

# Reelaboración de Planes en Agentes Inteligentes. Expansión de Grafos de Planning.

**Gerardo PARRA**

Departamento de Informática y Estadística  
UNIVERSIDAD NACIONAL DEL COMAHUE  
Neuquén – República Argentina  
e-mail: [gparra@uncoma.edu.ar](mailto:gparra@uncoma.edu.ar)

**Guillermo SIMARI**

Departamento de Ciencias de la Computación  
UNIVERSIDAD NACIONAL DEL SUR  
Buenos Aires – República Argentina  
e-mail: [grs@cs.uns.edu.ar](mailto:grs@cs.uns.edu.ar)

**Palabras Clave:** INTELIGENCIA ARTIFICIAL, PLANEAMIENTO, DINÁMICA DE CREENCIAS

## Resumen

Los agentes inteligentes autónomos, por su proactividad, se ven obligados a considerar la satisfacción de sus metas a través de un conjunto estructurado de acciones que conforman un plan. El modelo BDI (*Belief, Desires and Intentions*) para representar el conjunto cognitivo de un agente, es una posibilidad interesante que permite estudiar el problema que introduce el dinamismo natural del entorno en el que un plan particular se desenvuelve.

El dinamismo del entorno provoca que algunos de los planes deban ser modificados para poder alcanzar las metas finales. Es esta una actividad de replaneamiento.

En este trabajo, postulamos la conveniencia de adoptar el punto de vista del área de Dinámica de Creencias al considerar la actividad de replaneamiento de un agente inteligente. En esta primera aproximación, introducimos un modelo para representar expansiones en grafos de planning. Definimos un operador de expansión de grafos de planning, se introducen sus postulados característicos y su definición constructiva. El objetivo principal es la reutilización, en gran medida, del grafo de planning original.

## 1 Introducción

Los agentes inteligentes autónomos, por su proactividad, se ven obligados a considerar la satisfacción de sus metas a través de un conjunto estructurado de acciones que conforman un plan. El modelo BDI (*Belief, Desires and Intentions*)[10] para representar el conjunto cognitivo de un agente es una posibilidad interesante que permite estudiar el problema que introduce el dinamismo natural del entorno en el que un plan particular se desenvuelve. El entorno corriente, el mundo actual del agente, es representado con un modelo de creencias (*beliefs*) adecuado. Las metas del agente representan sus deseos (*desires*) y describen en formal parcial estados del entorno preferidos. Finalmente, los planes para alcanzar alguno de aquellos estados constituyen, en cierta forma, las intenciones.

El dinamismo del entorno provoca que algunos de los planes deban ser modificados para poder alcanzar las metas finales. Esta actividad de replaneamiento es, en realidad, una revisión del mismo. Ciertas partes pueden ser conservadas, pero otras deben ser removidas y reemplazadas por subplanes convenientes que ofrezcan la posibilidad de éxito para el plan global.

En este trabajo, postulamos la conveniencia de adoptar el punto de vista del área de Dinámica de Creencias[5,6,9] al considerar la actividad de replaneamiento de un agente inteligente. En esta primera aproximación, presentamos un modelo para representar expansiones en grafos de planning. El objetivo principal es la reutilización, en gran medida, del grafo de planning original.

El trabajo está organizado de la siguiente manera. En la sección 2, se presentan los principales conceptos relacionados al área de planning en Inteligencia Artificial. Se analiza un dispositivo de planning particular y se trata, en ese contexto, la construcción de grafos de planning. En la siguiente sección, presentamos los aspectos más relevantes relacionados con la dinámica de creencias. La sección 4 contiene las principales contribuciones de este trabajo. Se define un operador de expansión de grafos de planning, se introducen sus postulados característicos y su definición constructiva. Finalmente, la sección 5 incluye las conclusiones del trabajo, así como también las consideraciones sobre trabajos futuros.

## 2 Elaboración de Planes - Graphplan

El objetivo central del área de *planning* en el contexto de Inteligencia Artificial es construir algoritmos que hagan posible a un agente elaborar un curso de acción para lograr sus metas. El resultado producido por un dispositivo de planning (*planner*) es una secuencia de acciones que, cuando son ejecutadas en un mundo que satisface la descripción del estado inicial, permitirán la obtención de la meta. Existe una amplia variedad de lenguajes para representar el mundo, las metas del agente y las acciones posibles. En este trabajo de investigación, adoptamos, en primera instancia, la representación STRIPS[1] como lenguaje de representación.

Uno de los más recientes dispositivos de planning es Graphplan[2,3]. Se caracteriza, fundamentalmente, por ser un simple y elegante algoritmo que produce un planner muy eficiente. El funcionamiento de Graphplan alterna entre dos fases: la *construcción del grafo de planning* y la *extracción de la solución*. La primera fase construye un *grafo de planning* hacia delante en el tiempo hasta que se logra una condición necesaria (pero que puede ser insuficiente) para la existencia de un plan. Luego, la fase de extracción de solución realiza un recorrido hacia atrás sobre el grafo, buscando un plan que resuelva el problema. Si no es hallada una solución, el ciclo se repite mediante la construcción de un nuevo nivel del grafo de planning<sup>1</sup>.

Un problema de planning muy conocido es el *dinner-date problem*[4]. Este problema consiste en encontrar un plan para preparar una cita sorpresa a nuestra amada que se encuentra durmiendo. La meta del problema es sacar las bolsas de basura, preparar la cena y envolver un regalo. Existen cuatro acciones posibles: *cocinar*, *envolver*, *llevar\_a\_mano* y *llevar\_en\_carretilla*. La acción de *cocinar* requiere *manosLimpias* y produce *cena*. La acción *envolver* tiene como precondition *silencio* porque es una sorpresa y produce *regalo*. *Llevar\_a\_mano* elimina *basura* pero, el contacto manual con las bolsas, niega *manosLimpias*. La acción final, *llevar\_en\_carretilla*, también elimina *basura* pero, a causa del desplazamiento ruidoso, niega *silencio*. Inicialmente, tenemos *manosLimpias*, la casa tiene *basura* y está en *silencio*. Las demás proposiciones se consideran falsas.

Graphplan determina un plan para el *dinner-date problem* de la siguiente manera. En primer lugar construye el grafo de planning hasta que las metas del problema aparezcan como nodos del grafo. Luego, realiza un recorrido hacia

---

<sup>1</sup> Debido a las normas de presentación establecidas en cuanto a la longitud del artículo, se omite una descripción detallada de grafos de planning. Tal descripción, presente en las referencias propuestas, se incluirá en la versión extendida de este trabajo.

atrás sobre los nodos del grafo con el fin de encontrar un conjunto de acciones, no mutuamente excluyentes entre sí, que permitan lograr las metas a partir de las condiciones iniciales.

El problema de planning introducido tiene como finalidad plantear algunos de nuestros objetivos. Por ejemplo, ¿qué sucedería si, una vez hallado el plan definitivo, recordamos que a nuestra amada le encantan los postres? Ante esta situación sería necesaria una tarea de replaneamiento, puesto que sería necesario incorporar una nueva meta (*postre*) a la especificación del problema. Supongamos, por simplicidad, que se dispone de una acción adicional *preparar\_postre* con precondition *manosLimpias* y que produce *postre*. El mayor problema, desde un punto de vista computacional, es que el grafo de planning debe volver a construirse desde cero para intentar encontrar la solución del nuevo problema. En las siguientes secciones, consideramos la conveniencia de adoptar el punto de vista del área de Dinámica de Creencias con el fin de simplificar el problema.

### 3 Dinámica de Creencias

Uno de las más fundamentales aproximaciones a la formalización de la dinámica de creencias es el modelo AGM[5]. En este enfoque, las creencias de un agente inteligente son representadas mediante conjuntos de sentencias cerrados bajo consecuencia lógica.

**Notación:** Se adopta un lenguaje proposicional  $L$  con un conjunto completo de conectivos booleanos: negación, conjunción, disyunción e implicación. Las fórmulas en  $L$  serán denotadas por letras griegas minúsculas y los conjuntos de sentencias en  $L$  serán denotadas mediante letras mayúsculas. Se emplea un operador de consecuencia  $Cn$ .  $Cn$  toma un conjunto de sentencias en  $L$  y produce un nuevo conjunto de sentencias. Se asume que el operador  $Cn$  satisface las propiedades de *inclusión* ( $A \subseteq Cn(A)$ ), *iteración* ( $Cn(A) = Cn(Cn(A))$ ) y *monotonidad* (si  $A \subseteq B$  entonces  $Cn(A) \subseteq Cn(B)$ ).

Sea  $\mathbf{K} = Cn(\mathbf{K})$  un conjunto de creencias y  $\alpha$  una sentencia en un lenguaje proposicional  $L$ . Los tres principales tipos de operaciones de cambio de creencias son los siguientes[6]:

- **Expansión:** Una nueva sentencia es incorporada a un estado epistémico. Si '+' es un operador de expansión entonces  $\mathbf{K} + \alpha$  denota el conjunto de creencias  $\mathbf{K}$  expandido por  $\alpha$ .
- **Contracción:** Alguna sentencia presente en el estado epistémico es retraída sin incorporar nueva información. Si '-' es un operador de contracción entonces  $\mathbf{K} - \alpha$  denota el conjunto de creencias  $\mathbf{K}$  contraído por  $\alpha$ .
- **Revisión:** Una nueva sentencia es incorporada de manera consistente al estado epistémico. Con el fin de hacer posible esta operación, algunas sentencias deben ser retraídas del estado epistémico original. Si '\*' es un operador de revisión entonces  $\mathbf{K} * \alpha$  denota el conjunto de creencias  $\mathbf{K}$  revisado por  $\alpha$ .

Gärdenfors[6] propone los siguientes postulados básicos para la operación de expansión:

- ( $\mathbf{K}^+$  1) **Clausura:**  $\mathbf{K} + \alpha = Cn(\mathbf{K} + \alpha)$ .
- ( $\mathbf{K}^+$  2) **Exito:**  $\alpha \in (\mathbf{K} + \alpha)$ .
- ( $\mathbf{K}^+$  3) **Inclusión:**  $\mathbf{K} \subseteq (\mathbf{K} + \alpha)$ .
- ( $\mathbf{K}^+$  4) **Vacuidad:** Si  $\alpha \in \mathbf{K}$  entonces  $(\mathbf{K} + \alpha) = \mathbf{K}$ .
- ( $\mathbf{K}^+$  5) **Monotonidad:** Si  $\mathbf{K} \subseteq \mathbf{H}$  entonces  $(\mathbf{K} + \alpha) \subseteq (\mathbf{H} + \alpha)$ .

La operación de expansión puede ser definida explícitamente tomando la clausura lógica del conjunto de creencias  $\mathbf{K}$  unido a la sentencia  $\alpha$ . Es decir,

$$(\mathbf{K} + \alpha) = Cn(\mathbf{K} \cup \{\alpha\}).$$

Existe un teorema de representación que prueba la equivalencia entre esta definición operativa y los postulados mencionados[6]. No analizaremos los postulados propuestos para las operaciones de contracción y revisión, puesto que no son necesarios para el resto del trabajo. Una presentación exhaustiva de los diferentes modelos de cambios de creencias puede hallarse en [7,8].

## 4 Expansión de Grafos de Planning

La construcción del grafo de planning para un problema determinado no es una tarea trivial. Por lo tanto, sería interesante conservar buena parte del grafo ante una modificación del problema original. Si consideramos el *dinner-date problem*, el recordar que a nuestra amada le encantan los postres originaría la necesidad, en el contexto de Graphplan, de reconstruir el grafo de planning desde el nivel cero.

Con el fin de evitar este problema, en este trabajo proponemos la definición de una operación de expansión para grafos de planning. La idea fundamental es tratar de conservar tanto como sea posible del grafo de planning original. En primer lugar, necesitamos tratar a cada acción y a todas sus precondiciones y poscondiciones como una unidad.

**Definición 4.1.** Un *esquema de acción* es una terna  $(Pre, a, Pos)$  donde,  $a$  es una acción,  $Pre$  es un conjunto finito de proposiciones que constituyen las precondiciones de  $a$  y  $Pos$  es un conjunto finito de proposiciones que se verifican luego de aplicar la acción  $a$  (poscondiciones de  $a$ ).

A continuación, necesitamos especificar cuándo un esquema de acción pertenece a un grafo de planning.

**Definición 4.2.** Sea  $\Pi$  un grafo de planning y sea  $A = (Pre, a, Pos)$  un esquema de acción. Decimos que  $A$  pertenece a  $\Pi$  si la acción  $a$  y cada uno de los elementos de los conjuntos  $Pre$  y  $Pos$  existen como nodos en el grafo y si existen los arcos que modelan las relaciones correspondientes.

**Definición 4.3.** Sea  $\Pi$  un grafo de planning. Sea  $n$  un nodo de proposición o de acción perteneciente al grafo. Mediante la función  $Lev_{\Pi}(n)$  se indica el nivel correspondiente al nodo  $n$  en el grafo  $\Pi$ .

En este punto, ya estamos en condiciones de definir un operador de expansión para grafos de planning.

**Definición 4.4.** Sea  $\Pi$  la clase de los grafos de planning y sea  $A = (Pre, a, Pos)$  un esquema de acción. El *operador de expansión a nivel  $i$* , denotado  $\oplus^i$ , se define de la siguiente manera:

$$\oplus^i : \Pi \times A \rightarrow \Pi.$$

Dado un grafo de planning y un esquema de acción, esta función devuelve un nuevo grafo de planning  $\Pi'$ . En este grafo, la acción  $a$  se halla presente en el nivel de acción  $i$ , las precondiciones y las poscondiciones de  $a$  se encuentran en los niveles  $i - 1$  e  $i + 1$ , respectivamente y, además, existen los arcos, en  $\Pi'$ , que vinculan a los nodos involucrados.

A continuación, discutiremos un conjunto de postulados de racionalidad que el operador de expansión debería satisfacer. Sea  $\Pi$  un grafo de planning y sea  $A = (Pre, a, Pos)$  un esquema de acción.

**(P<sup>+</sup> 1) Éxito:**  $A \in (\Pi \oplus^i A)$ .

Establecer nuevas relaciones en el grafo de planning puede ser un proceso costoso. Consecuentemente, se impone el requerimiento de que las nuevas relaciones sean incorporadas al grafo.

**(P<sup>+</sup> 2) Inclusión:**  $\Pi \subseteq (\Pi \oplus^i A)$ .

La idea clave aquí es que, deberíamos retener tanto de nuestras relaciones previas como sea posible, cuando se realiza una operación de expansión sobre el grafo de planning.

(P<sup>+</sup> 3) **Vacuidad:** Si  $A$  pertenece a  $\Pi$  y  $Lev(a) = i$  entonces  $(\Pi \oplus^i A) = \Pi$ .

Lo que establece este postulado es que no hay relaciones que sean perdidas o ganadas por la incorporación de información redundante al grafo de planning.

(P<sup>+</sup> 4) **Monotonicidad:** Si  $\Pi_1 \subseteq \Pi_2$  entonces  $(\Pi_1 \oplus^i A) \subseteq (\Pi_2 \oplus^i A)$ .

Supongamos que el grafo de planning  $\Pi_2$  tiene más información que el grafo  $\Pi_1$ . Si  $\Pi_1$  y  $\Pi_2$  son expandidos por el esquema de acción  $A$ , entonces es natural que  $\Pi_1 \oplus^i A$  no contenga información que no esté también incluida en  $\Pi_2 \oplus^i A$ .

En este punto, introduciremos una definición constructiva de la expansión de grafos de planning.

**Definición 4.5.** Sea  $\Pi$  un grafo de planning y sea  $A = (Pre, a, Pos)$  un esquema de acción. Definimos la expansión a nivel  $i$  de  $\Pi$  por  $A$  como

$\Pi \oplus^i A = \Pi \cup \{(pre, a) : pre \in Pre\} \cup \{(a, pos) : pos \in Pos\} = \Pi'$ , donde  
 $Lev_{\Pi'}(pre) = i - 1$  para todo  $pre \in Pre$ ,  $Lev_{\Pi'}(a) = i$  y  $Lev_{\Pi'}(pos) = i + 1$  para todo  $pos \in Pos$ .

A partir de esta definición del operador de expansión, es posible establecer una vinculación muy importante y original con los postulados de racionalidad propuestos. El siguiente lema resume, entonces, algunas de las propiedades de interés del operador de expansión de grafos de planning.

**Lema 4.1.** Sea  $\oplus^i$  un operador de expansión de acuerdo a la Definición 4.5. Entonces  $\oplus^i$  satisface los postulados de éxito, inclusión, vacuidad y monotonicidad.

## 5 Comentarios Finales

La contribución principal de este trabajo de investigación es la introducción de un modelo para representar *operaciones de cambio* en grafos de planning. Hemos presentado un operador de expansión de grafos de planning, caracterizándolo mediante un conjunto de propiedades deseables. Además, hemos introducido su definición constructiva.

La definición de este operador de expansión hace posible la reutilización de gran parte del grafo de planning original. En trabajos futuros, se tratará la definición de operadores de contracción y revisión de grafos de planning, su caracterización mediante postulados de racionalidad y su definición constructiva.

## Referencias

- [1] R. Fikes and N. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *J. Artificial Intelligence*, 2(3/4), 1971.
- [2] A. Blum and M. Furst. Fast planning through planning graph analysis. In *Proceedings of the XIV International Joint Conference of AI*, pages 1636-1642, 1995.
- [3] A. Blum and M. Furst. Fast planning through planning graph analysis. *J. Artificial Intelligence*, 90(1-2):281-300, 1997.
- [4] D. S. Weld. Recent Advances in AI Planning. *AI Magazine*, 1999.

- [5] 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.
- [6] P. Gärdenfors. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. The MIT Press, Bradford Books, Cambridge, Massachusetts, 1988.
- [7] M. Falappa. *Teoría de Cambio de Creencias y sus Aplicaciones sobre Estados de Conocimiento*. Tesis Doctoral, Universidad Nacional del Sur, Bahía Blanca, 1999.
- [8] G. Parra. *Semi Revisión Plausible en Bases de Creencias*. Tesis de Magister, Universidad Nacional del Sur, Bahía Blanca, 1998.
- [9] S. O. Hansson. *A Textbook of Belief Dynamics*. Kluwer Academic Press, 1996.
- [10] M. Georgeff, B. Pell, M. Pollack, M. Tambe, and M. Wooldridge. The Belief-Desire-Intention Model of Agency. In J.P.Müller, M.P.Singh, and A.S. Rao, editors, *Intelligent Agents V* (LNAI Volume 1555), pages 1-10. Springer-Verlag: Berlin, Germany, 1999.