}\})$. La Figura 4.9 ilustra la base de creencias de **joe** luego de añadir y y eliminar $\sim y$ de $\Pi_{\mathbb{J}^2}$ (los otros agentes no tuvieron que hacer cambios). Note que $\mathbb{J}^3 = (\text{joe}, \Pi_{\mathbb{J}^3}, \Delta_{\mathbb{J}^3}, >)$ puede finalmente construir el argumento $\mathcal{A}_2 = \langle \sim a, \{\sim a \prec b; b \prec y, z\}, \{y, z\} \rangle$. ■

4.2.3.4. Pasar el turno cuando no hay nada más que opinar

Cuando un participante no puede hacer un movimiento **derrotar**, **rechazar** u **objetar** durante su turno, deberá **pasar**. Cuando esto sucede el agente es simplemente eliminado de la cola de espera, como es especificado por la regla de transición ri_{11} (ver β a β , y β a δ en la Figura 4.2).

$$ri_{11} : \frac{ID \in \mathbb{E} \quad \wedge \quad \text{AGENTE}(ID, \mathbb{Z}) = \mathbb{Z} \quad \wedge \quad \text{MOVIMIENTO}(\mathbb{I}, \mathbb{Z}) = \text{pasar}}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon], \mathbb{E}] \rightarrow (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{pasar} \rangle, \varepsilon], \mathbb{E} \setminus \{ID\})}$$

Como no hay información para adquirir a partir de este tipo de movimiento, cuando la regla de transición ri_{11} es aplicada una marca de espera ε es añadida al final de la ronda actual junto al movimiento $\langle ID, \text{pasar} \rangle$.

Ejemplo 4.14 Continuando con el Ejemplo 4.13, considere el estado de sesión $e_{15} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^4, \mathbb{C}^4, \mathbb{J}^3\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle, \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle, \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle, \varepsilon, \langle \text{shemp}, \text{objetar}(y, \sim y, y) \rangle, \varepsilon]], \{\text{joe}\})$ en el que es el turno de **joe** de

hacer un movimiento. De acuerdo a la Definición 4.5, $J^3 = (\text{joe}, \Pi_{J^3}, \Delta_{J^3}, >)$ no puede realizar otro movimiento además de **pasar**. Por lo tanto, la regla de transición ri_{11} es aplicada y la sesión evoluciona a $e_{16} = (\mathbb{I}^4, \{\mathbb{M}^4, \mathbb{L}^4, \mathbb{C}^4, \mathbb{J}^3\}, [[\varepsilon, \langle \text{larry}, \text{derrotar}(\mathcal{A}_1, \mathcal{A}_2) \rangle], \varepsilon, \langle \text{curly}, \text{derrotar}(\mathcal{A}_2, \mathcal{B}) \rangle], \varepsilon, \langle \text{moe}, \text{rechazar}(\mathcal{B}, \mathcal{C}) \rangle], \varepsilon, \langle \text{shemp}, \text{objetar}(y, \sim y, y) \rangle], \varepsilon, \langle \text{joe}, \text{pasar} \rangle, \varepsilon], \emptyset)$.

4.2.4. Fin de una ronda: ¿es necesario continuar la sesión?

Cuando la cola de espera se vacía, la ronda actual finaliza y la sesión continuará o no dependiendo de los movimientos realizados durante dicha ronda. Dado que la sesión debe continuar hasta que los participantes no tengan nada más para contribuir, si al menos un agente no pasó entonces se creará una nueva ronda y la sesión continuará. Esto es especificado por la regla de transición ri_{12} (ver δ a β en la Figura 4.2).

$$ri_{12} : \frac{\text{existe } \langle ID, \text{Movimiento} \rangle \in^* R_n \text{ tal que } \text{Movimiento} \neq \text{pasar} \wedge \mathbb{E} = \{ID' \mid (ID', \Pi_{\mathbb{Z}}, \Delta_{\mathbb{Z}}, >) \in \mathbb{Z}\}}{(\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n], \emptyset) \rightarrow (\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n, [\varepsilon]], \mathbb{E})} \quad n \geq 1$$

Observe que una nueva secuencia con una marca de espera ε es añadida como nueva ronda al final del tercer componente del nuevo estado de sesión. Además, la cola de espera es inicializada con los identificadores de los participantes en \mathbb{Z} .

Es importante tener en cuenta que, como los agentes adquieren nuevas creencias a partir de los movimientos de los demás, aún si un participante pasó en una ronda, nada impide que haga un movimiento durante la ronda siguiente. Por ejemplo, un agente podría adquirir un hecho o regla rebatible que le permite construir un nuevo derrotador para un argumento del árbol de indagación, el cual le permita realizar un movimiento **derrotar** que antes no podía. En cambio, si todos los agentes pasaron durante la última ronda, sus bases de creencias seguirán igual durante la próxima y tampoco tendrán movimientos para realizar. En este caso, la sesión puede finalizar, como es especificado por la regla de transición ri_{13} (ver δ a ε en la Figura 4.2).

$$ri_{13} : \frac{\text{para todo } \langle ID, \text{Movimiento} \rangle \in^* R_n : \text{Movimiento} = \text{pasar}}{(\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n], \emptyset) \rightarrow (\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n, [\text{fin}]], \emptyset)} \quad n \geq 1$$

Cuando los participantes no tienen nada más para contribuir y regla de transición ri_{13} es aplicada, la sesión evoluciona a un estado final con el terminador **[fin]** como último elemento de su tercer componente.

4.2.5. Ejemplo integrador

A continuación, se presentará un ejemplo integrador que formaliza la interacción entre los agentes del Ejemplo 4.1: un equipo de tres mineros de un videojuego que indagan sobre un argumento para darle mineral al jugador, quien necesita ese recurso. Las reglas de transición aplicadas serán denotadas entre corchetes, y los números dentro de los rectángulos serán utilizados para referirse a las diferentes partes de los árboles de indagación ilustrados en la Figura 4.10.

Ejemplo 4.15 Considere las bases de creencias de los agentes $P = (\text{peter}, \Pi_P, \Delta_P, >)$, $E = (\text{egon}, \Pi_E, \Delta_E, >)$ y $R = (\text{raymond}, \Pi_R, \Delta_R, >)$.

$$\begin{aligned} \Pi_P &= \left\{ \begin{array}{l} \text{ubicación}(\text{mineral}, \text{cavernas}) \\ \text{peligro}(\text{derrumbe}, \text{cavernas}) \\ \sim \text{difícil_encontrar}(\text{mineral}) \end{array} \right\} \\ \Delta_P &= \left\{ \begin{array}{l} \sim \text{dar}(R, A) \prec \text{necesita}(R, A), \text{valioso}(R) \\ \text{valioso}(R) \prec \text{ubicación}(R, U), \text{peligrosas}(U) \\ \text{peligrosas}(U) \prec \text{peligro}(P, U) \\ \sim \text{valioso}(R) \prec \text{mucho_en_inventario}(R) \end{array} \right\} \\ \Pi_E &= \left\{ \begin{array}{l} \text{peligro}(\text{derrumbe}, \text{cavernas}) \\ \text{enemigo}(\text{jugador}) \\ \text{equipamiento_seguridad}(\text{derrumbe}) \end{array} \right\} \\ \Delta_E &= \left\{ \begin{array}{l} \sim \text{dar}(R, A) \prec \text{necesita}(R, A), \text{enemigo}(A) \\ \sim \text{peligrosas}(U) \prec \text{peligro}(P, U), \text{equipamiento_seguridad}(P) \end{array} \right\} \\ \Pi_R &= \left\{ \begin{array}{l} \text{ubicación}(\text{mineral}, \text{cavernas}) \\ \text{escasas}(\text{cavernas}) \\ \sim \text{enemigo}(\text{jugador}) \end{array} \right\} \\ \Delta_R &= \left\{ \begin{array}{l} \text{valioso}(R) \prec \text{difícil_encontrar}(R) \\ \text{difícil_encontrar}(R) \prec \text{ubicación}(R, U), \text{escasas}(U) \end{array} \right\} \end{aligned}$$

Además, considere el argumento de indagación:

$$\mathcal{D}_1 = \left\{ \begin{array}{l} \text{dar}(\text{mineral}, \text{jugador}), \\ \text{dar}(R, A) \prec \text{necesita}(R, A) \end{array} \right\},$$

$$\left\{ \left\{ \text{necesita}(\text{mineral}, \text{jugador}) \right\} \right\}$$

Los tres agentes se unen a la sesión $[ri_1, ri_1, ri_1]$ y adquieren \mathcal{D}_1 , el único argumento en el árbol de indagación. La primer ronda comienza $[ri_3]$, **peter** realiza el movimiento $\text{derrotar}(\mathcal{D}_1, \mathcal{D}_2) [ri_4]$, donde

$$\mathcal{D}_2 = \left\langle \begin{array}{l} \sim\text{dar}(\text{mineral}, \text{jugador}), \\ \left\{ \begin{array}{l} \sim\text{dar}(R, A) \prec \text{necesita}(R, A), \text{valioso}(R) \\ \text{valioso}(R) \prec \text{ubicación}(R, U), \text{peligrosas}(U) \\ \text{peligrosas}(U) \prec \text{peligro}(P, U) \end{array} \right\}, \\ \left\{ \begin{array}{l} \text{peligro}(\text{derrumbe}, \text{cavernas}) \end{array} \right\} \end{array} \right\rangle$$

y $\mathcal{D}_2 > \mathcal{D}_1$. Por lo tanto, \mathcal{D}_2 es vinculado debajo de \mathcal{D}_1 $\boxed{1}$ y tanto **egon** como **raymond** adquieren $\mathcal{D}_2 [ri_5]$. En su turno, **egon** realiza el movimiento $\text{derrotar}(\mathcal{D}_2, \mathcal{P}) [ri_4]$, donde

$$\mathcal{P} = \left\langle \begin{array}{l} \sim\text{peligrosas}(\text{cavernas}), \\ \left\{ \begin{array}{l} \sim\text{peligrosas}(U) \prec \text{peligro}(P, U), \text{equipamiento_seguridad}(P) \end{array} \right\}, \\ \left\{ \begin{array}{l} \text{peligro}(\text{derrumbe}, \text{cavernas}) \\ \text{equipamiento_seguridad}(\text{derrumbe}) \end{array} \right\} \end{array} \right\rangle$$

y $\mathcal{P} > \mathcal{D}_2$. Por ende, \mathcal{P} es vinculado debajo de \mathcal{D}_2 $\boxed{2}$ y **peter** y **raymond** proceden a adquirir $\mathcal{P} [ri_5]$. En su turno, **raymond** realiza el movimiento $\text{derrotar}(\mathcal{D}_1, \mathcal{D}_3) [ri_4]$, donde

$$\mathcal{D}_3 = \left\langle \begin{array}{l} \sim\text{dar}(\text{mineral}, \text{jugador}), \\ \left\{ \begin{array}{l} \sim\text{dar}(R, A) \prec \text{necesita}(R, A), \text{valioso}(R) \\ \text{valioso}(R) \prec \text{difícil_encontrar}(R) \\ \text{difícil_encontrar}(R) \prec \text{ubicación}(R, U), \text{escasas}(U) \end{array} \right\}, \\ \left\{ \begin{array}{l} \text{ubicación}(\text{mineral}, \text{cavernas}) \\ \text{escasas}(\text{cavernas}) \end{array} \right\} \end{array} \right\rangle$$

y $\mathcal{D}_3 > \mathcal{D}_1$. En consecuencia, \mathcal{D}_3 es vinculado debajo de \mathcal{D}_1 $\boxed{3}$ y tanto **peter** como **egon** adquieren $\mathcal{D}_3 [ri_5]$. Observe que **raymond** no hubiese podido construir \mathcal{D}_3 antes de adquirir la regla rebatible $\sim\text{dar}(R, A) \prec \text{necesita}(R, A), \text{valioso}(R)$ del argumento \mathcal{D}_2 . La primer ronda finaliza y una nueva comienza ya que los agentes no pasaron $[ri_{12}]$. En su turno, **peter** realiza el movimiento $\text{rechazar}(\mathcal{D}_3, \mathcal{E}) [ri_6]$, donde $\mathcal{E} = \langle \sim\text{difícil_encontrar}(\text{mineral}), \emptyset, \{ \sim\text{difícil_encontrar}(\text{mineral}) \} \rangle$, y \mathcal{D}_3 es desvin-

culado [4]. Luego, **egon** y **raymond** adquieren \mathcal{E} [ri₇]. En su turno, **egon** realiza el movimiento **derrotar**($\mathcal{D}_1, \mathcal{D}_4$) [ri₄], donde

$$\mathcal{D}_4 = \left\langle \begin{array}{l} \sim dar(mineral, jugador), \\ \left\{ \begin{array}{l} \sim dar(R, A) \prec necesita(R, A), enemigo(A) \end{array} \right\}, \\ \left\{ \begin{array}{l} necesita(mineral, jugador) \\ enemigo(jugador) \end{array} \right\} \end{array} \right\rangle$$

y $\mathcal{D}_4 > \mathcal{D}_1$. Por lo tanto, \mathcal{D}_4 es vinculado debajo de \mathcal{D}_1 en el árbol de indagación [5] y **peter** y **raymond** proceden a adquirir \mathcal{D}_4 [ri₅]. En su turno, **raymond** realiza el movimiento **objetar**($enemigo(jugador), \sim enemigo(jugador)$) por lo que es necesario llegar a un consenso sobre cuál de los hechos es aceptado y cuál es rechazado. El hecho aceptado es $CONSENSO(enemigo(jugador), \sim enemigo(jugador)) = \sim enemigo(jugador)$ [ri₈], \mathcal{D}_4 es desvinculado [6] y todos proceden a actualizar sus creencias según el consenso [ri₁₀]. La segunda ronda finaliza y una nueva comienza ya que los agentes no pasaron [ri₁₂]. Finalmente, durante la tercer ronda todos los agentes realizan el movimiento **pasar** [ri₁₁, ri₁₁, ri₁₁] y la sesión finaliza [ri₁₃]. ■

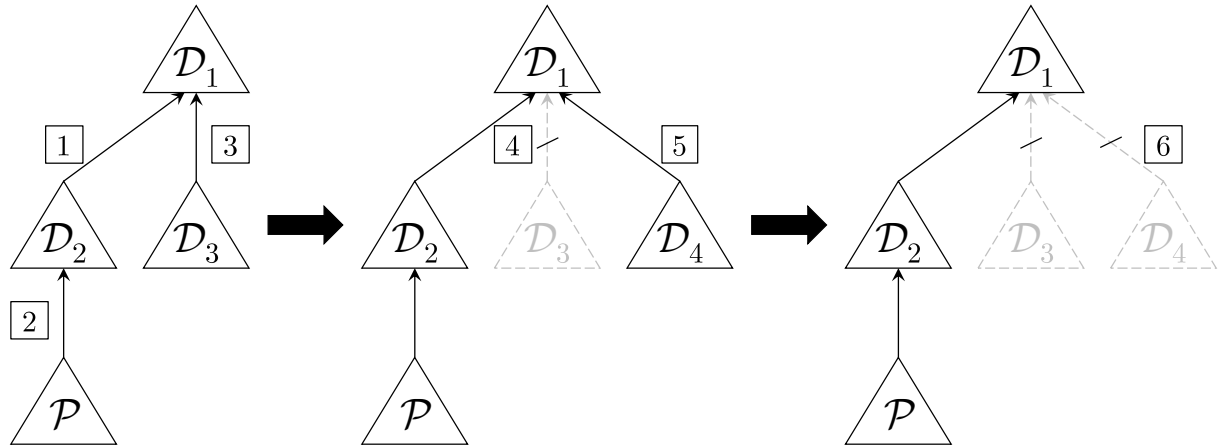


Figura 4.10: Evolución del árbol de indagación del Ejemplo 4.15.

En esta sección se presentó el núcleo de la contribución de este capítulo, cuyos resultados formales serán mostrados en la sección siguiente. Más adelante, en la Sección 4.4, se discutirán posibles extensiones para el formalismo.

4.3. Resultados formales

En esta sección, se presentarán resultados formales que garantizan que el protocolo y estrategia de diálogo de indagación definido por el sistema de transiciones $\Theta_I = \{ri_1, ri_2, ri_3, ri_4, ri_5, ri_6, ri_7, ri_8, ri_9, ri_{10}, ri_{11}, ri_{12}, ri_{13}\}$ se comporta de manera sensata.

Recuerde que, de acuerdo a la Definición 2.28, un estado e_a es alcanzable desde otro estado e_0 mediante un sistema de transiciones Θ si existe una secuencia finita de reglas de transición de Θ que, aplicadas en ese orden, hacen que e_0 evolucione a e_a . En esta sección, cuando se hable de un estado de sesión alcanzable y no se mencione cuál es el estado e_0 , implícitamente se hará referencia al estado de sesión inicial $e_0 = (\mathbb{I}, \emptyset, [], \emptyset)$. Además, implícitamente se hará referencia al sistema de transiciones Θ_I .

En primer lugar, se mostrará que el sistema de transiciones Θ_I se comporta de manera sensata incluso en dos situaciones extraordinarias: si todos los participantes abandonan la sesión y, por otro lado, si uno de los agentes rechaza el argumento de indagación que dio inicio a la sesión. La Proposición 4.1 garantiza que la sesión evolucionará a un estado final aún si todos los participantes la abandonan. Recuerde que cuando un agente abandona la sesión es eliminado del conjunto de participantes y de la cola de espera, *i. e.*, el segundo y cuarto componente del estado de sesión. Por lo tanto, un estado de sesión sin participantes es denotado $(\mathbb{I}, \emptyset, [R_1, \dots, R_n], \emptyset)$

Proposición 4.1 Sea $e_a = (\mathbb{I}, \emptyset, [R_1, \dots, R_n], \emptyset)$ con $n \geq 1$ un estado alcanzable y e_f un estado final, e_f es alcanzable desde e_a .

Demostración: ver Apéndice [pág. 136]. □

La Proposición 4.2 garantiza que la sesión podrá evolucionar a un estado final aún si un agente rechaza el argumento de indagación, lo cual causaría la desvinculación de la raíz del árbol de indagación y la eliminación de todos sus nodos.

Proposición 4.2 Sea $e_a = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle]], \mathbb{E})$ tal que $\mathbb{I} = \langle \emptyset, \emptyset \rangle$ un estado alcanzable y e_f un estado final, e_f es alcanzable desde e_a .

Demostración: ver Apéndice [pág. 137]. □

Una de las propiedades más importantes del sistema de transiciones Θ_I es que toda sesión finaliza bajo cierta condición razonable. Sin embargo, antes de probar esto se

mostrarán tres resultados auxiliares. En particular, el Lema 4.1 garantiza que desde todo estado de sesión alcanzable (excepto un estado en el que la primera ronda todavía no comenzó o un estado final) hay al menos una regla de transición aplicable aparte de ri_1 (unirse a la sesión) y ri_2 (abandonar la sesión). Esto implica que la ejecución de una sesión se detendrá sólo si evoluciona a un estado final.

Lema 4.1 Sea e_a un estado alcanzable tal que $e_a \neq (\mathbb{I}, \mathbb{Z}, [], \mathbb{E})$ con $|\mathbb{Z}| < 2$ y tal que $e_a \neq (\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n, [\text{fin}], \emptyset])$, existe al menos una regla de transición en $\Theta'_i = \{ri_3, ri_4, ri_5, ri_6, ri_7, ri_8, ri_9, ri_{10}, ri_{11}, ri_{12}, ri_{13}\}$ que es aplicable desde e_a .

Demostración: ver Apéndice [pág. 137]. □

El Lema 4.2 garantiza que si no se unen más agentes a la sesión luego de que un argumento es rechazado, éste no será vinculado nuevamente en el árbol de indagación.

Lema 4.2 Sea $e_a = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle]], \mathbb{E})$ un estado de sesión alcanzable, si la regla de transición $ri_1 \in \Theta_I$ no vuelve a ser aplicada entonces \mathcal{A} no será vinculado en el árbol de indagación nuevamente.

Demostración: ver Apéndice [pág. 138]. □

El lema anterior requiere que la regla de transición ri_1 no sea aplicada nuevamente luego del movimiento $\text{rechazar}(\mathcal{A}, \mathcal{B})$ ya que, de lo contrario, otro agente que puede construir \mathcal{A} podría unirse a la sesión y volver a vincularlo en el árbol de indagación. Esto se debe a que las razones de los rechazos (en este caso \mathcal{B}), no forman parte del árbol de indagación y por lo tanto no son adquiridas por los nuevos participantes mediante el operador ADQÁRBOL.

De forma similar, el Lema 4.3 garantiza que si no se unen más agentes a la sesión luego de que un hecho h es objetado y el consenso acepta \bar{h} y rechaza h , todo agente que creía en h abandona la sesión, o actualiza sus creencias y cree en h por el resto de la sesión.

Lema 4.3 Sea $e_a = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle OtroID, \text{objetar}(h, \bar{h}, \bar{h}) \rangle]], \mathbb{E})$ un estado de sesión alcanzable tal que $Z = (ID, \Pi_Z, \Delta_Z, >) \in \mathbb{Z}$ y $h \in \Pi_Z$, si la regla de transición $ri_1 \in \Theta_I$ no vuelve a ser aplicada entonces en todo estado alcanzable $e_b = (\mathbb{I}', \mathbb{Z}', \mathbb{R}', \mathbb{E}')$ desde e_a se cumple que si $Z' = (ID, \Pi_{Z'}, \Delta_{Z'}, >) \in \mathbb{Z}'$ entonces $\bar{h} \in \Pi_{Z'}$.

Demostración: ver Apéndice [pág. 139]. \square

El lema anterior también requiere que la regla de transición ri_1 no sea aplicada nuevamente luego del movimiento $\text{objetar}(h, \bar{h})$. De lo contrario, otro agente que cree en h podría unirse a la sesión y $\text{objetar} \bar{h}$, y el resultado del operador CONSENSO podría cambiar.

A continuación se presentará el antes mencionado Teorema 4.1, el cual garantiza que si en cierto punto de la sesión no se unen más agentes, la sesión eventualmente evolucionará a un estado final.

Teorema 4.1 Sea $e_a = (\mathbb{I}, \mathbb{Z}, \mathbb{R}, \mathbb{E})$ con $|\mathbb{Z}| \geq 2$ un estado de sesión alcanzable y e_f un estado de sesión final, si la regla de transición $ri_1 \in \Theta_I$ no es aplicada nuevamente entonces e_f es alcanzable desde e_a .

Demostración: ver Apéndice [pág. 139]. \square

Si los agentes pueden unirse a la sesión indefinidamente, nueva información de fuentes externas podría ser introducida a la sesión causando que el árbol de indagación sea modificado indefinidamente. Ni siquiera es necesaria una cantidad infinita de agentes para lograr esto: un único agente que repetidamente abandona la sesión, adquiere información externa, y vuelve a unirse podría causar el mismo efecto.

Por último, recuerde que según [WK95] un diálogo de indagación es exitoso si los participantes logran llegar a un acuerdo general sobre el tema en discusión. El Teorema 4.2 muestra que toda sesión desarrollada mediante el protocolo y estrategia de diálogo de indagación definido en este capítulo es exitosa en el contexto de DeLP: una vez finalizada la sesión, la marca de raíz del árbol de dialéctica construido para \mathcal{I} por cada uno de los participantes será la misma.

Teorema 4.2 Sea $e_f = (\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n, [\text{fin}], \emptyset)$ un estado de sesión final alcanzable tal que $\text{ETIQUETA}(\text{RAÍZ}(\mathbb{I})) = \mathcal{I}$, $\mathbf{Z} = (ID, \Pi_{\mathbf{Z}}, \Delta_{\mathbf{Z}}, >) \in \mathbb{Z}$ e $\mathbf{Y} = (ID', \Pi_{\mathbf{Y}}, \Delta_{\mathbf{Y}}, >) \in \mathbb{Z}$ dos agentes, y \mathcal{T}_1 y \mathcal{T}_2 los árboles de dialéctica para \mathcal{I} construidos por \mathbf{Z} e \mathbf{Y} , respectivamente, se cumple que $\text{MARCA}(\text{RAÍZ}(\mathcal{T}_1), \mathcal{T}_1) = \text{MARCA}(\text{RAÍZ}(\mathcal{T}_2), \mathcal{T}_2)$.

Demostración: ver Apéndice [pág. 140]. \square

4.4. Extensiones

En esta sección se presentarán brevemente tres posibles extensiones para el protocolo y estrategia definido en la sección anterior, las cuales proveen más versatilidad a cambio de algunas modificaciones en el sistema de transiciones Θ_I .

4.4.1. Indagación sobre un literal

En la sección anterior se supuso que el argumento de indagación ya fue propuesto antes de comenzar una sesión. Una posible extensión para el formalismo sería que, en vez de comenzar una sesión con un argumento de indagación, ésta comience con una consulta q . Luego, utilizando un nuevo tipo de movimiento (*e. g.*, **proponer**) los agentes podrían dar argumentos para q y para \bar{q} , creando así múltiples raíces de árboles de indagación. Una vez finalizada la sesión, se podría determinar si q está garantizado si existe al menos un árbol de indagación enraizado con un argumento para q que esté marcado como \textcircled{U} , de manera similar a la Definición 2.25.

El movimiento **proponer** podría realizarse incluso si la primer ronda ya comenzó, dado que adquirir nuevas creencias durante la sesión podría permitirle a un agente construir un nuevo argumento para q o para \bar{q} que antes no podía. De manera similar al movimiento **derrotar** un participante no podría **proponer** un argumento que utiliza hechos en contradicción con la evidencia de alguno de los árboles de indagación. En este caso, el agente debería primero **objetar** el hecho que considera contradictorio.

Incorporar esta extensión en el protocolo y estrategia de diálogo implicaría los siguientes cambios menores en el sistema de transiciones Θ_I :

- El primer componente de los estados de sesión debería ser un conjunto de árboles de indagación. En el estado inicial éste sería un conjunto vacío.
- Cuando un agente se une a la sesión, la regla de transición ri_1 debería aplicar el operador **ADQÁRBOL** a todos los árboles de indagación.
- La regla de transición ri_3 debería comenzar la primer ronda sólo si hay al menos dos participantes y hay al menos un árbol de indagación.
- Cuando se realiza un movimiento **derrotar**, la regla de transición ri_4 debería aplicar el operador **VINCULACIÓN** a todos los árboles de indagación.

- Cuando se realiza un movimiento **rechazar** o **objetar**, las reglas de transición ri_6 y ri_8 deberían aplicar el operador DESVINCULACIÓN a todos los árboles de indagación.

4.4.2. Construcción conjunta de argumentos

Recuerde que el movimiento **derrotar** permite contribuir argumentos vinculándolos como derrotadores en el árbol de indagación. Para poder hacer esto, es necesario que el agente que realiza el movimiento tenga en su base de creencias todas las reglas rebatibles y hechos necesarios para construir el derrotador. Por lo tanto, podría ocurrir que, a pesar de que entre varios agentes tengan todas las creencias necesarias para construir un argumento, éste nunca pueda ser construido y vinculado al árbol de indagación.

Considerando esto, se podría extender el protocolo y estrategia de diálogo para que los participantes también puedan contribuir reglas rebatibles que ayuden a otros agentes a construir nuevos argumentos derrotadores. Estas reglas rebatibles, para ser consideradas útiles y asistir en la construcción de argumentos válidos, deben cumplir ciertas restricciones. Por ejemplo, la cabeza de la regla, su complemento, y el complemento de los literales del cuerpo de la regla no pueden ser parte de la evidencia del árbol de indagación. De lo contrario, según las condiciones 3 y 4 de la Definición 2.13, un argumento que utilice dicha regla rebatible no podría ser construido en el contexto de esa sesión. Además, la cabeza de la regla tiene que ser el complemento de la cabeza de alguna regla rebatible de algún argumento en el árbol de indagación, o tiene que ser parte del cuerpo de otra regla rebatible previamente contribuida, creando así una “cadena” de reglas rebatibles. De esta manera, la regla rebatible contribuida podría ser utilizada para un ataque a un argumento en el árbol de indagación.

Incorporar esta extensión en el protocolo y estrategia de diálogo implicaría los siguientes cambios en el sistema de transiciones Θ_I : primero, el operador MOVIMIENTO debería considerar el nuevo tipo de movimiento con las condiciones antes mencionadas. Este requeriría que la secuencia de rondas \mathbb{E} sea un nuevo parámetro en el operador, ya que los agentes necesitarían poder verificar cuáles reglas rebatibles ya fueron contribuidas.

$$\text{MOVIMIENTO}(\mathbb{I}, \mathbb{R}, \mathbb{Z}) = \left\{ \begin{array}{l} \dots \\ \text{mostrar}(h \prec p_1, \dots, p_n), \dots \end{array} \right.$$

... si existe $h \prec p_1, \dots, p_n \in \Delta_Z$ tal que $h \notin \text{EVIDENCIA}(\mathbb{I})$, $\bar{h} \notin \text{EVIDENCIA}(\mathbb{I})$, $\bar{p}_i \notin \text{EVIDENCIA}(\mathbb{I})$ para todo $1 \leq i \leq n$, y $\text{mostrar}(h \prec p_1, \dots, p_n) \notin^* R \in^* \mathbb{R}$, y existe $\mathfrak{N}_{\mathcal{A}} \in \text{Nodos}$ con $\mathcal{A} = \langle a, \mathbb{R}, \mathbb{H} \rangle$ tal que $\bar{h} \prec p'_1, \dots, p'_m \in \mathbb{R}$ o existe $\langle ID, \text{mostrar}(c \prec p''_1, \dots, p''_s) \in^* R' \in^* \mathbb{R}$ tal que $h = p''_j$ para cualquier j ($1 \leq j \leq s$).

Además, dos reglas de transición adicionales deberían incorporarse al sistema de transiciones Θ_I : una para poder realizar el movimiento **mostrar** y otra para la adquisición de las reglas compartidas mediante dicho movimiento.

$$ri_{14} : \frac{ID \in \mathbb{E} \wedge \text{AGENTE}(ID, \mathbb{Z}) = \mathbb{Z} \wedge \text{MOVIMIENTO}(\mathbb{I}, \mathbb{Z}) = \text{mostrar}(h \prec p_1, \dots, p_n)}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon], \mathbb{E}] \rightarrow (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{mostrar}(h \prec p_1, \dots, p_n) \rangle], \mathbb{E} \setminus \{ID\})}$$

La regla de transición ri_{14} , a diferencia de ri_4 (movimiento **derrotar**), no modifica el árbol de indagación. Los agentes utilizarán el movimiento **mostrar** para contribuir reglas rebatibles relevantes, las cuales serán adquiridas por todos los participantes mediante la regla de transición ri_{15} , hasta que eventualmente uno de ellos pueda construir un nuevo argumento y realizar el movimiento **derrotar** correspondiente.

$$ri_{15} : \frac{\mathbb{Z}' = \{\mathbb{Z}' \mid (ID, \Pi_{\mathbb{Z}}, \Delta_{\mathbb{Z}}, >) \in \mathbb{Z} \wedge \mathbb{Z}' = (ID, \Pi_{\mathbb{Z}}, \Delta_{\mathbb{Z}} \cup \{h \prec p_1, \dots, p_n\}, >)\}}{(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle OtroID, \text{mostrar}(h \prec p_1, \dots, p_n) \rangle], \mathbb{E}] \rightarrow (\mathbb{I}, \mathbb{Z}', [\dots, [\dots, \langle OtroID, \text{mostrar}(h \prec p_1, \dots, p_n) \rangle], \varepsilon], \mathbb{E})}$$

4.4.3. Determinismo

Como el lector puede haber notado, el sistema de transiciones Θ_I fue definido de manera no-determinista. Esto se debe a que se evitó imponer restricciones innecesarias a la estrategia para permitir la implementación de las reglas de transición más adecuada para el dominio de aplicación.

En primer lugar, el no-determinismo puede aparecer entre las reglas de transición ri_4 , ri_6 , ri_8 (o ri_9), y ri_{11} dado que:

1. Pueden haber múltiples agentes esperando su turno.
2. El agente seleccionado puede tener múltiples movimientos disponibles.

En el caso de la razón 1, el determinismo puede ser forzado seleccionando de la cola de espera, por ejemplo, al agente con el menor identificador (si son números enteros) o al agente con mayor prioridad en base a cierto criterio. En el caso de la razón 2, el determinismo puede ser forzado asignándoles a los tipos de movimiento diferentes prioridades. En la siguiente sección se compararán empíricamente diferentes combinaciones de prioridades entre los movimientos **derrotar**, **rechazar**, y **objetar**, las cuales implican sesiones más o menos cortas.

El no-determinismo también puede aparecer entre ri_1 (unirse a la sesión), ri_2 (abandonar la sesión) y el resto de las reglas de transición, dado que ambas pueden ser ejecutadas en cualquier momento de la sesión sin importar el estado actual. En este caso, el determinismo puede ser forzado, por ejemplo, dándole prioridad a ri_2 por sobre ri_1 , y a ri_1 por sobre el resto de las reglas de transición.

4.5. Comparación empírica de criterios de selección de movimientos

En esta sección se comparará empíricamente el desempeño de tres criterios de selección de movimientos que determinan la forma en la que los agentes eligen entre sus movimientos disponibles durante sus turnos. El desempeño de un criterio en una sesión será medido una vez que la sesión finaliza, *i. e.*, existe una secuencia finita de reglas de transición que hacen evolucionar la sesión desde el estado inicial e_0 hasta un estado final e_f . La secuencia de estados comprendidos entre e_0 y e_f será llamada *ejecución de sesión* (ver Definición 2.29).

Siempre que un argumento es desvinculado del árbol de indagación mediante un movimiento **rechazar** o **objetar**, los subárboles enraizados con nodos que están etiquetados con uno de sus subargumentos² son eliminados. Por lo tanto, si la meta de los agentes es reducir la cantidad de movimientos que son realizados durante la sesión, los movimientos **derrotar** que vinculen nodos en aquellos subárboles son, en ese sentido, “desperdiciados”.

Claramente los agentes no pueden saber por adelantado cuáles nodos serán desvinculados del árbol de indagación. Por lo tanto, es imposible predecir la ejecución de sesión optimal en términos de la menor cantidad de movimientos realizados sin hacer un análisis

²En DeLP todo argumento es subargumento de sí mismo.

exhaustivo previo de todas las creencias de los participantes, lo cual sería contraproducente considerando el costo computacional y la violación de privacidad. A pesar de esto, en esta sección se probará empíricamente que hay un criterio de selección de movimientos que los agentes pueden utilizar para reducir la cantidad de movimientos realizados durante la ejecución de una sesión.

Un agente que utiliza el *Criterio de Selección de Objetar Primero* (CSOP) prioriza realizar el movimiento **objetar** por sobre **rechazar**, y el movimiento **rechazar** por derrotar. Por el contrario, un agente que utiliza el *Criterio de Selección de Derrotar Primero* (CSDP) prioriza realizar el movimiento **derrotar** por sobre **rechazar**, y el movimiento **rechazar** por sobre **objetar**. Por último, un agente que utiliza el *Criterio de Selección Aleatorio* (CSA) selecciona aleatoriamente entre sus movimientos disponibles. Para reducir la cantidad de movimientos realizados durante la ejecución de una sesión, los agentes deberían utilizar CSOP. La intuición detrás de esto es que el movimiento **rechazar** desvincula un único subargumento del árbol de indagación, mientras que el movimiento **objetar** puede desvincular muchos argumentos si el consenso rechaza el hecho que fue objetado (el cual forma parte de la evidencia del árbol de indagación).

El desempeño de un criterio de selección en una sesión es medido en términos de la *cantidad de movimientos efectivos* (CME), *i. e.*, la cantidad de movimientos diferentes de **pasar** que fueron realizados durante la ejecución de sesión correspondiente considerando que todo agente utilizó ese criterio. Un criterio de selección C1 tiene mejor desempeño en una sesión que otro criterio C2 si la CME obtenida con C1 es menor que la CME obtenida con C2. Con el fin de medir y comparar el desempeño de los tres criterios, se realizaron dos experimentos diferentes en los cual se generaron agentes que indagan en sesiones. Los parámetros involucrados en los experimentos fueron:

- **CantidadArgumentos**: La cantidad de argumentos diferentes que podrían ser vinculados en el árbol de indagación. Valores utilizados: 100, 150, 200, 250, 300.
- **PorcentajeMáximoObjetar** (PMO): El porcentaje máximo de argumentos (con respecto a **CantidadArgumentos**) que contienen un hecho que un agente puede objetar. Por ejemplo, si **PorcentajeMáximoObjetar** es 3 y **CantidadArgumentos** es 100, cada agente puede objetar como máximo 3 argumentos. Valores utilizados: 1, 3, 5.
- **ProbabilidadConstrucción**: Para cada agente y cada argumento, la probabilidad de que ese agente pueda construir ese argumento. Valores utilizados: 5, 15, 25, 35, 45, 55, 65, 75.

En ambos experimentos, para cada combinación de parámetros se generaron 1000 conjuntos de 10 agentes diferentes. Cada conjunto de agentes fue puesto a discutir en tres sesiones diferentes (para evaluar los tres criterios), y las CME fueron obtenidas. Finalmente, para cada combinación de parámetros la CME promedio entre las 1000 ejecuciones correspondientes a cada criterio fue calculada: CME_{CSOP} , CME_{CSDP} , y CME_{CSA} . Se utilizaron 10 agentes por sesión ya que los resultados no se vieron afectados después de utilizar diferentes cantidades de agentes en numerosas simulaciones. Cada conjunto de agentes fue generado de la siguiente manera:

1. Se generó un conjunto de argumentos correspondiente a **CantidadArgumentos**.
2. A cada agente se le asignó una cantidad de hechos en contradicción con aquellos utilizados por los argumentos del conjunto, de acuerdo a **PorcentajeMáximoObjetar**. Estos hechos fueron la fuente de los movimientos **objetar** en la simulación.
3. Los argumentos del conjunto se distribuyeron a los agentes utilizando round-robin, siempre manteniendo sus bases de creencias consistentes. Es decir, a los agentes no se les asignaron argumentos con hechos en contradicción con sus creencias.
4. Cada argumento fue replicado en algunos de los demás agentes de acuerdo a **ProbabilidadConstrucción**. Por ejemplo, dados tres agentes $z1$, $z2$ y $z3$, tres argumentos \mathcal{A}_1 , \mathcal{A}_2 y \mathcal{A}_3 , y **ProbabilidadConstrucción** igual a 35 %, \mathcal{A}_1 era construido por $z1$, tenía un 35 % de probabilidad de ser construido por $z2$, y un 35 % de probabilidad de ser construido por $z3$; \mathcal{A}_2 era construido por $z2$, tenía un 35 % de probabilidad de ser construido por $z1$, y un 35 % de probabilidad de ser construido por $z3$, y así sucesivamente. Los argumentos fueron distribuidos así ya que es poco realista considerar que cada argumento es conocido por un único agente en el sistema.

Respecto a la implementación del operador CONSENSO, los agentes votaban por el literal que querían aceptar (a o \bar{a}) con una probabilidad del 50 % cada uno, y las decisiones finales eran tomadas por mayoría simple.

En el primer experimento (ver Figura 4.11), el gráfico de la izquierda ilustra la variación de la ganancia de desempeño normalizada (%) de CSOP por sobre CSA (*i. e.*, $(CME_{CSA} - CME_{CSOP}) / (CME_{CSA} + CME_{CSOP})$) (eje vertical) con respecto al parámetro **ProbabilidadConstrucción** (eje horizontal). En cambio, el gráfico de la derecha ilustra la variación de la ganancia de desempeño normalizada de CSOP por sobre CSDP con respecto a **ProbabilidadConstrucción**. En ambos gráficos la curva corresponde a un valor diferente (1 %, 3 % y 5 %) de **PorcentajeMáximoObjetar** (PMO), y **CantidadArgumentos**

fue establecido en 100. Observe que, en ambos gráficos, CSOP tiene un mejor desempeño que el otro criterio para cada combinación de parámetros y, como era de esperarse, la diferencia de desempeño se vuelve más significativa a medida que incrementa `PorcentajeMáximoObjetar`. Mayores valores de `ProbabilidadConstrucción` también incrementan la ganancia de desempeño de CSOP por sobre el otro criterio. Esto se debe a que, mientras mayor es `ProbabilidadConstrucción`, más argumentos tienen los agentes, hay más movimientos derrotar disponibles, y menores son las probabilidades de que CSA seleccione un movimiento diferente de `objetar`.

La ganancia de desempeño de CSOP por sobre CSDP es mayor a la ganancia de desempeño de CSOP por sobre CSA dado que darle prioridad a los movimientos derrotar causa que las sesiones sean mucho más largas.

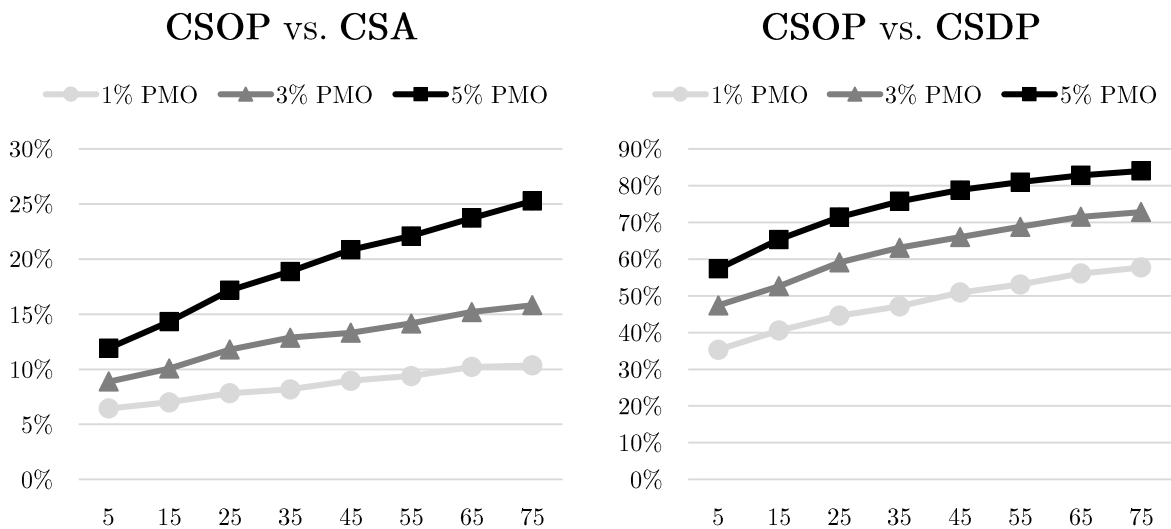


Figura 4.11: Variación de la ganancia de desempeño normalizada de CSOP por sobre CSA y de CSOP por sobre CSDP, con respecto a `ProbabilidadConstrucción`.

En el segundo experimento (ver Figura 4.12), el gráfico de la izquierda ilustra la variación de ganancia de desempeño normalizada de CSOP por sobre CSA (eje vertical) con respecto a `CantidadArgumentos` (eje horizontal). En cambio, el gráfico de la derecha ilustra la variación de ganancia de desempeño de CSOP por sobre CSDP con respecto a `CantidadArgumentos`. En ambos gráficos, cada curva corresponde a un valor diferente de `PorcentajeMáximoObjetar` (PMO), y `ProbabilidadConstrucción` fue fijado en 35%. Las conclusiones que fueron obtenidas en el primer experimento también pueden observarse

en estos dos gráficos: mayores valores de `CantidadArgumentos` incrementan la ganancia de desempeño de CSOP por sobre el otro criterio. Esto se debe a que, mientras mayor es `CantidadArgumentos`, más larga se vuelve la sesión a menos que los movimientos `objetar` tengan prioridad sobre los otros.

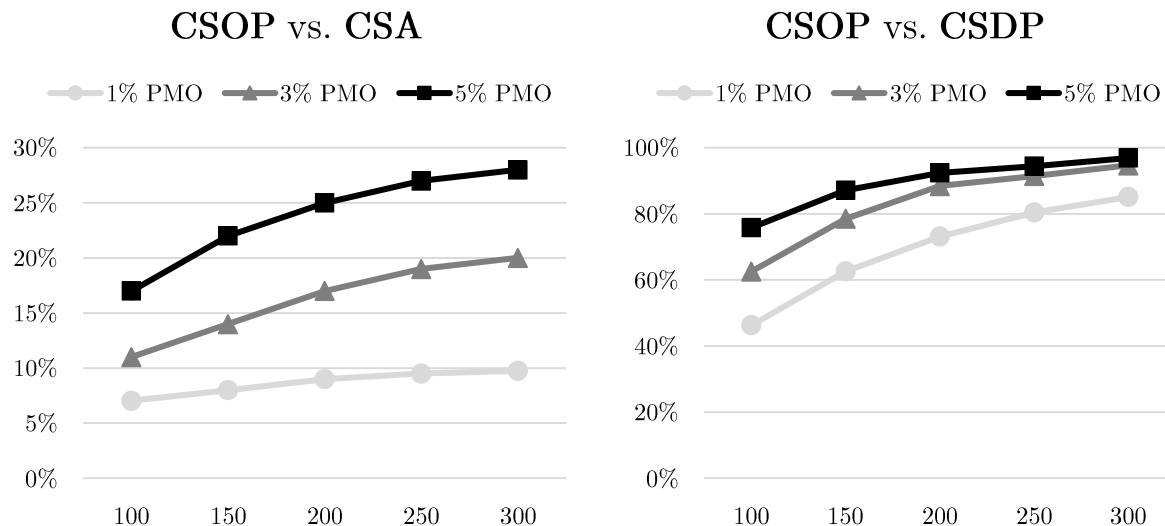


Figura 4.12: Variación de la ganancia de desempeño normalizada de CSOP por sobre CSA y de CSOP por sobre CSDP, con respecto a `CantidadArgumentos`.

4.6. Trabajo relacionado

Uno de los tipos de diálogos definidos por [WK95] es el *diálogo de indagación* (en inglés, *inquiry*), en el que un grupo de agentes busca llegar a un acuerdo general sobre la respuesta a una pregunta. Esto se hace mediante un proceso acumulativo de argumentos que finaliza cuando los participantes llegan a una conclusión que, desde su punto de vista, es indudable. Es necesario que los participantes sean imparciales, y que contribuyan todas las conclusiones intermedias y premisas correspondientes. Un diálogo de indagación es considerado exitoso si se logra llegar a un acuerdo general sobre el tema en discusión.

A diferencia de los trabajos de la literatura en los que se define un protocolo de diálogo y una estrategia de indagación modular, el formalismo definido en este capítulo es un *protocolo y estrategia* de diálogo de indagación dado que ambos aspectos están fusionados

en un sistema de transiciones que determinan tanto la coherencia de los diálogos como el comportamiento de los agentes. Como se mencionó en la Sección 3.6, definir un protocolo y estrategia específico para un tipo de diálogo permite concentrarse en los aspectos positivos que se desean lograr sin que la estrategia se vea limitada por el protocolo.

En la primera parte de esta sección se presentarán los protocolos y estrategias de diálogo de indagación de la literatura. Luego, se realizará un análisis comparativo entre ellos y el protocolo y estrategia de diálogo de indagación definido en este capítulo (el cual será denotado [CAP4]). Todas las características comparadas serán sintetizadas en la tabla de la Figura 4.13.

En la Sección 3.6 se presentó [PWA02], un protocolo de diálogo para dos agentes que representan su conocimiento con fórmulas de la lógica proposicional y razonan utilizando el sistema argumentativo propuesto en [AC98]. Además de proveer una estrategia de diálogo de búsqueda de información también proveen una de indagación, en la cual los dos participantes se turnan para afirmar fórmulas proposicionales con el fin de componer una cadena de implicaciones hasta obtener una prueba para p que sea aceptable para ambos, o hasta que ninguno tenga más fórmulas para afirmar. Cada vez que un agente afirma una fórmula, el otro puede aceptarla o desafiarla (dependiendo de su actitud de aceptación) para que muestre explícitamente el argumento que la concluye. El problema con esta estrategia es que puede terminar sin éxito a pesar de que ambos agentes tengan suficiente conocimiento colectivo para probar p . Esto se solucionaría agregando backtracking al protocolo, o permitiendo que los agentes puedan afirmar todas las reglas de inferencia que sean relevantes en cualquier punto del diálogo.

En la Sección 3.6 también se presentó [FT11], otro protocolo de diálogo para dos agentes a_1 y a_2 en el que se utiliza ABA como formalismo de representación de conocimiento y razonamiento, *i. e.*, tanto a_1 como a_2 están representados por un framework ABA. Como ABA no considera información estricta, es posible hablar de la unión de a_1 y a_2 , denotada \mathcal{F}_J , la cual representa las creencias colectivas de los participantes. Los autores, además de proveer dos estrategias de diálogo de búsqueda de información, definen dos subtipos de diálogo de indagación [FT12, FT15], cada uno con su propia estrategia:

- *I-Type I*: Es incierto si χ es aceptable en \mathcal{F}_J y el objetivo es evaluar la aceptabilidad de χ en \mathcal{F}_J .
- *I-Type II*: Los agentes a_1 y a_2 no pueden construir argumentos para χ y el objetivo es evaluar si a partir de \mathcal{F}_J puede construirse un argumento para χ .

Los autores prueban que los diálogos de tipo I-Type I son sensatos y completos sólo si los agentes utilizan la función de movimiento de estrategia ϕ_h *directa* (en inglés, *thorough*), la cual implica que hagan todas las declaraciones legales que puedan. Además, prueban que los diálogos de tipo I-Type II son sensatos y completos sólo si los agentes utilizan la función de movimiento de estrategia ϕ_{nh} , la cual implica que hagan todas las declaraciones legales que puedan que no involucren contrarios.

En [BH07, BH09] se define un protocolo de diálogo para dos agentes que también utilizan DeLP como formalismo de representación de conocimiento y razonamiento. Sin embargo, las bases de creencias sólo están compuestas de reglas rebatibles y presuposiciones (reglas rebatibles con cuerpo vacío) ya que suponen que todo el conocimiento es rebatible. Además, los autores definen dos subtipos de diálogo de indagación, cada uno con su propia estrategia:

- *Diálogos de indagación de argumento* (en inglés, *argument inquiry dialogues*): Permiten a dos agentes compartir creencias para construir conjuntamente argumentos para una afirmación específica que ninguno de ellos podría construir utilizando sólo sus propias creencias. Por ejemplo, un agente que quiere construir un argumento para ϕ puede iniciar un diálogo de indagación de argumento con la regla rebatible $\alpha_1 \wedge \dots \wedge \alpha_n \rightarrow \phi$. Si los dos agentes logran dar argumentos para cada uno de los elementos α_i ($i \geq 1 \geq n$), entonces es posible construir un argumento para ϕ .
- *Diálogos de indagación de garantía* (en inglés, *warrant inquiry dialogues*): Permiten a dos agentes determinar la aceptabilidad de un argumento compartiendo argumentos para construir conjuntamente un árbol de dialéctica que ninguno de ellos podría construir utilizando sólo sus propias creencias.

Los diálogos de indagación de argumento suelen estar embebidos dentro de diálogos de indagación de garantía. De lo contrario, podrían pasarse por alto argumentos útiles que involucran creencias no expresadas por los agentes. Los autores prueban que los diálogos generados a través de la estrategia de diálogo de indagación de garantía son sensatos y completos. Para esto, los autores prueban que el resultado de todo diálogo de indagación de garantía es igual al resultado que se obtendría a partir de la unión de las bases de creencias de los agentes involucrados. Esta es, en cierto sentido, la situación “ideal” donde no hay restricciones en la compartición de creencias. Sin embargo, dicha unión de bases de creencias sólo es posible ya que los agentes no tienen información estricta.

En [TGKIS08, Thi09] se define un protocolo de diálogo para múltiples agentes que también utilizan DeLP como formalismo de representación de conocimiento y razonamiento. Dada una consulta, los participantes pueden dar argumentos y contra-argumentos con el objetivo de llegar a una conclusión compartida. Cada agente tiene una base de creencias local compuesta de reglas rebatibles, la cual representa su propio conocimiento (incierto). Además, hay una base de creencias global compuesta de hechos y reglas estrictas, la cual representa el conocimiento común a todos los agentes. La base de creencias global no debe ser contradictoria ya que dicha información es considerada indiscutible por los agentes. El protocolo propuesto funciona de la siguiente manera:

1. Un moderador presenta una consulta, representada por un literal h .
2. Los agentes proponen argumentos iniciales para h o \bar{h} , los cuales se convierten en raíces de diferentes árboles de dialéctica.
3. Los agentes proponen contra-argumentos a los argumentos de otros agentes, que son válidos en la líneas de argumentación correspondientes.

El paso 2 se diferencia de la extensión propuesta en la Sección 4.4 en que, una vez que el moderador recibe los argumentos raíz iniciales y construye los árboles de dialéctica, los participantes ya no pueden proponer nuevos argumentos raíz. Además, en el paso 3, los participantes no están obligados a dar todos sus argumentos válidos. El comportamiento de los agentes está determinado por ciertas funciones que permiten modelar distintos tipos de estrategias de diálogo dependiendo de cómo sean instanciadas. Esto implica que, siguiendo la clasificación de diálogos provista por [WK95], se podría modelar una estrategia de diálogo indagación si dichas funciones se instancia de manera que los participantes den todos los argumentos y contra-argumentos válidos que pueden construir. Sin embargo, los agentes no podrían ser imparciales ya que el protocolo no les permite dar contra-argumentos para sus propios argumentos. Por lo tanto, según [WK95], considerarla una estrategia de diálogo de indagación es controversial.

En [TGKIS08] los autores también definen *colaboraciones* (en inglés, *collaborations*) que permiten a los agentes compartir sus creencias con otros para construir conjuntamente argumentos. En este caso, la argumentación ya no se realiza en un proceso iterativo, sino que los árboles de dialéctica simplemente se generan a partir de la unión de la base de creencias global y las bases de creencias locales de los agentes involucrados. Esto es posible ya que el formalismo no permite a los agentes tener sus propias bases de creencias con

información estricta, y por lo tanto no pueden surgir contradicciones a partir de la unión de las bases.

La tabla de la Figura 4.13 compara las características del protocolo y estrategia de diálogo de indagación definido en este capítulo ([CAP4]) con aquellas de los presentes la literatura. En particular, las primeras ocho filas son características deseables, mientras que las últimas dos podrían ser beneficiosas o no dependiendo del dominio de aplicación.

	[PWA02]	[BH07] [BH09]	[TGKIS08] [Thi09]	[FT12] [FT15]	[CAP4] (Capítulo 4)
Éxito garantizado	×	✓	✓	✓	✓
Hechos	×	×	✓	×	✓
Hechos propios	N/A	N/A	×	N/A	✓
Reglas estrictas	×	×	✓	×	×
Reglas estrictas propias	N/A	N/A	×	N/A	N/A
Tres o más agentes	×	×	✓	×	✓
Conjunto de participantes dinámico	N/A	N/A	×	N/A	✓
Construcción conjunta de argumentos	✓	✓	✓*	✓	✓
Efectiviza incorporación de creencias	×	×	×	×	✓
Argumentación exhaustiva	×	✓	✓*	✓	✓

Figura 4.13: Comparación de los protocolos y estrategias de diálogo de indagación.

Respecto a la característica *éxito garantizado*, el único que no la cumple es [PWA02] dado que los autores prueban que sus diálogos de indagación pueden terminar sin éxito a pesar de que ambos agentes tengan suficiente conocimiento colectivo para lograrlo.

Respecto a la representación de información estricta, ni [PWA02], [FT12, FT15] o [BH07, BH09] permiten a los agentes tener hechos o reglas estrictas en sus bases de creencias. En el caso de [PWA02] y de [FT12, FT15] esto es una consecuencia directa de los formalismos de representación de conocimiento utilizados. En cambio, en [BH07, BH09] se

optó por utilizar DeLP con sólo información rebatible dado que su motivación es el dominio de la medicina, el cual argumentan que está en constante cambio. En [TGKIS08, Thi09] se utiliza DeLP con hechos y reglas estrictas, pero sólo en una base de creencias global y consistente. Es decir, no pueden surgir contradicciones durante un diálogo ya que los participantes no pueden tener sus propios hechos o reglas estrictas. Si bien en [CAP4] se permite a los agentes tener sus propios hechos, se optó por utilizar DeLP sin reglas estrictas. Como se mencionó en la Sección 3.6, el tratamiento de reglas estrictas requiere de un análisis similar al de revisión de creencias en lógica proposicional, donde para actualizar Π incorporando una creencia α deben eliminarse todas las pruebas estrictas de $\bar{\alpha}$. Este tema ya fue estudiado y analizado en [FKIS02] y está fuera del foco de esta tesis.

Respecto a la cantidad de agentes que pueden participar de un diálogo, sólo en [CAP4] y en [TGKIS08, Thi09] se permiten diálogos entre tres o más agentes. Sin embargo, [CAP4] se destaca por permitirles unirse y abandonar la sesión libremente en cualquier momento.

Todos los formalismos permiten la construcción conjunta de argumentos, *i. e.*, los agentes pueden combinar reglas y premisas entre ellos para construir argumentos que no podrían individualmente. En [CAP4] esto es posible sólo si se utiliza la extensión definida en la Sección 4.4.2, la cual permite a los agentes compartir reglas rebatibles relevantes a la sesión. En el caso particular de [TGKIS08, Thi09], se le asignó \checkmark^* ya que la construcción conjunta de argumentos se realiza uniendo las bases de creencias de todos los participantes, lo cual podría ser una solución insensata o inviable si las bases de creencias son muy extensas o contienen información privada.

Por último, es importante comparar dos aspectos esenciales del funcionamiento de los formalismos. Primero, sólo en [CAP4] se efectiviza la adquisición de creencias. En otras palabras, en los trabajos de la literatura la información comunicada por los agentes queda almacenada en una estructura auxiliar: el *commitment store* en el caso de [PWA02] y de [BH07, BH09], un conjunto de árboles de dialéctica en el caso de [TGKIS08, Thi09], y el framework ABA \mathcal{F}_δ en el caso de [FT12, FT15]. Dichos protocolos no especifican qué es lo que hacen los agentes con esa información una vez finalizado el diálogo. Segundo, [PWA02] es el único formalismo en el que los agentes no realizan una argumentación exhaustiva, *i. e.*, no comparten toda la información relevante que tienen para contribuir. En el caso particular de [TGKIS08, Thi09] se le asignó \checkmark^* dado que, aunque se instancien las funciones para que los agentes compartan todos los argumentos que tienen para contribuir, el protocolo no les permite dar contra-argumentos para sus propios argumentos.

4.7. Conclusiones

En este capítulo se presentó un protocolo y estrategia de diálogo de indagación en el que los agentes utilizan DeLP como formalismo de representación de conocimiento y razonamiento. La interacción entre los agentes ocurre durante una sesión, la cual comienza cuando se propone un argumento de indagación sobre el cual se desea llegar a un acuerdo general, y finaliza cuando ningún participante tiene algo más para contribuir.

Los agentes pueden contribuir a la sesión derrotando al argumento de indagación con contra-argumentos, derrotando a esos contra-argumentos con otros contra-argumentos, y así sucesivamente. Siempre que un argumento es compartido, los participantes adquieren dicha información y la combinan con sus creencias previas, permitiéndoles potencialmente construir nuevos argumentos para contribuir. Además, los agentes pueden rechazar argumentos que consideran contradictorios o no-minimales u objetar hechos en contradicción con sus creencias. La meta de los participantes no es ganar una discusión o convencer a los otros de algo, sino contribuir imparcialmente todas sus creencias relevantes hasta que se llegue a un acuerdo general sobre el argumento de indagación.

El protocolo y estrategia de diálogo de indagación que se definió en este capítulo incluye cuatro mejoras respecto a los trabajos similares presentes en la literatura. Primero, se consideró que los agentes podrían tener hechos en contradicción con las creencias de los demás. Por lo tanto, si un agente informa que cree en un hecho en contradicción a otro hecho que fue utilizado como premisa durante la sesión, los participantes deben llegar a un consenso sobre cuál de los dos prevalece y deben actualizar sus creencias de acuerdo al consenso. Segundo, las sesiones no se limitaron a sólo dos agentes. Tercero, los participantes pueden optar por abandonar la sesión si no están dispuestos a seguir el protocolo, *e. g.*, cuando no quieren actualizar sus creencias luego de un consenso o cuando no quieren compartir cierta pieza de información privada. Por último, nuevos agentes pueden unirse libremente a la sesión incluso cuando ya ha comenzado.

Algunos aspectos del protocolo y estrategia de diálogo, como el operador CONSENSO, la selección del próximo agente que debe realizar un movimiento y el orden de prioridad entre las reglas de transición, no fueron especificados para evitar imponer restricciones innecesarias al formalismo. De esta manera, pueden utilizarse las implementaciones más adecuadas para el dominio de aplicación.

Este protocolo y estrategia de diálogo fue definido utilizando un sistema de transiciones con el objetivo formalizar la interacción entre los agentes y luego probar sus propiedades. En particular, considerando la noción de éxito dada por [WK95] para los diálogos de indagación, se probó que todo diálogo desarrollado mediante el protocolo y estrategia definido en este capítulo es exitoso en el contexto de DeLP: una vez finalizada una sesión, la marca de raíz del árbol de dialéctica construido para el argumento de indagación por cada uno de los participantes será la misma.

Finalmente, se comparó empíricamente el desempeño de tres criterios de selección de movimientos que determinan la forma en la que los agentes eligen entre sus movimientos disponibles durante sus turnos (CSOP, CSDP y CSA). Esto se logró realizando dos experimentos diferentes en los cuales múltiples sesiones con diferentes parámetros fueron ejecutadas y medidas. De estos experimentos se concluyó que para reducir la longitud de una sesión los agentes deberían usar el CSOP, dándole prioridad al movimiento **objetar** por sobre **rechazar**, y al movimiento **rechazar** por sobre **derrotar**.

Capítulo 5

Conclusiones y trabajo a futuro

En esta tesis se definieron dos protocolos y estrategias de diálogo para dos tipos de diálogo colaborativo-argumentativos en sistemas multi-agente: *consulta a experto* e *indagación*. En ambas propuestas, los agentes utilizan DeLP como formalismo de representación de conocimiento y razonamiento, y su interacción fue definida formalmente utilizando un sistema de transiciones.

En los diálogos de consulta a experto un agente es un experto en cierto dominio de conocimiento y el otro le consulta sobre un tema de ese dominio, el cual desconoce o conoce parcialmente, con el fin de adquirir información. Actualmente no existen otras estrategias de diálogo de consulta a experto en la literatura, y aquellas similares no consideran que el agente consultante podría tener hechos en contradicción con las creencias del agente experto, ni consideran que ambos agentes podrían tener diferentes preferencias entre argumentos.

Por otra parte, en los diálogos de indagación un grupo de agentes busca llegar a un acuerdo general sobre la respuesta a una pregunta, siendo imparciales y contribuyendo toda la información relevante que tengan. Si bien en la literatura existen otras estrategias de diálogo de indagación, éstas no consideran que los agentes podrían tener hechos en contradicción con las creencias de los demás, y las pocas que no se limitan a sólo dos agentes no permiten que el conjunto de participantes varíe durante del diálogo.

En los protocolos y estrategias de diálogo que se definieron en esta tesis se confrontaron dichas limitaciones para proveer una mayor versatilidad que las propuestas ya existentes.

En particular, en el protocolo y estrategia de diálogo de consulta a experto que presentó en el Capítulo 3, la sesión entre ambos agentes comienza cuando el agente consultante realiza una consulta al agente experto. En respuesta, si el experto tiene suficientes creencias para concluir el literal de la consulta o su complemento, elige un argumento que garantiza el literal correspondiente para enviar al consultante como justificación. El consultante considera al otro agente un experto en el tema y, por lo tanto, su meta es aceptar dicha justificación. Sin embargo, como podría tener argumentos que derroten a la justificación, tal vez necesite realizar preguntas de preferencia y preguntas de negador al experto para que éste le envíe sus preferencias entre los argumentos involucrados o argumentos a favor de la justificación, respectivamente. Además, el consultante podría tener derrotadores para dichos argumentos a favor, derrotadores para nuevos argumentos a favor, y así sucesivamente, creando un proceso iterativo de preguntas y respuestas que refinan sus creencias sobre la consulta adoptando información relevante del experto. Durante este proceso, el agente experto envía sólo la información necesaria para lograr la meta, y el agente consultante evita la eliminación innecesaria de creencias que son válidas desde la perspectiva del experto.

A diferencia de los protocolos y estrategias de diálogo de búsqueda de información presentes en la literatura, se consideró que el consultante podría tener hechos en contradicción con las creencias del experto sobre el tema consultado y que las preferencias entre argumentos de ambos agentes podrían ser diferentes. Por lo tanto, si el consultante está comprometido a lograr su meta, tal vez necesite actualizar sus preferencias y eliminar hechos de su base que—desde la perspectiva del experto—son inválidos.

Por otra parte, en el protocolo y estrategia de diálogo de indagación que presentó en el Capítulo 4, la sesión entre los agentes comienza cuando se propone un argumento de indagación sobre el cual se desea llegar a un acuerdo general, y finaliza cuando ningún participante tiene algo más para contribuir. Los agentes pueden contribuir a la sesión derrotando al argumento de indagación con contra-argumentos, derrotando a esos contra-argumentos con otros contra-argumentos, y así sucesivamente. Siempre que un argumento es compartido, los participantes adquieren dicha información y la combinan con sus creencias previas, permitiéndoles potencialmente construir nuevos argumentos para contribuir. Además, los agentes pueden rechazar argumentos que consideran contradictorios o no-minimales u objetar hechos en contradicción con sus creencias.

A diferencia de los protocolos y estrategias de diálogo de indagación presentes en

la literatura, se consideró que los agentes podrían tener hechos en contradicción con las creencias de los demás. Por lo tanto, si un agente informa que cree en un hecho en contradicción a otro hecho que fue utilizado para construir un argumento durante la sesión, los participantes deben llegar a un consenso sobre cuál de los dos prevalece y deben actualizar sus creencias de acuerdo al consenso. Además, las sesiones no se limitaron a sólo dos agentes, y se permitió que los participantes pueden optar por abandonar la sesión si no están dispuestos a seguir el protocolo, y que nuevos agentes pueden unirse libremente incluso cuando ya ha comenzado.

Considerando la noción de éxito provista por [WK95] para ambos tipos de diálogo, se probó que los diálogos desarrollado mediante los protocolos y estrategias definidos en esta tesis son exitosos.

5.1. Trabajo a futuro

El autor de [Wal10] explica el concepto de *apelación a la opinión experta* (en inglés, *appeal to expert opinion*), una forma de argumento basado en la suposición de que una fuente está en la posición de conocer algo sobre un tema porque él o ella tienen conocimiento experto sobre el tema. Además, menciona que hay una tendencia natural a respetar ciegamente a los expertos y considerarlos infalibles, lo cual es una práctica peligrosa ya que podrían estar equivocados. Por esta razón el autor presenta un conjunto de *preguntas críticas* para, dado un experto E y una afirmación A hecha por E , ayudar a decidir si aceptar o no la opinión de E :

1. ¿Qué tan creíble es E como fuente experta?
2. ¿ E es un experto en el dominio de A ?
3. ¿Qué cosas afirmó E que implican A ?
4. ¿ E es personalmente confiable como fuente?
5. ¿ A es consistente con lo que afirman otros expertos?
6. ¿ A está basada en evidencia?

Respecto a la pregunta 5, la *pregunta de consistencia* (en inglés, *consistency question*), el autor menciona que uno puede comparar A con lo que dicen otros expertos en el dominio.

La pregunta de consistencia está relacionada con la propuesta de [TGFS14, TGGS17, GTGS18], en la cual los autores consideran que los agentes pueden obtener información

de múltiples informantes y que la atribución de credibilidad a un informante particular puede ser más alta que la atribuida a los demás. En particular, cada agente tiene su propio orden parcial entre informantes, el cual representa la credibilidad que les asigna. Luego, cuando la información obtenida de diferentes informantes está en conflicto, el agente utiliza este orden parcial para decidir en cuál información confiar. La pregunta de consistencia de [Wal10] podría combinarse con la propuesta de [TGFS14, TGGS17, GTGS18] y el protocolo y estrategia de diálogo de consulta a experto definido en el Capítulo 3 para definir el operador RECHAZOOPINIÓN (ver Sección 3.5). Siempre que el agente consultante reciba del agente experto un argumento que contenga al menos un hecho en contradicción con sus creencias, el operador RECHAZOOPINIÓN podría comparar la credibilidad asignada a los informantes de cada hecho para decidir cuáles de ellos prevalecen. Luego, si al menos uno de los hechos del argumento no prevalece el operador podría rechazar el argumento.

Por otra parte, se buscará modificar el protocolo y estrategia de diálogo de indagación del Capítulo 4 para incluir las extensiones que fueron propuestas en la Sección 4.4. En particular, una de las extensiones permitirá a los agentes indagar sobre un literal q en vez de un argumento de indagación. Luego, utilizando un nuevo tipo de movimiento, los agentes podrían dar argumentos para q y para \bar{q} , creando así múltiples raíces de árboles de indagación. Por otra parte, otra de las extensiones permitirá a los agentes realizar un nuevo movimiento para contribuir reglas rebatibles individuales que ayuden a otros agentes a construir nuevos argumentos derrotadores.

Por último, si bien los protocolos y estrategias de diálogo propuestos en esta tesis se destacan de otros trabajos presentes en la literatura por permitir a los agentes tener sus propios hechos, se optó por utilizar DeLP sin reglas estrictas. Esta decisión se debió a que el tratamiento de reglas estrictas requiere de un tratamiento especial que ya fue estudiado en la literatura y estaba fuera del foco principal de esta tesis. No obstante, se buscará modificar ambos protocolos y estrategias para que los agentes también puedan tener reglas estrictas propias en sus bases de conocimiento, las cuales podrían causar derivaciones estrictas en contradicción con las derivaciones estrictas de los demás.

Apéndice A

Pruebas formales

Proposición 3.1 [pág. 43] Sea E un agente experto, C^1 un agente consultante, y \mathcal{A} un argumento construido por E , si E envía \mathcal{A} a C^1 y $\text{ADPARGUMENTO}(C^1, \mathcal{A}) = C^2$ entonces \mathcal{A} puede ser construido por C^2 .

Demostración: Sea $E = (\text{exp}, \Pi_{E^1}, \Delta_{E^1}, >_{E^1})$, $C^1 = (\text{con}, \Pi_{C^1}, \Delta_{C^1}, >_{C^1})$, $C^2 = (\text{con}, \Pi_{C^2}, \Delta_{C^2}, >_{C^2})$ y $\mathcal{A} = \langle a, R, H \rangle$. Supongamos que C^2 no puede construir \mathcal{A} luego de adoptarlo. Luego, según la Definición 2.13, alguna de las siguientes condiciones es cierta:

1. Existe una regla rebatible $r \in R_b$ tal que $r \notin \Delta_{C^2}$, o un hecho $h \in H_b$ tal que $h \notin \Pi_{C^2}$.
2. No existe una derivación para a a partir de $H \cup R$.
3. $\Pi_{C^2} \cup R$ es contradictorio, lo cual implica que existe una regla rebatible $\bar{h} \prec p_1, \dots, p_n \in R$ tal que el hecho $h \in \Pi_{C^2}$, o que $\{h, \bar{h}\} \in \Pi_{C^2}$.
4. \mathcal{A} no es minimal, lo cual implica que existe una regla rebatible $h \prec p_1, \dots, p_n \in R$ tal que el hecho $h \in \Pi_{C^2}$.

La condición 1 no es posible ya que, según la Definición 3.3, toda regla rebatible $r \in R$ fue añadida a Δ_{C^2} y todo hecho $h \in H$ fue añadido a Π_{C^2} . La condición 2 no es posible dado que, como E puede construir $\mathcal{A} = \langle a, R, H \rangle$, según la Definición 2.13 debe existir una derivación para a a partir de $H \cup R$. La condición 3 no es posible porque Π_{C^1} es inicialmente no contradictorio y, según la Definición 3.3, todo hecho $h \in \Pi_{C^1}$ tal que $\bar{h} \prec p_1, \dots, p_n \in R$ o $\bar{h} \in H$ fue eliminado de Π_{C^1} durante la adopción del argumento. Finalmente, la condición 4 no es posible ya que, según la Definición 3.3, todo hecho $h \in \Pi_{C^1}$ tal que $h \prec p_1, \dots, p_n \in R$ fue eliminado de Π_{C^1} luego de la adopción del argumento. Contradicción. \square

Proposición 3.2 [pág. 43] Sea E un agente experto, C^1 un agente consultante, \mathcal{A} un argumento construido por E , y \mathcal{B} un argumento construido por el E y por C , si E envía \mathcal{A} a C^1 y $\text{ADPARGUMENTO}(C^1, \mathcal{A}) = C^2$ entonces \mathcal{B} puede ser construido por C^2 .

Demostración: Sean $E = (\text{exp}, \Pi_E, \Delta_E, >_E)$, $C^2 = (\text{con}, \Pi_{C^2}, \Delta_{C^2}, >_{C^2})$, $\mathcal{A} = \langle a, R_a, H_a \rangle$ y $\mathcal{B} = \langle b, R_b, H_b \rangle$. Supongamos que luego de adoptar \mathcal{A} , el agente C^2 ya no puede construir \mathcal{B} . Luego, según la Definición 2.13, alguna de las siguientes condiciones es cierta:

1. Existe una regla rebatible $r \in R_b$ tal que $r \notin \Delta_{C^2}$.
2. Existe un hecho $h \in H_b$ tal que $h \notin \Pi_{C^2}$, lo cual según la Definición 3.3 implica que:
 - 2.1. $\bar{h} \in H_a$, o
 - 2.2. $\bar{h} \prec p_1, \dots, p_n \in R_a$, o
 - 2.3. $h \prec p_1, \dots, p_n \in R_a$.
3. No existe una derivación para b a partir de $H_b \cup R_b$.
4. $\Pi_{C^2} \cup R_b$ es contradictorio, lo cual implica que existe una regla rebatible $\bar{h} \prec p_1, \dots, p_n \in R_b$ tal que el hecho $h \in H_a \subseteq \Pi_{C^2}$, o que $\{h, \bar{h}\} \in \Pi_{C^2}$.
5. \mathcal{B} ya no es minimal, lo cual implica que existe una regla rebatible $h \prec p_1, \dots, p_n \in R_b$ tal que el hecho $h \in H_a \subseteq \Pi_{C^2}$.

La condición 1 no es posible ya que las reglas rebatibles nunca son eliminadas durante la adopción de un argumento. La condición 2.1 no es posible dado que, como $H_a \cup H_b \subseteq \Pi_E$, si $h \in H_b$ y $\bar{h} \in H_a$ entonces Π_E sería contradictorio. La condición 2.2 no es posible porque, como $H_b \subseteq \Pi_E$, si $h \in H_b$ y $\bar{h} \prec p_1, \dots, p_n \in R_a$ entonces $R_a \cup \Pi_E$ sería contradictorio y por lo tanto E no podría construir \mathcal{A} . La condición 2.3 no es posible ya que, como $H_b \subseteq \Pi_E$, si $h \in H_b$ y $h \prec p_1, \dots, p_n \in R_a$, el argumento \mathcal{A} no sería minimal para E y por ende E no podría construir \mathcal{A} . La condición 3 no es posible dado que, como E puede construir $\mathcal{B} = \langle b, R_b, H_b \rangle$, según la Definición 2.13 debe existir una derivación para b a partir de $H_b \cup R_b$. La condición 4 no es posible porque, como $H_a \subseteq \Pi_E$, si $\bar{h} \prec p_1, \dots, p_n \in R_b$ y $h \in H_a$ entonces $R_b \cup \Pi_E$ sería contradictorio y por lo tanto E no podría construir \mathcal{B} . Además Π_{C^1} es inicialmente no contradictorio y, según la Definición 3.3, todo hecho $h \in \Pi_{C^1}$ tal que $\bar{h} \in H_a$ fue eliminado de Π_{C^1} durante la adopción del argumento. Finalmente, la condición 5 no es posible ya que, como $H_a \subseteq \Pi_E$, si $h \prec p_1, \dots, p_n \in R_b$ y $h \in H_a$ entonces \mathcal{B} no sería minimal para E y por ende E no podría construir \mathcal{B} . Contradicción. \square

Lema 3.1 [pág. 64] Sea $e_a \neq (C, [\mathcal{F}_1, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset)$ un estado de sesión alcanzable, existe una y sólo una regla de transición de Θ_C aplicable desde e_a .

Demostración: Para esta prueba, comenzando desde el estado de sesión inicial, se analizarán las condiciones bajo las cuales las reglas de transición son aplicables y hacen evolucionar la sesión a todos los estados alcanzables posibles.

- Desde el estado inicial $(\mathbf{C}, [\], \emptyset, \emptyset)$ sólo rc_1 es aplicable y hace evolucionar la sesión al estado $(\mathbf{C}', [\mathcal{J}], \emptyset, \emptyset)$.
- En el estado $(\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \emptyset, \emptyset)$ con $n \geq 1$ vale $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{J}) = \textcircled{\mathbf{U}}$ o $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{J}) = \textcircled{\mathbf{D}}$. Si $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{J}) = \textcircled{\mathbf{U}}$, sólo rc_2 es aplicable y hace evolucionar la sesión al estado final $(\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset)$. Por el contrario, si $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{J}) = \textcircled{\mathbf{D}}$, sólo rc_3 es aplicable y hace evolucionar la sesión al estado $(\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P}, \emptyset)$ con $\mathbb{P} \neq \emptyset$.
- En el estado $(\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P}, \mathbb{N})$ con $n \geq 1$ y $(\mathcal{N}, \mathcal{A}) \in \mathbb{P}$ se cumple que $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{N}) = \textcircled{\mathbf{D}}$ o $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{N}) = \textcircled{\mathbf{U}}$. En el caso de que $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{N}) = \textcircled{\mathbf{D}}$, sólo rc_4 es aplicable y hace evolucionar la sesión al estado $(\mathbf{C}', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P} \setminus \{(\mathcal{N}, \mathcal{A})\}, \mathbb{N})$. Por el contrario, si $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{N}) = \textcircled{\mathbf{U}}$, sólo rc_5 es aplicable y hace evolucionar la sesión al estado $(\mathbf{C}', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P} \setminus \{(\mathcal{N}, \mathcal{A})\}, \mathbb{N} \cup \{\mathcal{N}\})$. Note que los elementos de \mathbb{P} son eliminados uno a uno por rc_4 y rc_5 hasta que $\mathbb{P} = \emptyset$.
- Desde el estado $(\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \emptyset, \mathbb{N})$ con $n \geq 1$ y $\mathcal{N} \in \mathbb{N}$ sólo rc_6 es aplicable con $\text{DERROTADOR}(\mathcal{J}, \mathcal{N}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n]) = (\mathcal{D}, \text{Pref})$ y hace evolucionar la sesión al estado $(\mathbf{C}', [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \mathcal{D}], \emptyset, \mathbb{N} \setminus \{\mathcal{N}\})$. Note que los elementos de \mathbb{N} son eliminados uno a uno por rc_6 hasta que $\mathbb{N} = \emptyset$.

Finalmente, por el sistema de transiciones $\Theta_{\mathbf{C}}$, el estado $(\mathbf{C}, [\], \mathbb{P}, \mathbb{N})$ con $\mathbb{P} \neq \emptyset$ o $\mathbb{N} \neq \emptyset$ no es alcanzable. \square

Teorema 3.1 [pág. 64] Sea e_f un estado final, e_f es alcanzable.

Demostración: Supongamos que dicha secuencia de transiciones no existe. Luego, alguna de las siguientes condiciones es cierta:

1. La sesión evolucionó a un estado en el cual no hay reglas de transición aplicables.
2. La sesión no puede evolucionar desde un estado alcanzable $e_a = (\mathbf{C}, [\mathcal{F}_1, \dots, \mathcal{F}_n], \mathbb{P}, \mathbb{N})$ hasta el estado final e_f , *i. e.*, la justificación no puede ser marcada como $\textcircled{\mathbf{U}}$. Esto implica que alguna de las siguientes condiciones es cierta:
 - 2.1. El agente \mathbf{C} tiene un negador \mathcal{N} para el cual no puede encontrar un derrotador marcado como $\textcircled{\mathbf{U}}$.

- 2.2. La sesión es infinita porque siempre que \mathbf{C} adopta del experto un nuevo argumento a favor \mathcal{D} que derrota a un negador, puede construir un nuevo negador para \mathcal{D} (que también es un negador para la justificación).

La condición 1 no es posible por el Lema 3.1. La condición 2.1 tampoco es posible: Si la sesión evolucionó a un estado $e_a = (\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \mathbb{P}, \mathbb{N})$ tal que $\mathcal{N} \in \mathbb{N}$, entonces \mathcal{N} es un negador para el argumento a favor enviado por el experto $\mathcal{F} \in [\mathcal{F}_1, \dots, \mathcal{F}_n]$. El experto puede construir el derrotador \mathcal{N} para \mathcal{F} ; caso contrario, hubiera respondido peor a la pregunta de preferencia $(\mathcal{N}, \mathcal{F})$, el agente \mathbf{C} hubiera actualizado sus preferencias para que $\mathcal{F} >_{\mathbf{C}} \mathcal{N}$, y \mathcal{N} no sería más un negador para \mathcal{F} . Además, por las reglas de transición rc_1 y rc_6 y las Definiciones 3.2 y 3.8, en el árbol de dialéctica del experto el argumento \mathcal{F} está marcado como $\textcircled{\mathbf{U}}$ a pesar de tener a \mathcal{N} como derrotador. Por lo tanto, el experto tiene un derrotador marcado como $\textcircled{\mathbf{U}}$ para \mathcal{N} para enviarle a \mathbf{C} . Finalmente, la condición 2.2 tampoco es posible, la base de creencias de cada agente es un programa lógico rebatible compuesto por un número finito de hechos y reglas rebatibles. Por lo tanto, el número de negadores que el consultante puede construir también es finito. Contradicción. \square

Corolario 3.1 [pág. 64] Sea e_f un estado final alcanzable, la secuencia de reglas de transición que hace evolucionar la sesión desde el estado inicial hasta e_f es única.

Demostración: De acuerdo al Teorema 3.1, dicha secuencia de transiciones debe existir. Además, según el Lema 3.1, en cada estado de sesión intermedio hay una sola regla de transición aplicable. Por lo tanto, la secuencia de transiciones que hacen evolucionar la sesión desde e_0 hasta e_f es única. \square

Teorema 3.2 [pág. 65] Sea $e_f = (\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n, \text{fin}], \emptyset, \emptyset)$ un estado de sesión final alcanzable, el agente consultante acepta \mathcal{J} .

Demostración: Una sesión evoluciona a e_f desde un estado alcanzable $e_a = (\mathbf{C}, [\mathcal{J}, \mathcal{F}_2, \dots, \mathcal{F}_n], \emptyset, \emptyset)$ sólo cuando se aplica la regla de transición rc_2 , *i. e.*, cuando se cumple que $\text{INTROSPECCIÓN}(\mathbf{C}, \mathcal{J}, \mathcal{J}) = \textcircled{\mathbf{U}}$. Luego, según la Definición 3.4, \mathbf{C} puede construir un árbol de dialéctica \mathcal{T} tal que $\text{ETIQUETA}(\text{RAÍZ}(\mathcal{T})) = \mathcal{J}$ y se cumple que $\text{MARCA}(\aleph_{\mathcal{J}}, \mathcal{T}) = \textcircled{\mathbf{U}}$ para todo nodo $\aleph_{\mathcal{J}}$ en \mathcal{T} (sólo la raíz de \mathcal{T}). Por lo tanto, \mathbf{C} acepta \mathcal{J} . \square

Proposición 4.1 [pág. 110] Sea $e_a = (\mathbb{I}, \emptyset, [R_1, \dots, R_n], \emptyset)$ con $n \geq 1$ un estado alcanzable y e_f un estado final, e_f es alcanzable desde e_a .

Demostración: Las reglas de transición aplicables desde e_a son ri_{13} , si todo movimiento

en R_n es **pasar**, o ri_{12} , si hay al menos un movimiento en R_n que no es **pasar**. Si ri_{13} es aplicable, la sesión evoluciona al estado final $(\mathbb{I}, [R_1, \dots, R_n, [\mathbf{fin}]], \emptyset)$. Por el contrario, si ri_{12} es aplicable, la sesión evoluciona al estado $(\mathbb{I}, [R_1, \dots, R_n, [\varepsilon]], \emptyset)$ en el que comenzó una nueva ronda. Note que la cola de espera se mantuvo vacía ya que no hay participantes. Por lo tanto, la única regla de transición aplicable desde este estado es ri_{13} , la cual hace evolucionar la sesión al estado final $(\mathbb{I}, [R_1, \dots, R_n, [\varepsilon], [\mathbf{fin}]], \emptyset)$. \square

Proposición 4.2 [pág. 110] Sea $e_a = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle], \mathbb{E})$ tal que $\mathbb{I} = \langle \emptyset, \emptyset \rangle$ un estado alcanzable y e_f un estado final, e_f es alcanzable desde e_a .

Demostración: La única regla de transición aplicable desde e_a es ri_7 , la cual hace evolucionar la sesión al estado $e_b = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle, \varepsilon], \mathbb{E})$ en el que los agentes en \mathbb{E} está esperando hacer un movimiento. De acuerdo a la Definición 4.5, el único movimiento disponible es **pasar** ya que todos los demás movimientos requieren que hayan nodos en el árbol de indagación. Por lo tanto, la única regla de transición aplicable desde e_b es ri_{11} , la cual es aplicada repetidamente hasta que \mathbb{E} se vacía (la ronda actual finaliza) y la sesión evoluciona al estado $e_c = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle, \varepsilon, \langle ID', \text{pasar} \rangle, \varepsilon, \dots, \langle ID'', \text{pasar} \rangle, \varepsilon], \emptyset)$ en el que todos los agentes de \mathbb{E} hicieron el movimiento **pasar**. Como en la ronda se realizó al menos un movimiento que no es **pasar**, la única regla de transición aplicable desde e_c es ri_{12} , el cual hace evolucionar la sesión al estado $e_d = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle, \varepsilon, \langle ID', \text{pasar} \rangle, \varepsilon, \dots, \langle ID'', \text{pasar} \rangle, \varepsilon], [\varepsilon], \mathbb{E}')$ en el que comenzó una nueva ronda. Nuevamente, como el árbol de indagación no tiene nodos, el único movimiento disponible es **pasar** y la única regla de transición aplicable es ri_{11} , la cual es aplicada repetidamente hasta que la ronda finaliza y la sesión evoluciona al estado $e_e = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle, \varepsilon, \langle ID', \text{pasar} \rangle, \varepsilon, \dots, \langle ID'', \text{pasar} \rangle, \varepsilon], [\varepsilon, \langle ID''', \text{pasar} \rangle, \varepsilon, \dots, \langle ID''', \text{pasar} \rangle, \varepsilon], \emptyset)$ en el que todos los agentes de \mathbb{E}' hicieron el movimiento **pasar**. Finalmente, como todos los movimientos que se realizaron en la ronda fueron **pasar**, la regla de transición ri_{13} es aplicada y la sesión evoluciona al estado final $e_f = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle, \varepsilon, \langle ID', \text{pasar} \rangle, \varepsilon, \dots, \langle ID'', \text{pasar} \rangle, \varepsilon], [\varepsilon, \langle ID''', \text{pasar} \rangle, \varepsilon, \dots, \langle ID''', \text{pasar} \rangle, \varepsilon], [\mathbf{fin}]], \emptyset)$ \square

Lema 4.1 [pág. 111] Sea e_a un estado alcanzable tal que $e_a \neq (\mathbb{I}, \mathbb{Z}, [], \mathbb{E})$ con $|\mathbb{Z}| < 2$ y tal que $e_a \neq (\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n, [\mathbf{fin}]], \emptyset)$, existe al menos una regla de transición en $\Theta'_i = \{ri_3, ri_4, ri_5, ri_6, ri_7, ri_8, ri_9, ri_{10}, ri_{11}, ri_{12}, ri_{13}\}$ que es aplicable desde e_a .

Demostración: Para esta prueba, comenzando desde $(\mathbb{I}, \mathbb{Z}, [], \mathbb{E})$ con $|\mathbb{Z}| \geq 2$ (un estado en

el que la primera ronda puede comenzar), se analizarán las condiciones bajo las cuales las reglas de transición de Θ'_I son aplicables y hacen evolucionar la sesión a todos los estados alcanzables posibles:

- Desde $(\mathbb{I}, \mathbb{Z}, [], \mathbb{I}, \mathbb{E})$ con $|\mathbb{Z}| \geq 2$ sólo ri_3 es aplicable y hace evolucionar la sesión al estado $(\mathbb{I}, \mathbb{Z}, [[\varepsilon]], \mathbb{E})$.
- Desde $(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon]], \mathbb{E})$ con $ID \in \mathbb{E}$, dependiendo de cuáles movimientos puede realizar $\mathbf{Z} = (ID, \Pi_{\mathbf{Z}}, \Delta_{\mathbf{Z}}, >)$, ri_4 , ri_6 , ri_8 , ri_9 o ri_{11} son aplicables. En particular, ri_4 , ri_6 , ri_8 y ri_9 hacen evolucionar la sesión a $(\mathbb{I}', \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{Movimiento} \rangle]], \mathbb{E} \setminus \{ID\})$ con $\text{Movimiento} \in \{\text{derrotar}, \text{rechazar}, \text{objetar}\}$ mientras que ri_{11} la hace evolucionar a $(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{pasar} \rangle], \varepsilon], \mathbb{E} \setminus \{ID\})$.
- Desde $(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon, \langle ID, \text{Movimiento} \rangle]], \mathbb{E} \setminus \{ID\})$ con $\text{Movimiento} \in \{\text{derrotar}, \text{rechazar}, \text{objetar}\}$ sólo ri_5 , ri_7 o ri_{10} es aplicable, dependiendo de si $\text{Movimiento} = \text{derrotar}$, $\text{Movimiento} = \text{rechazar}$ o $\text{Movimiento} = \text{objetar}$, respectivamente, y hace evolucionar la sesión a $(\mathbb{I}, \mathbb{Z}', [\dots, [\dots, \langle ID, \text{Movimiento} \rangle], \varepsilon], \mathbb{E})$.
- Desde $(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon]], \emptyset)$ sólo ri_{12} o ri_{13} es aplicable, dependiendo de si todos los agentes pasaron durante la última ronda. En particular, ri_{12} hace evolucionar la sesión a $(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon], [\varepsilon], \mathbb{E})$ mientras que ri_{13} la hace evolucionar a $(\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \varepsilon], [\text{fin}], \emptyset)$. \square

Lema 4.2 [pág. 111] Sea $e_a = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle ID, \text{rechazar}(\mathcal{A}, \mathcal{B}) \rangle]], \mathbb{E})$ un estado de sesión alcanzable, si la regla de transición $ri_1 \in \Theta_I$ no vuelve a ser aplicada entonces \mathcal{A} no será vinculado en el árbol de indagación nuevamente.

Demostración: Sea $\mathcal{A} = \langle a, R_a, H_a \rangle$ y $\mathcal{B} = \langle b, R_b, H_b \rangle$. \mathcal{A} fue rechazado por \mathcal{B} porque:

1. $\mathcal{B} = \langle b, \emptyset, \{b\} \rangle$ y $\bar{b} \prec c_1, \dots, c_n \in R_a$ (i. e., fue un rechazo por contradicción), o
2. $a = b$ y $R_b \subset R_a$ (i. e., fue un rechazo por no-minimalidad).

Supongamos que en un estado posterior a e_a un agente $\mathbf{Z} = (ID', \Pi_{\mathbf{Z}}, \Delta_{\mathbf{Z}}, >)$ realiza el movimiento $\text{derrotar}(\mathcal{C}, \mathcal{A})$. Luego, debe existir un par $(\mathcal{C}, \mathcal{A})$ que satisfaga la Definición 4.5, lo cual implica que \mathbf{Z} puede construir \mathcal{A} . Al realizarse el movimiento $\text{rechazar}(\mathcal{A}, \mathcal{B})$ todos adquirieron \mathcal{B} y, por ende, alguna de las siguientes condiciones es cierta:

- \mathbf{Z} puede construir \mathcal{B} . Luego, por la condición 1 o 2, el agente \mathbf{Z} no puede construir \mathcal{A} . Contradicción.
- \mathbf{Z} no puede construir \mathcal{B} . Luego, como \mathbf{Z} cree en todas las reglas rebatibles de \mathcal{B} , debe existir un hecho $h \in H_b$ tal que $h \notin \Pi_{\mathbf{Z}}$. Si $h \notin \Pi_{\mathbf{Z}}$ después de adquirir \mathcal{B} , entonces:

- Z creía en \bar{h} antes de la adquisición (ver Definición 4.3), o
- se consensuó el rechazo de h luego de un movimiento $\text{objetar}(h, \bar{h})$.

En ambos casos se cumple que $\bar{h} \in \Pi_Z$. Si la condición 1 fuese cierta entonces $h = b$ y, en consecuencia, $\bar{b} \in \Pi_Z$. Por lo tanto, Z no podría construir \mathcal{A} por no-minimalidad (*i. e.*, $\bar{b} \prec c_1, \dots, c_n \in \mathbb{R}_a$). Contradicción. Por el contrario, si la condición 2 fuese cierta entonces $h \in \mathbb{H}_a$ o $h \prec c_1, \dots, c_n \in \mathbb{R}_a$. En ambos casos, como $\bar{h} \in \Pi_Z$, Z no podría construir \mathcal{A} . Contradicción. \square

Lema 4.3 [pág. 111] Sea $e_a = (\mathbb{I}, \mathbb{Z}, [\dots, [\dots, \langle \text{OtroID}, \text{objetar}(h, \bar{h}, \bar{h}) \rangle]], \mathbb{E})$ un estado de sesión alcanzable tal que $Z = (ID, \Pi_Z, \Delta_Z, >) \in \mathbb{Z}$ y $h \in \Pi_Z$, si la regla de transición $ri_1 \in \Theta_I$ no vuelve a ser aplicada entonces en todo estado alcanzable $e_b = (\mathbb{I}', \mathbb{Z}', \mathbb{R}', \mathbb{E}')$ desde e_a se cumple que si $Z' = (ID, \Pi_{Z'}, \Delta_{Z'}, >) \in \mathbb{Z}'$ entonces $\bar{h} \in \Pi_{Z'}$.

Demostración: Las reglas de transición aplicables desde e_a son ri_2 , la cual implica que algún participante abandona la sesión, y ri_{10} , la cual implica que todos los agentes eliminan h y añaden \bar{h} a sus bases de creencias y hace evolucionar la sesión al estado $(\mathbb{I}, \mathbb{Z}', [\dots, [\dots, \langle ID, \text{objetar}(h, \bar{h}, \bar{h}) \rangle], \varepsilon], \mathbb{E})$. Luego:

- $Z = (ID, \Pi_Z, \Delta_Z, >)$ abandona la sesión antes de que ri_{10} se aplique, o
- Z se queda en la sesión y evoluciona a $Z' = (ID, \Pi_{Z'}, \Delta_{Z'}, >)$ con $\bar{h} \in \Pi_{Z'}$.

Los agentes en \mathbb{Z}' no creen en h y, como se supuso que la regla de transición $ri_1 \in \Theta_I$ no volverá a ser aplicada, no se unirán más agentes a la sesión. Por lo tanto, el movimiento $\text{objetar}(\bar{h}, h)$ no podrá ser realizado durante el resto de la sesión y en todo estado alcanzable $e_b = (\mathbb{I}', \mathbb{Z}', \mathbb{R}', \mathbb{E}')$ se cumple que si $Z' = (ID, \Pi_{Z'}, \Delta_{Z'}, >) \in \mathbb{Z}'$ entonces $\bar{h} \in \Pi_{Z'}$. \square

Teorema 4.1 [pág. 112] Sea $e_a = (\mathbb{I}, \mathbb{Z}, \mathbb{R}, \mathbb{E})$ con $|\mathbb{Z}| \geq 2$ un estado de sesión alcanzable y e_f un estado de sesión final, si la regla de transición $ri_1 \in \Theta_I$ no es aplicada nuevamente entonces e_f es alcanzable desde e_a .

Demostración: Supongamos que e_f no es alcanzable desde e_a . Luego, alguna de las siguientes condiciones es cierta:

1. La sesión evolucionó a un estado en el cual no hay reglas de transición aplicables.
2. La sesión es infinita porque en todas las rondas algún agente realiza un movimiento que no es **pasar**.

La condición 1 no es posible por el Lema 4.1. La condición 2 tampoco es posible: Recuerde que la base de creencias de todo agente es un programa lógico rebatible compuesta de

una cantidad finita de hechos y reglas rebatibles. Además, recuerde que los agentes deben abandonar la sesión para poder adquirir información de fuentes externas. Esto implica que la cantidad de argumentos que podrían construirse a partir de las creencias de los participantes actuales también es finita. Luego, según el Lema 4.2, si un argumento es rechazado y desde ese estado no se unen más agentes, dicho argumento no será vinculado nuevamente en el árbol de indagación. Por lo tanto, como se supuso que la regla de transición $ri_1 \in \Theta_I$ no volverá a ser aplicada:

1. la cantidad de movimientos **derrotar** potenciales es finita y, en consecuencia,
2. la cantidad de movimientos **rechazar** potenciales es finita, y
3. según el Lema 4.3, la cantidad de movimientos **objetar** es finita.

Según las razones 1, 2 y 3, eventualmente todos los participantes harán un movimiento **pasar** en la misma ronda y la sesión evolucionará a un estado final. Contradicción. \square

Teorema 4.2 [pág. 112] Sea $e_f = (\mathbb{I}, \mathbb{Z}, [R_1, \dots, R_n, [\text{fin}], \emptyset])$ un estado de sesión final alcanzable tal que $\text{ETIQUETA}(\text{RAÍZ}(\mathbb{I})) = \mathcal{I}$, $\mathbb{Z} = (ID, \Pi_{\mathbb{Z}}, \Delta_{\mathbb{Z}}, >) \in \mathbb{Z}$ e $\mathbb{Y} = (ID', \Pi_{\mathbb{Y}}, \Delta_{\mathbb{Y}}, >) \in \mathbb{Z}$ dos agentes, y \mathcal{T}_1 y \mathcal{T}_2 los árboles de dialéctica para \mathcal{I} construidos por \mathbb{Z} e \mathbb{Y} , respectivamente, se cumple que $\text{MARCA}(\text{RAÍZ}(\mathcal{T}_1), \mathcal{T}_1) = \text{MARCA}(\text{RAÍZ}(\mathcal{T}_2), \mathcal{T}_2)$.

Demostración: Supongamos que $\text{MARCA}(\text{RAÍZ}(\mathcal{T}_1), \mathcal{T}_1) = \textcircled{U}$ y $\text{MARCA}(\text{RAÍZ}(\mathcal{T}_2), \mathcal{T}_2) = \textcircled{D}$. Sin pérdida de generalidad, esto implica que alguna de las siguientes condiciones es cierta:

1. Existe un nodo $\aleph_{\mathcal{A}}$ en \mathcal{T}_1 que no está en \mathcal{T}_2 pero está en \mathbb{I} . Luego, $\mathcal{A} = \langle a, R, H \rangle$ fue vinculado a \mathbb{I} mediante un movimiento **derrotar**(\mathcal{B}, \mathcal{A}), y no puede ser construido por el agente \mathbb{Y} . Por la Definición 4.3, esto implica que:
 - 1.1. existe una regla rebatible $h \prec p_1, \dots, p_n \in R$ tal que $\bar{h} \in \Pi_{\mathbb{Y}}$, o
 - 1.2. existe una regla rebatible $h \prec p_1, \dots, p_n \in R$ tal que $h \in \Pi_{\mathbb{Y}}$, o
 - 1.3. existe un hecho $h \in H$ tal que $\bar{h} \in \Pi_{\mathbb{Y}}$, o
 - 1.4. existe un hecho $h \in H$ tal que $h \notin \Pi_{\mathbb{Y}}$ y $\bar{h} \notin \Pi_{\mathbb{Y}}$.
2. Existe un nodo $\aleph_{\mathcal{A}}$ en \mathcal{T}_1 que no está en \mathcal{T}_2 ni en \mathbb{I} , y existe otro nodo $\aleph_{\mathcal{B}}$ en \mathcal{T}_2 y en \mathbb{I} tal que $\aleph_{\mathcal{A}} \in \text{HIJOS}(\aleph_{\mathcal{B}})$.

Considere que, como la sesión ya finalizó, los participantes realizaron todos los movimientos que podían en \mathbb{I} . Las condiciones 1.1 y 1.2 no son posibles dado que, por la Definición 4.5, \mathbb{Y} tendría que haber realizado el movimiento **rechazar**(\mathcal{A}, \mathcal{C}), lo cual implica que \mathcal{A} no podría estar vinculado en \mathbb{I} y \aleph no podría existir. Contradicción. De manera análoga, la condición 1.3 tampoco es posible: \mathbb{Y} tendría que haber realizado el movimiento

objetar (h, \bar{h}) y, como $e \notin \Pi_{\gamma}$, el hecho rechazado por **CONSENSO** (h, \bar{h}) debería haber sido h , lo cual implica que \mathcal{A} no podría estar vinculado en \mathbb{I} y \aleph no podría existir. Contradicción. La condición 1.4 tampoco es posible por la Definición 4.3. La condición 2 tampoco es posible ya que, según la Definición 4.5, \mathbf{Z} podría haber realizado el movimiento **derrotar** $(\mathcal{B}, \mathcal{A})$. Contradicción. \square

Bibliografía

- [AC98] AMGOUD, L., AND CAYROL, C. On the acceptability of arguments in preference-based argumentation. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence* (1998), Morgan Kaufmann Publishers Inc., pp. 1–7.
- [AGG19a] AGIS, R. A., GOTTIFREDI, S., AND GARCÍA, A. J. An approach for distributed discussion and collaborative knowledge sharing: Theoretical and empirical analysis. *Expert Systems with Applications* 116 (2019), 377–395.
- [AGG19b] AGIS, R. A., GOTTIFREDI, S., AND GARCÍA, A. J. Acquiring knowledge from expert agents in a structured argumentation setting. *Argument & Computation*, 10 (2019), 149–189.
- [AK07] AMGOUD, L., AND KACI, S. An argumentation framework for merging conflicting knowledge bases. *Int. Journal of Approximate Reasoning* 45, 2 (2007), 321–340.
- [Amg05] AMGOUD, L. An argumentation-based model for reasoning about coalition structures. In *International Workshop on Argumentation in Multi-Agent Systems* (2005), Springer, pp. 217–228.
- [AMP00] AMGOUD, L., MAUDET, N., AND PARSONS, S. Modelling dialogues using argumentation. In *Proceedings Fourth International Conference on Multi-Agent Systems* (2000), IEEE, pp. 31–38.
- [AMP02] AMGOUD, L., MAUDET, N., AND PARSONS, S. An argumentation-based semantics for agent communication languages. In *15th European Conference on Artificial Intelligence* (2002), pp. 38–42.

- [AP93] ALFERES, J. J., AND PEREIRA, L. M. Contradiction: when avoidance equals removal part i. In *International Workshop on Extensions of Logic Programming* (1993), Springer, pp. 11–23.
- [AP01] AMGOUD, L., AND PARSONS, S. Agent dialogues with conflicting preferences. In *International Workshop on Agent Theories, Architectures, and Languages* (2001), Springer, pp. 190–205.
- [AP09] AMGOUD, L., AND PRADE, H. Using arguments for making and explaining decisions. *Artificial Intelligence* 173, 3-4 (2009), 413–436.
- [APM00] AMGOUD, L., PARSONS, S., AND MAUDET, N. Arguments, dialogue, and negotiation. *a a* 10, 11 (2000), 02.
- [ATT⁺12] AULINAS, M., TOLCHINSKY, P., TURON, C., POCH, M., AND CORTÉS, U. Argumentation-based framework for industrial wastewater discharges management. *Engineering Applications of Artificial Intelligence* 25, 2 (2012), 317–325.
- [BA09] BLACK, E., AND ATKINSON, K. Dialogues that account for different perspectives in collaborative argumentation. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2* (2009), International Foundation for Autonomous Agents and Multiagent Systems, pp. 867–874.
- [BDC08] BULLING, N., DIX, J., AND CHESÑEVAR, C. I. Modelling coalitions: AtI+ argumentation. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2* (2008), International Foundation for Autonomous Agents and Multiagent Systems, pp. 681–688.
- [Ben03] BENCH-CAPON, T. J. M. Persuasion in practical argument using value-based argumentation frameworks. *J. Log. Comput.* 13, 3 (2003), 429–448.
- [Ber14] BERARIU, T. An argumentation framework for bdi agents. In *Intelligent Distributed Computing VII*. Springer, 2014, pp. 343–354.

- [BH07] BLACK, E., AND HUNTER, A. A generative inquiry dialogue system. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems (2007)*, ACM, p. 241.
- [BH09] BLACK, E., AND HUNTER, A. An inquiry dialogue system. *Autonomous Agents and Multi-Agent Systems* 19, 2 (2009), 173–209.
- [BV14] BEDI, P., AND VASHISTH, P. Empowering recommender systems using trust and argumentation. *Information sciences* 279 (2014), 569–586.
- [CCS05] CAPOBIANCO, M., CHESÑEVAR, C. I., AND SIMARI, G. R. Argumentation and the dynamics of warranted beliefs in changing environments. *Autonomous Agents and Multi-Agent Systems* 11, 2 (2005), 127–151.
- [CI15] CARRERA, Á., AND IGLESIAS, C. A. A systematic review of argumentation techniques for multi-agent systems research. *Artif. Intell. Rev.* 44, 4 (2015), 509–535.
- [CMKMM14] COSTE-MARQUIS, S., KONIECZNY, S., MAILLY, J.-G., AND MARQUIS, P. On the revision of argumentation systems: Minimal change of arguments statuses. In *Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning (2014)*.
- [CSCC13] CHOW, H. K., SIU, W., CHAN, C.-K., AND CHAN, H. C. An argumentation-oriented multi-agent system for automating the freight planning process. *Expert systems with applications* 40, 10 (2013), 3858–3871.
- [DHL⁺18] DILLER, M., HARET, A., LINSBICHLER, T., RÜMMELE, S., AND WOLTRAN, S. An extension-based approach to belief revision in abstract argumentation. *International Journal of Approximate Reasoning* 93 (2018), 395–423.
- [DHP14] DOUTRE, S., HERZIG, A., AND PERRUSSEL, L. A dynamic logic framework for abstract argumentation. In *Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning (2014)*.

- [DKT06] DUNG, P. M., KOWALSKI, R. A., AND TONI, F. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence* 170, 2 (2006), 114–159.
- [DKT09] DUNG, P. M., KOWALSKI, R. A., AND TONI, F. Assumption-based argumentation. In *Argumentation in artificial intelligence*. Springer, 2009, pp. 199–218.
- [dOGAP⁺18] DE OLIVEIRA GABRIEL, V., ADAMATTI, D. F., PANISSON, A. R., BORDINI, R. H., AND BILLA, C. Z. Argumentation-based reasoning in bdi agents using toulmin’s model. In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)* (2018), IEEE, pp. 378–383.
- [DR09] DEVEREUX, J., AND REED, C. Strategic argumentation in rigorous persuasion dialogue. In *Argumentation in Multi-Agent Systems, 6th International Workshop, ArgMAS 2009, Budapest, Hungary, May 12, 2009. Revised Selected and Invited Papers* (2009), pp. 94–113.
- [Dun95] DUNG, P. M. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence* 77, 2 (1995), 321–358.
- [ESM12] EL-SISI, A. B., AND MOUSA, H. M. Argumentation based negotiation in multiagent system. In *2012 Seventh International Conference on Computer Engineering & Systems (ICCES)* (2012), IEEE, pp. 261–266.
- [FCS⁺13] FAN, X., CRAVEN, R., SINGER, R., TONI, F., AND WILLIAMS, M. Assumption-based argumentation for decision-making with preferences: A medical case study. In *International Workshop on Computational Logic in Multi-Agent Systems* (2013), Springer, pp. 374–390.
- [FKIS02] FALAPPA, M. A., KERN-ISBERNER, G., AND SIMARI, G. R. Explanations, belief revision and defeasible reasoning. *Artificial Intelligence* 141, 1-2 (2002), 1–28.
- [FO12] FERRANDO, S. P., AND ONAINDIA, E. Defeasible argumentation for multi-agent planning in ambient intelligence applications. In *Proceedings*

- of the 11th International Conference on Autonomous Agents and Multi-agent Systems-Volume 1* (2012), International Foundation for Autonomous Agents and Multiagent Systems, pp. 509–516.
- [FO13] FERRANDO, S. P., AND ONAINDIA, E. Context-aware multi-agent planning in intelligent environments. *Information Sciences 227* (2013), 22–42.
- [FT11] FAN, X., AND TONI, F. Assumption-based argumentation dialogues. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011* (2011), pp. 198–203.
- [FT12] FAN, X., AND TONI, F. Agent strategies for aba-based information-seeking and inquiry dialogues. In *ECAI* (2012), pp. 324–329.
- [FT15] FAN, X., AND TONI, F. Mechanism design for argumentation-based information-seeking and inquiry. In *International Conference on Principles and Practice of Multi-Agent Systems* (2015), Springer, pp. 519–527.
- [FTMW14] FAN, X., TONI, F., MOCANU, A., AND WILLIAMS, M. Dialogical two-agent decision making with assumption-based argumentation. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems* (2014), International Foundation for Autonomous Agents and Multiagent Systems, pp. 533–540.
- [GGB12] GRANDO, M. A., GLASSPOOL, D., AND BOXWALA, A. Argumentation logic for the flexible enactment of goal-based medical guidelines. *Journal of biomedical informatics 45*, 5 (2012), 938–949.
- [GGL14] GARCEZ, A. S. D., GABBAY, D. M., AND LAMB, L. C. A neural cognitive model of argumentation with application to legal inference and decision making. *Journal of Applied Logic 12*, 2 (2014), 109–127.
- [GGS09] GOTTIFREDI, S., GARCIA, A. J., AND SIMARI, G. R. Argumentation systems and agent programming languages. In *2009 AAAI Fall Symposium Series* (2009).

- [GGS10] GOTTIFREDI, S., GARCIA, A. J., AND SIMARI, G. R. Query-based argumentation in agent programming. In *Ibero-American Conference on Artificial Intelligence* (2010), Springer, pp. 284–295.
- [GS04] GARCÍA, A. J., AND SIMARI, G. R. Defeasible logic programming: An argumentative approach. *TPLP* 4, 1-2 (2004), 95–138.
- [GS14] GARCÍA, A. J., AND SIMARI, G. R. Defeasible logic programming: Delp-servers, contextual queries, and explanations for answers. *Argument & Computation* 5, 1 (2014), 63–88.
- [GTGS18] GOTTIFREDI, S., TAMARGO, L. H., GARCÍA, A. J., AND SIMARI, G. R. Arguing about informant credibility in open multi-agent systems. *Artificial Intelligence* 259 (2018), 91–109.
- [HGAB06] HSAIRI, L., GHEDIRA, K., ALIMI, M. A., AND BENABDELHAFID, A. Resolution of conflicts via argument based negotiation: extended enterprise case. In *2006 International Conference on Service Systems and Service Management* (2006), vol. 1, IEEE, pp. 828–833.
- [HGAB10] HSAIRI, L., GHEDIRA, K., ALIMI, A. M., AND BENABDELHAFID, A. Trust and reputation model for the 2-ibn framework. In *2010 IEEE International Conference on Systems, Man and Cybernetics* (2010), IEEE, pp. 2137–2144.
- [HJBJ13a] HERAS, S., JORDÁN, J., BOTTI, V., AND JULIÁN, V. Argue to agree: a case-based argumentation approach. *International Journal of Approximate Reasoning* 54, 1 (2013), 82–108.
- [HJBJ13b] HERAS, S., JORDÁN, J., BOTTI, V., AND JULIÁN, V. Case-based strategies for argumentation dialogues in agent societies. *Information Sciences* 223 (2013), 1–30.
- [HL10] HUANG, S.-L., AND LIN, C.-Y. The search for potentially interesting products in an e-marketplace: An agent-to-agent argumentation approach. *Expert Systems with Applications* 37, 6 (2010), 4468–4478.

- [JH12] JANJUA, N. K., AND HUSSAIN, F. K. Web@ idss—argumentation-enabled web-based idss for reasoning over incomplete and conflicting information. *Knowledge-Based Systems 32* (2012), 9–27.
- [KM03] KAKAS, A., AND MORAITIS, P. Argumentation based decision making for autonomous agents. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems* (2003), ACM, pp. 883–890.
- [Kra97] KRAUS, S. Negotiation and cooperation in multi-agent environments. *Artificial intelligence 94*, 1-2 (1997), 79–97.
- [MA11] MONTESERIN, A., AND AMANDI, A. Argumentation—based negotiation planning for autonomous agents. *Decision Support Systems 51*, 3 (2011), 532–548.
- [MB04] MORGE, M., AND BEAUNE, P. A negotiation support system based on a multi-agent system: specificity and preference relations on arguments. In *Proceedings of the 2004 ACM symposium on Applied computing* (2004), ACM, pp. 474–478.
- [MP14] MODGIL, S., AND PRAKKEN, H. The aspic+ framework for structured argumentation: a tutorial. *Argument & Computation 5*, 1 (2014), 31–62.
- [MRF+10] MOGUILLANSKY, M. O., ROTSTEIN, N. D., FALAPPA, M. A., GARCÍA, A. J., AND SIMARI, G. R. Argument theory change through defeater activation. In *COMMA* (2010), pp. 359–366.
- [MRN05] MARREIROS, G., RAMOS, C., AND NEVES, J. Dealing with emotional factors in agent based ubiquitous group decision. In *International Conference on Embedded and Ubiquitous Computing* (2005), Springer, pp. 41–50.
- [MS07] MORAITIS, P., AND SPANOUDAKIS, N. Argumentation-based agent interaction in an ambient-intelligence context. *IEEE Intelligent Systems 22*, 6 (2007), 84–93.
- [MS14] MAIO, P., AND SILVA, N. An extensible argument-based ontology matching negotiation approach. *Science of computer programming 95* (2014), 3–25.

- [MSC11] MAIO, P., SILVA, N., AND CARDOSO, J. Generating arguments for ontology matching. In *2011 22nd International Workshop on Database and Expert Systems Applications* (2011), IEEE, pp. 239–243.
- [MTB⁺13] MODGIL, S., TONI, F., BEX, F., BRATKO, I., CHESNEVAR, C. I., DVOŘÁK, W., FALAPPA, M. A., FAN, X., GAGGL, S. A., GARCÍA, A. J., ET AL. The added value of argumentation. In *Agreement technologies*. Springer, 2013, pp. 357–403.
- [MVEPA03] MCBURNEY, P., VAN EIJK, R. M., PARSONS, S., AND AMGOUD, L. A dialogue game protocol for agent purchase negotiations. *Autonomous Agents and Multi-Agent Systems* 7, 3 (2003), 235–273.
- [PA93] PEREIRA, L. M., AND ALFERES, J. J. Contradiction: When avoidance equals removal part ii. In *International Workshop on Extensions of Logic Programming* (1993), Springer, pp. 268–281.
- [Plo81] PLOTKIN, G. D. A structural approach to operational semantics.
- [PMVB15] PANISSON, A. R., MENEGUZZI, F., VIEIRA, R., AND BORDIN, R. H. Towards practical argumentation in multi-agent systems. In *2015 Brazilian Conference on Intelligent Systems (BRACIS)* (2015), IEEE, pp. 98–103.
- [Pol87] POLLOCK, J. L. Defeasible reasoning. *Cognitive Science* 11, 4 (1987), 481–518.
- [Pra06] PRAKKEN, H. Formal systems for persuasion dialogue. *Knowledge Eng. Review* 21, 2 (2006), 163–188.
- [Pra09] PRAKKEN, H. An abstract framework for argumentation with structured arguments. *Journal of Argument and Computation* 1 (2009), 93–124.
- [PS97] PRAKKEN, H., AND SARTOR, G. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics* 7, 1 (1997).
- [PSJ98] PARSONS, S., SIERRA, C., AND JENNINGS, N. Agents that reason and negotiate by arguing. *Journal of Logic and computation* 8, 3 (1998), 261–292.

- [PTB14] PASHAEI, K., TAGHIYAREH, F., AND BADIE, K. A negotiation-based genetic framework for multi-agent credit assignment. In *German Conference on Multiagent System Technologies* (2014), Springer, pp. 72–89.
- [PV02] PRAKKEN, H., AND VREESWIJK, G. Logics for defeasible argumentation. In *Handbook of Philosophical Logic*, D. Gabbay and F. Guenther, Eds., vol. 4. Kluwer Academic Pub., 2002, pp. 218–319.
- [PWA02] PARSONS, S., WOOLDRIDGE, M., AND AMGOUD, L. An analysis of formal inter-agent dialogues. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1* (2002), ACM, pp. 394–401.
- [RA06] RAHWAN, I., AND AMGOUD, L. An argumentation-based approach for practical reasoning. In *International Workshop on Argumentation in Multi-Agent Systems* (2006), Springer, pp. 74–90.
- [RAPB11] RILEY, L., ATKINSON, K., PAYNE, T., AND BLACK, E. An implemented dialogue system for inquiry and persuasion. In *International Workshop on Theorie and Applications of Formal Argumentation* (2011), Springer, pp. 67–84.
- [RGS02] RUEDA, S. V., GARCÍA, A. J., AND SIMARI, G. R. Argument-based negotiation among bdi agents. *Journal of Computer Science & Technology* 2 (2002).
- [RGS07] ROTSTEIN, N. D., GARCÍA, A. J., AND SIMARI, G. R. Reasoning from desires to intentions: A dialectical framework. In *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE* (2007), vol. 22, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, p. 136.
- [RK16] ROSENFELD, A., AND KRAUS, S. Strategical argumentative agent for human persuasion. In *Proceedings of the Twenty-second European Conference on Artificial Intelligence* (2016), IOS Press, pp. 320–328.

- [RRJ⁺03] RAHWAN, I., RAMCHURN, S. D., JENNINGS, N. R., MCBURNEY, P., PARSONS, S., AND SONENBERG, L. Argumentation-based negotiation. *The Knowledge Engineering Review* 18, 4 (2003), 343–375.
- [SFEA10] SCHLESINGER, F., FERRETTI, E., ERRECALDE, M., AND AGUIRRE, G. An argumentation-based bdi personal assistant. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (2010), Springer, pp. 701–710.
- [SR09] SIMARI, G. R., AND RAHWAN, I., Eds. *Argumentation in Artificial Intelligence*. Springer, 2009.
- [SR16] SNAITH, M., AND REED, C. Argument revision. *Journal of Logic and Computation* 27, 7 (2016), 2089–2134.
- [TCT15] THOMOPOULOS, R., CROITORU, M., AND TAMANI, N. Decision support for agri-food chains: A reverse engineering argumentation-based approach. *Ecological Informatics* 26 (2015), 182–191.
- [TG10] THIMM, M., AND GARCÍA, A. J. On strategic argument selection in structured argumentation systems. In *International Workshop on Argumentation in Multi-Agent Systems* (2010), pp. 286–305.
- [TGFS14] TAMARGO, L. H., GARCÍA, A. J., FALAPPA, M. A., AND SIMARI, G. R. On the revision of informant credibility orders. *Artificial Intelligence* 212 (2014), 36–58.
- [TGGS17] TAMARGO, L. H., GOTTIFREDI, S., GARCÍA, A. J., AND SIMARI, G. R. Sharing beliefs among agents with different degrees of credibility. *Knowledge and Information Systems* 50, 3 (2017), 999–1031.
- [TGKIS08] THIMM, M., GARCIA, A. J., KERN-ISBERNER, G., AND SIMARI, G. R. Using collaborations for distributed argumentation with defeasible logic programming. In *Proceedings of the 12th International Workshop on Non-Monotonic Reasoning (NMR'08)* (2008), pp. 179–188.
- [Thi09] THIMM, M. Realizing argumentation in multi-agent systems using defeasible logic programming. In *International Workshop on Argumentation in Multi-Agent Systems* (2009), Springer, pp. 175–194.

- [Thi14] THIMM, M. Strategic argumentation in multi-agent systems. *KI* 28, 3 (2014), 159–168.
- [Ton14] TONI, F. A tutorial on assumption-based argumentation. *Argument & Computation* 5, 1 (2014), 89–117.
- [Tou58] TOULMIN, S. E. *The Uses of Argument*. Cambridge University Press, 1958.
- [vdWDM⁺11] VAN DER WEIDE, T. L., DIGNUM, F., MEYER, J.-J. C., PRAKKEN, H., AND VREESWIJK, G. Multi-criteria argument selection in persuasion dialogues. In *International Workshop on Argumentation in Multi-Agent Systems* (2011), Springer, pp. 136–153.
- [VRO⁺12] VELAGA, N. R., ROTSTEIN, N. D., OREN, N., NELSON, J. D., NORMAN, T. J., AND WRIGHT, S. Development of an integrated flexible transport systems platform for rural areas using argumentation theory. *Research in Transportation Business & Management* 3 (2012), 62–70.
- [Wal10] WALTON, D. *Appeal to expert opinion: Arguments from authority*. Penn State Press, 2010.
- [WCBC12] WARDEH, M., COENEN, F., AND BENCH-CAPON, T. Multi-agent based classification using argumentation from experience. *Autonomous Agents and Multi-Agent Systems* 25, 3 (2012), 447–474.
- [Wis03] WISEAU, T. *The Room* (2003).
- [WK95] WALTON, D., AND KRABBE, E. C. *Commitment in dialogue: Basic concepts of interpersonal reasoning*. SUNY press, 1995.
- [WWW10] WANG, G., WONG, T., AND WANG, X. A negotiation protocol to support agent argumentation and ontology interoperability in mas-based virtual enterprises. In *2010 Seventh International Conference on Information Technology: New Generations* (2010), IEEE, pp. 448–453.
- [YYH⁺11] YUAN, J., YAO, L., HAO, Z., LIU, F., AND YUAN, T. Multi-party dialogue games for distributed argumentation system. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and*

Intelligent Agent Technology-Volume 02 (2011), IEEE Computer Society, pp. 329–332.

- [ZJH10] ZHANG, G., JIANG, G.-R., AND HUANG, T.-Y. Design of argumentation-based multi-agent negotiation system oriented to e-commerce. In *2010 International Conference on Internet Technology and Applications* (2010), IEEE, pp. 1–6.