

## **Tabla de contenido**

<b>Tabla de contenido .....</b>	<b>1</b>
<b>Introducción.....</b>	<b>4</b>
<b>1. Introducción a la Inteligencia Artificial .....</b>	<b>4</b>
1.1. DEFINICIONES DE IA.....	5
1.1.1. <i>La inteligencia</i> .....	5
1.1.2. <i>El pensamiento</i> .....	6
1.1.3. <i>Inteligencia Artificial</i> .....	7
1.2. EVOLUCIÓN HISTÓRICA .....	8
1.3. APLICACIONES.....	10
1.4. ÁREAS O CAMPOS DE LA IA .....	10
1.4.1. <i>Data Mining o Minería de Datos (DM o MD)</i> .....	11
1.4.2. <i>Redes Neuronales Artificiales (RNA)</i> .....	11
1.4.3. <i>Algoritmos Evolutivos o Algoritmos Genéticos (AG)</i> .....	12
1.4.4. <i>Sistemas Expertos (SS.EE.)</i> .....	12
1.4.5. <i>Agentes Inteligentes</i> .....	13
<b>2. Introducción a los Sistemas Expertos .....</b>	<b>14</b>
2.1. EVOLUCIÓN HISTÓRICA .....	14
2.2. DEFINICIONES.....	16
2.3. APLICACIONES.....	16
2.4. COMPOSICIÓN DE UN SS.EE. E INTERRELACIÓN ENTRE SUS PARTES .....	17
2.4.1. <i>Elementos de un SS.EE.</i> .....	17
2.4.2. <i>Interrelación entre los componentes</i> .....	22
2.5. CARACTERIZACIÓN DE LOS SS.EE.....	23
2.5.1. <i>Diferencias entre un S.E. y un programa tradicional</i> .....	23
2.5.2. <i>Razones para la utilización de SS.EE.</i> .....	24
2.5.3. <i>Límites y dificultades de los SS.EE.</i> .....	25
2.6. DESARROLLO DE UN SS.EE. ....	26
2.7. TIPOS DE SS.EE.....	28
2.7.1. <i>Sistemas Expertos basados en Reglas</i> .....	29
2.7.2. <i>Sistemas Expertos basados en Probabilidad</i> .....	29
2.7.3. <i>Sistemas Expertos basados en Redes Neuronales Artificiales</i> .....	29
2.7.4. <i>Sistemas Expertos basados en Lógica Difusa</i> .....	30
<b>3. Sistemas Expertos basados en Reglas .....</b>	<b>32</b>
3.1. BASE DE CONOCIMIENTOS .....	32
3.2. MOTOR DE INFERENCIA (MI).....	33
3.2.1. <i>Regla Modus Ponens y Modus Tollens</i> .....	34
3.2.2. <i>Mecanismo de resolución</i> .....	35
3.2.3. <i>Encadenamiento de Reglas</i> .....	37
3.2.3.1. <i>Ejemplos de Encadenamiento de Reglas</i> .....	37
3.2.4. <i>Encadenamiento de Reglas orientado a un objetivo</i> .....	39
3.2.4.1. <i>Ejemplos de Encadenamiento de Reglas Orientado a un Objetivo</i> .....	40
3.2.5. <i>Compilación de Reglas</i> .....	44
3.3. SUBSISTEMA DE CONTROL DE COHERENCIA.....	45
3.4. SUBSISTEMA DE EXPLICACIÓN O JUSTIFICACIÓN .....	46
<b>4. Sistemas Expertos basados en Probabilidad .....</b>	<b>47</b>
4.1. REGLAS GENERALIZADAS .....	48
4.2. LA BASE DE CONOCIMIENTOS .....	50
4.2.1. <i>Modelo de Síntomas Dependientes</i> .....	51
4.2.2. <i>Modelo de Síntomas Independientes</i> .....	52
4.2.3. <i>Modelo de Síntomas Relevantes Independientes</i> .....	53

4.2.4. <i>Modelo de Síntomas Relevantes Dependientes</i> .....	55
Modelo.....	57
4.3. MOTOR DE INFERENCIAS .....	57
4.4. CONTROL DE COHERENCIA.....	58
<b>5. Comparación entre los SS.EE. basados en Reglas y los basados en Probabilidades .....</b>	<b>59</b>
5.1. BASE DE CONOCIMIENTOS .....	59
5.2. MOTOR DE INFERENCIA .....	60
5.3. SUBSISTEMA DE EXPLICACIÓN .....	60
5.4. SUBSISTEMA DE APRENDIZAJE .....	60
<b>7. Conclusiones.....</b>	<b>61</b>
7.1. CONTRIBUCIONES .....	61
7.2. LIMITACIONES .....	61
<b>8. Futuros Trabajos.....</b>	<b>62</b>
<b>Referencias.....</b>	<b>63</b>
<b>Bibliografía adicional .....</b>	<b>64</b>

## Índice de gráficos y tablas

Gráfico 1. Componentes básicos de un S.E.; cada flecha representa el flujo de la información. ....	19
Gráfico 2. Ejemplo de la representación gráfica en árbol de dos reglas en un proceso de inferencia. ....	21
Tabla 1. Diferencias entre un Sistema Convencional y un Sistema Experto. ....	24
Gráfico 3. Etapas de desarrollo de un Sistema Experto.....	28
Tabla 2. Clasificación de las reglas utilizadas en la BC y las Estrategias utilizadas en el MI.....	33
Gráfico 4. Ejemplo de regla Modus Ponens. ....	34
Gráfico 5. Ejemplo de regla Modus Tollens.....	35
Tabla 3. Tabla de verdad de la Regla 1. ....	36
Tabla 4. Tabla de verdad de las reglas obtenidas por la combinación.....	36
Gráfico 6. Ejemplo de las etapas de sustitución y combinación de reglas.....	37
Gráfico 7. Ejemplo de 6 (seis) reglas relacionando 13 objetos.....	38
Gráfico 8. Ejemplo de encadenamiento de reglas.....	38
Gráfico 9a. La inclusión del conjunto B en A representa una implicación fuerte. ....	49
Gráfico 9b. La intersección de los conjuntos A y B representa la implicación débil. ....	49
Gráfico 9c. Los conjuntos disjuntos es el caso de la no implicación.....	49
Gráfico 10. Relaciones entre síntomas y enfermedades en el MSD. ....	51
Gráfico 11. Relaciones entre síntomas y enfermedades en el MSI.....	52
Gráfico 12. Relaciones entre síntomas y enfermedades en el MSRI.....	54
Gráfico 13. Relaciones entre síntomas y enfermedades en el MSRD. ....	55
Tabla 5. Cantidad de parámetros necesarios en cada modelo si se tienen m=100 enfermedades binarias, n=200 síntomas binarios y r=10 síntomas relevantes. ....	57
Tabla 6. Comparación entre Sistemas Expertos basados en Reglas y Sistemas Expertos basados en Probabilidad.....	61



# Monografía sobre Sistemas Expertos



**Bargiela Roberto Alejandro**

Cátedra Sistemas Operativos  
Dpto. Informática  
Licenciatura en Sistemas de Información  
Facultad de Ciencias Exactas y Naturales y Agrimensura  
Universidad Nacional del Nordeste. Corrientes Capital. Argentina.  
E-mail: alejandro\_1107@yahoo.com.ar

**Resumen.** Muchos son los problemas con los que se enfrenta el ser humano día a día; desde la aparición de la computadora esos problemas se han ido simplificando, ahora, con la ayuda de la Inteligencia Artificial (IA) se han simplificado aún más. La IA, entre todos sus aportes, sin duda alguna, el que mayor aceptación y aplicación ha tenido ha sido el desarrollo de Sistemas Expertos. Estos sistemas tienen su principal utilización en la medicina sin que esto sea un condicionante o limitante, ya que sus posibilidades son numerosas gracias al aporte que realizan otras ciencias como las matemáticas o, inclusive, la propia IA por medio de las Redes Neuronales. La posibilidad de conjugar en un mismo sistema varias técnicas, herramientas, en general, recursos, hace que los Sistemas Expertos sean potencialmente capaces de solucionar casi cualquier problema.

**Palabras Claves:** Sistemas Expertos, Inteligencia Artificial, Motor de Inferencia, Base de Conocimientos.

## Introducción

### 1. Introducción a la Inteligencia Artificial

Muchos piensan que la Inteligencia Artificial es crear computadoras o robots inteligentes, es decir, que posean características humanas, como pueden ser; el uso de razón, el pensar, el reconocer la voz y las imágenes, etc. En casos extremos alguno tienen el sueño de que algún día las computadoras tengan sentimientos.

La realidad es otra, si bien, la meta final o el objetivo implícito es conseguir que una computadora realice lo expresado en el párrafo anterior, hoy en día se está muy lejos de lograr muchas de esas cosas.

Para aclarar este tema, a continuación se presentan diferentes definiciones de Inteligencia Artificial con el objetivo de que el lector saque sus propias conclusiones.

## 1.1. Definiciones de IA

En algunos textos este tema se trata primero definiendo la IA y en otros se define primero qué es inteligencia y los distintos tipos de la misma y luego adaptar este concepto para conseguir la Inteligencia Artificial, éste último es el que adopta el presente trabajo.

### 1.1.1. La inteligencia

La palabra **inteligencia** proviene del latín, *intelligentia*, que significa la capacidad de entender o comprender. Sin embargo este termino tiene su origen en otro vocablo latino, *legere*, que significa "coger" o "escoger". He aquí que *intelligere* toma el significado de reunir elementos, escoger entre ellos y formar ideas, comprender y conocer.

Cuando se dice que un individuo es *inteligente* se hace referencia a que posee intuición, inspiración, capacidad para organizar cadenas lógicas de pensamiento, sentimientos y expresión lingüística, entre otras cosas [5].

Según el diccionario el término **Inteligencia** adquiere los siguientes significados:

**Definición.** Facultad de comprender, de conocer. Para la psicología es la aptitud para relacionar las percepciones sensoriales o para abstraer y asociar conceptos. Conocimiento. Habilidad. Avenencia.

**Definición.** La psicología experimental cuantifica la inteligencia y determina los criterios para objetivar dicha medida. La discusión central sobre la estructura y funcionamiento de la inteligencia gira en torno a si se considera a ésta como un factor general, base de las funciones específicas, o si de las funciones se deduce una capacidad global. Los primeros tests psicométricos fueron presentados por A. Binet y Th. Simon (1905). Otras investigaciones valiosas han sido las de Spearman, Thrustone, Guilford, Piaget, etc.

Además de estas definiciones, existen otras que son más bien lo que el humano cree que es la inteligencia, como por ejemplo se puede decir que la inteligencia es la capacidad de resolver problemas, de memorizar, de utilizar los conocimientos adquiridos para el logro de alguna meta, etc.

La inteligencia se puede clasificar en varios tipos, sin embargo, no son inteligencias distintas sino manifestaciones de una inteligencia cognitiva, es decir, por el propio conocimiento.

La clasificación admite los siguientes tipos:

1. Verbal.
2. Espacial.
3. Lógica - matemática.
4. Musical.

5. Intrapersonal.
6. Interpersonal.
7. Corporal - cinética.

Los distintos tipos se relacionan en cierta forma con las aptitudes que posee el individuo; algunas de estas son: verbales, numéricas, espaciales, mnemotécnicas, perspectivas, lógicas y psicomotoras.

Según Robert Sternberg, Psicólogo de la Universidad de Yale de Estados Unidos, la **inteligencia cognitiva** es la que se puede medir con los tests; la **inteligencia social** se refiere a la capacidad que posee el individuo de adaptarse al entorno que lo rodea y la **inteligencia experimental** es la aptitud para sacar el mejor provecho de las experiencias acumuladas por el individuo en el momento en que se enfrenta a problemas nuevos.

Está claro, a esta altura, que quizás la diversidad de definiciones o conceptos sobre qué es la inteligencia cubre un rango muy amplio, o bien, son diversas visiones de una misma cosa, sea como fuere, la inteligencia artificial todavía está lejos de cubrir todas estas definiciones.

A continuación se completarán las definiciones de inteligencia diciendo:

**Definición.** La inteligencia es un núcleo común de atributos que supuestamente tienen que ver con una actividad inteligente, como; la habilidad para resolver problemas nuevos, la planificación, la facilidad de adquirir nuevos conocimientos, la abstracción y la capacidad de adaptación.

### 1.1.2. El pensamiento

El pensamiento es utilizado por el hombre para resolver los problemas que se le presentan y se lo define como la capacidad de analizar una situación en base a lo percibido del medio o entorno.

En la mayoría de las personas se presenta un pensamiento lineal, es decir, actúan paso a paso, en secuencia, por lo que se ocupan de aspectos parciales del problema, pero nunca en conjunto.

A este grupo de personas les falta la capacidad de pensar de forma global, de comprender al problema como un todo compuesto por una intrincada red de distintas partes relacionadas entre sí.

Primero, para resolver un problema, se debe saber ¿qué es un problema?; según el psicólogo alemán K. Duncker, un problema:

**Definición.** [...] Parte de un **estado inicial no deseado** y debe llegar a un **estado final deseado**, entre ambos existe **al menos una barrera** que bloquea el paso de uno a otro.

Algunos expertos diferencian tres clases de problemas; los de interpretación, los de vinculación y los de lenguaje.

Los de la primer clase son del tipo en que se conoce el punto de partida y el destino pero se desconoce el camino más corto o rápido de llegar; el test de las Torres de Hanoi explica la solución de este tipo de problemas, según dicen los psicólogos.

La segunda clase de problemas son aquellos en que se conoce el objetivo pero no se sabe de donde partir para encontrar la solución. Para resolver esta clase de problemas es necesario creatividad, tener un punto de vista totalmente nuevo.

Y por ultimo, los problemas de lenguaje, son los más frecuentes en la vida diaria, pero a la vez son los más difíciles de resolver, puesto que sólo se pueden describir de forma muy vaga. Algunos ejemplos; ¿cómo se puede amoblar una vivienda para que quede más bonita o cómoda?, ¿cómo se proyectan las vacaciones más interesantes para el año siguiente?. Lo primero es definir los conceptos de bonito, cómodo o interesante; de esta forma se está en camino a la solución.

Se realizaron varios experimentos en los EE.UU. para encontrar la forma en que el ser humano piensa y también en animales; las conclusiones obtenidas son que los seres humanos no poseen una única forma de pensar, y por lo tanto, una única forma de resolver los problemas. Por otro lado los animales parecen resolver sus problemas por prueba y error.

Hoy en día resolver problemas al modo humano parece ser un nuevo desafío, ya que los programadores cuando intentan reproducir en una computadora los procesos del pensamiento humano tienen que conocer a la perfección la técnica para solucionar problemas. Pero es en ese punto donde los informáticos entran en una zona de incertidumbre.

Antes de la era de las computadoras los filósofos y psicólogos dividían el trabajo de pensar en dos métodos diferentes: inductivo y deductivo, es decir, argumentación y conclusión.

La deducción era la herramienta clásica utilizada por el personaje de novela, Sherlock Holmes. Este método consiste en ver objetivamente los hechos, seguir principios fijos y combinar todo esto para obtener una conclusión.

La inducción, por el contrario, es el método utilizado por la investigación científica. Se parte de algo concreto para desmenuzarlo hasta hallar las últimas causas sobre todo a través de la experimentación y así establecer leyes o reglas irrefutables.

### **1.1.3. Inteligencia Artificial**

Hasta aquí solo se ha hablado de inteligencia y no de Inteligencia Artificial, bien, ahora se expondrán las definiciones recopiladas sobre IA.

El término "*inteligencia artificial*" se acuñó en Dartmouth, en 1956, durante una conferencia convocada por McCarthy a la cual asistieron, entre otros, Minsky, Newell y Simon. Debido a que la inteligencia artificial tuvo muchos padres no hay un consenso para definir ese concepto.

La definición del diccionario dice:

**Definición.** Concepto que engloba todas las tecnologías que estudian la creación de máquinas (robots, autómatas, etc.), y también los programas que se ejecutan siguiendo un método parecido a la inteligencia humana (traducciones, juegos de ajedrez, etc.).

A. Barr y E. A. Feigenbaum definen, en su libro "*Barr y Feigenbaum*" de 1981 (en la página 4), a la IA como sigue:

**Definición.** La Inteligencia Artificial es la parte de la Ciencia que se ocupa del diseño de sistemas de computación inteligentes, es decir, sistemas que exhiben las características que asociamos a la inteligencia en el comportamiento humano que se refiere a la comprensión del lenguaje, el aprendizaje, el razonamiento, la resolución de problemas, etc.

También puede definirse a la IA como:

**Definición.** Métodos y algoritmos que permitan comportarse a las computadoras de modo inteligente.

**Definición.** Estudio de las facultades mentales a través del uso de modelos computacionales.

**Definición.** Desarrollo de autómatas (máquinas construidas por el hombre) para simular el proceso deductivo o inductivo humano de adquisición y aplicación de conocimiento.

El punto fundamental de la IA es que los procesos que se llevan a cabo en el cerebro pueden ser analizados, a un nivel de abstracción dado, como procesos computacionales de algún tipo.

McFarland establece que la inteligencia únicamente se la puede medir por el resultado, es decir, se puede analizar si el comportamiento es o no inteligente.

Básicamente la IA se implementa, en su gran mayoría, mediante modelos matemáticos de algún tipo, es decir, se trata de representar o simplificar un problema mediante un modelo; el modelo es la abstracción que se realiza para comprender o manejar algo complejo, obteniendo algo más simple y a la vez representable en un algoritmo computacional.

La IA posee dos paradigmas: el conexionista y el simbólico. En el primero se hace referencia a la forma en que las neuronas se conectan entre sí logrando el funcionamiento del cerebro; mientras que el otro, trata de la representación del conocimiento a través de símbolos.

## 1.2. Evolución histórica

La IA nació en los años cincuenta, llamándose, al período comprendido entre los años 1950 y 1965, subsimbólico. En este período se utilizaron representaciones numéricas (o subsimbólicas) del conocimiento. Aunque en la mayoría de la bibliografía se enfatizan los trabajos realizados con redes neuronales, es cierto, que también fueron importantes los algoritmos evolutivos o genéticos en el mismo período.

La primera generación (hasta 1955) estableció las bases lógicas y los primeros modelos matemáticos de esta disciplina.

Alan Turing, en 1950, publicó *Computing Machinery and Intelligence* donde sugiere: "*machines may someday compete with men in all purely intellectual pursuits.*"

Claude Shannon, en 1950, publicó "*A Chess-Playing Machine*" en *Scientific American* analizando el problema del juego automático de ajedrez (10<sup>120</sup> movimientos posibles).

En 1953, publicó "*Computers and Automata*" con nuevas y sugestivas interrogantes las cuales se refieren a si una máquina sería, en algún momento, capaz de realizar las siguientes tareas:

1. Localizar y reparar sus averías.
2. Programarse o reprogramarse a sí misma.
3. Aprender.

Warren McCulloch and Walter Pitts (1943) primer modelo matemático de red neuronal en "*A Logical Calculus of the Ideas Immanent in Nervous Activity*". Este modelo consistía en una red de neuronas binarias y sinapsis. Este modelo es esencialmente equivalente a una maquina de Turing.

En la segunda generación, el nombre **Inteligencia Artificial** fue lanzado por John McCarthy (desarrollador del Lenguaje LISP) en el encuentro "*Summer Research Project*" realizado en Dartmouth College (Hanover, New Hampshire, EE.UU) en 1956. Este encuentro sirvió como catalizador para los investigadores que estaban interesados en las posibilidades de la inteligencia de la máquina.

Por esta misma época, Newell, Simon y Shaw desarrollaron el "*Logic Theorist*", sistema capaz de probar teoremas de Lógica matemática tomados de la obra "*Principia Matemática*" de Whitehead y Russell.

"*Logic Theorist*" fue uno de los primeros programas eficientes desarrollado Allen Newell, Herbert Simon y J.C. Shaw. Fue capaz de probar 38 de los primeros 52 teoremas del Capítulo 2 del libro *Principia Mathematica* de Whitehead y Russell.

"Mycin" fue el pionero de los sistemas expertos (desarrollado por Edward Shortliffe). Puede diagnosticar infecciones bacterianas a partir de síntomas.

En 1957 Allen Newell predijo que en un plazo de 10 años, un ordenador sería el campeón del mundo de ajedrez.

Las características de los modelos y algoritmos "standard" de IA son:

1. El conocimiento se representa explícitamente usando reglas, redes semánticas, modelos probabilísticos, etc.

2. Se imita el proceso humano de razonamiento lógico para resolver los problemas, centrando la atención en las causas que intervienen en el problema y en sus relaciones (encadenamiento de reglas, inferencia probabilística).
3. Se procesa la información secuencialmente.

### **1.3. Aplicaciones**

Existen diferentes aspectos de la vida humana y áreas del continente científico en que es posible aplicar algunas o todas las técnicas de la IA para la resolución de problemas.

Algunas de las aplicaciones de la IA son:

1. Percepción y reconocimiento de imágenes (formas y colores).
2. Comprensión, generación y traducción del lenguaje natural.
3. Control de robots.
4. Juegos: ajedrez, damas, del tipo estratégicos, simulación, etc.
5. Matemáticas: calculo simbólico, demostración de teoremas.
6. Computación: verificación de programas, aprendizaje automático.
7. Ingeniería: diseño, detección de fallos, planificación de manufacturación.
8. Análisis científico.
9. Diagnóstico médico y técnico.
10. Análisis financiero, entre otros.

### **1.4. Áreas o campos de la IA**

Éstas áreas o campos hacen referencia a las subtareas que se engloban dentro de la IA y dependiendo del paradigma utilizado para el desarrollo del modelo se clasifican de diferente manera:

En el Paradigma Conexionista o Subsimbólico los más reconocidos son las Redes Neuronales Artificiales y los Algoritmos Genéticos.

Por otro lado, en el Paradigma Simbólico se encuentran los Sistemas Expertos, los programas de resolución de problemas matemáticos o los programas de juegos.

Sin embargo existen muchas otras técnicas utilizadas en la IA, pudiendo caer en discusión su clasificación.

A continuación se describen brevemente las técnicas más significativas y de mayor aplicación en la actualidad.

### **1.4.1. Data Mining o Minería de Datos (DM o MD)**

Muchos autores no la consideran como incluida dentro de la IA pero en su aplicación se encuentra el uso de redes neuronales artificiales y sistemas expertos entre otras técnicas estadístico-matemáticas.

La MD se basa en el concepto que su nombre propone; y es la "extracción" de datos de diferentes bases de datos, su análisis y procesamiento para luego presentar la información obtenida al que haya realizado la consulta.

Las bases de datos las cuales serán fuente (haciendo una analogía serían las minas) de los datos pueden o no estar relacionadas entre sí y si no lo estuvieran el sistema que implementa la MD debería relacionarlas para obtener los datos que se soliciten.

### **1.4.2. Redes Neuronales Artificiales (RNA)**

Al inicio de la IA, el conocimiento se representaba usando reglas, redes semánticas, modelos probabilísticos, etc. La metodología para la resolución de problemas era el uso del proceso humano de razonamiento lógico relacionando las causas del problema con los conocimientos que se poseían y luego se procesaba la información secuencialmente.

Al avanzar la IA, aparecieron problemas complejos en los que no era posible representar el conocimiento de manera explícita ni procesarlo utilizando el razonamiento lógico.

La posible solución a este inconveniente era utilizar estructuras computacionales paralelas basadas en redes neuronales biológicas; así surgen las redes neuronales artificiales.

Es sabido que el cerebro está compuesto por células especiales llamadas neuronas las cuales se comunican entre sí (se interconectan) a través del **axón**, esta conexión es llamada **sinapsis**. La neurona natural contiene un potencial interno continuo llamado potencial de membrana. Cuando este potencial supera un cierto valor **umbral**, la neurona está en condiciones de transmitir su potencial a la siguiente neurona en la red mediante el axón.

Las RNA se estructuran como un grupo numeroso de procesadores conectados entre sí y trabajando en paralelo.

### **1.4.3. Algoritmos Evolutivos o Algoritmos Genéticos (AG)**

Los AG se han desarrollado con la intención de optimizar poblaciones o conjuntos de soluciones posibles de un determinado problema mediante la utilización de la genética biológica.

La genética es la ciencia biológica que estudia la variabilidad y la herencia de los seres vivos [5] analizando el material genético a partir del cual nace todo ser vivo; este material genético se componen del ADN (ácido desoxirribonucleico) que se autoduplica en cada división celular. Esta ciencia es la base de para los AG ya que estos intentan, de alguna manera, representar un dato (que puede ser una posible solución) mediante una cadena de bits la cual, en analogía con la genética, sería un cromosoma.

Una vez representados todos los datos (posibles soluciones) se cruzan, mutan o recombinan entre sí, con lo cual se obtienen nuevos datos o cromosomas que pueden o no ser mejores que los anteriores. De ser mejores, este paso se repite hasta la obtención de nuevos cromosomas cuya mejora con respecto a los anteriores sea insignificante por lo que el proceso de evolución concluye con un nuevo grupo o población de datos (en la mayoría de los casos se intenta llegar a una sola solución, dato o cromosoma) que son interpretados como las mejores soluciones posibles del problema que se ha intentado solucionar.

### **1.4.4. Sistemas Expertos (SS.EE.)**

En una primera aproximación se puede decir que los SS.EE. son sistemas compuestos por dos partes componentes, principalmente; una llamado Motor de Inferencias y la otra Base de Conocimientos. En la primera se representa el procedimiento lógico que sigue un experto en determinada área del conocimiento y en la segunda se representa el propio conocimiento del experto, pudiendo ser uno sólo o un grupo de expertos dentro de la mismo área, inclusive pueden ser libros especializados en el tema de interés.

Este tipo de sistemas se desarrolla con los objetivo de imitar la inteligencia humana, en este caso de un experto, y de tener asistencia de un experto en cualquier lugar en poco tiempo con respecto al tiempo que tarda en formarse un profesional humano incluyendo el tiempo de experiencia.

Uno de los requisitos para realizar un SS.EE. es poseer el conocimiento de uno o varios expertos con amplia experiencia en el campo del conocimiento que les compete, ya que es la experiencia quien agrega valor al conocimiento. Un SS.EE. que se basa en los conocimientos de un novato poco puede aportar a un profesional que, si bien, no es un experimentado tampoco es un novato, pero que necesita la asistencia o los conocimientos de un experto que lo guíe en su búsqueda de una solución a un problema complejo o nuevo para el profesional.

Existen diferentes tipos de SS.EE. clasificados según el modelo que utilizan, siendo algunos de éstos los siguientes:

1. SS.EE. basados en Reglas.
2. SS.EE. basados en Probabilidad.

3. SS.EE. basados en Redes Neuronales Artificiales.
4. SS.EE. basados en Lógica Difusa.

### 1.4.5. Agentes Inteligentes

Un Agente Inteligente es:

**Definición.** Todo aquello que puede considerarse que percibe su ambiente mediante sensores y que responde o actúa en el ambiente por medio de efectores [4].

Martin Fischles y Oscar Firschein en 1987 describieron los atributos del agente inteligente, los cuales son:

1. Tener actitudes mentales tales como creencias e intenciones.
2. Tener la capacidad de obtener conocimiento, es decir, aprender.
3. Poder resolver problemas, incluso particionando problemas complejos en otros más simples.
4. Entender. Poseer la capacidad de crearle sentido, si es posible, a ideas ambiguas o contradictorias.
5. Planificar, predecir consecuencias (ser omnisciente<sup>1</sup>), evaluar alternativas (como en los juegos de ajedrez).
6. Conocer los límites de sus propias habilidades y conocimientos.
7. Poder distinguir a pesar de la similitud de las situaciones.
8. Poder ser original, creando incluso nuevos conceptos o ideas, y hasta utilizando analogías.
9. Poder generalizar.
10. Poder percibir y modelar el mundo exterior.
11. Poder entender y utilizar el lenguaje y sus símbolos.
12. Poder medir el desempeño de sus actos, realizando acciones que, se espera obtendrán el mejor desempeño con respecto a las acciones tomadas anteriormente de las cuales se tiene conocimiento de los resultados obtenidos.

En la definición publicada por Russel y Norving en 1996, describen a un Agente Inteligente como aquel que realiza lo correcto [4].

---

<sup>1</sup> Omnisciencia: conocimiento de todas las cosas reales y posibles.[3]

Un agente inteligente se lo considera autónomo cuando el tipo de acciones que lleva a cabo depende de su propia experiencia y no de los conocimientos que le fueron programados en tiempo de diseño.

El agente inteligente, al ser autónomo debe funcionar de la mejor manera posible dentro de los límites aceptables en grupo numeroso de ambientes teniendo en cuenta que se le ha otorgado el tiempo suficiente para adaptarse. Por otra parte el agente racional ideal es aquel que siempre emprende las acciones que mejor desempeño produzcan de acuerdo a las percepciones que recibe basándose en las percepciones y los conocimientos que tal agente ha incorporado.

La estructura de un agente se compone de una arquitectura y un programa. La arquitectura puede ser una computadora sencilla o hardware especial como cámaras de video, micrófonos, procesadores de imágenes y sonidos, etc.; el programa o software es el que contiene las acciones que deberá realizar el agente en cada tipo de secuencias de percepciones obtenidas del ambiente y que retroalimentará al agente para efectuar nuevas acciones.

## **2. Introducción a los Sistemas Expertos**

### **2.1. Evolución histórica**

Para entender el porqué y cuándo nacen los SS.EE. es necesario conocer un poco de historia sobre la IA, de lo cual ya se ha mencionado algo.

Los SS.EE. se desarrollaron al reconocerse la importancia primordial que tenían los conocimientos particulares de un dominio para que los métodos formales de búsqueda e inferencia sean eficaces en la solución de problemas.

En los inicios de los años '40 surgió un modelo formal de neurona, desarrollado por Warren McCulloch y Walter Pitts, publicado en "*A Logical Calculus of the Ideas Immanent in Nervous Activity*" en 1943.

El modelo se llamaba PERCEPTRON y consistía en un red de neuronas binarias, es decir, representaba la actividad nerviosa como un fenómeno "todo o nada", que se podía calcular con una lógica de proposiciones. Este modelo fue considerado el comienzo de las Redes Neuronales Artificiales.

Más tarde surgió la Cibernética, formulada en 1948 por el matemático Norbert Wiener (1894-1964); esta disciplina estudia los fenómenos de comunicación y control, de manera general, aplicable a sistemas biológicos, organizacionales sociales o a máquinas.

En 1956 se acuña el termino "**Artificial Intelligence**" en el encuentro "*Summer Research Project*" realizado en Dartmouth College en Hanover, New Hampshire, EE.UU. Los responsables de estas palabras y de la organización de la conferencia fueron John McCarthy y Marvin Minsky; el primero de ellos creó el lenguaje LISP (List Processing, cuya traducción es procesamiento de listas), en el Massachusetts Institute of Technology (MIT), justamente para su aplicación en la IA.

El lema que utilizó John McCarthy, para fundar las bases modernas de esta disciplina, fue el siguiente:

**Definición.** [...] To proceed on the basis of the conjecture that every aspect of learning or any other feature of the intelligence can in principle be so precisely described that a machine can be made to simulate it.

El cual explica que si cada aspecto del aprendizaje o de la inteligencia pudiera ser descrito con tal precisión, es decir, formalizar esos aspectos, una máquina puede ser construida para simular dichos aspectos.

Aproximadamente por la misma época Allen Newell, Herbert Simon y J.C. Shaw desarrollaron un sistema "*Logic Theorist*" que fue capaz de probar 38 de los 52 teoremas del Capítulo 2 del libro "*Principia Mathematica*" de Whitehead y Russell.

En la década de los '60, surgió un sistema llamado DENDRAL cuya función era determinar estructuras moleculares de compuestos orgánicos. Este sistema fue desarrollado por Lindsay, Buchanan, Feigenbaum y Lederberg.

En esta época los investigadores se centraban en los procesos de lógica y búsqueda, y en métodos de gran generalidad pero de poca utilidad como el "*General Problem Solver*" (GPS).

El GPS (Ernst y Newell, 1969) tenía como base la idea de un solo esquema para muchas aplicaciones. Todo problema debía ser formulado como la búsqueda de unos *estados objetivo* o *finés* en un *espacio de estados*, con la posibilidad de aplicar *operadores* o *medios* para pasar de un estado a otro. Este enfoque también se conoce como *análisis medios-finés*.

A principios de los '70 el sistema "*STRIPS*" (Fikes y Nilsson, 1971) aplicaba la metodología del probador de teoremas a la solución de otros problemas.

Hasta este momento los sistemas de este tipo eran llamados *Sistemas con Base de Conocimientos*, sin embargo, el sistema DENDRAL es considerado por algunos autores como el primer S.E., aunque este nombre aún no era conocido ni utilizado.

Finalmente en 1977, Edward Feigenbaum, en el Congreso Internacional de Inteligencia Artificial (IJCAI '77, International Joint Conference on A.I.) expuso la definición de **Sistemas Expertos** llamándolos así por primera vez.

## 2.2. Definiciones

Líneas atrás se menciona la primera vez que se usó el término S.E.; en esa ocasión la primera definición dada por E. Feigenbaum, en 1977, decía lo siguiente:

**Definición.** Sistema Experto puede definirse como un programa inteligente que utiliza conocimiento y procedimientos de inferencia para resolver problemas que son lo bastante complejos como para requerir Expertos Humanos en su resolución.

En la literatura existente se pueden encontrar muchas definiciones de sistema experto. Por ejemplo, Stevens (1984), da la definición siguiente:

**Definición.** Los sistemas expertos son máquinas que piensan y razonan como un experto lo haría en una cierta especialidad o campo. [...] Un Sistema Experto de verdad, no sólo realiza las funciones tradicionales de manejar grandes cantidades de datos, sino que también manipula esos datos de forma tal que el resultado sea inteligible y tenga significado para responder a preguntas incluso no completamente especificadas.

Entre otras definiciones más actuales se encuentran las siguientes [2]:

**Definición.** Sistema Experto es un conjunto integrado por mecanismos electrónicos que reproducen de un modo programado los procesos de aprendizaje, memorización, búsqueda, razonamiento, comunicación y ejecución de tareas propias del ser humano, con el objeto de imitar al Experto Humano en un determinado campo de especialización.

**Definición.** Sistema Experto puede definirse como un sistema informático (hardware y software) que simula a los Expertos Humanos en un área de especialización dada.

Como síntesis de todas estas definiciones pueden extraerse los siguientes conceptos: experto humano, conocimientos en una determinada área o campo de especialización, sistema informático compuesto por hardware y software, procedimientos de inferencia, resolución de problemas, etc.

Es de mucha importancia entender estos conceptos y sus alcances para comprender cómo se compone un S.E. y su funcionamiento.

## 2.3. Aplicaciones

En los últimos años se ha visto cómo la informática ha crecido a pasos agigantados no sólo en desarrollo de tecnología sino en posibles aplicaciones de la misma y aún más, es sorprendente cómo la IA ha ganado espacio en el mundo utilizándose en campos que no se relacionan con la informática; algunos de éstos, son [2]:

1. Medicina.
2. Geología.
3. Química.
4. Economía.

## 5. Ingeniería civil.

Pero no para todos los problemas que se presentan en estas áreas es efectiva la aplicación de los SS.EE. sino para aquellos problemas que sean de alguno de los siguientes tipos:

1. Clasificación o interpretación.
2. Diagnóstico: médico o técnico.
3. Predicción y previsión.
4. Diseño y configuración de equipos o sistemas.
5. Planeación.
6. Monitoreo y control.
7. Vigilancia y alarma.
8. Instrucción.

Es necesario aclarar que alguno de los tipos recién descritos se encuentran muy estrechamente relacionados por lo que los SS.EE. que puedan aplicarse a uno de ellos también podrían ser aplicados a otro, por la relación existente.

Estos problemas también pueden ser clasificados según su naturaleza, en *deterministas* ó *estocásticos*. El primero grupo está conformado por aquellos problemas que se pueden plantear utilizando un conjunto de reglas, mientras que los del segundo grupo manejan situaciones inciertas y necesitan incorporar en su planteo la medida intuitiva de incertidumbre que es la probabilidad [2].

## **2.4. Composición de un SS.EE. e interrelación entre sus partes**

### **2.4.1. Elementos de un SS.EE.**

Como todo sistema, un S.E. se compone de varias partes componentes o subsistemas, los cuales son:

1. Base de Conocimientos (BC).
2. Expertos Humanos e Ingenieros del Conocimiento.
3. Subsistema de Control de Coherencia.
4. Subsistema de Adquisición del Conocimiento.
5. Memoria de Trabajo.
6. Motor de Inferencia (MI).

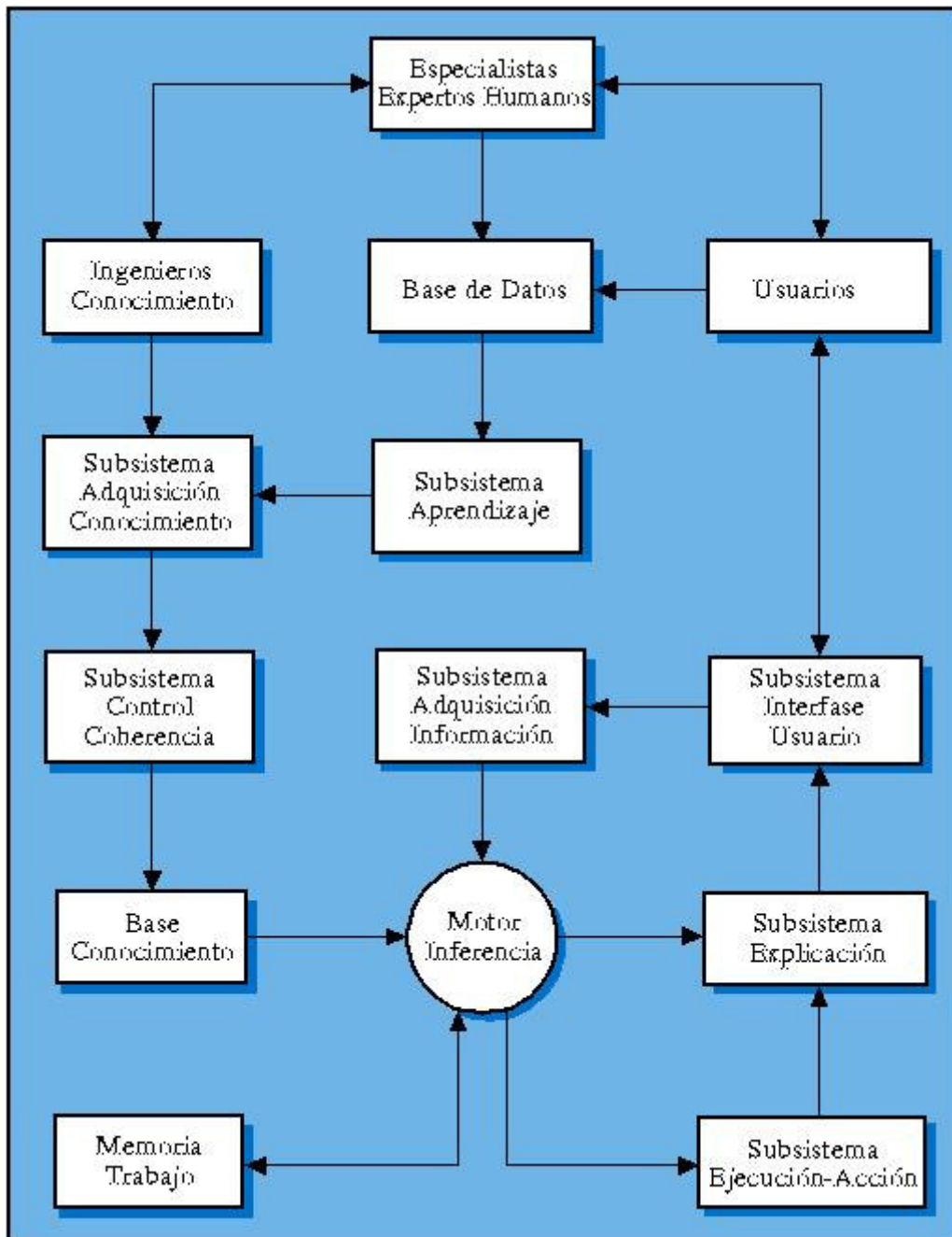
7. Subsistema de Demanda de Información.
8. Interface de Usuario.
9. Interface con el Desarrollador.
10. Interface con otros Sistemas.
11. Subsistema de Incertidumbre.
12. Subsistema de Ejecución de Tareas.
13. Subsistema de Aplicación.
14. Experiencia.
15. Subsistema de Aprendizaje.

Estos componentes se esquematizan en el **Gráfico 1**.

La **BC** es la representación del conocimiento permanente o abstracto, por medio de reglas o espacios probabilísticos.

El conocimiento se refiere a afirmaciones de validez general, por ejemplo, los síntomas, las enfermedades y las relaciones entre ellos forman parte del conocimiento.

Mientras los datos son efímeros (temporales), el conocimiento es permanente. Los datos son almacenados en la Memoria de Trabajo al igual que todos los procedimientos de los diferentes sistemas y subsistemas que son de carácter transitorio.



**Gráfico 1.** Componentes básicos de un S.E.; cada flecha representa el flujo de la información.

Los *Expertos Humanos* son los especialistas en el tema de estudio que aportan su experiencia (conocimiento) en una forma ordenada y estructurada, además de un conjunto de relaciones bien definidas y explicadas para que sea, todo, almacenado en la BC; mientras que los *Ingenieros del Conocimiento* trasladan esa BC a un lenguaje que sea inteligible por el S.E..

El *Subsistema de Control de Coherencia* previene el ingreso de conocimientos incoherentes, es decir, controla la consistencia de la base de datos y evita que unidades de conocimiento inconsistentes entren en la misma. Cuando se solicita información de los expertos humanos, este subsistema informa sobre las restricciones que debe cumplir la información requerida para que sea coherente con el conocimiento existente en la BC.

El *Subsistema de Adquisición de Conocimiento* controla el flujo de nuevos conocimientos provenientes del experto humano y que serán almacenados en la base de datos. El sistema determina que nuevos conocimientos son necesarios o si los conocimientos recientes son nuevos, si la respuesta es afirmativa entonces los almacena en la base de datos si debieran incluirse en la misma.

La *Memoria de Trabajo*, también llamada *tablero*, *pizarra* o *pizarrón* cuando el S.E. se compone de varios módulos (cada uno un S.E. especialista en un determinado nivel de detalle), es el lugar donde se almacena el conocimiento transitorio o concreto. Es un área de memoria reservada para la descripción del problema actual, tal como lo especifican los datos de entrada; también se usa para registrar resultados intermedios.

Conserva hipótesis y decisiones intermedias de tres tipos:

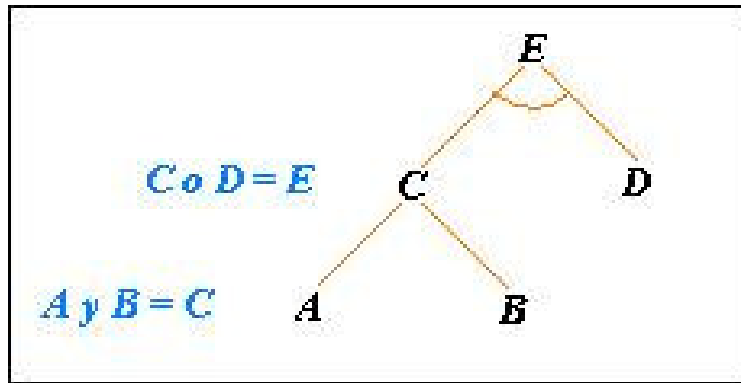
1. **Plan.** Cómo atacar el problema.
2. **Conciencia.** Acciones potenciales que esperan ejecución.
3. **Solución.** Hipótesis candidatas y cursos de acción alternativos que el sistema ha generado hasta el momento.

El MI es, en analogía con el ser vivo, el cerebro del SE. Básicamente es un programa que proporciona la metodología para razonar con la información en la BC y la Memoria de Trabajo hasta llegar a conclusiones satisfactorias.

Las conclusiones arribadas por el MI están basadas en conocimiento determinista (hechos) ó en conocimiento probabilístico (situaciones de incertidumbre). En los casos en los que se dispone de información aleatoria o difusa, el MI es responsable de la propagación de este conocimiento incierto. De hecho, en los SS.EE. basados en probabilidad, la propagación de incertidumbre es la tarea principal de el MI.

El esquema de organización del razonamiento a seguir por el MI para resolver los problemas que se le presentan puede ser alguno de los siguientes:

1. Esquema *hacia atrás*: se razona desde un objetivo o hipótesis hacia los datos; este esquema es muy utilizado en los SS.EE. utilizados para diagnóstico.
2. Esquema *hacia adelante*: el razonamiento se realiza desde los datos hacia un objetivo.
3. Esquema *oportunistico*: se aplica alguno de los esquemas anteriores según sea oportuno, este esquema se adapta muy bien en los sistemas con tablero (S.E. modular en el que cada modulo es a su vez un S.E. y sus Memorias de Trabajo son llamadas Tableros). El proceso de inferencia se lo puede representar mediante árboles donde los nodos representan hechos. La conjunción se esquematiza mediante un arco uniendo las premisas (Ver **Gráfico 2**).



**Gráfico 2.** Ejemplo de la representación gráfica en árbol de dos reglas en un proceso de inferencia.

El *Subsistema de Demanda de Información* completa conocimientos en la inferencia para obtener conclusiones válidas. Este componente se hace necesario debido a que hay situaciones en las que el conocimiento almacenado no es suficiente para sacar una conclusión satisfactoria por lo que es necesario implementar un mecanismo para obtener los conocimientos necesarios para continuar con el proceso de inferencia hasta que se haya sacado una conclusión aceptable. En algunos casos, el usuario proveerá estos conocimientos por lo que se hace necesaria una interfaz donde el usuario se comunique con el sistema.

La *Interface de Usuario* es la *cara* del sistema por lo que debe ser agradable y estar en un lenguaje comprensible para el usuario acompañando con gráficos si es necesario. Gracias a la interface de usuario el sistema podrá solicitar información adicional para el proceso de inferencia por lo que ésta debe ser lo más clara y amigable posible para lograr una buena comunicación entre el usuario y el sistema.

También hay que tener en cuenta que el sistema no sólo solicita datos e información sino que debe mostrarla al usuario para que éste pueda tomar buenas decisiones en tiempo y forma.

La *Interface con el Desarrollador* y la *Interface con otros Sistemas* tienen las mismas características que en el caso anterior diferenciándose de éste por el tipo de lenguaje empleado para comunicarse con el desarrollador y con otros sistemas, respectivamente, ya que en cada caso la información mostrada y/o solicitada será distinta según el caso.

El *Subsistema de Incertidumbre* se encarga de almacenar la información del tipo incierta y propagar la incertidumbre asociada a esa información. Se mencionó en párrafos anteriores que este subsistema es muy utilizado en los SS.EE. basados en probabilidad, y esto se debe a que en algunos casos se busca una conclusión partiendo de datos inciertos o difusos, por ejemplo, en un S.E. para el diagnóstico médico el paciente debe proporcionar los síntomas que presenta, pero éste no reconoce bien cuáles son sus síntomas por lo que los datos (síntomas) son inciertos o difusos.

El *Subsistema de Ejecución de Tareas* permite la realización de acciones en base a las conclusiones obtenidas por el motor de Inferencia. En SS.EE. aplicados al control, a la vigilancia o a las transacciones bancarias, entre otras, pueden ejecutar acciones como detener o retrasar el flujo de las tareas que se estén controlando (tráfico, cadenas de montaje, etc.), bloquear entradas en caso de detectarse un intruso, o expender dinero, confiscar tarjeta, etc. ( en cajeros automáticos).

El *Subsistema de Explicación* es el que brinda una explicación o justificación de la conclusión obtenida. Este subsistema es utilizado por el S.E. para explicar los siguientes interrogantes:

1. ¿Cómo llego a determinada conclusión?.
2. ¿Para qué solicita determinada información?.
3. ¿Porqué no llegó a otra determinada conclusión?.
4. ¿Qué falta determinar antes de poder llegar a una conclusión?.

Además de estas cuestiones es también necesario explicar o justificar las acciones realizadas por el subsistema de ejecución de tareas.

Esta parte es fundamental para que el sistema sea confiable desde el punto de vista del experto, del usuario y del ingeniero del conocimiento.

El *Subsistema de Aprendizaje*, el aprendizaje ayuda a los expertos a mejorar su calidad profesional al aprender de las nuevas experiencias aumentando su eficacia para los futuros problemas. Esto es altamente deseable también en los SS.EE. por lo que el Subsistema de Aprendizaje es el programa encargado de analizar las razones de sus éxitos y fracasos para perfeccionar la BC y ser más eficaz y efectivo en el futuro, como así también incluir nuevos conocimientos.

Se distinguen dos tipos de conocimientos: el *estructural* y el *paramétrico*. El primero hace referencia a los aspectos relacionados con la estructura del conocimiento (reglas, distribuciones probabilísticas, etc.), un ejemplo de éste es el descubrimiento de nuevos síntomas relevante para una enfermedad o la inclusión de una nueva regla en la BC. En el segundo tipo se estiman los parámetros necesarios para construir la BC, en este caso un ejemplo es la estimación de frecuencias o probabilidades asociadas a síntomas o enfermedades.

## **2.4.2. Interrelación entre los componentes**

La relación entre los componentes de un S.E. mantiene al sistema; dicha relación está dada de la siguiente manera:

La información, tanto de los expertos humanos como de los usuarios, puede ser incorporada de dos formas:

1. Por el Subsistema de Aprendizaje .
2. Por el Subsistema de Demanda o Adquisición de Información.

El *Motor de Inferencias* o *Máquina de Inferencia* (MI) saca conclusiones aplicando la información de la *BC* y de no poseer los datos necesarios le serán solicitados al experto humano o al usuario por el *Subsistema de Demanda* o *Adquisición de Información* a través de la *Interfase de Usuario*, por la cual el experto humano o el usuario ingresará la información necesaria.

Una vez obtenidas las conclusiones por el MI el S.E. puede iniciar acciones utilizando el *Subsistema de Tareas*.

El usuario obtendrá las explicaciones de las conclusiones obtenidas y de las acciones realizadas gracias al *Subsistema de Explicación*.

Al presentarse nuevas experiencias, éstas pueden ser aprendidas por el S.E. gracias al *Subsistema de Aprendizaje*.

## 2.5. Caracterización de los SS.EE.

### 2.5.1. Diferencias entre un S.E. y un programa tradicional

En la resolución de problemas o en la automatización de tareas existen algunas cuestiones que hacen difícil su cometido, esto tiene las siguientes causas:

1. El fenómeno de explosión combinatoria que se debe al crecimiento exponencial del número de estados posibles de un sistema o de alternativas a considerar para resolver un problema.
2. Los dominios no estructurados. Aquellos para los cuales no se dispone de teorías o modelos causales detallados, ni métodos matemáticos o suficientes para aplicar algoritmos que garanticen soluciones satisfactorias.
3. La necesidad de sentido común. La necesidad de aplicar gran cantidad de conocimientos, adquiridos durante la vida y que son aplicados de manera natural pero que no se tienen bien definidos ni formulados explícitamente.

En un programa convencional la implementación de un sistema para diagnóstico médico utilizando programación tradicional, podría ser:

```
Begin
Nombre := ask("Nombre ? ")
Edad := ask("Edad ? ")
.....
Sintoma_1 := ask("Sintoma ? ")
.....
if GRAMNEG in síntomas then
  Begin
    forma := ask ("Morfologia ? ")
    if forma := "COCUS" then
      Begin
        fiebre := ask("Ha tenido....")
        if fiebre := "SI" then
          .....
```

Este tipo de implementación presenta dificultades como las siguientes:

1. Es difícil evaluar lo que "sabe" el sistema.
2. Es engorroso modificarlo o actualizarlo, es demasiado rígido.
3. Se desconoce lo que sucede con la inexactitud e incertidumbre inherentes a los conocimientos, en general.

Con los SS.EE. no suceden estos inconvenientes debido a que los conocimientos están separados del procedimiento lógico, en contra parte con los sistemas programados de la forma tradicional.

En el siguiente cuadro se presentan las diferencias entre un sistema convencional y un S.E..

<i>Sistema Convencional</i>	<i>Sistema Experto</i>
Conocimiento y lógica de proceso mezclados en un programa.	Base de Conocimientos y Mecanismos de Inferencia separados.
No explican porqué se necesitan los datos ni porqué se llegó a un resultado.	Sí lo explican.
Es difícil efectuar cambios en los conocimientos programados.	Es más fácil modificar la BC.
Necesitan información completa para operar.	Deben ser más tolerantes para operar aún con alguna información desconocida.
Generalmente manejan datos cuantitativos.	Manejan datos cualitativos primordialmente.
Captura, amplifica y distribuye el acceso a datos numéricos o textuales.	Captura, amplifica y distribuye el acceso a juicios basados en conocimientos.

**Tabla 1.** Diferencias entre un Sistema Convencional y un Sistema Experto.

### **2.5.2. Razones para la utilización de SS.EE.**

Existen varias razones por las cuales se hace deseable aplicar un SS.EE. a un problema determinado, algunas de ellas, las más importantes, son las siguientes:

1. "El conocimiento tiene poder organizador" (Edward Feigenbaum).
2. Los expertos son escasos, costosos y tardan mucho tiempo en formarse.
3. Un SS.EE. permite mayor accesibilidad a la experiencia gracias a la posibilidad de duplicación y transferencia. Esto es útil en organizaciones que tienen varias sucursales dispersas geográficamente y sólo existe un único experto humano; con un SS.EE. en cada sucursal el conocimiento del experto humano puede duplicarse y ser utilizado en cada sucursal. También es posible transferir conocimientos de un experto a lugares donde haya escasos o nulos expertos humanos.

4. El SS.EE., al dedicarse a casos en donde no necesita asistencia o supervisión de un experto humano, permite el ahorro de tiempo a este último para que dedique todo su esfuerzo y tiempo a casos que excedan la capacidad del S.E..
5. Con la utilización de SS.EE. los novatos se vuelven expertos de manera incremental.
6. La mayoría de los SS.EE. han llegado a las mismas conclusiones que los expertos humanos en una gran cantidad de casos, comprobando que los SS.EE. son casi tan buenos como sus pares humanos.
7. Un SS.EE. posee conocimientos utilizables directamente de manera descentralizada y duradera.
8. El desarrollo de un SS.EE. obliga a la recuperación, depuración, formalización y conservación de conocimientos y experiencia.
9. Gracias a que posee un subsistema de explicación o justificación, el razonamiento del SS.EE. es visible para los usuarios brindando la confiabilidad necesaria para su utilización continua en casos futuros.
10. Por la razón anterior, los SS.EE. son utilizables para el entrenamiento de nuevo personal.
11. Posee un mejor y más fácil mantenimiento al poseer separados los conocimientos del procedimiento lógico.
12. Un SS.EE. aumenta la productividad por:
  - 12.1. Ahorrar tiempo valioso del experto.
  - 12.2. Ser más ágil que el humano, en especial cuando éste debe consultar grandes volúmenes de datos los cuales no los posee incorporados totalmente.
  - 12.3. La accesibilidad y rapidez para atender, por ejemplo, diagnóstico de equipos; puede reducir perdidas al disminuir el tiempo fuera de servicio de un equipo.
13. Un SS.EE. puede funcionar por tiempo indeterminado sin tomar descansos típicos de un humano.

### **2.5.3. Límites y dificultades de los SS.EE.**

Por ser invenciones del hombre y por el límite natural que poseen las máquinas, los SS.EE presentan las siguientes limitantes:

1. Son utilizables en un estrecho dominio de aplicación.

2. No existe gran variedad de lenguajes de programación para representar hechos y relaciones.
3. Los conocimientos llegan a ser demasiado superficiales.
4. Por ser máquinas o sistemas no poseen "conciencia" de su propio alcance y limitaciones.
5. El ser humano posee la capacidad de aplicar todos los conocimientos adquiridos para solucionar los problemas que se le presenten, aún estando fuera del campo de su especialidad consiguiendo, sin embargo, un éxito que disminuye conforme se aleja de su campo de especialidad. Esto no sucede con los SS.EE. ya que al alejarse de su dominio de aplicación fracasan catastróficamente y no de forma gradual.
6. La explicación que brinda, por lo general, se limita a la enumeración de las reglas que utilizó para llegar a determinada conclusión o acción.

Al introducir un SS.EE. en una organización se presentan las siguientes situaciones:

1. Los términos *experto*, *inteligencia artificial*, etc. suenan demasiado pretensiosos o poco serios con respecto a las preocupaciones de la organización.
2. Falta de adopción dentro del Dpto. de Procesamiento de Datos tradicional.
3. Es común que los empleados vean al sistema como una amenaza a sus puestos y no como una herramienta de apoyo y potenciación de sus habilidades.
4. Muchas veces las organizaciones creen que tienen la necesidad de implementar un SS.EE. antes de tener un problema para resolverlo con él, es decir, comienzan a buscar en dónde pueden implementar un SS.EE. antes de ver si los problemas que existen pueden resolverse muy fácilmente con las herramientas que poseen.

## 2.6. Desarrollo de un SS.EE.

Las etapas propuestas para el desarrollo de un SS.EE. fueron sugeridas por Weiss y Kulikowski [6]; para ampliar sobre este tema se recomienda ver [7] y [8].

Dichas etapas son siete, la secuencia de ejecución se ilustra en el **Gráfico 3**.

La primera de las etapas sugeridas es la de *Planteamiento del Problema*. Como en todo proyecto lo primero es definir clara y específicamente el problema a resolver para evitar todo un desarrollo en vano. Si en esta etapa el problema en cuestión es mal definido lo más probable es que el sistema funcione incorrectamente con lo cual las decisiones que se tomarán en base a las conclusiones que arroje el sistema serán equivocadas.

Una vez definido el problema se comienza con la *Búsqueda de los Expertos que puedan resolverlo*. Siempre es conveniente buscar y seleccionar aquellos que posean la mayor experiencia en la resolución del problema planteado en la etapa anterior; también es posible hacer, primero, una búsqueda de bibliografía especializada sobre dicho problema ya que sería la base para comenzar la carga de los conocimientos en el sistema. Se debe tener en cuenta que los conocimientos deben ser formalizados antes de ingresarlos a la Base de Conocimientos, por esto es que un libro será más fácil de estructurar en la Base de Conocimientos que los conocimientos de un Experto Humano.

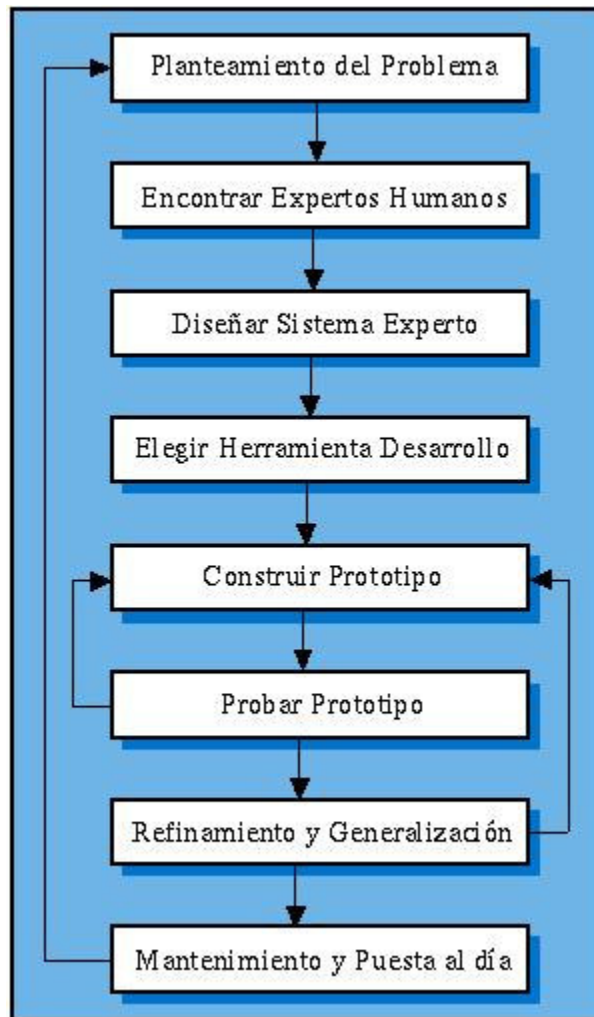
La tercera etapa es el *Diseño del Sistema Experto*. En este punto se comienza con el diseño de los componentes del sistema: Motor de Inferencias, estructura de la Base de Conocimientos, etc.

Con el Diseño del Sistema terminado se debe realizar la *Elección de la Herramienta de Desarrollo, shell o lenguaje de programación* para la construcción del sistema. En el caso de que existiera un shell que se ajuste al diseño hecho en la etapa anterior; utilizarlo debería ser la elección a tomar ya que, por lo general, un shell sufre controles de calidad lo cual la hace fiable y, por supuesto, es la solución más económica y rápida a implementar. Un shell, en pocas palabras, es un Sistema Experto (estructura) vacío, es decir, posee la BC vacía. Esto permite que el desarrollador sólo se enfoque en la carga de la BC y no en todo el desarrollo del sistema.

La quinta etapa consiste en el *Desarrollo y prueba de un prototipo*. Si por alguna razón el prototipo no resulta apropiado para los requerimientos predeterminados, se deben repetir las etapas anteriores hasta obtener un prototipo apropiado para la resolución del problema.

El *Refinamiento y generalización* es la etapa que sigue. Luego de aceptado el prototipo se corrigen los errores y se agregan las posibilidades no contempladas en el diseño inicial.

Por último resta realizar el *Mantenimiento y puesta al día*. Con la ayuda del usuario se plantean nuevos problemas o se detectan defectos del prototipo, se corrigen errores, se actualiza el producto con los nuevos avances, etc. Esta etapa provoca que las anteriores se repitan ya que son cambios que requieren cada una de ellas para obtener las mejores soluciones.



**Gráfico 3.** Etapas de desarrollo de un Sistema Experto.

## 2.7. Tipos de SS.EE.

Según el tipo de problema a resolver los SS.EE. pueden ser de dos tipos básicamente:

1. SS.EE. basado en Reglas.
2. SS.EE. basado en Probabilidad.
3. SS.EE. basado en Redes Neuronales.
4. SS.EE. Difuso.

Estos último no son más que una RNA "configurada" para funcionar como un sistema experto.

En las Secciones 2.7.1, 2.7.2, 2.7.3 y 2.7.4, respectivamente, se explican brevemente los cuatro tipos de SS.EE. mencionados, ampliándose en detalles en las Secciones 3 y 4 los dos primeros por su amplia aplicación en el campo de la medicina.

### **2.7.1. Sistemas Expertos basados en Reglas**

Los Sistemas Expertos Basados en Reglas, son las estructuras de sistemas expertos más comprensibles para la lógica humana, se definen a partir de un conjunto de objetos, que representan las variables del modelo, vinculadas mediante un conjunto de reglas encadenadas que representan las relaciones entre variables. Esta metodología es la que el ser humano usa en forma intuitiva en su vida diaria para resolver problemas.

Muchos de las situaciones con las que el ser humano se enfrenta a menudo, están gobernadas por reglas deterministas; en este ámbito los SS.EE. basados en reglas constituyen una herramienta eficiente para la solución de estos problemas.

### **2.7.2. Sistemas Expertos basados en Probabilidad**

Existen problemas en donde lo que interesa saber con exactitud solo se sabe con cierto grado de certeza, como, por ejemplo, las relaciones entre síntomas y enfermedades (no siempre la existencia de un grupo de síntomas asegura la existencia de una enfermedad específica). Este tipo de problemas reciben el nombre de *estocásticos*.

Para la solución de estos problemas se han propuesto formas de medir el grado de incertidumbre, algunas de esas medidas son llamadas *factores de certeza* y se utilizan en los shells para la generación de SS.EE.; otra medida intuitiva es la *probabilidad*.

La utilización del cálculo de probabilidades en los SS.EE. comenzó con la *probabilidad conjunta* pero la dificultad para definir la distribución de probabilidad conjunta de las variables hizo que se obstaculizara el desarrollo de los mismos; para solucionar este inconveniente se introdujeron los *modelos de redes probabilísticas*, los cuales incluyen las redes de *Markov* y las redes *Bayesianas*.

### **2.7.3. Sistemas Expertos basados en Redes Neuronales Artificiales**

Las Redes Neuronales Artificiales (RNA), son el resultado de las investigaciones que se centran en el desarrollo de sistemas análogos a las neuronas y sus redes.

En 1943, McCulloch y Pitts propusieron un modelo de neurona para explicar cómo el cerebro podría realizar funciones booleanas.

A partir de allí el desarrollo se lentificó (1960-1970); luego, en los años '80, con la utilización de conceptos de la física de los sistemas caóticos en las arquitecturas de redes de neuronas formales y la introducción de métodos de aprendizaje aplicables a redes multicapa de neuronas, se lograron abrir nuevas perspectivas en este campo.

Si bien las RNA no fueron concebidas para utilizarse como SS.EE., pueden comportarse como éstos ofreciendo una alternativa más.

Cuando se usan RNA, el dominio y la tarea no requieren ser tan bien comprendidos como para usar los shells tradicionales. No se requiere que el experto detalle cómo llega a una solución, ya que el sistema se adapta con base en las relaciones estímulo / respuesta, reconfigurándose para resolver el problema. La ventaja principal de usar este tipo de implementación es que no requiere algoritmos específicos del dominio; otra ventaja consiste en que se pueden degradar más gracilmente que los SS.EE. tradicionales si falla alguna parte. La principal desventaja es que no hay manera de examinar los conocimientos del sistema, y en este sentido es una especie de caja negra, en la cual se puede confiar sólo por su comportamiento entrada / salida.

#### 2.7.4. Sistemas Expertos basados en Lógica Difusa

Hay situaciones en la vida cotidiana en que los problemas que se presentan no poseen una solución muy específica en términos exactos, por ejemplo cuando se debe contestar a la pregunta ¿cuántas personas son muy bajas?, o ¿en cuáles ciudades hace mucho frío?. Las respuestas a estas preguntas son fácilmente contestadas por un humano pero a la hora de trasladar el problema para que sea resuelto por una computadora, el problema se complica. Para ampliar los conceptos, a continuación presentados, se sugiere consultar [11], [12] y [13].

Para resolver este tipo de inconvenientes surgió una teoría llamada *Lógica Difusa*, la cual crea una relación matemática entre un elemento y un determinado *conjunto difuso* con el objetivo de que una máquina realice una valoración semejante a la de los seres humanos. Esta teoría fue propuesta por L. Zadeh en el año 1965.

**Definición.** Teniendo un posible rango de valores  $U$  llamado *Universo de Discurso*. en  $U$  se tendrá un conjunto difuso de valores llamado  $F$  caracterizado por una función de pertenencia  $uf$  tal que  $uf: U \rightarrow [0,1]$ , donde  $uf(u)$  representa el grado de pertenencia de un  $u$  perteneciente a  $U$  en el conjunto difuso  $F$  [9].

Al igual que en los conjuntos tradicionales, los conjuntos difusos poseen operaciones entre ellos. La frontera de los conjuntos difusos no está precisamente definida, por lo que la relación de pertenencia de un elemento retorna un valor entre cero y uno. Dicha relación está dada por el grado de pertenencia simbolizado por  $\Leftarrow$ , este grado de pertenencia es subjetivo y depende del dominio.

Las operaciones que se puede aplicar sobre los conjuntos difusos son:

1. **Intersección.** Al igual que la intersección en los conjuntos clásicos, significa que se desean los elementos que pertenezcan a uno "y" (AND) otro conjunto, es decir, que pertenezcan a los dos conjuntos involucrados en la operación. Matemáticamente se expresa de la siguiente manera:

$$C = A \cap B \quad \forall x \in U$$

$$\Leftarrow(A \text{ and } B) = \min\{\Leftarrow(A), \Leftarrow(B)\}$$

2. **Unión.** La unión supone la idea de que se desean los elementos que pertenezcan a uno "o" (OR) a otro conjunto, es decir a cualquiera de los dos (o más) conjuntos involucrados.

$$C = A \cup B \quad \forall x \in U$$

$$\Leftarrow(A \text{ or } B) = \max\{\Leftarrow(A), \Leftarrow(B)\}$$

3. **Complemento.** En los conjuntos clásicos se define como el conjunto de todos aquellos elementos que le faltan a un conjunto A para ser igual al conjunto universo. Para los conjuntos difusos, el complemento, es el conjunto de valores de pertenencia faltantes para que un conjunto pueda obtener el valor máximo de pertenencia posible, siendo uno el valor máximo de pertenencia que un conjunto difuso puede suministrar. Dicho conjunto puede formarse restando, al valor uno, los valores de pertenencia del conjunto difuso al que se desea encontrar el complemento. Sea  $A = \{a_1, \dots, a_n\}$  un conjunto difuso y  $U = \{1\}$  el conjunto Universo. El complemento de A se define por extensión así:

$$A' = U - A = 1 - \{a_1, \dots, a_n\} = \{1 - a_1, \dots, 1 - a_n\} = \{a'_1, \dots, a'_n\}$$

Matemáticamente la operación se denota de la siguiente manera:

$$\Leftarrow(\text{not } A) = 1 - \Leftarrow(A)$$

El proceso de inferencia en los Sistemas Expertos Difusos utiliza reglas difusas donde si el antecedente tiene cierto grado de pertenencia, entonces el consecuente también lo tiene pero no puede tener mayor grado de pertenencia que su antecedente.

La cantidad de reglas necesarias en este tipo de Sistemas Expertos es de  $m^n$ , siendo  $m$  la cantidad de variables y  $n$  los valores difusos que pueden asumir cada una de esas variables.

Además de los operadores vistos anteriormente se pueden aplicar otros, llamados operadores difusos (hedges) los cuales son considerados como modificadores a los "términos" difusos:

**Concentración.** Por ejemplo, cuando se dice *alto muy alto*.

$$\text{con } (\Leftarrow)(a) = \Leftarrow^2(a)$$

**Dilatación.** Por ejemplo, al decir *alto medio alto*.

$$\text{dil } (\Leftarrow)(a) = \Leftarrow^{1/2}(a)$$

## Normalización.

$$\text{norm}(\Leftarrow)(a) = \Leftarrow(a) / \max\{\Leftarrow(a)\}$$

## Intensificación.

$$\text{int}(\mu)(a) = \begin{cases} \mu^2(a) & \text{para } 0 \leq \mu^2(a) \leq 0.5 \\ 1 - 2(1 - \mu(a))^2 & \text{para } 0.5 \leq \mu^2(a) \leq 1 \end{cases}$$

Debido a la gran utilidad de los Sistemas Expertos basados en Reglas y los Sistemas Expertos Probabilísticos; en las Secciones 3 y 4 se desarrollan con más detalle que lo expuesto hasta aquí. En ambas secciones se explican las partes componentes de cada sistema y luego, en la Sección 5, se los compara para observar sus diferencias y similitudes como así también sus ventajas y desventajas.

## 3. Sistemas Expertos basados en Reglas

### 3.1. Base de Conocimientos

En este tipo de sistemas intervienen dos elementos fundamentales, la Base de Conocimientos y los Datos.

Los datos son, por ejemplo en un sistema de diagnóstico médico, los síntomas que posee el paciente, es decir, hacen referencia a las verdades o realidades irrefutables de una situación particular, por ser aleatorios o variables de una situación a otra, son almacenados en la Memoria de Trabajo.

La base de conocimientos se conforma por las relaciones existentes entre objetos (datos) llamadas *reglas*. Las reglas son asociaciones entre una o varias *condiciones* o *premisas* y una *conclusión*. Dicha conclusión puede estar conformada a su vez por un grupo de conclusiones que suceden simultáneamente (conjunción "y") y/o una o varias (disyunción "o").

Los conocimientos almacenados en la BC son estáticos y permanentes de una aplicación a otra, excepto que se incorporen elementos de aprendizaje, como el subsistema de aprendizaje.

Para una mejor comprensión de lo que es una regla se presenta el siguiente ejemplo:

Se tiene un conjunto de objetos y que cada uno de esos objetos puede asumir un único valor de un conjunto finito (o infinito) de valores posibles; es verdad que puede asumir más de un único valor pero por simplicidad se elige esta condición. Con este enunciado se forman las siguientes reglas:

**Ejemplo.** Supóngase las siguientes reglas.

**Regla 1.** Si nota > 9 entonces calificación = sobresaliente.

**Regla 2.** Si puesto < 20 o nota > 7 entonces Admitir = sí y Notificar = sí.

Cada una de las reglas anteriores relaciona dos o más objetos y están formadas por las partes siguientes:

**Definición.** Una premisa. Expresión lógica que se encuentra entre las palabras clave si y entonces. La premisa puede contener una o más afirmaciones objeto-valor unidos por operadores lógicos ('y', 'o' ó 'no').

**Definición.** Una conclusión que se encuentra luego de la palabra clave entonces.

Una expresión lógica que contiene sólo una afirmación objeto-valor se denomina *expresión lógica simple*; en caso contrario, *expresión lógica compuesta*.

De igual manera, una regla que contiene solamente expresiones lógicas simples se denomina una *regla simple*; en otro caso, se llama *regla compuesta*.

### 3.2. Motor de Inferencia (MI)

El MI utiliza la BC y los datos para sacar conclusiones, las cuales pueden ser de dos tipos, *simples* o *compuestas*.

Las primeras son las que surgen como resultado de una regla simple, mientras que las compuestas surgen de más de una regla.

Para la obtención de conclusiones se utilizan diferentes tipos de reglas y estrategias; en el presente trabajo se describirán las siguientes:

<i>Reglas</i>	<i>Estrategias</i>
Modus Ponens	Encadenamiento de reglas
Modus Tollens	Encadenamiento de reglas orientado a un objetivo.
Resolución	Compilación de reglas

**Tabla 2.** Clasificación de las reglas utilizadas en la BC y las Estrategias utilizadas en el MI.

Las dos primeras reglas se utilizan para la obtención de conclusiones simples, mientras que el resto de reglas y estrategias se utilizan para la obtención de conclusiones compuestas.

Para obtener todas las conclusiones posibles es necesario implementar más de un tipo de reglas y estrategias.

### 3.2.1. Regla Modus Ponens y Modus Tollens

La regla Modus Ponens es, quizás, la más utilizada. Se utiliza para obtener conclusiones simples. En ella, se examina la premisa de la regla, y si es cierta, la conclusión pasa a formar parte del conocimiento. En el **Gráfico 4** se ejemplifica este tipo de reglas.

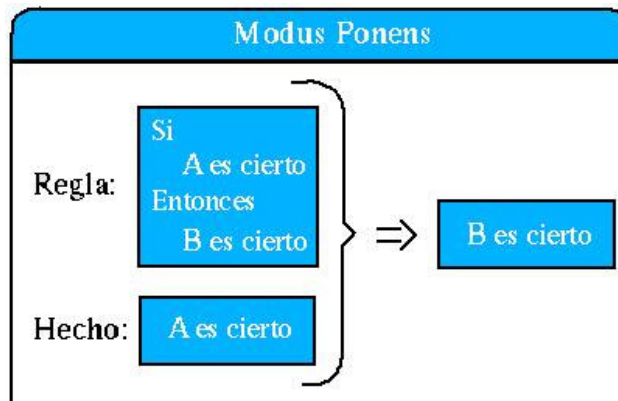
Esta regla de inferencia, que parece trivial, debido a su familiaridad, es la base de un gran número de sistemas expertos.

La regla Modus Tollens se utiliza también, para obtener conclusiones simples, pero contrariamente a la anterior, se examina la conclusión y si es falsa, se concluye que la premisa también lo es. Esta regla se esquematiza en el **Gráfico 5**.

En resumen, se dice, que la regla Modus Ponens se mueve hacia adelante debido a que parte de la premisa hacia la conclusión, mientras que la regla Modus Tollens se mueve hacia atrás, de la conclusión hacia la premisa.

La regla Modus Ponens necesita información de los objetos de la premisa y la regla Modus Tollens necesita información de los objetos de la conclusión.

Ambas reglas no deben ser vistas como alternativas sino como complementarias para ser usadas en una MI.



**Gráfico 4.** Ejemplo de regla Modus Ponens.

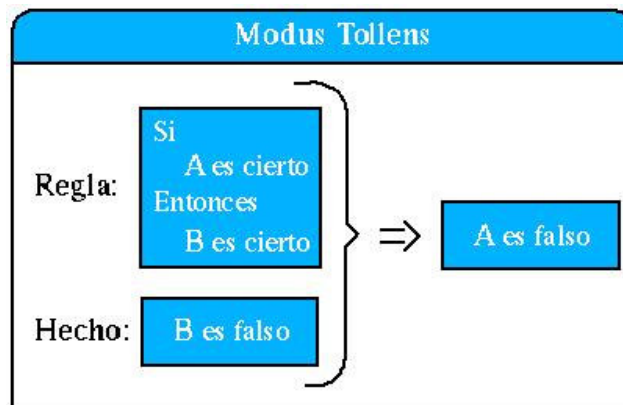


Gráfico 5. Ejemplo de regla Modus Tollens.

### 3.2.2. Mecanismo de resolución

Utilizando las reglas anteriores se conseguían conclusiones simples; con el mecanismo de resolución se obtienen conclusiones compuestas, éstas están basadas en dos o más reglas.

El Mecanismo de resolución posee las siguientes etapas:

1. Las reglas son sustituidas por expresiones lógicas equivalentes.
2. Estas expresiones se combinan en otra expresión lógica.
3. Y finalmente se utiliza esta expresión para obtener la conclusión.

Para lograr estas etapas se utilizan los conceptos de combinación y simplificación de expresiones lógicas.

A continuación se expone un ejemplo en donde se ilustran los conceptos de combinación y simplificación:

**Ejemplo.** Se supone que se tienen las siguientes reglas:

**Regla 1.** Si A es cierto, entonces B es cierto.

**Regla 2.** Si B es cierto, entonces C es cierto.

La primera etapa es la sustitución de las reglas por expresiones lógicas equivalentes, para ello se construye la siguiente tabla de verdad:

$A$	$B$	$\bar{A}$	Si $A$ , entonces $B$	$\bar{A} \circ B$
C	C	F	C	C
C	F	F	F	F
F	C	C	C	C
F	F	C	C	C

**Tabla 3.** Tabla de verdad de la Regla 1.

En la **Tabla 1** se observa que la *Regla 1* es equivalente a la expresión lógica *A' es cierto o B es cierto*; y que la *Regla 2*, por analogía, es equivalente a la expresión lógica *B' es cierto o C es cierto*.

La siguiente etapa es la combinación de las expresiones en otra expresión lógica resultante.

Las expresiones equivalentes, respectivamente, a cada una de las reglas son:

- a. **A** es falso o **B** es cierto
- b. **B** es falso o **C** es cierto

Estas expresiones implican la expresión:

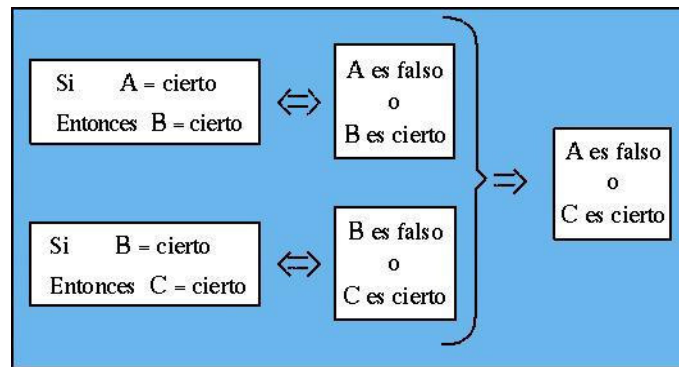
- c. **A** es falso o **C** es cierto

Lo cual queda demostrado por la siguiente tabla de verdad.

$A$	$B$	$C$	$\bar{A} \circ B$	$\bar{B} \circ C$	$(\bar{A} \circ B)$ y $(\bar{B} \circ C)$	$\bar{A} \circ C$
C	C	C	C	C	C	C
C	C	F	C	F	F	F
C	F	C	F	C	F	C
C	F	F	F	C	F	F
F	C	C	C	C	C	C
F	C	F	C	F	F	C
F	F	C	C	C	C	C
F	F	F	C	C	C	C

**Tabla 4.** Tabla de verdad de las reglas obtenidas por la combinación.

En el siguiente **Gráfico 6** se ilustran ambas etapas: sustitución y combinación.



**Gráfico 6.** Ejemplo de las etapas de sustitución y combinación de reglas.

Finalmente con esta última expresión: *A es falso o C es cierto* se obtiene la conclusión.

### 3.2.3. Encadenamiento de Reglas

Esta estrategia de inferencia es la más utilizada para la obtención de reglas compuestas. Su utilización se aplica cuando las premisas de una regla coinciden con las conclusiones de otra.

Cuando se encadenan las reglas, los hechos pueden utilizarse para dar lugar a nuevos hechos. Esto se repite sucesivamente hasta que no pueden obtenerse más conclusiones. La estrategia de encadenamiento de reglas se da en el algoritmo siguiente:

1. Para el inicio del algoritmo es necesario un conjunto de datos, compuesto por una base de conocimientos (objetos y reglas) y algunos hechos iniciales.
2. El algoritmo inicia asignando a los objetos sus valores conocidos tales como los dan los hechos conocidos o la evidencia.
3. Se ejecuta la regla de la base de conocimientos y concluye nuevos hechos si es posible.
4. Se repite el segundo paso hasta que no se puedan obtener nuevos hechos.

A continuación se presentan dos ejemplos donde se aprecia el encadenamiento de reglas y algoritmo aplicado, como así también, el uso de las reglas Modus Ponens y Modus Tollens.

#### 3.2.3.1. Ejemplos de Encadenamiento de Reglas

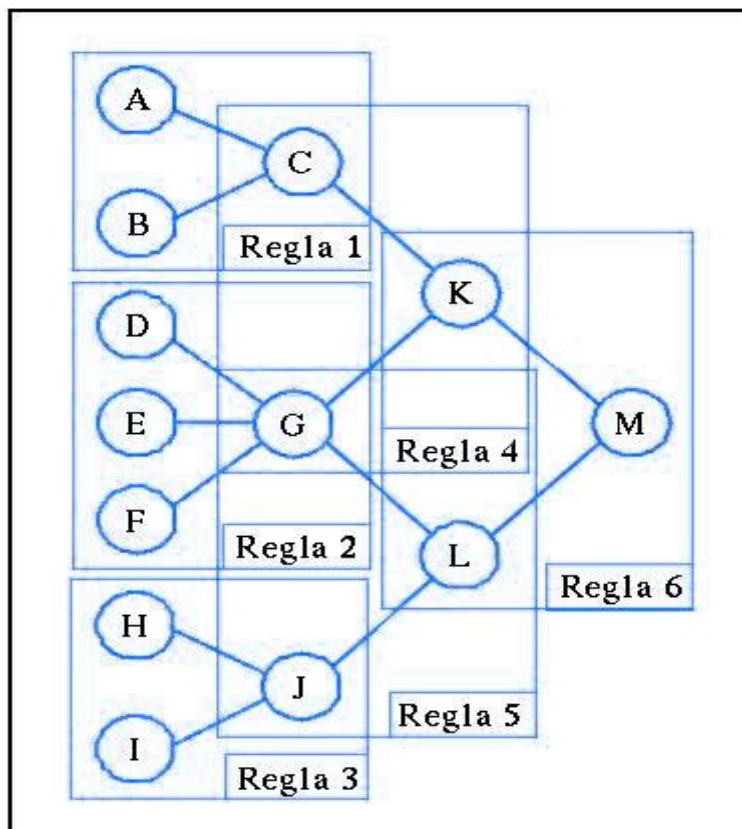
**Ejemplo.** Se tienen 13 (trece) objetos: A, B, ....., M; relacionados por medio de 6 (seis) reglas, ver **Gráfico 7**.

<b>Regla 1</b>	<b>Regla 2</b>	<b>Regla 3</b>
Si A y B Entonces C	Si D, E y F Entonces G	Si H e I Entonces J
<b>Regla 4</b>	<b>Regla 5</b>	<b>Regla 6</b>
Si C y G Entonces K	Si G y J Entonces L	Si K y L Entonces M

**Gráfico 7.** Ejemplo de 6 (seis) reglas relacionando 13 objetos.

Supóngase que los objetos A, B, D, E, F, H e I son ciertos y los restantes son de valor desconocido.

En el siguiente **Gráfico 8** se representan los objetos mediante círculos (nodos) y las relaciones son representadas por las aristas que unen las premisas de las reglas con las conclusiones; nótese que las premisas de algunas reglas coinciden con las conclusiones de otras.



**Gráfico 8.** Ejemplo de encadenamiento de reglas.

Supóngase que los objetos A, B, D, E, F, H e I son ciertos y que los restantes son de valor desconocido.

En este caso el algoritmo de encadenamiento procede así:

- La Regla 1 concluye que  $C = \text{cierto}$ .
- La Regla 2 concluye que  $G = \text{cierto}$ .
- La Regla 3 concluye que  $J = \text{cierto}$ .
- La Regla 4 concluye que  $K = \text{cierto}$ .
- La Regla 5 concluye que  $L = \text{cierto}$ .
- La Regla 6 concluye que  $M = \text{cierto}$ .

Al llegar a la Regla 6 no se pueden obtener más conclusiones por lo que el algoritmo se detiene.

**Ejemplo.** Teniendo en cuenta las reglas del **Gráfico 7** junto con que  $H$ ,  $K$  e  $I$  son ciertos y que  $M$  es falso se supone que el MI utiliza las dos reglas de inferencia (Modus Ponens y Modus Tollens). Entonces se aplica el algoritmo antes explicado con el cual se obtiene que:

1. La Regla 3 concluye que  $J = \text{cierto}$  (Modus Ponens).
2. La Regla 6 concluye (utilizando Modus Tollens) que  $K = \text{falso}$  o  $L = \text{falso}$ , pero, puesto que  $K = \text{cierto}$ , deberá ser  $L = \text{falso}$ .
3. La Regla 5 concluye (Modus Tollens) que  $G = \text{falso}$  o  $J = \text{falso}$ , pero, puesto que  $J = \text{cierto}$ , deberá ser  $G = \text{falso}$ .

Por lo que se obtiene la conclusión final de que  $G$  es falso. Este es, a la vez, un ejemplo de la utilización conjunta de las reglas Modus Ponens y Modus Tollens.

### 3.2.4. Encadenamiento de Reglas orientado a un objetivo

En el encadenamiento de reglas orientado a un objetivo, el sistema necesita del usuario para seleccionar una variable o nodo objetivo; desde allí el algoritmo navega las reglas en búsqueda de una conclusión para ese nodo objetivo. En caso de no obtener una conclusión con la información existente, el algoritmo fuerza al sistema a preguntarle al usuario por más información sobre los elementos importantes.

En el párrafo siguiente se describe el algoritmo mencionado:

**Datos.** Una base de conocimiento (objetos y reglas), algunos hechos iniciales, y un nodo o variable objetivo.

**Resultado.** El valor del nodo o variable objetivo.

**Etapas.** 1. Asigna a los objetos sus valores conocidos tales como están dados en los hechos de partida, si es que existe alguno, una vez hecho esto, el algoritmo marca todos los objetos cuyo valor ha sido asignado. Si el nodo objetivo está marcado, va a la **Etapas 7**; en otro caso realiza lo siguiente:

- a. Designa como objetivo inicial el objetivo en curso.
- b. Marca el objetivo en curso.
- c. Sea  $ObjetivosPrevios = \emptyset$ , donde  $\emptyset$  es el conjunto vacío.
- d. Designa todas las reglas como activas (ejecutables).
- e. Ir a la **Etapa 2**.

**Etapa 2.** Encontrar una regla activa que incluya el objetivo en curso y ninguno de los objetos en *ObjetivosPrevios*. Si se encuentra una regla, ir a la **Etapa 3**; en otro caso, ir a la **Etapa 5**.

**Etapa 3.** Ejecutar la regla referente al objetivo en curso. Si concluye, asignar el valor concluido al objetivo en curso, e ir a la **Etapa 6**; en otro caso, ir a la **Etapa 4**.

**Etapa 4.** Si todos los objetos de la regla están marcados, declarar la regla como inactiva e ir a la **Etapa 2**; en otro caso:

- a. Añadir el objetivo en curso a *ObjetivosPrevios*.
- b. Designar uno de los objetos no marcados en la regla como el objetivo en curso.
- c. Marcar el objetivo en curso.
- d. Ir a la **Etapa 2**.

**Etapa 5.** Si el *objetivo en curso* es el mismo que el objetivo inicial, ir a la **Etapa 7**; en otro caso, preguntar al usuario por el valor del objetivo en curso. Si no se da un valor, ir a la **Etapa 6**; en otro caso asignar al objeto el valor dado e ir a la **Etapa 6**.

**Etapa 6.** Si el objetivo en curso es el mismo que el objetivo inicial, ir a la **Etapa 7**; en otro caso, designar el objetivo previo como objetivo en curso, eliminarlo de *ObjetivosPrevios*, e ir a la **Etapa 2**.

**Etapa 7.** Devolver el valor del objetivo en curso si es conocido.

### 3.2.4.1. Ejemplos de Encadenamiento de Reglas Orientado a un Objetivo

**Ejemplo.** Se toman nuevamente las reglas del **Gráfico 7**, las relaciones dadas por el **Gráfico 8** y se supone que se selecciona como nodo objetivo el nodo M; además se sabe que los objetos D, E, F y L son ciertos.

El algoritmo se desarrolla en las siguientes etapas:

**Etapa 1.** Se asigna el valor *cierto* a los objetos D, E, F y L y se marcan. Puesto que el nodo objetivo M no está marcado, entonces

- a. Se designa el objeto M como en curso.
- b. Se marca el objeto M. Por tanto, se tiene  $ObjetosMarcados = \{D, E, F, L, M\}$ .
- c.  $ObjetosPrevios = 0$  (cero).
- d. Las seis reglas están activas. Por tanto, se tiene  $ReglasActivas = \{1,2,3,4,5,6\}$ .
- e. Se va a la **Etapa 2**.

**Etapa 2.** Se busca una regla que incluya el objetivo en curso M. Se encuentra la Regla 6, por lo que se va a la **Etapa 3**.

**Etapa 3.** La Regla 6 no puede concluir puesto que el valor del objeto K es desconocido. Así que se va a la **Etapa 4**.

**Etapa 4.** El objeto K no está marcado. Entonces:

- a.  $ObjetivosPrevios = \{M\}$ .
- b. Se elige el objeto K como objetivo en curso.
- c. El objeto K está marcado. Por tanto se tiene,  $ObjetosMarcados = \{D, E, F, L, M, K\}$ .
- d. Se va a la **Etapa 2**.

**Etapa 2.** Se busca una regla que incluya el objetivo en curso K pero no el anterior M. Se encuentra la Regla 4, y se continúa con la **Etapa 3**.

**Etapa 3.** La Regla 4 no puede concluir puesto que se desconocen los valores de los objetos C y G. Por ello, se continúa con la **Etapa 4**.

**Etapa 4.** Los objetos C y G no están marcados. Entonces:

- a.  $ObjetivosPrevios = \{M, K\}$ .
- b. Se elige uno de los objetos no marcados C o G como el nuevo objetivo en curso. Supóngase que se elige C.
- c. Se marca el objeto C.
- d. Por tanto, se tiene  $ObjetosMarcados = \{D, E, F, L, M, K, C\}$ .
- e. Se continúa con la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo en curso C pero no los objetos previos {M, K}. Se encuentra la Regla 1, así que se va a la **Etapa 3**.

**Etapa 3.** La Regla 1 no puede concluir puesto que se desconocen los valores de los objetos A y B. Por tanto se continúa con la **Etapa 4**.

**Etapa 4.** Los objetos A y B no están marcados. Entonces:

- a.  $ObjetivosPrevios = \{M, K, C\}$ .
- b. Se elige uno de los objetos no marcados A y B como nuevo objetivo en curso. Supóngase que se elige A.
- c. Se marca el objeto A.
- d. Por ello,  $ObjetosMarcados = \{D, E, F, L, M, K, C, A\}$ .
- e. Se continúa con la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo en curso A pero no los  $ObjetivosPrevios \{M, K, C\}$ . No se encuentra ninguna regla que satisfaga estas condiciones, así que se pasa a la **Etapa 5**.

**Etapa 5.** Puesto que el objetivo en curso A es diferente del inicial M, se pregunta al usuario por el valor del objeto A. Supóngase que A toma el valor cierto, entonces se hace  $A = \text{cierto}$  y se sigue con la **Etapa 6**.

**Etapa 6.** El objetivo en curso A no coincide con el previo M. Por tanto, el objeto C se designa como objetivo en curso y se elimina de la lista  $ObjetivosPrevios$ . Por ello,  $ObjetivosPrevios = \{M, K\}$  y se continúa con la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo C pero no los anteriores  $\{M, K\}$ . Se encuentra la Regla 1, por lo que se va a la **Etapa 3**.

**Etapa 3.** La Regla 1 no puede concluir porque el valor del objeto B es desconocido. Así que se va a la **Etapa 4**.

**Etapa 4.** El objeto B no está marcado. Entonces:

- a.  $ObjetivosPrevios = \{M, K, C\}$ .
- b. Se elige como objetivo en curso el único objeto no marcado, B.
- c. Se marca el objeto B. Por ello,  $ObjetosMarcados = \{D, E, F, L, M, K, C, A, B\}$ .
- d. Se va al **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo B pero no los  $ObjetivosPrevios \{M, K, C\}$ . Como no se encuentra ninguna regla, se va a la **Etapa 5**.

**Etapa 5.** Puesto que el objetivo en curso B no coincide con el inicial M, se pregunta al usuario el valor del objetivo en curso B. Supóngase que se da un valor cierto a B, entonces se hace  $B = \text{cierto}$  y se va a la **Etapa 6**.

**Etapa 6.** Como el objetivo en curso B no coincide con el inicial M, se designa el objetivo previo C como objetivo en curso y se elimina de  $ObjetivosPrevios$ . Por ello,  $ObjetivosPrevios = \{M, K\}$  y se va a la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo en curso C pero no los anteriores {M, K}. Se encuentra la Regla 1, por lo que se va a la **Etapa 3**.

**Etapa3.** Puesto que A = cierto y B = cierto, entonces C = cierto por la Regla 1. Ahora se va a la **Etapa 6**.

**Etapa 6.** El objetivo en curso C no coincide con el inicial M. Entonces, se designa el objetivo previo K como objetivo en curso y se elimina de *ObjetivosPrevios*. Por ello, *ObjetivosPrevios* = {M} y se va a la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo en curso K pero no los anteriores {M}. Se encuentra la Regla 4, por lo que se va a la **Etapa 3**.

**Etapa 3.** La Regla 4 no puede concluir puesto que el valor del objeto G es desconocido. Por tanto, se va a la **Etapa 4**.

**Etapa 4.** El objeto G no está marcado. Entonces:

- a. *ObjetivosPrevios* = {M, K}.
- b. El único objeto no marcado G se elige como objetivo en curso.
- c. Se marca el objeto G. Por ello, *ObjetosMarcados* = {D, E, F, L, M, K, C, A, B, G}.
- d. Se va a la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo en curso G pero no los anteriores {M, K}. Se encuentra la Regla 2, por lo que se va a la **Etapa 3**.

**Etapa 3.** Puesto que D = cierto, E = cierto y F = cierto, entonces G = cierto por la Regla 2. Ahora se va a la **Etapa 6**.

**Etapa 6.** El objetivo en curso G no coincide con el inicial M. Entonces, Se designa el objetivo previo K como objetivo en curso y se elimina de *ObjetivosPrevios*. Por ello, *ObjetivosPrevios* = {M} y se va a la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo en curso K pero no los anteriores {M}. Se encuentra la Regla 4, por lo que se va a la **Etapa 3**.

**Etapa 3.** Puesto que C = cierto y G = cierto, entonces K = cierto por la Regla 4. Seguidamente se va a la **Etapa 6**.

**Etapa 6.** El objetivo en curso K no coincide con el inicial M. Entonces, se designa el objetivo previo M como objetivo en curso y se elimina de *ObjetivosPrevios*. Por ello, *ObjetivosPrevios* =  $\emptyset$  y se va a la **Etapa 2**.

**Etapa 2.** Se busca una regla activa que incluya el objetivo en curso M. Se encuentra la Regla 6, por lo que se va a la **Etapa 3**.

**Etapa 3.** Puesto que K = cierto y L = cierto, entonces M = cierto por la Regla 6. Ahora se va a la **Etapa 6**.

**Etapa 6.** El objetivo en curso M coincide con el inicial. En consecuencia se va a la **Etapa 7.**

**Etapa 7.** El algoritmo devuelve el valor  $M = \text{cierto}$ .

### 3.2.5. Compilación de Reglas

Cuando datos y objetivos se han determinado previamente, las reglas pueden ser compiladas, es decir, pueden escribirse los objetivos en función de los datos para obtener las llamadas *ecuaciones objetivos*.

Para entender mejor este concepto se presenta el siguiente:

**Ejemplo.** Considérense las reglas expuestas en el **Gráfico 7** y su representación en el **Gráfico 8**, además, denotando al símbolo  $\wedge$  como el operador lógico y. Supóngase también que son conocidos los valores de los objetos A, B, D, E, F, H e I y que los demás objetos, C, G, J, K, L y M son los objetivos a los cuales se quiere llegar; entonces utilizando las seis reglas predefinidas pueden obtenerse las siguientes *ecuaciones objetivos*:

- a. La *Regla 1* implica la ecuación  $C = A \wedge B$ .
- b. La *Regla 2* implica la ecuación  $G = D \wedge E \wedge F$ .
- c. La *Regla 3* implica la ecuación  $J = H \wedge I$ .
- d. La *Regla 4* implica la ecuación  $K = (C \wedge G)$  pero como  $C = A \wedge B$  y  $G = D \wedge E \wedge F$ , entonces queda que  $K = (A \wedge B) \wedge (D \wedge E \wedge F)$ .
- e. La *Regla 5* implica la ecuación  $L = (G \wedge J)$  pero siendo  $G = (D \wedge E \wedge F)$  y  $J = (H \wedge I)$ , entonces  $L = (D \wedge E \wedge F) \wedge (H \wedge I)$ .
- f. La *Regla 6* implica la ecuación  $M = (K \wedge L)$  y tomando las ecuaciones obtenidas en (d) y (e) resulta que  $M = (A \wedge B) \wedge (D \wedge E \wedge F) \wedge (H \wedge I) = A \wedge B \wedge D \wedge E \wedge F \wedge H \wedge I$ .

Las últimas tres ecuaciones, (d), (e) y (f); equivalen, respectivamente a las siguientes tres reglas:

- a. *Regla 4.a. Si A y B y D y E y F, entonces K.*
- b. *Regla 5.a. Si D y E y F y H e I, entonces L.*
- c. *Regla 6.a. Si A y B y D y E y F y H e I, entonces M.*

En consecuencia, si los objetos {A, B, D, E, F, H, I} toman el valor cierto, inmediatamente se concluye, a partir de las *Reglas 4.a., 5.a. y 6.a.*, que los objetos {K, L, M} deben ser ciertos.

### 3.3. Subsistema de Control de Coherencia

Hay casos en la práctica en donde las situaciones son demasiado complejas y, aún los expertos, pueden dar información inconsistente, es decir ambigua o contradictoria; en estos casos es donde el SS.EE. debe estar preparado para evitar el ingreso de este tipo de información a la Base de Conocimientos, provocando que el sistema funcione de manera poco satisfactoria y llegue a conclusiones absurdas.

Por tanto, el objetivo del Subsistema de Control de Coherencia consiste en: (i) ayudar al usuario a no dar hechos inconsistentes mediante restricciones que debe cumplir la información solicitada, y (ii) evitar que entre en la Base de Conocimientos cualquier tipo de conocimiento inconsistente o contradictorio.

El control de coherencia se realiza sobre los hechos y las reglas que posee el sistema, como así también los hechos y las reglas que ingresan al mismo.

**Definición.** Un conjunto de reglas se denomina coherente si existe, al menos, un conjunto de valores de todos los objetos que producen conclusiones no contradictorias.

En consecuencia, basta con que algunos de los valores de los objetos de ese conjunto de reglas conduzca a conclusiones no contradictorias para que el conjunto de reglas se considere *coherente*.

Cabe aclarar que un conjunto de reglas puede ser coherente y aun así algunos conjuntos de valores pueden producir conclusiones inconsistentes; estos conjuntos de valores reciben el nombre de *valores no factibles*.

**Definición.** Se dice que un valor  $a$  para un objeto  $A$  no es factible si las conclusiones obtenidas al hacer  $A = a$  contradicen cualquier combinación de valores de los restantes objetos.

Cuando el Subsistema de Control de Coherencia detecta estos valores, automáticamente los elimina de la lista de posibles valores de ese objeto para que el mecanismo de inferencia no pueda llegar a conclusiones absurdas ni el usuario seleccione valores no factibles.

La revisión de consistencia, labor del Subsistema de Control de Coherencia, debe ser realizada antes de incorporar una nueva regla a la Base de Conocimientos, en caso de que la regla fuese consistente sería agregada a la Base de Conocimientos, sino, se devolvería al experto para su corrección.

La coherencia de los hechos también debe ser verificada; ya que el usuario puede ingresar al sistema, (a propósito, por desconocimiento o por error) hechos que no son consistentes con el conjunto de reglas y/o el conjunto de hechos existentes en cada etapa del proceso.

Existen algunas estrategias mediante las cuales puede lograrse dicha coherencia, éstas son:

1. Eliminar todos los valores no factibles de los objetos, una vez detectados. Cuando se pregunte al usuario por información sobre los valores de un conjunto de objetos, el sistema experto debería aceptar sólo los valores de cada objeto que sean consistentes con las reglas y con el conocimiento previo.
2. El motor de inferencia debe comprobar que los hechos conocidos no contradicen el conjunto de reglas.
3. Suministrar al usuario una lista de objetos a los que no se ha asignado valores previamente.
4. Para cada uno de los objetos, mostrar y aceptar sólo sus valores factibles.
5. Actualizar continuamente la Base de Conocimientos, es decir, tan pronto como se dé un hecho o se obtenga una conclusión, y eliminar los valores no factibles; el Motor de Inferencia obtiene todas las conclusiones posibles examinando, y posiblemente concluyendo, las reglas tan pronto como una simple unidad de información llega al sistema.

Es importante tener en cuenta que el orden en que se dan los hechos también puede ocasionar inconsistencias ya que es posible que al dar un hecho y luego dos hechos juntos, éstos últimos se contradigan con lo cual se debería ingresar uno u otro pero no ambos.

Para evitar este problema es necesaria la actualización constante de la Base de Conocimientos junto a la, ya mencionada, eliminación de valores no factibles.

### **3.4. Subsistema de Explicación o Justificación**

Es un clásico ejemplo aquel en que una persona va al médico para que el diagnostique su enfermedad y el médico, por medio de los síntomas de su paciente, llega a la conclusión de que posee tal enfermedad específica o un conjunto de posibles enfermedades, en cuyo caso el paciente es enviado a realizarse estudios clínicos para obtener más información y así poder diagnosticar cuáles y cuántas enfermedades padece el paciente.

Pero ni el paciente ni sus familiares se quedan conformes con el diagnóstico, sino que exigen al médico una explicación de porqué diagnosticó dicha enfermedad o enfermedades; es en ese momento cuando el médico demuestra lo que sabe justificando su diagnóstico mediante la enumeración de las "reglas" aprendidas en la universidad, donde estudió, más las aprendidas en el ejercicio de su profesión que le sirvieron para llegar al diagnóstico de su paciente.

Esta situación es también deseable en un sistema experto ya que no es suficiente con que el sistema obtenga una conclusión supuesta como verdadera, sino que debe explicar o justificar el camino que tomó para llegar a ella, tanto para brindar confianza y transparencia al usuario como al experto que esté a cargo de la actualización de los conocimientos del sistema.

Esta labor es función del Subsistema de Explicación o Justificación. En los SS.EE. basados en Reglas esto se reduce a la simple tarea de enunciar las reglas que utilizó la Máquina de Inferencia para llegar a una determinada conclusión.

#### **4. Sistemas Expertos basados en Probabilidad**

Para el desarrollo de esta sección se suponen conocidos, entre otros, los conceptos siguientes:

1. Medida de Probabilidad.
2. Distribuciones de probabilidad.
3. Dependencia e independencia de variables.
4. Teorema de Bayes.

A los efectos prácticos las fórmulas utilizadas sólo serán mencionadas y no se demostrarán ni se desarrollarán extensivamente. Para ampliar sobre el cálculo de probabilidades y otros temas relacionados ver [8].

La incertidumbre es uno de los inconvenientes con que se encuentra cualquier persona al momento de resolver un problema o al tomar una decisión; en general, los conocimientos que posee una persona en un momento dado poseen un cierto grado de incertidumbre.

Las razones por las cuales se presenta incertidumbre en los conocimientos son, entre otras:

1. *Conocimiento incierto.* El experto puede tener solamente un conocimiento heurístico o empírico respecto a algunos aspectos del dominio.
2. *Datos inciertos.* Aun cuando se tenga certidumbre del conocimiento por ejemplo de una relación causa-efecto, la certeza de los datos puede ser cuestionable.
3. *Información incompleta.* Con frecuencia es necesario tomar decisiones con información incompleta. A veces debemos asumir alguna información "por defecto".
4. *Azar.* En el mundo real existen hechos que se presentan de forma aleatoria, por ejemplo, al arrojar una moneda el hecho de que salga cara o cruz es aleatorio; también es posible que los datos posean valores aleatorios (variables aleatorias) como por ejemplo el primer premio de la quiniela puede asumir cualquier número entero entre 00000 y 99999, aunque muchos consideren números tres cifras.

Los SS.EE. basados en reglas poseen una incapacidad de tratar situaciones que posean un cierto grado de incertidumbre ya que las reglas con las que trabajan son deterministas; como solución a este problema se comenzaron a utilizar SS.EE. basados en probabilidad, es decir, cuya Base de Conocimientos contenga incertidumbre utilizando la probabilidad para medir esa incertidumbre.

El uso de las probabilidades en los SS.EE. no tuvo mucho éxito en un principio, debido al uso incorrecto de las hipótesis de independencia utilizadas para reducir la complejidad de los cálculos. Sin embargo, con la aparición de los redes probabilísticas, como las Bayesianas y las Markovianas, se han logrado medidas más intuitivas y aceptadas para tratar la incertidumbre [7].

Existen otros métodos para lograr dicha medición de la incertidumbre: el razonamiento monotónico, los factores de certeza, procedimientos basados en lógica difusa, la teoría de Dempster-Shafer, etc.

Lindley [10], por ejemplo, dice: "*La única descripción satisfactoria de la incertidumbre es la probabilidad. Esto quiere decir que toda afirmación incierta debe estar en forma de una probabilidad, que varias incertidumbres deben ser combinadas usando las reglas de la probabilidad, y que el cálculo de probabilidades es adecuado para manejar situaciones que implican incertidumbre. En particular, las descripciones alternativas de la incertidumbre son innecesarias.*"

#### 4.1. Reglas Generalizadas

La medida de probabilidad puede ser utilizada para medir la incertidumbre extendiendo los SS.EE. basados en reglas para que puedan abarcar situaciones de incertidumbre. Dado un SS.EE. basado en reglas, pueden usarse, en lugar de las reglas deterministas, reglas generalizadas; éstas se obtienen asociando una probabilidad a una afirmación; por ejemplo:

1.a) **Regla 1:** Si  $A$  es verdadero, entonces  $B$  es verdadero.

Mediante la asociación de un probabilidad a esta afirmación, se obtiene que:

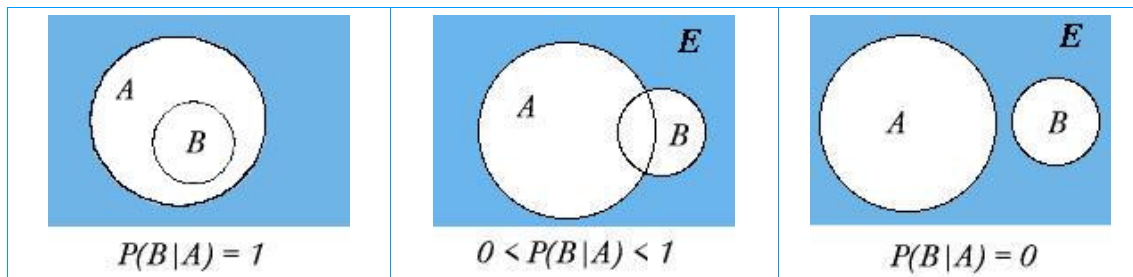
1.b) **Regla 2:** Si  $A$  es verdadero, entonces la probabilidad de que  $B$  sea verdadero es  $p(B)=\neg$ ; con  $0P\neg P1$ .

Es fácilmente observable que cuando  $\neg$  valga 1 (uno) la **Regla 2** será igual a la **Regla 1** por lo que se deduce que esta última es un caso particular de la **Regla 2**. Luego, cuando  $\neg$  se encuentra entre los valores uno y cero, es decir en situaciones de incertidumbre, la **Regla 1** no es apropiada, es por eso que debe utilizarse la **Regla 2** como una regla general o *generalizada*.

El valor de  $\neg$  determina el nivel de implicación según lo siguiente:

1. Implicación fuerte ( $\neg = 1$ ). **Gráfico 9a.**
2. Implicación débil ( $0P\neg P1$ ). **Gráfico 9b.**

3. No implicación ( $\neg = 0$ ). **Gráfico 9c.**



**Gráfico 9a.** La inclusión del conjunto B en A representa una implicación fuerte.

**Gráfico 9b.** La intersección de los conjuntos A y B representa la implicación débil.

**Gráfico 9c.** Los conjuntos disjuntos es el caso de la no implicación.

Además de la mera medida de probabilidad son necesarias las fórmulas de agregación para combinar la incertidumbre de los objetos en las premisas con la de las reglas para obtener la incertidumbre de los objetos en las conclusiones y también es necesario asociar probabilidades a priori a los objetos del modelo, y probabilidades condicionales a las reglas. Esta es la razón por la cual debe utilizarse algún método de propagación de probabilidades al presentarse nueva evidencia para que sea actualizada.

Para propagar las incertidumbres de la evidencia observada son necesarias hipótesis de independencia condicional que pueden no estar justificadas.

Una de las alternativas de la utilización de la medida de probabilidad es describir las relaciones entre objetos (variables) mediante una función de probabilidad conjunta.

Los SS.EE. que utilizan esta medida de incertidumbre son llamados SS.EE. de tipo probabilístico.

En el campo de la medicina los SS.EE. fueron ampliamente utilizados, principalmente, para el diagnóstico de enfermedades. En este tipo de problemas se trata de determinar qué enfermedades (una o varias) padece el paciente en base a los síntomas observados en él.

La dificultad del diagnóstico es que sólo se observan un subconjunto de síntomas, y basándose en ellos se debe realizar el diagnóstico con un grado de certeza razonable. En términos lógicos el problema se presenta de la siguiente manera. Dado que un paciente presenta un subconjunto de  $k$  síntomas<sup>2</sup>  $S_1 = s_1, \dots, S_k = s_k$ ; se desea conocer ¿cuál es la enfermedad más probable que el paciente padezca?. La respuesta se obtiene calculando la probabilidad condicional de que dicho paciente tenga la enfermedad  $e_i$ , dado que posee los síntomas  $s_i$ . Es decir,  $p(E=e_i | S_1 = s_1, \dots, S_k = s_k)$ .

Los SS.EE. probabilísticos pueden utilizarse para la resolución de otros tipos de problemas gracias a que:

1. Pueden memorizar información. Se puede almacenar y recuperar información de la base de datos.

<sup>2</sup> Las variables se denotan con mayúsculas y sus respectivos valores con minúscula.

2. Pueden contar o calcular las frecuencias absolutas y relativas de cualquier subconjunto de variables a partir de la base de datos. Estas frecuencias pueden utilizarse para calcular las probabilidades condicionales  $p(e_i | s_1, \dots, s_k)$  anteriormente expuesta.
3. Pueden aprender de la experiencia. Al presentarse nuevos casos es posible actualizar la base de datos incluyéndolos en la misma.
4. Pueden tomar, o ayudar al experto humano a tomar, decisiones tales como:
  - 4.a) Si se posee suficiente información para arribar a una conclusión satisfactoria.
  - 4.b) Si se necesita nueva información, es decir, más datos y si es así, qué debe preguntarse para obtener la información necesaria para concluir.

Para los fines prácticos sólo se desarrollarán los componentes Base de Conocimientos, Motor de Inferencias y el Sistema de Control de Coherencia de los SS.EE. basados en probabilidad.

## 4.2. La Base de Conocimientos

En los SS.EE probabilísticos la BC se construye relacionando las variables  $\{X_1, \dots, X_n\}$ , mediante la función de probabilidad conjunta definida sobre ellas,  $p(x_1, \dots, x_n)$ .

Es posible realizar un modelo general basado en la especificación directa de la función de probabilidad conjunta; asignando un valor numérico, llamado parámetro, a cada una de las posibles combinaciones de valores de las variables. Esto lleva a un problema debido a la incapacidad que poseen las computadoras de almacenar números exponencialmente grandes. El gran número de parámetros necesarios para, por ejemplo,  $n$  variables que asumen únicamente dos posibles valores ("presente" o "no presente"), es de  $2^n$  parámetros. Esta fue la razón por la cual los SS.EE. basados en probabilidades, en un principio, fueron criticados.

Diariamente se presentan situaciones en las que las variables o conjuntos de variables pueden ser independientes o dependientes entre sí; esto permite, teniendo en cuenta la estructura de independencia de las variables, obtener un modelo más general reduciendo de esta manera la cantidad de parámetros necesarios.

Cuatro modelos que ejemplifican dicha reducción de parámetros se presentan a continuación:

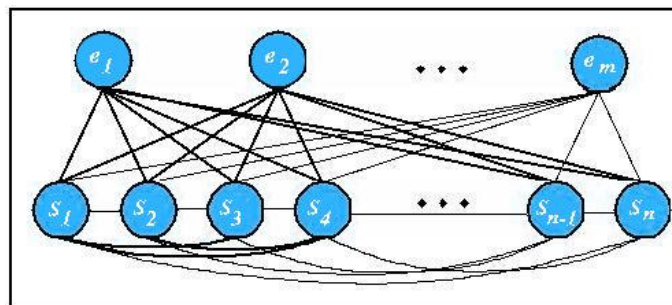
1. El Modelo de Síntomas Dependientes (MSD).
2. El Modelo de Síntomas Independientes (MSI).
3. El Modelo de Síntomas Relevantes Independientes (MSRI).
4. El Modelo de Síntomas Relevantes Dependientes (MSRD).

Estos modelos se aplican para un fin determinado, principalmente en el campo médico. Sin embargo, existen otros modelos más generales y potentes como:

1. Las Redes de Markov.
2. Las Redes Bayesianas.
3. Modelos especificados condicionalmente.

### 4.2.1. Modelo de Síntomas Dependientes

Para este modelo se realizan las siguientes suposiciones: que los síntomas son dependientes pero que las enfermedades son independientes entre sí, dados los síntomas. Esto se ilustra en la **Gráfico 10**.



**Gráfico 10.** Relaciones entre síntomas y enfermedades en el MSD.

La probabilidad conjunta para este modelo puede escribirse como:

$$p(e_i, s_1, \dots, s_n) = p(s_1, \dots, s_n)p(e_i/s_1, \dots, s_n). \quad (4.1)$$

Esta fórmula se deduce de hacer  $X=\{E\}$  e  $Y=\{S_1, \dots, S_n\}$  en la siguiente ecuación:

$$p(x, y) = p(y)p(x/y). \quad (4.2)$$

Utilizando la fórmula  $p(x_1, \dots, x_n) = p(X_1 = x_1, \dots, X_n = x_n)$ , se deduce la primera de las siguientes ecuaciones:

$$p(e_i | s_1, \dots, s_n) = \frac{p(e_i, s_1, \dots, s_n)}{p(s_1, \dots, s_n)} \quad (4.3)$$

$$= \frac{p(e_i)p(s_1, \dots, s_n | e_i)}{p(s_1, \dots, s_n)} \quad (4.4)$$

$$\propto p(e_i)p(s_1, \dots, s_n | e_i)$$

En la última expresión de la fórmula anterior se aprecia que los parámetros necesarios para la base de datos del MSD son:

1. Las probabilidades marginales  $P(e_i)$ , para todos los valores posibles de  $E$ ,
2. Las probabilidades condicionales  $P(s_1, \dots, s_n | e_i)$  de los síntomas para cada una de las enfermedades, cuyos valores pueden estimarse a partir de la información objetiva dada por las frecuencias de enfermedades y síntomas en la población.

En síntesis, para  $m$  enfermedades y  $n$  síntomas binarios, la función de probabilidad marginal de  $E$ ,  $P(e_i)$ , depende de  $m-1$  parámetros, ya que la suma de los parámetros debe dar uno. Para las probabilidades condicionales se necesitan especificar  $2^n-1$  parámetros para cada valor posible de  $E$ , siendo  $m$  la cantidad de enfermedades resulta:  $m(2^n-1)$  parámetros.

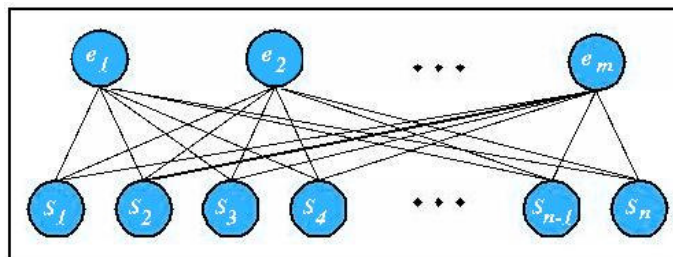
En total, el modelo, necesita  $m-1 + m(2^n-1) = m2^n-1$  parámetros.

El principal problema que posee el MSD es el número elevado de parámetros que se requiere a medida que aumenta la cantidad de valores (opciones o niveles) que puede asumir un síntoma, por ejemplo: fiebre alta, fiebre media, fiebre baja y sin fiebre.

#### 4.2.2. Modelo de Síntomas Independientes

Para solucionar el problema del MSD se procedió a simplificarlo; el resultado de esta simplificación consiste en suponer que, para una enfermedad dada, los síntomas son condicionalmente independientes entre sí.

Las relaciones de dependencia son representadas en la **Gráfico 11**.



**Gráfico 11.** Relaciones entre síntomas y enfermedades en el MSI.

Debido a la suposición hecha anteriormente se tiene la siguiente expresión:

$$p(s_1, \dots, s_n | e_i) = \prod_{j=1}^n p(s_j | e_i). \quad (4.5)$$

Es por esto que la función de probabilidad conjunta de la enfermedad  $E$  dados los síntomas  $s_1, \dots, s_n$ , puede escribirse así:

$$p(e_i | s_1, \dots, s_n) = \frac{p(e_i)p(s_1, \dots, s_n | e_i)}{p(s_1, \dots, s_n)} \quad (4.6)$$

$$= \frac{p(e_i) \prod_{j=1}^n p(s_j | e_i)}{p(s_1, \dots, s_n)} \quad (4.7)$$

$$\propto p(e_i) \prod_{j=1}^n p(s_j | e_i).$$

Sustituyendo la expresión (4.6) en (4.1) se obtiene el MSI.

En la ecuación (4.6) se observa la modificación de las probabilidades de todas las enfermedades cuando se conocen nuevos síntomas debido a la hipótesis de independencia expuesta anteriormente.

De la ecuación (4.7) puede distinguirse que los parámetros necesarios para la base de conocimientos son:

1. Las probabilidades marginales  $P(e_i)$ , para todos los valores posibles de la enfermedad  $E$ .
2. Las probabilidades condicionales  $P(s_j | e_i)$ , para todos los valores posibles del síntoma  $s_j$  y la enfermedad  $E$ .

Gracias a la hipótesis de independencia de los síntomas, el número de parámetros se reduce considerablemente. Es decir, que con  $m$  enfermedades posibles y  $n$  síntomas binarios, el número total de parámetros es

$$m(n+1) - 1 \quad (4.8)$$

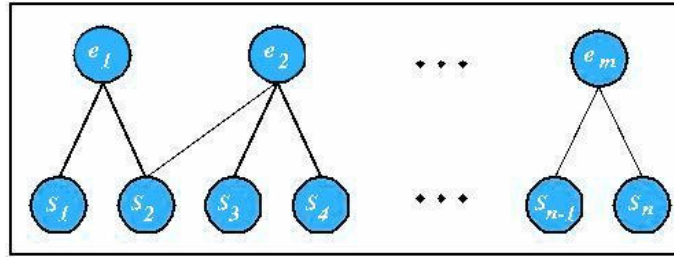
Con lo cual se obtiene un número menor de parámetros que en el MSD por ser:

$$m(n+1) - 1 < m2^n - 1, \text{ ya que el factor } n+1 < 2^n.$$

### 4.2.3. Modelo de Síntomas Relevantes Independientes

Todavía es posible reducir aún más la cantidad de parámetros suponiendo que cada enfermedad tiene un número reducido de síntomas relevantes. Para cada enfermedad  $E$ , es decir para cada valor de  $e_i$ , se seleccionan sólo una cantidad reducida de síntomas relevantes para cada una de las enfermedades y el resto de los síntomas no relevantes se suponen independientes para ese valor (enfermedad) de  $E$ .

Las relaciones de independencia o dependencia se ilustra en la **Gráfico 12**.



**Gráfico 12.** Relaciones entre síntomas y enfermedades en el MSRI.

Para simplificar la notación se tomará que  $s_1, \dots, s_{r_i}$  son los síntomas relevantes para la enfermedad  $e_i$  y que los síntomas  $s_{r_i+1}, \dots, s_n$  son los síntomas irrelevantes para la enfermedad  $e_i$ . Este modelo supone que las probabilidades condicionales  $P(s_j | e_i)$ , son idénticas para todos los síntomas irrelevantes de la enfermedad  $e_i$ . Por lo tanto la función de probabilidad conjunta de la enfermedad  $e_i$  dados los síntomas  $s_1, \dots, s_n$  puede escribirse como sigue:

$$\alpha p(e_i) \prod_{j=1}^{r_i} p(s_j | e_i) \prod_{j=r_i+1}^n p_j \quad (4.9)$$

$$\begin{aligned} p(e_i | s_1, \dots, s_n) &= \frac{p(e_i) p(s_1, \dots, s_n | e_i)}{p(s_1, \dots, s_n)} \\ &= \frac{p(e_i) \prod_{j=1}^{r_i} p(s_j | e_i) \prod_{j=r_i+1}^n p(s_j | e_i)}{p(s_1, \dots, s_n)} \\ &= \frac{p(e_i) \prod_{j=1}^{r_i} p(s_j | e_i) \prod_{j=r_i+1}^n p_j}{p(s_1, \dots, s_n)} \end{aligned} \quad (4.10)$$

En la expresión (4.9),  $p_j = p(s_j | e_i)$ , que es la misma para todas las enfermedades para las que  $s_j$  es irrelevante. Haciendo la sustitución de (4.8) en (4.1) se obtiene el MSRI.

De (4.9) se deduce que los parámetros necesarios para la base de conocimientos de este modelo son:

1. Las probabilidades marginales  $p(e_i)$ , para todos los valores posibles de la enfermedad  $E$ .

2. Las probabilidades condicionales  $p(s_j | e_i)$ , para cada valor posible de  $E$  y cada uno de sus correspondientes síntomas relevantes.
3. Las probabilidades  $p_j$ , para cada valor posible de  $E$  que tiene al menos un síntoma irrelevante. Esto implica que  $p_j = p(s_j | e_i)$  es idéntica para todos los síntomas irrelevantes  $e_i$ .

El número de parámetros para el MSRI se puede deducir de la fórmula (4.8) para calcular la cantidad de parámetros en el MSI. Para ello se deben imponer algunas restricciones adicionales en el MSI. Debido a que en el MSRI las probabilidades  $p(s_j | e_i)$  deben ser las mismas para todos los síntomas irrelevantes para las enfermedades  $e_i$ . El número de restricciones es:

$$a - n + \sum_{j=1, \dots, n} n_j; \text{ con } j=1, \dots, n \quad (4.11)$$

donde  $n_j$  es el número de enfermedades para las que  $S_j$  es irrelevante. Por lo tanto el número de parámetros total para el MSRI es igual a (4.8) menos (4.11):

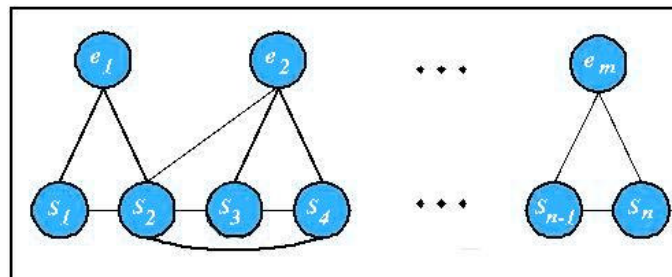
$$m(n + 1) - 1 + n - a - \sum_{j=1, \dots, n} n_j; \text{ con } j=1, \dots, n \quad (4.12)$$

#### 4.2.4. Modelo de Síntomas Relevantes Dependientes

Uno de los problemas del modelo anterior es el bajo nivel de realismo que posee, ya que los síntomas asociados a ciertas enfermedades suelen producirse en grupos o síndromes. Es por eso que puede ser poco razonable suponer que los síntomas relevantes son independientes. El presente modelo evita ese inconveniente.

A diferencia del anterior modelo, el MSRD, no obliga a que los síntomas relevantes sean independientes, dada la correspondiente enfermedad, por lo que se supone que solo los síntomas irrelevantes son independientes.

Lo expuesto en el párrafo anterior se ilustra en la **Gráfico 13**.



**Gráfico 13.** Relaciones entre síntomas y enfermedades en el MSRD.

Suponiendo que  $s_1, \dots, s_{r_i}$  son relevantes para la enfermedad  $e_i$ , y que los restantes síntomas  $s_{r_i+1}, \dots, s_n$  son irrelevantes; se tiene, entonces, que la función de probabilidad conjunta de  $e_i$  dados los síntomas  $s_1, \dots, s_n$  puede expresarse de la siguiente manera:

$$\begin{aligned}
 p(e_i | s_1, \dots, s_n) &= \frac{p(e_i)p(s_1, \dots, s_{r_i} | e_i) \prod_{j=r_i+1}^n p(s_j | e_i)}{p(s_1, \dots, s_n)} \\
 &= \frac{p(e_i)p(s_1, \dots, s_{r_i} | e_i) \prod_{j=r_i+1}^n p_j}{p(s_1, \dots, s_n)}
 \end{aligned} \tag{4.13}$$

$$p(e_i)p(s_1, \dots, s_{r_i} | e_i) \prod_{j=r_i+1}^n p_j \tag{4.14}$$

Donde  $p_j = p(s_j | e_i)$ , siendo la misma para todas las enfermedades para las que  $s_j$  es irrelevante. Realizando la sustitución hecha anteriormente en los modelos previos, es decir, (4.10) en (4.1), se obtiene el MSRD. Para este modelo son necesarias las siguientes probabilidades:

1. Las probabilidades marginales  $p(e_i)$ , para todos los posibles valores de la enfermedad  $E$ .
2. Las probabilidades condicionales  $p(s_1, \dots, s_{r_i} | e_i)$ , para todos los posibles valores de la enfermedad  $E$  y sus síntomas relevantes  $s_1, \dots, s_{r_i}$ .
3. Las probabilidades  $p_j$ , para cada valor posible de  $E$  que tenga al menos un síntoma irrelevante. Esto implica que  $p_j = p(s_j | e_i)$  coincide para todos los síntomas irrelevantes para  $e_i$ .

Por ello, si se tienen  $m$  enfermedades binarias y  $n$  síntomas binarios el número total de parámetros en el MSRD es:

$$m - 1 + n - a + \sum (2^{r_i} - 1) = n - 1 - a + \sum 2^r \tag{4.15}$$

Obsérvese que si  $r_i = r$  para todos los valores de  $e_i$ , entonces (4.15) resulta:

$$m2^r + n - 1 \tag{4.16}$$

También es importante destacar que si todos los síntomas son relevantes para todas las enfermedades el MSRD se convierte en el MSD.

En la **Tabla 5** se muestra una comparación del número de parámetros necesarios en cada modelo, teniendo en cuenta el caso en que se tienen  $m=100$  enfermedades binarias,  $n=200$  síntomas binarios y  $r=10$  síntomas relevantes por enfermedad.

<i>Modelo</i>	<i>Número de parámetros</i>	
	<i>Fórmula</i>	<i>Valor</i>
MSD	$m2^n - 1$	$> 10^{62}$
MSI	$m(n + 1) - 1$	20099
MSRI	$m(r + 1) n - 1$	1299
MSRD	$m2^r + n - 1$	102599

**Tabla 5.** Cantidad de parámetros necesarios en cada modelo si se tienen  $m=100$  enfermedades binarias,  $n=200$  síntomas binarios y  $r=10$  síntomas relevantes.

### 4.3. Motor de Inferencias

El conocimiento de los SS.EE. probabilísticos se clasifica en dos tipos:

1. El *conocimiento* está compuesto por el conjunto de variables y el conjunto de probabilidades asociadas necesarias para construir su función de probabilidad conjunta. Este tipo se almacena en la BC.
2. El otro tipo de conocimiento son los datos. Estos son un conjunto de valores de algunas variables conocidas por el usuario, es decir, la evidencia. La evidencia se almacena en la Memoria de Trabajo (MT).

El MI utiliza tanto la BC como la MT (los conocimientos y los datos o evidencia) para responder a las preguntas del usuario.

Como ejemplos de estas preguntas se pueden tomar las siguientes:

**Pregunta 1.** Un paciente, que aún no ha sido examinado por el doctor, desea saber ¿cuál es la enfermedad más probable que pueda padecer?. Para esta pregunta, la respuesta consiste en calcular la probabilidad marginal de la enfermedad  $E=e_i$  dado que el conjunto de síntomas es vacío, por no haber sido examinado aún por el doctor. Entonces la ecuación sería:

$$p(E=e_i), \text{ con } i = 1, \dots, m.$$

**Pregunta 2.** Ahora, el paciente ya ha sido examinado por el doctor y presenta  $k$  síntomas. ¿Cuál es la enfermedad más probable dado que padece un conjunto  $S$  de enfermedades?. La respuesta se obtiene calculando la probabilidad conjunta para cada enfermedad  $e_i$  dada la evidencia  $s_1, \dots, s_k$ :

$$p(E=e_i | s_1, \dots, s_k), i = 1, \dots, m.$$

En la primera pregunta la probabilidad marginal recibe el nombre de probabilidad *a priori* debido a que se calcula antes de conocer la evidencia, mientras que en la segunda pregunta, la probabilidad condicional se la llama *a posteriori* por ser calculada después de conocer la evidencia.

El MI posee tres tareas fundamentales en los SS.EE. probabilísticos:

1. Calcular las probabilidades condicionales cuando se conocen nuevos datos.
2. Actualizar las probabilidades condicionales:

$$p(e_i/s_1, \dots, s_k) = \frac{p(e_i, s_1, \dots, s_k)}{p(s_1, \dots, s_k)}; \quad i = 1, \dots, m \quad (4.17)$$

Para todos los valores posibles de los datos.

3. Decidir cuáles tienen probabilidades condicionales altas. De éstas se seleccionan unas pocas y se las muestra al usuario para que sean observadas y, así, obtenga las conclusiones pertinentes.

Es fácil observar que el Teorema de Bayes se puede aplicar para calcular las probabilidades condicionales cuando se tienen pocas variables (por ejemplo, enfermedades y síntomas) pero cuando este número es alto, lo más común en la realidad, se necesitan métodos y modelos más eficientes para calcular las probabilidades marginales y las probabilidades condicionales. Algunos de esos métodos, conocidos como *de propagación de evidencia o incertidumbre*, son las *Redes Bayesianas*, *Redes de Markov*, *Métodos Numéricos*, etc. En los Capítulos 8, 9 y 10 de [1] se encuentra un desarrollo de estos y otros métodos utilizados para la propagación de evidencia.

#### 4.4. Control de Coherencia

La presencia de incoherencias en la Base de Conocimientos y/o en la Memoria de Trabajo representa uno de los problemas que se encuentran en el uso de un SS.EE.

Esto se debe a varios motivos, por ejemplo:

1. Los expertos humanos pueden suministrar conocimiento incoherente. Por ejemplo al no especificar claramente las reglas a utilizar o al brindar hechos que aparentemente se contradicen; esto sucede muchas veces al simplificar el problema tomando sólo aquello más relevante quedando hechos que no lo son pero que hacen a la especificación puntual del problema.
2. El usuario puede suministrar datos incoherentes, ya sea por error o desconocimiento de los datos que debe proporcionar al sistema.
3. No hay un subsistema para controlar la coherencia que evite que llegue conocimiento inconsistente a la Base de Conocimiento y/o a la Memoria de Trabajo.

Para lograr la coherencia se especifican restricciones que deben cumplir los datos que ingresan al sistema; por ejemplo, una de las restricciones que debe estar presente en este tipo de sistemas es que las probabilidades, cualesquiera sean, deben estar acotadas entre cero y uno, de lo contrario se cometerá un error en los cálculos posteriores.

En el siguiente ejemplo se describen las restricciones para dos variables.

**Ejemplo.** Supóngase que se tienen solo dos variables binarias,  $E$  y  $S$ . Como se había deducido anteriormente las probabilidades necesarias para la base de conocimientos de cualquiera de los métodos anteriores son  $p(e)$ ,  $p(s)$ ,  $p(s | e)$ . Con esto el sistema comienza preguntando por los valores de las dos primeras. Estas deben satisfacer las restricciones siguientes:  $0 \leq p(e) \leq 1$  y  $0 \leq p(s | e) \leq 1$ . Luego el sistema debería preguntar por los valores de la probabilidad condicional informando al usuario debidamente de las restricciones que ésta debe cumplir.

Pero teniendo en cuenta la siguiente propiedad:

$$p(s | e') + p(s | e) = p(s) \quad (4.18)$$

Sólo es necesario conocer una de las probabilidades condicionales. Luego queda que hace falta conocer dos probabilidades más  $p(s' | e') + p(s | e')$ , pero éstas deben sumar uno, por lo que conociendo una de ellas basta para que el sistema calcule la restante.

## 5. Comparación entre los SS.EE. basados en Reglas y los basados en Probabilidades

En los siguientes párrafos se realiza una comparación de los dos tipos de SS.EE. expuestos en el presente trabajo. Dicha comparación se realiza teniendo en cuenta cada componente de los SS.EE. y describiendo sus ventajas y desventajas, analogías y diferencias.

### 5.1. Base de Conocimientos

El conocimiento en un SS.EE. basado en Reglas (SER) consiste en los objetos y el conjunto de reglas que relacionan dichos objetos.

Mientras que en el SS.EE. basado en Probabilidad (SEP) consiste en el espacio de probabilidad, que incluye las variables, sus posibles valores, y su función de probabilidad conjunta.

Además, los datos de ambos sistemas consisten en la evidencia asociada a los casos a analizar.

Para los SERs, la BC es fácil de implementar pero posee una escasa capacidad de almacenaje comparado con los SEPs, pero a su vez estos poseen un inconveniente que radica en la alta cantidad de parámetros necesarios, lo que hace difícil su especificación y definición.

## 5.2. Motor de Inferencia

En los SERs el Motor de Inferencia es rápido y fácil de implementar debido a la utilización de las estrategias de inferencia vistas en la **Sección 3.2**.

En los SEPs dicho componente es más complicado ya que se basa en la evaluación de las probabilidades condicionales utilizando uno o varios métodos propuestos por los diferentes tipos de SS.EE. El grado de dificultad dependerá del modelo elegido; los más simples son los modelos de independencia y los más complejos son los modelos de dependencia generales.

## 5.3. Subsistema de Explicación

En el caso de los SERs la explicación del camino recorrido para llegar a una conclusión se realiza con la simple enumeración de las reglas utilizadas en la inferencia.

Para el caso de los SEPs la información sobre qué variables influyen sobre otras, se encuentra codificada en la función de probabilidad conjunta. En este caso la explicación consiste en los valores relativos de las probabilidades condicionales que miden los grados de dependencia. Una comparación de las probabilidades condicionales para diferentes conjuntos de evidencia permite analizar sus efectos en las conclusiones.

## 5.4. Subsistema de Aprendizaje

Los SERs "aprenden" mediante la incorporación de nuevos objetos, nuevos conjuntos de valores factibles para los objetos, nuevas reglas modificaciones de los objetos existentes, de los conjuntos de valores posibles, o de las reglas.

En los SEPs el aprendizaje consiste en incorporar o modificar la estructura del espacio de probabilidad: variables, conjunto de posibles valores, o los parámetros (valores de probabilidades).

La **Tabla 6** muestra una comparación entre los SS.EE. basados en Reglas y los SS.EE. basados en Probabilidad utilizando un resumen de algunas componentes de cada tipo de SS.EE. y de la estructura, lógica o probabilística, en la que se basa.

Componente	Basados en Reglas	Probabilísticos
Base de Conocimiento	Objetos, reglas Hechos	Variables, FPC Hechos
Motor de Inferencia	Estrategias de inferencia Encadenamiento de reglas	probabilidad condicional métodos de evaluación
Subsistema de Explicación	Basado en reglas activas	Basado en probabilidad condicional
Aprendizaje	Cambio en objetos y reglas	Cambio en modelo probabilístico

**Tabla 6.** Comparación entre Sistemas Expertos basados en Reglas y Sistemas Expertos basados en Probabilidad.

## 7. Conclusiones

### 7.1. Contribuciones

El presente trabajo ha expuesto los diferentes tipos de Sistemas Expertos tratando de que el lector pueda conocer los Sistemas Expertos Difusos y los Sistemas Expertos basados en Redes Neuronales Artificiales, como así también otras técnicas de la Inteligencia Artificial (Minería de Datos y Algoritmos Genéticos).

También se ha logrado brindar una explicación simple de esta herramienta brindando los conceptos más importantes sin perder los detalles.

Se ha estructurado el presente trabajo para facilitar el aprendizaje del tema abordando, en los primeros capítulos, los conceptos más básicos que deben ser asimilados por el lector antes de comenzar con la lectura de los capítulos más avanzados.

Con el desarrollo del presente trabajo se ha conseguido elaborar una guía para el futuro desarrollo de sistemas de estas características y una base consistente para continuar el estudio de los mismos a un nivel más profundo.

### 7.2. Limitaciones

Dicho trabajo se ha enfocado en la presentación y explicación de conceptos básicos y generales sin profundizar en temas como la Teoría de Probabilidades, Lógica Difusa y Redes Neuronales, todos ellos, utilizados en algunos de los componentes de los SS.EE.

El tema "*Desarrollo de un SS.EE.*" quedó en la simple enumeración de las etapas junto con una breve explicación de cada una, evitando de esta manera la dispersión del lector y centralizando la atención en las características de un Sistema Experto, su funcionamiento, sus componentes y las funciones de cada uno.

Con el propósito de fijar los conceptos desarrollados a lo largo del presente trabajo se han presentado, en los casos pertinentes, ejemplos prácticos sin que ellos sugieran un fiel reflejo de los problemas reales a los cuales se deberá enfrentar un SS.EE.

## **8. Futuros Trabajos**

Basado en el desarrollo realizado del tema se presentan algunas signaturas que podrían ser tratadas de forma más amplia en futuros trabajos.

En relación con Inteligencia Artificial, se puede investigar y desarrollar con mayor profundidad áreas como la vida artificial, realidad virtual, reconocimiento de la voz e imágenes, entre otros temas relacionados.

Puntualmente, con respecto a los Sistemas Expertos, se puede extender la Sección 2.6 en la cual se trata la metodología de desarrollo de un SS.EE. incluyendo en dicho desarrollo un ejemplo práctico completo para la mejor comprensión del mismo.

También es posible investigar, en profundidad, las posibilidades y limitaciones de los Sistemas Expertos basados en Redes Neuronales y los Sistemas Expertos Difusos, con el propósito de encontrar nuevas aplicaciones de los mismos.

Y por último, teniendo en cuenta los desarrollos de software libre y código abierto, principalmente GNU/Linux, investigar las posibilidades de optimizar dicho sistema operativo utilizando Sistemas Expertos, Redes Neuronales o cualquier otra técnica de Inteligencia Artificial; esto sería más que un simple trabajo de investigación, sería la puerta que permita avanzar hacia una nueva generación, la de los Sistemas Operativos Inteligentes.

## Referencias

1. Castillo, E. Gutiérrez, J. M. y Hadi A. S. 1997. *Sistemas Expertos y Modelos de Redes Probabilísticas*.
2. Monserrat La Red Martínez, M. C. 2002. *Curso de Sistemas Expertos*. Universidad Nacional del Nordeste y Universidad Nacional de Misiones. Posadas, Misiones, Argentina.
3. Puigserver, S. Prats J. Ma. y Rovira J., editores. 1996. *OCEANO Uno Color Diccionario Enciclopédico*. OCEANO. Barcelona, España.
4. Mariño, S. I. 2003. *Tema1: Fundamentos*. Apuntes de la cátedra Inteligencia Artificial, Departamento de Informática, FaCENA, UNNE.
5. Causa, M. Klein, A. y Sort, R., editores. 1998. *Enciclopedia Didáctica de Computación OCEANO*. OCEANO. Barcelona, España. 63-74.
6. Hayes-Roth, F., Waterman, D. A., y Lenat, D. B., editores. 1983. *Building Expert Systems*. Addison-Wesley, Reading, MA.
7. Luger, G. F., y Stubblefield, W. A. 1989. *Artificial Intelligence and the Design of Expert Systems*. Benjamin/Cummings, Redwood City, CA.
8. Weiss, S. M. y Kulikowski, C. A. 1984. *A Practical Guide to Designing Expert Systems*. Rowan and Allanheld, Tolowa, N.J.
9. Mariño, S. I. 2003. Unidad 6: *Representación del conocimiento mediante razonamiento estadístico*. Apuntes de la cátedra Inteligencia Artificial, Departamento de Informática, FaCENA, UNNE.
10. Lindley, D. V. L.; editor 1988. *Computer Games*. Springer Verlag, New York.
11. Buckley, J. J.; Siler, W. and Tucker, D. 1986. *A Fuzzy Expert System. Fuzzy Sets and Systems*.
12. Yager, R. R.; Ovchinnikov, S.; Yong, R. M. and Nguyen, H. T. editores. 1987. *Fuzzy sets and Applications: Selected Papers by L. A. Zadeh*. John Wiley and Sons, New York.
13. Zadeh, L. A. 1983. *The Role of the Fuzzy Logic in the Management of Uncertainty in Expert Systems. Fuzzy Sets and Systems*.

## Bibliografía adicional

- Almond, R.G. 1995. *Graphical Belief Modeling*. Chapman and Hall, New York.
- Badiru, A. B. 1992. *Expert Systems Applications in Engineering and Manufatcuring*. Prentice-Hall, Englewood Cliffs.
- Bahill, A. T. 1991. *Verifynq and Validating Personal Computer - Based Expert Systems*. Prentice-Hall, Englewood Cliffs.
- Benchimol, G.; Levine P.; Pomerol, J. C. 1990. *Los sistemas Expertos en la Empresa*. Macrobit Editores, S.A. De C.V., Mexico D.F.
- Buchaman, B. G. and Shortliffe, E. H. 1984. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Reading, MA.
- Canavos, G. 1994. *Probabilidad y estadística*. Mc GrawHill.
- Castillo, E. and Alvarez, E. 1991. *Expert Systems: Uncertainty and Learning*. Computational Mechanics Publications and Elsevier Applied Science. London, U.K.
- Cheeseman, P. 1985. *In Defense of Proability*. In *Proceeding of the 9<sup>th</sup> International Joint Conference on Artificial Intelligence (IJCAI-85)*. Morgan Kaufmann Publishers, San Mateo, CA.
- De Groot, M. H. 1988. *Probabilidad y estadística*. Addison-Wesley Iberoamericana.
- D'lia, G. E. 1999. *Teoria de las Probabilidades, Estadística, Matemática*. Ed. Alsina.
- Durkin, J. 1994. *Expert Systems: Design and Development*. Maxwell Macmillan, New York.
- Feller, W. 1983. *Introducción a la Teoría de Probabilidades y sus aplicaciones*. Editorial Limusa.
- Garcia, O. N. and Chien, Y.-T.; editores. 1991. *Knowledge-Based Systems: Fundamentals and Tools*. IEEE Computer Society Press. Los Alamitos, CA.
- Hayes-Roth, F.; Waterman, D. A.; and Lenat, D. B.; editores. 1983. *Building Expert Systems*. Addison-Wesley, Reading, MA.
- Hayes-Roth, F. 1985. *Graph Theaory*. Addison-Wesley, Reading, MA.
- Kurinic, G. 2001. *Estadística, Probabilidades y Distribuciones*. Ed. Ediciones Cooperativas.

- Luger, G. F.; and Stubblefield, W. A. 1989. *Artificial Intelligence and the Design of Expert Systems*. Benjamin/Cummings, Redwood City, CA.
- Pedersen, K. 1989. *Expert Systems Programming: Practical Techniques for Rule-Based Expert Systems*. John Wiley and Sons, New York.
- Shafer, G. 1976. *Mathematical Theory of Evidence*. Princeton University Press, Princeton, N.J.
- Walpore, R.; Myers, R. 1992. *Probabilidad y estadística*. Mc GrawHill.
- Waterman, D. A. 1985. *A guide to Expert Systems*. Addison-Wesley, Reading, MA.
- Weiss, S. M. and Kulikowski, C. A. 1984. *A Practical Guide to Designing Expert Systems*. Rowman and Allanheld, Totowa, N.J.