

Universidad Nacional de Educación a Distancia
Ingeniería Técnica en Informática de Sistemas
Introducción a la Inteligencia Artificial (2º curso)
Soluciones exámenes junio 2007

Ejercicio 1. Primera Semana.

Considere un sistema basado en reglas con la siguiente base de conocimiento:

R ₁ :	SI $C_1(C_2(x))$	ENTONCES $C_3(x, C_2(x))$
R ₂ :	SI $C_3(x, y)$	ENTONCES $C_2(C_1(y))$
R ₃ :	SI $C_1(x)$ Y $C_2(y)$	ENTONCES $C_3(x, y)$
R ₄ :	SI $C_2(C_1(x))$ Y $C_1(y)$	ENTONCES $C_3(y, x)$
R ₅ :	SI $C_1(x)$ Y $C_2(y)$ Y $C_3(z, y)$	ENTONCES $C_3(y, z)$
R ₆ :	SI $C_3(x, x)$ Y $C_2(y)$	ENTONCES $C_3(x, C_1(y))$
R ₇ :	SI $C_1(x)$	ENTONCES $C_3(x, C_1(x))$
R ₈ :	SI $C_2(x)$ Y $C_3(x, y)$	ENTONCES $C_1(y)$
R ₉ :	SI $C_3(x, C_1(y))$	ENTONCES $C_3(y, x)$
R ₁₀ :	SI $C_3(C_2(x), y)$	ENTONCES $C_3(x, y)$

donde C_i ($i \in \{1, 2, 3\}$) representa un determinado predicado, $\{x, y, z\}$ es un conjunto de variables y $\{a, b\}$ es un conjunto de objetos. La base de hechos inicial es $BH_0 = \{C_1(a), C_2(b)\}$. Suponga que el citado sistema se encarga de que siempre sean asignados valores diferentes a variables diferentes dentro de un mismo antecedente de una regla. La resolución de conflictos en el sistema utiliza el siguiente criterio: en primer lugar se aplica un mecanismo de refractariedad según el cual la última regla ejecutada no podrá ser ejecutada en los dos ciclos siguientes, en segundo lugar se aplica un criterio de especificidad definido únicamente por las siguientes relaciones: $R_3 > R_7$ y $R_5 > R_7$ (donde “>” significa “es más específica que”) y que da preferencia a las reglas más específicas, en tercer lugar se elegirían las instanciaciones cuya regla asociada posee el mayor subíndice y, finalmente, entre instanciaciones correspondientes a la misma regla se prefiere aquella instanciación que haga referencia a información más actual.

Ejecute y explique detalladamente los pasos correspondientes a un proceso de encadenamiento hacia delante en el sistema descrito. El encadenamiento finalizará tan pronto como se llegue a un ciclo en que se intente añadir a la base de hechos un hecho que ya exista en la misma.

SOLUCIÓN:

En este ejercicio, cada expresión iniciada por C_i ($i \in \{1, 2, 3\}$) se entiende como un determinado patrón. Quizá, en el enunciado del ejercicio, “predicado” no sea la palabra más adecuada para definir los C_i . Una definición más adecuada sería “término de un patrón”.

Cuando se consideran los C_i estrictamente como predicados lógicos, es obvio que el sistema descrito no es válido. C_i no sería estrictamente un predicado, ya que aparece en ocasiones en el argumento de otro “predicado”. Es decir, allí donde debería aparecer un objeto, nos encontramos un valor booleano.

Aceptaremos estos dos puntos de vista en las contestaciones de los alumnos: como un problema de encadenamiento con comparación de patrones o como un problema donde intervienen predicados lógicos con el problema comentado anteriormente. En la presente solución incluimos el primero de los puntos de vista, que se corresponde con la filosofía inicial del ejercicio.

Ciclo 0:

$$BH_0=\{C_1(a), C_2(b)\}$$

Obviamente, en este ciclo no se aplica refractariedad. De las dos reglas que se pueden ejecutar, R_3 y R_7 , el criterio de especificidad nos indica que se ha de ejecutar R_3 . Como sólo hay una instanciación correspondiente al antecedente de R_3 , $\{C_1(a) \text{ Y } C_2(b)\}$, se ejecuta dicha instanciación.

Ciclo 1:

$$BH_1=\{C_1(a), C_2(b), C_3(a, b)\}$$

En este ciclo no se aplica R_3 por refractariedad. Dado que R_2 y R_7 serían las únicas reglas que se pueden ejecutar, no se aplicaría especificidad. Se elegiría R_7 para ser ejecutada, ya que su subíndice es mayor que el de R_2 . Como sólo hay una instanciación correspondiente al antecedente de R_7 , $\{C_1(a)\}$, se ejecuta dicha instanciación.

Ciclo 2:

$$BH_2=\{C_1(a), C_2(b), C_3(a, b), C_3(a, C_1(a))\}$$

En este ciclo no se aplican R_3 y R_7 por refractariedad. Dado que únicamente R_2 puede ser ejecutada, no es necesario aplicar el criterio de especificidad, así como tampoco el de elegir la regla con mayor subíndice. Debido a que existen dos instanciaciones correspondientes al antecedente de R_2 , $\{C_3(a, b)\}$ y $\{C_3(a, C_1(a))\}$, se elige la segunda de ellas debido al criterio de actualidad. Efectivamente, el hecho $C_3(a, b)$ fue incorporado a la base de hechos al final del ciclo 0, mientras que el hecho $C_3(a, C_1(a))$ lo fue al final del ciclo 1.

Ciclo 3:

$$BH_3=\{C_1(a), C_2(b), C_3(a, b), C_3(a, C_1(a)), C_2(C_1(C_1(a)))\}$$

En este ciclo no se aplican R_2 y R_7 por refractariedad. Dado que R_3 y R_4 serían las únicas reglas que se pueden ejecutar, no se aplicaría especificidad. Se elegiría R_4 para ser ejecutada, ya que su subíndice es mayor que el de R_3 . Como sólo hay una instanciación correspondiente al antecedente de R_4 , $\{C_2(C_1(C_1(a))) \text{ Y } C_1(a)\}$, se ejecuta dicha instanciación. Debido a que el hecho $C_3(a, C_1(a))$ ya se encuentra en la base de hechos, el proceso de encadenamiento hacia delante finaliza en este ciclo.

Ejercicio 2. Primera Semana.

Enumere los diferentes tipos de lógicas que resultan de la ampliación de la Lógica Clásica y resuma brevemente qué ventaja principal aporta cada una de dichas lógicas frente a la Lógica Clásica. Dado el siguiente razonamiento:

“Posiblemente, si me muero entonces me hagan un homenaje póstumo”

“Necesariamente he de morir”

Por tanto, “es posible que me hagan un homenaje póstumo”

dé una prueba del mismo.

SOLUCIÓN

- *Lógica de predicados de orden superior*: predicados cuantificados y en argumentos
- *Lógica de predicados con identidad*: introducción del predicado identidad
- *Lógicas multivaluadas*: más de dos valores de verdad
- *Lógica modal*: nuevos modos de verdad, a saber “necesidad” y “posibilidad”
- *Lógica difusa*: tratamiento de la imprecisión y subjetividad propia del lenguaje
- *Lógicas no monótonas*: tratan el carácter no monótono del razonamiento humano (explíquese brevemente en qué consiste la no monotonía).

Sea:

$A \equiv$ me muero

$B \equiv$ recibo un homenaje póstumo

Entonces, habría que probar:

$\Diamond (A \rightarrow B)$

$\Box A$

$\therefore \Diamond B$

Considérese la siguiente prueba por reducción al absurdo:

1. $\Diamond (A \rightarrow B)$
2. $\Box A$
3. $\neg (\Diamond B)$ (Negación de la conclusión)
4. $W1: A \rightarrow B$ (A partir de 1, donde W1 es cierto mundo)
5. $W1: A$ (A partir de 2)
6. $W1: B$ (A partir de 4 y 5)
7. $\Box (\neg B)$ (A partir de 3)
8. $W1: \neg B$ (A partir de 7, entrando en contradicción con 6)

Ejercicio 3. Primera Semana.

Realice un estudio comparativo entre el *Método AO**, también conocido como *Método YO**, y la *Búsqueda Primero el Mejor*.

SOLUCIÓN:

- Los dos son métodos de búsqueda heurística en un espacio de estados.

- Tipo de estructura de datos sobre la que representan el proceso de búsqueda:

Método YO*: grafo YO (defínase grafo YO y el coste asociado a un subgrafo YO).

Búsqueda Primero el Mejor: grafo O (defínase grafo O y el coste asociado a un camino en dicho grafo).

- Tipo de función de evaluación heurística utilizada:

Método YO*: Anota para cada nodo N cuál es el mejor enlace E hacia sus nodos hijos. El coste del subgrafo que parte del nodo N y contiene al enlace E es asociado a N como valor de su función de evaluación heurística. Cuando se atraviesa un arco en dicho subgrafo, se suma su coste (normalmente 1). Cuando se llega a un nodo intermedio, se toma el mejor enlace hacia sus nodos hijos. Finalmente, cuando se llega a un nodo hoja, se suma el valor de su función de evaluación heurística, que estima el mínimo coste de un subgrafo solución que parta de dicho nodo hoja.

Método Primero el Mejor: Utiliza como valor de función de evaluación heurística para cada nodo la estimación del coste del mejor camino solución desde dicho nodo hasta una meta.

- Tipos de estructuras de datos auxiliares que utilizan:

Método YO*: conjunto S que almacena el conjunto de nodos cuyo valor de la función de evaluación heurística queda por actualizar en la fase ascendente del algoritmo.

Búsqueda Primero el Mejor: listas ABIERTA y CERRADA (defínense), así como el árbol parcial de costes mínimos (defínase).

- Forma en que se decide la siguiente parte del espacio de búsqueda a explorar:

Método YO*: se expande cualquiera de los nodos hoja del mejor subgrafo parcial hallado para el nodo inicial.

Búsqueda Primero el Mejor: Se escoge el nodo más prometedor de ABIERTA

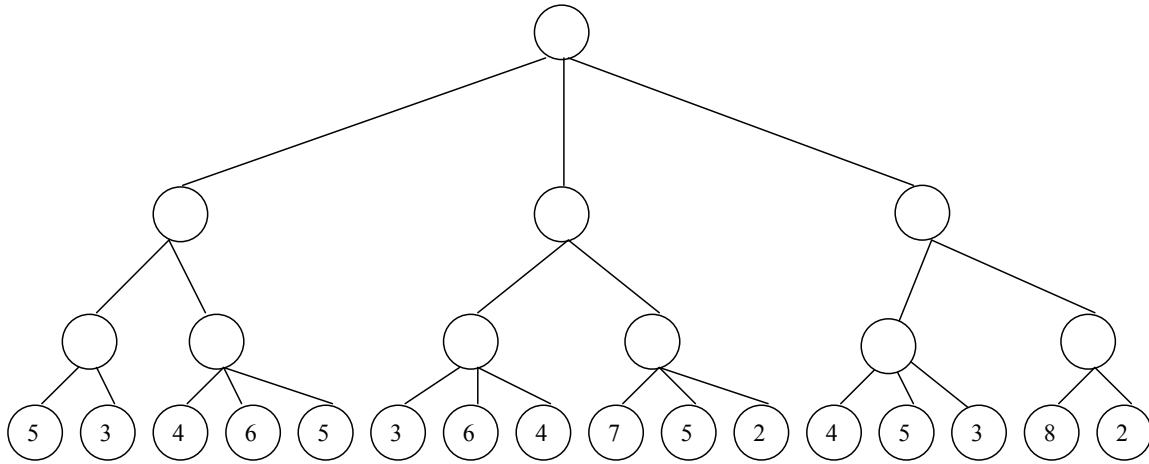
- Tipos de dominios a los que se puede aplicar:

Método AO*: Cualquiera en el que un problema se pueda dividir sucesivamente en subproblemas, de manera que la solución al problema se pueda hallar mediante la combinación de las soluciones de los subproblemas.

Búsqueda Primero el Mejor: aquellos problemas a los que se pueda asociar una búsqueda general en grafos.

Ejercicio 1. Segunda Semana. (Valoración: 4 puntos)

Considere el siguiente árbol:

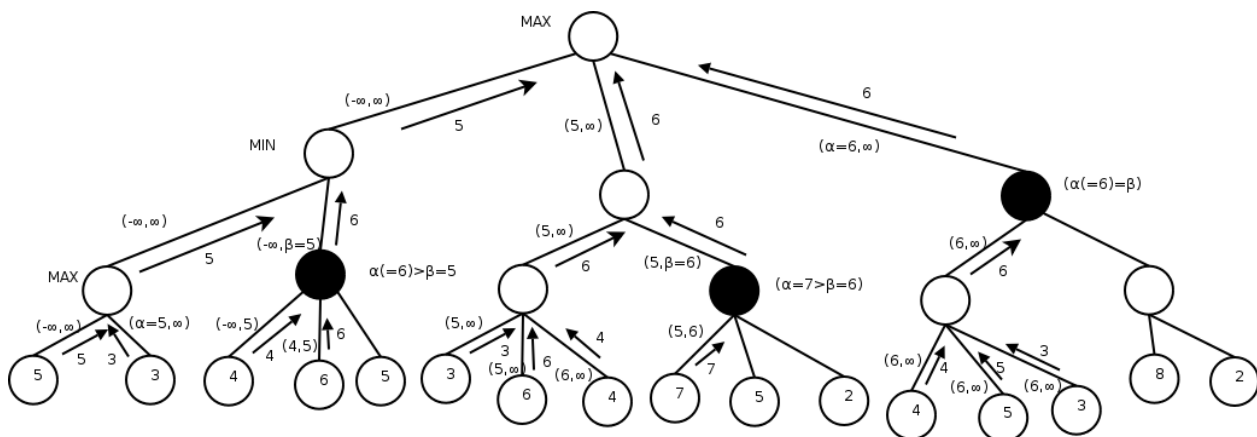


Donde los valores numéricos que aparecen en los nodos hoja corresponden a estimaciones de lo prometedoras que son para el jugador MAX las situaciones de la partida representadas por dichos nodos.

- Aplicar el método de poda alfa-beta al árbol anterior para el caso en el que el nodo raíz es un nodo MAX y el recorrido se realiza de izquierda a derecha.
- ¿En qué se asemejan y en qué se diferencian el método MINIMAX y el método de poda alfa-beta? Describa en detalle un ejemplo en el que dichas diferencias queden reflejadas.

Esquema de solución

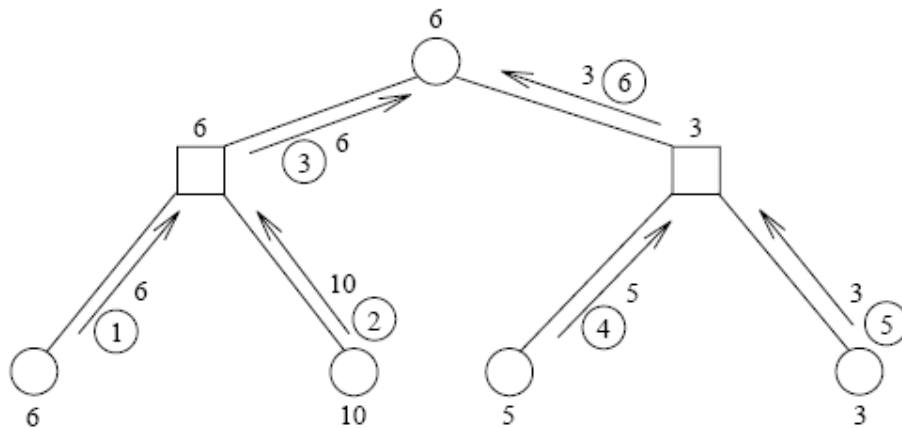
- 1) Aplicando el Método poda alfa-beta al árbol del enunciado nos quedaría un árbol final:



2) ¿En qué se asemejan y en qué se diferencian el método MINIMAX y el método de poda alfa-beta?. Describa en detalle un ejemplo en el que dichas diferencias queden reflejadas

Los métodos MINIMAX y de poda alfa-beta se utilizan en problemas de búsqueda con dos adversarios cuyo objetivo es ganar una partida en la que realizan movimientos alternativos. Normalmente, en cualquiera de estos dos métodos se genera un árbol de búsqueda con una profundidad limitada a partir de la situación inicial de la partida. Cada nivel de este árbol está asociado a un turno de movimiento para uno de los dos jugadores; lógicamente, se supone que cada jugador elegirá siempre aquel movimiento que más ventaja le dé en la partida. A los nodos hoja del árbol de búsqueda se les asocia el valor de una función de evaluación heurística (*fev*) que es una estimación del tipo de juego que cuelga de ese nodo hoja.

El método MINIMAX explora exhaustivamente el árbol de búsqueda mediante un método de búsqueda en profundidad. Considérese el siguiente ejemplo, donde se representa mediante círculos el turno del jugador que tiene que mover en la situación actual de la partida y mediante rectángulos el turno de su contrincante. Debajo de cada nodo hoja aparece un número que indica lo prometedora que es esa situación de la partida para que gane el jugador que tiene que mover en la situación actual (nodo raíz del árbol de búsqueda). Al lado de cada arco se especifica el valor de la jugada alcanzado si la misma se desarrolla por ese camino y, encerrado en un círculo, el ciclo del algoritmo de búsqueda en profundidad en que se devolvería dicho valor.

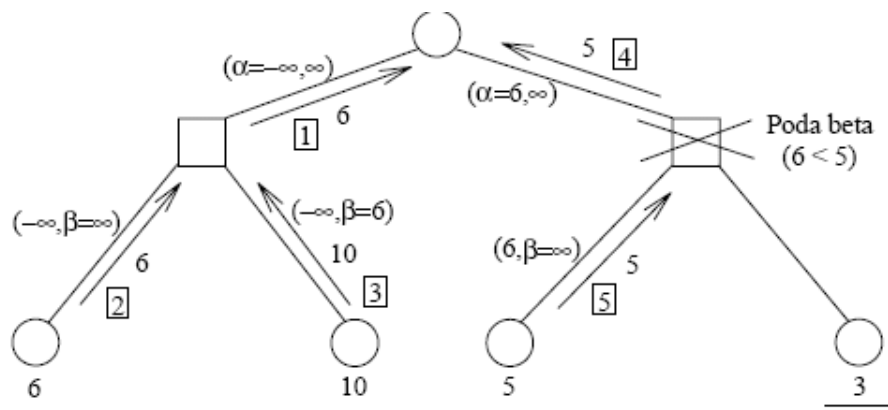


Finalmente, la mejor jugada es la de la izquierda, que conduce a un valor 6, frente a la jugada de la derecha, que conduce a un valor 3.

En el árbol de búsqueda anterior, en el ciclo 3 el jugador que tiene que mover sabe que por la izquierda puede llegar a una jugada de valor 6. En el ciclo 4, sin haber explorado todavía el nodo hoja etiquetado con 3, sabe que por la derecha podrá llegar a una jugada de valor 5 o menor que 5. Por tanto, siempre elegirá la jugada de la izquierda, sea cual sea el valor de la *f_{ev}* asociado al nodo hoja situado más a la derecha. Por tanto, la exploración que el método MINIMAX realiza del nodo hoja etiquetado con 3 es innecesaria.

El método de poda alfa-beta permite corregir situaciones como la anterior en las que el método MINIMAX realiza exploraciones innecesarias de ciertos subárboles del árbol de búsqueda. Para ello, a la vez que se realiza una exploración en profundidad del árbol de búsqueda, existe la posibilidad de realizar podas en el mismo. Una poda alfa se realiza en un nodo desde el que se intente acceder a situaciones de la partida con valores altos de la *f_{ev}* y una poda beta desde el tipo contrario de nodos. En cada llamada recursiva desde un nodo a uno de sus nodos hijo, se pasan dos parámetros: alfa y beta. El valor recibido como consecuencia de la llamada recursiva permite actualizar (incrementar o dejar igual) el valor de alfa, si la llamada se hizo desde un nodo desde el que se intente acceder a situaciones de la partida con valores altos de la *f_{ev}*; el valor actualizado es beta (se decrementa o se deja igual) si la llamada se hizo desde el tipo contrario de nodos. El valor que se devuelve al nodo padre después de realizada una llamada recursiva a cada nodo hijo es el último valor actualizado de alfa o beta, según corresponda. Al principio, en la primera llamada recursiva desde el nodo raíz, $\alpha = -\infty$ y $\beta = +\infty$. Se realiza una poda desde un nodo tan pronto como los valores que gestiona de alfa y beta cumplan la siguiente relación: $\alpha \geq \beta$.

El ejemplo anterior quedaría ahora de la siguiente forma, donde el orden de realización de cada llamada recursiva se representa mediante un número encerrado en un rectángulo:



En la llamada recursiva 2 se actualiza el valor de beta de ∞ a 6. Después de la llamada recursiva 3 se devuelve hacia arriba el último valor de beta actualizado: 6. En la llamada recursiva 1 se actualiza el valor de alfa de $-\infty$ a 6. En la llamada recursiva 5 se actualiza el valor de beta de ∞ a 5. En este momento se realiza una poda beta y el nodo etiquetado con “3” queda sin visitar. Como la última vez que se actualiza el valor de alfa en el nodo raíz es por el camino de la izquierda, ésta será la mejor jugada posible para esa situación actual de la partida.

Ejercicio 2. Segunda Semana. (Valoración: 3 puntos)

Dada una lógica modal que formalice los conceptos de necesidad y posibilidad ¿Cómo podría ser utilizada para abordar el tratamiento del razonamiento temporal? Dar una prueba para el siguiente razonamiento:

“Necesariamente, si Silvia tiene problemas entonces está preocupada”

“Es posible que Silvia no esté preocupada”

Por tanto, “es posible que Silvia no tenga problemas”

Esquema de solución

1) Dada una lógica modal que formalice los conceptos de necesidad y posibilidad ¿Cómo podría ser utilizada para abordar el tratamiento del razonamiento temporal?.

Problema 3.14 del libro de problemas

2) Dar una prueba para el siguiente razonamiento:

“Necesariamente si Silvia tiene problemas está preocupada”

“Es posible que Silvia no esté preocupada”

Por tanto “Es posible que Silvia no tenga problemas”

Sea:

A \equiv Silvia tiene problemas

B \equiv Silvia está preocupada

Entonces, habría que probar:

$\Box (A \rightarrow B)$

$\Diamond (\neg B)$

$\therefore \Diamond \neg A$

Considérese la siguiente prueba por reducción al absurdo:

1. $\Box (A \rightarrow B)$
2. $\Diamond (\neg B)$
3. $\neg (\Diamond \neg A)$ (Negación de la conclusión)
4. $\therefore \Box A$ (por 3)
5. $W \therefore \neg B$ (a partir de 2 donde W es cierto mundo)
6. $W \therefore (A \rightarrow B)$ (a partir de 1)
7. $W \therefore A$ (a partir de 4)
8. $W \therefore B$ (a partir de 6 y 7 y contradiciendo 5)

Ejercicio 3. Segunda Semana. (Valoración: 3 puntos)

Enumere las soluciones que conozca para aplicar control de razonamiento en reglas, también conocido como resolución de conflictos. Explique el funcionamiento de cada una de dichas soluciones a partir de un ejemplo sencillo.

Esquema de solución

a) Ordenación arbitraria de reglas en la base de conocimiento. Ejemplo:

R11: SI h1 ENTONCES h2

R2: SI h3 ENTONCES h4

BA: {h1, h3} (Base de afirmaciones)

En encadenamiento hacia delante, tanto R11 como R2 se podrían aplicar. Finalmente se aplicaría R11 por estar antes en la base de conocimiento.

b) Ordenación de cláusulas dentro de las reglas según probabilidad de fallo. Ejemplo:

R1: SI explosión-solar O guerra-nuclear ENTONCES entrar-en-refugio-subterráneo

En encadenamiento hacia atrás, si nuestro hecho objetivo fuera “entrar-en-refugio-subterráneo”, intentaríamos primero demostrar la primera cláusula, “explosión-solar”, cuyo fallo parece más probable. Esto nos evita desperdiciar recursos, ya que anticipamos el posible fallo del antecedente de la regla.

c) Adición de nuevas cláusulas en todas las reglas, según etapas. Ejemplo: en un sistema experto agrícola, podría interesar utilizar una cláusula referida a la estación del año en la que nos encontramos, para definir en qué estación es aplicable cada regla.

R1: SI primavera Y abril-lluvioso ENTONCES tarea10

R2: SI invierno Y disponibilidad-económica ENTONCES compra-de-maquinaria2

d) Asignación de prioridades a las reglas y utilización de agendas. Ejemplo:

R11: SI h1 ENTONCES h2 (prioridad 50)

R2: SI h3 ENTONCES h4 (prioridad 100)

BA: {h1, h3}

En encadenamiento hacia adelante podríamos introducir R11 y R2 en una agenda, la cual establecería que R2 es la regla a ejecutar, debido a que su prioridad es mayor.

e) Utilización de metarreglas. Ejemplo:

R1: SI h2 ENTONCES prioridad(R5, 20)

R1 es una metarregla porque permite razonar sobre conceptos asociados a reglas, “prioridad” en este caso. Lo que hace R1 es fijar la prioridad de R5 a 20, siempre que h2 sea cierto.

f) Aplicación de un mecanismo de refractariedad que impida, por ejemplo, que una regla se ejecute dos veces seguidas. Ejemplo:

R1: SI h1 ENTONCES h2

R2: SI h3 ENTONCES h4

BA: {h1, h3}

En encadenamiento hacia adelante, si suponemos que se ha ejecutado primero R1, a continuación debería ejecutarse R2 si se aplica este criterio.

g) Aplicación de un mecanismo de actualidad que obligue a que se ejecuten primero aquellas reglas cuyo antecedente se cumpla gracias a información obtenida en un ciclo más actual. Ejemplo:

R1: SI h1 ENTONCES h2

R2: SI h3 ENTONCES h4

BA: {h1(2), h3(3)}

En la base de afirmaciones, al lado de cada hecho se almacena el ciclo en que fue inferido. En encadenamiento hacia delante, habría que ejecutar R2 y no R1, ya que h3 es una información más actual que h1.

h) Aplicación de un criterio de especificidad que ejecute primero aquellas reglas más específicas.
Ejemplo:

R1: SI h1 ENTONCES h2

R2: SI h1 y h3 ENTONCES h4

BA: {h1, h3}

Según este criterio habrá que ejecutar antes R2, ya que es más específica que R1, es decir, el antecedente de R2 incluye al de R1.