

**DEPARTAMENTO DE ORGANIZACION DE EMPRESAS
UNIVERSIDAD DE CANTABRIA**

**CONOCIMIENTO Y BASES DE DATOS:
UNA PROPUESTA
DE INTEGRACION INTELIGENTE**

Margarita ALONSO MARTINEZ

**TESIS DOCTORAL DIRIGIDA POR:
Prof. Dr. D. José María SARABIA ALZAGA**

2.1 CONCEPTUALIZACION DE SISTEMA EXPERTO

Una primera concreción de sistema experto se debe a Edward Feigenbaum, de la Universidad de Stanford, quien en 1.977 lo definió como ⁶⁰: *"un programa inteligente para ordenador que usa conocimiento y procedimientos de inferencia para resolver problemas, que son lo suficientemente difíciles como para requerir la intervención de un experto humano para su resolución"*. En la misma línea, está la definición de un sistema experto, dada por d'Agapeyeff ⁶¹ en 1.983: *"programas para la resolución de problemas especialmente complicados y que requieren un elevado nivel de conocimientos. Se les califica como sistemas basados en conocimientos, porque sus prestaciones están determinadas por la utilización de procedimientos y técnicas heurísticas similares a las utilizadas por los expertos humanos"*. Las anteriores definiciones relacionan, los resultados de la aplicación de un sistema experto en la resolución de un problema

⁶⁰ FEIGENBAUM, E. A. (1.977): " The art of Artificial Intelligence: I. Themes and case studies of knowledge engineering". En BARR, A., COHEN, P. R. y FEIGENBAUM, E. (1.989) "The handbook of Artificial Intelligence" Volume IV Addison-Wesley Publishing Company Massachusetts.

⁶¹ D'AGAPEYEFF, A. (1.983): "Expert Systems, Fifth Generation and UK Suppliers". En SIMONS, G. L. (1.988): "Introducción a la Inteligencia Artificial". Ed. Diaz de Santos Madrid.

perteneciente al dominio sobre el que éste actúa, con el resultado que un experto humano obtendría al enfrentarse al mismo problema. Así, un sistema experto debe ser capaz, tanto de almacenar los conocimientos relativos al dominio del problema y necesarios para alcanzar su solución, y que el experto humano conoce, como de disponer de las mismas técnicas de búsqueda de la solución que el experto aplica en la resolución del problema. De esta forma los sistemas expertos incorporan gran cantidad de conocimientos sobre una determinada especialidad en su base de conocimientos, así como reglas prácticas, heurísticos, para orientar la búsqueda de la solución.

En este mismo sentido el comité de especialistas en sistemas expertos de la British Computer Society entiende los sistemas expertos como ⁶² *"el establecimiento en un sistema de computación de una base de conocimientos especializada y versada, de modo que la máquina puede ofrecer consejo inteligente o tomar una decisión inteligente sobre una función de proceso. Una característica adicional muy deseable, que puede considerarse como fundamental, es la capacidad del sistema, si así se solicita, de justificar su línea de razonamiento, de forma directamente inteligible por el usuario"*. El programa incorpora los procedimientos necesarios para explicar al usuario su línea de razonamiento. Un razonamiento que ha producido con la aplicación de heurísticos a los conocimientos almacenados, permitiendo de esta forma alcanzar determinadas conclusiones. Una enumeración de las operaciones que un sistema experto debería realizar está en la siguiente definición ⁶³ *"sistema informático que simula el proceso de aprendizaje, de memorización, de razonamiento, de comunicación y de acción de un experto humano en una determinada rama de la ciencia, suministrando de esta forma, un consultor que puede sustituirle con unas ciertas garantías de éxito"*.

Como consecuencia de las definiciones anteriores podemos caracterizar un sistema experto como:

⁶² SIMONS, G. L. (1.988): "Introducción a la Inteligencia Artificial". Ed. Diaz de Santos Madrid.

⁶³ CASTILLO, E. y ALVAREZ, E. (1.989): "Sistemas Expertos. Aprendizaje e Incertidumbre" Paraninfo S.A. Madrid.

- a) Aquel programa de ordenador capaz de llevar a cabo de forma eficiente procesos de búsqueda y recuperación de información interrelacionada y almacenada masivamente para su utilización por el programa.
- b) Programa que utiliza representación simbólica para establecer los conocimientos del dominio sobre el que actúa, admitiendo y representando información incompleta o contradictoria. Este tipo de información se puede producir en los problemas mal definidos, ⁶⁴ haciendo necesaria la utilización de inferencias y heurísticos. Estos permiten utilizar la experiencia acumulada, de forma práctica, para simplificar y orientar la resolución del problema planteado.
- c) Se configuran como programas interactivos y capaces de dialogar con el usuario, de tal manera que éste puede parar el proceso del programa y plantear preguntas sobre el razonamiento seguido hasta ese punto. El programa deberá incorporar los procedimientos necesarios para explicar su línea de razonamiento que, una vez conocida por el usuario, le permitirá confiar en la conclusión alcanzada.
- d) Pueden contemplarse como una eficaz ayuda para los usuarios que requieren una buena interpretación de información compleja, o bien un consejo sobre el dominio en el que el sistema experto trabaja, exigiendo a éste un rendimiento comparable al de los especialistas humanos correspondientes.

Sin embargo, el desarrollo de los sistemas expertos actualmente tiene un freno en varias deficiencias. Entre ellas, la capacidad de aprendizaje que relacionada con la posibilidad de generalizar procesos de resolución de problemas no está resuelta, como

⁶⁴ Un problema mal definido puede reunir alguna de las siguientes características:

- El objetivo a alcanzar no está definido.
- Los estados del problema, entendidos como fases en la búsqueda de la solución, no están delimitados.
- Las acciones a tomar orientadas a las búsqueda de las solución del problema no están especificadas.
- El espacio del problema es ilimitado.

tampoco lo está la capacidad de razonar por analogía. Así, los sistemas expertos actuales se limitan a la utilización de hechos y heurísticos almacenados, deteriorándose rápidamente su funcionamiento cuando el problema planteado desborda las tareas restringidas para las que fue diseñado.

Las características enunciadas de los sistemas expertos, hacen que su aplicación pueda extenderse en áreas muy amplias de actividad. En lo que respecta a su aplicación en el ámbito empresarial, los sistemas expertos se constituyen en una herramienta válida para optimizar la utilización de la información disponible y necesaria en la gestión de la empresa. Estos ofrecen, a la persona que tiene que tomar decisiones, un acercamiento sistemático a los problemas de gestión planteados, con un análisis rápido y en profundidad del volumen de información necesario para su resolución, proponiendo soluciones adecuadas y equivalentes a aquellas que un experto humano en gestión propondría. Por otra parte, los sistemas expertos incorporan ciertas características muy valoradas por los decisores -usuarios-, ya que la posibilidad de memorizar y explicar, cuando así se solicite, las etapas seguidas en la resolución de un problema, los hace transparentes al usuario, permitiendo su mejor comprensión y como consecuencia su mejor utilización. Esta característica de explicación del razonamiento seguido, además de proporcionar credibilidad al sistema, tiene un aspecto pedagógico para el usuario permitiéndole, en su caso, el aprendizaje de las técnicas de gestión incorporadas por el experto al programa. Otra característica importante, que ha de tenerse en cuenta en la construcción de un sistema experto para garantizar su empleo, es la facilidad de uso, simplificando en lo posible la comunicación usuario/programa. Las interfaces en lenguaje natural se orientan en este sentido.

Los sistemas expertos de gestión, se configuran de esta forma, como una valiosa herramienta de ayuda a la decisión. Pero, además, pueden colaborar a la solución de otros problemas que se plantean actualmente en la empresa ⁶⁵, entre ellos: la necesidad de disponer de conocimiento experto, susceptible de consulta, cuando un problema

⁶⁵ HARMON, P y KING, D. (1.988): "Sistemas Expertos. Aplicaciones de la inteligencia artificial en la actividad empresarial" .Ed. Díaz de Santos S.A. Madrid.

inesperado se plantea en cualquiera de los ámbitos en los que se desarrolla la gestión empresarial; la necesidad de personal especializado, con conocimientos actualizados que podrían ser adquiridos y puestos al día mediante la utilización de sistemas expertos con funciones educativas; la necesidad de mayores niveles de eficacia en la estructura organizativa de la empresa dotándola, en su caso, de la posibilidad de decisión distribuida y colaborando los sistemas expertos a que cada centro de decisión disponga de los conocimientos y controles necesarios para llevar a cabo su cometido.

2.2 COMPONENTES DE UN SISTEMA EXPERTO

El conocimiento contenido en un sistema experto se organiza basándose en dos elementos fundamentales ⁶⁶:

- 1.- Base de Conocimiento, que incorpora una representación de hechos y heurísticos referidos al dominio de aplicación del sistema.
- 2.- Motor de Inferencia, que controla y aplica el conocimiento contenido en la Base a la interpretación de los hechos que definen el problema a resolver.

La Base de Conocimiento, es relevante para determinar la potencia del sistema experto, puesto que éste depende ⁶⁷ de la amplitud de los conocimientos contenidos en aquélla. De hecho, se puede establecer una relación entre tamaño y calidad de la Base de Conocimiento y competencia del sistema experto. El conocimiento contenido en la Base se puede caracterizar, tomando como ejemplo la aplicación que un experto humano hace de su conocimiento para resolver de manera eficaz un problema de su ámbito de competencia. En la búsqueda de la solución éste utiliza, por una parte, conocimiento teórico sobre el modelo a aplicar y, por otra parte, conocimiento práctico obtenido en el desarrollo de su actividad profesional, con el objetivo de ajustar el modelo teórico de resolución a la realidad del problema planteado. Así, los sistemas expertos almacenan en su Base de Conocimiento, aquél relativo a objetos y sucesos característicos del dominio

⁶⁶ HARMON, P y SAWYER, B. (1.990): "Creating Expert Systems for Business and Industry". John Willey and Sons (eds.) New York.

⁶⁷ SIMONS, G. L. (1.988): "Introducción a la Inteligencia Artificial". Ed. Diaz de Santos, Madrid

sobre el que versa el sistema experto, considerándose éste conocimiento teórico. Pero, almacenan también, formando parte de su Base de Conocimiento, las relaciones que se pueden establecer entre aquél y cuyo conocimiento puede ser fruto de la experiencia. Se establece así, un conocimiento práctico que el sistema experto debe incorporar.

La elección de la forma de representación del conocimiento y su diseño son fundamentales de cara a obtener un buen rendimiento del sistema. La representación debe ser tal que permita identificar nuevo conocimiento, con la flexibilidad suficiente para incorporarlo a la Base de Conocimiento, si se juzga necesario, así como eliminar de ésta aquel conocimiento que se demuestre contradictorio u obsoleto. Se plantea, pues, otra cuestión no menos importante: la incorporación de nuevo conocimiento a la Base y/o su sustitución por otro más actualizado o preciso -aprendizaje-. El aprendizaje es básico en la evolución y capacidad de adaptación de la Base de Conocimiento. La incorporación de nuevos hechos y heurísticos mantiene la vigencia del programa.

El motor de inferencia, selecciona el conocimiento a aplicar en cada una de las fases de resolución del problema, comprobando al mismo tiempo la viabilidad de su aplicación. Esta comprobación la realiza el sistema a partir de información proporcionada, bien directamente por el usuario, o bien accesible al sistema como consecuencia del propio proceso, o por estar almacenado en su Base de Conocimiento. Las distintas soluciones parciales obtenidas en cada una de las fases intermedias del proceso de búsqueda de la solución son valoradas por el propio sistema, otorgándolas un determinado grado de confianza, permitiendo así el mantenimiento de la consistencia del sistema.

Alcanzadas las conclusiones pertinentes, el usuario puede estar interesado en conocer el proceso de razonamiento que ha conducido a las mismas. Por ello, debe ser posible que, tras el análisis de los procesos seguidos por el motor de inferencia, se comunique al usuario, en forma ordenada e inteligible, los hechos determinantes que diferencian las distintas alternativas de decisión. Se hace así necesaria la inclusión de un

módulo explicativo para permitir al usuario comprobar los razonamientos subyacentes en las respuestas del sistema.

Por otra parte, la comunicación entre el usuario de un sistema experto y el propio sistema, se lleva a cabo mediante una interface de usuario, cuyo objetivo es facilitar la comunicación hombre/máquina. Esta comunicación se orienta hacia la utilización del lenguaje natural para: interrogar y dar órdenes al sistema, recibir respuestas del mismo y, en su caso, explicaciones a las preguntas planteadas y, para permitir a este, realizar peticiones de nuevos datos al usuario, cuando sea necesario en la búsqueda de la solución.

Del estudio de los componentes de un sistema experto, se deriva la clara y explícita separación entre el conocimiento sobre el dominio del problema a resolver por un lado y el control de ese conocimiento por otro, en orden a alcanzar la mejor solución. Esta estructura implica, obviamente, un abandono de las técnicas tradicionales de programación. En efecto, en un programa convencional de ordenador, los conocimientos asociados al problema y los mecanismos de utilización de dichos conocimientos están interrelacionados, lo que implica su modificación conjunta. En cambio, la separación de los distintos componentes en un sistema experto, implica independencia, tanto para el desarrollo y modificación de alguna de las partes, sin afectar al resto, como para su utilización en variedad de funciones, basándose en la flexibilidad del sistema.

2.3 LA REPRESENTACION DEL CONOCIMIENTO

La capacidad de los sistemas expertos para realizar variedad de tareas de modo eficiente, tiene su base en el conocimiento especializado del que dispone el sistema. Ese conocimiento incluye: descripciones, relaciones y procedimientos pertenecientes al dominio del problema, y constituye la Base de Conocimiento del sistema. Las descripciones identifican objetos y clases o categorías de objetos. Conocer hechos acerca de los objetos es conocimiento que el programa incorpora en la Base de Conocimiento. Las descripciones se constituyen en sentencias en un determinado lenguaje y expresan

características y conceptos elementales acerca de los objetos. Las relaciones expresan y caracterizan las dependencias y asociaciones entre los elementos de la Base de Conocimiento. Los procedimientos, por su parte, especifican las operaciones necesarias para razonar sobre un problema y alcanzar su solución. El conocimiento práctico queda incorporado, también, a la Base de Conocimiento en forma de: asociaciones empíricas, métodos heurísticos, conceptos y restricciones aportados por la experiencia de un experto en el dominio. Así, con la integración de conocimiento práctico y teórico la Base de Conocimiento queda configurada por: descripciones simbólicas de objetos, relaciones - teóricas y prácticas- entre ellos y los procedimientos para manipular esas descripciones. El tener en cuenta la experiencia, en la definición de la Base de Conocimiento, permite disponer de conocimiento completo y efectivo sobre el dominio.

2.3.1 APROXIMACION TEORICA

La definición de la Base de Conocimiento requiere una decisión previa, sobre la *representación* del conocimiento más adecuada. Esta *representación* es una combinación de estructuras de datos y de los procedimientos para su interpretación. Hay variedad de *representaciones* posibles. En la elección se tendrán en cuenta las características del conocimiento del dominio donde se sitúa el problema a resolver.

Para decidir sobre la *representación* de conocimiento adecuada, hay tres aspectos a considerar: a) la adquisición de nuevo conocimiento, b) la búsqueda y localización en la Base del conocimiento relevante para el problema planteado, y c) el razonamiento que permite, a partir del conocimiento anterior, resolver el problema.

- a) **Adquisición de conocimiento:** El proceso de incorporar nuevo conocimiento a la Base -aprendizaje-, no debe contemplarse únicamente como una acumulación de nuevos hechos o de nuevas estructuras de datos. En orden a mejorar el rendimiento de un sistema experto, en la adquisición de nuevo conocimiento es necesario considerar que las nuevas estructuras de datos tienen que interactuar con las ya almacenadas, siendo necesaria la

compatibilidad de éstas, para evitar interferencias y bajas en el rendimiento. Es deseable, también, que el conocimiento adquirido se constituya a su vez en fuente de nuevo conocimiento mediante un adecuado proceso inferencial, de forma similar a como sucede con el aprendizaje de las personas.

- b) **Recuperación del conocimiento:** La recuperación se facilita con el enlace y unión de las distintas estructuras de datos vinculados. Las estructuras de uso conjunto se agrupan en una estructura mayor.
- c) **Razonamiento:** El sistema razona si es capaz de hacer algo que no le ha sido previamente especificado. En el proceso de razonamiento, el sistema debe deducir y verificar nuevos hechos a partir de aquellos explícitamente *representados* en la Base de Conocimiento.

La adquisición de nuevo conocimiento, su localización y el cómo usarlo posteriormente en el razonamiento, son aspectos que están interrelacionados y que deben tenerse en cuenta en el diseño de la Base de Conocimiento para conferir a ésta la máxima eficacia. De este modo, siguiendo a Rich ⁶⁸, un buen formalismo de representación de conocimiento, en un determinado dominio, debe poseer las siguientes propiedades:

- a) **Adecuación representacional:** habilidad para representar todo el conocimiento necesario en el dominio.
- b) **Adecuación inferencial:** habilidad de manipular las estructuras de representación, de tal manera que devengan o generen nuevas estructuras que correspondan a nuevos conocimientos inferidos de los anteriores.
- c) **Eficiencia inferencial:** capacidad de incorporar a la estructura de representación información adicional -metaconocimiento- ⁶⁹ que puede

⁶⁸ RICH, E. (1.984): "**Artificial Intelligence**". McGraw-Hill New York.

⁶⁹ **Metaconocimiento:** incluye conocimiento sobre aquello que se sabe, se domina, se duda o se dispone de experiencia; en definitiva, conocimiento acerca de lo que conocemos.

emplearse para dirigir los procedimientos de inferencia con el fin de optimizar el proceso.

- d) Eficiencia en la adquisición: capacidad de adquirir nueva información fácilmente.

Hay, además, toda una problemática que afecta, por un lado, a la forma en que se expresan los hechos que se van a registrar en la Base de Conocimiento y, por otro lado, a la flexibilidad y modularidad del esquema de representación elegido.

Así, en cualquier representación formal la elección de los atributos primarios, para representar los hechos en un dominio, afecta al poder de expresividad del esquema de representación. El problema de la selección de atributos se plantea por la multitud de formas posibles en que un hecho puede ser codificado.

La modularidad, hace referencia a la facilidad para añadir, modificar o borrar determinadas estructuras de datos sin afectar al conocimiento restante del sistema. Es deseable, para evitar complejas interdependencias, que el esquema de representación elegido permita la modificación de la base de Conocimiento, es decir, añadir o suprimir hechos sin afectar al resto del sistema. Hay, sin embargo, en todos los sistemas, un cierto grado de interacción entre las estructuras de datos que forman la Base de Conocimiento. La modularidad de los diferentes tipos de esquemas de representación se debe, en parte, a su propia forma de representación del conocimiento.⁷⁰

Por último, se debe también considerar, qué parte del conocimiento es *conocimiento explícito*, es decir, conocimiento directamente disponible para el sistema sin requerir elaboración previa. La ventaja de este *conocimiento explícito*, es el empleo de un mismo hecho representado para múltiples propósitos, confiriendo así, una significativa flexibilidad al sistema.

Los diferentes formalismos de representación de conocimiento -esquemas- se corresponden con distintas formas de comprender las estructuras de datos que han de ser

⁷⁰ WINOGRAD, T.(1.975): "Frame representation and the declarative/procedural controversy". En BARR, A. COHEN, P. R. y FEIGENBAUM, E. (1.989): "The handbook of Artificial Intelligence" Volume IV Addison-Wesley Publishing Company Massachusetts.

representadas, proporcionando cada esquema ventajas en ciertos dominios específicos. Los esquemas de representación de conocimiento utilizados en programas de Inteligencia Artificial, se corresponde con alguno de los siguientes:

- *Esquemas Lógicos.
- *Redes Semánticas.
- *Sistemas de Producción.
- *Analógicos o Directos.
- *Frames y Scripts.

2.3.2 ESQUEMAS LOGICOS

Los esquemas lógicos, fueron una de las primeras formas de representar el conocimiento en los programas de Inteligencia Artificial. En estos esquemas, los hechos se representan mediante fórmulas en algún sistema lógico -Lógica de primer orden, Multievaluada, Temporal, Difusa,...-. Una fórmula lógica es una combinación de predicados, variables, constantes, conectores, cuantificadores y funciones. La Base de Conocimiento se configura como una colección de fórmulas lógicas que proporcionan una descripción parcial del mundo real.

El cálculo de predicados es un lenguaje formal en el que se puede expresar, mediante una amplia variedad de sentencias, el conocimiento sobre el dominio. Se configura como una extensión de la lógica proposicional. En ésta, las proposiciones son sentencias que pueden ser *verdaderas* o *falsas*. Las sentencias pueden combinarse mediante *conectivos* -Y, O, NO, IMPLICA, EQUIVALENTE- para formar sentencias compuestas. A través de combinaciones sintácticas de sentencias y *conectivos*, se pueden construir sentencias en lógica proposicional, similares a las expresiones matemáticas. La estructura deductiva, *reglas de inferencia*, permite deducir una nueva sentencia a partir de sentencias anteriores, siempre que estas sean *verdaderas*. La regla de inferencia -modus ponens- indica que, si dos sentencias X y X IMPLICA Y son *verdaderas*, entonces se puede inferir que la sentencia Y es *verdadera*, pudiendo reemplazar esta sentencia simple

a la sentencia compuesta X IMPLICA Y en la Base de Conocimiento, eliminando una ocurrencia del conectivo IMPLICA.

La exclusiva representación del conocimiento mediante sentencias que pueden ser *verdaderas* o *falsas* puede, en ocasiones, resultar insuficiente. El cálculo de predicados, basándose en la lógica proposicional, permite representar sentencias sobre objetos y sus relaciones. El símbolo de predicado, se utiliza para representar una relación en un dominio. El empleo de conectivos, permite la combinación de fórmulas atómicas ⁷¹ (well-formed formulas -wff-), para formar sentencias más complejas. Un ejemplo de fórmula atómica (wff) sería el siguiente: considerese $P(a,b)$ con la siguiente interpretación:

- D (el dominio) es el conjunto de enteros.
- a es el entero 30.
- b es el entero 1.
- P es la relación *mayor que o igual que*

Bajo esta interpretación, la wff asevera que, *30 es mayor o igual que 1*. Esta aseveración es obviamente verdadera, y la wff toma el valor *verdadero* bajo esta interpretación. Si ésta cambia, con asignaciones diferentes a las variables, la evaluación de la fórmula podría dar como resultado *falso*.

En el cálculo de predicados, la representación formal queda ampliada con el uso de variables y *cuantificadores*. Las variables permiten referenciar en un predicado objetos que se definirán posteriormente, tomando el valor *verdadero* o *falso* dependiendo de la asignación. En cuanto a los *cuantificadores*, dos son los empleados: el cuantificador, "para todo...", y el cuantificador, "existe un...". La combinación sintáctica de conectivos, predicados, constantes, variables y cuantificadores permite la formalización de complejas sentencias para representar el conocimiento.

Un ejemplo, de cómo se puede representar en lógica un problema, lo constituye el conocido "puzzle" la Torre de Hanoi. En el, se dispone de tres varillas -1,2, y 3- y de tres discos de diferentes tamaños -A,B, y C-. Inicialmente los discos están en la varilla 1, el

⁷¹ NILSSON, J.(1.971): "Problem-Solving Methods in Artificial Intelligence". Ed. MacGraw-Hill New York.

disco grande C, esta debajo, y el pequeño, el disco A, esta arriba. Se desea transferir todos los discos a la varilla 3 teniendo en cuenta que: los discos se mueven de uno en uno; sólo el disco que esta situado en la parte superior de la varilla puede moverse; no se puede colocar un disco sobre otro mas pequeño que el. La situación inicial y final se muestra en la Figura 1. El problema se resuelve, estableciendo una secuencia de problemas simples y dando solución a cada uno de ellos.

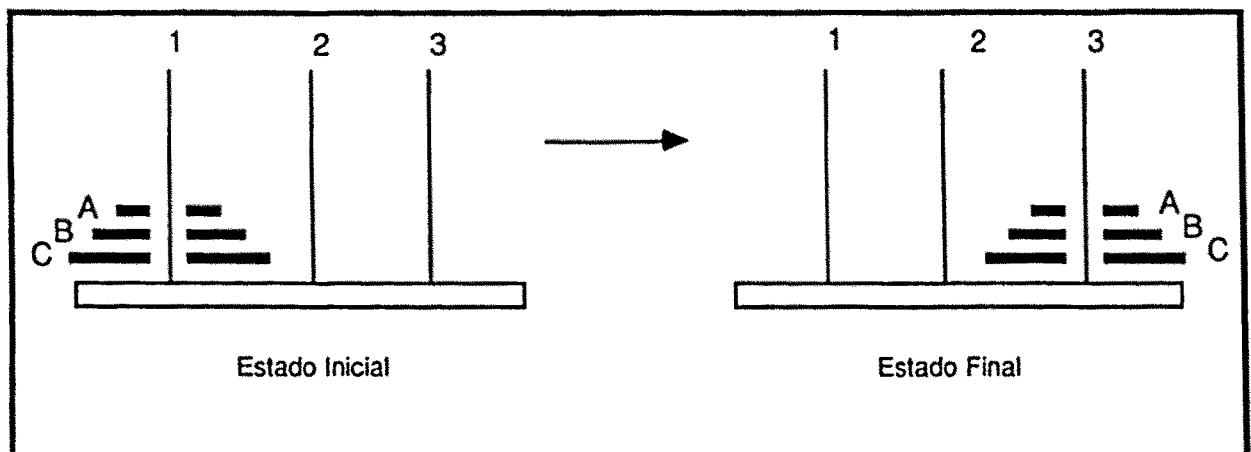


Fig. 1. "Puzzle" la Torre de Hanoi

Una representación parcial del conocimiento necesario para la solución del problema en lógica de primer orden ⁷², sería la siguiente:

Los objetos a representar son por un lado los discos A, B, y C y por otro las varillas, 1,2, y 3. Los predicados son DISCO y VARILLA. DISCO(A) es *verdadero* mientras que A sea un disco, VARILLA(C) es *falso* Para comparar el tamaño de los discos, se crea el predicado MENOR, que se define: MENOR(A,B) y sera *verdadero* si y solo si el disco A

⁷² Lógicas de primer orden: lógicas que permiten la cuantificación sobre objetos pero no sobre predicados y funciones.

es más pequeño que el disco B. La relación entre los objetos DISCO respecto al tamaño, podría quedar modelizada de la forma siguiente:

ABC (MENOR(A,B) Y MENOR(B,C)) -----> MENOR(A,C).

Con la interpretación: si disco A es menor que B, y B es menor que C, entonces A es menor que C.

La lógica de primer orden empleada en las sentencias del ejemplo, permite un razonamiento sencillo sobre los objetos representados, obteniendo resultados válidos - nuevas fórmulas-, que pueden a su vez utilizarse para derivar nuevos resultados que serán también válidos. La Base de Conocimiento es, así, consistente en todo momento, consistencia que no pueden asegurar otros esquemas de representación de conocimiento.

La lógica de primer orden, ofrece claridad entorno a los hechos representados: sobre el propio hecho a representar, sobre sus consecuencias y sobre las posibles inferencias que se pueden derivar de aquel. En este sentido, la lógica es idónea para explorar las posibilidades de representación del conocimiento en determinados problemas⁷³, por las siguientes razones:

- a) Es una forma natural de expresar ciertas nociones. Las descripciones contenidas en la Base de Conocimiento son muy simples.
- b) Es una forma precisa para determinar el significado de una expresión, mediante el uso de métodos lógicos deductivos.
- c) Es flexible, lo que significa que la modificación de la Base de Conocimiento es sencilla, se trata de introducir o eliminar fórmulas.
- d) Es modular, por lo que la representación del conocimiento es fácil de reformar y facilita la experimentación.

⁷³ La representación del conocimiento concierne a: ¿qué hechos del mundo están disponibles?, ¿cómo estos hechos pueden representarse en memoria? y ¿qué reglas permiten deducir conclusiones legítimas a partir de esos hechos?.

Sin embargo, la distinción entre representación y tratamiento del conocimiento es el mayor inconveniente de los esquemas basados en lógica de primer orden, al no poder solucionar satisfactoriamente la parte heurística ⁷⁴ debido a:

- a) Carencia de principios de organización entre los hechos que forman la Base de Conocimiento.
- b) Los objetos con los que trata son tan simples que no permiten representar las estructuras complejas del mundo real.
- c) Dificil manejo cuando el número de hechos en la Base de Conocimiento es muy grande, debido a la explosión combinatoria de las posibilidades de aplicación de reglas a hechos, en cada paso de la demostración. Sería necesario incorporar conocimiento sobre los hechos relevantes en situaciones particulares.

Los esquemas de representación de conocimiento de tipo procedural y los "frames" tratan de incorporar el aspecto heurístico en la propia representación.

2.3.3 REDES SEMANTICAS

Los primeros sistemas basados en redes semánticas, como formalismo de representación de conocimiento, fueron creados por R. Quillian en 1.968 ⁷⁵ y se diseñaron como modelos psicológicos de la memoria humana. Muchos otros sistemas han usado y usan este tipo de formalismo para representar el conocimiento en multitud de aplicaciones de Inteligencia Artificial. Todos ellos trabajan con una misma nomenclatura para representar el conocimiento, que consiste en: *nodos*, y *arcos* para conectar los *nodos*. Ambos conforman un grafo dirigido donde *nodos* y *arcos* pueden ser etiquetados. Estas etiquetas, en el caso de los *nodos*, se utilizan como referencia, e indican relaciones, en el caso de los *arcos*. A su vez, los *nodos* representan objetos -indivisibles o formados por

⁷⁴ El componente de tipo heurístico, se refiere al tratamiento dado al conocimiento una vez representado.
⁷⁵ QUILLIAM, R. (1.968): "Semantic Memory". En NILSSON, N. J. (1.987): "Principios de Inteligencia Artificial". Ed. Diaz de Santos S.A. Madrid.

otros objetos más simples-, conceptos -tipos de objetos-, o situaciones en un dominio. Los *arcos*, representan las relaciones entre ellos.

En una red semántica, es posible reconocer tres tipos básicos de *nodos*:

- a) Conceptos, para representar un objeto o una abstracción en un dominio.
- b) Hechos, para representar en el modelo las acciones que ocurren en la realidad.
- c) Situaciones, para representar estados y modificar conceptos, hechos y otras situaciones.

Los *arcos*, ponen de manifiesto la relación entre dos *nodos*. Las etiquetas de los *arcos*, explicitan la relación que puede ser de diversos tipos:

- a) Relación de un objeto con sus componentes -agregación-. Son del tipo "*es-parte*" o "*parte-de*". Pueden aplicarse recursivamente, de tal forma que pueden ser representados los componentes de un componente.
- b) Relación de un objeto con *una clase* más general a la que aquél pertenece -generalización-. Son del tipo "*es-un*" y producen una relación jerárquica. Al igual que el anterior pueden aplicarse recursivamente.
- c) Relación de un objeto con una situación o acción. En este caso, en cada *nodo* situación se puede establecer, a su vez, un conjunto de relaciones para especificar los distintos argumentos que identifican la situación.

La Base de Conocimiento se configura como una colección de estos grafos, produciéndose las modificaciones en la Base a través de la inserción o eliminación de *nodos* y la manipulación de las relaciones entre los *nodos*. Algunas características de las redes semánticas y las ventajas que confieren a la Base de Conocimiento, son las siguientes:

- a) La facilidad de acceso a la información contenida en la Base de Conocimiento, al poder utilizar las relaciones como patrones de direccionamiento.

- b) Las deducciones se producen siguiendo la jerarquía establecida en el propio grafo. Se asume así que los hechos de los nodos más altos en la jerarquía son aserciones sobre los nodos más bajos -herencia de propiedades-, no siendo necesario explicitar estas nuevas aserciones en la red. Un ejemplo de herencia jerárquica de propiedades puede ser el siguiente: de *Y es-un X* y de *Z parte-de X* se deduce que *Z parte-de Y*.
- c) La transitividad de las relaciones entre *nodos*. Por ejemplo, de *X es-un Y* e *Y es-un Z* se deduce *X es-un Z*.
- d) La posibilidad de representación gráfica de la Base de Conocimiento incrementa las facilidades para su comprensión.

Un ejemplo de red semántica como formalismo de representación de conocimiento, lo constituye el sistema FSA -Financial Statement Analyzer-,⁷⁶ un sistema experto para el análisis financiero desarrollado por Arthur Andersen, y uno de los primeros intentos para aplicar las técnicas de Inteligencia Artificial a la toma de decisiones financieras. La Base de Conocimiento del sistema contiene conocimiento contable y financiero. La Figura 2 muestra una pequeña parte de la red semántica que incorpora el sistema. En ella, cada *nodo* representa un objeto que se corresponde con algún dato contable. Los tipos de relación entre *nodos* son: *subcuenta-de* y *es-un*, que se corresponden con relaciones de agregación y generalización. El gráfico de la Figura 2 contiene varios niveles de cuentas y subcuentas, estas relaciones se representan por líneas de trazo grueso, las relaciones de generalización se representan en la figura mediante líneas de trazo fino. La herencia jerárquica se trasmite a través de la red, según el desglose establecido para las cuentas.

El conocimiento contable puede ser representado de forma eficaz por una red semántica, facilitando la disposición gráfica de las cuentas la adquisición del conocimiento del experto. Así, este formalismo de representación utilizado reúne las cualidades de: adecuación representacional del conocimiento necesario en el dominio;

⁷⁶ LIEBOWITZ, J. (ed.) (1.990): "Expert Systems for Business & Management" Prentice-Hall New Jersey.

adecuación inferencial, habilidad en la manipulación de estructuras de representación para derivar nuevas estructuras que correspondan a nuevos conocimientos inferidos de los anteriores; y eficiencia en la adquisición del conocimiento. Características citadas como cualidades que debe reunir un buen formalismo de representación del conocimiento 77.

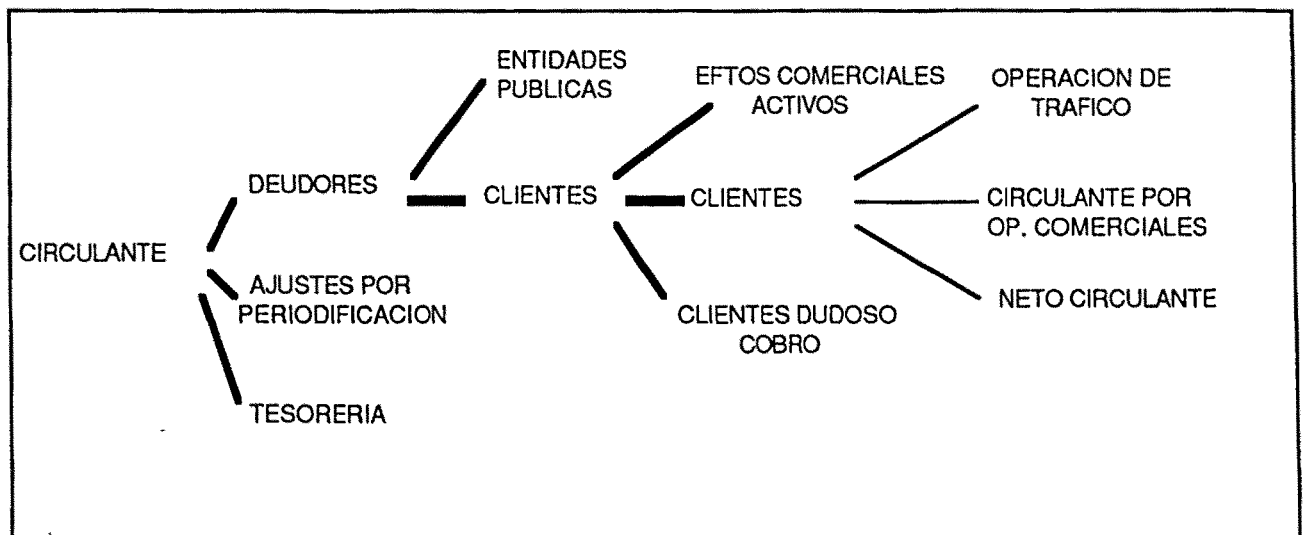


Fig. 2. Ejemplo de estructura jerárquica

En cuanto a la interpretación de las estructuras de una red semántica, depende del programa que las manipula y no existen convenciones acerca de su significado. Sobre una misma estructura se pueden usar diferentes procedimientos para llevar a cabo inferencias. El mecanismo de razonamiento usado por la mayoría de estos sistemas consiste en construir un fragmento de red por cada pregunta que se formule al sistema. Este fragmento de red se compara con la base de Conocimiento para ver si existe

77 RICH, E. (1.984): "Artificial Intelligence" McGraw-Hill New York.

respuesta. Así, los nodos variables que incorpora el fragmento construido obtienen, durante el proceso de comparación, los valores que permiten que el fragmento y la Base de Conocimiento encajen perfectamente. El comparador puede realizar inferencias durante el proceso de comparación para crear las estructuras de red que no están presentes explícitamente.

El formalismo de representación de conocimiento basado en *nodos y arcos*, que constituye la red semántica, ha encontrado aplicación en Inteligencia Artificial en numerosos sistemas sobre diferentes dominios. Desde los primeros trabajos de Quillian en psicología cognitiva, varios sistemas se han diseñado como modelos psicológicos de la memoria humana. Es el caso del programa HAM, diseñado por Anderson y Bower ⁷⁸ en 1.973, y el Active Structural Network de Norman y Rumelhart ⁷⁹ de 1.975. En comprensión del lenguaje natural, Raphael en 1.968 diseña el sistema SIR, uno de los primeros programas en utilizar las técnicas de representación semánticas y en 1.973 el sistema diseñado por Simmons da solución al problema de representación de situaciones y acciones en una red, con incorporación de los argumentos que la caracterizan. En 1.970, Carbonell usa en el diseño del programa SCHOLAR una red semántica para almacenar conocimiento geográfico que le permita al programa responder preguntas sobre geografía suramericana. Se han propuesto, también, algunos *lenguajes* de redes semánticas que poseen todo el poder expresivo del cálculo de predicados. Es el caso del formalismo de redes semánticas particionadas de Hendrix ⁸⁰, que da solución a la representación de los conectivos y cuantificadores lógicos, entre otros. El acceso a bases de datos en lenguaje natural, se diseña en el programa TORUS por Myopolous y el mantenimiento de bases de datos es estudiado por Doyle, junto con la representación de información por omisión⁸¹.

⁷⁸ ANDERSON, J. y BOWER, D. (1.973): "Human Associative Memory". En NILSSON, J. (1.987): "Principios de Inteligencia Artificial" Ed. Diaz de Santos Madrid.

⁷⁹ NORMAN, D. y RUMELHART, D. (Eds) (1.975): "Explorations in Cognition". En BARR, A. y FEIGENBAUM, E. (1.981): "The handbook of Artificial Intelligence" Volume I William Kaufmann. Los Altos, California.

⁸⁰ HENDRIX, G. (1.975): "Expanding the utility of semantic networks through partitioning". En BARR, A. y FEIGENBAUM, E. (1.981): "The handbook of Artificial Intelligence" Volume I William Kaufmann. Los Altos, California.

⁸¹ DOYLE, J. (1.979): "A truth maintenance system". En CHORAFAS, D. N. (1.990): "Knowledge Engineering: Knowledge Acquisition, Knowledge Representation, The role of the Knowledge engineer, and Domains Fertile for AI Implementation" Mac Graw-Hill New York.

Sin embargo, y a pesar de la popularidad de las redes semánticas como esquemas de representación del conocimiento en aplicaciones de Inteligencia Artificial, plantean problemas que afectan a su manipulación, que resulta complicada cuando la Base de Conocimiento es muy grande y, también, a su estructura, al no disponer de una semántica formal ni de una terminología estándar para etiquetar los arcos.

2.3.4 SISTEMAS DE PRODUCCION

Desarrollados por Newell y Simon en 1.972 ⁸² para modelizar el conocimiento humano, son esquemas modulares de representación de conocimiento. La idea básica de estos formalismos es constituir la Base de Conocimiento como un conjunto de reglas, llamadas también producciones, que se presentan en forma de pares condición (IF)-acción (THEN). La parte de condición de la regla contiene explícitamente las condiciones que deben cumplirse para que la regla sea aplicable, siendo la parte de acción aquella que se ejecuta en el caso de cumplimiento de la parte condición. Un sistema de producción tiene tres componentes principales: a) un conjunto de reglas; b) una o más bases de datos; c) un intérprete de reglas que dirige y controla la actividad del sistema.

- a) Conjunto de reglas. Componente básico de la Base de Conocimiento, permite formalizar el conocimiento sobre el dominio. Las reglas pueden expresar conocimiento declarativo o bien conocimiento procedural. Se configuran, en su forma más general, como un par ordenado condición-acción admitiendo, también, para incorporar al sistema conocimiento incierto, el asociar a cada regla un grado de certeza o grado de confianza en la verosimilitud de la misma. Durante la ejecución de un sistema de producción, una regla cuya parte de condición se satisfaga puede activarse, es decir, el intérprete ejecutará su parte de acción.

⁸² NEWELL, A. y SIMON, H. A. (1.972): "Human problem solving". En BARR, A., COHEN, P. R. y FEIGENBAUM, E. (1.989): "The handbook of Artificial Intelligence" Volume IV Addison-Wesley Publishing Company Massachusetts.

REGLA 1

SI (1) banco_cheque es Experto y,
(2) el emisor es reconocido y,
(3) el saldo es suficiente y,
(4) el receptor está identificado y,
(5) el cheque está completo
ENTONCES el pago del cheque está permitido

REGLA 2

SI (1) la fecha del cheque es correcta y,
(2) la firma del cheque existe y,
(3) las cantidades concuerdan
ENTONCES el cheque está completo

REGLA 3

SI (1) tipo cheque es al portador
ENTONCES el receptor está identificado

REGLA 4

SI (1) la fecha del cheque es hasta 90 días anterior
ENTONCES la fecha del cheque es correcta

REGLA 5

SI (1) el saldo de la cuenta es mayor que la cantidad del cheque
ENTONCES el saldo es suficiente

REGLA 6

SI (1) tipo cheque es personal y,
(2) la firma del receptor comprobada
ENTONCES el receptor está identificado

Fig. 3 Ejemplo de conjunto de reglas

Un ejemplo, muy simple de aplicación de reglas, se muestra en la Figura 3, donde se trata de resumir la estrategia seguida por el "Banco Experto" para el pago de los cheques emitidos por sus clientes ⁸³.

- b) Base de datos o contexto. Contiene una colección de hechos, que definen el estado actual del problema a resolver. La base de datos es consultada para verificar las condiciones de las reglas. La condición o conjunto de condiciones que pertenecen a una regla deben estar presentes en la estructura de datos del contexto para que la producción se active. Las acciones que originan las reglas de producción activadas, pueden variar el contexto y permitir, de ésta forma, que se satisfagan las condiciones de otras reglas.
- c) Intérprete. Es un programa cuyo trabajo consiste en decidir qué hacer en el instante siguiente, o lo que es lo mismo, qué regla activar a continuación. El intérprete actúa seleccionando en primer lugar, las reglas en las que se cumple la parte de condición. Una comprobación que realiza contra contexto para, a continuación, ejecutar la parte de acción de la regla. La ejecución de acciones puede producir diferentes efectos, bien dirigidos a provocar cambios en el contexto o, en su caso, producir la comunicación exterior del sistema.

Un sistema de producción, al entrar en ejecución, opera en ciclos. En cada ciclo se pueden distinguir tres fases: correspondencia, resolución de conflictos y acción. En la primera fase de correspondencia, el intérprete examina las reglas de forma exhaustiva, para conocer las apropiadas y, por tanto, candidatas a activarse. Se trata de encontrar todas las producciones cuya parte de condición sea *verdadera* para hacerlas aplicables. En la resolución de conflictos, si se halla más de una regla apropiada, una única producción es escogida de entre ellas. Los criterios de selección de la regla se basan en

⁸³ CASTILLO, E. y ALVAREZ, E. (1.989): "SISTEMAS EXPERTOS. Aprendizaje e Incertidumbre" Ed. Paraninfo S.A. Madrid.

criterios del tipo *la regla añade nueva información al contexto o se ajusta a la secuencia establecida para el conjunto de reglas*. En la acción, se activa la regla seleccionada.

A partir de las características de los sistemas de producción se pueden generalizar un conjunto de ventajas e inconvenientes de estos esquemas. Entre las ventajas destacan:

- a) **Modularidad:** una cualidad obvia de los sistemas de producción es aquella, que permite añadir, borrar o cambiar independientemente cada producción de la base de reglas. Cada una de ellas se configura, así, como una pieza independiente de conocimiento. Las producciones que forman el conjunto de reglas pueden modificarse independientemente unas de otras, pues el cambio de una regla no afecta al resto, al comunicarse las reglas a través de la estructura de datos del contexto. Esta relativa modularidad de las reglas es importante a la hora de construir sistemas con grandes Bases de Conocimiento al facilitar su formalización.
- b) **Uniformidad:** esta característica se refiere a la estructura del conjunto de reglas. El conocimiento a incorporar al sistema debe ser codificado de acuerdo con la estructura rígida de las reglas, facilitando la comprensión del conocimiento así codificado. Esta comprensión es mayor que en otras formas de representación relativamente más libres, como es el caso de las redes semánticas.
- c) **Naturalidad:** los sistemas de producción permiten expresar fácilmente cierto tipo de conocimiento, en particular, aquél que a partir de una situación predeterminada conoce la acción a llevar a cabo. Este conocimiento se corresponde con la propia estructura del formalismo a utilizar. Por otra parte, los expertos, para expresar la mayoría de sus técnicas de resolución de problemas, tienden a utilizar un conjunto de reglas heurísticas que podrían expresarse como un conjunto de reglas de producción.

- d) **Explicación:** los sistemas de producción explican sus conclusiones traduciendo las reglas que contribuyen a la toma de una decisión. La comprensión del lenguaje natural se hace necesaria, tanto para facilitar el diálogo con el usuario, como para posibles explicaciones solicitadas al sistema por éste o para solicitar, en su caso, nuevos datos al usuario.

Algunos de los inconvenientes significativos inherentes a los sistemas de producción, se pueden concretar en los siguientes:

- a) **Ineficiencia:** durante la ejecución de un sistema de producción, la ineficiencia se debe a la fuerte modularidad y uniformidad de las reglas, lo que eleva el coste de su utilización en la resolución de problemas, frente a otros sistemas de representación. La característica de ejecución de los sistemas de producción mediante ciclos, es ineficiente, puesto que impide secuencias más largas de razonamiento cuando esto es necesario y posible.
- b) **Opacidad:** que indica, la dificultad de seguir el flujo de control para la resolución del problema. Si el conocimiento situación-acción se expresa de forma natural en los sistemas de producción, el conocimiento algorítmico no se expresa naturalmente. Dos características de las reglas de producción contribuyen a ello: primero, la independencia de las producciones -no se llaman entre sí- y, después, el tamaño uniforme de las producciones -una producción no puede descomponerse en varias subproducciones-, lo que impide estructurar para hacerlo más comprensible el flujo de control.
- c) **Inconsistencia:** a partir de un elevado número de reglas, es difícil para el usuario, verificar directamente si una nueva regla es redundante o, si el conocimiento que almacena es contradictorio -inconsistente- con el ya almacenado en alguna otra regla existente.

La evolución en el diseño de los sistemas de producción, viene a paliar de alguna forma los anteriores inconvenientes. Así, por lo que se refiere a la complejidad de las

condiciones y acciones de las reglas, su estructura ha ido ampliándose progresivamente a medida que el tamaño y complejidad de los sistemas ha ido en aumento. En algunos, las condiciones de la regla se constituye en una función para evaluar condiciones complejas, en otros sistemas, la evaluación de las condiciones tiene efectos colaterales, alterando el contexto o cambiando la secuencia de control, antes de la activación de la regla. De forma similar, la parte de acción de la regla, puede incluir variables y permitir la ejecución de programas, antes de producir cambios en el contexto. Estos programas usualmente especifican operaciones, que pueden incluir la activación o desactivación de otras producciones.

La estructura del conjunto de reglas y el contexto, también han evolucionado. De las tres fases de cada ciclo de un sistema de producción, la fase de correspondencia es la que usa la mayor parte de los recursos dedicados a la ejecución. A medida que los sistemas de producción han evolucionado hacia mayores y más complejos sistemas, la eficiencia en la ejecución ha exigido que tanto el conjunto de reglas como el contexto se conviertan en estructuras de datos más elaboradas. Así, por ejemplo, para determinar rápidamente qué producciones pueden aplicarse, en una situación dada, sin tener que recorrer todo el conjunto de reglas, las producciones se indexan o se agrupan de acuerdo con las condiciones. También la estructura de datos del contexto ha evolucionado hacia estructuras más complejas, para así permitir representar situaciones más complicadas, caso del programa PROSPECTOR, donde los datos del contexto se estructuran en una red semántica.

Por lo que respecta a la resolución de conflictos, sucede con frecuencia en los grandes sistemas de producción que más de una regla puede ser activada en cada ciclo de ejecución. La elección de una única regla a aplicar, de entre el conjunto posible, puede llevarse a cabo utilizando diversos criterios de selección. Se enumeran a continuación alguno de estos:

- a) La primera regla que se adapte al contexto según un orden lineal del conjunto de reglas;

- b) La regla de mayor prioridad, donde la prioridad se define según las demandas y características de la tarea;
- c) La regla más específica, la que tras el proceso de correspondencia posee una mayor similitud de su parte condición con el contexto actual;
- d) La regla que se refiere al elemento añadido más recientemente al contexto;
- e) Una regla aún no activada;
- f) Una regla arbitraria;
- g) No escoger, sino explorar todas las reglas aplicables en paralelo, es el caso por ejemplo del sistema MYCIN.

Diferentes sistemas utilizan distintas combinaciones de estos métodos simples. McDermott y Forgy ⁸⁴ sugieren que la estrategia utilizada para la resolución de conflictos afecta a dos importantes características de los sistemas de producción: a su habilidad para responder rápidamente a los cambios en el entorno y a su habilidad para llevar a cabo una secuencia de acciones relativamente larga.

Respecto al proceso de razonamiento -dirección de inferencia-, se reconocen por los investigadores dos formas distintas de razonar ⁸⁵: el encadenamiento hacia adelante -forward chaining- y el encadenamiento hacia atrás -backward chaining-. El encadenamiento hacia adelante es un proceso de inferencia que a partir de la información disponible trata de extraer conclusiones apropiadas para los objetivos del sistema. El inconveniente que presenta es la falta de focalización: antes de alcanzar las conclusiones deseadas puede que cientos de ciclos se ejecuten sin que aporten nada sustancial a la solución. El encadenamiento hacia atrás, es un proceso de inferencia que a partir del objetivo propuesto trabaja en busca de evidencias que lo soporten o contradigan. Este método de razonamiento, requiere estudiar la parte de acción de la regla para buscar aquellas que han permitido alcanzar el objetivo. Una vez encontradas, se localiza en la

⁸⁴ McDERMOTT, J. y FORGY, C. (1.978): "Production system conflict resolution strategies". En HAYES-ROTH, F., WATERMAN, D., LENAT, D. (1.983): "Building Expert Systems" Addison-Wesley. Massachusetts

⁸⁵ BARR, A. y FEIGENBAUM, E. (1.981): "The handbook of Artificial Intelligence" Volume I William Kaufmann. Los Altos, California.

parte de condición de la regla aquéllas cuyo cumplimiento ha permitido su activación. El proceso continúa localizando, de forma sucesiva, otras reglas cuya parte de acción se corresponda con las condiciones satisfechas. Un ejemplo de utilización de este método de razonamiento, lo constituye el programa MYCIN.

Como conclusión, puede decirse que los sistemas de producción son aquellos que capturan conocimiento -básicamente procedural-, para la resolución de problemas, formalizándolo en un esquema de representación "manejable", de fácil manipulación. Es por esto que el tipo de conocimiento más adecuado para su representación mediante reglas es aquél conocimiento acerca de qué hacer en una determinada situación. Davis y King ⁸⁶ caracterizan los dominios en los que, usualmente, se utilizan las reglas de producción como esquemas de representación de conocimiento como aquellos en los que: a) el conocimiento está constituido por una gran abundancia de hechos; b) los distintos procesos pueden representarse como un conjunto de acciones independientes; c) el conocimiento puede fácilmente separarse de su tratamiento.

Por lo que respecta a la estructura de las reglas ésta, es similar al esquema que las personas usan para resolver problemas, lo que facilita su comprensión. Que unido a su modularidad, que facilita su manipulación, hace de los sistemas de producción el componente básico de populares sistemas expertos como: DENDRAL, MYCIN o PROSPECTOR. Es de destacar que la investigación en la Base de Conocimiento de estos sistemas, se llamó *ingeniería del conocimiento*. Se integra en ella, además de la representación del conocimiento, su adquisición de los expertos humanos y la explicación, a los usuarios, del razonamiento seguido por el sistema.

Al incorporar los sistemas de producción, básicamente, conocimiento procedural, puede ser de interés en este punto destacar las características y las diferencias entre las representaciones procedurales y las declarativas, así como, un comentario acerca de las

⁸⁶ DAVIS, R. y KING, J. (1.977) "An overview of production systems". En BARR, A. COHEN, P. R. y FEIGENBAUM, E. (1.989): "The handbook of Artificial Intelligence" Volume IV Addison-Wesley Publishing Company Massachusetts.

controversias a que dan lugar los dos tipos de representación en los desarrollos de Inteligencia Artificial.

Las representaciones declarativas se caracterizan por hacer hincapié en los aspectos estáticos del conocimiento. Se representan hechos acerca de objetos, sucesos y las relaciones entre éstos y con los estados del mundo real. La separación entre el conocimiento representado y el conjunto de procedimientos que manipula ese conocimiento es explícita en los esquemas de representación declarativos.

En las representaciones procedurales, el "*conocimiento acerca de cómo usar el conocimiento*" es relevante para el sistema. Así, los hechos característicos y la realización de inferencias se incorporan como conocimiento en el conjunto de procedimientos ⁸⁷ que configuran el esquema de representación. En los primeros esquemas procedurales como SIR -Raphael 1.968- el conocimiento sobre los hechos es almacenado en estructuras similares a las usadas en las representaciones declarativas. Los mecanismos para la realización de inferencias son representados mediante procedimientos especiales. Estos guían la resolución del problema eliminando ineficiencias propias de los sistemas declarativos, entre otras, la utilización de conocimiento irrelevante o el seguir líneas no naturales de razonamiento. Sin embargo, esta solución ha creado su propia fuente de problemas ya que, en general, a medida que la complejidad de los sistemas procedurales aumenta, su comprensión y modificación se hace más difícil.

El interés está, por tanto, en hacer compatible la facilidad de modificación de los sistemas declarativos, especialmente los esquemas lógicos, y la linealidad en las demostraciones con la búsqueda de soluciones de los sistemas procedurales. En esencia, se trata de representar el conocimiento declarativo en la forma en que se codifica en las expresiones lógicas, junto con las instrucciones para su uso. Los esquemas de representación de conocimiento procedurales, en la actualidad, tratan de hallar mejores formas de expresar información de control, es decir, información acerca de cómo y cuándo usar el conocimiento que el sistema posee. Esta información acerca de cómo usar el

⁸⁷ Procedimientos: pequeños programas donde se indica, cómo hacer tareas específicas y cómo proceder en situaciones bien determinadas

conocimiento almacenado, puede dirigir de varias formas el proceso de razonamiento. Una forma de control lleva a indicar la dirección en la cual puede usarse una implicación "*hacia adelante*" o "*hacia atrás*". Podría plantearse un problema al intentar direcciones de búsqueda en sentido contrario, no válidas al no estar soportadas por el sistema. Otra forma de control puede ser especificar el conocimiento relevante para alcanzar un determinado objetivo.

De entre los métodos que permiten especificar información de control a un sistema, se destacan tres como más relevantes: a) la especificación de control se lleva a cabo por la forma en que se expresan los hechos: sistema PLANNER (Hewitt 1.972); b) codificación, en bajo nivel, del lenguaje de representación, facilitando el acceso del usuario al conjunto de mecanismos de especificación del proceso de razonamiento: sistema CONNIVER (McDermott y Sussman 1.972); c) la definición de un lenguaje adicional para expresar la información de control, que actúa junto con el lenguaje de representación: proyecto GOLUX (Hayes 1.973) y cálculo de predicados (Kowalski 1.974).

Por otra parte, el control de un sistema puede reforzarse incorporando conocimiento heurístico. Así, entre las ventajas que ofrece el uso de procedimientos para la representación de conocimiento, es de destacar su facilidad para representar conocimiento heurístico, especialmente información sobre un dominio específico, que puede permitir procesos de deducción más directos. El conocimiento heurístico incorporado al sistema incluye información sobre la dirección en la que ha de usarse una implicación, el conocimiento a aplicar en una situación dada y los objetivos primarios a alcanzar. Esta información es crucial en grandes sistemas que no podrían funcionar si el mecanismo de resolución de problemas no fuese eficiente.

El razonamiento por defecto, es otra de las ventajas de las representaciones procedurales ya que permite integrar un tipo de razonamiento informal en un sistema que incorpora cierta lógica formal. Para Winograd ⁸⁸, es necesario incorporar esta clase de inferencias en un sistema que intente modelizar el razonamiento humano. Por último, una

⁸⁸ WINOGRAD, T. y FLORES, F. (1.986): " **Understanding computers and cognition: a new foundation for design**" Ablex Publishing Corporation Norwood.

ventaja destacada de las representaciones procedurales es la capacidad de modificar la base de datos que contiene el contexto del problema, como consecuencia de acciones tomadas por el sistema.

En lo que respecta a los inconvenientes que exhiben las representaciones procedurales frente a otros esquemas de representación, cabe citar la posibilidad de configurar sistemas incompletos y/o sistemas inconsistentes. Así muchos sistemas procedurales no son completos porque, a pesar de poseer toda la información necesaria para lograr una cierta conclusión, no son lo suficientemente potentes para alcanzar las deducciones requeridas. Sin embargo, el disponer siempre de sistemas completos puede, en ocasiones, no ser deseable. Quizás, se prefiera que el sistema trabaje rápidamente frente a que gaste mucho tiempo en hallar una respuesta específica o concluir que no puede lograr respuesta alguna. Por otra parte, la mayoría de los sistemas procedurales no son consistentes; frente a los sistemas que se basan en lógica deductiva que sí lo son. Un sistema es consistente si todas sus deducciones son correctas, esto es, si sus conclusiones se deducen, necesariamente, de ciertas premisas dadas. La incorporación al sistema de razonamiento por defecto puede introducir inconsistencias -contradicciones- en el mismo, si va unido a conocimiento incompleto. La consistencia de un sistema puede no ser necesaria, pero si es siempre deseable.

Otros inconvenientes de las representaciones procedurales se refieren a:

- a) En ocasiones, la información de control no es necesaria y, sin embargo, no se puede eliminar. Por ejemplo, no es posible enunciar un objetivo sin incluir información de control. Sin embargo, en una representación declarativa un hecho sólo se representa una vez, y son los procedimientos generales los que al accederlo lo utilizan de diferente forma. De este modo, no es necesario prever por anticipado cómo se usará posteriormente un hecho.
- b) La modularidad del conocimiento se sacrifica. En las representaciones procedurales la interacción entre los hechos es propia del conocimiento

heurístico. Se incrementa, así, la dificultad para verificar y modificar estos esquemas. Por el contrario, en los esquemas declarativos, los hechos representados son independientes entre sí, lo que facilita la comprensión y modificación de la Base de Conocimientos del sistema.

La controversia entre representaciones declarativas y procedurales ha contribuido a establecer la importancia de la representación del conocimiento en los trabajos de Inteligencia Artificial. Esta controversia, aún no está cerrada, se ha visto superada por una tendencia integradora, dado que para representar el conocimiento, en muchos de los dominios, se necesita de ambos tipos de representación. En la práctica, la mayoría de las representaciones utilizan bien una combinación de ambas tendencias o bien esquemas de representación, como "frames" o "scripts", que llevan incorporados ambos tipos de información en sí mismos.

2.3.5 ESQUEMAS DIRECTOS

Las representaciones directas se definen como esquemas en los que *las propiedades y relaciones entre las partes de una representación determinada, describen las propiedades y relaciones entre las partes en una compleja configuración, de modo que, la estructura de la representación proporciona información acerca de la estructura de lo que es representado*⁸⁹. Se requiere, por tanto, correspondencia entre las relaciones en la estructura de datos que constituye la representación y las relaciones en la situación real representada. Por ejemplo, un plano de una ciudad es una representación directa, en el sentido que las distancias entre dos puntos en el plano se corresponden con la distancia entre dos lugares en la ciudad representada. Existe, pues, una correspondencia entre las relaciones en la representación y las relaciones en la situación a representar.

Las distintas representaciones pueden utilizar estructuras de datos similares, pero difieren en como usarlas. Por ejemplo, una representación lógica no indicará nada acerca

⁸⁹ SLOMAN, A. (1.971): " Interactions Between Philosophy and AI: the role of intuition and non-logical reasoning in intelligence". En BARR, A. y FEIGENBAUM, E. (1.982): "The handbook of Artificial Intelligence" Volume II William Kaufmann. Los Altos, California

de la ubicación de los objetos con los que trabaja. Sin embargo, si un programa para examinar un plano utiliza una tabla interna para la localización de las distancias sin recurrir al propio plano para su medición, entonces éste ya no constituye un esquema de representación directa respecto a la distancia.

Una representación directa es la combinación de ciertas estructuras de datos y de la función de interpretación semántica que las manipula. Es a esta función a la que se denomina esquema de representación directa, siendo directa respecto a ciertas propiedades de lo representado. Volviendo al ejemplo anterior, si un plano es una representación directa respecto a la distancia, no lo es normalmente, respecto a la altitud; dato que no suele ser representado en planos de ciudades.

Los esquemas directos constituyen una significativa ventaja para la representación de ciertos tipos de problemas. En particular aquellos en los que la actualización de la representación, para reflejar cambios en el dominio real, es muy simple. En el ejemplo, si se añade una nueva ciudad al plano, sólo es necesario situarla en el lugar correcto, sin necesidad de indicar explícitamente la distancia de ésta con respecto a las ciudades ya recogidas en el plano.

Las ventajas que proporcionan las representaciones directas, respecto a otros esquemas de representación se pueden resumir en los puntos siguientes:

- a) Las representaciones directas a menudo facilitan la observación de importantes propiedades relativas a los objetos de un dominio, favoreciendo, en muchos casos, una más sencilla deducción.
- b) Pueden facilitar la búsqueda de soluciones, ya que las restricciones que se dan en la realidad son representadas mediante restricciones en los tipos de transformaciones que se aplican a la representación. De este modo, las estrategias no válidas son rechazadas inmediatamente.
- c) En otros esquemas, por ejemplo en el cálculo de predicados, la representación de una situación puede admitir varios modelos reales, es decir, pueden darse diversas situaciones del mundo real, cuya

representación se lleve a cabo con el mismo conjunto de sentencias, puesto que las diferencias entre aquellas, pueden consistir en aspectos no capturados en la representación. Por el contrario, un esquema de representación directo es, usualmente, más exhaustivo y específico, con lo que admite menos modelos y hace más eficiente la resolución de problemas.

Hay, sin embargo, inconvenientes asociados al uso de los esquemas de representación directa, entre ellos:

- a) La misma tendencia hacia esquemas específicos presenta ineficiencia, cuando se requieren modelos más generales para la resolución de problemas. Por ejemplo, dado el problema: un robot está en la habitación A y se mueve sucesivamente entre las habitaciones A y B. ¿en que habitación estará después de 377 movimientos?. En este caso, la mejor solución no es simular la acción sino generalizar los efectos según que el número total de movimientos sea par o impar.
- b) Ciertos tipos de información incompleta son difíciles de representar.

Un ejemplo de representación directa, lo constituye el sistema General Space Planner ⁹⁰. Este resuelve tareas relativas a la ordenación de objetos en el espacio, bajo ciertas restricciones que deben ser satisfechas. Un sencillo problema a resolver sería el representado en la Figura 4. En este, el sistema usa representación directa para reflejar la estructura del espacio y, lo describe mediante las propiedades de tamaño, forma y posición. El sistema resuelve el problema, encontrando localizaciones para los sucesivos objetos, comprobando, que no se violan las restricciones impuestas. La búsqueda de la solución se facilita estableciendo los efectos de las restricciones entre pares de objetos, presentándose el conjunto de restricciones de forma proposicional.

⁹⁰ EASTMAN, C.M. (1.970): "Representations for space planning". En BARR, A. y FEIGENBAUM, E. (1.982): "The handbook of Artificial Intelligence" Volume II William Kaufmann. Los Altos, California

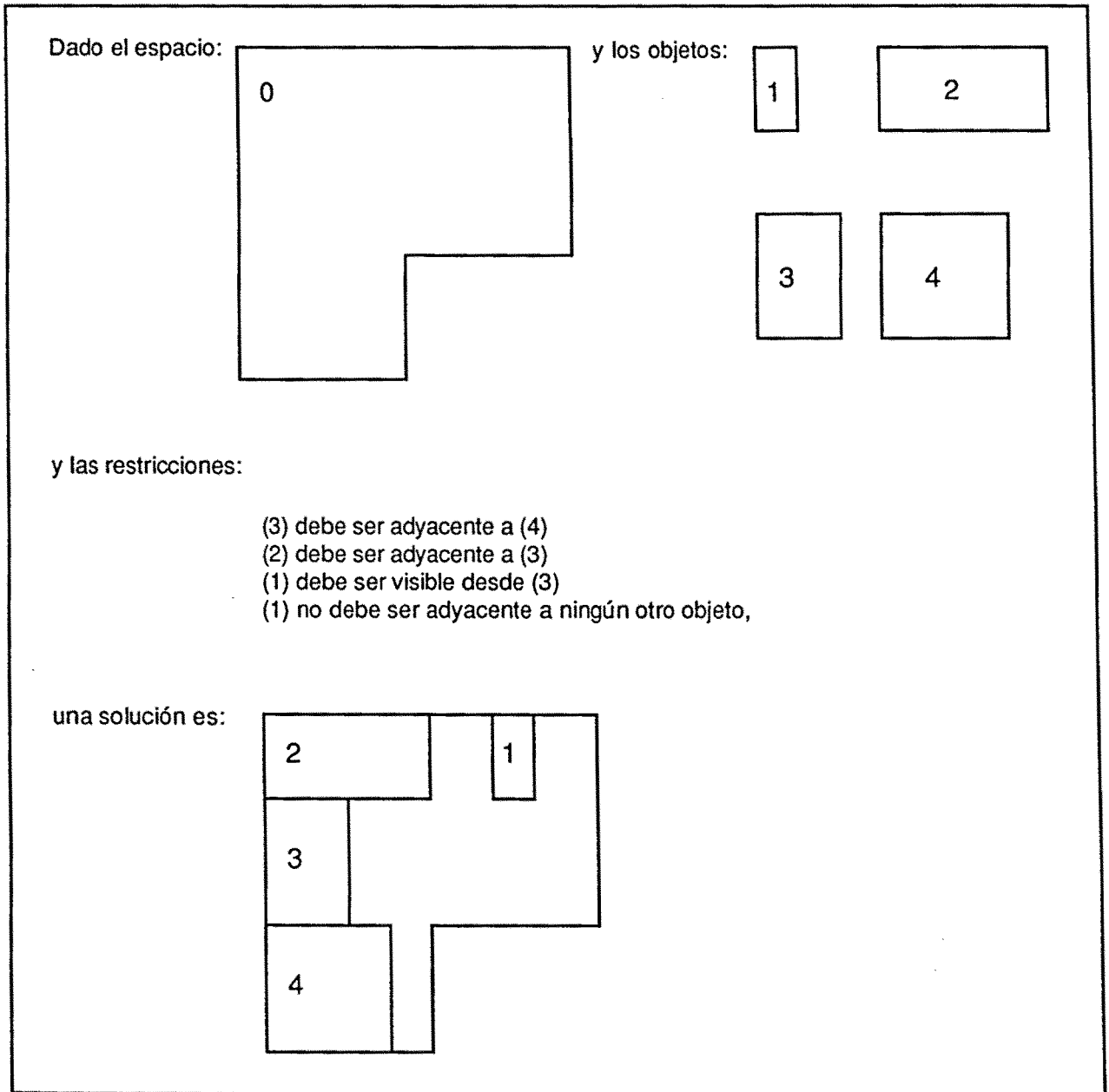


Fig.4 Ejemplo de representación directa

Otros sistemas que utilizan representación directa son el Geometry Theorem Prover diseñado en 1.959 por Gelenter, con el objeto de resolver problemas de geometría elemental y el WHISPER creado por Funt en 1.977, un sistema diseñado para razonar a partir de representaciones analógicas exclusivamente. El programa opera en un mundo de bloques simples, analizando su movimiento

2.3.6 FRAMES Y SCRIPTS

Se desarrollaron por investigadores en Inteligencia Artificial como consecuencia de la búsqueda de métodos eficientes de organización de la ingente cantidad de conocimiento requerido en las tareas cognitivas. Ambos son métodos que permiten representar conocimiento y están particularmente orientados a facilitar la integración de nuevos objetos y hechos, propios de situaciones determinadas, al esquema de representación.

Los "frames", fueron propuestos originalmente por Minsky en 1.975 ⁹¹ y se configuran como una estructura en la que los nuevos datos se interpretan según conceptos previos adquiridos a través de la experiencia. Este tipo de organización del conocimiento permite la existencia de procesos dirigidos por expectativas. La búsqueda de nuevo conocimiento se orienta según el contexto y de acuerdo con la experiencia adquirida. El mecanismo de representación que hace posible esta clase de razonamiento es el "slot". Este describe aspectos del objeto representado por el "frame", y es una pieza de conocimiento dentro de la estructura creada por el "frame". Un "frame", para un concepto genérico, posee un cierto número de "slots"; un "frame", para una concreción particular, o instancia, de ese concepto tendrá los mismos "slots", siendo su contenido más preciso que el del "frame" genérico, ya que se refiere a un objeto particular y no a un concepto. Un ejemplo muy simple de "frame" se utiliza en la Figura 5 para representar el concepto genérico de silla, incluyendo "slots" para representar sus características. El

⁹¹ MINSKY, M. (1.975): "A framework for representing knowledge". En BARR, A., COHEN, P. R. y FEIGENBAUM, E. (1.989): " *The handbook of Artificial Intelligence*" Volume IV Addison-Wesley Publishing Company Massachusetts.

"frame" donde se representa información sobre una silla determinada, en el ejemplo "SILLA-DE-JUAN", tiene los mismos "slots" que el anterior y heredados de el, pero su contenido es mucho más específico.

SILLA Frame	
Especialización-de:	MUEBLES
Estilo:	clásica, ergonómica,...
Número-de-patas:	un entero (DEFAULT=4)
Número-de-brazos:	0,1, o 2
SILLA-DE-JUAN Frame	
Especialización-de:	SILLA
Estilo:	ergonómica
Número-de-patas:	4
Número-de-brazos:	0

Fig. 5 Ejemplo de Frame

Algunos "slots" se usan para establecer jerarquías de propiedades entre los "frames", lo que permite que la información de los "frames" padres sea heredada por los "frames" hijos, de forma similar a como actúan los enlaces "es-un" en una red semántica. Otros "slots" poseen "subslots" que en ciertos casos pueden ser a su vez "frames", admitiendo también la asunción de valores por defecto y el establecer condiciones para cada "slot", con la posibilidad de generar, a su vez, "subframes". Cuando la información contenida en un "slot" varia, procedimientos contenidos en el propio "slot", pueden entrar en ejecución. Así, operaciones como añadir, suprimir o consultar información, pueden ejecutar sus propios procedimientos. Estos, a su vez, pueden modificar valores en otros

"slots" continuando, así, el proceso de activación de "slots", hasta que se alcance el objetivo dado. Puesto que cada "slot" puede corresponder a un "frame" y un mismo "frame" puede formar parte, como "slot", de otros "frames", disminuye la necesidad de representar información redundante con las ventajas que ello conlleva.

Los "scripts" desarrollados por Schank y Abelson en 1.977 ⁹², están especialmente diseñados para representar una secuencia de hechos. En un script se puede representar una situación, especificando secuencias de hechos habituales, pero también son admisibles secuencias por defecto, excepciones e información con posibilidad de errores. La estructura de representación de un script está fuertemente interconectada y el valor de un "slot" de una secuencia afecta al valor de otros "slots" en esa secuencia.

La diferencia entre un "frame" y un script estriba en la especialización de esta última estructura de representación. Los componentes esenciales de un script son ⁹³:

- a) *Condiciones de entrada*: son condiciones que se tiene que satisfacer antes de que los sucesos descritos en el script se cumplan.
- b) *Resultado*: son condiciones que serán verdad cuando los sucesos descritos en el script se hayan cumplido.
- c) *Propiedades*: son "slots" que representan objetos involucrados en los sucesos descritos en el script.
- d) *Personajes*: son "slots" que representan a las distintas personas que intervienen en los sucesos descritos en el script.
- e) *Instancia*: es una variación específica de un modelo más general que se ha representado mediante el script.
- f) *Escenas*: son las diferentes secuencias de sucesos que ocurren.

Los "scripts" requieren el uso de unas pocas descripciones que se refieran a "frames", tal como propiedades y personajes.

⁹² SHANK, R. C. y ABELSON, R. P. (1.977): "Scripts, Plans, Goals and Understanding". En GEVARTER, W. M. (1.987): "Maquinas Inteligentes. Una panorámica de la Inteligencia Artificial y de la Robótica" Ed. Diaz de Santos S.A. Madrid.

⁹³ RICH, E. (1.984): "Artificial Intelligence". McGraw-Hill. New York.

El razonamiento en ambas estructuras, a pesar de presentarse éstas como declarativas, puede ser dirigido por procedimientos ligados a los "slots". En algunos sistemas, los procedimientos son el mecanismo principal para dirigir el razonamiento. Los procedimientos pueden activarse para rellenar "slots" o pueden entrar en ejecución cuando un "slot" ha sido completado. En el primer caso, una vez que un "frame" o un script particular ha sido seleccionado para representar el problema en curso, el primer proceso en el sistema de razonamiento previsto es, completar los valores de sus "slots". La asignación de los valores, que puede llevarse a cabo bien por herencia o bien por defecto en caso de ausencia de valor, son métodos relativamente sencillos de completar "slots", puesto que no requieren complejos procesos de razonamiento. Así, un nuevo "frame" para interpretar una situación determinada, puede hacer uso de valores obtenidos por experiencias anteriores, sin necesidad de un nuevo proceso. Cuando la información que se requiere debe ser deducida, los procedimientos asociados proporcionan una forma de especificar métodos heurísticos apropiados que aprovechen el contexto actual. El completar "slots", provee información sobre si el "frame" o script es apropiado para comprender la escena o suceso. En caso de que se mostrara inapropiado, el procedimiento asociado accionaría la transferencia de control a otros "frames".

Otro tipo de procedimiento asociado es el consistente en rutinas que entran en ejecución cuando determinados hechos o datos ocurren o son modificados. En algunos casos, los procedimientos activados se asocian a "slots" especiales y se usan, por ejemplo, para decidir que hacer en caso que el "frame" no se adapte al contexto actual. Así, un "frame" proporciona una representación estructurada de un objeto o clase de objetos y puede ejecutar, también, de forma automática un conjunto de inferencias que amplíen el conocimiento representado. Además, los "frames" pueden usarse para representar reglas de producción con las que guiar el razonamiento en el dominio. Al estar cada regla representada como un "frame", éstas se pueden agrupar fácilmente en clases, permitiendo una organización de las reglas en pequeños módulos fácilmente manejables.

Un esquema de representación del conocimiento del tipo "frame" permite agrupar las reglas en una clase y asociarla con el "frame" que representa a la clase de los objetos sobre los que actuarán las reglas. Las reglas se invocarán como un grupo cada vez que el sistema ejecute la tarea de diagnóstico sobre cualquier objeto particular de la clase. Esta forma de agrupar reglas en clases constituye el conjunto de las meta-reglas del sistema.

Los "frames", como esquemas de representación de conocimiento, son utilizados por un número importante de sistemas. Así, en comprensión del lenguaje natural, en el sistema GUS desarrollado en 1.977 por Bobrow y diseñado como prototipo, para la reserva automática de plazas en una compañía aérea. El sistema utiliza los "frames" para representar el conocimiento necesario para la comprensión de diálogos. Otro sistema basado en "frames" es NUDGE, -desarrollado por Goldstein y Roberts, también en 1.977- para la interpretación y comprensión de incompletas o contradictorias solicitudes administrativas, para permitir su posterior tratamiento mediante métodos convencionales de proceso.

Los sistemas basados en "scripts", han sido desarrollados principalmente por Schank y Abelson y utilizados, entre otras, en tareas de comprensión de secuencias de hechos. En particular el programa SAM, intenta la comprensión de breves historias, usando un script para guiar la interpretación de los hechos que acaecen en ellas. El disponer de sistemas expertos como herramientas de construcción de nuevos sistemas, ha permitido extender la aplicación de estos a nuevas áreas, logrando éxitos notables, como es el caso de R1 ⁹⁴ un sistema experto para configurar los sistemas informáticos VAX de Digital Corporation, que representa la aplicación de más éxito de la herramienta OPS ⁹⁵ en la construcción de sistemas expertos.

⁹⁴ McDERMOTT, J. (1.980): "R1: A rule-based configurer of computer systems". En BENCH-CAPON (1.990): "Knowledge Representation: An Approach to Artificial Intelligence" Academic Press London.

⁹⁵ OPS es un lenguaje para programar sistemas expertos basados en reglas, derivado de la aplicación de psicología cognitiva, desarrollado por Forgy y McDermott en 1.977

2.4 SINTESIS EVOLUTIVA DE LOS SISTEMAS EXPERTOS

La evolución experimentada en el desarrollo de los sistemas expertos, como área de la Inteligencia Artificial, ha sido paralela a la seguida por ésta. En este sentido, los sistemas expertos se han beneficiado de las investigaciones llevadas a cabo en otros campos de la Inteligencia Artificial como, por ejemplo, los desarrollos de procesos en lenguaje natural o los desarrollos de métodos generales de resolución de problemas. Ambas investigaciones han influido en la evolución de los sistemas expertos, las primeras por su especial relevancia en la construcción de interfaces de usuario y las segundas por la importancia de su aplicación a la selección y control del conocimiento con el que opera el sistema. Los sistemas expertos han tenido también una evolución propia, en cierta forma espectacular, provocada por la incorporación de una, cada vez mayor, cantidad de conocimiento al sistema, lo que los convierte en programas de alto rendimiento para la resolución de complejos problemas simbólicos en dominios especializados.

En particular la evolución en el desarrollo de los sistemas expertos, puede concretarse en tres fases: ⁹⁶

- a) Etapa de iniciación, que abarca los años 1.965 a 1.970, en la que se construyen los primeros sistemas expertos. El sistema DENDRAL desarrollado en la Universidad de Stanford junto con el sistema MACSYMA desarrollado en el M.I.T. son los programas pioneros. Estos sistemas se diseñaron para manipular y explorar simbólicamente problemas complejos. Problemas que se caracterizan por el incremento del número de soluciones posibles, cuando crecen en complejidad. Alcanzar la mejor solución posible, a las especificaciones del problema, en un espacio de búsqueda de crecimiento exponencial está en el origen de su construcción. Es el caso del primero de los programas mencionados, utilizado para resolver problemas de estructura química molecular a partir del análisis de espectrometría de masas. El programa genera estructuras moleculares

⁹⁶ CASTILLO, E. y ALVAREZ, E. (1.989): "Sistemas Expertos. Aprendizaje e Incertidumbre" Paraninfo S.A. Madrid.

parciales, consistentes con los datos, y las elabora de todos los modos posibles. La generación y análisis sistemático de dichas estructuras posibles que cumplen las restricciones impuestas por los datos, evita la omisión de alguna posible solución, omisión que ocasionalmente podría ser cometida por un experto humano.

También, dentro de esta etapa, cabe mencionar otra línea en el desarrollo de sistemas expertos, que se inicia con el programa SAINT en 1.961 y tiene continuidad en el programa MACSYMA desarrollado por Martin y Faterman en 1971, en el Massachusetts Institute of Technology. MACSYMA es un sistema experto para el cálculo diferencial e integral, mediante manipulación simbólica de expresiones matemáticas. El programa opera con información simbólica e incorpora conocimiento experto que le permite alcanzar un alto rendimiento en la simplificación de expresiones matemáticas.

- b) Etapa de experimentación y desarrollo, entre 1.970 y 1.980, en la que se producen éxitos notables en la construcción de sistemas expertos, con el desarrollo de los sistemas más conocidos: CADUCEUS de la Universidad Carnegie-Mellon y MYCIN de la Universidad de Stanford son un ejemplo de ello. Ambos sistemas tienen su aplicación en el campo de la medicina para el diagnóstico de enfermedades. El conocimiento en CADUCEUS se estructura en una gran red semántica que en 1.982 estaba constituida por aproximadamente unas 100.000 asociaciones, representativas del 85 por ciento del conocimiento relevante de la época sobre medicina interna. Un conocimiento disponible para el programa en orden a relacionar síntomas con enfermedades. En MYCIN, tal conocimiento se representa en reglas, que en número aproximado a 400, plantean posibles condiciones asociadas a sus correspondientes interpretaciones, utilizadas en el diagnóstico y tratamiento de enfermedades infecciosas.

En ambos programas, tomados como ejemplo, se pone de manifiesto la importancia que tiene para el rendimiento de un sistema experto el disponer de abundante conocimiento y de una adecuada representación del mismo. A este respecto Feigenbaum, en la International Joint Conference on Artificial Intelligence -IJCAI- en 1977, sienta las bases para la orientación futura del trabajo de los investigadores en este área de la Inteligencia Artificial ⁹⁷. Pondría de manifiesto que el poder de un sistema experto deriva del conocimiento que éste posee, mientras que la representación del conocimiento y los procedimientos de inferencia son los mecanismos que utilizan dicho conocimiento. El papel central del conocimiento en la construcción de sistemas expertos es asumido, de este modo, por los investigadores dando origen a la construcción de sistemas más potentes con altos rendimientos. Es el caso de PROSPECTOR ⁹⁸, empleado para evaluar prospecciones geológicas con el fin de hallar yacimientos minerales. PROSPECTOR, tiene disponible cerca de una docena de Bases de Conocimientos referidas a los distintos tipos de depósitos minerales. Como en MYCIN, el conocimiento se representa mediante reglas, usadas para obtener el diagnóstico más probable en función del grado de confianza en las condiciones previas. El programa, por tanto, da a los usuarios posibles interpretaciones de los datos e identifica qué observaciones adicionales podrían ser útiles para alcanzar conclusiones más seguras.

En MYCIN primero y, posteriormente, en PROSPECTOR y programas siguientes, se utilizó razonamiento impreciso, con asignación a los elementos de conocimiento de grados de certidumbre con valores

⁹⁷ FEIGENBAUM, E. A. (1.977): "The art of artificial intelligence: Themes and case studies of Knowledge engineering" . En HAYES-ROTH, F., WATERMAN, D. y LENAT, D. (1.983): "**Building Expert Systems**" Addison-Wesley. Massachusetts.

⁹⁸ DUDA, R., GASCHING y HART, P. (1.979): "Model design in the PROSPECTOR consultant system for mineral exploration" . En MATE, J. L. y PAZOS, J. (1.988): "**Diseño y Construcción de Sistemas Expertos**" Sepa S.A. Argentina Córdoba.

comprendidos entre 0 y 1. Esta asignación es necesaria en diversos dominios donde, como en medicina, las implicaciones del problema no están suficientemente entendidas o completamente especificadas. En estos casos el conocimiento del que dispone el programa puede ser incierto e incompleto.

También se inicia con MYCIN la diferenciación entre la Base de Conocimiento y el mecanismo de inferencia, en contraste con los sistemas de la etapa anterior DENDRAL y SAINT cuyo Motor de Inferencia estaba especialmente diseñado para trabajar con masa espectrométrica y álgebra, respectivamente. La separación entre la Base de Conocimiento y el Motor de Inferencia permitió desarrollar "sistemas vacíos" o "conchas", como resultado de eliminar el conocimiento de la Base y conservar todos los demás módulos. Estos sistemas se constituyen así, en potentes herramientas para la construcción de nuevos programas, si bien presentan la limitación de obligar a representaciones de conocimiento mediante estructuras prefijadas por la "concha" empleada. Un ejemplo es EMYCIN⁹⁹, versión de MYCIN desarrollada en Stanford e independiente del dominio de aplicación de éste. EMYCIN, está formado por todos los componentes de MYCIN excepto su conocimiento sobre enfermedades infecciosas. EMYCIN facilitó la construcción de nuevos sistemas expertos como, PUFF utilizado en el estudio de enfermedades pulmonares, CLOT para problemas de coagulación y HEAMED en psicofarmacología, entre otros. Ampliaciones de EMYCIN serán; TEIRESIAS, que incorpora un módulo para adquisición y modificación del conocimiento y GUIDON, que incorpora un módulo para explicación y funciones de tutoría.

⁹⁹ Van MELLE, W. (1.979): "A domain-independent production-rule system for consultation programs" . En HAYES-ROTH, F., WATERMAN, D. y LENAT, D. (1.983): "Building Expert Systems" Addison-Wesley Massachusetts.

De manera similar, de PROSPECTOR se derivó la "concha" KAS ¹⁰⁰. Knowledge Acquisition System- y la Rand Corporation desarrolló ROSIE ¹⁰¹ en 1.981, como una herramienta de propósito general para la construcción de sistemas expertos. La construcción de ROSIE se inspira en MYCIN. Así, la representación del conocimiento en ROSIE se realiza mediante reglas y, se siguen similares procedimientos de explicación al usuario del razonamiento seguido por el programa. Pero, además, ROSIE incorpora nuevas facilidades, como la posibilidad de comunicación interactiva entre el usuario y el sistema y la de un entorno de programación próximo al lenguaje natural. ROSIE es la primera herramienta diseñada para la construcción de una amplia gama de sistemas expertos.

- c) Etapa de industrialización. A partir de 1.980 crece el interés comercial de las aplicaciones de sistemas expertos, lo que provoca la entrada en este nuevo campo de empresas ya establecidas como Xerox e IBM, entre otras, y la creación de nuevas empresas dedicadas al desarrollo y comercialización de estos sistemas.

El esfuerzo financiero asignado al desarrollo de sistemas expertos ha sido muy considerable. Una estimación de la inversión se indica en la Tabla 1., que integra información de Francia, Alemania, Italia, Reino Unido y Estados Unidos. Las categorías representadas en la tabla se refieren, por lo que respecta a Lenguajes de Inteligencia Artificial, al desarrollo global de estos, incluidos los dirigidos a su aplicación en sistemas expertos. Los "shells" o "conchas" genéricos se configuran como herramientas de desarrollo de sistemas expertos de aplicación general. Por su parte, los "shells" de aplicaciones son herramientas de desarrollo de sistemas expertos

¹⁰⁰ DUDA et al. (1.978) "Semantic network representations. In rule-based inference systems". En WATERMAN, D. A. (1.986): "A Guide to Expert Systems". Addison-Wesley Massachusetts.

¹⁰¹ FAIN, J. et al.(1.981): "The ROSIE language reference manual" . En HAYES-ROTH, F., WATERMAN, D., LENAT, D. (1.983): "Building Expert Systems" Addison-Wesley. Massachusetts.

orientadas hacia áreas de aplicación específica. Los sistemas aplicados son sistemas desarrollados a medida de las necesidades del usuario.

El disponer de sistemas expertos, como herramientas de construcción de nuevos sistemas, ha permitido extender la aplicación de estos a nuevas áreas, logrando éxitos notables, como es el caso de R1 ¹⁰² un sistema experto para configurar los sistemas informáticos VAX de Digital Corporation, que representa la aplicación de más éxito de la herramienta OPS en la construcción de sistemas expertos.

Productos software para S.E. (Valoración en millones \$)	1987	1988	1989	1990	1991	1992	1995
Inversión total en desarrollo	178	227	311	414	539	688	1,278
División por categorías:							
Lenguajes de Inteligencia Artificial	45	55	72	91	113	137	216
"Shells" Genéricos	80	102	140	186	243	310	577
"Shells" de Aplicaciones.....	16	23	35	51	72	98	221
Sistemas Aplicados.....	37	47	64	86	112	142	265

Tabla 1. Inversión en desarrollo de software orientado hacia sistemas expertos

Fuente OVUM PRESS ¹⁰³

También es de destacar, en ésta etapa, una línea importante de investigación y desarrollo en programas que tiene como objeto la identificación de la palabra hablada. Tal es el caso de la herramienta

¹⁰² McDERMOTT, J. (1980): "R1: A rule-based configurer of computer systems". En BENCH-CAPON (1.990): "Knowledge Representation: An Approach to Artificial Intelligence" Academic Press London.

¹⁰³ OVUM Ltd. (1.988): "Expert Systems in banking and securities" London.

HEARSAY-III ¹⁰⁴, derivada del sistema experto HEARSAY-II, desarrollado en la Universidad de Carnegie-Mellon, sistema capaz de comprender un discurso hablado con integración de un vocabulario de aproximadamente 1.000 palabras. La herramienta HEARSAY-III, está diseñada para ayudar al usuario en el desarrollo de métodos para representar y aplicar diversas fuentes de conocimiento en la resolución de un problema. Da al usuario facilidades para comunicarse con el sistema, para interrumpir el proceso, si se hace necesario, para la captura de nuevos datos y, finalmente para explicar el proceso seguido.

El proyecto Alvey patrocinado por el Reino Unido e iniciado a partir del informe del mismo nombre del año 1.982, recoge distintos proyectos de desarrollo de interfaces en lenguaje natural para sistemas expertos, entre ellos: el ALVEY IKBS 075 que promovido por la Imperial Cancer Research Found ¹⁰⁵ se orienta al diseño interfaces en lenguaje natural para sistemas expertos en el área médica.

La necesidad de disponer de interfaces hombre/máquina en lenguaje natural, es uno de los objetivos perseguidos en el proyecto japonés de construcción del *Ordenador de la Quinta Generación* ¹⁰⁶. Ordenador capaz de procesar conocimientos con las siguientes características: disponer de una interface capaz de admitir comunicación coloquial con el usuario, en lenguaje natural hablado y/o escrito, gráficos, etc. y, también, para comprender el significado de la información de entrada y poder, de esta forma, mantener un diálogo genuinamente inteligente con el usuario,

104 BALZER, R. ERMAN, P. Y LONDON (1.980): "HEARSAY-III: A domain-independent framework for expert systems". En HAYES-ROTH, F., WATERMAN, D.y LENAT, D. (1.983): "Building Expert Systems" Addison-Wesley Massachusetts.

105 SAVORY, S.E. (Ed). (1.988): "Artificial Intelligence y Expert Systems" Ellis Horwood Limited, West Sussex.

106 SIMONS, G.L. (1.983): "Los ordenadores de la quinta generación" Ed. Diaz de Santos S.A. Madrid.

respondiendo preguntas, haciendo sugerencias o solicitando del usuario respuestas concretas.

2.4.1 FUNCIONES

La versatilidad de un sistema experto viene determinada por su propia estructura. La cierta independencia de sus componentes y la variedad de conocimiento que puede ser incorporado al sistema, hacen posible que los sistemas expertos lleven a cabo una variedad de funciones. Entre ellas se pueden distinguir como básicas ¹⁰⁷:

- a) *interpretación*: análisis de datos para determinar su significado. La interpretación requiere un análisis riguroso y completo de los datos, capaz de considerar sistemáticamente todas las posibles interpretaciones, desechando aquéllas para las que se dispone de evidencia en contra. La interpretación se dificulta cuando los datos contienen errores o están distorsionados o simplemente faltan. En estos casos la interpretación debe producirse: 1) a partir de información parcial, caso de carencia de datos; 2) construyendo las hipótesis adecuadas sobre la validez de los datos, si estos presentan contradicciones; 3) dando poca credibilidad a los resultados de una interpretación, cuando ésta se basa en datos poco fiables, datos dudosos o incompletos. Para reforzar la credibilidad de la interpretación, es importante identificar que información es incompleta o dudosa y, en su caso, las presunciones realizadas.
- b) *diagnóstico*: proceso de búsqueda de los defectos de funcionamiento de un sistema, basado en las características o síntomas que presenta. Para proceder a la elaboración de un diagnóstico, se requiere conocer la estructura del sistema objeto de estudio y las relaciones e interacciones entre sus componentes. Será pertinente considerar que: 1) ciertos fallos pueden, en ocasiones, estar enmascarados presentando síntomas

¹⁰⁷ HAYES-ROTH, F., WATERMAN, D. y LENAT, D. (1.983): "Building Expert Systems" Addison-Wesley Massachusetts.

equivocos que el sistema experto debe ser capaz de identificar; 2) los fallos pueden presentarse de modo intermitente, a veces, es necesario forzar el sistema para ponerlos de manifiesto; 3) la obtención de información sobre el sistema puede resultar en ocasiones inaccesible, cara o incluso peligrosa. En la elaboración del diagnóstico se decidirá la información necesaria.

- c) *previsión*: descripción de una situación anticipadamente a partir del modelo construido sobre base histórica o por aprendizaje. La previsión requiere tener en cuenta el tiempo en el proceso de razonamiento, recogiendo el modo en que cierta información varía en el tiempo. La previsión se produce sobre problemas que conllevan información incompleta; cuando se dispone de toda la información, no procedería la aplicación de estas técnicas de la Inteligencia Artificial.
- d) *planificación*: definición en el tiempo y en el espacio de las acciones, encaminadas a lograr la consecución de un objetivo. La planificación requiere la construcción de un plan, donde: 1) se establezcan prioridades en caso de conflicto de objetivos; 2) se produzca el menor consumo posible de recursos; 3) se respeten las restricciones impuestas por el entorno y, 4) sea lo suficientemente flexible como para incorporar conocimiento incompleto o cambios relacionados con el tiempo. La planificación se requiere en problemas muy complejos, para los que se hace necesario tantear la posibilidad de planes alternativos y en aquellos que presentan múltiples interacciones entre los diferentes objetivos intermedios.
- e) *diseño*: definición de las especificaciones necesarias para la construcción de objetos, teniendo en cuenta ciertas restricciones. El diseño de circuitos digitales sería un ejemplo de sistema experto, aplicado a una función de diseño. Los problemas que plantea el diseño se asemejan a los comentados en la planificación. En éste también, es necesario el disponer de diseños alternativos puesto que, en la mayoría de los casos, las

consecuencias de las decisiones de diseño no son evidentes. De igual modo, cuando es un problema muy complejo, la subdivisión del proceso de diseño en partes, obliga a tener en cuenta las interacciones que se producen entre ellas.

2.4.2 DOMINIOS

La variedad de funciones que pueden ser incorporadas a un sistema experto hace que sus áreas de aplicación sean también amplias. En la actualidad se dispone de sistemas expertos aplicados en dominios como: química, matemáticas, medicina y enseñanza asistida por ordenador, como ejemplo de aplicaciones principales. Sin embargo, la investigación es muy activa en este campo de la Inteligencia Artificial y provoca el desarrollo de nuevas aplicaciones en variedad de áreas de conocimiento. Es conveniente comentar algunas de las características más relevantes de los sistemas expertos incluidos en las áreas de aplicación anteriormente mencionadas.

En química, y más específicamente en química orgánica, los sistemas expertos desarrollados, son de aplicación a dos grandes grupos de problemas: a) en la determinación de la estructura molecular de un compuesto orgánico desconocido y, b) en la planificación de la secuencia de operaciones para sintetizar un compuesto orgánico.

La determinación de estructuras moleculares tiene aplicación en diversas áreas; es clave, por ejemplo, en medicina para la identificación de medicamentos y su metabolismo en estudios clínicos y, en biología para la identificación de compuestos químicos en organismos terrestres y marinos. El proyecto DENDRAL (1965) ¹⁰⁸ de la Universidad de Stanford, uno de los pioneros en el desarrollo de sistemas expertos, incorpora técnicas para generar las posibles estructuras moleculares de un compuesto a partir de datos como: átomos que la forman y las relaciones que se establecen entre ellos en una estructura tridimensional. El programa utiliza el algoritmo desarrollado por

¹⁰⁸ LINDSAY, R. , BUCHANAN, B. G., FEIGENBAUM, E. A. y LEDERBERG, J.(1.980): "Applications of artificial intelligence for organic chemistry. The DENDRAL Project". McGraw-Hill, New York.

Lederberg en 1964 para la enumeración de estructuras moleculares y, conocimiento heurístico para determinar, de todas las posibles, un pequeño grupo de estructuras moleculares de acuerdo con los datos. Consecuencia de sucesivas revisiones y ampliaciones del programa DENDRAL son los programas, Meta-DENDRAL (1970) y CONGEN (1976) ¹⁰⁹.

En cuanto a la síntesis de compuestos orgánicos, tres son los principales programas LHASA ¹¹⁰ (Logic and Heuristics Applied to Synthetic Analysis) de la Universidad de Harvard, SECS ¹¹¹ (Simulation and Evaluation of Chemical Synthesis) de la Universidad de California y SYNCHEM ¹¹² (SYNthetic CHEMistry) de la Universidad de New York. En el programa SYNCHEM el procedimiento de búsqueda de soluciones está definido y no permite interactuar con el usuario para su modificación. Sin embargo, los programas LHASA y SECS permiten a éste guiar la búsqueda de la solución. Generalmente los resultados obtenidos del sistema mejoran con la ejecución guiada por el usuario. El interés de la síntesis de compuestos orgánicos reside en su aplicación a la creación de nuevos productos químicos y en el desarrollo de procesos de elaboración, más eficientes, para productos ya conocidos. Los sistemas expertos en este campo, ayudan a los químicos en la exploración sistemática de las posibilidades de síntesis de compuestos orgánicos y, en la consulta de la ingente información disponible sobre reacciones químicas. El método de exploración exhaustiva, se basa en el desarrollado por Corey y Wipke en 1969, donde la exploración del árbol de posibles reacciones químicas es guiado interactivamente por el usuario especializado, para designar aquellas ramas más interesantes y que, con mayor probabilidad, producirán los resultados deseados.

¹⁰⁹ BUCHANAN, B. G. y FEIGENBAUM, E. A. (1.978): "DENDRAL y MetaDENDRAL: Their applications dimension". En HAYES-ROTH, F., WATERMAN, D. y LENAT, D. (1.983): "Building Expert Systems" Addison-Wesley Massachusetts.

¹¹⁰ COREY, E.J. y WIPKE, W. T. (1.969): "Computer assisted design of complex organic synthesis". En WATERMAN, D. A. (1.986): "A Guide to Expert Systems". Addison-Wesley Massachusetts.

¹¹¹ WIPKE, W. T. et al. (1.978): "Simulation and evaluation of chemical synthesis-SECS: an application of artificial intelligence techniques". En BARR, A. y FEIGENBAUM, E. (1.981): "The handbook of Artificial Intelligence" Volume II William Kaufmann. Los Altos, California

¹¹² GELERNTER, H. L. et al. (1.977): "Empirical explorations of SYNCHEM". En WATERMAN, D. A. (1.986): "A Guide to Expert Systems". Addison-Wesley Massachusetts.

MACSYMA es un sistema experto interactivo, diseñado para ayudar a los usuarios, principalmente matemáticos e ingenieros, en la búsqueda de la solución de problemas matemáticos. El diseño original de MACSYMA fue realizado por Engleman, Martin y Moses,¹¹³ -investigadores del M.I.T.- en 1968, y tuvo sus primeros usuarios en 1971. Actualmente investigadores de universidades, centros estatales y empresas privadas utilizan el programa, accesible a través de los servicios ARPA. MACSYMA es un sistema que dispone de una gran base de conocimientos, pudiendo realizar al menos 600 operaciones matemáticas distintas, incluyendo diferenciación, integración, sistemas de ecuaciones, operaciones con matrices, etc.

CBC ¹¹⁴ es un sistema experto, desarrollado en los años 70, de ayuda en la reparación y modificación de complejos equipos electromecánicos. Dispone de una base de conocimientos especializada, que utiliza el sistema para contestar las preguntas del usuario, mostrando al mismo tiempo la tarea desarrollada, y proporcionando los avisos necesarios sobre el desarrollo de aquella. El sistema se ajusta a la formación del usuario, disponiendo de distintos niveles de detalle en la resolución del problema. Las preguntas pueden ser hechas en forma hablada y el usuario recibe las respuestas en la misma forma, para lo que dispone de un sintetizador de voz para generar las respuestas y de un sistema de reconocimiento de voz para entender las preguntas del usuario. La principal aportación del trabajo de investigación sobre los sistemas de consulta, como el CBC, ha sido la de crear tecnología básica independiente del dominio específico sobre el que versa la base de conocimientos.

A partir de 1975 nuevos sistemas de consulta se han comenzado a usar en dos dominios completamente diferentes: en la exploración de minerales y en la agricultura. PROSPECTOR ¹¹⁵ es un sistema experto de consulta desarrollado inicialmente por SRI

113 ENGELMAN, C. (1.971): "The legacy of MATHLAB 68". En HAYES-ROTH, F., WATERMAN, D. y LENAT, D. (1.983): "**Building Expert Systems**". Addison-Wesley Massachusetts.

114 NILSSON, N.J. (1.971): "**Problem-Solving Methods in Artificial Intelligence**". MacGraw-Hill New York.

115 DUDA, R. O. et al. (1.978): "Development of the PROSPECTOR consultation system for mineral exploration". En MATE, J. L. y PAZOS, J. (1.988): "**Diseño y Construcción de Sistemas Expertos**" Sepa S.A. Argentina Córdoba.

International para ayudar a los geólogos en la localización de depósitos de minerales. El programa toma los datos correspondientes a una determinada situación, datos que provienen de observaciones geológicas, y los compara con los *modelos* que tiene almacenados. Unos modelos que son descripciones formales de los más importantes tipos de yacimientos minerales. El sistema puede admitir datos inciertos o incompletos y, en estos casos, la conclusión se expresa en términos de probabilidad advirtiéndole al usuario de posibles interpretaciones diferentes de los datos y, solicitando más información, en caso necesario, para poder mejorar la conclusión alcanzada. En PROSPECTOR hay una clara separación entre el conocimiento de la base -geológico- y los mecanismos que emplean ese conocimiento para alcanzar la solución del problema planteado.

Los sistemas expertos desarrollados como ayuda al diagnóstico médico, se han mostrado eficientes y el dominio sobre el que están definidos fecundo para la investigación en este área de aplicación de las técnicas de Inteligencia Artificial. Dos causas básicamente lo explican. Primero, hay un claro beneficio para la sociedad derivado de la posibilidad de mejorar -hacerlo más fiable y exhaustivo- el diagnóstico médico. Así, los sistemas expertos partiendo de los datos adecuados de un paciente, pueden llevar a cabo un estudio exhaustivo de las enfermedades que mejor se ajustan a los síntomas que éste presenta, aconsejando un tratamiento y, en su caso, la dosificación de las medicinas recetadas. En segundo lugar, las características principalmente de la medicina clínica: la configuran como un campo muy apropiado para el estudio de los procesos cognitivos. Los requisitos de disponer de una gran base de conocimientos definida sobre un dominio específico y la existencia de expertos humanos capaces de aportar la mejor solución a problemas difíciles se da en este campo. Si a esto unimos, que en un determinado dominio los problemas a resolver son en cierta forma repetitivos, el diagnóstico médico se configura como un problema adecuado para su modelización como sistema basado en el conocimiento. Algunos de los programas más destacados en este

campo son: MYCIN (1.976), CASNET (1.978), INTERNIST (1.975), IRIS (1.977), EXPERT (1.979), ONCOCIN (1.981) y PUFF (1.981).

MYCIN, desarrollado por Shortliffe ¹¹⁶, es un programa de consulta para el diagnóstico de enfermedades infecciosas. El programa interroga al médico acerca de las características del paciente, a partir de las que infiere el diagnóstico y selecciona la terapia más apropiada. MYCIN fue uno de los primeros sistemas expertos y supuso un salto cualitativo importante, al aplicar con éxito las técnicas de Inteligencia Artificial a problemas del mundo real. CASNET (Casual ASSociational NETwork) desarrollado por Weiss, Kulikowski y Safir ¹¹⁷ de la Rutgers University, es un sistema de ayuda al diagnóstico médico especializado en casos de glaucoma. El programa trata de identificar el origen de la enfermedad para su clasificación y proceder, a continuación, a la prescripción del tratamiento adecuado. En CASNET, la separación de los conocimientos médicos, respecto a los mecanismos de razonamiento que incorpora el programa, facilitan en caso necesario, la actualización de los conocimientos incorporados al sistema. INTERNIST, programa de ayuda al diagnóstico en medicina interna, desarrollado por Pople y Myers ¹¹⁸ en la Universidad de Pittsburgh. El programa provee una lista de posibles enfermedades, derivadas de los datos disponibles del paciente, como por ejemplo; los síntomas que presenta, los datos de laboratorio, historia clínica, etc. dando el diagnóstico probable, pero no recomienda tratamiento. IRIS, fue diseñado por Triboff y Kulikowski ¹¹⁹ de la Rutgers University, como una herramienta para construir y experimentar con representaciones de conocimiento de medicina general, por un lado y posibles tratamientos clínicos por otro. Está diseñado para representar conocimiento médico sobre cualquier dominio y mostrar una variedad de tratamientos clínicos,

116 SHORTLIFFE, E.H. (1.976): "Computer-based medical consultations: MYCIN". En MATE, J. L. y PAZOS, J. (1.988): "Diseño y Construcción de Sistemas Expertos" Sepa S.A. Argentina Córdoba.

117 WEISS, S.M., KULIKOWSKI, C.A., AMAREL, S. y SAFIR (1.978): "A model-based method for computer-aided medical decision-making". En WATERMAN, D. A. (1.986): "A Guide to Expert Systems". Addison-Wesley Massachusetts.

118 POPLER, H.E., MYERS, J.D. y MILLER, R.A. (1.975): "DIALOG: A model of diagnostic logic for internal medicine". En BENCH-CAPON (1.990): "Knowledge Representation: An Approach to Artificial Intelligence" Academic Press London.

119 TRIGOBOFF, M. y KULIKOWSKI, C.A. (1.977): "IRIS: A system for the propagation of inferences in a semantic net". En WATERMAN, D. A. (1.986): "A Guide to Expert Systems". Addison-Wesley Massachusetts.

relacionándolos en su caso. Se ha desarrollado, utilizando IRIS un sistema de ayuda al diagnóstico del glaucoma.

EXPERT, desarrollado por Weiss y Kulikowski ¹²⁰ en 1979, es una herramienta diseñada independientemente de una aplicación concreta y utilizada por los investigadores como ayuda para el desarrollo de sistemas expertos. EXPERT se ha utilizado para la creación de sistemas expertos en distintos campos como, por ejemplo, la industria del automóvil y la industria química. No obstante, ha sido principalmente en medicina y, dentro de esta, en campos como endocrinología y oftalmología donde ha encontrado su máximo desarrollo.

Un ejemplo próximo de sistemas de diagnóstico es el sistema experto LAIDA ¹²¹ desarrollado conjuntamente por los laboratorios LABEIN -Centro de Investigación financiado por el Gobierno del País Vasco- y la empresa eléctrica IBERDUERO. LAIDA analiza los mensajes de alarma generados en la red eléctrica de IBERDUERO operando en tiempo real. Las características principales del sistema son: opera "on-line" junto con el sistema informático ya instalado, la entrada de datos se produce en tiempo real a partir de los datos de la red y, el diagnóstico se produce, también en tiempo real, avisando a tiempo al operador que utiliza el sistema para las oportunas correcciones.

En esta línea el sistema experto SPACE ¹²² desarrollado para el Ministerio de Industria y Energía incluido dentro del Plan Nacional de Calidad Industrial (1.991-1.994). Los objetivos que persigue el sistema son: difundir la importancia de la calidad en la empresa española, proporcionar una herramienta que facilite la propia evaluación del nivel de calidad según las normativas internacionales, europeas y españolas aplicables, destacando las desviaciones con respecto a las mismas, y sugerir por medio del informe

¹²⁰ WEISS, S. M. y KULIKOWSKI, C. A. (1.979): "EXPERT: A system for developing consultation models". En BARR, A. y FEIGENBAUM, E. (1.982): "The handbook of Artificial Intelligence" Volume II William Kaufmann. Los Altos, California

¹²¹ GALLASTEGUI, I., LARESGOITI, I. PEREZ, J. AMANTEGUI, J. y ECHAVARRI, J. (1.991): "Operating experience of an expert system for fault analysis en electrical networks" ACAI-91 Bilbao.

CAMPBELL, J. A. y CUENA, J. (1.989): "Perspectives in Artificial Intelligence. Volume I expert systems: applications and technical foundations" Ellis Horwood London.

¹²² DEL VALLE, M. L., IBAÑEZ, M., MALO, J. L. y PEÑAFIEL, L. F. (1.991): "Sistema Experto para el autodiagnóstico de calidad en la empresa" ACAI-91 Bilbao.

que el sistema emite, acciones correctoras que guíen a la empresa hacia la calidad total. El sistema utiliza CLIPS como herramienta para representar y manejar conocimiento. CLIPS es un entorno para la construcción de sistemas expertos desarrollado por la NASA escrito en lenguaje C.

Por otra parte, en el área de la enseñanza fue temprana la utilización del ordenador, siendo a partir de los primeros años de la década de los sesenta cuando se inicia la utilización de éste en aplicaciones docentes, especialmente en el campo de la enseñanza asistida por ordenador. En su utilización, el alumno puede interactuar con el programa, contestando las preguntas propuestas o haciendo elecciones dentro de las posibilidades que presenta la aplicación, esta se configura así como una herramienta que viene en asistencia y ayuda del profesor. Las técnicas de Inteligencia Artificial aplicadas en este campo, fijan su objetivo en la construcción de programas que incorporen un buen material docente además de una óptima docencia personalizada. Así, con los métodos de Inteligencia Artificial aplicados a la enseñanza asistida por ordenador se persigue el diseño de programas de educación que se acomoden a las necesidades y requerimientos de aprendizaje de cada usuario, creando nuevos y personalizados entornos de trabajo.

En programación convencional, la enseñanza asistida por ordenador responde, en general, a unos mismos esquemas pedagógicos. La materia objeto de docencia se muestra en pantalla, el alumno sigue el texto a través de ésta y responde a las preguntas que se le proponen que, generalmente requieren respuestas cortas. Una vez dada la contestación, el programa indica si la respuesta ha sido correcta o equivocada, en cuyo caso el programa bifurca para mostrar el texto que explique el problema al alumno. El autor del curso, debe así, anticiparse a cada posible respuesta errónea, para prefijar el material que explique el error, orientando este de acuerdo con la causa que el autor considere ha producido el error.

La enseñanza asistida por ordenador basada en el conocimiento ¹²³ -ICAI: Intelligent Computer-Assisted Instruction-, además de mostrar el texto, correspondiente a la materia objeto de docencia, intenta establecer un diálogo con el estudiante para, a través de sus posibles respuestas erróneas, diagnosticar fallos de conocimiento personales e incidir sobre ellos. De este modo, el estudiante, participa activamente en el sistema de instrucción y su interés y sus fallos son los que van guiando el dialogo que se establece con el programa. En este mismo sentido Koffman y Blount ¹²⁴ señalan que *a menudo no es suficiente decir a un estudiante que está equivocado e indicar el método adecuado de solución. Un sistema de ICAI debe poder realizar hipótesis basadas en los errores cometidos por el estudiante para conocer el origen real de los mismos. Para llevar a cabo la tarea de docencia encomendada un sistema ICAI dispone de los siguientes componentes principales: a) el conocimiento que el sistema intenta transmitir al estudiante, "módulo experto", b) información individualizada acerca de lo que el estudiante conoce o desconoce y, c) la presentación al alumno de la materia objeto de aprendizaje "módulo tutor".*

El "módulo experto" tiene asignada la tarea de generar los problemas a resolver por el estudiante y evaluar las soluciones dadas por éste. El desarrollo de esta tarea, requiere incorporar conocimiento del dominio sobre el que trabaja el sistema experto. La representación de ese conocimiento, en los primeros sistemas, se llevó a cabo utilizando redes semánticas, como es el caso de SCHOLAR y WHY, en otros sistemas más tardíos como, WUMPUS y GUIDON la representación se basó en reglas de producción. No hay, sin embargo, unanimidad en la aceptación de la forma más adecuada de representación de conocimiento en los sistemas ICAI. También el "módulo experto", para evaluar el aprendizaje del alumno, establece hipótesis acerca de los conceptos erróneos de éste y genera estrategias para su corrección, que serán llevadas a cabo por el "módulo tutor".

¹²³ CARBONELL, J. R. (1.970): "AI in CAI: An artificial intelligence approach to computer aided instruction". En BARR, A., COHEN, P. R. y FEIGENBAUM, E. (1.989): "**The handbook of Artificial Intelligence**" Volume IV Addison-Wesley Publishing Company Massachusetts.

¹²⁴ KOFFMAN, E. B. y BLOUNT, S. E. (1.975): "Artificial intelligence and automatic programming in CAI ". En BARR, A. y FEIGENBAUM, E. (1.982): "**The handbook of Artificial Intelligence**" Volume II William Kaufmann. Los Altos, California

Esta evaluación se realiza, generalmente, comparando las respuestas del estudiante con aquellas que daría el propio sistema experto sobre el mismo dominio de conocimiento.

El conocimiento sobre las aptitudes del alumno, en orden a ajustar la enseñanza y mejorar rendimientos, forma parte también del sistema y puede ser obtenido a través de varias fuentes ¹²⁵: a) de forma *implícita*, a través de cómo el alumno resuelve los problemas planteados; b) de forma *explícita*, con preguntas directas al estudiante; c) a través de la experiencia declarada por el estudiante en la materia objeto de estudio y, d) basada en la dificultad que ofrece por si misma la materia objeto de docencia. Los primeros programas como, por ejemplo SCHOLAR, usaban solo evidencia *explícita*, mientras que se intentaba incorporar evidencia *implícita* en desarrollos posteriores, teniendo en cuenta la dificultad que supone que el programa sea capaz de reconocer y analizar las estrategias de resolución aplicadas por el estudiante.

El "módulo tutor", se encarga de la comunicación con el alumno. Selecciona los problemas a resolver por este, muestra y critica resultados y provee, en su caso, el material de corrección adecuado. Para realizar esta tarea el "módulo tutor" incorpora conocimiento sobre: métodos de docencia, lenguaje natural y dominio.

Se comentan a continuación las características de algunos de los principales sistemas expertos orientados a la docencia. SCHOLAR ¹²⁶ creado por Carbonell y Collins en 1.970, fue uno de los primeros sistemas desarrollados en este área. El programa se plantea como objetivo, la docencia de la geografía suramericana. El conocimiento sobre el dominio se representa en el programa, mediante redes semánticas, poniéndose un especial énfasis en la investigación sobre lenguaje natural, para establecer la comunicación con el alumno. Con SCHOLAR, se inician los sistemas capaces de responder preguntas no prefijadas y capaces, también, de generar preguntas, con distinto nivel de detalle. Que adaptadas al conocimiento del alumno guían a éste en el estudio de

¹²⁵ GOLDSTEIN (1.977): "The computer as coach: An athletic paradigm for intellectual education". En BARR, A. y FEIGENBAUM, E. (1.982): "The handbook of Artificial Intelligence" Volume II William Kaufmann. Los Altos, California

¹²⁶ CARBONELL, J. R. (1.970): "AI in CAI: An artificial intelligence approach to computer aided instruction". En BARR, A., COHEN, P. R. y FEIGENBAUM, E. (1.989): "The handbook of Artificial Intelligence" Volume IV Addison-Wesley Publishing Company Massachusetts.

la materia determinando, al mismo tiempo, su nivel de conocimientos. SOPHIE ¹²⁷ -a SOPHisticated Instructional Environment- desarrollado por Brown y Burton en 1.975, es un programa que enseña a resolver problemas en el ámbito de un laboratorio de electrónica simulado. El problema a resolver por el estudiante, es el de encontrar los fallos de funcionamiento de una pieza o equipo, valiéndose para ello, de información obtenida a través de una secuencia de medidas, realizando a partir de estas las hipótesis de mal funcionamiento que serán evaluadas por el programa, se intenta, así, conocer la iniciativa de cada estudiante frente a la resolución de problemas.

GUIDON ¹²⁸ desarrollado por Clancey en la Universidad de Stanford, guía al estudiante en un diálogo sobre los síntomas de un paciente del que se sospecha sufre una enfermedad infecciosa, enseñándole a distinguir los datos clínicos y de laboratorio más relevantes y, el empleo de la información, así adquirida, para el diagnóstico de la enfermedad. En GUIDON se utilizan las reglas de MYCIN para el diagnóstico, materia de la docencia del sistema, pero también, GUIDON incorpora conocimiento para, guiar el diálogo con el estudiante, presentar el diagnóstico, ajustar la enseñanza a los conocimientos del estudiante y responder a sus preguntas. Este conocimiento está totalmente separado del contenido en MYCIN.

Con el diseño de GUIDON, se quiere poner de manifiesto el rendimiento del programa MYCIN al ser utilizado, el conocimiento que éste contiene, en un programa de enseñanza asistida por ordenador, permitiendo de esta forma la evaluación por una parte de las reglas contenidas en MYCIN y, de otra parte el conocimiento sobre docencia que es necesario incorporar a MYCIN, para utilizarlo como programa de enseñanza. El resultado es, que las reglas de MYCIN no tiene porque ser modificadas para la nueva aplicación, pero si son utilizadas de forma distinta para guiar, por ejemplo, el diálogo con el estudiante y modelizar el conocimiento de este. GUIDON pone de manifiesto que el

¹²⁷ BROWN, J. S., BURTON, R. R. y BELL, A. G. (1.975): "SOPHIE: A step toward creating a reactive learning environment". En WATERMAN, D. A. (1.986): "A Guide to Expert Systems". Addison-Wesley Massachusetts.

¹²⁸ CLANCEY, W. J. (1.979): "Dialogue management for rule-based tutorials". En WATERMAN, D. A. (1.986): "A Guide to Expert Systems". Addison-Wesley Massachusetts.

conocimiento que incorpora un programa de enseñanza y el conocimiento sobre el dominio de cualquier otro sistema experto, pueden tener las mismas características, es decir: a) pueden codificarse en reglas y, b) pueden adquirir conocimiento a partir de la experiencia como consecuencia de su puesta en práctica.

BUGGY ¹²⁹ desarrollado por Brown, Burton y Larkin en 1.978 es un programa que explica los fallos en aritmética cometidos por un estudiante y, el por qué de éstos. Para ello dispone de un modelo detallado de los conocimientos de cada estudiante, así como, de sus errores conceptuales. La experiencia con BUGGY pone de manifiesto la dificultad que presenta modelizar los errores cometidos por el estudiante enfrentado a una tarea. En este sentido, el programa trabaja incorporando los procedimientos de resolución correctos e incorrectos para una determinada tarea, simulando el comportamiento de un estudiante frente a un problema particular, comparando y poniendo de manifiesto, que parte de la solución es correcta y que parte no lo es. Esta comparación se lleva a cabo descomponiendo la tarea en pasos elementales.

EXCHECK ¹³⁰ desarrollado por Suppes en la Universidad de Stanford, es un programa para la docencia a nivel universitario de lógica matemática, teoría de conjuntos y demostraciones matemáticas. La materia se presenta al estudiante a través del terminal de ordenador, con empleo de ventanas para visualizaciones simultáneas y con voz generada por ordenador. Una de las características principales del programa es, la de producir la comunicación con el alumno en un estilo natural muy próximo a la práctica matemática habitual.

A modo de resumen, el empleo del ordenador y de las técnicas de la Inteligencia Artificial aplicadas a la enseñanza, permiten crear entornos de trabajo, que pongan al alcance del estudiante enseñanza individualizada, de acuerdo con sus propias preferencias y aptitudes, y poder trabajar en un entorno práctico donde el estudiante,

129 BROWN, J. S. y BURTON, R. R. (1.978): "A paradigmatic example of an artificially intelligent instructional system". En BARR, A., COHEN, P. R. y FEIGENBAUM, E. (1.989): "The handbook of Artificial Intelligence" Volume IV Addison-Wesley Publishing Company Massachusetts.

130 SUPPES, P. (1.981): "University-level computer assisted instruction at Stanford: 1968-1980". En BARR, A. y FEIGENBAUM, E. (1.982): "The handbook of Artificial Intelligence" Volume II William Kaufmann Los Altos, California

además de aprender la teoría de la materia correspondiente, puede poner en práctica aquélla, tutorizado por los propios programas con los que está interactuando.

2.4.3 LOS SISTEMAS EXPERTOS EN EL DOMINIO EMPRESA

Los sistemas expertos tienen también un campo de desarrollo muy importante en su aplicación a la actividad empresarial. En los años 80, las empresas comienzan a interesarse y hacer uso de esta tecnología. En la actualidad, hay sistemas construidos y en uso referidos a distintas áreas de la actividad empresarial, principalmente: fabricación, finanzas, marketing y contabilidad. El interés de las empresas en utilizar las técnicas de Inteligencia Artificial y más concretamente los sistemas expertos, obedece a distintas razones. En primer lugar, los sistemas expertos pueden usarse como medio para capturar el conocimiento y la experiencia de un experto humano. De esta forma, los conocimientos que permiten al experto desarrollar de forma eficaz su trabajo en una parcela definida dentro de la empresa, van a poder transmitirse a otros empleados, que tengan que desarrollar las mismas tareas, disponiendo del sistema experto para ese trabajo de formación. Una segunda razón, sería la posibilidad de utilizar un sistema experto en sustitución de un experto humano, en el caso de no ser posible la contratación de éste, siempre que el rendimiento del sistema experto lo hiciera deseable y sus resultados estuvieran suficientemente validados. Otro motivo de interés está en la utilización de un sistema experto como consultor, cuando se deben considerar un número importante de alternativas y sus distintas implicaciones, en un proceso de toma de decisiones. Y por último, su utilización puede significar una mejora tecnológica en la empresa que la permita mantenerse en vanguardia respecto a las tecnologías de la información.

La aplicación de los sistemas expertos a las distintas áreas de la actividad empresarial se muestra en la actualidad en expansión. Se comentan, a continuación, las características de algunos de estos sistemas. El IMS -Intelligent Management System- desarrollado en la Universidad Carnegie-Mellon, con el objetivo de hacer posible un sistema integrado de fabricación, en un entorno constituido por redes de ordenadores,

para llevar a cabo el enlace de las distintas fases del proceso de producción. El sistema provee una *interface* de usuario para acceder al sistema a través de un terminal inteligente conectado a la red de comunicaciones. El empleo del lenguaje natural, para establecer el diálogo con el sistema, es importante a la hora de facilitar al usuario la realización de consultas y requerimientos de operación. IMS está diseñado como un sistema distribuido, para la realización de múltiples actividades en paralelo, en un entorno multiproceso.

ISIS es un sistema experto de planificación de producción desarrollado en 1.982 por Fox¹³¹ en la Universidad Carnegie-Mellon. El objetivo del sistema consiste en generar un plan de fabricación de entre todos los planes posibles, guiado por búsqueda heurística, que evalúa limitaciones asociadas con costes, disponibilidad de maquinaria y posibilidad de puesta en práctica del proceso.

CALLISTO sistema diseñado conjuntamente por Digital Equipment Corporation y la Universidad Carnegie-Mellon, tiene como objetivo principal la gestión de proyectos, teniendo en cuenta las interacciones que tienen lugar en el transcurso de las diferentes fases del ciclo de fabricación. CALLISTO programa las actividades necesarias para la realización de tareas, controla el estado de actividades paralelas, determina si se requieren cambios en el plan establecido para cumplir los objetivos previstos y administra, en su caso, las modificaciones a introducir en el plan. La representación del conocimiento se lleva a cabo, mediante la utilización de marcos de referencia, representación también utilizada por ISIS, y reglas de producción, lo que facilita su empleo tanto por directivos como por diseñadores de proyectos.

Los sistemas expertos aplicados al análisis financiero se pueden agrupar en dos tipos básicos: a) sistemas consultores para la ayuda a la toma de decisiones y, b) sistemas cuyo objetivo final es el de proveer decisiones financieras. La primera categoría de sistemas expertos producen, como resultado de su ejecución, principalmente informes

¹³¹ FOX, M. S., ALLEN, B. y STROHM, G. (1.982) "Job-Shop Scheduling: An Investigation in Constraint-Directed Reasoning". En GEVARTER, W. M.(1.987): "Maquinas Inteligentes. Una panorámica de la Inteligencia Artificial y de la Robótica" Ed. Diaz de Santos S.A. Madrid.

sobre tendencias y ratios además de proporcionar al usuario un informe acerca de las expectativas financieras, obtenidas a partir de datos iniciales, informe similar al que un analista experto produciría. La segunda categoría de programas, incluyen todas las operaciones necesarias para llevar a cabo un completo análisis financiero, partiendo de unos datos iniciales hasta alcanzar una decisión final.

Un ejemplo de sistema experto correspondiente al primer tipo, es el programa ANSWERS, desarrollado por Blocher ¹³² en la Universidad de North Carolina en 1.988. El programa está diseñado para trabajar con información proporcionada por la contabilidad, nómina e inventario. La toma de datos se produce, preferentemente, a partir de los balances o, si se considera necesario, pueden ser requeridos directamente al usuario. Como resultado de ejecución, el programa produce un informe donde se incluyen comentarios y sugerencias sobre la situación estudiada, acompañados por informes gráficos para facilitar su interpretación.

FSA -Financial Statement Analyzer- es un sistema experto para el análisis financiero desarrollado por U.S. Securities and Exchange Commission y Arthur Andersen. FSA es uno de los primeros intentos para aplicar las técnicas de Inteligencia Artificial a la toma de decisiones financieras. En el programa la toma de decisiones se contempla dividida en dos etapas ¹³³. La primera engloba las actividades cognitivas para reconocer la información relevante en la toma de decisiones; la segunda implica un proceso de razonamiento sobre la información anterior para tomar la decisión adecuada. El conocimiento del sistema, además de incorporar datos sobre el tipo de empresa y la evolución de esta, incluye información sobre el entorno como conocimiento necesario para llegar a la toma de decisiones. Se utiliza en FSA técnicas de representación del conocimiento orientadas al objeto y proceso en lenguaje natural para obtener información de los informes financieros anuales, a utilizar en el cálculo de ratios, como paso necesario para llegar a la toma de decisiones.

¹³² BLOCHER, E. y SCALF, K. A. (1.988): "Developing an Expert System for Financial Analysis". En LIEBOWITZ, J. (1.990): "Expert Systems for Business and Management". Prentice-Hall. New Jersey.

¹³³ SIMON, H.A.(1.982): "La nueva ciencia de la decisión gerencial". El Ateneo Buenos Aires.

Por lo que se refiere al área de ventas y marketing, la mayor parte de los sistemas expertos comercializados, se han diseñado para empresas que ofrecen servicios financieros; bancos y empresas de seguros. Hay razones que explican esta orientación, entre ellas: las características de los servicios que estas empresas ofertan con alto contenido en información; el incremento de servicios ofertados por estas empresas, principalmente bancos, a sus clientes; la competencia entre las mismas y la sofisticación cada vez mayor de los productos y servicios ofertados. Los sistemas expertos aplicados a este ámbito deben ayudar a analizar las necesidades financieras del cliente, aconsejarle acerca de los productos más adecuados y dirigir las ventas hacia aquellos que el cliente necesita. APEX -Applied Expert Systems- es una herramienta, para la construcción de sistemas expertos que obedece a las características anteriormente mencionadas.

Los sistemas expertos desarrollados con APEX requieren, como punto de partida para efectuar el análisis financiero, una colección de datos en los que se incluyen, además de información básica acerca del cliente como, nombre, D.N.I., nómina, volumen de gastos etc., tomada en su caso de la Base de Datos de la entidad, información acerca de los objetivos perseguidos por el cliente, con indicación de las prioridades en la consecución de los mismos y el riesgo que el cliente está dispuesto a asumir. Esta información dirige el razonamiento del sistema y estructura los informes de salida individualizándolos. La salida del sistema se configura, por tanto, como un conjunto de informes, ajustados a la situación financiera del cliente y en los que se emplea el lenguaje natural para facilitar su comprensión.

Por lo que se refiere a los sistemas expertos aplicados a la contabilidad destacan, por número de sistemas e interés de los mismos, aquellos orientados al ámbito de la auditoría. El diseño de los primeros prototipos de sistemas expertos en auditoría, se llevó a cabo principalmente en universidades y centros de investigación dependientes de las mismas, pero en la actualidad, son las empresas que se desenvuelven en el ámbito de la auditoría quienes comercializan y ponen en funcionamiento sus propios productos. Este interés, por aplicar las técnicas de la Inteligencia Artificial a los problemas de auditoría,

responde a varias causas, y específicamente la referida al entorno altamente complejo en el que se desenvuelven los problemas en auditoría. Presentando, por lo general, varias posibles soluciones y planteando dificultades al auditor la selección de la mejor de entre ellas. Son problemas de éstas características, aquellos que, con frecuencia, se resuelven más adecuadamente aplicando conocimiento heurístico para una buena aproximación en la búsqueda de la mejor solución, aunque ésta no sea, la solución óptima. También la posibilidad de comisión de errores de inconsistencia, en este ámbito, es alta. Se producen éstos cuando diferentes auditores consultados, partiendo de igual información financiera y, de un conjunto idéntico de datos, producen diferentes opiniones. La aplicación de un sistema experto en el proceso de toma de decisiones en auditoría, puede conferir a éste, consistencia y fiabilidad permitiendo, además, registrar el proceso seguido en la resolución del problema. De este modo se obtiene una documentación completa del mismo a utilizar para posteriores consultas o docencia.

AUDITOR, desarrollado en 1.983 por Dungan de la Universidad de Illinois, ¹³⁴ es uno de los primeros sistemas expertos orientados a la auditoría y, construido para analizar los problemas de créditos fallidos analizando las cuentas de los clientes individualmente. La representación del conocimiento en AUDITOR, se lleva a cabo mediante reglas de producción, utilizando para el proceso la concha AL/X. El programa informa, en términos de probabilidad, sobre las cuentas de incobrables que figuran en balance.

ICE -Internal Control Evaluation- desarrollado en 1.984 por Kelly de la Universidad de Pittsburg ¹³⁵. Es un prototipo creado como herramienta de ayuda en la planificación de los procesos de auditoría. En ICE el conocimiento se organiza jerárquicamente en distintos niveles. El primer nivel incluye conocimiento de la industria y entorno económico. El segundo nivel, centra su atención en el cliente con información acerca de la gestión empresarial y los procedimientos contables seguidos. El tercer nivel,

134 DUNGAN, C. (1.983): "A model of an Audit Judgment in the Form of an Expert System". En PAU, L. F., MOTIWALLA, J. PAO, Y. H. y TEH, H. H. (eds.) (1.989): "Expert Systems in Economics Banking and Management" Elsevier Science Publishing Amsterdam.

135 KELLY, K. P. (1.984): "Expert Problem Solving for the Audit Planning Process". En LIEBOWITZ, J. (1.990): "Expert Systems for Business and Management". Prentice-Hall. New Jersey.

se centra en el control de las operaciones. ICE está programado en LISP y la representación del conocimiento se lleva a cabo mediante reglas de producción y marcos.

TICOM -The Internal Control Model- desarrollado en 1.985 por Bailey, ¹³⁶ es una herramienta de ayuda al auditor para diseñar y evaluar un modelo de control interno. Recientemente en 1.988, aparece una nueva versión TICOM-IV que permite al auditor el diseño, análisis y evaluación de un modelo de control interno. El sistema está programado en lenguaje PROLOG.

Son numerosos también los sistemas expertos en auditoría desarrollados y comercializados por empresas del sector. Un ejemplo de ello, son los programas AY/ASQ diseñado para trabajar en el entorno Apple Macintosh y, CFILE -Credit FILE analysis- sistema basado en reglas y diseñado para la evaluación de préstamos bancarios.

En resumen, puede señalarse que los sistemas expertos en auditoría, se configuran como una herramienta válida para a) simular los procedimientos empleados por un auditor y, b) contrastar el conocimiento y las técnicas de Inteligencia Artificial necesarias para llevar a cabo un proceso de auditoría.

2.4.4 TIPOLOGIA BASICA

El conocimiento incorporado a un sistema experto, le proporciona una alta efectividad en la resolución de problemas que pertenecen al ámbito de su aplicación. Esta efectividad ha generado un número importante de aplicaciones, la mayoría de las cuales se pueden agrupar entorno a los siguientes tipos:

- a) Sistemas que predicen, con inferencia de consecuencias probables a partir de situaciones dadas. Un sistema de predicción modeliza una situación, para utilizar las conclusiones obtenidas del modelo como base de sus predicciones. Pertenecen a esta clase de sistemas aquellos que llevan a cabo predicciones meteorológicas, demográficas, evolución del tráfico, etc.

¹³⁶ BAILEY, A. D. et. al. (1.985): "TICOM and the Analysis of Internal Controls". En PAU, L. F., MOTIWALLA, J. PAO, Y. H. y TEH, H. H. (eds.) (1.989): "Expert Systems in Economics Banking and Management" Elsevier Science Publishing Amsterdam.

- b) Sistemas que diagnostican, infiriendo un mal funcionamiento del sistema a partir de sensores. A esta categoría pertenecen, sistemas médicos, electrónicos y mecánicos entre otros. La misión básica de los sistemas de diagnóstico es, detectar las irregularidades de funcionamiento observadas, con indicación de sus causas. Uno de los métodos posibles para llevar a cabo el diagnóstico, es usar una tabla de asociaciones entre el comportamiento del sistema y el correspondiente diagnóstico.
- c) Sistemas que diseñan, configurando objetos que satisfacen las restricciones del diseño. Se intenta además, minimizar una función objetivo que incluye, entre otras variables, los costes necesarios para la consecución de los objetivos. Sistemas de este tipo son los dedicados a la elaboración de presupuestos y a la construcción de variedad de diseños, como por ejemplo, la construcción de circuitos eléctricos.
- d) Sistemas que planifican, dando lugar al diseño de acciones. Estos sistemas están especializados en el diseño de objetos que pueden realizar ciertas funciones. Pertenecen a esta categoría sistemas de, programación automática, robótica, planificación de proyectos militares y comunicaciones. Los sistemas de planificación utilizan modelos de comportamiento para inferir los efectos de las actividades planeadas del sistema.
- e) Sistemas que supervisan, comparando la conducta observada del sistema en aspectos característicos, con aquella que debería ser, para tener éxito en la consecución del plan previsto. Los aspectos característicos, sobre los que se realiza la supervisión, son puntos vulnerables del plan que se corresponden con posibles fallos del mismo. Generalmente la supervisión se lleva a cabo sobre aspectos clave del sistema, cuya no correcta ejecución puede producir, bien la anulación de todo el plan, bien la violación de alguna de las restricciones establecidas en éste. Ejemplos de

sistemas de supervisión, son los dedicados a controles de tráfico aéreo, centrales nucleares, etc.

- f) **Sistemas que depuran.** Se basan en la planificación, diseño y capacidad de predicción de los sistemas, para crear especificaciones o recomendaciones dirigidas a corregir un problema diagnosticado. Los programas que ayudan a encontrar y corregir defectos de funcionamiento en sistemas, suelen estar constituidos por una Base de Conocimiento y un editor de texto y, están dirigidos fundamentalmente a la depuración de programas de ordenador. No se pueden considerar como verdaderos sistemas expertos.
- g) **Sistemas que reparan.** Desarrollan y ejecutan planes para dar solución a problemas diagnosticados. Estos programas incorporan características de los sistemas de depuración y planificación y, se han desarrollado principalmente en el ámbito del mantenimiento de ordenadores.
- h) **Sistemas que enseñan.** Diagnostican y corrigen errores de aprendizaje. Básicamente, estos sistemas comienzan por construir una hipotética descripción del conocimiento que tiene el estudiante sobre una determinada materia, para así, interpretar su comportamiento y diagnosticar, más tarde, sus fallos de conocimiento. Finalmente el sistema planea una acción tutorial que provee el remedio para corregir los fallos de conocimiento del alumno.
- i) **Sistemas que controlan, interpretan, reparan y supervisan** el comportamiento de un sistema. Para realizar estas funciones, el sistema de control, debe sucesivamente, interpretar la situación actual del sistema, predecir el futuro, diagnosticar y anticipar los problemas, formular un plan para su solución y supervisar su ejecución para asegurar el éxito. Problemas que caen bajo el ámbito de aplicación de estos sistemas son, por ejemplo, los relacionados con el control del tráfico aéreo y la gestión empresarial. Los sistemas expertos podrían dar solución a problemas de control que resisten un tratamiento estrictamente matemático.

2.5 SISTEMAS EXPERTOS BASADOS EN REGLAS

2.5.1 LA RESOLUCION INTELIGENTE DE PROBLEMAS

La resolución inteligente de problemas tiene como idea central, la construcción de la mejor solución, de forma selectiva y eficiente, de entre el espacio de alternativas de solución posibles. Cuando los recursos son limitados, se requiere una búsqueda selectiva en el espacio de soluciones posibles, minimizando búsquedas infructuosas. El conocimiento incorporado al sistema permite, disponer de información adecuada y conocer los procedimientos correctos para su utilización, en orden a construir la mejor solución evitando aquellas no viables.

Para un sistema que pretenda incorporar el conocimiento y la inteligencia de un experto, en la resolución de problemas en un dominio, hay dos procesos básicos a los que prestar especial atención: una apropiada representación simbólica del conocimiento y, el empleo de búsqueda heurística para la correcta resolución del problema.

Por lo que respecta a la representación simbólica del conocimiento, Newell y Simon ¹³⁷ definen un símbolo, como un modelo físico de un hecho, que forma parte de una estructura simbólica, compuesta por una secuencia de símbolos relacionados. Se puede así concebir un símbolo como una cadena de caracteres, y las estructuras simbólicas como tipos de estructuras de datos que pueden denominarse estructuras en lista. El desarrollo de los primeros lenguajes para procesos de listas orientados al tratamiento de información simbólica, fue una de las primeras aportaciones de la investigación en Inteligencia Artificial.

Por otra parte, si se examinan los métodos de resolución de problemas aplicados en los desarrollos de Inteligencia Artificial, encontramos que muchos de ellos utilizan procedimientos de búsqueda de solución basados en métodos de prueba-error. En estos métodos, la descripción de la solución que se pretende, se denomina objetivo y el conjunto de etapas que llevan desde las condiciones iniciales del problema, hasta la

¹³⁷ NEWELL, A. y SIMON, H.A. (1.976): "Computer Science as empirical enquiry: Symbols and search". En SIMON, H. A. (1.982): "La Nueva Ciencia de la Decisión Gerencial" El Ateneo Buenos Aires.

consecución de objetivos, se denomina espacio de búsqueda de la solución. Se trata de encontrar la solución que permita alcanzar el objetivo, de entre el espacio de soluciones posibles. Son varios los métodos a aplicar para la búsqueda de la solución, la búsqueda exhaustiva es uno de ellos, en la que se exploran sistemáticamente todas las posibles etapas hasta alcanzar la solución. Como consecuencia, a medida que aumenta la complejidad del problema, aumenta el espacio de soluciones posibles hasta, en ocasiones, hacer inviable el procedimiento. En muchas aplicaciones, es posible incorporar conocimiento específico sobre el dominio, para guiar el proceso de búsqueda y reducir tiempos de ejecución. La información incorporada al sistema con este fin, se denomina información heurística y los procedimientos de búsqueda que utilizan esta información, se denominan procedimientos de búsqueda heurísticos. La búsqueda heurística pondrá fin a la ejecución del programa tan pronto se alcance una solución satisfactoria.

2.5.2 LAS PRIMITIVAS SEMANTICAS

Un formalismo de representación de conocimiento, como es el caso de las reglas de producción, es un modo de expresión del conocimiento en un dominio. Una vez decidido el formalismo de representación adecuado, surge otra cuestión no menos importante en el diseño de un sistema experto: la relativa al vocabulario a emplear en el formalismo seleccionado. La investigación en *primitivas semánticas* incide en el problema de establecer un vocabulario representacional.

El término *primitiva semántica*, no tiene una única definición. Para algunos autores, se puede pensar en una primitiva como, en cualquiera de los símbolos, que es usado pero no definido dentro de un sistema. Los trabajos desarrollados por Wilks ¹³⁸ se ajusta a este enfoque. Una segunda interpretación, configura las *primitivas semánticas* como, elementos de conocimiento mediante los cuales se representan las palabras y

¹³⁸ WILKS, Y. (1.975): "An Intelligence analyzer and understanding of English". En BARR, A., COHEN, P. R. y FEIGENBAUM, E. (1.989): "The handbook of Artificial Intelligence" Volume IV Addison-Wesley Publishing Company Massachusetts.

sentencias que constituyen el conocimiento del sistema. Ejemplos de ésta interpretación son los trabajos desarrollados por lingüistas y psicólogos, como Jackendorff ¹³⁹ y Miller ¹⁴⁰.

Una descripción semántica con utilización de primitivas es, para Wilks, *un reducido micro-lenguaje, con todas las ambigüedades del lenguaje natural*. Su empleo permite, sin embargo, *el establecer hipótesis sobre significados, en los sistemas de proceso del lenguaje natural*. Wilks inició en 1.968 el desarrollo de un sistema en lenguaje natural para traducción automática. El sistema acepta párrafos en inglés, produciendo a partir de ellos una representación interna que se corresponde con una estructura de datos compuesta por nodos, representados éstos, por *primitivas semánticas*. A partir de esta estructura, se genera una traducción en francés del texto original. Esta traducción sirve como comprobación de la comprensión del lenguaje natural original, el inglés en el trabajo comentado, e implica directamente la validez de la representación interna utilizada.

Uno de los mayores problemas de la representación, es la ambigüedad de muchos de los términos que se emplean en el lenguaje. En el trabajo de Wilks el empleo de "preferencias semánticas" intenta dar solución al problema. Se trata de emplear conocimiento sobre los posibles significados de las palabras, para deshacer ambigüedades. El sistema propuesto por Wilks requiere de un diccionario para distinguir entre los distintos significados que pueden asociarse a las palabras del texto a procesar. Para las definiciones en el diccionario se utiliza un vocabulario de *primitivas semánticas*. La elección del conjunto de primitivas debe, según Wilks, ajustarse a las siguientes características:

139 JACKENDOFF, R. (1.972): "Semantic interpretation in generative grammar" MIT Cambridge Massachusetts.

140 MILLER, G. A. y JOHNSON-LAIRD, P. N. (1.976): "Language and perception". En BARR, A. y FEIGENBAUM, E. (1.982): "The handbook of Artificial Intelligence" Volume I William Kaufmann Los Altos, California

- a) **Finitud:** El número de primitivas debe ser finito y debe ser más pequeño que el número de palabras cuyo significado se incorpora en el esquema de representación.
- b) **Comprensibilidad:** El conjunto de primitivas debe ser el adecuado para expresar y distinguir de entre los posibles significados de una palabra, aquellos que deben ser representados. La definición mediante primitivas, de los diferentes significados de una palabra, no tiene porque ser exhaustivo.
- c) **Independencia:** Una primitiva no debe definirse en función de otras primitivas.
- d) **No circularidad:** Dos primitivas no pueden definirse una en función de la otra.
- e) **Atomicidad:** Un subconjunto de primitivas no pueden reemplazarse por un conjunto más pequeño, no se pueden reducir.

Las anteriores propiedades, son de aplicación al conjunto de primitivas que recogen el significado de las palabras del texto a procesar. Los hechos se expresan mediante un conjunto de reglas de inferencia y separados del conjunto de primitivas. Son en último término, las reglas de inferencia las que permitirán deshacer ambigüedades.

Otro enfoque en la utilización de *primitivas semánticas* es el dado por Schank ¹⁴¹, para quién un conjunto de primitivas debe permitir la representación de cualquier acción. Presentan, por tanto, la característica de independencia respecto de la tarea a representar. Los requisitos que debe reunir una representación para capturar la estructura conceptual del lenguaje se centran en la no ambigüedad de la representación, a pesar que el lenguaje presente ambigüedades tanto de tipo sintáctico como semántico y en la unicidad, esto es, una única representación para contenidos conceptuales iguales, aunque su formulación se lleve a cabo mediante expresiones diferentes. Schank, se ocupa especialmente en su trabajo, de las sentencias que, con una construcción léxica distinta, poseen el mismo significado.

¹⁴¹ SCHANK, R. C. (1.975): "Conceptual Information Processing" North-Holland Amsterdam.

En orden a lograr representaciones que cumplan los requisitos de unicidad y no ambigüedad, Schank agrupa las primitivas en categorías y utiliza reglas de inferencia para indicar como los elementos de las diferentes categorías se combinan en la representación del conocimiento. Una combinación básica, es aquella que involucra un objeto y la descripción de su estado: atributos del objeto. Las primitivas, en el trabajo de Schank, no representan palabras sino conceptos y reflejan un cierto nivel de comprensión del lenguaje. Consecuentemente, la representación de un texto mediante conceptos, se lleva a cabo con la utilización de un lenguaje libre que explicita la información contenida en una sentencia. El empleo de inferencias es básico en el proceso, asociando a cada primitiva un conjunto de inferencias.

Para Schank, la representación basada en primitivas, unidas a un conjunto de estados posibles de los objetos, permite representar el significado de cualquier expresión. Sin embargo algunos aspectos de las situaciones a representar, como el caso de las descripciones físicas de una acción, no quedaban suficientemente explicadas por el mecanismo de representación propuesto. Revisiones sobre el sistema de representación, han incorporando nuevas categorías de primitivas que amplían el marco anterior, dando cabida, por ejemplo, a aquellas acciones que tienen una cierta implicación social: AUTORIZAR, ORDENAR, PEDIR, etc.

2.5.3 LA FORMALIZACION DEL CONOCIMIENTO.

Un sistema basado en el conocimiento, pretende transformar la línea de pensamiento de un experto humano en un conjunto de formalismos simbólicos capaces de recoger el conocimiento del experto. La transferencia del conocimiento de un experto humano a una máquina es una tarea de diseño del conocimiento y requiere de adecuadas técnicas, en orden a establecer una "continuidad cognitiva" entre el individuo y la máquina. El nivel de abstracción del conocimiento incorporado al sistema debe ser similar al que utilizan los humanos en la resolución de problemas, un nivel en el que se incluyen no solo hechos o deducciones, sino también intuiciones.

Al considerar la destreza individual en la búsqueda de solución de un problema, hay varias dimensiones que intervienen en la tarea. Por ejemplo, hay un componente situación -características del problema planteado- y un componente temporal -tiempo requerido para la resolución-. Hay también, un componente de interacción -los otros agentes que participan en la tarea-, un componente de deducción -las reglas en uso- y, un componente de inducción -intuición, nuevos datos, etc..-. Todas estas dimensiones deben tenerse en cuenta al estudiar y construir un sistema experto.

Por otra parte, el análisis del proceso de razonamiento seguido en la resolución de un problema, sitúa al experto en una primera fase del razonamiento, tiempo "t", concentrado en el estudio del problema, constituyendo éste su foco de atención específico. En una fase posterior, y como resultado de la aplicación de conocimiento -deducción, sentido común- del propio experto y, en su caso, aportación de nueva información del entorno, podemos considerar al experto en el tiempo "t+1", donde ha variado su estado de conocimiento, respecto a la solución del problema planteado. La variación producida es consecuencia de la aplicación de conocimiento a la resolución del problema. En el proceso de razonamiento el experto utiliza nueva información o sus propias deducciones, con el objetivo de alcanzar la solución. Para que un máquina sea capaz de este comportamiento, se requiere una formalización del proceso de razonamiento del experto.

Para la formalización hay que tener en cuenta dos aspectos principales:

- a) Los distintos estados del conocimiento en el proceso de resolución de un problema están relacionados entre sí y esa relación es de inferencia. El experto, en el proceso de razonamiento, evalúa un conjunto restringido de elementos de conocimiento donde se entrecruzan datos del contexto, condiciones del problema y posibles acciones. Este conjunto es el marco de las diferentes alternativas posibles de razonamiento. La evaluación de las alternativas relevantes constituye el siguiente paso en el proceso de razonamiento.

- b) Las distintas etapas del proceso de razonamiento, no son independientes del entorno en el que sitúan. Cada una, representa una situación en la cual la información que llega del exterior es interiorizada y se le da una respuesta o interpretación por parte del experto. De este modo, si queremos construir un modelo procesable por un ordenador, necesitamos tener una representación del entorno en relación con el razonamiento.

Los formalismos de representación basados en reglas, permiten capturar y formalizar los procesos de razonamiento seguidos en la resolución de un problema. Las reglas se representan como estructuras simbólicas, mediante las cuales se puede representar una situación y sus inmediatas consecuencias. En una regla se pueden establecer un conjunto de condiciones que, en caso de cumplirse, conducen a conclusiones que permitirían el establecimiento de hipótesis u objetivos a tener en cuenta en otras reglas, así como la realización de un conjunto de acciones resultado de la aplicación de la regla. Esta facilidad de las reglas para la representación de situaciones concretas de conocimiento, unida a las operaciones de concatenación de las reglas en su ejecución, permiten establecer una relación entre el proceso de reglas y los procesos de razonamiento seguidos por un experto para la resolución de problemas.

En una primera aproximación, se considera que cada una de las condiciones establecidas en una regla -y que como resultado de su evaluación, pueden tomar los valores de "verdadero, "falso" o "desconocido"- son todas ellas condiciones verificadas o "verdaderas". En este punto, la regla de referencia puede ser aplicada o "disparada". Al verificarse que todas las condiciones de la regla son verdaderas, se provoca un proceso de inferencia en el que se asumen como "verdaderas" las hipótesis que dependían de aquellas e integradas en la parte de acción de la regla. Estas hipótesis podrían formar parte del conjunto de condiciones de otra de las reglas del esquema de representación, de tal modo que la verificación de las hipótesis anteriores, condiciones en la nueva regla, podrían ser un paso previo en el proceso de activación de esta última.

El nombre que se asigna a este proceso es el de razonamiento deductivo y en él la ejecución de las reglas se lleva a cabo mediante el encadenamiento hacia atrás de las mismas, con la posibilidad de diferentes niveles de profundidad.

Por otra parte, la regla activada también incluye, de forma general, acciones, llamadas así porque son inductoras de cambios en el conjunto del sistema. Algunas de estas acciones pueden afectar a los datos incorporados al sistema -modificándolos- o bien pueden ordenar la ejecución de programas externos al sistema o afectar a la activación de otras reglas. Cuando una regla cambia los valores de los datos que intervienen en las condiciones de otras reglas, está posibilitando la activación de las reglas afectadas. Este tipo de enlace entre reglas se denomina encadenamiento hacia adelante y se corresponde con un razonamiento de tipo inductivo.

Los sistemas de representación basados en reglas, admiten la integración de ambos tipos de mecanismos de inferencia para facilitar el proceso. Además, el sistema puede adaptarse a los cambios producidos en el entorno, propiciando una exploración selectiva de las reglas evitando, así, costosas búsquedas exhaustivas. Todo ello consecuencia de que la activación en el tiempo, de una determinada regla depende, tanto del proceso actual de razonamiento como de acontecimientos que signifiquen cambios derivados, bien de información externa, bien resultado de una acción.

La *calidad* del conocimiento es crítica en el rendimiento del sistema. Los datos demasiado generales, por ejemplo, aquellos presentes en numerosas reglas que presentan condiciones similares, inducen al sistema a búsquedas más extensivas, mientras que los datos que pueden asociarse a condiciones específicas, ayudan a limitar el alcance de la búsqueda, actuando como discriminantes. En la Figura 6 se muestra como las condiciones actúan como canales de comunicación entre el conjunto de reglas afectadas y cada hipótesis. Los círculos en blanco representan hipótesis, los círculos oscuros son datos, cada línea recta representa una condición en una regla, los arcos unen condiciones dentro de la regla.

A partir de una hipótesis inicial, el sistema determina dos nuevas hipótesis potenciales. Uno de los datos comunes a la hipótesis inicial y a una de las nuevas hipótesis potenciales no verifica la condición establecida en la regla de la que depende dicha hipótesis. Por lo que ésta no se consideraría en este punto. La siguiente hipótesis considerada sería aquella para la que los datos relevantes sí verifican sus condiciones. De este modo, se vincula el grado de especificidad de los datos contenidos en la definición del problema en curso, con el rendimiento en la búsqueda de la solución.

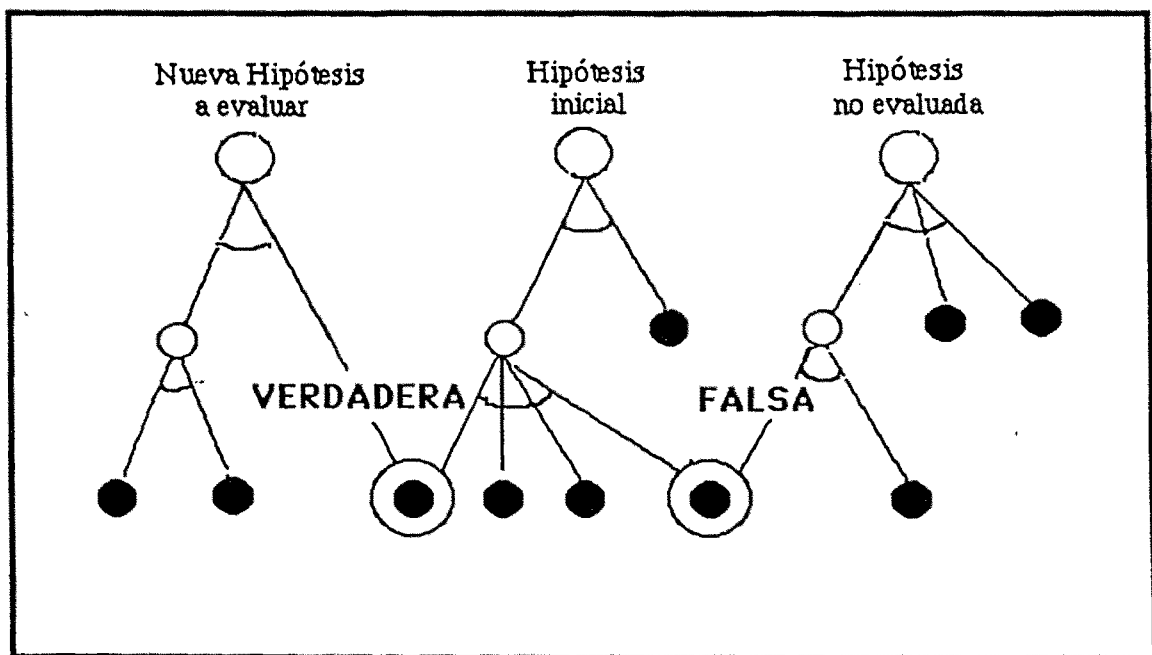


Fig. 6 Enlace entre Reglas

Otro posible nivel de inferencia, que admite su inclusión en el modelo, es aquél que recoge el conocimiento práctico del experto, y que permite establecer relaciones entre grupos de reglas en orden a guiar el proceso de razonamiento.

El conocimiento contenido en las reglas, puede completarse con información externa, información que hace referencia al contexto y que proviene del mundo exterior al modelo. Esta información, se considera relevante, puesto que puede, en su caso, alterar los procesos de razonamiento en curso. Conocimiento acerca de objetos, propiedades de los objetos y sus estados, es conocimiento complementario a las reglas y, generalmente, admite otros modelos de representación, como por ejemplo, los formalismos en red semántica o los "frames".

En el supuesto de una red semántica, los nodos quedan constituidos por conocimiento acerca de los objetos. Definiéndose sobre éstos propiedades que los caracterizan. Así mismo, los objetos pueden agruparse en clases, configurándose una clase como una colección de objetos que tienen propiedades comunes. La clase puede contener información relevante sobre los objetos que la componen y los objetos pueden heredar las propiedades de la clase a la que pertenecen. Clases, objetos y propiedades estructuran un esquema de representación del dominio del problema, registrando conocimiento estático, frente al conocimiento incorporado a las reglas, que puede considerarse, como conocimiento dinámico y, que se relaciona con las inferencias del experto.

2.5.4 EL PROBLEMA DE LA REPRESENTACION EN LA BUSQUEDA DE LA SOLUCION.

La resolución de problemas requiere:

- a) Conocimiento acerca del dominio del problema y del propio problema a resolver.
- b) Conocimiento acerca del razonamiento necesario para su resolución.
- c) Una estrategia de control para llevar a cabo la resolución.

El conocimiento sobre el dominio sitúa el razonamiento dentro de los límites del problema a resolver.

El razonamiento del sistema, se materializa en un conjunto de operadores que aumentan su eficacia al convertirse en operadores muy especializados en el dominio, capaces de generar nuevas aserciones a partir de las ya existentes.

La estrategia de control, establece qué hacer en el siguiente paso y, en particular, qué operador aplicar y en qué orden. Normalmente, el control sobre cómo usar los recursos, está centralizado y forma una estructura aparte dentro del sistema. Aunque, en ocasiones, cierto control puede estar contenido en el razonamiento formando parte del propio conjunto de operadores.

La elección de una determinada estrategia de control puede afectar al contenido y organización del conocimiento. Un tipo usual de estrategia de control es aquella que intenta alcanzar el objetivo mediante la aplicación de una apropiada secuencia de operadores a un estado inicial de la tarea a resolver. Cada aplicación provoca una modificación de la situación anterior. El sistema puede evaluar el resultado de la aplicación de secuencias alternativas de operadores, en orden a elegir aquella que ofrezca las mejores expectativas para la resolución.

El encadenamiento de operadores que supone -la estrategia de control-, se corresponde con dos tipos básicos: a) encadenamiento hacia adelante; se trata de, a partir de una situación inicial, alcanzar aquella que satisfaga las condiciones del problema, b) encadenamiento hacia atrás; los operadores se aplican, en este caso, al objetivo, reduciendo éste a subobjetivos -que se suponen más simples- y cuya resolución permitirá alcanzar la solución del problema inicial propuesto. La reducción continuará hasta que los subobjetivos se acepten como problemas triviales o se alcance su solución.

En el encadenamiento hacia adelante, el sistema emplea razonamiento inductivo y cada nueva situación del problema, obtenida de la anterior mediante la aplicación de los operadores correspondientes, forma parte del espacio de búsqueda de la solución. La representación del espacio de búsqueda emplea dos clases de entidades: estados y operadores. Un estado, es una estructura de datos que contiene una instantánea de la situación del problema en cada etapa de su solución; los operadores transforman el

problema de un estado a otro. Se trata, por tanto, de encontrar una secuencia de operadores que puedan ser aplicados, para alcanzar, a partir de un estado inicial, el objetivo final propuesto. Como consecuencia, la representación del espacio de búsqueda de la solución, debe especificar ¹⁴² : a) la descripción de los estados del problema, en particular la descripción del estado inicial, b) el conjunto de operadores y sus efectos sobre los estados anteriores y, c) la descripción de las características del estado objetivo.

Los diferentes estados de un problema, que forman el espacio de búsqueda de la solución, admiten representación gráfica, representación que facilita su comprensión y diseño. Las estructuras jerárquicas en árbol, permiten una representación gráfica adecuada del conjunto de los estados del problema. En el árbol, el nodo raíz representa el estado inicial, cada uno de los nuevos estados, producidos como consecuencia de la aplicación de un operador, se representa como un nodo sucesor del nodo raíz. La aplicación de nuevos operadores, produce sucesores de aquéllos. Cada aplicación de un operador se representa en el árbol, mediante un arco que une un nodo a su sucesor.

La Figura 7 muestra un ejemplo de representación gráfica de un espacio simple de búsqueda. En ella, un estado inicial, nodo raíz, al que se puede aplicar el operador A o B, se transforma, como consecuencia, en el estado 1 o 2, nodos sucesores, que a su vez admiten la aplicación de nuevos operadores hasta alcanzar el objetivo final.

En el encadenamiento hacia atrás, razonamiento deductivo, tomando como punto de partida el problema a resolver, se establece una sucesión de subproblemas, que reducirá finalmente, el problema original a un conjunto de problemas elementales. El empleo de operadores, permite llevar a cabo la transformación del problema en subproblemas específicos.

¹⁴² NILSSON, N. J. (1.971): "Problem-Solving Methods in Artificial Intelligence". MacGraw-Hill New York

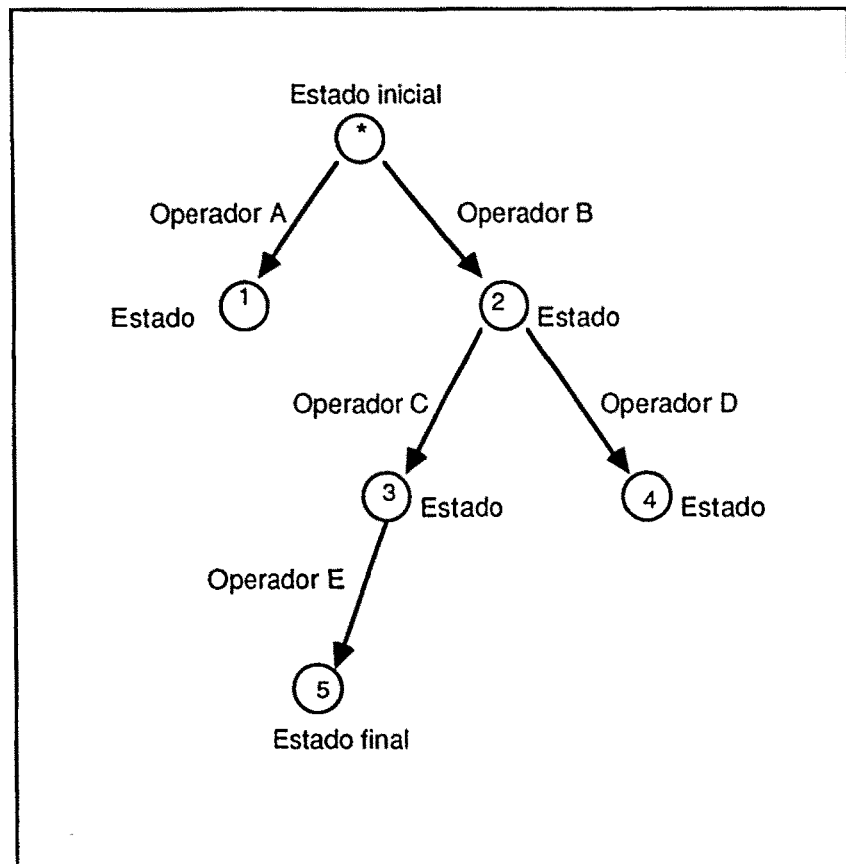


Fig. 7 Gráfico de búsqueda.

La representación del proceso de razonamiento que permite pasar de la descripción de un problema inicial a un conjunto de subproblemas de solución inmediata, precisa de operadores que apliquen la secuencia de transformaciones necesarias. Un operador transforma un problema en numerosos subproblemas que deben tener una solución, siendo posible la aplicación de más de un operador a un único problema. La representación del proceso de razonamiento queda, por tanto, definida por tres elementos básicos: a) la descripción del problema inicial, b) el conjunto de operadores que transforman el problema en subproblemas y, c) la descripción de un conjunto de problemas elementales. Estos elementos admiten representación gráfica, denominándose gráficos AND/OR aquellos, utilizados en su representación.

143:

La elaboración de los gráficos AND/OR, debe ajustarse a un conjunto de reglas

- a) Cada nodo representa un único problema o un conjunto de problemas a resolver. El gráfico contiene un nodo inicial, correspondiente al problema original.
- b) Un nodo que representa un problema elemental, se denomina *nodo terminal* y, no tiene descendientes.
- c) Cada aplicación de un operador a un problema lo transforma en un conjunto de subproblemas, existiendo un arco que enlaza el problema al conjunto de subproblemas obtenido. La Figura 8 muestra la reducción del problema P, en tres subproblemas A, B, y C. De modo que, P será resuelto si uno de los subproblemas A, B, y C son resueltos, A, B, y C se denominan nodos OR.

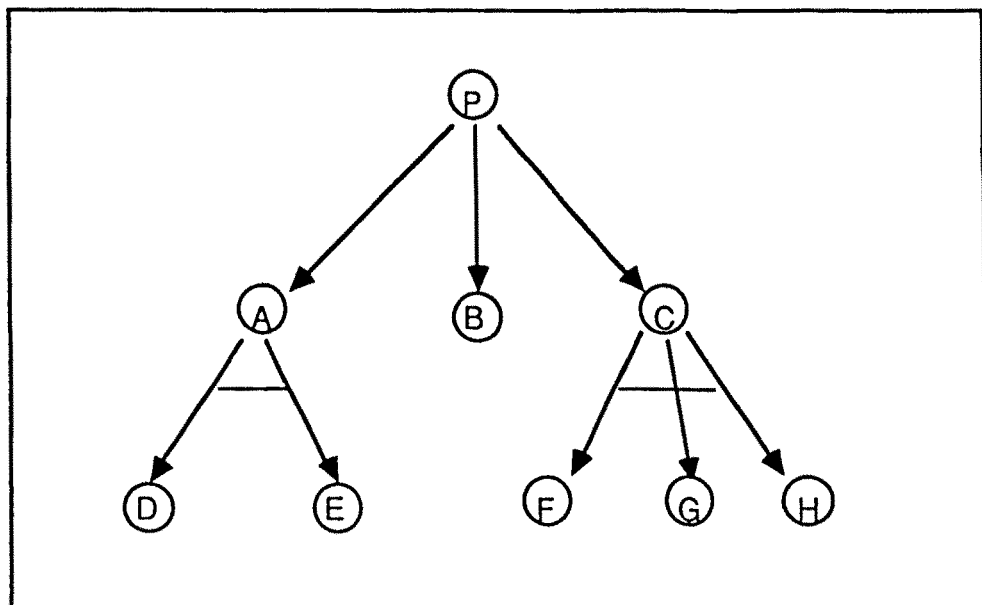


Fig. 8 Gráfico AND/OR.

143 NILSSON, N. J. (1.971): "Problem-Solving Methods in Artificial Intelligence". MacGraw-Hill New York

- d) Por cada nodo que representa un conjunto de dos o más subproblemas, existen arcos que unen el nodo, con el conjunto de nodos que individualizan cada subproblema. El conjunto de subproblemas tiene solución, si todos sus miembros pueden resolverse. Los nodos que representan un conjunto de subproblemas se denominan nodos AND y, para distinguirlos de los nodos OR, los arcos que unen los nodos sucesores AND, a un nodo común, se enlazan mediante una línea horizontal. En la Figura 8 los nodos F, G, y H, son nodos AND y todos ellos deben tener solución, para resolver el subproblema C.

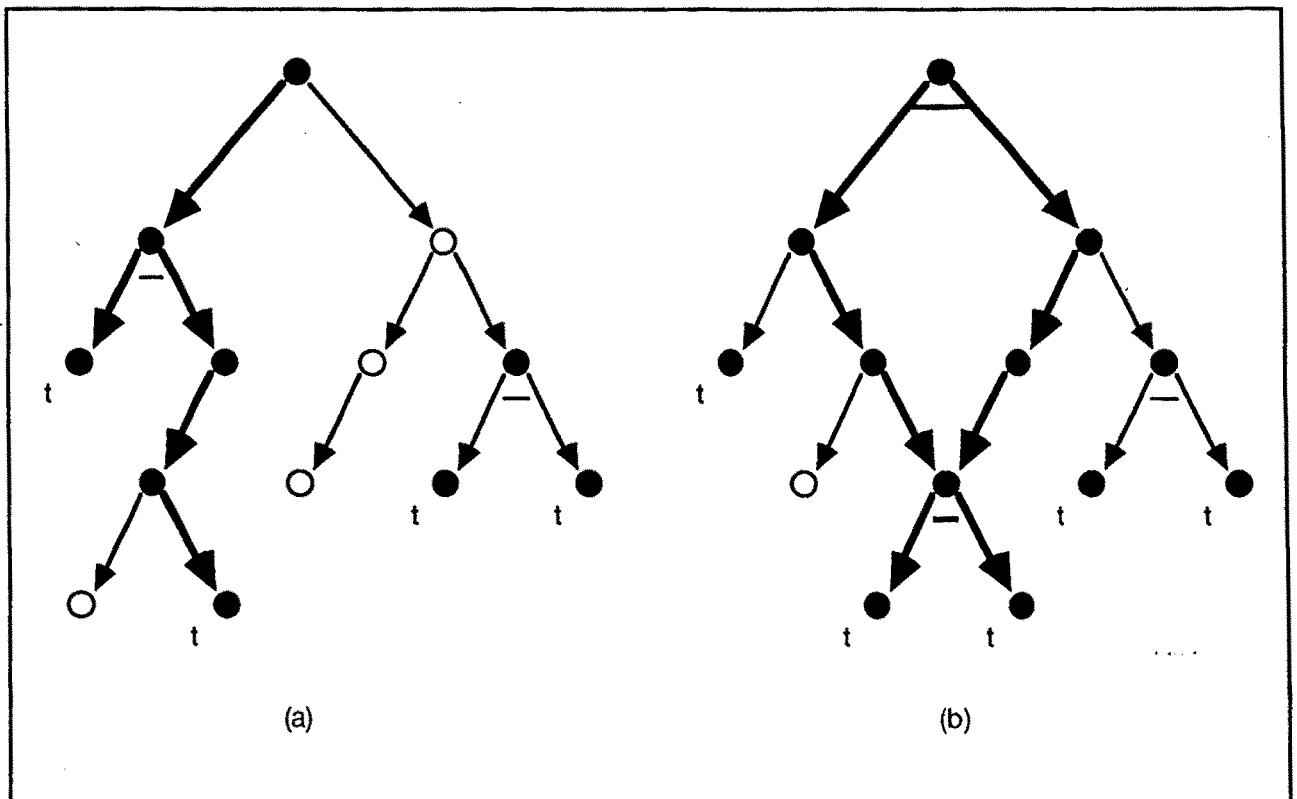


Fig. 9. Ejemplos de gráficos AND/OR y soluciones gráficas.
El gráfico (b) tiene más de una solución.

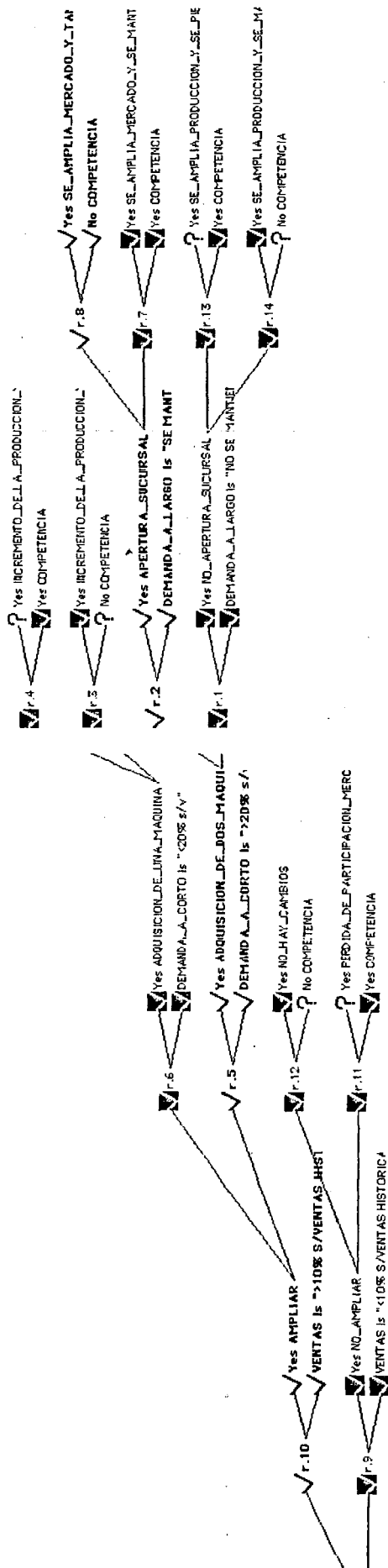
Así, el proceso de búsqueda en un gráfico AND/OR finaliza, dando por resuelto el nodo inicial, cuando: a) los nodos terminales, contienen problemas triviales, b) un nodo no terminal, que tiene sucesores OR, se resolverá, si al menos uno de sus nodos sucesores es resuelto, c) un nodo no terminal, que tiene sucesores AND, será resuelto si todos sus nodos sucesores son resueltos.

La Figura 9 muestra algunos ejemplos de gráficos AND/OR. En ella los nodos terminales se indican por la letra t, los nodos con solución se representan mediante un círculo oscuro y, la solución gráfica se indica mediante las líneas de trazo más grueso.

La representación gráfica de la estrategia seguida en la búsqueda de la solución de un problema, tiene importancia al proporcionar al usuario una visión completa de la secuencia del razonamiento. Un ejemplo concreto de representación gráfica de búsqueda de solución se muestra en la Figura 10, que presenta un problema en el que se plantean diferentes decisiones de inversión en función de, las expectativas del decisor, la evolución de la demanda, la actuación de la competencia, etc.

El conocimiento sobre el dominio en el que actúa el sistema son hechos del entorno del decisor referidos a, la evolución de las ventas, las expectativas de evolución de la demanda, comportamiento de la competencia, entre otros. Los operadores que concretan el razonamiento son reglas de producción que relacionadas presentan una estructura jerárquica en árbol.

La estrategia de control seguida en la búsqueda de la solución se muestra gráficamente mediante un signo de aceptación que indica cuando se cumplen las condiciones establecidas en la regla, verificando como consecuencia la hipótesis relacionada, y como la verificación de una regla implica la evaluación y en su caso verificación de la(s) reglas relacionadas. El signo de aceptación sobre fondo negro marca las hipótesis y reglas evaluadas pero no verificadas, y el signo de interrogación indica la no evaluación de la hipótesis.



En el ejemplo mostrado en la Figura el gráfico de búsqueda se corresponde con un gráfico AND/OR en el que se ha producido un encadenamiento hacia atrás de las reglas. La solución implica sucesivas respuestas a los subproblemas en que se ha reducido el problema inicial planteado, que no es otro, que la decisión de expansión mediante la realización de nuevas inversiones en la empresa.

La adecuada representación de los elementos que intervienen en la búsqueda de la solución de un problema, tiene gran influencia en el propio proceso de búsqueda. En general, se prefieren, las representaciones que reducen el espacio de búsqueda y, en las que un menor número de nodos quedan implicados. Una buena representación puede hacer trivial el espacio de búsqueda de un problema que en principio se mostraba complejo. Por otra parte, la representación debe ser tal que admita la posibilidad de cambios en los operadores inicialmente propuestos, si éstos se muestran ineficaces, o, incluso, una reformulación del problema, cuando es posible la reducción del espacio de búsqueda inicialmente propuesto. Para la adecuada representación, es importante, la experiencia obtenida en los intentos de resolución que impliquen representaciones similares, en orden, a distinguir estados del problema que pueden ser simplificados, u operadores que admitan agrupaciones en otros más potentes "macro-operadores".

El empleo de variables en la representación amplía, también, la eficacia de la misma. Una expresión que integra variables, puede usarse para describir un conjunto de estados, en uno solo. Los diferentes valores que pueden ser asignados a las variables, permiten la representación de cada uno de los anteriores estados.

2.5.5 METODOS DE BUSQUEDA.

La definición de la estrategia de control, a incorporar en un sistema experto, se facilita con la utilización de *gráficos de búsqueda*. Tanto si se trata de estructuras en árbol, donde cada nodo representa un estado del problema, como si se trata de gráficos AND/OR, obtenidos por reducción del problema objetivo, la explícita representación de las

estructuras de datos, mediante nodos y arcos, pone de manifiesto el camino hacia la solución del problema planteado.

Los métodos de búsqueda pretenden la adecuada representación gráfica del espacio de soluciones de un problema. Como consecuencia de su aplicación, se obtiene un gráfico de búsqueda que se constituye en un subgrafo del más general correspondiente al espacio de las soluciones posibles del problema. Se estudiarán dos métodos básicos de búsqueda: a) el método de búsqueda ciega "blind search" o, búsqueda exhaustiva y, b) el método de búsqueda heurística "heuristic search", en el que se incorpora información específica del dominio, para mejorar tiempos de proceso.

- a) **Búsqueda exhaustiva:** el método de búsqueda exhaustiva, aplicado a los procesos de razonamiento inductivo, conduce a la elaboración de un gráfico de búsqueda en el que, de modo explícito, se muestra el camino que lleva de un estado inicial a un estado objetivo. Cada estado del problema se representa mediante un nodo y cada arco del gráfico representa la aplicación de un operador, que transforma el estado del problema sobre el que se aplica, en otro estado sucesor de aquél.

En la búsqueda exhaustiva, es arbitrario el orden en que se evalúan los caminos que representan soluciones potenciales. En cualquier caso, se pretende encontrar todos los nodos sucesores a un nodo dado; es decir, todos los estados que pueden derivarse del estado actual del problema, como consecuencia de la aplicación de un operador. Es el proceso de *expansión de un nodo*.

El método se ajusta a los siguientes supuestos teóricos:

- 1) El gráfico de búsqueda es una estructura jerárquica en árbol. Un sólo nodo inicial -nodo raíz- se asocia a la descripción del estado inicial del problema. El camino que enlaza el nodo inicial con cualquier otro nodo es único.

- 2) La aplicación de un operador a un nodo provoca, la creación de un nodo por cada uno de los sucesores posibles. Cada nodo sucesor, contiene un puntero que lo relaciona con el nodo padre. Estos punteros, indican el camino recorrido, desde el nodo inicial hasta el nodo objetivo, cuando, finalmente, este se alcanza.

Una completa especificación del proceso de búsqueda, requiere una descripción del orden en que se ha producido la expansión de los nodos. Si los nodos se expanden en el orden en el cual se han generado, el proceso se denomina "breadth-first" o búsqueda horizontal. Si la expansión se produce cada vez, sobre el último nodo generado, el proceso se denomina "Depth-first" o, búsqueda vertical.

En el método de búsqueda horizontal, la expansión de los nodos se produce según su proximidad al nodo inicial y viene medida por el número de arcos definidos entre ellos. Se consideran, por tanto, todas las secuencias posibles de operadores de longitud n antes de pasar a la secuencia de longitud $n+1$. El proceso de búsqueda horizontal, se puede convertir, dependiendo de la complejidad del problema, en un proceso largo con grandes requerimientos de memoria, al explorar en paralelo diversidad de caminos posibles. Garantiza, por otra parte, el encontrar la secuencia de solución más corta posible, siempre que exista solución. En la Figura 10, se muestra un árbol de búsqueda generado por búsqueda horizontal y, aplicado sobre un problema clásico, como es el puzzle-8.

El problema cuya resolución se plantea, es un ejemplo simple de representación de los estados de un problema. El puzzle, es un cuadrado que contiene 8 celdas, numeradas del 1 al 8. El espacio correspondiente a la celda 9, está vacío. El problema a resolver es, transformar la configuración:

2	8	3
1	6	4
7	■	5

en

1	2	3
8	■	4
7	6	5

Los operadores a aplicar a cada estado para producir su transformación son los siguientes:

ARRIBA → Mover la celda vacía, una posición hacia arriba.

ABAJO → Mover la celda vacía, una posición hacia abajo.

IZQUIERDA > Mover la celda vacía, una posición hacia la izquierda.

DERECHA → Mover la celda vacía, una posición hacia la derecha.

Los nodos se numeran en la Figura según el orden en que son generados, teniendo en cuenta que el orden de generación de sucesores se produce moviendo la celda vacía, primero, a la izquierda, después, hacia arriba, derecha, y abajo. La rama de trazo más grueso, muestra la solución después de cinco movimientos ¹⁴⁴.

El gráfico muestra, un total de 26 nodos expandidos y 46 nodos generados, antes de encontrar la solución del problema. Además, se pone de manifiesto en el gráfico, que no existe otra solución que implique un número menor de movimientos.

¹⁴⁴ Se han omitido las flechas en la representación de los arcos en el gráfico, al no producirse ambigüedad en la búsqueda de la solución.

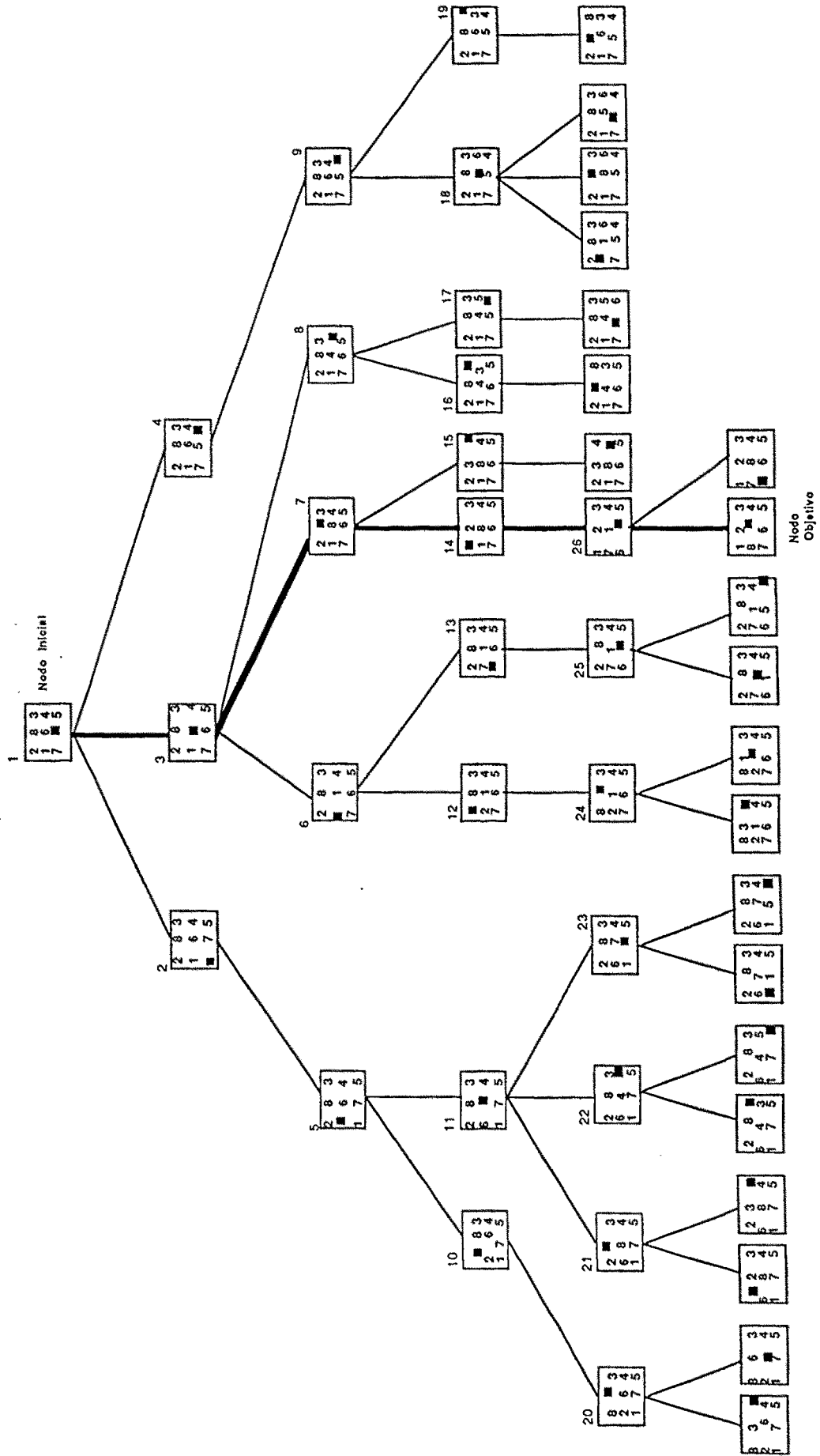


FIG 10 Arbol búsqueda horizontal.

En el método de búsqueda vertical la expansión de los nodos se produce sobre el último nodo generado. La verticalidad o profundidad de un nodo en el árbol de búsqueda, queda definida de la siguiente forma: a) la profundidad del nodo raíz es cero, b) la profundidad de cualquier otro nodo descendiente del raíz, es uno más que la profundidad de su predecesor - nodo padre-.

La consecuencia de expandir primero el nodo más profundo hace que la búsqueda siga un solo camino a través del espacio de búsqueda y a partir del nodo inicial. Sólo si la búsqueda conduce a un estado que no tiene sucesores, se considerará un camino alternativo.

En algunos problemas, el árbol de estados puede tener una verticalidad infinita, o puede ser más profundo que cualquier aceptable secuencia de solución. Para prevenir la excesiva longitud de los caminos de búsqueda de la solución, a menudo, se marca un máximo en la verticalidad de expansión de los nodos y cualquier nodo que alcance aquella, es tratado como nodo terminal sin sucesores. Si se marca un límite en la expansión de los nodos la solución encontrada, es posible no sea aquélla que se corresponda con el camino más corto para alcanzarla.

La Figura 11, muestra un árbol generado por búsqueda vertical y referido al problema del puzzle planteado en el apartado anterior.

La solución del problema, pretende la misma transformación partiendo de la misma situación inicial. Los nodos se numeran en el orden en que son expandidos y se asume que el límite de profundidad del árbol es cinco. De este modo los caminos a explorar para la solución tendrán longitud cinco o menor.

El procedimiento de búsqueda vertical explora sólo un camino, hasta alcanzar la máxima profundidad. Entonces inicia la consideración de otros caminos alternativos, de la misma o menor profundidad y que difieren del anterior en el último paso; el siguiente, en dos pasos y así sucesivamente.

En los procesos de razonamiento deductivo, el método de búsqueda exhaustiva, se apoya en la elaboración de gráficos AND/OR para alcanzar la solución del problema planteado. La generación de estos gráficos, se corresponde con un proceso búsqueda que termina, satisfactoriamente, cuando se alcanza la solución gráfica. Esta, requiere, la generación de un gráfico AND/OR que demuestre que el nodo inicial tiene solución.

Formalmente, un nodo o problema se dice que tiene solución, si se cumple una de las condiciones siguientes:

- 1) El nodo, forma parte de un conjunto de nodos terminales, que se corresponden con problemas elementales y no tiene sucesores.
- 2) El nodo tiene sucesores OR, y al menos uno de sus nodos sucesores tiene solución.
- 3) El nodo tiene sucesores AND, y todos sus nodos sucesores tienen solución.

Por el contrario, un nodo no tiene solución si alguna de las siguientes condiciones es cierta:

- 1) El nodo no tiene sucesores y no pertenece al conjunto de nodos terminales. Es decir, el nodo se corresponde con un problema no elemental y no existen operadores que puedan serle aplicados.
- 2) El nodo tiene sucesores AND y uno o más de sus sucesores no tiene solución.
- 3) El nodo tiene sucesores OR y todos sus sucesores no tienen solución.

La definición de un nodo como insoluble o con solución, es recursiva. Las condiciones de la definición pueden usarse en procedimientos iterativos

para etiquetar todos los nodos de un gráfico. La búsqueda terminará, satisfactoriamente, tan pronto como el nodo inicial pueda etiquetarse como nodo con solución. También terminará, pero en este caso sin solución, si el nodo inicial se etiqueta como insoluble.

En el caso, de etiquetar el nodo inicial como nodo con solución, finalizando así el proceso de búsqueda, la solución gráfica obtenida es un subgrafo del gráfico general de solución del problema y contiene solo los nodos con solución, suficientes, de acuerdo con las condiciones anteriores, para indicar la solución del nodo inicial.

Así, el proceso de búsqueda tiene las fases siguientes:

- 1) Un nodo inicial se asocia a la descripción inicial del problema.
- 2) Los operadores de reducción se aplican sobre el nodo inicial, obteniendo como consecuencia un conjunto de nodos sucesores - proceso de expansión de un nodo-.
- 3) Para cada nodo sucesor, se establecen punteros, que los relacionan con los nodos padres. Estos punteros se utilizarán cuando el nodo inicial quede etiquetado, para establecer la solución gráfica.
- 4) El proceso de expansión de nodos y establecimiento de punteros, continuará hasta etiquetar el nodo inicial.

Los métodos de búsqueda aplicables a los árboles AND/OR difieren básicamente en el orden en que se expanden los nodos. Estos métodos son similares a los utilizados en la búsqueda exhaustiva de los estados de solución, en el razonamiento inductivo. El método de búsqueda horizontal, expande los nodos en el orden en que son generados, y el método de búsqueda vertical, expande primero el último nodo generado.

Un ejemplo del orden en que se produce la expansión de los nodos en la búsqueda horizontal se muestra en la Figura 12(a), donde el número asociado a cada nodo indica el orden de expansión, los nodos con solución

se marcan mediante círculos oscuros y el árbol de solución se muestra por líneas gruesas.

Los nodos A, B, C no tienen sucesores y no son nodos terminales. Serían, por tanto, nodos sin solución.

La búsqueda vertical, intenta encontrar el árbol de solución, con la expansión, hasta un determinado nivel de profundidad, del último nodo generado cada vez. La Figura 12(b), muestra un ejemplo de expansión de los nodos.

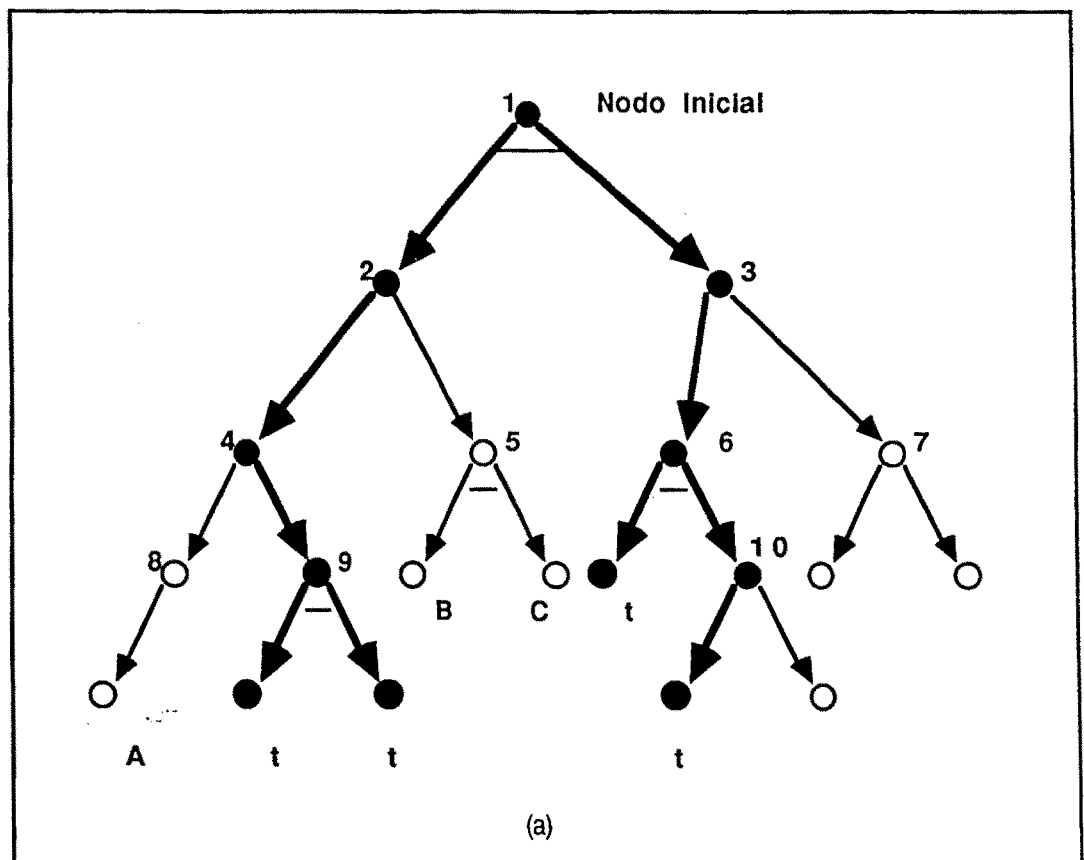


Fig. 12(a). Ejemplos de expansión de nodos en gráficos AND/OR
Árbol AND/OR y orden de expansión de los nodos
en una búsqueda horizontal

Los árboles AND/OR y los métodos de búsqueda aplicables, permiten representar procedimientos de búsqueda de solución. Sin embargo, hay ciertos aspectos que, de tenerse en cuenta, pueden mejorar el rendimiento de la representación:

- 1) La reducción de un problema produce un sucesor que es equivalente a algún subproblema ya resuelto en una etapa anterior del proceso de búsqueda. El reconocimiento de esa equivalencia reduce el esfuerzo de la búsqueda, puesto que subproblemas equivalentes se resuelven solo una vez,

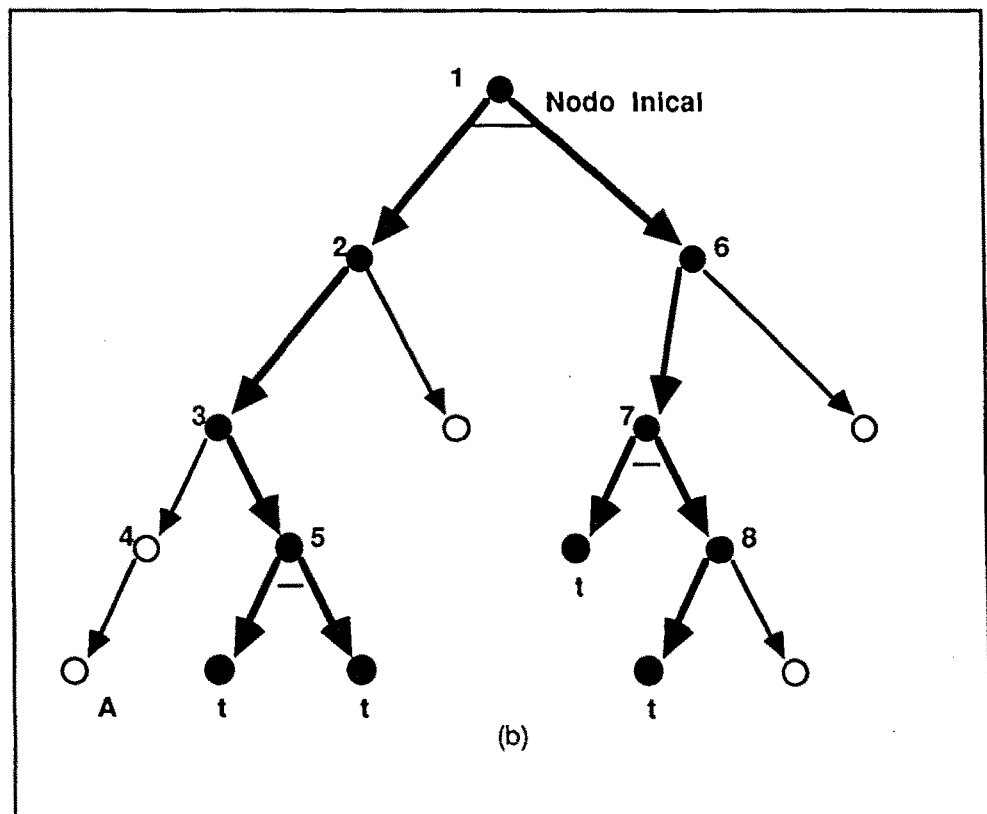


Fig. 12(b). Ejemplos de expansión de nodos en gráficos AND/OR
 Arbol AND/OR y orden de expansión de los nodos en una búsqueda vertical,
 con un nivel de profundidad de 4

- 2) En ocasiones, la estructura del gráfico de búsqueda de la solución requiere la utilización de bucles en el razonamiento -razonamiento circular-, es decir, admite en determinadas circunstancias volver a evaluar un razonamiento anterior.

En ambos casos mencionados, el proceso de búsqueda tiene una representación más eficiente en un gráfico de búsqueda AND/OR, que admita la posibilidad de múltiples padres para un mismo nodo, requisito no permitido en las estructuras AND/OR en árbol.

El considerar más de un padre a un nodo sucesor, afecta a los métodos de búsqueda que requieren la definición de más de un puntero por nodo y en los que se define la profundidad de un nodo, no inicial, como una unidad más que el nodo padre menos profundo. Un ejemplo de aplicación es el planteado en la Figura 13. Se trata de un juego que emplea las técnicas de reducción para encontrar una estrategia de juego para uno de los jugadores que intervienen.

Las reglas del juego son las siguientes ¹⁴⁵: Dos jugadores tienen en frente una pila de objetos, por ejemplo, una columna de 7 monedas. El primer jugador divide la columna en dos con la condición de tener distinto número de unidades cada una. Cada jugador alternativamente hará lo mismo con una única columna. El juego termina cuando en cada columna hay una o dos monedas como máximo. El primer jugador que no pueda mover pierde la partida.

¹⁴⁵ NILSSON, N. J. (1.971): "Problem-Solving Methods in Artificial Intelligence". MacGraw-Hill New York

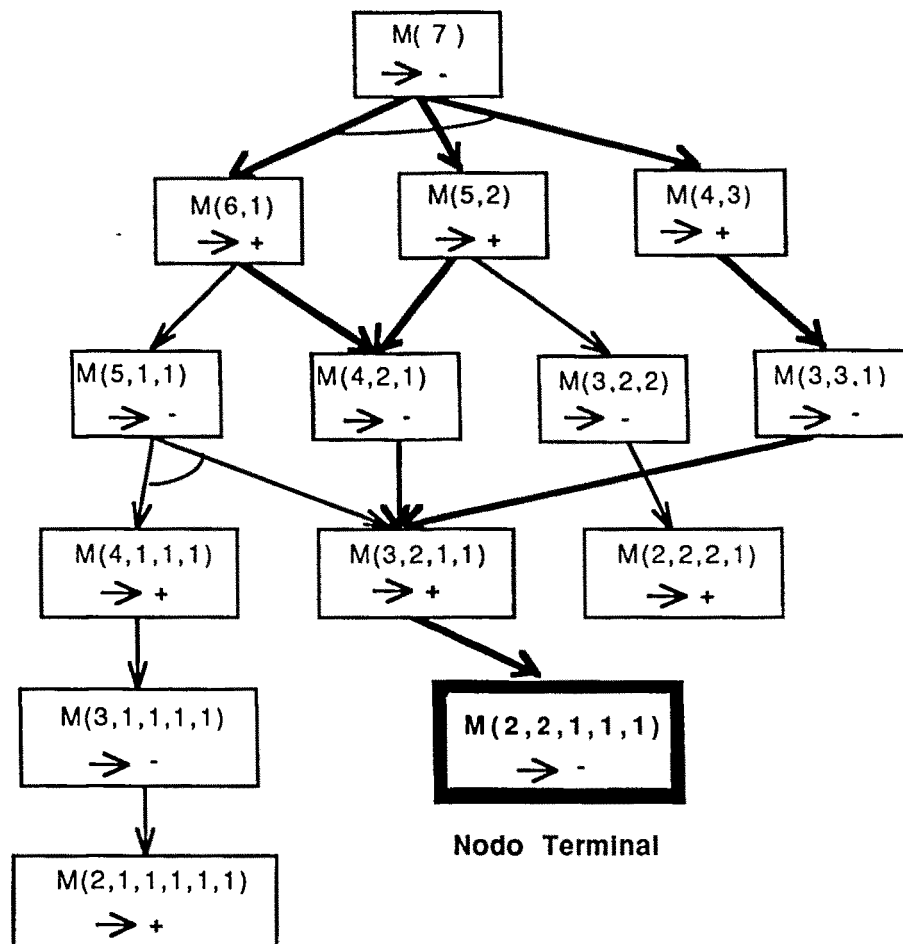


Fig. 13. Ejemplo de aplicación de gráficos AND/OR

En el gráfico los jugadores se identifican como MAS y MENOS, y la flecha indica el próximo a jugar. Inicia el juego MENOS, la estrategia para ganar se prepara para el jugador MAS. En la posición inicial MENOS tiene tres alternativas de juego: (6,1), (5,2) y (4,3). Hay tres nodos terminales, es decir, configuraciones en las cuales el jugador no puede seguir moviendo: (2,1,1,1,1,1), (2,2,1,1,1) y (2,2,2,1). El nodo objetivo que dará ganador a MAS es el (2,2,1,1,1).

Otra mejora en el proceso de búsqueda está en la posibilidad de combinar razonamiento inductivo y deductivo en un mismo proceso. Esta técnica se denomina búsqueda bidireccional. El objetivo es reemplazar un único gráfico, que un proceso de búsqueda exhaustiva puede hacer crecer exponencialmente en cada nivel, por dos gráficos más limitados. En uno, el nodo inicial recogería la situación inicial del problema; en el otro, el nodo inicial se correspondería con el objetivo cuya validez se quiere comprobar. La búsqueda terminaría con la intersección de los dos gráficos.

En los sistemas expertos con razonamiento codificado en reglas de producción, el encadenamiento entre las reglas se produce teniendo en cuenta las relaciones entre ellas. Así, dos reglas estarán relacionadas si entre ellas se pueden establecer enlaces basados en la parte de condición o en la parte de acción.

Los enlaces basados en condición, ocurren cuando las reglas comparten datos o variables de la parte de condición de la regla. Los enlaces basados en acción, ocurren cuando se comparten datos entre la parte de acción de la regla y la parte de condición de la regla con la que se relaciona.

Son también posibles otros enlaces entre reglas, que recogen un tipo de razonamiento que en ocasiones no puede ser expresado explícitamente mediante alguna de las relaciones anteriores. Estos enlaces, permiten poner de manifiesto relaciones entre reglas debidas, por ejemplo, a la experiencia o a la intuición del experto que, por otra parte, presentan una difícil formalización. Es el caso, de las relaciones que se establecen entre reglas, cuando una de ellas se plantea como un subproblema de la clase de problemas al que pertenecen las otras reglas. Estos enlaces suelen resolverse en la práctica con algún tipo de encadenamiento hacia adelante.

- b) **Búsqueda Heurística:** los métodos de búsqueda exhaustiva, tanto la búsqueda horizontal como vertical, permiten encontrar la solución de un

problema. Pero, a menudo, su aplicación no es posible, al provocar éstos en el proceso de búsqueda la expansión de un número demasiado elevado de nodos antes de encontrar la solución del problema. En la práctica, siempre hay límites a la cantidad de tiempo y almacenamiento disponibles para el proceso de búsqueda, por lo que parece necesaria la existencia de otros métodos alternativos a la búsqueda exhaustiva.

En muchas aplicaciones, es posible guiar el proceso de búsqueda, mediante información específica sobre el dominio del problema, reduciendo de este modo el tiempo de proceso. La información especializada que guía la búsqueda se denomina *información heurística*. La búsqueda heurística finalizará tan pronto como se encuentre una solución satisfactoria.

La aplicación de información heurística al proceso de búsqueda se puede llevar a cabo:

- 1) Decidiendo el siguiente nodo a expandir, sin tener en cuenta el orden que le correspondería en una búsqueda horizontal o vertical.
- 2) Decidiendo durante la expansión de un nodo, qué sucesor o sucesores van a ser generados. Difiere de la búsqueda exhaustiva, que genera todos los sucesores posibles en cada expansión de un nodo.
- 3) Decidiendo qué nodos van a ser eliminados del árbol de búsqueda.

En el primer caso, la información heurística, permite decidir qué nodo se va a expandir a continuación, teniendo en cuenta que la expansión completa de un nodo genera todos sus sucesores y la no expansión no genera ningún sucesor. El nodo a expandir, es aquél que se muestra más "prometedor" en la búsqueda de la solución. El proceso de búsqueda que utiliza en esta forma información heurística, se denomina *búsqueda ordenada* o "best-first search".

En el segundo caso, se decide qué sucesores de un nodo van a ser generados. Se trata, por tanto, de determinar que operadores, de entre los posibles, se van a aplicar a un nodo dado para su expansión. El nodo sobre el que se aplica el operador seleccionado se denomina nodo parcialmente desarrollado o expandido

El tercer uso de información heurística, va dirigido a decidir qué nodos nunca van a ser expandidos. En algunos casos, se puede determinar qué nodo no va a formar nunca parte de la solución, procediendo entonces a su eliminación del árbol de búsqueda. En otros casos, se eliminan los nodos que se consideran no esenciales en la búsqueda de la solución del problema. Una razón para la eliminación puede ser el no hacer demasiado larga la lista de candidatos posibles a una futura expansión, eliminando aquellos que en principio se muestran poco "prometedores" para la solución.

La posibilidad de empleo de información heurística para determinar la rama del árbol de búsqueda a expandir, requiere de algún criterio de ordenación de los nodos en cada nivel de expansión del árbol. Para aplicar un adecuado procedimiento de ordenación, es necesario algún tipo de baremo que permita evaluar el nodo más "prometedor" en la búsqueda de la solución. Este baremo a aplicar se denomina *función de evaluación*.

El propósito de una *función de evaluación* es el de disponer de un criterio para clasificar los nodos candidatos a la expansión y seleccionar de entre ellos, el más adecuado para alcanzar la solución. En la definición de las funciones de evaluación, se pueden tener en cuenta distintos criterios. Uno de ellos es aquel que define la probabilidad de que un nodo forme parte del mejor camino para la solución. Otro criterio, es el estimar la distancia entre el nodo a evaluar y el nodo que contiene la solución.

Nilsson ¹⁴⁶, propone un algoritmo básico para llevar a cabo búsquedas ordenadas en el espacio de los estados de un problema y define una función de evaluación f para ordenar los nodos en la expansión. La definición de la función es tal que el nodo más prometedor es el que minimiza la función.

Una aplicación de la función de evaluación, sugerida por Nilsson, es aquella que propone un algoritmo de ordenación para la búsqueda en el problema del puzzle-8. La función propuesta por Nilsson es la siguiente:

$$f(n) = g(n) + W(n)$$

donde $g(n)$ es la distancia -profundidad- en el árbol de búsqueda, entre el nodo "n" y el nodo inicial, y $W(n)$, cuenta el número de celdas no ubicadas correctamente en el estado del problema asociado al nodo "n".

La Figura 14, muestra el resultado de aplicar el algoritmo mencionado para guiar la búsqueda en el ejemplo del puzzle. El valor de f para cada nodo se rodea por un círculo, mostrándose también los valores que corresponden al orden de expansión de los nodos. En la Figura, el camino que conduce a la solución del problema, coincide con el camino propuesto por otros métodos de búsqueda, pero en este caso, el empleo de la función de evaluación produce un apreciable descenso en el número de nodos expandidos y, como consecuencia, un menor tiempo de proceso. La comparación se puede establecer con el árbol generado por búsqueda vertical de la Figura 14 donde el nodo objetivo se alcanza después de la expansión del nodo 18, mientras que la búsqueda heurística genera la solución después de la expansión del nodo 6.

¹⁴⁶ NILSSON, N.J. (1971): "Problem-Solving Methods in Artificial Intelligence". MacGraw-Hill New York

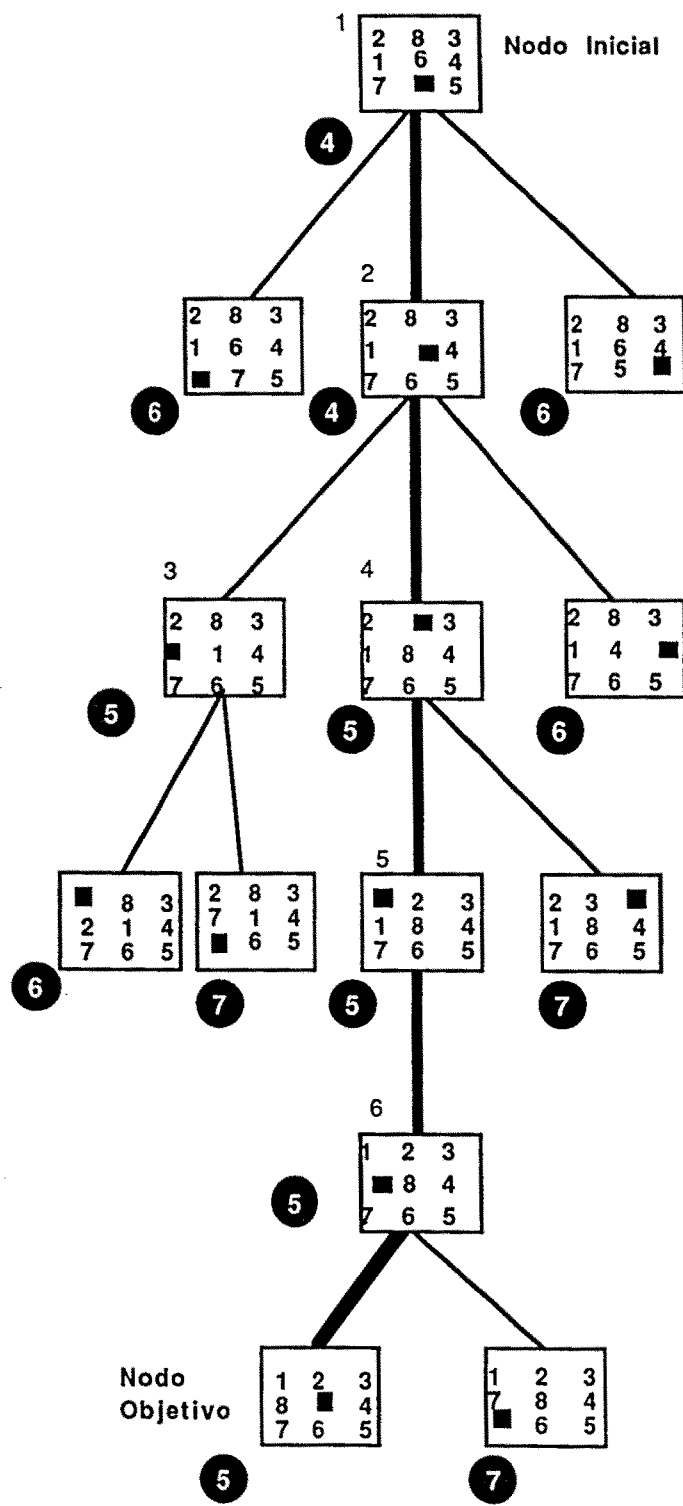


Fig. 14. Arbol de búsqueda ordenada

El propósito de la búsqueda ordenada, es reducir el número de nodos expandidos, respecto a los métodos de búsqueda exhaustiva. La efectividad de la reducción, depende fuertemente de la función de evaluación elegida. Esta, al discriminar los nodos a expandir, puede llevar a una pérdida de la solución óptima, o a no encontrar la solución del problema, si la discriminación es inadecuada.

Por otra parte, la elección del nodo a expandir y, como consecuencia, la elección de la función de evaluación apropiada, depende del problema a resolver. Así, un primer grupo estaría formado por aquéllos en los que el espacio de los estados del problema contiene más de una solución. En este caso, la función de evaluación, debe ser tal, que permita encontrar la solución óptima, es decir, aquella que suponga un coste mínimo.

Un segundo grupo, lo integran aquellos problemas, que además de tener varias soluciones posibles, el proceso de búsqueda de la solución óptima excede los límites de tiempo y memoria disponibles. La solución en estos casos está en encontrar una buena solución -no la óptima- dentro de límites razonables establecidos para la búsqueda.

En un tercer tipo de problemas, no se trata de buscar la solución óptima, porque el problema o bien tiene una única solución, o hay una solución mejor que ninguna otra. En estos casos, la función de evaluación debe tratar de minimizar el esfuerzo de búsqueda.

Otros posibles métodos de búsqueda guiada, son aquéllos que hacen abstracción del espacio de las soluciones del problema para reducir el espacio de búsqueda, o bien aquéllos que dividen el proceso de búsqueda en dos partes: un generador de soluciones posibles y un analizador que contrasta las soluciones obtenidas, con las restricciones del problema y elimina aquéllas que no las cumplen. Los procesos de generación de soluciones y su análisis pueden solaparse en el tiempo.

En los sistemas de producción, ciertos enlaces entre reglas o grupos de reglas son consecuencia de procesos cognitivos del experto, derivados de su experiencia, en la resolución de problemas similares y que presentan, además, una difícil formalización. Estos enlaces, pueden actuar como heurísticos dirigidos a restringir el espacio de búsqueda de la solución guiando ésta. En estos casos, la estrategia de enlace entre grupos de reglas puede ser definida con carácter general y modificada, en su caso, dependiendo de información del contexto. Es necesario, también, arbitrar un procedimiento que sea capaz de seleccionar, en caso de conflicto -más de dos grupos de reglas relacionadas-, el orden de las relaciones.

2.5.6 EL PROBLEMA DEL CONTROL.

La resolución de un problema es un proceso que desarrolla una secuencia de acciones encaminadas a alcanzar un objetivo. Cada acción es generada, bien como consecuencia de los datos almacenados en el sistema, bien como consecuencia de acciones anteriores. Cada acción puede provocar, a su vez, la modificación o la generación de nuevas acciones. Teniendo en cuenta que, en cada fase del proceso de resolución se pueden producir un conjunto de acciones similares, el problema del control radica en la selección, en cada una de las fases del proceso de resolución, de las acciones adecuadas en orden a alcanzar el objetivo.

El problema del control, puede ser equiparable a un problema de planificación, que incluya al menos: ¹⁴⁷

- a) Un conjunto ordenado de acciones de resolución, para decidir en cada fase qué acciones se van a realizar y en qué orden.
- b) El establecimiento de prioridades de ejecución de las acciones, diferenciando las acciones posibles de las deseables.
- c) Integrar conocimiento heurístico, para la selección de acciones.

¹⁴⁷ HAYES-ROTH, B. (1.985): "A Blackboard Architecture for Control" Artificial Intelligence nº26 251-321.

- d) Producir acciones de control encaminadas a dirigir el sistema.
- e) Interpretar situaciones dinámicas y disponer de un plan dinámico de secuencias de acciones.
- f) Interpretar la situación actual del sistema, en orden a conocer cuándo se ha resuelto el problema.

La *planificación* es una técnica de resolución de problemas, que permite establecer un plan, una línea de acción, de resolución. Un plan contiene, generalmente, una lista ordenada de los objetivos a satisfacer. La variedad de objetivos contenidos en el plan, hace que cada uno de ellos pueda ser reemplazado por un subplan que registre con detalle las características del objetivo a lograr. Un plan queda así configurado como una estructura jerárquica de subobjetivos.

La planificación aporta un conjunto de ventajas en la resolución de problemas, que se pueden resumir en las siguientes: reducción del proceso de búsqueda, resolución de conflictos entre objetivos y recuperación de errores. A su vez, el plan permite supervisar el proceso de resolución, de especial importancia en aquellos problemas cuyo entorno es variable.

Respecto a la reducción del proceso de búsqueda, el plan pretende encontrar una secuencia ordenada de acciones, de entre todas las posibles, que logren el objetivo previsto. En cuanto a la resolución de conflictos entre objetivos, ésta se plantea cuando hay más de una condición a satisfacer y el orden en el que estas deben cumplirse no se especifica en el problema, siendo sin embargo crítico para su resolución. De hecho, el proceso de búsqueda se ve afectado en estos casos y puede suceder que se requiera búsqueda adicional, si se establece un orden arbitrario en la consecución de los subobjetivos relacionados.

En el plan tiene cabida, también, un tipo de razonamiento basado en el sentido común del experto. Un razonamiento que permite realizar ciertas asunciones, pudiendo éstas ser rechazadas más tarde como consecuencia de nueva información. Este

razonamiento se denomina no monótono ¹⁴⁸, en contraste con el razonamiento monótono empleado en la lógica deductiva.

El razonamiento por defecto cae dentro de la categoría de razonamiento no monótono y permite incorporar un tipo de conocimiento al sistema que es aceptado, al menos, hasta no tener evidencia en contra ¹⁴⁹. El razonamiento por defecto, forma parte de las técnicas de Inteligencia Artificial y es usado para establecer generalizaciones, cuyas excepciones deben ser conocidas para evitar posibles graves errores en el razonamiento del sistema. Una dificultad que se puede plantear en éste tipo de razonamiento es la circularidad: el conocimiento que se puede inferir depende de las reglas de inferencia aplicadas, al mismo tiempo, la selección de la regla de inferencia a aplicar, depende del conocimiento inferido.

El razonamiento por defecto, es una forma de realizar asunciones, necesarias, cuando se precisa obtener conclusiones a partir de información parcial o incompleta. En estos casos, el sentido común del experto permite la realización de las asunciones, dependiendo de las condiciones del problema y, mientras no exista evidencia en contra.

Una posible formalización del razonamiento por defecto, es la propuesta por Reiter, ¹⁵⁰ quién utiliza para su representación, reglas de inferencia características, que entran en relación con el resto de reglas de inferencia. De este modo, la expresión $(P \longrightarrow Q)$: "Si P es cierto, entonces Q es cierto" precede a una regla básica de inferencia en lógica, el "modus ponens":

$$(((P \longrightarrow Q) \text{ and } P) \longrightarrow Q)$$

"Si $(P \longrightarrow Q)$ es verdadero y P es verdadero, entonces se puede afirmar que Q es verdadero". Las reglas de inferencia que recogen el razonamiento por defecto, propuestas por Reiter, tienen la siguiente formalización:

¹⁴⁸ La lógica matemática formal es monótona, en el sentido, que las inferencias son acumulativas; ninguna conclusión necesita nunca refutarse. WINOGRAD, T (1.980) "Extended inference modes in reasoning by computer system". En WINOGRAD, T. y FLORES, F. (1.986): "**Understanding computers and cognition: a new foundation for design**" Ablex Publishing Corporation Norwood.

¹⁴⁹ McDERMOTT, D. y DOYLE, J. (1.980): "**Non-Monotonic logic**". En ANDERSON, J. (1.983): "**The Architecture of Cognition**" Harvard University Press Massachusetts.

¹⁵⁰ REITER, R. (1.980): "A logic for default reasoning". En BENCH-CAPON (1.990): "**Knowledge Representation: An Approach to Artificial Intelligence**" Academic Press London.

P:Q/R

"Si P es verdadero y se puede asumir Q, entonces se puede inferir R". En este caso, la condición dada es P, y el razonamiento por defecto se produce con la inferencia de R, como consecuencia que Q no se puede refutar.

2.5.7 APRENDIZAJE.

Simon ¹⁵¹ define el aprendizaje como: *cualquier proceso por el que un sistema mejora su rendimiento*. Aplicada esta definición a los sistemas expertos, se puede interpretar que el trabajo de resolución de problemas desarrollado por el sistema, puede perfeccionarse con aplicación de nuevos métodos de resolución y nuevo conocimiento o, mejorando los métodos que aplica y el conocimiento de que dispone, para alcanzar mayor rapidez, fiabilidad y exactitud.

La Inteligencia Artificial se interesa pronto en los procesos de aprendizaje, por dos motivos principales: el primero, responde al objetivo de conocer mejor el propio proceso de aprendizaje; el segundo, el de proporcionar a los programas de ordenador la habilidad del aprendizaje. En los sistemas expertos, la necesidad del sistema de adquirir conocimiento, lleva a considerar el aprendizaje como una parte importante del propio sistema.

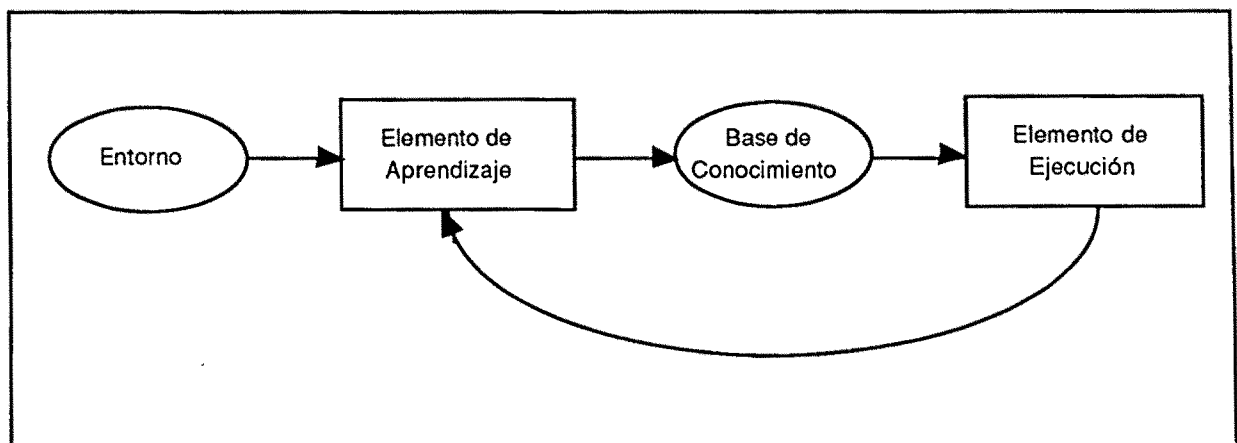
Muchos sistemas expertos, representan su conocimiento mediante un gran conjunto de reglas y se considera el aprendizaje como la adquisición de conocimiento explícito, con la incorporación de nuevas reglas y su integración en la base de conocimiento del sistema.

En otros casos, se considera que el aprendizaje del experto consiste en adquirir habilidad en el desarrollo de una tarea y, en este sentido, se informa al sistema cómo llevar a cabo un trabajo cuyo resultado puede ir mejorando con la práctica. Es posible, también, considerar el aprendizaje como el proceso de construir una teoría que describa y explique un fenómeno complejo. Sería necesario, en este caso, la elaboración de

¹⁵¹ SIMON, H. A. (1.982): "La Nueva Ciencia de la Decisión gerencial". El Ateneo Buenos Aires.

hipótesis que interpretaran un conjunto determinado de datos, en el contexto de una teoría más general.

Un modelo elemental de aprendizaje, derivado de la definición de Simon, puede ser el siguiente:



Donde se representan mediante formas circulares conjuntos de información y mediante rectángulos procedimientos. Las flechas indican la dirección del flujo de datos a través del sistema de aprendizaje. En el sistema, el entorno suministra información al elemento de aprendizaje. Este usa la información para perfeccionar la base de conocimiento y el elemento de ejecución emplea la base de conocimiento para llevar a cabo su tarea. Finalmente, la información requerida en el intento de realización de la tarea, es nuevamente reciclada por el elemento de aprendizaje.

El modelo descrito es elemental, pero permite identificar las cuatro unidades funcionales de que consta un sistema de aprendizaje: entorno, base de conocimiento, ejecución de la tarea y procedimiento de aprendizaje. En una aplicación determinada, el entorno, la base de conocimiento y la tarea a ejecutar determinan el problema particular de aprendizaje y las funciones que éste debe realizar.

El entorno tiene una influencia fundamental sobre el diseño del sistema de aprendizaje. Este depende del tipo de información que llega del entorno, valorándose el grado de generalidad de la información y la calidad de la misma, como dos de sus aspectos básicos. El grado de generalidad o grado de abstracción de la información lleva a considerar dos tipos de información: de *alto nivel*, como información abstracta, que es relevante respecto a la clase de problemas a la que pertenece la tarea a realizar; de *bajo nivel*, que hace referencia a información detallada, relevante para el problema concreto a resolver. La tarea del elemento de aprendizaje consiste en acomodar el nivel de información que llega al sistema procedente del entorno, al nivel que el elemento ejecutor necesita para llevar a cabo su trabajo.

Si al sistema de aprendizaje llega información demasiado abstracta, para la tarea a realizar, aquél debe completarla para que el elemento de ejecución pueda interpretar esta información en situaciones específicas. Si por el contrario, del entorno llega información muy específica, de bajo nivel, relativa a un problema particular, el elemento de aprendizaje debe generalizar esta información, para guiar la resolución del problema.

Por otra parte, si el conocimiento es imperfecto, el elemento de aprendizaje puede no conocer exactamente cómo completar la información o cómo generalizarla, teniendo, en estos casos, que establecer hipótesis para hacer posible su empleo. A estos efectos, el sistema, debe disponer de un mecanismo que le permita revisar y, en su caso, rectificar las hipótesis realizadas. El sistema de aprendizaje así planteado se denomina aprendizaje por prueba y error.

El grado de abstracción de la información, que proviene del entorno, determina la clase de hipótesis que el sistema debe generar, y por tanto las situaciones de aprendizaje. Entre ellas:

- a) Aprendizaje, en que el entorno ofrece información con igual grado de abstracción que se requiere en la resolución del problema. En este caso, no es necesaria la elaboración de hipótesis. El sistema de aprendizaje no

llevará a cabo ningún proceso para entender o interpretar la información ya que simplemente almacenará dicha información para su posterior empleo.

- b) Aprendizaje, en que el entorno ofrece información demasiado abstracta o general, propiciando la elaboración de hipótesis, por parte del elemento de aprendizaje, para cubrir los detalles omitidos. Este conocimiento de propósito general que llega al sistema, conocimiento de alto-nivel, debe ser transformado mediante un proceso que incluye: la realización de asunciones, la deducción de consecuencias, solicitud de nueva información, en su caso. Todo ello para conseguir adecuar el conocimiento a las necesidades de utilización del elemento de ejecución.
- c) Aprendizaje, en que la información que proviene del entorno es demasiado específica, obligando al elemento de aprendizaje a establecer hipótesis que transformen aquélla en conocimiento más general "conocimiento de alto-nivel" para su empleo efectivo por el elemento de ejecución. Una forma de enseñar al sistema cómo llevar a cabo una tarea es presentarle ejemplos de cómo realizarla. El sistema generalizará éstos, para encontrar reglas que guíen al elemento de ejecución.
- d) Aprendizaje, en que la información que proviene del entorno es relevante en problemas que reúnen ciertas características. El elemento de aprendizaje debe descubrir las características comunes en la tarea a realizar y establecer las hipótesis correctas. Si el sistema dispone de conocimiento adecuado para llevar a cabo una determinada tarea, puede reconocer analogías con otras tareas que se le presenten y transferir el conocimiento relevante a otra base de conocimiento.

La calidad de la información tiene, también, un efecto significativo en la tarea de aprendizaje. Si la información se ofrece poco depurada y sin orden, los procesos de razonamiento y de aprendizaje por inducción presentan dificultades.

El segundo factor que afecta al diseño de un sistema de aprendizaje, es la base de conocimiento. La forma en que el conocimiento se expresa en la base, decidido en su diseño, afecta a su expresividad: facilidad para expresar conocimiento relevante, a su facilidad para realizar inferencias teniendo en cuenta el coste de los recursos de proceso dedicados, a su modificabilidad y, a su capacidad de ampliación.

Además, para que el elemento de aprendizaje, pueda manipular el conocimiento adquirido, debe disponer de información acerca de cómo se ha estructurado la representación -Meta-conocimiento-. Este, puede quedar incorporado en los procedimientos de manipulación de las estructuras de datos, pero también puede estar presente, de modo explícito, formando parte de la base de conocimiento del sistema. El propósito de este meta-conocimiento es permitir al programa examinar y modificar su propia estructura representacional.

Un sistema de aprendizaje requiere, por tanto, el empleo de conocimiento, para comprender la información que proviene del entorno, para formar hipótesis y para contrastar y refinar estas. Un sistema de aprendizaje se puede entender, en esta línea, como una extensión y perfeccionamiento de un determinado cuerpo de conocimientos.

El tercer factor básico de un sistema de aprendizaje es el elemento de ejecución, ya que son las acciones que éste lleva a cabo las que el sistema de aprendizaje intenta mejorar. Tres son los aspectos básicos a considerar relativos al elemento de ejecución: complejidad, retroacción y transparencia.

La complejidad del problema es un factor importante, teniendo en cuenta que tareas complejas requieren más conocimiento que las tareas simples. La resolución de un problema sencillo puede implicar unas pocas decenas de reglas para alcanzar la solución. Por el contrario, un sistema de diagnóstico médico, como es el caso de MYCIN, puede emplear cientos de reglas para almacenar el conocimiento. Así, a medida que el problema a resolver se vuelve más complejo, la base de conocimiento crece y aumenta la dificultad para integrar nuevas reglas y detectar aquellas incorrectas. La integración de una nueva regla en un conjunto de reglas ya existentes plantea problemas, ya que el

sistema de aprendizaje debe considerar las posibles interacciones entre la nueva regla y las ya almacenadas.

Respecto a la retroacción, todos los sistemas de aprendizaje deben disponer de algún método de evaluación de las hipótesis propuestas por el elemento de aprendizaje. Los programas pueden disponer de conocimiento diferenciado para llevar a cabo dicha evaluación.

La transparencia del elemento de ejecución hace referencia a la posibilidad de acceso del elemento de aprendizaje al proceso de razonamiento seguido por el elemento de ejecución, para evaluar individualmente la aplicación de las reglas de la base de conocimiento.