



# Inteligencia Artificial



Perspectivas y  
Realizaciones

G. Choque Aspiazu

La Paz, Bolivia  
Diciembre de 2002

---

# INTELIGENCIA ARTIFICIAL

## Perspectivas y Realizaciones

---

Guillermo Choque Aspiazu

Universidad Mayor de San Andrés  
Ciencias de la Computación

---

## DEDICATORIA

A mi querido hijo Andrés ...  
Hoy en tu inocencia estas contemplando el desarrollo de la  
inteligencia artificial, mañana podrás utilizar esta tecnología para  
reconstruir lo que no está escrito.

## PREFACIO

Durante mucho tiempo he conocido de cerca muchas personas que incrédulas contemplaban como la inteligencia artificial comenzaba a crecer apoderándose de los principales titulares en seminarios, conferencias, encuentros profesionales y congresos sobre ciencias de la computación en muchos lugares del orbe.

La incredulidad, manifiesta en juicios de valor mezquinos, consideraba a esta rama multidisciplinaria como un hijo no deseado; el tiempo en su devenir ha mostrado la utilidad de la inteligencia artificial en la solución de problemas recurrentes con nuevos enfoques y paradigmas que hacen interesante su aplicación al interior de la fascinante ciencia de las computadoras.

La Inteligencia Artificial, o IA en términos cortos, es una combinación de la ciencia de las computadoras, la psicología y la filosofía. La IA cubre un espectro amplio, consistente en campos de estudio diferentes, estos van desde la visión por computadora hasta las redes neuronales y los sistemas expertos. El elemento que los campos de la IA tienen en común es la creación o simulación de máquinas que puedan “pensar”.

Para clasificar las máquinas como pensantes, es necesario definir inteligencia. El problema que aparenta ser simple es uno de los más complejos del área, en el entendido que la inteligencia puede representar, por ejemplo, la solución adecuada de problemas complejos o a establecer generalizaciones y relaciones entre objetos. ¿Qué se podría decir acerca de la percepción y la comprensión? ¿Dónde podrá situarse las formas de adquisición del conocimiento? y, si el pensar es un acto natural concomitante con la naturaleza ¿Cuáles son las herramientas que se necesitan para simular un comportamiento natural del pensar humano?

Estos cuestionamientos son centrales en la concepción de este texto resumen de la inteligencia artificial. Una primera versión vio la luz en el anterior siglo, específicamente en el año 1998, cuando fue presentado como texto de lectura preliminar para la materia del mismo nombre en la Universidad Mayor de San Andrés. A inicios del siglo, y con el advenimiento acelerado de las tecnologías web, se publicó el texto en el sitio web de la UMSA. Aún es posible obtener una descarga gratuita del contenido en este sitio: [www.umsanet.edu.bo/docentes/gchoque/M420texto.htm](http://www.umsanet.edu.bo/docentes/gchoque/M420texto.htm)

El texto está compuesto por siete capítulos: el capítulo uno hace referencia a los fundamentos sobre los cuales se edifica la inteligencia artificial, el segundo capítulo hace referencia a la representación del conocimiento, el tercer capítulo toma como elemento fundamental el estudio de las lógicas como elementos para la representación del conocimiento, específicamente se considera a la lógica difusa, el capítulo cuatro se refiere al primer paradigma biológico que se aborda, el estudio de las redes neuronales artificiales, el capítulo cinco toma como elemento central a los algoritmos genéticos, el capítulo seis está referido a la vida artificial y el capítulo siete considera a la realidad virtual.

Con muchas mejoras y la revisión siempre acertada de mi amigo y hermano artificial Carlos Collazos, quiero presentar esta pequeña aventura intelectual y literaria que se ha subtitulado como perspectivas y realizaciones porque rescata lo esencial y básico de las diferentes áreas de la inteligencia artificial, y además, porque propone una perspectiva futura al intentar utilizar estos paradigmas en la solución de los problemas que aquejan a la sociedad y que requieren de manera apresurada de nuevas propuestas que puedan ser útiles para lograr la satisfacción de algunas demandas insatisfechas.

## AGRADECIMIENTOS

Una de las grandes emociones que me tocó vivir fue la de salir campeones en la práctica del fútbol, en los diferentes campeonatos barriales en estos últimos quince años. Por analogía acaba de terminarse esta aventura, el texto está escrito, las imágenes están hechas, las anteriores versiones han sido revisadas, se ha revisado también la composición final y ya está preparada para su paso final, para dar su vuelta de popularidad alrededor de las innumerables canchas que le tocará visitar.

Quiero agradecer a los alumnos de la materia Inteligencia Artificial, el espacio disponible no permitiría que nombre a los aproximadamente 600 alumnos que pasaron la materia en estos últimos cinco años, por todas las agradables clases que pudimos disfrutar de manera conjunta. Se que muchos hicieron sus tesis de grado copiando partes de este texto y no mencionaron a los autores de la teoría utilizada, ahora se incluyen referencias bibliográficas y referencias electrónicas a sitios web para que corrijan este detalle.

Cuando converse con el Decano de la Facultad de Ciencias Puras y Naturales, Msc. Ing. Rolando Campuzano Arzabe, no dubitó un momento en ofrecer los servicios del Centro de Publicaciones de la Facultad para que este texto se replique y en este momento se encuentre en sus manos. En este entendido agradezco al Msc. Ing. Campuzano por su colaboración y al personal del Centro de Publicaciones por todas las atenciones.

Finalmente quiero agradecer a mi familia por haber soportado mi alejamiento virtual al comprometerme indirectamente con la computadora, mientras duro la redacción y corrección del texto. Andresito adorado hijito este texto es para ti y disculpas por haber robado tus minutos de juego, en el patio con la pelota de fútbol y en el estudio con esta tonta útil que refleja mi rostro en su pantalla y que en el futuro querrá también ser agradecida y reclamará bloqueando el teclado o apagando su procesador en señal de protesta.

La Paz, Bolivia.  
Diciembre de 2002.

G.I.C.A.

## Síntesis del Contenido

Pág.

1. Fundamentos
  - 1.1. Inteligencia
  - 1.2. Inteligencia Artificial
  - 1.3. Evaluación Persona-Computadora
  - 1.4. Ciencia Cognitiva
  - 1.5. Áreas de Investigación
2. Representación del Conocimiento
  - 2.1. Introducción
  - 2.2. Definición
  - 2.3. Evaluación de los Sistemas Basados en Conocimiento
  - 2.4. Formalismos de Representación del Conocimiento
3. Lógica Difusa
  - 3.1. Introducción
  - 3.2. Definición
  - 3.3. Teoría de Conjuntos Difusos
  - 3.4. Características Esenciales
  - 3.5. Diferencias respecto a Sistemas Tradicionales
  - 3.6. Teoría de la Posibilidad
  - 3.7. Semántica con Resultados de Ensayo
  - 3.8. Aplicaciones
4. Redes Neuronales Artificiales
  - 4.1. Introducción
  - 4.2. Neuronas
  - 4.3. Redes Neuronales Artificiales
  - 4.4. Modelos de Redes Neuronales
  - 4.5. Topologías Generales
  - 4.6. Topologías Específicas
  - 4.7. Aplicaciones
5. Algoritmos Genéticos
  - 5.1. Fundamentos
  - 5.2. Teoría de la Evolución
  - 5.3. Comparación con lo convencional
  - 5.4. Algoritmo Genético Simple
  - 5.5. Procedimiento General
  - 5.6. Algoritmo Genético con Objetos de Tamaño Fijo
  - 5.7. Algoritmo Genético General
  - 5.8. Problema del Agente Viajero
  - 5.9. Programación Genética
  - 5.10. Aplicaciones
6. Vida Artificial
  - 6.1. Introducción

- 6.2. Vida Artificial
- 6.3. Características de los Modelos de Vida Artificial
- 6.4. Mecánica de la Vida
- 6.5. Aplicaciones
  
- 7. Realidad Virtual
- 7.1. Introducción
- 7.2. Historia
- 7.3. Comunicación Hombre-Maquina
- 7.4. Espacio de la Realidad Virtual
- 7.5. Definición
- 7.6. Dispositivos
- 7.7. Clasificación
- 7.8. Aplicaciones

## Contenido

Pág.

### **1. Fundamentos**

- 1.1. Inteligencia
- 1.2. Inteligencia Artificial
  - 1.2.1. Raíces
  - 1.2.2. Metodologías
  - 1.2.3. Objetivos
- 1.3. Evaluación Persona-Computadora
  - 1.3.1. Capacidades de las Computadoras
  - 1.3.2. Capacidades de las Personas
  - 1.3.3. Actitudes Esenciales de la Inteligencia
- 1.4. Ciencia Cognitiva
- 1.5. Áreas de Investigación
  - 1.5.1. Sistemas Expertos
  - 1.5.2. Procesamiento del Lenguaje Natural
  - 1.5.3. Reconocimiento del Habla
  - 1.5.4. Visión por Computadora
  - 1.5.5. Robótica
  - 1.5.6. Otras Áreas

Ejercicios # 1

Bibliografía

### **2. Representación del Conocimiento**

- 2.1. Introducción
- 2.2. Definición
  - 2.2.1. Vista Atómica
  - 2.2.2. Vista Holística
  - 2.2.3. Notas Finales
- 2.3. Evaluación de los Sistemas Basados en Conocimiento
  - 2.3.1. Dato
  - 2.3.2. Sistema de Programación Convencional
  - 2.3.3. Sistema de Base de Datos
  - 2.3.4. Sistema Basado en Conocimiento
- 2.4. Formalismos de Representación del Conocimiento
  - 2.4.1. Redes Semánticas
  - 2.4.2. Armazones
  - 2.4.3. Reglas de Producción
  - 2.4.4. Redes de Herencia Estructurales
  - 2.4.5. Representación Terminológica
  - 2.4.6. Guiones

Ejercicios # 2

Bibliografía

### **3. Lógica Difusa**

- 3.1. Introducción
- 3.2. Definición
- 3.3. Teoría de Conjuntos Difusos



- 3.3.1. Conjuntos Clásicos
- 3.3.2. Conjuntos Difusos
- 3.3.3. Función de Pertenencia
- 3.4. Características Esenciales
- 3.5. Diferencias respecto a Sistemas Tradicionales
  - 3.5.1. Valores de Verdad
  - 3.5.2. Predicados
  - 3.5.3. Modificadores de Predicados
  - 3.5.4. Cuantificadores
  - 3.5.5. Probabilidades
  - 3.5.6. Posibilidades
- 3.6. Teoría de la Posibilidad
- 3.7. Semántica con Resultados de Ensayo
- 3.8. Aplicaciones
  - 3.8.1. Área de Control
  - 3.8.2. Área de Inteligencia Artificial
- Ejercicios # 3
- Bibliografía

#### **4. Redes Neuronales Artificiales**

- 4.1. Introducción
- 4.2. Neuronas
  - 4.2.1. Modelo Biológico Básico
  - 4.2.2. Modelo Idealizado
  - 4.2.3. Modelo Computacional
  - 4.2.4. Modelos de Neuronas
- 4.3. Redes Neuronales Artificiales
  - 4.3.1. Taxonomía
  - 4.3.2. Aprendizaje
- 4.4. Modelos de Redes Neuronales
  - 4.4.1. Caracterización
  - 4.4.2. Topologías
- 4.5. Topologías Generales
  - 4.5.1. Neurona Simple
  - 4.5.2. Red Simple
  - 4.5.3. Red con Elementos de Asociación
  - 4.5.4. Redes Multicapa
  - 4.5.5. Modelo Interactivo
  - 4.5.6. Modelo Interactivo Desarrollado
- 4.6. Topologías Específicas
  - 4.6.1. Perceptrón
  - 4.6.2. Propagación de Errores
  - 4.6.3. Hopfield
  - 4.6.4. Kohonen
- 4.7. Aplicaciones
- Ejercicios # 4
- Bibliografía

#### **5. Algoritmos Genéticos**

- 5.1. Fundamentos
- 5.2. Teoría de la Evolución
- 5.3. Comparación con lo convencional
- 5.4. Algoritmo Genético Simple
  - 5.4.1. Anatomía
  - 5.4.2. Codificación de Variables
- 5.5. Procedimiento General
- 5.6. Algoritmo Genético con Objetos de Tamaño Fijo
  - 5.6.1. Selección
  - 5.6.2. Apareamiento
  - 5.6.3. Mutación
- 5.7. Algoritmo Genético General
- 5.8. Problema del Agente Viajero
- 5.9. Programación Genética
  - 5.9.1. Elección del Lenguaje de Programación
  - 5.9.2. Iniciación
  - 5.9.3. Programas de Apareamiento
  - 5.9.4. Mutación de Programas
  - 5.9.5. Función de Adaptación
- 5.10. Aplicaciones
- Ejercicios # 5
- Bibliografía

## **6. Vida Artificial**

- 6.1. Introducción
- 6.2. Vida Artificial
  - 6.2.1. Evolución del Área
  - 6.2.2. Intereses del Área
  - 6.2.3. Herramientas del Área
- 6.3. Características de los Modelos de Vida Artificial
  - 6.3.1. Algunos Modelos Representativos
- 6.4. Mecánica de la Vida
  - 6.4.1. Reproducción Propia
  - 6.4.2. Comportamiento Emergente
  - 6.4.3. Metabolismo y Adaptabilidad
  - 6.4.4. Evolución
- 6.5. Aplicaciones
- Ejercicios # 6
- Bibliografía

## **7. Realidad Virtual**

- 7.1. Introducción
- 7.2. Historia
- 7.3. Comunicación Hombre-Maquina
- 7.4. Espacio de la Realidad Virtual
- 7.5. Definición
- 7.6. Dispositivos
- 7.7. Clasificación
  - 7.7.1. Sistemas de Ventana

- 7.7.2. Sistemas de Mapeo por Vídeo
- 7.7.3. Sistemas Inmersivos
- 7.7.4. Sistemas de Telepresencia
- 7.7.5. Sistemas de Realidad Mixta
- 7.7.6. Sistemas de Pecera
- 7.8. Aplicaciones

Ejercicios # 7

Bibliografía

Anexo A “Glosario de Términos”

# 1 FUNDAMENTOS

## 1.1. INTELIGENCIA

Es completamente posible establecer una escala aproximada de la inteligencia: muchas personas son más inteligentes que algunos chimpancés, un procesador de textos es una máquina más inteligente que una máquina de escribir mecánica, etc. Sin embargo no existe una definición científica de inteligencia.

Se reconoce que es difícil precisar el significado de la Inteligencia, sin embargo algunas definiciones mencionadas por algunos investigadores son:

- a) BINET: buen discernimiento, buena comprensión, buen razonamiento.
- b) SPEARMAN: la inteligencia general..... supone principalmente saber relacionar y correlacionar.
- c) Terman: capacidad para desarrollar conceptos y captar su significado.
- d) VERNON: todo lo relacionado con capacidad para pensar o eficiencia mental.
- e) BURT: capacidad innata, general, habilidad cognoscitiva.
- f) HEIM: la inteligencia denota capacidad para captar lo esencial de una situación y responder en consecuencia.
- g) WECHSLER: capacidad general o global del individuo para actuar según un propósito, pensar racionalmente y relacionarse eficazmente con su entorno.
- h) PIAGET: adaptación al entorno físico y social.
- i) MINSKY: inteligencia es todo lo que puede ser medido mediante pruebas de inteligencia.

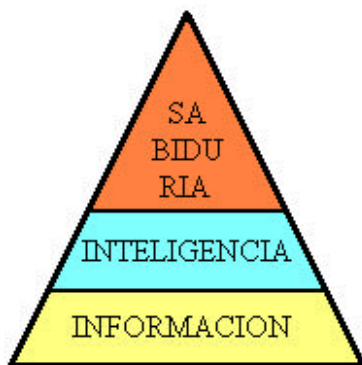


Fig. 1.1. Jerarquía de la Inteligencia

La combinación de aptitudes, criterio, comprensión, razonamiento, conceptualización, saber responder adecuadamente, adaptación, etc. que se encuentran presentes en estas definiciones señalan la naturaleza polifacética de la inteligencia.

La inteligencia por otra parte está relacionada con la habilidad para reconocer patrones, deducir conclusiones razonables, analizar sistemas complejos como elementos simples y resolver sus contradicciones, y aun es mucho más que todo lo mencionado. La inteligencia está en un nivel mayor que la información y el conocimiento pero debajo del nivel de la sabiduría.

Una de las formas más aceptadas para explicar la inteligencia, desde el punto de vista computacional, constituye la prueba de Alan Turing elaborada en 1950. La idea de la prueba de Turing es que si un interrogador decide erróneamente que una máquina es una persona, entonces se dice que dicha máquina exhibe inteligencia.

## 1.2. INTELIGENCIA ARTIFICIAL

Resulta obvio que la inteligencia es de por sí un concepto bastante complicado de definir. Por esta razón intentar definiciones para la inteligencia artificial es también compleja.

Quizá la inteligencia artificial debería llamarse inteligencia sintética para que concuerde mejor con el lenguaje comercial. Así los diamantes artificiales son falsas imitaciones, mientras que los diamantes sintéticos son diamantes auténticos, sólo que manufacturados en lugar de desenterrados. No obstante el nombre, la inteligencia artificial aspira a una inteligencia auténtica, no a una falsa imitación.

Marvin Minsky sugiere el carácter práctico de la inteligencia artificial mencionando: “La inteligencia artificial es la ciencia de hacer que las máquinas hagan cosas que requerirían inteligencia si fueran hechas por seres humanos”. La definición aceptada de la inteligencia artificial, para los fines de la presente publicación, es la que menciona la investigadora Elaine Rich: *“La inteligencia artificial es el estudio de cómo hacer que los ordenadores hagan cosas que por el momento las personas realizan de una forma más perfecta”*.

### 1.2.1. Raíces

La Inteligencia Artificial es un campo interdisciplinario que se deriva de múltiples raíces, incluyendo contribuciones de la psicología, filosofía, lingüística, ingeniería eléctrica y la ciencia de las computadoras. La intersección entre la psicología y la inteligencia artificial se centra en las subáreas conocidas como psicología cognitiva y psico lingüística. La filosofía y la inteligencia artificial caminan de manera conjunta en las áreas de la lógica, filosofía del lenguaje y la filosofía de la mente. Las intersecciones con la lingüística incluyen lingüística computacional, psico lingüística y socio lingüística. Concentraciones mutuas entre la ingeniería eléctrica y la inteligencia artificial incluyen proceso de imágenes, reconocimiento de patrones y robótica. Finalmente, la ciencia de las computadoras apoya a la inteligencia artificial de manera propia y a los campos relacionados de los sistemas adaptativos.

### 1.2.2. Metodologías

Las suposiciones metodológicas bajo las cuales puede ser categorizada la investigación en la inteligencia artificial son: métodos formales y programas de escritura. En la IA, como en otros campos, es importante contar con modelos formales. Una variedad de maquinaria formal ha probado ser útil en la IA, incluyendo lógica de primer orden<sup>1</sup> y otras lógicas, sistemas para la semántica formal<sup>2</sup>, y los trabajos sobre matemáticas y gramáticas. Sin embargo en la inteligencia artificial no es suficiente probar las cosas de manera formal, la construcción de programas activos para su ejecución es también bastante importante. Usualmente los sistemas de inteligencia artificial son implementados en lenguajes simbólicos especializados tales como LISP, PROLOG, sistemas de producción, o lenguajes de representación de conocimiento.

---

<sup>1</sup> First Order Logic (F.O.L.).

<sup>2</sup> Tales como la semántica de Montague, semántica situacional, semántica procedimental y semántica denotacional.

Frecuentemente la línea entre las dos metodologías no es clara y muchos de los mejores trabajos sobre inteligencia artificial involucran una mezcla juiciosa de modelos formales y sistemas implementados.

### **1.2.3. Objetivos**

Puesto que la meta de la inteligencia artificial es desarrollar sistemas que se comporten de manera inteligente, es decir de manera similar a los seres humanos, Cercone & McCalla en 1984 diseñaron objetivos que cualquier sistema ideal de inteligencia artificial debe cumplir. Un investigador de la inteligencia artificial debe ser capaz de:

- a) Desarrollar un sistema de trabajo que se comporte de una manera inteligente;
- b) Validar el comportamiento del sistema mediante comparaciones con seres humanos (objetivo de diseño externo);
- c) Entender las implicaciones del sistema - su cobertura, lo que puede hacer (objetivo de diseño interno);
- d) Determinar que es lo que puede ser generalizado acerca del sistema, las lecciones que enseña, las áreas de impacto.

## **1.3. EVALUACION PERSONA – COMPUTADORA**

### **1.3.1. Capacidades de las Computadoras**

Una computadora estándar cualquiera, normalmente tiene las siguientes tres capacidades que la identifican de los seres humanos:

- a) Cálculo numérico.
- b) Almacenamiento de Información.
- c) Operaciones repetitivas.

### **1.3.2. Capacidades de las Personas**

Una persona en promedio cuenta con las siguientes capacidades innatas que la diferencian de manera abierta de las computadoras:

- a) Proceso y entendimiento de información.
- b) Sentido a lo que se ve y escucha (visión, audición.)
- c) Deducción de nuevas ideas.
- d) Sentido común.
- e) Capacidad motriz.
- f) Autocontrol.
- g) Razonamiento.
- h) Intuición.
- i) Sentimientos.

### **1.3.3. Actitudes Esenciales de la Inteligencia**

La inteligencia presenta las siguientes actitudes que son importantes para poder ser categorizada:

- a) Responder de manera flexible a las situaciones.
- b) Obtener el sentido de mensajes contradictorios o ambiguos.
- c) Reconocer la importancia relativa de los diferentes elementos de una situación.
- d) Encontrar semejanzas entre situaciones a pesar de las diferencias que puede haber entre las mismas.
- e) Extraer diferencias entre situaciones a pesar de las similitudes que puede haber entre las mismas.

#### **1.4. CIENCIA COGNITIVA**

Se entiende por ciencia cognitiva a las ciencias que abordan conceptos tales como sensación, emoción, pensamiento, deseo, etc. Es decir no solamente aquellos conceptos que tienen que ver con el conocimiento sino también otros conceptos mentales. Las investigaciones cognitivas presuponen que ese tipo de conceptos no se agotan en algo neurofisiológico, o bien que poco o nada tiene que ver con lo neurofisiológico. De esta suerte esas investigaciones persiguen otros métodos de búsqueda, a saber, el uso de modelos. Estos modelos se ven favorecidos con el uso de lenguajes de computación y de allí la afinidad entre las ciencias cognitivas y la inteligencia artificial.

La ciencia cognitiva construye teorías y prueba las mismas experimentando con los seres humanos como sujetos, mientras que el investigador en inteligencia artificial construye y prueba teorías implementando las mismas a manera de programas computacionales.

#### **1.5. AREAS DE INVESTIGACION**

Los mayores esfuerzos de la investigación en inteligencia artificial se concentraron en el entendimiento del lenguaje natural, la visión por computadora, aprendizaje, prueba de teoremas y programación lógica, búsqueda, solución y planeación de problemas, sistemas expertos, representación de conocimiento y otras categorías tales como instrucción y tutoría inteligente asistida por computadoras, juegos, pronunciación, programación automática y herramientas de inteligencia artificial.

##### **1.5.1. Sistemas Expertos**

Son programas computacionales diseñados para actuar como expertos en un dominio particular restringido<sup>3</sup>. Es importante debido a que trabaja con conocimiento en lugar del tradicional dato. La investigación en representación de conocimiento es central para el avance de los sistemas expertos. Los componentes básicos de un sistema experto son:

- a) Base de Conocimiento. Es la representación del conocimiento del dominio para la solución de problemas específicos, normalmente dicho conocimiento se estructura en forma modular y declarativa.
- b) Máquina de Inferencia. Es el procedimiento que se encarga de realizar el razonamiento a partir de los datos y utilizando el conocimiento acumulado en la base de conocimiento. Es genérica, es decir, se puede aplicar a diferentes dominios sólo cambiando la base de conocimiento.
- c) Memoria de Trabajo. Es el lugar donde se almacenan los datos de entrada y conclusiones intermedias que se van generando durante el proceso de razonamiento.

---

<sup>3</sup> Hace referencia al área de conocimiento de un experto, esta debe contener conocimiento altamente especializado.

- d) Interfaz de Usuario. Se refiere a la entrada/salida al usuario del sistema, incluyendo, normalmente, mecanismos de pregunta (porqué) y de explicación (cómo).
- e) Interfaz de Adquisición. Hace referencia a la interface para la adquisición del conocimiento del dominio, puede incluir mecanismos para facilitar su adquisición y depuramiento interactivo y para automatizar la adquisición (aprendizaje).

Los Sistemas Expertos son uno de los puntos que componen las investigaciones en el campo de la IA. Un sistema de ordenadores que trabaje con técnicas de IA deberá estar en situación de combinar información de forma "inteligente", alcanzar conclusiones y justificarlas. Los Sistemas Expertos son una expresión de los sistemas basados en el conocimiento<sup>4</sup>. Con la aplicación de técnicas de Inteligencia Artificial finaliza la transición del procesamiento de datos al procesamiento de conocimientos.

Los sistemas expertos se aplican por norma general en problemas que implican un procedimiento basado en el conocimiento. Un procedimiento de solución basado en el conocimiento comprende las siguientes capacidades:

- a) Utilización de normas o estructuras que contengan conocimientos y experiencias de expertos especializados.
- b) Deducción lógica de conclusiones.
- c) Capacidad para interpretar datos ambiguos.
- d) Manejo de conocimientos afectados por valores probabilísticos.

Debido a la gran capacidad de almacenamiento de las computadoras, los sistemas expertos tienen el potencial de interpretar estadísticas, para interpretar el formalismo de representación conocido como reglas de producción. Un sistema experto trabaja de manera parecida a un detective encargado de resolver un misterio. Utilizando la información, y un formalismo de representación de conocimiento, un sistema experto puede resolver el problema que se le plantee. Por ejemplo un sistema experto diseñado para distinguir aves puede tener la siguiente configuración:

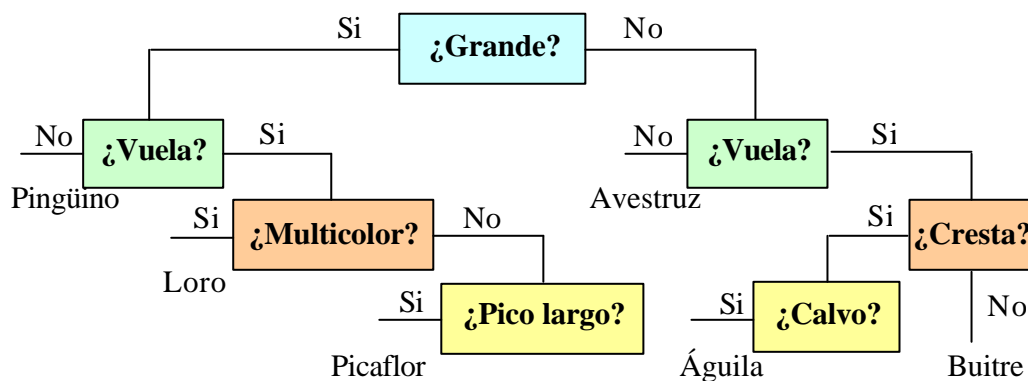


Fig. 1.2. Reglas del Sistema Experto en Aves

### 1.5.2. Procesamiento del Lenguaje Natural

<sup>4</sup> La representación del conocimiento y los sistemas basados en conocimiento se discuten en el capítulo 2.



El Procesamiento del Lenguaje Natural (PLN) es una parte esencial de la IA que investiga y formula mecanismos computacionalmente efectivos que faciliten la interrelación hombre-máquina y permitan una comunicación mucho más fluida y menos rígida que los lenguajes formales. Todo sistema de PLN intenta simular un comportamiento lingüístico humano; para ello debe tomar conciencia tanto de las estructuras propias del lenguaje, como del conocimiento general acerca del universo de discurso. De esta forma, una persona que participa en un diálogo sabe cómo combinar las palabras para formar una oración, conoce los significados de las mismas, sabe cómo éstos afectan al significado global de la oración y posee un conocimiento del mundo en general que le permite participar en la conversación.

El PLN hace referencia a programas que permiten la comunicación hombre-máquina en un lenguaje parecido al humano<sup>5</sup>. Cuenta con dos áreas: comprensión del lenguaje natural y generación del lenguaje natural. La comprensión del lenguaje natural investiga los métodos que permitan al ordenador comprender las instrucciones dadas en lenguaje natural, de tal manera que las computadoras puedan entender de manera más fácil al ser humano. La generación del lenguaje natural intenta mostrar que el ordenador puede ser capaz de expresarse en lenguaje ordinario, de tal forma que se pueda entender a las computadoras de una manera mucho más sencilla.

### **1.5.3. Reconocimiento del Habla**

Hace ya tiempo que se estudia la posibilidad de desarrollar interfaces hombre-máquina controlados por la voz para sustituir en ciertas aplicaciones a las interfaces tradicionales basadas en teclados, paneles y dispositivos similares. Este nuevo tipo de interfaz constaría de dos módulos de entrada/salida: uno de reconocimiento de habla, mediante el cual el ordenador sería capaz de extraer información de los comandos orales del operador o usuario, y otro de síntesis de voz, que podría ser una de las vías de presentación de resultados.

La utilización de la voz, y el Reconocimiento de Habla, como vía de dar ordenes a los ordenadores ofrece varias ventajas respecto al método tradicional de comunicación entre el usuario y la máquina:

- a) Hace esta comunicación más rápida, y más agradable para los nuevos usuarios, ya que al ser la forma natural de comunicarse no se necesita ninguna habilidad especial.
- b) Permite tener las manos libres para utilizarlas en alguna otra actividad, a la vez que se van dando ordenes por medio de la voz.
- c) Permite movilidad, ya que la voz se puede enviar a distancia y ser recogida por un micrófono, por oposición a un teclado que no se puede mover de la mesa de trabajo.
- d) Permite acceso remoto, al poder acceder a un ordenador usando la red telefónica, que es la red de comunicaciones más extendida.
- e) Permite la disminución del tamaño de los paneles de control. Piense en el panel de un avión, cuantos conmutadores manuales podrían suprimirse si se utilizara la voz como forma de comunicación con un sistema de control

Esta área de investigación permite que los ordenadores entiendan el habla humana, de tal manera que se puedan oír voces y reconocer palabras habladas, simplificando el proceso de

---

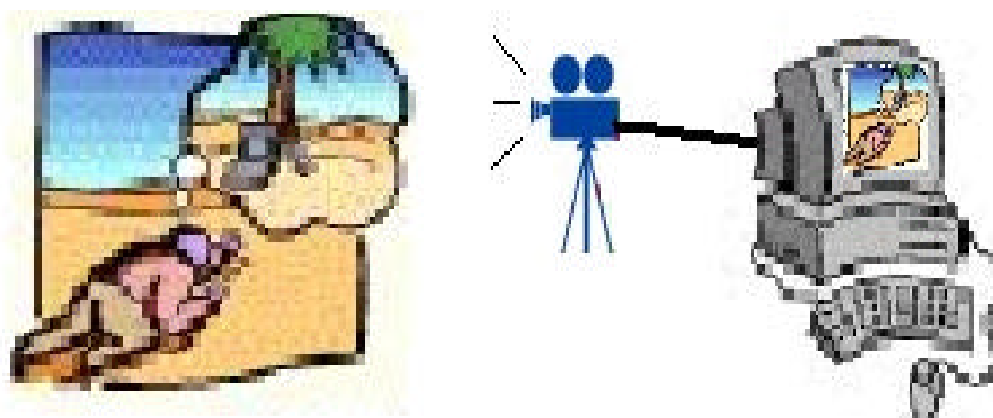
<sup>5</sup> Hace referencia al lenguaje natural.

comunicación interactiva hombre - maquina. Incrementa el método interactivo de comunicación primaria utilizada por las personas, el habla.

#### **1.5.4. Visión por Computadora**

El término Visión por Computadora (VC) dentro del campo de la Inteligencia Artificial puede considerarse como el conjunto de todas aquellas técnicas y modelos que permiten el procesamiento, análisis y explicación de cualquier tipo de información espacial obtenida a través de imágenes digitales. Desde sus inicios la VC ha inspirado sus desarrollos en el estudio del sistema visual humano el cual sugiere la existencia de diferentes tipos de tratamiento de la información visual dependiendo de metas u objetivos específicos, es decir, la información visual percibida es procesada en distintas formas con base en las características particulares de la tarea a realizar, por lo que la VC propone varias técnicas que permiten obtener una representación del mundo a partir del análisis de imágenes obtenidas desde cámaras de video.

Debido a que la información visual es una de las principales fuentes de datos del mundo real, resulta útil el proveer a una computadora digital del sentido de la vista (a partir de imágenes tomadas con cámaras digitales o analógicas), que junto con otros mecanismos como el aprendizaje hagan de esta una herramienta capaz de detectar y ubicar objetos en el mundo real.



**Fig. 1.3. Procedimiento macro para obtener una imagen digital**

Por consiguiente la investigación en visión por ordenador tiene como objetivo dotar a las computadoras con la herramienta de visualización para el entendimiento y comprensión del entorno que la computadora está observando.

#### **1.5.5. Robótica**

Cuando se escucha la palabra Robot, algunas ocasiones se piensa de manera directa en esas películas que han sorprendido por presentar Robots que realizan acciones superiores a las capacidades del ser humano. Los modelos más famosos de robots han sido los creados por George Lucas en su película Star Wars a quienes se conoce como C3PO y R2D2.



**Fig. 1.4. Robots famosos**

Mikell Groover, en su libro *Automation, Production Systems and Computer Integrated Manufacturing*, define al robot industrial como: “una máquina programable, de propósito general, que posee ciertas características antropomórficas, es decir, con características basadas en la figura humana...”. Cabe destacar que la característica antropomórfica más común en nuestros días es la de un brazo mecánico, el cual realiza diversas tareas industriales. Existen en el mercado diversas empresas dedicadas a la fabricación de robots industriales por lo que existen diferentes marcas y modelos. Estos últimos son normalmente asignados para identificarlos o de acuerdo a su función.

Un resumen general de lo que constituye un robot industrial puede ser considerado sobre la base de los siguientes puntos:

- a) Un robot industrial es una máquina programable de propósito general que posee ciertas características antropomórficas.
- b) El componente principal lo constituye el manipulador, el cual consta de varias articulaciones y sus elementos.
- c) Las partes que conforman el manipulador reciben los nombres de cuerpo, brazo, muñeca y efector final o gripper. Otros elementos son el controlador, los mecanismos de entrada y salida de datos y los dispositivos especiales.
- d) Existen dos categorías de efectores finales (grippers): las pinzas y las herramientas. Las pinzas pueden ser de tipo pivotante o de movimiento lineal entre otras. Entre las herramientas se tiene a los desarmadores y las pistolas para soldar.
- e) Los movimientos de un robot están relacionados con los grados de libertad que posea. Un grado de libertad es un número o tipo de movimiento del manipulador. Los grados de libertad se determinan por los movimientos que ejecutan el brazo y la muñeca del robot que pueden ser de uno a tres cada uno.

La investigación en esta área tiende al estudio de las capacidades de los robots de poder insertarse en la sociedad a manera de seres mecánicos capaces de controlar y resolver los problemas discretos y mecanizados de su entorno.

### **1.5.6. Otras Áreas**

Otras áreas significativas para su estudio al interior de la inteligencia artificial son: las lógicas no clásicas como la Lógica Difusa, las Redes Neuronales, los Algoritmos Genéticos, la Realidad Virtual, la Vida Artificial, los Agentes Inteligentes, etc. Estas áreas pueden ser visualizadas en la Fig. 1.5.



**Fig. 1.5. Áreas de la Inteligencia Artificial**

En el resto del texto la visión que se intenta brindar es la comprensión, análisis y aplicación de estas áreas de la inteligencia artificial. No es intención revisar a detalle cada una de estas áreas, sin embargo es menester reconocer la importancia que han cobrado en esta última década de investigación en la inteligencia artificial. La mayor de las expectativas es compartir los principios y fundamentos básicos que proporcionan la importancia natural a estas áreas, tan trilladas y comentadas en este ultimo tiempo.

### **Ejercicios #1**

1. Desarrolle la hipótesis del Sistema de Símbolos Físicos que dice: “Un sistema de símbolos físicos tiene los medios suficientes y necesarios para la acción inteligente”.
2. Escriba, de manera sucinta, las biografías de los padres de la inteligencia artificial, debe incluir entre otros a: Marvin Minsky, John McCarthy, Claude Shannon , Nataniel Rochester, Allen Newell y Herbert Simon.
3. Lea el artículo original<sup>6</sup> de Alan Turing sobre la IA. En este artículo Turing predijo que para el año 200 es probable que una computadora tenga 30% de oportunidad de aprobar una prueba de Turing con duración de cinco minutos aplicada por un evaluador inexperto. ¿Considera razonable lo anterior?
4. Son bien conocidos ciertos tipos de problemas inmanejables por las computadoras, así como otros que probablemente evidencian que ninguna computadora puede tomar decisiones. ¿Significa esto entonces que la IA es un imposible?
5. Algunos consideran que la percepción y las habilidades motoras son la parte más importante de la inteligencia y que las capacidades de “alto nivel” son más bien parásitas (meros agregados a las capacidades básicas). Es un hecho que la mayor parte de la evolución y del cerebro se han concentrado en la percepción y las habilidades motoras, en tanto se ha encontrado que en la IA tareas como juegos e inferencia lógica resultan más sencillas, en muchos sentidos, que percibir y actuar en el mundo real. ¿Consideraría usted que ha sido un error la concentración tradicional de la IA en las capacidades cognoscitivas de alto nivel?

<sup>6</sup> Turing, A.M. Computing machinery and intelligence. *Mind*, 59: 433-460. 1950.

## Bibliografía

- Borrajo, D. y otros *Inteligencia artificial: Métodos y técnicas*. Centro de estudios Ramón Areces, 1993.
- Copeland, J. *Inteligencia artificial*. Editorial Alianza, 1996.
- Crevier, D. *Inteligencia artificial*. Editorial Acento, 1996.
- Dean, T.; Allen, J. & Aloimonos, Y. *Artificial Intelligence: Theory and Practice*. Benjamin Cummings, 1995.
- Ginsberg, M. *Essentials of Artificial Intelligence*. Morgan Kaufmann Publishers, 1993.
- Haton, J.P. & Haton, M.C. *La Inteligencia Artificial: una aproximación*. Editorial Paidós, 1991.
- Luger, G.F. & Stubblefield, W.A. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. 2nd edition. Benjamin Cummings, 1993.
- McCorduck, P. *Máquinas que piensan*. Editorial Tecnos, 1991.
- Milner, W.L. *Common Lisp: A tutorial*. Prentice Hall, 1988.
- Mira, J.; Delgado, A.E.; Boticario, J.G. & Díez, F.J. *Aspectos básicos de la Inteligencia Artificial*. Editorial Sanz y Torres, 1995.
- Rich, E. & Knight, K. *Inteligencia Artificial*. Segunda edición. Editorial McGraw-Hill Interamericana, 1994.
- Russell, S. y Norvig, P. *Artificial Intelligence: A Modern Approach*. Prentice-Hall International, 1995.
- Shirai, Y. y Tsujii, J. *Inteligencia Artificial: Conceptos, técnicas y aplicaciones*. Editorial Ariel, 1987.
- Tanimoto, S.L. *The Elements of Artificial Intelligence: Using Common Lisp*. Computer Science Pres, 1990.
- Winston, P.R. *Inteligencia artificial*. Tercera edición. Editorial Addison-Wesley, 1994.

## Referencias Electrónicas

- Advanced Computing Research Centre (ACRC).  
<http://www.fen.bris.ac.uk/engmaths/research/aigroup/acrc.html>
- Asociación Argentina de Inteligencia Artificial.  
<http://www.laguia.com.ar/aaia.htm>
- Aircenter  
<http://www.aircenter.net/>
- Aphex's Website  
<http://www.aphex.ipfox.com/>
- Aplicación de las calculadoras  
[http://decsai.ugr.es/docencia/lm/lm\\_ac.html](http://decsai.ugr.es/docencia/lm/lm_ac.html)
- Artificial Intelligence group at the University of Bristol  
<http://www.fen.bris.ac.uk/engmaths/research/aigroup/aigroup.html>
- Bibliografía de Inteligencia Artificial  
<http://grial.uc3m.es/docweb/ia/bibliografia.html>

Descripción de "Inteligencia Artificial"

<http://grial.uc3m.es/docweb/ia/descripcion.html>

DLSI: Reconocimiento de Formas e Inteligencia Artificial

<http://www.dlsi.ua.es/rfia/rfia.html>

ELITE - European Laboratory for Intelligent Techniques Engineering

<http://www.mitgmbh.de/elite/elite.html>

Hybrid Intelligent Systems

<http://www.cas.american.edu/research/his.html>

Laboratorio de Inteligencia Artificial

<http://delicias.dia.fi.upm.es/>

Octava Conferencia Nacional en Inteligencia Artificial

<http://www.cc.gatech.edu/aimosaic/OLD/cniase95/call-for-papers-spanish.html>

¿Qué es la Inteligencia Artificial (AI)?

[http://www.fortunecity.com/tinpan/radiohead/92/qes\\_ai.htm](http://www.fortunecity.com/tinpan/radiohead/92/qes_ai.htm)

Research Group on Artificial Intelligence

<http://www.etse.urv.es/LSI/rgai.html>

# 2

## REPRESENTACION DE CONOCIMIENTO

### 2.1. INTRODUCCION

La representación constituye el núcleo de la ciencia de las computadoras. Cada pieza de código escrita es significativa para la representación de algún objeto. En la actualidad la representación del conocimiento es potencialmente investigada, de tal forma que cualquier libro que trate sobre inteligencia artificial o sistemas expertos considera a la representación del conocimiento como un resultado central.

Los ingredientes básicos para la representación del conocimiento son tres: el primero está referido a un lenguaje de representación, el segundo a la capacidad de inferencia de la representación y el tercero al conocimiento del dominio.

En general una representación debe de tener dos capacidades: por un lado una expresividad adecuada y por otro una eficiencia de razonamiento. La expresividad y el razonamiento le confieren al formalismo la capacidad adecuada para ser considerado como una alternativa útil para la representación.

Se establece, a priori la definición de los formalismos, que los criterios para juzgar una representación adecuada son tres: lo primero que hay que observar es la capacidad lógica, referida a que el formalismo sea capaz de expresar el conocimiento que se desea expresar; lo segundo es el poderío heurístico, que se refiere a la capacidad para resolver problemas utilizando inferencias; finalmente el tercer criterio es la conveniencia de la notación, que significa la simplicidad para acceder al conocimiento y la facilidad de su entendimiento.

### 2.2. DEFINICIÓN

La definición del concepto de representación de conocimiento se realiza desde dos puntos de vista, atómico y holístico.

#### 2.2.1. Vista Atómica

En su etimología la palabra representación viene del latín “*Repraesentare*” que significa: hacer aparecer como presente. Por otro lado la palabra Conocimiento deriva de la palabra latina “*Gnosco*” que significa: aquello que ha sido sujeto al acto de reconocimiento. Si se combinan ambos significados se tiene que: “la representación de conocimiento es hacer que aparezca como presente lo que ha sido sujeto al acto de reconocimiento”.

Otra definición sostiene que la representación de conocimiento puede ser algo como la correspondencia de reglas conocidas y el estado del mundo en alguna estructura apropiada.

#### 2.2.2. Vista Holística

Esta vista corresponde al holismo que dice, él todo es mas que la suma de sus partes<sup>7</sup>. Así la representación de conocimiento es mas que solo la conjunción de los conceptos de conocimiento y representación.

### a) Modelo Realístico Simple

Este modelo conceptual corresponde al proceso de cognición. El dominio de interés es alguna parte (P) del mundo (W) y el resultado será una teoría acerca de dicha parte P. Un modelo realístico-simple es una abstracción extractada de lo concreto. El proceso de abstracción o construcción tiene tres fases:

- Se realiza la selección de un conjunto de datos acerca del mundo. La selección es en principio sin ninguna teoría de respaldo.
- Los datos se combinan en una estructura.
- Se refina la estructura hasta que la misma refleja al mundo concreto, lo suficientemente bien.

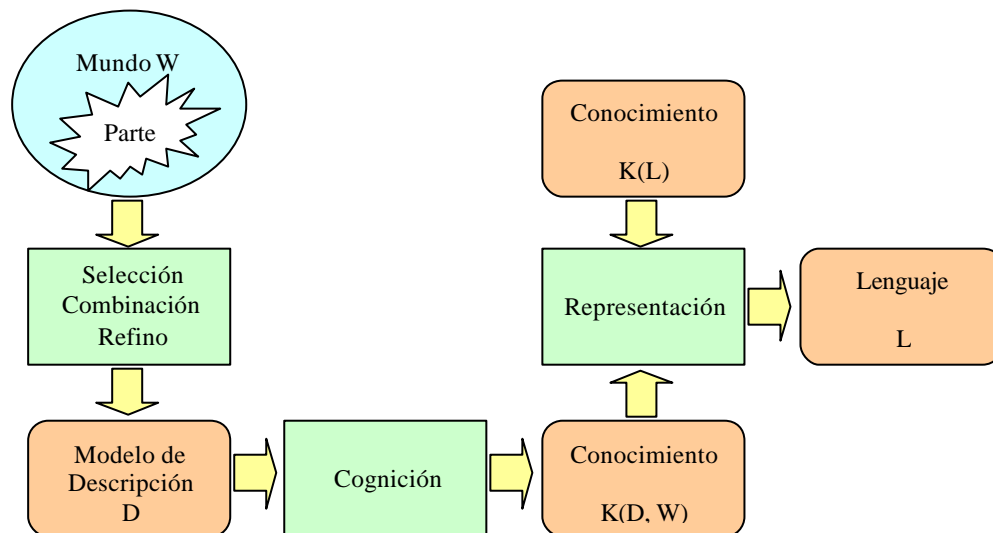


Fig. 2.1. Modelo Realístico Simple

Ejemplo:

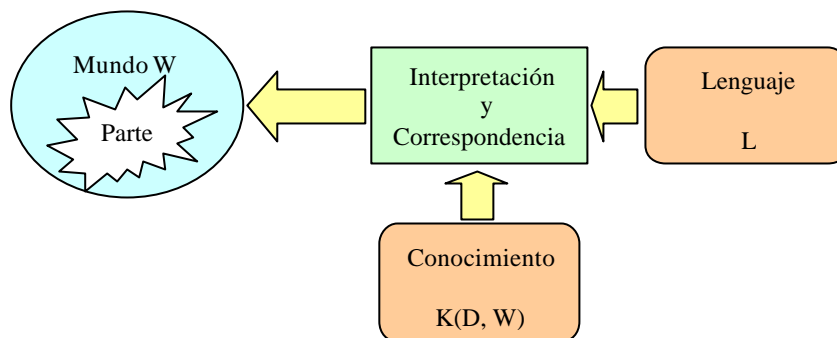
Considere a una persona X preguntarse ¿por qué un yate anclado en un puerto no se encauza por efecto del viento?. X primero observa el yate y toma notas acerca de su forma (fase a); luego X construye un modelo a escala en una pieza de madera (fase b); en tercer lugar examina el modelo en un cubo lleno de agua y lo refina hasta que el mismo se comporte como el yate del puerto (fase c). Finalmente X entiende que una quilla grande es crucial para los movimientos del yate.

### b) Modelo Lógico Matemático

Este modelo conceptual corresponde a la asignación de significado a la representación. El modelo lógico matemático de una teoría o realización, es una estructura para la cual todos los axiomas de la teoría son validos.

<sup>7</sup> Según Aristóteles 384-322 a. C.





**Fig. 2.1. Modelo Lógico Matemático**

Ejemplo:

Considere la teoría de cómo el yate se comporta en el agua. Un teorema de esta teoría puede ser expresado de manera muy informal como: “un yate estable tiene una quilla grande”. Una realización de la teoría son yates de madera con grandes quillas de hierro.

### 2.2.3. Notas Finales

Sobre la base de lo observado en los dos modelos holísticos para la representación de conocimiento se puede decir que:

- El conocimiento es algo que tiene los seres humanos. Se relaciona con algún mundo  $W$ , entendido o conocido por una persona.
- La representación es un proceso, pero no necesariamente un proceso de representar conocimiento.
- El conocimiento representado es expresado en algún lenguaje<sup>8</sup>  $L$ . Para hablar de conocimiento representado también se requiere que el mismo pueda ser interpretado por alguna persona.
- La representación de conocimiento es el acto de representar conocimiento.

Con lo mencionado en los anteriores cuatro puntos se tiene que la representación de conocimiento involucra a:

- Un dominio del discurso, es decir algún mundo físico abstracto  $W$ .
- Un lenguaje  $L$ , en el que se representa conocimiento acerca del mundo  $W$ .
- Conexiones entre el mundo y el conocimiento representado y viceversa.
- La representación<sup>9</sup> de conocimiento de algún mundo  $W$ , en algún lenguaje  $L$ .
- La interpretación<sup>10</sup> del conocimiento representado.

## 2.3. EVOLUCION DE LOS SISTEMAS BASADOS EN CONOCIMIENTO

### 2.3.1. Dato

<sup>8</sup> En este contexto el lenguaje debe ser entendido en el sentido más amplio que sea posible.

<sup>9</sup> Se considera como una codificación del conocimiento.

<sup>10</sup> Se considera como una decodificación del conocimiento.

El concepto de dato es, por supuesto, el concepto central en la ciencia de las computadoras. Dicha afirmación es evidente en la siguiente definición proporcionada por Eriksen, Helms y Romer en 1975: “Los datos constituyen una representación formalizada de hechos o ideas en forma tal que puedan ser comunicados o transformados mediante un proceso”.

### 2.3.2. Sistema de Programación Convencional

Un sistema de programación convencional consta de un conjunto de archivos, los que normalmente pueden ser utilizados por programas, además de un programa que trabaja sobre esos archivos. El programa y el conjunto de archivos están relacionados de manera cerrada.

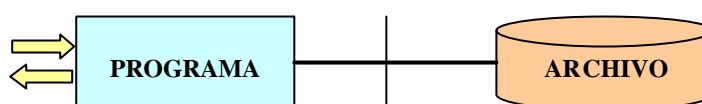


Fig. 2.3. Sistema de Programación Convencional

Ejemplo:

Programa convencional para la emisión de una planilla de pagos.

| Programa                           | Archivo                     |
|------------------------------------|-----------------------------|
| Método de búsqueda                 | Identificación de empleados |
| Método de clasificación            | Numero de horas trabajadas  |
| Reglas de cálculo del total ganado | Beneficios                  |

### 2.3.3. Sistema de Base de Datos

De manera esencial un Sistema de Base de Datos es un sistema computarizado para la acumulación de registros. Consta de una Base de Datos (BD), descriptores de la Base de Datos y un Sistema de Administración de la Base de Datos<sup>11</sup>, el cual es utilizado por un programa.

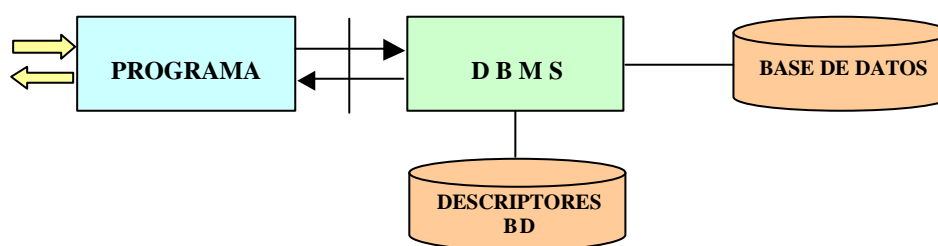


Fig. 2.4. Sistema de Base de Datos

Una BD es una colección de archivos de datos computarizados. Los descriptores de la BD describen por ejemplo los campos que están en los archivos y la longitud de los mismos. El DBMS proporciona facilidades al usuario<sup>12</sup> relativos a la operación sobre los archivos. Algunas de las operaciones son: adicionar y remover archivos, insertar, recuperar, actualizar y eliminar datos de los archivos existentes.

<sup>11</sup> Corresponde al Data Base Management System (DBMS).

<sup>12</sup> En este caso incluso puede ser un programa.

Mas aún el DBMS forma el control y cuidado del mantenimiento de la integridad de la BD. El sistema de BD es una entidad general que puede ser utilizado por muchos usuarios programas en diferentes conexiones.

Ejemplo:

Programa convencional para la emisión de una planilla de pagos.

| Programa                | DBMS                    | Base de Datos              |
|-------------------------|-------------------------|----------------------------|
| Cálculo de total ganado | Método de búsqueda      | Identificación empleado    |
| Uso del DBMS            | Método de clasificación | Numero de horas trabajadas |
|                         | Método de actualización | Beneficios                 |

#### 2.3.4. Sistema Basado en Conocimiento

Un Sistema Basado en Conocimiento (SBC) puede ser visto como la evolución de un sistema de BD. En muchos casos incluye un sistema de BD, e involucra una maquina de inferencias que opera sobre la base de conocimiento interactuando con el sistema de BD.

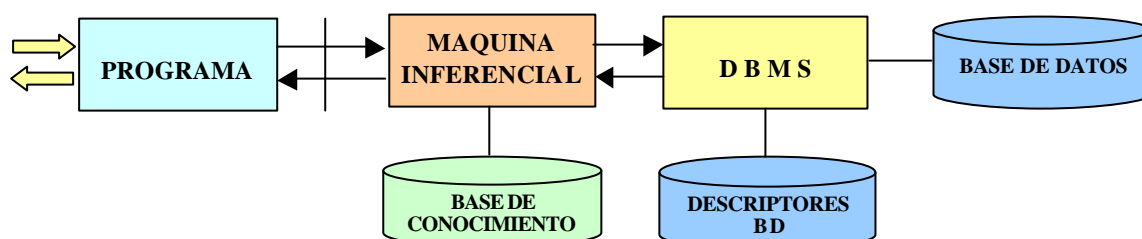


Fig. 2.5. Sistema Basado en Conocimiento

Ejemplo:

Programa convencional para la emisión de una planilla de pagos.

| Programa | Máquina Inferencial         | Base de Conocimiento                    | DBMS                    | Base de Datos           |
|----------|-----------------------------|---|-------------------------|-------------------------|
| Dialogo  | Control de la BC y del DBMS | Reglas para el cálculo del total ganado | Método de búsqueda      | Identificación empleado |
|          |                             |   | Método de clasificación | Beneficios              |
|          |                             |   | Método de actualización |                         |

#### 2.4. FORMALISMOS DE REPRESENTACION DEL CONOCIMIENTO

Es útil considerar la representación de conocimiento como una combinación de estructuras de datos y procesamientos interpretativos capaces de hacer que un programa exhiba un comportamiento experto.

Un sistema de inteligencia artificial necesita distintos tipos de conocimiento para comportarse inteligentemente o de modo experto. Algunos tipos son:

- Conocimiento de Objetos: donde se guardan los hechos relacionados con objetos (saber que...).

- b) Conocimiento de acciones y sucesos: donde se guardan los hechos sobre los sucesos ocurridos (saber que...).
- c) Conocimiento sobre prestaciones: donde se guardan habilidades como montar una bicicleta, nadar, jugar fútbol, etc. (saber como...).
- d) Meta conocimiento: saber sobre lo que se sabe, como ejemplo conocer las limitaciones de nuestro conocimiento.

### 2.4.1. Redes Semánticas

La red semántica fue introducida por Quillian en 1966, básicamente es un modelo de memoria humana que se encarga de capturar la semántica de las palabras y lograr uso del significado de manera parecida a los seres humanos. Se llama red semántica porque se utilizó originalmente para representar el sentido en las expresiones escritas en lenguaje natural.

Desde la perspectiva de la teoría de grafos se dice que una red semántica es un grafo compuesto por nodos y arcos. Los nodos representan conceptos de palabras y los arcos se encargan de enlazar los conceptos para establecer las definiciones. Cada palabra o nodo conceptual se considera la cabeza de un plano que contiene su definición. Los enlaces en el plano representan su definición.

En una red semántica existen apuntadores a superclases<sup>13</sup>, modificaciones, disyunciones, conjunciones y las relaciones sujeto-objeto. Los apuntadores fuera del plano hacen referencia a otros objetos en los cuales se definen.

Una red semántica distingue dos enlaces principales:

- a) subclase es-un: las clases de “arriba” están definidas en términos de conceptos generales que se asume se cumplen en todas sus subclases.
- b) modificadores: propiedades particulares de conceptos específicos.

Ejemplo 1:

Relación semántica simple que representa “Piolín es un ave”.

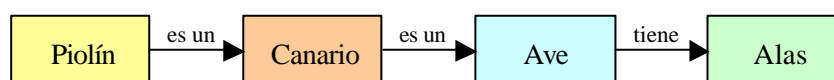


Fig. 2.6. Red semántica “Piolín es un ave”

<sup>13</sup> El enlace más notable entre clases es el enlace: es-un (IS-A). A través de este enlace se establecen también relaciones jerárquicas y herencia.

Ejemplo 2:

Relación semántica para representar la relación entre una silla y su propietario.

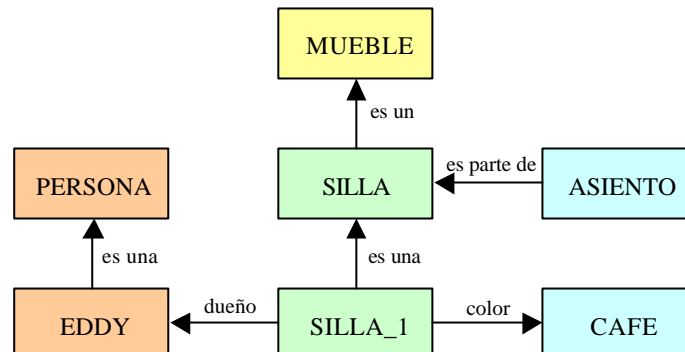


Fig. 2.6. Red semántica propietario de una silla

Las redes semánticas también son denominadas redes asociativas, debido principalmente al carácter asociativo de los enlaces<sup>14</sup>.

#### 2.4.2. Armazones<sup>15</sup>

En pocas palabras se puede decir que un armazón es una estructura de datos que se encarga de representar situaciones prototípicas. Una de las ideas intuitivas detrás de los armazones, es que la memoria se basa mucho en estereotipos<sup>16</sup>. Los sistemas de armazones razonan acerca de clases de objetos utilizando representaciones prototípicas, estas pueden modificarse para capturar las complejidades del mundo real.

Los armazones son grandes trozos de información que hacen énfasis en el razonamiento por omisión (default). Constan básicamente de una estructura de datos, con la característica principal de que esta estructura puede incluir información declarativa y procedimental. Un armazón consta de un conjunto de ranuras que describen el aspecto de los objetos. Estas ranuras se llenan mediante otros armazones que describen otros objetos. Asociado con cada ranura puede existir un conjunto de condiciones que deben cumplir las entidades que vayan a llenarla. Cada ranura puede también llenarse con un valor por defecto, de forma que, en ausencia de información específica, pueda suponerse que las cosas sean lo que usualmente son. Puede también asociarse información procedimental con cada ranura.

Los armazones están puestos en una jerarquía en donde los armazones de “abajo” pueden heredar los valores de las ranuras de los armazones de “arriba”. Normalmente la herencia se hace por medio de los arcos es-un (*is-a*).

<sup>14</sup> Conocidas también como relaciones o asociaciones.

<sup>15</sup> En el lenguaje inglés corresponde a la palabra "frame", en algunos textos está traducido como marco, la traducción correcta nombrada por Frost corresponde a "Armadura" o "Armazón".

<sup>16</sup> En este caso hace referencia a las propiedades típicas de los objetos.

Ejemplo 1:

Especificar el almacén para representar a un camión.

(Camión: Almacén:

(nombre camión)

(es-un objeto)

(color rojo)

(llantas 10) ...)

Ejemplo 2:

Especificar el almacén para describir la silla de Eddy

Almacén: SILLA

Especialización de MUEBLE

Numero de patas un entero (por omisión 4)

Tipo de forro tela, cuero, etc.

Numero de brazos 0, 1 o 2

Almacén: SILLA DE EDDY

Especialización de SILLA

Numero de patas 4

Tipo de forro cuero

Numero de brazos 0

Almacén: Persona

Nombre hilera de caracteres

Dirección hilera de caracteres

Edad rango entre 0 y 100 (en años)

Profesión hilera de caracteres

#### **Nota:**

En el almacén persona se puede adicionar una ranura denominada fecha de nacimiento y un procedimiento que permita calcular la edad actual de la persona. Aquí se puede notar una diferencia sustancial de los almacenes, la posibilidad de manejar conocimiento procedimental.

### **2.4.3. Reglas de Producción**

La inteligencia se asocia muchas veces con las “regularidades” y el comportamiento inteligente parece que se ejecuta a través de reglas. Newell y Simon en los años setenta proponen los sistemas de producción como un modelo psicológico del comportamiento humano. En este modelo parte del conocimiento humano se representa en forma de producciones o reglas de producción.

Las reglas de producción se asemejan al proceso de memoria humano: memoria a corto plazo (deducciones intermedias) y memoria a largo plazo (producciones). Normalmente las reglas de producción se ven como un formalismo en el cual se representa el conocimiento de manera simple y es el formalismo más utilizado en los sistemas expertos.

El formalismo de reglas de producción debe su origen a los trabajos de Post, enriquecidos en la lingüística con los trabajos de Noam Chomsky. Una regla de producción consta de un par ordenado de símbolos, miembros izquierdo y miembro derecho, los que forman una pareja de situación-acción. Existe una amplia variedad de este tipo de sistemas de producción, desde los

más simples, con un antecedente y un consecuente, hasta las más complejas de múltiples antecedentes y múltiples consecuentes. A cada regla de producción puede añadirse un factor de certeza probabilístico tanto al antecedente como al consecuente.

En una regla de producción se pueden observar los siguientes elementos:

- a) Un conjunto N de nombres de objetos en el dominio
- b) Un conjunto P de propiedades que representan atributos de los objetos
- c) Un conjunto V de valores que los atributos pueden tener

De manera general para representar una regla de producción se utiliza una tripleta: (objeto atributo valor). De modo formal las reglas se colocan como:  $P_1, P_2, \dots, P_m \Rightarrow Q_1, \dots, Q_n$ , que significa: si las condiciones  $P_1$  y  $P_2$  y ... y  $P_m$  se cumplen entonces se realiza las acciones  $Q_1$  y ... y  $Q_n$ .

La estructura general de una regla de producción es:

|          |                   |              |
|----------|-------------------|--------------|
| SI       | miembro izquierdo | (suposición) |
| ENTONCES | miembro derecho   | (conclusión) |

Las propiedades de las reglas de producción son las siguientes:

- a) Modularidad. Cada regla define un pequeño y relativamente independiente pedazo de conocimiento.
- b) Incrementalidad. Nuevas reglas pueden ser añadidas a la base de conocimiento de manera relativamente independiente de las demás.
- c) Modificabilidad. Como consecuencia de la modularidad, las reglas viejas pueden ser modificadas.
- d) Transparencia. Hacer referencia a la habilidad de explicar sus decisiones y soluciones

Finalmente se puede decir que un sistema de producción tiene:

- a) un conjunto de reglas (base de conocimiento)
- b) un interprete de reglas o máquina de inferencia (que decide que regla aplicar, controla la actividad del sistema)
- c) una memoria de trabajo (que guarda los datos, metas, y resultados intermedios)

Ejemplo No 1:

IF: (\$AND (SAME CNTXT INFECT PRIMARY\_BACTEREMIA)  
(MEMBF CNTXT SITE STERILESITES)  
(SAME CNTXT PORTAL GI))  
THEN: (CONCLUDE CNTXT INDENT BACTEROIDES TALLY.7)

Que puede ser interpretado como:

SI: la infección es una bacteremia primaria, y el lugar de cultivo es un sitio estéril y el conducto sospechoso para su entrada al organismo es vía canal gastrointestinal.

ENTONCES: Existe una evidencia sugestiva de 0.7 de que la identidad del organismo sea un bacteroide.

Ejemplo 2:

SI Animal es-un carnívoro Y Animal color café Y Animal tiene rayas  
ENTONCES Animal es tigre

Ejemplo No 3:

SI un empleado divulga proyectos secretos.  
ENTONCES él viola la política de la compañía

Ejemplo No 4:

SI un estudiante no lee mucho (.80)  
ENTONCES corre el peligro de quedar desactualizado (.95)

En este ejemplo puede observarse los factores de certeza probabilísticos en la suposición y en la conclusión.

#### 2.4.4. Redes de Herencia Estructurales

Como principio de la explicación se indica que Ronald J. Brachmann identifica cuatro niveles diferentes en los que se utilizan las redes semánticas:

- a) El nivel de implementación, en el cual las redes son vistas como estructuras de datos, con punteros y listas como primitivas.
- b) El nivel lógico, en el cual los operadores lógicos y predicados son primitivas, llevando a la red semántica como una variante notacional de alguna lógica.
- c) El nivel conceptual que enfoca sobre relaciones conceptuales y semánticas, las primitivas en este nivel son, por ejemplo, las relaciones de casos y acciones primitivas y objetos<sup>17</sup>.
- d) El nivel lingüístico, en el cual las palabras arbitrarias y las expresiones son utilizadas como primitivas, las cuales son por tanto grandemente dependientes del lenguaje.

Además Brachman argumenta que un nivel importante no es considerado en los anteriores cuatro niveles es el nivel epistemológico<sup>18</sup>. El nivel epistemológico es una capa intermedia entre el nivel lógico y conceptual. En este nivel son formadas e interrelacionadas descripciones intencionales. El formalismo denominado red de herencia estructural se encarga de capturar el nivel epistemológico utilizando un conjunto pequeño de primitivas epistemológicas.

Los principales bloques de construcción de este formalismo son los *conceptos*. Los conceptos son descritos por superconceptos, por roles, que son posibles relaciones a otros conceptos, y por descripciones estructurales, que se encargan de contar las relaciones entre roles.

Los conceptos son en su división:

- a) Conceptos genéricos: destinados para denotar clases de individuos.
- b) Conceptos individuales: destinados para denotar individuos.

La clase formadora de los conceptos puede ser dividida a su vez en:

- a) Conceptos definidos: completamente determinados por su descripción.
- b) Conceptos primitivos: parcialmente determinados por su descripción.

---

<sup>17</sup> En este nivel conocidos como grafos de dependencia conceptual.

<sup>18</sup> Hace referencia a lo estructural o descriptivo.



- c) Clase natural: que no puede ser descrita de forma exhaustiva, por ejemplo las especies de animales.

Ejemplo:

Taxonomía simple de conceptos

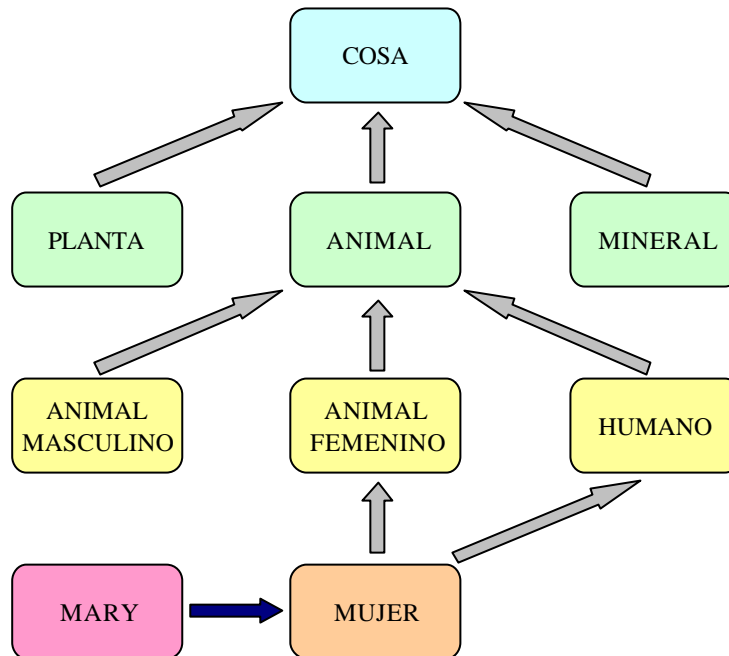


Fig. 2.7. Taxonomía simple de conceptos

## 2.4.5. Representación Terminológica

La primera implementación de las redes de herencia estructural fue KLONE<sup>19</sup>, que evolucionó con el paso de los años y ha provocado esfuerzos de investigación substanciales en la forma de aplicaciones, trabajo teórico y el desarrollo de sistemas similares, documentados en muchos eventos y reuniones.

KL-ONE ha logrado un estatus afrontado a los pocos esfuerzos en la breve historia de la IA, en un periodo significativo de tiempo (al menos seis años) y sobre un numero de proyectos, ha servido como el fundamento central para mucha de la investigación básica sobre la representación del conocimiento, y al mismo tiempo ha proporcionado soporte para la representación en un numero grande de sistemas implementados de IA.

En esencia KL-ONE representa:

- las relaciones entre un concepto y sus roles,
- las relaciones entre un concepto y sus descripciones estructurales,
- la estructura interna de un rol,
- la estructura interna de una descripción estructural,
- relaciones entre partes de las descripciones estructurales y los roles.

<sup>19</sup> Esta denominación original sufrió una variación para denominarse KL-ONE.

KL-ONE distingue entre conocimiento terminológico y conocimiento asercional. El nivel terminológico tiene que ver con un conjunto de descripciones intencionales normadas por un pequeño conjunto de operadores formadores de conceptos y sus relaciones taxonómicas, las cuales están determinadas por sus propiedades estructurales. Esas relaciones taxonómicas no tienen importancia asercional pero pueden utilizarse las descripciones para hacer aserciones acerca del mundo, y por virtud de su estructura pueden realizar inferencias adicionales.

En esencia KL-ONE ha resultado sobre el interés en la funcionalidad, en la que en lugar de preguntar. "¿Que estructuras debería mantener el sistema para el usuario?" se pregunta "¿Que es lo que exactamente debería hacer el sistema para el usuario?". Visto desde este ángulo KL-ONE puede ser dividido en dos componentes:

- a) TBOX es un componente terminológico utilizado para hacer definiciones y para razonar acerca de las relaciones entre esas definiciones.
- b) ABOX es un componente utilizado para hacer aserciones.

Mientras que el TBOX está muy bien definido, el tratamiento del ABOX en el KL-ONE ha sido superficial y fortuito.

#### 2.4.6. Guiones<sup>20</sup>

Un guión es una forma de representar una situación prototípica<sup>21</sup>, pero en lugar de tener una descripción de un objeto, el guión describe *una secuencia de eventos*. A diferencia del armazón, se presenta en un contexto particular. Para describir una secuencia de eventos, el guión utiliza un conjunto de ranuras que contienen información acerca de gente, objetos y acciones involucradas en los eventos.

Un guión es una estructura utilizada para guardar prototipos de secuencias de sucesos. Se pueden emplear muchos componentes diferentes para construir un guión. Algunos de los más comunes son:

- a) Condiciones de Entrada: Condiciones que deben existir para que el libreto se pueda aplicar.
- b) Resultados: Condiciones que serán verdaderas después de que hayan ocurrido los eventos en el guión.
- c) Utillería: Ranuras que representan objetos involucrados en el guión.
- d) Papeles: Ranuras que representan agentes que realizan acciones en el guión.
- e) Escenas: Secuencias específicas de eventos que hacen al guión.

Ejemplo:

Guión que representa el viaje a un estadio de fútbol.

|  |  |
|--|--|
| Guión : Viaje al estadio                             |  |
| Utilería:<br>Automóvil<br>Llaves<br>Puerta automóvil | Escena 1. ARRANQUE<br>Dueño busca llaves<br>Dueño abre puerta automóvil<br>Dueño arranca automóvil |

<sup>20</sup> Traducción de la palabra inglesa Script.

<sup>21</sup> En algún sentido parecido a los armazones.

|   |  |
|---|--|
| Espacio de parqueo  | Dueño pone automóvil en marcha   |
| Papeles: (Roles)<br>Dueño (propietario)<br>Acomodador Vehículos (Valet) | Escena 2. CONDUCCION<br>Dueño encuentra transito libre<br>Dueño ingresa al trafico<br>Dueño se dirige al estadio   |
| Condiciones de ingreso:<br>Dueño y automóvil<br>en punto de partida     | Escena 3. CONTACTO CON ACOMODADOR<br>Dueño buscar primer espacio libre<br>Dueño parquea automóvil<br>Dueño sale del automóvil<br>Dueño encarga al acomodador |

RESULTADO: Propietario en el estadio y automóvil parqueado

## Ejercicios #2

1. Establecer un modelo conjunto que utilice los dos modelos holísticos observados en este capítulo..
2. Desarrolle un método para representar conocimiento con redes semánticas a partir de los armazones.
3. Complemente su visión de las redes semánticas investigando lo referente a los sistemas de redes semánticas ARCH y SCHOLAR.
4. Construir una red semántica que represente la información relativa a una competencia ciclista.
5. Representar conocimiento con guiones para viajar de una capital departamental de Bolivia a otra, empleando el transporte adecuado (emplear costo-beneficio).
6. Empleando el modelo conjunto realístico-simple y lógico-matemático, desarrolle el marco de explicación de la geometría euclidiana.
7. Establecer las reglas necesarias para realizar el diagnóstico de evaluación de los alumnos en una determinada materia de un plan de estudios.
8. Definir una red de herencia estructural para representar el conocimiento de la burocracia (emplear la Dirección de una determinada Carrera universitaria y los diferentes procesos que realiza). Establecer roles y restricciones.
9. Elegir el formalismo adecuado para representar el conocimiento asociado a la compra de una computadora, la computadora debe ser utilizada como cliente de un sistema distribuido.
10. Construir un guión para representar la ingesta de alimentos por parte de un comensal con dinero en un restaurante.

## Bibliografía

Lucas, P. y Van der Gaag, L. *Principles of Expert Systems*. Addison Wesley, 1991.

Brachman, R y Levesque, H. *Readings in Knowledge Representation*. Morgan Kaufmann, 1985.

Russel, S., Norvig, P., *Artificial Intelligence: A Modern Approach*, Prentice-Hall, 1995.

Jackson, P. *Introduction to Expert Systems*. Addison-Wesley, 1990 (2a. edición).

Winston, P., *Artificial Intelligence*. Addison-Wesley (Tercera Edición) 1992.

Shapiro, S.C. *Encyclopedia of Artificial Intelligence*. Wiley, New York (segunda edición), 1992.

Is the Brain Mind a Computer Program? John R. Searle. *Scientific American*, Jan. 1990, pp. 26-31.

Could a Machine Think? Paul M. Churchland, Patricia Smith Churchland. *Scientific American*, Jan. 1990, pp. 32-37.

On Computational Wings: Rethinking the Goals of Artificial Intelligence. Kenneth M. Ford, Patrick J. Hayes. *Scientific American*, Vol. 9 (4): pp. 78-83.

Christopher John Hogger, *Essential of Logic Programming*, Oxford University Press, 1990.

## **Referencias Electrónicas**

### **ARPA Knowledge Sharing Effort**

Información y programas relacionados con el ARPA Knowledge Sharing Effort, en Stanford.  
<http://www-ksl.stanford.edu/knowledge-sharing/README.html>

### **CODE4**

Entornos por descripción orientada a conceptos, en la Universidad de Ottawa.  
<http://www.csi.uottawa.ca/~tcl/CODE4.html>

### **Description Logics**

Teorías y sistemas para expresar conocimiento estructurado.  
<http://dl.kr.org/dl/>

### **KADS Esprit Project**

Metodología para el desarrollo basado en el conocimiento.  
<http://www.swi.psy.uva.nl/projects/CommonKADS/home.html>

### **Knowledge-Base Projects**

Proyectos KB alrededor del mundo; lista completada por Peter Clark.  
<http://www.cs.utexas.edu/users/mfkb/related.html>

### **Knowledge Sharing and the Logic Group at Stanford**

Información sobre el KIF (Formato de intercambio de conocimiento) en Stanford.  
<http://hpdce.stanford.edu/logic.html>

### **KQML**

Investigaciones sobre el conocimiento y la manipulación del lenguaje, en la Universidad de Maryland, Baltimore Country.  
<http://www.cs.umbc.edu/kqml/>

### **Multifunctional Knowledge Base Group**

Universidad de Texas en Austin.  
<http://www.cs.utexas.edu/users/mfkb/index.html>

### **SNePS Research Group**

SNePS es un sistema KRR integrado totalmente y desarrollado en SUNY Buffalo.  
<http://www.cs.buffalo.edu/pub/sneps/WWW/index.html>

# 3

## LÓGICA DIFUSA

### 3.1. INTRODUCCIÓN

Uno de los problemas con los que el ser humano se enfrenta a diario es que *“hada en la vida es cierto excepto la muerte”*. La solución de problemas de la vida real demanda la aceptación de la incertidumbre, para minimizar las dificultades que conlleva su solución.

Algunos problemas básicos subyacentes que se encuentran presentes en la actualidad son: por una parte existen conceptos sin una clara definición, muchos conceptos que manejan los seres humanos, a menudo, no tienen una definición clara, entre otras cosas se anotan las siguientes, ¿Qué representa ser una persona alta? ¿A partir de qué edad una persona deja de ser joven?. Por otra parte la lógica clásica o bivalente es demasiado restrictiva, si se analiza una afirmación esta siempre es verdadera o falsa en términos de esta lógica, pero hay algunas cosas que pueden no ser ni verdad ni falso, si se considera las siguientes frases: “Yo leeré la trilogía del Señor de los Anillos”, ¿En qué medida es cierto? depende de quien lo diga y “Él es bueno en Matemáticas”, ¿Es bueno, muy bueno o un poco mejor que regular?.

Muchas veces es necesario conocer cuando utilizar la tecnología difusa, esta se utiliza generalmente en:

- a) en procesos complejos, si no existe un modelo de solución sencillo,
- b) en procesos no lineales,
- c) cuando haya que introducir la experiencia de un operador “experto” que se base en conceptos imprecisos obtenidos de su experiencia,
- d) cuando ciertas partes del sistema a controlar son desconocidas y no pueden medirse de forma fiable<sup>22</sup>,
- e) cuando el ajuste de una variable puede producir el desajuste de otras,
- f) en general, cuando se quieran representar y operar con conceptos que tengan imprecisión o incertidumbre<sup>23</sup>.

### 3.2. DEFINICION

La palabra *fuzzy* viene del ingles *fuzz*<sup>24</sup> y se traduce como difuso o borroso, para fines de la presente publicación se utiliza la acepción difuso. En la actualidad la lógica difusa es un campo de investigación muy importante, tanto por sus implicaciones matemáticas o teóricas como por sus aplicaciones prácticas.

La lógica difusa o lógica posibilística fue creada por Lotfi Zadeh en 1965, quien extendió la clásica lógica booleana a los números reales. En el álgebra booleana 1 representa verdadero y 0 falso. Lo mismo ocurre en la lógica difusa, pero en adición, todas las fracciones entre cero y uno

---

<sup>22</sup> Es decir contiene aún posibles errores.

<sup>23</sup> Como ocurre cuando se utiliza base de datos difusas.

<sup>24</sup> La traducción de diccionario es: vello, pelusa, tamo.

son empleadas para indicar verdades parciales, así:  $P(\text{alto}(X)) = 0.75$ , establece que la proposición de que “X sea alto” es en algún sentido verdadero en una proporción de tres cuartos.

Para combinar valores de verdad no enteros, la lógica difusa define el equivalente de los operadores AND, OR y NOT como:

$P1 \text{ AND } P2 = \text{MIN}(P1, P2)$  Smaller

$P1 \text{ OR } P2 = \text{MAX}(P1, P2)$  Greater

$\text{NOT } P1 = 1 - P1$  Inverse

Así las piezas de evidencia pueden ser combinadas de una manera rigurosa y consistente.

### 3.3. TEORIA DE CONJUNTOS DIFUSOS

Considere un árbol. Ahora considere un arbusto. ¿En que etapa de su desarrollo un arbusto llega a convertirse en árbol?. ¿Esta etapa se realiza con las dimensiones físicas de la planta o es la esencia de hacer un árbol algo más espiritual que el tamaño?

Por supuesto, no existe un instante en el cual la planta pase de ser un arbusto a un árbol, y muchas personas pueden afirmar que la solución al problema es completamente obvia; que existe alguna etapa cuando la planta es en algún grado un arbusto y en algún grado un árbol.

Es sobre esta idea simple en la cual está basada el cuerpo completo de la teoría difusa. Un conjunto difuso es cualquier conjunto al cual los elementos pueden llegar a pertenecer a los extremos<sup>25</sup> de “miembro” o “no-miembro”. Un elemento puede ser un miembro de grado 0.5 por ejemplo, lo cual significa que es solamente un miembro parcial del conjunto. Un ejemplo de esto puede ser considerar el grado de pertenencia de un arbusto en el conjunto “árbol” que fue mencionado en el párrafo precedente. Un arbusto no es completamente un árbol, pero sin embargo tiene muchas cosas en común con un árbol, así puede ser razonable clasificar un arbusto como un árbol de grado 0.5.

#### 3.3.1. Conjuntos Clásicos<sup>26</sup>

Para ingresar en la arena de los conjuntos difusos propiamente dicho, es importante mencionar que los conjuntos clásicos surgen de forma natural, por la necesidad del ser humano de clasificar objetos y conceptos.

Así un conjunto de frutas puede ser representado como:  $F = \{\text{manzana, uva, durazno, chirimoya}\}$ . En el conjunto F se puede observar la función de pertenencia absoluta simbolizada por:  $\mu_F$ . Así manzana  $\mu_F$  y lechuga  $\mu_F$ .

La función de pertenencia es absoluta y se define en los siguientes términos: sea el conjunto  $A(x)$  con elementos  $x \in X$ . Se dice que X es el universo del discurso y la restricción de la función asociada a la lógica bivalente es que el conjunto  $A: X \rightarrow \{0,1\}$ .

<sup>25</sup> Estos extremos son representados por los números 0 y 1 en la teoría clásica de los conjuntos.

<sup>26</sup> Conocidos como *crisp sets*.

$$A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases}$$

En este dominio se define el conjunto vacío como  $\emptyset$  ( $x \in \emptyset = 0$ ),  $x \in X$ ; el conjunto universo se define como  $U(x)=1$ ,  $x \in X$ .

### 3.3.2. Conjuntos Difusos

¿Cuál es la definición de alto?. Para hacerlo fácil, asuma que los objetos a clasificar son adultos humanos antes que rascacielos u hojas de afeitar. La respuesta simple a la pregunta formulada es que no existe una definición adecuada. Ningún diccionario dice que alto está definido como que significa mas de 165 cm para las mujeres y 180 cm para los hombres o algo parecido. Una respuesta más informativa es que la definición de la palabra alto aplicada a una persona es definida de manera difusa, y aunque se tiene una idea buena de lo que es y no es alto, aún pensando que todas las personas sujetas de comparación son completamente diferentes. Alguien con 2 metros de estatura es indudablemente alto, pero ¿qué acerca de sus vecinos cercanos en altura?

Para resolver este problema, es necesario definir algunos conjuntos difusos. Es importante entender que no existe una manera rígida para definir un conjunto difuso. Estos son designados con la visión de la meta a alcanzar. Si un conjunto es definido para utilizarlo en algún sistema de control, por ejemplo, y el sistema no presenta un buen rendimiento, entonces es probable que los conjuntos necesiten ser redefinidos. El conjunto “alto” es definido para ser capaz de comunicar dimensiones físicas, y lo que es más importante, comunicar dimensiones imprecisas. Cuando se describe a alguna persona no es necesario especificar su altura en milímetros, sin embargo comunicar una idea aproximada de su estatura es probablemente lo más usual.

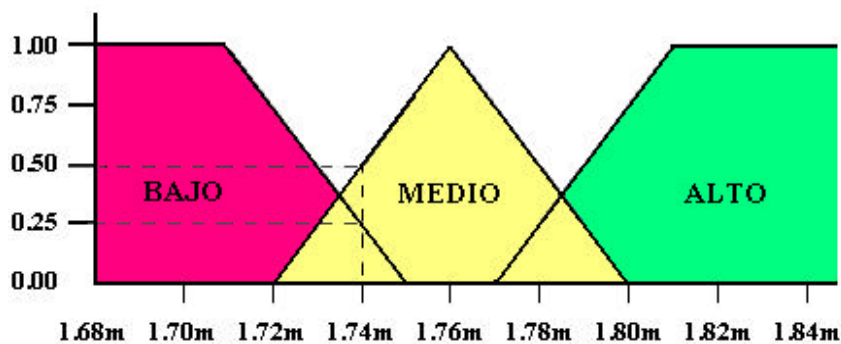


Fig.3.1. Conjuntos difusos

Se definen tres conjuntos que describen la estatura de los seres humanos: “bajo”, “mediano” y “alto”, ilustrados en la figura 3.1; donde la altura está indicada en el eje x del gráfico, y el grado de pertenencia del conjunto a la correspondiente altura es proporcionado por la coordenada del eje y. Así el grado al cual una altura de 1.74 m es “mediano” es de 0.50, y el grado en el cual es “bajo” 0.25 (ilustrado por las líneas punteadas).

Esto origina otro punto importante; a saber un elemento, en este caso un ejemplo de altura, puede ser miembro de mas de un conjunto aún si los dos conjuntos aparenten ser mutuamente

exclusivos en términos de la teoría clásica de los conjuntos. Tales conjuntos difusos pueden ser definidos a lo largo de algún eje o ejes numéricos, discretos o continuos, o pueden ser definidos utilizando ejemplos de los miembros del conjunto, en alguna forma deseada. Como se mencionó en los párrafos anteriores, la correctitud de la definición de un conjunto difuso puede ser evaluada mediante el rendimiento del sistema que hace uso del conjunto difuso.

Para la definición adecuada de los conjuntos difusos debe considerarse fundamentalmente el contexto en el que se define un conjunto difuso. En el ejemplo de la figura 3.1, no es lo mismo el concepto “Alto” aplicado a personas que a edificios.

Para formalizar la discusión de los grados de pertenencia, un conjunto difuso  $A$  está completamente definido sobre algún universo del discurso<sup>27</sup>  $\mathcal{U}$ , mediante la siguiente función de pertenencia:  $\mu_A : \mathcal{U} \rightarrow [0,1]$

El valor  $\mu_A(x)$  para  $x \in \mathcal{U}$  representa el grado de pertenencia de  $x$  en  $A$ , o el grado de compatibilidad de  $x$  con el concepto  $A$

Como ejemplo se desarrolla la función de pertenencia del subconjunto difuso Bajo al conjunto  $A$ .

$$\mu_{Bajo}(h) = \begin{cases} 1 & h \leq 1.72 \\ (h-1.72)/(1.76-1.72) & \text{si } h \in (1.72, 1.76) \\ (1.80-h)/(1.80-1.76) & \text{si } h \in (1.76, 1.80) \\ 0 & h \geq 1.80 \end{cases}$$

Donde  $h$  representa la altura.

### 3.3.3. Función de Pertenencia

Un conjunto difuso puede representarse también gráficamente como una función, especialmente cuando el universo de discurso  $\mathcal{U}$  es continuo o no discreto. Las abscisas ubicadas en el eje  $X$  representan el universo del discurso  $\mathcal{U}$ ; las ordenadas ubicadas en el eje  $Y$  corresponden a los grados de pertenencia en el intervalo  $[0,1]$ .

Ejemplo:

Concepto de temperatura “Alta”.

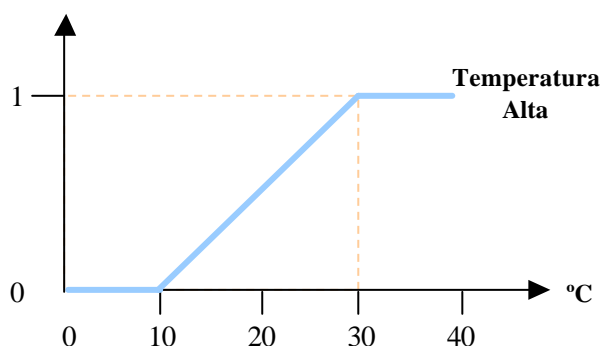


Fig. 3.2. Temperatura difusa “alta”

<sup>27</sup> En este caso, todos los valores posibles que pueden ser tomados por un atributo.

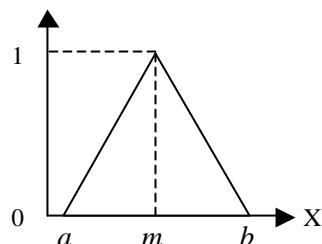


Cualquier función de pertenencia  $A$  es válida, su definición exacta depende del concepto a definir, del contexto al que se refiera y de la aplicación. En general, es preferible usar funciones simples, debido a que simplifican muchos cálculos y no pierden exactitud, debido a que precisamente se está definiendo un concepto difuso.

Entre las funciones de pertenencia típicas pueden distinguirse las siguientes:

- a) **Función Triangular.** Se encuentra definida por sus límites: inferior  $a$  y superior  $b$ , y el valor modal  $m$ , tal que  $a < m < b$ .

$$A(x) = \begin{cases} 0 & \text{si } x \leq a \\ (x-a)/(m-a) & \text{si } x \in (a, m) \\ (b-x)/(b-m) & \text{si } x \in (m, b) \\ 0 & \text{si } x \geq b \end{cases}$$

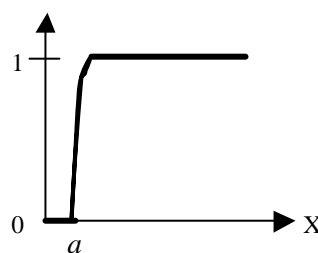


También puede representarse así:  $A(x; a, m, b) = \max\{\min\{(x-a)/(m-a), (b-x)/(b-m)\}, 0\}$

- b) **Función G (gamma).** Se encuentra definida por su límite inferior  $a$  y un valor constante  $k > 0$ . Esta función se caracteriza por un rápido crecimiento a partir de  $a$ . Cuanto mayor es el valor de  $k$ , el crecimiento de la función es más rápido aún. Nunca toman el valor 1, aunque tienen una asíntota horizontal en 1.

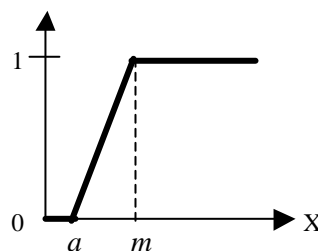
$$A(x) = \begin{cases} 0 & \text{si } x \leq a \\ 1 - e^{-k(x-a)} & \text{si } x > a \end{cases}$$

$$A(x) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{k(x-a)^2}{1 + k(x-a)^2} & \text{si } x > a \end{cases}$$



Esta función se aproxima linealmente con lo siguiente:

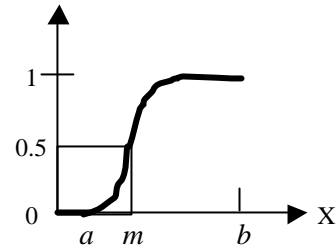
$$A(x) = \begin{cases} 0 & \text{si } x \leq a \\ (x-a)/(m-a) & \text{si } x \in (a, m) \\ 1 & \text{si } x \geq m \end{cases}$$



La función opuesta se llama Función L.

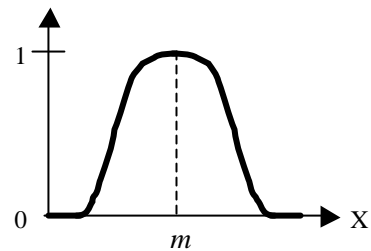
- c) **Función S.** Se encuentra definida por sus límites inferior  $a$  y superior  $b$ , además del valor  $m$ , denominado punto de inflexión, tal que  $a < m < b$ . Un valor típico de esta función es el promedio  $m = (a+b)/2$ . El crecimiento es más lento cuanto mayor sea la distancia  $(a-b)$ .

$$A(x) = \begin{cases} 0 & \text{si } x < a \\ 2\{(x-a)/(b-a)\}^2 & \text{si } x \in (a, m] \\ 1-2\{(x-b)/(b-a)\}^2 & \text{si } x \in (m, b) \\ 1 & \text{si } x \geq b \end{cases}$$



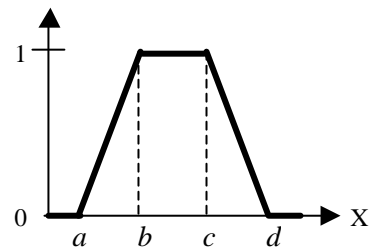
- d) **Función Gaussiana.** Definida por su valor medio  $m$  y el valor constante  $k > 0$ . Esta función hace referencia a la típica campana de Gauss. Se caracteriza porque cuanto mayor es  $k$ , más estrecha es la forma de la campana.

$$A(x) = e^{-k(x-m)^2}$$



- e) **Función Trapezoidal.** Se encuentra definida por sus límites inferior  $a$  y superior  $d$ , y los límites de su soporte,  $b$  y  $c$ , inferior y superior respectivamente.

$$A(x) = \begin{cases} 0 & \text{si } x < a \\ (x-a)/(b-a) & \text{si } x \in (a, b] \\ 1 & \text{si } x \in (b, c) \\ (d-x)/(d-c) & \text{si } x \in (c, d) \\ 0 & \text{si } x > d \end{cases}$$



### 3.4. CARACTERISTICAS ESENCIALES

La lógica difusa es la lógica fundamental de los modos de razonamiento que son aproximados antes que exactos. Alguna de las características esenciales de la lógica difusa se relacionan con lo siguiente:

- En la lógica difusa el razonamiento exacto es considerado como un caso limite del razonamiento aproximado.
- En la lógica difusa todo es materia de grados.
- Cualquier sistema lógico puede ser llevado a términos difusos.
- En la lógica difusa el conocimiento esta interpretado como una colección de restricciones elásticas.
- La inferencia es considerada como un proceso de propagación de restricciones elásticas.

### 3.5. DIFERENCIAS RESPECTO A SISTEMAS TRADICIONALES

La lógica difusa difiere de los sistemas tradicionales de lógica tanto en espíritu como en detalle. Algunas de las principales diferencias son las siguientes:

### **3.5.1. Valores de Verdad**

En los sistemas de lógica bivalente, los valores de verdad solamente pueden tener dos valores: verdadero o falso. En los sistemas multivaluados, los valores de verdad de una proposición pueden ser elementos de: un conjunto finito, un intervalo tal como  $[0,1]$  o un álgebra booleana.

En la lógica difusa los valores de verdad de una proposición pueden ser subconjuntos difusos de cualquier conjunto parcialmente ordenado pero usualmente este es un subconjunto difuso del intervalo  $[0,1]$  o simplemente un punto en ese intervalo. Los denominados valores de verdad lingüísticos expresados como: verdadero, demasiado verdadero, completamente verdadero, etc. son interpretados como etiquetas de subconjuntos difusos del intervalo unitario.

### **3.5.2. Predicados**

En los sistemas bivalentes, los predicados son precisos por decir: mortal, aun, mas grande que. En la lógica difusa los predicados son difusos, por decir: alto, malo, gordo, veloz, mucho mas grande que. Se puede decir que la mayoría de los predicados en un lenguaje natural son difusos antes que precisos.

### **3.5.3. Modificadores de Predicados**

En los sistemas clásicos, el único modificador de predicados ampliamente utilizado es la negación. En la lógica difusa existe una variedad de modificadores de predicados tales como: muy, mas o menos, completamente antes, extremadamente. Tales modificadores de predicados juegan un rol esencial en la generación de los valores de una variable lingüística tales como: muy joven, no muy joven, mas o menos joven, etc.

### **3.5.4. Cuantificadores**

En los sistemas lógicos clásicos existen solamente dos Cuantificadores: universal y existencial. La lógica difusa admite, en adición, una variedad amplia de Cuantificadores tales como: pocos, varios, usualmente, mas que, casi siempre, frecuentemente, etc. En la lógica difusa un cuantificador difuso es interpretado como un numero difuso o una proporción difusa.

### **3.5.5. Probabilidades**

En los sistemas de lógica clásica, la probabilidad es numérica o de valor en intervalos. En la lógica difusa se tiene la opción adicional de emplear lingüísticos, o de manera general, las probabilidades difusas ejemplificadas por probable, improbable, muy probable, alrededor de 0.7, grandemente probable. Tales probabilidades pueden ser interpretadas como números difusos que pueden ser manipulados mediante el uso de la aritmética difusa.

### **3.5.6. Posibilidades**

El concepto de posibilidad en la lógica difusa es graduada, antes que bivalente. Además, como en el caso de las probabilidades, las posibilidades pueden ser tratadas como variables lingüísticas

con valores tales como: posible, completamente posible, casi imposible, etc. Tales valores pueden ser interpretados como etiquetas de subconjuntos difusos de la línea real.

### 3.6. TEORIA DE LA POSIBILIDAD

La teoría de la posibilidad sirve para tratar lo difuso de manera distinta a la aleatoriedad. Esta teoría esta basada en los primeros trabajos de Zadeh sobre conjuntos difusos.

La teoría de los “conjuntos difusos” proporciona un medio para tratar con lo difuso de términos tales como “grande”. Si S es un conjunto y e es un miembro de S, entonces se puede definir un conjunto difuso F de S introduciendo una función de pertenencia como miembro @F tal que  $@F(e)=d$ , donde d es el grado hasta el cual e es miembro de F.

Por ejemplo, si S es el “conjunto de todos los animales” y F es el “conjunto de animales grandes”, entonces se puede definir @F como sigue:

|                |   |        |
|----------------|---|--------|
| @F(elefante)   | = | 1.0    |
| @F(hipopótamo) | = | 1.0    |
| @F(tigre)      | = | 0.8    |
| @F(hombre)     | = | 0.6    |
| @F(perro)      | = | 0.1    |
| @F(ratón)      | = | 0.0001 |

Una declaración difusa tal como “animal grande” se interpreta que tiene una detonación imprecisa caracterizada por un conjunto difuso.

Una “variable difusa” X es aquella que puede tomar valores en S y valores “asignados” por declaraciones tales como: “X es un F”.

El valor de X es una distribución de posibilidad tal que la posibilidad de que  $X=e$  es @F(e). Por ejemplo, la declaración: “X es un animal grande”, significa que la posibilidad de que “X=elefante” es 1.0, la posibilidad de que “X=raton” es 0.0001, etc.

La teoría de la posibilidad proporciona reglas para calcular las posibilidades de expresiones que implican variables difusas. Tales reglas incluyen:

|                        |   |                           |
|------------------------|---|---------------------------|
| pos (X=a OR Y=b)       | = | max {pos (X=a), pos(Y=b)} |
| pos (X=a AND Y=b)      | = | min {pos (X=a), pos(Y=b)} |
| pos (X <sup>1</sup> a) | = | 1 - pos(X=a)              |

La lógica difusa de Zadeh puede ayudar a detectar la redundancia en un conjunto de declaraciones difusas. Por ejemplo considérese el siguiente conjunto de declaraciones:

- a) Las personas esbeltas tienden a correr rápido
- b) Los jóvenes mayores de quince años tienden a correr rápido
- c) Un gran numero de corredores nacieron en 1969
- d) Es más probable que un muchacho sea esbelto si tiene mas de dieciséis años.

Suponiendo que Mary esta en el colegio, que tiene diecisiete años y que nació en 1969, un razonador ingenuo procedería de la siguiente manera:

- e) De b) se concluye que Mary podría correr rápido.
- f) De d) y los datos dados se concluye que Mary podría ser esbelta
- g) De f), a) y e) se concluye que es probable que Mary corra rápido
- h) De los datos y c) juntamente con g), se concluye que Mary debe ser una corredora rápida.

Sin embargo, esta conclusión es algo precipitada dadas las declaraciones anteriores. Un razonador más analítico puede ver que esto es demasiado rotundo, puesto que puede detectar la redundancia que este presente.

El problema que podría tener un razonador ingenuo se debe al hecho de que la redundancia está implícita. La confusión aparece porque las declaraciones anteriores se expresan como equivalencias o disposiciones. Es decir, no se ha indicado con claridad cual es la dirección de la implicación causal. Por ejemplo, no esta claro si los jóvenes de diecisiete años corren rápido porque son esbeltos o porque nacieron en 1969. Zadeh en el año 1983 describió como se puede utilizar la lógica difusa para superar tales problemas.

### 3.7. SEMANTICA CON RESULTADOS DE ENSAYO

La semántica con resultados de ensayo se utiliza para asignar un significado a una proposición de la lógica difusa. En la semántica con resultados de ensayos se considera que una proposición es una colección de “restricciones elásticas”.

Por ejemplo, la proposición “*Mayte es castaña*” representa una restricción elástica del color de cabello de *Mayte*, y la proposición “*la mayoría de los gordos no son muy ágiles*” representa una restricción elástica del numero de *gordos ágiles*. En la semántica con resultados de ensayo, el significado de una proposición viene dado por el procedimiento que se utiliza para calcular los resultados de los ensayos de esa propiedad.

En general, este procedimiento implica:

- a) Identificar las variables  $X_1, \dots, X_n$ , cuyos valores estén limitados por la proposición. En el ejemplo de antes  $X_1$ =color del cabello de *Mayte*.
- b) Identificar las restricciones  $C_1, \dots, C_m$  inducidas por la proposición.
- c) Caracterizar cada restricción  $C_i$  describiendo un procedimiento de ensayo que asocie a  $C_i$  un resultado de ensayo  $t_i$  que represente el grado hasta el cual se satisface  $C_i$ .
- d) Agregar los resultados parciales de los ensayos  $t_1, \dots, t_m$  a un número más pequeño de resultados de ensayo  $t_1, \dots, t_k$  que representen un resultado global de un vector de ensayo  $t$ . En la mayoría de los casos,  $t=1$ . El procedimiento de ensayo en c) hace uso de una colección de relaciones difusas que constituye una base de datos explicativa<sup>28</sup>.

El significado de las relaciones en una base de datos explicativa es conocido por la persona que quiera saber el significado de la proposición. Indirectamente, los procedimientos de ensayo y agregación utilizados en semántica de resultados de ensayo describen un proceso mediante el cual los significados de las proposiciones están compuestos de significados de las relaciones constituyentes en la base de datos explicativa.

---

<sup>28</sup> De la palabra inglesa Explanatory Database (ED).

Como ejemplo, suponga que se quiere determinar el significado de la proposición: “la nieve es generalmente blanca”.

Una base de datos explicativa apropiada incluiría una relación BLANCA y una relación GENERALMENTE, tal que:

#### BLANCA

| Muestra de nieve | Grado hasta el cual la muestra es “blanca” |
|------------------|--|
| $S_1$            | 0.4  |
| $S_2$            | 0.5  |
| $S_3$            | 0.6  |
| $S_4$            | 0.7  |
| $S_5$            | 0.8  |

#### GENERALMENTE

| Proporción | Grado hasta el cual la proporción representa “generalmente” |
|------------|---|
| 0.3        | 0.45  |
| 0.4        | 0.55  |
| 0.5        | 0.65  |
| 0.6        | 0.75  |
| 0.7        | 0.85  |

Ahora suponga que  $S_1, \dots, S_m$  representan muestras de nieve y que  $t_i$  ( $1 \leq i \leq m$ ) representa el grado hasta el cual el color de  $S_i$  es blanco. Así pues,  $t_i$  es el resultado del ensayo para la restricción del color de  $S_i$  inducido por “blanco”.

Utilizando esta notación, los pasos en el procedimiento de ensayo son:

- Encontrar la proporción de muestras cuyo color sea blanco:  $a = (t_1 + t_2 + \dots + t_m)/m$ . Para el caso del ejemplo:  $a = 0.6$ , a partir de la tabla BLANCA.
- Calcular el grado hasta el cual  $a$  satisface la restricción inducida por “generalmente”  $t =$  grado hasta el cual la proporción representa generalmente, donde la proporción  $= a$ . En el ejemplo utilizando la tabla GENERALMENTE se tiene que la proporción de 0.6 tiene un valor de 0.75. Es decir 75% considera que la nieve generalmente es blanca.

Como se observa, simplemente se utiliza  $a$  para buscar un valor en la relación “generalmente”. El significado de “la nieve es generalmente blanca” está representado por el procedimiento de ensayo utilizado para calcular el valor de  $t$ .

### 3.8. APLICACIONES

Es necesario proporcionar un entendimiento correcto de lo que puede realizar un sistema difuso en un número de áreas determinado.

#### 3.8.1. Área de Control

El uso de los sistemas difusos en el área del control fue una de las primeras aplicaciones prácticas del paradigma.

Palm demostró que un controlador difuso simple de afinado manual podía ejecutar de manera correcta un algoritmo de control sobre un manipulador de robots, con una ventaja principal consistente en la estabilidad del control difuso bajo condiciones inusuales, por ejemplo un manipulador de diversos tipos de metales.

Murukami, Takemoto y Fujimura, realizaron un trabajo similar para un controlador difuso, el cual describe el control de un arco de soldadura sobre una línea de soldadura. Nuevamente una de las principales características del controlador es su robustez en un medio ambiente ruidoso causado por los sensores que operan de manera cerrada en un soplete de soldadura.

Horton & Jones en 1994, aplicaron un sistema difuso al problema de la pista de las multi tarjetas, tradicionalmente del dominio de los sistemas analíticos tales como el filtro de Kalman. Ellos demostraron que un rendimiento comparable al filtro Kalman puede obtenerse con costos computacionales reducidos. Kosko, en 1992, también examina el problema de las pistas, comparando un sistema difuso con un filtro Kalman. Kosko demostró que aunque ambos sistemas se comportan de manera similar bajo condiciones ideales, cuando los parámetros del sistema son perturbados, mediante la adición de reglas fraudulentas a la base de reglas difusas y mediante el cambio de la covarianza estimada del estado del proceso del filtro de Kalman, que el rendimiento del sistema difuso se degrada de manera más adecuada que la del filtro de Kalman.

Fodor & Driankov examinaron el rendimiento de un número de controladores bajo la transgresión de lo que ellos denominaron suposiciones ontológicas, o el cuerpo formal de conocimiento acerca del medio ambiente de control utilizado para designar un controlador. Fodor & Driankov demostraron que es posible diseñar un sistema difuso que mantiene estabilidad bajo una clase general de transgresiones a las suposiciones ontológicas, bajo las cuales los controles podrían fallar.

De manera comercial, los trenes subterráneos en la ciudad japonesa de Sendai utilizan controladores desarrollados con lógica difusa para asegurar un frenado suave y económico en las proximidades a las estaciones, y para detener los trenes en el lugar correcto de tal forma que las puertas queden a la altura de las líneas marcadas en las plataformas.

### **3.8.2. Área de Inteligencia Artificial**

El poder de la lógica difusa como un medio para representar la incertidumbre, y su simplicidad desde el punto de vista computacional comparado con la teoría de la probabilidad, ha conducido a un conjunto amplio de aplicaciones en la toma de decisiones y los sistemas inteligentes, frecuentemente en aplicaciones donde no existen soluciones convencionales.

Un número de aplicaciones financieras fueron documentadas para esta área, por ejemplo Rommelfanger describe un sistema para el asesoramiento del valor del crédito en pequeñas firmas. Sotirov, Mincoff y Krasteva, propusieron el uso de los sistemas difusos para ayudar en la asignación de precios a los nuevos productos de venta en un mercado.

También en el campo de la IA, los sistemas difusos fueron utilizados para detectar fallas en las plantas generadoras de energía eléctrica, en las fabricas de cemento para el control de la producción y, en las luces de control de tráfico para optimizar el flujo del trafico vehicular.

### Ejercicios #3

1. En el contexto desarrollado en el presente capítulo, ¿Qué se entendería por razonamiento difuso?
2. Empleando bibliografía de lógica difusa explique formalmente el significado de una variable lingüística.
3. Utilizando como universo del discurso los grados centígrados defina cinco conjuntos difusos acerca de la temperatura en la Ciudad de La Paz.
4. Defina cinco conjuntos difusos para la temperatura de la Ciudad de Cobija.
5. Utilizando teoría de posibilidades explique la “dedicación” de su persona a las materias de estudio semestrales.
6. Sea S es el “conjunto de todos los relojes” y F es el “conjunto de relojes pulsera”, definir @F e interpretar reloj grande, reloj mediano y reloj pequeño.
7. Considere el conjunto de las siguientes proposiciones
  - a) Las personas esbeltas tienden a correr rápido
  - b) Los jóvenes mayores de 15 años tienden a correr rápido
  - c) Un gran número de corredores nació en el año 1980
  - d) Es más probable que una muchacha sea esbelta si tiene mas de 16 añosUtilizando la lógica difusa y suponiendo que, Vilema está en el colegio, que tiene 17 años y que nació en 1980, ¿cual es la conclusión del razonamiento asociado a las proposiciones?.
8. Determinar el significado de la proposición: “Las manifestaciones normalmente perjudican a los trabajadores”.
9. Utilizando semántica con resultados de ensayo, determinar el significado de la proposición: “el color de cabellos de las personas de manera frecuente es negro”.
10. Determinar el significado de la proposición: “La lluvia generalmente es una bendición para el agricultor y normalmente una maldición para el ciudadano”.

### Bibliografía

- Bezdek, J. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York, 1981.
- Dychkhoff, H. & W. Pedrycz. Generalized Means as a Model of Compensative Connectives. *Fuzzy Sets and Systems 14*, pp. 143-154, 1984.
- Kosko, B. *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. Englewood Cliffs, NJ: Prentice Hall, 1992.
- Kruse, R., J. Gebhardt, F. Klawonn, *Foundations of Fuzzy Systems*. John Wiley & Sons, 1994. ISBN 0-471-94243X.
- McNeill, F.M., E. Thro, *Fuzzy Logic: A Practical Approach*. AP Professional, 1994. ISBN 0-12-485965-8.
- Menger, K. Statistical Metric Spaces. *Proc. of the National Academy of Sciences 37*, pp. 535-537 (USA), 1942.
- Mohammed, J, N. Vadiiee, T.J. Ross, Eds. “*Fuzzy Logic and Control. Software and Hardware Applications*”. Eaglewood Cliffs, NJ:PTR. Prentice Hall, 1993.
- Pedrycz, W., F. Gomide, *An introduction to Fuzzy Sets: Analysis and Design*. A Bradford Book. The MIT Press, Massachusetts, 1998. ISBN 0-262-16171-0.



Saaty, T.L. *The Analytic Hierarchy Processes*. McGraw Hill, New York, 1980.

Schwizer, B. & A. Sklar. *Probabilistic Metric Spaces*. Amsterdam: North Holland, 1983.

Sur A&C, Omron Electronics, S.A., *Lógica Difusa para Principiantes*. Ed. I. Hernández, 1997. ISBN 84-920326-3-4.

Yager, R. On Ordered Weithted Averaging Aggregation Operations in Multicriteria Decision Making. *IEEE Transactions on Systems, Man and Cybernetics* 18, pp. 183-190, 1988.

Zadeh, L.A. Fuzzy Sets as a Basis for a Theory of Possibility. *Fuzzy Sets and Systems* 1, pp. 3-28, 1978.

Zadeh, L.A. Fuzzy Sets. *Information and Control*, 8, pp. 338-353, 1965.

Zadeh, L.A. The Concept of a Linguistic Variable and its Application to Approximate Reasoning. *Information Sciences* 8, pp. 199-249 (part I), 8, pp. 301-357 (part II), 9, pp. 43-80 (part III), 1975.

Zimmermann, H. & P. Zysno. Latent connectives in human decision making. *Fuzzy Sets and Systems* 4, pp. 47-51, 1980.

Zimmermann, H. *Fuzzy Set Theory and its Applications*. 2d ed. Dordrecht, the Netherlands: Kluwer Academic Publishers, 1993.

## Referencias Electrónicas

Fuzzy Logic and Neurofuzzy Resources at ISIS Research Group  
<http://www-isis.ecs.soton.ac.uk/research/nfinfo/fuzzy.html>

Fuzzy Logic Sources of Information  
<http://www.cms.dmu.ac.uk/People/rj/fuzzy.html>

A tutorial on Fuzzy Systems by James F. Brule.  
<http://www.quadralay.com/www/Fuzzy/tutorial.html>

International Fuzzy Systems Association (IFSA)  
<http://www.abo.fi/~rfuller/ifsa.html>

European Network in Uncertainty Techniques Developments for Use in Information Technology. (ERUDIT)  
<http://www.mitgmbh.de/erudit/erudit.html>

North American Fuzzy Information Processing Society (NAFIPS)  
<http://seraphim.csee.usf.edu/nafigs.html>

Int. J. Uncertainty, Fuzziness and Knowledge-Based Systems  
<http://www.wspc.co.uk/wspc/Journals/ijufks/ijufks.html>

IEEE Transactions on Fuzzy Systems.  
[http://www.ieee.org/pub\\_preview/fuzz\\_toc.html](http://www.ieee.org/pub_preview/fuzz_toc.html)

Elsevier Science on Fuzzy Sets and Systems  
<http://www.elsevier.nl/catalogue/SA5/515/08410/08417/Menu.html>

World Scientific on Fuzzy Sets and Systems  
[http://www.wspc.co.uk/wspc/Books/compai\\_hilites.html](http://www.wspc.co.uk/wspc/Books/compai_hilites.html)

The Center for Fuzzy Logic and Intelligent Systems Research

<http://www.cs.tamu.edu/research/CFL/>

Fuzzy CLIPS

<http://ai.iit.nrc.ca/fuzzy/fuzzy.html>

Approximate Reasoning and Artificial Intelligence Group - University of Granada, Spain

<http://decsai.ugr.es/difuso/fuzzy.html>

Research Group on Fuzzy Systems and Soft Computing at Technical University of  
Braunschweig,

<http://sol.ibr.cs.tu-bs.de/ibr/fuzzy/fuzzysys.html>

Fuzzy Logic at the "Politecnico di Milano AI and Robotics Project"

[http://www.elet.polimi.it/section/compeng/air/fuzzy\\_intro.html](http://www.elet.polimi.it/section/compeng/air/fuzzy_intro.html)

Conferences on Fuzzy Logic from the WWW Virtual Library on Conferences.

[http://www.iao.fhg.de/Library/conferences/index/Fuzzy\\_Logic.html](http://www.iao.fhg.de/Library/conferences/index/Fuzzy_Logic.html)

# 4

## REDES NEURONALES ARTIFICIALES

### 4.1. INTRODUCCION

Existen tareas para las cuales todavía no existen algoritmos, o para las cuales es virtualmente imposible escribir una serie de pasos lógicos o aritméticos que proporcionen la solución adecuada a una tarea determinada. Estas tareas tienen características importantes en común:

- a) los humanos saben como realizarlas;
- b) se puede generar grandes cantidades de ejemplos de ellas;
- c) cada tarea requiere realizar una asociación entre objetos de dos conjuntos.

Los ejemplos de este tipo de tareas son las tareas cognoscitivas como reconocer un rostro familiar, hablar, comprender el lenguaje y recuperar contextualmente información apropiada desde la memoria. Estas tareas están mas allá del alcance de las computadoras programadas de manera convencional, así como de los sistemas expertos tradicionales.

Como alternativa para el procesamiento automatizado de la información, se establece la neurocomputación, de manera acelerada, como una disciplina. Formalmente la neurocomputación es la disciplina relacionada con los sistemas de procesamiento de la información adaptativos no programados, que desarrollan asociaciones o transformaciones entre objetos. En lugar de un procedimiento paso a paso para ejecutar la transformación de la información, la red neuronal genera sus propias reglas internas que gobiernan la asociación y refina estas reglas comparando sus resultados con los ejemplos.

La neurocomputación es fundamentalmente un paradigma nuevo y diferente para el procesamiento de la información, la alternativa más radical de la programación algorítmica. Pero la neurocomputación no puede aún reemplazar a la programación algorítmica, por el momento la neurocomputación se complementa con la programación algorítmica.

El renovado interés en las redes neuronales está motivado por los avances en la tecnología así como por una comprensión más profunda de como trabaja el cerebro. Una motivación particular es el deseo de construir una nueva generación de computadoras para resolver tareas como las mencionadas en los párrafos anteriores, para las cuales las arquitecturas actuales resultan insuficientes. Otra motivación es el deseo de desarrollar modelos cognitivos que puedan servir como fundamento para la inteligencia artificial.

Aunque es conocido que el cerebro no es tan bueno como una computadora convencional para ejecutar operaciones aritméticas, existen varias funciones del cerebro que no se pueden duplicar en una computadora. Algunas de tales funciones son la asociación, categorización, generalización, clasificación y extracción de rasgos. Estos aspectos están relacionados de manera estrecha a la propiedad asociativa y de auto organización del cerebro.

La propiedad asociativa significa la capacidad de recuperar un cuerpo de información complejo utilizando una pequeña parte del mismo como una llave para un proceso de búsqueda. El cerebro hace esto extraordinariamente bien. La autoorganización significa la habilidad para adquirir conocimiento a través de un proceso de aprendizaje por tanteo que comprende organizarse y reorganizarse en respuesta a estímulos externos.

Por otra parte los modelos y algoritmos estándar de la IA tienen las siguientes características:

- a) El conocimiento se representa explícitamente utilizando reglas, redes semánticas, modelos probabilísticos, etc.,
- b) Se imita el proceso humano de razonamiento lógico para resolver los problemas, centrando la atención en las causas que intervienen en el problema y en sus relaciones (encadenamiento de reglas, inferencia probabilística), y
- c) Se procesa la información secuencialmente.

Con el rápido desarrollo de la IA apareció un gran número de problemas complejos donde no se disponía de una representación explícita del conocimiento y de un procedimiento de razonamiento lógico para resolverlo. Las posibles causas de este fallo son los procedimientos algorítmicos y la estructura computacional empleados son incorrectos. Una posible solución es el uso de estructuras computacionales paralelas inspiradas en redes neuronales biológicas.

Se dice que para el manejo del paradigma de las redes neuronales es necesario salir de los esquemas cerrados que plantea la computación actual con el modelo propuesto por Von Neumann. En este sentido una computadora biológica que tenga las características descritas en la tabla 4.1 constituiría una buena opción para el manejo adecuado del paradigma conexionista y otros paradigmas inspirados en la biología.

|                                 | <b>COMPUTADORA VON NEUMANN</b> | <b>COMPUTADORA BIOLÓGICA</b> |
|---------------------------------|--------------------------------|------------------------------|
| <b>Procesador</b>               | Compleja                       | Simple                       |
|                                 | Alta velocidad                 | Baja velocidad               |
|                                 | Uno o unos cuantos             | Gran número                  |
| <b>Memoria</b>                  | Separada del procesador        | Integrada en el procesador   |
|                                 | Localizada                     | Distribuida                  |
| <b>Computación</b>              | Contenido no direccionable     | Contenido direccionable      |
|                                 | Centralizada                   | Distribuida                  |
|                                 | Secuencial                     | Paralela                     |
| <b>Confiabilidad</b>            | Programa almacenado            | Auto aprendizaje             |
|                                 | Muy vulnerable                 | Robusta                      |
| <b>Experiencia</b>              | Manejo numérico y simbólico    | Problemas perceptuales       |
|                                 | Bien definido                  | Pobrementemente definido     |
| <b>Medio ambiente operativo</b> | Bien restringido               | Irrestringido                |
|                                 |                                |                              |

**Tabla 4.1. Computadora Von Neumann vs. Computadora Biológica**

## 4.2. NEURONAS

El sistema nervioso y especialmente el cerebro constan de una cantidad increíble de células nerviosas denominadas neuronas, conexiones entre las neuronas denominadas sinapsis y soportan células denominadas células glia. Las neuronas son las células más importantes del cerebro, debido a que realizan todo el proceso, mientras que el rol de las células glia es soportar las neuronas mediante el control de las concentraciones iónicas al interior de las neuronas.

La habilidad del cerebro para almacenar información es distribuida sobre todas las neuronas y el proceso de recordar la información almacenada es un proceso colectivo entre todas las neuronas de la red neuronal.

#### 4.2.1. Modelo Biológico Básico

El modelo propuesto es bastante común en los libros de texto sobre neuro fisiología. Una neurona consta de tres partes básicas:

- a) **Dendritas**: Un árbol de fibras de entrada que lleva los potenciales de acción de las neuronas transmisoras en la neurona.
- b) **Soma**: El cuerpo principal de la célula con un núcleo. Es en este lugar donde los potenciales de acción son construidos antes que la neurona se active.
- c) **Axón**: Es una fibra de salida simple que bifurca a otras neuronas y transmite los potenciales de acción generados por la neurona.

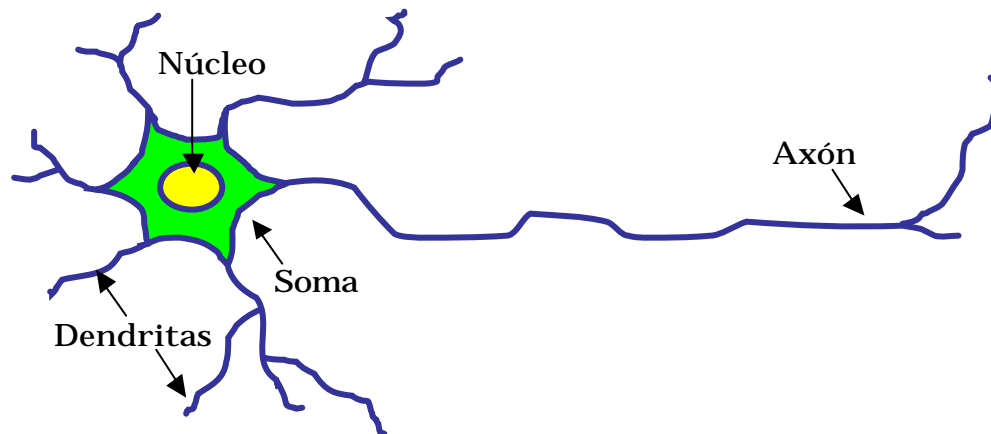


Fig. 4.1. Neurona biológica

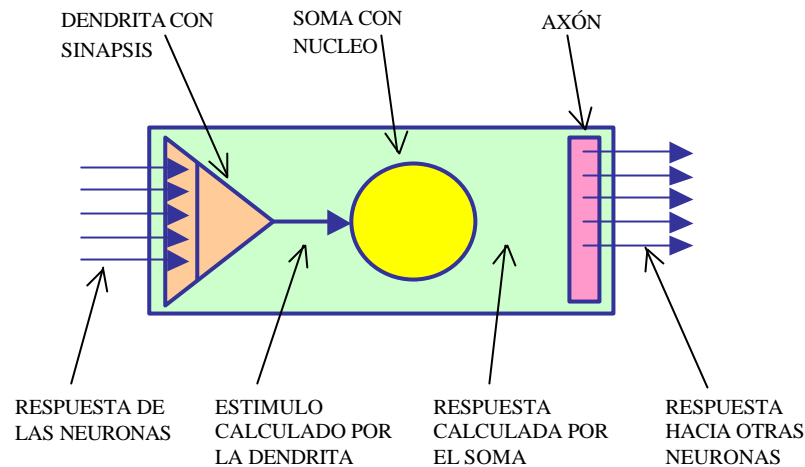
Las dendritas y los axones pueden ser vistos como cadenas de comunicación entre las neuronas, y no se conocen los detalles exactos acerca de cómo funcionan actualmente. Las partes más interesantes son el soma y la sinapsis los cuales constituyen los puntos de conexión entre los axones y las dendritas.

Las neuronas reciben señales (inputs) de otras neuronas vía conexiones sinápticas que pueden ser activadoras o inhibidoras. En función de las señales recibidas, una neurona envía a su vez una señal a otras neuronas por medio del axón.

Una neurona contiene un potencial interno continuo denominado potencial de membrana. Cuando éste excede un cierto valor umbral, la neurona puede transmitir todo su potencial por medio del axón. Se estima que el cerebro humano contiene más de cien mil millones ( $10^{11}$ ) de neuronas y que hay más de 1000 sinápsis a la entrada y a la salida de cada neurona.

#### 4.2.2. Modelo Idealizado

La neurona que se presenta es utilizada en muchas redes neuronales artificiales actuales. De manera similar al modelo de neurona funcional básico consta de tres partes básicas: Dendritas, Soma con núcleo y Axón.



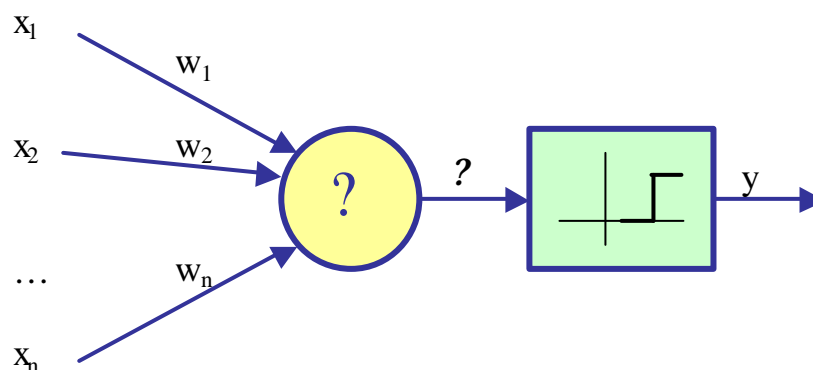
**Fig. 4.2. Modelo idealizado de una neurona**

La neurona tiene una sola dendrita. La dendrita acumula el estímulo enviado a la neurona desde las otras neuronas. El soma procesa los estímulos recibidos a través de su dendrita y decide la respuesta de la neurona. El axón toma cuidado de distribuir esta respuesta a las otras neuronas.

Los modelos de redes neuronales artificiales, o simplemente redes neuronales, se conocen por diversos nombres como modelos conexionistas o modelos de procesamiento distribuido paralelo. En lugar de ejecutar un programa secuencialmente como en una arquitectura Von Neumann, la red neuronal explora muchas hipótesis de manera simultánea utilizando redes masivamente paralelas compuestas de muchos elementos de procesamiento conectados por enlaces con pesos.

#### 4.2.3. Modelo Computacional

McCulloch & Pitts proponen una unidad binaria con umbral como un modelo computacional para una neurona. Un diagrama esquemático de la neurona de Mcculloch-Pitts se observa en la Fig. 4.3.



**Fig. 4.3. Modelo computacional de una neurona**

La neurona matemática calcula una suma de pesos de las  $n$  señales de entrada,  $x_j$ ,  $j=1, 2, n$ , y genera una salida de “1” si su suma esta sobre un cierto umbral  $\theta$ , y una salida de “0” en otros casos. Matemáticamente,

$$y = \theta(\sum w_j x_j - \theta), j=1..n$$

donde  $\theta(.)$  es una función de paso unitario, y  $w_j$  es el peso de la sinapsis asociado con la  $j$ -ésima entrada. Para la simplicidad de la notación, frecuentemente se considera el umbral  $\theta$  como otro peso  $w_0 = -\theta$ , la cual está ligada a la neurona con una entrada constante,  $x_0 = 1$ . Los pesos positivos corresponden a sinapsis activadoras, mientras que los pesos negativos corresponden a sinapsis inhibidoras.

Existe una analogía cruda<sup>29</sup> con una neurona biológica: las interconexiones modelan las dendritas y los axones, los pesos de las conexiones representan la sinapsis, y la función umbral aproxima la actividad en el soma.

La neurona de McCulloch-Pitts ha sido generalizada de muchas maneras. Una generalización obvia consiste en utilizar de funciones de activación distintas a la función umbral, por decir, una función lineal, una sigmoideal o una gaussiana. La función sigmoideal es la que se utiliza con mayor frecuencia en las RNAs. Es una función estrictamente incremental que exhibe propiedades asintóticas y uniformes. La función sigmoideal estándar es la función *logística* definida como

$$g(x) = 1/(1+\exp(-\theta x))$$

donde  $\theta$  es el parámetro de inclinación.



**Fig. 4.4. Funciones de activación**

#### 4.2.4. Modelos de Neuronas

El modelo de la neurona define el comportamiento de la misma al recibir una entrada para producir una respuesta, a esta respuesta se la conoce como nivel de activación. La activación de cada neurona se calcula en dos etapas. Primero se computa la entrada total a la neurona, la cual no es más que la suma pesada de todas las entradas a la unidad<sup>30</sup>, más un término que indica la predisposición de la neurona a reaccionar<sup>31</sup>.

$$S_i = \sum W_{ij} * X_j + \text{BIAS}_i$$

En la segunda etapa se calcula el nivel de activación utilizando una función que tiene como argumento la entrada total a la neurona. Entre las funciones más utilizadas con este propósito están las siguientes:

<sup>29</sup> Puede decirse una analogía tosca, rudimentaria.

<sup>30</sup> Se refiere al producto del peso del enlace y el valor de la información que fluye por esa conexión.

<sup>31</sup> Este termino es conocido como "bias".

- a) **Modelo lineal.** En este modelo la activación de la neurona es igual a la entrada total.
- b) **Modelo lineal con umbral.** El nivel de activación toma un valor binario en dependencia del signo de la entrada total. Ejemplo de este modelo es la función signo.
- c) **Modelo estocástico.** La activación se calcula según una regla probabilística, tomando valor 1, con una probabilidad  $p$ , según la expresión:  $P(O_i=1) = ( 1 / (1 + e^{-S_i/T}) )$ . Este modelo de neurona se utiliza en la maquina de Boltzmann.
- d) **Modelo continuo.** La salida está relacionada no linealmente con la entrada según la función:  $O_i = 1 / (1 + e^{-S_i})$ . Esta función continua transforma el valor de la entrada total a valores reales entre 0 y 1. Se utiliza en algoritmos de aprendizaje que requieren la continuidad de la función de activación, como es el caso del de propagación de los errores hacia atrás. Una alternativa para calcular la entrada total es utilizar un polinomio de grado mayor que uno. El enfoque polinomial permite implementar numerosos clasificadores no lineales, los cuales superan las limitaciones de clasificadores lineales como el modelo de Perceptrón original. En este sentido esta variante es equivalente al uso de redes con capas ocultas.

#### 4.3. REDES NEURONALES ARTIFICIALES

Una red neuronal artificial, "intenta ser" la representación matemática de una red neuronal biológica. Se dice que intenta ser, porque dada la complejidad todavía no resuelta del funcionamiento del cerebro, todos los trabajos hasta ahora desarrollados son muy burdos en comparación de lo que realmente es, esto es en gran parte debido a las limitantes tecnológicas actuales.

Modelar significa abstraer y simplificar a partir de características generales reales. La elaboración de modelos de neuronas y circuitos de neuronas que intentan reproducir de una manera plausible la arquitectura y funcionamiento del sistema nervioso central, ha sufrido profundas transformaciones en su enfoque a lo largo de la última década, sobre todo por los continuos avances de la neurobiología experimental.

Desde hace algunos años, algunos investigadores han estado creando modelos, tanto en hardware como en software, que interpretan la actividad cerebral en un esfuerzo por producir una forma de inteligencia artificial. Muchos modelos teóricos o paradigmas, datan desde los años 50's. Muchos de ellos tenían aplicaciones limitadas en el mundo real, teniendo como consecuencia que las Redes Neuronales Artificiales (RNA) permanecieran en la oscuridad por décadas.

Las RNA están compuestas de un gran número elementos de procesamiento altamente interconectados, denominados neuronas, trabajando al mismo tiempo para la solución de problemas específicos. Las RNA, tal como las personas, aprenden de la experiencia.

En cualquier caso, se trata de una nueva forma de computación que es capaz de manejar las imprecisiones e incertidumbres que aparecen cuando se trata de resolver problemas relacionados con el mundo real<sup>32</sup>, ofreciendo soluciones robustas y de fácil implementación.

Las RNA están compuestas de muchos elementos sencillos que operan en paralelo, el diseño de la red está determinado mayormente por las conexiones entre sus elementos. Al igual que las conexiones de las neuronas cerebrales.

---

<sup>32</sup> Entre otras cosas clasificación, reconocimiento de patrones, predicción y toma de decisiones.



La idea de las redes neuronales fue concebida originalmente como un intento de modelar la fisiología del cerebro humano, esto es, entender y explicar como funciona y opera el cerebro. La meta era crear un modelo capaz en emular el proceso humano de razonamiento. La mayor parte de los trabajos iniciales en redes neuronales fue realizada por fisiólogos y no por ingenieros.

Existen varias formas de hacer las conexiones en una RNA, así como existen varias formas de conectar neuronas biológicas en el cerebro. Cada tipo sirve para diferentes procesos, el elegir la correcta topología y sus características, es imprescindible para lograr fácilmente la solución del problema.

#### 4.3.1. Taxonomía

Existen dos fases en toda aplicación de las redes neuronales: la *fase de aprendizaje o entrenamiento* y la *fase de prueba*. En la fase de entrenamiento, se usa un conjunto de datos o patrones de entrenamiento para determinar los pesos (parámetros de diseño) que definen el modelo neuronal. Una vez entrenado este modelo, se usará en la llamada fase de prueba o funcionamiento directo, en la que se procesan los patrones de prueba que constituyen la entrada habitual de la red, analizándose de esta manera las prestaciones definitivas de la red.

- a) **Fase de Prueba**. En esta fase los parámetros de diseño de la red neuronal se obtienen a partir de unos patrones representativos de las entradas que se denominan *patrones de entrenamiento*. Los resultados pueden ser calculados tanto de una vez como adaptados iterativamente, según el tipo de red neuronal, y en función de las ecuaciones dinámicas de prueba. Una vez calculados los pesos de la red, los valores de las neuronas de la última capa, se comparan con la salida deseada para determinar la validez del diseño.
- b) **Fase de Aprendizaje**. En esta fase se reconoce que una característica de las redes neuronales es su capacidad de aprender. Aprenden por la actualización o cambio de los pesos sinápticos que caracterizan a las conexiones. Los pesos son adaptados de acuerdo a la información extraída de los patrones de entrenamiento nuevos que se van presentando. Normalmente, los pesos óptimos se obtienen optimizando<sup>33</sup> alguna "función de energía".

#### 4.3.2. Aprendizaje

La habilidad para aprender es un rasgo fundamental de la inteligencia. Aunque lo que se entiende por aprendizaje es difícil de describir, un proceso de aprendizaje, en el contexto de las redes neuronales artificiales puede ser considerado como el problema de actualizar la arquitectura de la red y los pesos de conexión de modo tal que una red pueda ejecutar de manera eficiente una tarea específica.

Para las redes neuronales artificiales existen dos métodos de aprendizaje ampliamente conocidos:

- a) **Supervisado**. En este aprendizaje los datos están constituidos por varios patrones de entrada y de salida. El hecho de conocer la salida implica que el entrenamiento se beneficia la supervisión de un maestro.
- b) **No Supervisado**. En este aprendizaje, el conjunto de datos de entrenamiento consiste solamente en los patrones de entrada. Por lo tanto, la red es entrenada sin el beneficio de

---

<sup>33</sup> Aunque un tanto redundante, entiéndase en este contexto como los procesos de maximización y minimización.

un maestro. La red aprende a adaptarse basada en las experiencias recogidas de los patrones de entrenamiento anteriores.

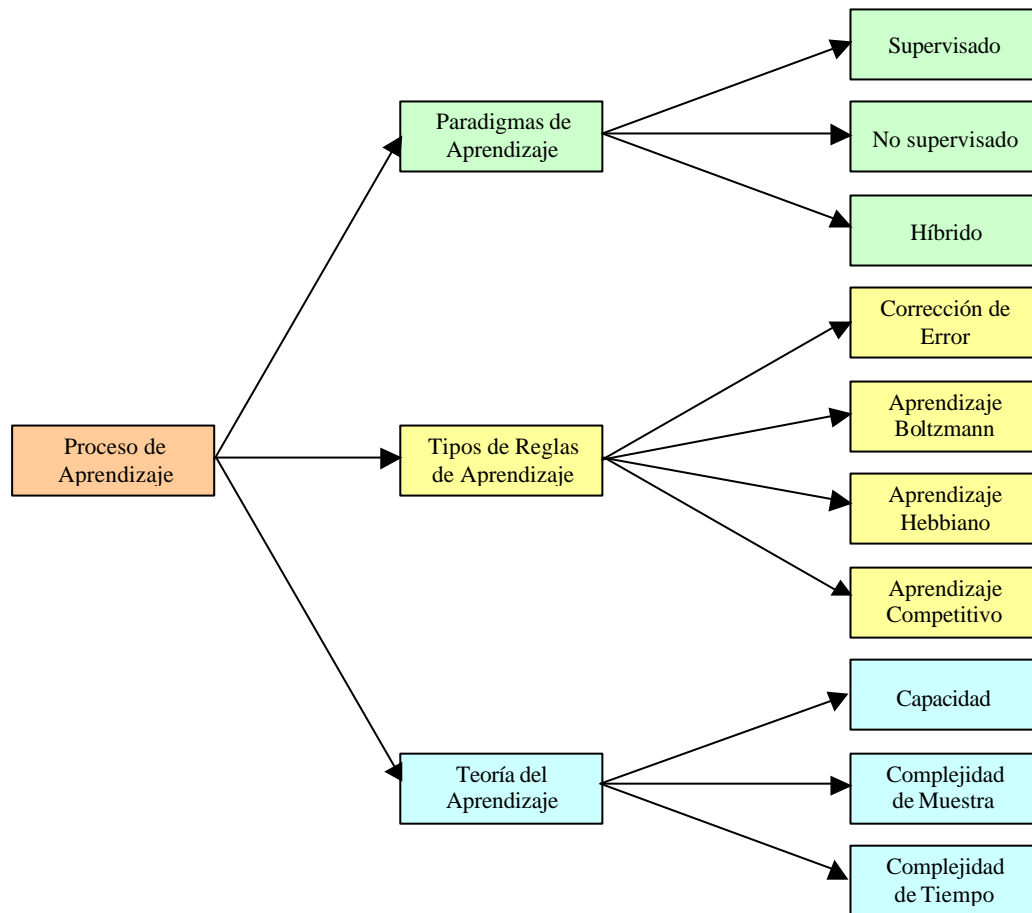


Fig. 4.5. Taxonomía del aprendizaje conexionista

#### 4.4. MODELOS DE REDES NEURONALES

Los modelos de redes neuronales son especificados por las tres siguientes características:

- Topología de la red.** Hace referencia al lugar donde se describen la estructura y el tipo de enlaces.
- Características de los nodos.** Que se refiere al modelo de la neurona.
- Reglas de aprendizaje.** Hace referencia al método de ajuste de los pesos.

En principio, siempre se puede encontrar una red neuronal que pueda resolver un problema dado, partiendo de que no hay restricción sobre el tamaño de la red y existe disponible una cantidad infinita de datos.

##### 4.4.1. Caracterización

Una red neuronal es un modelo computacional que pretende simular el funcionamiento del cerebro a partir del desarrollo de una arquitectura que toma rasgos del funcionamiento de este

órgano sin llegar a desarrollar una replica del mismo. El cerebro puede ser considerado como un equipo integrado por aproximadamente 10 billones de elementos de procesamiento cuya velocidad de cálculo es lenta, pero que trabajan en paralelo y con este paralelismo logran alcanzar una alta potencia de procesamiento.

A partir de esta visión del cerebro, el modelo computacional desarrollado consiste de un conjunto de elementos computacionales simples, las cuales constituyen neuronas artificiales, que están unidas por arcos dirigidos que les permiten comunicarse. Cada arco tiene asociado un peso numérico  $W_{ij}$  que indica la activación alcanzada por la unidad  $U_i$  sobre la unidad  $U_j$ . Cada celda  $U_i$  calcula una activación a partir de las activaciones de las celdas conectadas directamente a ellas, de los pesos de los arcos a través del que llega cada activación y utilizando un algoritmo que generalmente es el mismo para todas las unidades. En la literatura sobre redes neuronales tales elementos son frecuentemente referidos como neurona adaptativa.

Matemáticamente el funcionamiento de una red neuronal se representa del siguiente modo. Toda unidad  $U_j$ , excepto las entradas, calcula una nueva activación  $U'_j$  como una función<sup>34</sup> que tiene como argumento la suma pesada de las entradas a la unidad:

$$\begin{aligned} S_j &= \sum_i W_{ij} U_i && \text{(nivel de voltaje)} \\ U'_j &= F(S_j) && \text{(intensidad)} \end{aligned}$$

El valor  $S_j$  se puede interpretar como el nivel de voltaje que excita la neurona, y el valor  $U'_j$  denota la intensidad de la salida resultante de la neurona.

Si  $U_i$  no está conectada a  $U_j$  entonces  $W_{ij}=0$ . Por convenio existe una unidad  $U_0$  con activación siempre igual a 1 que está conectada al resto de los elementos de procesamiento y el peso  $W_{0j}$  es una constante que representa una predisposición de la unidad.

Los pesos positivos ( $W_{ij}>0$ ) indican que el nivel de actividad de la unidad  $U_i$  refuerza el nivel correspondiente del elemento  $U_j$ , mientras que un peso negativo ( $W_{ij}<0$ ) representa una inhibición. Los pesos determinan el comportamiento de la red, desde el punto de vista de la inteligencia artificial el conjunto de pesos  $W$  encierra el conocimiento del dominio de aplicación que la red neuronal modela.

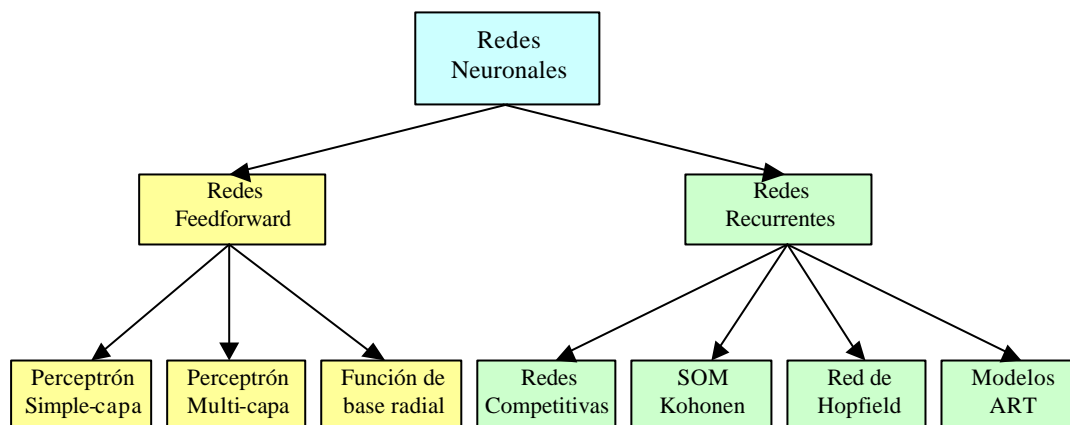
La función  $F$  se conoce como la función de activación y puede ser una función sólida como la función signo, o una función no lineal diferenciable creciente monótonamente en forma de S tal como la tangente hiperbólica.

#### 4.4.2. Topologías

Se define como topología de una red neuronal a la organización o arquitectura del conjunto de neuronas que la forman; esta organización comprende la distribución espacial de las mismas y los enlaces entre ellas. Las topologías básicas se describen en la Fig. 4.6.

---

<sup>34</sup> Esta función también es referenciada como algoritmo de cálculo.



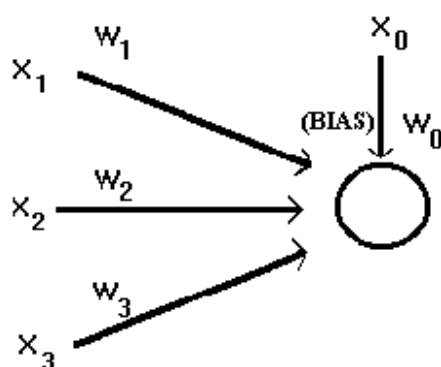
**Fig. 4.6. Clasificación de topologías**

Como se observa en la clasificación de las topologías o arquitecturas de las redes neuronales artificiales, existen diferentes topologías que pueden ser aplicadas con diferentes soluciones. Las topologías que resaltan para su estudio ulterior son las siguientes: Perceptrón, Propagación de Errores hacia Atrás<sup>35</sup>, Hopfield y Kohonen.

## 4.5. TOPOLOGÍAS GENERALES

### 4.5.1. Neurona Simple

La expresión mas simplificada de una red es aquella en la cual se tiene solamente una neurona. Esta funciona como una unidad de procesamiento que recibe entradas y calcula un nivel de activación que definirá su respuesta. Un ejemplo de esta arquitectura es la neurona de McCulloch y Pitts.



**Fig. 4.7. Topología de neurona simple**

### 4.5.2. Red Simple

En esta topología se organiza un conjunto de  $N$  neuronas de la siguiente forma. Se utilizan  $M$  unidades sensoras, las cuales captan la información de entrada, estas  $M$  unidades se conectan a las  $N$  neuronas mediante caminos pesados. No existe conexión entre las neuronas. Cada neurona

<sup>35</sup> Mas conocida como Red Backpropagation.

calcula una respuesta, y la salida de la red será un vector con N componentes, uno por cada neurona.

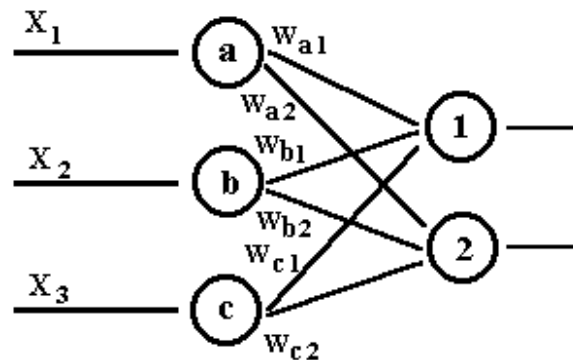


Fig. 4.8. Topología de red simple

#### 4.5.3. Red con Elementos de Asociación

La arquitectura de red simple se puede desarrollar introduciendo un conjunto de unidades de asociación en las cuales se combinan las entradas y las combinaciones producidas sirven de entrada a las neuronas, las cuales calculan las salidas de la red. Esta topología es muy similar a la arquitectura propuesta originalmente para el Perceptrón.

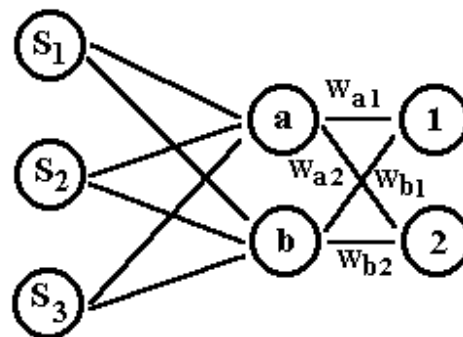


Fig. 4.9. Topología de red con unidades de asociación

#### 4.5.4. Redes Multicapa

Una de las topologías más poderosas y difundidas ordena el conjunto de elementos de procesamiento en niveles, de modo que los enlaces se establecen desde unidades en el nivel  $i$  a unidades en el nivel  $j$  ( $i < j$ ), por lo que la información fluye unidireccionalmente desde las unidades de entrada a las unidades de salida. Esta arquitectura se conoce con el nombre de capa dirigida hacia adelante.

Típicamente existe una capa de unidades sensorias, una o más capas ocultas de neuronas y una capa de neuronas que producen la salida. El calificativo de oculta es para denotar que las capas

permanecen "desconocidas" para el usuario de la red. La utilización de esta topología fue posible gracias al desarrollo de algoritmos de aprendizaje como el backpropagation. Esta topología es la base del multilayer perceptrón y del modelo MADALINE.

En esta topología resulta de interés analizar lo referente a cuantos niveles o capas, así como la cantidad de neuronas por capas, son necesarios. En el nivel inicial existe una unidad sensoria por cada rasgo de entrada a la red, en la capa de salida se colocan tantas neuronas como sean necesarias. Se ha probado que dos capas ocultas son suficientes para resolver cualquier problema. No existe un criterio riguroso para determinar la cantidad de neuronas en las capas ocultas.

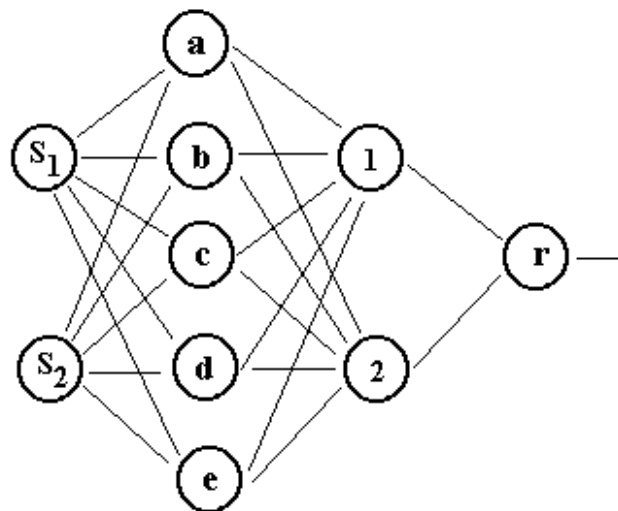


Fig. 4.10. Topología de red con capas ocultas

#### 4.5.5. Modelo Interactivo

Se tiene un conjunto de  $N$  neuronas las cuales se conectan completa y mutuamente, es decir, todas las unidades sirven como unidades de entrada y como neuronas para calcular la salida; cada neurona se conecta a las  $N-1$  restantes mediante caminos pesados. El procedimiento de cálculo de las salidas se realiza mediante un procedimiento iterativo en el cual las neuronas se inicializan con los valores de entrada y recalculan estos valores hasta caer en un estado estable. La red de Hopfield y el modelo con activación interactiva y competencia se basan en esta topología.

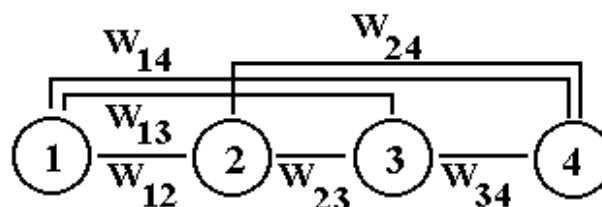


Fig. 4.11. Topología del modelo interactivo

#### 4.5.6. Modelo Interactivo Desarrollado

En este modelo la red está formada por dos subredes. La primera tiene la topología de una red simple y la segunda la del modelo interactivo. La red simple recibe las entradas y las procesa, las salidas de esta primera subred sirven de entrada a la red interactiva, la cual realiza el proceso iterativo de cálculo de las salidas. Esta topología ha mostrado que tiene las mismas capacidades del modelo interactivo y es más eficiente. Los modelos de Hamming, de Carpenter/Grossberg y de Kohonen se basan en esta topología.

#### 4.6. TOPOLOGÍAS ESPECÍFICAS

##### 4.6.1. Perceptrón

###### a) Antecedentes

En 1943, Warren McCulloch y Walter Pitts originaron el primer modelo de operación neuronal, el cual fue mejorado en sus aspectos biológicos por Donald Hebb en 1948.

En 1962 Bernard Widrow propuso la regla de aprendizaje Widrow-Hoff, y Frank Rosenblatt desarrolló una prueba de convergencia, y definió el rango de problemas para los que su algoritmo aseguraba una solución. Widrow propuso el Perceptrón como herramienta computacional.

###### b) Funcionamiento

En la Fig. 4.12. se representa una neurona artificial, que intenta modelar el comportamiento de la neurona biológica. Aquí el cuerpo de la neurona se representa como un sumador lineal de los estímulos externos  $x_j$ , seguida de una función no lineal  $y_j = f(z_j)$ . La función  $f(z_j)$  es llamada la función de activación, y es la función que utiliza la suma de estímulos para determinar la actividad de salida de la neurona.

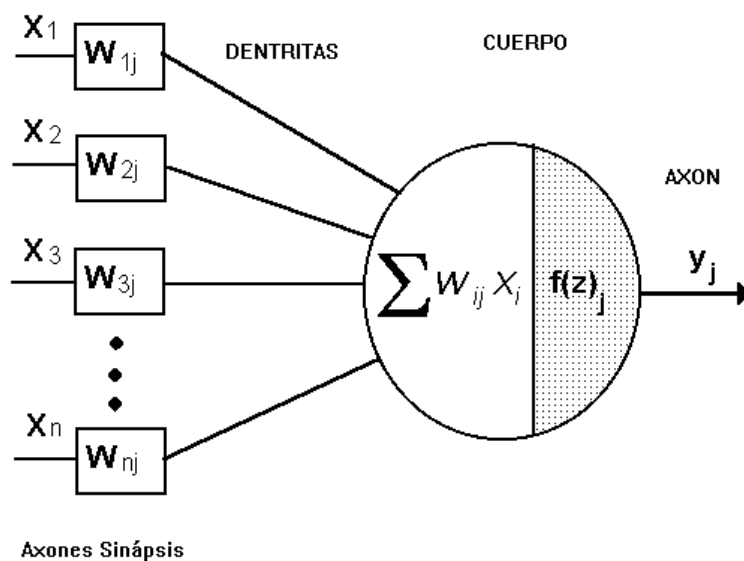


Fig. 4.12. Perceptrón

Este modelo se conoce como Perceptrón de McCulloch-Pitts, y es la base de la mayor parte de las arquitecturas de RNA que se interconectan entre sí. Las neuronas emplean funciones de activación diferentes según la aplicación, algunas veces son funciones lineales, otras funciones sigmoidales y otras funciones de umbral. La eficiencia sináptica se representa por factores de peso de interconexión  $w_{ij}$ , desde la neurona  $i$ , hasta la neurona  $j$ .

Los pesos pueden ser positivos o negativos<sup>36</sup>. Los pesos junto con las funciones  $f(z)$  dictan la operación de la red neuronal. Normalmente las funciones no se modifican de forma tal que el estado de la red neuronal depende del valor de los factores de peso o sinapsis que se aplica a los estímulos de la neurona.

En un Perceptrón, cada entrada es multiplicada por el peso  $W$  correspondiente, y los resultados son sumados, siendo evaluados contra el valor de umbral, si el resultado es mayor al mismo, el Perceptrón se activa.

### c) Limitantes

El Perceptrón es capaz tan sólo de resolver funciones definidas por un hiperplano<sup>37</sup>, que corte un espacio de dimensión  $N$ . Un ejemplo de una función que no puede ser resuelta es el operador lógico XOR.

Una explicación más sencilla de un hiperplano sería, hablando en un plano de dos dimensiones, una línea que separa a los elementos existentes en dos grupos. El Perceptrón sólo puede resolver una función, si todos los posibles resultados del problema pueden separarse de ésta forma, es decir, que no se combinen entre sí. Estos elementos son linealmente separables.

### d) Entrenamiento

El entrenamiento de un Perceptrón es por medio de la regla de aprendizaje delta: Para cada peso  $W$  se realiza un ajuste  $dW$  según la regla:  $dW = LR(T - Y)X$ . Donde  $LR$  es la razón de aprendizaje,  $T$  el valor deseado,  $Y$  el valor obtenido y  $X$  la entrada aplicada al Perceptrón.

### e) Tipos de Perceptrón

El Perceptrón básico de dos capas<sup>38</sup> solo puede establecer dos regiones separadas por una frontera lineal en el espacio de patrones de entrada, donde se tendría un hiperplano.

Un Perceptrón con tres niveles de neuronas puede formar cualquier región convexa en este espacio. Las regiones convexas se forman mediante la intersección entre las regiones formadas por cada neurona de la segunda capa, cada uno de estos elementos se comporta como un Perceptrón simple, activándose su salida para los patrones de un lado del hiperplano.

Un Perceptrón con cuatro capas puede generar regiones de decisión arbitrariamente complejas. El proceso de separación en clases que se lleva a cabo consiste en la partición de la región deseada en pequeños hipercubos. Cada hipercubo requiere  $2n$  neuronas en la segunda capa (siendo  $n$  el

---

<sup>36</sup> Estos pesos también reciben el nombre de pesos activadores o positivos y pesos inhibidores o negativos.

<sup>37</sup> Objeto de dimensión  $N-1$  contenido en un espacio de dimensión  $N$ .

<sup>38</sup> Entrada con neuronas lineales, analógicas, y la de salida con función de activación de tipo escalón, digital.



numero de entradas a la red), una por cada lado del hipercubo, y otra en la tercera capa, que lleva a cabo la conjunción lógica de la salida de los nodos del nivel anterior. La salida de los nodos de este tercer nivel se activaran solo para las entradas de cada hipercubo. Los hipercubos se asignan a la región de decisión adecuada mediante la conexión de la salida de cada nodo del tercer nivel solo con la neurona de salida (cuarta capa) correspondiente a la región de decisión en la que este comprendido el hipercubo llevándose a cabo una operación lógica OR en cada nodo de salida. Este procedimiento se puede generalizar de manera que la forma de las regiones convexas sea arbitraria, en lugar de hipercubos.

En teoría, el Perceptrón de 4 capas puede resolver una gran variedad de problemas cuyas entradas sean analógicas, la salida sea digital y sea linealmente separable. El problema práctico radica en el numero de neuronas, en el numero idóneo de capas ocultas, la extensión de la función de activación, el tiempo de entrenamiento de la red, las implicaciones en la generación de ruido<sup>39</sup> en contraparte con la ventaja de tener un sistema tolerante a fallas al tener un número de neuronas redundante.

#### **f) Aplicaciones del Perceptrón**

El rango de tareas que el Perceptrón puede manejar es mucho mayor que solamente simples decisiones y reconocimiento de patrones. Por ejemplo, se puede entrenar una red para formar el tiempo pasado de los verbos en ingles, leer texto en ingles y manuscrito. El Perceptrón multicapa (MLP) puede ser usado para la predicción de una serie de datos en el tiempo; este modelo ha sido exitoso en la medición de la demanda de gas y electricidad, además de la predicción de cambios en el valor de los instrumentos financieros.

En la predicción de mercados financieros, diagnósticos médicos, el Perceptrón como una red codificadora, el Perceptrón como sumador de números enteros.

NETtalk es un Perceptrón que es capaz de transformar texto en ingles en sonido individual (representaciones fonéticas) y la pronunciación con la utilización de un sintetizador de voz; cuenta con aproximadamente 300 nodos de neuronas (siendo 80 en la capa escondida) y 20000 conexiones individuales.

#### **4.6.2. Retropropagación (Backpropagation)**

El Perceptrón solo es el ejemplo más elemental de una red neuronal, de hecho, no puede siquiera ser considerado una "red", puesto que no intervienen otros elementos. Si se combinan varios perceptrones en una "capa", y los estímulos de entrada después se suman, recién se tendrá una red neuronal. Una red neuronal muy eficaz para resolver fundamentalmente problemas de reconocimiento de patrones es la red neuronal de retropropagación, en inglés back propagation network.

---

<sup>39</sup> Normalmente se produce al tener un numero excesivo de neuronas.

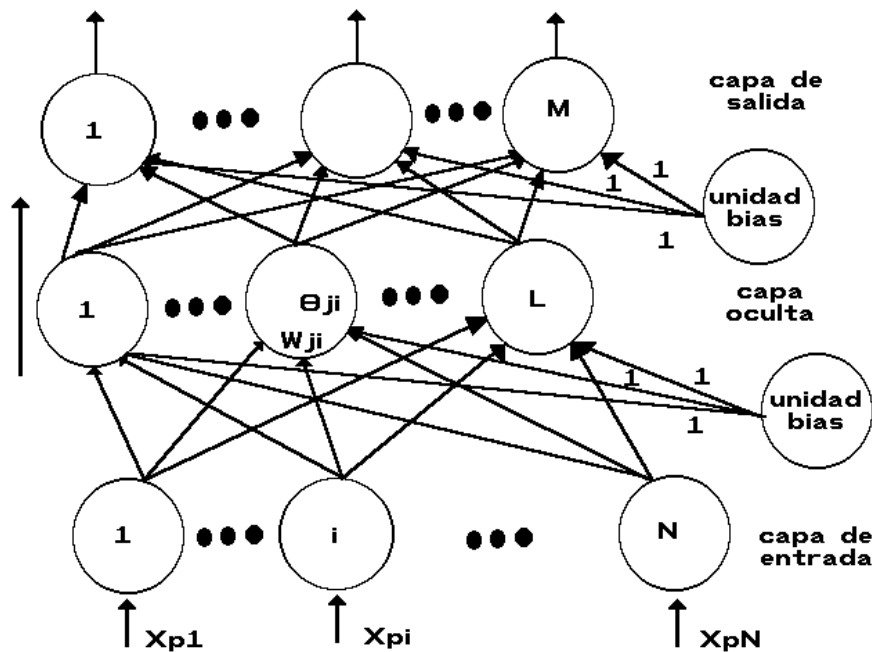


Fig. 4.13. Red de retropropagación

En esta red, se interconectan varias unidades de procesamiento en capas, las neuronas de cada capa no se interconectan entre sí. Sin embargo, cada neurona de una capa proporciona una entrada a cada una de las neuronas de la siguiente capa, esto es, cada neurona transmitirá su señal de salida a cada neurona de la capa siguiente. La Fig. 4.13. muestra un ejemplo esquemático de la arquitectura de este tipo de redes neuronales.

#### b) Algoritmo

El algoritmo de aprendizaje proporciona una forma de entrenar una red multicapa con alimentación hacia adelante. Comienza alimentando los valores de la entrada de acuerdo a las siguientes ecuaciones:

$$net_i = \sum_{j \in A} O_j W_{ji} + \theta_i \quad i \in B$$

donde  $A$  es el grupo de neuronas en una capa, y  $B$  es la otra.  $O_j$  es la activación para la neurona  $j$ ,  $W_{ji}$  son los pesos asignados a la conexión entre las neuronas  $j$  e  $i$ . En la ecuación anterior, se toman los valores de salida y se alimentan a la siguiente capa a través de los pesos. Esta operación se realiza para cada neurona en la siguiente capa, produciendo un valor de red. Este valor es la suma de todos los valores de activación en las neuronas de la capa anterior, y cada valor de red es aplicado a la siguiente ecuación, conocida como función de activación, para producir la activación de esa neurona.

$$O_i = f(net_i) = \frac{1}{1 + e^{-net_i}}$$

Después de que todas las neuronas tienen un valor de activación asociado a un patrón de valores de entrada, el algoritmo sigue buscando errores en cada neurona que no es de entrada.

Los errores encontrados para las neuronas de salidas, son propagados hacia atrás, a la capa anterior para que puedan ser asignados a neuronas de las capas escondidas, esto se calcula por:

$$\delta_i = f'(net_i) \sum_{j \in E} W_{ij} \delta_j$$

donde  $D$  es el grupo de neuronas en una capa que no es de entrada y  $E$  es el grupo de neuronas de la siguiente capa. Este cálculo se repite para cada capa escondida en la red.

Después de que se ha encontrado la activación y el error asociado a cada grupo de neuronas, los pesos se actualizan, primero encontrando el valor con el que cada peso debe modificarse, esto se logra calculando:

$$\Delta W_{ij} = CO_i \delta_j \delta_i, i \in A, j \in B$$

donde  $C$ , conocida como la razón de aprendizaje, es una constante que controla el valor del cambio de los pesos y  $\Delta W_{ij}$  es el cambio de los pesos entre la neurona  $i$  y  $j$ . El peso se cambia evaluando:

$$W_{ij,t+1} = W_{ij,t} + \Delta W_{ij}$$

#### 4.6.3. Hopfield

Las redes de Hopfield son redes de adaptación probabilística, recurrentes, funcionalmente entrarían en la categoría de las memorias autoasociativas, es decir, que aprenden a reconstruir los patrones de entrada que memorizaron durante el entrenamiento. Son arquitecturas de una capa con interconexión total, son funciones de activación booleana de umbral<sup>40</sup> donde se produce adaptación probabilística de la activación de las unidades, conexiones recurrentes y simétricas, y regla de aprendizaje no supervisado. Mientras que las redes en cascada (no recurrentes) dan soluciones estables, los modelos recurrentes dan soluciones inestables (dinámicas), lo que no siempre es aconsejable.

La principal aportación de Hopfield consistió precisamente en conseguir que tales redes recurrentes fueran así mismo estables. Imaginó un sistema físico capaz de operar como una memoria autoasociativa, que almacenara información y fuera capaz de recuperarla aunque la misma se hubiera deteriorado.

La Red de Hopfield es recurrente y completamente conectada. Funciona como una memoria asociativa no lineal que puede almacenar internamente patrones presentados de forma incompleta o con ruido. De esta forma puede ser utilizada como una herramienta de mejora u optimización

---

<sup>40</sup> Cada unidad puede tomar dos estados, 0 o 1, dependiendo de si la estimulación total recibida supera determinado umbral.

de problemas. El estado de cada neurona puede ser actualizado un número indefinido de veces, independientemente del resto de las neuronas de la red pero en paralelo.

### a) Máquina de Boltzmann

En la Máquina de Boltzmann, generalización de la red de Hopfield que incluye unidades ocultas, la operación de actualización se basa en un concepto de termodinámica estadística conocido como "temple simulado"<sup>41</sup>. La red de Hopfield, la máquina de Boltzmann y un derivado conocido como la máquina del teorema de campo medio se han utilizado en aplicaciones de segmentación y restauración de imágenes y optimización combinatorial.

### b) Características

La red de Hopfield consiste en un conjunto de N elementos de procesamiento interconectados que actualizan sus valores de activación de forma asíncrona e independiente del resto de los elementos de proceso. Todos los elementos son a la vez de entrada y salida. Los valores de activación son binarios.

El estado del sistema está dado por los valores de activación  $Y_k$ . La entrada de la neurona  $k$  en el ciclo temporal  $t+1$  viene dada por

$$s_k(t+1) = \sum_{j \neq k} y_j(t) w_{jk} + \theta_k$$

Para obtener el nuevo valor de activación se aplica una función umbral. Cuando un elemento de procesamiento mantiene su valor de activación se dice que es estable. Se llama estado estable a aquel en el cual todos los elementos de procesamiento son estables.

Con la restricción extra de simetría en los pesos de conexión,  $W_{jk} = W_{kj}$ , el sistema puede ser descrito mediante una función energía de la forma

$$E = -\frac{1}{2} \sum_{j \neq k} \sum_k y_j y_k w_{jk} - \sum_k \theta_k y_k$$

### c) Funcionamiento

A cada estado de la red se le puede atribuir una cierta cantidad de energía, el sistema evoluciona tratando de disminuir la energía mediante un proceso de relajación, hasta alcanzar un mínimo donde se estabiliza. Los mínimos de energía se corresponden con los recuerdos almacenados durante el aprendizaje de la red.

Ante la presentación de un estímulo nuevo se obtendrá una configuración inicial más o menos parecida a alguno de los estímulos almacenados, el sistema evolucionará hasta caer en una configuración estable que representa el recuerdo asociado a ese estímulo. Si la configuración inicial discrepa mucho de los recuerdos almacenados es posible alcanzar algún mínimo que no se corresponde a ningún recuerdo almacenado, recuperando en ese caso una información espuria, o se podría no alcanzar ningún mínimo, quedando inestable: en ese caso se dice que la red está "confundida", no es capaz de reconocer el estímulo, no recuerda.

---

<sup>41</sup> Del término original *simulated annealing*.

Una tercera posibilidad es que al cabo de unos pasos de evolución empiece a repetir periódicamente una secuencia definida de estados; con esta dinámica se han modelado ciertas excitaciones nerviosas que regulan acciones rítmicas y repetitivas; y se ha tratado de reproducir la memoria de secuencias temporales, pe. el recuerdo de melodías.

#### d) Modelo de Red de Hopfield de 3 unidades

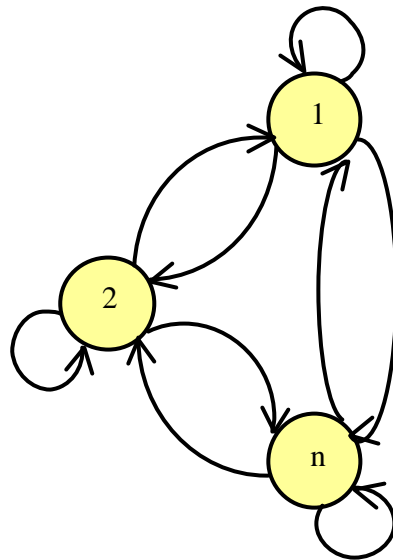


Fig. 4.14. Neuronas interconectadas

Como se había mencionado, las neuronas se conectan todas entre sí, y consigo mismas. Para empezar se le asigna a cada unidad el valor o estado correspondiente del patrón de entrada. En cada ciclo se elige una neurona al azar y se calcula su activación según la función de umbral:

Sea  $n$  el número de neuronas en la red, la estimulación total se calcula como la sumatoria de todas las entradas ponderadas, incluida la procedente de la misma unidad.

$$x_j = \sum_{i=1}^n W_{ij} Y_i$$

Se puede trabajar con cualquier valor de umbral para la función de activación, pero típicamente se usa el 0 como umbral; tiene la ventaja de simplificar las ecuaciones.

$$y_i = 1 \text{ si } x_i > 0 \quad y_i = 0 \text{ si } x_i < 0$$

Por definición los recuerdos<sup>42</sup> son puntos fijos en la dinámica de la red, es decir, configuraciones que no cambian en el tiempo aunque se siga aplicando la regla de evolución. Para almacenar un recuerdo habrá que lograr que la presentación del patrón de entrada lleve a la red a alcanzar un punto fijo, esto se logrará mediante alguna regla de aprendizaje que modifique los pesos de las conexiones.

<sup>42</sup> En este caso hace referencia a ítems o patrones almacenados.

### e) Aprendizaje de las redes de Hopfield: Regla de Cooper-Hebb

La elección de la regla de aprendizaje no es trivial, depende de la interrelación de los patrones que se desea memorizar. Si estos patrones están poco correlacionados (pseudoortogonales) se puede aplicar la regla de Cooper-Hebb, basada en la regla de Hebb o regla del producto. Suponga que se tiene M ítems o patrones (binarios) que almacenar ( $p_1.. p_m$ ), se calcula los pesos de la siguiente forma:

$$W_{ij} = W_{ji} = \sum_{m=1}^M (2p_i^m - 1)(2p_j^m - 1)$$

$w_{ij} = 0$  si  $i = j$  (y la diagonal vale 0)

Esta regla fortalece las conexiones cuando las unidades i-ésima y j-ésima tienen la misma activación o valor de estado, y las debilita en caso contrario.

El entrenamiento cuenta con los siguientes pasos.

1. Elegir un número de neuronas que cumpla el criterio del 15%
2. Codificar los ítems que se requieran memorizar de forma que los patrones para representarlos se parezcan lo menos posible entre sí, para aproximarse a la condición de pseudo-ortogonalidad.
3. Calcular los pesos de las conexiones según la regla de Cooper-Hebb.
4. Una vez conseguida la red lista para usarla, asignando a las unidades el estado inicial, se deja que la red evolucione hasta alcanzar un mínimo (esto se lo consigue comprobando que el decremento de energía sea nulo).

### f) Condiciones de estabilidad

Como demostró Hopfield, una red de adaptación asíncrona estable se puede conseguir haciendo que la matriz de pesos sea simétrica y con ceros en la diagonal principal. La idea para alcanzar estados estables es encontrar una función del estado del sistema, que se llama energía, con la propiedad de decrecer cuando se produce un cambio en el estado de la red. Un tipo de funciones con esa propiedad son las funciones de Liapunov. Una función de este tipo, empleando umbral=0, es la siguiente:

$$E = -\frac{1}{2} \sum_i \sum_{j \neq i} (w_{ij} y_i y_j) - \sum_i \theta_i y_i$$

dónde  $y_i$  representa la activación o salida de la unidad i-ésima

El cambio de energía del sistema ante el cambio de estado de una unidad  $i$  elegida al azar ( $\Delta y_i$ ) es:

$$\begin{aligned} \Delta_k E &= E(k+1) - E(k) \\ \Delta_k E &= -\text{net}_i(k+1) \cdot \Delta y_i(k+1) - \frac{1}{2} w_{ii} \end{aligned}$$

Que como  $w_{ii} = 0$  queda

$$\Delta_k E = -net_i(k+1) \cdot \Delta y_i(k+1)$$

Se puede demostrar que dicho cambio de energía es siempre negativo o nulo, de tal modo que el sistema tiende a converger hacia un mínimo. El criterio de simetría es suficiente, pero no necesario, para alcanzar la estabilidad. Por otro lado la simetría aproximada es usualmente suficiente para lograrlo.

### g) Valoración

Las redes de Hopfield se han aplicado a campos como la percepción el reconocimiento de imágenes y optimización de problemas, mostrando gran inmunidad al ruido y robustez. Incluso se han llegado a desarrollar chips específicos para este tipo redes. El estudio de las representaciones de secuencias temporales es un área de gran interés, con aplicaciones en reconocimiento automático de voces y movimientos.

Hopfield ha mostrado como aplicar los mismos principios con funciones de activación continuas como la función sigmoideal, con muy pocas modificaciones

Pero pese a sus evidentes ventajas no están exentas de los siguientes problemas:

- ?? El número máximo de patrones no correlacionados que puede almacenar es igual al 15% del número de neuronas de la red.
- ?? Requieren mucho tiempo de procesamiento hasta converger a una solución estable, lo que limita su aplicabilidad.
- ?? Otro de los problemas achacados a las redes de Hopfield es su tendencia a caer en mínimos locales, como en las redes de retropropagación. La solución pasa por aplicar los métodos estadísticos, comentado en el apartado dedicado a las redes de retropropagación, relacionados con el equilibrio termodinámico simulado.

#### 4.6.4. Kohonen

Existen evidencias que demuestran que en el cerebro existen neuronas que se organizan en muchas zonas, de forma que las informaciones captadas del entorno a través de los órganos sensoriales se representan internamente en forma de capas bidimensionales. Por ejemplo, en el sistema visual se han detectado mapas del espacio visual en zonas de córtex<sup>43</sup>. También en el sistema auditivo se detecta organización según la frecuencia a la que cada neurona alcanza la mayor respuesta<sup>44</sup>.

Aunque en gran medida esta organización neuronal está predeterminada genéticamente, es probable que se origine mediante el aprendizaje. Esto sugiere, por tanto, que el cerebro podría poseer la capacidad inherente de formar mapas topológicos de la información recibida del exterior. De hecho, esta teoría podría explicar su poder de operar con elementos semánticos: algunas áreas del cerebro simplemente podrían crear y ordenar neuronas especializadas o grupos con características de alto nivel y sus combinaciones. Se trataría, en definitiva, de construir mapas espaciales para atributos y características.

---

<sup>43</sup> Se refiere a la capa externa del cerebro.

<sup>44</sup> Es decir la organización tonotópica.

### **a) Historia**

A partir de estas ideas, T. Kohonen presentó en 1982 un sistema con un comportamiento semejante. Se trataba de un modelo de red neuronal con capacidad para formar mapas de características de manera similar a como ocurre en el cerebro.

El objetivo de Kohonen era demostrar que en un estímulo externo, considerado como información de entrada, por si solo, suponiendo una estructura propia y una descripción funcional del comportamiento de la red, era suficiente para forzar la formación de mapas.

Este modelo tiene dos variantes, denominadas LVQ (Learning Vector Quantization) y TPM (Topology-Preserving Map) o SOM (Self-Organizing Map). Ambas se basan en el principio de formación de mapas topológicos para establecer características comunes entre la información (vector) de entrada a la red, aunque difieren en las dimensiones, siendo de una sola dimensión en el caso de LVQ, y bidimensional, e incluso tridimensional, en la red TPM.

### **b) Características**

Pertenece a la categoría de las redes competitivas o mapas de autoorganización, es decir, aprendizaje no supervisado. Poseen una arquitectura de dos capas (entrada-salida) (una sola capa de conexiones), funciones de activación lineales y flujo de información unidireccional<sup>45</sup>.

Las unidades de entrada reciben datos continuos normalizados, se normalizan así mismo los pesos de las conexiones con la capa de salida. Tras el aprendizaje de la red, cada patrón de entrada activará una única unidad de salida.

El objetivo de este tipo de redes es clasificar los patrones de entrada en grupos de características similares, de manera que cada grupo activará siempre la misma salida. Cada grupo de entradas queda representado en los pesos de las conexiones de la unidad de salida triunfante. La unidad de salida ganadora para cada grupo de entradas no se conoce previamente, es necesario averiguarlo después de entrenar a la red.

### **c) Arquitectura**

En la arquitectura de la versión original (LVQ) del modelo Kohonen no existen conexiones hacia atrás. Se trata de una de las  $N$  neuronas entrada y  $M$  de salida. Cada una de las  $N$  neuronas de entrada se conecta a las  $M$  de salida a través de conexiones hacia adelante (feedforward).

Entre las neuronas de la capa de salida, puede decirse que existen conexiones laterales de inhibición implícitas, pues aunque no estén conectadas, cada una de las neuronas tiene cierta influencia sobre sus vecinas. El valor que se asigne a los pesos de las conexiones hacia adelante entre las capas de entrada y salida ( $W_{ji}$ ) durante el proceso de aprendizaje de la red depende precisamente de esta interacción lateral.

La influencia que una neurona ejerce sobre las demás es función de la distancia entre ellas, siendo muy pequeñas cuando están muy alejadas. Es frecuente que dicha influencia tenga la forma de un sombrero mexicano.

---

<sup>45</sup> Es decir son redes en cascada.



Por otra parte, la versión del modelo denominada TPM (Topology Preserving Map) trata de establecer una correspondencia entre los datos de entrada y un espacio bidimensional de salida, creando mapas topológicos de dos dimensiones, de tal forma que ante datos de entrada con características comunes se deben activar neuronas situadas en zonas próximas de la capa de salida.

#### **d) Aprendizaje**

Suponga que se tiene patrones de entrada  $n$ -dimensionales. Inicialmente se tiene que: aleatorizar los pesos de las conexiones; normalizar los pesos de las conexiones incidentes de cada unidad de salida sobre la unidad, dividiendo cada conexión por la raíz cuadrada de la suma de los cuadrados de las conexiones de cada unidad; normalizar igualmente los datos de entrada. Luego se aplica el siguiente conjunto de pasos:

1. Aplicar un patrón de entrada.
2. Calcular alguna medida de similitud/disimilitud (producto interno, distancia euclídea o de Mahalanobis, etc.) entre las entradas y los pesos de las conexiones.
3. La unidad de salida con los pesos más parecidos al patrón de entrada es declarada ganadora. El vector de pesos de la unidad ganadora, se convierte en el centro de un grupo de vectores cercanos a él.
4. Modificar los pesos de los vectores de pesos  $W_j$  "ceranos" a  $W_c$  (distancia menor a  $D$ ). De esta manera se consigue que los vectores de pesos de la unidad ganadora y de su "vecindario" se parezcan cada vez más al patrón de entrada que hace ganar a esa unidad.
5. Repetir los pasos 1 a 4 con todos los patrones de entrada.

A medida que avanza el aprendizaje hay que ir reduciendo  $D$  y  $a$ . Kohonen recomienda empezar con un valor de  $a$  cercano a 1 y reducirlo gradualmente hasta 0.1.  $D$  puede empezar como la máxima distancia existente entre los pesos de las conexiones al principio y acabar siendo tan pequeño que no quede ninguna unidad en el vecindario de la unidad ganadora. En ese momento solo se entrenará una unidad, que al final tendrá su vector de pesos igual al vector de entrada.

La precisión de la clasificación de los patrones de entrada aumenta con el número de ciclos de aprendizaje. Kohonen recomienda una cantidad de ciclos no inferior a 500 veces el número de neuronas de salida para obtener buenos resultados.

### **4.7. APLICACIONES**

Las redes neuronales artificiales constituyen una tecnología computacional emergente que puede ser utilizada en una amplia variedad de aplicaciones, tanto científicas como comerciales. Las redes neuronales pueden ser desarrolladas en periodos de tiempo razonables y pueden realizar tareas concretas de mejor manera que tecnologías convencionales similares, incluyendo los sistemas expertos. Cuando se implementan en el hardware de la computadora, presentan una alta tolerancia a fallas del sistema y proporcionan un grado de paralelismo bastante grande para el procesamiento de los datos.

Desde el punto de vista de las aplicaciones, la ventaja de la RNA reside en el procesamiento paralelo, adaptativo y no lineal. Las RNA han encontrado muchas aplicaciones con éxito en la visión artificial, en el procesamiento de señales e imágenes, reconocimiento del habla y de caracteres, sistemas expertos, análisis de imágenes médicas, control remoto, control de robots, inspección

industrial y exploración científica. El dominio de aplicación de las RNA se puede clasificar de la siguiente forma: asociación y clasificación, regeneración de patrones, regresión y generalización, y optimización.

Existen muchos tipos diferentes de redes neuronales, las aplicaciones actuales se agrupan en los siguientes conjuntos:

- a) **Biología:** Estudio del cerebro y el sistema nervioso. Obtención de modelos de la retina.
- b) **Empresa:** Evaluación de la probabilidad en formaciones geológicas y petrolíferas. Explotación de bases de datos. Optimización de plazas y horarios en líneas de vuelo. Reconocimiento de caracteres escritos.
- c) **Medio Ambiente:** Análisis de tendencias y patrones climáticos. Previsión del tiempo.
- d) **Finanzas:** Previsión de la evolución de los precios. Valoración del riesgo de los créditos. Identificación de falsificaciones. Interpretación de firmas.
- e) **Manufactura:** Robots automatizados y sistemas de control. Control de producción en líneas de proceso. Inspección de calidad.
- f) **Medicina:** Analizadores del habla para ayuda en la audición de sordos. Diagnostico y tratamiento a partir de síntomas. Monitorización de la cirugía. Predicción de reacciones adversas a los medicamentos. Lectores de rayos X. Comprensión de la causa de ataques epilépticos.

#### Ejercicios #4

1. Complemente la importancia de las representaciones probabilísticas del conocimiento investigando lo referente al aprendizaje bayesiano.
2. ¿Cuál es la relación entre el razonamiento difuso y las redes neuronales?
3. Defina la micro estructura de una red neuronal artificial y luego construya una RNA para la disyunción lógica, con los valores “gordo” y “flaco”.
4. Construir una RNA para clasificar los resultados de evaluación de una materia del plan de estudios en los conjuntos difusos: totalmente aprobados, casi aprobados, casi reprobados y totalmente reprobados.
5. Construir una red neuronal artificial para el problema lógico AND, con los valores “arriba” y “abajo”.
6. Diseñe una Red Neuronal Artificial para la siguiente tabla:

| E1   | E2   | E3   | R    |
|------|------|------|------|
| Nada | Nada | Nada | Nada |
| Nada | Nada | Algo | Nada |
| Nada | Algo | Nada | Nada |
| Nada | Algo | Algo | Nada |
| Algo | Nada | Nada | Nada |
| Algo | Nada | Algo | Algo |
| Algo | Algo | Nada | Algo |
| Algo | Algo | Algo | Algo |

Donde Ei representa las entradas y R la salida.

7. Construir una red neuronal artificial para el problema lógico XOR, con los valores “peor” y “mejor”.
8. Utilizar un Perceptrón con dos unidades de entrada y una de salida para modelar las funciones lógicas AND y OR. ¿Se podrá modelar también una puerta XOR?

9. Construir un archivo de datos que contenga tres columnas de datos ( $x$ ;  $\sin(x)$ ;  $\cos(x)$ ) e intentar aproximarlos con un Perceptrón multicapa. Probar varios valores de los parámetros de aprendizaje y momento, luego comparar la convergencia en los distintos casos. ¿Qué valores se recomienda para este problema?
10. Considerar la función no lineal  $y(x) = 20e^{-8.5x}(\ln(0.9x + 0.2) + 1.5)$ . Luego generar un fichero con 50 pares ( $x$ ,  $y(x)$ ) en el intervalo (0,1) para entrenar un Perceptrón multicapa. Generar también un fichero con otros 50 puntos distintos para comprobar la validez de la aproximación.

## Bibliografía

- Arbib, M.A., Erdi, P. y Szentagothai, J. *Neural organization: structure, function and dynamics*. Cambridge, Mass.: MIT Press, 1997.
- Bishop, C.M. *Neural Networks for Pattern Recognition*. Clarendon Press, 1996.
- Demartines, P. y Blayo, F. Kohonen self-organizing maps: Is the normalization necessary?. *Complex Systems*, 6, 1992, pp. 105-123.
- Fausett L. *Fundamentals of Neural Networks*. Prentice-Hall, 1994.
- Garson, G.D. Interpreting neural-network connection weights. *AI Expert*, April 1991, pp. 47-51.
- Hilera, J.R. y Martínez, V.J. *Redes Neuronales Artificiales: Fundamentos, modelos y aplicaciones*. Madrid: Ra-Ma, 1995.
- Gurney K. *An Introduction to Neural Networks*. UCL Press, 1997.
- Hair, J., et al. *Multivariate Data Analysis*. Prentice Hall, 1998.
- Hampshire, J.B., Pearlmutter, B.A. Equivalence Proofs for Multilayer Perceptron Classifiers and the Bayesian Discriminant Function. *Proceedings of the 1990 Connectionist Models Summer School*.
- Haykin, S. *Neural Networks: a Comprehensive Foundation*. Prentice-Hall, 1998.
- Kohonen, T. Analysis of a simple self-organizing process. *Biological Cybernetics*, 44, 1982, pp. 135-140.
- Kohonen, T. *Self-organization and associative memory*. New York: Springer-Verlag, 1989.
- Martín del Brío, B. y Sanz, A. *Redes neuronales y sistemas borrosos*. Madrid: Ra-Ma, 1997.
- Masters, T. *Practical neural networks recipes in C++*. London: Academic Press, 1993.
- Kohonen, T. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, 1982, pp. 59-69.
- Kung S. Y. *Digital Neural Networks*. Prentice Hall, Inc, 1993.
- Lau, C. *Neural Networks, Theoretical Foundations and Analysis*. IEEE Press, 1991.
- Mitchell, T. *Machine Learning*. McGraw-Hill, 1997.
- Murray, A.F. *Applications of Neural Networks*. Kluwer Academic Publishers, 1995.
- Ripley, B.D. *Pattern Recognition & Neural Networks*. Cambridge University Press, 1996.
- Rumelhart, D.E., Hinton, G.E. y Williams, R.J. Learning internal representations by error propagation. En: D.E. Rumelhart y J.L. McClelland (Eds.). *Parallel distributed processing* (pp. 318-362). Cambridge, MA: MIT Press, 1986.
- Taylor, J.G. *Neural networks and their applications*. John Wiley and Sons, 1996.

## Referencias Electrónicas

Computing Research Repository CoRR  
<http://xxx.lanl.gov/archive/cs/intro.html>

Group Method of Data Handling (GMDH)

<http://www.inf.kiev.ua/GMDH-home/>

Int. Journal of Neural Systems

<http://www.wspc.com.sg/journals/ijns/ijns.html>

Intro to Neural Networks - Ron Meir

<http://tiger.technion.ac.il/~rmeir/NNETS2000/NNETcourse.html>

Kernel Machines

<http://www.kernel-machines.org/>

Lyapunov Exponents and Neural Networks

<http://econ.uwindsor.ca/>

Mike Orr- RBF Networks

<http://www.anc.ed.ac.uk/~mjo/>

Neural Networks FAQ

<ftp://ftp.sas.com/pub/neural/FAQ.html>

Neural Networks Search Engine

<http://nnc.cs.tamu.edu/>

NNets Frequently Added Questions FAQ

<http://www.faqs.org/faqs/ai-faq/neural-nets/>

Networks: Computation in Neural Systems

<http://www.iop.org/Journals/ne#featured>

Neural Computing Surveys

<http://www.icsi.berkeley.edu/~jagota/NCS/>

Neuroinformatics at DTU

<http://eivind.imm.dtu.dk/thor/>

Neuro Labs in Europe

<http://neural-server.aston.ac.uk/NN/labs/europe.html>

Neural Networks (Elsevier Science)

<http://www.elsevier.nl/inca/publications/store/8/4/1/>

NETLAB: Neural Networks in Mathlab (Aston University)

<http://www.ncrg.aston.ac.uk/netlab/index.html>

Neural Networks Course (Thorsteinn S. Rögnvaldsson)

[http://www.hh.se/staff/denni/sls\\_course.html](http://www.hh.se/staff/denni/sls_course.html)

Numerical Analysis Group (NAG)

<http://www.nag.co.uk/>

Polynomial Networks by Dr. John F. Elder IV

<http://www.datamininglab.com/>

Scientific C++

<http://extreme.indiana.edu/~tveldhui/papers/techniques/>

The Neuroprose Archive

<ftp://ftp.cis.ohio-state.edu/pub/neuroprose/>

Web Sources for Neural Networks

<http://www-dsi.eng.unifi.it/neural/w3-sites.html#web>

# 5

## ALGORITMOS GENETICOS

### 5.1. FUNDAMENTOS

Los algoritmos genéticos son métodos de optimización de problemas basados en los mecanismos de la reproducción natural, los cuales son operaciones sobre el material genético de los organismos vivientes. Esta información genética está contenida en el núcleo de las células sexuales de las criaturas. La información genética contiene un cierto número de cromosomas, denominados genes, que se encargan de transportar la información genética. Así, por razones obvias, los algoritmos genéticos operan sobre estructuras que están organizadas de manera similar a los cromosomas.

Como ejemplo considere el siguiente problema de optimización:

Encontrar un  $x_0$  que pertenece a  $X$  tal que  $f$  es máximo en  $x_0$ , donde  $f : X \rightarrow \mathbb{R}$  es una función arbitraria de valor real, es decir  $f(x_0) = \sup f(x)$ .

Se denomina a  $X$  el espacio de búsqueda no codificado asociado al problema;  $f$  es denominada la función objetivo del problema de optimización. Considerando  $S$  un conjunto de strings y  $G$  una gramática, la cual describe la sintaxis de los strings contenidos en  $S$ , se define la función:

$$c: X \rightarrow S \\ x \mapsto c(x)$$

y la función:

$$c': S \rightarrow X \\ s \mapsto c'(s)$$

son denominadas funciones de codificación y decodificación respectivamente, si y solo si  $c$  es inyectiva y  $(c \circ c') = \text{id}_S$ ;  $S$  es denominado el espacio de búsqueda del siguiente problema alternativo codificado de optimización:

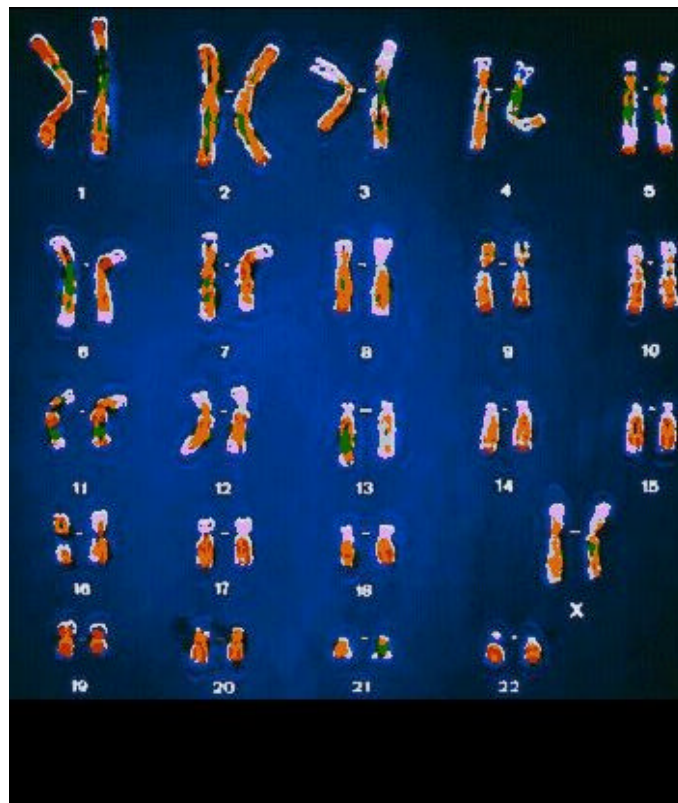
Encontrar un  $s_0$  que pertenece a  $S$  tal que  $f' := f \circ c'$  es máximo en  $s_0$ , es decir  $f'(s_0) = \sup [f'(s)]$

Un algoritmo genético es por consiguiente un método de optimización probabilístico que trata de resolver la tarea de optimización, planteada en el párrafo precedente, mediante la aplicación de operaciones genéticas.

### 5.2. TEORIA DE LA EVOLUCION

La teoría de la evolución<sup>46</sup>, fue descrita por Charles Darwin 20 años después de su viaje por las islas Galápagos en el *Beagle*, en el libro *Sobre el Origen de las Especies por medio de la Selección Natural*. Este libro fue bastante polémico en su tiempo, y en cualquier caso es una descripción incompleta de la evolución. La hipótesis de Darwin, presentada junto con Wallace, que llegó a las mismas conclusiones independientemente, es que pequeños cambios heredables en los seres vivos y la selección son los dos hechos que provocan el cambio en la Naturaleza y la generación de nuevas especies. Pero Darwin desconocía cual es la base de la herencia, pensaba que los rasgos de un ser vivo eran como un fluido, y que los "fluidos" de los dos padres se mezclaban en la descendencia; esta hipótesis tenía el problema de que al cabo de cierto tiempo, una población tendría los mismos rasgos intermedios.

Fue Mendel quien descubrió que los caracteres se heredaban de forma discreta, y que se tomaban del padre o de la madre, dependiendo de su carácter dominante o recesivo. A estos caracteres que podían tomar diferentes valores se les llamaron *genes*, y a los valores que podían tomar, *alelos*. En realidad, las teorías de Mendel, que trabajó en total aislamiento, se olvidaron y no se volvieron a redes cubrir hasta principios del siglo XX. Además, hasta 1930 el genetista inglés Robert Aylmer no relacionó ambas teorías, demostrando que los genes mendelianos eran los que proporcionaban el mecanismo necesario para la evolución.



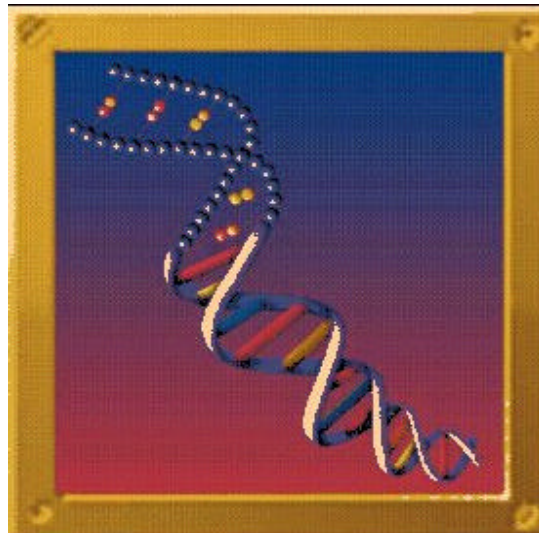
**Fig. 5.1. Cromosoma biológico**

Más o menos por la misma época, el biólogo alemán Walther Flemming describió los cromosomas biológicos, descritos en la Fig. 5.1., como ciertos filamentos en los que se agregaba

<sup>46</sup> Más que teoría es una serie de hechos probados.

la cromatina del núcleo celular durante la división; más adelante se descubrió que las células de cada especie viviente tenían un número fijo y característico de cromosomas.

Y no fue hasta los años 50, cuando Watson y Crick descubrieron que la base molecular de los genes está en el ADN, ácido desoxiribonucleico, vista en la Fig. 5.2. Los cromosomas están compuestos de ADN, y por tanto los genes están en los cromosomas.



**Fig. 5.2. Ácido Desoxiribonucleico**

La macromolécula de ADN está compuesta por bases *púricas* y *pirimídicas*, la adenina, citosina, guanina y timina. La combinación y la secuencia de estas bases forma el **código genético**, único para cada ser vivo. Grupos de 3 bases forman un *codon*, y cada codon codifica un aminoácido; el código genético codifica todas las proteínas que forman parte de un ser vivo. Mientras que al código genético se le llama *genotipo*, al cuerpo que construyen esas proteínas, modificado por la presión ambiental, la historia vital, y otros mecanismos dentro del cromosoma, se llama *fenotipo*.

No toda la cadena de ADN codifica proteínas, es decir, no todos son *genes*; las zonas que codifican proteínas se llaman *intrones*, las zonas que no lo hacen, *exones*. La cantidad de *ADN basura* aumenta desde los seres vivos más simples, como las bacterias, donde no hay nada, hasta los seres humanos, donde gran cantidad del ADN no codifica. Un gen comienza con el sitio 3' o *acceptor* y termina con el sitio 5' o *donante*. Proyectos como el del Genoma Humano tratan de identificar cuáles son estos genes, sus posiciones, y sus posibles alteraciones, que habitualmente conducen a enfermedades.

Todos estos hechos forman hoy en día la teoría del neo-darwinismo, que afirma que la historia de la mayoría de la vida está causada por una serie de procesos que actúan en y dentro de las poblaciones: reproducción, mutación, competencia y selección. La evolución se puede definir entonces como cambios en el *pool* o conjunto genético de una población.

### **5.3. COMPARACIÓN CON LO CONVENCIONAL**

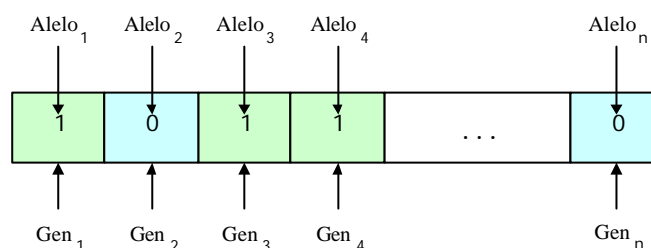
La tabla 5.1 muestra las analogías entre la evolución natural y el paradigma de los algoritmos genéticos.



| EVOLUCION NATURAL | ALGORITMO GENETICO           |
|-------------------|------------------------------|
| Genotipo          | string codificado            |
| Fenotipo          | punto no codificado          |
| Cromosoma         | string                       |
| Gen               | posición en string           |
| Alelo             | valor en cierta posición     |
| Adaptación        | valor de la función objetivo |

**Tabla 5.1. Analogías de la evolución natural y los algoritmos genéticos**

La representación gráfica de un cromosoma idealizado puede ser visto en la Fig.5.3.. Por otra parte, un algoritmo genético puede ser visto solamente como un modelo simplificado del proceso de evolución sin embargo, la diferencia más importante es que la "adaptabilidad" en el mundo real no puede expresada por un valor simple. La adaptabilidad en el mundo real es un vector formado por componentes tales como: inteligencia, crecimiento o fertilidad.



**Fig. 5.3. Estructura de un cromosoma**

Se ha mencionado que un algoritmo genético trata de resolver un problema de optimización, codificado o transformado, en reemplazo del problema real. La segunda diferencia significativa entre los métodos convencionales y los algoritmos genéticos es que los mismos no operan sobre puntos simples, sino sobre conjuntos completos de puntos<sup>47</sup>.

## 5.4. ALGORITMO GENETICO SIMPLE

John Holland<sup>48</sup> desde pequeño, se preguntaba cómo logra la naturaleza, crear seres cada vez más perfectos<sup>49</sup>. Lo curioso era que todo se lleva a cabo sobre la base de interacciones locales entre individuos, y entre estos y lo que les rodea. No sabía la respuesta, pero tenía una cierta idea de como hallarla: tratando de hacer pequeños modelos de la naturaleza, que tuvieran alguna de sus características, y ver cómo funcionaban, para luego extrapolar sus conclusiones a la totalidad. De hecho, ya de pequeño hacía simulaciones de batallas célebres con todos sus elementos: copiaba mapas y los cubría luego de pequeños ejércitos que se enfrentaban entre sí.

En los años 50 entró en contacto con las primeras computadoras, donde pudo desarrollar algunas de sus ideas, aunque no se encontró con un ambiente intelectual fértil para propagarlas. Fue a principios de los 60, en la Universidad de Michigan en Ann Arbor, donde, al interior del grupo *Logic of Computers*, sus ideas comenzaron a desarrollarse y a proporcionar frutos. Además,

<sup>47</sup> Referido como una población o generación de strings.

<sup>48</sup> En la historia de los algoritmos genéticos es considerado su principal representante.

<sup>49</sup> Aunque, como se ha visto, esto no es totalmente cierto, o en todo caso depende de qué entienda uno por *perfecto*.

leyendo un libro escrito por un biólogo evolucionista, R. A. Fisher, titulado *La teoría genética de la selección natural*, comenzó a descubrir los medios de llevar a cabo sus propósitos de comprensión de la naturaleza. De ese libro aprendió que la evolución era una forma de adaptación más potente que el simple aprendizaje, y tomó la decisión de aplicar estas ideas para desarrollar programas bien adaptados para un fin determinado.

En esa universidad, Holland impartía un curso titulado *Teoría de sistemas adaptativos*. Dentro de este curso, y con una participación activa por parte de sus estudiantes, fue donde se crearon las ideas que más tarde se convertirían en los algoritmos genéticos.

Por tanto, cuando Holland se enfrentó a los algoritmos genéticos, los objetivos de su investigación fueron dos:

- a) imitar los procesos adaptativos de los sistemas naturales, y
- b) diseñar sistemas artificiales (normalmente programas) que retengan los mecanismos importantes de los sistemas naturales.

Unos 15 años más adelante, David Goldberg, actual delfín de los algoritmos genéticos, conoció a Holland, y se convirtió en su estudiante. Golberg era un ingeniero industrial trabajando en diseño de *pipelines*, y fue uno de los primeros que trató de aplicar los algoritmos genéticos a problemas industriales. Aunque Holland trató de disuadirle, porque pensaba que el problema era excesivamente complicado como para aplicarle algoritmos genéticos, Goldberg consiguió lo que quería, escribiendo un algoritmo genético en un ordenador personal Apple II. Estas y otras aplicaciones creadas por estudiantes de Holland convirtieron a los algoritmos genéticos en un campo con base suficientemente aceptada para celebrar la primera conferencia en 1985, ICGA '85. Tal conferencia se sigue celebrando bianualmente.

#### 5.4.1. Anatomía

Los *algoritmos genéticos* son métodos sistemáticos para la resolución de problemas de búsqueda y optimización que aplican a estos los mismos métodos de la evolución biológica: selección basada en la población, reproducción sexual y mutación.

Los algoritmos genéticos son métodos de optimización, que tratan de resolver un conjunto de problemas que se ha contemplado anteriormente, es decir, hallar  $(x_i, \dots, x_n)$  tales que  $F(x_i, \dots, x_n)$  sea máximo. En un algoritmo genético, tras parametrizar el problema en una serie de variables,  $(x_i, \dots, x_n)$  se codifican en un cromosoma. Todas los operadores utilizados por un algoritmo genético se aplicarán sobre estos cromosomas, o sobre poblaciones de ellos. En el algoritmo genético va implícito el método para resolver el problema; son solo parámetros de tal método los que están codificados, a diferencia de otros algoritmos evolutivos como la programación genética. Hay que tener en cuenta que un algoritmo genético es independiente del problema, lo cual lo hace un algoritmo *robusto*, por ser útil para cualquier problema, pero a la vez *débil*, pues no está especializado en ninguno.

Las soluciones codificadas en un cromosoma *compiten* para ver cuál constituye la mejor solución<sup>50</sup>. El *ambiente*, constituido por las otras camaradas soluciones, ejercerá una presión selectiva sobre la población, de forma que sólo los mejor adaptados<sup>51</sup> sobrevivan o transmitan su

---

<sup>50</sup> Aunque no necesariamente la mejor de todas las soluciones posibles.

<sup>51</sup> Entiéndase como aquellos que resuelvan mejor el problema.

legado de material genético a las siguientes generaciones, igual que en la evolución de las especies. La diversidad genética se introduce mediante mutaciones y reproducción sexual.

En la Naturaleza lo único que hay que optimizar es la supervivencia, y eso significa a su vez maximizar diversos factores y minimizar otros. Un algoritmo genético, sin embargo, se usará para optimizar habitualmente sólo una función, no diversas funciones relacionadas entre sí simultáneamente. Este tipo de optimización, denominada optimización multimodal, también se suele abordar con un algoritmo genético especializado.

Por lo tanto, un algoritmo genético consiste en lo siguiente: hallar los parámetros de los cuales depende el problema, codificarlos en un cromosoma, y luego aplicar los métodos de la evolución: selección y reproducción sexual con intercambio de información y alteraciones que generan diversidad.

### 5.3.2. Codificación de Variables

Los algoritmos genéticos requieren que el conjunto se codifique en un *cromosoma*. Cada cromosoma tiene varios genes, que corresponden a sendos parámetros del problema. Para poder trabajar con estos genes en una computadora, es necesario codificarlos en una *cadena*, es decir, una cadena de símbolos (números o letras) que generalmente están compuestas de ceros y unos.

El número de bits usado para cada parámetro dependerá de la precisión que se quiera en el mismo o del número de opciones posibles (alelos) que tenga ese parámetro. Hay otras codificaciones posibles, usando alfabetos de diferente cardinalidad; sin embargo, uno de los resultados fundamentales en la teoría de algoritmos genéticos, el *teorema de los esquemas*, afirma que la codificación óptima, es decir, aquella sobre la que los algoritmos genéticos funcionan mejor, es aquella que tiene un alfabeto de cardinalidad 2.

Aquí se está codificando cada parámetro como un número entero de  $n$  bits. En realidad, se puede utilizar cualquier otra representación interna: bcd, código *Gray* y codificación en forma de números reales, por ejemplo.

La mayoría de las veces, una codificación correcta es la clave de una buena resolución del problema. Generalmente, la regla heurística que se utiliza es la llamada *regla de los bloques de construcción*, es decir, parámetros relacionados entre sí deben de estar cercanos en el cromosoma.

En todo caso, se puede ser bastante creativo con la codificación del problema, teniendo siempre en cuenta la regla anterior. Esto puede llevar a usar cromosomas bidimensionales, o tridimensionales, o con relaciones entre genes que no sean puramente lineales de vecindad. En algunos casos, cuando no se conoce de antemano el número de variables del problema, caben dos opciones: codificar también el número de variables, fijando un número máximo, o bien, lo cual es mucho más natural, crear un cromosoma que pueda variar de longitud. Para ello, claro está, se necesitan operadores genéticos que alteren la longitud.

Normalmente, la codificación es estática, pero en casos de optimización numérica, el número de bits dedicados a codificar un parámetro puede variar, o incluso lo que representen los bits dedicados a codificar cada parámetro. Algunos paquetes de algoritmos genéticos adaptan automáticamente la codificación según convergen los bits menos significativos de una solución.

## 5.5. PROCEDIMIENTO GENERAL

La forma general de un algoritmo genético está constituida por las líneas asociadas al siguiente algoritmo:

```
Algoritmo 1
   $t := 0$  ;
  Calcular  $B_0$  ;
  WHILE la condición de salida no se cumpla DO
    Begin
       $B_{t+1} := ?(B_t)$  ;
       $t := t+1$ 
    End
```

? es denominado operador de transición probabilístico que se encarga de calcular la siguiente generación  $B_{t+1}$  partiendo de la generación anterior  $B_t$  y tomando en cuenta la adaptabilidad de sus componentes.

Normalmente ? es una composición de varios operadores probabilísticos. Comparado con los métodos tradicionales de optimización, tales como los métodos de Newton o del gradiente descendente se puede establecer las siguientes diferencias significativas:

- Los algoritmos genéticos manejan versiones codificadas asociadas a los parámetros pertenecientes a los problemas, en reemplazo de los parámetros originales de los mismos.
- Mientras que casi todos los métodos convencionales se encargan de realizar la búsqueda de un punto simple, los algoritmos genéticos siempre operan sobre una población completa de puntos (strings). Esto contribuye a que los mismos sean robustos, reconociéndose el hecho de que proporcionan la opción de tomar el óptimo global, reduciendo el riesgo de quedar atrapado en un punto estacionario local.
- Los algoritmos genéticos normales no utilizan ninguna información auxiliar acerca del valor de la función objetivo tales como las derivadas.
- Los algoritmos genéticos utilizan operadores de transición probabilísticos, los métodos convencionales para la optimización continúan aplicando exclusivamente operadores de transición determinísticos.

Desde este punto de vista, está claro que los algoritmos genéticos son métodos robustos, los cuales pueden, debido a su generalidad, ser aplicados a un rango amplio de diferentes problemas de optimización. Por otra parte, los algoritmos genéticos pueden ser de rendimiento débil debido a que desagregan toda la información que puede ser útil.

En el resto del texto, se considera la clase más simple de un algoritmo genético la cual es ampliamente utilizada en la solución de problemas de optimización continuos como también en la optimización discreta. Esto está caracterizado como un algoritmo genético simple que opera sobre strings de tamaño fijo  $n$  (no necesariamente binario).

## 5.6. ALGORITMO GENETICO CON OBJETOS DE TAMAÑO FIJO

Un Algoritmo Genético simple, el cual opera sobre strings de tamaño fijo, incorpora los siguientes métodos en su operador de transición (?):

- a) **Selección:** Los individuos con adaptabilidad alta son favorecidos en el proceso de reproducción. De manera concreta, los individuos con adaptabilidad alta tienen una gran probabilidad de sobrevivir y de reproducirse por cuenta propia.
- b) **Emparejamiento**<sup>52</sup>: Se denomina así al método de mezclar la información genética de dos individuos; este mecanismo ha contribuido bastante a la adaptación rápida de las especies que se reproducen sexualmente.
- c) **Mutación:** En la evolución real el material genético puede ser cambiado randomicamente por reproducción errónea u otras deformaciones de los genes (ej. radiaciones gama). En los Algoritmos Genéticos la mutación puede ser realizada como una deformación randómica de los strings con una cierta probabilidad.

#### Algoritmo 2

```

Sea  $m$  el tamaño de la población
 $t := 0$ 
Calcular la población inicial  $B = (b_{1,0}, \dots, b_{m,0})$ ;
WHILE no se cumpla la condición de parada DO
Begin
  FOR  $i = 1$  TO  $m$  DO
  Begin
    Seleccionar un elemento  $g$  de  $B_t$ ;
    IF  $\text{Random}[0,1] \leq P_c$  THEN
      Aparear  $g$  con un elemento individual de  $B$ ;
      Mutar  $g$ 
       $b_{i,t+1} := g$ 
    End
  End
   $t := t + 1$ 
End

```

Este algoritmo contiene los mecanismos de selección, mutación y apareamiento.

#### 5.6.1. Selección

La selección es el componente que guía el algoritmo hacia la solución de un problema. Puede ser un mecanismo que favorece a los individuos o elementos más aptos con desventajas para los menos aptos. Puede ser una operación determinista, pero, en la mayoría de las implementaciones, tiene componentes aleatorios.

Una variante, la cual es muy popular, constituye el siguiente esquema, donde la probabilidad de elegir un cierto elemento es proporcional a su capacidad de adaptación. Esto puede ser considerado como un experimento randómico con:

$$P[b_{j,t} \text{ sea seleccionada}] := f(b_{j,t}) / \sum f(b_{k,t}) \quad (\text{Ec. 4.1.1})$$

---

<sup>52</sup> En otra bibliografía es referida como apareamiento.

Por supuesto esta fórmula solamente tiene sentido si todos los valores adaptables son positivos. Si este no es el caso, se puede aplicar una transformación apropiada. Este experimento aleatorio es, en algún sentido, un juego de ruleta en aquellos casos en los cuales la probabilidad para seleccionar un cierto individuo o elemento depende de su adaptabilidad.

Puede ser necesario utilizar una función de adaptabilidad transformada en la selección. Luego las probabilidades pueden ser expresadas como:

$$P[b_{j,t} \text{ sea seleccionada}] := f(b_{j,t}) / \sum_k f(b_{k,t}) \quad (\text{Ec. 4.1.2})$$

donde  $f: \mathbb{R} \rightarrow \mathbb{R}^+$  es una función no decreciente. La función  $f$  puede también ser utilizada para acelerar la acumulación de individuos o elementos con gran capacidad de adaptación. Considere por ejemplo que  $f(x) = x^p$  con  $p > 1$ , en este caso llega a ser, dependiendo de  $p$ , menos probable la selección de strings menos adaptables.

La formulación algorítmica del esquema de selección (Ec. 4.1.1) puede ser escrito como sigue (de manera análoga puede escribirse el caso asociado a la (Ec. 4.1.2))

#### Algoritmo 3

```

 $x := \text{Random}[0,1];$ 
 $i := 1$ 
WHILE  $i < m$  &  $x < f(b_{j,i}) / \sum_k f(b_{k,i})$  DO
     $i := i + 1;$ 
seleccionar  $b_{i,t};$ 

```

De manera concreta esta selección matemática es un equivalente a la selección natural en la cual los individuos más adaptados tienen mejores opciones de sobrevivir. Este método es denominado también selección proporcional.

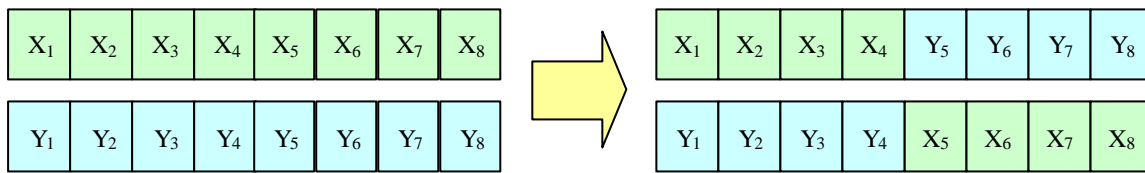
### 5.6.2. Apareamiento

En la reproducción sexual, el material genético de dos padres es mezclado durante el proceso de meiosis. Este mecanismo es una herramienta bastante poderosa para introducir nuevo material genético, el cual es llevado a cabo con la probabilidad más alta, o con la capacidad de adaptación más alta que la de sus padres o predecesores. Muchas investigaciones demostraron que el apareamiento es la razón por la cual las especies que se reproducen sexualmente se hayan adaptado más rápido y de mejor forma que las especies que se reproducen asexualmente.

Básicamente, el apareamiento es el intercambio de genes entre los cromosomas de los dos padres. En la meiosis<sup>53</sup> real esto es, de manera simple, el intercambio de partes de los cromosomas. En la investigación sobre algoritmos genéticos con objetos de tamaño fijo el apareamiento puede, en el caso más simple, ser realizado como el corte de dos strings en una posición aleatoriamente elegida y el intercambio de las partes. Este proceso es visualizado en la Fig. 5.4.

---

<sup>53</sup> Conocida también como meiosis, se refiere a la división de la célula viva, en la que las células hijas tienen la mitad de cromosomas que la célula madre, y constituye el estadio esencial de la formación de las células.



**Fig. 5.4. Apareamiento en el gen 4**

Se denomina a este proceso apareamiento en un punto, y responde al siguiente algoritmo:

```

Algoritmo 4
  pos := Random{1,...,n-1} ;
  FOR i := 1 TO pos DO
    Begin
      Child1[i] := Parent1[i] ;
      Child2[i] := Parent2[i] ;
    End
  FOR i := pos + 1 TO n DO
    Begin
      Child1[i] := Parent2[i] ;
      Child2[i] := Parent1[i] ;
    End

```

El apareamiento en un punto es un método simple y frecuentemente utilizado para los algoritmos genéticos que operan sobre strings binarios. Para otros problemas, con diferentes códigos, pero también para el caso de strings binarios, existen diferentes técnicas de apareamiento, tales como:

- Apareamiento de N-puntos** : en lugar de solo un punto N puntos de corte son elegidos de manera aleatoria. Cada segunda sección es intercambiada.
- Apareamiento Segmentado**: similar al apareamiento de N-puntos con la diferencia que el numero de puntos de corte puede variar.
- Apareamiento Uniforme**: Para cada posición se decide de manera aleatoria como intercambiar las posiciones.
- Apareamiento con permutación**: Primero es aplicada una permutación elegida de manera aleatoria a los dos padres, luego es aplicado el apareamiento de N-puntos a los parientes permutados, finalmente, los hijos permutados son transformados con la permutación inversa.

### 5.6.3. Mutación

El ultimo ingrediente del algoritmo genético simple es la mutación. En el mundo real, la mutación hace referencia a la deformación randómica de la información genética de un individuo mediante una radiación radioactiva u otras influencias similares. En la reproducción real es la probabilidad de que un cierto gen mutado sea casi igual para todos los genes.



Fig. 5.5. Mutación en los genes impares

De esta manera se puede utilizar la siguiente técnica de mutación para un string binario S:

*Algoritmo 5*

```
FOR i := 1 TO n DO
  IF Random[0,1] <  $P_M$  THEN
    invert S[i] ;
```

Nuevamente, de manera similar al apareamiento, la elección de la técnica de mutación apropiada depende de la codificación misma del problema. A continuación se menciona algunas alternativas:

- a) **Inversión de bits simples**: Con probabilidad  $P_M$  se niega un bit elegido de manera aleatoria.
- b) **Inversión**: Con probabilidad  $P_M$  el string completo es invertido bit a bit.
- c) **Selección Aleatoria**: Con probabilidad  $P_M$  el string es reemplazado por otro elegido de manera aleatoria.

## 5.7. ALGORITMO GENETICO GENERAL

En base a los operadores genéticos descritos en la anterior sección, se muestra a continuación la estructura general de un algoritmo genético en términos de su estructura básica, la modificación para su implantación como procedimiento computacional depende del análisis y diseño de la solución a un problema determinado.

*Algoritmo 6*

```
t := 0
Calcular la población inicial B=(b1,0,... , bm,0) ;
WHILE no se cumpla la condición de parada DO
  Begin
    FOR i=1 TO m DO
      Begin
        (* selección proporcional *)
        x := Random[0,1];
        l := 1
        WHILE l < m & x < (f(bj,t)/ (f(bj,t) DO
          l := l+1;
        g := bl,t ;
        (* apareamiento en un punto *)
        IF Random[0,1] <= Pc THEN
          Begin
            Parent1 := g ;
            Parent2 := bRandom{1,...,m},t ;
```



```

pos := Random{1,n-1};
FOR i := 1 TO pos DO
Begin
    Child1[i] := Parent1[i];
    Child2[i] := Parent2[i];
End
FOR i := pos+1 TO n DO
Begin
    Child1[i] := Parent2[i];
    Child2[i] := Parent1[i];
End
IF Random[0,1] < 1/2 THEN
    g := Child1;
ELSE
    g := Child2;
End
(* mutación *)
FOR i := 1 TO n DO
    IF Random[0,1] < pM THEN
        invert g[i];
    bi,t+1 := g
end
t := t+1;
End

```

En este algoritmo, como es usual,  $n$  es el tamaño de los strings y  $m$  es el tamaño de la población.

## 5.8. PROBLEMA DEL AGENTE VIAJERO

Considere el ejemplo de un viajante que parte de una ciudad "origen" y tiene que pasar por un número determinado de ciudades, por decir 20. La ruta a escoger resulta diferente según el orden en que se visiten las ciudades. Si se van visitando ciudades en un orden en que cada una está lo más cerca posible de la próxima entonces el viaje resultará más rentable que si se visita primero una ciudad que esté muy lejos de la ciudad "origen", luego se vuelve a otra cerca de ésta, luego se vuelve a ir al otro extremo etc. Por eso resulta interesante visitar las ciudades en un orden en que la distancia total sea la más corta posible.

Ahora se intenta desarrollar un programa que consista en lo siguiente: dadas las coordenadas (x,y) de las ciudades origen y de las demás ciudades a visitar, que el programa se encargue de evaluar todas las maneras posibles de efectuar el recorrido y vaya calculando sus respectivas distancias una a una, proporcionando como resultado el recorrido más pequeño.

Pero esto sin duda tiene un problema. En el ejemplo con 20 ciudades el número de maneras diferentes de recorrerlas es  $20!$ , este es un número astronómico aún para las computadoras contemporáneas. Si se diseña un programa que intente calcular la solución exacta por evaluación de todas las rutas posibles, el programa no será rentable en tiempo. Si el número de ciudades en lugar de ser 20 fuese 40 el número de maneras distintas de recorrerlas resulta del orden de  $8 \times 10^{47}$ . Es decir, se descarta este método por ser demasiado costoso en tiempo, incluso irrealizable en muchos casos.

Los algoritmos genéticos permiten encontrar una solución aproximada a este problema, una solución que puede ser muy cercana a la mejor. Se identifica cada ciudad con un número secuencial (1, 2, 3... 20) y se genera 20 ristas<sup>54</sup> con los 20 números de las ciudades ordenados de manera aleatoria se generan 20 rutas diferentes generadas al azar. A continuación se reproducen esas ristas; esto significa que a partir de cada una se generan 10 más, a la primera de ellas se la nombra como la rista “padre” y en cada una de las otras 9 se permutan dos ciudades al azar. Por ejemplo, si una de las 20 ristas generadas de manera aleatoria contiene un orden concreto de visita de las ciudades: 20 3 4 1 15 16 18 2 5 6 7 11 12 8 9 10 19 17 14 13, entonces la primera de las 10 ristas que se llamarán “hijas” será ella misma y la segunda será la misma, excepto que se habrá permutado 2 elementos (por ejemplo la posición del 20 por la del 4), obteniendo así una rista diferente que implica una nueva manera de recorrer las ciudades. Y las demás análogamente a esta segunda.

Se tenía 20 ristas “padres”, de cada una se ha generado 10 ristas hijas diferentes, por tanto se tienen 200 ristas diferentes. Ahora es posible calcular la distancia que supondría recorrer las ciudades en cada una de estas 200 rutas ordenadas. Con este método se hará solamente 200 operaciones frente a las 2.432.902.008.176.640.000 que tendrían que hacerse si se evaluara todas las posibilidades. De estas 200 rutas se opta por las 20 mejores, esto es, con las ristas que generaban las 20 maneras más rentables (en distancia) para realizar el viaje. Con estas 20 se repetirá el proceso de generar otras 200, de ellas se calcula de nuevo sus 20 mejores y se puede repetir toda esta operación (es lo que se llama el algoritmo genético) todas las veces que se quiera. Se puede comprobar cómo en cada “reproducción” y selección de las 20 mejores ristas van apareciendo mejores resultados que en la anterior reproducción. Al terminar del programa, de esas 20 mejores ristas solamente se elige la mejor, considerándola como resultado final. Ésa es una buena aproximación a la solución.

Pero, ¿cómo saber si esa aproximación es buena, si lo único que se ha realizado es evaluar un cierto número de ristas que es mucho menor al número de maneras posibles de visitar todas las ciudades? Y si se repite el algoritmo, por decir 50 veces, ¿no sería menos complicado considerar  $50 \times 200 = 10000$  ristas y evaluar las distancias de cada una de ellas prescindiendo de los operadores genéticos de mutación y reproducción?

Las dos preguntas son ampliamente razonables. Es molesto considerar, si se repite el algoritmo genético 50 veces y por lo tanto analizar  $50 \times 200 = 10000$  ristas aleatorias, que la mejor solución no pueda acercarse a la manera de recorrer las ciudades por el camino más corto, ya que el número de maneras distintas es 2.432.902.008.176.640.000 y solamente se ha considerado 10.000 de ellas. Por otra parte, también se hace la pregunta acerca de la evaluación “genética” de esas 10.000 ristas y la razón por la que no se considera una evaluación secuencial, cosa que ahorraría tiempo.

Continuando con las consideraciones, al igual que un padre se parece a su hijo, las ristas generadas se parecen a las ristas “padres” que se usaron para generarlas. Cuando se generaron las 20 primeras ristas, éstas contaban con una posibilidad de ser buenas o malas aleatoriamente, pero las ristas “hijas” que generan una rista buena tienen tendencia a ser en su mayoría buenas, y así, al ir seleccionando las mejores, es posible recorrer adecuadamente el espacio de las

---

<sup>54</sup> Hace referencia a una serie de cosas inmateriales que van o se suceden una tras otra. En es caso presente un orden secuencial.

soluciones buenas y no es lo mismo generar 10.000 ristas aleatoriamente y buscar el mejor resultado.

Con este ejemplo se ilustra los algoritmos genéticos. La eficiencia de éstos depende de la elección correcta, a priori desconocida, del número de iteraciones, ristas seleccionadas inicialmente, ristas hijas que genera cada rista padre, etc. Se puede decir que los algoritmos genéticos resuelven a diario problemas similares al del viajante.

## **5.9. PROGRAMACION GENETICA**

La segunda clase importante de algoritmos genéticos está comprendida bajo el termino de programación genética. Esto constituye un enfoque genético bastante importante para el problema de la inducción de programas en la inteligencia artificial. La inducción de programas es el método para enseñar a una computadora como resolver un problema sin que el mismo sea programado de manera explícita. Esto es, en algún sentido, un problema inverso, en lugar de decirle a una computadora como resolver de manera explícita un problema, se presenta el problema a la computadora y se le alienta para encontrar un programa que resuelva el problema.

La idea de aplicar los algoritmos genéticos al problema de la inducción de programas puede ser atribuida a J. Holland. Otra fuente importante de investigación fue realizada por J.R. Koza quien introdujo el término "*programación genética*".

Obviamente, la aplicación de los algoritmos genéticos a programas completos requiere mas o menos modificaciones significativas a la maquinaria genética discutida previamente. La más importante es la referida a los strings de tamaño fijo. Aunque es posible restringir el tamaño de un programa a un cierto valor, no se puede preservar la universalidad de un lenguaje de programación si se restringe el tamaño de los programas a un numero determinado de instrucciones. En lo que sigue se asume que la sintaxis de un lenguaje de programación esta proporcionada en la forma Backus-Naur (BNF) la cual describe la sintaxis de los lenguajes de manera recursiva. Un programa, escrito en dicho lenguaje, puede ser considerado como un árbol<sup>55</sup> que puede ser escrito como una lista anidada. El proceso de reescribir un programa como un árbol o una lista anidada puede ser realizado para cualquier lenguaje de programación utilizado de manera común tal como: C, FORTRAN, PASCAL o LISP. Es fácil ver que la implementación es menos dificultosa para lenguajes restringidos, los cuales solo permiten recursiones, tales como el LISP.

### **5.9.1. Elección del Lenguaje de Programación**

Es posible reescribir cada programa como una lista o un árbol, pero esta puede ser una tarea crucial para muchos de los lenguajes de programación procedimentales comunes. En muchas aplicaciones y tratamientos teóricos es utilizado el lenguaje LISP. Los programas LISP son listas anidadas fáciles en su manejo, debido a que no es necesaria su reescritura.

Otro aspecto importante a considerar es cual subconjunto del lenguaje se podría utilizar. Esta claro que no es útil en cada aplicación utilizar el conjunto completo de instrucciones que el lenguaje ofrece. Considere por ejemplo un problema en el que se tiene que aprender operaciones lógicas. Para tal caso se puede restringir solamente a las operaciones lógicas AND, OR y NOT o

---

<sup>55</sup> La estructura de este árbol no es necesariamente binaria.

a un subconjunto de las mismas. Bajo la suposición de que el problema puede ser resuelto dentro del subconjunto, una elección inteligente de dicho subconjunto puede incrementar el rendimiento de manera marcada. La razón simple es que, el espacio de búsqueda es bastante pequeño, consecuentemente esto incrementa la elección de una solución considerablemente buena.

### 5.9.2. Iniciación

En secciones anteriores no se prestó demasiada atención en la creación de la población inicial. Se asumía implícitamente que los individuos de la primera generación podían ser generados de manera aleatoria con respecto a una cierta distribución de probabilidad. La generación aleatoria de árboles o listas anidadas es una tarea más compleja.

Considere por ejemplo la BNF del siguiente lenguaje que puede ser utilizado para representar funciones lógicas ternarias:

```
Programa    := <expresión>;
<expresión> := "("<variable>")" /
              "("<unario><expresión>")" /
              "("<binario><expresión><expresión>")";
<variable>  := "x"/"y"/"z";
<unario>    := "NOT";
<binario>    := "AND"/"OR";
```

Obviamente, los elementos sintácticos de este lenguaje parecido al LISP son los paréntesis, las variables  $x$ ,  $y$  y  $z$  y los operadores NOT, AND y OR. Está claro que un procedimiento, el cual crea strings con entradas randómicas a partir de este conjunto de elementos sintácticos, no es una variante buena de un procedimiento de inicialización, a causa de que la probabilidad de creación de strings sintácticamente correctos es demasiado baja.

Una mejor alternativa podría constituir un procedimiento el cual está basado en el BNF del lenguaje mismo. Tal algoritmo puede ser establecido como sigue :

#### *Algoritmo BNFL*

*Empezar con la raíz de la sintaxis (en nuestro caso el Programa).*

*Seleccionar una alternativa de la expresión sintáctica actual de manera randómica.*

*Llenar la alternativa y aplicar el procedimiento de manera recursiva para todas las subexpresiones atómicas de la alternativa.*

*Fin BNFL*

Es comprensible, de manera intuitiva, que se deben incluir mecanismos que eviten el final de la recursión en este método. Una oportunidad común consiste en fijar una profundidad máxima y evitar alternativas no atómicas si esta profundidad es alcanzada o excedida. El siguiente ejemplo ilustra este método de manera mas clara para el caso del lenguaje definido.

Ejemplo:

El punto de inicio es, por supuesto, la expresión raíz<sup>56</sup>.

<expresión>

segunda alternativa

(<unario> <expresión> <expresión>)

<unario> es una expresión atómica con solo una alternativa

(NOT <expresión> <expresión>)

tercera alternativa

(NOT (<binario> <expresión> <expresión>) <expresión>

primera alternativa

(NOT (AND (<expresión> <expresión>) <expresión>

primera alternativa

(NOT (AND (<variable> <expresión>) <expresión>

segunda alternativa

(NOT (AND (y) <expresión>) <expresión>)

... y así sucesivamente

### 5.9.3. Programas de Apareamiento

La segunda cuestión, que difiere del caso asociado a los strings de manera notable, es el apareamiento sobre la operación. Por supuesto, las listas anidadas pueden ser consideradas como strings para los cuales un apareamiento estándar puede ser aplicado sobre la operación. El problema es nuevamente que la probabilidad de generar strings sintácticamente correctos mediante esta forma es muy pequeña. Por consiguiente, es necesario encontrar una alternativa la cual preserve la correctitud sintáctica.

Para este propósito, se utiliza la interpretación de los programas como árboles. Por supuesto, existen muchas interpretaciones posibles de esta clase, el enfoque que se discute es el propuesto por A. Geyer-Schulz. Un programa correcto puede derivarse de una gramática resaltada en al menos una forma (no necesariamente única). Esta derivación de árboles por si misma es utilizada luego como una representación. No es trivial proporcionar una formulación exacta del procedimiento para determinar la derivación de árboles para una expresión dada. A continuación se presenta un ejemplo simple el cual aclara los conceptos mencionados.

---

<sup>56</sup> En este ejemplo se subraya la evaluación que se realiza.

Ejemplo:

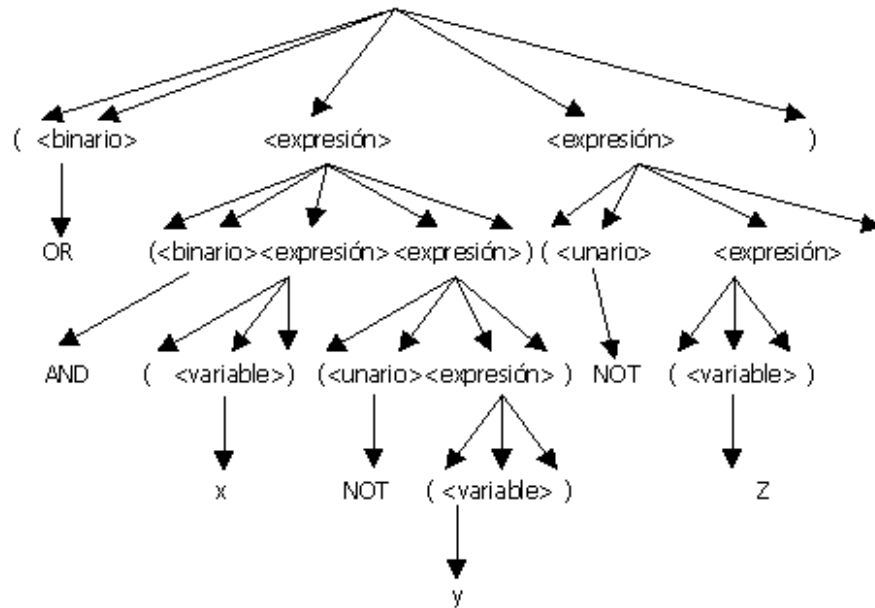


Fig. 5.6. Árbol de derivación para  $(OR(AND(x)(NOT(y)))(NOT(z)))$

En la Fig. 5.6. se muestra la derivación de árboles de la expresión:  $((x \text{ ? } ?y) \text{ ? } ?z)$ , que puede ser escrito como:  $(OR(AND(x)(NOT(y)))(NOT(z)))$ , en el lenguaje establecido en el anterior párrafo.

Puede considerarse fácilmente que cada subarbol corresponde a una subexpresión. Las raíces de estos subárboles son denominadas etiquetas. El método más común para aparear dos expresiones, las que pueden ser expresadas como árboles de derivación, consiste en intercambiar subárboles que empiezan con nodos iguales, es decir los cuales cuentan con etiquetas iguales. Esto garantiza que la descendencia a partir de los nodos intercambiados es correcta sintácticamente.

Ejemplo 2:

La figura adjunta muestra un ejemplo simple para aparear dos árboles de derivación. El resultado en forma de lista anidada es el siguiente:

$$\begin{array}{l} (AND(x)(NOT(z))) \\ (NOT(OR(y)(x))) \end{array} \quad \longrightarrow \quad \begin{array}{l} (AND(x)(OR(y)(x))) \\ (NOT(NOT(z))) \end{array}$$

#### 5.9.4. Mutación de Programas

Después de todo este trabajo preparatorio resulta comparativamente fácil proporcionar una técnica de mutación, de la manera usual, para los programas. El método más común es seleccionar aleatoriamente un subarbol del árbol de derivación y reemplazarlo por otro subarbol que fue generado aleatoriamente aplicando el mismo método discutido en conexión con el procedimiento de inicialización. Por supuesto, se debe prestar atención especial a la profundidad de este subarbol. Luego se garantiza que la correctitud sintáctica del programa no es dañada por la operación de mutación.

### 5.9.5. Función de Adaptación

Otra tarea no trivial en conexión con la programación genética es la definición de una función de adaptación apropiada la cual se encargue de medir cuan bien, un determinado programa, resuelve un problema. La solución de este problema depende fuertemente del problema mismo, una receta universal para definir la medida de adaptación apropiada no puede ser proporcionada de manera eficiente.

Una técnica utilizada comúnmente consiste en aplicar un programa a un numero finito de pruebas de entrada para las cuales la salida deseada es conocida. Por supuesto, esos casos deben ser seleccionados de manera clara de tal forma que sean realmente representativos. Luego, el número de casos para los cuales la salida correcta es obtenida puede ser tomada como una medida para la correctitud del programa. El siguiente ejemplo muestra un caso donde este método no es útil.

Ejemplo:

```
Función      := <expresión>;
<expresión> := "("<variable>")" /
              "("<constante>")" /
              "("<binario><expresión><expresión>")";
<variable>  := "x";
<constante> := <numero> "/" <numero> ;
<numero>    := "0" / ... / "9" / <numero> ;
<binario>   := "+" / "-" / "*" ;
```

Esta gramática describe un lenguaje de programación para representar polinomios con coeficientes racionales. La tarea a emprender es encontrar un polinomio de algún grado tal que corresponda a un conjunto de puntos representativos de prueba  $\{(x_1, y_1), \dots, (x_n, y_n)\}$  tan buenos como sea posible. Puesto que la probabilidad de que un polinomio elegido aleatoriamente corresponda al menos a un punto de manera exacta es aproximadamente 0, el número de emparejamientos exactos son usados como medida de adaptación. Una alternativa puede ser:

$$f(P) := \|(y_1 - P(x_1), \dots, y_n - P(x_n))\|$$

la cual puede ser minimizada, donde  $f$  es la función de adaptación del polinomio  $P$ , y  $\| \cdot \|$  es una norma arbitraria sobre  $\mathbb{R}^n$  (por ejemplo la norma euclideana).

### 5.10. APLICACIONES

En toda ejecución de un algoritmo genético hay que decidir con qué frecuencia se va a aplicar cada uno de los algoritmos genéticos; en algunos casos, como en la mutación o el apareamiento uniforme, se debe de añadir algún parámetro adicional, que indique con qué frecuencia se va a aplicar dentro de cada gen del cromosoma. La frecuencia de aplicación de cada operador estará en función del problema; teniendo en cuenta los efectos de cada operador, tendrá que aplicarse con cierta frecuencia o no. Generalmente, la mutación y otros operadores que generen diversidad se suele aplicar con poca frecuencia; la recombinación se suele aplicar con frecuencia alta.

En general, la frecuencia de los operadores no varía durante la ejecución del algoritmo, pero hay que tener en cuenta que cada operador es más efectivo en un momento de la ejecución. Por ejemplo, al principio, en la fase denominada de *exploración*, los más eficaces son la mutación y el apareamiento; posteriormente, cuando la población ha convergido en parte, el apareamiento no es útil, pues se está trabajando con individuos bastante similares, y es poca la información que se intercambia. Sin embargo, si se produce un estancamiento, la mutación tampoco es útil, pues está reduciendo el algoritmo genético a una búsqueda aleatoria; y hay que aplicar otros operadores. En todo caso, se pueden usar operadores especializados.

Una de las primeras aplicaciones de las técnicas evolutivas fue la reportada por Friedman, quien propuso la evolución de circuitos de control, similares a las actuales redes neuronales.

Lawrence Davis propone varios ejemplos de algoritmos evolutivos con representaciones y operadores especiales para resolver problemas del mundo real. Este cambio es útil en la mayoría de los casos, pero no es fácil generalizarlo para problemas similares.

- a) **Llaves aleatorias.** Propuestas por James C. Bean, consisten de una representación llamada "*Random Key Encoding*" la cual se utiliza para eliminar la necesidad de operadores especiales de apareamiento y mutación codificando las soluciones con números aleatorios los cuales serán las llaves para decodificar sus valores. La propuesta es interesante, pero los resultados obtenidos con ella no han sido muy exitosos.
- b) **GENOCOP.** GENetic algorithm for Numerical Optimization for CONstrained Problems, propuesto por Michalewicz asume puntos iniciales para comenzar la búsqueda y sólo admite restricciones lineales. Los resultados obtenidos con este algoritmo genético han sido exitosos.
- c) **Algoritmo Genético Consistente en Restricciones.** Propuesto por Kowalczyk para acotar el espacio de búsqueda previniendo la instanciación de variables que no satisfacen a las restricciones. Por lo tanto, sólo se producen soluciones factibles.
- d) **Localización de Límites de Región Factible.** Su motivación radica en que en optimización no lineal se suelen tener restricciones activas en el óptimo global, razón sugerente para explorar el límite entre la zona factible y la no-factible del espacio de búsqueda.
- e) **Decodificadores.** Consiste en que un cromosoma proporcione la manera de construir una solución factible. Cada decodificador impone una relación entre una solución factible y una solución decodificada. Propuesta por Koziel y Michalewicz.

## Ejercicios # 5

1. Encontrar un  $x_0$  que pertenece a  $X$  tal que  $f$  es mínimo en  $x_0$ , donde  $f: X \rightarrow \mathbb{R}$  es una función arbitraria de valor real, es decir  $f(x_0) = \inf_{x \in X} f(x)$ .
2. Escribir los pasos que representen un algoritmo genético general.
3. Construir un algoritmo genético para resolver el juego del tic tac toe (tres en raya).
4. Modificar el algoritmo genético de las ciudades imponiendo bloqueos entre algunas de ellas (por ejemplo que no se pueda ir directamente de la ciudad 8 a la 5 ni de la 2 a la 12 porque sus carreteras suelen estar muy frecuentadas y se tarda más).



5. Crear un algoritmo genético que reparta los asientos de los invitados de una boda según sus preferencias, es decir cada invitado dará una lista de personas al lado de las que querría estar sentado y otra lista con las personas que no quiere que se sienten a su lado. Hay que conseguir que se cumplan en su mayoría sus preferencias
6. Construir un algoritmo genético que se utilice para el juego de las damas en un tablero de 8x8 entradas.
7. Se tiene una lista de profesores, cada uno puede impartir una o varias asignaturas, y una lista de clases con el numero de horas que debe recibir cada asignatura y se pide diseñar un algoritmo genético que reparta el horario de las asignaturas y asigne a ellas profesores de manera que no se solapen clases del mismo profesor. El problema se puede complicar si se impone restricciones como que un profesor no puede dar clase a una cierta hora, etc.
8. Explicar las razones por las cuales se utiliza la programación genética, en lugar de la programación tradicional, y en que casos es útil contar con este paradigma.
9. Construir un árbol de derivación para la siguiente expresión  $(a \ ? \ ? \ b) \ ? \ ? \ c$ .
10. Desarrollar el árbol de derivación para la expresión  $((? \ x \ ? \ y) \ ? \ ? \ z)$  y luego aparearlo con el árbol  $(x \ ? \ ? \ y)$ .

## Bibliografía

- Daniels, R.L. & Carrillo, J.E. Beta-Robust scheduling for single-machine systems with uncertain processing times, *IEEE Transactions*, Volume 29, pages 977-985, 1997.
- Davis, L. Job Shop Scheduling with Genetic Algorithms, *Proceedings of the First International Conference on Genetic Algorithms*, 1986.
- Gideon, W. *A Tutorial in Stochastic Scheduling*. Editors: Chretienne, P. and Coffman, E.G. Jr. and Lenstra, J.K. and Liu, Z., *Scheduling: Theory and its applications*, chapter 3, pages 33-64, John Wiley & Sons, 1995.
- Goldberg, D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, MA, 1989.
- Goldberg, D.E. Genetic and Evolutionary Algorithms come of Age. *Communications of the ACM*, Volume 37, No. 3, pages 113-119, 1994.
- Holland, J.H. *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge MA, 1975.
- James C. Bean. Genetics and random keys for sequencing and optimization. *ORSA Journal on Computing*, 6(2):154-160, 1994.
- Jia, C. & Tu, F. Stochastic Single Machine Scheduling Problem with V-Shaped or Lambda-shaped Optimal Sequences. *Proceedings of the 35th Conference on Decision and Control*, 1996.
- Jones, A. & Rabelo, L.C. *Survey of Job Shop Scheduling Techniques*, NISTIR, National Institute of Standards and Technology, Gaithersburg, MD, 1998 (on-line publication).
- Kowalczyk, R. Constraint Consistent Genetic Algorithms. In *Proceedings of the 1997 IEEE Conference on Evolutionary Computation*, pages 343-348, Indianapolis, USA, April 1997. IEEE.
- Koziel S. and Z. Michalewicz. A Decoder-based Evolutionary Algorithm for Constrained Parameter Optimization Problems. In T. Back, A. E. Eiben, M. Schoenauer, and H.-P. Schwefel, editors, *Proceedings of the 5th Parallel Problem Solving from Nature (PPSN V)*, pages 231-240, Amsterdam, September 1998. Springer-Verlag.
- Lawrence D. editor. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York, 1991.
- Lin, Shyh-Chang, Goodman, E.D. and Punch, W.F.III. A Genetic Algorithm Approach to Dynamic Job Shop Scheduling Problem. *Proceedings of the Seventh International Conference on Genetic Algorithms*, 1997.
- Matfeld, D. *Evolutionary Search and the Job Shop*. Heidelberg, Physica Verlag, 1995.

Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, third edition, 1996.

Montana, D., Brinn, M., Moore, S. & Bidwell, G. Genetic Algorithms for Complex, Real-Time Scheduling, *IEEE International Conference on Systems, Man and Cybernetics*, Volume 3, pages 2213-2218, 1998.

Morton, T.E. & Pentico, D.W. *Heuristic Scheduling Systems with Applications to Production Systems and Project Management*. Wiley series in engineering and technology management, Wiley-Interscience, 1993.

Pinedo, M. *Scheduling: Theory, Algorithms and Systems*. Englewood Cliffs, Prentice Hall, N.J., 1995.

## **Referencias Electrónicas**

The Hitch-Hiker's Guide to Evolutionary Computation  
<ftp://ftp.cerias.purdue.edu/pub/doc/EC/Welcome.html>

Guía resumen de computación evolutiva  
<http://alife.santafe.edu/~joke/encore/www/>

Programación por expresión genética  
<http://www.gene-expression-programming.com/>

Gramáticas evolutivas  
<http://www.grammatical-evolution.org/>

Algoritmos genéticos aplicados a clustering de KDD  
<http://www.cs.uwa.edu.au/robvis/theses.html#CO>

Algoritmos genéticos  
<http://www-cs.us.es/~delia/sia/html98-99/pag-alumnos/web12/indice.html>

Algoritmos genéticos y Programación evolutiva  
<http://gal4.ge.uiuc.edu/techreps.php3>

Computación evolutiva  
<ftp://ftp.krl.caltech.edu/pub/EC/Welcome.html>

Algoritmos Genéticos en C++  
<http://ka-el.ugr.es/gags.html>

Biblioteca de algoritmos genéticos  
<http://lancet.mit.edu/ga/>

Código de programación genética  
<ftp://ftp.mad-scientist.com/pub/genetic-programming/code>

Programación genética exhaustiva  
<ftp://ftp.cs.bham.ac.uk/pub/authors/W.B.Langdon/biblio>

Programación genética

<http://www.genetic-programming.org/>

# 6

## VIDA ARTIFICIAL

### 6.1. INTRODUCCION

La Biología, de manera práctica, es el estudio científico de la vida basada en la química de las cadenas de carbono. No existe nada en su característica que limite, a la Biología, solamente al estudio de la vida basada en las cadenas de carbono; la razón es que esta es la única forma de vida disponible para su estudio.

La definición de vida no solamente tiene como objeto los caracteres que abarcan los organismos, sino también los mecanismos que los originan. Se afirma que la vida, como un proceso físico dinámico, podría "frecuentar" otro material físico, con la condición de que este material sea organizado de manera correcta. Ciertamente el proceso dinámico que constituye la vida debe compartir ciertas características universales que permitan reconocerla de manera simple mediante su forma dinámica, sin referenciar a su materia. Este fenómeno general de vida -la vida a través de todos los posibles sustratos materiales- es el verdadero asunto sujeto de la biología.

Según Nason, "una forma viviente es, en esencia, un sistema complejo, altamente organizado, independiente, con una estructura físico-química definida, capaz de utilizar la materia y energía del medio ambiente por medio de cadenas integradas y autoestablecidas de reacciones físico-químicas, para poder así crecer y reproducirse".

### 6.2. VIDA ARTIFICIAL

Es extremadamente difícil distinguir las propiedades esenciales de la vida de las propiedades que son incidentales a la vida, las cuales parecen ser universales a la vida en la tierra debido a una combinación de los accidentes históricos locales y los descendientes genéticos comunes. Según Langton, una forma vigente es tratar de sintetizar las formas de vida de manera alternativa, tal como sucede con la vida artificial, caracterizada como una forma de vida hecha por el hombre antes que por la naturaleza.

La vida artificial es un campo relativamente nuevo que emplea un enfoque sintético para el estudio de la vida como debería ser. En el campo, de la vida como debería ser, se considera a la vida como una propiedad de la organización de la materia, antes que una propiedad de la materia organizada.

Mientras que la biología está ampliamente interesada en las bases materiales de la vida, la vida artificial está interesada en las bases formales de la vida. La biología comenzó su estudio de arriba hacia abajo, observando los organismos vivos como una máquina bioquímica compleja, y trabajando de manera analítica hacia abajo<sup>57</sup> en busca de los mecanismos de vida. Langton

---

<sup>57</sup> Esta referencia se realiza hacia los órganos, tejidos, células, membranas, y finalmente moléculas.

menciona que la Vida Artificial empieza de la parte de abajo, considerando a un organismo como una gran población de maquinas simples, y trabaja hacia arriba de manera sintética<sup>58</sup>.

La relación entre la vida artificial y la biología teórica es de dos pliegues:

- a) debido a que uno de los principales objetivos de la Vida Artificial es producir la evolución conduciendo al incremento espontaneo de la diversidad y la complejidad, existe un campo rico de teoría biológica que sugiere los factores que pueden contribuir aquel proceso; y
- b) a la extensión de los procesos corrientes de la vida que son los mismos en la vida artificial y la vida orgánica, en este contexto los modelos de vida artificial proporcionan una nueva herramienta para el estudio experimental de tales procesos, los cuales pueden ser utilizados para demostrar que la biología teórica no puede ser probada por técnicas analíticas y experimentales tradicionales.

El concepto clave de la vida artificial es el "comportamiento emergente"<sup>59</sup>. La vida natural emerge de interacciones organizadas de un gran numero de moléculas no vivientes, sin ningún control global responsable del comportamiento de cada parte. Antes, cada parte es un comportamiento propio, y la vida es el comportamiento que emerge desde fuera de todas las interacciones locales entre comportamientos individuales. Esta es una determinación local, distribuida, hacia-arriba<sup>60</sup>, del comportamiento que emplea la vida artificial en su enfoque metodológico primario para la generación de comportamientos similares a la vida.

La Vida Artificial se puede considerar como la parte de la Inteligencia Artificial que pretende reproducir los procesos y comportamientos típicos de los seres vivos con el objetivo de resolver problemas. También podemos definirla como el intento de crear vida, o algo parecido a la vida, mediante la combinación de símbolos (datos) y procesos de símbolos (programas) independientemente del soporte físico de estos símbolos y procesos

### **6.2.1. Evolución del Área**

En 1984, cuando fue fundado el Instituto de Santa Fe, entidad privada dedicada a la enseñanza superior e investigación interdisciplinaria en Sistemas Complejos. La Vida Artificial es un área de investigación de reciente creación dentro del terreno más amplio de los Sistemas Complejos. El rito fundacional se celebró en los Estados Unidos en Septiembre de 1987, con sede en el Laboratorio Nacional Los Álamos de Santa Fe, Nuevo México. El nombre de este evento fue "Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems", también conocido como ALIFE I, y su gestor el investigador Christopher Langton. Dicho evento, que inaugura oficialmente la Vida Artificial como área independiente de investigación, fue apoyado por el "Center of Nonlinear Studies", el "Santa Fe Institute" y "Apple Computer Inc". Allí se dieron cita científicos dispersos procedentes de diversas disciplinas como la biología, informática, física, filosofía y antropología. Tras la segunda edición tomó carácter de congreso bienal. En Mayo de 1996 se celebró en Japón ALIFE IV.

---

<sup>58</sup> De manera similar a la construcción de grandes agregados de objetos simples gobernados por reglas que interactúan una con otra de manera no lineal.

<sup>59</sup> Una referencia inicial se realiza en el apartado 2, una referencia complementaria completa puede ser consultada en el apartado "Mecánica de la Vida".

<sup>60</sup> Traducción de la palabra inglesa "bottom-up".

La siguiente definición de Vida Artificial se debe a Chistopher Langton: "... es el campo de estudio dedicado a la comprensión de la vida, intentando abstraer los principios dinámicos fundamentales que subyacen a los fenómenos biológicos, y recreando esas dinámicas en otros medios físicos<sup>61</sup> haciéndolos accesibles a nuevos tipos de manejo experimental y de pruebas."

Es posible apuntar algunas causas del nacimiento de la Vida Artificial como campo específico:

- a) Las grandes posibilidades prácticas de la Biología. que inducen la necesidad de una Biología Teórica con leyes universales. Se trata de determinar las características que definen a los procesos vivos independientemente de la materialidad terrestre. Es decir, definir la vida tal y como podría ser en otra materialidad, e investigar los mecanismos bioquímicos que hacen emerger "lo vivo" como un fenómeno no reducible a las partes.
- b) Los progresos en la teoría matemática de los Sistemas Dinámicos. La Caracterización de ciertos comportamientos caóticos que se producen en los sistemas dinámicos no lineales realizada en los años 80 lleva a interesarse por el análisis de la dinámica cualitativa de los sistemas que presentan patrones emergentes complejos, es decir, que alcanzan configuraciones que no son absolutamente regulares ni absolutamente caóticas.
- c) Nuevas necesidades y posibilidades en el campo de la robótica. Como la exploración espacial, trabajos penosos, cooperación en cuadrillas, etc. El radical empujamiento de los procesadores permite construir pequeños robots dotados de autonomía, que imitan conductas animales, individuales y colectivas. Se puede así estudiar las conductas adaptativas y la cognición desde la perspectiva de la corporeidad, cuestionando a la Inteligencia Artificial.
- d) Nueva crisis del Software. Los ordenadores crecen exponencialmente en velocidad de proceso y capacidad de memoria decreciendo en tamaño, lo que hace posible la distribución distribuida de información y su procesamiento. El espectacular avance de la conectividad lleva a la creación de espacios On-line, como Internet, que están adquiriendo verdadera importancia y significado social. Estos son sistemas complejos que plantean nuevos problemas de procesamiento de información y que requieren nuevos lenguajes. Los sistemas biológicos son una referencia importante, ya que sistemas como un hormiguero o un ecosistema pueden ser entendidos y servir de inspiración a para construir sistemas informáticos capaces de resolver problemas

### 6.2.2. Intereses del Área

La Vida Artificial se ha ocupado de los siguientes temas:

- a) Procesos de nivel prebiótico, o de los mecanismos bioquímicos y evolutivos que llevan a la aparición de la célula. Los elementos al nivel bioquímico son proteínas, lípidos, carbohidratos y sus combinaciones forman un número tal que la organización de la célula viva tiene probabilidad casi cero. Se ha intentado explicar el surgimiento de la célula mediante modelos como la autopoiesis, desarrollado por Maturana y Varela, que sitúa el punto de partida para la evolución de la célula en los procesos de formación de membrana que permiten el flujo energético y que surja un metabolismo. Otros modelos se basan en

---

<sup>61</sup> Específicamente se refiere a las computadoras

la auto-organización de unidades autorreplicantes involucradas en procesos autocatalíticos, es decir, en los que el producto creado acelera el proceso de producción, y produce unidades cada vez más complejas.

- b) Procesos a escala celular: El fenómeno de la auto-reproducción celular es el que ha ocupado a la Vida Artificial, ya desde su origen remoto en el planteamiento del autómata autorreplicativo de von Neumann. Christopher Langton es el autor que, siguiendo la línea de von Neumann, ha estudiado los autómatas celulares como modelos de cómputo que pueden dar cuenta de la auto-reproducción, característica básica de lo viviente, tratando de crear expresamente una base para la Vida Artificial dentro de la Informática Teórica, de forma análoga a como fue instituida la Inteligencia Artificial.
- c) Procesos al nivel del organismo: La morfogénesis, es decir, el proceso de desarrollo de un individuo desarrollado o fenotipo en su ambiente a partir de una codificación genética es uno de los temas más importantes a investigar. La naturaleza de la cognición y del aprendizaje, el papel del sistema nervioso en la conducta adaptativa de los organismos, son los que más interés han despertado en la comunidad informática, ya que son básicamente los que planteó la cibernética clásica en su campo de intersección con la Inteligencia Artificial. Las Redes Neuronales se aplican en un amplio número de investigaciones en Vida Artificial, y ellas mismas son objeto de investigación.
- d) Procesos al nivel colectivo. Buscando determinar los mecanismos de interacción entre los individuos de una colectividad que hacen emerger comportamientos adaptativos o inteligentes al nivel de toda la colectividad de organismos. La simulación de hormigueros es un ejemplo, y sugiere múltiples aplicaciones a la Informática, sobre todo en cuanto a que tiene un campo de interés común con la Inteligencia Artificial Distribuida. La investigación en la comunicación animal y en el origen del lenguaje entran también en este nivel.
- e) Evolución filogenética. Investigando las leyes que rigen la evolución de las poblaciones, los mecanismos de transmisión genética, selección natural y adaptación de las especies. El problema de la evolución ha tenido consecuencias en el mundo de la informática: los algoritmos genéticos ya desarrollados por Holland desde los años 70 han encontrado importantes aplicaciones en la optimización de funciones. Esto ha dado lugar al nacimiento de la computación evolutiva y de la programación evolutiva como áreas de investigación específicas.

### **6.2.3. Herramientas del Área**

Las herramientas informáticas usadas en vida artificial son principalmente las redes neuronales y los algoritmos genéticos<sup>62</sup>. En unas pocas palabras, los primeros tratan de imitar la forma como funciona el sistema nervioso de los animales, y los segundos tratan de resolver problemas de optimización imitando la selección natural y su base molecular: mutación y entrecruzamiento de material genético.

---

<sup>62</sup> En los capítulos 4 y 5 se hace una referencia extensa a estos dos temas.

También se usan autómatas celulares y lógica difusa. Generalmente, se dota de algún tipo de "cerebro" a un agente (con reglas difusas o con redes neuronales), y se les hace evolucionar usando algoritmos genéticos. Esta es la base, pero el fin de la evolución puede ser muy diferente.

### 6.3. CARACTERISTICAS DE LOS MODELOS DE VIDA ARTIFICIAL

Christopher Langton propone las siguientes características, como esenciales para los modelos de vida artificial basados en computadoras:

- a) Constan de poblaciones de programas simples o especificaciones simples;
- b) no existe un programa simple que dirija a todos los otros programas;
- c) cada programa detalla la manera en la cual una entidad simple reacciona a las situaciones locales en su medio ambiente, incluyendo encuentros con otras entidades.
- d) no existen reglas en el sistema que dicten un comportamiento global.
- e) cualquier comportamiento hacia los niveles superiores antes que a los programas individuales es por consiguiente emergente.

En la búsqueda de las características universales que definen de manera simple a la vida, reconocida como sistema complejo, altamente organizado y con una estructura definida, se llega a recoger las ideas asociadas con la definición de la vida según sus propiedades mecánicas.

Ludwig menciona que las propiedades mecánicas de la vida son:

- a) reproducción propia,
- b) comportamiento emergente,
- c) metabolismo y adaptabilidad y
- d) evolución<sup>63</sup>.

#### 6.3.1. Algunos Modelos Representativos

- a) **Virus Computacional.** Los virus computacionales ofrecen un ejemplo gráfico de la VA e ilustran muchas de las propiedades importantes que poseen de manera típica los sistemas vivientes (reproducción, integración de partes, impredecibilidad, etc.).
- b) **Evolución Simulada en Computadora.** Son anfitriones de sistemas que incorporan la metáfora de los sistemas vivientes como procesos o patrones en tiempo y espacio capaces de reproducirse, absorbiendo información de su medio ambiente, modificar su espacio circundante, etc. Son importantes para reconocer que no es la computadora la que se considera viva, sino los procesos mismos que interactúan con el sustrato material que soporta al mismo (procesadores y memoria, incluyendo discos cuando existe memoria virtual) que son consideradas de manera propia como poseedores de las propiedades de la vida.
- c) **Biomorfismo y Procesos Realísticos Ontogenéticos.** La habilidad de los sistemas vivientes para evolucionar y su posesión de una morfogenesis completa que genera patrones complejos de partes integradas, parece casi única en el mundo natural. Patrones hermosos y complejos, frecuentemente realizados con geometría fractal, pueden emerger

---

<sup>63</sup> Estas características son desarrolladas mas adelante, concretamente en el apartado "Mecánica de la Vida".



de los resultado colectivos y repetidos de las acciones simples de células vivas. Los ejemplos de esta categoría incluyen a los L-Systems de Lindenmeyer y Prusinkiewicz.

- d) **Robótica**. Los robots representan un enfoque diferente a los sistemas vivientes. Al igual que el biomorfismo carecen de la habilidad de reproducción propia, aunque algunas compañías de computadoras (ej NeXT) no están lejos de contar con microcomputadoras controladores de robots que generen otras muchas microcomputadoras. Los robots poseen muchas de las propiedades que son asociadas de manera normal con la vida (complejidad, integración de partes, movimiento, etc).
- e) **Redes Autocatalíticas**. La antípoda de los robots de alta tecnología, los cuales son diseñados y construidos por humanos con los más altos niveles de tecnología, son sistemas simples que generan de manera espontánea vida, procesos de reproducción propia (o promoción propia) en una sopa química simulada. Dos de estos tipos de sistemas están bajo investigación activa: las redes catalíticas y los autómatas celulares. Un ejemplo especial bastante bien desarrollado de las redes autocatalíticas es la teoría de los hiperciclos (Manfred Eigen y Peter Schuster). Los hiperciclos son redes conectadas de elementos reactivos (por ejemplo RNAs) que permiten la evolución coherente de entidades funcionalmente acopladas y de replicación propia. Estas entidades son asociadas de manera típica a redes acopladas de reactivos químicos.
- f) **Autómatas Celulares**. Son arreglos de células, cada uno de los cuales asume datos discretos. Estos estados pueden cambiar hacia tiempos discretos de acuerdo a reglas bien definidas. Las reglas de transición son tomadas como las mismas en todo el arreglo y se toman en cuenta del estado corriente de la célula junto con los estados de sus vecinos inmediatos. Todas las células son asumidas para ser actualizadas de manera simultánea. Tanto las redes autocatalíticas como los autómatas celulares generan su propia organización, sus propios patrones de replicación los cuales persisten a través del tiempo.
- g) **Nucleotidos Artificiales**. Es importante reconocer en principio que la VA no está confinada solamente al medio ambiente de las computadoras. Mientras que pueden existir varios sustratos que soporten las características de la vida, se conoce de manera incuestionable que solo existe una, la relacionada con la vida natural, y que la variación de la vida como compuesto de componentes químicos es solamente el principio de la exploración de un área de grandes posibilidades. La sociedad debe prepararse con mucho cuidado para los cambios en gran escala provocados por la investigación en el campo de la biología molecular. Muchos de los cambios de la ingeniería genética en nuevas formas de vida pueden ser materiales de investigación para mejora o para ser eliminados, lo ultimo debido al uso de un código genético alternativo y algún método forzado.
- h) **Evolución Cultural**. Las ideas actuales convertidas en realidad, quizá originadas de culturas foráneas, pueden cambiar la cultura milenaria en formas favorables para su supervivencia, pueden extenderse a otras sociedades y luego morir. Las colecciones de ideas frecuentemente forman medio ambientes de soporte mutuo para su *propagación*. Actualmente las distintas tendencias religiosas y las novedades como las tortugas ninja son solamente dos ejemplos.

#### 6.4. MECANICA DE LA VIDA

Parece razonable sugerir que existe un cierto conjunto de funciones con los cuales cualquier sistema físico debe ser capaz de ejecutarse si es clasificado como vivo. Que equivale a decir que un organismo vivo debe ser capaz de poder hacer ciertas cosas. Desafortunadamente, definir tal lista de funciones es una confusión casi intratable, aún cuando solo se investigue sobre los organismos basados en la química del carbono. No importa que clase de lista se establezca, esta no puede ser general, siempre existen excepciones que pueden ser encontradas.

El problema mencionado en el párrafo precedente es particularmente agudo cuando se trata con funciones macroscópicas. Si se enfoca en los detalles microscópicos de como trabajan los organismos basados en la química del carbono, se puede dibujar una línea, entre lo que es la vida y lo que no es, de manera mas cerrada. Para definir la vida de manera abstracta, es necesario alejarse de los detalles microscópicos que caracterizan los sistemas específicos y hacer énfasis en sus propiedades abstractas.

Para el estudio emprendido el interés está centrado en organismos individuales, antes que las poblaciones. En este dominio, los investigadores de la VA toman con gran énfasis muchos de los aspectos funcionales de la vida, entre los cuales destacan:

- a) La habilidad de reproducirse y el método de reproducción.
- b) El concepto de comportamiento emergente.
- c) La posesión de un metabolismo.
- d) La habilidad de funcionar bajo perturbaciones del medio ambiente e interactuar con el medio ambiente.
- e) La habilidad de evolucionar.

El estudio de estos aspectos funcionales es realizado por Ludwig en 1993 y se desarrollan a continuación.

#### **6.4.1. Reproducción Propia**

Los organismos vivos, en general, son capaces de reproducirse. Aunque individuos específicos no puedan ser capaces de reproducirse debido a circunstancias accidentales, al final de su etapa en el ciclo de vida, o a una inusual combinación genética, miembros de la población entera deben tener la capacidad de realizar copias completas o cercanas de si mismos.

#### **6.4.2. Comportamiento Emergente**

En un organismo biológico, el ácido desoxirribonucleico (DNA) determina las características físicas de un organismo. Es esencialmente el medio en el cual es escrito el "programa de la vida". Sin embargo, la relación entre el programa de la vida en su forma mas cruda y la expresión manifiesta de este en un organismo vivo es extremadamente obtusa. En lo obtuso de esta relación es que se forma el núcleo de la idea del comportamiento emergente. Para comprender de mejor manera este concepto, se revisa el mismo en el contexto de los organismos biológicos.

Se empieza visualizando como trabaja el DNA. La molécula de DNA podría ser descrita propiamente como un cristal de una dimensión o una fibra. Sin embargo, a pesar de lo que se piensa de los cristales, el DNA es un cristal aperiódico. Antes que ser un compuesto de una molécula simple (denominada nucleótido) repetida infinitas veces, el DNA consiste de cuatro diferentes nucleotidos denominados bases, los cuales son funcionalmente idénticos con respecto a

la estructura del DNA mismo. Esas bases pueden ser sustituidas por cualquier otra en las celdas del cristal sin alterar la estructura física de la molécula de DNA. Por consiguiente se puede concebir la construcción de una molécula de DNA para codificar cualquier clase de información deseada.

Los genes en el DNA pueden ser considerados como "instrucciones" complejas que pueden ser "ejecutadas" por las proteínas que realizan todas las funciones de la célula. Las instrucciones codificadas en el DNA son referenciadas como su genotipo<sup>64</sup>. En el caso de los organismos vivos, la "ejecución de las instrucciones" consiste del funcionamiento propio de las distintas proteínas en una célula para producir vida. Esta complejidad extrema, relacionada con la ejecución de las instrucciones, es denominada su fenotipo<sup>65</sup>.

El concepto de comportamiento emergente da vueltas alrededor de la pregunta de como el genotipo determina al fenotipo. El comportamiento emergente es simplemente la idea de que el comportamiento, en un organismo complejo, no está dirigida por un programa centralizado. Mas bien, el comportamiento emerge de la interacción compleja de las partes individuales.

#### **6.4.3. Metabolismo y Adaptabilidad**

Muchos organismos vivos utilizan energía para mantenerse en un estado de baja entropía y llevar a cabo sus actividades, incluida la reproducción propia. Los organismos vivos hacen uso de la energía directa tomada del sol, o convierten la energía almacenada en moléculas orgánicas complejas en formas que puedan utilizar. El proceso mediante el cual los organismos vivos utilizan materia y energía es denominado el metabolismo del organismo.

Por otra parte, los organismos vivos individuales pueden generalmente interactuar y modificar su medio ambiente, como también adaptarse a los cambios en dicho ambiente. Esta capacidad de adaptación es conocida como adaptabilidad, la misma hace que un organismo sea flexible y estable en la faceta del cambio. Por supuesto que la habilidad de un organismo para adaptarse a su medio ambiente es limitado, especialmente cuando se enfrenta a un cambio extremo<sup>66</sup>.

#### **6.4.4. Evolución**

La capacidad de los organismos de cambiar sus características, influenciada por los factores de herencia de los padres, a través del tiempo es conocida como evolución. Muchos organismos biológicos parecen tener, al menos alguna capacidad, para contar con la evolución al estilo Darwiniano, especialmente el referido a que cualquier organismo vivo tiene una prole. Generalmente, la derivación genética de esta prole no es la misma que la derivación genética de los padres. Así la composición genética de una población puede cambiar a través del tiempo. En realidad, la composición genética de una población puede ser influenciada por factores externos, donde un gen prueba tener mas capacidad de supervivencia que otros.

### **6.5. APLICACIONES**

---

<sup>64</sup> Conjunto de potencialidades genéticas de un individuo, determinadas por la combinación de sus alelos. Constituye el patrimonio genético que permanece fijo en la fecundación.

<sup>65</sup> Conjunto de caracteres morfológicos y fisiológicos observables en un organismo.

<sup>66</sup> Por ejemplo, cuando un pez sale fuera del agua, el organismo no es capaz de adaptarse, en cuyo caso muere.

Entre las aplicaciones más importantes se menciona las siguientes:

- a) Los Autómatas Celulares y entre ellos, el Juego de la Vida de Conway, constituyen los ejemplos más sencillos de Vida Artificial.
- c) En Dioses, Hombres, Demonios y Máquinas se tratan temas de interés científico en computación que incumben a las religiones, como el de las máquinas que se autorreproducen; las máquinas que aprenden y la relación hombre-máquina.
- d) En Autómatas como analogías de nuestro Universo, se describe el trabajo con analogías entre nuestro universo real y universos simulados en la creación de hipótesis acerca del tiempo, el espacio, la vida y el azar.
- e) El tamagotchi y mi hijo ¿Quién es la mascota de quién? En este documento se analiza la influencia sobre los niños de las simulaciones de vida como juguetes, del tipo del Tamagotchi.
- f) En el artículo "Agentes Autodidactas ¿Futuro o Realidad?" se describe cómo es posible crear un sistema de Vida Artificial con múltiples agentes capaces de aprender a resolver por sí mismos cualquier tipo de problema.

## Ejercicios # 6

1. Complementar la temática relativa a la vida artificial observado ejemplos de vida, en el sitio [http://www.aircenter.net/gaia/ejv/ejv\\_c.htm](http://www.aircenter.net/gaia/ejv/ejv_c.htm), en este lugar se pueden estudiar varios mundos habitados por seres artificiales, entre ellos, un mundo virtual donde conviven hormigas y plantas y el dilema del prisionero como ejemplo de las posibles aplicaciones de la Vida Artificial en cuanto al estudio de comportamientos sociales.
2. Realizar una comparación de las razones científicas y teológicas acerca del origen de la vida.
3. Intentar una explicación del origen de la vida con el siguiente argumento: una sola bacteria, contiene más de 2000 enzimas. Las posibilidades de que se forme una cadena de enzimas completa en un caldo con todas las enzimas individuales son ínfimas. El tiempo para que se formara oscila entre cuarenta y cien mil millones de años. Ahora bien, la vida surgió hace tres mil quinientos millones de años, por lo que no hay tiempo material para que se forme una enzima, y muchísimo menos para que evolucione a las formas de vida actuales.
4. Cual es su opinión acerca del origen de la vida en otro astro, donde tras miles de millones de años se formó una enzima, y por una casualidad llegó a la Tierra.
5. Desarrollar los argumentos a favor de la explicación del origen de la vida en una nube de gas cósmico.
6. Si se acepta la actual teoría evolucionista, la vida se reduce a una acumulación de sucesos aleatorios producidos en el tiempo. ¿Cuál es su opinión acerca de la perfección provocada por la supuesta aleatoriedad en la configuración perfecta de los seres vivos? ¿No cree que hay una suerte de problema coordinatorio?
7. Si se considera el problema selectivo se puede decir que no siempre sobrevive el más adaptado, ni es necesario adaptarse para sobrevivir. Algunos animales como los tiburones no han cambiado nada desde la extinción de los dinosaurios, pero su entorno ha cambiado radicalmente. ¿Cuál es su opinión acerca de que posiblemente hayan llegado al máximo punto de perfección para su hábitat?
8. La interacción de dos elementos se puede conocer matemáticamente, la de tres es muy complicada y la de cuatro prácticamente imposible. En una sola célula interactúan cientos de miles de objetos. Igualmente, en un cuerpo humano hay cientos de miles de células. La experiencia demuestra que de cuantos más factores depende un problema, más posibilidades

- hay de que falle. ¿Que opina acerca de que en la vida todo funciona mas o menos a la perfección?
9. Utilizando la teoría relativa a la mecánica de la vida, demostrar que un virus computacional es un organismo vivo.
  10. Diseñar un mundo de dos dimensiones en el que conviven hormigas y plantas. Las plantas crecen cuando son regadas. Las hormigas pueden comer plantas, moverse, regar plantas, pelear y reproducirse.

## Bibliografía

- Langton, Ch. *Artificial Life*. Addison-Wesley Publishing Co. 1989.
- Gleick, J. *Caos: La creación de una ciencia*. Seix Barral. 1994.
- Wiener, N. *Cibernética, o el control de comunicaciones en los animales y las máquinas*. Tusquets Editores. 1985.
- Leinecker, R. *Developing Dinosaurs and Ancient Worlds*. Sams Publishing. 1994.
- Monod, J. *El azar y la necesidad*. Barral Editores. 1977.
- Dawkins, R. *El gen egoísta: Las bases biológicas de nuestra conducta*. Tusquets Editores. Barcelona. 1994.
- Dawkins, R. *El relojero ciego*. Editorial Labor. 1988.
- De Guzman, M., M. A. Martín, M. Moran & M. Reyes. *Estructuras fractales y sus aplicaciones*. Editorial Labor Matemáticas. 1993.
- Meyer, J.A. & S. Wilson. *From animals to animats*. MIT. Press. 1991.
- Crick, F. *La búsqueda científica del alma*. Editorial Debate Pensamiento. 1994.
- Lorenz, E.N. *La esencia del caos*. Ed. Debate Pensamiento. 1994.
- Mandelbroth, M. *La geometría fractal de la naturaleza*. Tusquets Editores. Barcelona. 1997.
- Goodwin, B. *Las manchas del leopardo. La evolución de la complejidad*. Tusquets Editores. Barcelona. 1998.
- Mandelbrot, B. *Los objetos fractales. Forma, azar y dimensión*. Tusquets Editores. 1984.
- Gardner, M. *Ruedas, vida y otras diversiones matemáticas*. Editorial Labor. 1988.
- Von Bertalanffy, L. *Teoría general de los sistemas*. Fondo de Cultura Económica. 1980.
- Fernandez J. & A. Moreno. *Vida Artificial*. Eudema Biología. 1992.
- Prata, S. *Vida Artificial. La evolución en la punta de tus dedos*. Anaya Multimedia. 1994.

## Referencias Electrónicas

Universidad de Santa Fe (USA). Grupo precursor de los estudios de sistemas dinámicos no lineales, complejos y el caos.

<http://alife.santafe.edu/>

Instituto Tecnológico de Massachusset (USA).

<http://mitpress.mit.edu/journals-legacy.tcl>

Universidad Politécnica de Cataluña (España).

<http://aleph.ac.upc.es/aliweb/>

Universidad Yale (USA).

<http://www.nnc.yale.edu/HTML/YALE/NNC/neuron/html/nrmsim.html>

Instituto Tecnológico de California (USA).

<http://www.bbb.caltech.edu/cns.html>

Universidad Charles Sturt (Australia).

<http://complex.csu.edu.au/complex>

Universidad de Granada (España).

<http://Ka-el.ugr.es/gags.html>

Universidad de Sussex (Reino Unido).

<http://www.cogs.susx.ac.uk/lab/adapt/index.html>

ISIS. (Reino Unido).

<http://www.isis.ecs.soton.ac.uk/>

# 7

## REALIDAD VIRTUAL

### 7.1. INTRODUCCION

Soñar es una forma de realizar todos aquellos deseos negados. Soñando es posible volar, alcanzar las más altas cumbres del planeta o descender a las más profundas simas. Por desgracia en general no se puede dominar los sueños y al final la realidad acaba limitada a lo que ofrece la percepción de los sentidos. Franquear la barrera de los sentidos y ofrecer sensaciones de extremo realismo debería ser casi como un sueño controlado.

La sensación de estar en otro mundo mediante imágenes estereoscópicas y realistas proyectadas en una pequeña pantalla al interior de un casco, al cual se añaden sonidos, convencen a la mayoría de los usuarios de que este tipo de sensaciones se acercan a una realidad creada por la computadora. Los sensores de movimiento colocados en el casco y en los guantes de los usuarios son los medios de interacción con este nuevo mundo.

En este nuevo mundo, la velocidad y potencia de la computadora se combinan con los avances en el procesamiento de imágenes y, los mecanismos de búsqueda e intuición humana en la comunicación por computadora, para dar lugar al medio experimental denominado "realidad virtual".

Una computadora diseñada para desarrollar buenas imágenes y en rápida sucesión debe tener gran potencia y velocidad además de buenos recursos de visualización. En cada instante de tiempo estas tecnologías generalmente se encuentran en diferentes fases de desarrollo. Su coordinación conlleva irregularidades en el tiempo o en la calidad. Para la creación de mundos virtuales, cada una de las tecnologías involucradas debe alcanzar en cada fase una intensidad y unos recursos que puedan ser utilizados de manera efectiva junto con las demás. La convergencia debe dar como resultado una inmersión que tenga las siguientes características:

- a) Sea percibida como auténtica;
- b) Permita la interacción intuitiva y responda a ella rápidamente;
- c) Esté definida y enfocada, incluso como medio de entretenimiento;
- d) Facilite la recopilación, percepción y análisis de datos.

Desde sus principios la Realidad Virtual (RV) ha tratado de ofrecer una interfaz entre los sentidos y las máquinas, pero entonces, ¿qué es la realidad virtual? Un teclado y una pantalla también ofrecen una interfaz adecuada, pero no brindan una experiencia realista. El término clave para definir qué es y qué no es la RV es el de "inmersión". La frontera entre los sistemas de RV y los sistemas convencionales está en que mientras que en los primeros se percibe la información como una verdad representada mediante caracteres, imágenes, audio o incluso gráficos tridimensionales, en los segundos se encuentra la información, tal y como si compartiese el mismo espacio en el que uno se encuentra.

La realidad virtual explota todas las técnicas de reproducción de imágenes y las extiende, usándolas dentro del entorno en el que el usuario puede examinar, manipular e interaccionar con los objetos expuestos. De este modo los investigadores y usuarios son capaces de utilizar imágenes para transmitir, no solamente la información, sino también la capacidad de interpretarla.

## **7.2. HISTORIA**

La historia anterior de la realidad virtual, antes de 1990, puede ser descrita mediante un número pequeño de investigaciones de laboratorio que siguen caminos independientes. Los años 90 significaron el crecimiento tanto del número de investigadores de RV y la calidad de investigación, guiada por el decremento en los costos de los equipos computacionales. Durante este periodo fueron realizadas muchas conferencias sobre RV, resultando en el desarrollo de una comunidad de investigación sobre realidad virtual con una agenda y visión compartida. Hacia 1993 el IEEE llegó a patrocinar en gran escala conferencias técnicas sobre realidad virtual.

El simposio internacional anual sobre RV (VRAIS '93) se llevó a cabo en Seattle, Washington, patrocinado por la IEEE NCC (Neural Networks Council). En octubre el simposio de la IEEE sobre las fronteras de la investigación en RV fue realizado como parte del simposio sobre Visualización '93 realizado en San José, California, patrocinado por la IEEE TCCG (Technical Committee on Computer Graphics). Ambos eventos lograron la cohesión de la comunidad de investigación enfocada en los problemas de la RV. El VRAIS '95 se llevó a cabo en Carolina del Norte en marzo de 1995. El programa técnico del VRAIS '95 reflejó el rango amplio de resultados que se ejecutan en la RV. Este simposio realizó un énfasis amplio sobre los factores humanos, logrando siete artículos en dos sesiones. La RV distribuida mereció también una gran atención, con una sesión sobre infraestructura de la RV y una sobre aplicaciones de la RV distribuida. Una sesión se ocupó sobre las técnicas para popularizar los medio ambientes virtuales.

## **7.3. COMUNICACIÓN HOMBRE MÁQUINA**

Diseñar los medios para comunicarse de manera fácil y precisa con una computadora, además de controlar lo que debe ocurrir, no es una tarea fácil. Un campo de estudio amplio se desarrolló en torno al problema de la interacción humana con máquinas complejas y es denominada de manera alterna: ingeniería humana, factorización humana, análisis de factores humanos, tecnología de interfaces humanas, interacción hombre - máquina, etc.

Los problemas más complejos de las interfaces radican en la interacción entre el hombre y la máquina, esta tarea normalmente es bastante complicada y no se ha conseguido perfeccionar todavía. Se recomienda que las interfaces deben ser diseñadas por personas con un alto nivel de conocimientos en varios dominios (psicológico, temático y técnico), con el objeto de minimizar la pérdida de información, recogiendo el máximo beneficio del esfuerzo humano realizado. Las interfaces deben servir como escenarios de una comunicación excelente y deben ser sencillas de utilizar para todo tipo de usuarios. Las interfaces deben ahorrar la constante interrupción en el entrenamiento relativo al pensamiento del usuario.

## **7.4. ESPACIO DE LA REALIDAD VIRTUAL**



Un gráfico útil que muestra la ubicación de la realidad virtual en un espacio doble de transportación y artificialidad se muestra en la Fig. 7.1.

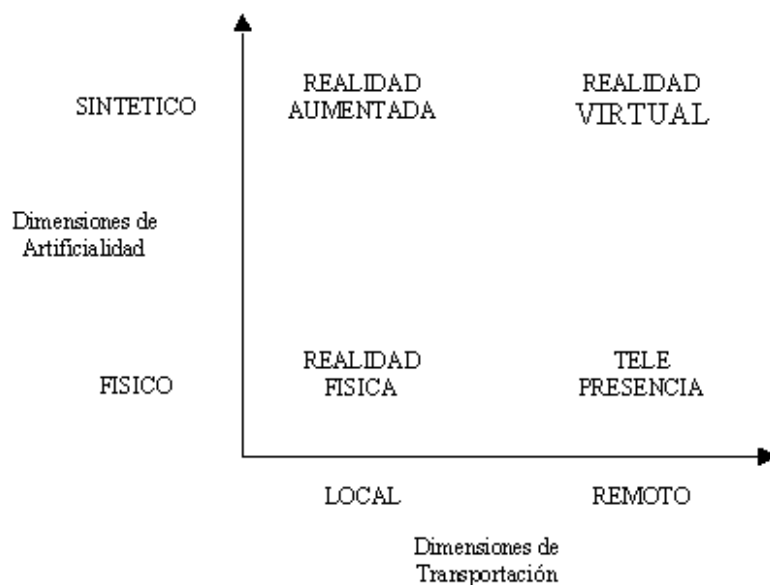


Fig. 7.1. Espacio de la realidad virtual

## 7.5. DEFINICION

Cualquier definición sobre RV se considera como transitoria, debido a que es una tecnología en plena evolución. Sin embargo una de las definiciones más aceptada es la siguiente: *'Realidad Virtual es simulación por computadora, dinámica y tridimensional, con alto contenido gráfico, acústico y táctil, orientado a la visualización de situaciones complejas y variables, durante la cual el usuario ingresa, mediante el uso de sofisticados dispositivos de entrada a "mundos" que aparentan ser reales, resultando inmerso en ambientes altamente participativos, de origen artificial'*.

## 7.6. DISPOSITIVOS

Cuando se escucha hablar de Realidad Virtual a casi todos nos viene a la mente la imagen de una persona con un casco y un guante lleno de cables acariciando el aire, estos son probablemente dos de los dispositivos de Realidad Virtual más clásicos y paradójicamente menos utilizados en la actualidad. ¿Cuales son, entonces, esos dispositivos que consiguen introducir al ser humano en una experiencia virtual?. Se pueden distinguir los siguientes tipos de dispositivos:

- a) **Generadores de imagen.** Constituyen el cerebro de los sistemas de Realidad Virtual, generalmente consisten en grandes máquinas que se encargan de construir en tiempo real los gráficos de la aplicación que se desarrolla.
- b) **Dispositivos de control y manipulación** Se refiere a diversos tipos de ratones, joysticks o los ya clásicos guantes de datos que permiten posicionarse en el mundo virtual, así

como actuar sobre él. El problema de todos estos dispositivos está en el tiempo de latencia<sup>67</sup> y en que es un único usuario el que participa de la experiencia.

- c) **Dispositivos de seguimiento.** Permiten determinar la posición dentro del mundo virtual, así como los distintos giros que se puedan realizar sobre cualquiera de los ejes de coordenadas. Generalmente esta información debe generar una respuesta visual adecuada porque de lo contrario puede llegar a producirse una sensación de mareo en el usuario<sup>68</sup>.
- d) **Visión estereoscópica.** Es un elemento casi imprescindible en un sistema de RV. Consiste en generar por separado las imágenes del ojo izquierdo y del ojo derecho que el cerebro se encargará de mezclar para proporcionar una sensación de tridimensionalidad. Un dispositivo clásico de visión estereoscópica es el HMD<sup>69</sup>, generalmente constituido por dos pequeñas pantallas integradas en un casco que se conectan al resto del sistema mediante una serie de cables que salen por la nuca. El problema de estos dispositivos es que son excesivamente caros, sirven sólo para una persona y limitan en gran medida su movilidad. Otro de los dispositivos clásicos son las gafas estereoscópicas, que mediante diversas tecnologías se encargan de filtrar la imagen para cada ojo, a partir de una pantalla en la que se proyectan alternativamente y a gran velocidad las imágenes izquierda y derecha. Esto permite que varios usuarios perciban la misma experiencia

## 7.7. CLASIFICACION

La RV está presente en cada vez en más áreas, tales como la arquitectura, educación, investigación, simulación y entrenamiento, información y publicidad, o el cada vez más exigente mercado del entretenimiento. Dentro de los sistemas de Realidad Virtual, según el tipo de interface con el usuario, se pueden distinguir los siguientes:

- a) Sistemas de ventana
- b) Sistemas de mapeo por vídeo
- c) Sistemas inmersivos
- d) Sistemas de telepresencia
- e) Sistemas de realidad mixta
- f) Sistemas de pecera

### 7.7.1. Sistemas de Ventana

Son también conocidos como Sistemas de RV sin inmersión. Muchos de estos sistemas utilizan un monitor convencional para mostrar un mundo virtual. Estos sistemas son conocidos como "mundos en ventanas"<sup>70</sup> o como RV de escritorio. Los sistemas de ventana siguen la línea definida por Ivan Sutherland, quien a finales de los años sesenta estableció que: *"uno debe mirar a la pantalla de la computadora como si fuera una ventana a través de la cual contempla un mundo virtual. El reto es hacer que la imagen que aparece en la pantalla luzca real, suene real y que los objetos que en ella se representan actúen con realismo"*.

De manera practica consiste en utilizar monitores convencionales para representar las imágenes realistas. Esta modalidad existe desde prácticamente el origen de las propias tecnologías de

---

<sup>67</sup> Instante desde el que se ejerce acción sobre el mundo hasta que se perciben los resultados.

<sup>68</sup> Se refiere al hecho de girar la cabeza y no percibir un cambio simultáneo en el punto de vista de la escena, o cuando el cambio del punto de vista no es lo suficientemente progresivo.

<sup>69</sup> De la palabra inglesa Head Mounted Display.

<sup>70</sup> De la palabra inglesa window on world (WOW).

Realidad Virtual y su capacidad de inmersión es escasa. De todos durante más de 30 años este ha sido el campo de investigación sobre algoritmos de visualización 3D por excelencia, y la práctica totalidad de estos ha surgido del desarrollo de los sistemas WoW o "Desktop Virtual Reality".

### **7.7.2. Sistemas de Mapeo por Vídeo**

Constituyen una forma particular de sistema inmersivo. Estos sistemas fueron inspirados en el pensamiento del artista Myron Krueger. La realidad artificial de estos sistemas se basa en la filmación mediante cámaras de vídeo de una o más personas y la incorporación de dichas imágenes a la pantalla de la computadora, donde interactúan en tiempo real con otros usuarios o con otras imágenes gráficas generadas por el sistema computacional. De esta manera las acciones que realiza el usuario en el exterior de la pantalla se reproducen en la pantalla de la computadora permitiendo interactuar lo que está afuera con lo que está adentro. El usuario puede, mediante este enfoque, simular su participación en aventuras, deportes y otras formas de interacción física.

En otras palabras se puede decir que constituyen una variación de los sistemas WoW donde se mezcla la silueta del usuario -capturada mediante vídeo convencional- con un escenario generado por ordenador. Estos no son sistemas inmersivos, de hecho estos sistemas "dan la vuelta" al concepto de Realidad Virtual ya que en vez de estar orientados a hacer que el usuario se sienta integrado en el mundo, tratan de generar un mundo que integra la imagen del usuario<sup>71</sup>. Se puede ver este tipo de sistemas funcionando cada vez que se observa la predicción del tiempo en la televisión.

### **7.7.3. Sistemas Inmersivos**

Son los sistemas más perfeccionados de RV, los cuales permiten que el usuario pueda sentirse "sumergido" en el interior de un mundo virtual. El fenómeno de inmersión puede ser experimentado mediante cuatro modalidades diferentes, dependiendo de la estrategia adoptada para generar esta ilusión, estas modalidades son:

- a) Operador aislado.
- b) Cabina personal.
- c) Cabina colectiva.
- d) Caverna

Estos sistemas tratan de introducir al propio usuario dentro del mundo virtual, proporcionándole sensaciones realistas como la estereoscopia<sup>72</sup> o el sonido envolvente. Tradicionalmente se han utilizados "video cascos" (HMD), aunque cada vez se encuentran más en boga los sistemas de tipo "caverna" en los que el usuario se introduce en habitaciones cerradas sobre cuyas paredes se proyectan las imágenes del mundo virtual. La ventaja de estos últimos sistemas es que permiten la participación de varios usuarios en una misma experiencia virtual.

Los sistemas inmersivos generalmente se encuentran equipados con un casco visor HMD. Este dispositivo está dotado de un casco o máscara que contiene recursos visuales, en forma de dos mini pantallas que interactúan para producir visión estereoscópica y recursos acústicos de efectos tridimensionales.

---

<sup>71</sup> El usuario se transforma en virtual, en vez de serlo el mundo.

<sup>72</sup> Hace referencia a la visión tridimensional.

Una variante de este tipo de sistemas se basa en el uso de múltiples pantallas de proyección de gran tamaño, dispuestas de manera ortogonal, para crear un ambiente tridimensional o caverna en la cual se ubica a un grupo de usuarios. De estos usuarios, uno es el que asume la tarea de navegación, mientras los demás pueden dedicarse a visualizar los ambientes de RV dinámicos en tiempo real.

#### **7.7.4. Sistemas de Telepresencia**

Consiste en proporcionar al usuario las sensaciones recogidas por sensores distantes situados en un determinado campo de operaciones. Esto permitiría teleoperar sondas-robot en el espacio, realizar operaciones quirúrgicas a distancia o dirigir vehículos de forma remota.

Esta tecnología se encarga de fusionar sensores remotos en el mundo real con los sentidos de un operador humano. Los sensores remotos utilizados pueden encontrarse instalados en un robot. De esta forma, el usuario puede operar el equipo como si fuera parte componente de sus sentidos. La telepresencia contempla, de manera obligatoria, un grado de inmersión que involucra el uso de control remoto, pero posee características propias suficientes como para asignarle una clasificación particular.

#### **7.7.5. Sistemas de Realidad Mixta**

La fusión de los Sistemas de Telepresencia y la RV se obtienen los denominados Sistemas de Realidad Mixta. Estos sistemas son también conocidos como sistemas de simulación sin costura, donde las entradas generadas por la computadora se mezclan con las entradas de telepresencia y/o con la visión de los usuarios del mundo real. Este tipo de sistemas se ha venido utilizando tradicionalmente en los aviones de combate, en los que al piloto se le proyecta información de vuelo sobre la propia visera de su casco

#### **7.7.6. Sistemas de Pecera**

Son sistemas parecidos a los WoW, en los que se combina una pantalla con un dispositivo de posicionamiento, de forma que la pantalla supone una verdadera ventana al escenario virtual y moviendo esta se puede cambiar el punto de vista sobre el mundo. A veces se utilizan visores estereoscópicos en vez de pantallas planas.

Este sistema combina un monitor de despliegue estereoscópico utilizando lentes LCD con obturadores acoplados a un rastreador. El sistema resultante es superior a la combinación simple del sistema estéreo WoW debido a los efectos de realismo tridimensional de movimiento introducidos por el rastreador.

### **7.8. APLICACIONES**

La investigación y desarrollo en aplicaciones de medio ambientes virtuales puede ser encontrada en muchos lugares del mundo. Las aplicaciones que son comercialmente disponibles o están en investigación, con importantes posibilidades pueden ser divididas en las siguientes categorías: diseños de arquitectura y diseño asistido por computadora; soporte para decisiones; telecomunicaciones, interfaces, y lugares remotos; entrenamiento; exploración científica; entretenimiento. Debido al amplio desarrollo de la revolución para el campo medico, todas las técnicas relacionadas al medio ambiente virtual funcionan de manera conjunta en este dominio.

Se puede establecer cinco categorías de aplicaciones profesionales de la Realidad Virtual:

- a) **Prototipado-modelado.** Las aplicaciones de prototipado son aquellas en las que se utiliza la realidad virtual para establecer un modelo de cómo sería un determinado desarrollo que aún está en sus etapas más tempranas. Esto es muy útil en áreas como las del diseño industrial, por ejemplo, es posible pasear en el último modelo de todo terreno antes de que se haya realizado sobre él una sola soldadura, arquitectura, planificación de operaciones, por ejemplo misiones en el espacio o lugares de alta inaccesibilidad, o sencillamente se pueden realizar paseos virtuales sobre terrenos existentes o inexistentes.
- b) **Entrenamiento-simulación-enseñanza.** La Realidad Virtual puede resultar útil en la enseñanza y entrenamiento por medio de simuladores. Un ejemplo clásico son los simuladores de vuelo donde tradicionalmente se han entrenado los pilotos militares, pero cada vez existen sistemas de entrenamiento de máquinas más cotidianas, como por ejemplo los simuladores de conducción que resultarán aparatos cotidianos en las autoescuelas.
- c) **Aplicaciones metafóricas.** Estas aplicaciones pretenden utilizar la RV para dotar de una representación visual que aumente la capacidad de comprensión de los seres humanos, sobre un determinado proceso. Las aplicaciones metafóricas típicas suelen ir encaminadas a la representación de datos científicos, por ejemplo, el túnel de viento virtual desarrollado en el NASA Ames Center, que representa los flujos de aire sobre un determinado modelo virtual, pero no solamente se encuentran limitadas a este campo, por ejemplo, Maxus Systems International desarrolló en 1992 un sistema de este tipo que se empleaba para manipular carteras de valores de bolsa.
- d) **Teleoperación y tele-robótica** Esta área está fuertemente ligada a los sistemas de telepresencia y de realidad aumentada, y se emplea en ámbitos como la programación de robots<sup>73</sup>, los sistemas de exploración sobre terrenos inaccesibles o en las cada vez más cercanas aplicaciones de telemedicina.
- e) **Aplicaciones de ocio.** Cada vez es más frecuente el uso de sistemas inmersivos en parques de atracciones, cada vez los videojuegos resultan más complejos y las empresas que los desarrollan más poderosas, es cuestión de tiempo el que estas se lancen a los sistemas de Realidad Virtual y que estos lleguen a gran cantidad de hogares.

Es difícil predecir cual puede ser el futuro de esta tecnología, el mayor problema a resolver en la actualidad es el proporcionar una interfaz adecuada entre el mundo virtual y el usuario que lo habita. Acciones cotidianas como tomar algo que está sobre la mesa y dejarlo sobre el suelo pueden no ser tan evidentes en un entorno de Realidad Virtual, en el que a lo mejor la acción de tomar un objeto se realiza señalando con el dedo y la de dejarlo abriendo la palma de la mano. Una cosa es el posicionarse en un escenario y otra bien distinta es poder interactuar con los propios objetos. Obtener una información sobre el tacto, dureza o peso de los objetos podría ayudar a realizar sobre el mundo virtual las mismas acciones que se realizan sobre nuestro mundo real y que resultan evidentes, de forma más natural.

Otra área en la que se investiga es las interfaces gráficas de usuario, campo en el que parece evidente que algún día deberá migrar de las interfaces planas actuales a nuevas y espectaculares interfaces gráficas tridimensionales. Llegados a este punto habrá que rescribir muchas de las analogías que se utilizan en los Sistemas Operativos actuales y que todos damos por válidas, por

---

<sup>73</sup> Hace referencia a la programación de movimientos sobre un robot virtual que reproducirá un robot real.

ejemplo el "cortar" y "pegar" es algo que se realiza sobre el plano, mientras que tal vez sobre el espacio debería empezar a hablarse de "coger" y "dejar".

Por último, también el concepto de "comunidad virtual" empieza a tomar importancia, tal vez en un futuro bastante más cercano de lo que se cree la videoconferencia sea sustituida por salas virtuales en la que personajes -conocidos como "avatares"- que representan a usuarios del mundo real charlen e intercambien información de una forma tan natural como si todos ellos estuviesen presentes en la misma estancia.

## Ejercicios # 7

1. Complementar la temática propuesta con la cronología de la realidad virtual.
2. Comentar acerca de las razones por las cuales términos como "sociedad virtual" y "realidad virtual" tienen hoy una popularidad como la que tuvieron en la década del cincuenta los términos "atómico" y "nuclear".
3. Describir que es la ciberantropología utilizando los resultados de los congresos mundiales sobre el ciberespacio.
4. Analizar las tendencias actuales de la sociedad virtual. Incluir temas como el estudio etnográfico de las comunidades virtuales —los hackers, los cultores de los juegos de rol en línea (*muds*), los *cyberpunks*, etc.—, las nuevas identidades sociales que vienen emergiendo en el mundo virtual, y la forma cómo se plantean viejos temas de la dinámica social real en la sociedad virtual, como, por ejemplo, las percepciones y las interacciones entre raza, etnia y género, o la relaciones entre economía, comunicaciones y poder.
5. Escribir un ensayo que muestre las diferencias y similitudes que existen entre la sociedad real y la sociedad virtual.

## Bibliografía

Ellis Stephen R., M. K. Kaiser & A. J. Grunwald (Eds.) *Pictorial communication in Virtual and Real Environments*. Taylor & Francis, London 1991.

Krueger, M. *Artificial Reality II*. Addison-Wesley, Reading, Massachusetts, 1991.

Pimentel, K. & K. Teixeira. *Virtual Reality: Trough the New Looking Glas*. Windcrest Books, TAB imprint, McGraw-Hill, New York, 1993.

Salvadori, M. *Why Buildings Stand Up: The Strength of Architecture*. W. W. Norton & Company, New York, 1990.

Blanchard, C., S. Burgess, Y. Harvill, J. Lanier & A. Lasko. *Reality Build for Two: A Virtual Reality Tool*. ACM SIGGRAPH 1990

Csuri, Ch. A. The Art of Artificial Reality. *IEEE Computer Graphics & Applications*, Vol. 11, 1991. Department of Computer Science, University of North Carolina, Chapel Hill, N. C.

Holloway, R., H. Fuchs & W. Robinett. Virtual-Worlds Research at The University of North Carolina at Chapel Hill, *Computer Graphics '91*, London U. K., 1991.

Ramos, P. M. *Virtual Interface Environment Workstation (VIEW), a Virtual Reality System*. UCLA 1991.

## Referencias Electrónicas

Ciudad virtual que ayuda a conocer la realidad  
<http://www.civila.com/ciudades/reales.asp>

Que es realidad virtual. Proyectos

[www.geocities.com/CapeCanaveral/Lab/3925/index.html](http://www.geocities.com/CapeCanaveral/Lab/3925/index.html)

Aplicaciones de la realidad virtual a la realidad real

[www.geocities.com/CapeCanaveral/Lab/3925/apli.htm](http://www.geocities.com/CapeCanaveral/Lab/3925/apli.htm)

Club de realidad virtual

<http://urania.eafit.edu.co/manual/vhosts/index.html> (en inglés).

Realidad virtual y Superscape Software

<http://www.superscape.com> (en inglés)

Página de animación

<http://animation.about.com>

Ensayo sobre virtualización de la realidad.

[www.funredes.org/liendo/charlas/virtual/virtual2.htm](http://www.funredes.org/liendo/charlas/virtual/virtual2.htm)

Enlaces a realidad virtual

<http://www.interaccess.cl/eurrutia/virtual.htm>

Página mexicana sobre realidad virtual y sus aplicaciones

<http://cecusac.gdl.iteso.mx/virtual/01a.html>

Clemson University, VR research and activities.

<http://fantasia.eng.clemson.edu/vr>

Georgia Institute of Technology, The Graphics, Visualization and Usability Center

<http://www.gatech.edu/gvu/research.html>

Ian's VR Buying Guide.

<http://www.cs.jhu.edu/~feldberg/vr/vrbg.html>

Michigan University, Virtual Reality Laboratory (VRL)

<http://www-vrl.umich.edu/>

Rutgers CAIP Virtual Reality Lab

<http://www.caip.rutgers.edu/vrlab>

University of Washington, The Human Interface Technology Lab

<http://www.hitl.washington.edu/>

Georgia Tech, The Interactive Visualizer

<http://www.ce.gatech.edu/Projects/IV/iv.html>

VRML polyhedra, Virtual Polyhedra

<http://www.li.net/~george/virtual-polyhedra/vp.html>

Mississippi State Virtual Environment/Interactive Systems Program

<http://www.erc.msstate.edu/vr>

Montgomery Blair High School Virtual Walkthrough

<http://www.mbhs.edu/~dpeck/mwalk/mwalk.html>

The MR Toolkit

<http://web.cs.ualberta.ca/~graphics/MRToolkit.html>

VENUS the Virtual Environment Navigation in the Underground Sites

<http://sgvenus.cern.ch/VENUS/>



# GLOSARIO

## **Algoritmos Genéticos (Genetic Algorithm)**

Son algoritmos matemáticos aplicables a problemas de optimización, basados en la teoría de la evolución de Darwin, operando en un ciclo simple de selección y reproducción, implicando una recombinación y mutación del "material genético" de las soluciones. Una "población" de posibles soluciones se genera al azar, se evalúan con respecto a un objetivo y las más aptas se combinan entre sí para producir nuevas soluciones. El ciclo se repite hasta llegar a una solución aceptable o al determinarse el óptimo de una función.

## **Base de Conocimiento (Knowledge Base)**

En inteligencia artificial, una recopilación del conocimiento y reglas sobre un tema específico.

## **C**

Lenguaje de alto nivel usado para la programación de sistemas, estructurado en bloques y con grandes facilidades para controlar las máquinas al nivel de hardware.

## **Conversor (Transducer)**

Es un equipo que convierte una forma de energía en otra forma de energía. Por ejemplo, un "transducer" puede acompañarse de un amplificador que convierte o transduce electricidad en sonido.

## **Minería de Datos (Data Mining)**

Son sistemas de búsqueda de conocimiento, tratando de determinar relaciones y patrones de comportamiento en bases de datos, de una forma inteligente y automática.

## **Fractales (Fractals)**

Modelos matemáticos para describir la naturaleza irregular de líneas, planos o volúmenes. Se pueden aplicar para representar modelos de datos.

**Inteligencia Artificial** Es el campo de la ciencia de la computación dedicado a analizar y desarrollar sistemas que reproduzcan e imiten los procesos de pensamiento y razonamiento del hombre.

## **LISP (List Processing)**

Un idioma de programación basado en lógica de predicado y uno de los más usados en aplicaciones de inteligencia artificiales.

## **Lógica Deductiva (Deductive Logic)**

Lógica tradicionalmente usada en sistemas expertos, definiendo un método para el razonamiento que parte de un concepto general para llegar a otros específicos.

## **Lógica Difusa (Fuzzy Logic)**

Es un conjunto de técnicas matemáticas para la representación y tratamiento de datos que no tienen una precisión definida y concreta. La evaluación de este tipo de datos es a menudo una cuestión relativa (pe. alto / bajo)

## **Lógica Inductiva (Inductive Logic)**

Lógica donde la agrupación de reglas que describen comportamientos particulares conducen a una regla general

### **Método Heurístico (Heuristic Method)**

Resolución de problemas, probando diferentes métodos y comparando cual es el que ofrece la mejor solución.

### **Percepción (Perception)**

Es un proceso de organización e interpretación de información sensorial que se lleva a cabo en el cerebro y cuyo propósito es brindar significado a la información que ingresa por los sentidos. Tanto la sensación como la percepción son procesos inseparables. Cuando el cerebro recibe información sensorial de los nervios aferentes, por ejemplo, dicha información es automáticamente interpretada. Por tanto, muchos psicólogos se refieren a la sensación y a la percepción como un sistema unificado de procesamiento de información. El medio ambiente es un lugar lleno de significados, sonidos, olores, y tacto. En este sentido es importante que dentro de la experiencia sensorial se cuente con la capacidad de detectar y discriminar estímulos.

### **Programación Genética (Genetic Programming)**

En inteligencia artificial, es una forma de desarrollo para generar programas automáticamente, partiendo de un juego de instrucciones primitivas.

### **Realidad (Reality)**

Existencia real y efectiva de una cosa. Verdad, ingenuidad, sinceridad.

### **Realidad Virtual (Virtual Reality)**

Un ambiente creado por una computadora y donde una persona puede experimentar utilizando equipo especial.

### **Redes Neuronales (Neural Networks)**

Son programas de Inteligencia Artificial capaces de simular algunas de las funciones de aprendizaje del ser humano. Sin reglas convencionales, una red neuronal obtiene experiencia analizando automática y sistemáticamente una cantidad de datos, para determinar reglas de comportamiento. Sobre la base de estas reglas, puede realizar predicciones sobre nuevos casos. Estas técnicas se aplican a problemas de clasificación, reconocimiento de patrones y series de tiempo, ofreciendo el potencial de identificar conexiones que otras técnicas no pueden, porque utiliza relaciones lineales y no-lineales entre los datos, puede trabajar con cualquier tipo de distribución (no solamente distribución normal) y maneja datos con redundancia y/o inconsistencia en la información

### **Sensación (Sensation)**

Es el proceso de detección y codificación de estímulos provenientes del mundo. Los estímulos emiten energía física -por ejemplo, luz, sonido, y calor-. Los órganos de los sentidos detectan esta energía y la transforman, o "transducen", en códigos que pueden ser transmitidos al cerebro. El primer paso en las sensaciones se encuentra en las células receptoras, las cuales responden a ciertas formas de energía. En este sentido, la retina del ojo es sensible a la luz, y las células ciliares del oído son sensibles a las vibraciones que generan los sonidos. La energía física es transformada a impulsos eléctricos; la información que lleva estos impulsos eléctricos viajan por las fibras nerviosas que conectadas los órganos de los sentidos con el sistema nervioso central. La información acerca del mundo externo viaja para apropiarse de áreas de la corteza cerebral.

**Sistemas de Clasificación (Classifier Systems)**

En inteligencia artificial, estos sistemas ejecutan un tipo de aprendizaje que genera reglas a partir de ejemplos.

**Sistemas Determinísticos (Deterministic Systems)**

Sistemas en los que el resultado es determinado por una ecuación, donde la causa y sus efectos son fácilmente reconocibles.

**Sistemas Expertos (Expert Systems)**

Los sistemas expertos permiten el desarrollo de otros sistemas que representan el conocimiento como una serie de reglas. Las distintas relaciones, conexiones y afinidades sobre un tema pueden ser compilados en un sistema experto pudiendo incluir relaciones altamente complejas y con múltiples interacciones.

**Teoría del Caos (Chaos Theory)**

Teoría que describe el comportamiento de sistemas no lineales, basada en que la dinámica y el comportamiento irregular de un conjunto complejo de datos, tiene un orden subyacente. Comprende un conjunto de técnicas para examinar y tratar de determinar las relaciones entre datos que han sido inicialmente clasificados como al azar. Una característica de este tipo de datos es que una diferencia menor en las condiciones iniciales puede resultar en cambios muy grandes en los resultados, en un período muy breve. También muestra la tendencia de los resultados a oscilar alrededor de un área en oposición a estabilizarse en un nivel de equilibrio. Por ejemplo, este comportamiento está presente en los mercados financieros, donde es imposible un entendimiento comprensible de los factores que influyen y hay un gran número de participantes con distintos objetivos.

**Virtual**

Que tiene "virtud" para producir un efecto, aunque no lo produce de frecuente. Implícito, tácito. Que tiene existencia aparente y no real.

**Virtud**

Actividad o fuerza de las cosas para producir o causar sus efectos.