Hacia una Teoría de Revisión Temporal

María Laura Cobo* Marcelo A. Falappa
Laboratorio de Investigación y Desarrollo en Inteligencia Artificial (L.I.D.I.A)†
Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur
Bahía Blanca - Argentina
fax: +54 291 4595136
[mlcobo,mfalappa]@cs.uns.edu.ar

Palabras Clave: Revisión de Creencias, Lenguajes Lógico-Temporales, Bases de Conocimiento Temporal, Razonamiento Temporal.

Resumen

Uno de los aspectos que no ha sido considerado en forma profunda en el área de revisión de creencias es el de como tratar con información temporal, es decir información que no solo hace referencia al tiempo, sino que también depende de él. Los formalismos actuales de revisión de creencias utilizan principalmente un lenguaje proposicional, sin contar con operadores modales con referencias temporales. Esto representa una limitación importante puesto que el tiempo es un factor determinante para la toma de decisiones, fundamentalmente, en entornos computacionales y en las formalizaciones de éstos. En estos entornos, al producirse un cambio en la información, la relevancia del cambio no está sólo puesta en el *que* sino también en el *cuando* se produce esa modificación. En este trabajo brindaremos las bases para dar comienzo a una teoría de revisión de creencias temporal, en la cual la representación del tiempo es adoptada en la forma de operadores modales temporales.

1 Introducción

Los mecanismos para garantizar la consistencia del estado epistémico de cualquier agente, requiere el desarrollo de métodos de revisión de creencias. Hasta el momento no se han encontrado desarrollos tendientes a revisar conocimiento temporal, es decir, conocimiento que está afectado por el tiempo, no sólo en cuanto al cambio, sino al momento en que tal conocimiento es valedero.

Desde la filosofía y la lógica se ha tratado de capturar la noción de tiempo y, de esta manera, se pueden encontrar diversas alternativas, a veces contrapuestas desde el punto de vista filosófico. Se encuentran así, lógicas temporales que se desprenden de las lógicas modales y en las que la noción de tiempo se encuentra capturada a través de operadores.

*Becaria de la Comisión de Investigaciones Científicas de la provincia de Buenos Aires (CIC)

†Laboratorio LIDIA miembro del IICyTI (Instituto de Investigación en Ciencia y Tecnología Informática)

Las lógicas más conocidas fueron desarrolladas por Arthur Prior [Pri67a, Pri67b]. Estas lógicas han servido de base para muchos lenguajes de especificación [CA00, CA99, Gab87, BFG$^+$96].

En el presente trabajo, se plantea como línea de investigación el desarrollo de métodos de revisión para lenguajes temporales basados en operadores modales.

2 Lenguaje Temporal $\mathcal{L}$

El lenguaje temporal utilizado esta formado por fórmulas de la lógica de primer orden afectadas por operadores temporales. Los operadores involucrados en el lenguaje y su significado intuitivo son:

$\boxplus\,\phi$	"Siempre en el futuro ϕ"
$\boxminus\,\phi$	"Siempre en el pasado ϕ"
$\Box\,\phi$	$\boxminus\phi \wedge \phi \wedge \boxplus\phi$ ("Siempre en el pasado, ahora y futuro")
$\diamondplus\,\phi$	"Alguna vez en el futuro ϕ"
$\diamondminus\,\phi$	"Alguna vez en el pasado ϕ"
$\Diamond\,\phi$	$\diamondminus\phi \vee \phi \vee \diamondplus\phi$ ("Alguna vez el pasado, ahora o en el futuro")
$\oplus\,\phi$	"En el próximo instante ϕ"
$\ominus\,\phi$	"En el instante previo ϕ"
$\mathcal{S}(\phi,\psi)$	"ϕ desde que se verifica ψ"
$\mathcal{U}(\phi,\psi)$	"ϕ hasta que se verifica ψ"

Los operadores pueden ser caracterizados por distintas propiedades, entre ellas las siguientes:

Reflexividad: esta propiedad establece que la semántica del operador debe ser involucrada en forma directa al momento de evaluación. Esto es, la fórmula afectada por un operador temporal reflexivo asegurará la veracidad de la fórmula en el momento de evaluación, *presente*, y/o los momentos que correspondan de acuerdo al operador. De esta manera, el hecho de que operador $\diamondplus$ cumpla con esta propiedad indicará que la sentencia $\diamondplus\alpha$ será verdadera si α es verdadera en el momento *presente* o si lo es en cualquier momento del futuro. En el caso de los operadores binarios como *Since* y *Until* la *reflexividad* del operador implica que, para un par de proposiciones α y β, puede darse que β sea verdadera en el momento de la evaluación de la sentencia provocando que $\mathcal{S}(\alpha,\beta)$ se verifique ante la veracidad de β en ese momento de tiempo. En cambio si se pide que verifique *irreflexividad*, α debe, al menos, ser verdadera en el presente, para que la proposición en cuestión sea veraz.

Fuerte: esta propiedad asegura la existencia de un próximo instante donde la proposición debe verificarse. El contrapuesto, un operador *débil*, no asegura siempre la existencia de un próximo instante de tiempo. La restricción de que un operador binario como *Since*(α,β) sea *fuerte* implica que la proposición β debe ser verdadera en algún momento, mientras que exigir que sea *débil* implica que puede darse el caso de que β no sea verdadera en ningún momento. Esto podría darse si siempre se deduce α y no se deduce nunca β de la base de conocimiento.

3 Aspectos importante sobre el lenguaje temporal $\mathcal{L}$

Desde el punto de vista de la implemantabilidad del sistema de revisión existen ciertos aspectos que deben ser tenidos en cuenta al momento de determinar el lenguaje a utilizar. Es claro que la posibilidad de utilizar conjuntos clausurados es inaplicable en un contexto real de implementación. Por esta razón, el uso de bases de conocimiento (conjuntos no necesariamente clausurados) resulta una aproximación más adecuada.

Teniendo en cuenta estos aspectos de eficiencia se puede notar que el lenguaje presentado anteriormente presenta ciertas falencias que pueden llegar a comprometer la implementabilidad de la teoría que se intenta desarrollar. Este problema es usualmente conocido en los medios científicos como el *problema de actualización de bases de datos*. En el entorno particular planteado en este trabajo, la base de información sería una base deductiva, utilizando las formulas lógicas para representar la información de una manera declarativa. La noción de tiempo y cambio conduce a ciertos problemas. En primer término, puede ser interesante mantener la historia de los estados previos, no sólo el estado actual. Esto puede aplicarse a contextos reales como la evolución de pacientes en hospitales (historias clínicas), eventos en un entorno industrial, etc. y por otro lado aspectos importantes de eficiencia en la toma de decisiones. Aplicando esta idea al problema de lograr alguna forma de revisión, el problema cobra un sentido aún mayor. Presentándose por otro lado los problemas eficiencia, que no son despreciables,

Desde el punto de vista de los lenguajes temporales, la capacidad de manipular esta noción se ve afectada por el grado de precisión temporal requerido. Además, está la delicada decisión de que concepción temporal se adoptará en el lenguaje. El lenguaje elegido adopta parte de los estudios de Arthur Prior [Pri67a, Pri67b]. Prior siguió una de las formas de representar el tiempo más conocidas, A-series (pasado, presente y futuro). Otros investigadores siguieron la otra representación, B-series (antes, después), donde se utilizan referencias explícitas de tiempo. La lógica desarrollada por Rescher [RU71] conduce a los lenguajes más conocidos que formalizan esta concepción temporal. Existen circunstancias, en las cuales puede ser muy útil contar con ambas formas de referenciar el tiempo simultáneamente, a pesar de que filosóficamente éstas sean contrapuestas.

El problema de considerar lenguajes basados en A-series es el costo de mantener la base actualizada. Esto se debe a que el momento presente cambia continuamente, y es necesario que la base refleje ese cambio. Por ejemplo, todos los hechos que son verdaderos en el presente deben modificarse para que sean verdaderos en el pasado, y algunos hechos del futuro pasarán a ser verdaderos en el (*nuevo*) presente [Kow92]. Si el escenario es altamente dinámico, el proceso de mantener la base actualizada puede llegar a ser prohibitivamente costoso desde el punto de vista computacional.

Ejemplo 1 Suponga que se cuenta con la siguiente base que almacena información sobre vuelos de cabotaje:

$$
\begin{aligned}
BaseConocimiento_1 = \{ \quad & \diamondminus\, abordando\_vuelo\_1, \\
& \diamondminus\, vuelo\_1, \\
& \diamondminus\, abordando\_vuelo\_2, \\
& \diamondminus\, vuelo\_2, \\
& abordando\_vuelo\_3, \\
& \diamondplus\, vuelo\_3, \\
& \diamondplus\, abordando\_vuelo\_4, \\
& \diamondplus\, vuelo\_4\}
\end{aligned}
$$

una actualización debe llevarse a cabo para ajustar la base a la realidad. En el ejemplo, luego de la partida de $vuelo\_3$, se debe actualizar $BaseConocimiento_1$ a la siguiente base:

$$
\begin{array}{rl}
BaseConocimiento_2 = \{ & \diamondminus\, abordando\_vuelo\_1, \\
& \diamondminus\, vuelo\_1, \\
& \diamondminus\, abordando\_vuelo\_2, \\
& \diamondminus\, vuelo\_2, \\
& \diamondminus\, abordando\_vuelo\_3, \\
& \diamondplus\, vuelo\_3, \\
& abordando\_vuelo\_4, \\
& \diamondplus\, vuelo\_4\}
\end{array}
$$

Para evitar estos problemas, se pueden combinar las dos concepciones temporales mencionadas a través del uso de un operador temporal prefijo $\mathsf{At}(\alpha,\ n)$. Este operador liga la sentencia α a un momento específico de tiempo n. El significado intuitivo de este nuevo operador es:

$$\mathsf{At}(\phi,\ n) \quad \text{“}\phi \text{ es verdadera en el momento } n\text{-ésimo”}$$

EJEMPLO **2** Utilizando este operador $\mathsf{At}(\alpha,\ n)$ se puede ligar la información de las bases anteriores como se muestra en $BaseConocimiento_3$. En este caso, no se requiere ningún tipo de modificación una vez que el $vuelo_3$ despegó:

$$
\begin{array}{rl}
BaseConocimiento_3 = \{ & \mathsf{At}(abordando\_vuelo\_1,\ 10:00), \\
& \mathsf{At}(vuelo\_1,\ 10:30am), \\
& \mathsf{At}(abordando\_vuelo\_2,\ 11:00), \\
& \mathsf{At}(vuelo\_2,\ 11:30am), \\
& \mathsf{At}(abordando\_vuelo\_3,\ 13:00), \\
& \mathsf{At}(vuelo\_3,\ 01:30pm), \\
& \mathsf{At}(abordando\_vuelo\_4,\ 13:15), \\
& \mathsf{At}(vuelo\_4,\ 13:45)\}
\end{array}
$$

Por otro lado, si se decide mantener un lenguaje basado solo en las B-series entonces no se permite ningún tipo de imprecisión en cuanto al momento de ocurrencia de un evento. Este tipo de información es bastante común en los entornos de razonamiento temporal, razón por la cual no contar con ella supone una pérdida importante en cuanto a expresividad.

Definimos luego el lenguaje $\mathcal{L}^{\mathcal{A}}$, el cual se obtiene aplicando el operador prefijo $\mathsf{At}(\alpha,\ n)$ a cualquier sentencia α perteneciente a $\mathcal{L}$.

4 Hacia una teoría de revisión temporal

A continuación, se presentarán las características que deben tener las operaciones de cambio sobre bases de conocimiento temporales asumiendo la representación dada anteriormente. Es necesario remarcar que el tiempo se puede referenciar en dos estratos diferentes en la teoría de revisión:

- Uno es el tiempo denotado en las sentencias del lenguaje $\mathcal{L}$.

- Otro el tiempo del sistema. Este tiempo irá registrando la historia del estado del conocimiento. La estructura temporal del sistema se considerará discreta con un momento inicial y sin momento final, ya que se desconoce el momento en el cuál el sistema dejará de funcionar.

Llamaremos $\mathcal{K}_{\mathcal{T}}$ a la base de conocimiento que representa el conocimiento en el momento $\mathcal{T}$ del sistema. La misma, está compuesta por fórmulas temporales pertenecientes al lenguaje $\mathcal{L}^{\mathcal{A}}$. La base $\mathcal{K}_{\mathcal{T}}$ debe ser consistente y mantenerse de esa forma sin importar que momento esté representando $\mathcal{T}$. Por lo tanto, frente a una sentencia α cualquiera del sistema, se pueden adoptar las siguientes actitudes epistémicas con respecto a la base de conocimiento actual:

- α es **aceptada** en $\mathcal{K}_{\mathcal{T}}$ si y solo si $\mathcal{K}_{\mathcal{T}} \vdash \alpha$
- α es **rechazada** en $\mathcal{K}_{\mathcal{T}}$ si y solo si $\mathcal{K}_{\mathcal{T}} \vdash \neg\alpha$
- α es **indeterminada** en $\mathcal{K}_{\mathcal{T}}$ si y solo si $\mathcal{K}_{\mathcal{T}} \not\vdash \alpha$ y $\mathcal{K}_{\mathcal{T}} \not\vdash \neg\alpha$

Es decir α es aceptada en la base de conocimiento $\mathcal{K}_{\mathcal{T}}$, si se puede inferir de ésta, es rechazada si se infiere su negación e indeterminada si no se infiere de la base ni α ni su negación.

Las operaciones de cambio permiten cambiar la actitud epistémica que se determina para una sentencia cualquiera del lenguaje $\mathcal{L}$, definiendo operaciones de cambio. Básicamente existen tres operaciones de cambio diferentes:

Expansión: Sea $+$ un operador de expansión. La expansión de $\mathcal{K}_{\mathcal{T}}$ por α será notada como $\mathcal{K}_{\mathcal{T}} + \alpha$. Los cambios epistémicos que pueden obtenerse son:

- Si α es indeterminada en $\mathcal{K}_{\mathcal{T}}$ entonces α es aceptada en $\mathcal{K}_{\mathcal{T}} + \alpha$
- Si α es indeterminada en $\mathcal{K}_{\mathcal{T}}$ entonces α es rechazada en $\mathcal{K}_{\mathcal{T}} + (\neg\ \alpha)$

Contracción: Sea $\sim$ un operador de contracción. La contracción de $\mathcal{K}_{\mathcal{T}}$ por α será notada como $\mathcal{K}_{\mathcal{T}} \sim \alpha$. Los cambios epistémicos que pueden obtenerse luego de una contracción son:

- Si α es aceptada en $\mathcal{K}_{\mathcal{T}}$ entonces α es indeterminada en $\mathcal{K}_{\mathcal{T}} \sim \alpha$
- Si α es rechazada en $\mathcal{K}_{\mathcal{T}}$ entonces α es indeterminada en $\mathcal{K}_{\mathcal{T}} \sim (\neg\ \alpha)$

Revisión: Sea $\star$ un operador de revisión. La revisión de $\mathcal{K}_{\mathcal{T}}$ por α será notada como $\mathcal{K}_{\mathcal{T}} \star \alpha$. Los cambios epistémicos que pueden lograrse luego de una revisión son:

- Si α es aceptada en $\mathcal{K}_{\mathcal{T}}$ entonces α es rechazada en $\mathcal{K}_{\mathcal{T}} \star (\neg\ \alpha)$
- Si α es rechazada en $\mathcal{K}_{\mathcal{T}}$ entonces α es aceptada en $\mathcal{K}_{\mathcal{T}} \star \alpha$

En las expansiones solo se permite agregar conocimiento nuevo, mientras que en las contracciones solo se puede eliminar conocimiento. Por su parte, el operador de revisión es el más complejo ya que permite la incorporación de nuevas creencias pero intentando respetar dos hipótesis importantes de la teoría de cambio de creencias: *mantener consistencia en la base de conocimiento* y *producir el menor cambio posible en el estado* *epistémico que se obtiene*. Obtener un operador de revisión apropiado para $\mathcal{L}$ no es trivial, razón por la cual en las próximas secciones se presentarán las propiedades o postulados que los operadores deberían satisfacer, sin hacer referencia a la construcción de los mismos. De esta manera se presentarán postulados básicos, basados en los presentados en el modelos AGM [AGM85] y un conjunto de postulados nuevos, destinados a mostrar la dinámica de la información temporal.

4.1 Postulados básicos para los operadores de cambio

Sean α y β sentencias de un lenguaje $\mathcal{L}$ presentado anteriormente $\sim$ un operador de contracción. Reformulando los postulados de los modelos tradicionales de cambio, podríamos obtener los siguientes postulados básicos para contracciones:

Clausura: Si $\mathcal{K}_\mathcal{T} = Cn(\mathcal{K}_\mathcal{T})$ entonces ${\mathcal{K}_\mathcal{T}}^{\sim\alpha} = Cn({\mathcal{K}_\mathcal{T}}^{\sim\alpha})$.

Inclusión: ${\mathcal{K}_\mathcal{T}}^{\sim\alpha} \subseteq \mathcal{K}_\mathcal{T}$.

Vacío: Si $\alpha \notin Cn(\mathcal{K}_\mathcal{T})$ entonces ${\mathcal{K}_\mathcal{T}}^{\sim\alpha} = \mathcal{K}_\mathcal{T}$.

Éxito: Si $\alpha \in \mathcal{K}_\mathcal{T}$ entonces $\alpha \notin Cn({\mathcal{K}_\mathcal{T}}^{\sim\alpha})$.

Recuperación: $\mathcal{K}_\mathcal{T} \subseteq ({\mathcal{K}_\mathcal{T}}^{\sim\alpha})^{+\alpha}$.

Preservación: Si $\vdash \alpha \leftrightarrow \beta$ entonces $Cn({\mathcal{K}_\mathcal{T}}^{\sim\alpha}) = Cn({\mathcal{K}_\mathcal{T}}^{\sim\beta})$.

Es importante destacar que estos postulados están basados en los presentados como parte del modelo AGM [AGM85, Gä88]. Se recuerda que $\mathcal{K}_\mathcal{T}$ representa la base de conocimiento en el momento $\mathcal{T}$ del sistema.

De la misma manera, contamos con postulados análogos para un operador de revisión, sea $\star$ un operador de revisión. Reformulando nuevamente los postulados de los modelos tradicionales de cambio, podríamos obtener los siguientes postulados básicos para revisiones:

Clausura: Si $\mathcal{K}_\mathcal{T} = Cn(\mathcal{K}_\mathcal{T})$ entonces ${\mathcal{K}_\mathcal{T}}^{\star\alpha} = Cn({\mathcal{K}_\mathcal{T}}^{\star\alpha})$.

Éxito: $\alpha \in {\mathcal{K}_\mathcal{T}}^{\star\alpha}$.

Inclusión: ${\mathcal{K}_\mathcal{T}}^{\star\alpha} \subseteq {\mathcal{K}_\mathcal{T}}^{+\alpha}$.

Vacío: Si $\neg\alpha \notin Cn(\mathcal{K}_\mathcal{T})$ entonces ${\mathcal{K}_\mathcal{T}}^{\star\alpha} = {\mathcal{K}_\mathcal{T}}^{+\alpha}$.

Consistencia: Si $\not\vdash \neg\alpha$ entonces $Cn(\mathcal{K}_\mathcal{T})^{\star\alpha} \neq K_\perp$.

Preservación: Si $\vdash \alpha \leftrightarrow \beta$ entonces $Cn({\mathcal{K}_\mathcal{T}}^{\star\alpha}) = Cn({\mathcal{K}_\mathcal{T}}^{\star\beta})$.

4.2 Postulados adicionales para los operadores de cambio

Para el lenguaje $\mathcal{L}$ se propusieron los siguientes postulados [CF03], asumiendo que $\mathcal{L}^e$ representa el conjunto de sentencias específicas, esto es, aquellas sentencias de la forma $\oplus\alpha$ o $\ominus\alpha$, con $\alpha \in \mathcal{L}$. Se cuneta entonces con los siguientes postulados adicionales:

Clausura temporal: Si $\vdash \alpha$ entonces $\mathcal{K_T}^{\star\beta} \vdash \Box\alpha, \forall\beta \in \mathcal{L}$.

Este postulado plantea la necesidad de contar con todos los teoremas en forma temporal, esto es, los teoremas son válidos en todos los momentos de tiempo. Es por eso que, sin importar con que fórmula β del lenguaje se esté revisando la base de conocimiento, se deducirá a la misma la sentencia *"siempre α"*, para cualquier teorema α.

Prioridad de información específica: [1]

En el Futuro: (a) Si $\mathcal{K_T} \vdash \oplus\alpha$ entonces $\mathcal{K_T}^{\star\beta} \nvdash \Diamond\alpha, \forall\beta \in \mathcal{L}$.
(b) Si $\mathcal{K_T} \vdash \oplus\alpha$ entonces $\mathcal{K_T}^{\star\beta} \nvdash \diamondplus\alpha, \forall\beta \in \mathcal{L}$.

En el Pasado: (a) Si $\mathcal{K_T} \vdash \ominus\alpha$ entonces $\mathcal{K_T}^{\star\beta} \nvdash \Diamond\alpha, \forall\beta \in \mathcal{L}$.
(b) Si $\mathcal{K_T} \vdash \ominus\alpha$ entonces $\mathcal{K_T}^{\star\beta} \nvdash \diamondminus\alpha, \forall\beta \in \mathcal{L}$.

Desde el punto de vista del conocimiento que se desea mantener, es de suma utilidad tratar de mantener la mayor cantidad de conocimiento temporalmente específico, ya que de alguna manera representa un conocimiento más fino o detallado del entorno.

Prioridad de información $\Diamond$: Si $\mathcal{K_T} \vdash \Diamond\neg\alpha$ entonces $\mathcal{K_T}^{\star\beta} \nvdash \Box\alpha, \forall\beta \in (\mathcal{L} - \mathcal{L}^e)$.

De manera análoga se tienen los siguientes postulados para sentencias de $\mathcal{L}$ afectadas por los operadores $\diamondplus$ o $\diamondminus$:

En el Futuro: Si $\mathcal{K_T} \vdash \diamondplus\neg\alpha$ entonces $\mathcal{K_T}^{\star\beta} \nvdash \boxplus\alpha$ y $\mathcal{K_T}^{\star\beta} \nvdash \Box\alpha, \forall\beta \in (\mathcal{L} - \mathcal{L}^e)$.

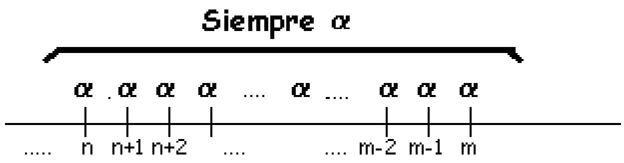
En el Pasado: Si $\mathcal{K_T} \vdash \diamondminus\neg\alpha$ entonces $\mathcal{K_T}^{\star\beta} \nvdash \boxminus\alpha$ y $\mathcal{K_T}^{\star\beta} \nvdash \Box\alpha, \forall\beta \in (\mathcal{L} - \mathcal{L}^e)$.

De la misma manera que en el conjunto de postulados anteriores se prefiere la información más concreta, desde el punto de vista temporal, en este caso se plantea la preferencia de la información que es verdadera *"alguna vez"* sobre la que *"siempre"* lo es. Esto se debe a que en los entornos más concretos de aplicación, la información del tipo *siempre* es menos probable o creíble desde el punto de vista temporal. Se puede pensar en cualquier entorno probable o conocido, donde el conocimiento de las posibles excepciones es más enriquecedor que el conocer que una determinada cosa sea verdadera por siempre.

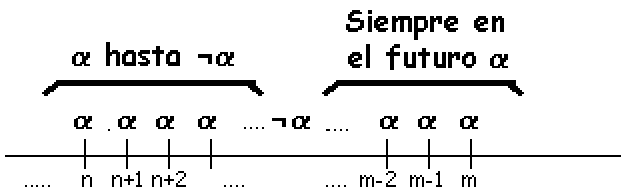
[1] El lenguaje temporal que se presentó cuenta con operadores temporales que capturan el tiempo de una forma imprecisa. Se cuenta con un momento de tiempo destacado, usualmente llamado *"presente"*, y todas las referencias temporales hechas a través de los operadores dependen de ese momento destacado. Se puede observar que la definición que la mayoría de estos operadores conduce a especificaciones de tiempo vagas. Por ejemplo, $\diamondplus\alpha$ representa que α será verdadera en algún momento del futuro pero no se tiene precisión del momento de tiempo exacto en el que lo será. Los únicos operadores que proveen una cierta precisión sobre el momento en el que la formula afectada por ellos es verdadera son los operadores $\oplus$ (*en el próximo momento*) y $\ominus$, (*en el momento previo*).

Excepciones:

> **En el Futuro:** Si $\mathcal{K}_\mathcal{T} \vdash \Box\alpha$ entonces $Cn({\mathcal{K}_\mathcal{T}}^{\star\diamondplus\neg\alpha}) = Cn(({\mathcal{K}_\mathcal{T}}^{\sim\Box\,\alpha})^{+[\mathcal{U}(\alpha,\neg\alpha)\wedge\diamondplus\neg\alpha\wedge\boxminus\alpha]})$.
>
> **En el Pasado:** Si $\mathcal{K}_\mathcal{T} \vdash \Box\alpha$ entonces $Cn({\mathcal{K}_\mathcal{T}}^{\star\diamondminus\neg\alpha}) = Cn(({\mathcal{K}_\mathcal{T}}^{\sim\Box\,\alpha})^{+[\mathcal{S}(\alpha,\neg\alpha)\wedge\diamondminus\neg\alpha\wedge\boxplus\alpha]})$.
>
> Esta excepciones tratan con la necesidad de mantener la mayor cantidad de información original como sea posible. Al conocer una determinada información α como verdadera en todos los momentos de tiempo y revisar con respecto a la posibilidad de que esa misma información sea falsa en algún momento, lo que podemos concluir es que α seguirá siendo verdadera desde o hasta (según el caso) que se compruebe que en determinado momento α es falsa.
>
> Veamos gráficamente el resultado de revisar utilizando este postulado en su versión *en el futuro*



Base $\mathcal{K}_\mathcal{T}$ que deduce $\Box\alpha$



Base ${\mathcal{K}_\mathcal{T}}^{\star\diamondplus\neg\alpha}$

Aceptación de Contraejemplo: Si $\mathcal{K}_\mathcal{T} \vdash \Box\alpha$ entonces $Cn({\mathcal{K}_\mathcal{T}}^{\star\neg\alpha}) = Cn(({\mathcal{K}_\mathcal{T}}^{\sim\Box\,\alpha})^{+(\boxplus\,\alpha\wedge\boxminus\,\alpha)})$

> Este postulado presenta la capacidad de mantener la mayor cantidad de información temporal una vez que se produce un cambio en el momento destacado o *"presente"*. De esta manera, a pesar del contraejemplo, se puede seguir manteniendo la información anterior para el resto de la estructura temporal.

Es importante destacar que algunos de los postulados deben ser considerados para otros operadores temporales del lenguaje $\mathcal{L}$. Por otra parte, si tomamos el lenguaje $\mathcal{L}^\mathcal{A}$ completo, en lugar del fragmento $\mathcal{L}$, algunos postulados adicionales se requieren para poder revisar correctamente la información que está afectada por algún operador $\mathsf{At}(\alpha,\ n)$.

Prioridad de información anterior: Si $\mathcal{K}_\mathcal{T} \vdash \mathsf{At}(\alpha,\ n)$ entonces:

1. ${\mathcal{K}_\mathcal{T}}^{\star\mathsf{At}(\alpha,\ m)} = \mathcal{K}_\mathcal{T}$ solo si $n \leq m$
2. ${\mathcal{K}_\mathcal{T}}^{\star\mathsf{At}(\alpha,\ m)} = ({\mathcal{K}_\mathcal{T}}^{\sim\mathsf{At}(\alpha,\ n)})^{+\mathsf{At}(\alpha,\ m)}$ solo si $n > m$

En este postulado se establece la necesidad de mantener la información en el espectro temporal más amplio posible. Así por ejemplo, si se revisa la base de conocimiento con información que ya está presente en la base pero ligada a un momento de tiempo diferente, siempre se preferirá la información más antigua, desde el punto de vista temporal. Es importante notar que el postulado 1 no respeta el postulado de éxito completamente, ya que de alguna forma la sentencia se puede deducir de la información previa.

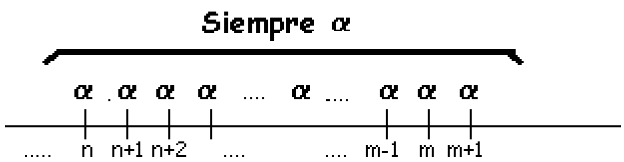
Este postulado solo es aplicable sobre aquellas sentencias $\alpha \in \mathcal{L}^e$, es decir, aquellas que no estén afectadas por los operadores $\oplus$ y $\ominus$. Esto se debe a que su precisión temporal establece un vínculo importante con el momento en que son afirmadas.

Descomposición: Si $\mathcal{K}_\mathcal{T} \vdash \mathsf{At}(\boxplus\alpha,\ n)$ entonces

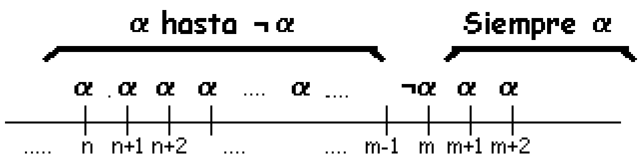
$$\mathcal{K}_\mathcal{T}{}^{\star \mathsf{At}(\neg\alpha,\ m)} = (\mathcal{K}_\mathcal{T}{}^{\sim[\mathsf{At}(\alpha,\ n)])+[\mathsf{At}(\mathcal{U}(\alpha,\neg\alpha),\ m)\wedge \mathsf{At}(\boxplus\alpha,\ m)]} \text{ solo si } n < m$$

La necesidad de mantener la mayor cantidad de información temporal lleva a que cada vez que se cuente con información temporal de la forma *"siempre"*, se intentará mantener la sentencia en la mayor cantidad de momentos posibles.

Gráficamente la situación podría verse de la siguiente manera:



Base $\mathcal{K}_\mathcal{T}$ que deduce $\mathsf{At}(\boxplus\alpha,\ n)$



Base $\mathcal{K}_\mathcal{T}{}^{\star \mathsf{At}(\neg\alpha,\ m)}$

Preferencia de hechos: Si $\mathcal{K}_\mathcal{T} \vdash \mathsf{At}(\oplus\alpha,\ n)$ entonces:

$$\mathcal{K}_\mathcal{T}{}^{\star\ \mathsf{At}(\alpha,\ n+1)} = (\mathcal{K}_\mathcal{T}{}^{\sim \mathsf{At}(\oplus\alpha,\ n)})^{+\mathsf{At}(\alpha,\ n+1)}$$

Los operadores temporales le agregan dificultad al lenguaje, es por eso que en este postulado se establece que siempre que sea posible se preferirá la información que esté afectada por la menor cantidad de operadores temporales.

5 Conclusiones y Trabajos Futuros

En este trabajo se presentan los postulados preliminares para el desarrollo de una teoría de cambio de creencias temporal sobre conjuntos no clausurados. Es claro que los postulados presentados también son aplicables a conjuntos clausurados de sentencias pero el hincapié en este trabajo está en la ventaja que aportan las bases de conocimiento sobre los conjuntos clausurados. Es necesario definir algoritmos para operadores de contracción y revisión que respeten los postulados presentados en este trabajo. Como trabajo futuro, resta obtener teoremas de representación de estos operadores, esto es, dar definiciones constructivas, postulados básicos, y demostrar que todo operador construido con los algoritmos formulados satisface los postulados propuestos y viceversa.

Referencias

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

[BFG+96] H. Barringer, M. Fisher, D. Gabbay, R. Owens, and M. Reynolds. *The Imperative Future: Principles of Executable Temporal Logic.* Research Studies Press Ltd., 1996.

[CA99] M. L. Cobo and J. C. Augusto. Fundamentos lógicos e implementación de una extensión a temporal prolog. *The Journal of Computer Science and Technology (JCS&T), sponsored by ISTEC (Iberoamerican Science & Technology Education Consortium*, 1(2):22–36, 1999.

[CA00] M. L. Cobo and J. C. Augusto. Towards a Programming Language Based on Prior's Metric Temporal Operators. In *Proceedings del VI Congreso Argentino de Cs. de la Computación, CACiC2000*, pages 453 –464, 2000.

[CF03] M. L. Cobo and M. A. Falappa. Revisando bases modales temporales. Presentado en el Workshop de Investigadores de Ciencias de la Computación (WICC'03), realizado en la ciudad de Tandil los días 23 y 24 de Mayo de 2003, 2003.

[Gä88] P. Gärdenfors. *Knowledge in Flux: Modelling the Dynamics of Epistemic States.* The MIT Press, Bradford Books, Cambridge, Massachusetts, 1988.

[Gab87] D. M. Gabbay. Modal and temporal logic programming. In Antony Galton, editor, *Temporal Logic and their Applications*, pages 197–236. Academic Press, 1987.

[Kow92] Robert Kowalski. Database updates in the event calculus. *Journal of Logic Programming*, 12:121–146, 1992.

[Pri67a] A. Prior. *Past, Present and Future.* Clarendon Press, 1967.

[Pri67b] A. Prior. Stratified metric tense logic. *Theoria 33*, pages 28–38, 1967.

[RU71] Nicholas Rescher and Alasdair Urquhart. *Temporal Logic.* Springer-Verlag, 1971.