



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO



FACULTAD DE INGENIERIA

**SISTEMA EXPERTO TUTORIAL
DE LOGICA DIFUSA**

T E S I S

QUE PARA OBTENER EL TITULO DE

INGENIERO EN COMPUTACION

P R E S E N T A :

MARIA DE LOS ANGELES PLAZA MUNGUIA



Director de tesis: Dr. Felipe Lara Rosano
Asesor: M.I. Nicolas Kemper Valverde

Ciudad Universitaria México, D. F.

1996

TESIS CON
FALLA DE ORIGEN

TESIS CON
FALLA DE ORIGEN



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

“ Todos los triunfos nacen cuando nos atrevemos a comenzar “

Eugene Wore

“ Aunque los expertos estén de acuerdo, bien pueden estar equivocados”

Bertrand Russell

“ Las oportunidades a menudo son cosas que no advertimos cuando se nos presentaron por primera vez “

Catherine Deneuve

“ Lo que distingue a un verdadero profesional es que dá más de lo que recibe “

Robert Kirby

“ No mires lo que das, sino el corazón con que lo das “

Alfonso Milagro

“ Quienes más se quejan de que no tienen tiempo son los que menos saben usarlo “

Aldo Cammarota

“ No basta con alcanzar la sabiduría, es preciso saber utilizarla “

Marco Tulio Cicerón

“ La única forma de hacer que un hombre sea digno de confianza, es confiar en él “

Henry Stimson

“Un profesional es quien puede realizar su mejor trabajo cuando se siente desganado“

Alistair Cooke

“ La búsqueda de la perfección a menudo detiene el progreso “

George Will

“ La felicidad que da el dinero, está en no tener que preocuparse por él “

José Ingenieros

“ Cualesquiera que hayan sido nuestros logros, alguien nos ayudó siempre a alcanzarlos “

Althea Gibson

AGRADECIMIENTOS

A la Universidad Nacional Autónoma de México, por su colaboración en mi formación profesional.

A la Facultad de Ingeniería, a quien la considero mi segunda casa.

Al Instituto de Ingeniería, por todo su apoyo, entusiasmo y facilidades brindadas para la realización de este trabajo de tesis.

Muy en especial a mi maestro y director de tesis, el **Dr. Felipe Lara Rosano** quien despertó en mi el interés por la Inteligencia Artificial y los Sistemas Expertos, así como toda su ayuda, entusiasmo, dedicación y paciencia en los momentos difíciles para mi, y sus valiosos consejos.

Al coordinador del Laboratorio de Inteligencia Artificial del Instituto y mi asesor de tesis, el **M.I. Nicolás Kemper Valverde** por toda su colaboración, entusiasmo, accesibilidad y consejos para la realización de este trabajo Gracias Nico !!!

A todos mis compañeros del Laboratorio de Inteligencia Artificial que ya no están conmigo, pero que siempre me apoyaron y a los que siguen por todo su entusiasmo, cariño y consejos a lo largo de mi trabajo.

A todos los integrantes de la Coordinación de Cómputo del Instituto, por todas sus facilidades brindadas para el desarrollo de este trabajo.

Siempre pensaba en cómo serían los agradecimientos de mi tesis, pero ahora que ha llegado el momento de la verdad no se ni qué decir

Antes que a nadie, le agradezco a Dios que me dio la vida !!!

A mis ancianos lindos “ mis padres queridos “ a quienes les debo todo lo que soy, su apoyo, cariño y confianza, que me hicieron seguir adelante. ¡Los quiero mucho!

A mis hermanos Salvador, Enrique y Eduardo porque siempre me apoyaron a lo largo de toda mi carrera. ¡Los quiero mucho escuincles!

A Edgar por su cariño, comprensión y apoyo para terminar este trabajo y seguir adelante. ¡Gracias patito!

A mis queridos amigos (casi hermanos) Oscar y Cristy, con quienes he contado siempre, tanto en las buenas como en las malas
(y ustedes cuando, ya se están tardando eh).

A mis abuelos, tíos y primos que siempre me apoyaron.

A todos mis amigos, que cerca o lejos siempre me impulsaron y apoyaron para seguir adelante. Gracias !!!

María de los Angeles Plaza Munguía

SISTEMA EXPERTO

TUTORIAL

DE

LÓGICA DIFUSA

MARIA DE LOS ANGELES PLAZA MUNGUÍA

ÍNDICE

INTRODUCCIÓN

Capítulo I Inteligencia Artificial y Sistemas Expertos

1. Inteligencia Artificial	1
1.1 Técnicas de base	2
1.2 Áreas de aplicación	3
1.3 Sistemas basados en el conocimiento	3
2. Sistemas Expertos	5
2.1 Introducción	5
2.2 Análisis del conocimiento	7
2.3 Definición de un sistema experto	7
2.4 Sistemas de producción	8
2.5 Estructura interna de un sistema experto	9
2.5.1 Base de conocimiento	12
2.5.2 Motor de inferencia	13
2.5.3 Base de datos	14
2.5.4 Interfaces de los sistemas expertos	15
2.5.4.1 Interfaz con el usuario	15
2.5.4.2 Interfaz con el ingeniero de conocimiento	17
2.5.4.3 Interfaces con base de datos	17
2.5.5 Control de ejecución	18
2.6 Ventajas de los sistemas expertos	18
2.7 Tipos de sistemas expertos	19
2.8 Arquitectura de los sistemas expertos	20
2.8.1 Sistemas expertos aislados	20
2.8.2 Sistemas expertos integrados	20
2.8.3 Sistemas expertos embebidos	21
3. Desarrollo de un Sistema Experto	21
3.1 Fases en la construcción de un sistema experto	21
3.2 Personas que intervienen en la construcción	23
3.3 Elección de la herramienta	24
3.4 Adquisición del conocimiento	25
4. Razonamiento	26
4.1 Inferencia	26
4.2 Control del razonamiento	27
4.3 Aprendizaje	29

Capítulo II El Aprendizaje y los Sistemas Expertos Tutoriales

1. Introducción	31
2. El Aprendizaje Tradicional	32
2.1 Aprendizaje por absorción	33
2.2 Consideraciones en el aprendizaje	33
2.3 Ambiente educativo	34
3. La Computadora en la Educación	35
3.1 Aprendizaje por computadora (tutoriales)	36
3.2 La computadora como herramienta para el aprendizaje	36
3.3 El software educativo	38
4. Sistemas Expertos Tutoriales	39
4.1 Módulos básicos de un sistema experto tutorial	40
4.2 Tipos de sistemas expertos tutoriales	43
4.3 Sistemas expertos tutoriales desarrollados	44

Capítulo III Lógica Difusa
Parte I Matemáticas Difusas

1. Conjuntos Difusos	47
1.1 Definiciones básicas	47
2. Medidas Difusas y Medidas de Fuzzificación	48
2.1 Medidas difusas	48
2.2 Medidas de fuzzificación	50
3. El Principio de Extensión y Aplicaciones	53
3.1 El principio de extensión	53
3.2 Operaciones algebraicas con números difusos	54
4. Relación Difusa y Gráficas Difusas	56
4.1 Relación difusa en conjuntos y conjuntos difusos	55
4.1.1 Composición de relaciones difusas	59
4.1.2 Propiedades de la composición min-max	60
4.1.2.1 Asociatividad	60
4.1.2.2 Reflexibilidad	61
4.1.2.3 Simetría	61
4.1.2.4 Transitividad	62
4.2 Gráficas difusas	62
5. Análisis Difuso	64
5.1 Funciones difusas en conjuntos difusos	64
5.2 Extremos de funciones difusas	65
6. Teoría de Posibilidad, Teoría de Probabilidad y Teoría de Conjuntos Difusos	66
6.1 Teoría de posibilidad	66
6.1.1 Conjunto difuso y distribuciones de posibilidad	66
6.1.2 Medidas posibles y necesarias	69
6.2 Probabilidad de eventos difusos	71
6.2.1 Probabilidad de un evento difuso como un escalar	71

6.2.2 Probabilidad de un evento difuso como un conjunto difuso	72
6.3 Posibilidad contra probabilidad	74

Parte II Aplicaciones de la Teoría de Conjuntos Difusos

7. Lógica Difusa y Razonamiento Aproximado	79
7.1 Variables lingüísticas	79
7.2 Lógica difusa	83
7.2.1 Revisión de la lógica clásica	83
7.3 Razonamiento aproximado	86
8. Aplicaciones de Lógica Difusa	88
8.1 Introducción	88
8.1.1 Algunos pioneros en el tema	88
8.1.2 Orígenes del pensamiento difuso (fuzzy)	89
8.2 Sistemas expertos en el control de procesos	90
8.2.1 Sistemas expertos y control difuso	90
8.2.1.1 Teoría de conjuntos difusos	90
8.2.1.2 Controladores lógicos difusos	94
8.2.1.3 Algoritmos genéticos	95
8.3 Ejemplos de aplicaciones de sistemas expertos en el control	99

Capítulo IV Desarrollo del Sistema Experto Tutorial SETLODI

1. Introducción	102
1.1 Definición del problema	102
1.2 Posibles soluciones	103
1.3 Por qué un sistema experto?	103
1.4 Análisis y formulación del sistema	105
2. Desarrollo de un Sistema Experto Tutorial	107
2.1 Proceso para implementar un Sistema Experto	108
2.1.1 Inicialización del proyecto	108
2.1.2 Evaluación de necesidades	109
2.1.3 Implementación	109
2.1.4 Depuración del sistema	110
3. Desarrollo del sistema experto tutorial SETLODI	110
3.1 Inicialización del proyecto	111
3.2 Evaluación de necesidades	111
3.2.1 Selección del shell de desarrollo	111
3.2.2 Tipo de inferencia utilizada	125
3.2.3 Requerimientos técnicos del sistema experto	125
3.3 Implementación	126
3.3.1 Diseño de la interfaz del usuario	126
3.3.2 Estructura general del sistema SETLODI	126
3.3.3 Proceso de evaluación	128

3.3.4 Bases de datos	129
3.4 Depuración del sistema	130
3.4.1 Operación general	130
3.4.2 Validación	145

Conclusiones

Bibliografía

INTRODUCCIÓN

INTRODUCCIÓN

El concepto de lógica difusa hay que asociarlo a Zadeh que, a partir de 1965, publica sus trabajos sobre teoría difusa, por lo que las fundamentales referencias deben ser suyas.

Para hablar de Lógica hay que partir de su posible definición: *Ciencia que estudia las condiciones formales de validez de una inferencia y, en general, de una argumentación cualquiera.* Dicho de otra manera más fácil *Lógica* es la formalización del pensamiento humano.

Para definir una lógica hay que empezar por determinar las *variables* o enunciados elementales y las *conectivas* que pueden unir a las variables para dar lugar a *sentencias* o enunciados compuestos.

En un segundo paso hay que enunciar los *axiomas* como enunciados básicos pertenecientes a la lógica que estamos definiendo, y las *reglas operativas* que permiten derivar un enunciado de otro.

Llegados a este punto se pueden enunciar *teoremas* que serán sentencias que mediante *demostración*, esto es, aplicación consecutiva de reglas operativas, se pueden obtener a partir de los axiomas. Se dice asimismo que una sentencia es una *tesis* si es o bien un axioma o bien un teorema.

El siguiente paso es la *semántica*, esto es, dar significado o interpretar las sentencias. Para ello hay que definir un *conjunto de valores semánticos*, con un mínimo de dos elementos, y un conjunto de *operaciones* cerradas con los elementos del conjunto, de modo que a cada variable le corresponda un valor semántico y a cada conectiva una operación. De este modo será posible calcular el valor semántico de cualquier sentencia.

En este punto hay que resaltar, por las implicaciones en el tema que nos ocupa, que aunque tradicionalmente el conjunto de valores semánticos es de dos elementos {verdadero, falso}, {1,0}, desde un punto de vista general no se impone ninguna restricción al número máximo de elementos, ni siquiera que sea finito.

Un concepto relacionado con la semántica es el de *tautología*: cuando dentro del conjunto de valores semánticos está el valor '1' o 'verdadero', se define tautología como aquella sentencia que para cualquier interpretación de sus variables la interpretación de la sentencia siempre es verdadera. Cuando la interpretación es siempre 'falsa' se llama *contradicción*.

La relación entre sentencias válidas o tesis, con su valor semántico es lo que lleva a los conceptos de completitud y consistencia de las lógicas. Si en una

lógica toda tautología es una tesis se dice que es *completa*, y se dice que es *consistente* cuando toda tesis es una tautología.

Con estos conceptos fundamentales se construyen las diferentes lógicas según se definan sus variables y conectivas, sus reglas operativas y su semántica.

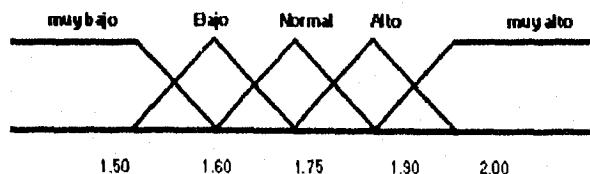
La *Lógica Difusa* surge como un intento de formalización del razonamiento con incertidumbre. En particular, y al contrario que otras formas de razonamiento, intenta abordar problemas definidos en términos lingüísticos, y por tanto imprecisos, donde los datos están expresados en términos cualitativos.

El primer punto que debe afrontarse al enfocar el tema, es el de la representación matemática de las expresiones lingüísticas. Podemos formalizar que una persona varón es *alto* cuando mide 1.90, y que hace frío cuando la temperatura es de 10°C. Evidentemente, una estatura de 1.75 ya no correspondería a un varón alto, sino a uno de estatura *normal*. "*1.90 es alto*" es cierto al 100%, "*1.75 es alto*" es cierto al 0%, pero "*1.75 es normal*" es cierto al 100%. Pues bien, la aplicación de la lógica difusa consiste en asignar a estaturas comprendidas entre 1.75 y 1.90, valores borrosos entre 0 y 1. Si la interpolación se hace de forma lineal, entonces "*1.80 es alto*" es algo cierto (0.33 difuso).

La lógica difusa tendrá por una parte el mismo aspecto sintáctico de la lógica de predicados y por otra su semántica se basa en el concepto de difusidad que se formaliza en la teoría de conjuntos difusos.

Para abordar y entender la teoría de los conjuntos difusos es casi preferible olvidarse, por lo menos durante un tiempo, de nuestros conocimientos de la teoría clásica de conjuntos, ya que la idea de partida es completamente distinta. En la teoría clásica un elemento cualquiera o bien pertenece a un conjunto o bien no pertenece al mismo. En teoría difusa un elemento siempre pertenece en cierto grado a un conjunto y nunca pertenece del todo al mismo.

El grado de pertenencia no tiene un sentido probabilístico, como en otras lógicas, sino más bien representa un grado de compatibilidad de un cierto predicado o un grado de posibilidad de que éste sea cierto. En la figura se representan las funciones de pertenencia de distintos conjuntos del ejemplo de las estaturas.



CAPITULO I

INTELIGENCIA ARTIFICIAL Y

SISTEMAS EXPERTOS

1. INTELIGENCIA ARTIFICIAL

La solución de los problemas nacionales como alimentación, educación, salud y vivienda, está muy relacionada con la capacidad del país de aprovechar más eficientemente sus recursos naturales y de insertarse en una economía global, ya no como simple productor de materias primas, sino como una nación industrial, con un nivel competitivo en relación con el resto de las naciones, que pueda colocar sus productos ventajosamente en el mercado internacional. Clave para ello es la mejoría tecnológica de sus procesos productivos, que es a su vez función de la calidad de su ingeniería. Este avance tecnológico sería la base, no sólo de una mayor calidad y menores costos de producción, sino de una mayor capacidad de adaptación frente al mercado y mejores posibilidades para ocupar en él un lugar preferente.

Esta situación hace imperativos el desarrollo y la adopción de nuevas filosofías y enfoques relacionados con la planeación, el análisis, el diseño, la gestión y el control en todas las ramas de la ingeniería, así como la utilización de técnicas innovadoras que permitan lograr una mayor competitividad.

Entre los desarrollos tecnológicos de mayor impacto en la ingeniería en general destaca la computadora, que ha transformado radicalmente la forma de concebir los procesos de análisis y diseño, tanto en ingeniería en computación como en las restantes ramas ingenieriles. Es la computadora la que ha permitido vislumbrar la "fábrica del futuro", como un sistema productivo altamente integrado, tanto en lo que se refiere a la planeación, la gestión y el control de la producción (manufactura integrada por computadora), como por lo que toca al proceso productivo mismo, sobre todo, a partir de los sistemas flexibles de manufactura.

Dentro de las ciencias computacionales, cuyas fronteras son actualmente muy dinámicas, la **Inteligencia Artificial**, representa un área con un gran potencial de aplicación para la resolución de problemas.

Las bases para el término Inteligencia Artificial (en adelante I.A.) se establecieron y fue usado por primera vez en el verano de 1956 en una conferencia en el Colegio Dartmouth en los Estados Unidos. Expertos en el tema acuerdan que I.A. está asociada con dos ideas básicas. La primera involucra el estudio de los procesos de pensamiento humano (para entender lo que es la inteligencia); y la segunda trata con la representación de estos procesos a través de las máquinas (computadoras, robots, etc.).

La Inteligencia Artificial es el conjunto de técnicas que se aplican en el diseño de programas para computadora que tengan capacidad de razonar, en el sentido de inferir nueva información, y que por la dificultad del problema a resolver requieran una solución con un grado de inteligencia.

Esta cualidad de inteligencia suele implicar una capacidad de aprendizaje, de autocorrección y de razonamiento que son elementos fundamentales y distintivos de la I.A. La cantidad y calidad de estas capacidades, difíciles de medir, provocan largas discusiones para determinar si un producto puede o no incluirse dentro de la I.A.

1.1 TÉCNICAS DE BASE

El contenido de la I.A. es abundante porque la I.A. cubre muchas áreas de aplicación, pero como contenidos básicos y comunes a la mayoría de las áreas hay que considerar las técnicas de base:

- Búsqueda heurística
- Lógica y razonamiento
- Lenguajes y herramientas
- Representaciones del conocimiento

El término heurístico procede del griego "heurisken" y tiene el significado de descubrir, en este sentido la *búsqueda heurística* es una técnica que hace posible el proceso de selección de una solución de un determinado problema en el que el número de alternativas es muy alto.

Una técnica de búsqueda heurística sería aquella que permite encontrar una solución "aceptable" sin tener necesidad de revisar una por una todas las posibilidades. Normalmente la búsqueda heurística se basa en un compromiso entre tiempo de cálculo y bondad de la solución. No existe una técnica para cada problema, ni existen técnicas universales.

Desde el punto de vista de la I.A. *la lógica y el razonamiento* agrupan las técnicas que permiten deducir nueva información. Son los métodos y algoritmos de trabajo. Se intenta reproducir con ellos los mecanismos habituales en el razonamiento humano: Modus Ponens, Modus Tollens, Silogismos, etc.

La potencia, versatilidad y facilidad de uso dependen de la herramienta de programación utilizada.

El propósito de *la representación del conocimiento* es organizar la información requerida por los métodos y algoritmos en una forma tal que les permita tomar decisiones, planificar, reconocer objetos y situaciones, en definitiva razonar.

Una división clásica de los esquemas de representación es considerar, por un lado los declarativos y, por otro los procedurales. Los primeros se refieren a la representación de hechos y afirmaciones, los procedurales a las acciones y métodos.

De acuerdo con esta clasificación, un programa de I.A. contendrá de los dos tipos ya que necesitará hechos y métodos para trabajar.

1.2 ÁREAS DE APLICACIÓN

La I.A. tiene aplicación en casi todas las áreas tecnológicas actuales. Históricamente se han venido considerando cuatro áreas principales de aplicación:

- El procesamiento del lenguaje natural
- Visión por computadora
- Sistemas basados en el conocimiento
- Resolución de problemas y planificación

Resulta difícil establecer limitaciones y clasificaciones cuando, como ocurre normalmente, un problema implica diferentes áreas, por ejemplo en la fabricación integrada por computadora (CIM) pueden encontrarse las cuatro áreas descritas interrelacionadas. Esta situación, que está comenzando a ser tenida en cuenta por los centros de investigación y desarrollo, nos lleva a ampliar áreas de aplicación con una nueva que podría denominarse Aplicaciones Industriales.

El término "Sistemas Basados en el Conocimiento" (knowledge based system) está siendo utilizado para referirse a aquellos programas de computadora que realizan funciones que suponen alguna de las propiedades asociadas con la inteligencia humana y que además incluyen dos características fundamentales:

- ⇒ La absoluta separación entre los mecanismos (algoritmos, métodos) de operación y el conocimiento (información).
- ⇒ La representación del conocimiento y las propiedades o relaciones entre sus componentes son simbólicos y tienen una gran componente semántica.

Este tipo de sistemas está siendo utilizado en todas las áreas y la reciente aparición en el mercado de estaciones de trabajo especialmente diseñadas, junto a las herramientas disponibles, hacen prever una avalancha de productos de todo tipo utilizando estos sistemas.

1.3 SISTEMAS BASADOS EN EL CONOCIMIENTO

Dentro de la Inteligencia Artificial existen dos grupos de aplicaciones fundamentales. El primero es aquél que se ocupa de interactuar con el mundo exterior de un modo efectivo y eficiente y se le denomina "Rama de Manipulación Física". El segundo grupo se ocupa de tratar temas más relacionados con el hombre, y en particular, su forma de razonamiento. Se le denomina "Rama de Manipulación del Pensamiento".

En la parte de la Inteligencia Artificial que se ocupa de la manipulación del pensamiento, hay dos facetas primordiales: la parte que estudia la posibilidad de comunicarse con las máquinas de un modo lo más próximo posible al humano, y la que intenta construir programas que funcionen de modo idéntico al razonamiento de las personas para una serie de campos sobre los que actúen. A los primeros se les llama Sistemas de Procesamiento del Lenguaje Natural y a los segundos *Sistemas Basados en el Conocimiento*.

Dentro de los Sistemas de Procesamiento del Lenguaje Natural se tienen dos grupos de sistemas: de reconocimiento de la voz humana y de generación de dicha voz. Los primeros sistemas se ocupan de crear programas capaces de comprender las órdenes y preguntas de un usuario que se exprese de la misma manera a como lo haría si se relacionase con otras personas. Los segundos se ocupan de que, a partir de unos resultados internos conseguidos por la computadora, éste sea capaz de comunicárselos al usuario en la misma forma en que lo haría un humano que alcanzase dichos resultados.

Los Sistemas Basados en el Conocimiento se suelen dividir, tal como se muestra en la figura 1-1, en Sistemas Expertos, Bases de Datos Inteligentes, Entornos de Programación, Sistemas de Desarrollo, Sistemas de Simulación y Programas de Ayuda al Operador.

Introducción a los Sistemas Expertos

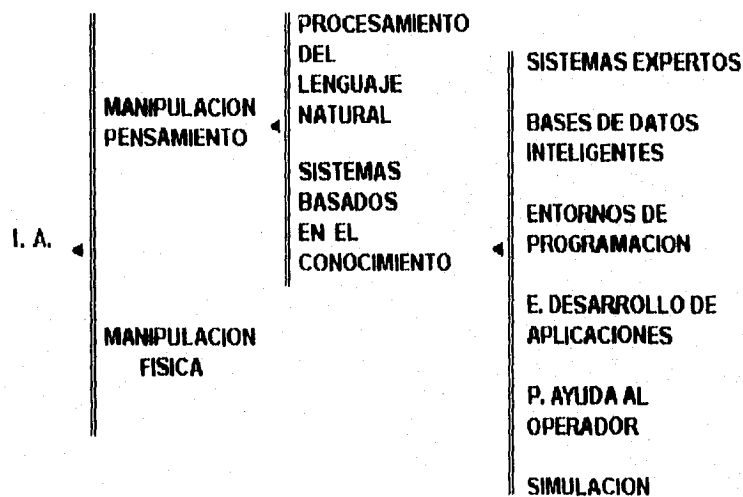


Figura 1-1: "Aplicaciones de la I.A. "

Entre todos ellos destaca por sus particulares propiedades y, por el gran desarrollo que están alcanzando, los "*Sistemas Expertos*". En ellos, el conocimiento requerido para resolver un problema viene dado por la experiencia obtenida por un técnico en la materia que se denomina *Experto*.

Este conocimiento suele venir dado por una serie de reglas heurísticas de comportamiento obtenidas a partir del examen de casos similares aparecidos a lo largo de sus años de trabajo en dicho campo.

Los demás grupos, de estructura similar, suelen tener su conocimiento basado en información que no depende tanto de la experiencia como del conocimiento teórico sobre la materia. Son casos de bases de datos inteligentes en que se unen las posibilidades de las estructuras de la Inteligencia Artificial para hacerlas funcionar de un modo más eficiente. o de las simulaciones, en que se aprovecha la gran modularidad y flexibilidad de las herramientas de la I.A. para desarrollar procedimientos heurísticos de control sobre el sistema simulado a partir de pruebas constantes realizadas con el simulador.

La gran importancia adquirida por los Sistemas Expertos ha producido que muchas veces los conceptos de Sistema Basado en el Conocimiento y Sistema Experto se confundan.

El hecho de que presentan una estructura similar y la misma finalidad de ayudar al máximo al usuario en aplicaciones que requieran un gran conocimiento específico, ha contribuido a esta confusión. Sin embargo, la aparición de la figura primordial del Experto como fuente de todo el conocimiento en los Sistemas Expertos, hace que éstos adquieran características que les diferencian de los demás.

2. SISTEMAS EXPERTOS

2.1 INTRODUCCIÓN

El año de 1956 es considerado como el momento en que nace la Inteligencia Artificial. Desde entonces, esta rama de la ciencia ha evolucionado vertiginosamente y, desde entonces, se busca, con esta disciplina resolver problemas complejos mediante la computarización de los procesos utilizados por la mente humana para captar y clasificar información, identificar estructuras y patrones en lo percibido, captar significados, plantear problemas, buscar soluciones, plantear su implantación y hacer el control, seguimiento y adaptación de dichas soluciones a través del tiempo.

Dentro de las ciencias computacionales, cuyas fronteras son actualmente muy dinámicas, la Inteligencia Artificial, y en particular los *Sistemas Expertos*, representan áreas con un gran potencial de aplicación para la resolución de

problemas en los que la experiencia humana desempeña un papel importante y cuya solución algorítmica o no existe, o es inadecuada.

Por otro lado, la toma de decisiones en ingeniería se suele realizar a través de una combinación de:

- A) Teorías científicas y técnicas analíticas,
- B) Métodos experimentales,
- C) Experiencia, juicio y sentido común.

Esto obedece a varios factores entre los que cabe mencionar:

- a) La escasez de información específica sobre los sistemas físicos,
- b) La incertidumbre de la información disponible sobre los mismos,
- c) La gran diversidad de fenómenos y variables que intervienen en el comportamiento de esos mismos sistemas,
- d) La presentación de situaciones y condiciones imprevistas en el campo,
- e) La complejidad de los problemas abordados en la ingeniería,
- f) La existencia de teorías fundamentales en criterios y condiciones de aplicación muy específicos.

Los problemas cuya solución depende de las reglas empíricas asimiladas por un experto a lo largo de su experiencia son bastante comunes en la ingeniería, tanto en lo que se refiere a los procesos de análisis, diseño y fabricación o de construcción, como en lo relacionado a la planeación, gestión y toma de decisiones.

Ahora bien, dada la insuficiencia de expertos de alta calificación en ingeniería y su inadecuada distribución en el país, así como el alto costo de la transferencia de conocimientos y de capacitación, una manera de perfeccionar la disponibilidad de conocimiento en el campo para mejorar esta toma de decisiones, sería la de incorporar el conocimiento de los expertos en una base de conocimientos computarizada, y a partir de la cual se pudiera recuperar este conocimiento en forma selectiva, de acuerdo con las necesidades de las empresas consultantes.

Esta forma de capturar el conocimiento de uno o varios expertos y de hacerlo disponible cuando se necesita, a través de una computadora, es lo que se conoce con el nombre de **Sistema Experto**.

Desde mediados de la década de los 60's, la Inteligencia Artificial ha tenido un éxito considerable en el desarrollo de Sistemas Expertos. Los Sistemas Expertos se emplean para ejecutar una variedad muy complicada de tareas que en el pasado solamente podían llevarse a cabo por un número limitado de personas expertas intensamente entrenadas.

A través de la aplicación de las técnicas de Inteligencia Artificial, los Sistemas Expertos captan el conocimiento básico que permite a una persona desempeñarse como un experto frente a problemas complicados.

Tal vez la característica más fascinante y poderosa de los Sistemas Expertos, que los distingue de la mayoría de las aplicaciones tradicionales de la computación es, su capacidad para enfrentar problemas que constituyen un reto del mundo real, por medio de la aplicación de procesos que reflejan el discernimiento y la intuición humanas.

Dentro de los Sistemas Basados en el Conocimiento, los Sistemas Expertos han supuesto que la I.A. vuelva a ser una tecnología de punta.

2.2 ANÁLISIS DEL CONOCIMIENTO

Existen varios componentes del conocimiento, que dan origen a la habilidad experta en su desempeño. Se pueden ver generalmente como:

- **Hechos.** Declaraciones que relacionan algunos elementos de la realidad con referencia al área específica.
- **Reglas de procedimiento.** Reglas bien definidas e invariables que describen secuencias fundamentales de eventos y relaciones relativas al área.
- **Reglas heurísticas.** Reglas generales en forma de opiniones o reglas empíricas que sugieren procedimientos que se pueden seguir cuando no existen disponibles reglas de procedimiento invariables. Dichas reglas son aproximadas y han sido generalmente acuñadas por un experto a través de años de experiencia.

2.3 DEFINICIÓN DE UN SISTEMA EXPERTO

Los Sistemas Expertos son programas de computadora en los que, a diferencia de los programas tradicionales, no se ha vertido una solución dada a un problema, sino el conjunto de conocimientos y reglas de operación de un experto humano, en torno a un problema específico, que le permiten al programa, a semejanza del experto humano, buscar la mejor solución, entre un gran número de posibilidades atendiendo a la naturaleza del problema y a la situación contextual en el que éste se da.

Además, los Sistemas Expertos tienen incorporadas facilidades de explicación de cómo llegaron a determinada solución, y en ciertos casos, mecanismos para aprender de la experiencia.

Un Sistema Experto es entonces un paquete de computación, construido en forma modular, con una base de conocimientos, que es donde se plasma la información

de los expertos, un módulo de inferencias que deriva conclusiones útiles a partir de la base de conocimientos y un sistema de interfaces con el usuario, para que éste pueda manejar e interrogar al sistema en forma fácil, útil y amena.

“Los Sistemas Expertos son sistemas computarizados que proporcionan soluciones a problemas en temas específicos, debido a que incorporan, en una base de conocimientos, conocimiento extraído de expertos humanos, del cual obtienen conclusiones válidas mediante mecanismos de inferencia.” (*Lara 1991*).

Una de las definiciones más completas de Sistema Experto es la debida a **Feigenbaum**:

“Un sistema experto es un programa inteligente de computadora, que usa procedimientos de conocimiento e inferencia para solucionar problemas que son suficientemente difíciles como para requerir experiencia humana en su solución. El conocimiento necesario para operar en tal nivel, añadido a los procedimientos de inferencia utilizados, puede ser considerado como un modelo de la experiencia de los mejores operadores en dicho campo”.

Revisando esta definición se puede observar que las características más importantes de un sistema experto son:

- Es un programa para computadora
- Dispone de una gran cantidad de conocimiento sobre el problema, fruto de la experiencia.
- Realiza un razonamiento similar al que haría un experto humano ante el problema planteado.
- Puede operar con datos cualitativos además de cuantitativos.
- Puede sacar conclusiones a partir de datos incompletos o inciertos.
- Tiene una total independencia entre la base de conocimientos y las reglas con las que opera sobre los conocimientos.

Por el contrario el humano es más creativo, adaptable, disponible de “sentido común” y experiencia sensorial. Frente a programas convencionales, los Sistemas Expertos permiten representar y usar conocimiento, no datos; son heurísticos, no algorítmicos; y disponen de un proceso de inferencia.

Todo esto ha hecho posible el desarrollo espectacular que se ha producido en la última década de estos sistemas.

2.4 SISTEMAS DE PRODUCCIÓN

Antes de pasar a estudiar la estructura interna de los sistemas expertos es necesario hablar de los *sistemas de producción*. De forma muy simple, un sistema de producción es un conjunto de reglas. Cada regla es una línea de

razonamiento que representamos mediante una parte izquierda y una parte derecha.

La parte izquierda representa antecedentes, situaciones o premisas y la parte derecha consecuentes, acciones o conclusiones. Ejemplos de *reglas de producción* son:

Si Juan es padre de Antonio y

Antonio es padre de Luis **ENTONCES** Juan es abuelo de Luis.

Si la temperatura es superior a 30 grados **ENTONCES** encender el ventilador.

Si el solicitante figura en el RAI **ENTONCES** no conceder el crédito.

Las reglas de producción se denominan también reglas *IF-THEN* (SI-ENTONCES).

Las principales características de los sistemas de producción son:

- Es fácil de construir.
- Es fácil de comprender por su parecido a nuestros esquemas mentales.
- Es muy flexible.
- Está muy próximo a los usuarios finales.

Los sistemas de producción proceden de la lógica y es posible encontrar definiciones formales y estudios de las propiedades de los sistemas de producción en libros de la especialidad e incluso en libros de I.A.

No obstante, un sistema de producción es una abstracción. es una de las formas en las que nosotros organizamos nuestras ideas o "reglas" pero no es algo que se pueda poner en una computadora de forma inmediata.

La mayoría de los Sistemas Expertos están *basados en reglas* y por tanto son aplicaciones prácticas de los sistemas de producción. La forma concreta en que se codifica la regla, así como la información que aparece en ella dependerá del lenguaje o herramienta de desarrollo que estemos utilizando.

2.5 ESTRUCTURA INTERNA DE UN SISTEMA EXPERTO

Tradicionalmente, un Sistema Experto está formado internamente por:

- Base de conocimiento
- Motor de inferencia
- Base de datos
- Interfaces de los sistemas expertos
- Control de ejecución

Anteriormente se explicó la necesidad existente en todo Sistema Experto de diferenciar claramente entre la zona del programa que se ocupa de contener el conocimiento general sobre el tema en cuestión y aquella que se ocupa de contener los métodos de razonamiento para operar, de un modo efectivo e independiente, con dicho conocimiento. A la primera de dichas zonas se le denominará Base de Conocimiento (BC) y se ocupará de contener y organizar dicho conocimiento. A la segunda se le denominará Motor de Inferencia (MI) y su misión será de realizar las operaciones necesarias con dicho conocimiento para lograr el fin requerido por el usuario.

Dentro de la denominada Base de Conocimiento se tienen dos tipos generales de unidades: los hechos y las reglas de producción. Los primeros representan a los conocimientos existentes sobre las variables generales del sistema. Serán verdades indiscutibles, relaciones aceptadas entre dichas variables, distintos valores posibles para dichas variables, etc. Las reglas de producción representarán las unidades de razonamiento de un experto ante dichas variables, las consecuencias que pueden tener para algunas de ellas los cambios efectuados en otras, la relación causal entre los distintos hechos, las condiciones para que se de un suceso determinado, etc.

El Motor de Inferencia se encargará de utilizar de un modo ordenado dichas reglas de producción para, a partir de una situación determinada, poder responder a las preguntas de un usuario, o aconsejar la acción más oportuna ante un cambio del entorno. Su función primordial será la de aplicar las reglas que intervengan ante una situación determinada y establecer los cambios producidos por ellas en la base de conocimiento.

Sin embargo, en la mayoría de los Sistemas Expertos es necesario tratar con una gran cantidad de datos de carácter general que en algunas ocasiones incluso proceden de procesos externos. Estos datos, que no dependen directamente del problema en cuestión como los hechos de la Base de Conocimiento se suelen organizar en una Base de Datos independiente de la anterior, que puede ser incluso físicamente exterior al sistema construido. En casos especiales, estos datos podrán venir dados por sensores conectados al Sistema Experto.

Pero en los Sistemas Expertos cada vez es más necesaria la existencia de unas cómodas interfaces que permitan comunicarlo con el exterior. La principal de todas ellas será situada entre el Motor de Inferencia y el Usuario. Su existencia permitirá una mayor comodidad y facilidad de comprensión de las operaciones realizadas por parte del usuario, características sin las cuales es incomprendible la existencia de un sistema de este tipo.

Otras interfaces también necesarias serán la existente con el Ingeniero de Conocimiento para facilitar la construcción del sistema y las presentes en la Base

de Datos, una para recibir señales exteriores en caso de necesidad y la otra, a veces necesaria, para conectarla con el Motor de Inferencia.

Finalmente en muchos Sistemas Expertos y fundamentalmente en casi todos los Sistemas Basados en el Conocimiento existe un último bloque al que se denominará Control de Ejecución.

Este bloque, realizado en el lenguaje de programación del sistema, tendrá condiciones comunes en todos los anteriores por lo que muchas veces se le considera integrado en ellos. En general se ocupará de modificar el control de la inferencia del sistema, simplificar la edición de las reglas y facilitar la interfaz con el usuario.

La existencia de este bloque es particularmente importante para casos de sistemas desarrollados con herramientas de construcción de Sistemas Expertos. En estos casos, el Motor de Inferencia suele venir dado por la herramienta y la única manera de cambiar el mecanismo de razonamiento es mediante el código externo. Sin embargo, como ya se ha enunciado, la existencia de este bloque es optativa y muchos Sistemas Expertos no lo tienen implementado.

En la figura 1-2 se presenta el diagrama de bloques de un sistema experto incluyendo todos los bloques descritos.

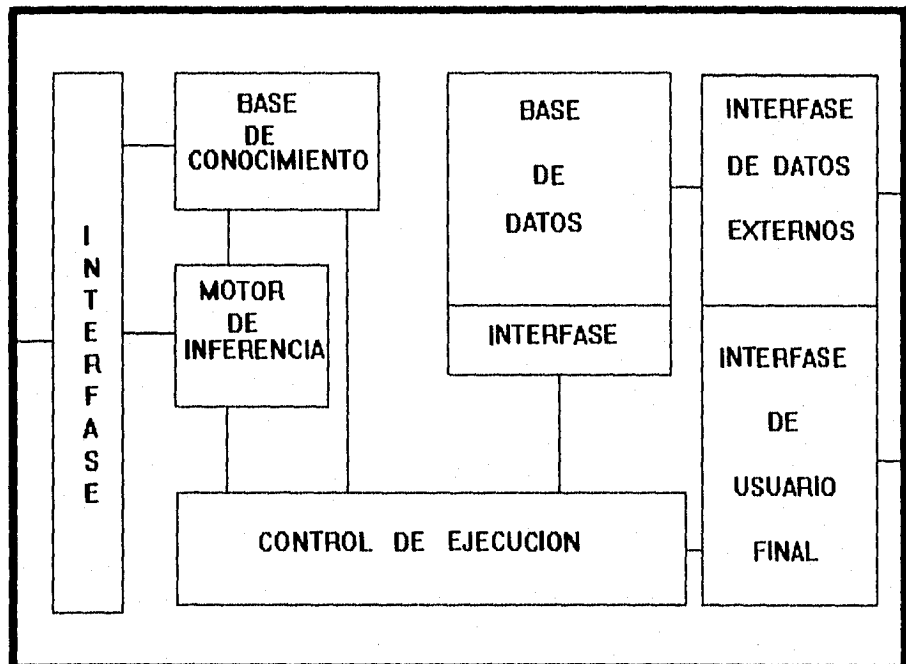


FIGURA 1.2 Estructura Interna de un Sistema Experto

2.5.1 BASE DE CONOCIMIENTO

La **Base de Conocimiento** constituye uno de los bloques fundamentales de un Sistema Experto. Está constituida por todo el conocimiento teórico y experimental sobre el problema en cuestión que haya podido recoger el Ingeniero de conocimiento a partir del experto en la materia. Parte primordial de la aplicación, todos los demás bloques se verán grandemente influenciados por las decisiones tomadas al realizarla. De ahí la importancia dada a su desarrollo y el tiempo elevado dedicado a su construcción de todo el Sistema Experto.

Aún cuando se presenta como bloque unitario, este módulo contiene dos diferentes tipos de conocimiento sobre el problema a tratar, los hechos y las reglas de producción.

Los hechos son verdades, situaciones presentes o relaciones admitidas entre las distintas variables del sistema. Así, por ejemplo, un hecho a tratar de la unidad persona, será la afirmación de que, si no se decide lo contrario, toda persona tiene dos manos, o que tiene dos piernas y una cabeza. La labor del Ingeniero de Conocimiento al tratar los hechos no se limitará a plantear dichas verdades admitidas sino que, al mismo tiempo, para favorecer el funcionamiento del sistema, deberá:

- ⇒ Organizar dichos hechos en una forma eficaz no sólo para ser operada sino también para ser comprendida.
- ⇒ Localizar las variables más significativas de la aplicación y estructurarlas facilitando el acceso a las mismas.
- ⇒ Relacionar dichas variables conforme al conocimiento existente de la materia.
- ⇒ Conseguir que toda la estructura tenga una gran capacidad de modificación y crecimiento.

Las reglas de producción representarán los conocimientos causales existentes entre los distintos tipos de hechos. Serán las relaciones admitidas entre los valores de las distintas variables, las consecuencias que tendrán las modificaciones de ciertos elementos respecto a otros, etc. Por ejemplo, una regla podría ser tratando el caso del hombre el establecer que Si una persona tiene una única mano **ENTONCES** es manco. Al tratar la construcción de las reglas, el Ingeniero de Conocimiento deberá atender a las siguientes condiciones:

- ⇒ Las reglas de producción deben definir todas las posibilidades de relación del sistema.
- ⇒ Deben construir cada una de ellas un conocimiento independiente sobre las consecuencias producidas por una situación determinada.
- ⇒ Deben ser estructuradas de un modo eficiente para el buen funcionamiento del sistema.

- ⇒ En general, deberán admitir la presencia de variables para simplificar su construcción y aumentar sus prestaciones de funcionamiento.
- ⇒ Debe existir siempre la capacidad de añadir de un modo sencillo nuevas reglas a la base.

Tanto la estructura de los hechos como la de las reglas deberán admitir también la propiedad de estar representadas del modo más sencillo posible y ser inteligibles por cualquier usuario que no sea un experto en la materia.

Como base de conocimiento en general, interesará siempre el que el conocimiento sobre un tema en cuestión sea muy amplio, aunque sin redundancia, y lo más rápidamente accesible sea posible. Así se disminuirán los tiempos de búsqueda necesarios, reduciéndose con ellos el tiempo de respuesta y la posibilidad de error del sistema.

2.5.2 MOTOR DE INFERENCIA

El ***Motor de Inferencia*** constituye con la Base de Conocimiento los dos bloques fundamentales de un Sistema Experto. Su misión fundamental es la de utilizar todos los conocimientos incluidos en el sistema en forma de hechos y reglas de producción para alcanzar el resultado apetecido.

En general, se procurará que utilice un mecanismo de razonamiento lo más parecido posible al humano, reproduciéndose el modo de deducción del experto al máximo. Deberá también ser comprensible en sus pasos por el usuario para facilitar su seguimiento posterior por el mismo. Tendrá dos acciones fundamentales:

- ⇒ Examen de hechos existentes y reglas añadiendo hechos a la Base de Conocimiento en caso de así indicarlo las reglas aplicadas.
- ⇒ Decidir el orden en que se ejecutan las reglas cuando hay varias posibles.

Cada vez que el Motor de Inferencia examina una regla para ver si es aplicable, comprobará si sus condiciones iniciales o antecedentes se cumplen. En caso de ser así, afirmará las condiciones indicadas en su segunda parte o consecuentes. Cuando hay varias reglas ejecutables en el mismo instante, deberá elegir la más prioritaria. Este proceso tiene los siguientes pasos:

- 1) Se seleccionan todas las reglas que dada la situación existente pueden ejecutarse en el instante dado.
- 2) Se toman todas esas reglas y se introducen en el denominado "conjunto de conflicto"
- 3) Se establecen métodos de resolución para ver cuál es la regla mas prioritaria entre todas ellas. El criterio de selección aquí aplicado

dependerá de la aplicación y del tipo de búsqueda elegido para ella. Criterios frecuentemente elegidos serán el escoger aquella cuyos antecedentes hayan sido activados los últimos, la regla que tenga asignado un coeficiente de prioridad más alto, la más específica, etc.

- 4) Se aplica dicha regla elegida como más prioritaria y se modifica la Base de Conocimiento conforme a los consecuentes de la regla aplicada.
- 5) Se regresa al punto 1) con la nueva situación del sistema ya establecida.
- 6) El proceso termina cuando ya no hay reglas aplicables o cuando se llega a la solución que partía como objetivo de la búsqueda.

Este proceso de operación debe realizarlo el Motor de Inferencia cada vez que de alguna forma se active el mecanismo de búsqueda.

2.5.3 BASE DE DATOS

La **Base de Datos** es aquel módulo de los Sistemas Expertos en la que se almacena toda la información general que no depende explícitamente del problema tratado. Aparece especialmente en casos en los que se requieran una gran cantidad de valores dato para las variables y constantes del sistema. Estos valores, que generalmente no se ven influenciados por las demás variables existentes y que por tanto permanecen constantes durante la aplicación, se estructuran del mejor modo posible para facilitar su acceso. A la estructura resultante de datos utilizable por el sistema se le denomina Base de Datos del Sistema Experto.

Como se puede observar por lo descrito, la relación entre la Base de Datos y la Base de Conocimientos es muy acusada. Este hecho ha provocado que en muchas ocasiones se incluyan las dos en el mismo bloque de conocimiento. Sin embargo, aparecen características particulares de ambas bases que permiten diferenciarlas:

BASE DE CONOCIMIENTO

Conocimiento particular

Introducido por Experto en materia

Estructura dada por Ing. del Conocimiento

Variación de valores por acción de S.E.

Construcción adecuada para el S.E. particular

BASE DE DATOS

Datos generales

Datos que no requieren especial introducción

Estructura general de base de datos

Suele limitarse a valores de lectura, sin variación

Muchas veces, utilización de base ya construida

Estas características, sin embargo, no son tan claras en todo sistema Experto. A veces, la Base de Conocimiento presenta caracteres de Base de Datos y viceversa.

En todo caso, al realizar un sistema, conviene siempre diferenciar claramente ambas bases, aunque sólo sea conceptualmente. Se evitarán así complicaciones en el trabajo posterior con el sistema y se facilitará el crecimiento modular del mismo.

Hay casos en que varios Sistemas Expertos se conectan todos ellos a una Base de Datos exterior común de la que obtienen los valores necesarios para sus operaciones. En estos casos es frecuente que exista un bloque de interfaz entre ellos y la base. Este bloque se ocupará de atender a las peticiones de los sistemas y de transmitirles los datos requeridos cuando estén disponibles.

Otras veces, la recepción de datos externos se realizará a partir de sensores. En estos casos dicha información ha de ser operada antes de llegar al sistema. Deberá estar sincronizada con su funcionamiento y convertida a valores comprensibles internamente. Es frecuente la existencia de código de conversión entre los valores externos y la representación interna. Este código suele estar incluido entre los valores externos y la representación interna. Este código suele estar incluido en el bloque de Control de Ejecución.

El bloque de Base de datos ha ido apareciendo cada vez con más importancia en los Sistemas Expertos. A medida que éstos han ido creciendo en tamaño y prestaciones, la necesidad de tener la información sobre el mundo exterior accesible y ordenada ha aumentado a la misma velocidad. Actualmente, casi todas las herramientas de ayuda al diseño de Sistemas Expertos admiten la posibilidad de conectar el sistema a programas y bases de datos exteriores. Es de esperar que, a medida que los Sistemas Expertos sigan aumentando en tamaño y en necesidades de información exterior, mayores y más específicas bases de datos sean construidas para trabajar en ellos.

2.5.4 INTERFACES EN LOS SISTEMAS EXPERTOS

Los bloques de interfaz entre el exterior e interior de un Sistema Experto se ocupan de traducir las representaciones exteriores del conocimiento a una forma interna que pueda ser comprendida por el código del sistema. Se puede hablar de cuatro interfaces generales, la del usuario, la del Ingeniero de conocimiento y dos de la Base de Datos, una entre ella y el sistema y la otra de conexión con el mundo exterior. De todas ellas, la más importante es la del usuario.

2.5.4.1 INTERFAZ CON EL USUARIO

Existen dos facetas fundamentales en la Interfaz del Usuario:

- *Respuestas.* Es la parte de la interfaz que se ocupa de responder a las preguntas del usuario. Para ello debe trasladar dichas preguntas al Motor de Inferencia y comunicar posteriormente sus respuestas del modo más comprensible posible. además, deberá tener una capacidad gráfica que le permita presentar dichas respuestas del modo más cómodo posible.

- *Explicación.* Es la parte de la interfaz que se ocupa de presentar el resultado obtenido y, sobre todo, encargarse en todo instante de poder indicar el razonamiento seguido. Este podrá venir dado ante una pregunta del usuario sobre cómo se ha llegado a un resultado dado o porqué el sistema le pregunta por el valor de un cierto dato. Esta parte dependerá en gran medida del Motor de Inferencia implementado.

Desde el principio de las técnicas de la Inteligencia Artificial en los años cincuenta, una de las prioridades de la misma fue la construcción de una interfaz cómoda entre la computadora y el usuario. Esta interfaz se pensó que debía estar constituida por un lenguaje de carácter humano al que se denominó Lenguaje Natural. Los fines primordiales de este lenguaje eran:

- ⇒ Que el usuario sea comprendido y respondido por el ordenador en dicho lenguaje.
- ⇒ Que el vocabulario sea extenso y cercano al utilizable normalmente en la vida real.
- ⇒ Comunicación posible de la máquina con un no técnico en la materia.
- ⇒ Intercomunicación de viva voz.

Desgraciadamente, los objetivos se demostraron pronto como demasiado ambiciosos y la meta está todavía lejos de conseguirse. De todas formas, actualmente se está investigando mucho en el tema y se están alcanzando buenos resultados para lenguajes de vocabularios relativamente amplios, aunque con introducción de valores por teclado. En todo caso, todavía los Sistemas Expertos deberán esperar hasta que dichos lenguajes sean aplicables a sus estructuras. Por ello, en la actualidad, aquellos que están en funcionamiento y principalmente las herramientas de desarrollo de Sistemas Expertos han dedicado un gran esfuerzo a las construcción de gráficos y representaciones asociables a los elementos de un sistema. Como gráficos más destacados tenemos:

- ⇒ Representación de valores de variable en forma digital. A veces, incluso modificables actuando sobre él.
- ⇒ Representación gráfica de valores. Barras, menús de elección...
- ⇒ Histogramas.
- ⇒ Pantallas de texto, variables en tamaño, forma, posición...
- ⇒ Gráficos interactivos que permiten modificar valores de variables actuando sobre ellos.

⇒ Actuadores de lanzamiento de métodos de resolución, lanzamiento del sistema...

Como ayuda que se va imponiendo cada vez más para la modificación de los gráficos y para la respuesta del usuario entre menús, aparece la facilidad del *ratón*. Por medio de sus movimientos y actuación en sus teclas, se consigue cambiar las formas y valores de los gráficos. El *ratón* es una nueva herramienta de ayuda para el programador que poco a poco se va imponiendo como apoyo para la interfaz máquina-usuario.

Todo Sistema Experto, debe tener también la facilidad de explicar su razonamiento ante preguntas del usuario. Para ello utilizará la información existente en el Motor de Inferencia e irá indicando para cada regla aplicada las condiciones antecedentes de la misma y las conclusiones ejecutadas. Esta faceta de la interfaz está generalmente del tipo textual. Sin embargo, cada vez es más frecuente introducir gráficos que permitan seguir el razonamiento de un modo más cómodo. Se representan las reglas aplicadas en forma de árboles jerarquizados cuyos nodos representan a las reglas y las ramas el orden en que han sido aplicadas. Este método mucho más visual de trabajo permite seguir el razonamiento del sistema con mucho menor esfuerzo por parte del usuario.

2.5.4.2 INTERFAZ CON EL INGENIERO DE CONOCIMIENTO

Se ocupa de facilitar la labor del Ingeniero de Conocimiento en la construcción del Sistema Experto. Utiliza todas las facilidades de la interfaz del usuario. Además de ellas, suele presentar cómodos editores y mecanismos de seguimiento del razonamiento y de búsqueda de errores más complejos que los del usuario. Es muy importante, ya que una buena interfaz puede reducir en mucho el tiempo necesario para la construcción de un sistema.

2.5.4.3 INTERFACES CON BASE DE DATOS

Ya se ha hablado de ambas interfaces, de conexión exterior e interior. La primera se ocupa de recibir datos exteriores introducidos a partir de sensores. En general deberá haber un mecanismo de sincronización intermedio entre ellos y el sistema. La conexión interior es particularmente interesante en casos de Bases de Datos ya construidas, especialmente si varios sistemas pueden acceder a la misma.

La importancia de las cuatro interfaces, particularmente la del usuario, se está incrementando día a día. El buen resultado de un Sistema Experto muchas veces depende de la comodidad con la que un usuario puede trabajar con él, especialmente desde el punto de vista de la comercialización.

2.5.5 CONTROL DE EJECUCIÓN

Cuando se construyeron los primeros Sistemas Expertos, se consideró que todo el control del razonamiento del sistema debía estar incluido en el Motor de Inferencia. El usuario se limitaba a hacer preguntas al sistema y a procurar entender sus contestaciones, pudiendo pedirle aclaraciones sobre el razonamiento empleado. El sistema era activado directamente por el usuario y daba por concluida su labor al conseguir un determinado objetivo. Al haberse construido el programa especialmente enfocado para la aplicación dada, no se requería ningún control externo y todo el mecanismo del razonamiento dependía del Motor de Inferencia.

Sin embargo, a medida que nuevos Sistemas Expertos fueron realizándose, se descubrió que la realización del bloque de Motor de Inferencia era un proceso muy costoso. Se comenzaron a utilizar bloques ya construidos que facilitaban la labor de desarrollo. Este proceso de simplificación tiene su último logro con la construcción de las herramientas de ayuda al diseño de Sistemas Expertos.

Pero estos bloques de razonamiento ya construido tenían el problema de, por ser muy generales, muchas veces no adaptarse por completo al problema tratado. Al mismo tiempo apareció el concepto paralelo de Sistemas Basados en el Conocimiento. Estos sistemas desarrollados a partir de las mismas herramientas que los Sistemas Expertos y muchas veces confundidos con ellos, intentaban no dedicarse a incluir únicamente en un sistema de información proveniente de un experto humano, sino a partir de conocimientos generales ayudar a un usuario en diversas tareas que requiriesen un conocimiento elevado de una materia. Ejemplos de este tipo de sistemas serán los simuladores, programas de ayuda a usuarios para utilización de nuevas máquinas, programas de enseñanza etc. Al desarrollar estos programas, los Ingenieros del Conocimiento descubrieron que muchas facetas eran más fácilmente representables en forma de código que de valores unitarios y que, para que el programa tuviese control sobre el razonamiento empleado, debían existir formas de activarlo y modificar su camino por medio de instrucciones programables.

2.6 VENTAJAS DE LOS SISTEMAS EXPERTOS

Algunas de las ventajas de los Sistemas Expertos son:

1. **Autonomía:** Una vez que el Sistema Experto ha sido diseñado y completado, se hace autónomo, es decir, independiente de la presencia física tanto del experto como del desarrollador del sistema.
2. **Productividad:** Se aumenta la productividad del conocimiento experto al relevar a especialistas de tareas repetitivas, y así aprovechar sus capacidades intelectuales en asuntos más creativos.

3. **Manejo de gran cantidad de información:** Posibilidad de operar en tiempo real con gran cantidad de información durante el proceso de raciocinio, rebasando por mucho los niveles de manejo de información accesibles a los expertos humanos.
4. **Rapidez:** La velocidad de respuesta puede ser muchas veces mayor a la de los expertos humanos sin perder efectividad.
5. **Homogeneización de criterios:** Los Sistemas Expertos pueden desarrollarse basándose en los conocimientos de más de un experto.
6. **Imparcialidad:** Los Sistemas Expertos proporcionan juicios imparciales, aún operando con información ambigua, inexacta, incompleta o con algún grado de incertidumbre.
7. **Reproducibilidad:** El Sistema Experto en sí y el conocimiento que abarca son reproducibles a voluntad de su autor. Esto implica que, en caso de ser necesario, el producto puede reproducirse para dar servicio a miles de usuarios. Por otro lado, en contraposición con la lentitud del proceso de formación de expertos, la reproducción de un Sistema Experto es del orden de unos minutos solamente.
8. **Bajo costo de adquisición y operación:** En tanto de disponer permanentemente de un experto supone altos costos, un Sistema Experto.

2.7 TIPOS DE SISTEMAS EXPERTOS

Dependiendo del problema que resuelven, la mayoría de los Sistemas Expertos pueden clasificarse dentro de una de las siguientes categorías:

- a) **De interpretación:** Infieren el significado de un conjunto de datos partiendo de su contexto.
- b) **De predicción:** Infieren las probables consecuencias de una situación dada.
- c) **De diagnóstico:** Identifican la naturaleza y las causas del mal funcionamiento de un sistema a partir de ciertos signos y síntomas.
- d) **De diseño:** Desarrollan las configuraciones que resuelven un problema satisfaciendo sus restricciones.
- e) **De planeación:** Diseño de estrategias para lograr un objetivo, utilizando ciertos recursos y sujetándose a ciertas restricciones.
- f) **De monitoreo:** Analizan observaciones del comportamiento de un sistema para detectar posibles desviaciones de la norma y tomar eventualmente medidas correctivas en tiempo real.
- g) **De depuración:** Describen soluciones para los casos en que programas de computadora o equipos de toda índole muestran condiciones anómalas en su comportamiento.
- h) **De reparación:** Desarrollan y ejecutan planes para corregir fallas.

- i) *De instrucción o tutoriales:* Ayudan en el aprendizaje, no solo presentando la información necesaria al usuario sobre un tema específico, sino también haciendo un seguimiento del aprendizaje del mismo.
- j) *De control:* Comparan la situación actual de un sistema con una norma predefinida y corrigen las variables conforme sea necesario, requiere únicamente pagar por su diseño y construcción. Este costo después se distribuye entre todos los usuarios del sistema, por lo que el costo por copia es muy reducido.
- k) *Facilidad de distribución:* Por naturaleza del sistema, pueden cubrirse necesidades en localizaciones geográficas muy dispersas y bajo condiciones de trabajo difíciles.

2.8 ARQUITECTURA DE LOS SISTEMAS EXPERTOS

Aunque los Sistemas Expertos comenzaron funcionando como bloques independientes de intercomunicación con el usuario, poco a poco ha ido avanzando la necesidad de llegar a estructuras más complejas a fin de utilizar al máximo sus propiedades. Así, hoy en día, se comienza a pensar en sistemas específicos distribuidos ayudándose unos a otros en la resolución de problemas. Por eso han nacido varios modos fundamentales de estructurar los Sistemas Expertos con el objeto de mejorar su funcionamiento.

Se puede hablar de tres tipos de estructuras:

2.8.1 SISTEMAS EXPERTOS AISLADOS

Es el caso normal de aplicación de estos sistemas en la actualidad. El Sistema Experto controla su propia Base de Datos y recibe directamente todas las entradas externas. Su estructura es la más simple. Ha dado muy buenos resultados y su comprensión es básica para poder entender las demás estructuras.

2.8.2 SISTEMAS EXPERTOS INTEGRADOS

Son aquellos en que, debido a la complejidad del sistema a tratar, existen varios Sistemas Expertos conectados a una única Base de Datos. El acceso a dicha base suele ser a través de una interfaz especial. Existen dos tipos fundamentales de estructuras de este tipo:

- ⇒ *Sistemas Front-End.* El Sistema Experto pide un dato necesitado a la Base de Datos y este se lo devuelve directamente.
- ⇒ *Sistema Back-End.* El Sistema Experto pide un dato necesitado a la Base de Datos. Esta observa que no dispone de él, pregunta a otro Sistema Experto por dicho valor, espera a que se lo manden y lo devuelve al programa que realizó la llamada inicial.

2.8.3 SISTEMAS EXPERTOS EMBEBIDOS

Son aquellos en que uno o varios Sistemas Expertos están introducidos en una aplicación general que actúa sobre ellos. Son un bloque más de ella, siendo activados cuando el sistema de control global lo considere oportuno. Devuelven los valores perdidos y esperan a nueva petición.

Aunque la mayoría de los Sistemas Expertos pertenecen al grupo de sistemas aislados, poco a poco se va viendo la necesidad de construir sistemas más complejos y de superiores prestaciones. Es de esperar que en un futuro próximo aparecerán más aplicaciones y herramientas que permitan construir de un modo sencillo otras estructuras más complicadas para resolver problemas más amplios. La tecnología de los Sistemas Expertos, dado que son programas de computadora, puede considerarse independiente de un hardware específico. Lógicamente por el tamaño en disco o memoria, velocidad, lenguaje básico cada sistema experto tiene unos requisitos mínimos. Pero es posible desarrollar sistemas expertos en computadoras personales, microcomputadoras, grandes computadoras o máquinas de diseño específico.

3. DESARROLLO DE UN SISTEMA EXPERTO

3.1 FASES EN LA CONSTRUCCIÓN DE UN SISTEMA EXPERTO

Se suelen distinguir cinco fases, altamente interrelacionadas, pero que al menos con fines didácticos se deben considerar: Identificación, Conceptualización, Formalización, Implementación y Verificación.

Durante la fase de **identificación** el ingeniero de conocimiento y el experto determinan los aspectos importantes del problema. Es decir, la identificación del problema en cuanto a tipo, rango, etc.; los participantes en el desarrollo del problema (otros expertos); los recursos necesarios (tiempo, facilidades de cálculo, etc.) y los objetivos de la construcción del sistema experto. De estas actividades, sin duda la determinación del rango del problema es la más crítica, a menudo el primer intento de identificación pretende resolver un tema excesivamente complejo, ambiguo y general.

El Ingeniero de Conocimiento obtiene en esta fase una medida de la complejidad intentando resolver alguno de los sus bjetivos mas interesantes de forma práctica, creando e implementando rutinas, haciendo programas previos.

Durante la **conceptualización**, el ingeniero de conocimiento y el experto deciden que conceptos, relaciones y mecanismos de control necesitan para describir el problema. Las estrategias, subtareas y limitaciones en la resolución del

problema, se exploran también en esta fase. Se puede determinar en este momento a qué nivel de detalle tiene que ser representado el conocimiento.

El ingeniero de conocimiento tomará los niveles más abstractos de detalle para poder distinguir entre los conceptos clave. Los encargados del desarrollo deben comenzar en este momento un análisis completo antes de empezar los programas de implementación.

La **formalización** implica determinar los conceptos clave y las relaciones formalmente, generalmente usando una estructura tipo "frame" o similares. El ingeniero de conocimiento podrá ahora tener algunas ideas sobre las posibles herramientas adecuadas al problema.

Durante la **implementación** el ingeniero de conocimiento transforma el conocimiento definido en las fases de la formalización en un programa para computadora. La construcción de este programa requiere contenido, forma e integración. El contenido llega del dominio del conocimiento, es decir, estructura de datos, reglas de inferencia y estrategias de control necesarias para la resolución del problema. La forma viene definida por el lenguaje escogido. La integración supone la combinación y reorganización de las diferentes piezas del conocimiento para eliminar los errores globales entre estructuras de datos, reglas y especificaciones de control. La implementación debe abordarse relativamente pronto, porque la implementación del prototipo inicial permite verificar la efectividad del diseño de las decisiones hechas durante las fases anteriores. Esto significa que hay muchas probabilidades de que el código inicial sea revisado durante el desarrollo.

Finalmente la fase de **verificación** supone la evaluación de las capacidades y utilidad del programa prototipo y de su revisión si es necesario. El experto en el problema evalúa el prototipo y ayuda al Ingeniero de conocimiento a su revisión. Tan pronto como el prototipo funcione con unos pocos ejemplos, debe medirse su eficiencia y utilidad. Esta evaluación puede poner de manifiesto: olvidos en la fase de diseño en conceptos o relaciones, conocimiento representado de forma inadecuada o fallos en los mecanismos de control. Tales problemas pueden forzar a los encargados del desarrollo a volver a realizar total o parcialmente fases anteriores.

El sistema experto debe ser realizado pensando en la realización de verificaciones y pruebas posteriores, que permiten modificaciones no excesivamente costosas en tiempo. Los usuarios finales piden al sistema que sea rápido, fiable, fácil de usar, fácil de comprender y muy generoso cuando se cometan errores. Antes de estar listo para ser usado como un sistema comercial, el sistema experto debe superar gran cantidad de pruebas y verificaciones.

Los sistemas expertos suelen clasificarse en función de su estado de realización de la forma siguiente:

- Prototipo de demostración.(50-100 reglas). El sistema resuelve una porción del problema, y debe demostrar la viabilidad del proyecto.
- Prototipo de investigación.(200-500 reglas). El sistema tiene unas prestaciones creíbles, pero es frágil y aún no ha superado las pruebas de campo.
- Prototipo de campo. (500-1000 reglas). El sistema es ya operativo, pero aún debe mejorar sus cualidades de velocidad, fiabilidad y adaptación al entorno.
- Modelo de producción. (500-1500 reglas). El sistema está terminado, es correcto pero sirve para un usuario.
- Sistema comercial. Se trata de un modelo de producción que se ha generalizado para convertirlo en una herramienta de amplio espectro.

Como unidad de medida puede usarse el número de reglas que componen un sistema. En la actualidad la mayoría de los sistemas son prototipos de una u otra categoría, pero prototipos y existen muy pocos sistemas comerciales.

3.2 PERSONAS QUE INTERVIENEN EN LA CONSTRUCCIÓN

Los principales jugadores en el juego de los sistemas expertos son:

- El sistema experto
- El experto en el problema
- El ingeniero de conocimiento
- El usuario

El sistema experto es el conjunto de programas o software para computadora que resuelve problemas en un campo de interés. Se le denomina sistema porque es bastante más que un simple programa, contiene los programas y el conocimiento con el que operarán estos programas. Este soporte ayuda al usuario a interactuar con el programa básico principal y puede incluir sofisticados mecanismos de ayuda y depuración.

El experto en el problema es la persona que por su conocimiento y reputación está en condiciones de dar soluciones a los problemas en un área en particular. El conoce trucos y atajos para buscar una solución más eficiente y dará al sistema experto sus estrategias. El experto en el área no tiene porque ser una persona sola. Además pueden usarse libros, artículos, etc. todo aquello que dé experiencia.

El Ingeniero de Conocimiento es una persona, generalmente con mucha experiencia en informática e I.A. y sabe cómo diseñar un sistema experto. El realizará las entrevistas a los expertos, organizará el conocimiento y decidirá como se representará éste. Generalmente este ingeniero de conocimiento ayuda al experto a estructurar el área del problema, interpretando e integrando respuestas de los humanos en preguntas, estableciendo analogías, planteando contraejemplos y trayendo a la luz dificultades conceptuales. El ingeniero de conocimiento es, por lo general el desarrollador del sistema.

El usuario es la persona que usará el sistema experto cuando esté terminado. El usuario puede ser un científico usando el sistema para descubrir nuevos depósitos de minerales, un abogado usándolo para resolver un caso o un estudiante tratando de aprender química orgánica. El término usuario es un poco ambiguo, normalmente se refiere al usuario final, la persona para la que se desarrolla el sistema experto y también cualquier posible usuario, durante el desarrollo todos los participantes serán de una u otra forma usuarios.

3.3 ELECCIÓN DE LA HERRAMIENTA

La elección de la herramienta es junto con la definición del problema una de las mayores dificultades del diseño. Existen bastantes herramientas y tal como define Randall Davis en su conocida ley "Para cada herramienta existe una tarea para la que resulta especialmente adecuada", pero desgraciadamente la inversa no es cierta. De hecho para una tarea existirá más de una herramienta que lo resuelva igualmente bien. En la práctica la mayoría de las veces el criterio de selección de una herramienta obedece a:

- El ingeniero de conocimiento estaba familiarizado con ella.
- La herramienta venía impuesta por el hardware del diseñador.
- Se desarrolló una herramienta y para probarla se desarrollaron aplicaciones.

Estas razones no son muy recomendadas aunque deben tenerse en cuenta. El desarrollo de los sistemas expertos requiere tiempo, dinero, personal, hardware y todo ello debe tenerse en cuenta en la elección. Básicamente la elección se centra entre lenguajes de programación y entornos de ingeniería del conocimiento. Los primeros tienen como ventaja una gran flexibilidad, una adaptación total al problema concreto pero suponen más tiempo y la implementación del motor de inferencia. Por otro lado los entornos ofrecen el motor de inferencia y programas de ayuda complejos, pero el sistema puede no ser suficientemente eficiente y rápido. Un grupo sin mucha experiencia estaría más protegido con un entorno aunque tuviera menos flexibilidad.

Por último, debe tenerse en cuenta que el mercado ofrece cada día más herramientas que van mejorando sus prestaciones haciendo más difícil la elección.

3.4 ADQUISICIÓN DEL CONOCIMIENTO

Adquisición del conocimiento es la acumulación, transferencia y transformación de la expertés de alguna fuente de conocimiento a un programa de computadora para construir y expandir la base de conocimiento.

El conocimiento en un sistema experto puede provenir de diferentes fuentes tales como libros de texto, artículos, bases de datos, casos prácticos, datos empíricos y experiencia personal. Esta última fuente es la más completa y la más compleja en la mayoría de los sistemas. *El Ingeniero de Conocimiento* obtiene la información interrogando la experto, esta comunicación personal encuentra muchos problemas. Uno de ellos se conoce con el nombre de la paradoja del ingeniero de conocimiento y que viene a decir que el más competente de los expertos es el que menos sabe explicar su conocimiento y sus formas de razonamiento. El ingeniero de conocimiento debe *transformar* la representación del conocimiento como parte del proceso de transporte.

Para adquirir el conocimiento necesario, el ingeniero de conocimiento primero debe establecer una comprensión global del área, formar un diccionario mental de los términos y desarrollar una comprensión básica de los conceptos claves. Luego debe condensar el conocimiento sucinto a partir de la información suministrada por el experto.

La función de adquisición de conocimiento es comúnmente, el aspecto de mayor dificultad en la construcción de sistemas expertos. Esto se debe principalmente al hecho de que el proceso requiere comunicaciones humanas ampliadas, entre el experto en el área y el ingeniero de conocimiento y en consecuencia enfrenta los problemas asociados con esta actividad. Por lo tanto el proceso de adquisición del conocimiento no está bien entendido ni bien definido. Si el proceso mismo de desarrollo de un sistema experto se visualizara como un área para expertos, el conocimiento asociado con el procedimiento de adquisición sería considerado como heurístico.

Aunque en los libros clásicos de sistemas expertos se apuntan métodos de trabajo, para garantizar un adecuado avance en la adquisición del conocimiento, en general, resultan demasiado utópicos porque las disponibilidades de los expertos en las empresas no permiten optimizar la forma, frecuencia y calidad de las entrevistas.

4. RAZONAMIENTO

4.1 INFERENCIA

Antes de comenzar a estudiar las posibles formas de razonamiento, se describirán ciertos conceptos básicos sobre el razonamiento en sí y que se encuadrarán en la denominada *inferencia*.

- INFERENCIA

Denominaremos inferencia al proceso por el cual un programa de computadora es capaz de obtener nueva información.

- MODUS-PONENS

Todo el método de razonamiento de un Sistema Experto está basado en este principio. Según él, cuando en cualquier regla se puedan demostrar las condiciones antecedentes de la misma, se considerarán como ciertos sus consecuentes.

- MODUS TOLLENS

Según este principio, si en una regla aparece una afirmación entre sus consecuentes y se puede demostrar que no es cierta, se podrá deducir que alguno de sus antecedentes es falso. Ejemplo podrá ser la afirmación de que Si en una casa no hay corriente eléctrica, ENTONCES no se pueden encender las luces. Si se observa que en una casa hay lámparas encendidas, aplicando la regla anterior, se podrá deducir que esa casa tiene corriente eléctrica. Este razonamiento, que parece sencillo para el razonamiento humano, no ha podido ser incluido en casi ningún sistema de funcionamiento debido a las numerosas dificultades que introduce.

- UNIFICACION

Es el mecanismo por medio del cual se identifican las reglas que pueden ser ejecutadas en cada instante. Para ello se examinarán los antecedentes de las mismas, comprobando si todos ellos son ciertos.

Los antecedentes indican condiciones que deben cumplir elementos de la base de conocimiento. Estos elementos pueden expresarse, en la sintaxis de una regla como:

- a) Un único elemento de la base de conocimiento. Se indica el símbolo que lo identifica.

- b) Todos los elementos de una clase. Se indica mediante un símbolo especial que denota esta polivalencia. Se denominan variables a estos símbolos.

En el caso de operar con reglas que permitan el uso de variables, si aparece una de ella en la primera condición de la regla y existe un elemento en la base que cumpla dicha condición, ésta se considera como cierta y a partir de entonces la variable queda inicializada a dicho elemento para las demás condiciones de la regla que aparezca.

- RESOLUCIÓN

Es el proceso de demostración de una afirmación ante una serie de conocimientos y reglas de aplicación. El método más famoso por marcar el inicio de las técnicas de Inteligencia Artificial, es el de demostración por contradicción lógica. Dado que la mayoría de los sistemas existentes están basados en él, se dará una explicación breve sobre su funcionamiento.

Toda regla del sistema se simplifica en forma de una cláusula con conectivas "O" (OR). Desde el punto de vista lógico, las dos siguientes expresiones son equivalentes:

$$\text{IF A THEN B} \quad \Leftrightarrow \quad \text{NOT(A) OR B}$$

Realizando el mismo proceso para las siguientes reglas se tiene una serie de cláusulas de información. A continuación, se procede a simplificar dichas cláusulas reduciendo aquella en las que una misma condición aparezca como negada y afirmada al mismo tiempo.

Por ejemplo, de:

$$\text{A OR B} \quad \text{y} \quad \text{NOT(A)}$$

se obtiene: B

El método general de resolución para comprobar una afirmación consistente en situarla como condición negada en las cláusulas del sistema. Si al final de la operación de simplificación, existe una incoherencia en las cláusulas remanentes, esto es, que existen dos cláusulas con una misma condición, en una de ellas negada y en la otra no, se aceptará como válida la condición inicial.

4.2 CONTROL DEL RAZONAMIENTO

Como ya se ha indicado el motor de inferencia es la parte del sistema experto que se encarga de generar nueva información a partir de la información existente y de un conjunto de reglas de producción. La forma en que se realiza la obtención de la información se denomina *control del razonamiento*.

El *control del razonamiento* se encarga de establecer los mecanismos necesarios para resolver situaciones de conflicto utilizados al examinar las reglas y para lanzar el procedimiento de inferencia inicialmente. Dos son los métodos de control del razonamiento utilizados mayoritariamente: hacia atrás y hacia adelante.

El razonamiento hacia atrás (BACKWARD CHAINING) es un mecanismo de búsqueda ordenada, cuyo proceso de razonamiento es descendente y guiado por los objetivos o inductivos. Se aplica cuando el punto de partida es una hipótesis u objetivo; esto es, comprueba que una hipótesis es cierta en base a los hechos contenidos en la base de conocimientos.

Las reglas seleccionadas son las que verifican los consecuentes o efectos de acuerdo a la hipótesis inicial. Las condiciones o datos desconocidos existirán mientras existan subobjetos por demostrar; cuando estos hayan sido demostrados en su totalidad, se alcance la verificación de la hipótesis inicial o se termine la posibilidad de seleccionar otra regla, se habrá terminado la búsqueda.

Este tipo de razonamiento de reglas difiere del razonamiento hacia adelante principalmente en que asumimos que una conclusión es cierta y usamos las reglas de la base de conocimientos para tratar de probarla, verificando la certeza de las premisas de la regla que contiene la conclusión a probar.

Para probar la certeza de una premisa, buscamos una regla que tenga esa premisa como una de sus conclusiones; si tal regla es encontrada, razonamos hacia atrás esa regla y nos damos a la tarea de probar la certeza de las premisas de esa segunda regla de la misma forma que la inicial.

El razonamiento hacia atrás se basa en el "modus tollens" de la lógica formal y emplea procesos recursivos. Las soluciones deben ser conocidas de antemano.

El razonamiento hacia adelante (FORWARD CHAINING) es un mecanismo de búsqueda ordenada, cuyo proceso de razonamiento es ascendente y guiado por los datos o deductivo. Utilizado cuando se cuenta con datos básicos como punto de partida, esto es, parte de los hechos para llegar a los resultados.

La búsqueda se realiza seleccionando las reglas que verifican las condiciones necesarias para llevar a cabo el efecto, es decir, se aplica la regla para la cual la condición sea cierta según los datos con los que se cuenta. El proceso se sigue hasta que no existan más reglas aplicables o se encuentre la solución.

Explicándolo de otra forma, imaginaremos que somos el motor de inferencia. Sabemos que la base de conocimientos está formada por una lista de reglas que usamos para tomar decisiones. Cuando usamos el razonamiento hacia adelante, queremos encontrar una solución basada en hechos, entonces nos referimos a la

base de conocimientos como si no supiéramos nada y nos damos a la tarea de encontrar algo. Primeramente tenemos que tener algunos hechos acerca del problema y podemos preguntar por ellos explícitamente.

El método elemental del razonamiento hacia adelante nos indica que comencemos de manera arbitraria con la primera regla de la base de conocimientos y tratemos de usarla, para lo cual debemos conocer como ciertas sus premisas. En algún lugar vacío inicialmente, al que llamaremos "lista de certezas", almacenaremos todas las cosas que conocemos como ciertas.

Cuando la certeza de una premisa no es conocida y no puede ser determinada por otra regla, preguntamos al usuario o a una base de datos para obtener tal información.

Una vez que se usa una regla, adicionamos todas sus conclusiones a la lista de certezas y vamos por la siguiente regla en la base de conocimientos, que use la primera solución de la regla anterior como una de sus premisas y tratamos de usarla. Esto es a lo que llamamos razonamiento hacia adelante, el cual está basado en el "modus ponens".

4.3 APRENDIZAJE

Ya en la definición de Inteligencia Artificial se mencionó la capacidad de aprendizaje como algo importante y trascendente. En los sistemas expertos sería deseable, en muchos casos, disponer de mecanismos que permitieran de forma automática incorporar al sistema nueva experiencia.

Hay que distinguir, al hablar de aprendizaje, si se trata de incorporar conocimiento (hechos, relaciones, información) o reglas de inferencia.

El primer aspecto, más sencillo de abordar, está casi resuelto en las herramientas de desarrollo actuales. Físicamente las herramientas permiten crear nuevas variables, marcos, objetos, etc. ¿Cuándo se incorpora ese conocimiento? Es necesario escribir las reglas adecuadas o preparar un control del razonamiento capaz de ello.

El segundo aspecto es muy complicado, se trata de que un sistema experto autogenera nuevas reglas fruto de consultas realizadas. En la actualidad sólo existen dos o tres prototipos de investigación con estas características.

Es posible, sin embargo, utilizar soluciones intermedias:

⇒ Algunos sistemas expertos han previsto que determinados *parámetros* que aparecen en algunas reglas pueden ser modificados por reglas en

función de datos estadísticos. Esto es posible con casi cualquier herramienta de desarrollo.

⇒ Algunas herramientas generan las reglas a partir de ejemplos (combinaciones de datos de partida y conclusiones o situaciones finales).

La dificultad está en prever cuándo y cómo se incorpora una nueva regla. Hay que recordar que un sistema experto es un sistema vivo y que el ingeniero de conocimiento puede fácilmente ampliar y modificar el prototipo, por lo que el aprendizaje automático es en muchos casos innecesario o excesivamente costoso frente a la incorporación manual.

CAPITULO II
EL APRENDIZAJE Y LOS
SISTEMAS EXPERTOS
TUTORIALES

1. INTRODUCCIÓN

La educación del futuro desde la perspectiva de la "pedagogía del lenguaje total", será planteada como aquella que le brinde al individuo la oportunidad de ubicarse dentro de las condiciones reales que configuran a la sociedad y ya que los medios de comunicación forman parte muy importante en el concepto de conceptualización del mundo que se da en el individuo, la escuela del futuro deberá incorporar los medios de comunicación social como un instrumento para expresar la realidad, ofreciendo así al alumno la posibilidad de incorporarse a un mundo verdadero.

Larry Shorter nos dice que hoy en día la educación es decisión del individuo, y según sus necesidades, intereses e ideales es permanente o continua. La educación en el futuro deberá estar al alcance de todos, lo cual significa un aprendizaje activo, de investigación y exploración continua.

El aprendizaje es un factor que influye de modo importante en todos los seres vivos. Permite al hombre adaptarse a las modificaciones del ambiente para poder sobrevivir. La lucha por la supervivencia requiere realizar muchos ajustes y adaptaciones. Desde el momento de la concepción, la experiencia empieza a dar forma al organismo en desarrollo.

El término *aprendizaje* tiene una gran amplitud que abarca cambios tan distintos como la respuesta de un modo diferente a una señal, la adquisición de una habilidad, la alteración de la manera de percibir una cosa, el conocimiento de un hecho y el desarrollo de una actitud frente a algo. Las consecuencias últimas del aprendizaje van, inclusive, mucho más allá del individuo, puesto que las formas que presentan los idiomas, las religiones, las culturas, las leyes y las costumbres, son producto directo de modificaciones practicadas por la experiencia humana. El aprendizaje nunca es observado directamente, es siempre una inferencia derivada de la observación de un cambio, o de una serie de cambios.

Es difícil definir el concepto de educación porque la palabra "educación" se emplea de múltiples maneras. En su aplicación más común equivale a asistir a la escuela y traer a la mente toda gama de actividades que se llevan a cabo en los jardines de la infancia, escuelas, colegios, institutos y universidades. En este sentido su significado es vago, puesto que designa cualquiera de las diversas clases de aprendizaje.

En este contexto se podría decir que "la educación" también implica actitudes, creencias que se aprenden a través de la participación en la vida social. Así pues, *educación* designa el proceso general por el cual aceptamos los valores de nuestra sociedad, y por esta razón podemos decir que la educación es un proceso que dura toda la vida.

Una tercera concepción de la educación se refiere a la percatación y curiosidad intelectual por todo lo que ocurre en la tierra y la consiguiente búsqueda por satisfacer la curiosidad.

2. EL APRENDIZAJE TRADICIONAL

Aunque difieren mucho en sus puntos filosóficos, Platón y Aristóteles plantearon las bases fundamentales en la que se sustentan la perspectiva tradicional de la educación.

Ambos afirman la prioridad del intelecto y establecen la noción de educación como trascendencia lograda por una élite intelectual, donde las masas se mantienen en un estado físico y material suficientemente cómodo, pero de vasallaje mental.

Aunque sobre este aspecto han habido cambios substanciales y se han integrado otros elementos, la educación tradicional se basa en la filosofía de Platón y Aristóteles en el aspecto que se subraya la autoridad del maestro y se considera que su función consiste en impartir a sus alumnos el conjunto requerido de la asignatura.

"La tarea del maestro es la base y condición del éxito de la educación; a él le corresponde organizar el conocimiento, aislar y elaborar la materia que ha de ser aprendida, en una palabra, trazar el camino y llevar por él a sus alumnos".

"La escuela queda separada de la vida exterior y lo que ocurre en ella se ve primordialmente como una preparación para el futuro, más que como un enriquecimiento del presente". "La escuela se construye en un mundo aparte, al margen de la vida diaria, en un reintentado reservado y preservado del mundo exterior".

La escuela tradicional concede importancia a los conocimientos y a la cultura en general y sostiene que " la mejor forma de preparar al individuo para la vida, es formar su inteligencia, su capacidad de resolver problemas, sus posibilidades de atención y de esfuerzo.

Los conocimientos son valorados por su utilidad para ayudar al individuo en el progreso de su personalidad: edificando sólidos conocimientos se favorece el desarrollo global del individuo". De esta forma, la escuela tradicional significa por encima de todo, método y orden.

Algunas observaciones hechas en diferentes sistemas educativos permiten detectar que los estudiantes, casi siempre, permanecen sentados física y mentalmente escuchando, acatando ordenes y, por supuesto, aprendiendo a adivinar las respuestas correctas a formulaciones sesgadas. En otras palabras,

están aprendiendo a manipular, a percibir que resulta mejor la aceptación pasiva que la crítica, o que el aprendizaje memorístico es la mejor forma para triunfar.

2.1 APRENDIZAJE POR ABSORCIÓN

Si la educación se concibe como la absorción de datos, escuchando pasivamente en una aula en la que el profesor habla *ex-cátedra* en todo el momento, como un dueño único y absoluto de la verdad, los estudiantes no tienen nada que hacer. O más bien, si tienen algo que hacer: acumular memorísticamente hechos, cifras, nombres, citas célebres. Todo ello sin relación alguna entre sí.

El mejor estudiante, será aquel que sea capaz de repetir exactamente cuanto el profesor ha dicho durante horas y horas de transmisión verbalista: pues, además se cree que a mayor número de palabras escuchadas y memorizadas, será mayor el índice de aprendizaje.

Los resultados previsibles en este aprendizaje pueden ser desastrosos. Por una parte, los estudiantes lograrán ser muy buenos burócratas o muy buenos consumistas, obedientes lo mismo a la voz demagógica de las autoridades que a los reclamos publicitarios de los mensajes radiofónicos, etc.

Por otra parte, se aprende que nada importa lo que se diga, siempre y cuando esto agrade al oído de los interlocutores del sistema familiar, escolar o laboral. Este tipo de mensajes implícitos en el ambiente es el que más repercusiones tiene sobre las personas.

2.2 CONSIDERACIONES EN EL APRENDIZAJE

Es más educativo considerar a los estudiantes como objetos que como personas. Desde el punto de vista de *Skinneriano*, los individuos únicamente son máquinas, manipuladas mediante reforzamientos positivos o negativos.

Además, no conviene involucrarse con los estudiantes ni establecer con ellos relaciones profundas de amistad, porque ¿cómo un amigo puede ser verdugo, juez y examinador?. En consecuencia habrá que mantenerlos a distancia y conservar relaciones de "respeto".

Los estudiantes-objeto, aprenderán a considerarse como objetos y no sujetos de su proceso educativo, el cual, esencialmente, es un proceso de auto-desarrollo. Además si la propia institución educativa no se encuentran ni solidaridad ni amistad ¿cómo podríamos pensar que asimilen estos valores?. Este ambiente, en consecuencia, puede ser educativo o des-educativo.

2.3 AMBIENTE EDUCATIVO

Es necesarios crear un ambiente educativo: laboral, familiar, escolar, en el cual, quienes lo forman:

- a) **Confíen** profundamente de que no serán reprimidos, sino alentados para poner en juego su propia capacidad de resolver problemas y de realizar críticas desde su propia perspectiva científica e ideológica. Esta confianza mutua no implica que siempre se acierte en la solución. También se aprende por ensayo y error. Es más educativo crear un sistema en el que se respire confianza, no sólo por principios éticos o pedagógicos sino por la necesidad de responder a las demandas de una sociedad con agudas tensiones y un cambio acelerado.
- b) **Disfruten** a todo lo largo del proceso de investigación, pues la alegría también influye en una mayor capacidad de aprendizaje. La perspectiva de ser responsables del proceso de investigación deberá ser alentada no dando respuestas a todas las preguntas que se formulen a quien supuestamente sabe más, aunque el profesor tenga sinceros deseos de transformar a sus alumnos.
- c) **Distingan** claramente entre lo importante, lo esencial y lo secundario, lo accidental. En otras palabras, hay que aprender a *mirar* (en profundidad) no simplemente a ver (superficialmente), pues de otra manera no es posible *capturar* la diferencia entre lo importante y lo trivial.
- d) **Sean flexibles** tanto en lo que respecta a sus propias "verdades científicas" como en lo que respecta al valor de sus respuestas. Aunque confiados en su propio criterio, investigan y emiten opiniones fundamentadas una vez que tuvieron acceso a toda la información, directa o indirecta, disponible.

En otras palabras, comprenden que las verdades científicas son modelos provisionales para interpretar la realidad y que, por lo tanto, las respuestas son relativas, pues dependen de diversos enfoques teóricos, así como de diferentes sistemas metodológicos y perspectivas.

Al no cambiar actitudes en este sentido se corre el riesgo de ser manipuladores, al servicio de ideologías dominantes. De aquí que resulte más difícil establecer diálogo entre profesor y alumno. En consecuencia para formar la conciencia crítica resulta necesario cambiar los ambientes educativos diversos en los que hay que *aprender a ver* de un modo u otro.

Obviamente, no podrán introducirse innovaciones relevantes si no se modifican las actitudes de los "educadores", ya que sus actitudes, creencias y prejuicios determinan el clima de estos ambientes de aprendizaje.

3. LA COMPUTADORA EN LA EDUCACIÓN

La solución que se propone a este problema es la planeación educativa basada en la educación continua como principio básico, cuyo propósito sea el pleno desarrollo personal, y su resultado sería una sociedad individualmente más creativa y más satisfactoria.

Los planteamientos teóricos parecen interesantes, el problema fundamental es cómo ponerlos en práctica y además crear la infraestructura, y generar los recursos para llevar a cabo la tarea de poner la educación al alcance de todos. Algunas de las respuestas se presentan a continuación:

1. Crear programas de extensión Universitaria que establezcan fuentes alternas de educación, con base a las necesidades sentidas y manifiestas.
2. Integrar grupos de expertos en áreas de educación, pedagogía y telecomunicaciones, para utilizar la infraestructura existente, así como nuevos desarrollos y ofrecer posibilidades reales de educación continua, educación a distancia, etc.;
3. Evaluar la estructura de la currícula actual de las Instituciones de educación superior y las necesidades presentes y futuras de la sociedad en su conjunto. Referente a la educación superior, desarrollarla en la modalidad de continua, con una tecnología adecuada, para ofrecer alternativas reales.

La educación en cómputo e informática es uno de los grandes retos que enfrenta la sociedad de hoy, el lograr la formación e información en cómputo e informática para toda la comunidad es una vía indispensable para entender a la sociedad actual, sus estructuras, su interés, y todas las relaciones que se dan en un ámbito cada vez más tecnificado, y que incorporan innovaciones tecnológicas a un ritmo vertiginoso.

La educación actual se plantea consciente e inconscientemente la preparación del hombre para una sociedad futura, una sociedad en la que la electrónica, la computación y la informática juegan un papel relevante.

El cómputo y la informática han sido señalados como un conjunto de conocimientos para incorporarse en el proceso educativo; sin embargo, muchas de las experiencias han tenido pobres resultados, principalmente debido a que se han hecho como una introducción de modernidad mediante la incorporación de equipos y programas de cómputo, pero con escasa o nula vinculación con las necesidades reales en su conjunto y mucho menos con la práctica profesional, lo cual resulta como un parche de modernidad con pobres resultados en el proceso educativo.

3.1 APRENDIZAJE POR COMPUTADORA (TUTORIALES)

Las computadoras están apareciendo rápidamente en las escuelas de todo el mundo. En todos los países el mayor problema es el entrenamiento de los maestros para el uso de estas. Debido al rápido incremento en el número de computadoras en las escuelas, la pregunta es: ¿Qué sucede con las computadoras cuando llegan a las escuelas? Se tienen dos problemas:

- 1) La preparación de los maestros para usar el hardware.
- 2) Proveer adecuados materiales para la enseñanza con la computadora.

Casi la totalidad de los maestros simplemente *no tienen acceso a ningún programa apropiado* al empleo de las computadoras en la educación, ya que no tienen tradición en esta área. La solución al problema del entrenamiento de los maestros es desarrollar un buen material para el aprendizaje basado en el uso de computadoras *sobre la computadora*, solo para maestros. Los maestros aprenderían *en la computadora, no respecto a la computadora*.

La instrucción por computadora como un término general "*Instrucción*", como se aplica entre otras áreas, es ser capaz de leer y escribir de modo competente, ser capaz de realizar la aritmética esencial y de entender a la ciencia en un sentido general.

Estamos interesados en los maestros como maestros auxiliando a los estudiantes en el proceso de aprendizaje. Por lo que podemos enfocarnos al uso de la computadora en sus propias clases.

Una computadora por sí sola no despierta interés a los educadores como medio de enseñanza. El interés surge a partir del desarrollo de herramientas educativas para computadoras.

Se debe tener claro que el uso de las computadoras en la educación, al igual que otros elementos y técnicas, por sí mismas no se pueden catalogar de buenas o malas. Dado su potencial, pueden inclinarse hacia cualquier lado, por lo tanto la labor radica en aminorar sus defectos y en dar énfasis a sus cualidades.

3.2 LA COMPUTADORA COMO HERRAMIENTA PARA EL APRENDIZAJE

La clave está en que la computadora por sí misma proporciona una solución posible y factible. La computadora es un excelente instrumento para el aprendizaje y puede usarse para auxiliar tanto de los maestros como de los estudiantes. Programas de cómputo interactivos aunados al acceso de solución de problemas por computadora, pueden proporcionar una gama de posibilidades no alcanzables de ninguna otra forma. Cada persona puede ser tratada como un

individuo en particular, con todas las características que le distinguen de cualquier otra persona que se considere. Cada persona puede avanzar a su propio ritmo. Además, la experiencia del aprendizaje puede ser activa, con la persona como participante en lugar de como simple espectador.

En casi todas las Universidades la mayor parte del tiempo de los estudiantes se dedica a la lectura de libros y tal vez las conferencias estén en segundo lugar. La computadora sólo cuenta con una pequeña fracción del tiempo de aprendizaje del estudiante.

Al usar una computadora, primeramente contamos con un elevado factor de motivación que ésta despierta en los estudiantes, pues la gente recibe constantemente referencia de ellas en películas, televisión y medios de comunicación impresos. Los estudiantes que no tienen contacto con las computadoras ven en ellas máquinas apasionantes y ansían tener contacto con ellas.

Por sus experiencias en el Centro de Tecnología Educacional, A. Bork menciona que no ha encontrado estudiantes que no quisieran utilizar la computadora siempre que el uso de ellas no fuera prolongado. Así pues, la idea de estar frente a una máquina que "contesta" resulta atrayente para el estudiante, pues éste se siente como si estuviera unido a la aplicación. Aunado a esto, la computadora se presenta como un mecanismo que saca al estudiante de la rutina.

Dado que el grado de aprendizaje depende del tiempo que se dedique a estudiar un tema, es necesario poner énfasis a los factores motivadores, tales como la computadora, ya que inducen a que los estudiantes empleen más tiempo en aprender, por medio del repaso, al ser el tutorial un material adicional.

Con una computadora como material único de aprendizaje, los estudiantes adquieren conocimientos invirtiendo menos tiempo del que les tomaría hacerlo en una clase convencional, debido a que la computadora está dedicada a un solo estudiante. A Bork maneja una disminución del 30% del tiempo invertido gracias a la individualización: una vez que se verifican los conocimientos del estudiante, la aplicación puede pasar a otros temas, no antes. La cuestión del tiempo que se emplea para aprender es especialmente importante en los lugares donde el personal es apartado de su trabajo productivo en una compañía para estudiar algo.

Otra ventaja que poseen las computadoras es el que pueden proporcionar información visual interactiva. La interacción es en esencia, un diálogo entre el usuario final y la computadora, donde se usan datos como medio de comunicación; entre los datos se encuentran los gráficos que llaman mucho la atención ya que estimulan la parte creativa del cerebro, fomentando así la estimulación de ambos hemisferios del cerebro.

El uso de la computadora en la educación trae consigo la ventaja de poder presentar una lección como un juego en el que las respuestas incorrectas recibidas por la computadora hacen desplegar de alguna forma las correctas, fomentando así la instrucción-práctica.

Un tutorial nunca quedaría obsoleto, ya que éste se puede ir actualizando en información y en presentación. Además puede irse perfeccionando conforme se vayan presentando errores, mediante su uso, o bien se descubran puntos en los que se necesite remarcar un concepto.

3.3 EL SOFTWARE EDUCATIVO

Aunque este tipo de software puede tener algún valor didáctico, su objetivo principal no es auxiliar el proceso enseñanza-aprendizaje, en los procesos administrativos y de investigación de otra índole.

El software calificado como educativo es aquel que tiene como objetivo intervenir en el proceso de enseñanza-aprendizaje, ya sea en la forma tradicional, apoyando la transmisión o ejercitación de un conocimiento particular de un tema y materia determinada, o bien de una forma más innovadora, tratando de provocar en el alumno un interés genuino por la investigación. En este último caso el software plantea problemáticas y no contenidos particulares, que el alumno habrá de manipular a través de variables, advirtiendo, asimismo, las transformaciones surgidas de su interacción con el software durante el proceso simulado.

Este software proporciona al alumno una guía para la comprensión de un concepto, a partir de ejemplos particulares y tangibles que le permitan interaccionar y observar los efectos de esa interacción en diferentes niveles de representación del fenómeno.

Como instrumento de medición, el software y la computadora presenta un fuerte apoyo a las ciencias experimentales, ya que con éstos se pueden desarrollar experimentos sobre algunos conceptos que especialmente en los niveles educativos básicos son transmitidos como dogma.

Ante estas aseveraciones se pueden obtener algunas respuestas al problema planteado.

Es evidente que no podemos pensar que lógicamente el software educativo sustituye al maestro. Tampoco podemos pensar que representa una solución a todos los problemas de la enseñanza de las ciencias, especialmente de las ciencias básicas, ni mucho menos creer que constituye una forma rápida de aprender un tema en particular, sin el menor esfuerzo.

En cambio, sí podemos considerar el software educativo como un poderoso instrumento de apoyo, dinamización y revisión de nuestros actuales conceptos de aprendizaje, y de las formas actuales de transmisión y generación de conocimiento.

4. SISTEMAS EXPERTOS TUTORIALES

Desde el surgimiento de la computadora como medio de fácil acceso o adquisición, ha surgido el interés en sistemas de instrucción basados en computadoras que desplieguen cierto grado de inteligencia. Estos sistemas tutoriales inteligentes, lo que hemos llamado Sistemas Expertos Tutoriales (SET), son bien avenidos por muchos educadores y especialistas de instrucción, quienes los ven como desafíos en la investigación.

Estos sistemas se definen de la forma:

"Son programas de computadora que utilizan técnicas de inteligencia artificial para ayudar a una persona a aprender". (Kearsley 1987).

Todos los Sistemas Expertos son por sí mismos herramientas para la enseñanza. Los SET se enfocan a la educación, son inteligentes no en cuanto a su contenido, sino a la forma en como el sistema enseña su contenido. Un SET proporciona apoyo a una tarea experta: la enseñanza. Es valioso económicamente pero técnicamente limitado. Los SET se desarrollan generalmente en una computadora personal y se considera asistente. Es importante que un sistema de este tipo no presente solamente pantallas cargadas de información.

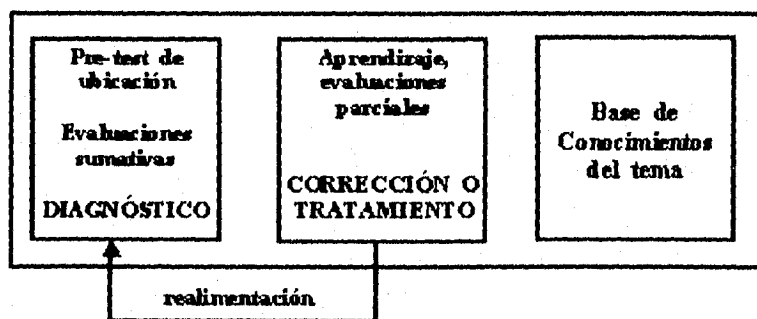


Figura 2.1 Estructura de un SET

Los primeros intentos de creación de software educacional tuvieron resultados variados. Comúnmente los proyectos habían hecho más sacando a la luz problemas que cumpliendo las promesas, tanto de realizaciones técnicas como pedagógicas y su integración dentro de los fines de las instituciones educacionales. La dificultad de producir buen material dejó cierta incertidumbre, llegando a la desilusión; fue la Inteligencia Artificial quien dio nuevas esperanzas al viejo sueño. Dado el gran interés generado alrededor de la I.A. , ampliado por lo referente al gran potencial de la educación e instrucción, se ha motivado el desarrollo de varios proyectos de investigación en el área de los SET.

Mucho de la investigación sobre el tema de software educacional dentro de la I.A. ha sido conducido bajo el nombre ICAI (*Intelligent Computer-Aided Instruction*, Instrucción Inteligente Apoyada en una Computadora), más recientemente al acrónimo del ICAI ha sido reemplazado por el de ITS (*Intelligent Tutoring System*, Sistemas Tutoriales Inteligentes o SET).

En los sistemas ICAI, la experiencia ha ser comunicada es contenida en bloques de presentación acumulados anteriormente, algunas veces llamados *frames* los cuales son diseñados por el maestro experto y son desplegados al estudiante bajo condiciones dadas. Entre la transición del ICAI a SET, se encuentra históricamente el primer aspecto en el que el conocimiento de la materia da como resultado la representación explícita en los sistemas de la experiencia del maestro.

En los sistemas que comunican el conocimiento, hay un módulo especial llamado comúnmente experto, que contiene una representación del conocimiento a ser comunicado. En la mayoría de los casos esta representación del conocimiento no es sólo una descripción de varios conceptos y destrezas que el estudiante debe adquirir, el modelo actual provee al sistema con una forma dinámica de experiencia.

Con su interés en la Inteligencia Artificial: la epistemología, la psicología y la ciencia cognitiva están ligadas a la investigación automatizada tanto como la I.A. lo está. Estas disciplinas buscan entender y construir modelos de cómo la gente conoce, como aprende. Los psicólogos han incorporado los propósitos y metodologías de los SET dentro de su investigación y han encontrado herramientas en el desarrollo de teorías cognitivas.

Obviamente la educación y la psicología educacional están muy ligadas en su intento de capturar los componentes de la experiencia pedagógica para acumular material de técnicas empíricas útiles y para producir herramientas educacionales útiles.

4.1 MÓDULOS BÁSICOS DE UN SET

El área en particular que cubrirá el SET necesita llegar a ser comprendida mediante preguntas profundas acerca de la naturaleza del conocimiento que abarca, la forma en cómo debe ser comunicado el conocimiento y la forma en cómo el aprendizaje del área es óptima. Los SET permanecen aún difíciles de construirse, la mayoría de los proyectos requieren años de trabajo, de gente talentosa, y que son mejorados a través de varias generaciones de prototipos, algunos de los cuales se quedan en el laboratorio.

La enseñanza es difícil y el hecho de diseñar SET requiere de comprender todo el proceso que envuelve una aplicación en particular, (no todos los problemas se atacan de la misma forma; así que no se puede sistematizar).

A pesar de la diversidad de arquitecturas que se manejan alrededor de los sistemas, es posible distinguir los módulos básicos que los conforman:

1. **Módulo experto:** Consiste de conocimientos sobre un dominio específico del cual el sistema intenta enseñar al estudiante. Este módulo se utiliza para generar el conocimiento instruccional, así como para evaluar el desempeño del estudiante. Los métodos de I.A. más representativos son las redes semánticas sistema de reglas de producción, representación procedural y la construcción de marcos o guiones.
2. **Módulo del estudiante:** Se utiliza para estimar el estado de conocimientos del estudiante y hacer hipótesis acerca de las estrategias de razonamiento utilizadas para alcanzar el estado actual de conocimientos.

Como la mayoría de los SET representan el conocimiento del estudiante como un subconjunto de la base de conocimientos de un sistema experto, el modelo se construye comparando el desempeño del estudiante con el desempeño de un sistema experto en la misma tarea. A esta técnica se le llama el "*Modelo de Sobreposición*".

Otra técnica es la de representar las tareas erróneas o mal aprendidas. Las cuales no son subconjuntos de la base de conocimientos expertos como variantes de este conocimiento. A esta técnica se le conoce como "*Modelo de Depuración*".

El modelado del conocimiento del estudiante y el comportamiento de aprendizaje, básicamente utilizan dos rutinas:

- *Trazar entre la estructura de conocimientos aquellas áreas en las que el estudiante ha dominado o ha intentado aprender; y*

- *Aplicar patrones de reconocimiento al histórico de respuestas del estudiante para hacer inferencias acerca del entendimiento de la habilidad y el proceso de razonamiento utilizado para llegar a esa respuesta.*

Para mantener el modelo del estudiante, existen cuatro fuentes principales de información:¹

- I. El comportamiento del estudiante es la solución de problemas, observado por el sistema.
- II. Cuestionamientos directos al estudiante.
- III. Suposiciones basadas en la experiencia de aprendizaje del estudiante.
- IV. Suposiciones basadas en alguna medida de la dificultad del material presentado.

Sin embargo, la mayoría de los sistemas utilizan únicamente las dos primeras fuentes de información para mantener el modelo del estudiante.

3. **Módulo de la evaluación:** Es la parte inteligente del sistema, ya que esta es quien decide si un estudiante se encuentra apto para pasar al siguiente tema o debe repetir el anterior. La manera de como se desarrolle este módulo dependerá de los fines para los cuales este pensada la aplicación. Es un conjunto de especificaciones acerca de cuál es el material que el sistema debe presentar, y de cómo y cuándo debe hacerlo.

En los SET existentes, las estrategias de instrucción básicamente se representan por dos métodos:

- a) **El método Socrático**, provee a los estudiantes de cuestionamientos para guiarlos a través de un proceso de depuración de sus propias malas interpretaciones. En el proceso de depuración, se asume que el estudiante razona acerca de lo que sabe y lo que no sabe y de esta forma modifica sus interpretaciones.²
- b) **El método de entrenamiento**, provee a los estudiantes un ambiente en el cual los compromete en actividades como juegos por computadora con el fin de aprender habilidades relacionadas o habilidades generales para la solución de problemas. La meta del programa es tener a los estudiantes entretenidos y aprender como consecuencia de la diversión (Burton 1979).

¹ Clancey 1979

² Carbonell 1970 y Stevens 1977

Los tres componentes no están totalmente desarrollados en cada sistema dado la complejidad y el tamaño de los SET. La mayoría de ellos se enfoca en el desarrollo de un solo componente de lo que constituye un sistema completo utilizable.³

4.2 TIPOS DE SISTEMAS EXPERTOS TUTORIALES

TUTORES MEZCLADOS CON INICIATIVAS

Estos programas representan a los más antiguos SET, en donde, se compromete al estudiante a una conversación en dos sentidos y se intenta enseñarle vía el método Socrático con un descubrimiento guiado. Este tipo de sistemas se utiliza con mayor frecuencia en la enseñanza de habilidades conceptuales o procedurales. Un ejemplo representativo lo conforman los sistemas SCHOLAR y SOPHIE.

TUTORES DE ENTRENAMIENTO

Un entrenador observa el desempeño del estudiante y lo provee de consejos que le ayudan a mejorar su desempeño. Los sistemas de entrenamiento se utilizan en programas del tipo solución de problemas como son simulaciones y juegos. Ejemplos de este tipo de sistemas son WEST y WUSOR.

TUTORES DE DIAGNÓSTICO

Este tipo de sistemas intentan depurar el trabajo del estudiante. Los sistemas manejan un catálogo de errores que identifican los conceptos mal entendidos o faltantes que un estudiante puede tener al momento de resolver un problema. Los tutores de diagnóstico son apropiados para casi cualquier tipo de situación de resolución de problemas, aunque son sencillos de implantar para problemas con soluciones cerradas. Ejemplos de este tipo de sistemas son BUGGY y PROUST.

EL CONCEPTO DE MICROMUNDOS

Este tipo de sistemas involucran el desarrollo de herramientas computacionales que le permiten al estudiante explorar dominios como la geometría, la física o la música. El mejor ejemplo conocido es LOGO.

SISTEMAS EXPERTOS

Este tipo de sistemas pueden ser utilizados como herramientas de trabajo y para proveer prácticas en la solución de problemas y toma de decisiones. Claramente

³ Clancey 1979

se tiene un gran potencial de aplicaciones en el área aunque el uso de sistemas expertos ya sea en entrenamiento o educación no ha sido muy difundido.

La diferencia con los sistemas expertos convencionales es que los SET incluyen un módulo adicional que contiene las reglas que controlan el proceso de enseñanza.

Al estudiar esta clasificación es importante tener en cuenta que cada tipo de SET trata con un conjunto de elementos de la ciencia cognitiva e ignora los otros. Actualmente ningún tipo cubre lo concerniente a los SET. Sin duda alguna, en el futuro se generarán nuevos tipos de SET.

A pesar de sus grandes diferencias, la mayoría de estos sistemas tratan con elementos y componentes similares. Por ejemplo, el elemento más importante en el desarrollo de un SET es la forma apropiada de representar el conocimiento del que se trata. Otro elemento de gran importancia es como modelar el comportamiento actual del estudiante.

El compartir los conceptos y metodología proveen en los SET un objetivo común y una fundamentación teórica. Esta situación es análoga al período de la aviación en la que los aviones de propulsión a chorro sustituyeron a los aviones de hélice. La mayoría de los conocimientos acerca de volar fueron aún relevantes pero tuvieron que ser replanteados en términos de la nueva tecnología.

4.3 SISTEMAS EXPERTOS TUTORIALES DESARROLLADOS

Los SET tienen sus raíces durante los años 20's con el desarrollo de lo que se llamó "*Las máquinas de enseñanza*". Estas máquinas fueron un intento de construir dispositivos interactivos de enseñanza. Skinner en los años 50's en sus trabajos sobre la instrucción, aportó las bases de la metodología para la instrucción lineal programada tan difundida durante los años 50's y 60's. La instrucción programada fue el modelo para muchos esfuerzos iniciales en SET.

Uno de los esfuerzos más representativos fue desarrollado en la Universidad de Stanford por el investigador Suppes (1968), en donde se demostró que en un ciclo regular añadiendo un poco de instrucción y sesiones de práctica con una computadora, se podrían incrementar significativamente las habilidades del estudiante.

Durante los 70's se desarrollaron varios proyectos tanto en Universidades como Institutos de entrenamiento (la mayoría se iniciaron en los sesenta): En los más significativos se encontraron SOLO, en la Universidad de Pittsburgh, con el fin de enseñar el uso de computadoras como herramientas personales y los sistemas PLATO y TCCIT que se desarrollaron con la finalidad de proporcionar sistemas de instrucción de bajo costo.

Estos sistemas fueron probados en Institutos y Universidades y eventualmente se convirtieron en productos comerciales. Su principal beneficio fue el permitir a un gran número de investigadores el obtener experiencia práctica, así como la apertura de una nueva era en los SET con la aparición de las microcomputadoras.

Otro desarrollo importante en los SET se dio en la estructura de los componentes de instrucción. En sus primeros años todos los componentes estaban combinados y almacenados en un mismo archivo. Esto implicaba que cualquier modificación al contenido o a las reglas de instrucción derivaba en prácticamente un sistema nuevo. A principios de los 70's Seidel desarrolló un proyecto para la Armada Americana llamado IMPACT en el cual el contenido de la materia (textos y gráficas) y las reglas de instrucción, estaban separados en diferentes archivos de datos. Asimismo, este sistema permitía que una variedad de actividades sucedieran en forma simultánea.

Aunque el prototipo del sistema IMPACT tenía la característica de una presentación sensible a las respuestas, el proceso de instrucción incluyendo la presentación de formatos y las interacciones entre la computadora y el estudiante, tenían que ser especificadas en un programa antes de la instrucción, lo que traía un método muy rígido de enseñanza.

Con el fin de superar estas limitaciones (Inherentes a las estructuras de formas preespecificadas), aunado a las características de generar nuevos problemas a partir de las combinaciones de diferentes elementos, utilizando para ello técnicas modernas de representación de conocimientos, surgieron los Sistemas Expertos Tutoriales.

El inicio de los SET lo marcó el sistema SCHOLAR desarrollado para enseñar geografía sudamericana⁴. La base de datos de SCHOLAR es compleja pero bien estructurada con la forma de una red de conceptos, datos y procedimientos. Los elementos de la red son unidades de información definiendo palabras y eventos en la forma de arboles multiniveles. El método de enseñanza es el diálogo Socrático. El sistema primero diagnostica los conceptos erróneos o mal interpretados y luego presenta material que fuerza al estudiante a ver sus propios errores. Los mecanismos de inferencia para contestar las preguntas de los estudiantes y evaluar sus respuestas son independientes del contenido de la red semántica y es aplicable a diferentes áreas de dominio.

Como extensión a SCHOLAR surge el sistema WHY⁵ diseñado para enseñar la causas del fenómeno de lluvia repentina, un complejo proceso geofísico que está en función de muchos factores no relacionados, WHY implanta el método de

⁴ Carbonell 1970

⁵ Stevens 1977

evaluación Socrático Heurístico para describir las estrategias globales utilizadas por los seres humanos para guiar los diálogos.

Otro avance importante en la aplicación de técnicas de I.A. en el campo de los SET fue el crear nuevos ambientes educacionales a través del control total de la experiencia aprendida por el estudiante. Como ejemplo el Sistema LOGO.

Un aspecto interesante de los Sistemas Tutoriales fue el desarrollo de características auto-mejorables. Estos tipos de sistemas tenían dos grandes componentes: Un programa de enseñanza adaptativo que se expresa en reglas de producción y un componente auto-mejorable que lleva a cabo cambios experimentales en las reglas de producción del programa de enseñanza. Este trabajo fue particularmente importante por su naturaleza adaptativa. Como ejemplos de esta línea de investigación existen el tutor cuadrático auto-mejorable y el tutor de integración simbólica.

CAPITULO III
LÓGICA DIFUSA

PRIMERA PARTE: MATEMÁTICAS DIFUSAS

1. CONJUNTOS DIFUSOS

1.1 DEFINICIONES BÁSICAS

Un *conjunto clásico* (crisp o determinístico) es normalmente definido como una colección de elementos u objetos $x \in X$ que puede ser finito, contable o incontable. Cada elemento simple puede pertenecer o no pertenecer a un conjunto A , $A \subseteq X$. En el caso formal, al declarar "x pertenece a A" es verdadero, puesto que en el otro caso esta declaración es falsa.

Como un conjunto clásico puede ser descrito en diferentes formas: uno puede enumerar o no los elementos que pertenecen a el conjunto; describir el conjunto analíticamente, por instancia, por estados de condiciones para pertenencia

$$(A = \{x|x \leq 5\})$$

o definir los elementos miembros para usar la función característica, en que 1 indica membresía (pertenencia) y 0 no-membresía (no-pertenencia). Para un conjunto difuso, las funciones características toman varios grados de pertenencia para los elementos de un conjunto dado.

Definición 1

Si X es una colección de objetos denotados genéricamente por x entonces un *conjunto difuso* \tilde{A} en X es un conjunto de parejas ordenadas:

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in X\}$$

$\mu_{\tilde{A}}(x)$ es llamada la función de pertenencia o grado de pertenencia (también grado de compatibilidad o grado de raíz) de x en \tilde{A} con mapas X a el espacio de pertenencia M . (Cuando M contiene sólo los puntos 0 y 1, \tilde{A} es no-difusa y $\mu_{\tilde{A}}(x)$ es igual a la función característica de un conjunto no-difuso). El rango de la función de pertenencia es un subconjunto de la no-negatividad de los números reales cuyo supremo es finito. Elementos con un cero grado de pertenencia son normalmente no listados.

Definición 2

El soporte de un conjunto difuso \tilde{A} , $S(\tilde{A})$, es el conjunto determinístico de toda $x \in X$ tal que $\mu_{\tilde{A}}(x) > 0$.

Definición 3

Al grado menor α de el conjunto de elementos que pertenecen al conjunto difuso \tilde{A} se le llama el *conjunto nivel- α* :

$$A_\alpha = \{x \in X \mid \mu_{\tilde{A}}(x) \geq \alpha\}$$

$A'_\alpha = \{x \in X \mid \mu_{\tilde{A}}(x) > \alpha\}$ es llamado "conjunto fuerte nivel- α " ó "fuerte corte- α "

Definición 4

Un conjunto difuso \tilde{A} es *convexo* si:

$$\mu_{\tilde{A}}(\lambda x_1 + (1-\lambda)x_2) \geq \min(\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)), \quad x_1, x_2 \in X, \lambda \in [0, 1]$$

Alternativamente, un conjunto difuso es convexo si todos los conjuntos nivel- α son convexos.

Definición 5

Para un conjunto difuso finito \tilde{A} la *cardinalidad* $|\tilde{A}|$ se define como:

$$|\tilde{A}| = \sum_{x \in X} \mu_{\tilde{A}}(x)$$

$$\|\tilde{A}\| = \frac{|\tilde{A}|}{|X|}$$

2. MEDIDAS DIFUSAS Y MEDIDAS DE FUZZIFICACIÓN**2.1 MEDIDAS DIFUSAS**

Para prevenir confusiones entre medidas difusas y medidas de fuzzificación, comenzaremos con una breve descripción del significado y características de medidas difusas. A finales de 1970's, Sugeno define una medida difusa como sigue:

Sugeno [1977]: \mathfrak{B} es un campo Borel de un conjunto arbitrario (universo) X .

Definición 6

Dada una función g definida en \mathfrak{B} que tiene las siguientes propiedades es llamada medida difusa:

1. $g(\phi) = 0$, $g(X) = 1$
2. Si $A, B \in \mathcal{B}$ y $A \subseteq B$, entonces $g(A) \leq g(B)$.
3. Si $A_n \in \mathcal{B}$, $A_1 \subseteq A_2 \subseteq \dots$ entonces $\lim_{n \rightarrow \infty} g(A_n) = g(\lim_{n \rightarrow \infty} A_n)$.

La medida de Sugeno difiere de la medida clásica esencialmente por la disminución de la propiedad de aditividad⁶. Una aproximación diferente, sin embargo, es usada por Klement y Schwyhla [1982]. El lector interesado podrá referirse a su artículo.

Banon [1981] muestra que muchas medidas con universo finito, tales como las medidas de probabilidad, funciones de creencia, medidas de plausibilidad y muchas más, son medidas difusas en el sentido de Sugeno.

En el marco de la teoría de conjuntos difusos Zadeh incorpora la noción de distribución de posibilidad y el concepto de medida de posibilidad, la cual es un tipo especial de medida difusa propuesta por Sugeno. Una medida de posibilidad es definida como sigue⁷

Definición 7

Dado $P(X)$ como el conjunto potencia de un conjunto X .

Una medida de posibilidad es una función $\Pi: P(X) \rightarrow [0, 1]$ con las propiedades

1. $\Pi(\phi) = 0$, $\Pi(X) = 1$
2. $A \subseteq B \Rightarrow \Pi(A) \leq \Pi(B)$
3. $\Pi(\bigcup_{i \in I} A_i) = \sup_{i \in I} \Pi(A_i)$ con un conjunto índice I

Esto puede ser determinado únicamente por una función de distribución de posibilidad $f: X \rightarrow [0, 1]$ por $\Pi(A) = \sup_{x \in A} f(x)$, $A \subset X$. De esto resulta directamente que f es definida por⁸ $f(x) = \Pi(\{x\}) \forall x \in X$

Una posibilidad no es siempre una medida difusa⁹. Esta es, sin embargo, una medida difusa, si X es finita y si la distribución de posibilidad es un mapeo dentro de $[0, 1]$.

Ejemplo

Dado $X = \{0, 1, \dots, 10\}$.

$\Pi(\{x\})$: = posibilidad que x es cercano a 8.

⁶ [Murofushi y Sugeno 1989, pag. 201].

⁷ [Zadeh 1978; Higashi y Klir 1982].

⁸ [Klir y Folger 1988, pag. 122].

⁹ [Puri y Ralescu 1982].

x	0	1	2	3	4	5	6	7	8	9	10
$\Pi(\{x\})$.0	.0	.0	.0	.0	.1	.5	.8	1	.8	.5

$\Pi(A)$ = Posibilidad que A contenga un entero cercano a 8.

$$A \subset X \Rightarrow \Pi(A) = \sup_{x \in A} \Pi(\{x\})$$

$$\begin{aligned} \text{Para } A = \{2, 5, 9\} \text{ calcularemos } \Pi(A) &= \sup_{x \in A} \Pi(\{x\}) \\ &= \sup\{\Pi(2), \Pi(5), \Pi(9)\} \\ &= \sup\{0, .1, .8\} \\ &= .8 \end{aligned}$$

2.2 MEDIDAS DE FUZZIFICACIÓN

Las medidas de fuzzificación en contraste con las medidas difusas intentan indicar el grado de fuzzificación de un conjunto difuso. Varios planteamientos para este fin han llegado a ser conocidos. Algunos autores, fuertemente influenciados por la entropía de Shannon como una medida de información, y siguiendo a de Luca y Termini [1972], consideran una medida de fuzzificación como un mapeo del conjunto potencia $P(X)$ hacia $[0, +\infty]$ satisfaciendo un número de condiciones. Otros (Kaufmann 1975) sugirieron un índice de fuzzificación como una distancia normalizada, y otros ¹⁰ basan su concepto en una medida de fuzzificación en el grado de distinción entre el conjunto difuso y su complemento.

Deberíamos discutir, como una ilustración, dos de estas medidas. Suponiendo para ambos casos que el apoyo de A es finito.

Primer caso: Dada $\mu_{\tilde{A}}(x)$ es una función de pertenencia de un conjunto difuso \tilde{A} para $x \in X$, X es finito. Parece plausible que la medida de fuzzificación $d(\tilde{A})$ debe entonces tener las siguientes propiedades (de Luca y Termini 1972):

1. $d(\tilde{A}) = 0$ si \tilde{A} es un conjunto clásico en X .
2. $d(\tilde{A})$ toma un máximo único si $\mu_{\tilde{A}}(x) = \frac{1}{2} \forall x \in X$.
3. $d(\tilde{A}) \geq d(\tilde{A}')$ si \tilde{A}' es "crisper" que \tilde{A} , i.e. si $\mu_{\tilde{A}'}(x) \leq \mu_{\tilde{A}}(x)$ para $\mu_{\tilde{A}}(x) \leq \frac{1}{2}$ y $\mu_{\tilde{A}'}(x) \leq \mu_{\tilde{A}}(x)$ para $\mu_{\tilde{A}}(x) \geq \frac{1}{2}$.
4. $d(\tilde{C}\tilde{A}) = d(\tilde{A})$ donde $\tilde{C}\tilde{A}$ es el complemento de \tilde{A} .

De Luca y Termini sugieren como una medida de fuzzificación la "entropía" de un conjunto difuso ¹¹ que ellos definen a continuación:

¹⁰ [Yager 1979; Higashi y Klir 1982]

¹¹ [de Luca y Termini 1972, pag. 305].

Definición 8a

La entropía como una medida de un conjunto difuso $\tilde{A} = \{(x, \mu_{\tilde{A}}(x_i))\}$ es definida como:

$$d(\tilde{A}) = H(\tilde{A}) + H(\mathbf{C}\tilde{A}), \quad x \in X$$

$$H(\tilde{A}) = -K \sum_{i=1}^n \mu_{\tilde{A}}(x_i) \ln(\mu_{\tilde{A}}(x_i))$$

donde n es el número de elementos en apoyo de \tilde{A} y K es una constante positiva.

Usando la función de Shannon $S(x) = -x \ln x - (1-x)$, simplificaron la definición 8a para llegar a:

Definición 8b

La entropía como una medida de fuzzificación de un conjunto difuso $\tilde{A} = \{(x, \mu_{\tilde{A}}(x_i))\}$ esta definida como:

$$d(\tilde{A}) = K \sum_{i=1}^n S(\mu_{\tilde{A}}(x_i))$$

Ejemplo

Dada $\tilde{A} = \text{"enteros cercanos a 10"}$

$\tilde{A} = \{(7, .1), (8, .5), (9, .8), (10, 1), (11, .8), (12, .5), (13, .3)\}$
 $K = 1$, así

$$d(\tilde{A}) = .325 + .693 + .501 + 0 + .501 + .693 + .325 = 3.038$$

El segundo caso de medida es: Knopmacher (1975), Loo (1977), Gottwald (1979b), y otros, basaron sus contribuciones en algunos aspectos sugeridos por de Luca y Termini.

Si \tilde{A} es un conjunto difuso en X y $\mathbf{C}\tilde{A}$ su complemento, entonces por contraste a los conjuntos clásicos, no es necesariamente cierto que

$$\tilde{A} \cup \mathbf{C}\tilde{A} = X$$

$$\tilde{A} \cap \mathbf{C}\tilde{A} = \emptyset$$

De esta manera los conjuntos difusos no siempre satisfacen la ley de tercero excluido, la cual es una de sus mayores distinciones para los conjuntos tradicionales.

Algunos autores ¹² consideran la relación entre \tilde{A} y $C\tilde{A}$ para ser la esencia de la fuzzificación.

Yager (1979) notó que los requerimientos de distinción entre \tilde{A} y $C\tilde{A}$ no es satisfactorio para los conjuntos difusos. Por lo tanto sugirió que cualquier medida de fuzzificación deberá ser una medida de falta de distinción entre \tilde{A} y $C\tilde{A}$ ó $\mu_{\tilde{A}}(x_i)$ y $\mu_{C\tilde{A}}(x_i)$.

Como una métrica posible para la medida de la *distancia* entre un conjunto difuso y su complemento, Yager sugiere:

Definición 9

$$D_p(\tilde{A}, C\tilde{A}) = \left[\sum_{i=1}^n \left| \mu_{\tilde{A}}(x_i) - \mu_{C\tilde{A}}(x_i) \right|^p \right]^{1/p} \quad p = 1, 2, 3, \dots$$

donde $S = \sup(\tilde{A})$: $D_p(S, CS) = \|S\|^{1/p}$

Definición 10 ¹³

Una medida de fuzzificación de \tilde{A} puede ser definida como:

$$f_p(\tilde{A}) = 1 - \frac{D_p(\tilde{A}, C\tilde{A})}{\| \sup p(\tilde{A}) \|}$$

Así $f_p(\tilde{A}) \in [0, 1]$. Esta medida además satisface las propiedades 1 a 4 requeridas por de Luca y Termini.

Para $p = 1$, $D_1(\tilde{A}, C\tilde{A})$ produce la métrica Hamming

$$D_1(\tilde{A}, C\tilde{A}) = \sum_{i=1}^n \left| \mu_{\tilde{A}}(x_i) - \mu_{C\tilde{A}}(x_i) \right|$$

Porque $\mu_{C\tilde{A}}(x) = 1 - \mu_{\tilde{A}}(x)$, este comienza

$$D_1(\tilde{A}, C\tilde{A}) = \sum_{i=1}^n \left| 2\mu_{\tilde{A}}(x_i) - 1 \right|$$

¹² [Yager 1979; Higashi y Klir 1982].

¹³ [Yager 1979].

Para $p = 2$, llegamos a la métrica Euclidiana

$$D_2(\tilde{A}, C\tilde{A}) = \left(\sum_{i=1}^n \left(\mu_{\tilde{A}}(x_i) - \mu_{C\tilde{A}}(x_i) \right)^2 \right)^{1/2}$$

y para $\mu_{C\tilde{A}}(x) = 1 - \mu_{\tilde{A}}(x)$, tenemos

$$D_2(\tilde{A}, C\tilde{A}) = \left(\sum_{i=1}^n 2\mu_{\tilde{A}}(x_i)^2 \right)^{1/2}$$

3. EL PRINCIPIO DE EXTENSIÓN Y APLICACIONES

3.1 EL PRINCIPIO DE EXTENSIÓN

Uno de los conceptos más básicos en la teoría de conjuntos difusos el cual puede ser usado para generalizar conceptos matemáticos determinísticos hacia los conjuntos difusos es el principio de extensión. En su forma elemental ya estuvo implícito en la primera contribución de Zadeh (1965). Mientras tanto, han sido sugeridas modificaciones (Zadeh 1973a; Zadeh et al. 1975; Jain 1976). Siguiendo a Zadeh y Dubois y Prade (1980a) definimos el principio de extensión como sigue:

Definición 11

Supongamos que X es un producto cartesiano de universos $X = X_1, \dots, X_r$, y $\tilde{A}_1, \dots, \tilde{A}_r$ son r conjuntos difusos en X_1, \dots, X_r , respectivamente. f es mapeo de X hacia el universo Y , $y = (X_1, \dots, X_r)$. Entonces el principio de extensión nos permite definir un conjunto difuso \tilde{B} en Y por:

$$\tilde{B} = \{(y, \mu_{\tilde{B}}(y)) \mid y = f(x_1, \dots, x_r), (x_1, \dots, x_r) \in X\}$$

donde

$$\mu_{\tilde{B}}(y) = \begin{cases} \sup_{(x_1, \dots, x_r) \in f^{-1}(y)} \min \left\{ \mu_{\tilde{A}_1}(x_1), \dots, \mu_{\tilde{A}_r}(x_r) \right\} & \text{si } f^{-1}(y) \neq \emptyset \\ 0 & \text{en otro caso} \end{cases}$$

donde f^{-1} es el inverso de f .

Para $r = 1$, el principio de extensión, claro esta se reduce a:

$$\tilde{B} = f(\tilde{A}) = \{(y, \mu_{\tilde{B}}(y)) \mid y = f(x), x \in X\}$$

donde

$$\mu_{\tilde{B}}(y) = \begin{cases} \sup_{x \in f^{-1}(y)} \mu_{\tilde{A}}(x) & \text{si } f^{-1}(y) \neq \emptyset \\ 0 & \text{en otro caso} \end{cases}$$

El principio de extensión como se indica en la *definición 11* puede y ha sido modificado para usarse en la suma algebraica en lugar de *sup*, y el producto en vez de *min* (Dubois y Prade 1980a). Desde entonces, sin embargo, es generalmente usado como se señaló en la *definición 11*, limitaremos nuestras consideraciones a esta versión "clásica".

3.2 OPERACIONES ALGEBRAICAS CON NÚMEROS DIFUSOS

Definición 12

Un *número difuso* \tilde{M} es un conjunto difuso convexo normalizado \tilde{M} de la línea real \mathfrak{R} tal que:

1. Existe exactamente un $x_0 \in \mathfrak{R}$ con $\mu_{\tilde{M}}(x_0) = 1$ (x_0 es llamado el valor medio de \tilde{M})
2. $\mu_{\tilde{M}}(x)$ es un camino de piezas continuas

Hoy en día la *definición 12* es frecuentemente modificada. Por la eficiencia computacional y la fácil adquisición de datos, la función de pertenencia trapezoidal es frecuentemente usada. La figura (3-1) muestra tal conjunto difuso, el cual puede ser llamado "aproximadamente 5" y el cual puede ser normalmente definido como un cuádruple $\{3, 4, 6, 7\}$. Estrictamente hablando, esto es un intervalo difuso. Un número difuso triangular es, por supuesto, un caso especial de esto.

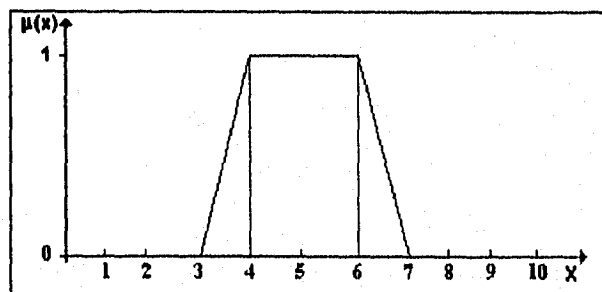


figura 3-1 "número difuso" Trapezoidal

Definición 13

Un número difuso \tilde{M} es llamado *positivo* (*negativo*) si su función de pertenencia es tal que $\mu_{\tilde{M}}(x_0) = 0, \forall x < 0$ ($\forall x > 0$).

Todos estamos familiarizados con operaciones algebraicas con números determinísticos. Si queremos utilizar conjuntos difusos en aplicaciones tendríamos que utilizar números difusos y el principio de extensión, es una manera de extender números determinísticos de operaciones algebraicas a números difusos. Necesitamos un poco más de definiciones: Suponiendo $F(\mathfrak{R})$ sea el conjunto difuso de los números reales y $X = X_1 \otimes X_2$. Podemos definir las siguientes propiedades de las operaciones binarias:

Definición 14

Una operación binaria $*$ en \mathfrak{R} es llamada *creciente* (*decreciente*) si para:

$$x_1 > x_2 \text{ y } y_1 > y_2$$

$$x_1 * x_2 > y_1 * y_2 \quad (x_1 * x_2 < y_1 * y_2)$$

Ejemplo

- | | |
|----------------------|--|
| $f(x,y) = x + y$ | es una operación creciente. |
| $f(x,y) = x \cdot y$ | es una operación creciente en \mathfrak{R}^+ . |
| $f(x,y) = -(x + y)$ | es una operación decreciente. |

Teorema 1

Si \tilde{M} y \tilde{N} son números difusos cuyas funciones de pertenencia son continuas y subjetivas de \mathfrak{R} para $[0, 1]$ y $*$ es una operación binaria continua creciente (*decreciente*), entonces $\tilde{M} \otimes \tilde{N}$ es un número difuso cuya función de pertenencia es continua y subjetiva de \mathfrak{R} para $[0, 1]$.

Dubois y Prade (1980a) presentaron procedimientos para determinar las funciones de pertenencia $\mu_{\tilde{M} \otimes \tilde{N}}$ en base de $\mu_{\tilde{M}}$ y $\mu_{\tilde{N}}$.

Teorema 2

Si $\tilde{M}, \tilde{N} \in F(\mathfrak{R})$ con $\mu_{\tilde{N}}(x)$ y $\mu_{\tilde{M}}(x)$ funciones de pertenencia continuas, entonces por la aplicación de el principio de extensión para la operación binaria $*$: $\mathfrak{R} \otimes \mathfrak{R} \rightarrow \mathfrak{R}$ la función de pertenencia de el número difuso $\tilde{M} \otimes \tilde{N}$ N esta dado por:

$$\mu_{\tilde{M} \otimes \tilde{N}}(z) = \sup_{z = x \times y} \min \left\{ \mu_{\tilde{M}}(x), \mu_{\tilde{N}}(y) \right\}$$

Observación 1

1. Para cualquier operación conmutativa * la operación extendida \otimes es también conmutativa .
2. Para cualquier operación asociativa * la operación extendida \otimes es también asociativa.

4. RELACION DIFUSA Y GRÁFICAS DIFUSAS

4.1 RELACIÓN DIFUSA EN CONJUNTOS Y CONJUNTOS DIFUSOS

Las relaciones difusas y subconjuntos difusos de $X \times Y$, esto es, mapeos de $X \rightarrow Y$, han sido estudiados por un número de autores, en particular por Zadeh (1965_1971), Kaufmann (1975) y Rosenfeld (1975). Aplicaciones de relaciones difusas son muy difundidas e importantes. Consideraremos algunas de ellas e indicaremos sus usos más posibles. Ejemplificaremos solo relaciones binarias. Una generalización de relación *n-aria* es aclarada más adelante.

Definición 15

Suponiendo $X, Y \subseteq \mathfrak{R}$ son conjuntos universales, entonces

$$\tilde{R} = \left\{ \left((x, y), \mu_{\tilde{R}}(x, y) \right) \mid (x, y) \in X \times Y \right\}$$

es llamada una *relación difusa* en $X \times Y$.

Ejemplo

Suponiendo $X = Y = \mathfrak{R}$ y $\tilde{R} :=$ "es considerablemente grande". La función de pertenencia de una relación difusa, la cual es, por supuesto, un conjunto difuso en $X \times Y$ puede entonces ser:

$$\mu_{\tilde{R}}(x, y) = \begin{cases} 0 & \text{para } x \leq y \\ \frac{(x-y)}{10y} & \text{para } y < x \leq 11y \\ 1 & \text{para } x > 11y \end{cases}$$

Una función de pertenencia diferente para esta relación sería:

$$\mu_{\tilde{R}}(x, y) = \begin{cases} 0 & \text{para } x \leq y \\ \left(1 + (y - x)^{-2}\right)^{-1} & \text{para } x > y \end{cases}$$

Para soporte discreto; las relaciones difusas pueden también ser definidas por matrices.

En la *definición 15* fue asumido que $\mu_{\tilde{R}}$ fue un mapeo de $X \times Y$ hacia $[0, 1]$, esto es, asignando para cada par (x, y) un grado de pertenencia en el intervalo unitario. En algunas instancias, tales como en teoría de gráficas, esto es útil para considerar las relaciones difusas de este mapa de conjuntos difusos contenidos en el conjunto universal dentro de un intervalo unitario. Entonces la *definición 15* tiene que ser generalizada ¹⁴

Definición 16

Suponiendo $X, Y \subseteq \mathfrak{X}$ y

$$\begin{aligned} \tilde{A} &= \{(x, \mu_{\tilde{A}}(x)) \mid x \in X\}, \\ \tilde{B} &= \{(y, \mu_{\tilde{B}}(y)) \mid y \in Y\} \end{aligned} \quad \text{dos conjuntos difusos}$$

Entonces $\tilde{R} = \{(x, y, \mu_{\tilde{R}}(x, y)) \mid (x, y) \in X \times Y\}$ es una *relación difusa* en \tilde{A} y \tilde{B} si

$$\mu_{\tilde{R}}(x, y) \leq \mu_{\tilde{A}}(x), \quad \forall (x, y) \in X \times Y$$

y

$$\mu_{\tilde{R}}(x, y) \leq \mu_{\tilde{B}}(y), \quad \forall (x, y) \in X \times Y.$$

Esta definición deberá particularmente de ser útil cuando se definan gráficas difusas: dejando los elementos de la relación difusa de la *definición 16*, ser nodos de la gráfica difusa la cual es representada por una relación difusa. Los grados de pertenencia de los elementos del conjunto difuso definen "la fuerza" de el flujo en los límites de la gráfica, mientras los grados de pertenencia de los pares correspondientes en la relación son el "flujo" o "capacidades" de la gráfica.

El requerimiento adicional de la *definición 16* ($\mu_{\tilde{R}}(x, y) \leq \min\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(y)\}$) entonces asegura que los "flujos" en los límites de la gráfica, nunca excederá los flujos en los nodos respectivos (= par de límites).

La relación difusa es obviamente un conjunto difuso de espacios producto. Por lo tanto la teoría de conjuntos y las operaciones algebraicas pueden ser definidas por esta en analogía a definiciones utilizando el principio de extensión.

¹⁴ [Rosenfeld 1975].

Definición 17

Suponiendo que \tilde{R} y \tilde{Z} son dos relaciones difusas en el mismo producto espacio. La unión/intersección \tilde{R} y \tilde{Z} es entonces definida por:

$$\mu_{\tilde{R} \cup \tilde{Z}}(x, y) = \max\{\mu_{\tilde{R}}(x, y), \mu_{\tilde{Z}}(x, y)\}, \quad (x, y) \in X \times Y$$

$$\mu_{\tilde{R} \cap \tilde{Z}}(x, y) = \min\{\mu_{\tilde{R}}(x, y), \mu_{\tilde{Z}}(x, y)\}, \quad (x, y) \in X \times Y$$

Hasta aquí "min" y "max" han sido usados para definir intersección y unión. Puesto que la relación difusa es un conjunto difuso, las operaciones también pueden ser definidas usando las definiciones alternativas. Algunos conceptos adicionales, tales como la proyección y la extensión cilíndrica de relaciones difusas.

Definición 18

Suponiendo $\tilde{R} = \{(x, y), \mu_{\tilde{R}}(x, y) | (x, y) \in X \times Y\}$ es una relación binaria difusa. La *primera proyección* de \tilde{R} es entonces definida como

$$\tilde{R}^{(1)} = \{(x, \max_y \mu_{\tilde{R}}(x, y)) | (x, y) \in X \times Y\}$$

la *segunda proyección* es definida como

$$\tilde{R}^{(2)} = \{(y, \max_x \mu_{\tilde{R}}(x, y)) | (x, y) \in X \times Y\}$$

y la *proyección total* como

$$\tilde{R}^{(T)} = \max_x \max_y \{\mu_{\tilde{R}}(x, y) | (x, y) \in X \times Y\}$$

La relación resultante de aplicar una operación de proyección a otra relación es también llamada "sombra"¹⁵. Dejemos ahora una consideración más general de espacio: $X = X_1 \times \dots \times X_n$ y suponiendo que \tilde{R}_q es una proyección en $X_{i_1} \times \dots \times X_{i_k}$ donde (i_1, \dots, i_k) es una subsecuencia de $(1, \dots, n)$. Es obvio que distintas relaciones difusas en el mismo universo pueden tener la misma proyección. Sin embargo, debe de ser una relación más grande definida únicamente $\tilde{R}_{qL}(X_1, \dots, X_n)$ con $\mu_{\tilde{R}_{qL}}(x_{i_1}, \dots, x_{i_k})$ por cada proyección. Esta relación más grande es llamada la *extensión cilíndrica de la relación proyección*.

¹⁵ [Zadeh 1973a].

Definición 19

$\tilde{R}_{qL} \subseteq X$ es la relación más grande en la proyección X en la cual es \tilde{R}_{qL} . \tilde{R}_{qL} es entonces llamada la *extensión cilíndrica* de \tilde{R}_q y \tilde{R}_q es la base de \tilde{R}_{qL} .

Definición 20

Suponiendo que \tilde{R} es una relación difusa en $X = X_1 \times \dots \times X_n$ y \tilde{R}_1 y \tilde{R}_2 son dos proyecciones difusas en $X_1 \times \dots \times X_r$ y $X_{s+1} \times \dots \times X_n$ respectivamente, con $s \leq r+1$ y \tilde{R}_{1L} , \tilde{R}_{2L} sus respectivas extensiones cilíndricas.

La unión de \tilde{R}_1 y \tilde{R}_2 esta definida como $\tilde{R}_{1L} \cup \tilde{R}_{2L}$ y su intersección como: $\tilde{R}_{1L} \cap \tilde{R}_{2L}$.

4.1.1 COMPOSICIÓN DE LAS RELACIONES DIFUSAS

La relación difusa en diferentes espacios producto puede ser combinada una u otra por la operación "composición". Diferentes versiones de "composición" que han sido sugeridas, difieren en sus resultados y también con respecto a sus propiedades matemáticas. La composición max-min ha llegado a ser la más conocida y la más frecuentemente usada. Sin embargo, otras composiciones llamadas *max-product* (producto-máximo) o *max-average* (promedio-máximo) encabezan resultados que son más atractivos.

Definición 21

La composición max-min: Suponiendo $\tilde{R}_1(x, y)$, $(x, y) \in X \times Y$ y $\tilde{R}_2(y, z)$, $(y, z) \in Y \times Z$ son dos relaciones difusas. La composición max-min \tilde{R}_1 max-min \tilde{R}_2 es entonces un conjunto difuso.

$$\tilde{R}_1 \circ \tilde{R}_2 = \left\{ \left[(x, z), \max_y \left\{ \min \left\{ \mu_{\tilde{R}_1}(x, y), \mu_{\tilde{R}_2}(y, z) \right\} \right\} \right] \mid x \in X, y \in Y, z \in Z \right\}$$

$\mu_{\tilde{R}_1 \circ \tilde{R}_2}$ es otra vez una función de pertenencia de una relación difusa en un conjunto difuso.

Una definición más general de composición es la "composición max-*".

Definición 22

Suponiendo que \tilde{R}_1 y \tilde{R}_2 son definidas como en la *definición 21*. La *composición max-** de \tilde{R}_1 y \tilde{R}_2 son entonces definidas como:

$$\tilde{R}_1 \circ \tilde{R}_2 = \left\{ \left[(x,z), \max_y \left\{ \left\{ \mu_{\tilde{R}_1}(x,y), \mu_{\tilde{R}_2}(y,z) \right\} \right\} \right] x \in X, y \in Y, z \in Z \right\}$$

Si $*$ es una operación asociativa y es monotónicamente no decreciente en cada argumento, entonces la composición max- $*$ corresponde esencialmente a la composición max-min. Dos casos especiales de la composición max- $*$ son propuestos en la siguiente definición.

Definición 23

Suponiendo \tilde{R}_1 y \tilde{R}_2 son definidas como en la definición 21. La composición max_prod $\tilde{R}_1 \circ \tilde{R}_2$ y la composición max-av de $\tilde{R}_1 \circ \tilde{R}_2$ es entonces definida como sigue ¹⁶

$$\tilde{R}_1 \circ_{\text{prod}} \tilde{R}_2 = \left\{ \left[(x,z), \max_y \left\{ \left\{ \mu_{\tilde{R}_1}(x,y) * \mu_{\tilde{R}_2}(y,z) \right\} \right\} \right] x \in X, y \in Y, z \in Z \right\}$$

$$\tilde{R}_1 \circ_{\text{av}} \tilde{R}_2 = \left\{ \left[(x,z), 1/2 * \max \left\{ \left\{ \mu_{\tilde{R}_1}(x,y) + \mu_{\tilde{R}_2}(y,z) \right\} \right\} \right] x \in X, y \in Y, z \in Z \right\}$$

4.1.2 PROPIEDADES DE LA COMPOSICIÓN MIN-MAX

4.1.2.1 ASOCIATIVIDAD

La composición max-min es asociativa, esto es,

$$(\tilde{R}_3 \circ \tilde{R}_2) \circ \tilde{R}_1 = \tilde{R}_3 \circ (\tilde{R}_2 \circ \tilde{R}_1).$$

Por lo tanto $\tilde{R}_1 \circ \tilde{R}_1 \circ \tilde{R}_1 = \tilde{R}_1$ ³ la tercera potencia de una relación difusa esta definida.

4.1.2.2 REFLEXIBILIDAD

Definición 24

Suponiendo que \tilde{R} es una relación difusa en $X \times X$.

1. \tilde{R} es llamada reflexiva (Zadeh 1971) si

$$\mu_{\tilde{R}}(x, x) = 1 \quad \forall x \in X$$

¹⁶ [Rosenfeld 1975]

4.1.2.4 TRANSITIVIDAD

Definición 27

Una relación difusa (max-min) R es llamada *transitiva* si

$$\tilde{R} \circ \tilde{R} \subseteq \tilde{R}$$

Observación 3

Las combinaciones de las propiedades anteriores dan algunos resultados interesantes para las composiciones max-min:

1. Si \tilde{R} es simétrica y transitiva, entonces $\mu_{\tilde{R}}(x, y) \leq \mu_{\tilde{R}}(x, x)$ para toda $x, y \in X$.
2. Si \tilde{R} es reflexiva y transitiva, entonces $\tilde{R} \circ \tilde{R} = \tilde{R}$.
3. Si \tilde{R}_1 y \tilde{R}_2 son transitivas y $\tilde{R}_1 \circ \tilde{R}_2 = \tilde{R}_2 \circ \tilde{R}_1$, entonces $\tilde{R}_1 \circ \tilde{R}_2$ es transitiva.

Las propiedades mencionadas en las observaciones 2 y 3 son para la composición max-min. Para la *composición max-prod*, la propiedad 3 de la observación 3 es también verdadera pero no las propiedades 1 y 3 de la observación 2 o la propiedad 1 de la observación 3. Para la *composición max-av*, las propiedades 1 y 3 de la observación 2 así como las propiedades de la observación 3. La propiedad 5 de la observación 2 es cierta para cualquier operador conmutativo.

4.2 GRÁFICAS DIFUSAS

Ya fue mencionado en las *definiciones 16 y 17* que una relación difusa también puede ser interpretada como una definición de una gráfica difusa. Para continuar en la línea con la terminología de la teoría de gráficas tradicional usaremos la siguiente definición de gráfica difusa.

Definición 28

Suponiendo que E es el conjunto (crisp) de nodos. Una *gráfica difusa* esta entonces definida por

$$\tilde{G}(x_i, x_j) = \{((x_i, x_j), \mu_G(x_i, x_j)) \mid (x_i, x_j) \in E \times E\}$$

Si \tilde{E} es un conjunto difuso, una gráfica difusa deberá ser definida como una analogía a la *definición 16*.

Enfocaremos nuestra atención en gráficas binarias indirectas finitas, esto es, asumiremos en lo siguiente que la relación difusa representa a una gráfica

simétrica. Los arcos pueden entonces ser considerados como pares desordenados de nodos. En analogía con la teoría de gráficas tradicional, los conceptos teóricos de las gráficas difusas pueden ser definidos.

Definición 29

$\tilde{H}(x_1, x_2)$ es una *subgráfica difusa* de $\tilde{G}(x_1, x_2)$ si

$$\mu_{\tilde{H}}(x_i, x_j) \leq \mu_{\tilde{G}}(x_i, x_j) \quad \forall (x_i, x_j) \in N \times N$$

$\tilde{H}(x_i, x_j)$ *abarca* la gráfica $\tilde{G}(x_i, x_j)$ si el conjunto de nodos de $\tilde{H}(x_i, x_j)$ y $\tilde{G}(x_i, x_j)$ son iguales, esto es, si difieren solo en su peso de arco.

Definición 30

Una *ruta* en una gráfica difusa $\tilde{G}(x_i, x_j)$ es una secuencia de distintos nodos, x_0, x_1, \dots, x_n , tal que para todo (x_i, x_{i+1}) $\mu_{\tilde{G}}(x_i, x_{i+1}) > 0$. La *fuerza* de la ruta es $\min\{\mu_{\tilde{G}}(x_i, x_{i+1})\}$ para todos los nodos contenidos en la ruta. La *longitud* de una ruta $n > 0$ es el número de nodos contenidos en la ruta. Cada par de nodos (x_i, x_{i+1}) $\mu(x_i, x_{i+1}) > 0$ es llamado *límite* (arco) de la gráfica. Una ruta es llamada un *ciclo* de $x_0 = x_n$ y $n \geq 3$.

Sería claro llamar a la longitud de la ruta más corta entre dos nodos de la gráfica la distancia entre estos nodos. Sin embargo, esta definición tiene algunas desventajas. Es por lo tanto más razonable definir la distancia entre dos nodos como sigue¹⁹

Definición 31

La *longitud- μ* de una ruta $p = x_0, \dots, x_n$ es igual a

$$L(p) = \sum_{i=1}^n \frac{1}{\mu(x_i, x_{i+1})}$$

La *distancia- μ* $d(x_i, x_j)$ entre dos nodos x_i, x_j es la *longitud- μ* más pequeña de cualquier ruta x_i a x_j , $x_i, x_j \in \tilde{G}$.

Definición 32

Dos nodos que son unidos por una ruta son llamados *nodos conectores*.
Conectividad es una relación que también es transitiva.

¹⁹ [Rosenfeld, pag. 58]

Definición 33

Una gráfica difusa es un *bosque* si no tiene ciclos, esto es, un gráfica difusa acíclica. Si el bosque difuso esta conectado es llamado un *árbol* (Una gráfica difusa que es un bosque, debe de ser distinguida de una gráfica difusa que es un bosque difuso).

5. ANÁLISIS DIFUSO**5.1 FUNCIONES DIFUSAS EN CONJUNTOS DIFUSOS**

Una función difusa es una generalización del concepto de función clásica. Una función clásica f es un mapeo (correspondencia) del dominio D de la definición de la función dentro de un espacio S ; $f(D) \subseteq S$ es llamada rango de f . Diferentes características del concepto clásico de una función pueden ser considerados difusos más que determinísticos. Por lo tanto diferentes "grados" de fuzzificación de la noción clásica de un función son concebibles.

1. Puede haber un mapeo determinístico de un conjunto difuso el cual acarrea a lo largo de la fuzzificación del dominio y por lo tanto generar un conjunto difuso. La imagen de un argumento determinístico puede volver a ser determinístico.
2. El propio mapeo puede ser difuso, por eso enturbia la imagen de un argumento difuso. Llamaremos esto *función difusa*. Estos son llamados "*funciones de fuzzificación*"²⁰
3. Funciones ordinarias pueden tener propiedades difusas o ser restringidas por restricciones difusas.

Naturalmente, tipos híbridos pueden ser considerados. Sin embargo, enfocaremos nuestras consideraciones solo en usar casos puros.

Definición 34²¹

Una *función clásica* $f: X \rightarrow Y$ mapea de un dominio *difuso* \tilde{A} en X a un rango difuso \tilde{B} en Y si

$$\forall x \in X, \mu_{\tilde{B}}(f(x)) \geq \mu_{\tilde{A}}(x)$$

Dada una función clásica $f: X \rightarrow Y$ y un dominio difuso \tilde{A} en X el principio de extensión produce el rango difuso \tilde{B} con la función de pertenencia:

$$\mu_{\tilde{B}}(y) = \sup_{x \in f^{-1}(y)} \mu_{\tilde{A}}(x)$$

²⁰ Dubois y Prade (1980a, pag. 106).

²¹ [Dubois y Prade 1980a; Negoita y Ralescu 1975]

De aquí f es una función correspondiente a la *definición 34*.

Definición 35

Suponiendo que X y Y son universos y $\tilde{P}(Y)$ el conjunto de todos los conjuntos difusos en Y (conjunto potencia).

$$\begin{aligned} \tilde{f}: X \rightarrow \tilde{P}(Y) & \text{ es un mapeo} \\ \tilde{f} & \text{ es una función difusa si} \\ \mu_{\tilde{f}(x)}(y) &= \mu_{\tilde{R}}(x, y), \forall (x, y) \in X \times Y \end{aligned}$$

donde $\mu_{\tilde{R}}(x, y)$ es una función de pertenencia de una relación difusa.

5.2 EXTREMOS DE FUNCIONES DIFUSAS

Tradicionalmente un extremo (máximo o mínimo) de un función determinística f sobre un dominio dado D es conseguido en un punto preciso x_0 . Si la función f pasa a ser la función objetiva del modelo de decisión, posiblemente forzada por un conjunto de otras funciones, entonces el punto x_0 en el cual la función consigue el óptimo es generalmente llamada la decisión óptima, esto es, en la teoría clásica hay casi una relación única entre el extremo de la función objetiva y la noción de la decisión óptima de un modelo de decisión.

En modelos en los cuales la fuzzificación está envuelta, esta relación única no existe mucho. El extremo de una función, el óptimo de un modelo de decisión, la "decisión óptima" es frecuentemente considerada para ser un conjunto determinístico, D_m , el cual contiene esos elementos del conjunto difuso "decisión" consiguiendo el máximo grado de pertenencia²².

La noción de una "decisión óptima" como se menciona arriba corresponde a el concepto de un "conjunto maximizado" cuando se consideran funciones en general.

Definición 36

Suponiendo que f es una función X valuada en los reales. Supongamos que f está limitado de abajo por $\inf(f)$ y de arriba por $\sup(f)$.

El conjunto difuso $\tilde{M} = \{(x, \mu_{\tilde{M}}(x)), x \in X$ con:

²² [Bellman y Zadeh 1970, pag. 150].

$$\mu_{\tilde{M}}(x) = \frac{f(x) - \inf(f)}{\sup(f) - \inf(f)}$$

es entonces llamado el *conjunto maximizado*.

En la *definición 36*, f es una función determinística valuada en los reales, similar a la función de pertenencia del conjunto difuso "decisión" y el conjunto maximizado proporciona información acerca de la vecindad del extremo de la función f , del cual el dominio también es determinístico.

Consideramos ahora los extremos de funciones difusas de acuerdo con la *definición 35*, los cuales son definidos sobre un dominio determinístico: a partir de entonces la función difusa $f(x)$ es un conjunto difuso, digamos en \mathfrak{X} , el máximo generalmente no será un punto en \mathfrak{X} pero también es un conjunto difuso, al cual llamaremos "máximo difuso de $f(x)$ ". Una aproximación más clara es definir una operación max extendida en analogía a las otras operaciones extendidas definidas en el tema 3. Max y min son operaciones crecientes en \mathfrak{X} . El máximo o mínimo, respectivamente, de n números difusos, denotados por $\text{m}\tilde{\text{a}}\text{x}(\tilde{M}_1, \dots, \tilde{M}_n)$ y $\text{m}\tilde{\text{i}}\text{n}(\tilde{M}_1, \dots, \tilde{M}_n)$ son de nuevo un número difuso.

Definición 37

Suponiendo que $\tilde{f}(x)$ es una función difusa de X a \mathfrak{X} , definida sobre un dominio determinístico y finito D . El *máximo difuso* de $f(x)$ es entonces definido como:

$$\tilde{M} = \text{m}\tilde{\text{a}}\text{x}_{x \in D} \tilde{f}(x) = \left\{ \left(\sup \tilde{f}(x), \mu_{\tilde{M}}(x) \right) \mid x \in D \right\}$$

para $|D| = n$ la función de pertenencia de $\text{m}\tilde{\text{a}}\text{x} \tilde{f}(x)$ esta dada por

$$\mu_{\tilde{M}}(x) = \min_{j=1, \dots, n} \mu_{\tilde{f}(x_j)}(f(x_j)), \quad f(x) \in D$$

6. TEORÍA DE POSIBILIDAD, TEORÍA DE PROBABILIDAD Y TEORÍA DE CONJUNTOS DIFUSOS

6.1 TEORÍA DE POSIBILIDAD

6.1.1 CONJUNTOS DIFUSOS Y DISTRIBUCIONES DE POSIBILIDAD

La teoría de posibilidad se enfoca primariamente en imprecisión, la cual es implícita en los lenguajes naturales y es asumida para ser más bien "posibilística"

que probabilística. Por lo tanto el término *variable* es muy frecuentemente usado en más de un sentido que en uno estrictamente matemático. Esto es una razón de porque la terminología y el simbolismo de la teoría de posibilidad difiere en algo respecto de la teoría de conjuntos difusos. Para facilitar el estudio de la teoría de posibilidad, usaremos la terminología posibilística común pero siempre mostraremos la correspondencia de la teoría de conjuntos difusos.

Supondremos, por ejemplo, que queremos considerar la proposición "X es \tilde{F} ", donde X es el nombre de un objeto, una variable o una proposición. Por ejemplo, en "X es un entero pequeño", X es el nombre de una variable. En "John es joven", John es el nombre de un objeto. \tilde{F} (i.e. "entero pequeño" o "joven") es un conjunto difuso caracterizado por su función de pertenencia $\mu_{\tilde{F}}$.

Uno de los conceptos centrales de la teoría de posibilidad es esta, de una distribución de posibilidad (como opuesta a una distribución de probabilidad). Para definir una distribución de posibilidad, es conveniente primero introducir la noción de una restricción difusa. Para visualizar un restricción difusa el lector imaginará una maleta elástica, que actuará en lo posible del volumen de su contenido como un fuerza. Para una maleta dura, el volumen es un número determinístico. Para una maleta suave, el volumen de su contenido depende de un cierto grado de fuerza que es usada para estirlo. La variable en este caso será el volumen de maleta; los valores que esta variable puede asumir son $u \in U$, y el grado para el cual la variable (X) puede asumir diferentes valores de u es expresado por $\mu_{\tilde{F}}(u)$. Zadeh define estas relaciones como sigue:

Definición 38

Supongamos que \tilde{F} es un conjunto difuso del universo U caracterizado por una función de pertenencia $\mu_{\tilde{F}}(u)$. \tilde{F} es una *restricción difusa* en la variable X si \tilde{F} actúa como una fuerza elástica en los valores que pueden ser asignados a X, en este sentido la asignación de valores u a X tiene la forma

$$X = u: \mu_{\tilde{F}}(u)$$

$\mu_{\tilde{F}}(u)$ es el grado en el cual la fuerza representada por \tilde{F} está satisfecha cuando u es asignada a X. Equivalentemente, esto implica que $1 - \mu_{\tilde{F}}(u)$ es el grado en el cual la fuerza está estirada para permitir la asignación de valores u a la variable X.

En todo caso un conjunto difuso puede ser considerado o no como una restricción difusa, depende obviamente en su interpretación: este es el caso sólo si actúa como una restricción en los valores de una variable, que podría tomar la forma de un término lingüístico o una variable clásica.

Suponiendo que $\tilde{R}(X)$ es una restricción difusa asociada con X tal como está definida en la *definición 38*. Entonces $\tilde{R}(X) = \tilde{F}$ es llamada una *ecuación de asignación relacional* que asigna el conjunto difuso \tilde{F} a la restricción difusa $\tilde{R}(X)$. Ahora asumimos que $A(X)$ es un atributo implícito de la variable X . Por ejemplo, $A(X)$ = "edad de Jack" y \tilde{F} es el conjunto difuso "joven". La proposición "Jack es joven" (o mejor aún "la edad de Jack es joven") puede entonces ser expresada como:

$$\tilde{R}(A(X)) = \tilde{F}$$

Zadeh (1978) relaciona el concepto de una restricción difusa a el de una distribución de posibilidad como sigue:

Considerar una edad numérica, digamos $u = 28$, del cual el grado de pertenencia en el conjunto difuso "joven" es aproximadamente 0.7. Interpretaremos primero 0.7 como el grado de compatibilidad de 28 con el etiquetado con joven. Entonces postularemos que la proposición "John es joven" convierte el significado de 0.7 del grado de compatibilidad de 28 con joven hacia el grado de posibilidad de que John es 28 determinado por la proposición "John es joven". En resumen, la compatibilidad de un valor u con joven llega a ser convertido en la posibilidad de que el valor de u dado "John es joven".

El concepto de un distribución de posibilidad puede ahora ser definido como sigue:

Definición 39²³

Suponiendo que \tilde{F} es un conjunto difuso en un universo de discurso U el cual es caracterizado por su función de pertenencia $\mu_{\tilde{F}}(u)$, que es interpretado como la compatibilidad de $u \in U$ con el concepto etiquetado llamado \tilde{F} .

Suponiendo que X es una variable que toma valores en U y \tilde{F} actúa como una restricción difusa, $\tilde{R}(X)$, asociada con X . Entonces la proposición " X es \tilde{F} ", la cual se traslada a $\tilde{R}(X) = \tilde{F}$ asociada a una *distribución de posibilidad*, π_x , con X que es postulada a ser igual a $\tilde{R}(X)$.

La función de distribución de posibilidad $\pi_x(u)$, caracterizando la distribución de posibilidad π_x es definida para ser numéricamente igual a la función de pertenencia $\mu_{\tilde{F}}(u)$ de \tilde{F} , esto es,

$$\pi_x \hat{=} \mu_{\tilde{F}}$$

²³ [Zadeh 1978, pág. 6].

El símbolo $\hat{=}$ siempre representará "denotar" ó "es definido para ser". Para permanecer en la misma línea con el símbolo común de la teoría de posibilidad denotaremos una distribución de posibilidad con π_x más bien con $\tilde{\pi}_x$, aunque esto es un conjunto difuso.

6.1.2 MEDIDAS POSIBLES Y NECESARIAS

En el tema 2.1 una medida de posibilidad ya fue definida (*definición 7*) para el caso en que A es un conjunto determinístico. Si A es un conjunto difuso una definición más general de medida de posibilidad es dada como ²⁴

Definición 40

Suponiendo que \tilde{A} es un conjunto difuso en el universo U es π_x una distribución de posibilidad asociada con la variable X que toma valores en U . La *medida de posibilidad* $\pi_x(\tilde{A})$, de \tilde{A} es entonces definida por

$$\begin{aligned} \text{poss} \{X \text{ es } \tilde{A}\} &\hat{=} \pi(\tilde{A}) \\ &\hat{=} \sup_{u \in U} \min \{ \mu_{\tilde{A}}(u), \pi_x(u) \} \end{aligned}$$

Similar a la teoría de probabilidad existen también posibilidades condicionales. Como una distribución de posibilidad condicional se definió lo siguiente ²⁵

Definición 41

Suponiendo que X y Y son variables en los universos U y V , respectivamente. La *distribución de posibilidad condicional* de Y dado X es entonces inducida por una proposición de la forma "si X es \tilde{F} entonces Y es \tilde{G} " y es denotada por $\pi_{(Y|X)}(v/u)$.

Proposición 1

Suponiendo que $\pi_{(Y|X)}$ es la función de distribución de posibilidad condicional de X y Y , respectivamente. La unión de la función de distribución de posibilidad de Y dado X , $\mu_{(Y|X)}$ está dada entonces por

$$\mu_{(Y|X)}(u,v) = \min\{\pi_x(u), \pi_{(Y|X)}(v/u)\}$$

²⁴ [Zadeh 1978 pag. 9].

²⁵ [Zadeh 1981b, pag.81].

Al parecer no quedo aún completamente establecida la cuestión de como se derivan las funciones de distribución de posibilidad condicional de la unión de la función de distribución de posibilidad.

Las medidas difusas como se definieron en la *definición 7* expresan el grado para el cual un subconjunto de un universo Ω es cierto, o un evento es posible. De aquí tenemos

$$g(\phi) = 0 \text{ y } g(\Omega) = 1$$

como consecuencia de la condición 2 de la *definición 7*, esto es,

$$A \subseteq B \Rightarrow g(A) \leq g(B)$$

tenemos

$$g(A \cup B) \geq \max(g(A), g(B)) \quad \text{y} \quad (6.1)$$

$$g(A \cap B) \geq \min(g(A), g(B)) \quad \text{para } A, B \subseteq \Omega \quad (6.2)$$

las medidas de posibilidad son definidas para los casos limites:

$$\pi(A \cup B) = \max(\pi(A), \pi(B)) \quad (6.3)$$

$$\pi(A \cap B) = \min(\pi(A), \pi(B)) \quad (6.4)$$

Si CA es el complemento de A en Ω , entonces

$$\pi(A \cup CA) = \max(\pi(A), \pi(CA)) = 1 \quad (6.5)$$

lo cual expresa que el factor en cada A ó CA es completamente posible.

En la teoría de posibilidad otra medida adicional esta definida, que usa la relación conjuntiva y, en este sentido, es doble para la medida de posibilidad:

$$N(A \cup B) = \min(N(A), N(B)) \quad (6.6)$$

N es llamada la medida necesidad, $N(A) = 1$ indica que A es necesariamente verdadera (A es segura). La doble relación de posibilidad y necesidad requiere que:

$$\pi(A) = 1 - N(CA); \quad \forall A \subseteq \Omega \quad (6.7)$$

Las medidas de necesidad satisfacen la condición

$$\min(N(A), N(\bar{A})) = 0 \quad (6.8)$$

La relación entre las medidas de posibilidad y las medidas de necesidad satisfacen también las siguientes condiciones ²⁶

$$\pi(A) \geq N(A), \quad \forall A \subseteq \Omega \quad (6.9)$$

$$N(A) > 0 \Rightarrow \pi(A) = 1$$

$$\pi(A) < 1 \Rightarrow N(A) = 0 \quad (6.10)$$

aquí Ω asume siempre un valor finito.

6.2 PROBABILIDAD DE EVENTOS DIFUSOS

Por ahora esto tendrá que llegar a ser claro, que la posibilidad no es un sustituto para la probabilidad, sino más bien otro tipo de incertidumbre.

Dejaremos ahora de suponer que un evento no está determinísticamente definido excepto por una distribución de posibilidad (un conjunto difuso) y que estaremos en una situación de incertidumbre estocástica, esto es, que el suceso (descrito difusamente) de este evento no es cierto y queremos expresar la probabilidad de su acontecimiento. Dos vistas en esta probabilidad pueden ser adoptadas:

6.2.1 PROBABILIDAD DE UN EVENTO DIFUSO COMO UN ESCALAR

En la teoría clásica de probabilidad en un evento A , es un miembro de un campo- α a , de subconjuntos de un espacio muestra Ω . Una medida de probabilidad P es una medida normalizada sobre un espacio medible (Ω, a) esto es, P es una función real que asigna a cada A en a una probabilidad $P(A)$ tal que:

1. $P(A) \geq 0 \quad A \in a$
2. $P(\Omega) = 1$
3. Si $A_i \in a, i \in I \subset \mathbb{N}$, par juicioso disjunto, entonces

$$P\left(\bigcup_{i \in I} A_i\right) = \sum_{i \in I} P(A_i)$$

Si Ω es, por ejemplo, un espacio- n euclidiano y a el campo- σ de conjuntos Borel en \mathfrak{R}^n entonces la probabilidad de A puede ser expresada como

$$P(A) = \int_A dP$$

²⁶ [Dubois y Prade 1988].

Si $\mu_A(x)$ denota la función característica de un conjunto determinístico de A y $E_p(\mu_A)$ la esperanza de $\mu_A(x)$ entonces:

$$P(A) = \int_{\mathbb{R}^n} \mu_A(x) dP = E_p(\mu_A).$$

Si $\mu_A(x)$ no denota la función característica de un conjunto determinístico más bien la función de pertenencia de un conjunto difuso, la definición básica de la probabilidad A no cambiaría. Zadeh (1968) por lo tanto definió la probabilidad de un evento difuso \tilde{A} (i.e., un conjunto difuso \tilde{A} con función de pertenencia $\mu_{\tilde{A}}(x)$) como sigue.

Definición 42

Suponiendo que $(\mathfrak{X}^n, \mathcal{a}, P)$ son un espacio probabilidad en cual \mathcal{a} es el campo- σ de conjuntos Borel en \mathfrak{X}^n y P es una medida de probabilidad sobre \mathfrak{X}^n . Entonces un *evento difuso* en \mathfrak{X}^n es un conjunto difuso \tilde{A} en \mathfrak{X}^n cuya función de pertenencia $\mu_{\tilde{A}}(x)$ es medible en Borel.

La *probabilidad de un evento difuso* \tilde{A} es entonces definida por la integral Lebesgue-Stieltjes

$$P(\tilde{A}) = \int_{\mathbb{R}^n} \mu_{\tilde{A}}(x) dP = E(\mu_{\tilde{A}})$$

6.2.2 PROBABILIDAD DE UN EVENTO DIFUSO COMO UN CONJUNTO DIFUSO

En adelante consideraremos los conjuntos con un número finito de elementos. Asumiremos que allí existe una medida de probabilidad P definida en el conjunto de todos los subconjuntos determinísticos del universo X , el conjunto Borel. $P(x_i)$ denotara la probabilidad del elemento $x_i \in X$.

Suponiendo que $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in X\}$ es un conjunto difuso que representa a un evento difuso. El grado de pertenencia del elemento $x_i \in \tilde{A}$ es denotado por $\mu_{\tilde{A}}(x_i)$. Los conjuntos nivel- α o corte- α como ya se definió en la definición 3 serán denotados por \tilde{A}_α .

Yager (1979,1984) sugirió que es totalmente natural definir la probabilidad de un conjunto nivel- α como $P(A_\alpha) = \sum_{x \in A_\alpha} P(x)$. En base a esto la probabilidad de un evento difuso es definida como sigue²⁷

²⁷ [Yager 1984].

Definición 43

Suponiendo que A_α es el conjunto nivel- α de un conjunto difuso \tilde{A} que representa un evento difuso. Entonces la *probabilidad de un evento difuso* \tilde{A} puede ser definida como

$$P_Y(\tilde{A}) = \{(P(A_\alpha)), \alpha\} \mid \alpha \in [0, 1]$$

con la interpretación de "la probabilidad de al menos un grado α de satisfacción de la condición \tilde{A} ".

La subíndice Y de P_Y indica que P_Y es una definición de probabilidad debida a Yager que difiere de la definición de Zadeh la cual se denota por P . Estará muy claro que Yager considera α , que es usada como el grado de pertenencia de las probabilidades $P(A_\alpha)$ en el conjunto difuso $P_Y(A)$, como una clase de nivel de significancia para la probabilidad de un evento difuso.

En base a la comunicación privada con Klement, Yager también sugirió otra definición para la probabilidad de un evento difuso, la cual se deriva como sigue.

Definición 44

La *verdad de la proposición* "la probabilidad de \tilde{A} es al menos w " está definida como el conjunto difuso $P_Y^*(\tilde{A})$ con la función de pertenencia

$$P_Y^*(\tilde{A})(w) = \sup_{\alpha} \{\alpha \mid P(A_\alpha) \geq w\}, \quad w \in [0, 1]$$

El lector debe darse cuenta ahora de que el "indicador" de significancia de la medida de probabilidad es w y ya no α ! El lector debe también estar consciente del hecho que hemos usado la terminología de Yager denotando los valores de la función de pertenencia por $P_Y^*(\tilde{A})(w)$. Esto facilitara la lectura del trabajo de Yager (1984).

Si denotamos el complemento de \tilde{A} por $\tilde{C}\tilde{A} = \{(x, 1 - \mu_{\tilde{A}}(x)) \mid x \in X\}$ y el conjunto nivel- α de $\tilde{C}\tilde{A}$ por $(\tilde{C}\tilde{A})_\alpha$ entonces $P_Y^*(\tilde{C}\tilde{A})(w) = \sup_{\alpha} \{\alpha \mid P(\tilde{C}\tilde{A})_\alpha \geq w\}$, $w \in [0, 1]$ puede ser interpretado como la verdad de la proposición " la probabilidad de no \tilde{A} es al menos w ".

Definiremos $P_Y^*(\tilde{A}) = 1 - P_Y^*(\tilde{C}\tilde{A})$. Si $P_Y^*(\tilde{C}\tilde{A})(w)$ es interpretado como la verdad de la proposición "la probabilidad de \tilde{A} es a lo sumo w ", entonces podemos

argumentar lo siguiente: La combinación “y” de “la probabilidad de \tilde{A} es al menos w ” y “probabilidad de \tilde{A} es a lo sumo w ” podría ser considerada como “la probabilidad de \tilde{A} es exactamente w ”. Si $P_y^*(\tilde{A})$ y $\bar{P}_y^*(\tilde{A})$ son considerados como distribuciones de posibilidad entonces su conjunción es su intersección. De aquí la siguiente definición (Yager 1984):

Definición 45²⁸

Suponiendo $P_y^*(\tilde{A})$ y $\bar{P}_y^*(\tilde{A})$ son definidas como arriba. La distribución de posibilidad asociada con la proposición “la probabilidad de \tilde{A} es exactamente w ” pueden ser definidas como:

$$\bar{P}_y(\tilde{A})(w) = \min \{ P_y^*(\tilde{A})(w), \bar{P}_y^*(\tilde{A})(w) \}$$

6.3 POSIBILIDAD CONTRA PROBABILIDAD

Preguntas concernientes a relación entre la teoría de conjuntos difusos y la teoría de probabilidad son muy frecuentemente hechas, particularmente por los “nuevos” en el área de los conjuntos difusos. Hay probablemente dos grandes razones para esto: De un lado, hay ciertas similitudes formales entre teoría de conjuntos difusos (en particular cuando se usan conjuntos difusos normalizados) y la teoría de probabilidad; por el otro lado, en el pasado las probabilidades han sido el único significado para expresar “incertidumbre”. Esto parece apropiado y provechoso, por lo tanto, hay que aclarar más esta pregunta.

Como ya fue mencionado tal comparación es difícil por la falta de definiciones únicas de los conjuntos difusos. Esta falta de una definición única se debe en parte a la variedad de posibilidades sugeridas para definir matemáticamente los conjuntos difusos. Es también debido a las diferentes clases de fuzzificación que pueden ser modeladas con los conjuntos difusos.

Otro problema es la selección de los aspectos con respecto a cual de estas teorías serán comparadas.

En el punto 6.1 “teoría de posibilidad” fue brevemente explicada. Allí fue mencionado que la teoría de posibilidad es más que la versión max-min de la teoría de conjuntos difusos. Fue también mostrado que las “medidas de incertidumbre” usadas en la teoría de posibilidad son las medidas de posibilidad y las medidas de necesidad, dos medidas que en cierto sentido son dobles a cada una. En comparación la teoría de posibilidad con la teoría de probabilidad, primero consideraran sólo funciones de posibilidad –y medidas (dejando la existencia de medidas dobles)– de la teoría de posibilidad. Al final del tema

²⁸ [Yager 1984].

investigaremos un poco de la relación entre la teoría de posibilidad y la teoría de probabilidad.

Ahora dejaremos el turno a las probabilidades e intentaremos caracterizar y clasificar las nociones de probabilidades disponibles. Tres aspectos deben ser de concierne principal:

1. La expresión lingüística de probabilidad,
2. Los diferentes contextos de información de los diferentes tipos de probabilidades, y
3. La interpretación semántica de probabilidades y sus consecuencias axiomáticas y matemáticas.

Lingüísticamente, podemos distinguir formulaciones de probabilidad explícitas de implícitas. Con respecto a la información contenida, podemos distinguir entre probabilidades clasificatorias (dado E , H es probable), comparativa (dado E , H es más probable que K), parcial (dado E la probabilidad de K está en el intervalo $[a, b]$) y cuantitativa (dado E la probabilidad de H es p) probabilidades.

Finalmente la interpretación de una probabilidad puede variar considerablemente: consideramos dos interpretaciones muy importantes y comunes de probabilidades cuantitativas. Koopman(1940), [Carnap&Stegmüller] interpretan probabilidades esencialmente como grados de confiabilidad de las declaraciones de la lógica dual. Axiomáticamente Koopman deriva un concepto de probabilidad q , que matemáticamente es un aro Booleano.

Kolmogoroff (1950) interpreta las probabilidades "estadísticamente". Él considera un conjunto Ω y una asociación algebra- σ F , los elementos del cual son interpretados como eventos. En base a la teoría de medidas definió una función $P: F \rightarrow [0, 1]$ con las siguientes propiedades:

$$P: I \rightarrow [0, 1] \quad (6.11)$$

$$P(\Omega) = 1 \quad (6.12)$$

$$\forall (X_i) \in F (\forall i, j \in \mathbb{N}: i \neq j \rightarrow X_i \cap X_j = \emptyset) P(\cup_{i \in \mathbb{N}} X_i) = \sum_{i \in \mathbb{N}} P(X_i) \quad (6.13)$$

de estas, la siguientes expresiones pueden ser fácilmente derivadas:

$$X, CX \in F \rightarrow P(CX) = 1 - P(X) \quad (6.14)$$

$$X, Y \in F \rightarrow P(X \cup Y) = P(X) + P(Y) - P(X \cap Y) \quad (6.15)$$

donde CX denota el complemento de X .

La tabla 6-1 ilustra la diferencia entre los conceptos de probabilidades Koopman y Kolmogoroff, tomando en cuenta las diferentes lingüísticas y posibilidades informacionales mencionadas arriba.

Tabla 6-1 "Probabilidades Koopman's vs Kolmogoroff"

<i>Koopman</i>	<i>Kolmogoroff</i>
<i>D, D', H, H'</i> son declaraciones de lógica dual, <i>Q</i> es un número real no-negativo (generalmente $Q \in [0,1]$)	<i>W</i> es un conjunto de eventos, <i>W_i</i> es subconjunto de <i>W</i> .
<i>Clasificatoria</i>	
1. Implícita: <i>D</i> soporta <i>H</i>	1. <i>W_i</i> es un subconjunto lleno de <i>W</i>
2. Explícita: <i>H</i> es probable en la base de <i>D</i>	2. Si al momento de tirar los dados <i>W</i> es no probable <i>W_i</i> es vacío.
<i>Comparativa</i>	
1. Implícita: <i>D</i> soporta <i>H</i> más que <i>D'</i> soporta <i>H'</i>	1. Para el momento de tirar los dados <i>W</i> , <i>W_i</i> es de igual tamaño como <i>W_j</i>
2. Es más probable <i>H</i> dado <i>D</i> a que sea <i>H'</i> dado <i>D'</i> .	2. Al tiempo de lanzar una moneda <i>W</i> , <i>W_i</i> es tan probable como <i>W_j</i>
<i>Cuantitativa</i>	
1. El grado de soporte para <i>H</i> en la base de <i>D</i> es <i>Q</i> .	1. La radio de el número de eventos en <i>W_i</i> y <i>W</i> es <i>Q</i> .
2. La probabilidad para <i>H</i> dado <i>D</i> es <i>Q</i> .	2. La probabilidad de que el resultado de tirar los dados <i>A</i> sea 1 cuando se tiran los dados <i>M</i> al mismo tiempo es <i>Q₁</i> .

Ahora estamos listos para comparar "conjuntos difusos" con "probabilidades" o al menos una versión cierta de teoría de conjuntos difusos con una de teoría de probabilidad. Probabilidades implícitas no son comparables a los conjuntos difusos, desde los modelos de conjunto difuso trataremos particularmente de modelar explícitamente la incertidumbre. Las probabilidades comparativas y parcial son más comparables a las declaraciones probabilísticas usando "variables lingüísticas".

Por lo tanto las versiones mas frecuentemente usadas que mostramos ahora son cuantitativas, probabilidades explicitas Kolmogoroff con posibilidades.

La Tabla 6-2 describe algunas de las principales diferencias matemáticas entre tres áreas que son similares en muchos aspectos.

Tabla 6-2 "Relación entre álgebra Booleana, probabilidades y posibilidades"

	<i>Algebra Booleana</i>	<i>Probabilidades (cuantitativas explicitas)</i>	<i>Posibilidades</i>
Dominio	Conjunto de declaraciones (lógica)	álgebra- σ	Un universo X
Rango de valores de pertenencia	{0,1}	[0,1]	[0,1]
Restricciones especiales		$\sum_{\Omega} p(u) = 1$	
Unión (independiente, no-interactiva)	max	$\sum w_i$	max
Intersección	min	Π	min
Condional igual a salto?	si	no	después
Qué puede ser usado para inferencia?	condicional	condicional salto	o condicional, después salto

Retornaremos ahora al aspecto de "dualidad" de medidas de posibilidad y medidas de necesidad.

Una medida de posibilidad $P(A)$, satisface el axioma de aditividad, esto es, $\forall A, B \subseteq \Omega$ para la cual $A \cap B = \phi$

$$P(A \cup B) = P(A) + P(B) \tag{6.16}$$

Esta medida es monotónica en el sentido de la condición 2 de la *definición 7*.

La ecuación (6.12) es el equivalente probabilístico de (6.1) y (6.2).

Las condiciones (6.5) y (6.8) de la teoría de posibilidad implican

$$N(A) + N(\overline{CA}) \leq 1 \quad (6.17)$$

$$\pi(A) + \pi(\overline{CA}) \geq 1 \quad (6.18)$$

que es menos estricta que la relación equivalente

$$P(A) + P(\overline{CA}) = 1 \quad (6.19)$$

de la teoría de probabilidad.

En ese sentido posibilidad corresponde más a la teoría de evidencia (Shafer 1976) que a la teoría de probabilidad clásica, en la cual las probabilidades de un elemento (un subconjunto) son relacionados únicamente a la probabilidad del elemento contrario (complemento). En la teoría de Shafer, la cual es probabilística por naturaleza, esta relación es también flexible para introducir una "probabilidad superior" y una "probabilidad inferior" que son tan "duales" a cada una de las otras como son posibilidad y necesidad.

En realidad, medidas de posibilidad y necesidad pueden ser consideradas como caso límite de medidas de probabilidad en el sentido de Shafer, esto es,

$$N(A) \leq P(A) \leq \pi(A) \quad \forall A \subseteq \Omega \quad (6.20)$$

Con respecto a las teorías consideradas en este tema podemos concluir: Teoría de conjuntos difusos, teoría de posibilidad y teoría de probabilidad no son substitutos, pero se complementan entre ellos. Mientras que la teoría de conjuntos difusos tiene realmente un número "grados de libertad" con respecto a los operadores intersección y unión, clases de conjuntos difusos (funciones de pertenencia), etc., las últimas dos teorías están bien desarrolladas y únicamente definidas con respecto a la operación y la estructura. La teoría de conjuntos difusos parece ser más adaptable a contextos diferentes. Esto, así también, implica la necesidad de adaptar la teoría a un contexto si uno desea una herramienta apropiada de modelado.

SEGUNDA PARTE:**APLICACIONES DE LA TEORÍA DE CONJUNTOS DIFUSOS****7. LÓGICA DIFUSA Y RAZONAMIENTO APROXIMADO****7.1 VARIABLES LINGÜÍSTICAS**

"En retroceso de precisión a pesar de la abrumadora complejidad, es natural explorar el uso de lo que podría ser llamado variables *lingüísticas*, esto es, variables cuyos valores no son números pero palabras o sentencias en un lenguaje natural o artificial.

La motivación para el uso de palabras o sentencias en vez de números es que las caracterizaciones lingüísticas son, en general, menos específicas que las numéricas"²⁹

Esta cita presenta en pocas palabras la motivación y la justificación para la lógica difusa y el razonamiento aproximado. Otra cita podría ser agregada, la cual es más vieja. El famoso filósofo B. Russell :

Toda la lógica tradicional asume que símbolos precisos son empleados. Esto por consiguiente, no es aplicable a esta vida, pero solo a una existencia celestial imaginada (Russell 1923).

Una de las herramientas básicas de la lógica difusa y del razonamiento aproximado es la noción de una variable lingüística que en 1973 fue llamada *variable de alto orden* en vez de variable difusa y se definió como sigue³⁰

Definición 46

Una *variable lingüística* es caracterizada por una quintuple $(x, T(x), U, G, \tilde{M})$ en la cual x es el nombre de la variable; $T(x)$ (o simplemente T) denota el conjunto termino de x , esto es, es conjunto de nombres de *valores lingüísticos* de x , con cada valor siendo una variable difusa denotada genéricamente por x , y con un rango sobre un universo de discusión U el cual es asociado con la variable base u ; G es una regla sintáctica (la cual tiene usualmente la forma de una gramática) de generación del nombre X , de valores de x ; y M es una *regla semántica* para asociar con cada X su significado, $\tilde{M}(X)$ que es un subconjunto difuso de U . Una X particular, que es un nombre generado por G , es llamado *termino*. Deberá ser notado que la variable base u puede también ser un vector valuado.

²⁹ [Zadeh 1973a].

³⁰ [Zadeh 1973].

Dos variables lingüísticas de particular interés en lógica difusa y en teoría de probabilidad (difusa) son las dos variables lingüísticas "verdad" y "probabilidad". La variable lingüística "probabilidad" es representada ejemplarmente en la figura 3-2.

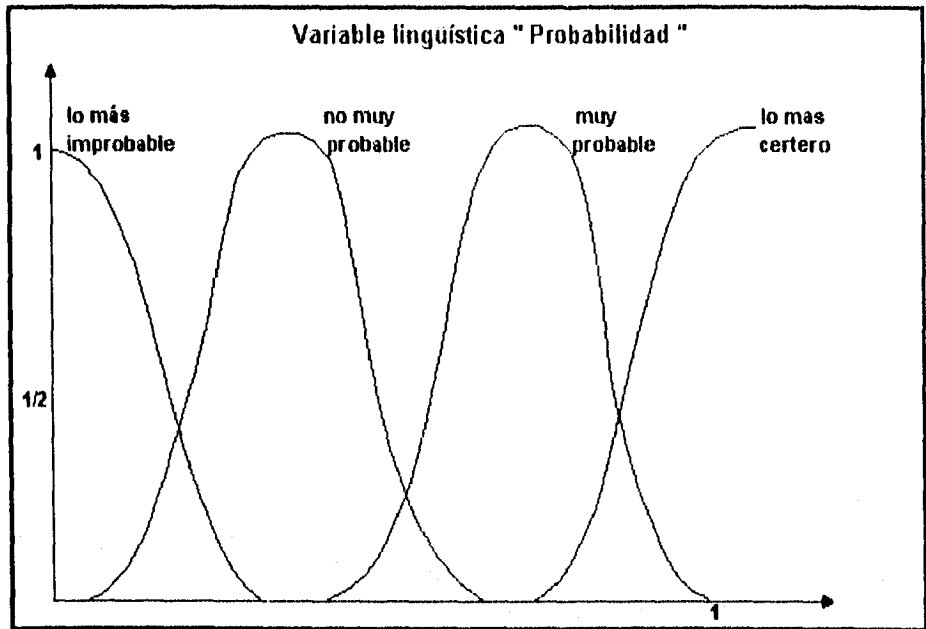


Figura 3-2

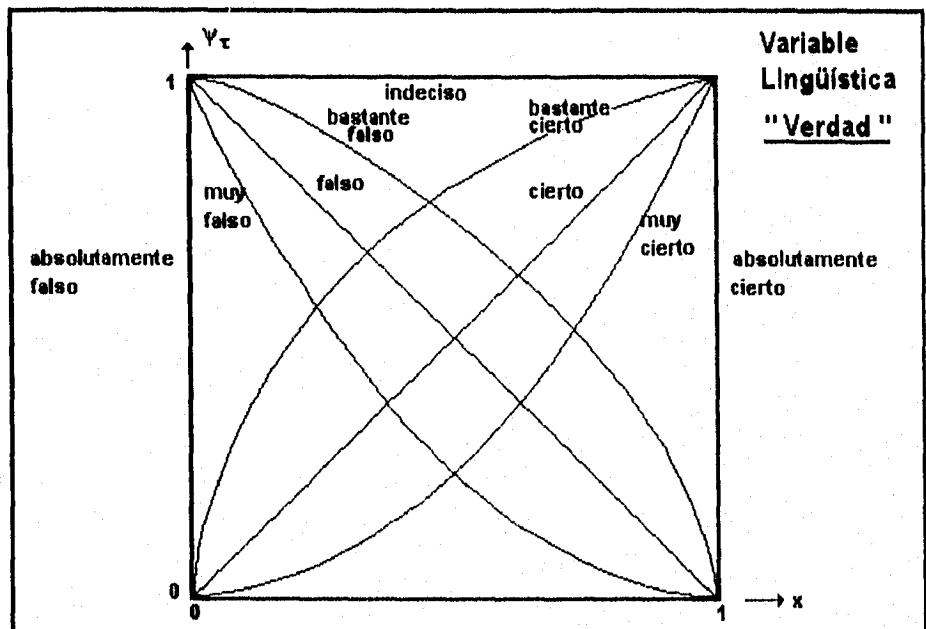


Figura 3-3

El conjunto termino de la variable lingüística "verdad" ha sido diferentemente definida por varios autores. Baldwin (1979) definió algunos de los términos como muestra en la figura 3-3.

Aquí

$$\begin{aligned} \mu_{\text{very true}}(v) &= (\mu_{\text{true}}(v))^2 & v \in [0, 1] \\ \mu_{\text{fairly true}}(v) &= (\mu_{\text{true}}(v))^{1/2} & v \in [0, 1] \end{aligned}$$

y así sucesivamente. Zadeh (1973a) sugiere para el termino *verdad* la función de pertenencia:

$$\mu_{\text{true}}(v) = \begin{cases} 0 & \text{para } 0 \leq v \leq a \\ 2 \cdot \left(\frac{v-a}{1-a}\right)^2 & \text{para } a \leq v \leq \frac{a+1}{2} \\ 1 - 2 \cdot \left(\frac{v-1}{1-a}\right)^2 & \text{para } \frac{a+1}{2} \leq v \leq 1 \end{cases}$$

donde $v = (1+a)/2$ es llamado el *punto de paso*, y $a \in [0, 1]$ es un parámetro que indica el juicio subjetivo del valor mínimo de v en orden a considerar una declaración como "verdadera" del todo.

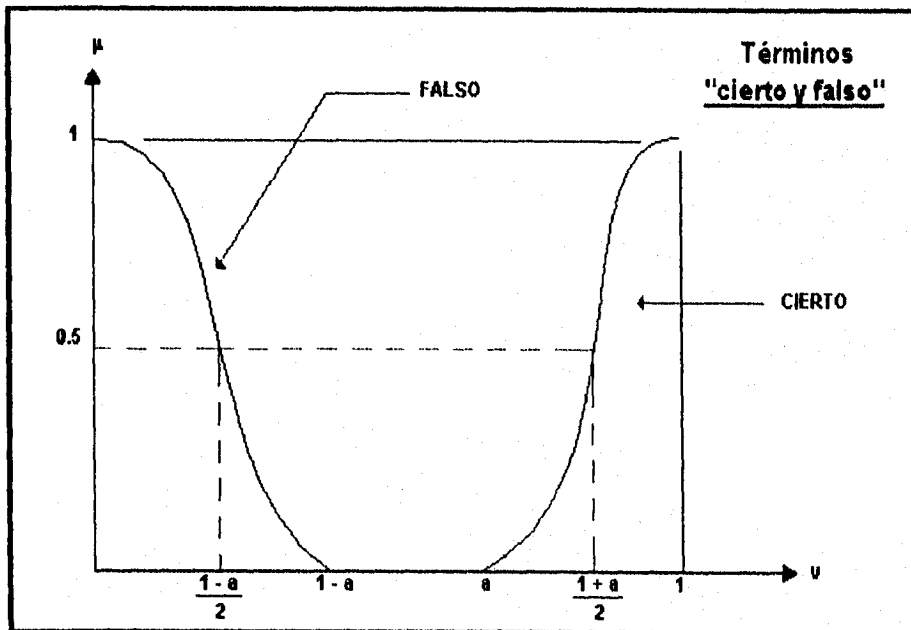


Figura 3-4

La función de pertenencia de "falso" es considerada como la imagen del espejo de "verdad", esto es,

$$\mu_{\text{false}}(v) = \mu_{\text{true}}(1 - v) \quad 0 \leq v \leq 1$$

la figura 3-4 (Zadeh 1973a) muestra los términos *verdad* y *falso*.

Claro que las funciones de pertenencia de verdad y falso, respectivamente, pueden también ser escogidas del universo finito de valores verdad. El conjunto término de la variable lingüística "verdad" es entonces definido como³¹

$$T(\text{verdad}) = \{\text{verdadero, no verdadero, muy verdadero, no muy verdadero, ..., falso, no falso, muy falso, ..., no muy verdadero y no muy falso, ...}\}$$

Los conjuntos difusos (distribución de posibilidad) de estos términos pueden ser esencialmente determinados por el término *verdadero* o el término *falso* por la aplicación de los modificadores mencionados abajo (hileras).

Definición 47

Una variable lingüística x es llamada *estructurada* si el conjunto término $T(x)$ y el significado $\tilde{M}(x)$ pueden ser caracterizados algorítmicamente. Para una variable lingüística estructurada, $\tilde{M}(x)$ y $T(x)$ pueden ser considerados como algoritmos los cuales generan los términos del conjunto término y significados asociados a ellos.

Definición 48

Una *hilera lingüística* o un *modificador* es una operación que modifica el significado de un término o, más generalmente, de un conjunto difuso. Si \tilde{A} es un conjunto difuso entonces el modificador m genera el término (compuesto):

$$\tilde{B} = m(\tilde{A}).$$

Modelos matemáticos frecuentemente usados son:

$$\text{concentración: } \mu_{\text{con}(\tilde{A})}(u) = (\mu_{\tilde{A}}(u))^2$$

$$\text{dilatación: } \mu_{\text{dil}(\tilde{A})}(u) = (\mu_{\tilde{A}}(u))^{1/2}$$

intensificación por contraste:

$$\mu_{\text{int}(\tilde{A})} = \begin{cases} 2(\mu_{\tilde{A}}(u))^2 & \text{para } \mu_{\tilde{A}} \in [0, .5] \\ 1 - 2(1 - \mu_{\tilde{A}}(u))^2 & \text{en otro caso} \end{cases}$$

³¹ [Zadeh 1973a].

Generalmente el seguimiento de las hileras lingüísticas (modificadores) es asociado con los operadores arriba mencionados.

Si \tilde{A} es un termino (un conjunto difuso) entonces

$$\begin{aligned} \text{muy } \tilde{A} &= \text{con } (\tilde{A}) \\ \text{más o menos } (\tilde{A}) &= \text{dil } (\tilde{A}) \\ \text{mas } \tilde{A} &= \tilde{A}^{1.25} \\ \text{un poco } \tilde{A} &= \text{int } [\text{mas } \tilde{A} \text{ y no (muy } \tilde{A})] \end{aligned}$$

donde "y" es interpretado posibilísticamente.

Definición 49³²

Una *variable lingüística Booleana* es una variable lingüística cuyos términos, X , son expresiones Booleanas en variables de la forma $X_p, m(X_p)$ donde X_p es un termino primario y m es un modificador. $m(X_p)$ es un conjunto difuso resultado de la actuación con m en X_p .

7.2 LÓGICA DIFUSA

7.2.1 REVISIÓN DE LA LÓGICA CLÁSICA

Lógica como base del razonamiento puede ser distinguida esencialmente por sus tres tópicos neutrales (contexto independiente): valores verdad, vocabulario (operadores) y procedimiento de razonamiento (tautologías, silogismos).

En la lógica Booleana, los valores de verdad pueden ser 0 (falso) ó 1 (verdadero) y por medio de estos valores de verdad el vocabulario (operadores) es definido vía las tablas de verdad.

Consideraremos dos declaraciones, A y B, ambas pueden ser verdaderas o falsas, esto es, tenemos el valor verdad 1 ó 0. Podemos construir la siguiente tabla verdad:

A	B	\wedge	\vee	$\times\vee$	\Rightarrow	\Leftrightarrow	?
1	1	1	1	0	1	1	1
1	0	0	1	1	0	0	1
0	1	0	1	1	1	0	0
0	0	0	0	0	1	1	0

³² [Zadeh 1973a, pag. 87].

Hay $2^2 = 16$ tablas de verdad, cada una definiendo un operador. Asignar significados (palabras) a estos operadores no es difícil para los primeros 4 ó 5 columnas: obviamente el primero en definir es el "AND", el segundo es el "OR inclusivo", el tercero es el "OR exclusivo" y el cuarto y quinto la implicación y la equivalencia. Sin embargo, tendremos dificultades en interpretar las restantes 9 columnas en términos de nuestro lenguaje. Si tenemos tres declaraciones en vez de tener dos, esta tarea de asignación de significados a las tablas de verdad se vuelve incluso más difícil.

Hasta aquí ha sido asumido que cada declaración, A y B, podría claramente ser clasificada como verdadera o falsa. Si esto ya no es verdadero entonces los valores de verdad adicionales, tales como "indecisión" o similar, puede y tiene que ser introducido, el cual encabeza los sistemas existentes de lógica multivaluada. Esto no es difícil ver, como los problemas mencionados arriba de la lógica bi-valuada en las "llamadas" tablas de verdad u operadores de crecimiento, como movernos a lógica multivaluada. Para solo dos declaraciones y tres posibles valores de verdad hay ya $3^2 = 729$ tablas de verdad! La unicidad de interpretación de las tablas de verdad, la cual es así conveniente en lógica Booleana, desaparece inmediatamente porque muchas tablas de verdad en lógica trivaluada se ven muy parecidas.

El tercer tópico neutral de los sistemas lógicos es su mismo procedimiento de razonamiento que generalmente se basa en tautologías tales como

modus ponens:	$(A \wedge (A \Rightarrow B)) \Rightarrow B$
modus tollens:	$((A \Rightarrow B) \wedge \neg B) \Rightarrow \neg A$
silogismo:	$((A \Rightarrow B) \wedge (B \Rightarrow C)) \Rightarrow (A \Rightarrow C)$
contraposición:	$(A \Rightarrow B) \Rightarrow (\neg B \Rightarrow \neg A)$

Consideraremos el modus ponens que podría ser interpretado como: "si A es verdadero y si la declaración "si A es verdadero entonces B es verdadero" es también verdadera entonces B es verdadero".

El termino *verdadero* es usado en diferentes lugares y en dos diferentes sentidos: pero todos los últimos "verdaderos" son material verdadero, esto es, son tomados como una materia de hecho, mientras que el ultimo "verdadero" es un tópico neutral lógico "verdadero". En la lógica Booleana, sin embargo, esas "verdades" son tratadas del mismo modo. Una distinción entre verdad material y lógica (necesaria) es hecha en la llamada lógica extendida: la lógica modal distingue entre verdad necesaria y posible, tiempo lógico entre los enunciados que son verdad en el pasado y que serán verdad en el futuro. La lógica epistémica trata con el conocimiento y creencia y la lógica deóntica con la que debe ser hecha y con la que se permite ser verdad. La lógica Modal, en particular, debería ser una muy buena base para aplicar deferentes medidas y teorías de incertidumbre.

Otra extensión de la lógica Booleana es el cálculo del predicado, el cual es un conjunto teórico de lógica usando cuantificadores (todo, etc.) y predicados en adición a los operadores de lógica Booleana.

La lógica difusa es una extensión del conjunto teórico de lógica multivaluada en la cual los valores verdad son variables lingüísticas (o términos de variable lingüística verdad).

De operadores como \wedge , \vee , \neg , \Rightarrow en lógica difusa son también definidos para usar con tablas de verdad, el principio de extensión puede ser aplicado para derivar definiciones de operadores. Hasta aquí, la teoría de posibilidad ha sido usada primariamente para definir operadores en lógica difusa, aunque otros operadores también han sido investigados, y podrían también ser usados. Nos limitaremos a consideraciones para interpretaciones posibilísticas de variables lingüísticas.

Si $v(A)$ es un punto en $V[0, 1]$, representando el valor de verdad de la proposición "u es A" o simplemente A, entonces el valor verdad de no A esta dado por:

$$\bar{v}(\text{no } A) = 1 - v(A)$$

Definición 50

Si $\bar{v}(A)$ es un conjunto difuso normalizado, $\bar{v}(A) = \{(v_i, \mu_i) \mid i = 1, \dots, n, v_i \in [0, 1]\}$ entonces aplicando el principio de extensión, el valor verdad de $\bar{v}(\text{no } A)$ es definido como

$$\bar{v}(\text{no } A) = \{(1 - v_i, \mu_i) \mid i = 1, \dots, n, v_i \in [0, 1]\}$$

En particular "falso" es interpretado como "no verdadero", esto es

$$\bar{v}(\text{falso}) = \{(1 - v_i, \mu_i) \mid i = 1, \dots, n, v_i \in [0, 1]\}$$

Definición 51

Para tablas de verdad numéricas $v(A)$ y $v(B)$ los operadores lógicos *and*, *or*, *not* e *implica* son definidos como:

$$\begin{aligned}
 v(A) \wedge v(B) = v(AyB) &= \{(v, \min\{\mu_A(v), \mu_B(v)\})\} \\
 v(A) \vee v(B) = v(AoB) &= \{(v, \max\{\mu_A(v), \mu_B(v)\})\} \\
 \neg v(A) &= \{(v, 1 - \mu_A(v))\} \\
 v(A) \Rightarrow v(B) = v(A \Rightarrow B) &= \neg v(A) \vee v(B) \\
 &= \{(v, \max\{(1 - \mu_A(v)), \mu_B(v)\})\}
 \end{aligned}$$

si

$$\begin{aligned}
 \tilde{v}(A) &= \{(v_i, \alpha_i)\}, \quad \alpha_i \in [0,1], \tilde{v}_i \in [0,1] \\
 \tilde{v}(B) &= \{(w_j, \beta_j)\}, \quad \beta_j \in [0,1], \tilde{w}_j \in [0,1] \\
 & \quad i = 1, \dots, n; j = 1, \dots, m
 \end{aligned}$$

entonces

$$\tilde{v}(A y B) = \tilde{v}(A) \wedge \tilde{v}(B) = \left\{ \left(v = \min\{v_i, w_j\}, \max_{u=\min\{v_i, w_j\}} \min\{\alpha_i, \beta_j\} \right) \right\}$$

$i = 1, \dots, n; j = 1, \dots, m$

(Esto es equivalente a la intersección de dos tipos de conjuntos difusos) Los otros operadores por consiguiente son definidos.

7.3 RAZONAMIENTO APROXIMADO

Ya mencionamos que en la lógica tradicional la principales herramientas del razonamiento son las tautologías tales como, por ejemplo, el modus ponens, esto es $(A \wedge (A \Rightarrow B)) \Rightarrow B$ ó

Premisa	A es verdadero
Implicación	Si A entonces B
Conclusión	B es verdadero

A y B son enunciados y proposiciones (definidas determinísticamente) y la B en el enunciado condicional es idéntica a la B de la conclusión.

En base de lo que ha sido dicho en las secciones 7.1 y 7.2, dos generalizaciones completamente obvias del modus ponens son:

1. Permitir enunciados que son caracterizados por conjuntos difusos.
2. Relajar (ligeramente) la identidad de la "B" en la implicación y la conclusión.

Esta versión del modus ponens es entonces llamada "modus ponens generalizada"³³

Definición 52

Suponiendo $\tilde{R}(x)$, $\tilde{R}(x, y)$ y $\tilde{R}(y)$, $x \in X$, $y \in Y$ son relaciones difusas en X , $X \times Y$ y Y respectivamente, el cual actúa como restricción difusa en x , (x, y) y y , respectivamente. Suponiendo \tilde{A} y \tilde{B} denotan conjuntos difusos particulares en X y $X \times Y$. Entonces la *regla composicional de inferencia* afirma que la solución de ecuaciones de asignación relacional $\tilde{R}(x) = \tilde{A}$ y $\tilde{R}(x, y) = \tilde{B}$ esta dada por $\tilde{R}(y) = \tilde{A} \circ \tilde{B}$, donde $\tilde{A} \circ \tilde{B}$ es la composición de \tilde{A} y \tilde{B} .

Ejemplo

El universo es $X = \{1, 2, 3, 4\}$.

$\tilde{A} = \text{pequeño} = \{(1, 1), (2, .6), (3, .2), (4, 0)\}$.

$\tilde{R} = \text{"aproximadamente igual"}$ es una relación difusa definida por

\tilde{R} :		1	2	3	4
	1	1	.5	0	0
	2	.5	1	.5	0
	3	0	.5	1	.5
	4	0	0	.5	1

Para la inferencia formal denota

$$\tilde{R}(x) = \tilde{A}, \tilde{R}(x, y) = \tilde{B}, \text{ y } \tilde{R}(y) = \tilde{A} \circ \tilde{B}$$

Aplicando la composición max-min de la inferencia para calcular $\tilde{R}(y) = \tilde{A} \circ \tilde{B}$ produce

$$\begin{aligned} \tilde{R}(y) &= \max\text{-min} \{ \mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x, y) \} \\ &= \{(1, 1), (2, .6), (3, .5), (4, .2)\} \end{aligned}$$

Una posible interpretación de la inferencia puede ser la siguiente:

x es pequeño

x y y son aproximadamente iguales

y es más o menos pequeño

³³ [Zadeh 1973a, p. 56; Mizumoto et al. 1979, Mamdani 1977a].

Una aplicación directa del razonamiento aproximado es el algoritmo difuso (una secuencia ordenada de instrucciones en la cual algunas de las instrucciones pueden contener etiquetas de conjuntos difusos) y el diagrama de flujo difuso.

8. APLICACIONES DE LÓGICA DIFUSA

8.1 INTRODUCCIÓN

Desde las primeras ideas formalizadas de la Lógica Difusa en los años 60, se ha venido intentando con mayor o menor éxito la aplicación de esta rama de la lógica al Control de Procesos.

La Lógica Difusa surge como un intento de formalización del razonamiento con incertidumbre. En particular, y al contrario que otras formas de razonamiento, intenta abordar problemas definidos en términos lingüísticos, y por tanto imprecisos, donde los datos están expresados en términos cualitativos.

8.1.1 ALGUNOS PIONEROS EN EL TEMA

Aunque las primeras formulaciones serias sobre Lógica Difusa no comienzan a desarrollarse hasta bien entrada la segunda mitad del siglo, no es menos cierto que la intranquilidad de conciencia de algunos pensadores del siglo XIX ya les había llevado a plantear algunas de las ideas que, sin saberlo, darían mucho después lugar a la aparición de la teoría de conjuntos difusos.

Resulta bien conocida la controversia suscitada entre G. Cantor y L. Kronecker (segunda mitad del siglo XIX) sobre el significado matemático de los conjuntos infinitos. Según el propio Kronecker (1886) no es posible aplicar la formulación del concepto *un sistema modular con elementos no finitos*, pues si se intentara una aplicación matemática de este concepto tan poco especificado en términos aritméticos, sólo podría probarse para cada subsistema concreto, por lo que en cada uno de dichos subsistemas la formulación de sistemas modulares con elementos no finitos resultaría innecesaria.

Ante esto, la relación de R. Dedekind (1888) fue ésta:

De la misma forma que un sistema S es objeto de nuestro razonamiento (...) completamente determinado, para cada situación se puede determinar si es o no un elemento de S . En qué forma esta determinación se lleva a cabo y si disponemos o no de un modelo para decirlo, es completamente indiferente para las siguientes consideraciones (...) Y menciono esto explícitamente, pues el Sr. Kronecker intentó recientemente imponer ciertas restricciones a la libre formulación de conceptos matemáticos, lo que en mi opinión no tiene ningún fundamento.

Una postura intermedia a la de Kronecker y Dedekind³⁴ consistía en decir que un conjunto S está completamente determinado si y sólo si existe un criterio de decisión que especifique el grado de pertenencia de un elemento a S. Según la lógica clásica estaríamos en el caso de funciones características, según la lógica multivalorada derivaríamos hacia el concepto de funciones de pertenencia introducido por Zadeh.

En este sentido es en el que se pueden considerar a Kronecker y Dedekind como los precursores de la teoría de conjuntos difusos.

8.1.2 ORÍGENES DEL PENSAMIENTO DIFUSO (FUZZY)

Procedente del inglés, el término *fuzzy* se ha venido traduciendo por difuso o borroso (¡pero no por confuso, como lo hacen algunos!), según las preferencias de cada autor. Sin embargo, es también una práctica aconsejable no traducirlo ya que, aparte de dar una idea poco afortunada sobre el mismo, se evita la ingrata tarea de tener que traducir también otros términos que aparecen frecuentemente en la literatura sobre *fuzzy logic* como fuzziificación y desfuzziificación.

En un primer acercamiento, la lógica difusa puede ser contemplada en contraposición a la lógica clásica. Esta última establece que cualquier enunciado o proposición puede tomar un valor lógico VERDADERO o FALSO, en definitiva, 1 ó 0 (parece evidente que una puerta, o está abierta o está cerrada). De esta forma, es posible desarrollar toda una lógica que se base en leyes del tipo:

$p \wedge \bar{p}$ es siempre falso
 $p \vee \bar{p}$ es siempre verdadero
 $(p \Rightarrow q) \wedge p$, por tanto q
 $(p \Rightarrow q) \wedge \bar{q}$, por tanto \bar{p}
 etc.

Sin embargo, en el año 1965, la aparición de un trabajo del soviético Lofti Zadeh sobre *La teoría de los Conjuntos Difusos*, revoluciona la matemática al proponer una nueva lógica denominada *fuzzy*. Según la misma, una puerta no tiene porque estar necesariamente abierta (1) o cerrada (0), sino que puede estar *abierta a medias* (0.5), *bastante abierta* (0.8), *casi cerrada* (0.1), etc.

La principal ventaja de utilizar términos lingüísticos como *a medias*, *bastante*, *casi*, *poco*, *mucho*, *algo*, etc., está en que permite plantear el problema en los mismos términos en lo que lo haría un experto humano. De hecho, el propio Zadeh justifica el inexplicable éxito de sus trabajos con la frase: *El mundo es fuzzy*. En otras palabras, no tiene sentido buscar la solución a un problema no

Según Höle y Neff Stout, 1991.

perfectamente definido, por medio de un planteamiento matemático muy exacto, cuando es el ser humano el primero que razona empleando inexactitud.

Según el teórico Berkeley, ... *cualquiera de nosotros es capaz de estacionar su automóvil en unos pocos segundos, porque no hace falta encajarlo exactamente en un espacio perfectísimamente delimitado. Si pretendiéramos hacerlo así, tardaríamos tres años ...*

Según lo dicho, puede deducirse que el empleo de las lógicas difusa y clásica, es función directa del problema a resolver. El conjunto de los números pares e impares encajaría perfectamente dentro de la clásica: cualquier número es par o es impar.

Sin embargo, pretendamos contestar al siguiente interrogante: *¿Carlos es alto, alto bajo o tiene una estatura normal?* Más aún, *¿Qué significa ser alto bajo y cómo se representa este concepto en términos matemáticos?* Este tipo de problemas precisamente son los que aborda la lógica difusa.

8.2 SISTEMAS EXPERTOS EN EL CONTROL DE PROCESOS

8.2.1 SISTEMAS EXPERTOS Y CONTROL DIFUSO

8.2.1.1 TEORIA DE CONJUNTOS DIFUSOS

La incertidumbre, en los procesos de toma de decisión, es un hecho que si bien, no es deseable, está presente en los casos en que para tomar decisiones no se tiene información determinística. Los conjuntos difusos ofrecen un método para manejar estas situaciones de instrucciones inciertas y poco claras, y ofrecen una respuesta computacional al trabajo con leyes de no-contradicción y la exclusión media de la lógica Aristotélica.

Los primeros sistemas difusos datan de los años veinte cuando fueron propuestos por Lukasiewicz. En 1965, Lofti Zadeh codificó y expandió el trabajo de Luka y proporcionó la teoría matemática suficiente para trabajar con conjuntos difusos. La teoría de conjuntos difusos, entonces establecida, permitió el desarrollo del primer sistema comercial difuso en el control industrial. En los inicios de los años 70's, empezó a notarse una gran influencia de los conjuntos difusos en ingeniería de control con el surgimiento de los controladores difusos.

Para abordar y entender la teoría de conjuntos difusos es casi preferible olvidarse, por lo menos durante un tiempo, de nuestros conocimientos de la teoría clásica de conjuntos, ya que la idea de partida es completamente distinta. En la teoría clásica un elemento cualquiera o bien pertenece a un conjunto o bien no pertenece al mismo. El teoría difusa un elemento siempre pertenece en cierto grado a un conjunto y nunca pertenece del todo al mismo.

Una definición podría ser la siguiente:

Un *subconjunto difuso* A de un universo $X = \{x\}$ es un conjunto de pares ordenados $a = \{(x | \mu_A(x)) \forall x \in X\}$, donde $\mu_A : X \rightarrow [0,1]$ es la *función de pertenencia* asociada a A .

El grado de pertenencia no tiene un sentido probabilístico, como en otras lógicas, sino más bien representa un grado de compatibilidad de un cierto predicado o un grado de posibilidad de que éste sea cierto.

El primer punto que debe afrontarse al enfocar el tema, es el de la representación matemática de las expresiones lingüísticas. Podemos formalizar que una persona varón es *alto* cuando mide 1.90, y que hace frío cuando la temperatura es de 10°C. Evidentemente, una estatura de 1.75 ya no correspondería a un varón alto, sino a uno de estatura *normal*. "*1.90 es alto*" es cierto al 100%, "*1.75 es alto*" es cierto al 0%, pero "*1.75 es normal*" es cierto al 100%. Pues bien, la aplicación de la lógica difusa consiste en asignar a estaturas comprendidas entre 1.75 y 1.90, valores borrosos entre 0 y 1. Si la interpolación se hace de forma lineal, entonces "*1.80 es alto*" es algo cierto (0.33 difuso).

Siguiendo con el ejemplo de las estaturas, se representan en la figura (4-4) las funciones de pertenencia de distintos conjuntos. Por ejemplo, la función de pertenencia del conjunto muy bajo se ha tomado como:

$$\begin{array}{ll} \text{si } x \leq 1.50 \text{ m} & \mu(x) = 1 \\ \text{si } 1.50 \leq x \leq 1.60 \text{ m} & \mu(x) = 1 - 10 * (x - 1.5) \\ \text{si } x \geq 1.60 \text{ m} & \mu(x) = 0 \end{array}$$

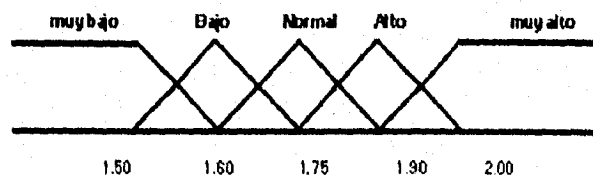


Figura 4-4

En la lógica difusa los *valores semánticos* serán subconjuntos difusos del intervalo real $[0,1]$, siendo necesario definir para cada predicado los correspondientes subconjuntos que, en el caso normal, será un número finito.

Llegados a este punto, conviene definir o recordar algunos conceptos que se emplean con asiduidad:

- Universo de Discurso
- Etiquetas Lingüísticas
- Funciones de Pertenencia

Como ya se ha definido anteriormente, se denomina *universo del discurso* al conjunto de posibles valores particulares que pueden tomar variables que intervienen en el predicado. En el eje de abscisas del ejemplo de la figura (4-4) se representa el Universo de Discurso, es decir, el rango de variación de las variables estatura, que se han elegido entre 1.50 y 2.00 m.

Se denominan *etiquetas lingüísticas* los valores semánticos correspondientes a un predicado. En el ejemplo se han tomado cinco etiquetas lingüísticas. Estas etiquetas se corresponden en gran medida con el lenguaje natural. Habitualmente se toman entre 5 y 9 términos, ya que menos no definen bien el problema, y más se salen fuera de los límites que sería capaz de discernir una persona (hay diferencia entre *muy bueno* y *buenísimo*, pero ¿en qué se diferenciaría *muy muy bueno* si lo intercalásemos entre los dos anteriores?. Se define *granularidad* como la capacidad de discernir entre dos términos lingüísticos. Ya hemos indicado que se suelen emplear alrededor de 7, pero cabe advertir que los tests se han hecho en inglés, cuya semántica posee matrices diferentes en español.

Las *funciones de pertenencia* de un término lingüístico, son las que aparecen en la figura (4-5). Cada término lingüístico corresponde a un subconjunto difuso que lleva asociada una función de pertenencia. Esta representa el grado de asociación de un valor numérico x con ese término. Por ejemplo, y como ya se dijo antes, "1.80 es ALTO" toma el valor 0.33.

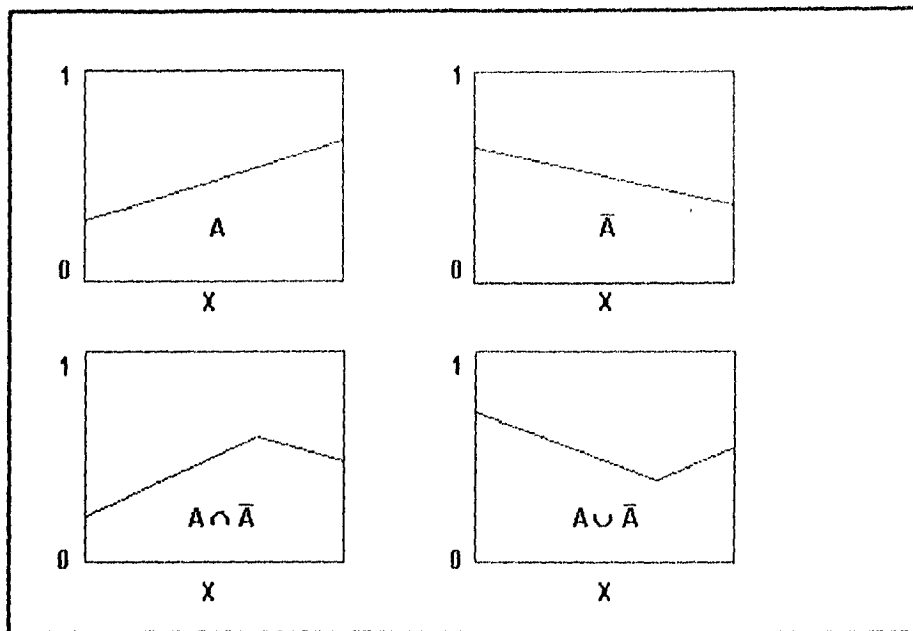


Figura 4-5

La teoría de control difuso fue desarrollada para resolver problemas en los cuales la pertenencia de un conjunto no tiene límites definidos (valores discretos), sino una escala continua, es por ello que puede aplicarse en una amplia variedad de situaciones para confrontar ambigüedad de forma racional. En la teoría de conjuntos tradicional, la intersección de dos conjuntos determina los miembros que pertenecen a los dos conjuntos y se realiza mediante una operación "and" lógica. En teoría de conjuntos difusos, la intersección es realizada mediante la comparación miembro a miembro de cada elemento de los conjuntos difusos y aplicando la función "min" (minimización) a esta comparación, como se muestra en la Tabla 1.

" TABLA 1 "

	sujeto 1	sujeto 2	sujeto 3	sujeto 4
gusta de A	0.0	0.8	0.6	0.4
gusta de B	0.9	0.8	0.4	0.7
gusta de ambas	min (0.0, 0.9), 0.0	min (0.8, 0.8), 0.8	min (0.6, 0.4), 0.6	min (0.4, 0.7), 0.7
gusta al menos de uno	MAX (0.0, 0.9), 0.9	MAX ((0.8, 0.8), 0.8)	MAX (0.6, 0.4), 0.6	MAX (0.4, 0.7), 0.7
que gusta de A implica que gusta de B	MAX ((1.0-0), 0.9), 1.0	MAX ((1.0-0.8), 0.8) 0.8	MAX ((1.0-0.4), 0.6), 0.6	MAX ((1.0-0.4), 0.7), 0.7

La unión de dos conjuntos difusos, es similar a la operación "or" lógica convencional, se realiza aplicando la función "max" (máximo) miembro por miembro a los dos conjuntos (ver Tabla 1). En el manejo de dos conjuntos difusos se requiere de otra operación llamada "implicación" generalmente es denotada por $A \rightarrow B$ y se lee "si A es cierto" implica que B lo es?", que en lógica binaria se expresa con las siguientes relaciones:

$$1 \rightarrow 0 = 0, \quad 1 \rightarrow 1 = 1, \quad 0 \rightarrow 0 = 1, \quad 0 \rightarrow 1 = 1$$

donde $0 \rightarrow 1 = 1$ se debe a que en la operación de implicación, si A no es verdadera no se concede información o se implica algo acerca de B; por lo que cualquier valor de B es aceptado. La operación de implicación puede calcularse en dos formas; la más fácil es calcular la expresión $(\text{NOT } A) \cup B$, la primer parte es realizada restando el valor de A de su máximo valor posible y tomando la unión a continuación (Ver tabla 1).

Se pueden realizar muchas operaciones con conjuntos difusos, pero estas son las básicas.

Los procedimientos de toma de decisión difusa son muy útiles cuando la información es una opinión, no es clara, es vaga o imprecisa en cualquier forma. Los conjuntos difusos pueden ser usados por sí mismos para construir una herramienta de toma de decisiones.

Un ejemplo de este tipo de herramienta fue la presentada en 1981, por Ronald Yager en Iona Collage, cuyo producto de toma de decisiones emplea conjuntos difusos para decidir varias posibilidades, particularmente cuando se debe alcanzar más de un criterio u objetivo.

8.2.1.2 CONTROLADORES LÓGICOS DIFUSOS

Los Controladores Lógicos Difusos (CLD's) son sistemas expertos basados en reglas que emplean variables lingüísticas difusas (alto, bajo, poco, etc.) para modelar los métodos de toma de decisión humana en la solución de algunos problemas, con esto superan la limitación de los sistemas de control, para representar las decisiones humanas.

Como los conjuntos difusos pueden significar diferentes cosas para diferentes personas, son necesarios algunos mecanismos de interpretación constituidos por las llamadas funciones de pertenencia. Lo característico de los sistemas expertos difusos es que sus reglas obtenidas del conocimiento del experto humano, manejan el proceso de decisión, y las funciones de pertenencia convierten las variables lingüísticas en valores numéricos precisos, necesarios para la computadora.

La tarea que más tiempo consume en el diseño de los CLD es la definición de las funciones de pertenencia difusas, pues un pequeño cambio en éstas, puede alterar significativamente el desempeño del controlador; el proceso de determinación de aquellas funciones que producen un máximo desempeño del CLD involucra una búsqueda enorme debido a la gran cantidad de factores no lineales que pueden aparecer en el proceso.

8.2.1.3 ALGORITMOS GENÉTICOS

Un método para realizar esta búsqueda, es el empleo de los Algoritmos Genéticos (AG), que constituyen una técnica de búsqueda gráfica que se enfoca a la optimización. Estos algoritmos emplean operaciones encontradas en la genética natural para guiar sus rutas a través del espacio de búsqueda. Hollstien y De Jong, demostraron que son una técnica eficiente en la optimización de funciones particularmente, establecieron que los AG's son robustos y eficientes para un amplio espectro de problemas, si se les compara con los esquemas tradicionales. La naturaleza robusta y los mecanismos simples de los AG's los han convertido en una herramienta muy útil, no sólo para la definición de las funciones de pertenencia, sino también para la elección de las reglas que serán usadas en los CLD's.

Si bien existen muchas formas para desarrollar AG's, casi todos ellos emplean una composición de reglas de inferencia, que son instrucciones matemáticas que describen como se manipularán las variables lingüísticas, para controlar el ambiente del problema; estas reglas con frecuencia son descritas por funciones difusas complejas.

La oficina norteamericana de minas (*U.S. Bureau of Mines*), desarrolló un método que difiere de esta filosofía, y establece de forma general, lo siguiente:

1. El primer paso es determinar qué variables son importantes para decidir una acción de control efectiva; una vez elegidas estas, se deben identificar las variables de control. A diferencia de las variables de decisión, el número de variables de control no tiene impacto en el tamaño del conjunto de reglas.
2. En seguida, hay que definir los términos difusos que representarán tanto a las variables de control como a las de decisión (algunas veces estos términos son llamados conjuntos difusos). La elección de estos conjuntos difusos define el número necesario de reglas difusas; se escribe una regla por cada condición posible en el sistema físico. las reglas, comúnmente llamadas reglas de condición tiene la siguiente forma:

IF [C, es (C₁) y es (C₂), y ...]
THEN [a₁, es (A₁) y a₂, es (A₂) y ...]

donde: C_i es el conjunto difuso que caracteriza a las variables de decisión, y A_i es el conjunto difuso que caracteriza a las variables de control. La introducción de términos difusos en las reglas hace mucho más fácil su desarrollo comparado con los métodos tradicionales.

Las funciones de pertenencia difusas se pueden considerar como una aproximación de la confianza con la cual un cierto valor numérico preciso pertenece a un conjunto difuso; los valores de la funciones de pertenencia difusas (f) son representaciones numéricas de esa confianza: un valor de $f = 1$ indica la máxima confianza de que el valor numérico dado, pertenece al conjunto difuso. Para cada valor de decisión específico, el conjunto difuso tiene una función de pertenencia; por ejemplo, cada variable de decisión (A_i) puede ser descrita en términos difusos: *baja*, *media* y *alta* simultáneamente, variando el grado de certeza. (Figura 4-2).

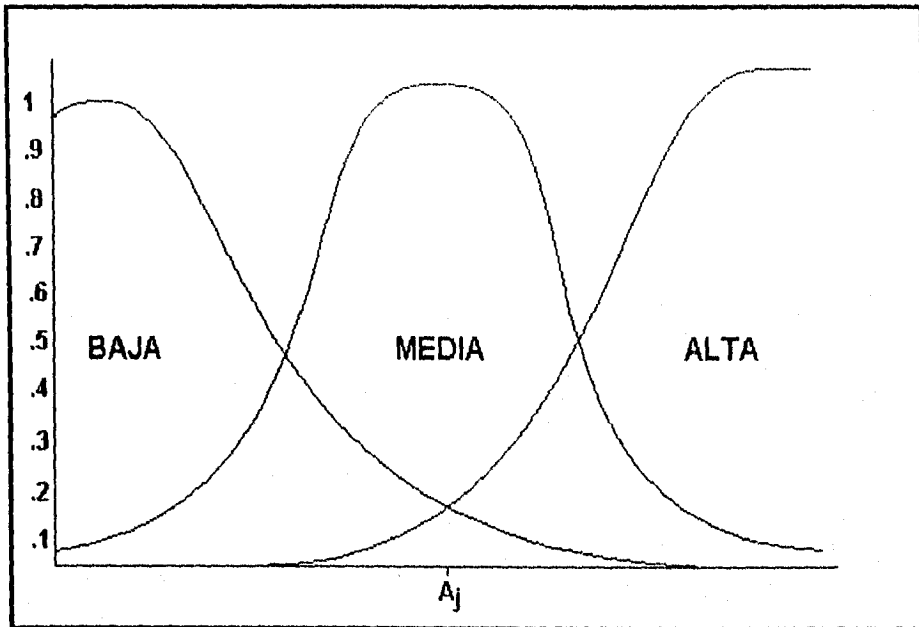


Figura 4-2

Es necesario, por lo tanto, escribir el conjunto de reglas que proporcionen una acción difusa (elegida del conjunto difuso que describe las variables de control) para cualquier condición que pudiera existir en el ambiente del problema.

El uso de subconjuntos difusos permite formar reglas que se derivan básicamente de la intuición del experto, sin embargo, la experiencia con el sistema a ser controlado hace que el desarrollo del conjunto de reglas sea mucho más eficiente.

Una vez obtenido lo anterior, se requiere convertir las acciones difusas proporcionadas por las reglas difusas, a un conjunto específico de acciones

precisas a ser tomadas sobre el sistema físico. L. I. Larckin encontró un procedimiento llamado "esquema del centro de gravedad de área, que es un método eficiente para determinar dichas acciones, en el cual las funciones de pertenencia difusas definen las acciones de control necesarias mediante un procedimiento de suma de pesos. La suma de los pesos se puede obtener fácilmente en términos gráficos: cada regla de producción tiene su función de pertenencia, graficada con una altura igual a la mínima confianza (grado de pertenencia) asociada con la porción de condición de la regla que se cumple (Figura 4-3). A la suma gráfica de las áreas se le determina el centroide, que da el valor de la acción de la acción a aplicar al sistema físico.

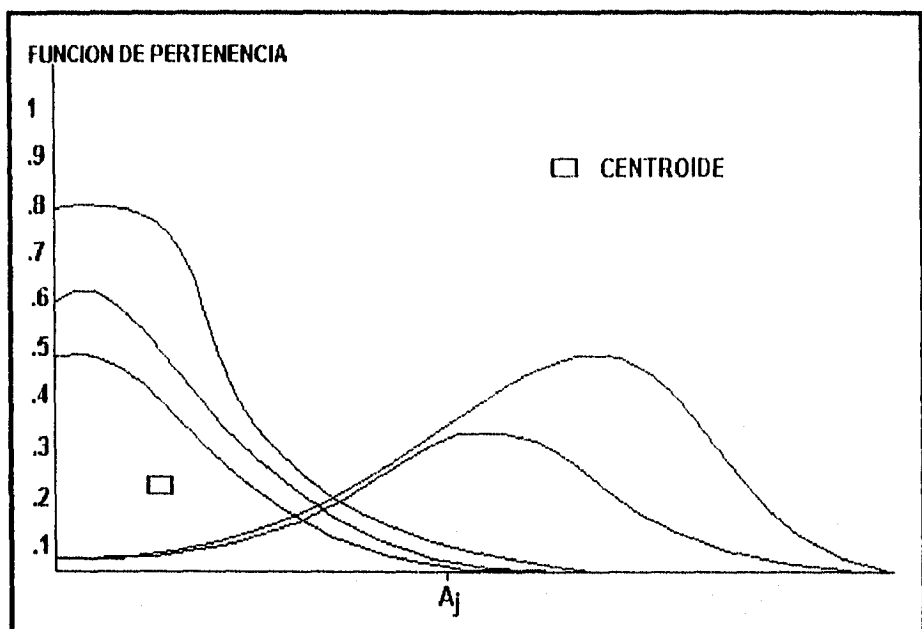


Figura 4-3

Los algoritmos genéticos tienen algunas propiedades que permiten lograr un alto desempeño en la selección de las funciones de pertenencia para los CLD, propiedades que los distinguen de la mayoría de las técnicas de búsqueda convencional, por ejemplo:

- Consideran una población de puntos y no sólo un punto.
- Trabajan directamente con las cadenas de caracteres que representan el conjunto de parámetros, no con los parámetros por sí mismos.
- Usan reglas probabilísticas para dirigir su búsqueda, no reglas determinísticas.

Los AG's consideran muchos puntos del espacio de búsqueda simultáneamente y por tanto tienen una reducida posibilidad de converger a un óptimo local. La población de puntos se genera y prueba en forma iterativa, en un proceso que es similar al proceso de población natural de las criaturas biológicas, en el cual las

generaciones sucesivas de organismos son producidas y eliminadas hasta que ellos mismos son capaces de reproducirse.

En la mayoría de las técnicas convencionales, la búsqueda punto a punto consiste en: seleccionar un punto, probarlo y usarlo en alguna regla de decisión para elegir el siguiente punto a ser seleccionado. Estos métodos pueden ser peligrosos en sistemas multimodales porque pueden converger, a un óptimo local. Los AG's generan poblaciones enteras de puntos (cadenas codificadas), prueban cada punto independientemente y combinan las cualidades de los puntos existentes para formar una nueva población con mejores puntos. Aunado a la producción de búsquedas más globales, la consideración simultánea de muchos puntos se realizan los AG's los hace muy apropiados para el procesamiento en paralelo, dado que la evaluación de cada punto es un proceso independiente. La compatibilidad con los procesadores en paralelo es importante en la selección de las reglas para el sistema de control basados en los CLD's adaptables (aquellos en los que el conjunto de reglas cambia durante el proceso) que contribuye a incrementar la velocidad de ejecución.

Es difícil que los AG's fallen porque no dependen de la continuidad del espacio de parámetros, característica muy importante en el diseño de CLD's, dado que el significado de "continuidad en el espacio de búsqueda" no es muy claro.

Un AG sólo requiere información de la calidad de la solución producida por cada conjunto de parámetros (información objetiva del valor de la función), característica que les da mayor flexibilidad y la diferencia de otros métodos de optimización que requieren de información derivada, o peor aún, de un completo conocimiento de la estructura del problema y los parámetros. También son distintos de los otros métodos en que los AG's usan selección aleatoria para definir la regla de decisión en que se basará la búsqueda. esto no significa que los AG's no recorran de forma aleatoria el espacio de búsqueda, sino que emplean de forma efectiva la selección aleatoria en su exploración del conocimiento prioritario para localizar rápidamente a partir de ella, una solución cercana a la óptima.

Los AG's requieren que el conjunto natural de parámetros del problema a ser codificado, se pueda empezar como una cadena de caracteres, por ello para aplicar los AG's en un problema de búsqueda, se deben tomar dos decisiones:

1. cómo codificar las posibles soluciones al problema en forma de cadenas finitas de bytes y
2. cómo evaluar el mérito de cada una de estas.

En los casos de codificación del diseño de un CLD para el cual se ha establecido un conjunto de reglas, se emplea una codificación común, llamada código binario no signado de concatenación y mapeo. Para la segunda decisión, las cadenas de

bytes que representan el problema de búsqueda deben ser juzgadas y asignadas con una calificación (un valor apropiado de la función) que represente el grado con el cual dicha cadena cumple con la meta de definir un CLD de alto desempeño.

8.3 EJEMPLOS DE APLICACIONES DE SISTEMAS EXPERTOS EN EL CONTROL

SISTEMA DIFUSO APLICADO A LA PRODUCCIÓN DEL CEMENTO

La complejidad del control de una planta de cemento está directamente relacionada con la magnitud de los elementos que intervienen en el proceso, por ejemplo, la planta Lafarge en Ontario, Canadá, posee un orno de 19 ft. de diámetro, 655 ft. de longitud y puede producir un millón de toneladas de cemento anualmente. Para controlar esta instalación, F.L. Smidth & Co. de Copenhague, junto con un equipo de investigación del Queen Mary Collage en Dinamarca, desarrollaron en 1980, un sistema experto en control aplicando teoría de conjuntos difusos, para atacar este problema. Los factores considerados en el manejo del horno por el controlador difuso incluyen:

- tasa de introducción de aceite,
- velocidad de rotación del horno,
- alimentación del producto,
- ventilación del horno (cantidad de aire introducido)

Para simplificar el problema se asumió que lo único que se controlaría sería el consumo de aceite como una medida de la cantidad de oxígeno en la parte posterior del horno.

El sistema difuso completo está incorporado en el sistema de supervisión, diálogo y reporte (SDR), que está a su vez implantado en una micro Hewlett-Packard 26-A600 con 256k de memoria y sin discos.

Antes del SDR Lafarge era controlada por operadores humanos usando una computadora Foxboro I. El SDR fue conectado al sistema existente con una conexión serial RS-232. Lafarge reportó substanciales ahorros en consumo de aceite y un significativo aumento en la vida del ladrillo del horno, lo que trajo menos tiempos muertos y mayor productividad en la planta. Lafarge es uno de los 30 lugares donde se emplea el SDR.

Esta aplicación de sistemas difusos muestra algunos puntos importantes:

1. es posible, en algunos casos, agregar los sistemas difusos al sistema de control existente con ahorros substanciales y en computadoras pequeñas;

2. se eliminan los conflictos de reglas resolviéndolos racionalmente; y
3. se pueden generar funciones de control complejas con unas cuantas reglas.

CONTROL DIFUSO DE CLIMA ARTIFICIAL

La conservación y ahorro de energía se ha convertido en una prioridad, por lo que la Mitsubishi Heavy Industries Inc. (MHII) junto con la empresa Togai Infraclogic Inc. (TIL) decidieron, en 1988, desarrollar un sistema de control de clima para edificios.

Los sistemas convencionales generalmente tienen sensores en cada habitación para la temperatura del cuarto y la pared, pueden, además controlar el intercambio de calor, la velocidad del ventilador y la dirección de las persianas del radiador. La MHII con su sistema difuso realiza mejor la función que el método convencional basado en un control PID.

El sistema TIL usa 25 reglas para calor y 25 reglas para frío, las entradas al sistema tienen una precisión de 8 bits e incluyen además de la temperatura del cuarto y de la pared, la tasa de cambio de estas temperaturas. El sistema final fue desarrollado aproximadamente en cuatro meses, redujo los tiempos de enfriamiento y calentamiento en un factor de cinco, y mejoró la estabilidad de la temperatura, por un factor de dos. El sistema requiere menos sensores y, lo más importante, reduce el consumo de energía de los sistemas de calentamiento y enfriamiento en un 24%.

Una de las razones de esta mejoría estuvo en que casi todos los sistemas convencionales están diseñados para ser incondicionalmente estables, lo que significa que no presentan oscilación bajo ninguna condición, esta situación limita al controlador PID en la cantidad de calor que puede enviar al cuarto, mientras que las reglas del sistema difuso pueden incrementar el calor en la habitación en una tasa mayor. Hasta ahora no se sabe si el sistema difuso entra en oscilación bajo algunas condiciones de entrada, pero se sigue investigando.

PILOTO AUTOMÁTICO DIFUSO PARA EL CONTROL DE UN BARCO

Otra aplicación de la teoría de conjuntos difusos, se encuentra en un modelo para controlar un barco de guerra de la Marina Real Inglesa, mientras este mantiene su curso.

En la tarea de mantener el curso, el timonel debe mantener la ruta y compensar las desviaciones causadas por la dinámica del barco, el estado del mar y/o las perturbaciones por viento. El timonel tiene la siguiente información:

- a) La dirección deseada F_c dada verbalmente por el vigía.
- b) La dirección real F indicada por la cinta del compás.
- c) La tasa de variación D_F que no se tiene directamente disponible, pero se estima por los movimientos de la cinta del compás o por observación de marcas en la costa.

El experto humano realiza la tarea combinando reglas heurísticas precisas e imprecisas, basadas en la experiencia y la intuición.

El elemento básico del modelo del piloto propuesto es un algoritmo difuso de mantenimiento de ruta o curso, con un error de elevación $e = F_c - F$, una tasa de elevación D_F como entradas y la demanda de giro d como salida.

Los controles convencionales de prueba, fueron considerados como un elemento de dirección de fase empleando el error de elevación como entrada y la demanda de giro como salida.

La comparación del desempeño del modelo de control difuso se hizo además de con el experto humano, con un modelo integral difuso que también se desarrolló, y que calcula el promedio de movimientos de error de elevación y la tasa de elevación para los últimos 2.5 seg. El promedio de movimientos es probado con la regla difusa:

“ si el valor del error de elevación está entre $\pm 1^\circ$ y el de la tasa de elevación entre $\pm 0.05\%$, entonces se lleva a cabo la acción integral, en caso contrario, se usa el algoritmo difuso ”

La respuesta del timonel fue observada en tiempo real mientras conducía para mantener el curso. La respuesta del modelo difuso se parece mucho a la del experto humano, y posee menos precisión que la que ofrece el modelo que combina la acción integral con la difusa.

CAPITULO IV
DESARROLLO DEL
SISTEMA EXPERTO TUTORIAL
DE LÓGICA DIFUSA

1. INTRODUCCIÓN

La incorporación de metodologías de enseñanza auxiliada por computadora incrementan la eficiencia en la transmisión y adquisición del conocimiento, permitiendo la manipulación de grandes volúmenes de información, la homogeneización del conocimiento en los educandos, la disponibilidad, masificación y la permanente actualización del mismo.

La educación en México requiere transformar los mecanismos de transmisión de conocimiento, por lo que el proceso enseñanza-aprendizaje debe ser homologado con los avances tecnológicos. Una de las herramientas tecnológicas que han permitido modificar las técnicas y metodologías del proceso enseñanza-aprendizaje es la computación. Es así, que en la actualidad, los sistemas expertos tutoriales computarizados constituyen una excelente alternativa de capacitación, tanto para estudiantes, técnicos u otros usuarios que requieran información especializada en un dominio de su interés. Un sistema experto tutorial permite la búsqueda y obtención rápida del conocimiento, el cual ha sido debidamente organizado, clasificado y estructurado para tales fines.

En la actualidad el desarrollo de aplicaciones educativas en computadora es foco de interés en instituciones educativas, de investigación y difusión, de empresas industriales y comerciales, consultoras, etc., aunado a esto, el desarrollo de la tecnología computacional se encuentra en una de las mejores etapas en lo que a recursos, herramientas y sistemas se refiere para desarrollar paquetes educativos en computadora, de los cuales los sistemas expertos tutoriales son una de las mejores alternativas por el enorme campo de acción potencial en todos los sectores, principalmente en la industria donde permitirían la posibilidad de capacitación personalizada del personal tendiente a mejorar productividad, calidad y eficiencia.

1.1 DEFINICIÓN DEL PROBLEMA

Actualmente en Ingeniería y en la División de Estudios de Posgrado de la Facultad Ingeniería (DEPFI) se sigue estudiando el control y sus mecanismos para el progreso del mismo, es por ello que siempre se debe estar al tanto de las nuevas teorías y aplicaciones que surgen para el mismo, como es el caso de la teoría de la lógica difusa, que no se ha difundido ampliamente más sin embargo en otros lugares como Japón y Estados Unidos, se han obtenido resultados positivos en su aplicación hacia el control.

Existe una gran necesidad de obtener información teórica básica y general sobre las diversas aplicaciones de la lógica difusa tanto en el sector industrial como en el académico puesto que la falta de difusión se debe a diversas razones, incluyendo la falta de expertos en esta nueva teoría, así como los problemas que conlleva a capacitar a otras personas en lo mismo.

1.2 POSIBLES SOLUCIONES

Existen diversas soluciones al problema de la capacitación tales como la publicación por diversos medios, ya sean boletines, revistas, folletos, libros, videos, documentales, etc., de técnicas y teoría en general, así como de los logros obtenidos en la implementación de sistemas de lógica difusa mundialmente.

Entre otras posibilidades existe el poder formar sociedades cuyo objetivo específico sea brindar ayuda técnica de este tópico dependiendo del tipo de información que le sea requerida, es decir, si la petición necesita de una amplia información, se le puede proporcionar dependiendo de la tecnología con que cuente quien así lo requiera, puesto que de ella depende la forma de solucionar los requerimientos, ya sea que implique ir personalmente a brindar esta asesoría o que se pueda resolver el problema por otros medios.

1.3 ¿POR QUÉ UN SISTEMA EXPERTO?

Las ventajas de la aplicación de un sistema experto tutorial para la enseñanza y consulta sobre cualquier dominio del conocimiento son principalmente:

- a) *Autonomía*: Luego que el sistema experto tutorial ha sido diseñado y completado, se hace autónomo, es decir, independiente de la presencia física del desarrollador del sistema y del especialista en el dominio a aprender que participó en tal desarrollo.
- b) *Reproducibilidad*: El sistema experto tutorial en sí y el conocimiento que abarca son reproducibles a voluntad de su autor. Esto implica que, en caso de ser necesario, el producto puede reproducirse para dar servicio a miles de usuarios. Por otro lado, en contraposición con la lentitud del proceso de formación de especialistas, la reproducción de un sistema experto tutorial es de orden de unos minutos solamente.
- c) *Bajo costo de adquisición y operación*: En tanto que disponer permanentemente de instructores supone altos costos, un sistema experto tutorial requiere únicamente de pagar por su diseño y construcción. Estos costos después se distribuyen entre todos los usuarios del sistema, por lo que el costo por copia es muy reducido.
- d) *Facilidad de distribución*: Por la naturaleza del sistema, pueden cubrirse necesidades en localizaciones muy dispersas y bajo condiciones de trabajo difíciles.

e) *Flexibilidad para modificaciones y expansión*: La información de la base de conocimientos del sistema puede ser actualizada con fines de ampliación o profundización, con solo acceder a los archivos de texto, ya que estos no forman parte integrante de las bases de conocimiento, y sin necesidad de reestructurar la secuencia de trabajo.

Un sistema experto tutorial se diseña considerando los siguientes aspectos relacionados con el proceso de enseñanza-aprendizaje:

Presentar opciones para que el usuarios decida hacer la evaluación de entrada o iniciar directamente la capacitación en todos los temas.

A partir de la evaluación de entrada (de todo el módulo), el sistema guía al usuario hacia las áreas que se hayan identificado como las más débiles.

A partir de la evaluación formativa, el sistema debe recomendar al usuarios retornar a las áreas en las que se haya identificado que falta reforzar su conocimiento. (Esto constituye la retroalimentación). Este proceso puede hacerse por temas o grupo de subtemas.

Luego de las retroalimentaciones formativas de todos los subtemas, se efectúa la evaluación sumativa de todo el tema. Se tendrán opciones de aprobado o desaprobado en el tema. Si se aprueba, se pasa al tema siguiente, en caso contrario, se regresa al proceso para estudiar los temas no aprobados.

Dicho todo lo anterior es posible desarrollar un sistema experto tutorial orientado para brindar información teórica básica y general sobre las matemáticas difusas y su empleo está destinado a estudiantes de licenciatura y de posgrado, así como también a profesionales que deseen iniciarse en el estudio de este tema. Además, será una buena opción para la capacitación tanto en empresas como en instituciones que así lo soliciten.

Al mismo tiempo, con este sistema se pretende dar a conocer la importancia de la lógica difusa y sus aplicaciones, ya que al proporcionar capacitación sin la necesidad física de un experto las empresas e instituciones pueden tomar decisiones con respecto a la mejor aplicación que se puede implementar en las mismas. Además, cada una de ellas puede individualizar al sistema aportando información en base a las experiencias que se obtengan al poner en práctica los conocimientos de este tutorial; tomando en cuenta que el sistema puede tener una retroalimentación en la cual el buen funcionamiento depende de los datos aportados por cada una de los educandos.

1.4 ANÁLISIS Y FORMULACIÓN DEL PROBLEMA

En esta etapa se evalúa la conveniencia de que un problema determinado sea abordado a través de un sistema experto. Los factores evaluados son: el problema, los resultados que se esperan alcanzar y las personas fundamentales que participan en el desarrollo, como son el experto y el diseñador. De acuerdo con los requerimientos del usuario, los recursos disponibles y las características del problema, se fijan los objetivos y las tareas que deben ser realizadas por el sistema experto.

Criterio de selección de problemas: La decisión de desarrollar un sistema experto sólo puede tomarse después de un examen minucioso del contexto de la tarea por encarar. Cierta cantidad de factores hacen que sea razonable pensar en este tipo de desarrollo:

- El problema planteado pone en juego no sólo informaciones cuantitativas sino también cualitativas.
- Los conocimientos no sólo vinculan con el sentido común sino que tienen carácter intuitivo.
- Existen expertos reconocidos, motivados y disponibles, capaces de resolver el problema y de explicar su proceso de solución.
- El problema es de dificultad razonable y no posee solución algorítmica posible o deseable.

Es necesario, además, asegurarse de que la solución de tipo de sistema experto para el problema planteado esté realmente justificado. Una justificación sobre la base de criterios puramente económicos no es suficiente, ya que por lo general, el rédito de la inversión de un sistema experto suele ser difícil de calcular.

Por el contrario, acudir a los sistemas expertos se justifica si:

- La habilidad en el tema es rara o insegura, por ejemplo cuando un experto está pronto a jubilarse.
- La habilidad se concreta en un punto, pero se le utiliza de manera dispersa.
- Las decisiones deben tomarse en condiciones "estresantes"

Participantes: En el estado actual de la Inteligencia Artificial, el desarrollo de un sistema experto hace que intervengan por lo menos dos personas (o dos grupos de personas):

- Un experto en el dominio considerado.
- Un especialista de inteligencia artificial, al que se le conoce como especialista en conocimiento o ingeniero del conocimiento, que tenga una excelente cultura informática.

El participante de mayor importancia es el usuario, quien a pesar de que no participa activamente en el desarrollo del sistema experto, es el factor esencial para evaluar las posibilidades de que el problema sea desarrollado a través de un sistema experto y cuando la decisión sea afirmativa, es fundamental tener en cuenta al usuario final del sistema experto en lo que concierne al diseño de las interfaces respectivas.

Formulación del problema

Los requerimientos de validación del sistema se basaron en la secuencia que consideró el experto para el mejor aprendizaje de los temas y ésta consistió en restringir el acceso a ciertos temas dependiendo del nivel de conocimientos que tenga el usuario así como de haber cubierto los requisitos necesarios para la comprensión de temas subsecuentes. De hecho, la mayor restricción para la continuidad de los temas son los exámenes que se aplican al término de cada tema y es el factor decisivo para que el usuario pueda seguir con la seriación de temas programada. Dicha secuencia de temas se puede ver en forma gráfica con el siguiente diagrama de bloques general:

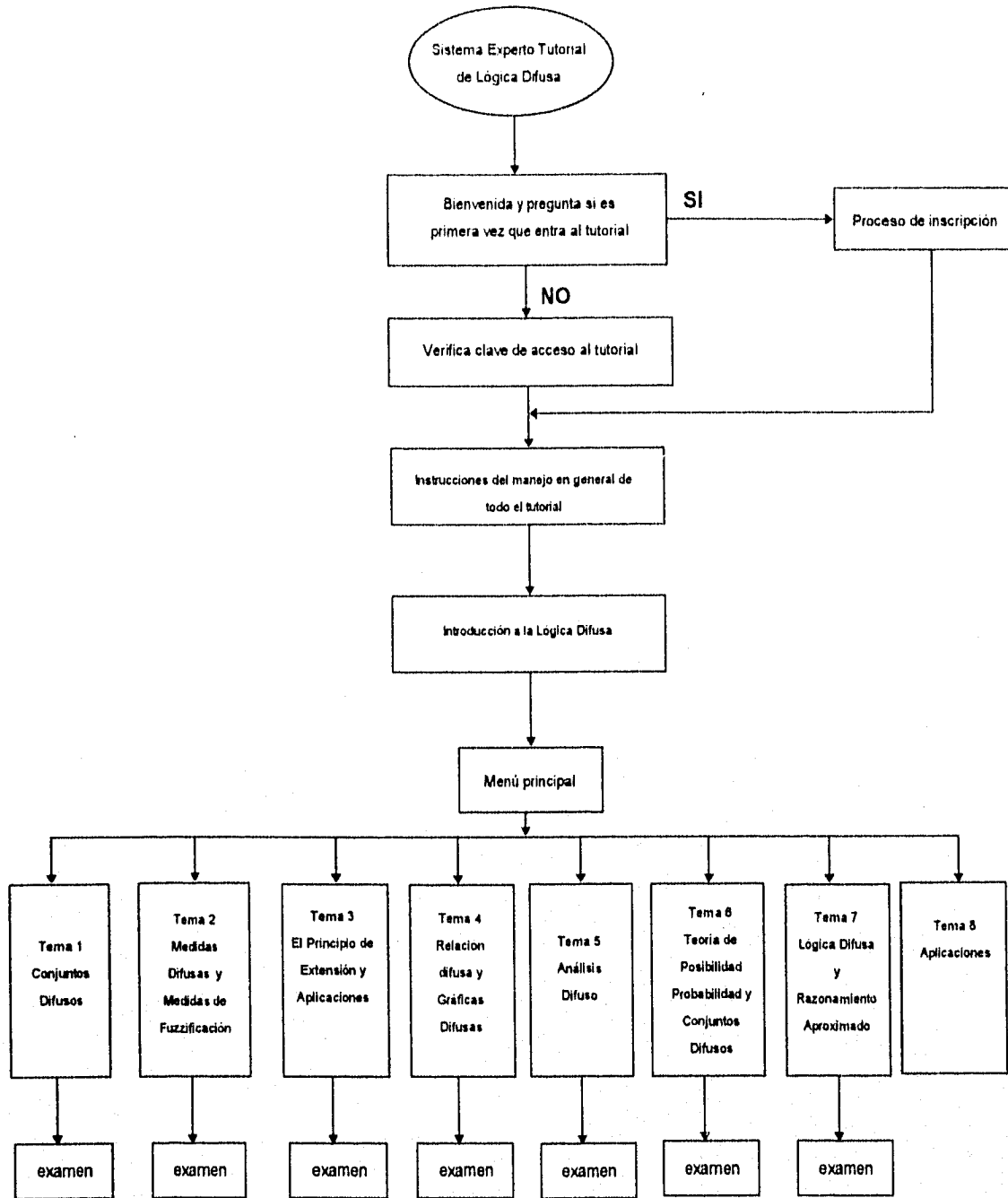


Diagrama 4.1 Sistema Experto Tutorial de Lógica Difusa

2. DESARROLLO DE UN SISTEMA EXPERTO TUTORIAL

Una vez identificado el problema que solucionará el SET, empieza la gran labor de desarrollarlo.

Para desarrollar un SET es necesario identificar a los expertos que participarán, y formar un conjunto de factores de decisión esenciales del área que abarca su experiencia; para esto, es necesario y de su importancia, entrevistar a expertos en el área para la cual se elaborará la aplicación. Hay que poner atención en el hecho de que las decisiones que tome el experto durante las entrevistas, seguramente cambiarán si el experto se encontrara bajo presión.

También se deben crear algunas reglas generales para iniciar la base de conocimientos que contendrá el prototipo. El simular el proceso de decisión del experto, es un trabajo difícil, para empezar, es necesario leer todo el material introductorio como si el elaborador del SET fuera un estudiante de la materia y posteriormente simular el proceso que realiza el experto en una máquina que no posee intuición.

2.1 PROCESO PARA IMPLEMENTAR UN SISTEMA EXPERTO

Todos los Sistemas Expertos comienzan con la percepción de un problema que es difícil de resolver usando métodos de programación tradicionales. Los SET solamente son una extensión del conjunto de herramientas que provee una computadora y tienen el potencial de asistir a los usuarios, de la manera en que un libro no podría hacerlo, presentando sólo las preguntas que necesitan ser contestadas y que se basan en respuestas anteriormente dadas por el usuario.

Dado que un SET requiere de experimentación para su buen desempeño, su desarrollo es gradual. El procedimiento para extraer el conocimiento del experto y codificarlo en forma de programa, se llama *adquisición del conocimiento*. Esta transferencia de información es el corazón del proceso del desarrollo del SET.

La figura 4-1 nos muestra los pasos envueltos en la implementación de un Sistema Experto, mismos que se encuentran en el desarrollo de un SET; están clasificados de acuerdo a las cuatro fases principales de la implementación.

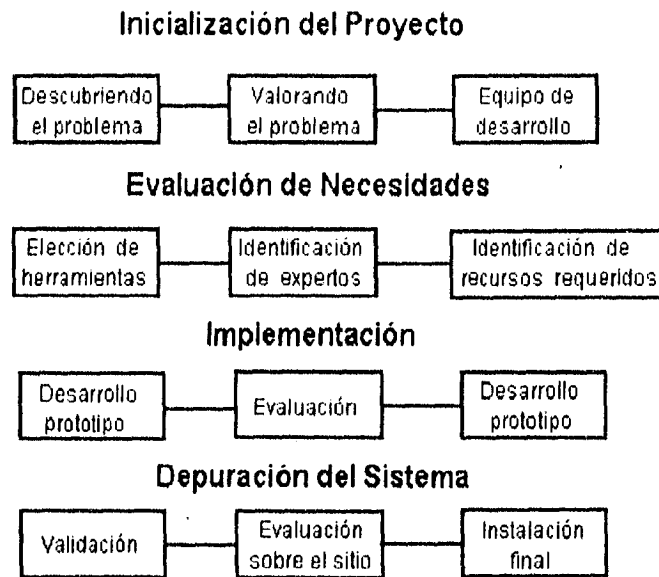


figura 4.1 Proceso para implementar un SE

2.1.1 INICIALIZACIÓN DEL PROYECTO

Esta primera parte es una introducción general al proceso que conduce a los requerimientos de un SE. Comienza con la percepción de un problema, examinando la manera en la cual los expertos presentan una posible solución. De la necesidad percibida, el grupo de trabajo puede valorar la utilidad de un SE desarrollando criterios con los cuales evaluar la factibilidad del proyecto. A través de este primer nivel, se asume que no se cuenta con el conocimiento implementado.

Durante la identificación del problema, el ingeniero del conocimiento y el experto trabajan juntos para identificar el área del problema y definir su alcance.

La decisión de elaborar un SE requiere de una evaluación previa del problema, el cual debe ser de dificultad razonable, además de no presentar una solución algorítmica posible o deseable, como es el caso de la elaboración de tutoriales inteligentes SET, los cuales contienen los conocimientos y habilidades de los pocos expertos existentes en el área.

1.1.2 EVALUACIÓN DE NECESIDADES

En esta etapa, se identifican los participantes en el proceso de desarrollo (experto en el área en que se desarrollará el SE, especialista en IA, expertos adicionales y usuarios finales), se determinan los recursos necesarios (tiempo, facilidades de cómputo) y se deciden los objetivos de construcción del SET.

Una vez definido el problema que atenderá el SE, se localizan los conceptos claves y relaciones dentro de una representación formal sugerida por alguna herramienta de construcción de SE o lenguaje. El ingeniero del conocimiento debe seleccionar el lenguaje y, con ayuda del experto, representar los conceptos básicos y relaciones dentro del área de trabajo del lenguaje.

Para poder elegir un *shell* adecuado, debe tenerse en cuenta su costo y su capacidad de contener objetos y/o reglas, además de que la distribución del SE final sea adecuada a los requerimientos del problema. El *shell* debe presentar una interfaz del usuario agradable, con las facilidades que requiera el problema para el cual se desarrollará la aplicación.

Durante la conceptualización el experto y el ingeniero del conocimiento explican los conceptos clave, relaciones, reglas de inferencia, restricciones y métodos de razonamiento, características necesarias para describir el proceso de resolución del problema en el dominio dado. El experto y el ingeniero del conocimiento trabajan en estrecha colaboración para conformar la base de conocimientos del SET.

1.1.3 IMPLEMENTACIÓN

Durante la implementación, el ingeniero del conocimiento combina y reorganiza el conocimiento formalizado para hacerlo compatible con las características del flujo de información del problema. El conjunto resultante de reglas y la estructura de control asociada, define un programa prototipo que puede ejecutarse y ponerse a prueba.

1.1.4 DEPURACIÓN DEL SISTEMA

Finalmente, la prueba envuelve la evaluación del prototipo y lo revisa para conformar a estándares de excelencia definida por expertos en el área (dominio del problema). Típicamente, el experto evalúa el programa y asiste al ingeniero de conocimiento en revisiones subsecuentes.

Estas etapas del desarrollo del SE no están bien definidas, sólo muestran una guía (no estricta) para la elaboración de un SE y las etapas pueden fundirse, volverse más o menos complejas, dependiendo de la aplicación para el cual se desarrolle el SE.

3. DESARROLLO DEL SISTEMA EXPERTO TUTORIAL DE LÓGICA DIFUSA SETLODI

SETLODI es un Sistema Experto Tutorial de Lógica Difusa, que se enfoca principalmente en las matemáticas difusas, esto es, conjuntos, principios, relaciones, gráficas, análisis y teorías, entre otros, ayudado de las facilidades que ofrece una computadora como elemento para la enseñanza.

3.1 INICIALIZACIÓN DEL PROYECTO

La investigación ha existido y continuará existiendo, puesto que siempre hay algo nuevo por descubrir.

La lógica difusa es un concepto nuevo que empezó a surgir desde 1965, pero que no había sido tan aceptado como lo es actualmente. Podría decirse que la lógica es la formalización del pensamiento, por lo que la lógica difusa surge como un intento de formalización del razonamiento con incertidumbre. En particular y al contrario que otras formas de razonamiento, intenta abordar problemas definidos en términos lingüísticos, y por tanto imprecisos, donde los datos están expresados en términos cualitativos.

El objetivo principal de la lógica difusa es asignar a valores comprendidos entre cierto rango, valores difusos entre 0 y 1, lo que se conoce como grado de pertenencia de la función de pertenencia, por medio de una interpolación lineal. Esto se refiere a que ahora ya no sólo existirán los valores de verdadero y falso ó 1 y 0, sino que se existirán otras opciones que surgen de la interpolación.

El tutorial abarca principalmente la teoría de las matemáticas difusas y ejemplos de las mismas, y con ellas se puede lograr crear un aplicación real, que puede ser tan sencilla o complicada como una quiera.

3.2 EVALUACIÓN DE NECESIDADES

Esta etapa fue una de las más difíciles en este proyecto, primeramente porque como el concepto de Lógica Difusa es todavía algo no muy conocido, no hay mucha bibliografía al respecto, y en los pocos artículos que se publican con respecto al tema, se hace referencia a alguna aplicación muy específica.

3.2.1 SELECCIÓN DEL SHELL DE DESARROLLO

Un "shell" de desarrollo es un ambiente computacional o lenguaje de alto nivel, que presenta una serie de facilidades y herramientas pre-programadas para el desarrollo de Sistemas Expertos.

Se puede decir que un "shell" de desarrollo es la estructura del sistema experto ya que define en términos estructurales el tipo de modelado y la forma del razonamiento del sistema. De esta forma, dependiendo del problema particular que se quiera enfrentar se utilizará uno u otro shell en función de sus características.

Existen dos factores principales que determinan las técnicas que son adecuadas para resolver un problema en particular. Por un lado está, por supuesto, el tipo de problema. Se identifican nueve tipos generales que son: procedural, de diagnóstico, de evaluación, recomendación de acciones, monitoreo, configuración y diseño, distribución de recursos, planeación y calendarización. Por otro lado se encuentra la complejidad del problema: pequeña, mediana o grande.

Con excepción de los problemas procedurales, mientras la complejidad del problema se mantiene reducida, diferentes herramientas son adecuadas para diferentes tipos de problemas, sin embargo, conforme aumenta la complejidad se vuelve necesario el uso generalizado de herramientas híbridas que manejen tanto objetos como diferentes tipos de reglas.

La decisión de qué "shell" de desarrollo deberá utilizarse dependerá, inicialmente, de las herramientas que se requieran y en segundo término, de factores como las facilidades para crear la interfaz con el usuario o la posibilidad de comunicarse con otros tipos de aplicaciones, etc.

Para la elaboración de SETLODI, se analizaron distintos shell's que manejan lógica difusa, primeramente se estudió *Fuzzy Logic Designer*, el cual tiene la capacidad de manejar un gran número de variables tanto de entrada como de salida, estas a su vez pueden ser de diferentes tipos (enteras, reales, booleanas, negativas, etc.).

Uno mismo puede crear las funciones de pertenencia para las variables que esté manejando, aunque se recomienda que estos valores sean proporcionados por un experto, para no caer en incongruencias, así como también se recomienda que él mismo proporcione las reglas con las que trabajará el shell.



Active file: C:\USUARIOS\ANGELES\FLD\DEMO2.FLD

Number inputs: 2

TEMPERATURA	?
TEMPORADA	?

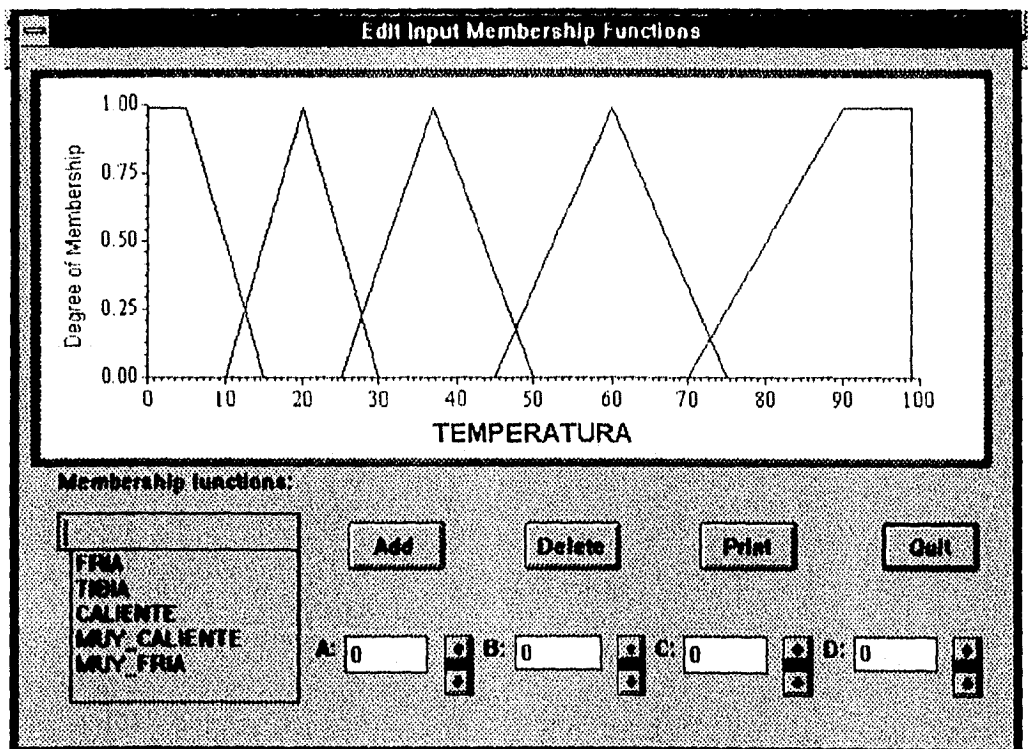
Number outputs: 1

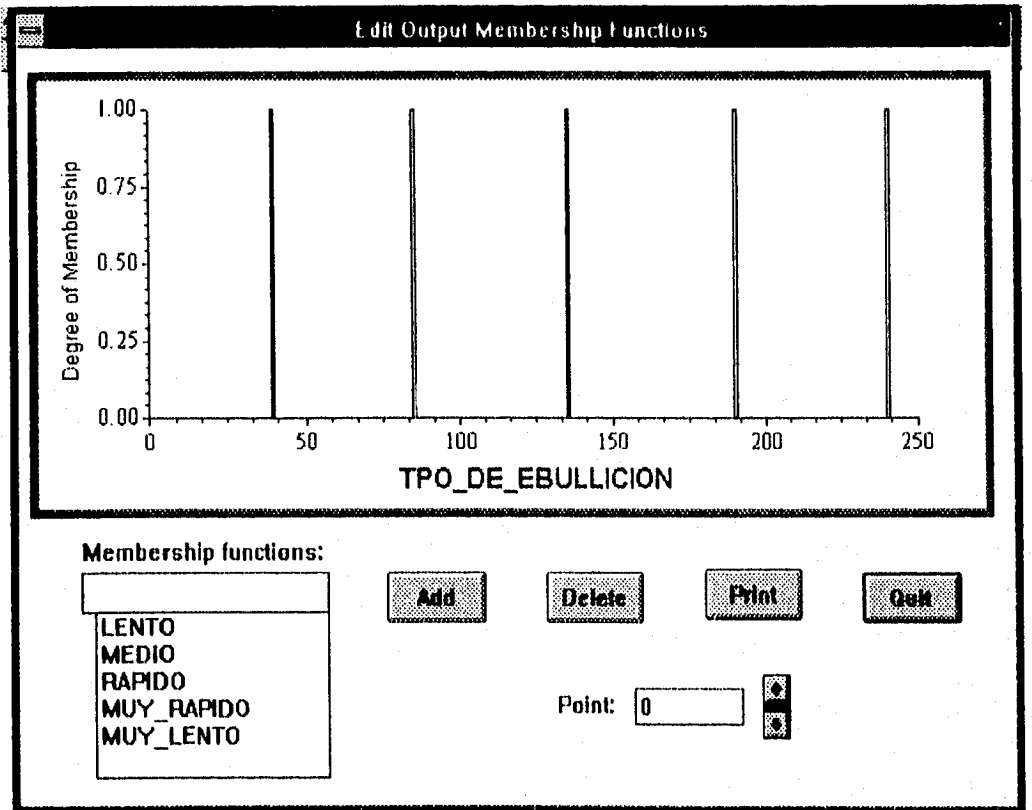
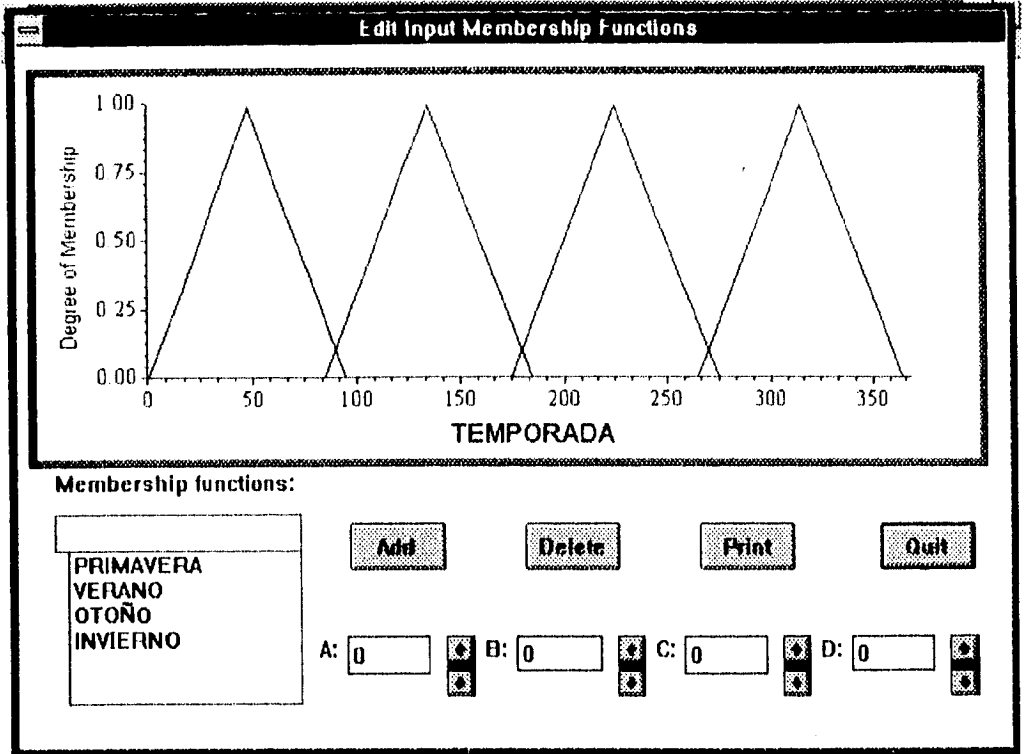
TPO_DE_EBULLICION	?
-------------------	---

Number rules: 20

```

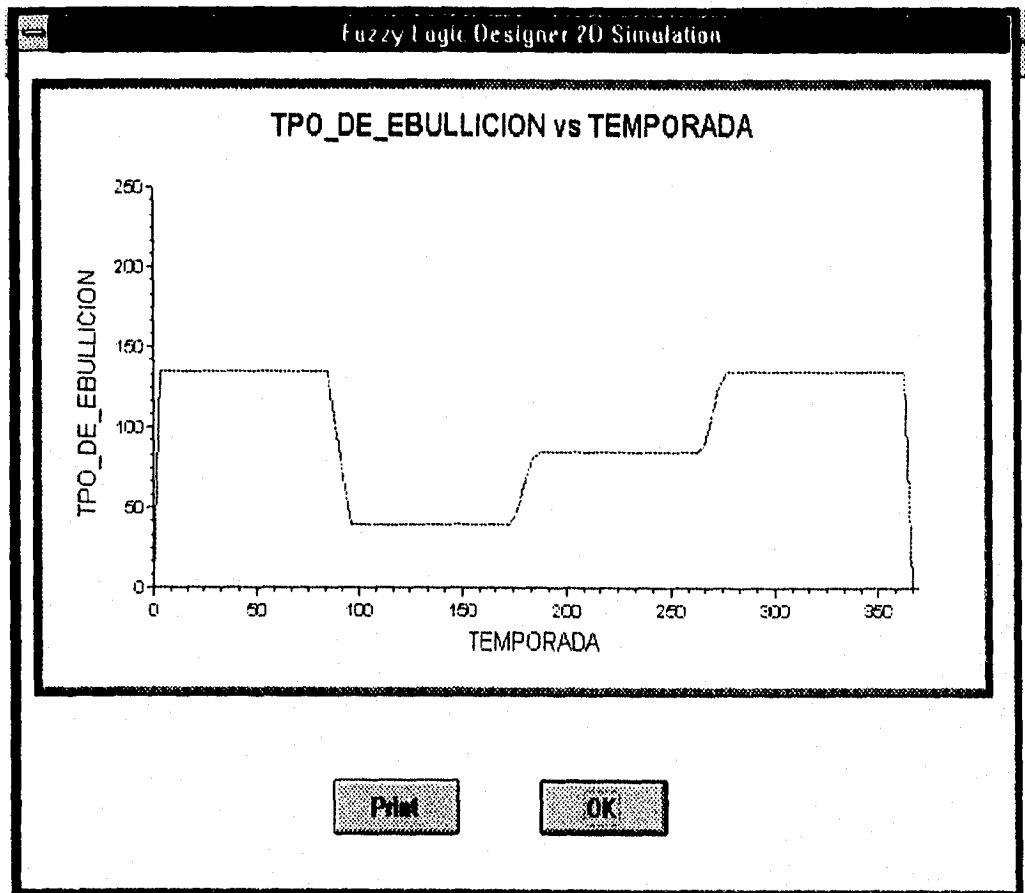
IF TEMPERATURA is FRIA and TEMPORADA is INVIERNO then TPO_DE_EBULLICION is LENTO
IF TEMPERATURA is FRIA and TEMPORADA is OTONO then TPO_DE_EBULLICION is LENTO
IF TEMPERATURA is CALIENTE and TEMPORADA is PRIMAVERA then TPO_DE_EBULLICION is MUY_RAPIDO
IF TEMPERATURA is FRIA and TEMPORADA is VERANO then TPO_DE_EBULLICION is MEDIO
IF TEMPERATURA is MUY_CALIENTE and TEMPORADA is VERANO then TPO_DE_EBULLICION is MUY_RAPIDO
IF TEMPERATURA is TIBIA and TEMPORADA is PRIMAVERA then TPO_DE_EBULLICION is RAPIDO
IF TEMPERATURA is TIBIA and TEMPORADA is VERANO then TPO_DE_EBULLICION is MEDIO
IF TEMPERATURA is TIBIA and TEMPORADA is INVIERNO then TPO_DE_EBULLICION is LENTO
IF TEMPERATURA is FRIA and TEMPORADA is PRIMAVERA then TPO_DE_EBULLICION is MEDIO
IF TEMPERATURA is CALIENTE and TEMPORADA is OTONO then TPO_DE_EBULLICION is MEDIO
IF TEMPORADA is VERANO and TEMPERATURA is CALIENTE then TPO_DE_EBULLICION is RAPIDO
    
```

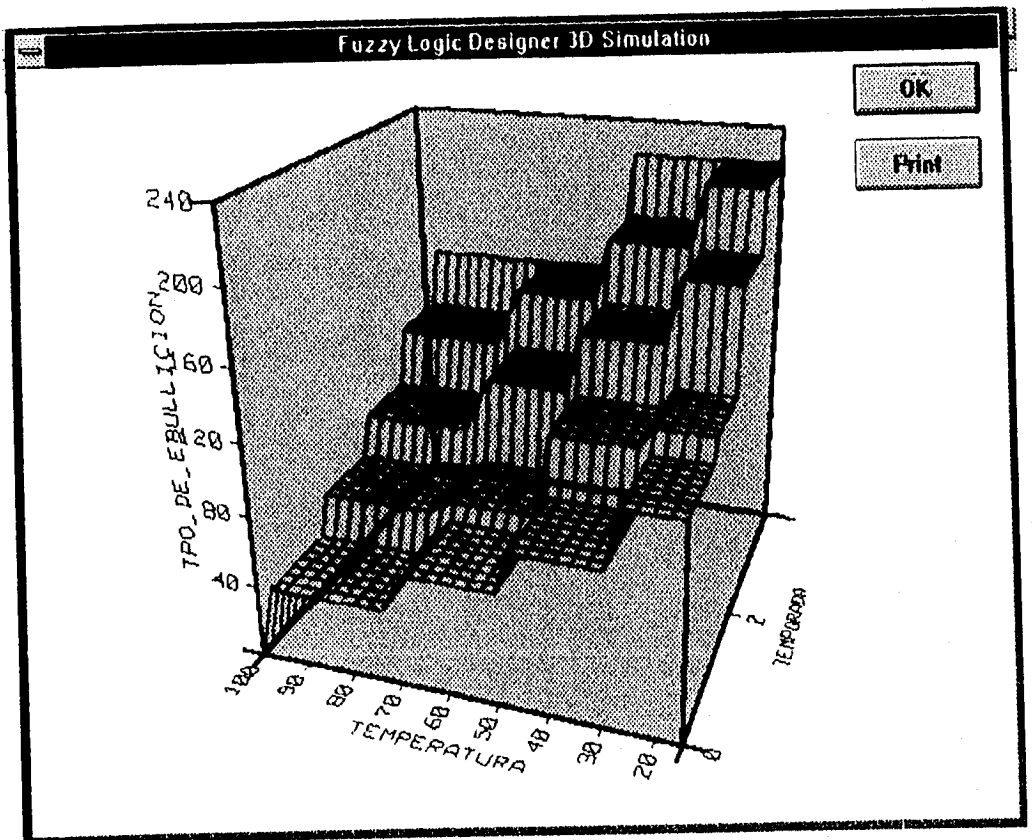
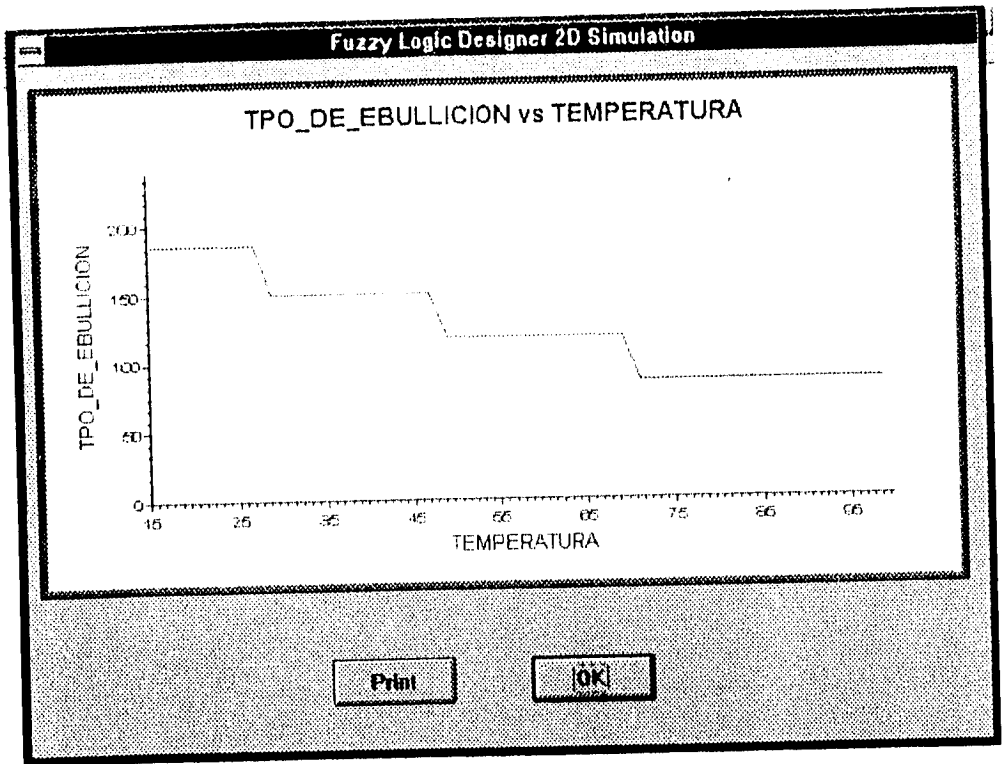




Puede graficar las variables en dos ó tres dimensiones, obteniendo las condiciones iniciales del usuario, o tomarlas aleatoriamente en caso contrario. Cuenta con la opción de generar un código en lenguaje "C" de toda la parte difusa, quedando ésta como una función, de tal manera que pueda llamársele dentro de un programa más completo para que realice la parte difusa de una aplicación específica.

Este shell es útil y práctico para aplicaciones muy sencillas, como comprobación de valores, graficar dos variables para obtener el grado de pertenencia de la función de pertenencia, o graficar en tres dimensiones para visualizar el comportamiento de las variables a las reglas dadas, pero no es un buen shell para realizar sistemas expertos mas complejos, porque su interfaz con el usuario todavía es muy robusta y no es la adecuada para desarrollar la aplicación.

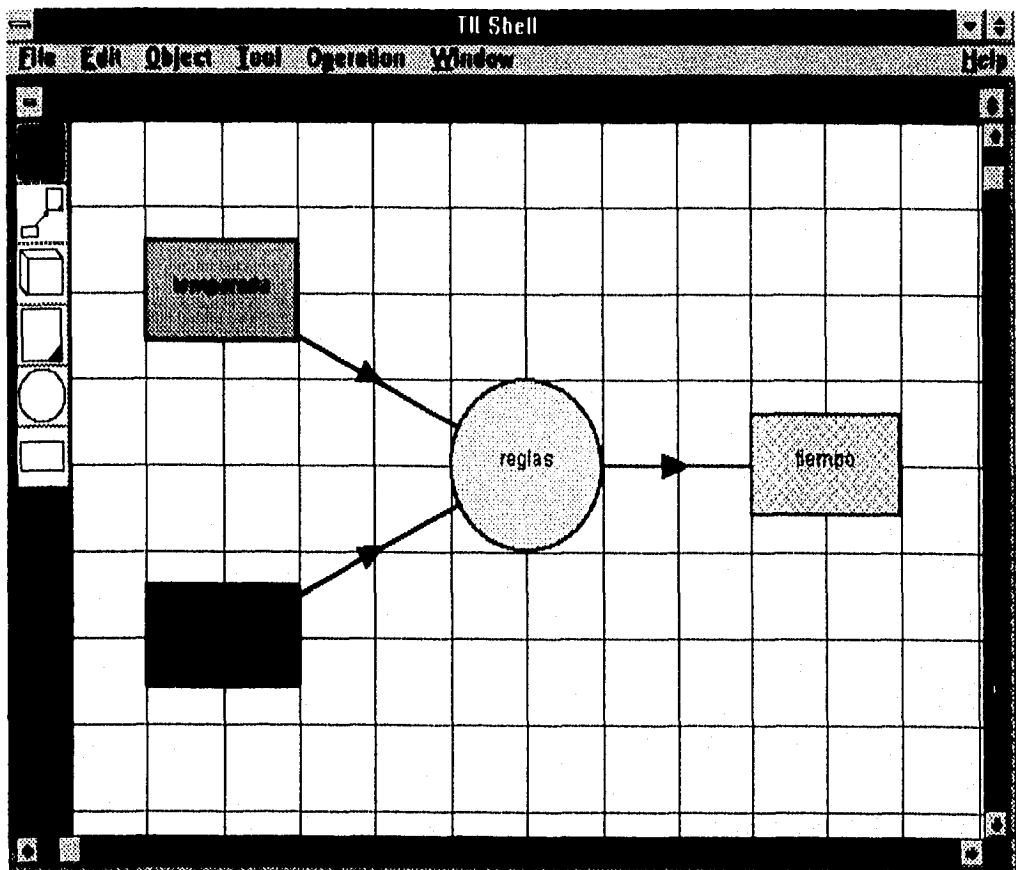




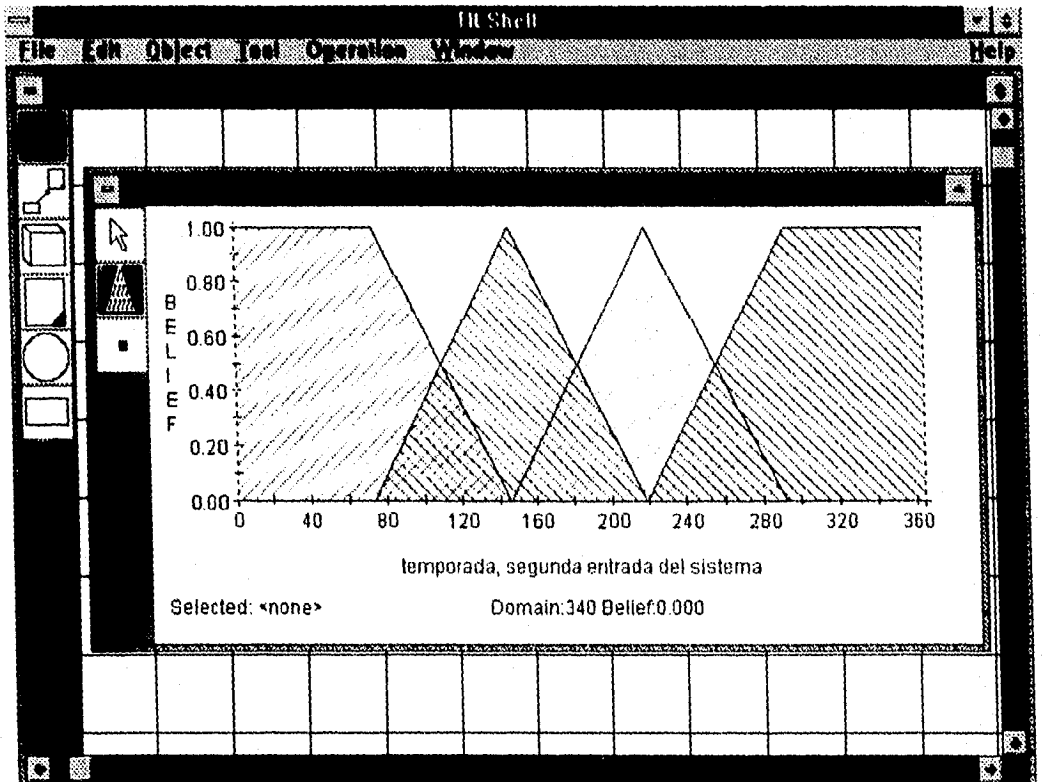
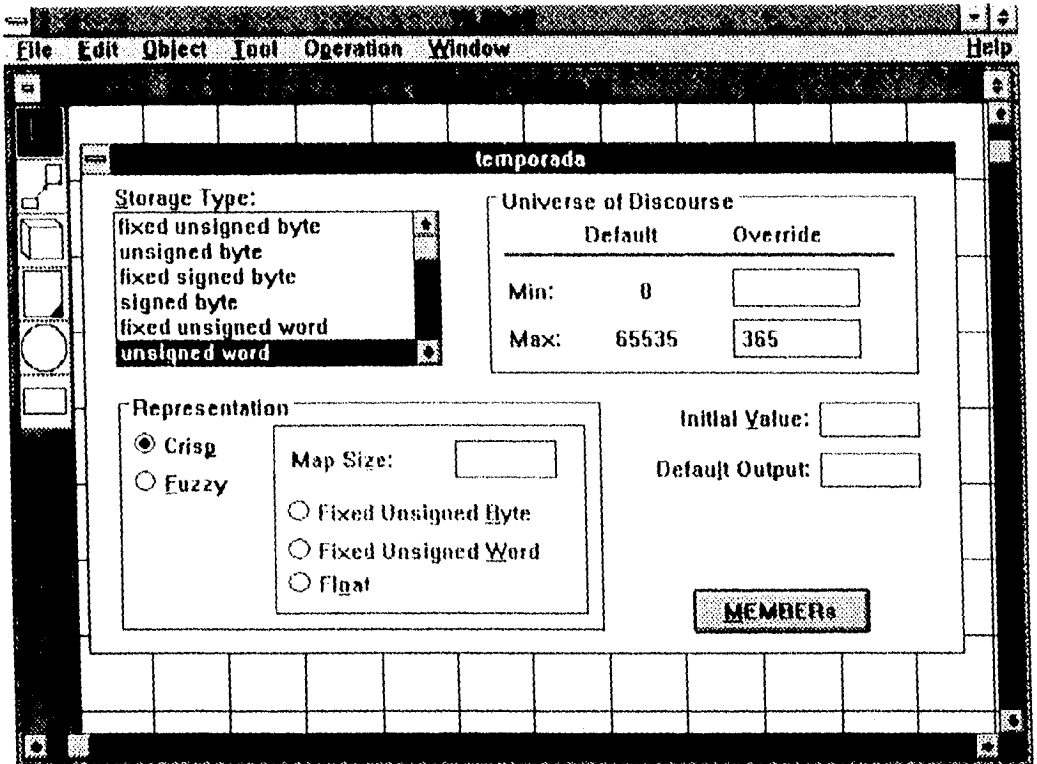
Otro shell que se estudió fue *TilShell*, el cual también maneja lógica difusa y se enfoca a los sistemas expertos. En este shell podemos definir variables de varios tipos, cuyos rangos vienen ya establecidos; para la creación de las funciones de pertenencia nos proporciona dos tipos, que son los más utilizados actualmente por su facilidad de manejo.

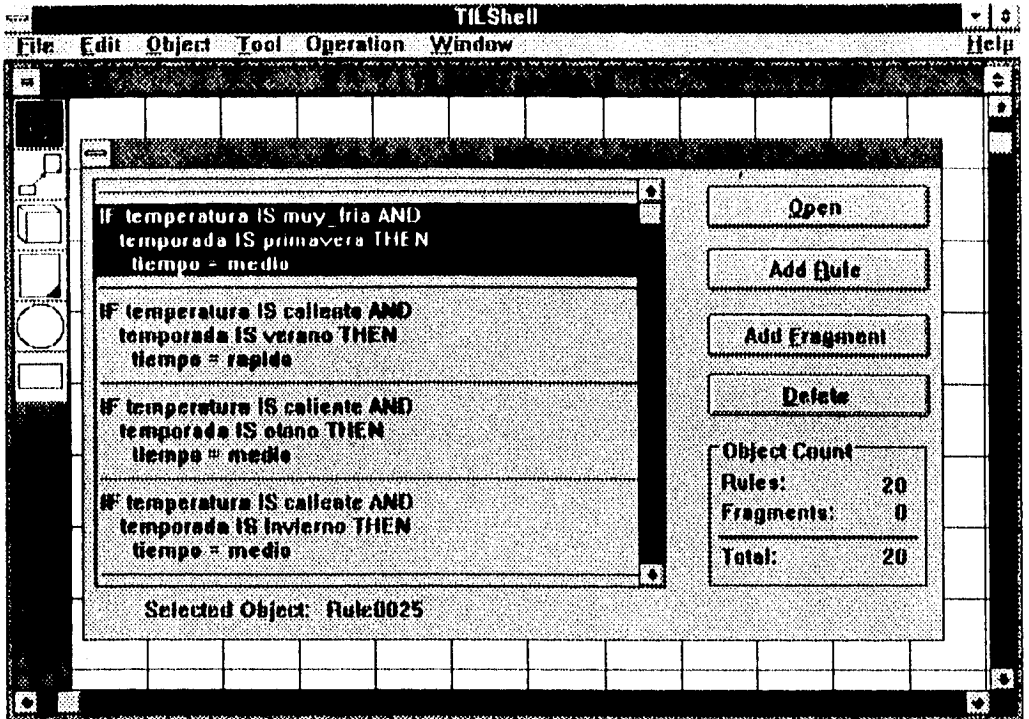
La definición de las reglas es muy sencilla, ya que nos permite visualizarlas al crearlas y así tener un mejor control de las mismas. Genera un código en lenguaje "C", para que sea llamado como una función desde un programa más completo y se entienda mejor la parte difusa.

Por su ambiente gráfico permite que la interfaz del usuario sea más agradable y fácil de manejar, pero este también resultó ser un shell muy robusto y no adecuado para esta aplicación, más sin embargo es de gran ayuda para hacer sistemas expertos sencillos, que manejen 2 ó 3 variables difusas.

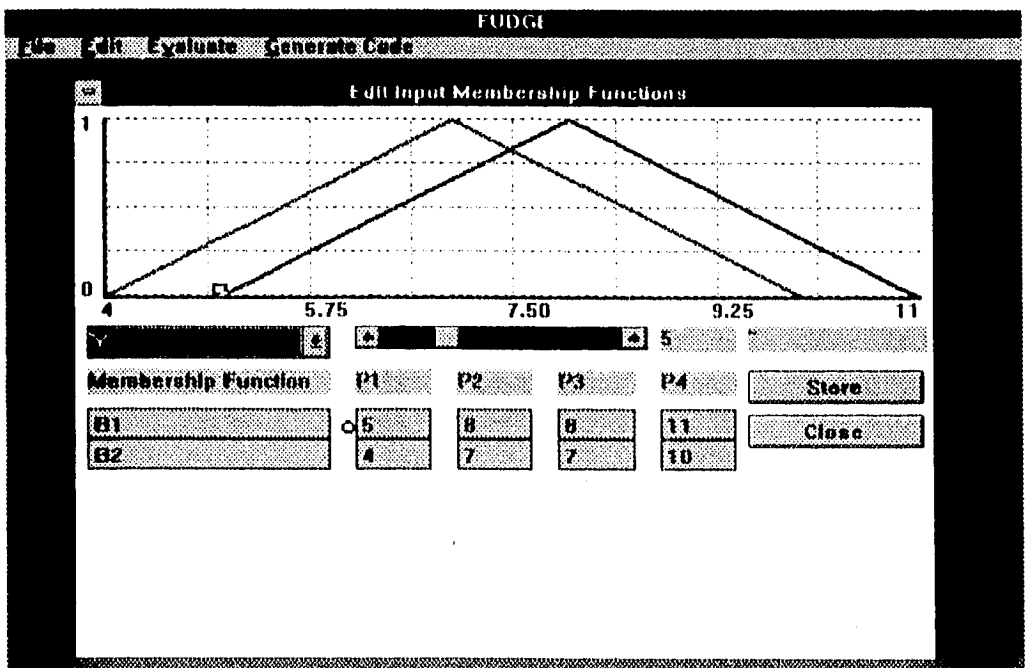


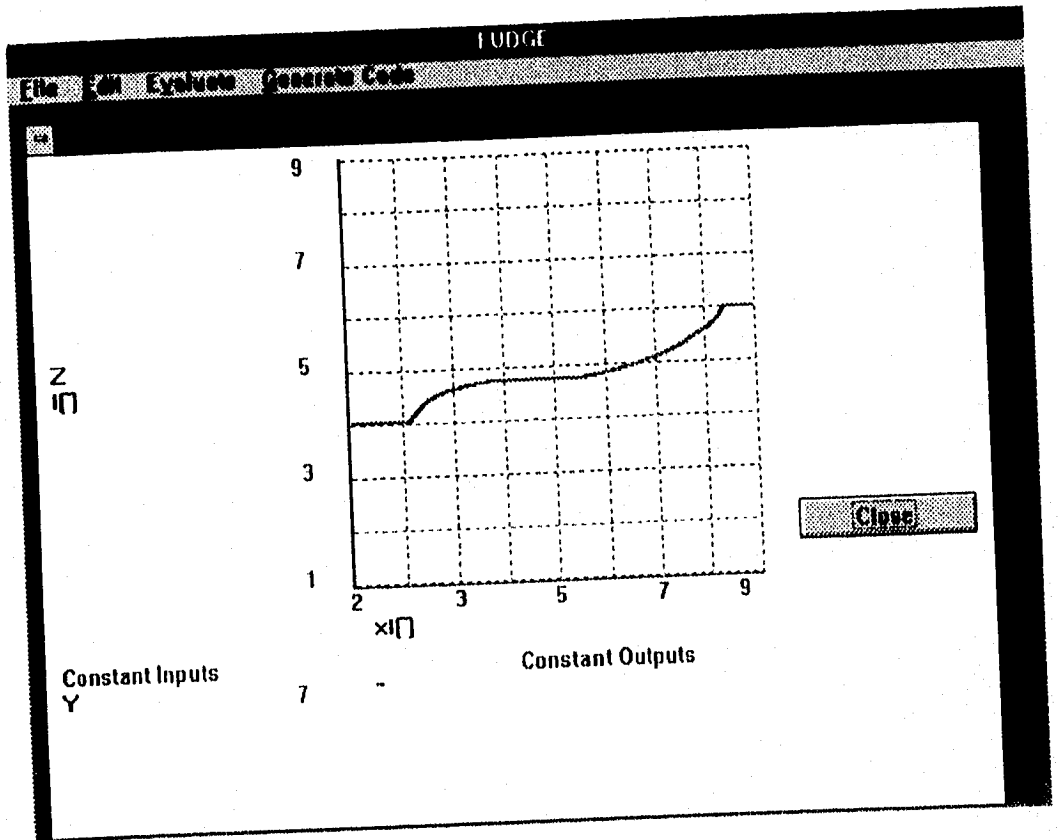
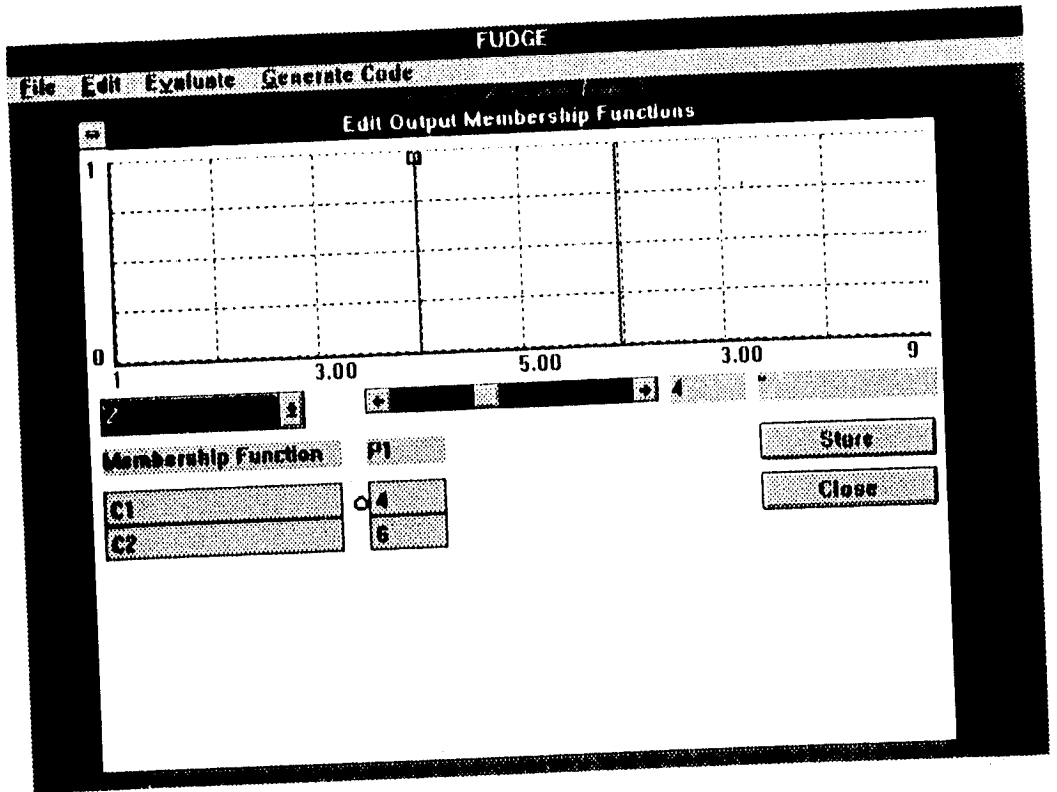
DESARROLLO DEL SISTEMA EXPERTO TUTORIAL SETLODI

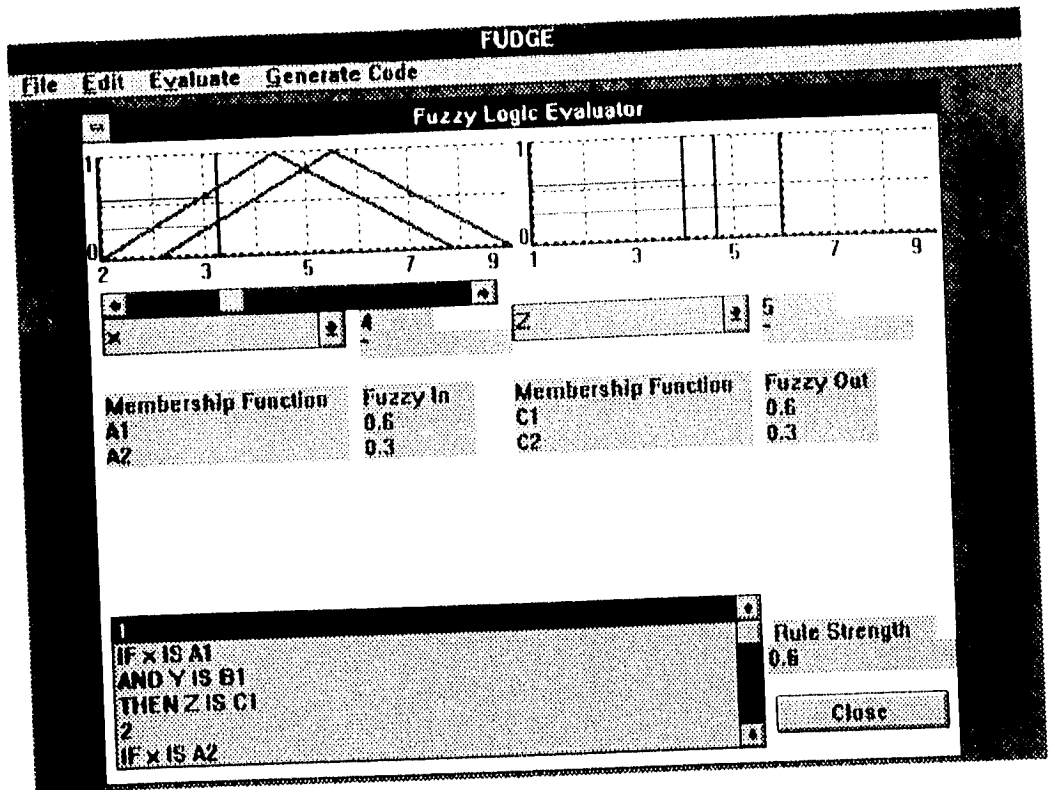




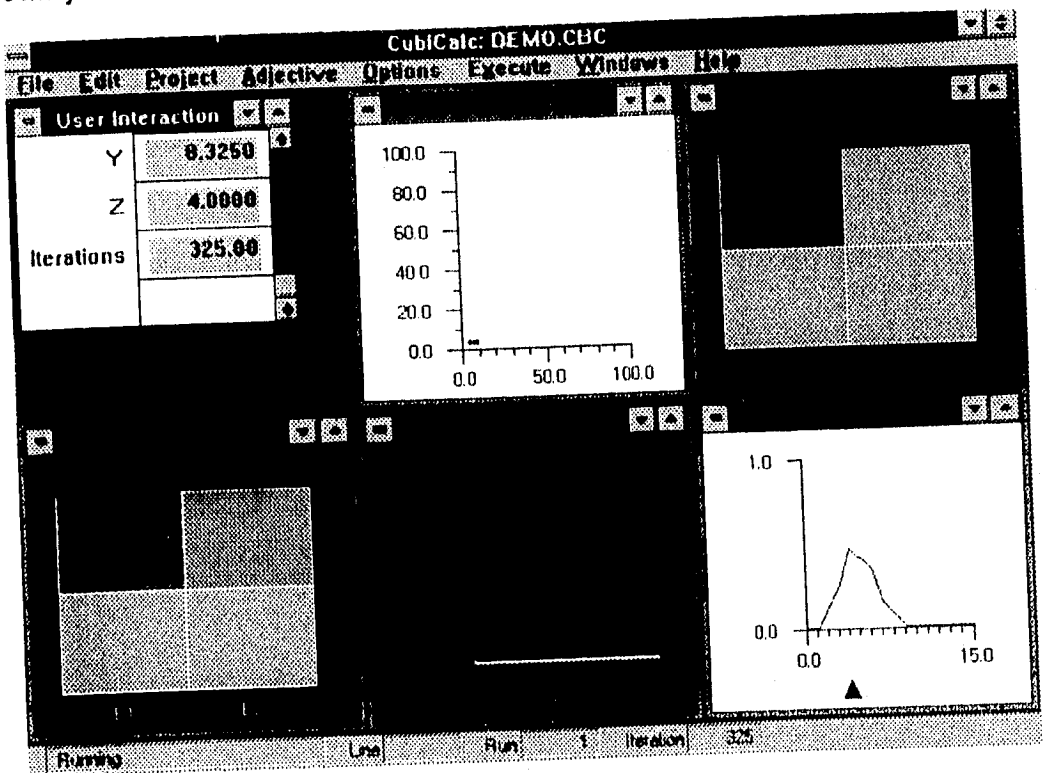
También se manejó *FUDGE* que es un shell muy sencillo ya que su principal objetivo es brindar el grado de pertenencia de la función de pertenencia de una variable.

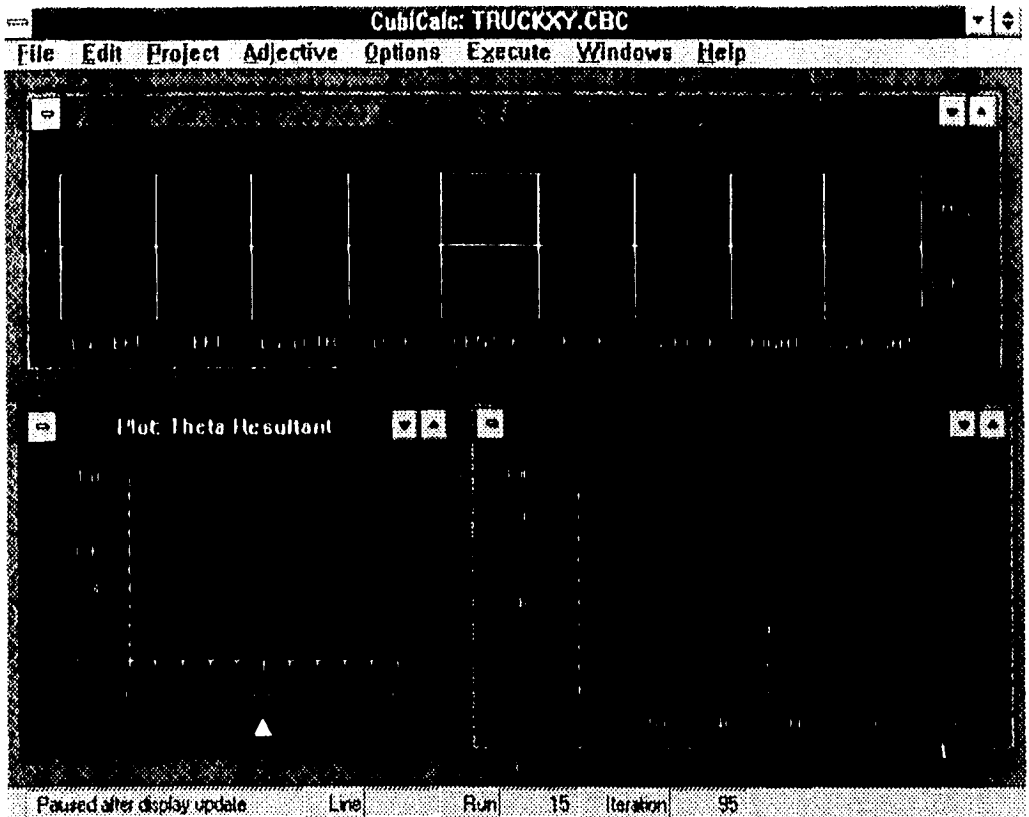






Otro shell fue *CubiCalc*, pero éste no se estudió a fondo, porque es muy semejante a *TilShell*.





Para la decisión de la elección del shell se hizo un análisis del costo-beneficio de las diversas alternativas como fueron la programación en un lenguaje de alto nivel así como software especializado en sistemas expertos. En base a los resultados obtenidos se tomó la decisión de hacer el sistema en un shell que proporcionara el mejor ambiente al usuario por tratarse de un sistema experto tutorial, cuya transmisión del conocimiento debe hacerse de la forma más amigable posible, lo que redundaría en un mayor aprendizaje, minimizando el factor económico que pudiera ocasionar.

Finalmente *el shell elegido* para la elaboración del Sistema Experto Tutorial de Lógica Difusa " SETLODI " fue **LEVEL5 OBJET** ya que es un ambiente de desarrollo de aplicaciones que combina la tecnología de Sistemas Expertos, programación orientada a objetos, modelos de bases de datos y herramientas de depurado. El ambiente que proporciona actúa como mediador gráfico entre el usuario y el programa en sí.

Las características y ventajas principales de LEVEL5 son:

1. Se basa principalmente en la programación orientada a objetos. Esto facilita la ordenación del conocimiento, ya que este se puede dividir en clases, atributos e instancias.

La idea principal de la programación orientada a objetos es tan simple como lo describe Morril [Morris 1989]:

"Nosotros percibimos el mundo como una variedad de objetos; cuando observamos una planta, nosotros miramos una planta, no una masa de átomos individuales. Podemos dividir la planta en hojas, flores, tallo y raíz; pero seguimos viendo esas partes como unidades, como objetos."

"Si subdividiéramos las partes y piezas de la planta en moléculas, siguen agrupadas en diferentes átomos que también percibimos como unidades simples. Para llevar la analogía un paso más, la programación procedural trata los átomos, mientras que la programación orientada a objetos trata a la planta."

Las realizaciones software de objetos del mundo real están categorizadas de la misma forma. Todos los objetos son miembros de una *clase* más amplia, y *heredan* la estructura de datos privada y las operaciones que se han definido para esa clase. Es decir, una clase es un conjunto de objetos que tienen las mismas características. Un objeto individual es por tanto una *instancia* de una clase más amplia.

El uso de clases y herencia es crucialmente importante, la *reutilización* de componentes de programas se consigue creando objetos (instancias), que se construyen sobre los atributos y operaciones existentes que se heredan de una clase. Sólo se necesita especificar en qué difieren los nuevos objetos de los de la clase, en lugar de definir todas las características del nuevo objeto.

La definición de objetos y operaciones es una buena forma para comenzar el análisis de los dominios funcionales y de información. Un objeto puede verse como un elemento de información en una operación, como un proceso-función que se aplica a uno o más objetos.

El método de análisis orientado a objetos puede describirse de la siguiente forma:

- a) El software asignado (Level 5) se describe utilizando una estrategia informal. La estrategia no es más que una descripción en lenguaje natural de la solución del problema que se tiene que resolver mediante el software representado a un nivel consistente de detalle.
- b) Los objetos se determinan subrayando cada nombre o cláusula nominal e introduciéndolos en una tabla sencilla. Deben anotarse los sinónimos. Si se requiere que el objeto se implemente como una solución, entonces es parte del espacio de solución; en otros casos, si un objeto es necesario sólo para describir una solución, es parte del espacio del problema.

- c) Los atributos de los objetos se identifican subrayando todos los adjetivos y luego asociándolos con sus objetos respectivos (nombres).
- d) Las operaciones se determinan subrayando todos los verbos, frases verbales y predicados (una frase verbal indica un Test condicional) y relacionando cada operación con el objeto apropiado.
- e) Los atributos de las operaciones se identifican subrayando todos los adverbios y luego asociándolos con sus operaciones respectivas (verbos).

Puede observarse que el análisis orientado a objetos suministra al analista un mecanismo sencillo para representar los elementos clave en el dominio de la información y las funciones clave en el dominio funcional. Cada elemento y función puede ser subdividido más adelante y aplicados de nuevo, de forma iterativa, los pasos del análisis.

- 2. LEVEL5 es lo que se conoce como un sistema híbrido, es decir, que puede manejar tanto objetos como reglas. De este modo el conocimiento puede ser conducido de manera muy similar a como es manejado por el ser humano; por un lado se tienen un conjunto de objetos estructurados y jerarquizados dentro de clases, un conjunto de reglas mediante las que podemos razonar sobre esa representación del conocimiento, y llegar a nuevas condiciones que modifiquen los objetos anteriores y que nos permitan resolver problemas.
- 3. Su interfaz gráfica es altamente desarrollada. No se requiere casi teclear nada, ni al programar ni cuando corremos el sistema, ya que tanto clases, objetos y atributos como funciones, símbolos y comandos, pueden ser obtenidos e integrados a reglas, demonios o métodos por medio del mouse. Además permite desarrollar aplicaciones de gran atractivo para el usuario como son las animaciones.
- 4. Cuenta con la capacidad de manejar información de manera no secuencial y de acuerdo al nivel de conocimiento y a los requerimientos del usuario.
- 5. Capacidad de interactuar con bases de datos relacionales, como es el caso de Dbase III Plus, que en cuestión de códigos de diseño son muy útiles ya que cuando éstos son modificados, basta con cambiar los datos de la bases que maneje el sistema para actualizarlo.

6. La máquina de inferencia puede trabajar de varias maneras: hacia atrás (utilizando reglas), hacia adelante (usando demons) o con encadenamiento mixto (combinación de los dos anteriores).
7. Confiabilidad, pues existen ya un gran número de aplicaciones desarrolladas en este shell.

3.2.2 TIPO DE INFERENCIA UTILIZADA

El tipo de inferencia que se usó fue la de *forward chaining* o encadenamiento hacia adelante, ya que se requiere de un razonamiento guiado por los datos o deductivo, pues cada estudiante es diferente (algunos repetirán lecciones mientras otros avanzarán rápidamente).

A partir de los datos de entrada (como el contraseña del usuario), el sistema decidirá que conocimientos emplear, de tal manera que el resultado que arroje, una decisión, corresponda a los datos de entrada requeridos para tomar la siguiente decisión.

Fuentes de conocimiento utilizadas

Para la adquisición del conocimiento se contó con la colaboración del experto en el tema, el Dr. Felipe Lara Rosano y con la asesoría del M.I. Nicolás Kemper Valverde, ellos fueron la fuente principal de conocimientos tanto en la parte educativa, es decir, en la manera en que se imparten y evalúan los temas; como en la información y contenido de los mismos. La información de cada uno de los 8 temas que comprende el tutorial fue tomada en su base de el libro: *Fuzzy Set Theory and Its Applications*.

Este SET no está basado completamente en el conocimiento de un experto humano, pero mucho del contenido de los temas son experiencias propias del experto y de el asesor. Las evaluaciones y exámenes de cada tema, fueron realizados especialmente por el experto y constituyen también parte importante del tutorial.

3.2.3 REQUERIMIENTOS TÉCNICOS DEL SISTEMA EXPERTO

Para poder hacer uso de este sistema se debe contar con una computadora personal con las siguientes características como mínimo:

Hardware:

- Procesador 80386 a 33 MHz. ó superior
- Monitor Color VGA
- Teclado

- Mouse
- 4 Mbytes en RAM como mínimo
- Espacio en disco duro para el Sistema Experto SETLODI y su ejecución de 20 Mbytes mínimo
- Drives de 3 ½ " de alta densidad

Software:

- MS-DOS 5.1 ó superior
- Windows 3.1 y superior
- LEVEL5 OBJET versión 2.5 (El sistema es compatible con versiones posteriores utilizando la correcta adaptación).

3.3 IMPLEMENTACIÓN

3.3.1 DISEÑO DE LA INTERFAZ DEL USUARIO

La interfaz del usuario es completamente gráfica, de tal manera que la navegación por las lecciones se realiza completamente por medio de botones y pussbutton, los cuales se activan haciendo un *click* en él con el mouse.

El diseño de la interfaz del usuario significó un gasto de mucho tiempo puesto que a la interfaz se le tuvo que rediseñar en varias ocasiones. Los colores usados para fondos, pantallas, ejemplos y botones, así como también el tipo de letra, tipo de botones y la ubicación de ventanas de texto, gráficas, tablas y dibujos, fueron examinados por un Lic. en Diseño Gráfico y por un Lic. en Comunicación Gráfica. Amigos y voluntarios que probaron la interfaz, mostraron entusiasmo e interés en el uso del tutorial, así como también valiosas opiniones acerca del mismo, y que al conjuntar éstas, dio como resultado la interfaz del usuario que presenta SETLODI.

Los botones, ilustraciones, pinturas y fondos de pantalla fueron elaborados en Paintbrush con ayuda en algunos casos y detalles de Visual Basic Profesional.

3.3.2 ESTRUCTURA GENERAL DEL SISTEMA SETLODI

Todos los temas están interrelacionados en forma secuencial, sin embargo el diseño y estructura de los mismos tendrán cierta independencia en cuanto a su empleo y secuencia de aprendizaje, pero manteniendo características de herencia de la información obtenida en alguno de ellos y que pueda ser empleada para fines comparativos en cualquiera de los restantes. Con ello se estaría garantizando flexibilidad, mucho dinamismo y fácil repetitividad en el proceso de capacitación.

Una de las causas por las cuales el sistema se reestructuró manejando los diversos temas en distintos módulos es debido a la importancia del concepto de modularidad en el software de computación, ya que es el atributo más sencillo que permite a un programa ser manejable intelectualmente. El software monolítico (es decir, un gran programa compuesto de un único módulo) no puede ser fácilmente abarcado por otros programadores. El número de caminos de control, la expansión de las referencias, el número de variables y la complejidad global podría hacer imposible comprenderlo bien.

La independencia funcional se obtuvo desarrollando módulos por tema ya que cada módulo tiene una clara función: mostrar el contenido de un tema específico, y no se tiene una excesiva interacción con otros módulos (la única forma de acceder a cada tema es por medio de uno de ellos). De ahí que sea fácil de desarrollar cada tema, ya que su función puede ser compartida con sencillas interfaces. Los módulos independientes son fáciles de mantener debido a que se limitan los efectos secundarios causados en las modificaciones de diseño/código, se reduce la programación de errores y es posible reutilizar los módulos.

La independencia se mide usando dos criterios cualitativos: cohesión y acoplamiento. La cohesión es una medida de la fuerza funcional relativa de un módulo. El acoplamiento es una medida de la interdependencia relativa entre módulos.

Un módulo coherente ejecuta una tarea sencilla en un procedimiento de software y requiere poca iteración con procedimientos que se ejecutan en otras partes de un programa. Dicho sencillamente, un módulo coherente sólo debe hacer (idealmente) una cosa.

El acoplamiento es una medida de la interconexión entre módulos en una estructura de programa. En el diseño del sistema se buscó el más bajo acoplamiento posible, puesto que la conectividad sencilla entre módulos da como resultado un software más entendible y menos propenso a errores, que ocurren en una posición y que se propagan a lo largo del sistema.

Tomando en cuenta lo anterior, se elaboró la secuencia de validación de los temas expuestos en el diagrama 4.1, basándose en el análisis y experiencias de los expertos en Lógica Difusa.

El tutorial se compone de 4 partes principales, conformadas por 20 bases de conocimiento, es decir archivos KNB, los cuales están ligados por medio de una base de datos con formato Dbase III, ya que éste es el formato con el que trabaja LEVEL5 OBJET 2.5:

- Recepción de datos
- Definiciones y ejemplos

- Exámenes
- Aplicaciones

La recepción de datos se emplea tanto para guardar el registro del usuario y moverlo dentro de SETLODI, como para manejar la base de datos. El manejo de la base de datos es únicamente permitido para el ingeniero de conocimientos y para el experto, ya que les permite borrar registros, reemplazar datos de un registro en particular e insertar registros.

En la recepción de datos es en donde SETLODI decide a dónde va mandar al alumno en ese momento, por lo que el tutorial siempre iniciará en esta parte y requerirá de la contraseña del usuario para colocarlo en el tema que le corresponda, según haya avanzado en el tutorial la última vez que entró.

La parte de definiciones y ejemplos contiene todo el conocimiento de las matemáticas difusas, las principales definiciones, funciones y teoremas entre otros, constanding de texto informativo, tablas, gráficas y ejemplos en el pizarrón. Las definiciones están divididas en 7 temas que son de manera secuencial, esto es, que el alumno sólo tendrá acceso a aquellos temas anteriores en los que su examen fue acreditado, esto es con el fin de evitar que emplee material nuevo sin los antecedentes requeridos.

En cada tema, el alumno puede avanzar y retroceder cuantas veces sea necesario, así como también regresar al menú principal y salir del tutorial. El menú será diferente para cada alumno en cuanto a la habilitación de los botones de los temas, esto es debido al avance de cada alumno. La habilitación de un botón se identifica porque está iluminado en un tono azul claro, en tanto que aparecerá en tono gris cuando este se encuentre deshabilitado.

Para acreditar un tema, este cuenta con un examen al final del mismo, el cual debe contestar el alumno correctamente casi en su totalidad para acreditarlo con una calificación igual o superior a ocho en una escala del 1 al 10 para poder tener derecho a pasar al siguiente tema, en caso contrario al tutorial automáticamente lo regresa al mismo tema.

3.3.3 PROCESO DE EVALUACIÓN

El proceso de evaluación contempla las siguientes alternativas:

- a) Las evaluaciones formativas podrán ser: cuestionarios de preguntas. Para estas evaluaciones, existe la posibilidad de hacerlas cada cierto tema. El sistema a través de un proceso inteligente de retroalimentación, recomendará al usuario-alumno los temas que debe volver a estudiar.

- b) La evaluación contempla la integración de: cuestionario de preguntas. Esta evaluación se ejecuta en forma conjunta al final de todos los temas con excepción del último.
- c) Los exámenes tienen un grado de dificultad adecuado al perfil del usuario de acuerdo a los expertos. Cada examen consta de preguntas que son de opción múltiple, es decir, la respuesta correcta se proporciona como alternativa a escoger.

Por otra parte, las preguntas son de opción múltiple debido a la ambigüedad que se tendría si el usuario proporcionara la respuesta, incluyendo que la validación de las respuestas del usuario no sería objetiva por diversos factores como: el teclear mal una palabra, extenderse en la respuesta, etc., que no podrían considerarse en la evaluación. Así mismo, el poder elegir una respuesta entre varias posibilidades implica facilidad tanto para contestar como para evaluar, aunado al esfuerzo lógico y no solo del conocimiento que el usuario deberá tener para contestar el examen.

3.3.4 BASES DE DATOS

Actualmente los sistemas que requieran mantener información para la retroalimentación del mismo, así como la toma de decisiones, necesitan almacenar datos que ayuden a la distribución de la información y al mejor control sobre las evaluaciones del sistema. De acuerdo a los requerimientos del tutorial que comprenden considerar que el alumno pudiera continuar con el aprendizaje del tema cuando lo deseara, incluyendo el poder repasar la última lección vista, fue necesario contemplar ciertos datos del alumno e información del avance en el tutorial para llevar a cabo lo anterior.

Para el efecto, fue necesario almacenar el valor de ciertas variables específicas del tutorial como son: tema, total de temas, último tema visto, calificación de cada uno de los exámenes presentados, datos del alumno (clave referencial, apellidos y nombre).

Level 5 Objet, versión 2.5, maneja dos tipos de manejadores de bases de datos: Dbase III y FOCUS. Dada la familiaridad que se tiene con el primero de ellos y la facilidad para manejar bases de datos entre otras características, se optó por implementar la base de datos en Dbase III.

La base de datos ha de ser de alguna forma accesible al usuario, ya que es de lo que se ocupa el lenguaje de programación Dbase III PLUS.

El Dbase III PLUS puede manejar 1,000 millones de registros y hasta 128 campos de registro. Cada registro puede contener hasta 4,000 caracteres. Puede ordenar varios campos a la vez y trabajar hasta con 10 archivos a la vez.

Estas características tan poderosas son más impresionantes que realistas. Unos simples cálculos indican que si una base de datos contiene 1,000 millones de registros de 128 campos cada uno, a una PC le llevaría más de un mes el leer una sola vez la base de datos y dicha base de datos ocuparía 30 millones de diskettes. Pero por otro lado, el Dbase III PLUS ordena muy rápido: la ordenación de un archivo lleva menos de 60 segundos. Además permite el acceso de múltiples usuarios al programa cuando se trabaja en una red de área local. Las posibilidades de protección de datos de que dispone el programa se pueden usar con las más populares redes de área local para el IBM PC y compatibles.

Limitaciones: Las limitaciones de Dbase son pocas, pero de hecho existen. La mayor de ellas es que necesita a menudo información que está contenida en disco, para poder operar. Esto puede disminuir la velocidad de sus programas debido al acceso al disco. Los retrasos son más evidentes en un sistema basado en diskettes que en un sistema con disco duro.

Requerimientos de Dbase: Para usar el Dbase se necesita una computadora de 16 bits con MS-DOS o PC DOS. La computadora debe tener un mínimo de 256K de memoria siendo recomendable disponer de 320K o más para un uso más eficiente. Se debe usar la versión 2.0 del MS-DOS o PC DOS, o una posterior. El sistema debe tener dos unidades de disco flexible o una unidad de disco flexible más un disco duro. (Debido al tamaño de los archivos del disco del sistema del Dbase no es práctico el uso del Dbase en un sistema con una única unidad de disco flexible). Se puede usar cualquier impresora que imprima como mínimo 80 columnas de texto.

3.4 DEPURACIÓN DEL SISTEMA

3.4.1 OPERACIÓN GENERAL

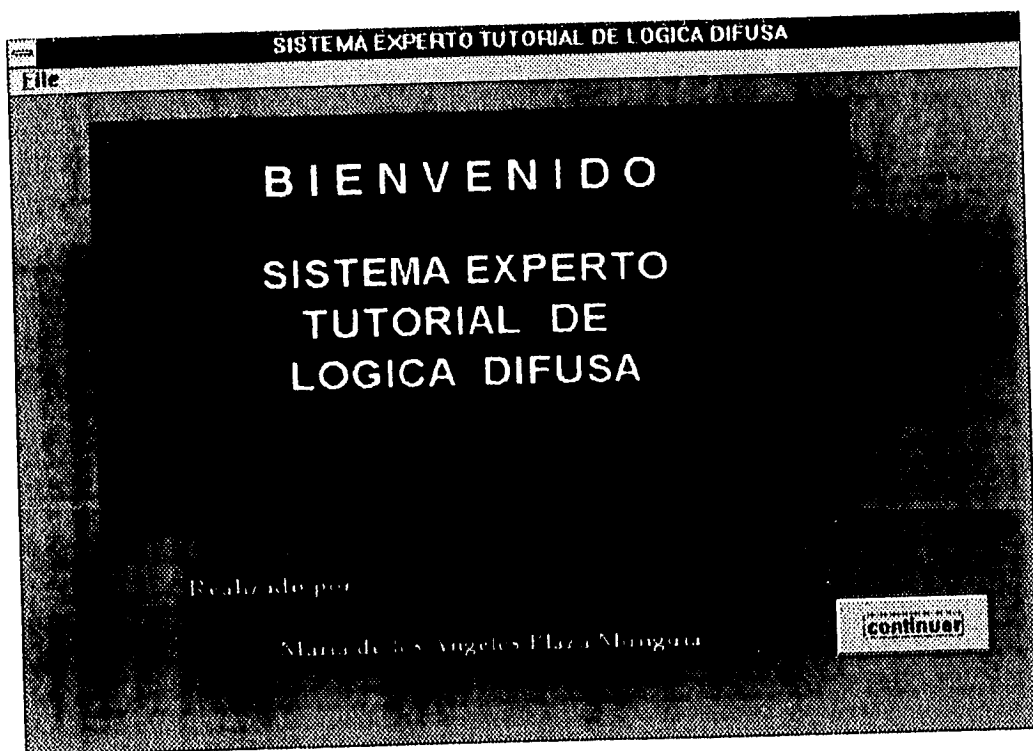
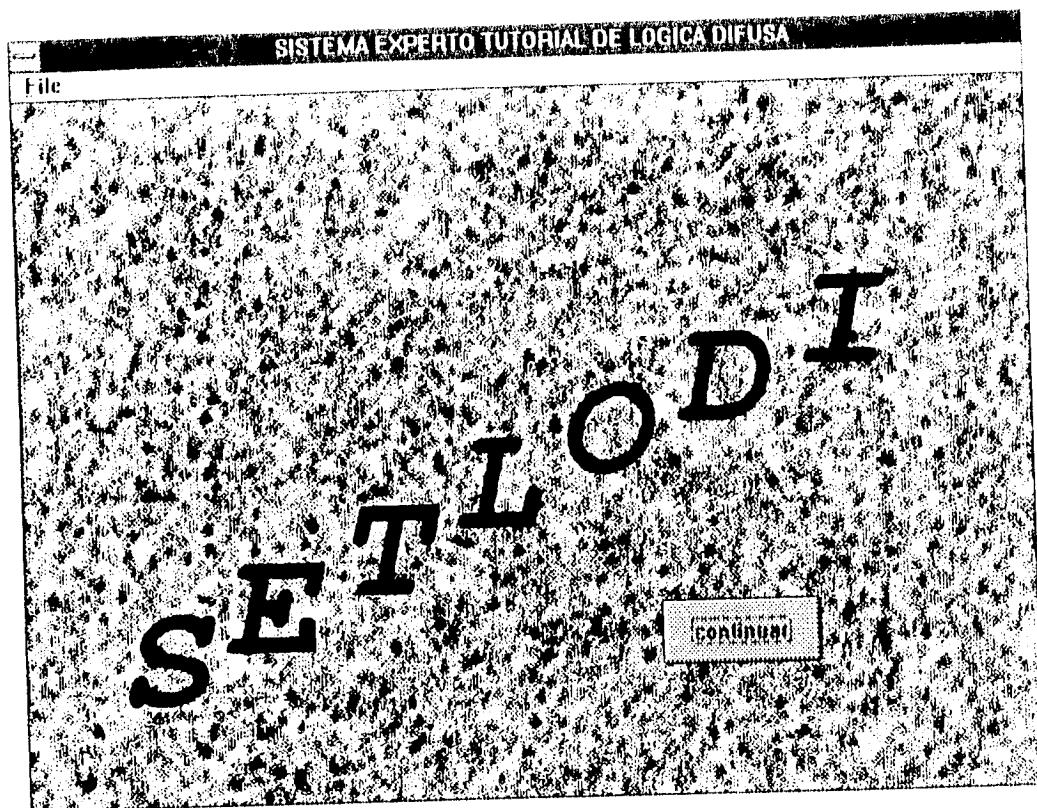
La operación de SETLODI es muy fácil e intuitiva debido a su interfaz del usuario.

La forma en como operan los botones y las pantallas de los temas se encuentra explicada en la parte de INSTRUCCIONES a la cual el alumno entra una vez que ya se inscribió en el tutorial.

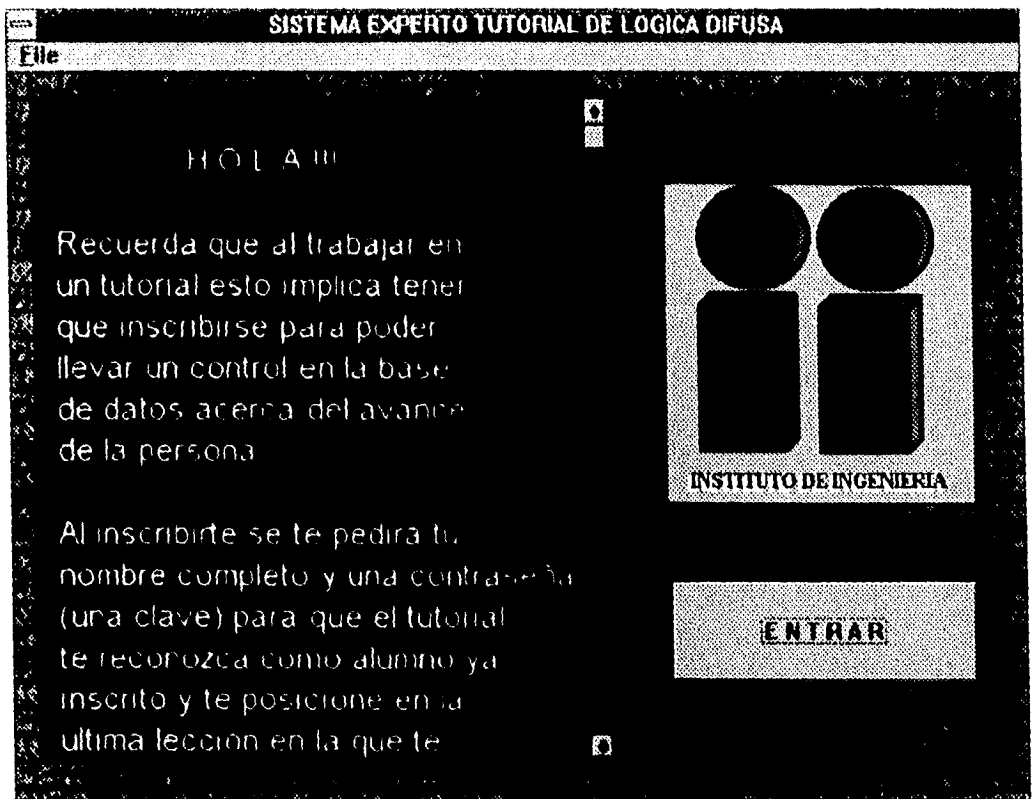
A continuación se explica la manera en cómo SETLODI funciona pantalla por pantalla antes de entrar al menú de temas para iniciar con las definiciones.

Las pantallas que se muestran a continuación, son la presentación del sistema, ambas cuentan con un pushbutton que dice continuar, el cual debe oprimirse para pasar a la siguiente pantalla.

Pantallas de presentación del sistema

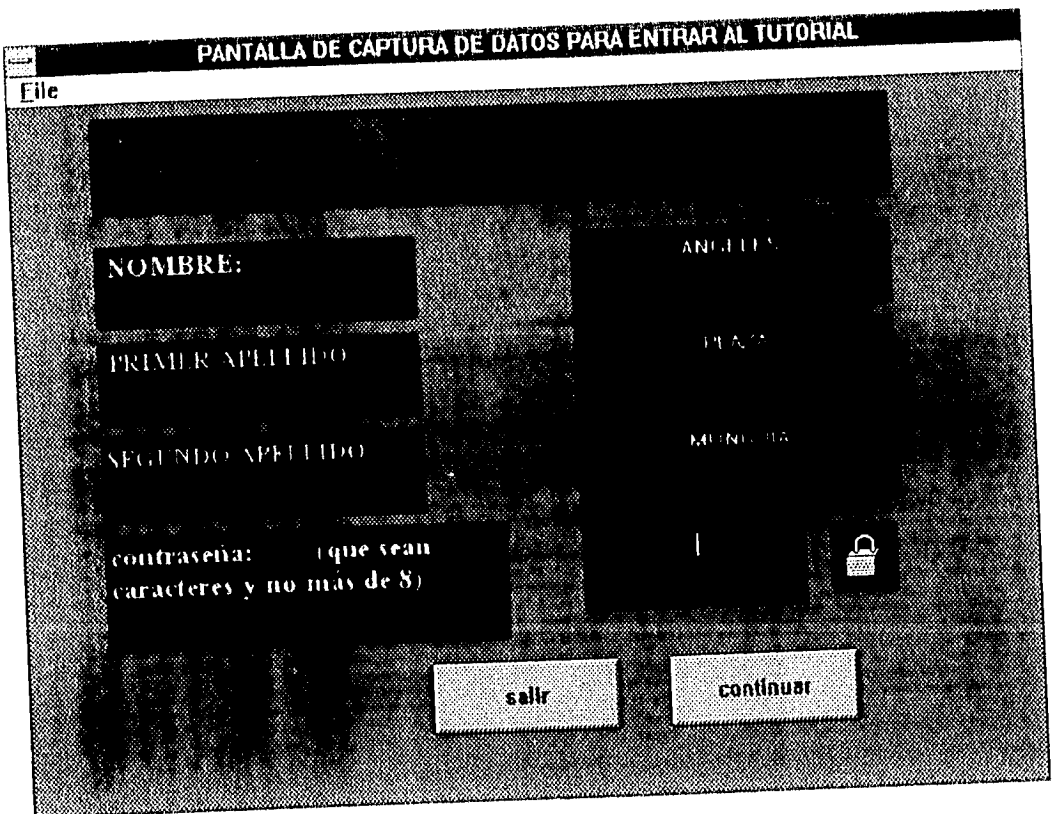
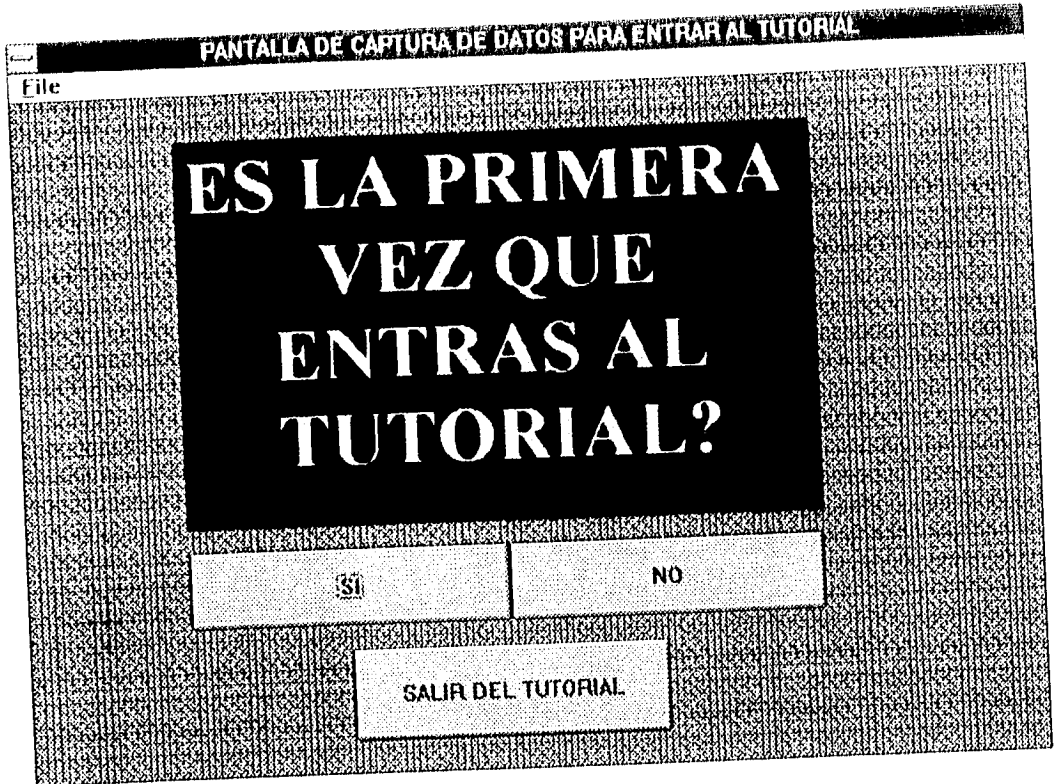


La pantalla siguiente, es en donde el tutorial le da la bienvenida al usuario y en ella se le indica como inscribirse al tutorial, en el caso de ser la primera vez que entra a el.



Pantalla de instrucciones iniciales

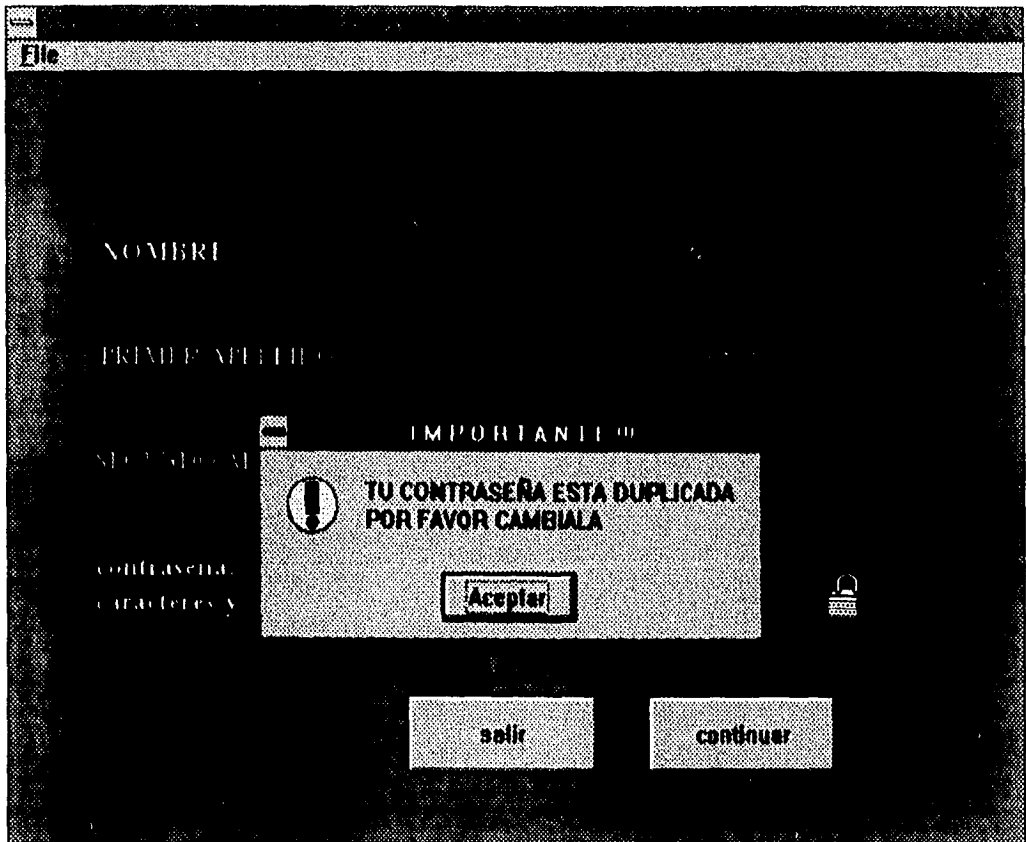
La siguiente pantalla aparece después de la bienvenida, en ella se le pregunta al usuario si es la primera vez que entra al tutorial para entonces enviarlo a la pantalla de inscripción, en donde debe proporcionar sus datos completos, esto es nombre, apellido paterno, apellido materno y una contraseña con la que el sistema lo va a identificar cada vez que accese al tutorial, en caso contrario, es decir, que sea la primera vez que entra al tutorial, el sistema lo manda a otra pantalla en la cual sólo le pedirá su contraseña para que pueda continuar con las definiciones correspondientes al último tema en el que se quedó.

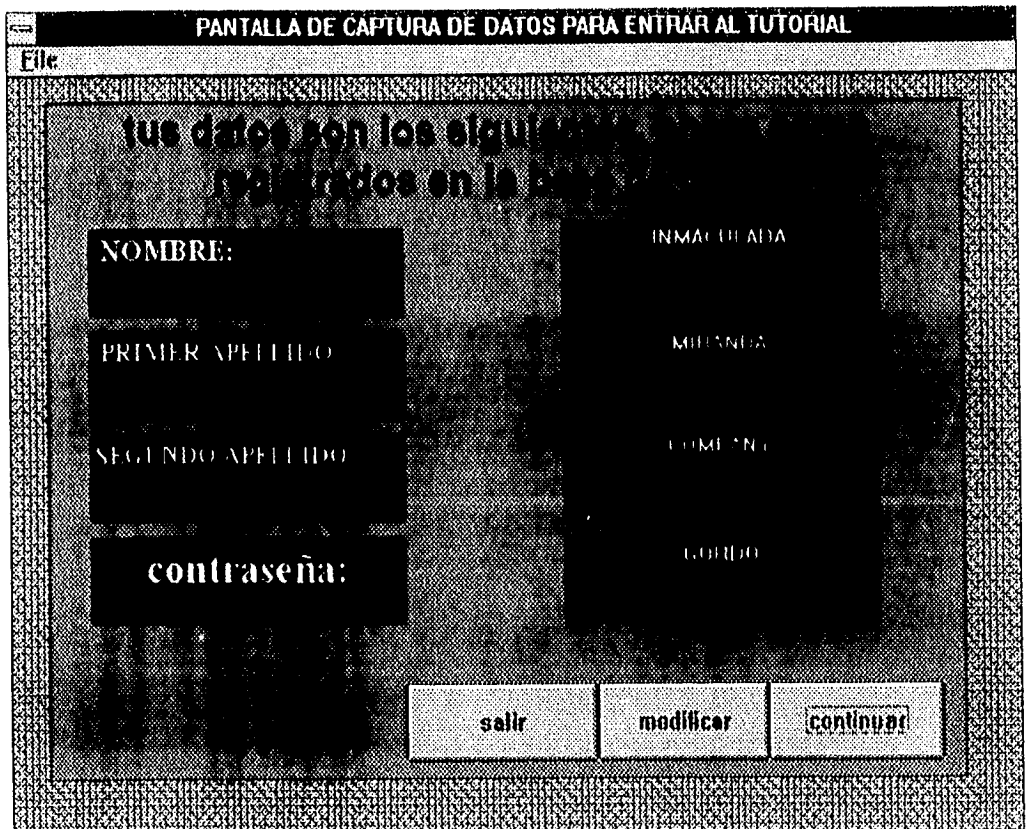
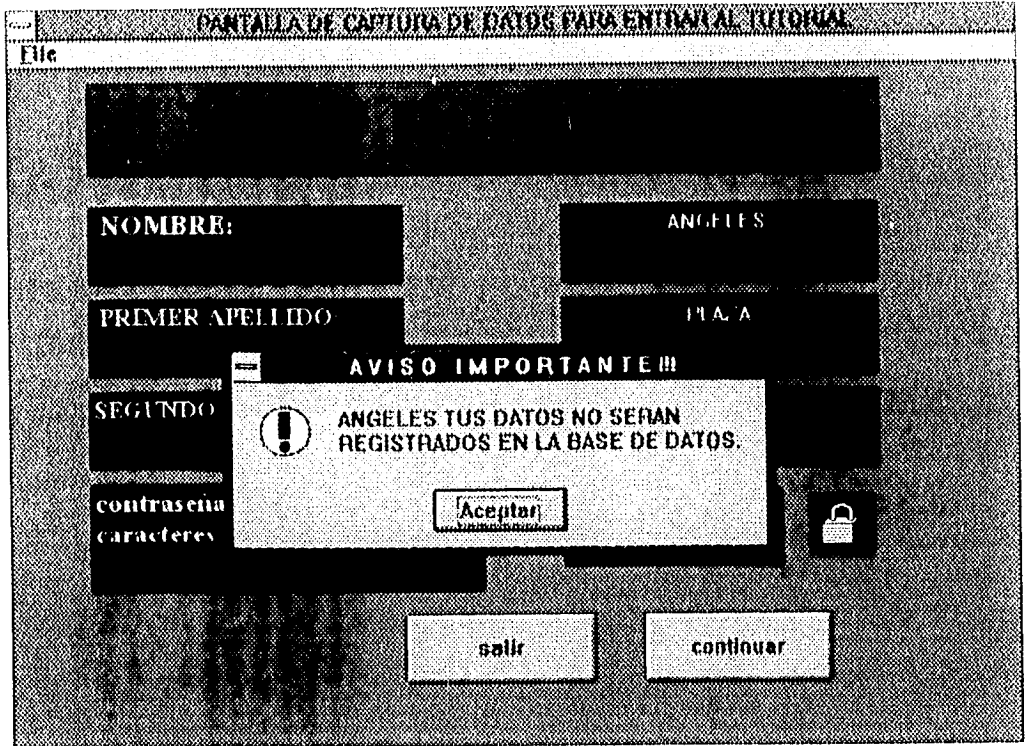


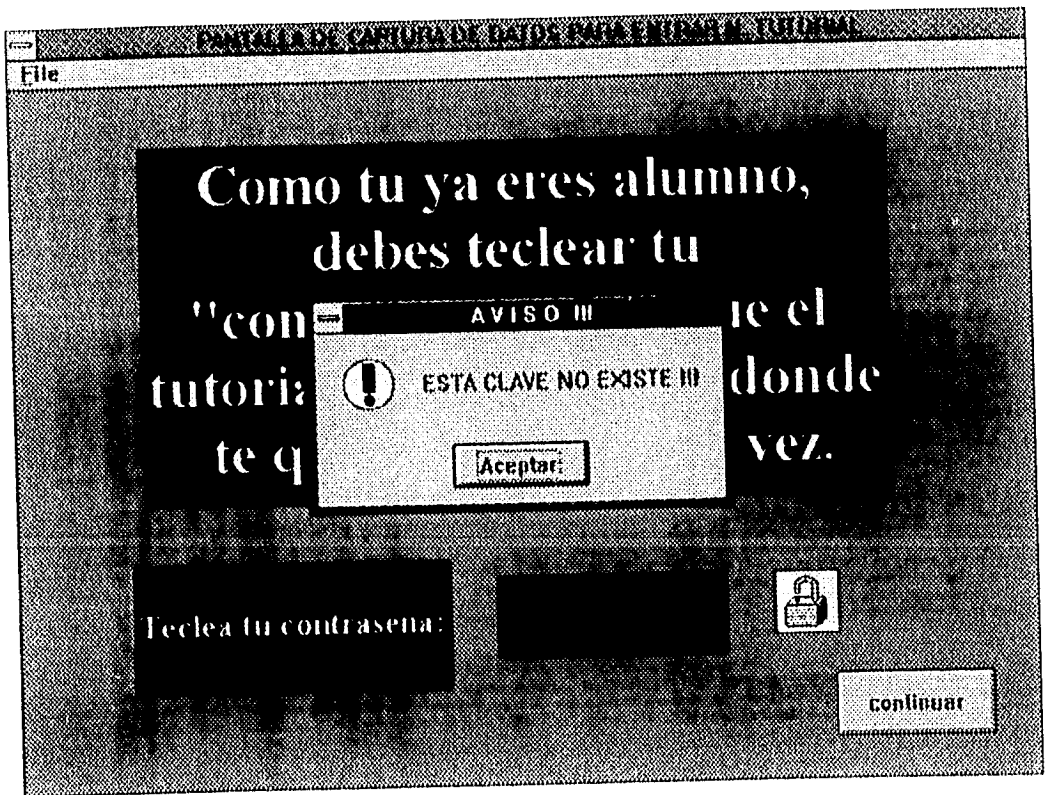
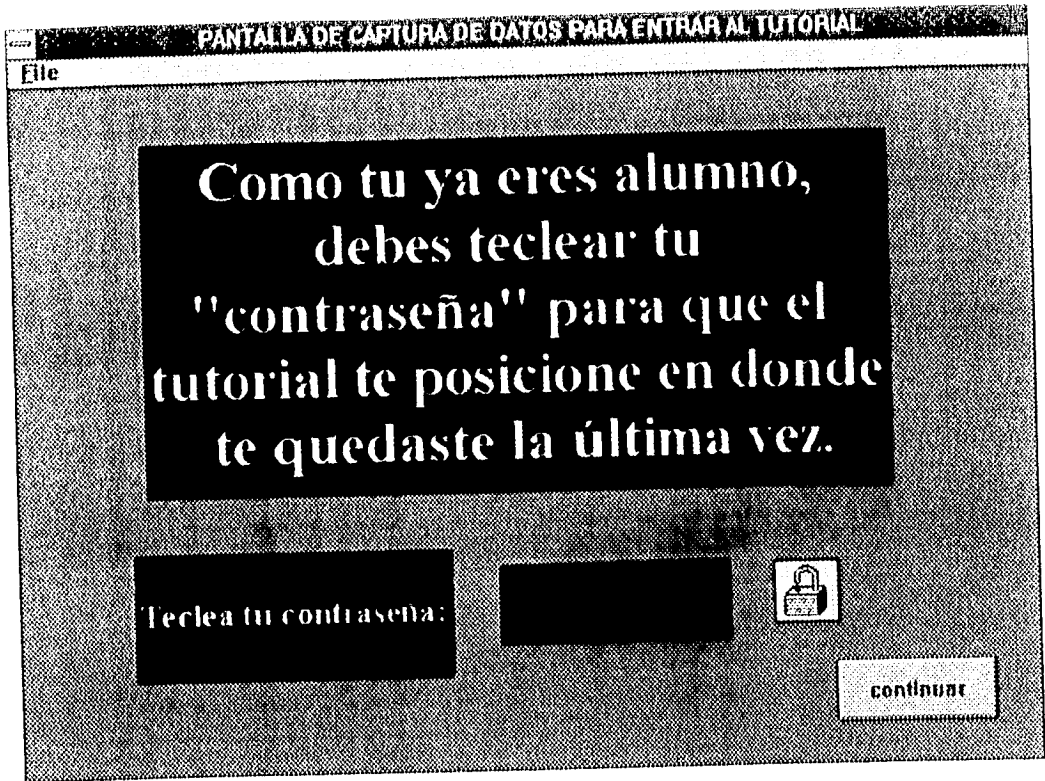
Cuando un nuevo usuario se está inscribiendo y utiliza como contraseña una que ya existe le envía el sistema un mensaje en el que le indica que ya existe y debe cambiarla por favor, y en el caso de que ya no quiera entrar y decida salirse del tutorial, también el sistema le envía un mensaje en el que se le avisa que sus datos no serán registrados en la base de datos por si en otra ocasión quiere volver a entrar al sistema, tome en cuenta que no fue dado de alta.

Una vez que el usuario ya le proporcionó sus datos al tutorial, el sistema le muestra en otra pantalla como quedaron finalmente sus datos y su contraseña, para que los verifique y se almacenen en la base de datos, en caso de que exista algún error el sistema le da la opción de cambiarlos.

También el sistema le envía un mensaje a un usuario cuando éste teclea mal una contraseña y ésta no existe, enviándolo nuevamente a la pantalla en donde se le pregunta si es la primera vez o no que ingresa al tutorial.

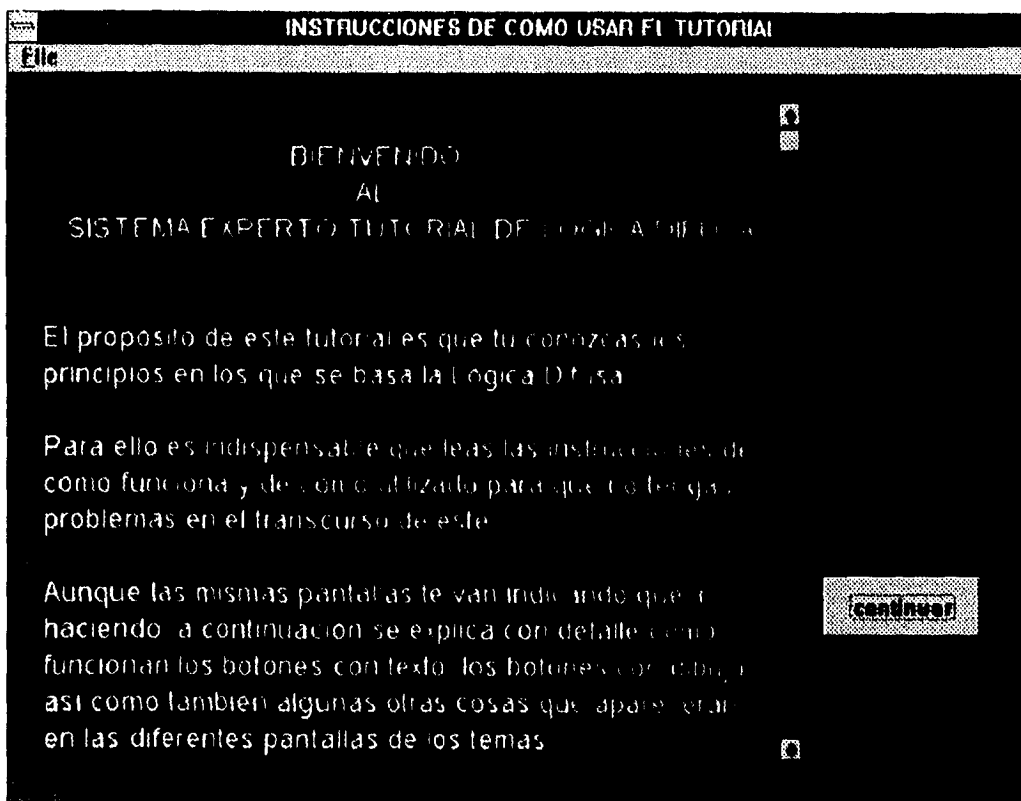






La pantalla siguiente aparecerá después de que un alumno acaba de inscribirse al tutorial, en ella se le muestran al alumno las instrucciones de como utilizar el tutorial, ésta pantalla no tiene la opción de ir directamente al menú principal, ya que es muy importante que todo usuario lea completamente las instrucciones antes de pasar a la introducción y por consiguiente al menú principal.

La pantalla de instrucciones, se verá en el tutorial únicamente la primera vez que se entra a él, ya que su manejo es muy sencillo y basta tan solo con leerla en una ocasión para entender su funcionamiento, además de que en el transcurso del tutorial se va indicando lo que se debe de hacer para poder continuar.



INSTRUCCIONES DE COMO USAR EL TUTORIAL


File


En las pantallas de definiciones encontrarás botones con texto, "continuar y regresar", son como su nombre lo indica para regresarte a la pantalla anterior o llevarte a la siguiente, "salir" es para salirte del tema en el que estas e irte al menu principal.

También encontrarás en las pantallas botones que dicen ejemplo, al presionarlo, este te llevará a una pantalla en donde puedes ver un ejemplo de lo que se le acaba de mencionar.

Este tipo de botones son opcionales aunque se le recomienda que los veas para que te queden mas claras las definiciones.

Cada vez que termines un tema, te aparecerá un mensaje avisándote del acontecimiento y a la vez

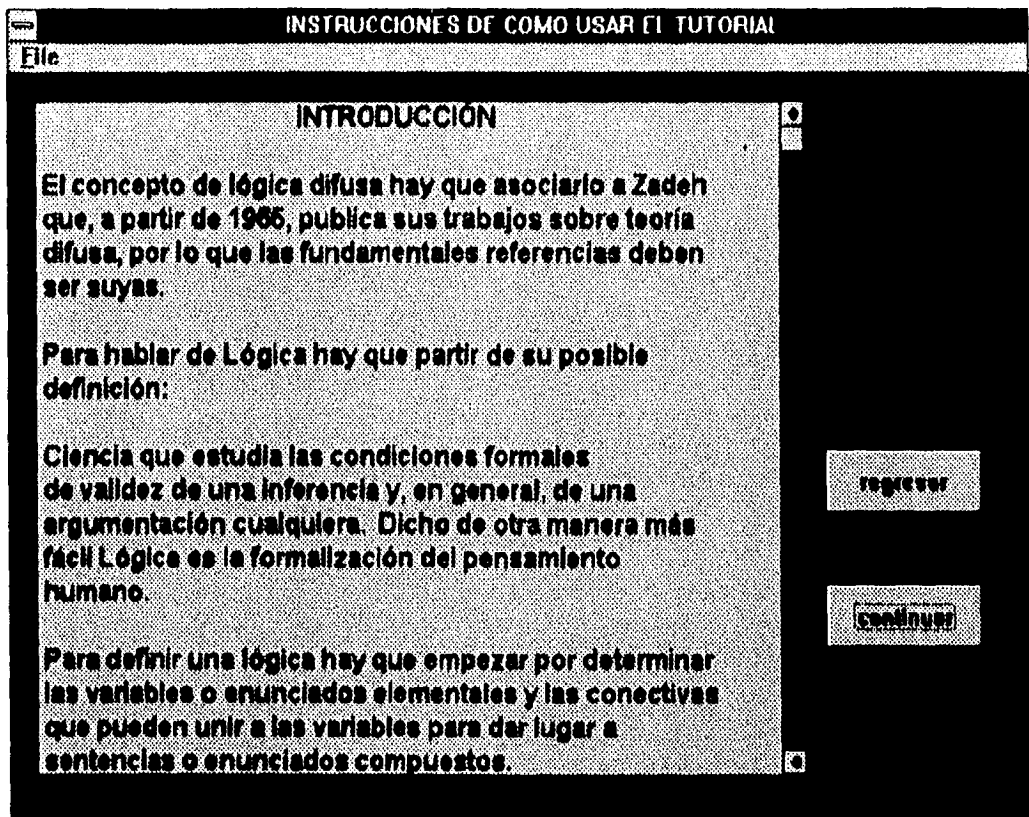




INSTRUCCIONES DE COMO USAR EL TUTORIAL

File

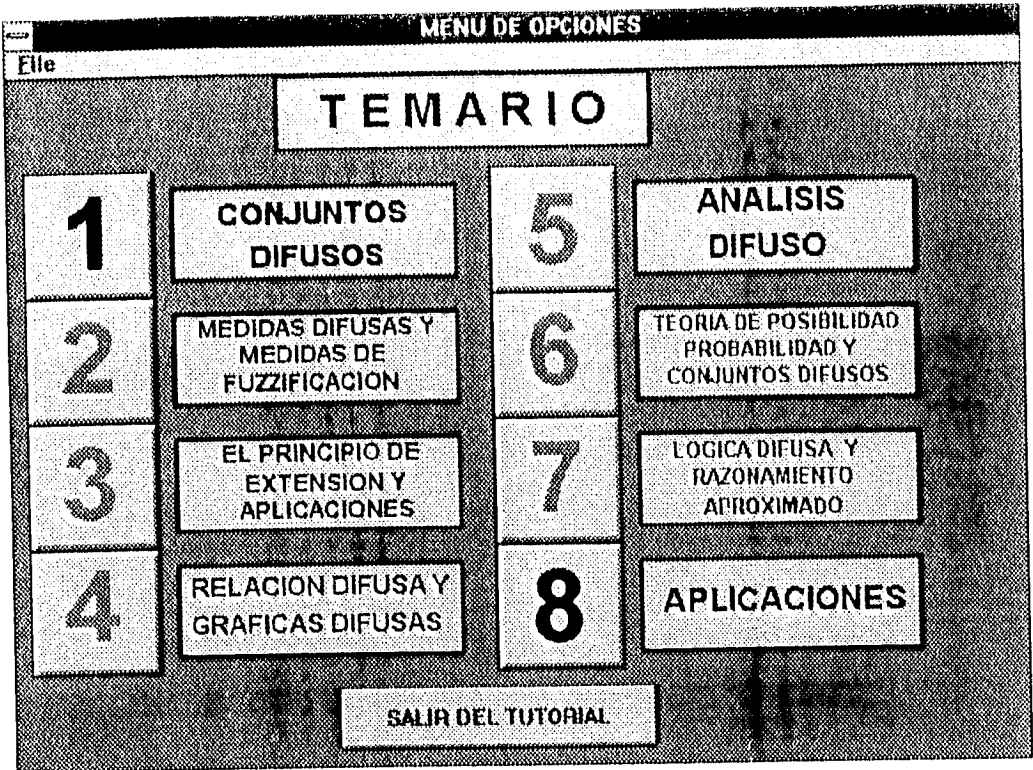
INTRODUCCIÓN A LA LÓGICA DIFUSA



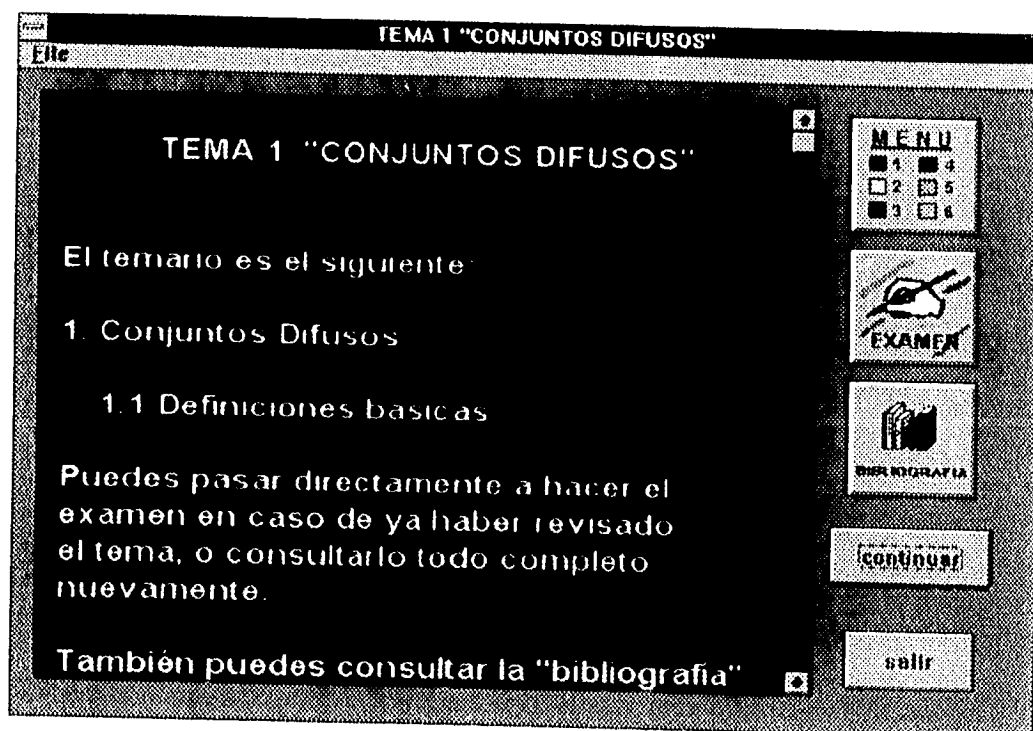
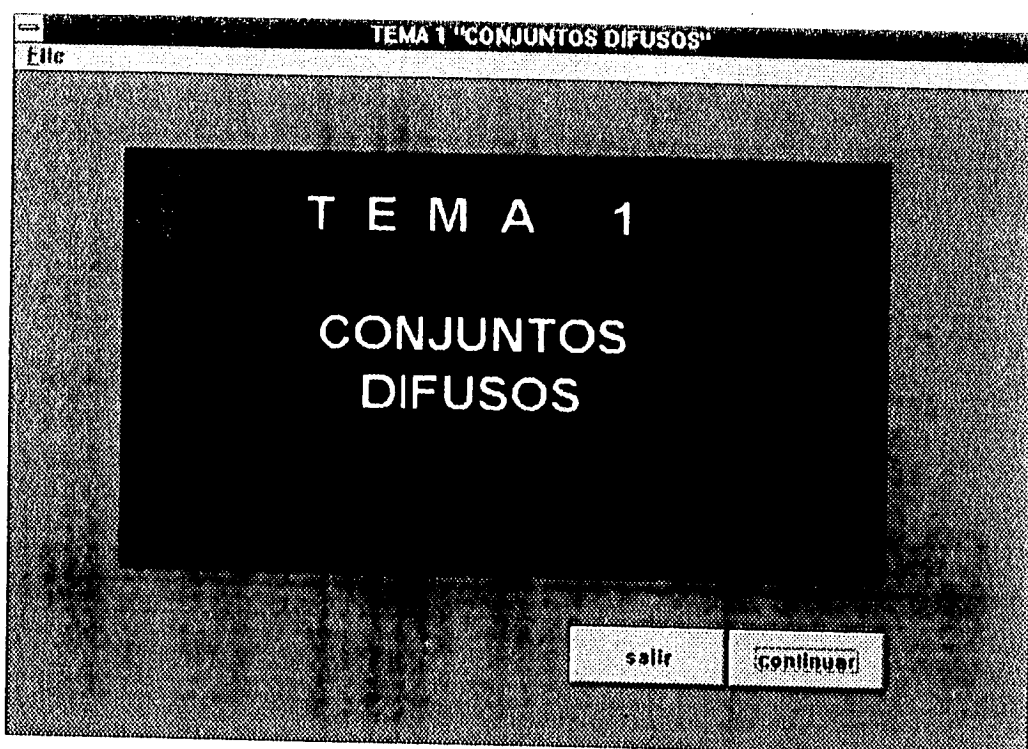
Una vez que el usuario ha leído las instrucciones del manejo del tutorial, el sistema lo lleva a una breve introducción de la Lógica Difusa, para que posteriormente lo pase al menú principal.

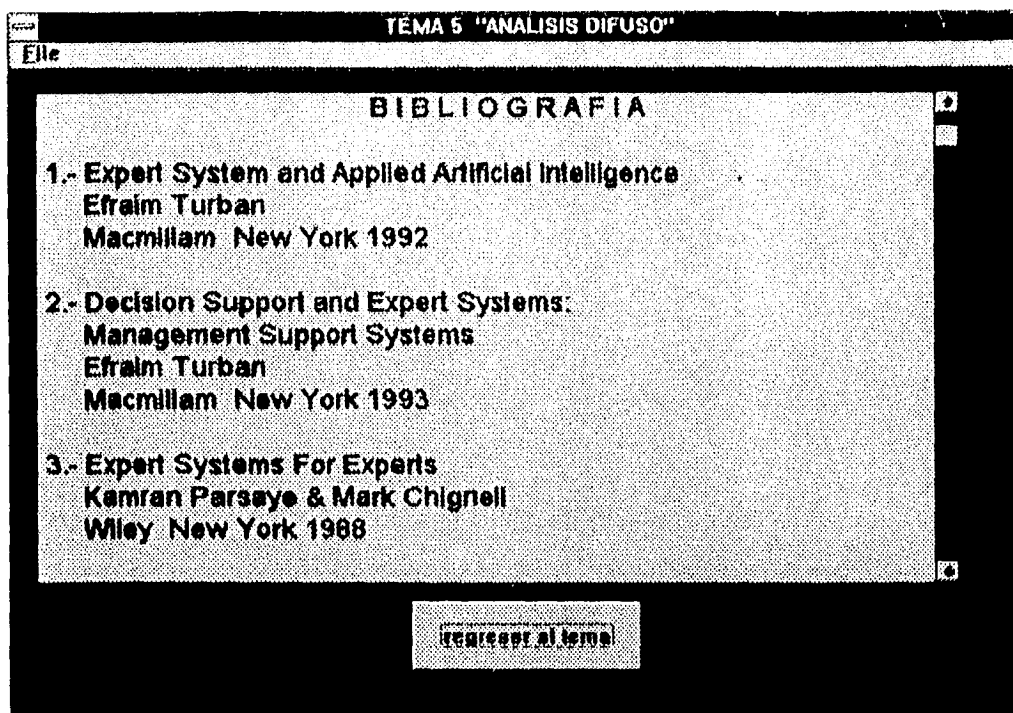
En la pantalla del menú principal se observa que el tutorial cuenta con 8 temas, a los cuales el alumno podrá acceder conforme vaya avanzando, esto quiere decir que son seriados y están inhabilitados, es decir, que si no acredita un tema, no podrá pasar al otro, por el contrario, tiene que volver a estudiarlo.

Dentro del menú principal hay una opción para salir del tutorial en caso de así desearlo el alumno, y el sistema le envía un mensaje de agradecimiento mencionándole su nombre para que observe que ya el tutorial lo considera inscrito por consiguiente lo reconocerá en su próxima sesión.

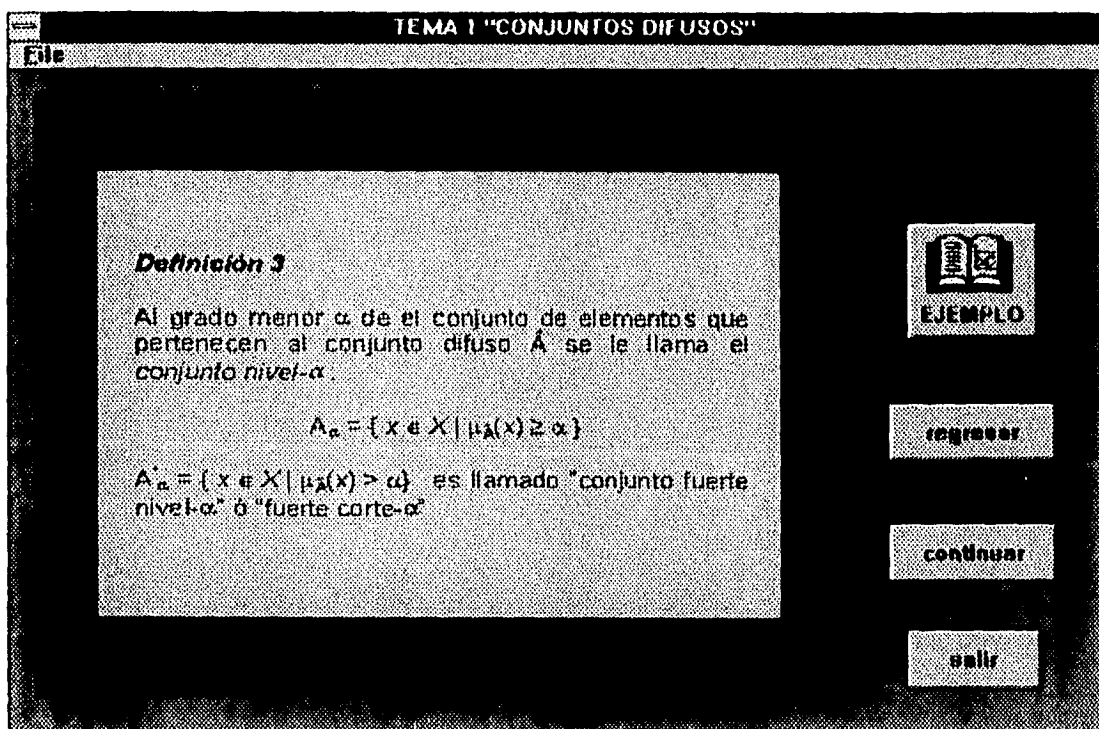


A continuación se muestra el orden de aparición de las pantallas de cada módulo, la primera indica el número del tema y el nombre del mismo, la otra nos da una breve explicación de las definiciones que contiene el tema en la cual se tiene la opción de consultar bibliografía referente al tema.





Las otras pantallas nos muestran como es el contenido del tutorial, en ellas se le explica al alumno detalladamente una definición y tiene la opción de continuar con la siguiente definición, ver el ejemplo relacionado con la definición que está viendo, o regresarse a la definición anterior dentro de la misma pantalla.



TEMA 6 "ANÁLISIS DIFUSO"

File

Los números difusos triangulares representan la función:
 $f(x)$ en $X = X_1, X_2, X_3, X_4$ y X_5 como se muestra en la figura.

" números difusos triangulares representando una función difusa "

IR

[regresar] [continuar]

TEMA 7 "LÓGICA DIFUSA Y RAZONAMIENTO APROXIMADO"

File

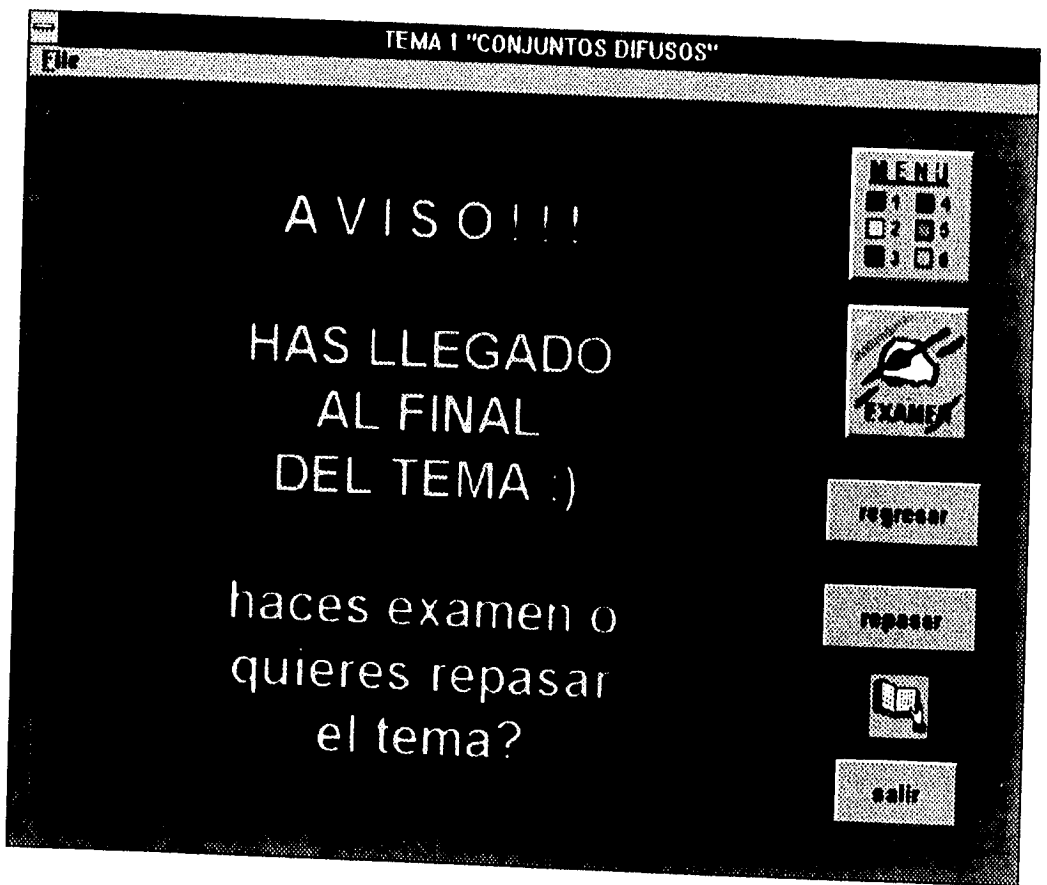
AGE Variable lingüística "Edad"

valores de edad

VARIABLE BASE años

[regresar] [continuar]

La otra es la pantalla en que el tutorial le muestra al alumno que ha concluido con un tema, en ella le da la opción de regresarse a la definición anterior, repasar todo el tema, ir directamente a realizar el examen o salirse del tutorial.



3.4.2 VALIDACIÓN

La aplicación de SETLODI se llevó a cabo bajo tres propósitos:

- Probar la respuesta del sistema a los diferentes eventos
- Probar el diseño de la interfaz del usuario
- Probar el sistema como Sistema Experto Tutorial en sí

Los dos primeros puntos se realizaron con ayuda de los dos colaboradores principales en la interfaz del usuario, la Lic. en Diseño Gráfico y el Lic. en Comunicación Gráfica, también se contó con compañeros que están en el mismo campo de investigación, "la lógica difusa" y finalmente con voluntarios cuya área de desarrollo profesional es diferente, obteniendo excelentes resultados e identificando que la mayoría de las dificultades con las que se encuentra un usuario dentro del tutorial, es debido a que no lee las instrucciones con atención.

El tercer punto y más importante, correspondiente a la aplicación del sistema bajo el punto de vista de SET, se llevó a cabo en tres partes. La primera parte correspondió a la aplicación de SETLODI a los expertos, la segunda fue la aplicación a un usuario y la tercera consistió en mostrar el SET a otros desarrolladores de sistemas basados en lógica difusa, para su opinión.

Los expertos manejaron el tutorial desde el principio y hasta el final, simulando la forma en como la haría un estudiante. El recorrido completo del tutorial se hizo de una forma fluida, sin necesidad de regresar a un tema por no acreditar el examen, claro, hay que tomar en cuenta que el material utilizado es conocido y manejado por los expertos.

Un usuario en cambio, se puede ver en la necesidad de tener que repetir la evaluación o incluso el tema en el que se encuentra, para poder avanzar; hay quienes cada vez que entran al tutorial, repasan desde la última lección hasta un tema completo, con la finalidad de estar al corriente y evitar retrocesos.

Por otra parte, el usuario se sintió a gusto trabajando en la computadora, ya que podía regresar a pantallas anteriores y revisar conceptos tantas veces como quisiera, mostrando gran aceptación en el uso de ejemplos, pues de esta forma la teoría presentada anteriormente se reafirma. También manifestó que el uso de gráficos indicativos, hacen al tutorial más ameno y fácil de manejar.

La depuración total del tutorial requiere de más tiempo y pruebas, ya que puede decirse que un sistema nunca queda perfecto en su totalidad y día a día se presentan nuevas ideas para mejorarlo, entre las cuales se incluye la migración del sistema a nuevas versiones de LEVEL5 y la inclusión a éste de multimedia.

CONCLUSIONES

CONCLUSIONES

Como ya se mencionó, la *Inteligencia Artificial* es el conjunto de técnicas que se aplican en el diseño de programas para computadora que tengan capacidad de razonar, en el sentido de inferir nueva información, y que por la dificultad del problema a resolver requieran una solución con un grado de inteligencia.

Esta cualidad de inteligencia suele implicar una capacidad de aprendizaje, de autocorrección y de razonamiento que son elementos fundamentales y distintivos de la inteligencia artificial.

Dentro de las aplicaciones de la inteligencia artificial destacan por sus particulares propiedades y, por el gran desarrollo que están alcanzando, los "*Sistemas Expertos*". En ellos, el conocimiento requerido para resolver un problema viene dado por la experiencia obtenida por un técnico en la materia que se denomina *Experto*.

Los *Sistemas Expertos Tutoriales*, actualmente son de gran ayuda para poder consultar y aprender conocimientos por medio de la computadora sin la necesidad de tener que ir directamente con el experto en la materia.

Una computadora por sí sola no despierta interés a los educadores como medio de enseñanza. El interés surge a partir del desarrollo de herramientas educativas para computadoras.

La clave está en que la computadora por sí misma proporciona una solución posible y factible. La computadora es un excelente instrumento para el aprendizaje y puede usarse para auxiliar tanto de los maestros como de los estudiantes. Programas de cómputo interactivos aunados al acceso de solución de problemas por computadora, pueden proporcionar una gama de posibilidades no alcanzables de ninguna otra forma.

En un tutorial, cada persona es tratada como un individuo en particular, con todas las características que le distinguen de cualquier otra persona que se considere. Cada persona puede avanzar a su propio ritmo. Además, la experiencia del aprendizaje puede ser activa, con la persona como participante en lugar de como simple espectador.

Un tutorial nunca quedaría obsoleto, ya que éste se puede ir actualizando en información y en presentación. Además puede irse perfeccionando conforme se vayan presentando errores, mediante su uso, o bien se descubran puntos en los que se necesite remarcar un concepto.

En el desarrollo de SETLODI se encontraron dificultades en su interfaz con el usuario, ya que esta parte del sistema es la más importante, puesto que es el medio por el cual le presentamos al usuario la información, teniendo ésta que ser principalmente clara, sobre todo porque la simbología que se maneja en lógica difusa es no muy común y esto conlleva a que la persona se confunda o no entienda el concepto, por otra parte, esta interfaz también debe ser agradable y amena, para que la persona no se canse el transcurso del tutorial, ni se aburra, sino que por el contrario, se interese por seguir adelante.

SETLODI presenta la información contenida de una manera tal que el usuario asimile en gran parte lo visto, por lo cual se optó por desarrollar el sistema experto tutorial en Level 5 Object, porque es un lenguaje de alta programación orientado a objetos y eventos, con el fin de desarrollar sistemas expertos, sistemas expertos con tutorial y sistemas expertos tutoriales.

Este tutorial aún puede ampliarse en lo referente a las aplicaciones, ya que en el área de control se está avanzando constantemente así como también en otras áreas afines y diferentes.

BIBLIOGRAFÍA

BIBLIOGRAFÍA

Efraim Turban
Expert System and Applied Artificial Intelligence
Macmillan New York 1992

Efraim Turban
Decision Support and Expert Systems: Management Support Systems
Macmillan New York 1993

David W. Rolston
Principios de Inteligencia Artificial y Sistemas Expertos
Mc-Graw-Hill Septiembre 1990

"Aplicaciones de los Sistemas Expertos en la Ingeniería", Dr. Felipe Lara Rosano
Inteligencia Artificial en México
Universidad Tecnológica de la Mixteca
México 1992

Ramón Galán López
Introducción a los Sistemas Expertos
Universidad Politécnica de Madrid
Escuela Superior de Ingenieros Industriales
Madrid, España 1990

Felipe Lara Rosano, Nicolas kempfer Valverde, Luis Rodríguez Viqueira
Aplicación de Sistemas Expertos al ahorro energético para la iluminación en la pequeña y mediana industria
Informe Interno. Proyecto 1510. Programa Universitario de Energía
Instituto de Ingeniería. UNAM Febrero 1992

Felipe Lara Rosano, Julio García Sacristán, Eugenio López Ortega
Aplicación de Sistemas Expertos al diagnóstico operativo en la pequeña y mediana industria.
Informe interno. Proyecto 1508. Programa Universitario de Energía
Instituto de Ingeniería. UNAM Febrero 1992

Kamran Parsaye & Mark Chignell
Expert Systems For Experts
Wiley New York 1988

Etienne Wenger

Artificial Intelligence and Tutoring Systems

Computational And Cognitive Approaches to the Communication of Knowledge

Morgan Kaufmann Publishers, Inc.

California Enero 1987

Luger, G F & Stubble field, W A.

Artificial Intelligence and the Design of Expert Systems

Red Wood City, C.A. Benjamin/Cummings 1989

H. J. Zimmermann

Fuzzy Set Theory and Its Applications

Kluwer Academic Publishers USA 1993

Toshiro Terano, Kiyoji Asai, Michio Sugeno

Applied Fuzzy Systems

AP Professional Tokyo, Japan 1994

Agustín Jiménez Avello, Ramón Galán López, Ricardo Sanz B. y Juan Velásco P.

Curso de Control Inteligente de Procesos

Universidad Politécnica de Madrid

Escuela Superior de Ingenieros Industriales

Madrid, España 1993

Dr. Felipe Lara Rosano, Nayeli de Paz Fragoso

Estudio del estado de arte de los Sistemas Expertos aplicados en el control de procesos.

Instituto de Ingeniería. UNAM Agosto 1992

REVISTAS

"Fuzzy Fundamentals", Cox, E. **IEEE Spectrum** Octubre de 1992, pp. 58-61

"Adaptive Fuzzy Systems", Cox, E. **IEEE Spectrum** Febrero de 1993, pp. 27-31

"Fuzzy Logic in Control Systems: Fuzzy Logic Controller Part I, Part II"

Lee, C.C. **IEEE Transactions System, Man and Cybernetics**

Vol. 20, No. 2, pp. 404-435, Marzo-Abril de 1990

"Fuzzy Gain Scheduling of PID Controllers" Zhao, Z.

IEEE Transactions System, Man and Cybernetics

Vol. 23, No. 5, 1392-1398, 1993

"Fuzzy Identification of Systems and its Applications to Modeling and Control"
Takagi, T. and M. Sugeno. *IEEE Transactions System, Man and Cybernetics*
Vol. 15, No. 1, 116-132, 1985

"Fuzzy Control Theory: A Nonlinear Case" Ying, H. *Automática*
Vol. 26, 513-520, 1990

"Theoretical and Linguistic Aspect of the Fuzzy Logic Controller"
M. Braae and D.A. Rutherford. *Automática* Vol 15, No. 5, 553-577, 1979

"Application of a Fuzzy Controller in a Warm Water Plant"
W.J. Kicket and H.R. Van Nauta Lemke, *Automática* Vol. 12 (4), 301-308, 1976

"The Application of Fuzzy Control Systems to Industrial Processes"
P.J. King and E.H. Mamdani, *Automática* Vol 13 (3), 235-242, 1977

"Theoretical an Linguistic Aspect of the Fuzzy Logic Controller"
Braae & D.A. Rutherford, *Automática* Vol 15 (5), 553-577, 1979

"Fuzzy Control Theory : A Nonlinear Case"
Hao Ying, William Sile and James J. Buckley. *Automática*
Vol. 26 (3), 513-520, 1990

"Fuzzy Logic: Concept to Constructs" Greg Viot,
IA Expert The Magazine of Artificial Intelligence in Practice November 1993

"Fuzzy Risk Analysis: Using AI Systems" Mark Jablonowski,
IA Expert The Magazine of Artificial Intelligence in Practice December 1994

"Relational Database Queries Using Fuzzy Logic" Earl Cox,
IA Expert The Magazine of Artificial Intelligence in Practice January 1995

"Fuzziness in Real State Clarification Through Fuzzification" Gene dilmore,
IA Expert The Magazine of Artificial Intelligence in Practice March/April 1995

MATERIAL VISUAL

I.- INFORMATION PROCESSING WITH FUZZY LOGIC

Presented by: Piero Bonissone General Electric CR & D
Produced by the: IEEE Educational Activities Board
In Cooperation with the: IEEE Neuronal Networks Council

II.- INTRODUCTION TO FUZZY SET

Theory and Fuzzy Logic: Basic Concepts and Structures

Presented by: Enrique Ruspini SRI International
Produced by the: IEEE Educational Activities Board
In Cooperation with the: IEEE Neuronal Networks Council

III.- FUZZY LOGIC: Advances Concepts and Structures

Presented by: Lofti Zadeh UC/Berkeley
Produced by the: IEEE Educational Activities Board
In Cooperation with the: IEEE Neuronal Networks Council