Actualización de Bases de Conocimiento con Referencias Temporales [1]

(Resumen Extendido [2])

Marcelo A. Falappa - Juan C. Augusto - Guillermo R. Simari
(G.I.R.I.T. [3] - G.I.I.A.[4] - I.C.I.C. [5])
Departamento de Ciencias de la Computación
Universidad Nacional del Sur
Bahía Blanca - Argentina
[ccfalapp,ccaugust,grs]@criba.edu.ar

Key Words: Representación de Conocimiento, Bases de Conocimiento Temporal, Revisión de Creencias, Razonamiento Temporal.

Resumen

Dada una base de conocimiento temporal para modelar las creencias de un agente racional sobre un mundo dinámico, surge la necesidad de revisar sus creencias. En [SA95], se utilizó una base de conocimiento temporal destinada a representar el conocimiento de un agente. A modo de simplificación, dicha base de conocimiento fue asumida estática. Esto es, no se consideró en que orden el conocimiento es agregado, cambiado o eliminado en la misma. El presente trabajo ataca dicho problema proveyendo medios para considerar la dinámica de las creencias del agente modelado.

El hecho de contar con referencias temporales explícitas requiere la consideración de esta información desde dos aspectos distintos. El primero es que la mención de momentos de tiempo como parte del conocimiento adquirido introduce la posibilidad de exponer nuevas contradicciones. El segundo es la mención de un tiempo del sistema necesario para referenciar los distintos estados de conocimiento del agente a medida que ingresa información externa.

Además, algunas consideraciones sintácticas permitieron la homologación de ciertos tipos de contradicción a otros clásicamente conocidos. En base a las simplificaciones antes citadas se brindó un algoritmo para realizar la revisión de una base de conocimiento temporal que asegura la consistencia en la base resultante. En este último caso se considera el tiempo del sistema como medio para guiar el proceso de revisión de creencias.

---

[1]Parcialmente financiado por la Secretaría de Ciencia y Tecnología (Universidad Nacional del Sur).

[2]La versión completa de este trabajo puede ser accedida como Reporte Técnico UNS-DCC-96-XXX del Departamento de Ciencias de la Computación (UNS).

[3]Grupo de Investigación en Representación de Información Temporal.

[4]Grupo de Investigación en Inteligencia Artificial.

[5]Instituto de Ciencias e Ingeniería de la Computación.

1 Introducción

Uno de los problemas más importantes en Inteligencia Artificial es el de la representación de conocimiento de un agente. Este agente se desenvolverá en un ambiente dinámico y recibirá en forma constante nueva información que pasará a formar parte de su estado epistémico. Ese conocimiento debe ser consistente con el fin de poder obtener conclusiones que sean coherentes con el mismo Esta dinámica en el conocimiento se conoce como *cambio de creencias* o *revisión de creencias*. Uno de los principales objetivos de esta disciplina es poder obtener a partir de estados epistémicos en equilibrio y de información externa, nuevos estados epistémicos en equilibrio.

Por su parte, el estudio del tratamiento de las nociones temporales también ha adquirido relevancia dentro del ámbito de la Inteligencia Artificial. Dado que el objetivo del área es capturar la habilidad humana de resolución de problemas, la consideración de conceptos temporales se torna imprescindible. Un agente inteligente utiliza referencias temporales en su comunicación y en sus procesos de deducción en forma constante y decisiva. A modo de ejemplo basta mencionar su relevancia en la comprensión de textos escritos en lenguaje natural, en la elaboración de diagnósticos médicos, en sistemas de detección de fallas y en los sistemas de planificación de actividades en fábricas o robots. Como reflejo de esta relevancia, en los últimos años se han propuesto gran cantidad de formalismos para proveer capacidades de Razonamiento Temporal.

Aquí consideraremos el problema de actualización de una base de conocimiento temporal basada en un lenguaje temporal reificado (ver [AS94] y [SA95]). En la Sección 2 mostraremos el tipo de base de conocimiento temporal a considerar. En la Sección 3 presentaremos consideraciones sintácticas que permiten la homologación de diversos casos de contradicción en una base de conocimiento temporal [FAS96], y en base a las simplificaciones antes citadas se brinda un algoritmo para realizar la revisión de una base de conocimiento temporal que asegura la consistencia en la base de conocimiento resultante.

2 Sobre el Lenguaje Temporal

El conocimiento de un agente $\mathcal{A}^{\mathcal{T}}$ con habilidad para razonar acerca de conceptos temporales será representado a través del sistema formal $\mathbb{L}(\mathcal{T})$que describiremos a continuación. A modo de simplificación, en $\mathbb{L}(\mathcal{T})$ se adopta una ontología discreta basada en *momentos de tiempo*. El dominio temporal es asumido como un conjunto de puntos discretos, ilimitado y linealmente ordenado por una relación de orden, R. Esencialmente, nuestro dominio temporal $\mathcal{T}$ es isomorfo a $\mathbb{Z}$. La sintáxis para el predicado R es:

"R(" *término-temporal* "," *término-temporal* ")"

Reflejando esta característica del dominio temporal, utilizaremos la frase *línea de tiempo* para referenciarlo. La relación R establece la precedencia de los momentos en la linea temporal. Dados dos puntos T y T', "R(T, T')" expresa que T es anterior a T' en la linea de tiempo. De esta forma, dados dos puntos de tiempo, uno precede al otro o referencian el mismo punto en la línea de tiempo. Para facilitar la legibilidad de nuestro lenguaje temporal reemplazaremos "R(T, T')" por $T < T'$, "R(T, T') y R(T', T'')" por $T < T' < T''$, "$\neg$R(T, T') y $\neg$R(T', T)" por $T = T'$.

$\mathbb{L}(\mathcal{T})$ contiene un lenguaje lógico tipado [6], $\mathcal{L}^{\mathcal{T}}$. Los momentos constituyen un tipo distinguido de individuos referenciables. El otro tipo de individuos es determinado por el dominio del problema a ser representado. $\mathcal{L}^{\mathcal{T}}$ es un lenguaje temporal donde pueden utilizarse fórmulas que hacen referencia a la validez de predicados en determinados momentos.

[6]En Inglés, "many-sorted".

Se trata de un lenguaje con menciones explícitas del momento en que un predicado es válido. Los predicados temporales son de la forma $P(\varphi, T)$ donde P es el nombre de un predicado del lenguaje temporal, φ un literal de un lenguaje clásico de primer orden y T es un término temporal. Las fórmulas del lenguaje temporal permiten el uso de cuantificadores aplicados a los predicados temporales. Por ejemplo, $\forall T_1 \exists T_2(\mathrm{OCCURS}(\alpha, T_1) \rightarrow \mathrm{HOLDS}(\neg\beta, T_2))$ es una fórmula bien formada de $\mathcal{L}^{\mathcal{T}}$. Como puede observarse nuestro lenguaje temporal es de tipo "reificado". [7] Esto debe ser entendido como una forma de cambiar el status de algunos elementos del lenguaje. Por ejemplo, *compra(pasaje_avion,juan,10/10/95)* puede ser reescrito como OCCURS(*compra*(*pasaje_avion*, *juan*), 10/10/95) y el evento *compra*(*pasaje_avion*, *juan*) puede ser referido como un objeto de tipo especial.

También pueden ser utilizados predicados que solo referencian momentos tales como aquellos que establecen hechos acerca de la relación de orden entre varios puntos de tiempo. Por ejemplo, como ya hemos propuesto en la sección anterior $\mathrm{R}(T_1, T_2)$ puede ser usado para representar la precedencia de dos momentos. Además de fórmulas que usan etiquetas de tiempo, pueden ser usadas fórmulas atemporales de un lenguaje clásico de primer orden. Tales fórmulas pueden ser reinterpretadas en un contexto temporal como fórmulas válidas para todos los momentos. Esto es, la fórmula $\forall x P(x)$ está en correspondencia con la fórmula temporal $\forall T\ \forall x \mathrm{HOLDS}(P(x), T)$.

Algunos conceptos aparecen como particularmente significativos para la tarea de representar conocimiento temporal. Las nociones de *propiedad, evento* y *acción* que se desarrollan en momentos de tiempo particulares constituyen elementos claves para el razonamiento temporal. Nuestra concepción de tales nociones está fuertemente basada en las expuestas en [AKPT91]. No obstante, dado que no es el tema central del trabajo no realizaremos un análisis del lenguaje utilizado, limitándonos a su ejemplificación. Mayores precisiones del mismo pueden ser obtenidas de [SA95].

En nuestro lenguaje lógico temporal representaremos el suceso de una propiedad mediante el predicado: "HOLDS(" *literal* "," *término-temporal* ")" HOLDS relaciona literales con términos del dominio temporal. Por ejemplo, HOLDS(roja(casa_17),3/5/95) indica que el color de la casa_17 fué rojo para el día 3 de mayo de 1995. El predicado OCCURS es usado para señalar la ocurrencia de eventos en cierto momento de tiempo. Su forma general de uso es: "OCCURS(" *literal-positivo* "," *término-temporal* ")". Tales eventos son el efecto de una acción o conjunto de acciones realizadas por uno o varios agentes. El predicado DO, cuya forma general es: "DO(" *literal-positivo* "," *término-temporal* ")" denota que una acción, designada por medio de un literal positivo, es realizada en un momento de tiempo denotado por el término temporal. Luego, DO(*plantar(jazmín), 21/9/97*) indica la intención de plantar un jazmín en el día de la primavera del proximo año. Utilizando las relaciones primitivas R, HOLDS, OCCURS y DO se pueden definir predicados y reglas de uso más natural y frecuente en el lenguaje cotidiano. Sin embargo, trataremos de mantener un lenguaje simple centrando nuestra atención en la presentación del problema de actualización de la base de conocimiento temporal. De aquí en más asumiremos que cada elemento del espacio temporal es una referencia a un momento de tiempo. Las referencias temporales pueden ser tan diversas como: "lunes", "lunes 5", "5/9/94", "ayer" y "el primer día del año". Para razonar con referencias temporales tan diversas se requieren funciones de traducción a través de las distintas unidades consideradas. Por ejemplo, si conocemos que HOLDS(*rojo_techo_casa_17,enero*) podemos inferir por el principio de homogeneidad de las propiedades [8] que también es válida en cada uno de los días que constituyen el mes. Sin embargo, este problema esta fuera del foco central del presente trabajo; luego asumiremos que dichas referencias temporales utilizan la misma granularidad temporal.

[7] La palabra utilizada en la literatura técnica utilizada en el idioma Inglés es "reified".

[8] Si una propiedad es verdadera sobre un período de tiempo también es verdadera para todo punto de tiempo que integra el período.

3 Operaciones de Cambio en BC Temporales

A continuación, presentaremos las operaciones de cambio sobre bases de conocimiento temporales asumiendo la representación dada anteriormente. Es necesario remarcar que en nuestro sistema existen dos formas de tiempo:

- Uno que se refiere al tiempo denotado por sentencias del lenguaje $\mathcal{L}^{\mathcal{T}}$. Como la base de conocimiento es un subconjunto de $\mathcal{L}^{\mathcal{T}}$, existirán referencias a momentos de tiempo mediante sentencias que utilizan predicados del estilo de OCCURS(*nace(Luis)*,20/06).

- Otro que se refiere a las diferentes bases de conocimiento asociadas a un agente a lo largo del tiempo. Esta segunda noción de tiempo se refiere al tiempo del sistema, es decir, a los diferentes estados de conocimiento del agente que se generan como consecuencia de la incorporación de información externa. Asumiremos que el tiempo del sistema consiste de una estructura isomorfa a la de los números naturales. Es decir, es discreta, lineal, con un momento inicial y sin momento final.

La base de conocimiento $\mathcal{K}^{\mathcal{T}}$ está compuesta por dos conjuntos disjuntos: $\mathcal{K}^{\mathcal{T}}_P$, el cual representa un conjunto de hechos básicos particulares, y $\mathcal{K}^{\mathcal{T}}_G$, el cual representa un conjunto de reglas irrebatibles, válidas en cualquier momento del tiempo. Los hechos particulares de $\mathcal{K}^{\mathcal{T}}_P$ son irrebatibles en un estado de conocimiento, pero pueden dejar de ser aceptados en un estado subsiguiente si se incorpora una nueva creencia en ese momento. En cambio, las reglas que componen el conjunto $\mathcal{K}^{\mathcal{T}}_G$ son válidas en cualquier momento de tiempo, esto es, son aceptadas en todo estado de conocimiento del agente. Gráficamente, podemos representar estas intuiciones a través del siguiente diagrama:

$$
\begin{array}{lllll}
 & \alpha_1 & \alpha_2 & \alpha_n & \text{Entradas Epistémicas} \\
 & \searrow & \searrow & \searrow & \\
\mathcal{K}^{\mathcal{T}}_{P(0)} & \mathcal{K}^{\mathcal{T}}_{P(s(0))} & \mathcal{K}^{\mathcal{T}}_{P(s(s(0)))} & \mathcal{K}^{\mathcal{T}}_{P(s^n(0))} & \\
\hline
\mathcal{K}^{\mathcal{T}}_{G(0)} & \mathcal{K}^{\mathcal{T}}_{G(s(0))} & \mathcal{K}^{\mathcal{T}}_{G(s(s(0)))} & \mathcal{K}^{\mathcal{T}}_{G(s^n(0))} & \text{Tiempo del Sistema}
\end{array}
$$

El subíndice (t) en los conjuntos $\mathcal{K}^{\mathcal{T}}_P$ y $\mathcal{K}^{\mathcal{T}}_G$ indica el estado de conocimiento del agente en el tiempo del sistema t. Asumimos que $t = 0$ es el momento que denota el estado epistémico inicial $({\mathcal{K}^{\mathcal{T}}}_{(0)})$, $t = s(0)$ (donde s se entiende como estado *siguiente* o estado *sucesor*) es el momento siguiente y ${\mathcal{K}^{\mathcal{T}}}_{(s(0))}$ es el estado epistémico resultante después de la primera entrada externa, $\ldots$, $s^n(0)$ el momento resultante después de la n-ésima entrada epistémica, etc. En nuestro sistema, asumimos que el conocimiento de $\mathcal{K}^{\mathcal{T}}_G$ es irrebatible en el tiempo del sistema. Por lo tanto, en caso de realizar una revisión del conocimiento ${\mathcal{K}^{\mathcal{T}}}_{(t)}$ podremos eliminar y/o agregar sentencias a $\mathcal{K}^{\mathcal{T}}_{P(t)}$ sin afectar el contenido de $\mathcal{K}^{\mathcal{T}}_{G(t)}$, *i.e.*, $\mathcal{K}^{\mathcal{T}}_{P(s(t))}$ posiblemente sea distinto de $\mathcal{K}^{\mathcal{T}}_{P(t)}$ pero $\mathcal{K}^{\mathcal{T}}_{G(s(t))} = \mathcal{K}^{\mathcal{T}}_{G(t)}$ para todo momento t del sistema.

Partiendo de las suposiciones dadas anteriormente, podemos definir una operación de cambio sobre una base de conocimiento temporal en el estado de conocimiento t con respecto a una creencia perteneciente a $\mathcal{L}^{\mathcal{T}}$. Con el objeto de evitar notaciones confusas, eliminaremos el indicador $\mathcal{T}$ para indicar que $\mathcal{K}^{\mathcal{T}}$ es una base de conocimiento temporal. En lo sucesivo, notaremos $K_{(t)} = \mathcal{K}_{P(t)} \cup \mathcal{K}_{G(t)}$ asumiendo que $K_{(t)}$ es la base de conocimiento temporal resultante en el momento t del sistema . Esto es, si "#" es una operación de cambio, $K_{(t)}$ es una base de conocimiento temporal en el momento t del sistema , y $P(A,T)$ una sentencia de $\mathcal{L}^{\mathcal{T}}$, [9] entonces

[9] $P(A,T)$ podría ser una sentencia de primer orden reificada del tipo HOLDS(*rinde*(*Juan*), 27/06) o bien del tipo OCCURS(*nace*(*Luis*), 20/03). El argumento t indica el momento del tiempo en que la sentencia A satisface la situación P, y no tiene ninguna relación con el tiempo del sistema t.

$K_{(s(t))} = K_{(t)}{}^{\#P(A,T)}$. El nuevo estado epistémico se obtiene en el siguiente momento del sistema $s(t)$ como consecuencia de modificar el contenido de $K_{(t)}$ con respecto a $P(A, T)$.

Naturalmente, las operaciones de cambio que podemos realizar son variadas y dependen del modelo epistémico adoptado para representar el conocimiento de $K_{(t)}$. En el modelo adoptado anteriormente, utilizamos sentencias del cálculo de predicados reificadas para representar nociones temporales. Adoptaremos las actitudes epistémicas básicas adoptadas en el modelo AGM [AGM85]. Sea $P(A,T)$ una sentencia cualquiera en $\mathcal{L}^T$ y $K_{(t)}$ una base de conocimiento consistente. La actitudes epistémicas que pueden adoptarse para la sentencia $P(A,T)$ con respecto a la base de conocimiento $K_{(t)}$ son las siguientes:

- $P(A,T)$ es **aceptada** en $K_{(t)}$ si y solo si $K_{(t)} \vdash P(A, T)$.
- $P(A,T)$ es **rechazada** en $K_{(t)}$ si y solo si $K_{(t)} \vdash \neg P(A, T)$.
- $P(A,T)$ es **indeterminada** en $K_{(t)}$ si y solo si $K_{(t)} \nvdash P(A, T)$ y $K_{(t)} \nvdash \neg P(A, T)$.

Las operaciones de cambio permiten un cambio de actitud epistémica frente a sentencias de $\mathcal{L}^T$. Sea $K_{(t)}$ una base de conocimiento temporal en el momento t del sistema, $P(A,T)$ una sentencia de $\mathcal{L}^T$. Básicamente, podemos identificar tres operaciones de cambio diferentes:

- **Expansión**: Sea "+" un operador de expansión. La expansión de $K_{(t)}$ por $P(A,T)$ será notada como $K_{(t)}{}^{+P(A,T)}$. Los cambios epistémicos que pueden lograrse mediante una operación de expansión son los siguientes:
  - Si $P(A,T)$ es indeterminada en $K_{(t)}$ entonces $P(A,T)$ es aceptada en $K_{(t)}{}^{+P(A,T)}$.
  - Si $P(A,T)$ es indeterminada en $K_{(t)}$ entonces $P(A,T)$ es rechazada en $K_{(t)}{}^{+\neg P(A,T)}$.
- **Contracción**: Sea "−" un operador de contracción. La contracción de $K_{(t)}$ por $P(A,T)$ será notada como $K_{(t)}{}^{-P(A,T)}$. Los cambios epistémicos que pueden lograrse mediante una operación de contracción son los siguientes:
  - Si $P(A,T)$ es aceptada en $K_{(t)}$ entonces $P(A,T)$ es indeterminada en $K_{(t)}{}^{-P(A,T)}$.
  - Si $P(A,T)$ es rechazada en $K_{(t)}$ entonces $P(A,T)$ es indeterminada en $K_{(t)}{}^{-\neg P(A,T)}$.
- **Revisión**: Sea "*" un operador de revisión. La revisión de $K_{(t)}$ por $P(A,T)$ será notada como $K_{(t)}{}^{*P(A,T)}$. Los cambios epistémicos que pueden lograrse mediante una operación de revisión son los siguientes:
  - Si $P(A,T)$ es aceptada en $K_{(t)}$ entonces $P(A,T)$ es rechazada en $K_{(t)}{}^{*\neg P(A,T)}$.
  - Si $P(A,T)$ es rechazada en $K_{(t)}$ entonces $P(A,T)$ es aceptada en $K_{(t)}{}^{*P(A,T)}$.

La operación de expansión permite agregar nuevo conocimiento al conjunto $K_{(t)}$ de manera indiscriminada, sin prestar atención a las nociones de consistencia. La forma más directa de definir una operación de expansión es la siguiente:

$$K_{(t)}{}^{+P(A,T)} = K_{(t)} \cup \{P(A, T)\}$$

Este tipo de operación es adecuada para el modelo epistémico adoptado para representar el conocimiento mediante una base de conocimiento temporal $\mathcal{K}^T$. Sea Cn un operador de consecuencia con las siguientes propiedades:

- **Clausura:** $Cn(K) = Cn(Cn(K))$.
- **Monotonicidad:** Si $H \subseteq K$ entonces $Cn(H) \subseteq Cn(K)$.
- **Compacidad:** Sea A una sentencia tal que $A \in Cn(K)$. Entonces existe un subconjunto finito de H de K tal que $A \in Cn(H)$.

Si $\mathcal{K}^{\mathcal{T}}$ estuviera clausurada mediante el operador de consecuencia Cn, la operación de expansión se define de la siguiente forma:

$$K_{(t)}{}^{+P(A,T)} = Cn(K_{(t)} \cup \{P(A,\, T)\})$$

Una característica obvia de la operación de expansión, conocida como propiedad de *inclusión*, es que $K_{(t)} \subseteq K_{(t)}{}^{+P(A,T)}$. Si $K_{(t)} \vdash \neg P(A,\, T)$, como $K_{(t)}{}^{+P(A,T)} \vdash P(A,\, T)$, estaríamos en la situación en que $K_{(t)}{}^{+P(A,T)}$ es inconsistente. Esta propiedad hace que la operación de expansión no sea la más indicada para incorporar nuevo conocimiento puesto que permite agregar sentencias a una base de conocimiento de manera indiscriminada sin interesarse en preservar consistencia. Sin embargo, veremos que la operación de expansión puede ser de suma utilidad si se utiliza bajo ciertas restricciones como componente de una operación de cambio más compleja. La operación de contracción consiste en la eliminación de sentencias básicas de un conjunto determinado. Una de las propiedades de la operación de expansión es que $K_{(t)}{}^{-P(A,T)} \subseteq K$. De esto se deduce que la operación de contracción no permite agregar sentencias.

Por su parte, la operación de revisión permite incorporar nuevas creencias pero tratando de respetar dos premisas fundamentales en la teoría de cambio de creencias:

1. Producir un mínimo cambio en el estado epistémico resultante.
2. Preservar consistencia en el conjunto de sentencias revisado.

Sin embargo, como la operación de revisión requiere agregar nuevas sentencias posiblemente eliminando otras para preservar consistencia, no es una tarea simple definir un operador de revisión sobre una base de conocimiento cualquiera, y menos aún sobre una base de conocimiento temporal. Para definir un operador de revisión, utilizaremos la *identidad de Levi*, [AGM85, Gär88, Han93] la cual permite obtener un operador de revisión a partir de la contracción y posterior expansión de un conjunto de sentencias. Formalmente, la **identidad de Levi** es la siguiente:

$$K_{(t)}{}^{*P(A,T)} = (K_{(t)}{}^{-\neg P(A,T)})^{+P(A,T)}$$

El significado de esta igualdad es el siguiente: antes de incorporar consistentemente una sentencia $P(A,T)$ a un conjunto $K_{(t)}$, debemos eliminar a $\neg P(A,\, T)$ de $K_{(t)}$, para luego agregar al conjunto resultante la sentencia $P(A,T)$ sin generar inconsistencia. Puesto que la operación de expansión sobre una base de conocimiento $K_{(t)}$ con respecto a una sentencia $P(A,T)$ consiste solamente en el agregado de $P(A,T)$,[10] debemos interesarnos en definir un operador de contracción apropiado para luego, identidad de Levi mediante, obtener el operador de revisión.

En la siguiente sección presentaremos un operador de contracción aplicable a una base de conocimiento temporal en un momento del sistema determinado.

[10]La razón de esto es que cada estado de conocimiento de un agente es representado mediante una base de conocimiento $K_{(t)}$, la cual es una base de conocimiento temporal en el momento t del sistema , y no está clausurada bajo consecuencia lógica. De ser un conjunto clausurado, la expansión de $K_{(t)}$ con respecto a una sentencia $P(A,T)$ consistiría en agregar $P(A,T)$ y luego clausurar el conjunto resultante bajo consecuencia lógica.

4 El Operador de Contracción

Antes de definir el operador de contracción sobre una base de conocimiento temporal con respecto a una sentencia de $\mathcal{L}^{\mathcal{T}}$, estudiaremos las propiedades que debe satisfacer. Luego, definiremos un algoritmo para obtener un operador de contracción sobre una base de conocimiento.

4.1 Propiedades del Operador de Contracción

Los siguientes postulados, fueron introducidos por Gärdenfors [Gär88] para caracterizar las operaciones de contracción sobre conjuntos clausurados. Los mismos han sido apropiadamente modificados para caracterizar operaciones de contracción sobre conjuntos no clausurados. Sea $K_{(t)}$ una base de conocimiento, α y β sentencias en $\mathcal{L}^{\mathcal{T}}$ y $-$ un operador de contracción.

1. **Clausura**: Si $K_{(t)} = Cn(K_{(t)})$ entonces $K_{(t)}{}^{-\alpha} = Cn(K_{(t)}{}^{-\alpha})$.
2. **Inclusión:** $K_{(t)}{}^{-\alpha} \subseteq K_{(t)}$.
3. **Vacío**: Si $\alpha \notin Cn(K_{(t)})$ entonces $K_{(t)}{}^{-\alpha} = K_{(t)}$.
4. **Exito**: Si $\nvdash \alpha$ entonces $\alpha \notin Cn(K_{(t)}{}^{-\alpha})$.
5. **Recuperación**: $Cn(K_{(t)}) \subseteq Cn((K_{(t)}{}^{-\alpha})^{+\alpha})$.
6. **Preservación**: Si $\vdash \alpha \leftrightarrow \beta$ entonces $Cn(K_{(t)}{}^{-\alpha}) = Cn(K_{(t)}{}^{-\beta})$.
7. **Solapamiento Conjuntivo**: $Cn(K_{(t)}{}^{-\alpha} \cap K_{(t)}{}^{-\beta}) \subseteq Cn(K_{(t)}{}^{-(\alpha\wedge\beta)})$.
8. **Inclusión Conjuntiva** Si $K_{(t)}{}^{-(\alpha\wedge\beta)} \nvdash \alpha$ entonces $Cn(K_{(t)}{}^{-(\alpha\wedge\beta)}) \subseteq Cn(K_{(t)}{}^{-\alpha})$.

Nuestro interés es definir un operador de contracción que satisfaga estos postulados. Sin embargo, puesto que el modelo epistémico adoptado para representar bases de conocimiento temporales consiste en utilizar conjuntos no clausurados, estamos inhabilitados a encontrar un operador que satisfaga *todos* estos postulados. La razón de esto, es que el postulado de recuperación, en general, no se satisface sobre conjuntos no clausurados excepto para casos triviales con lenguajes muy reducidos [Han93, Gär88]. Por las razones anteriormente expuestas, definiremos un operador que satisface los postulados para contracciones salvo recuperación.

Definición 4.1: Sea $K_{(t)}$ una base de conocimiento temporal en el momento t del sistema, y α una sentencia de $\mathcal{L}^{\mathcal{T}}$. Entonces, el conjunto de α-*kernels* [Han93] de $K_{(t)}$ se define como sigue:

$$K_{(t)} \perp\!\!\perp \alpha = \begin{cases} \{X : X \subseteq K_{(t)}, X \vdash \alpha, \text{ y no existe } Y \subset X \text{ tal que } Y \vdash \alpha\} & \text{si } K_{(t)} \vdash \alpha \\ \emptyset & \text{en caso contrario} \end{cases}$$

■

Cada X perteneciente a los α-kernels de $K_{(t)}$ contiene sentencias de $K_{(t)}$ que permiten inferir α. Si se elimina al menos una sentencia de cada α-kernel, podemos garantizar que no puede inferirse α a partir de la base de conocimiento (temporal) contraída. En caso de que α no se infiera a partir de $K_{(t)}$ entonces $K_{(t)} \perp\!\!\perp \alpha = \emptyset$. Aún resta especificar un criterio de preferencia para decidir que elementos de $K_{(t)}$ serán eliminados.

Puesto que podemos modificar solamente el contenido de $\mathcal{K}_{P(t)}$, debemos recurrir a cierta función de selección tal que elija los "peores" elementos de cada α-kernel. Una característica relevante de los sistemas de revisión de creencias es que estos le dan *primacía* [Dal88] a la información nueva. Por lo tanto, cuanto más reciente es la información, más plausible será.

Para mantener consistencia con este criterio, podemos optar por rechazar de cada α-kernel al menos un elemento de $\mathcal{K}_{P(t)}$ que haya sido incorporado más temprano en el tiempo (del sistema).

El criterio de primacía de la información externa no siempre es el más adecuado ya que existen casos particulares en los cuales genera resultados contraintuitivos. En [FS95a, FS95b] se han formulado operadores de revisión que no siguen este criterio, sino que deciden en función de un criterio de comparación entre justificaciones o argumentos lógicos.

A continuación, definiremos una función de selección (dependiente del tiempo) que elija para eliminación aquellos hechos de cada α-kernel menos recientes.

Definición 4.2: Sea $K_{(t)} = \mathcal{K}_{P(t)} \cup \mathcal{K}_{G(t)}$ una base de conocimiento temporal, α un hecho básico de $\mathcal{L}^{\mathcal{T}}$ y $K_{(t)} \perp\!\!\!\perp \alpha \neq \emptyset$ el conjunto de α-kernels de $K_{(t)}$. Entonces, decimos que γ es una *función de selección dependiente del tiempo para* $K_{(t)} \perp\!\!\!\perp \alpha$ si para cada $X \in K_{(t)} \perp\!\!\!\perp \alpha$ se satisfacen las siguientes condiciones:

1. $\gamma X = Y$ tal que $\emptyset \neq Y \subseteq \mathcal{K}_{P(t)}$.[11]
2. $\gamma X = \{Z : Z \in X, Z \in \mathcal{K}_{P(u)}, u \leq t$ y no existe $W \in X$ tal que $W \in \mathcal{K}_{P(v)}, v < u\}$. ∎

La primera condición garantiza que se seleccione un subconjunto no vacío de cada α-kernel y el mismo sea un subconjunto de $\mathcal{K}_{P(t)}$. La segunda condición hace que γ seleccione de cada α-kernel aquellos elementos que fueron incorporados hace más tiempo a la base de conocimiento.

A partir del conjunto de α-kernels y de la función de selección γ sobre cada conjunto del mismo, podemos definir un operador de contracción que satisfaga los postulados expuestos anteriormente:

$$K_{(t)}{}^{-\alpha} = \begin{cases} K_{(t)} - \cup\{\gamma X : X \in K_{(t)} \perp\!\!\!\perp \alpha\} & \text{si } (K_{(t)} \perp\!\!\!\perp \alpha) \neq \emptyset \\ K_{(t)} & \text{en caso contrario} \end{cases}$$

Básicamente, este operador es un operador de *safe contraction* [AM85] o de *kernel contraction* [Han93] sobre $K_{(t)}$. En [Han93] se demuestra que toda operación de kernel contraction es una generalización de una operación de safe contraction. A su vez, es posible caracterizar una operación de safe contraction en función de aquellos elementos que son *seguros con respecto a α en $K_{(t)}$* [AM85] o en términos de α-kernels de $K_{(t)}$ como lo hicimos en este caso.

Las operaciones de kernel contraction o safe contraction satisfacen todos los postulados para contracciones cuando se aplican sobre conjuntos clausurados sobre consecuencia lógica. Cuando se aplican sobre conjuntos no clausurados (las bases de conocimiento temporales utilizadas en nuestro sistema tienen esa característica), generalmente no satisfacen recuperación. Independientemente de esto, para que se respeten el resto de los postulados, es **necesario** que exista una relación de orden *acíclica* [AM85, Han93] entre los elementos del conjunto a contraer.

Definición 4.3: Sea K un conjunto de sentencias y "$\prec$" una relación entre los elementos de K. Entonces "$\prec$" es una relación *acíclica* sobre K si para todo subconjunto $\{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subseteq K$ no es el caso en que $\alpha_1 \prec \alpha_2 \prec \ldots \prec \alpha_n \prec \alpha_1$. ∎

Puesto que nuestro operador de contracción elimina solamente elementos de $\mathcal{K}_{P(t)}$ necesitamos una relación de orden acíclica entre los elementos de $\mathcal{K}_{P(t)}$ en **cada** momento t del sistema. Nuestro sistema determina la importancia de los hechos de $\mathcal{K}_{P(t)}$ en función del momento de arribo de cada hecho.

Definición 4.4: Sea $\{\alpha, \beta\} \subseteq \mathcal{K}_{P(t)}$. Si α fue introducida en un momento s y β fue introducida en un momento u, tales que $u < s < t$, entonces diremos que *β es menos importante que α* y lo notaremos como $\beta \prec \alpha$. ∎

[11]El caso en que $X = \emptyset$ no se considera pues asumimos que se desean incorporar a la base de conocimiento temporal hechos básicos y no teoremas. Puesto que α nunca es un teorema, no puede darse el caso en que $\emptyset \vdash \alpha$.

Para garantizar que "$\prec$" sea acíclica, es necesario asegurar que ninguna sentencia $\alpha \in \mathcal{K}_{P(t)}$ sea reintroducida en la revisión de $K_{(t)}$ con respecto a α. Veremos a continuación que si esto fuera garantizado, tal relación podría no existir. Sea el perdicado INGRESA el que denota el momento del sistema en que un predicado temporal es admitido en la base de conocimiento. Si se tuviera la secuencia de ingreso de información descripta en la definición, obtendríamos el siguiente conjunto representativo del estado actual del sistema:

$$\{\text{INGRESA}(\alpha, u), \text{INGRESA}(\beta, s), \text{INGRESA}(\alpha, t)\}$$

Luego tenemos que α según el criterio descripto en la definición 4.4 es menos importante que si mismo, lo cual es contradictorio.

Teniendo en cuenta las ideas anteriores, podemos definir el operador de revisión de una base de conocimiento temporal $K_{(t)}$ con respecto a un hecho básico $P(A, T) \in \mathcal{L}^T$ basándonos en la identidad de Levi. Sin embargo, antes de incorporar α a la base de conocimiento que resulta de contraer $K_{(t)}$ con respecto a $\neg P(A, T)$, es necesario determinar si $P(A,T)$ pertenece o no a la base de conocimiento contraída.

$$K_{(t)}{}^{*P(A,T)} = \begin{cases} K_{(t)}{}^{*P(A,T)} = K_{(t)}{}^{-\neg P(A,T)} \cup \{P(A,T)\} & \text{si } P(A,T) \notin K_{(t)}{}^{-\neg P(A,T)} \\ K_{(t)}{}^{*P(A,T)} = K_{(t)}{}^{-\neg P(A,T)} & \text{en caso contrario} \end{cases}$$

Cabe observar que una simple unión entre $K_{(t)}{}^{-\neg P(A,T)}$ y $\{P(A,T)\}$ no logra el efecto deseado puesto que, si $P(A,T)$ pertenece a $K_{(t)}{}^{-\neg P(A,T)}$ entonces $P(A,T)$ también pertenece a algún $K_{(u)}$ con $u < t$. Esto significa que $P(A,T)$ ya fue introducida anteriormente. De agregar $P(A,T)$ sin ninguna restricción, la misma ingresaría en dos momentos distintos por lo que se violaría la propiedad de aciclicidad entre las sentencias básicas de $K_{(t)}$

De este modo, el operador de revisión "$*$" permite actualizar el conocimiento de $K_{(t)}$ sin que se reintroduzcan sentencias que fueron incorporadas anteriormente. Esto asegura que se satisfaga la propiedad de aciclicidad entre las sentencias de $K_{(t)}$ en todo momento t del sistema, por lo que el operador de contracción, "$-$", utilizado para definir el de revisión, "$*$", satisface los postulados para contracciones excepto el de recuperación. Por lo tanto, si la base de conocimiento $K_{(t)}$ es actualizada mediante el operador de revisión "$*$", $K_{(t)}$ es consistente en cada momento del sistema t siempre y cuando partamos de una base de conocimiento inicial $K_{(0)}$ consistente.

4.2 Aplicaciones del Operador de Revisión

En esta sección, analizaremos como se comportaría el operador de revisión definido para una base de conocimiento temporal.

Ejemplo 4.1: Supongamos que, en el momento t del sistema, tenemos una base de conocimiento temporal $K_{(t)} = \mathcal{K}_{P(t)} \cup \mathcal{K}_{G(t)}$ con el siguiente conocimiento:
$\mathcal{K}_{P(t)} = \{\text{HOLDS}(\text{nace}(Luis), 05/03)\}$
$\mathcal{K}_{G(t)} = \{(\text{OCCURS}(\text{nace}(X), T_1) \wedge (T_1 \neq T_2)) \rightarrow \neg\text{OCCURS}(\text{nace}(X), T_2)\} \cup \{\text{Otros Axiomas}\}$
El hecho OCCURS(nace($Luis$), 05/03) necesariamente arribó a la base de conocimiento en un momento $s < t$. Supongamos ahora que en el momento t del sistema arriba el hecho $P(A, T)$ = OCCURS(nace($Luis$), 20/03). Obviamente, este hecho básico se contradice con el conocimiento existente en $\mathcal{K}_{P(t)}$. Es de esperar que $K_{(s(t))} = (K_{(t)})^{*P(A,T)}$ permita inferir $P(A,T)$ pero no sea un conjunto contradictorio. Siguiendo el algoritmo propuesto, primeramente debemos tratar de eliminar toda posible demostración de $\neg P(A, T)$, que en nuestro lenguaje temporal equivale a bloquear toda posible prueba de $\neg$OCCURS(nace($Luis$), 20/03). El único $(\neg P(A,T))$-kernel en $K_{(t)}$ es el siguiente:

$$\{\text{OCCURS}(\text{nace}(Luis), 05/03), (\text{OCCURS}(\text{nace}(X), T_1) \wedge (T_1 \neq T_2)) \rightarrow \neg\text{OCCURS}(\text{nace}(X), T_2)\}$$

Como solamente podemos eliminar sentencias de $\mathcal{K}_{P(t)}$, el único predicado posible de eliminar en este caso es OCCURS(nace($Luis$), 05/03). Luego, la base de conocimiento temporal revisada es $K_{(s(t))} = (K_{(t)})^{*(\text{OCCURS(nace}(Luis),20/03))}$ donde:
$\mathcal{K}_{P(t)} = \{\text{OCCURS(nace}(Luis), 20/03)\}$
$\mathcal{K}_{G(t)} = \{(\text{OCCURS(nace}(X), T_1) \wedge (T_1 \neq T_2)) \rightarrow \neg\text{OCCURS(nace}(X), T_2)\} \cup \{\text{Otros Axiomas}\}$

En este caso, existe un solo kernel del cual eliminar sentencias. De existir más de un kernel, debe eliminarse al menos una sentencia básica (esto es, perteneciente a $\mathcal{K}_{P(t)}$) de cada kernel para asegurar que se satisfaga el postulado de éxito en la contracción y de esta forma garantizar consistencia en la base de conocimiento revisada. ■

A su vez, notemos que no solo se obtiene una nueva base de conocimiento temporal, sino que también se modifican las referencias a momentos de tiempo entre los predicados que componen la base de conocimiento. El problema que surge es que no se sabe *a priori* cuál es el predicado de una regla que puede generar la contradiccón junto con la información ingresante. Luego, según el esquema presentado se debe disponer en la base de conocimiento general de tantas reglas como predicados intervinientes haya en cada regla expresando una situación contradictoria. [12]

A continuación, mostraremos como evitar dicho inconveniente. Supongamos tener la siguiente sentencia en $\mathcal{K}_{G(t)}$:

$$\text{OCCURS}(E_1, T_1) \wedge \text{OCCURS}(E_2, T_2) \wedge \text{ANTERIOR}(E_1, E_2) \wedge \neg(T_1 < T_2) \rightarrow \bot$$

Esta sentencia puede reescribirse de varias formas, entre ellas:

$$\text{OCCURS}(E_1, T_1) \wedge \text{OCCURS}(E_2, T_2) \wedge \text{ANTERIOR}(E_1, E_2) \rightarrow (T_1 < T_2)$$

$$\text{OCCURS}(E_1, T_1) \wedge \text{OCCURS}(E_2, T_2) \wedge \neg(T_1 < T_2) \rightarrow \neg\text{ANTERIOR}(E_1, E_2)$$

Para evitar escribir la regla original de varias formas, la representaremos como una disyunción de literales de la forma:

$$\neg\text{OCCURS}(E_1, T_1) \vee \neg\text{OCCURS}(E_2, T_2) \vee \neg\text{ANTERIOR}(E_1, E_2) \vee (T_1 < T_2)$$

Ejemplo 4.2: Consideremos la base de conocimiento $K_{(0)}$ compuesta por los siguientes conjuntos:

$\mathcal{K}_{P(0)} = \{\}$
$\mathcal{K}_{G(0)} = \{\text{OCCURS}(E_1, T_1) \wedge \text{OCCURS}(E_2, T_2) \wedge \text{ANTERIOR}(E_1, E_2) \wedge \neg(T_1 < T_2) \rightarrow \bot,$
$\quad \text{OCCURS(nace}(x), T_1) \wedge \text{OCCURS(muere}(x), T_2) \wedge (T_1 > T_2) \rightarrow \bot\}$

Ahora supongamos que arriban los siguientes hechos en el orden especificado:

1. OCCURS(nace(Bach), 1675). [13]
2. OCCURS(nace(Vivaldi), 1678).
3. ANTERIOR(nace(Vivaldi),nace(Bach)).

La primera entrada epistémica, aunque errónea, no es inconsistente con la base de conocimiento, pues no existe ninguna demostración de ¬OCCURS(nace(Bach), 1675) en $K_{(0)}$. Por lo tanto, la base de conocimiento en el momento $s(0)$ es la siguiente:

[12] Ver una discusión más detallada de este problema en [FAS96].
[13] En realidad Bach nació en 1685.

$$\mathcal{K}_{P(s(0))} = \{\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Bach}), 1675)\}$$
$$\mathcal{K}_{G(s(0))} = \{\mathrm{OCCURS}(E_1, T_1) \wedge \mathrm{OCCURS}(E_2, T_2) \wedge \mathrm{ANTERIOR}(E_1, E_2) \wedge \neg(T_1 < T_2) \rightarrow \perp; \\ \mathrm{OCCURS}(\mathrm{nace}(x), T_1) \wedge \mathrm{OCCURS}(\mathrm{muere}(x), T_2) \wedge (T_1 > T_2) \rightarrow \perp\}$$

La segunda entrada epistémica, tampoco produce inconsistencia, por lo que la base de conocimiento en el momento $s(s(0))$ es la siguiente:

$$\mathcal{K}_{P(s(s(0)))} = \{\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Bach}), 1675), \mathrm{OCCURS}(\mathrm{nace}(\mathrm{Vivaldi}), 1678)\}$$
$$\mathcal{K}_{G(s(s(0)))} = \{\mathrm{OCCURS}(E_1, T_1) \wedge \mathrm{OCCURS}(E_2, T_2) \wedge \mathrm{ANTERIOR}(E_1, E_2) \wedge \neg(T_1 < T_2) \rightarrow \perp, \\ \mathrm{OCCURS}(\mathrm{nace}(x), T_1) \wedge \mathrm{OCCURS}(\mathrm{muere}(x), T_2) \wedge (T_1 > T_2) \rightarrow \perp\}$$

La tercera entrada epistémica es inconsistente con el conocimiento de $K_{(s(s(0)))}$. Esto significa que $K_{(s(s(0)))} \cup \{\mathrm{ANTERIOR}(\mathrm{nace}(\mathrm{Vivaldi}),\mathrm{nace}(\mathrm{Bach}))\} \vdash \perp$. La demostración usando literales complementarios es la siguiente:

1)	$\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Bach}), 1675)$	
2)	$\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Vivaldi}), 1678)$	
3)	$\neg\mathrm{OCCURS}(E_1, T_1) \vee \neg\mathrm{OCCURS}(E_2, T_2) \vee \neg\mathrm{ANTERIOR}(E_1, E_2) \vee (T_1 < T_2)$	
4)	$\neg(1678 < 1675)$	Conocimiento Implícito
5)	$\mathrm{ANTERIOR}(\mathrm{nace}(\mathrm{Vivaldi}),\mathrm{nace}(\mathrm{Bach}))$	
6)	$\neg\mathrm{OCCURS}(E_1, T_1) \vee \neg\mathrm{ANTERIOR}(E_1, \mathrm{nace}(\mathrm{Bach})) \vee (T_1 < 1675)$	1 y 3
7)	$\neg\mathrm{ANTERIOR}(\mathrm{nace}(\mathrm{Vivaldi}), \mathrm{nace}(\mathrm{Bach})) \vee (1678 < 1675)$	2 y 6
8)	$(1678 < 1675)$	5 y 7
9)	$\perp$	4 y 8

Esta es la única demostración para la sentencia $\neg\mathrm{ANTERIOR}(\mathrm{nace}(\mathrm{Vivaldi}),\mathrm{nace}(\mathrm{Bach}))$ en $K_{(s(s(0)))}$. El único kernel en favor de $\neg\mathrm{ANTERIOR}(\mathrm{nace}(\mathrm{Vivaldi}),\mathrm{nace}(\mathrm{Bach}))$ es el siguiente:

$$\{\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Bach}), 1675), \mathrm{OCCURS}(\mathrm{nace}(\mathrm{Vivaldi}), 1678)\}\cup$$
$$\cup\{\neg\mathrm{OCCURS}(E_1, T_1) \vee \neg\mathrm{OCCURS}(E_2, T_2) \vee \neg\mathrm{ANTERIOR}(E_1, E_2) \vee (T_1 < T_2)\}$$

Puesto que podemos eliminar solamente sentencias de $\mathcal{K}_{P(s(s(0)))}$, debemos seleccionar para su eliminación, al menos una sentencia del conjunto:

$$\{\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Bach}), 1675), \mathrm{OCCURS}(\mathrm{nace}(\mathrm{Vivaldi}), 1678)\}$$

Puesto que el hecho $\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Bach}), 1675)$ arribó antes que el hecho $\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Vivaldi}), 1678)\}$, debemos eliminar el primero. De este modo, la base de conocimiento revisada es la siguiente:

$$\mathcal{K}_{P(s(s(s(0))))} = \{\mathrm{OCCURS}(\mathrm{nace}(\mathrm{Vivaldi}), 1678), \mathrm{ANTERIOR}(\mathrm{nace}(\mathrm{Vivaldi}),\mathrm{nace}(\mathrm{Bach}))\}$$
$$\mathcal{K}_{G(s(s(s(0))))} = \{\mathrm{OCCURS}(E_1, T_1) \wedge \mathrm{OCCURS}(E_2, T_2) \wedge \mathrm{ANTERIOR}(E_1, E_2) \wedge \neg(T_1 < T_2) \rightarrow \perp, \\ \mathrm{OCCURS}(\mathrm{nace}(x), T_1) \wedge \mathrm{OCCURS}(\mathrm{muere}(x), T_2 \wedge (T_1 > T_2) \rightarrow \perp\}$$

■

LEMA **4.1** La base de conocimiento temporal permanece consistente al arribar nuevo conocimiento.

Justificación: Tal como fue mencionado en secciones previas, la base de conocimiento inicialmente es asumida consistente. Cuando arriba nuevo conocimiento a $K_{(t)}$ se utiliza una operación de revisión, "*", para mantener su consistencia. Veamos que la misma satisface este objetivo.

Antes de introducir una sentencia en la base de conocimiento se utiliza la identidad de Levi para definir un operador de revisión en función de un operador de contracción. Luego, se determina si la misma no pertenece a la base de conocimiento en cuestión (ver final de la sección 3). Esto permite establecer en cada estado de conocimiento del agente una relación acíclica entre las sentencias del conocimiento particular, ver definiciones 4.3 y 4.4. Dado que cada estado de conocimiento del agente constituye una base de conocimiento no clausurada y que podemos definir una relacion acíclica sobre ellas, cuando se utiliza una operación de kernel-contraction son eliminados los elementos que provocarían contradicción al agregar el nuevo conocimiento. De esta forma cuando arriba el nuevo elemento no produce contradicción. ■

5 Conclusiones y Trabajo Futuro

Dada una base de conocimiento temporal para modelar las creencias de un agente racional sobre un mundo dinámico, surge la necesidad de permitir la actualización de las creencias del agente. En [SA95], se ha utilizado una base de conocimiento temporal destinada a representar el conocimiento de un agente. A modo de simplificación, dicha base de conocimiento fue asumida estática. Es decir, no se consideró en que orden el conocimiento es agregado, cambiado o eliminado en la misma. El presente trabajo consideró dicho problema proveyendo medios para considerar la dinámica de las creencias del agente modelado.

El hecho de contar con referencias temporales explícitas requiere la consideración de información temporal desde dos aspectos distintos. El primer aspecto considerado fue que la mención de momentos de tiempo como parte del conocimiento adquirido introduce la posibilidad de generar nuevos tipos de contradicciones. No obstante, algunas consideraciones sintácticas permitieron la homologación de estos casos a otros clásicamente conocidos. En base a las simplificaciones antes citadas se brindó un algoritmo para realizar la revisión de una base de conocimiento temporal que asegura la consistencia en la base considerada. En este último caso se consideró el tiempo del sistema como medio para guiar el proceso de revisión de creencias.

Referencias

[AGM85] Carlos Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50:510–530, 1985.

[AKPT91] J. Allen, H. Kautz, R. Pelavin, and J. Tenenberg. *Reasoning About Plans.* Morgan Kaufmann, San Mateo, California, 1991.

[AM85] Carlos Alchourrón and David Makinson. On the logic of theory change: Safe contraction. *Studia Logica*, 44:405–422, 1985.

[AS94] J. C. Augusto and G. R. Simari. Un sistema argumentativo con referencias a momentos de tiempo. In *Proceedings de las 23as Jornadas Argentinas en Informática e Investigación Operativa (JAIIO 94)*, pages 81–92. SADIO, Buenos Aires, 1994.

[Dal88] Mukesh Dalal. Investigations into a theory of knowledge base revision: Preliminary report. *Seventh National Conference on Artificial Intelligence*, (AAAI–88):475–479, 1988.

[FAS96] M. Falappa, J. C. Augusto, and G. Simari. Actuallización de bases de conocimiento temporal usando un lenguaje reificado. Technical report, Universidad Nacional del Sur, Departamento de Ciencias de la Computación, 1996.

[FS95a] Marcelo A. Falappa and Guillermo R. Simari. *Condicionales Derrotables a partir de la Revisión de Condicionales Estrictos. Jornadas Argentinas de Informática e Investigación Operativa*, JAIIO '95:7.13–7.28, 1995.

[FS95b] Marcelo A. Falappa and Guillermo R. Simari. *Propiedades del Operador de Revisión mediante Argumentos. Primer Congreso Argentino de Ciencias de la Computación*, CACIC '95:325–336, 1995.

[Gär88] Peter Gärdenfors. *Knowledge in Flux: Modelling the Dynamics of Epistemic States.* The MIT Press, Bradford Books, Cambridge, Massachusetts, 1988.

[Han93] Sven Ove Hansson. A textbook of belief dymanics: Theory change and database updating. first draft. Uppsala University, Departament of Philosophy, Uppsala, Sweden, 1993.

[SA95] G. R. Simari and J. C. Augusto. On the construction of temporal arguments. In *Proceedings del XV Congreso Internacional de Ciencias de la Computación (SCCC'95)*, pages 402–413. SCCC, Arica, Chile, 1995.

Lenguajes de Programación