

03081
6



**INTERACCION DINAMICA EN SISTEMAS DE
ENSEÑANZA INTELIGENTES**

Ana Lilia Concepción Laureano Cruces

TESIS DOCTORAL

PRESENTADA AL INSTITUTO DE INVESTIGACIONES BIOMEDICAS DE LA
UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

COMO REQUISITO PARA OBTENER EL GRADO DE DOCTORA EN
INVESTIGACION BIOMEDICA BASICA

CIUDAD UNIVERSITARIA
2000

DIRECTORES

Dr. José Negrete Martínez y Dr. Fernando de Arriaga Gómez *

273111



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

COMITE TUTORIAL

Dra. Sandra Castañeda Figueiras^{**} y
Dr. Enrique Ruiz-Velasco Sánchez^{***}

ASESORA EXTERNA

Dra. María del Carmen García-Alegre Sánchez^{****}

DICIEMBRE, 1997

* Departamento de Matemática Aplicada a las Tecnologías de la Información
UNIVERSIDAD POLITECNICA DE MADRID, ESPAÑA.

** Facultad de Psicología - UNAM

*** Centro de Estudios Sobre la Universidad - UNAM

**** Instituto de Automática Industrial - Consejo Superior de Investigaciones
Científicas (CSIC), Madrid, ESPAÑA.

Dedicatorias

A mi papá, por haberme construido una sólida cimentación i anti-sísmica !

A mi mamá, porque cada paso que he dado, lo ha hecho suyo.

A mis hermanas Reina, Grisel y Diana, mi equipo favorito y sin el cual, este quarterback, no hubiera podido ganar el partido. Que dicho sea de paso se ganó en 'muerte súbita' (Interactive Learning Environments).

A Javier, por todo su amor e incansable apoyo y por ser quien diariamente endulzó y continúa endulzando mi vida con su paz.

A Emiliano, mi pequeño hijo, por su paciencia, quien a su corta edad no se entera de nada, sólo de ausencias y esperas.

Reconocimientos

Al Dr. José Negrete, por haber sembrado la semilla y por su apoyo a lo largo del trayecto.

A la Dra. Sandra Castañeda, por haberme descubierto y guiado a través de una nueva perspectiva de la inteligencia artificial.

A Enrique Ruiz-Velasco, por compartir conmigo sus conocimientos.

A Fernando de Arriaga, por todo su entusiasmo y trabajo durante año y medio, en nuestras sesiones semanales, por su amistad y por haber creído en mi.

A María García-Alegre, por arrancarle gotitas al tiempo y siempre escucharme y ayudarme.

A mi valioso equipo de software formado por Santiago Arriaga y Mauricio Martínez.

Al Lic. Edmundo Jacobo Molina, que siendo Rector de la UAM-A, apoyó la parte académica de esta tesis.

A mis amigas Isabel Quintas, Angela Ribeiro, Montserrat Parrilla y Teresa Sánchez, por el tiempo empleado en la lectura del manuscrito y sus valiosos consejos.

A mis amigos, Rafael López Bracho, Germán Téllez, Angel Hernández, Clarita Durán y a todos aquellos que me apoyaron durante los años que duró el proyecto.

INTERACCION DINAMICA EN SISTEMAS DE ENSEÑANZA INTELIGENTES

Resumen:

Se presenta una forma de análisis y diseño de un sistema de enseñanza inteligente (SEI), basado en los sistemas reactivos y los modelos cognitivos; lo que conduce a una arquitectura multi-agente, donde el comportamiento global del sistema reactivo es modelado tomando en consideración los resultados del análisis cognitivo de tareas (ACT), la gráfica genética (GG) y los errores de la tarea a desarrollar.

La característica principal de una arquitectura reactiva es que modela el problema de forma "bottom-up", en forma contraria a como lo hace la inteligencia artificial tradicional, "top-down". Los SEI, clásicos trabajan con agentes cognitivos y su comportamiento global es basado en objetivos explícitos, así como una representación centralizada del entorno. Por el otro lado están los agentes reactivos, que no cuentan con una representación centralizada del entorno, y actúan con base en un modelo de estímulo-respuesta, pudiendo de esta forma responder de acuerdo con el estado del entorno en el cual están embebidos.

Con esta propuesta se pretende ofrecer un análisis más claro y puntualizado de los errores y de los planes de enseñanza que corresponden a las diferentes tácticas instruccionales manejadas por cada agente, donde cada agente será del tipo asistente experto.

Aún más la implementación con agentes reactivos es simple porque representan habilidades sencillas vistas de forma aislada, con lo que se puede uno beneficiar de todas las ventajas de la programación orientada a objetos (POO). La dificultad real radica en encontrar esas habilidades, así como el análisis y diseño de los mecanismos de interacción entre el entorno y el sistema reactivo, en este caso un SEI.

En este trabajo se importa la filosofía de diseño de los sistemas reactivos utilizada en los robots móviles, al diseño de SEI. Han sido utilizadas diferentes herramientas para el desarrollo del SEI, una de las cuales es el ACT. En la actual propuesta se muestra su potencial, para encontrar las habilidades de una tarea cognitiva (TC). Uno de los objetivos de cualquier sistema reactivo es lograr un comportamiento global emergente, donde emergente significa adaptación al entorno en curso; basándose en conductas simples. Pero en el caso de un SEI, el propósito es enseñar una habilidad compleja dividiéndola en sub-habilidades; de aquí que cada sub-habilidad este representada por un agente tipo experto asistente. Con base en interacciones entre los agentes y entre los agentes y el entorno, es que logramos ese comportamiento emergente y coherente. Para lograrlo tenemos aspectos del entorno y objetivos trabajando de forma distribuida en los diferentes asistentes expertos.

Se presenta como ejemplo un SEI, llamado Makatsiná¹, donde se utilizó esta técnica, que centra su enseñanza en el aprendizaje de la habilidad para resolver estructuras triangulares por el método de los nodos; este dominio se ubica dentro de habilidades integradas.

¹ Makatsiná, significa tutor en totonaca, una lengua mexicana prehispánica

INTELLIGENT TUTORING SYSTEMS WITH DINAMIC INTERACTION

Abstract:

This thesis presents a new approach to the analysis and design of intelligent tutoring systems (ITS), based on reactive principles and cognitive models, this way leading to multi-agent architecture.

In the reactive models, the analysis problem is treated in a bottom-up manner, as opposed to that of traditional artificial intelligence (AI), i.e., top down. We present one ITS example called Makatsiná (meaning tutor in totonaca, a Mexican pre-Columbian language) constructed according to this approach, which teaches the skills necessary to solve the truss analysis problem by the method of joints.

This learning domain is an integration skill. The classical ITS work is based on explicit goals and an internal representation of the environment. The new approach has reactive agents which have no representation of their environment and act using a stimulus/response behavior type.

In this way they can respond to the present state of the environment in which they are embedded. With these elements, errors and teaching plans, each agent behaves as an expert assistant that is able to handle different teaching methods.

Reactive agent programming is found to be simple because agents have simple behaviors. The difficulty lies in the interaction mechanism analysis and design between the environment and the intelligent reactive system.

This work translates the design philosophy for reactive systems, used in mobile robots, to the design of intelligent tutoring systems (ITS). Different tools have been chosen to develop this ITS methodology; one of them is the cognitive task analysis (CTA), whose potential to find the cognitive task (CT) skills is shown in this work. One of the reactive system objectives is to find an intelligent global behavior based on simple conducts, but the main ITS purpose is to teach a complex skill which has been divided into subskills. As a consequence, each subskill is represented by an expert assistant to get an intelligent interaction among the experts and the environment to finally exhibit a global coherent ITS behavior. To get it we have only the environment features and objectives distributed among the expert assistants.

We present one ITS example called, Makatsiná, which teaches the skill necessary to solve the truss analysis problem by the joints method; this domain is an integration skill.

Curriculum Vitae de una Idea

Después de haber recorrido varias Instituciones de la Ciudad, buscando opciones para iniciar mis estudios de doctorado, llegué con el Dr. Negrete, quien amablemente me recibió y escuchó. Una vez conocida la historia de mi trabajo me dijo que *la idea* que tenía era; la de aplicar la filosofía de los agentes reactivos utilizados en robótica a los sistemas de enseñanza inteligentes. Así lo hice y preparé (supervisada por él) lo que era *la idea* de investigación para ser presentada al comité de admisión nombrado por el Subcomité del Instituto de Investigaciones Biomédicas en turno.

Posteriormente estuve en el Departamento de Biomatemáticas, en el Instituto de Investigaciones Biomédicas; mis años de créditos obligatorios. Trabajé arduamente durante este periodo (3 años). Faltaba el último requisito; el que mi comité tutorial aprobara los resultados y me diera el visto bueno para comenzar a escribir *la tesis y la preparación de al menos una comunicación técnica o científica*; este requisito lo cubrí el 14 de octubre de 1996, cuando presenté el sistema de enseñanza inteligente trabajando; llamado Makatsiná, el Dr. Negrete lo bautizó con ese nombre *totonaca* que significa exactamente tutor, además del último informe de avance de proyecto. Había cumplido ya todos los requisitos previos a la escritura de la tesis.

Para llegar a este punto muchas personas participaron, además del Dr. Negrete. La Dra. Sandra Castañeda de reconocido prestigio en su área, me enseñó cómo llevar a cabo el análisis cognitivo de tareas y no solo me animó a escribir artículos sobre el tema sino que me supervisó en la difícil tarea de encontrar las sub-habilidades de una habilidad integrada, utilizando su método sobre análisis cognitivo de tareas. El Dr. Enrique Ruiz-Velasco estudió un doctorado en Robótica Pedagógica en Ottawa y siempre me aconsejó sobre las mejores técnicas didácticas y obviamente con literatura sobre el tema.

Además tuve la oportunidad de trabajar con una encantadora persona que a pesar de estar siempre ocupada nunca me negó sus conocimientos y cooperación, él es el *experto en el domino* el Ing. Miguel Angel Rodríguez Vega; Jefe del Departamento de Estructuras de la Facultad de Ingeniería y de acuerdo a una encuesta personal, uno de los mejores profesores del área de estructuras, su método innovador quedó representado en Makatsiná. El me dio la oportunidad de acudir a sus clases, aprender el nuevo método, me facilitó exámenes de los que se pudieron obtener los errores comunes así como los de clase. En esta etapa participó una alumna brillante del área, Alejandra Medina Arévalo, quien de acuerdo al método mencionado y bajo mi supervisión, desarrolló los problemas que posteriormente fueron codificados en el sistema de enseñanza. El Ing. Rodríguez Vega también me indicó donde poner los puntos críticos dentro de dicho sistema y su consulta continuó a lo largo del desarrollo de Makatsiná. El fue uno de los primeros evaluadores del citado sistema trabajando.

Finalmente desarrollamos Makatsiná para lo cual reuní un pequeño, pero, poderoso equipo de exalumnos míos: Santiago Arriaga Noguez y Mauricio Martínez Medina; y así aplicando las técnicas de: tipos abstractos de datos y programación orientada a

objetos se desarrolló Makatsiná. Cabe señalar que el sistema de enseñanza inteligente se desarrolló en año y medio.

El Dr. Negrete estuvo al pendiente de cada paso y logro mío, no permitiendo que me saliera ni un milímetro de *la idea* en su más alto nivel de abstracción, él sabía cómo debían trabajar los agentes en un nivel reactivo y me supervisaba en este sentido, dos veces me salí y pensé en representaciones globales o representaciones compartidas y el Dr. Negrete en sesiones específicas me reorientaba. Otras tantas veces perdía el rumbo y él me volvía a ubicar.

Finalmente en una sesión *estrella* después de analizar y analizar el modelo mental del experto; dedujimos que el sistema de enseñanza inteligente era un tutor de habilidades y no de conocimiento como hasta entonces lo habíamos estado enfocando. Esta conclusión era de vital importancia; ya que de esto dependía la forma y el conocimiento que se debía extraer del *análisis cognitivo de tareas*. Acudí a un seminario dirigido por él donde nos enseñó a mí y a otros, los principios de la reactividad de agentes, ahí analizamos todo un libro. De él aprendí entre otras cosas, todo lo referente a robótica reactiva; cabe mencionar que en México el Dr. Negrete es pionero en el tema.

Encontré financiamiento para salir del país y escribir mi tesis, pero... había otro problema que consistía en encontrar quién se interesara en *una idea* ya finalizada y que quisiera involucrarse, estudiar y ponerse al tanto de todo lo referente *a la idea*. Toqué tres puertas y afortunadamente en la tercera estaba una maravillosa persona detrás del escritorio, que después de escucharme atentamente me dijo con todo gusto la ayudo en el ámbito intelectual, ya que económicamente me es imposible, esa persona es el Dr. Fernando de Arriaga Gómez.

Así fue como entró en escena el Dr. Arriaga; Director del programa de Doctorado en Inteligencia Artificial y Sistemas Expertos de la Escuela Técnica Superior de Ingeniería en Telecomunicaciones de la Universidad Politécnica de Madrid; quien con mucho respeto y seriedad tanto a mi persona, como a *nuestra idea*; leyó artículos y se documentó. Con él completé la formalización matemática de mi software, me ayudó a redondear la escritura de mi tesis, y participó en la escritura de mis artículos.

Detrás de todo este trabajo existe otro equipo valiosísimo que nunca dejó de trabajar y esta constituido por: mis amigos (imposible nombrar a todos), mis queridas hermanas, mi papá, mi mamá, y Javier; siempre corriendo y entregando papeles y sobre todo siempre dándome ánimo, ellos tenían muchas veces más fe que yo misma, en que algún día concluiría esta ardua tarea de 4 años y más...

A todos ellos mil gracias por fin se cumplió mi sueño...

Lili

Indice

Indice	1
Indice de tablas y figuras	v
Capítulo I	7
Introducción	7
I. 1 Los Sistemas de enseñanza inteligentes	7
I. 2 Premisas	10
I. 2. 1 Los Sistemas de enseñanza inteligentes	10
I. 2. 2 La inteligencia artificial distribuida	10
I. 2. 3 Los agentes reactivos	11
I. 3 Motivaciones	11
I. 3. 1 Aplicaciones	11
I. 3. 2 Un sistema de enseñanza reactivo	11
I. 3. 3 Descomposición de habilidades	11
I. 3. 4 Metodología de análisis y diseño	12
I. 3. 5 Formalización de agentes	12
I. 4 Objetivos	12
I. 4. 1 Crear una nueva arquitectura	12
I. 4. 2 Proponer una forma de análisis y diseño	13
I. 4. 3 Ofrecer un conocimiento detallado en habilidades	14
I. 4. 4 Combinar el uso de un tutor tipo entrenador deportivo con características reactivas	14
I. 4. 5 Proponer una forma de control de calidad de software, en base al comportamiento deseado de los agentes	15
I. 4. 6 Proponer un mecanismo de arbitraje par a los SEIs	15
I. 5 Estructura de la memoria	16
Capítulo II	17
Técnicas de diseño en Sistemas de Enseñanza Inteligentes	17
II. 1 Introducción	17
II. 2 El módulo del dominio o módulo experto	18
II. 2. 1 Experto caja negra	18
II. 2. 2 Experto caja de cristal	20
II. 2. 3 Los modelos cognitivos	20
II. 2. 4 Diferentes tipos de conocimiento	21
II. 2. 5 Representación del conocimiento	22
II. 3 El modelo de estudiante	23

II. 3. 1 Las dimensiones del espacio a ser consideradas en el modelo del estudiante	23
II. 3. 2 Clasificación del modelo de estudiante de acuerdo a distintas características	26
II. 3. 3 Modelos individualizados implícitamente	28
II. 3. 4 Técnicas de diagnóstico	30
II. 4 Módulo tutorial	32
II. 4. 1 Aspectos centrales	32
II. 4. 2 Curriculum	37
II. 4. 3 El proceso tutorial (enseñanza)	39
II. 5 La Interfaz	44
II. 5. 1 La ayuda	44
II. 5. 2 Multimedia y cognición	46
II. 6 Trabajos relacionados	48
II. 6. 1 Arquitecturas MultiAgente	48
II. 6. 2 Agentes pedagógicos	50
II. 7 Nuestra aproximación	52
II. 7. 1 Metodología	52
Capítulo III	55
Agentes reactivos	55
III. 1 Introducción	55
III. 1. 1 Agentes deliberativos	56
III. 1. 2 Agentes reactivos	56
III. 1. 3 Aproximación de las arquitecturas reactivas	58
III. 2 Antecedentes	59
III. 2. 1 Ideas de Brooks sobre cómo lograr un comportamiento inteligente con agentes autónomos [BROR91](2)	59
III. 3 Arquitecturas reactivas	63
III. 3. 1 Arquitectura de Brooks	63
III. 3. 2 Arquitectura de Garcia-Alegre, Bustos & Guinea [GARM95]	66
III. 3. 3 Arquitectura de Kaelbling	70
III. 4 Propiedades de un sistema altamente reactivo	75
III. 4. 1 Reactividad	75
III. 4. 2 Trabajo de equipo	75
III. 4. 3 La robustez	76
III. 5 Elementos que constituyen a los SEIs y a los robots reactivos	76
III. 5. 1 Master	76
III. 5. 2 Entorno	76
III. 5. 3 Robustez conductual y perceptual	77
III. 5. 4 Percepción e interpretación del entorno	78
III. 5. 5 Reactividad	79
III. 5. 6 Arbitros	79
III. 5. 7 Flexibilidad	81
III. 5. 8 Modularidad	81
III. 5. 9 Ubicación y funcionalidad emergente	82
III. 5. 10 Los niveles y la granularidad conductual	82
III. 6 Conclusiones con respecto a los SEIs y los agentes reactivos	83

III. 6. 1 ¿Dónde se implementará la reactividad?	83
III. 6. 2 Distintas formas de diseño de robots no reactivos.....	83
III. 6. 3 Diseño de la conducta básica del agente reactivo.....	84
III. 6. 4 Análisis cognitivo de tareas	85
III. 6. 5 El Control de los sistemas reactivos	85
Capítulo IV	87
Análisis y diseño del dominio	87
IV. 1 Introducción	87
IV. 1. 1 Problemas básicos.....	87
IV. 1. 2 Definiciones básicas	88
IV. 2 Antecedentes	88
IV. 2. 1 Análisis cognitivo de tareas (ACT)	89
IV. 3. Metodología	99
IV. 3. 1 El dominio de Makatsiná.....	100
IV. 3. 2 Por qué un solo nivel de competencia	104
IV. 3. 3 Número de habilidades	105
IV. 3. 4 Habilidades que se detectan.....	107
IV. 4 Reflexiones sobre el dominio cognitivo y su enseñanza.....	109
IV. 4. 1 Experto Vs. Novato	110
IV. 4. 2 Representación del conocimiento.....	111
IV. 5 Aspectos cognitivos en la instrucción	113
IV. 5. 1 La didáctica del conocimiento procedimental	113
Capítulo V	115
Formalización de los agentes de Makatsiná	115
V. 1 Introducción.....	115
V. 1. 1 Modelos formales.....	115
V. 1. 2 Asistentes expertos.....	116
V. 1. 3 Robots autónomos.....	117
V. 1. 4 Softbots	117
V. 1. 5 Agentes sintéticos	117
V. 2 Definiciones.....	117
V. 2. 1 Definiciones básicas.....	118
V. 2. 2 Definiciones de los esquemas de un Agente.....	120
V. 3 Ingeniería de software.....	120
V. 3. 1 Arquitectura MultiAgente en software	121
V. 3. 2 Organización de un sistema interactivo	121
V. 3. 3 Modelo MultiAgente.....	122
V. 4 Agentes de Makatsiná.....	123
V. 4. 1 Percepción de un agente.....	125
V. 4. 2 La Acción de un agente.....	126
V. 4. 3 Definición de los esquemas de los agentes de Makatsiná.....	126
V. 5 Tratamiento de los errores	130
V. 5. 1 Clasificación de los errores.....	130
Capítulo VI	131
Ejemplos del desarrollo de Makatsiná	131

VI. 1 Introducción	131
VI. 2 Ejemplo 1 en el modo de resolución	135
VI. 3 Ejemplo 2 en el modo de resolución	139
VI. 4 Ejemplo 3 en el modo de autoresolución	149
VI. 5 Programación de Makatsiná	152
Capítulo VII.....	153
Conclusiones y trabajos futuros	153
VII. 1 Resumen.....	153
VII. 2 Resultados y Aportaciones.....	154
VII. 3 Trabajos Futuros	155
VII. 3. 1 La robustez conductual y perceptual de Makatsiná	155
VII. 3. 2 Diagnóstico de errores.....	156
VII. 3. 3 El impacto pedagógico de Makatsiná.....	156
VII. 3. 4 La generalización de la técnica propuesta de análisis y diseño del SEI MultiAgente.....	156
VII. 3. 5 La generalización de la arquitectura de software para construir un SEI MultiAgente.....	157
REFERENCIAS	159

Indice de tablas y figuras

Figura I. 1 – Dominio de los SEIs.....	9
Figura I. 2 – Componentes de un SEI.....	10
Tabla II. 1 - Técnicas de Diagnóstico propuesta por VanLehn [VANK88].....	31
Tabla II. 2 - Estrategias de Diálogo de acuerdo a diferentes objetivos instruccionales según Collins [HALH88]	40
Figura II. 1 - Anatomía de un asunto.....	41
Figura II. 2 - Modelo Tutorial con asuntos y ejemplos de Burton y Brown [BURR82] Parte 1(a).....	42
Figura II. 2 - Modelo Tutorial con asuntos y ejemplos de Burton y Brown [BURR82] Parte 1(b)	43
Figura III. 1 - Arquitectura-Incluida (subsumption- architecture)	65
Figura III. 2 - Arquitectura Jerárquica MultiAgente	68
Figura III. 3 - Flujo y Tipos de Representación en Percepción y Acción	72
Figura III. 4 - Descomposición Horizontal	73
Figura III. 5 - Control Jerárquico Adaptativo.....	74
Figura IV. 1 - Modelo Mental para la Solución de Estructuras Triangulares por el Método de los Nodos.....	97
Figura IV. 2 - Tipos comunes de estructuras triangulares tipos Comunes de estructuras triangulares	101
Figura IV. 3 - Diagrama de cuerpo libre de los nodos.....	102
Figura IV. 4 - Estructuras con más de tres reacciones.....	102
Figura IV. 5 - Estructura con tres reacciones.....	103
Figura IV. 6 - Diagrama de Cuerpo Libre de los Nodos.....	103
Tabla IV.1 - Especificación del ACT de la tarea análisis de estructuras triangulares tomando en cuenta las habilidades	105
Figura IV. 7 - Diagrama conceptual del dominio (basándonos en la GG)	106
Tabla IV. 2 - ACT de la Solución de Estructuras Triangulares (basándonos en Ryder & Redding).....	107
Figura IV. 8 - Reactividad del Módulo Tutorial.....	109
Tabla IV. 3 - Errores críticos para controlar la intervención de los SUB-TUTORES (obtenidos a partir del desarrollo global de la TC).....	112
Figura V. 1 - Arquitectura para un Sistema Interactivo.....	121
Figura V. 2 - Jerarquía de Agentes.....	122
Figura V. 3 - Anatomía de un Agente.....	123
Figura V. 4 - Reactividad del Módulo Tutorial.....	124
Figura V. 5 - Agente Global Del Sistema	125
Figura V. 6 - Modelo Funcional (Parte -1-).....	126
Figura V. 7 - Modelo Funcional (Parte -2-).....	127

Figura VI. 1	132
Figura VI. 2	133
Figura VI. 3	134
Figura VI. 4	136
Figura VI. 5	136
Figura VI. 6	137
Figura VI. 7	137
Figura VI. 8	138
Figura VI. 9	138
Figura VI. 10	139
Figura VI. 11	140
Figura VI. 12	140
Figura VI. 13	141
Figura VI. 14	141
Figura VI. 15	142
Figura VI. 16	143
Figura VI. 17	143
Figura VI. 18	144
Figura VI. 19	144
Figura VI. 20	145
Figura VI. 21	145
Figura VI. 22	146
Figura VI. 23	146
Figura VI. 24	147
Figura VI. 25	147
Figura VI. 26	148
Figura VI. 27	148
Figura VI. 28	149
Figura VI. 29	150
Figura VI. 30	150
Figura VI. 31	151
Figura VI. 32	151
Figura VI. 33	152

Capítulo I

Introducción

I. 1 Los Sistemas de enseñanza inteligentes

Los sistemas de enseñanza inteligentes (SEI) son interesantes debido a que simulan y hacen explícita una clase importante de inteligencia humana; y es la enseñanza en el nivel de abstracción más alto. Los sistemas de enseñanza inteligentes (SEIs) pueden ser vistos como agentes activos que adaptan sus estrategias de enseñanza basándose en los cambios que percibe del estudiante, cuando este último se encuentra en un proceso de aprendizaje. La descripción de las estrategias de enseñanza, así como el control, son aspectos que tienen un papel importante en la construcción de SEIs.

Los primeros sistemas de enseñanza que hicieron su aparición fueron los llamados Computer Aided Instruction (CAI). Estos últimos han evolucionado desde sus inicios a la fecha; utilizando diferentes técnicas que faciliten su uso y los vuelvan más atractivos.

La incorporación de las técnicas de inteligencia artificial (a partir de los 70) a estos sistemas dieron origen a los sistemas de enseñanza inteligentes. El término de inteligencia se asocia a la capacidad de adaptación dinámica a diferentes tipos de estudiantes. Lo anterior lo logran incorporando las mencionadas técnicas en: 1) el conocimiento que el sistema tiene del dominio, 2) los principios del proceso tutorial y los métodos bajo los cuales son aplicados, y 3) la representación del conocimiento que supuestamente tiene el usuario.

Los SEIs enfocan el proceso de aprendizaje como una cooperación entre el tutor y el alumno. El tutor basándose en la percepción del alumno decide en cada momento que estrategia es adecuada. Estas estrategias serán elegidas en base a la medida de una serie de parámetros como: errores cometidos, estilo de aprendizaje, conocimientos dominados, etc. Lo anterior para poder decidir: qué explicar, con que nivel de detalle, cuándo y cómo interrumpir al alumno.

La investigación en el área de los SEIs, ha estado centrada en diferentes aspectos, desde la inspección de cada uno de sus módulos constituyentes ([GOLI79], [URRM91], [WOOB92], [SLED85], [FERI89]), hasta la elaboración de arquitecturas genéricas [FERI89]. Así como el aprendizaje automático y la construcción de ayudas de diseño de sistemas de enseñanza [ARRA98].

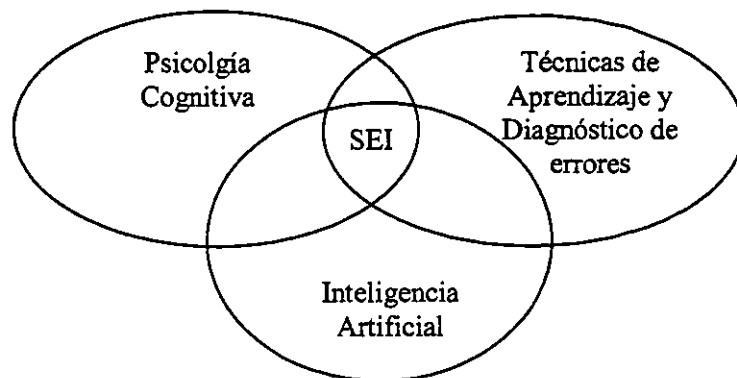
El dominio de SEIs se encuentra en la intersección de tres diferentes áreas: 1) las ciencias de la computación (inteligencia artificial), 2) la pedagogía (recursos educacionales) y 3) la psicología cognitiva (métodos de análisis de los diferentes procesos cognitivos), ver Figura I.1. Además se encuentra el área del conocimiento que se pretenda enseñar a través del sistema.

En el caso de los recursos educacionales la contribución a los SEIs se refiere a las técnicas de aprendizaje que soportan el proceso de resolución de problemas y así poder diagnosticar los errores del estudiante. En el caso de diagnóstico de errores, nos permite saber de acuerdo a los errores que comete que ha o no ha comprendido y crear estrategias y tácticas remediales. En el caso de las técnicas de aprendizaje nos permite implementar los estilos de aprendizaje más apropiados, así como las estrategias y tácticas instruccionales.

En el caso de las ciencias de la computación la contribución se centra básicamente en las técnicas de inteligencia artificial que el caso de los ITSs pueden ser incluidas en: 1) el conocimiento que el sistema tiene del dominio (motivo de la enseñanza) y 2) los principios del proceso tutorial, y los métodos bajo los cuales son aplicados estos principios. Sin embargo, es importante mencionar que la implementación de sistemas sofisticados como los sistemas que involucran inteligencia artificial se verían altamente beneficiados con la utilización de técnicas de formalización, y análisis y diseño pertenecientes a la ingeniería de software. Por lo anterior se considera que también existe una contribución importante en este sentido.

En el caso de la contribución de la psicología cognitiva, ésta es valorada cada vez más en el desarrollo de los SEIs. En esta rama se encuentra el estudio y esclarecimiento de los procesos cognitivos. Estudios con resultados invaluable cuando se trata de emular procesos humanos en sistemas con inteligencia artificial.

Figura I.1 - Dominio de los SEIs



A continuación se exponen brevemente las características de los componentes de un Sistema de enseñanza inteligente Figura I.2.

El módulo experto: es el lugar donde se encuentra acumulado el conocimiento que el sistema intenta enseñar al estudiante. La implementación de este componente está íntimamente ligada al módulo tutor. Debido a que el tutor enseñara el dominio haciendo énfasis en la organización del módulo experto. De aquí que interesa que este módulo esté organizado de forma pedagógica.

El modelo de estudiante: es una base de datos que contiene información del estudiante que permite desarrollar las siguientes funciones: 1) adaptación del sistema con base en la competencia que tenga el estudiante de un determinado material (objeto de la enseñanza), 2) hacer un reporte del material cubierto de acuerdo al currículum, 3) seleccionar el nivel adecuado de intervención y explicación, y 4) facilitar la retroalimentación del estudiante.

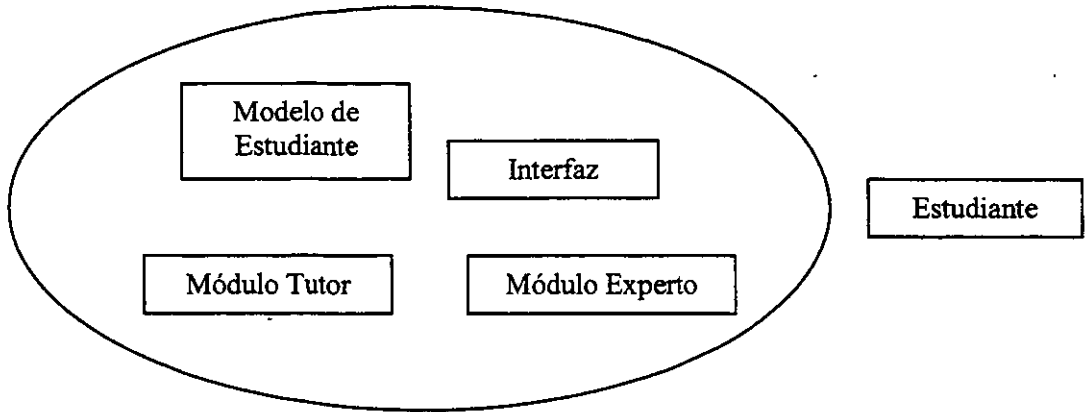
El módulo tutor: tiene la responsabilidad de decidir que acciones tomar para enseñar o corregir un determinado dominio basándose en la información del currículum y de los objetivos que el planificador² tiene; con respecto a uno o varios temas específicos a enseñar. Es él quien selecciona los problemas a mostrar al estudiante, analiza las respuestas, presenta la solución de ciertos problemas o decide mostrar algunos ejemplos. Gestiona el material didáctico y se encarga de seleccionar el material más adecuado en función de las situaciones reportadas. Estas situaciones son principalmente determinadas por las demandas del planificador y del comportamiento del alumno percibido a través de la interfaz.

La interfaz: puede ser considerada como un entorno de simulación en el sentido de que es el lugar donde tienen representación las salidas y entradas del sistema. Su responsabilidad

² Se entiende por planificador el lugar donde se encuentran implementados los métodos de control de la selección y secuenciación de las estrategias de enseñanza. Puede ser implementado como un componente aparte o como parte del módulo tutor. Dependiendo del mecanismo de control elegido.

básica es la comunicación entre el sistema y el estudiante, aunque al ser el medio de salida de las acciones del SEI, también tiene una responsabilidad didáctica.

Figura I. 2 - Componentes de un SEI



I. 2 Premisas

I. 2. 1 Los Sistemas de enseñanza inteligentes

Los sistemas de enseñanza inteligentes son sistemas que incorporan técnicas de inteligencia artificial en su desarrollo, con el propósito de emular una enseñanza personalizada. Para lograrlo estos sistemas requieren de un continuo proceso de percepción de la actividad del estudiante durante la evaluación de la solución de los problemas que se plantean. En el otro lado se encuentra la experiencia de un tutor que se basa en: el conocimiento de la materia que enseña, en los errores que cometen los alumnos, en las características de aprendizaje del estudiante, etc.

I. 2. 2 La inteligencia artificial distribuida

La inteligencia artificial distribuida (IAD) descentraliza el proceso de control; lo que lleva al uso de agentes autónomos. Estos agentes son especialistas que cooperan entre si, de forma contraria a como lo hacen los sistemas que cuentan con un control general. Con lo anterior se trata de minimizar en lo posible la incapacidad de planificar dentro de un mundo dinámico, al permitir la existencia de varios especialistas resolviendo un mismo problema en vez de uno [GIRJ92].

Dentro de la IAD se encuentran las arquitecturas MultiAgente. Los agentes con que se diseñen estos sistemas pueden ser de diversos tipos: agentes colaborativos, agentes interfaz,

agentes móviles, agentes informativos, agentes reactivos, agentes híbridos o agentes inteligentes [NWAH96]. En esta tesis se trabajará con agentes reactivos, así que serán las características de éstos los que estudiemos a lo largo del desarrollo de la misma.

I. 2. 3 Los agentes reactivos

Los agentes reactivos dotan a los sistemas de la capacidad de reaccionar con rapidez en un mundo dinámico y en continuo cambio. Para lograrlo cada agente sintetiza los aspectos de ese mundo que son relevantes para su funcionamiento. Con esta división se creará por otro lado, la interacción de estos agentes con el mundo y entre ellos para que el funcionamiento global del sistema emerja. Lo anterior de acuerdo a la tarea específica para la que fue creado el sistema.

Como consecuencia de lo anterior y pariendo de la idea: que un sistema de enseñanza inteligente es un sistema que opera en un ambiente: cognitivo, impredecible, complejo y por lo tanto difícil de modelar. Proponemos que sea visto como un sistema con características reactivas.

I. 3 Motivaciones

I. 3. 1 Aplicaciones

En el rango de aplicaciones de sistemas reactivos no se encontraban los SEIs. Las áreas de aplicación básicamente han sido: la robótica ([BROR86], [BROR91](1), [BROR91](2)) y algunos problemas de simulación y solución de problemas [FERJ92].

I. 3. 2 Un sistema de enseñanza reactivo

El uso de un SEI reactivo es interesante cuando se pretende enseñar una única estrategia, sea porque pertenece a un conjunto de estrategias aplicables a un determinado problema o porque interesa el aprendizaje de la mejor, la más rápida o la más sencilla. Este tipo de sistemas sería aplicable a estudiantes avanzados en un tema determinado. Se entrenaría al estudiante en la mejor solución; donde se exigiera optimizar el tiempo de ejecución. No funcionaría para principiantes.

I. 3. 3 Descomposición de habilidades

Lograr la descomposición de una habilidad integrada en subhabilidades e implementarla en un sistema MultiAgente; donde los agentes son reactivos, cuenta con tres tipos de ventaja

desde el punto de vista de la enseñanza: a) con base en los agentes reactivos se puede implementar un mayor detalle en las técnicas y estrategias tutoriales; al poder crear subexpertos en las distintas subhabilidades, b) en un SEI reactivo quedaría implícito el orden de aprendizaje de una estrategia. El hecho de no utilizar alguna subhabilidad en un determinado momento; donde ésta sea necesaria dentro de ese contexto, provocaría la reacción inmediata del sistema y c) en el caso que se pretenda enseñar no una sino varias estrategias pertenecientes a un mismo dominio, es muy probable que algunas de ellas compartan subhabilidades; con lo que se lograría una re-utilización de los agentes reactivos. Por lo tanto el *kid* del asunto es encontrar una metodología que permita la descomposición de habilidades integradas con el fin de desarrollar SEIs reactivos.

I. 3. 4 Metodología de análisis y diseño

Hasta el momento no existe una metodología clara para llevar a cabo el análisis y diseño de sistemas reactivos. Este trabajo propone una instrumentación con herramientas provenientes de la inteligencia artificial y la psicología cognitiva, para lograr traspasar el análisis y diseño reactivo llevado hasta el momento principalmente en robots móviles a los SEIs. Lo anterior con el fin de esclarecer el proceso y que este pueda ser probado en otro tipo de sistemas reactivos.

I. 3. 5 Formalización de agentes

Muchos intentos de formalización de agentes por separado han sido propuestos y la formalización de estos sistemas durante la etapa de diseño no ha sido abordada en su totalidad, en este trabajo se propone una forma de lograrlo basándonos en las investigaciones realizadas por Luck y d'Inverno [LUCM95](1), acompañada de una representación de agentes dentro de un sistema interactivo propuesta por Nigay y Coutaz [NIGL91]. El objetivo es poder lograr un seguimiento de los errores en las conductas y las mismas conductas, a través de los distintos agentes y sus comunicaciones. Así como un acercamiento desde la teoría a la implementación.

I. 4 Objetivos

I. 4. 1 Crear una nueva arquitectura

Crear una nueva arquitectura en el área de los SEIs relacionada con la corriente reactiva y basada en las arquitecturas de: Brooks [BROR86], Kaelbling [KAEL87] y García-Alegre, Bustos & Guinea [GARM95]. La filosofía medular de la robótica reactiva (RR) esta basada

en la idea de los agentes de la mente de Minsky [MINM87] y en los sistemas basados en la conducta, Beer [BERJ90].

Esta nueva corriente ha dado frutos precisamente en el campo de la robótica, donde se pretende crear modelos de robots autónomos, esto es, que puedan trabajar sin la guía del hombre. Esta corriente sostiene la tesis de lograr objetivos complejos apoyándose en un conjunto de conductas sencillas representadas por agentes.

Otro aspecto importante en el desarrollo de esta corriente es que no está de acuerdo con la modelización exhaustiva del mundo dentro del sistema seguida por la inteligencia artificial tradicional, ni con la representación explícita de objetivos dentro del sistema. Esta corriente arguye que el mejor modelo es el propio mundo.

I. 4. 2 Proponer una forma de análisis y diseño

Proponer una forma de análisis y diseño del dominio que se fundamenta en los principios de la reactividad y los modelos cognitivos, conduciéndonos a una arquitectura multi-agente. La parte medular de esta arquitectura, es que comienza a modelar el problema de forma 'bottom-up', en forma contraria a como lo hace la inteligencia artificial (IA) tradicional 'top-down', en otras palabras, organiza el funcionamiento global del sistema en agentes independientes con sus propios objetivos. Esta técnica comenzó a ser aplicada en la robótica reactiva en los 80 en el curso de su desarrollo ha ido dejando sus experiencias, así como los principios bajo los cuales se elaboran los sistemas basados en agentes reactivos. Esta corriente de diseño actualmente forma parte de la inteligencia artificial distribuida (IAD) y se presenta como una herramienta poderosa para ciertos problemas; donde hay que tener en cuenta las características implícitas del problema que se pretenda modelar.

Para importar esta teoría se han tenido que tomar en cuenta diferentes aspectos en la modelización de los agentes, debido básicamente al cambio de entorno de físico a cognitivo. En base a esta última reflexión se propone utilizar el análisis cognitivo de tareas (ACT) como instrumento dentro de la etapa de análisis y diseño.

De este análisis se desprenderán las características fundamentales de la tarea cognitiva (TC) a ser enseñada, y en este caso son las habilidades y los elementos no encontrados de forma explícita en los enunciados de los problemas; punto importante dentro del modelado del entorno en una arquitectura multi-agente. Con este diseño se pretende tener una representación interna del mundo distribuida en los agentes. Que permita la toma de decisión en tiempo real, con lo que el sistema se adaptará más rápidamente y de forma más flexible al entorno.

El sistema que hemos desarrollado enfoca el proceso de aprendizaje como un proceso de interacción basado en el mecanismo estímulo-respuesta. El estímulo es proporcionado por la percepción del entorno, que en el caso de un SEI esta representada por el desarrollo del estudiante en la solución de un problema, y basándose en ésta proporcionar una respuesta.

En un SEI quien controla *qué y cómo* se va a enseñar es el módulo tutorial y es este módulo el foco de interés dentro de esta investigación. Se divide básicamente en dos aspectos que son *el curriculum y la instrucción*, estos elementos se refieren al desarrollo de métodos para la selección y secuencia del material a ser enseñado de acuerdo a Halff [HALH88].

I. 4. 3 Ofrecer un conocimiento detallado en habilidades

Con esta propuesta se pretende ofrecer un conocimiento más específico, detallado en áreas, que en nuestro caso son las habilidades, con lo que ganaremos tiempo al ir directamente a la falla del usuario, logrando además que el manejo de errores sea más completo, al ir dividido en zonas especializadas.

Cuando hablamos de SEIs es necesario distinguir dos tipos básicos, aquellos que enseñan conocimiento y aquellos que enseñan el uso de habilidades, el sistema *Makatsiná*³, centra su enseñanza en el aprendizaje de la habilidad para resolver estructuras triangulares por el método de los nodos, este dominio cae dentro de habilidades integradas, de acuerdo a la taxonomía de habilidades propuesta por Ryder & Redding [RYDJ93].

Es el momento para mencionar que existen dos niveles de enseñanza: 1) cuando se enseña el uso de una determinada estrategia y 2) cuando interesa enseñar al estudiante a ubicarse dentro de un contexto que le permita escoger la mejor estrategia de solución entre una gama, dependiendo de las características del problema. Nuestro trabajo se sitúa en la enseñanza de una sola estrategia.

I. 4. 4 Combinar el uso de un tutor tipo entrenador deportivo con características reactivas

Utilizar el entrenador deportivo (coach) con un modelo tutorial tipo reactivo, con la finalidad de implementar un sistema de enseñanza reactivo.

Como ya se mencionó en *Makatsiná* se encuentra implementada la enseñanza de una determinada estrategia y trabaja en base a ejercicios y ejemplos para aprender la habilidad de análisis de estructuras triangulares por el método de los nodos.

³*Makatsiná*, significa **TUTOR** en **TOTONACA**, una lengua mexicana prehispánica. Es el nombre del SEI.

Tomando en cuenta la teoría desarrollada por VanLhen [VANK88] 'step theory' sobre la inducción de un procedimiento a partir de ejemplos presentados en base a una secuencia de aprendizaje; que en este caso será encontrada por medio del ACT, y la apreciación de Anderson comentada en [HALH88] en el sentido de que no existe teoría de enseñanza suficientemente precisa y poderosa que soporte un proceso tutorial interactivo, se tomaron en cuenta los siguientes aspectos:

- **Manejabilidad:** todos los ejercicios deben tener solución y los ejemplos deben ser comprensibles a los estudiante. Estos últimos deberán tener el conocimiento previo para su comprensión.
- **Transparencia estructural:** los ejemplos deben estar orientados a la estructura de la tarea, de forma que el estudiante descubra el procedimiento general de la tarea.
- **Individualización:** los ejercicios deben ser elegidos por el tutor de acuerdo a las habilidades enseñadas y donde el estudiante muestra dificultades.

1. 4. 5 Proponer una forma de control de calidad de software, en base al comportamiento deseado de los agentes'

Proponer una forma de control y depuración de las conductas de los agentes a través de una formalización de los mismos.

Al encontrarse es sus inicios la corriente reactiva no cuenta con técnicas específicas que permitan: a) controlar su crecimiento y b) depurar los posibles errores en los comportamientos de los agentes y en consecuencia del sistema en general. Una forma de resolver este problema es aplicando la formalización matemática al comportamiento de los agentes. En esta Tesis proponemos utilizar las técnicas de formalización propuestas por Luck & d'Inverno [LUCM95] y una visualización de los agentes en base a ideogramas propuestos por Nigay y Coutaz [NIGL91] y [NIGL93] para sistemas interactivos.

1. 4. 6 Proponer un mecanismo de arbitraje par a los SEIs

En cuanto a los mecanismos de arbitraje hasta el momento la mayoría ha sido enfocada para el desarrollo de robots móviles. Como este concepto esta íntimamente ligado al diseño de la granularidad conductual; hay que investigar como y cuál sería el mejor para ser empleado en los SEIs. En esta Tesis se analizarán tres distintas arquitecturas y en consecuencia tres mecanismos de arbitraje propuestos por: Brooks [BROR86], Kaelbling [KAEL87] y García-Alegre et al. [GARM95]. De su estudio se propondrá uno para ser implementado en los SEIs.

Partiendo de estas suposiciones es que a continuación se presenta el orden de exposición del siguiente trabajo.

I. 5 Estructura de la memoria

En el capítulo II, se hace la revisión de algunas técnicas de desarrollo de los SEIs existentes, que gira en torno a los cuatro componentes de una arquitectura tradicional. Se analizan las características de estos componentes y se comentan; haciendo especial hincapié en aquellas utilizadas en el desarrollo de Makatsiná. Se explica como esta trabajando Makatsiná en cada uno de estos componentes. Se describen algunos trabajos relacionados. Finalmente se expone nuestra propuesta, y se explica la metodología utilizada para lograr el objetivo final.

En el capítulo III, se presentan los principios de reactividad revisando las primeras arquitecturas que dieron origen a esta nueva corriente como son las de Brooks y Kaelbling, así como una más reciente, la de García-Alegre, Bustos & Guinea. Se abordan sus características, que básicamente, se centran en la forma de inter-relacionar sus elementos y la forma de integrar la información del entorno, para su posterior uso. Finalmente, se descubren los elementos importantes para poder traspolar esta teoría al entorno cognitivo de los SEIs.

En el capítulo IV, se aborda el diseño y análisis del dominio utilizando modelos mentales, análisis cognitivo de tareas (ACT) y la gráfica genética (GG), todo esto como pivote para poder encontrar los agentes que representan las distintas habilidades necesarias para resolver estructuras triangulares por el método de los nodos y los errores como las características del entorno necesarias para lograr la reactividad de los agentes.

En el capítulo V, se presenta una formalización de agentes basada en lenguaje Z y en un diagrama de agentes vistos como parte de un sistema interactivo MultiAgente. Antes de su implementación.

En el capítulo VI, se presentan unos ejemplos del desarrollo de Makatsiná.

En el capítulo VII, finalmente, se exponen las conclusiones, aportaciones y el trabajo futuro.

Capítulo II

Técnicas de diseño en Sistemas de Enseñanza Inteligentes

II. 1 Introducción

En este capítulo se hace una revisión de algunos de los Sistemas de Enseñanza Inteligentes existentes (SEIs). Describiendo su comportamiento en base a las técnicas de diseño utilizadas en sus cuatro componentes y haciendo hincapié en las utilizadas en el desarrollo de *Makatsiná*.⁴

A lo largo del capítulo en pequeñas secciones específicas se describe *Makatsiná* en función de dichas técnicas. Al final se concluye nuestra aproximación. Además se hace una reseña de SEIs que de alguna forma se relacionan a nuestro trabajo.

Empezaremos por el tipo de conocimiento que puede ser transmitido por un sistema de enseñanza inteligente (SEI). Procedimental (relacionado con habilidades), declarativo (relacionado con hechos o conceptos), cualitativo (relacionado con la habilidad mental de simular y razonar en base a procesos dinámicos).

⁴ *Makatsiná* es un sistema de enseñanza inteligente, cuyo dominio es la enseñanza de análisis de estructuras triangulares por el método de los nodos. Este dominio es un tema de mecánica de los cuerpos sólidos [LAUA92](2), [BEEF80] y [HUAT80].

En Makatsiná, el conocimiento es procedimental, y está formado por las habilidades que se deben manejar para resolver estructuras triangulares por el método de los nodos.

El objetivo de un SEI es tener la capacidad de adaptación durante el proceso tutorial. Este objetivo se logra inter-relacionando los cuatro componentes básicos del sistema:

1. Módulo del Dominio o Módulo Experto.
2. Modelo del Estudiante.
3. Módulo Tutorial.
4. Interfaz.

Este tipo de sistemas necesitan utilizar técnicas de inteligencia artificial para:

- Representar el conocimiento que el sistema tiene del dominio.
- Controlar el proceso tutorial y los métodos bajo los cuales son aplicados estos principios.

A continuación se describen las características y las diferentes técnicas de diseño que los componentes de un SEI pueden utilizar para lograr el objetivo antes mencionado.

II. 2 El módulo del dominio o módulo experto

En este componente vamos a encontrar el conocimiento específico y detallado obtenido de los expertos humanos que llevan años dedicándose a la tarea cognitiva (TC) que se pretende enseñar. Las investigaciones en este campo se centran en cómo codificar el conocimiento y cómo representar la experticia. Existen diferentes técnicas para abordar el problema:

- Caja Negra (Black Box).
- Caja de Cristal (Glass Box).
- Modelos Cognitivos.

Las tres tienen ventajas y desventajas cuando se utilizan en la construcción de un SEI.

II. 2. 1 Experto caja negra

En este tipo de representación, el conocimiento se encuentra compilado, esto es, la respuesta se dirige en base a entradas y salidas, lo que proporciona una medida de lo correcto del resultado, pero no tenemos acceso al detalle de por qué se tomó una determinada decisión.

El ejemplo clásico de este tipo de tutores es SOPHIE [BROJ75], que utilizaba un simulador electrónico de propósito general. Se trataba de enseñar a los estudiantes cómo encontrar fallas en un circuito electrónico. El tutor utilizaba su simulador para determinar si las medidas que el estudiante realizaba sobre varios puntos del circuito eran correctas.

El simulador trabajaba en base a un conjunto de ecuaciones, no como lo haría un humano, por lo que no era posible para SOPHIE explicar las decisiones en detalle.

Este tipo de representación puede utilizarse con entornos de aprendizaje reactivos, que indique al estudiante si la operación realizada es correcta o incorrecta y, posiblemente, cuál es el siguiente movimiento apropiado. Esta combinación de tutor reactivo y caja negra es una filosofía interesante a la hora de crear un SEI en base a sistemas expertos (no limitados a cajas negras), ya que permite ahorrar tiempo en la fase de implementación.

En el caso de SOPHIE el tutor funcionó bien con su entorno reactivo, aunque se hicieron versiones posteriores de SOPHIE [BROJ82], donde se incorporó un modelo causal de circuitos orientado a permitir la explicación de decisiones en detalle.

Otro sistema creado a partir de un sistema experto fue GUIDON [CLAW82]. El propósito era enseñar diagnóstico y terapia de infecciones a estudiantes de medicina. El conocimiento compilado ocasionó problemas de operación debido al bajo nivel de detalle y a la falta de relaciones necesarias para el aprendizaje. GUIDON demostró la necesidad del desarrollo de una base cognitiva en el módulo experto así como la importancia de la ubicación de abstracciones lógicas y relacionales.

El objetivo principal de GUIDON era revisar las 450 reglas del experto MYCIN [SHOE76]. La implementación no resultó exitosa, siendo uno de los principales problemas el hecho de que los diagnósticos médicos no son un libro de recetas de cocina, esto es, no se receta teniendo como base un perfecto conjunto de hechos y reglas.

Posteriores investigaciones en el aspecto psicológico demostraron que las reglas utilizadas para diagnosticar funcionan además con razonamiento casual y cruce de referencias, lo que ocasiona que una regla sea invocada en el instante apropiado. Con esto último quedó demostrada la necesidad de algo más que un módulo experto para implementar un SEI.

Finalmente la lista original de 450 reglas se incrementó, incluyendo estas relaciones entre reglas, NEOMYCIN [CLAW84].

Sin embargo, un SEI, no debe limitarse a decir correcto/incorrecto, sino que tiene que saber qué puntos son críticos, o cuándo se debe explicar con más detalle. Existen otras técnicas, como el experto caja de cristal, que contemplan este otro aspecto de los sistemas de adiestramiento.

II. 2. 2 Experto caja de cristal

La metodología básica para construir este tipo de experto involucra a un ingeniero del conocimiento y a un experto en el dominio; este último identifica el área y el alcance del problema. En esta etapa se enumeran y formalizan los conceptos claves del dominio; formulando un sistema para implementar el conocimiento y finalmente de manera iterativa probarlo y refinarlo.

Estos sistemas se caracterizan por contener de forma articulada una gran cantidad de conocimiento experto. Su construcción se divide básicamente en dos etapas: la primera se refiere a la adquisición del conocimiento (tiempo consumido para construir el componente experto), y la segunda trata de automatizar este conocimiento.

Debido a que el sistema experto resultante contiene conocimiento menos compilado, la enseñanza resulta más factible que con un experto de caja negra. En este tipo de sistema experto la experticia del conocimiento humano queda representada de forma articulada. Lo anterior conlleva la ventaja de poder llegar a niveles de detalle más profundos durante la enseñanza del conocimiento, motivo del sistema.

Una de las principales conclusiones de GUIDON fue que no sólo se debe atender a la representación del conocimiento en el módulo experto, sino también a la forma en que se presenta ese conocimiento. Para que un tutor actúe de forma apropiada el módulo experto debe desplegar ese conocimiento en la forma en que lo hacen los humanos, de acuerdo con sus restricciones.

Los trabajos de Clancy de gran importancia mostraron que los SEIs están limitados si sólo usan sistemas expertos para su proceso tutorial. Esta última reflexión es la que ha conducido a la utilización de modelos cognitivos.

II. 2. 3 Los modelos cognitivos

El objetivo de un modelo cognitivo es desarrollar una efectiva simulación de la solución del problema en un determinado dominio desde el punto de vista del humano. En esta técnica el conocimiento se divide en componentes que guardan una relación directa con la forma en que el humano los clasifica y los utiliza.

El mérito de esta aproximación es que proporciona un módulo experto cuya taxonomía permite un proceso tutorial y una comunicación con el estudiante más profunda. Aunque ha habido avances importantes en los últimos diez años, por parte de las ciencias cognitivas, estos modelos requieren gran cantidad de tiempo en su desarrollo; por el gran número de detalles que tienen que ser incorporados.

En esta técnica se tienen que considerar básicamente tres cuestiones: qué componentes procedentes del análisis cognitivo son importantes para el proceso tutorial, con qué nivel deben ser representados los componentes, y finalmente cómo deben ser tratados los diferentes tipos de conocimiento: procedimental, declarativo y cualitativo en esta técnica de modelado.

En Makatsiná se utilizó un modelo cognitivo cuyo análisis y diseño se desarrolló de acuerdo a las investigaciones realizadas por Castañeda [CASS93], Redding[REDR92] y Ryder & Redding [RYDJ93].

II. 2. 4 Diferentes tipos de conocimiento

Otro de los aspectos importantes de este módulo experto es el referente al tipo de conocimiento que manejará, ya que ello conlleva la forma en la que será representado.

II. 2. 4. 1 El conocimiento procedimental

Se refiere básicamente al conocimiento que subyace al desarrollo de una tarea y esta directamente relacionado con las reglas de producción consideradas por los investigadores como una representación que capta lo esencial del proceso humano con su mecanismo cíclico de válida-actúa (si - entonces). Tal es el caso de los sistemas LMS (enseñaba procedimientos algebraicos [SLED82]), BIP (enseñaba programación en BASIC [BARA76]), MENO (enseñaba programación en Pascal [WOOB84]), WEST (enseñaba aritmética a nivel elemental [BURR82]) y CAPRA (enseñaba programación [FERI89]).

En Makatsiná tenemos conocimiento procedimental, relacionado con la tarea de análisis de estructuras triangulares por el método de los nodos.

II. 2. 4. 2 El conocimiento declarativo

Este tipo de conocimiento se refiere básicamente a hechos que no guardan relación con un uso especializado para un caso particular, es como decir *América fue descubierta por Cristóbal Colón*, esta frase representa un hecho concreto y aislado. Tal es el caso de los sistemas: SCHOLAR, de enseñanza de geografía de Sudamérica [CARJ70], WHY, de enseñanza de meteorología (básicamente posibilidades de lluvia y los factores que en ellas influyen), LISP Tutor, que enseñaba el conocimiento declarativo del lenguaje LISP [ANDJ85] y GUIDON.

II. 2. 4. 3 El conocimiento cualitativo

Es el conocimiento que subyace en la habilidad de los humanos para simular y razonar con respecto a procesos dinámicos de manera mental. Un ejemplo de este tipo de conocimiento lo encontramos en SOPHIE [BROJ75].

II. 2. 5 Representación del conocimiento

Dentro de los formalismos que existen para representar conocimiento procedimental contamos con: reglas de producción y expertos compilados, el tipo de representación elegida esta íntimamente ligada a la técnica que será utilizada para detectar los errores y para la intervención tutorial por eso es tan importante.

II. 2. 5. 1 Reglas de producción

Dentro de los sistemas de producción que utilizan reglas de producción [LAUA95] existen muchas variaciones, pero todos involucran un conjunto de reglas, que emparejan con una memoria de trabajo de hechos, para poder ser disparadas. Dentro de las ventajas para el proceso tutorial se encuentra el hecho de que el módulo tutor pueda realizar sus decisiones en base a correr una simulación del aprendizaje deseado, otra es la modularidad de poderlas aplicar por bloques, algo inherente al proceso de enseñanza, y finalmente el estado de conocimiento del estudiante puede ser diagnosticado como un conjunto de reglas de producción. Ejemplo de esto último son LISP Tutor [ANDJ85] y LMS [SLED82].

II. 2. 5. 2 El experto compilado

Se utiliza cuando por limitaciones en los requerimientos del equipo o por no resultar conveniente no se utiliza el modo de representación de reglas de producción. En este caso se compilan las soluciones de los problemas o la mayor parte de los cálculos de la solución. Este tipo de representación se conoce en la literatura como 'compiling the expert out' [ANDJ88] y obliga a un diseño que tenga en cuenta estructuras de archivo que permitan un fácil y rápido acceso a disco. Existen implementaciones en hardware de estas bases de datos. Con esta técnica se pierde la capacidad de poder resolver cualquier problema propuesto por el estudiante; sus características permiten implementar el proceso tutorial desde otra perspectiva.

En Makatsiná se han elegido las reglas y un experto compilado básicamente por el tipo de proceso tutorial que se realiza; entrenador-deportivo aplicado a una tarea cognitiva que conlleva integrar habilidades. Además debido al objetivo de implementar un proceso tutorial de tipo reactivo, resulta de primordial interés conocer el siguiente paso de antemano.

II. 3 El modelo de estudiante

Este módulo contiene todos los datos en un instante dado del estudiante y sirve para diagnosticar los efectos del proceso tutorial. Esta información se utiliza para elegir el siguiente tema de enseñanza y qué tipo de táctica será la adecuada para el adiestramiento. En caso de error será considerada una táctica remedial.

El modelo del estudiante consta de dos componentes: a) la base de datos que representa el comportamiento del estudiante durante el proceso tutorial y b) el proceso de diagnóstico que manipula la base de datos.

II. 3. 1 Las dimensiones del espacio a ser consideradas en el modelo del estudiante

De acuerdo a VanLehn [VANK88], existen tres espacios a considerar en la modelización del estudiante:

1. El ancho de banda.
2. Tipo de conocimiento a ser enseñado.
3. Diferencias entre el estudiante y el experto.

II. 3. 1. 1 El ancho de banda

Se refiere a la entrada de diferentes tipos de información, que permiten saber que es lo que el estudiante está haciendo o diciendo. La cantidad y la calidad de esta información es importante; ya que esta se comunica al componente de diagnóstico. Este componente utiliza esta información, en sus inferencias y creencias, con respecto al estado del usuario.

VanLehn propone los siguientes niveles de información, autocontenidos, en orden decreciente de acuerdo a la información que pueden proporcionar:

- Nivel. Estados Mentales.
- Nivel. Estados Intermedios.
- Nivel. Estados Finales.

En el nivel más alto se encuentran los estados mentales por donde atraviesa el estudiante cuando resuelve un problema, éstos a su vez tendrán estados intermedios, quienes a su vez contendrán a los estados finales.

Un estado mental podría ser aquel que reflejase los estados mentales por donde el usuario puede atravesar durante la resolución de un problema. En el caso de los estados intermedios, éstos están formados por los cambios de estados que guían la solución de un problema, desde sus estado inicial hasta su estado final. Estos estados son los que interesan al proceso tutorial. Los estados finales son como su nombre lo indica el estado final de la solución de un problema.

Entre más información tenga el módulo de diagnóstico, obviamente éste será mejor. Pero no siempre se puede tener acceso a los estados intermedios. Por otro lado no existe forma de acceder los modelos mentales desde un SEI (sería tanto como poder *ver* lo que el estudiante tiene en cada momento del desarrollo del problema en su mente). A través de preguntas se puede llegar a una aproximación.

Por ejemplo en el caso de un proceso tutorial tipo *entrenador-deportivo (coach)*, solamente estamos observando los *estados intermedios* para la intervención tutorial y los *estados mentales* para guiar la secuencia del desarrollo del problema.

La información obtenida del ancho de banda es importante en el sentido que de ella depende el algoritmo a usar en el diagnóstico.

II. 3. 1. 2 El Tipo de conocimiento

Se refiere a las diferentes técnicas de tratamiento existentes de acuerdo a la clasificación del conocimiento: procedimental, declarativo o cualitativo.

En el caso del *conocimiento procedimental*, que esta incluido en la resolución de un problema, hace falta una clase de interpretación que relacione al modelo del estudiante con el conocimiento de resolución del problema, debido a que el intérprete tiene que tomar decisiones basadas en conocimiento local. Existen dos tipos de conocimiento procedimental:

- Ordenado o jerárquico.
- No ordenado.

Cuando hablamos de *conocimiento ordenado*, sabemos que lleva implícito subobjetivos, esta característica va íntimamente ligada con el tipo de diagnóstico utilizado, esto es, se necesita conocer las condiciones que disparan otro estado del problema, así como el conjunto total de estados y los subobjetivos.

II. 3. 1. 3 Las diferencias entre el estudiante y el experto

Están representadas de manera general por el modelo del experto, mas una lista de los elementos que el estudiante no tiene (*missing conception*) y el experto si; y los que el

estudiante tiene y el experto no (*misconception*). Existen diferentes técnicas de implementación:

- Overlay.
- Archivos de errores.
- Librería de errores hechas en partes.

En la *forma overlay*, el conocimiento del modelo del estudiante queda representado como un subconjunto del conocimiento del experto, lo que es equivalente al modelo del experto mas una lista de los elementos que el estudiante no tiene.

En *los archivos de errores*, se resaltan las diferencias teniendo en cuenta, los *misconceptions* y los *missing conceptions*, aquí el modelo del estudiante esta representado por el modelo del experto mas una lista de errores, el sistema diagnostica al estudiante encontrando los errores en esta lista.

Dado que los errores son muy importantes para el diagnóstico, existen diferentes técnicas para encontrarlos:

- Los errores se obtienen a partir de los trabajos en educación sobre el tema.
- Los errores se encuentran observando directamente a los estudiantes, realizando la Tarea Cognitiva (TC).
- Si existe una teoría de aprendizaje sobre el dominio, se utiliza para predecir los errores en el desarrollo de la TC.

La librería de errores hecha en partes, representa una alternativa a la técnica anterior, aquí los errores se construyen durante el diagnóstico, en lugar de estar pre-definidos. Esta técnica se ha implementado en ACM (Langley & Ohlsson 1984), aquí cada error es una regla de producción, con una parte condición constituida por una conjunción de predicados y una acción, tanto los predicados como la acción provienen de archivos pre-definidos. Si la librería de predicados tiene P predicados y la librería de acciones tiene A acciones entonces se pueden componer $A * 2^P$ errores distintos, de esta forma un estudiante puede tener mas de un error. Esta técnica permite representar un gran número de modelos de usuario, utilizando solamente dos pequeños archivos. Los problemas para obtener estos archivos son análogos a los de la técnica anterior.

En Makatsiná tenemos, para guiar el proceso tutorial, los estados mentales y para el diagnóstico utilizamos los estados intermedios. Utilizamos conocimiento procedimental ordenado; dado que estamos abordando el diseño de un tutor reactivo y es necesario conocer las condiciones de disparo hacia otros estados del problema de forma previa a su

ejecución. Se utilizó la técnica de archivos de errores, para modelar las diferencias entre el estudiante y el experto.

II. 3. 2 Clasificación del modelo de estudiante de acuerdo a distintas características

En Urretavizcaya [URRM91] se aborda una clasificación del modelo de estudiante de acuerdo a las diferentes características de los modelos. En el trabajo realizado pro Urretavizcaya se explica con más detalle esta clasificación. A continuación se presentan algunos aspectos relevantes.

- Modelos genéricos o canónicos Vs. Modelos individualizados (grado de especialización).
- Modelos especificados explícitamente Vs. Modelos inferidos implícitamente (posibilidad de actualización del modelo).

Las siguientes además de ser características importantes de modelos de forma individual, pueden ser acopladas en mayor o menor medida en cualquier modelo que se utilice.

- Modelos a largo plazo Vs. Modelos a corto plazo (extensión temporal del modelo).
- Modelos según la representación del modelo (representación del modelo).
- Métodos de uso de los modelos.
- Tipos de usuarios que utilizan el sistema.
- Número de modelos para cada usuario.
- Cobertura del conocimiento (overlay/diferencial).

Un modelo de usuario *canónico o genérico*, representa a todos los usuarios de la misma forma, como si estos fueran homogéneos. Este modelo es limitado frente a grupos de usuarios heterogéneos. Tal es el caso de ZOG, que es un sistema de selección de menús para definir la comunicación hombre-máquina [ROBG81] y MACSYMA un sistema que enseña manipulación simbólica de expresiones matemáticas [GENM78].

Un modelo *individualizado*, contiene información específica para cada usuario, de esta forma modela el comportamiento individual de cada uno.

Existen dos técnicas para construir modelos *individualizados explícitamente*:

1. Permitir al usuario configurar su modo de interacción con el sistema para adaptarlo a sus necesidades. La desventaja, es que le deja la responsabilidad al usuario y se pueden presentar problemas cuando el usuario es neófito o esporádico.
2. Proporcionar al sistema la suficiente información del usuario para que este pueda configurar el modelo. Las preguntas estarían relacionadas con el nivel de conocimiento que el usuario tiene, pero en la enseñanza se necesita saber más cosas como: qué sabe, qué no sabe, qué sabe incorrectamente, y en estos casos el usuario no siempre conoce la respuesta.

Para resolver parte de los problemas que presenta el anterior modelo aparecen los modelos *individualizados inferidos*. Se construyen a partir de un modelo inicial de usuarios basado en características comunes a todos ellos, añadiendo progresivamente información característica de cada estudiante. Para lograrlo son necesarias dos acciones: actualizar el modelo, con la información inferida de cada interacción contrastándola con la ya obtenida, y expresar el grado de aprendizaje sobre cada punto que se toca.

Un modelo *explícito o estático* tiene como característica principal que no cambia durante la interacción con el usuario, mientras que un modelo *implícito o dinámico*, se actualiza con nueva información en cada interacción con el usuario.

Un modelo *estático* podría tener datos iniciales producto de una sesión preliminar con el usuario o estar pre-codificado de antemano. En el caso de los modelos *dinámicos* el sistema infiere la información necesaria, directamente de la interacción con el usuario.

Las características de los siguientes modelos, como ya se mencionó pueden ser implementadas en cualquier tipo de modelo.

Los modelos *a largo plazo* son útiles cuando el sistema desarrolla un proceso tutorial de varias sesiones que incluye varios temas, aunque también serían deseables en sesiones repetidas de un mismo tema. La información a largo plazo que manejan agrupa el conocimiento sobre el dominio, la capacidad de aprendizaje, los errores frecuentes. La información a corto plazo que manejan es básicamente los errores actuales; esta información posteriormente actualizará la información a largo plazo. Este tipo de modelos hace hincapié en el desarrollo de la sesión del estudiante, cuántos errores y de que clase ha cometido, con que frecuencia en las primeras sesiones. Los modelos a largo plazo son bastante mas complejos que los modelos a corto plazo.

De las posibles formas de representar el conocimiento (reglas de producción, marcos, redes semánticas, etc., [LAUA95]) por lo general los SEIs utilizan una combinación de los diferentes modos. La forma elegida esta relacionada con la interacción y con el modelo del estudiante.

En cuanto a los *métodos de uso del modelo* se refieren básicamente a la técnica sobre la que se apoya el uso del modelo del estudiante. Puede ser *descriptiva*, lo que implica que el modelo del estudiante solamente es una base de datos con información sobre el usuario o *predictiva* en cuyo caso realiza una simulación.

En cuanto a los *tipos de usuarios*, se refiere a la capacidad de un sistema para precisar la modelización del usuario en base al tipo de características de este, por ejemplo sistemas que trabajan con varios usuarios simultáneamente.

El *número de modelos* se refiere a que un usuario puede ser modelado de acuerdo al subdominio, esto sucede cuando un dominio grande se rompe en subconjuntos, permitiendo la modelización de un mismo usuario por subdominios. Por ejemplo un sistema que enseñe sistemas operativos; podría tener subdominios de tipos de memoria, administración de memoria, etc.

La *cobertura del conocimiento* se refiere a la forma en como éste se representa. Puede ser básicamente de dos tipos: 1) *overlay* en el que el conocimiento del usuario es un subconjunto del conocimiento utilizado por el sistema o 2) *diferencial* en el que la cobertura está formada por las diferencias entre el conocimiento del usuario y el conocimiento del sistema.

II. 3. 3 Modelos individualizados implícitamente

Estos modelos son los que se han utilizado en los SEIs, dadas sus ventajas para el modelado del usuario. Se distinguen dos subclases, *los modelos cuantitativos y los modelos cualitativos*.

II. 3. 3. 1 Los modelos cuantitativos

Se centran en valoraciones numéricas de diferentes aspectos del usuario. La información que producen es escueta y se recomiendan en sistemas que no necesiten gran adaptación a usuarios o donde se sepa de antemano que el grupo de usuarios es en algún sentido homogéneo. Entre los diferentes tipos de modelos cuantitativos existen aquellos que contiene varios modelos pre-definidos de antemano e identificables por un número que clasifican al usuario de acuerdo a una interacción preliminar dentro de alguno de los modelos. A estos modelos se les llama modelos escalares. Tal es el caso de los sistemas LEEDS ARITHMETIC SYSTEM, sistema de enseñanza de aritmética [WOOP71] y KEYSTROKE sistema de enseñanza de tratamiento de textos [CARS83] ambos reducen toda la complejidad de las tareas que un estudiante puede abordar a un simple número.

II. 3. 3. 2 Los modelos cualitativos (también llamados simbólicos)

Describen a los usuarios en términos de relaciones espaciales, temporales o causales y son más complejos que los anteriores, cuentan con dos subclases:

- Modelos por valoración/calificación.
- Modelos ejecutables o modelos de simulación.

II. 3. 3. 2. 1 Los modelos por valoración

Estos modelos realizan una aproximación mediante la valoración de diversas situaciones que forman la base de conocimiento del modelo, este tipo de modelo nos permite diagnosticar el error pero no podemos reconstruir el comportamiento del usuario. El comportamiento del usuario puede ser:

- Perfil.
- Ad-Hoc.
- Overlay y diferencial.

Modelos perfil, califican distintos aspectos del usuario pre-definidos en el sistema. Tal es el caso de GRUNDY que selecciona libros apropiados a los gustos del usuario [RICE79].

Modelos Ad-Hoc, se caracterizan porque el conocimiento que se incluye, en los modelos del usuario, es específico del dominio de aplicación. Tal es el caso de SOPHIE [BROJ75], WEST [BURR82] y MACSYMA [GENM78].

Modelos overlay se caracterizan por representar el conocimiento del usuario como un subconjunto del conocimiento del experto (sistema). Tal es el caso de SCHOLAR y WUMPUS, este último es un juego que pretende ejercitar el razonamiento lógico y probabilístico del jugador [GOLI79].

Modelos diferenciales, se caracterizan por representar el modelo del usuario como una diferencia entre el conocimiento del usuario y el conocimiento del experto. Tal es el caso de ACE [SLED82], su dominio es el análisis de explicaciones complejas (espectros NMR) y WEST[BURR82].

II. 3. 3. 2. 2 Los modelos ejecutables

Su principal característica es que se pueden comportar de forma predictiva, esto es el sistema tiene la posibilidad de predecir el comportamiento del usuario en diferentes situaciones en base a un modelo del estudiante que incluye un mecanismo de inferencia; como en el caso de

NEOMYCIN [CLAW84] y HERACLES [CLAW86](1) y [CLAW86](2). Ambos cuentan con un sistema general de diagnóstico para múltiples dominios.

En Makatsiná, contamos con un modelo individualizado implícitamente tipo cualitativo cuya precisión cae dentro de los modelos por valoración y dentro de esta clasificación tenemos una combinación ad-hoc, porque la base se llena con aspectos propios del desarrollo de la tarea cognitiva, diferencial en base a archivos de errores, dado que así se implementaron las diferencias entre el experto y el estudiante.

II. 3. 4 Técnicas de diagnóstico

En la siguiente sección abordaremos el segundo componente de un modelo de estudiante que es el *procedimiento de diagnóstico*, empleado sobre la base de datos. Según la clasificación de VanLehn [VANK88], existen nueve técnicas de diagnóstico, que se numeran en la Tabla II.1, donde además se hace una recomendación de su uso de acuerdo a dos de las dimensiones de modelización del estudiante (ancho de banda y tipo de conocimiento).

1. Seguimiento de una Ruta (Model-Tracing).
2. Encontrando la Ruta (Path Finding).
3. Condición por Inducción (Condition Induction).
4. Reconocimiento de un Plan (Plan Recognition).
5. Seguimiento de asuntos (Issue-Tracing).
6. Sistemas expertos (Expert systems).
7. Árboles de decisión (Decision Trees).
8. Generación y Prueba (Generate and Test).
9. Diagnóstico Interactivo (Interactive Diagnosis).

De acuerdo a las características de Makatsiná, abordaremos la 1, la 4 y la 5.

En la técnica de *seguimiento de una ruta* propuesta por Anderson, J. y comentada en [VANK88], se supone que todos los estados mentales significativos del estudiante están disponibles para el procedimiento de diagnóstico. La idea básica radica en el uso de un intérprete no determinista para modelar la solución del problema, así en cada paso del proceso de resolución, el interprete puede sugerir un conjunto de reglas aplicables. El procedimiento de diagnóstico disparará estas reglas y obtendrá de esta forma el conjunto de posibles estados sucesores (en el caso de un intérprete determinista sólo contaremos con un estado sucesor), uno de estos estados será el que corresponda a la solución del estudiante.

Como todas las técnicas ésta tiene sus ventajas y desventajas; entre las desventajas está qué hacer si ninguno de los estados disparados coincide con el del estudiante, ya que esto no tiene sentido en el caso del determinista.

En la técnica de reconocimiento de un plan se necesita cumplir dos condiciones: 1) que el conocimiento sea procedural y ordenado y 2) que todos o casi todos los estados físicos observables del estudiante puedan ser utilizados por el procedimiento de diagnóstico. En esta técnica el dominio se representa en un árbol, donde los nodos terminales son las acciones más primitivas, los nodos no-terminales incorporan los subobjetivos y la raíz el objetivo final.

La técnica seguimiento de asuntos [BURR82] esta basada en el análisis de episodios cortos en la solución de un problema, dividiendo su observación en conjuntos de micro-habilidades o asuntos que se emplean durante ese episodio. Este tipo de análisis no explica cómo estos asuntos interactúan, ni el papel que desempeñan en la solución global del problema, sólo interesa saber si se usan o no.

Tabla II. 1 - Técnicas de Diagnóstico propuesta por VanLehn [VANK88]

	<i>Tipo de Conocimiento</i>		
	Procedimental no-ordenado	Procedimental ordenado	Declarativo
Ancho de Banda			
Estados Mentales		Seguimiento de una Ruta (1)	
Estados Intermedios	Seguimiento de Asuntos (5)	Reconocimiento de un Plan (4)	Sistema Experto (6)
Estados Finales	Encontrando la Ruta (2)	Arbol de Decisiones (7)	Generación y Prueba (8)
	Condición por Inducción (3)	Generación y Prueba (8)	
		Diagnóstico Interactivo (9)	

En Makatsiná debido a que el objetivo es lograr un tutor reactivo se utiliza el seguimiento de una ruta; dado que sólo tenemos que enseñar una estrategia para la solución de estructuras triangulares. Se usa el seguimiento de asuntos, dadas las características del conocimiento procedimental-ordenado. Finalmente, debido a la división de la solución del

problema en etapas más finas representadas por subtutores que revisarán el uso de varias habilidades, utilizamos el reconocimiento de un plan, dentro de esos micromundos.

II. 4 Módulo tutorial

A este módulo le concierne todo lo referente a los problemas en el desarrollo del curriculum y de la forma de enseñar ese curriculum. El curriculum se refiere a la selección y a la secuencia del material de enseñanza. La enseñanza también conocida como proceso tutorial se refiere a los métodos para presentar ese material, de acuerdo a Halff [HALH88]. De aquí que los SEIs puedan utilizar diferentes técnicas de enseñanza. En general las intervenciones tutoriales deben contener, si no todas, al menos alguna combinación de las siguientes características:

- Tener algún control sobre el curriculum y su secuencia.
- Ser capaz de responder a preguntas hechas por el usuario.
- Darse cuenta de cuándo el usuario necesita ayuda y de qué tipo.

II. 4. 1 Aspectos centrales

Los aspectos centrales que se tratan en el diseño del módulo tutorial son: el control sobre la selección y la secuenciación, el estilo de aprendizaje, el estilo de la enseñanza y el tipo del dominio de la TC; estos tres aspectos íntimamente ligados son fundamentales en el diseño de este módulo.

II. 4. 1. 1 Formas de control de la selección y secuenciación en los SEIs

De acuerdo a Fernández [FERI89] existen 3 tipos de control donde se pueden agrupar los diferentes SEIs:

1. Sistemas basados en reglas.
2. Sistemas basados en autómatas de estados pedagógicos.
3. Sistemas basados en planificación.

En el trabajo realizado por Fernández [FERI89] se explica a fondo en que consisten cada uno de ellos. A continuación se presenta un resumen de las ideas básicas de los tres tipos de control existentes.

A los sistemas con control *basado en reglas* pertenecen aquellos que han sido construidos en base a un sistema experto. A este tipo corresponden los sistemas: GUIDON, NEOMYCIN y HERACLES. Comentaremos como funciona GUIDON.

GUIDON [WENE87] fue diseñado para satisfacer los siguientes objetivos: 1) demostrar la utilidad pedagógica de la base de datos del sistema experto MICYN, 2) descubrir el conocimiento adicional en un sistema de enseñanza inteligente y 3) expresar las estrategias para el proceso en términos independientes del dominio.

En GUIDON [CLAW82] se separan las capacidades de diálogo tutorial de las de resolución del problema. El adiestramiento se logra en base a un programa de consulta basado en reglas y las capacidades de diálogo.

GUIDON compara las preguntas del estudiante con las del experto y las critica en base al siguiente patrón. Cuando el usuario plantea una hipótesis la compara con la conclusión a la que llegó el experto (MYCIN) a partir de los mismos datos. La transferencia de conocimiento se realiza a través de diálogos de diferentes casos. El propósito del sistema es ampliar el conocimiento del estudiante señalando líneas inapropiadas de razonamiento y haciendo sugerencias sobre aspectos que el estudiante no consideró.

Existen básicamente dos paquetes de reglas: las que componen a la base del conocimiento del sistema experto y las reglas encargadas del proceso tutorial, las cuales a su vez se dividen en reglas para acumular certeza (premisa) y reglas para seleccionar un procedimiento de discurso (acción).

Como sistema *basado en autómatas de estados pedagógicos*, comentaremos a MENO-Tutor [WOOB84].

Este sistema fue desarrollado con el fin de proporcionar un marco de trabajo adecuado para definir y probar reglas pertenecientes al proceso tutorial, con el fin de generalizar estas reglas.

El sistema utiliza dos componentes diferentes para la planificación y generación del proceso tutorial. El componente encargado del proceso tutorial y el generador de lenguaje natural.

El componente encargado del proceso tutorial esta formado por un conjunto de unidades de decisión organizadas en tres niveles de planificación que refinan sucesivamente las acciones del tutor. En el nivel superior (Nivel Pedagógico) lo que se decide es la frecuencia de interrupción al estudiante y la frecuencia con que se verificará su conocimiento. En el segundo nivel (Nivel Estratégico) se refina la decisión pedagógica en estrategia especificando el método que se va a usar, como podría ser el determinar la competencia del

estudiante con preguntas. En el tercer nivel (táctico) se selecciona una táctica para implementar la estrategia.

La implementación de este componente es similar a una red de transición que se atraviesa de estado a estado mediante una rutina iterativa a través de un espacio de caminos. Los caminos entre estados no son fijos ya que, la estructura de control provee un conjunto de metareglas que producen cambios en los caminos por defectos.

Los sistemas *basados en planificación* nacen como una necesidad para construir sistemas de enseñanza que utilicen estrategias para cursos largos. La planificación puede ser estática o dinámica.

Peachey y McCalla [PEAD86] proponen usar técnicas de planificación para crear cursos largos e individualizados que traten campos más amplios. Este sistema pertenece al tipo de *planificación estática*.

Las técnicas de planificación se han utilizado principalmente en robótica [GARM93] y [TORC93]. También en áreas como la modelización del razonamiento [HAYB79] y comprensión del lenguaje natural [WILR83] entre otras.

El proceso de planificación consiste en decidir el curso de una acción antes de que se realice, y un plan es la representación del curso de dicha acción. Un planificador es un programa que parte de un estado inicial e intenta alcanzar el estado final deseado mediante la aplicación de un conjunto de operadores sobre los objetos que componen el mundo en el que opera. Esto es, construye la secuencia de operadores que permite transformar el estado inicial en el estado final. Un planificador incorpora una base de conocimientos formada por: 1) los operadores de cambio de estado, 2) una base de datos en la que se caracteriza el estado final y los objetivos y 3) un mecanismo de inferencia.

El sistema tutor propuesto por Peachey et al. [PEAD86] está formado por cinco componentes: 1) base de conocimientos del dominio, 2) modelo del estudiante, 3) colección de operadores de enseñanza, 4) planificador y 5) ejecutor.

El planificador construye el plan mediante una secuencia de pasos para lograr un objetivo instruccional. Cada paso es una instancia de un operador y cada operador tiene una acción asociada. El ejecutor ejecuta el plan invocando las acciones asociadas a cada paso del plan. El planificador crea su plan de enseñanza simulando los efectos de las acciones de los operadores sobre el modelo del estudiante. Un operador es semejante a una regla en un sistema de producción e incluye un conjunto de precondiciones y un conjunto de efectos esperados. Mediante el uso del operador el planificador simula la llamada de los pasos del plan de enseñanza y considera los resultados que puede obtener sobre un modelo de estudiante virtual. Realmente no hay conexión directa entre los efectos esperados de un

operador y los efectos reales que produce. El ejecutor debe detectar y recuperar estas desviaciones, haciendo uso de las opciones construidas dentro del plan o inclusive llegando a re-invocar al planificador para revisar un plan que no ha tenido éxito.

Fernández [FERI89] propone un tipo de *planificador dinámico*. En este tipo de sistema no se separan las técnicas de planificación de la resolución del problema. Uno de las características de este sistema es que evita los costos de re-planificación cada vez que el camino trazado no se ajusta a la realidad.

Se establecen unos objetivos pedagógicos en la sesión. Para lograrlos se eligen unas estrategias de enseñanza relacionadas con un conjunto de planes. Cada vez que se interacciona con el alumno el sistema es capaz de detectar si existen conflictos en cuyo caso se reconsideran los objetivos a nivel local o global. Contiene estrategias de enseñanza basada en estados, que son independientes del dominio a enseñar, con el fin de que pueda ser utilizado en otro dominio de naturaleza estructurada. El control de las estrategias se presenta de forma más flexible; ya que no existe una forma preestablecida de transiciones entre estados. Las transiciones se establecen en términos de los planes y las reglas de resolución de conflictos.

El planificador dinámico propuesto por Fernández [FERI89] esta compuesto por la cooperación de 4 módulos: 1) decisión pedagógica, 2) decisión temática, 3) módulo de enseñanza y 4) módulo supervisor.

La decisión pedagógica se encarga de: 1) seleccionar la combinación de estrategias de enseñanza más adecuada para conseguir los objetivos del alumno teniendo en cuenta información del usuario. 2) desarrollar las estrategias establecidas, generando subobjetivos progresivamente y 3) comunicarse con el alumno sobre la elección de distintas actividades didácticas a desarrollar.

La decisión temática es el componente que refina los objetivos establecidos por otros módulos. Tomando en consideración el dominio objeto de la enseñanza.

El módulo de enseñanza explica o verifica los conceptos seleccionados por decisión temática. También se ocupa de la comunicación y de la actualización del modelo del alumno.

El módulo supervisor actúa cada vez que se produce una interacción usuario-sistema. Para detectar posibles conflictos entre los objetivos del usuario y los objetivos establecidos por el tutor (en curso o en espera). Selecciona una estrategia local o notifica la necesidad de replantear la estrategia.

Para lograr cada una de las tareas, estos módulos cuentan con: una base de conocimiento formada por objetivos, planes que los resuelven y reglas que seleccionan un plan para

resolver un objetivo. Las reglas de selección utilizan condiciones sobre el estado de la sesión actual y la historia de aprendizaje del alumno. La ejecución de cada plan dará lugar a la ejecución de acciones básicas o al planteamiento de nuevos objetivos.

En Makatsiná no existe el significado de control en los términos mencionados anteriormente. Uno de los objetivos en los sistemas reactivos es lograr la distribución del control entre los agentes para lograr que el ciclo de toma de decisión sea lo más 'rápido' posible. Hay que hacer hincapié que en el caso de los agentes reactivos no existe un modelo simbólico exhaustivo del entorno sobre el que el sistema pueda razonar. El modelo sobre el que toma decisiones es el 'mundo real'; de aquí que el control de un sistema reactivo dependa de los objetivos de cada agente y de su interacción con el 'mundo real'.

II. 4. 1. 2 El estilo de aprendizaje

Se refiere a la mejor forma de enseñar un determinado dominio. Sobre esta cuestión existe gran cantidad de teoría desarrollada dentro del campo de la educación por pedagogos y psicólogos. Generalmente se acude a experiencias para desarrollar técnicas que de una u otra forma den mejores resultados. Por ejemplo es conocida la capacidad de los estudiantes de ingeniería para captar en un problema aspectos visuales implícitos significativos y utilizarlos en la resolución de éste. En este caso el uso de métodos visuales resaltaría estas características. Actualmente se puede recurrir a técnicas que permiten conocer más acerca de la visualización y con ello poder orientar a los estudiantes hacia el desarrollo de esta capacidad, que les permitiría un mejor aprendizaje.

En Alonso y Gallego [ALOC94], se hace una reseña histórica de definiciones y aportaciones, que finaliza con el desarrollo de un cuestionario. Este cuestionario puede ser aplicado a grupos de estudiantes y con ello conocer las preferencias del estilo de aprendizaje. El cuestionario distingue entre cuatro estilos de aprendizaje: activo, reflexivo, teórico y pragmático; estos estilos a su vez representan las experiencias que se tienen durante el proceso del aprendizaje: vivir la experiencia, reflexión, generalización y elaboración de hipótesis y finalmente aplicación. Sin embargo, existen diferencias claras en base a las diferentes áreas del conocimiento. Podríamos pensar que el hecho de que las distintas Facultades Universitarias, agrupadas de acuerdo a como manejan el conocimiento en carreras: técnicas, de humanidades y experimentales, se inclinen hacia un tipo dado de aprendizaje; esta directamente relacionado con el tipo de dominio y la forma de utilizarlo en la práctica. Poder conocer el estilo de aprendizaje de los estudiantes y sus preferencias, implica diseñar cursos que aporten mejores resultados. En el caso concreto de los SEIs, podríamos diseñarlos haciendo énfasis en estos estilos de aprendizaje.

En Alonso et al. [ALOC94] se realiza un estudio estadístico riguroso en base a un *Cuestionario Honey-Alonso* diseñado por los autores donde los estudiantes de ingeniería civil muestran preferencias por los estilos de aprendizaje activo y pragmático, predominando el pragmático; debido a que las ingenierías caen dentro de las ciencias aplicadas.

Este hecho es tomado en consideración en la implementación de Makatsiná. Ya que se hace énfasis en el desarrollo de ejercicios, esto es, la aplicación de un método de resolución de problemas.

II. 4. 2. 3 El estilo de enseñanza

Representa la contraparte del aprendizaje y debe ser diseñado de forma tal que saque ventaja del *estilo de aprendizaje*. Debemos ver el proceso de *enseñanza - aprendizaje* como un mecanismo de comunicación que debe darse entre dos agentes. De aquí que el proceso tutorial se aborde como un proceso de comunicación [GIRJ92], más que como un proceso de enseñanza convencional.

II. 4. 1. 4 El tipo del dominio

Se refiere a la diferencia que existe entre los diversos tipos de tutores: *expositores* y *procedimentales*. En un *tutor expositor* el énfasis se encuentra en el *conocimiento factual* (*representa los hechos que conforman al conocimiento declarativo*) y a partir de ese conocimiento se infieren las habilidades de primer orden que pertenecen a la clasificación de *conocimiento declarativo*.

Los *tutores procedimentales*, se encargan de enseñar *habilidades y procedimientos*, que tienen aplicaciones en el mundo real, estos tutores funcionan más como entrenadores, presentando ejemplos y mostrando el uso de las habilidades a través del desarrollo de problemas. Dentro de su contenido se encuentran problemas orientados a pruebas y prácticas especiales.

II. 4. 2 Curriculum

Este problema, como ya se mencionó, se aborda desde dos perspectivas:

1. Encontrar una adecuada representación para el material a enseñar.
2. Seleccionar y secuenciar ese material

El primer problema está relacionado con el conocimiento referente a las instrucciones de la enseñanza y con el módulo experto. Y el segundo esta relacionado con el tipo de control (sección II.4.1.1).

II. 4. 2. 1 Selección y secuenciación

En cuanto a la *selección y secuenciación*, existen diferencias de acuerdo al tipo de tutor, en los *expositores* el problema se centra en mantener la coherencia en la presentación del material para su posterior reflexión. Los *tutores procedimentales* tienen además el problema de tener que ordenar las subhabilidades integrantes de la *habilidad-objetivo* y de tener que seleccionar correctamente los ejercicios y ejemplos que reflejen ese orden.

Makatsiná es un tutor tipo procedimental.

II. 4. 2. 2 Selección y secuenciación en los tutores procedimentales

Este tipo de tutores *basan su enseñanza en ejercicios y ejemplos*. En este caso el punto central es contar con mecanismos adecuados de selección y secuencia del material. Lo ideal sería poderlos elegir de acuerdo al *estilo de aprendizaje*, pero, según Anderson [ANDJ88], no existe una teoría de aprendizaje que sea lo suficientemente precisa y poderosa para sustentar un proceso tutorial interactivo.

Las investigaciones recomiendan tener en cuenta los siguientes puntos con relación a los ejercicios y ejemplos:

- *Manejabilidad*, todos los ejercicios deben tener solución y deben poder ser comprendidos por un estudiante que haya cubierto el material previo.
- *Transparencia estructural*, la secuencia de ejercicios y ejemplos debe reflejar la estructura del proceso a ser enseñado y deberá tener la cualidad de guiar al estudiante en la adquisición de la *habilidad-objetivo*.
- *Individualización*. El ejercicio presentado al estudiante debe poseer la(s) subhabilidad(es) que el estudiante ya maneja y poder relacionarse de forma fácil con la(s) subhabilidad(es) que quiere adquirir en esos momentos.

VanLhen [HALH88] expone una teoría llamada 'step-theory', donde habla de que el curriculum debe estar dividido de acuerdo a aspectos o características importantes. Estas divisiones representan los *pasos-de-teoría*. Cada lección podrá utilizar solo un *paso-de-teoría*, de forma que el procedimiento a ser aprendido saque ventaja de la división del curriculum por pasos. Así mismo se habla de las *felicity conditions* o la capacidad de poder encontrar las divisiones del dominio que más ayuden al proceso tutorial. Esta división coincidirá con un paso en el procedimiento que deba ser aprendido. En relación a esto se considera que lo que se necesita es la ayuda de un buen *análisis cognitivo de tareas de (ATC)* para poder encontrar esas rupturas críticas del dominio que ayuden a esclarecer puntos en la

enseñanza como los expuestos en Castañeda [CASS93], Redding [REDR92] y Ryder & Redding [RYDJ93].

En cuanto al *proceso de selección y secuenciación del material*, es importante diseñar el curriculum para que cumpla con las siguientes funciones:

- Contener el material dividido de forma que sea fácil su manipulación, en base a objetivos instruccionales.
- Secuenciar el material de forma que su estructura convenga al usuario.
- Asegurar que los objetivos planteados en cada unidad pueden ser logrados.
- Actualizarse como consecuencia de los mecanismos de evaluación del impacto del proceso tutorial en el estudiante.

II. 4. 3 El proceso tutorial (enseñanza)

Este componente incluye, las funciones de cubrir la *presentación del material*, formas de poder contestar al estudiante preguntas y las condiciones y el contenido de la *intervención del proceso tutorial*.

II. 4. 3. 1 Métodos de presentación

Los métodos de presentación dependen del tipo del dominio a enseñar y de los objetivos que se desean alcanzar durante el proceso tutorial.

Los *tutores expositores* utilizan diferentes formas de diálogo, los *tutores procedimentales* orientados al manejo de habilidades utilizan ejemplos y ejercicios con entrenador, para lograr el manejo de esas habilidades.

Las diferentes formas de diálogo implican diferentes objetivos de enseñanza [HALH88]. Collins propone una guía para la selección del dialogo de acuerdo a los objetivos de enseñanza. Esta clasificación coincide con las propuestas por Gagné y Merrill y se muestra resumida en la Tabla II.2.

II. 4. 3. 2 Intervención tutorial

La intervención tutorial, se necesita para mantener el control de la situación durante el desarrollo del proceso tutorial, con el objetivo de mantener al usuario lejos de un aprendizaje inapropiado o incorrecto y mantenerlo al margen de caminos fuera del objetivo de enseñanza. Automatizar este proceso implica diseñar reglas para decidir la intervención o la no intervención, además de formular el contenido de la intervención.

Existen básicamente dos formas para guiar una intervención tutorial, ya comentadas como *técnicas de diagnóstico*. Estas son: 1) seguimiento de una ruta (Model-Tracing) y 2) seguimiento de asuntos (Issue-Tracing).

Tabla II. 2 - Estrategias de Diálogo de acuerdo a diferentes objetivos instruccionales según Collins [HALH88]

Objetivos Instruccionales	Estrategias
Enseñar hechos y conceptos	Separar hechos o conceptos
Explicar hechos o conceptos	Enseñar reglas y relaciones Estrategias de selección Trampear
Enseñar habilidades por inducción	Ejercicios y ejemplos orientados a mostrar las subhabilidades

II. 4. 3. 2. 1 Intervención con seguimiento de una ruta

En este tipo de intervención se cuenta con los caminos que podría elegir el usuario en la solución de un problema. Se revisa el comportamiento de ambos tratando de emparejar el desarrollo del usuario con alguno de los caminos que se pueden tomar. Cuando el emparejamiento fracasa, el tutor interviene emitiendo un consejo que permite al alumno retomar el camino correcto. La desventaja es que el tutor intervendrá siempre que no pueda reconocer el camino seguido por el usuario, aún cuando este sea mejor.

II. 4. 3. 2. 2 Intervención con seguimiento de asuntos

Esta forma de intervención tutorial fue desarrollada por Burton y Brown e implementada en WEST [BURR82]. Aquí las habilidades o conocimientos que se desea que el estudiante aprenda se llaman asuntos (Issues).

Cada *asunto* representa una *mini-teoría articulada*, esto es, un pedazo del experto visto como caja de cristal que se caracteriza por dos procedimientos.

El primer procedimiento tiene el papel de observador del desarrollo del estudiante. Su misión es obtener evidencia para saber si el estudiante usa o no ese concepto o habilidad particular. A este procedimiento se le denomina *reconocedor de asuntos* y sus observaciones se utilizan para construir un modelo del desarrollo del estudiante.

El segundo procedimiento sabe como utilizar la información vertida en el modelo del estudiante para decidir si el estudiante domina o no el asunto. A este procedimiento se le denomina *evaluador de asuntos*. En la Figura II.1 se muestra la anatomía de un asunto.

En el caso concreto de WEST, se utilizó el tutor tipo entrenador. Para conocer el progreso del estudiante se ejecutaban todos los evaluadores del modelo. Cuando el estudiante ejecuta una mala jugada, ésta conlleva una debilidad o carencia de habilidad; ya que no fue buena. Entonces su debilidad es comparada con las habilidades (asuntos) necesarias para ejecutar una mejor jugada y de esta forma encontrar porque el no realizó esta jugada.

Figura II. 1 - Anatomía de un asunto



Una vez que ha sido determinado en que *asunto* falla, el entrenador presenta una explicación de ese *asunto* seguida de un ejemplo de uso. Lo que se persigue con este tipo de intervención es que el estudiante reciba la información cuando este más receptivo a procesarla, en otras palabras, nada más cometer el falla.

La Figura II.2 compuesta por Parte 1(a) y Parte 1(b) presenta el proceso del modelado tutorial utilizando *asuntos y ejemplos*. En la Parte 1(a) de la figura se muestra el proceso de construcción de un modelo del desarrollo del estudiante durante la resolución de una serie de problemas. En el caso particular de WEST son jugadas ya que el aprendizaje se realiza en el contexto de un juego. Cada vez que el estudiante realiza una jugada, el reconocedor de *asuntos* elabora la abstracción de los aspectos importantes del desarrollo en el estudiante y en el experto del sistema en el mismo entorno, utilizando para ambos los mismos *reconocedores de asuntos*. Se comparan las dos abstracciones para obtener un modelo diferencial que permita conocer los *asuntos* que el estudiante no domina.

Es importante resaltar que sin el experto no sería posible determinar el grado de dominio de un asunto (habilidades o conocimiento) por parte del estudiante o si el estudiante no lo ha utilizado por falta de experiencia en su uso.

Figura II. 2 - Modelo Tutorial con asuntos y ejemplos de Burton y Brown [BURR82] Parte 1(a)

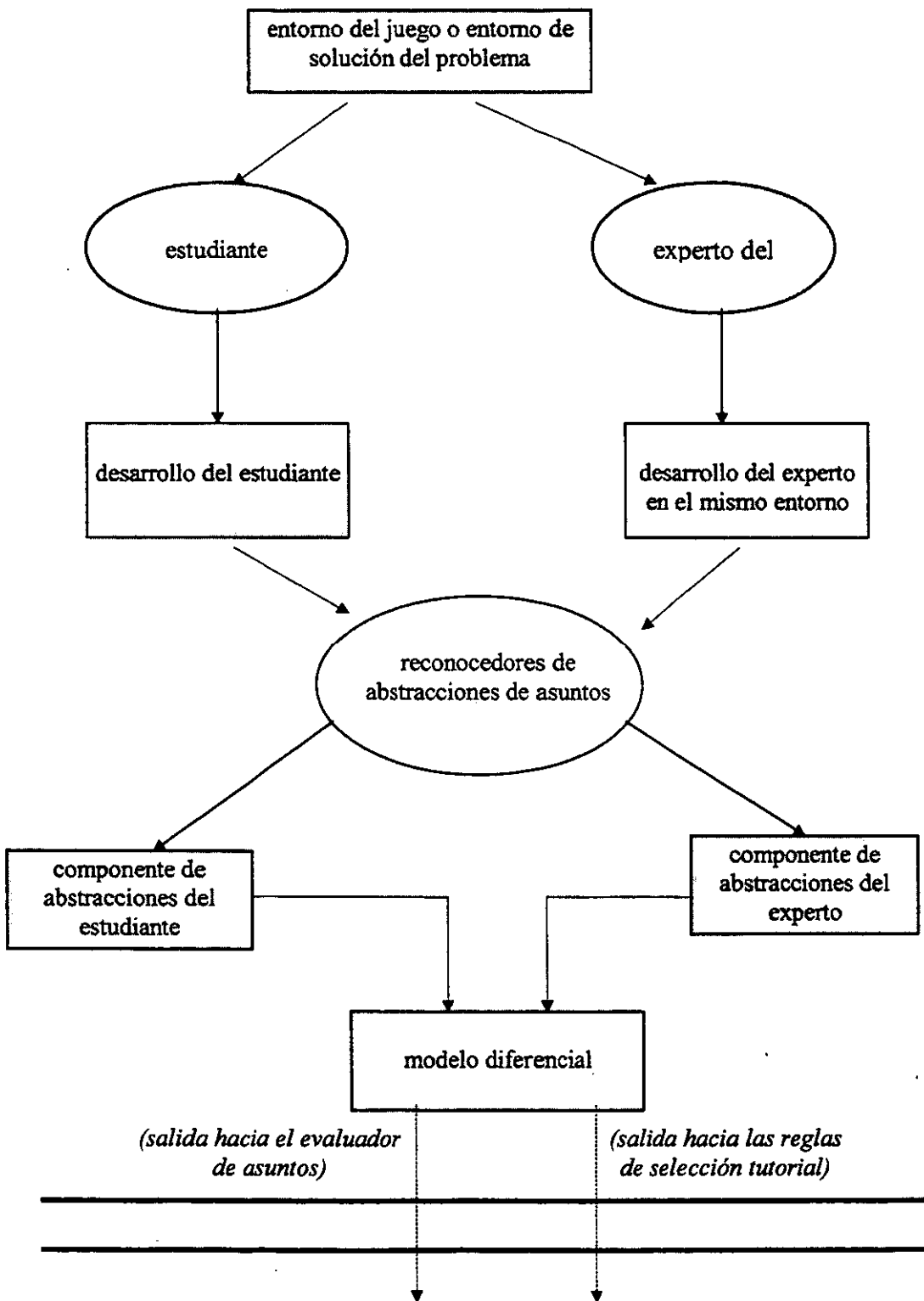
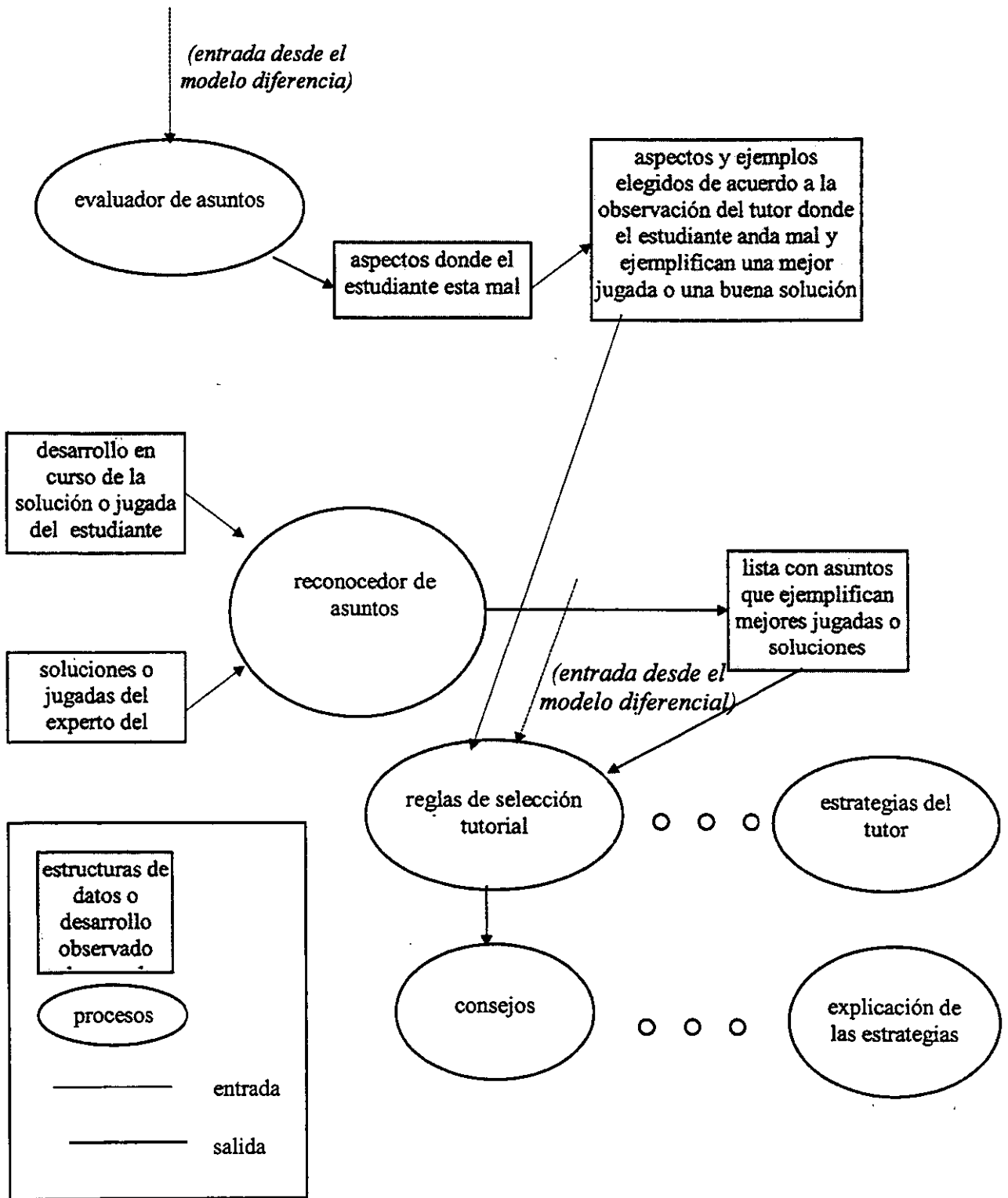


Figura II. 2 - Modelo Tutorial con asuntos y ejemplos de Burton y Brown [BURR82] Parte 1(b)



La Parte 1(b) representa el nivel de abstracción mas alto en la intervención tutorial con un tipo *entrenador*. Cuando el estudiante realiza una jugada no considerada como óptima (comparándola con el experto), el entrenador utiliza al *evaluador de asuntos* para crear una

lista de asuntos en los cuales el estudiante anda mal, por otro lado cuenta con la lista del experto sobre las mejores jugadas; entonces el entrenador invoca al *reconocedor de asuntos* para determinar que *asuntos* están registrados en las mejores jugadas. A partir de estas dos listas, el entrenador selecciona un *asunto* de la *lista de asuntos* que le fallan al estudiante y una jugada buena del experto que contenga este *asunto* para mostrarla. Si no existen asuntos en común en las dos listas, se deduce que el problema del estudiante esta fuera del alcance del tutor y el entrenador no dice nada. En base a otros principios tutoriales se decide interrumpir o no al estudiante, pero, si el entrenador lo decide, el asunto y el ejemplo se pasan al alumno; se espera que esta acción genere una retroalimentación en el estudiante.

Un SEI puede utilizar una de estas dos técnicas para la intervención tutorial o combinarlas, lo que se hace frecuentemente.

En Makatsiná utilizamos la técnica de seguimiento de asuntos para la intervención tutorial, combinada con un tutor tipo entrenador. Se detectan las habilidades que el estudiante utiliza de forma errónea; para lograrlo se utiliza el mecanismo propuesto por Burton et al. [BURR82].

II. 5 La Interfaz

Representa un elemento muy importante en la arquitectura de los SEIs, ya que cubre varias actividades en el funcionamiento global del sistema:

- Es el puente de comunicación entre el estudiante y el sistema.
- Es el único medio físico para poder captar el desarrollo del estudiante.

Las siguientes actividades tienen enfoque de herramientas didácticas.

- Representa el medio a través del cual el tutor (sistema) realizará las intervenciones.
- De acuerdo al dominio de enseñanza el potencial de la interfaz debe ser explotado al máximo, utilizando los medios más adecuados (video, audio, etc.) para la mejor comprensión de los conceptos o el manejo de habilidades.

II. 5. 1 La ayuda

Esta ayuda se refiere básicamente a la proporcionada cuando el estudiante esta resolviendo algún problema, pero no incluye la que necesita inteligencia artificial, la cual, se encuentra considerada en *el diseño del curriculum y la enseñanza*.

Los SEIs cuentan con diferentes tipos de ayuda:

II. 5. 1. 1 Ayuda del sistema

La gran mayoría de las interfaces de los SEIs cuentan con este tipo de ayuda que es útil cuando el estudiante la solicita o cuando se comenten errores. Por ejemplo, MACSYMA [GENM78] construye un plan que explica las acciones que pueden haber conducido al error y propone una lista de probables 'misconceptions' causantes de que el usuario cometiera ese error.

II. 5. 1. 2 Asistencia

Se trata de mecanismos que permiten desarrollar parte de la tarea o la totalidad de ésta. Este tipo de ayuda deja al estudiante concentrarse en las partes donde encuentra dificultad y dejar las partes que domina al sistema. En Algebra Land de Brown & Foss y Algebra Tutor de Anderson [BURR88] los estudiantes cuentan con esta opción que les permite ver como son realizadas las operaciones.

II. 5. 1. 3 Herramientas de poder

Se refiere a la captura de decisiones y acciones del estudiante en estructuras para su posterior observación. Este tipo de mecanismo ayuda al estudiante a reflexionar en las actividades desarrolladas durante la solución del problema. Algebra Land y Geometry Tutor de Anderson & Skwarecki [BURR88] cuentan con esta ayuda. En el caso particular de Algebra Tutor cada decisión del estudiante es capturada en una estructura de árbol que le permite visualizar el espacio de búsqueda:

II. 5. 1. 4 Reactivo

En este caso el sistemas dispone de mecanismos que responden inmediatamente a la acción del estudiante, en el contexto de la situación específica. SOPHIE - I [BROJ75]. Un ejemplo de entorno reactivo es reaccionar a la solicitud de evaluar las hipótesis relativas a las medidas que el estudiante tiene, no le dice que esta mal, dado que la lógica puede ser correcta de acuerdo a las medidas que tiene el estudiante. En el caso de que sea inconsistente la hipótesis propuesta, lo confronta con ejemplos que tienen esa característica (de forma correcta) y que el estudiante esta pasando por alto. Una característica de estos sistemas es que permiten al estudiante articular sus hipótesis en forma opuesta a aquellas que ellos realizan.

II. 5. 1. 5 Modelado

En este caso el sistema desarrolla la tarea mientras el estudiante observa. La ayuda permite al estudiante observar como se comportaría un experto en el desarrollo de la tarea; para lograrlo es necesario que el modelo del experto articule las decisiones y que estas coincidan con las

estrategias de selección de esas decisiones. El experto articulado de SOPHIE - II [BROJ82] es un buen ejemplo, ya que permite al estudiante crear una avería en el circuito y hacerlo funcionar. El estudiante realizará las medidas hasta donde le sea posible y posteriormente el experto explicará cada medida; el por qué se ha realizado (en base a sus estrategias) y al análisis cualitativo del sistema.

II. 5. 1. 6 Entrenador

El tutor se ha desarrollado con el objetivo de seguir de cerca al estudiante en la realización de la tarea e interrumpirlo cuando esta realización no sea óptima o se haya equivocado, proporcionándole sugerencias en estos casos. Es importante mencionar que para la implementación de este tipo de mecanismo es necesario contar con el *camino del experto*. Ejemplos ya clásicos son WUSOR y WEST [BURR82]; ambos tutores se interrumpen cuando el estudiante realiza un mal movimiento y dan consejos. Crean el modelo del estudiante comparando el desarrollo de éste con el del experto. Con WEST se demostró que, aunque no se pueda rastrear todo lo que el estudiante hace, se pueden reconocer patrones de un desarrollo no óptimo.

En Makatsiná se manejan varios tipos de ayuda, que son herramientas de poder (se guarda todas las acciones desarrolladas por el estudiante, permitiéndole observar en cualquier momento los puntos de decisión), modelado (en cualquier instante cuenta con la opción de poder ejecutar parte o toda la tarea por el experto) y entrenador (filosofía con la que fue desarrollado el modelo tutorial).

II. 5. 2 Multimedia y cognición

La interfaz es tan importante que podríamos pensar en ella en función de los sentidos que captan su funcionamiento al interactuar con algún sistema.

II. 5. 2. 1 Los Multimedia como herramienta didáctica

En Rickel [RICJ89] se menciona que las personas retienen aproximadamente el 25% de lo que escuchan, el 45% de lo que escuchan y observan, y el 70% de lo que escuchan, observan y desarrollan. Estos datos representan un potente argumento para desarrollar interfaces que incluyan texto, sonidos y gráficas; además de que tengan la capacidad de generar una interacción con el estudiante como si éste fuese parte del sistema.

Un SEI que utiliza este tipo de interfaz es el citado en Rickel [RICJ89]. Los autores del sistema consideran que la descripción gráfica utilizada en relaciones causales y en topologías son muy importantes. Algunos conceptos pueden ser presentados utilizando gráficas, lo que

facilita la comprensión del alumno. Por otro lado *manejar una interfaz en base a ideogramas permite una conexión más profunda entre el estudiante y el sistema.*

II. 5. 2. 2 El aprendizaje por asociación (cognición)

Es importante mencionar que el uso de multimedia tiene una relación directa con el aprendizaje por asociación como se menciona en Laureano [LAUA93](2). La memorización por asociación puede ser mecanizada, esta reflexión nos conduce directamente al uso correcto de multimedia, con el objeto de mecanizar conocimiento procedimental.

El método de aprendizaje por asociación descrito en el año 55 a. de c. por Cicerón y ampliado por Godden y Baddeley indica que la memoria no sólo depende de un conjunto de trucos formalizados para recordar elementos ubicados en distintas localidades físicas, sino que, está basada en algo más profundo que depende del contexto de aprendizaje. De la reflexión anterior se desprende la importancia de modelar interfaces con una *fidelidad de display* profunda, sobre todo en principiantes.

Sin embargo, el hecho de que la asociación juegue un papel importante en la forma de ligar nuestros esquemas mentales personales en un todo coherente, no implica que ésta sea la mejor forma de impartir información y conocimiento. Una evidencia indirecta contra los métodos asociativos es el éxito, durante mas de 500 años, del libro convencional y el desarrollo de la narrativa durante más de mil años. Esto por otro lado tiene las siguientes desventajas:

- No existe una respuesta inmediata al estudiante.
- Los estudiantes son entes pasivos.
- No se generan ejercicios de distintas complejidades de acuerdo a la habilidad de los estudiantes.

Desventajas que se convierten en ventajas al hablar de los SEIs, de aquí que los sistemas multimedia, por el momento, no sean más que un conjunto de técnicas experimentales útiles en casos específicos que se separan de la corriente principal del pensamiento educacional convencional.

Resumiendo, en la interfaz de un SEI se pretende que la entrada de la información sea robusta, pero a su vez que permita flexibilidad en la entrada de información sintetizada que se desprende de la observación del desarrollo del estudiante. En cuanto a la información de salida, esta debe ser rica, esto es, *deletreada al máximo*, utilizando la potencia de los sistemas multimedia; aprovechando así todos los sentidos del estudiante, *sin caer en el exceso.*

En Makatsiná se hizo especial hincapié en el desarrollo de una interfaz controlada a través de ideogramas. Las intervenciones tutoriales, y la interacción del estudiante son a través de gráficas. Lo anterior debido a que el dominio de enseñanza es considerado ad-hoc para ser manejado por ideogramas. En el pizarrón se explica con gises de colores y un continuo escribir al lado de las líneas que conforman la estructura triangular, para mostrar los sentidos de las fuerzas de equilibrio, donde también éstas tienen diferentes colores de acuerdo al tipo de esfuerzo (tensión o compresión).

En cuanto al uso de Multimedia, se considera un dominio que se presta al aprendizaje por asociación. En Makatsiná se utilizaron efectos visuales de animación, colores y BIP's de sonido para indicar errores o felicitar en caso de éxito.

II. 6 Trabajos relacionados

Existen algunos SEIs que han utilizado arquitecturas MultiAgente para su implementación. Por otro lado también se han utilizado agentes autónomos con características pedagógicas. A continuación describiremos algunos.

II. 6. 1 Arquitecturas MultiAgente

El trabajo realizado por Girard, Gauthier y Levesque (1992) se hace una propuesta de diseño basada en una arquitectura MultiAgente muy interesante. Para ser implementada en un SEI que podríamos definir como tradicional si no fuera por el hecho de que todo el sistema incluyendo el alumno y el profesor están considerados como agentes. La motivación de esta propuesta parte de considerar a la actividad humana de enseñar como una actividad cargada de oportunismo; de aquí que el mecanismo que desarrolla un SEI sea considerado como un proceso de planificación oportunista. El SEI esta formado por: tutor, planificador, curriculum y MicroMundo.

Desde el punto de vista de la inteligencia artificial tradicional, la planificación es vista como una exploración de un espacio de estados; esta técnica tiene la desventaja bien conocida de la explosión combinatoria y de la 'resistencia a las perturbaciones'. Este último concepto se refiere al hecho de que un sistema durante la resolución de un problema por exploración de un espacio de estados sea capaz de tener en cuenta los cambios que se producen en el entorno, durante esa exploración; los cuales entrañan modificaciones en el problema a resolver. En otras palabras, es la incapacidad de planificar dentro de un mundo en continuo cambio, esto es, dinámico. En la mayor parte de los casos este tipo de sistemas se contenta con re-comenzar la exploración. Provocando un retraso de la respuesta en el mejor de los casos. De aquí que la distribución de las actividades sea una solución prometedora.

En este trabajo un agente es una entidad real o abstracta que es: a) capaz de elegir sobre el mismo y sobre el entorno, b) dispone de una representación parcial del entorno y c) dentro de un universo MultiAgente es capaz de comunicarse con otros agentes. El comportamiento de este agente es producto de sus observaciones, de sus objetivos y de sus interacciones con otros agentes. (Ferber, 1989).

Esta forma de aproximación se basa en la interacción y el propósito de hacer cooperar a muchos agentes especialistas, en lugar de utilizar un programa central y general para resolver un problema como lo hace la inteligencia artificial tradicional.

Los componentes del sistema emergen con una semántica de capas que se obtienen de una descomposición del problema, que en este caso es enseñar. Con esta descomposición se podrá conocer los agentes, sus responsabilidades y sus interacciones.

Las capas obtenidas pueden ser vistas de forma conceptual como: los puntos donde se toman decisiones, como los niveles de abstracción de un plan que constituye la solución de un problema o como soluciones parciales del problema a resolver.

Así los agentes del sistema quedan constituidos por: humanos y programados. Los humanos son: el alumno y el profesor, y los programados son: el planificador, el tutor y el MicroMundo.

El curriculum no forma parte de los componentes activos ya que no influye en el comportamiento de un SEI durante el proceso de enseñanza. El es un objeto de comunicación sobre el que recaen las interacciones del agente profesor y del agente planificador.

Finalmente el proceso de enseñanza se realizará en base a dos políticas: 1) la planificación para conjeturar, anticipar y provocar los eventos pertenecientes a la enseñanza y 2) la evaluación para observar, analizar y comprender los eventos de la enseñanza.

Por otro lado la responsabilidad de cada agente en base a estas dos políticas tiene un comportamiento interno y externo. 1) Visto desde el interior cada agente debe conjeturar los eventos productos de la enseñanza, y analizar y comprender otros eventos productos de la enseñanza en estado de evaluación. 2) Visto desde el exterior cada agente debe provocar los eventos de enseñanza para planificar y observar la evaluación.

La propuesta de este trabajo es interesante por dos aspectos. El primero es que trata de aproximarse desde un aspecto de distribución de tareas a la imposibilidad de poder considerar los cambios antes de que el sistema regrese de su inferencia sobre su mundo simbólico. Y la otra es que contempla la representación parcial en cada uno de los agentes en base a sus tareas. La descomposición es vista en un grano más grueso que en la filosofía

reactiva. Otro aspecto importante es el oportunismo del proceso de enseñanza mismo que puede ser logrado si por separado cada cual busca su mejor oportunidad de actuación.

En el trabajo realizado por Néhémie (1992) se propone una forma MultiAgente para modelar al estudiante en los diferentes sistemas de enseñanza entre los que se destaca a los SEI. Basa su estudio en una arquitectura de pizarrón, implementada de forma MultiAgente. Su principal aportación es que se concibe al módulo del estudiante como un sistema abierto; que analiza la información proveniente del sistema global. Este módulo es un agente que modela al estudiante y lo hace en base al análisis de la información proveniente de varios flujos.

Identifica dos flujos de información, dependiendo de las condiciones de operación. 1) Las entradas; su objetivo es asegurar que el modelo del estudiante sea actualizado regularmente y que incluya eventos recientes que hayan ocurrido durante la sesión. En el contexto de las arquitecturas de pizarrón este flujo se logra en base a la comprobación de los reportes de varias actividades e interacciones que son llevadas a cabo en el pizarrón por diferentes agentes. 2) Las salidas que produce este agente durante el proceso de aprendizaje son un modelo sintético del estudiante con un resumen de sus tendencias. Este modelo esta disponible de forma permanente a los otros agentes y se mantiene actualizado.

II. 6. 2 Agentes pedagógicos

Rickel y Johnson (1997) desarrollan un agente autónomo con propósitos pedagógicos dentro de un entorno virtual de aprendizaje. Los entornos virtuales son utilizables en entrenamiento, especialmente donde la vida depende de ello; como es combate aéreo, o en procesos de manufactura complejos.

El dominio representado en el entorno esta relacionado con el entrenamiento de uso y compostura de maquinaria compleja. El agente en cuestión se llama Steve y esta capacitado para verificar y manipular entornos virtuales dinámicos. También puede adoptar distintas formas como el de una figura humana o manos que realizan tareas de apuntadores.

Steve utiliza una serie de capacidades inteligentes durante sus interacciones con el estudiante y el entorno que le permiten realizar acciones de: revisión o ejecución de un plan, explicaciones y un proceso de monitorización sobre el desarrollo del estudiante.

Un agente virtual autónomo puede ser de invaluable valor cuando los estudiantes no reconocen que sus acciones son inapropiadas o simplemente no son óptimas; en cuyo caso un agente virtual puede intervenir con consejos apropiados. Otras veces pueden encontrarse con situaciones que no son familiares y debido a la insuficiencia de conocimiento para afrontar la situación se podrían ver beneficiados si tuvieran alguien que les guaira, contestara sus dudas

o mostrara el procedimiento. Otro aspecto importante es que pueden simular el hecho de perder personal, permitiendo al estudiante entrenarse en tareas multi-persona sin necesidad de otros humanos.

Steve *habita* en el entorno virtual y registra constantemente el estado del entorno; el cual periódicamente controla a través de acciones virtuales. El objetivo de Steve es ayudar en el aprendizaje del desarrollo de tareas procedurales; como la operación y reparación de dispositivos complejos. Todas estas habilidades tutoriales están integradas en un agente.

Steve consta de dos componentes: 1) el cognitivo implementado en SOAR (Laird, Newel & Rosenbloom, 1987) que maneja el procesamiento cognitivo de alto nivel y 2) el que maneja el motor de detección. El componente cognitivo: interpreta el estado del mundo virtual, lleva a cabo planes para lograr objetivos y toma decisiones con respecto a acciones. El componente del motor de detección es la interfaz de Steve al mundo virtual, permitiendo al componente cognitivo percibir el estado del mundo virtual y provocar cambios en él.

En el caso de Steve podríamos considerar al componente cognitivo como un SEI, capaz de decidir cuándo, cómo y que decir.

En el trabajo de Lester y Stone (1997) se enfatiza la confianza basada en la credibilidad que estos agentes pedagógicos deben dar al estudiante. La confianza es producto de dos fuerzas: la calidad visual del agente y la maquina-de-secuencias-de-comportamientos que los administra. Esta administración se realiza en base a la evolución de las interacciones con el usuario.

Lester y Stone (1997) desarrollan un marco para estos agentes representados por distintos comportamientos de animación pedagógica, de acuerdo a sí son o no son propicios en ese momento. El derecho de hacerse presentes depende de sí son o no apropiados para una determinada situación en curso con el usuario.

Estos comportamientos compiten por el derecho de ser exhibidos. Asociados a ellos esta un parámetro que se llama fuerza-de-exhibición, la cual es una medida del grado de propiedad para aparecer en el estado en curso. El administrador de comportamientos decrementa o incrementa este parámetro. En cada tic de reloj se les premia o se les penaliza con puntos de acuerdo a sí son o no propios para ese momento. A continuación el administrador analiza ese puntaje entre todos los elegibles evaluando sus parámetros de acuerdo a: la historia de la solución del problema desarrollada por el estudiante y la historia del comportamiento de ese agente. El más fuerte es el elegido permitiendo que se exhiba. Posteriormente es marcado como exhibido y se le pone a cero su fuerza-de-exhibición y continúa la competencia.

Estos agentes están implementados en un entorno de biología y psicología. Los diferentes comportamientos están representados en un insecto que se llama Hernan. Hernan esta capacitado para poder: volar, crecer, nadar, pescar, caminar, brincar y realizar series acrobáticas.

Un punto importante en esta implementación es la capacidad de generar distintos comportamientos ad-hoc a los motivos pedagógicos. Y la forma del administrador de comportamientos para premiarlos o penalizarlos; que es registrando el desarrollo y comparándolo con los comportamientos ad-hoc para determinadas situaciones. Una vez que se elige un comportamiento este es un agente autónomo que es capaz de interactuar con el estudiante.

II. 7 Nuestra aproximación

En este capítulo han quedado plasmadas las diferentes técnicas que utilizan los sistemas de enseñanza inteligentes para diseñar los diferentes módulos constitutivos de un SEI. Se ha hecho especial hincapié en aquellas utilizadas por el sistema Makatsiná.

Resulta claro que para lograr el objetivo de aprendizaje existe además del modelo del estudiante el control que debe realizar el módulo tutorial.

Bien nuestra propuesta no utilizará modelo de estudiante, ni ningún tipo de control en el módulo pedagógico. Lo que en términos de la filosofía reactiva se traduce en no contar con objetivos previamente predeterminados, ni representación exhaustiva del entorno. Proponemos un sistema que base sus acciones en lo que observa. Siguiendo la filosofía reactiva creada por Brooks.

Lo que ganamos con esta nueva arquitectura es: 1) tiempo de respuesta y 2) contar con subexpertos en distintas habilidades; donde los errores serán tratados de forma puntual. Lo anterior abre la posibilidad de un diagnóstico de errores, que se realizará de forma exhaustiva en los diferentes subexpertos. La posibilidad de ser expertos en una sola habilidad es difícil de encontrar en los instructores humanos, en general su conocimiento es global.

II. 7. 1 Metodología

Para lograrlo estudiaremos la forma en que lo hacen los robots móviles. La aplicación de los agentes reactivos ha estado centrada principalmente en ellos.

Analizaremos las ideas básicas de acuerdo a Brooks, considerado uno de sus principales exponentes. Este punto es de primordial importancia ya que para lograr este tipo de diseño, es necesario pensar de forma distinta desde la etapa de análisis y diseño.

La corriente reactiva es más experimental que formal, de aquí que estudiemos tres distintas arquitecturas. De cada una de ellas extraeremos conclusiones como: qué tipo de niveles conductuales tienen los SEIs, qué métodos de diseño global existen en los sistemas reactivos, qué tipo de mecanismos de arbitraje utilizan los robots y cómo pueden ser aplicables a los SEIs. Todo ello nos permitirá, traspolar esta metodología de diseño a los sistemas de enseñanza inteligentes.

Posteriormente con las conclusiones obtenidas del estudio anterior se sabrá: 1) qué método de diseño global se utilizará y 3) qué mecanismo de arbitraje se adaptará al SEI.

Por otro lado se realizará el análisis y diseño del entorno cognitivo, cuyo resultado será: 1) el número de agentes necesarios y su especialización y 2) el número de niveles. Una vez especificados los agentes se continuará con su formalización conductal. Para finalizar con la implementación del SEI.

Capítulo III

Agentes reactivos

III. 1 Introducción

El principal objetivo de este capítulo* es dejar claro los elementos que tienen que ser considerados en un sistema de enseñanza inteligente (SEI), con el fin de aplicar la filosofía reactiva.

Para lograrlo se expondrán los conceptos de los agentes reactivos y lineamientos de esta corriente. Se analizarán tres arquitecturas de robots que nos permitirán ver reflejados estos lineamientos. Posteriormente se realizará una analogía entre robot y sistema de enseñanza inteligente que nos permitirá esclarecer los elementos y qué características deben de tener en un SEI para lograr la reactividad.

Los agentes de software se han desarrollado rápidamente. Existen muchos tipos de agentes como agentes: colaborativos, de interfaz, móviles, informativos (internet), reactivos, deliberativos, híbridos e inteligentes. En Nwana [NWAH96] y Muller [MULJ98] se encuentra una discusión profunda de sus características de acuerdo a la topología de cada autor. En este trabajo el foco de interés se centra en los agentes reactivos. Son estos últimos

* Los contenidos presentados en este capítulo amplían las ideas aportadas en las publicaciones referenciadas como: [LAUA97] (2) y [LAUA98] (2).

los que describiremos haciendo algunas veces una comparativa con los agentes deliberativos; considerados como su contra parte.

Uno de los principales objetivos de la IA, es tratar de desarrollar mecanismos de selección - acción que puedan ser utilizados por agentes autónomos para determinar qué acciones tienen que desarrollar los sistemas inteligentes.

Desde 1970 los fundamentos de las investigaciones en IA radican en que una tarea inteligente, puede ser implementada, basándose en un proceso de razonamiento que opere sobre un modelo simbólico interno es decir, un agente tiene una representación interna de acciones, objetivos y eventos. Esta representación es utilizada, por el proceso de razonamiento (llamado planificación en robots) para determinar que secuencia de acciones lograrán el objetivo. Esta corriente ha tenido éxito en algunos campos, pero sus resultados no han sido los esperados cuando hablamos de agentes autónomos operando en entornos dinámicos y complejos, de aquí que la inteligencia artificial distribuida (IAD) en los últimos 10 años haya tenido avances significativos. Dentro de esta rama contamos con las arquitecturas Multi-Agente y dentro de esta última se encuentran los *agentes reactivos*; en Ferber y Drogoul [FERJ92] y Ferber [FERJ93], se explica con más detalle las características de la IAD.

III. 1. 1 Agentes deliberativos

En la IAD tradicional los sistemas están compuestos por un grupo de agentes, donde cada uno de ellos puede ser considerado como un sistema experto en sí, que cuenta con una base de conocimiento que le permite realizar su tarea, así como una vez desarrollada, comunicar los resultados a los demás agentes. En realidad los sistemas tradicionales son vistos como solucionadores de problemas, donde los agentes basan su mecanismo en la intencionalidad; para ello cuentan con objetivos y planes explícitos que les permiten lograr el objetivo final. Los problemas que se presentan son de cooperación, donde varios agentes deben coordinar sus actividades y ocasionalmente resolver los conflictos que se puedan presentar.

III. 1. 2 Agentes reactivos

La otra corriente busca que los agentes no sean inteligentes de forma individual, pero sí, que su comportamiento de forma individual genere un comportamiento inteligente de forma global. Estos agentes no poseen modelos internos del entorno y basan su acción en un ciclo de estímulo-reacción, de acuerdo a los eventos que se presentan en el estado en curso del entorno en el cual están embebidos; de ahí que no cuenten con objetivos, ni mecanismos de planificación explícitos, y aún así puedan resolver los problemas clasificados como complejos. Esta escuela es menos representativa y más experimental en contraposición con la

deliberativa, que es más formal; sin embargo el uso de estos sistemas se está extendiendo a otros campos como simulación, juegos [FERJ92] y sistemas de enseñanza inteligentes (SEIs) [LAUA97](1) y [LAUA97](3).

Uno de los puntos más importantes es el hecho de que son relativamente simples e interactúan con otros agentes de forma básica. No obstante comportamientos complejos pueden emerger cuando el ensamblado global entre los agentes y el entorno es visto.

Maes [MAEP93](1) menciona tres puntos claves relativos a estos agentes: 1) la funcionalidad emergente; lograda en base a la dinámica de la interacción de estos agentes con el entorno, 2) la descomposición de tareas: los agentes reactivos son vistos como una colección de módulos que operan de forma autónoma y son responsables de tareas específicas. La comunicación entre éstos cuando existe es mínima y se da en niveles bajos. No existe un modelo global dentro de cada agente, sin embargo, es alcanzado el comportamiento global. 3) Los agentes reactivos tienden a operar con representaciones cercanas a los datos sin tratar.

La hipótesis esencial de los sistemas basados en agentes reactivos es que la especificación se realiza en base al entorno real, no a uno simbólico. La inteligencia artificial tradicional aseguraba que para demostrar que una acción fuera inteligente la condición suficiente y necesaria es que fuera desarrollada por un sistema que tuviera una representación simbólica del entorno. Por el contrario la corriente reactiva sostiene que para construir un sistema inteligente es necesario tener una representación basada en el mundo real.

Así la hipótesis de los agentes reactivos puede ser descrita de esta forma: los sistemas de agentes inteligentes pueden ser desarrollados a partir de agentes simples que no cuenten con modelos simbólicos internos y cuya inteligencia se derive del comportamiento emergente producto de las interacciones entre varios módulos.

Los agentes reactivos son simples y fáciles de entender [FERJ94]. Ellos están ubicados en el mundo, no planifican y no revisan ningún modelo previo del mundo. Sus acciones dependen de lo que sucede en el mundo en un momento preciso.

Dentro de los beneficios en el uso de agentes reactivos están: son más robustos y tolerantes a las fallas que otros sistemas basados en agentes; un agente puede perderse sin efectos catastróficos. Además son más flexibles y adaptables en contraste con la rigidez, lenta respuesta, y fragilidad por parte de los sistemas con agentes deliberativos.

Esta corriente ha sido criticada por el hecho de tener pocas aplicaciones (simulaciones y juegos). Sin embargo hay que hacer notar que esta corriente se encuentra en sus inicios y si la inteligencia artificial simbólica tardó dos décadas en llegar a las aplicaciones industriales

(hasta finales de los 80), aún le falta camino por recorrer a la corriente reactiva. Es necesario expandir los idiomas, teorías, arquitecturas y aplicaciones.

III. 1. 3 Aproximación de las arquitecturas reactivas

Las arquitecturas reactivas empiezan a ser investigadas en los 80's, por Maes [MAEP93](1) y Brooks [BROR91](2); donde se menciona la nueva tendencia de desarrollo de estos sistemas que comienza el modelado de forma 'bottom-up', por el momento centrada básicamente en los robots móviles. Esta corriente está en desacuerdo principalmente en dos puntos referentes a la forma de modelado de los sistemas inteligentes dentro de la corriente de la inteligencia artificial tradicional. Estos puntos son: la modelización exhaustiva del mundo y la representación explícita de objetivos dentro del sistema. A estos sistemas se les conoce como sistemas basados en la conducta, siendo sus características principales:

- Interacción dinámica con el entorno.
- Mecanismos internos que permitan trabajar con: recursos limitados y con información incompleta.

En estos sistemas la funcionalidad emergente se dará en base a una interacción intensiva del sistema con el entorno, [BEEJ90], [BROR91](1) y [MAEP93].

La especificación de conducta de un agente de forma individual no explica la funcionalidad de dicho agente visto en acción, lo que significa que ésta depende en gran proporción de las propiedades dinámicas del entorno al interactuar con el sistema (agentes).

Interesa modelar las características dinámicas del entorno, ya que serán éstas las que den la pauta al funcionamiento del sistema.

Esta filosofía de diseño lleva a la siguiente reflexión: en este tipo de arquitecturas no es posible decirle de forma simple a los agentes cómo alcanzar un objetivo; más bien desde la etapa de diseño hay que encontrar qué características del entorno tienen que ser tomadas en cuenta. Con lo anterior se busca diseñar un ciclo de interacción entre los agentes y el entorno, que permita converger hacia el objetivo deseado; esta interacción sistema (agentes) \Leftrightarrow entorno terminará en el instante en que los objetivos sean alcanzados.

Esta forma de ver la acción de un sistema implica que es en tiempo real cuando se determinará la acción de los agentes para un objetivo, es decir, la acción dependerá de la manera en que se vayan presentando las características del entorno (a ser analizadas) y del objetivo final. Esta nueva forma de control conlleva implementaciones distribuidas al tener interacción paralela local entre agentes y entre agentes y el entorno.

La dinámica de selección de acciones de este tipo de sistemas emergerá en respuesta a dos aspectos: las condiciones del entorno y los objetivos internos de cada agente, lo que nos hace concluir que las selecciones de acción de un agente son realizadas en tiempo real.

Resumiendo, lo que se busca con estas arquitecturas es que la deliberación de los mecanismos de selección esté implícita, es decir, que sea una propiedad emergente de un proceso dinámico. En este tipo de arquitecturas los esquemas de control del sistema emergerán de una manera distribuida, basándose en interacciones locales entre un conjunto de módulos simples.

Uno de los principales investigadores de esta nueva forma de ver las cosas es Rodney Brooks [BROR86]. Una vez que nació la arquitectura base, diversos autores han contribuido con sus ideas y se han propuesto infinidad de cambios, para mejorar las implementaciones en los diferentes robots, de acuerdo a las tareas que se desea que desarrollen o adaptarlas a diferentes arquitecturas para ser implementadas en otro tipo de sistemas inteligentes. [KAEL87], [SANJ88], [AGRP87], [ROSS87], [NILN89], [SCHM87], [GARM95], [LAUA96](2), (3), [LAUA97](1), [NEGJ97](1) y [NEGJ97](2).

Nuestro trabajo crea una nueva arquitectura de SEI [LAUA97](3) y [LAUA98](3), relacionada con las arquitecturas de Brooks [BROR86], Kaelbling [KAEL87] y la de García-Alegre, Bustos & Guinea [GARM95], para su adaptación en los SEIs.

En la siguiente sección se hará un acercamiento a los lineamientos de partida para poder crear sistemas con agentes reactivos.

III. 2 Antecedentes

III. 2. 1 Ideas de Brooks sobre cómo lograr un comportamiento inteligente con agentes autónomos [BROR91](2)

Las siguientes ideas representan la nueva forma de ver y pensar en los sistemas inteligentes, cuando consideramos la forma de modelarlos para su posterior construcción.

III. 2. 1. 1 Ubicación (situatedness)

En la construcción de los sistemas basados en los principios de la inteligencia artificial tradicional, los agentes son construidos de forma que trabajan con un dominio abstracto y simbólico, en el que estos símbolos tienen relación con el mundo de acuerdo al diseñador que los construyó, pero no existe nada que fundamente estas relaciones con el mundo real; es más, los agentes no se encuentran ubicados en el mundo real, en realidad son solucionadores

de problemas, dónde, dado uno, es resuelto por ellos. En estos sistemas no existe el mundo externo con sorpresas y continuidad, lo que en realidad existe es una confusión entre los conocimientos que el agente tiene del mundo y el mundo en el cual se supone debe operar; en otras palabras, los agentes tienen acceso a una percepción ideal que conlleva un isomorfismo entre el mundo real y el mundo simbólico lo que conduce a una acción directa y perfecta.

En el modelado de los sistemas basados en la conducta se desarrolla una aproximación diferente, donde el robot móvil utiliza el mundo real como su modelo, referenciándolo continuamente a través de sus sensores, en vez de hacerlo hacia un modelo interno del mundo.

En este tipo de sistemas las representaciones son en términos de las relaciones que se establecen a partir de los *objetos del mundo real* hacia el robot y no se tienen representaciones de estos *objetos del mundo de forma individual*. Pensando en esta forma de representación, se hace posible construir sistemas computacionales que cambien la implementación de los cálculos en profundidad por la implementación de los cálculos a lo ancho.

Una acción ubicada de los agentes, es una acción adecuada a su situación; donde la situación es relativa a la combinación compleja que se da entre los eventos internos y externos.

La clave de esta idea es: el mundo real es el mejor modelo

III. 2. 1. 2 Incorporación (embodiment)

La incorporación es vista en términos de la participación activa en lo que sucede en el mundo, en base a una percepción también activa de ese mundo. Básicamente existen dos razones por las que la incorporación resulta crítica en este tipo de sistemas:

Sólo un agente incorporado, es capaz de ser validado totalmente para enfrentarse al mundo real.

Sólo a través de una interacción con el mundo físico puede un simbolismo interno encontrar significado para trabajar de forma bottom-up o en forma interna al sistema.

Sin una incorporación real, no existe significado para un agente. Johnston, comentado en [BROR91](2), sugiere que gran cantidad de nuestro lenguaje es actualmente metafórico y está relacionado con nuestras conexiones físicas al mundo. Por otro lado Smith, comentado

en el mismo artículo, menciona que nuestros conceptos mentales están basados en experiencias físicas ejemplares. Smith sugiere que en los sistemas basados en el conocimiento sin una interacción con el mundo físico podría no existir un retorno. Esto es, que el sistema con su mecanismo de inferencia representado en forma de enciclopedia podría no proporcionar un camino de retorno dado que sus razonamientos son sin considerar el mundo real.

La clave de esta idea es: el mundo fundamenta el retorno en un mecanismo de inferencia.

III. 2. 1. 3 Inteligencia

Brooks, considera que la clase de actividades en que usualmente pensamos, para demostrar la inteligencia en los humanos, han tenido un lugar de tamaño reducido, dentro de la historia de toda nuestra evolución. También subraya que las actividades simples que se realizan utilizando percepción y movilidad en un entorno dinámico, necesariamente han tenido que ser perfeccionadas durante un largo tiempo y que estas capacidades son la base del intelecto de un nivel superior; de aquí su propuesta de observar animales con comportamientos sencillos, para aprender a modelar de forma bottom-up, con la consecuente construcción de sistemas inteligentes.

Simon, comentado en [BROR91](2), hace una reflexión, sobre una hormiga que camina a lo largo de una playa y nos hace notar que la complejidad observada en el comportamiento de la hormiga es más un reflejo, dada la complejidad del entorno, que debido a la complejidad interna de la hormiga; de aquí que Brooks proponga a la interacción robot \Leftrightarrow entorno, como determinante primaria en la estructura de la inteligencia de un robot.

De forma concluyente podemos decir que es difícil trazar una línea divisoria entre la la inteligencia y el entorno dinámico en el cual se desenvuelva un sistema y carece de importancia. lo que sí hay que tener presente es que los sistemas inteligentes deben ser situados en algún mundo o de otra forma carecerán de utilidad.

La clave de esta idea es: la inteligencia es determinada por la dinámica de interacción con el mundo

III. 2. 1. 4 Emergencia (emergence)

Cuándo nos preguntamos dónde podemos encontrar la inteligencia de un sistema, la respuesta se torna difícil de contestar; dado que es difícil ubicar el corazón de un programa. En el sólo hay ciclos y secuencias de operaciones primarias, así que en lugar de preguntarnos en dónde se ubica la inteligencia, en realidad debemos decir que la inteligencia es un

comportamiento que emerge como resultado de la interacción de los componentes del sistema. La forma en que emerge es distinta en los sistemas que utilizan los fundamentos de la inteligencia artificial tradicional y aquéllos que son el punto de nuestro interés. Brooks realiza una aproximación informal de los métodos de trabajo en las dos corrientes.

En la inteligencia artificial tradicional, los módulos están definidos como procesadores de información o funcionales, siendo los módulos típicos de los robots-móviles los siguientes: percepción, planificador, modelador del mundo y aprendizaje; *el comportamiento inteligente del sistema* como evitar obstáculos, pararse, o controlar la visión, emerge como resultado de la interacción de los módulos mencionados con anterioridad.

En el caso de los sistemas inteligentes basados en el comportamiento, los módulos definidos son *productores de comportamiento* como: evitar obstáculos, pararse, controlar la visión, etc. En estos sistemas la inteligencia funcional del sistema, como percepción, planificador, modelado y aprendizaje, *emerge como resultado de la interacción de los componentes entre ellos y el mundo*.

Esta aproximación resulta ser la parte romántica, porque en realidad hay más problemas; a los sistemas que usan la inteligencia artificial tradicional, les ha sido difícil poder lograr la conexión con el mundo, de aquí que la emergencia del comportamiento inteligente este representada por una expectativa más que por un fenómeno establecido. Por el contrario, en un sistema basado en la conducta, dada la multiplicidad de los comportamientos que proporcionan una dinámica propia y están conectados de forma individual con el mundo. Es difícil con frecuencia decir que una serie particular de acciones ha sido producida por un comportamiento particular.

Steels, comentado en [BROR91](2), define *la funcionalidad emergente* como una función que es alcanzada de forma indirecta dada la interacción entre componentes primitivos y de algunos o todos con el mundo.

De los párrafos anteriores concluimos que la ubicación de la inteligencia en un sistema, resulta difícil, ya que se logra a partir de la interacción de sus componentes. Por esta causa la inteligencia solamente puede ser determinada dado el comportamiento global del sistema y de la forma en que se da este comportamiento en relación al entorno.

La clave de esta idea es: la inteligencia se encuentra en el ojo del observador

III. 2. 1. 5 La representación en los sistemas basados en el comportamiento

En estos tipos de sistemas la representación implica un punto de discusión muy importante, de aquí que sea pertinente hacer una aclaración, para ello se transcribe a continuación un fragmento de Brooks [BROR91](2).

My earlier paper [BROR91](1), is often criticized for advocating absolutely no representation of the world within a behavior-based robot. This criticism is invalid. I make it clear in the paper that I reject traditional Artificial Intelligence representations schemas. I also made it clear that I reject explicit representations of goals within the machine.

Brooks, habla de que pueden existir representaciones pero éstas serán modelos parciales del mundo; de hecho, las capas individuales toman del mundo aquellos aspectos relevantes que les son necesarios para lograr su funcionamiento. Para lograrlo, creó la arquitectura reactiva incluida (subsumption architecture), cuya premisa es subsumir las capas conductuales de niveles inferiores; para lograr la competencia de los superiores.

III. 3 Arquitecturas reactivas

En esta sección se describirán tres arquitecturas utilizadas para la implementación de robots. La arquitectura de Brooks es reactiva pura, la de García-Alegre et al. y la de Kaelbling son híbridas.

III. 3. 1 Arquitectura de Brooks

Brooks [BROR86] creó esta arquitectura teniendo en cuenta las siguientes suposiciones:

1. Una conducta compleja, no necesariamente tiene que ser producto de un sistema de control complejo, más bien una conducta compleja puede ser vista como el reflejo de entornos complejos. Esta suposición está relacionada con la idea de inteligencia (III.2.1.3).
2. En este tipo de diseño la comunicación de las interfaces es importante. Como estamos trabajando con un diseño incremental, se sugiere que si encuentran en algún momento dificultades se regrese y se cambien (reduciendo o agrandando) los módulos.
3. Deseamos construir robots baratos que puedan vagabundear en espacios humanos, sin la intervención de éstos.
4. El mundo humano es tridimensional, no es sólo un mapa de dos dimensiones, así que el robot debe ser capaz de modelarlo en tres dimensiones, para que sea viable la cohabitación con humanos.
5. Los sistemas de coordenadas absolutas son fuente de errores acumulativos; para evitarlo se deben tener mapas relacionales que son mas útiles para los robots móviles, sin embargo, esta idea altera los espacios de diseño para los sistemas de percepción.

6. Si realmente deseamos que los robots nos ayuden en tareas humanas, no se deben construir mundos artificiales perfectos para los robots; esta suposición esta relacionada con la idea de ubicación (III.2.1.1).
7. Aunque los datos de los sensores de ultrasonidos se obtienen fácilmente no representan una fuente rica para describir el mundo siendo los datos visuales mejores para este propósito. Por esta causa el sonar debe utilizarse en las interacciones de los niveles bajos, para evitar obstáculos en tiempo real.
8. Para la robustez, el robot debe ser capaz de darse cuenta cuando un sensor falla o comienza a dar lecturas erróneas y debe ser capaz de recobrase rápidamente; esto implica que el robot debe ser capaz de calibrarse solo, y además debe hacerlo constantemente.
9. Estamos interesados en fabricar robots que puedan sobrevivir por días, semanas y meses sin la asistencia humana en un entorno complejo y dinámico, tales robots deben ser capaces de mantenerse a sí mismos.

Partiendo de estas suposiciones Brooks creó una arquitectura dividida en capas conductuales; esta división es realizada desde el punto de vista de las manifestaciones externas deseadas en el sistema de control del robot. En lugar de fijarse en el trabajo interno de la solución de estas conductas, esta suposición esta relacionada con la idea de inteligencia (III.2.1.3).

A continuación vamos a ver la manera en que él realizó la abstracción conductual de un robot móvil autónomo:

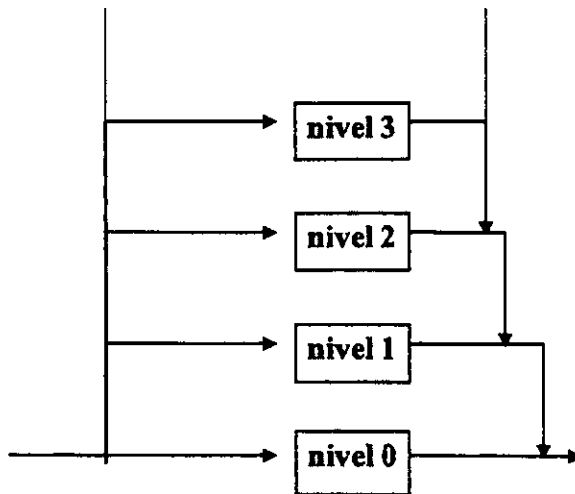
- 0o. Nivel. Evitar contacto con objetos (dinámicos o estáticos).
- 1er. Nivel. Vagabundear sin golpear las cosas.
- 2o. Nivel. Explorar el mundo buscando lugares en la distancia donde se pueda llegar y dirigirse hacia ellos.
- 3er. Nivel. Construir un mapa del medio ambiente y planear rutas de un lugar a otro.
- 4o. Nivel. Darse cuenta de cambios estáticos del medio ambiente.
- 5o. Nivel. Razonar con respecto al mundo en términos de objetos identificables y desarrollar tareas relacionadas con estos objetos.
- 6o. Nivel. Desarrollar y ejecutar planes que cambien el medio ambiente *de forma deseada*.
- 7o. Nivel. Razonar con respecto al comportamiento de objetos del mundo y modificar los planes de acuerdo a ellos.

Es importante hacer notar que cada nivel de competencia (conducta) incluye un conjunto de niveles de competencia más bajos, este conjunto representa capas de control, que funcionan de la siguiente forma.

El *nivel cero* es visto como la cimentación en la granularidad de abstracciones, sobre este nivel se construirá el *primero*, que es capaz de examinar datos provenientes del *nivel cero* y también le es permitido proporcionar datos de manera interna al *nivel cero* a través de las interfaces de este último, suprimiendo así el flujo normal de datos. De esta manera el *nivel primero*, logra su competencia ayudado por el *nivel cero*. Es importante mencionar que independientemente del trabajo de la capa del *nivel primero*, la capa del *nivel cero* continuará trabajando.

Para efectos de nuestro trabajo se describirán algunas características más de esta arquitectura, ya que las demás carecen de sentido en los SEIs.

Figura III. 1 - Arquitectura-Incluida (subsumption- architecture)



Lo anterior se traduce en que, independientemente del nivel de capa en donde se encuentre la acción, a ese nivel no le importarán las capas de arriba, que solamente pueden interactuar con ella a través del flujo de datos. De esta forma cada nivel logra su competencia con la ayuda de las capas que tiene debajo de él. Este proceso continuará hasta el último nivel donde se encontrará la conducta más compleja. A esta arquitectura Brooks la llamó Arquitectura-Incluida (subsumption- architecture) Figura III.1.

Esta arquitectura permite múltiples objetivos, al tener capas trabajando de manera independiente y concurrente en objetivos individuales. La forma de mediar las acciones que se toman se basa en un mecanismo de supresión, en esta forma de acción se resalta la ventaja de no tener que realizar una decisión a priori del objetivo que se persigue.

No es necesario que las lecturas de los múltiples sensores sean vertidas en una representación central, solamente se sintetizarán las lecturas que cada capa necesite para lograr sus objetivos y se hará de forma independiente a cada capa.

La robustez en este caso dependerá del diseño de la granularidad de las abstracciones, esto es, si llegase a fallar la conducta de un nivel superior, las capas de niveles inferiores seguirán trabajando y en un momento dado le proporcionarían los datos necesarios para que pueda trabajar.

III. 3. 2 Arquitectura de Garcia-Alegre, Bustos & Guinea [GARM95]

Proponen una Arquitectura Jerárquica Multi-Agente para la generación incremental de conducta de complejidad creciente en el marco de la Robótica Autónoma. Dentro de este marco de referencia define los cuatro componentes básicos de la arquitectura: agente, agentes básicos, nivel, comunicaciones.

III. 3. 2. 1 Componentes básicos

Agente: Constituye la unidad fundamental de la arquitectura y abarca tanto aspectos reactivos como deliberativos sin ninguna restricción en su grado de complejidad. Actúa como una herramienta de encapsulación que posibilita una modularidad de tipo funcional; son tratados de forma indistinta en el sentido de un proceso computacional o de control dirigido a alcanzar o a mantener un determinado objetivo.

Agentes Básicos: Se definen como los bloques elementales de construcción que son necesarios para resolver una clase específica de problemas, por ejemplo movimiento libre de colisión del sistema o movimientos del ojo.

Nivel: Engloba a un conjunto de Agentes que comparten un lenguaje compuesto por un conjunto de términos relacionados con la percepción y la actuación. Gravita en torno a una representación específica del conocimiento del sistema \Leftrightarrow entorno y los Agentes que lo componen son modulados por los Agentes del nivel adyacente superior. Posee una memoria de corto o largo alcance que permite optimizar el comportamiento observable y compensar un exceso de actividad.

Comunicaciones: En esta jerarquía, los niveles se comunican entre sí, a través de canales bi \Leftrightarrow direccionales que no tienen asociada una semántica, por lo cual los protocolos de comunicación deben definirse entre los elementos comunicantes. Desde una perspectiva global de Percepción \Leftrightarrow Acción, existen dos flujos de información :

De arriba \Rightarrow abajo guiado por los objetivos que depende de los mecanismos de **coordinación** diseñados para utilizar las habilidades de los Agentes del Nivel inferior.

De abajo \Rightarrow arriba dirigido por los datos sensoriales, se encarga de, **la propagación** de la información del estado del robot-entorno.

La **coordinación** es un mecanismo fundamental que se realiza en tiempo real y depende a su vez de los objetivos y de la situación en que se encuentre el robot en el mundo. De esta forma los Agentes de un determinado Nivel se hayan coordinados por los del Nivel superior que les pide la solución de un determinado tipo de problemas en los cuales ellos son especialistas, a fin de alcanzar su objetivo. En la práctica no existe una garantía de que la petición se alcance con éxito, pero sí una respuesta ante el problema encontrado y una petición de ayuda o un mecanismo de corte que devuelva el control al Nivel superior.

La **propagación** hace referencia al flujo de información perceptual que progresa desde el Nivel inferior al superior. Un Agente propaga este tipo de información cuando se le solicita de forma explícita o bien cuando no puede realizar sus tareas dentro de las restricciones impuestas por un Agente de Nivel superior.

Se ha intentado mantener el marco general tan simple como fuese posible, bajo la hipótesis de que la complejidad debe crecer en paralelo para: el robot, el entorno y el conjunto de tareas que se quieren realizar de forma autónoma.

En la Figura III.2 referente a la arquitectura jerárquica multi-agente, se muestra esquemáticamente la arquitectura de locomoción guiada por visión ultrasónica, desarrollada para un robot que debe realizar un conjunto de tareas de navegación en un ambiente de oficina.

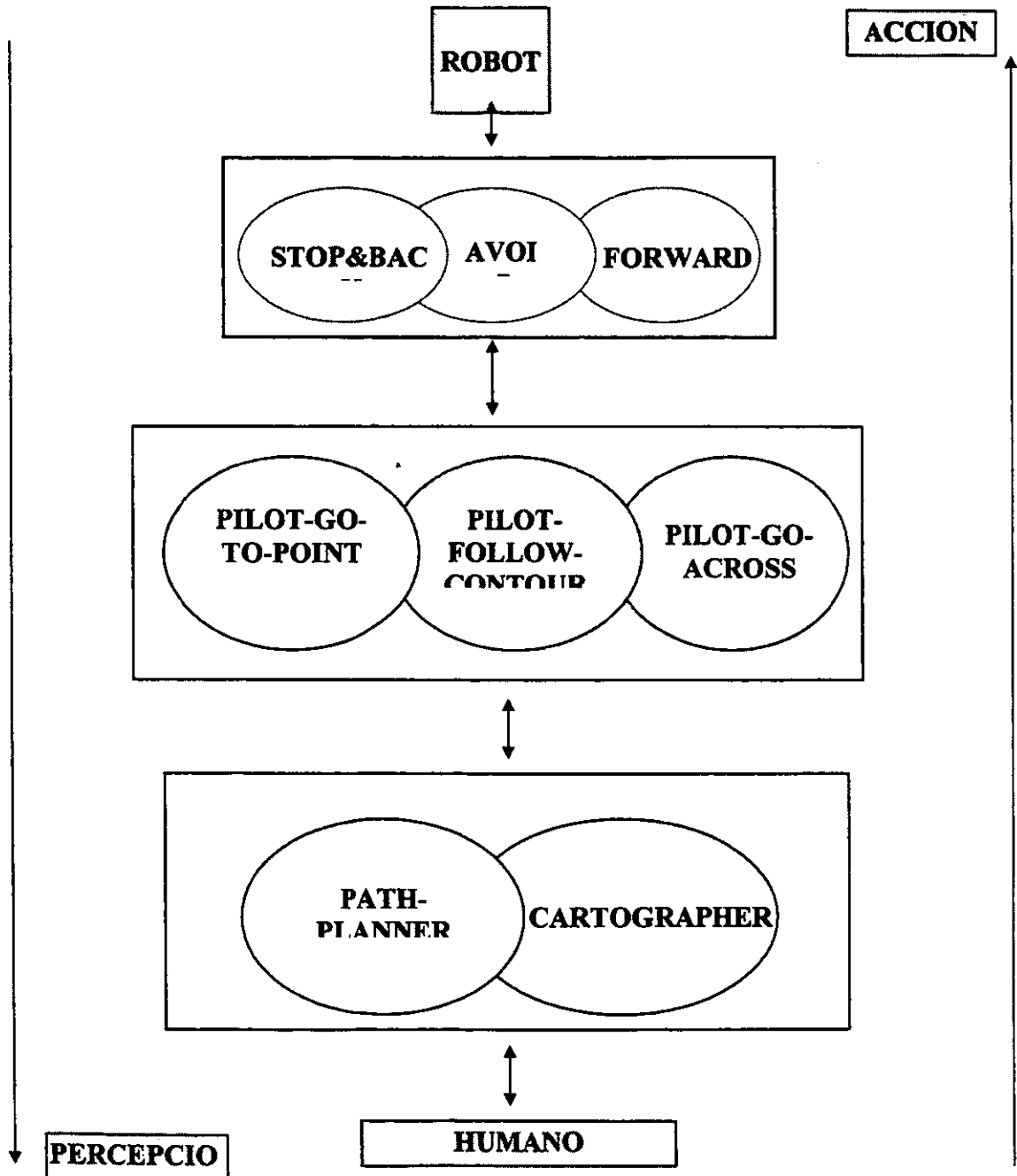
La arquitectura posee tres niveles asociados a una determinada representación del mundo, a la que le corresponde una granularidad espacio-temporal y un tiempo de respuesta de los agentes conductuales. Los niveles superiores van asociados a una actividad deliberatoria creciente frente a una capacidad de reacción decreciente. El grado de abstracción de los conceptos es superior conforme nos aproximamos al nivel de definición de misiones por el humano.

En el primer nivel se encuentran los agentes que han sido seleccionados como básicos para la realización del repertorio de tareas de navegación en oficinas. *Esta selección depende del criterio del diseñador en la selección de la granularidad conductual elemental.* En este nivel la percepción se basa en valores crudos o filtrados de distancias ultrasónicas, de intensidad luminosa en el infrarrojo y de activación de sensores de impacto, y la actuación se traduce en cambios en orientación o en velocidad del robot. Los agentes en este nivel poseen un tiempo de respuesta inmediato y una memoria de muy corto plazo, por lo que se pueden asociar a conductas reflejas.

III. 3. 2. 2 Agentes básicos de locomoción (arcos reflejos)

FORWARD: este Agente está guiado por los sensores de ultrasonidos. Tiene como objetivo la detección de una clase de estados, aquellos asociados a espacio-libre. Genera un movimiento del robot en la dirección y con la velocidad que le indica un Agente del nivel superior.

Figura III. 2 - Arquitectura Jerárquica MultiAgente



AVOID: tiene como base las sensaciones ultrasónicas y es activado por los Agentes del Nivel superior. Se dispara cuando percibe un obstáculo en un ángulo frontal amplio, generando un movimiento desviatorio cuando se sobrepasa una distancia de seguridad; mediante cambios en este parámetro se obtiene una variedad de trayectorias para evitar obstáculos asociados a un comportamiento de mayor o menor precaución.

STOP&BACK: está dirigido por sensores táctiles y sensores de infrarojo. Es necesario para hacer frente a los imprevistos originados por fallas en los sensores, obstáculos en movimiento rápido u objetos con perfiles afilados.

III. 3. 2. 3 Nivel de locomoción relativa a objeto (estímulo)

En este nivel se muestran tres ejemplos de navegación guiada por la percepción de un objeto en el dominio de trabajo; manejándose términos de percepción más elaborada como son: hueco, obstáculo, camino libre, etc. Su actuación consiste en el disparo coordinado de los agentes del nivel inferior.

Posee una granularidad mayor que el nivel anterior, un tiempo de respuesta ligeramente superior y una memoria a medio plazo que le permite un mayor grado de optimización de sus agentes.

PILOT_GO_TO_POINT: tiene como base las medidas propioceptivas de orientación y distancia a una localización en el mundo. Representa un comportamiento de locomoción segura entre dos posiciones relativas en el espacio 2D. Modula a los agentes básicos para conseguir su objetivo.

PILOT_FOLLOW_CONTOUR: se activa al detectar el estímulo_obstáculo. Trata de mantenerse alineado con el mismo a una determinada distancia, que finaliza mediante una condición de final o 'timeout'. Permite el encadenamiento de otras tareas como pueden ser la generación de mapas y el descubrimiento de estructuras conductuales claves, como es un hueco de salida.

PILOT_GO_ACROSS: se activa al detectar un estímulo clave, que en este caso es un hueco_puerta; realiza una coordinación especial de los agentes del nivel inferior a fin de atravesar con éxito el hueco.

III. 3. 2. 4. Nivel de navegación absoluta

PATH-PLANNER: constituye un comportamiento puramente deliberativo a partir de un mapa que ha sido generado por el cartógrafo. Maneja un mundo de estructuras poligonales a las que se intenta dotar de contenido semántico. Permite una optimización de las estrategias de pilotaje local guiadas por estímulos. Es modulado por el nivel Humano a través de un

heurístico o función de costo global; genera un camino tentativo óptimo, en base al conocimiento de que dispone, que ofrece al nivel adyacente inferior.

CARTOGRAPHER: es otro comportamiento deliberativo encargado de generar un mapa y actualizarlo con las nuevas percepciones provenientes del nivel inferior. El mapa generado, constituye una percepción compartida con los agentes de este nivel. Organiza la secuenciación de activaciones de los agentes del nivel adyacente inferior a fin de conseguir el objetivo indicado por el Nivel superior.

III. 3. 2. 5 Características principales de la arquitectura

- **Modularidad:** en el sentido de posibilitar su expansión en sistemas de proceso heterogéneos para evitar el agotamiento de recursos.
- **Flexibilidad:** para evitar restricciones “a priori” en relación al tamaño y complejidad de cada elemento software.
- **Desarrollo incremental:** que permita dinámicamente añadir y eliminar Agentes sin perturbar gravemente al sistema. Los Agentes de conducta más compleja se organizan a partir de la modulación de los Agentes Básicos.
- **Flujo bidireccional de información:** Top-down guiada por los objetivos o misiones y Bottom-Up guiada por el estado del Sistema-Entorno.
- **Las Conductas Básicas;** son mutuamente excluyentes por lo que no es necesario la inhibición explícita intra-Agentes (R.Brooks) o la mediación (L.Kaelbling) de las salidas de los distintos Agentes.
- **Se considera fundamental la selección y optimización de lo que se denomina Conductas Básicas o Agentes del Nivel más bajo.**

III. 3. 3 Arquitectura de Kaelbling

Kaelbling [KAEL87], propone el mismo tipo de arquitectura general propuesta por Brooks incluyendo algunos cambios que ahora mencionaremos. La propuesta de Kaelbling se basa en los siguientes problemas detectados en las arquitecturas anteriores:

El **primer problema es la robustez de la información** al encontrarse ésta distribuida en muchas capas; de aquí que ella sugiera aprovechar la redundancia de información proveniente de muchos sensores integrándola.

El **segundo problema** también proviene del manejo de la información y consiste en que las conductas mas sofisticadas (las de niveles superiores), tienden a depender más de las

condiciones del mundo que de las características particulares de las lecturas de un sensor. Para resolver este problema se propone un mecanismo de percepción general que sintetice la información de diferentes sensores en información acerca del mundo; esta información podría ser utilizada por muchas conductas. El objetivo principal de esta propuesta se centra en la capacidad de la implementación de conductas sofisticadas.

Cabe mencionar en este apartado, que actualmente existen distintos modelos de arquitecturas híbridas que apoyan la idea de compartir una representación del entorno por los agentes, García-Alegre & Recio [GARM97] propone que se integren deliberación y reacción siguiendo un modelo de agente jerarquizado. En cada nivel de esta jerarquía los distintos agentes que la componen compartirían: una representación del entorno, una historia y un módulo de aprendizaje. Sin embargo los agentes de cada nivel se encuentran diseñados como módulos de: percepción, toma de decisión y acción.



Donde, tanto percepción como acción pueden representarse por valores numéricos o expresiones lingüísticas ver Figura III.3.

En la propuesta básica de Kaelbling, la arquitectura primaria quedaría igual, seguirá existiendo la granularidad de las capas conductuales, pero, dentro de cada nivel de competencia existe una descomposición horizontal mayor que se puede apreciar en la Figura III.4.

Con esta mayor descomposición horizontal dentro de cada nivel de competencia, se pretende que cualquiera de los subcomponentes de acción pueda tomar ventaja de cualquiera de las salidas del componente de percepción.

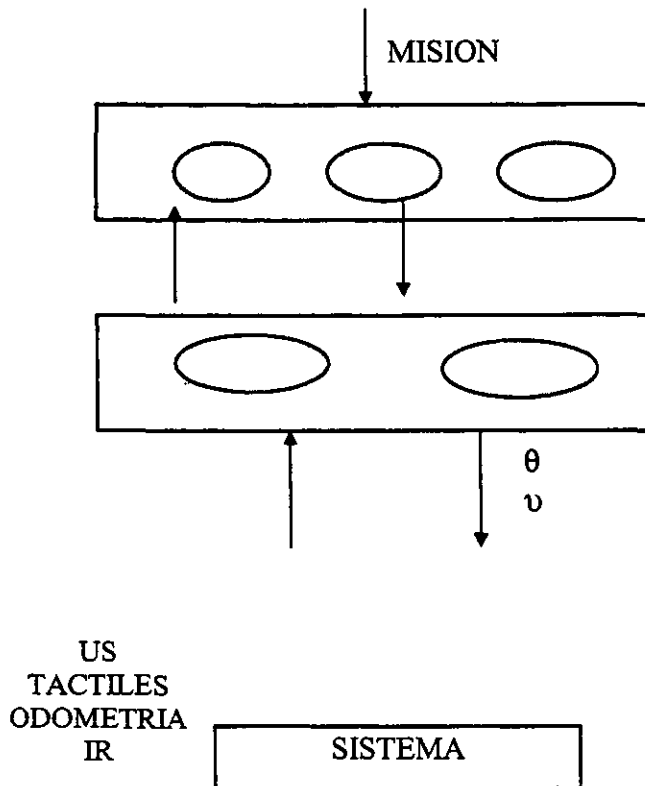
Kaelbling propone una mayor división, dónde cada conducta será descompuesta en capas de abstracción, con lecturas de sensor sin interpretación en los niveles mas bajos y modelos del mundo sofisticado en los niveles mas altos; en este caso el componente de acción consistirá en un conjunto de conductas.

Otra característica que añadió Kaelbling a esta arquitectura básica, es lo que llamó un marco para el control jerárquico adaptativo, que es vista en términos de la composición de diferentes conductas, para lo cual creó los procedimientos llamados mediadores.

Hay que hacer hincapié en que en este tipo de arquitecturas *el diseño por niveles de abstracción conductual es importante y es en donde se basa el diseño del sistema inteligente de control del robot*; ahora bien, debe existir una reciprocidad del diseño por niveles de abstracción, con el subcomponente de percepción. No obstante estos niveles de percepción, a

diferencia de los de Brooks, no tienen una correspondencia directa hacia los niveles de competencia del componente de acción (debido a los mediadores).

Figura III. 3 - Flujo y Tipos de Representación en Percepción y Acción



III. 3. 3. 1 Propiedades de un sistema reactivo propuestas por Kaelbling

Kaelbling propone un sistema reactivo que debe cumplir cuatro propiedades para garantizar que el sistema siempre se desarrolle de la mejor manera, de acuerdo a la información disponible.

Antes de enumerar estas propiedades es necesario comentar que existe una salida conductual no-se (no-command), cuyo significado es que un nivel de competencia no sabe qué hacer, estando también incluida en la arquitectura de Garcia-Alegre & Recio [GARM97].

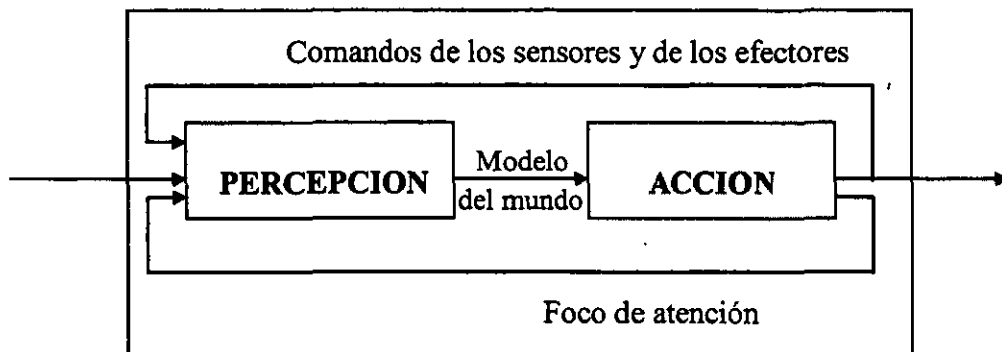
- En el nivel más bajo de competencia nunca existirá una salida no-sé.
- Una vez que se ha emitido una salida no-sé, ningún otro nivel la emitirá, a menos que sea una salida correcta.
- Los niveles bajos de competencia trabajan con información débil.

- Si existen dos niveles que emiten comandos en un mismo tic de reloj, el comando del nivel superior será el elegido para generar acciones en el entorno en curso. Esta condición se cumplirá si se parte del supuesto de que existe un mecanismo que sea capaz de emitir un juicio sobre dos comandos en base a la meta y a la situación en curso.

De esta forma se garantiza que el robot esté controlado por componentes de niveles inferiores, donde la información necesaria para trabajar es débil y de esta forma asegurar el trabajo congruente hasta que se recuperen los niveles de mayor competencia.

Dentro de cada uno de los niveles conductuales (competencia) la descomposición está basada en abstracciones, más que en competencias. La propuesta de Kaelbling queda representada en la Figura III.5.

Figura III. 4 - Descomposición Horizontal



En este caso la conducta del nivel más alto es construída en base a un mediador de conductas de nivel medio y éstas a su vez están construídas en base a un mediador de nivel bajo.

Hemos dado una descripción de forma breve de las principales características de la robótica reactiva, de donde podemos resumir las propiedades que debe presentar un sistema reactivo.

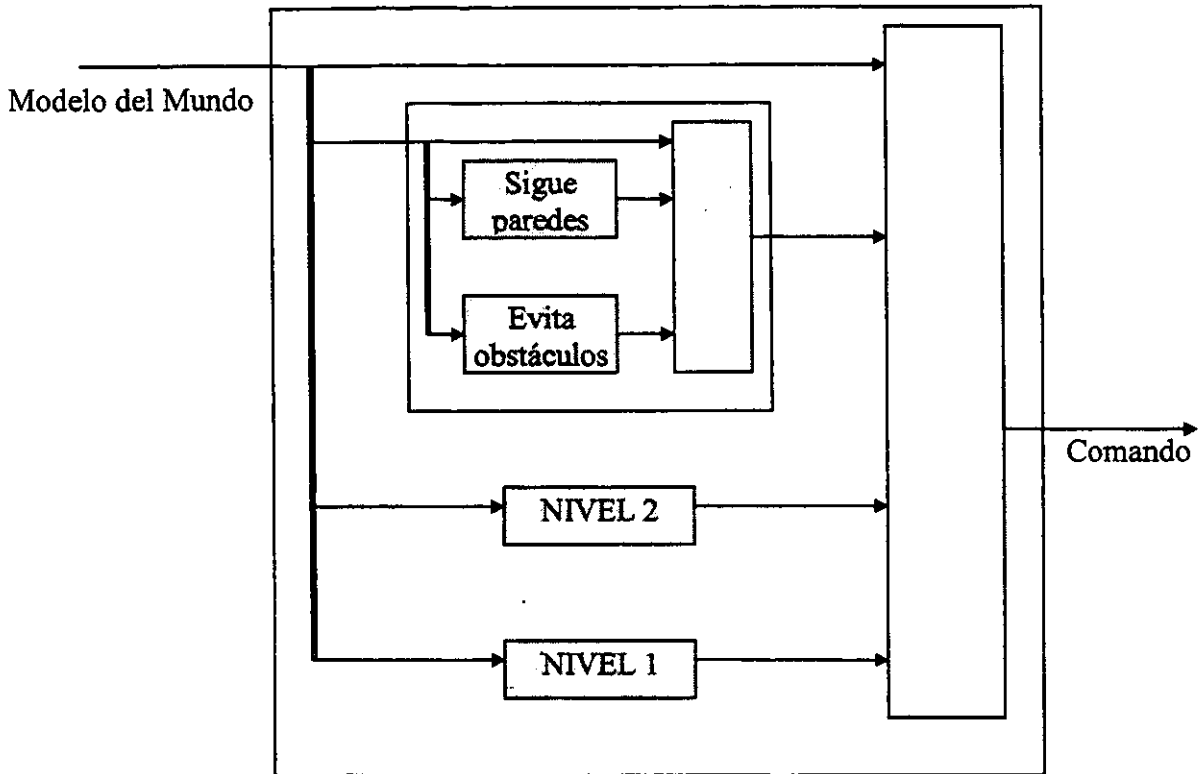
III. 3. 3. 2 Ventajas de las arquitecturas reactivas de acuerdo a Kaelbling

III. 3. 3. 2. 1 Modularidad

Es un concepto importado de la ingeniería de software (IS) tratado en Laureano y Ortíz [LAUA96](1) y en Laureano [LAUA93](1), que implica diseñar sistemas robustos, comprensibles y reusables, basándose en un diseño incremental sobre componentes pequeños fáciles de entender e implementar. Como consecuencia podemos decir que la correctez de

una conducta compleja puede ser probada de forma composicional desde el punto de vista computacional.

Figura III. 5 - Control Jerárquico Adaptativo



III. 3. 3. 2. 2 Conciencia

En ningún instante un sistema reactivo no debe darse cuenta de lo que sucede, debiendo además de ser capaz de reaccionar a la lectura de datos inesperados, lo que se logra con la granularidad conductual; si un nivel no está listo para contestar por diversas razones (esta calculando, acaba de ser prendido, está leyendo e interpretando datos), los niveles inferiores seguirán trabajando para mantener al robot, fuera de peligro, hasta que se recupere.

III. 3. 3. 2. 3 Robustez (conductual y perceptual)

El sistema debe comportarse coherentemente en situaciones nuevas aún cuando existan fallas en los sensores. La robustez conductual es lograda realizando las dos etapas anteriores, y la robustez perceptual se consigue, siguiendo la propuesta de Kaelbling, al integrar toda la información en una estructura que represente el conocimiento, o la carencia del conocimiento con respecto al mundo. También se puede lograr incrementando o diversificando los sensores. Así diremos que si un sensor falla y ésta es detectada, la información que logra el

robot del mundo es mas débil, lo que hará que el robot no pueda hacer tantas discriminaciones con respecto al mundo. No obstante la información integrada de los demás sensores hará que trabaje, aunque con un poco de degradación.

III. 4 Propiedades de un sistema altamente reactivo

III. 4. 1 Reactividad

La reactividad esta representada por un modelo basado en la conducta (agentes reactivos) de una determinada actividad, por el otro lado un modelo de manipulación simbólica (agentes cognitivos) utiliza la planificación, como es el caso de la inteligencia artificial tradicional. Esta propiedad nos hace reflexionar sobre el costo de la arquitectura global del sistema, durante el diseño con cualquiera de los dos modelos para llevar a cabo una determinada tarea.

En el caso del modelo simbólico, se cuenta con una representación interna explícita y exhaustiva del mundo, que en consecuencia necesita de una adecuación con el entorno, esta última actividad es considerada compleja, de aquí su alto costo.

En el caso de los agentes reactivos, éstos son simples y fáciles de comprender; no basan su funcionamiento en una representación interna del entorno, esto es, desarrollan acciones complejas con arquitecturas simples, de aquí su menor costo.

- Otra característica de estos agentes es que se consideran ubicados (situated), lo que significa que no toman en consideración eventos pasados y son incapaces de ver hacia el futuro; sus acciones están basadas en lo que sucede en un determinado instante y en la capacidad para obtener las características del mundo que le permiten actuar, por lo tanto, no planean lo que harán.

III. 4. 2 Trabajo de equipo

Dada la complejidad de los agentes construidos con el modelo simbólico, éstos con frecuencia son considerados autosuficientes, debido a que pueden trabajar solos o en su defecto con poca ayuda por parte de otros agentes involucrados en la misma tarea.

De forma contraria los agentes reactivos necesitan del trabajo de equipo para lograr la tarea encomendada, esto es, no pueden trabajar de forma aislada.

III. 4. 3 La robustez

La robustez y la tolerancia a fallas representan una de las principales propiedades de los sistemas desarrollados con agentes reactivos; un grupo de estos últimos puede completar la tarea aún en el supuesto caso de que algunos se hayan dañado, dicho en otras palabras, la pérdida de un agente no implica que no se termine la tarea global, debido a que los papeles que desempeña cada agente han sido asignados haciendo hincapié en sus necesidades y en la percepción del entorno para obtener los datos necesarios que permitan la ejecución de esos papeles; para lograrlo se tienen en cuenta dos propiedades modularidad y conciencia.

Por todo lo anterior, los agentes reactivos como subcampo de la IAD, están siendo considerados en otras áreas de IA; dado que se les considera flexibles y adaptativos, al poder manejar sus recursos en entornos impredecibles y completar sus tareas aún con fallas parciales del sistema [SOML93].

En la siguiente sección analizaremos en base a una correspondencia unívoca las partes constituyentes entre un robot y un sistema de enseñanza inteligente. Para descubrir que elementos en el SEI tienen una representación y cuáles deben ser sus características.

III. 5 Elementos que constituyen a los SEIs y a los robots reactivos

La analogía se realizará elemento a elemento, comentando las funciones en cada caso.

III. 5. 1 Master

El propósito de este elemento es la ignición del sistema. En el caso del SEI, estamos hablando del usuario que pretende utilizar el SEI, y que le dará vida al programa mediante su invocación. En el caso de los robots reactivos, estamos hablando del humano que le encarga una tarea al robot que debe ser completada.

III. 5. 2 Entorno

El entorno constituye el medio en el cual se desenvolverá el sistema y en consecuencia aquel que proporcionará las características necesarias para que el sistema reaccione.

En el caso de los SEIs solamente contamos con una representación del entorno, esto es, estamos registrando por decirlo de alguna manera sólo una dimensión, que además, para procesos cognitivos hasta ahora es la única conocida, y es a través de la tarea desarrollada. En otras palabras, aún no nos podemos asomar a la mente.

En el caso de los robots estamos hablando del mundo real y las distintas formas de registrarlo implica que se cuenta con distintas representaciones del entorno, de aquí que se mida: distancia, luz, temperatura, etc. Los distintos niveles conductuales están relacionados directamente con un tipo de representación, lo que conlleva comprender las distintas clases de representación del entorno [GARM97] y [MOLJ96].

III. 5. 3 Robustez conductual y perceptual

La robustez en estos sistemas va relacionada con la descentralización y la distribución. Si se desea mayor grado de robustez conductual, la respuesta es incrementar el número de agentes especialistas en la misma tarea y que éstos fueran capaces de darse cuenta cuando alguno de ellos falla. El caso idóneo sería el del humano suponiendo que trabajara con agentes; donde sin tener duplicación de agentes en el desarrollo de una misma tarea, cuando hay algún daño en alguna parte del cerebro, muchas veces hay posibilidad de que otros agentes se hagan cargo del trabajo aprendiéndolo nuevamente. Estas características están relacionadas con los conceptos de modularidad y conciencia, vistos en la sección III.3.3.2 y III.4.3.

En los SEIs no se da propiamente ninguna, meditemos la perceptual, no tenemos muchos componentes de detección extrayendo el mismo tipo de información y no tenemos una división física de los subtutores, de aquí que si falla uno, en realidad supondría que fallaría todo el PC y por lo tanto, todos los subtutores.

Sin embargo, es posible su implementación si tomamos en cuenta los siguientes aspectos: una división física en que los subtutores estuvieran ubicados en Transputer, esto sería un paso para lograr las dos robusteces, ya que podríamos tener a varios subtutores revisando los mismos errores.

La robustez conductual, la podríamos lograr, al tener a varios agentes trabajando un mismo aspecto del problema, con el mismo método o con métodos diferentes; además en este tipo de sistemas se podría con esta duplicación subsanar incertidumbres en la interpretación de los errores. Otra forma sería creando una granularidad más fina en la revisión de un mismo error, cuando éstos pertenecieran a la clase de graves.

En el caso de los robots se pretende que éstos se comporten coherentemente en situaciones nuevas y aún cuando existan fallas de sensores.

La robustez conductual en el caso de la arquitectura de Brooks y Garcia-Alegre se consigue con los flujos de información a través de los niveles conductuales la cual se debe a dos tipos de necesidad básicamente (sección III.3.1 y III.3.2):

- Dirección de objetivos (flujo de información top-down).

- Manejo de excepciones (flujo de información bottom-up).

La dirección de objetivos implica un mecanismo de arbitraje a través de la cual se tiene acceso al comportamiento de agentes de niveles básicos, proveniente de agentes de niveles superiores que le permiten de esta forma cumplir su objetivo.

En el caso del manejo de excepciones se refiere a la información que fluye de abajo hacia arriba, ya sea, porque se le haya pedido de forma explícita o frente a la incapacidad de resolver los problemas por conductas de niveles inferiores, en cuyo caso los niveles superiores lo resolverán integrando a su información parcial del mundo, la información proveniente de los niveles conductuales inferiores

En el caso de Brooks, queda aclarado, cuando habla de que el nivel primero es capaz de examinar datos provenientes del nivel cero y también le es permitido a este nivel proporcionar datos de manera interna a través de sus interfaces al nivel cero.

En el caso de la arquitectura de García-Alegre, esta información fluye entre niveles dependiendo de lo que se trate de resolver, de manera similar a Brooks.

En el caso de Kaelbling, es diferente ya que una de sus propuestas es la modificación para integrar la información de forma que cualquiera de los subcomponentes de acción pueda tomar ventaja de cualquiera de las salidas del componente de percepción, con lo que se integra toda la información en una estructura que represente el conocimiento del robot, o la carencia del conocimiento con respecto al mundo.

La robustez perceptual básicamente se debe a la gran cantidad de sensores que se tienen midiendo una misma representación de entorno y los mecanismos que detectan sus fallas como por ejemplo si un sensor da una lectura fuera de rango lo que se podría hacer es: no tomar en cuenta esa lectura de ese ciclo o ponerle la lectura del sensor que tiene a su lado; así diremos que si un sensor falla y se detecta su falla, la información que logra el robot del mundo es mas débil, lo que hará que el robot ahora no pueda hacer tantas discriminaciones con respecto al mundo, pero la información integrada de los demás sensores hará que trabaje, aunque con un poco de degradación.

III. 5. 4 Percepción e interpretación del entorno

Este punto esta estrechamente relacionado con la forma de implementar la representación del conocimiento en los sistemas reactivos; debido a que estos sistemas nacen como una nueva forma de ver las cosas dentro del análisis y diseño de los sistemas inteligentes.

Se ha mencionado la importancia de la caracterización de los aspectos del entorno, ya que serán ellos los que harán reaccionar al sistema. Partiendo de la definición de que los

asistentes expertos son agentes que monitorizan un proceso [HAYB97] se dirá que los agentes de un SEI, son asistentes expertos. En este caso las características del entorno que nos interesa monitorizar son los errores cometidos durante ese desarrollo; estos errores se repartirán entre los diferentes agentes encargados de vigilar aspectos diferentes de ese desarrollo, y en los cuales son expertos.

En el caso de los robots contamos con niveles conductuales, los cuáles, como ya se mencionó están relacionados directamente con un tipo de representación, por lo que de acuerdo a estos diferentes tipos de representación, contaremos con diferentes tipos de reacciones, así por ejemplo, se tendrán sensores detectando la distancia a objetos cercanos o lejanos, donde en caso de detectar algún objeto cercano, la conducta de evitar, actuará de acuerdo a las lecturas, caso contrario continuará su camino.

III. 5. 5 Reactividad

En el caso de ambos tipos de sistemas esta característica es lograda en base a los mecanismos que provocan la funcionalidad emergente (sección III.5.9). Esta última se basa en el diseño de ciclo de interacción entre el sistema (agentes) y el entorno; siendo necesaria una constante percepción del entorno.

III. 5. 6 Arbitros

Los mecanismos de administración de conflictos nacen como una necesidad al hecho de tener a varios agentes deseando intervenir, estos mecanismos obviamente dependerán de las distintas prioridades que se puedan manejar en el entorno donde se encuentra inmerso el sistema reactivo, de esta forma, evitar un obstáculo será prioritario a alcanzar un objetivo. En el caso de los asistentes expertos podemos pensar en que algunos agentes tendrán a su cargo la revisión de errores. Estos errores formarán parte de una jerarquía; donde la eliminación de algunos serán la base para continuar la enseñanza. Ya sea porque son graves o porque es el orden necesario para adquirir una habilidad.

III. 5. 6. 1 Mecanismos de arbitraje

En las tres arquitecturas vistas (sección III.3) se distinguen tres distintos tipos de arbitraje. En el caso de Brooks éste es realizado a través de inhibiciones conductuales. Brooks tiene ensamblado el comportamiento reactivo en el marco de la arquitectura incluida. En el caso García-Alegre et al. el arbitraje esta orientado a tareas [CAÑJ99]. Este tipo de arbitraje cuenta con niveles que tienen su propio ciclo de control, objetivos y percepciones. Cuando se producen colisiones de reacciones, el agente del nivel superior resolverá el problema, este arbitraje es orientado a una tarea específica, esto es, el agente del nivel superior conoce de

antemano las combinaciones que pueden presentarse y sabe a quien le tiene que dar el control de acuerdo a su objetivo. En el caso de Kaelbling éste se realiza a través de mediadores. Los mediadores producen las combinaciones deseadas de los agentes necesarios de acuerdo a un determinado estímulo.

III. 5. 6. 1. 1 Mecanismo de arbitraje propuesto para sistemas de enseñanza inteligentes

En el caso de los SEIs hemos dicho que sólo contamos con una conducta que es enseñar y esta se activa cuando el sistema se inicia. Esta conducta necesita de sus agentes básicos que en el caso específico del SEI son los subtutores; encargados de revisar el desarrollo del alumno en la habilidad de la cual son expertos. Cuando la conducta es activada todos los agentes se ponen en estado activo. Y sólo reaccionarán cuando se les presente un estímulo que pertenezca a su competencia. Sin embargo, aunque están revisando espacios mutuamente excluyentes (en el caso de los SEIs errores) podrían presentarse una colisión de reacciones; debido a que varios subtutores detecten errores a la vez. Lo que se propone en estos casos es establecer una jerarquía, que puede estar implementada en base a: 1) la importancia de errores, o 2) el orden de aprendizaje de diferentes subhabilidades. Esto es, identificar a los errores más importantes dentro del contexto global y en base a ellos implementar el mecanismo de arbitraje. O identificar el orden de utilización de diferentes subhabilidades por un experto para resolver un determinado problema y en consecuencia el orden de aprendizaje de dichas habilidades. En este último caso las jerarquías de errores estaría relacionada con el orden de aprendizaje de las subhabilidades.

En el caso específico de Makatsiná el arbitraje esta representado por una jerarquía que se estableció en base al orden de aprendizaje de diferentes subhabilidades. En el caso específico de la tarea cognitiva de Makatsiná debido a la forma en que es integrado el aprendizaje de la habilidad global los errores se encuentran ubicados en cada subtutor; por orden de importancia en orden ascendente. Esto es, los errores del subtutor -1- son más importantes que los del subtutor -2- y éstos a su vez más importantes que los del subtutor -3-. Cabe mencionar que en base al comportamiento reactivo que deseamos (diagnóstico de errores) es posible combinar diferentes tipos de mecanismos de arbitraje.

En el caso de los robots, de forma general la instanciación de una conducta genera el mismo comportamiento. Esto es, cuando se activa una conducta, ésta pone a trabajar a determinados agentes de niveles más bajos, que actuarán de acuerdo a los tres mecanismos de arbitraje mencionados.

III. 5. 6. 1. 2 Formas de activación de los agentes

En un robot los agentes se activan por: 1) tiempo, esto es, cada determinado intervalo de tiempo los agentes se despiertan y revisan el entorno para tratar de encontrar posibles estímulos que los hagan reaccionar, y 2) por una condición de alerta que esta de centinela todo el tiempo. Cada agente tiene asociado un límite (de acuerdo a sus funciones) el cual si es sobrepasado, implica que se debe despertar al agente.

Continuando con la analogía, en un SEI la activación de los agentes puede ser por: 1) ciclo de tiempo como en los robots, 2) activación por puntos críticos del desarrollo; generalmente asociados con puntos de decisión o cuando finaliza una etapa del problema y 3) errores directos, esto es, que no involucren pasos.

En el caso de Makatsiná se implemento la activación en base a puntos críticos ubicados por el experto y por errores directos.

III. 5. 7 Flexibilidad

Esta característica se refiere a la capacidad de poder adaptares a las diferentes características del entorno.

En el caso específico de los SEIs, estamos hablando de la capacidad en tiempo de ejecución de adaptarse al alumno, pero no sólo eso sino a la diferencia substancial que existe cuando dos personas cometen el mismo error, pero por razones distintas. Esto se logra al tener a distintos asistentes expertos trabajando subhabilidades, con lo cual abarcan un mayor nivel de detalle en los errores.

En los robots, esta flexibilidad se logra al tener diferentes niveles conductuales razonando sobre diferentes tipos de representación y a la implementación de los flujos de informaión en base a prioridades de comportamiento (existentes en los niveles).

III. 5. 8 Modularidad

La modularidad se da en cualquier sistema diseñado con agentes reactivos (sección III.3.3.2.1).

Hasta el momento no existe una metodología que nos permita llevar a cabo la formalización de agentes para: 1) poder entender y seguir su comportamiento y 2) dar un seguimiento a los errores conductuales. En este trabajo se hace una propuesta de formalización de agentes, que permitirá formalizar la arquitectura MultiAgente y las operaciones asociadas (Capítulo V).

III. 5. 9 Ubicación y funcionalidad emergente

Tanto en los robots como en los SEIs, esta característica se logra al estar percibiendo constantemente a través de los diferentes mecanismos de medición el entorno. (sección III.2.1.1).

En el caso de un SEI es percibiendo los errores del desarrollo del usuario, con lo que se logra en todo momento la reacción del SEI. En el caso de los robots, a través de los sensores se perciben los cambios del entorno físico y se logra la reacción en base a estos cambios y al procesamiento de esta información en los distintos niveles de representación (sección III.2.1.4).

III. 5. 10 Los niveles y la granularidad conductual

Este concepto está relacionado con los mecanismos de arbitraje (sección III. 5. 6). La granularidad conductual en los robots tiene dos propósitos. El primero está relacionado con la capacidad de ensamblar conductas complejas en base a conductas simples. El segundo está relacionado con la capacidad de arbitraje de esas conductas.

Analicemos un ejemplo: supongamos que un robot se encuentra desplazándose hacia un punto y de pronto se le atraviesa un objeto. El robot no sabe que es, sólo sabe que es lo que debe hacer ante una situación así; en este caso será esquivarlo o si está muy próximo parar. Su prioridad es no hacerse daño antes que alcanzar el objetivo. Una vez evitada la colisión continuará su trayectoria para alcanzar su objetivo final.

Se deduce de los párrafos anteriores que la granularidad conductual está íntimamente ligada a como se desea reaccione el robot ante determinadas situaciones del entorno (sección III.3.1 y sección III.6.3). Y es a través de esta jerarquía que logra el ensamblado de conductas complejas y prioridades.

En el caso de los SEIs, no es posible visualizar esta granularidad a través del ensamblado de conductas complejas en base a conductas simples, debido a que sólo contamos con una conducta que es enseñar. Sin embargo, la capacidad de reacción, va directamente relacionada al comportamiento deseado del sistema ante ciertos eventos del entorno. Específicamente en un SEI se desea que reaccione ante los errores; donde éstos pueden ser clasificados por niveles de importancia (de acuerdo a taxonomías ad-hoc a la resolución de problemas relacionados con el dominio de enseñanza o basados en la forma en que se desea integrar la enseñanza de distintas habilidades, sección III.5.6.1.1). El diagnóstico de estos errores daría como resultado un árbol. En base a este árbol se crearía la jerarquía de intervención de los distintos agentes.

Analicemos un ejemplo: supongamos un estudiante que aprende a multiplicar. Siempre que aparece un 9 comete errores. La causa del errores puede ser variada: no se sabe la tabla del 9, no sabe multiplicar, o a lo mejor coincide que cuando ha aparecido el 9, el multiplicador es de dos cifras y no sabe hacer el desplazamiento necesario. Aquí la pregunta obligada sería, qué es más grave no saber la tabla del 9 o no saber multiplicar. En base a ello entrarían en acción los diferentes agentes; cuya representación en un SEI, sería la de subtutores. Esto es, habría inhibición de la actuación de alguno de ellos; lograda por cualquiera de los mecanismos de arbitraje citados (sección III. 5. 6.1.1).

III. 6 Conclusiones con respecto a los SEIs y los agentes reactivos

III. 6. 1 ¿Dónde se implementará la reactividad?

En un SEI, quien lleva la batuta de qué y cómo se va a enseñar es el módulo tutorial y lo logra interactuando con los demás módulos del SEI, pero por ahora olvidemos a los demás módulos y centrémonos en éste.

Para lograr la funcionalidad emergente de los agentes es imprescindible la interacción intensiva sistema-entorno; en un robot quienes se dan cuenta de los cambios del entorno son básicamente los sensores, que integran estas lecturas en los diferentes agentes.

En un SEI, lo tenemos un poco más difícil, porque el estado cognitivo del usuario que representa nuestro entorno, no se ve. La forma en que podemos percibir el estado cognitivo del usuario es observando su desarrollo en la tarea que se desea enseñar; objetivo que logramos con la ayuda de la interfaz que representa el puente entre el sistema y el usuario; aquí es donde el sistema podrá asomarse para ver ese estado del entorno. Por otro lado, se encuentra el tratamiento que le demos al análisis y diseño del proceso de enseñanza - aprendizaje del dominio que se desee enseñar. En Laureano y De Arriaga [LAUA96](3) y Laureano [LAUA98](4), se desarrollan ejemplos que hacen hincapié en este análisis.

III. 6. 2 Distintas formas de diseño de robots no reactivos

Hasta antes del nacimiento de la filosofía reactiva las conductas de los robots eran diseñadas haciendo énfasis en los módulos y sus funciones.

La modular-funcional; donde existe una fuerte correspondencia entre los componentes estructurales del sistema y su descomposición funcional en otras palabras, a cada elemento del sistema le corresponde una función determinada y ésta se encuentra encapsulada en una

función, objeto, paquete, módulo, agente, etc. Este tipo de arquitectura puede haber sido encontrado de forma top-down o bottom-up.

Valavanis y Saridis [VALK92] proponen un diseño analítico para construir robots inteligentes, basándose en la dualidad de la función de entropía; como una medida de la incertidumbre y como función de energía. El modelo del sistema inteligente (robot) se basa en el principio de incrementar la precisión, decrementando la inteligencia para lo cual propone una arquitectura de diseño de tres niveles interactivos jerárquicos que corresponden a: la organización, la coordinación y la ejecución de tareas. Se presenta como una analogía a las letras y las palabras donde de forma *bottom-up* se genera un lenguaje en base a 27 letras, estas son seguidas por la formulación de planes y la organización; los planes son formulados en base a un conjunto de eventos/acciones primitivas y será verificada su compatibilidad y completos.

Vlavanis et al. proponen apoyarse en modelos matemáticos para realizar la descripción de los 3 niveles jerárquicos (coordinación, control de hardware y ejecución de procesos) y de esta forma garantizar una operación óptima del sistema, donde el control inteligente es postulado como: el problema matemático de encontrar: la secuencia correcta de decisiones internas y controles para un sistema estructurado. De acuerdo al principio de incremento de precisión en base al decremento de inteligencia, de forma que minimize la entropía total.

III. 6. 3 Diseño de la conducta básica del agente reactivo

El diseño de la granularidad conductual en los sistemas reactivos esta íntimamente ligado al diseño de la conducta de los agentes reactivos. En general se hace en base a la experticia del diseñador, pero, es de vital importancia (sección III.3.2.5 y pag. 13).

Atkinson-Abutridy y Carrasco [ATKJ99] destacan 3 enfoques básicos utilizados para diseñar la conducta de los agentes reactivos: 1) el diseño guiado etológicamente, 2) el diseño por experimentos y 3) el diseño guiado por actividad situada.

El diseño *guiado etológicamente* permite diseñar el agente tomando en cuenta los estudios fundamentales de la conducta animal; moldeándolos, experimentándolos y evaluándolos de una forma repetitiva, hasta obtener un comportamiento adecuado. Se puede profundizar este diseño en: [BERJ90], [BROR91](1), [GARA93].

El diseño *por experimentos* se lleva a cabo de forma 'bottom-up' dotando al sistema de conductas básicas; experimentándolas, evaluándolas e incorporando conductas adicionales a medida que vayan siendo requeridas en función de el comportamiento final, [GARM95].

El diseño *guiado por actividad situada* se enfoca sobre las acciones del agente, y por lo tanto, sus conductas según las situaciones con las que se encuentre en un determinado

momento y entorno. Por lo tanto es fundamental encontrar las percepciones específicas que provocarán una determinada acción sobre un entorno preestablecido. Se puede profundizar en este tipo de análisis y diseño en: [LAUA96](3), [LAUA97](1), [LAUA99] y [LAUA00](1).

Independientemente del método de diseño global utilizado, el control del agente dependerá de dos aspectos claves: 1) la representación y codificación de las conductas, y 2) la coordinación entre ellas.

Makatsiná fue diseñado por actividad situada. En este caso es necesario conocer de la manera más completa la *conducta enseñar* de un determinado dominio. Lo anterior implica conocer: 1) las subhabilidades que integran una de mayor complejidad, 2) las relaciones entre ellas, 3) el proceso de enseñanza y 4) los errores. Para lograrlo proponemos como método de instrumentación de análisis y diseño, al análisis cognitivo de tareas (Capítulo IV). En Laureano y Arriaga [LAUA96](3) se presenta un método para obtener las lecturas de los agentes y se presenta el desarrollo de un ejemplo. Posteriormente se propone la formalización del comportamiento de estos agentes a través de un lenguaje simple propuesto por Luck y d'Inverno [LUCM95]; basado en el lenguaje Z (Capítulo V).

III. 6. 4 Análisis cognitivo de tareas

El ACT tiene un papel primordial ya que a partir de este análisis se obtendrán las características del entorno que serán registradas por los distintos agentes que configuren el sistema, punto muy importante para lograr la funcionalidad emergente.

De acuerdo a Redding [REDR89] el análisis cognitivo de tareas compara estructuras y procesos de expertos con aquellos que tienen menos experiencia en el desarrollo de una determinada tarea cognitiva (TC) para determinar cómo puede ser aprendida, cómo puede ser enseñada, y la organización de los modelos mentales óptimos, así como las habilidades para su desarrollo. Para lograrlo se utilizan una variedad de técnicas entre las que podemos citar: la observación, análisis de protocolo, modelado del conocimiento utilizando redes semánticas o gráfica genética y análisis de errores. El análisis cognitivo de tareas ha sido utilizado con éxito en el desarrollo de SEI, [GOTS89], [FLEB96], [LAUA97](1) y [LAUA98](1); así como, en sistemas de enseñanza [RYDJ93].

III. 6. 5 El Control de los sistemas reactivos

Una de las conclusiones más importantes de este capítulo es la que trata sobre el control. Aún cuando hablemos de comportamiento emergente y de carencia de representaciones simbólicas y de objetivos globales.

El control se encuentra en el análisis y el diseño, de aquí que éste sea tan importante. Se mencionó que el modelado interno del sistema se realiza en base a como se desea se comporte el sistema. Esto es, debemos tener una clara y completa idea de que es lo que deseamos realice el sistema. De otra forma sería imposible fijar objetivos parciales de acuerdo a sus tareas; a cada uno de los agentes.

Lo anterior se traduce en que los diseñadores de sistemas con agentes reactivos deben invertir mucho tiempo en el diseño para dejar precomplejado o fijo en hard-wire el mecanismo de acción-selección. Y así se concluye que mientras los mecanismos de planificación (inteligencia artificial tradicional) dejan mucho trabajo al agente deliberativo. La aproximación reactiva deja mucho trabajo deliberativo al diseñador.

En el siguiente capítulo se estudiará cómo fue tratado el dominio, para lograr extraer estas características, así como conocer el número de agentes necesarios para desarrollar la conducta enseñar.

Capítulo IV

Análisis y diseño del dominio

IV. 1 Introducción

IV. 1. 1 Problemas básicos

Dentro de los problemas que se resolvieron durante la etapa de análisis y diseño del sistema de enseñanza inteligente (SEI) están los siguientes:

- Cómo determinar la interpretación del entorno dinámico.
- Cómo encontrar las particularidades que, de forma distribuida, serán interpretadas por cada agente.

A lo largo de este capítulo* se verá en detalle la resolución de estos dos puntos. Es necesario tener en cuenta desde la etapa de análisis y diseño, qué tipo de mecanismo de inferencia tendrá nuestro sistema inteligente. En el caso específico de nuestro SEI este mecanismo estará implementado dentro de un sistema MultiAgente; por lo tanto será necesario proveerlo con la capacidad de generar la representación del entorno dentro de cada agente. La representación del entorno estará en función de los valores que obtenga de los

* Los contenidos presentados en éste capítulo amplían las ideas aportadas en las publicaciones referenciadas como: [LAUA96] (3), [LAUA97] (1), [LAUA98] (4), [LAUA99] (1), [LAUA00] (1)

perceptores. Valores que en nuestro caso estarán determinados por los errores que cometa el estudiante en su tarea.

IV. 1. 2 Definiciones básicas

Antes de comenzar el tema principal de este capítulo mencionaremos unas definiciones concernientes a los sistemas de enseñanza inteligentes (SEIs).

Cuando hablamos de un SEI, hablamos de un tipo de sistema que debe contar al menos con las siguientes características.

- El dominio debe ser suficientemente comprendido por el sistema, para conocer la solución de los problemas.
- El módulo encargado del proceso tutorial debe ser capaz de darse cuenta del nivel de conocimiento que tiene el usuario.
- La táctica instruccional se debe seleccionar de forma inteligente para que, en cada caso se elija aquella que reduzca la diferencia entre el usuario y el experto en el dominio.

En un SEI las técnicas de inteligencia artificial pueden ir en dos lugares: 1) el conocimiento que el sistema tiene del dominio y 2) los principios del proceso tutorial y los métodos por los que son aplicados estos principios.

En Makatsiná las técnicas de inteligencia artificial se han integrado básicamente en los principios del proceso tutorial; el cual es reactivo. El conocimiento del dominio esta integrado en un experto articulado, debido a las características de una interacción reactiva.

IV. 2 Antecedentes

El principal problema durante la construcción de un SEI, esta relacionado con las limitaciones del modelado del entorno, por tratarse de un proceso cognitivo. Nuestra propuesta de análisis y diseño fundamentada en la reactividad, permitirá modelar de forma distribuida ese entorno.

Para lograrlo se utilizan como fuente de conocimiento herramientas originadas dentro de distintas áreas de investigación, pertenecientes a los tres campos que emplean los SEIs. Abordaremos en primer lugar a la psicología cognitiva, donde una de las preocupaciones es esclarecer el desarrollo de las tareas cognitivas (TC) como: pensar, aprender, enseñar, resolver problemas, además de poder seguir un proceso de monitorización. En segundo lugar se analizará la rama de inteligencia artificial, donde se encuentran los distintos modelos de representación del conocimiento; en este caso se abordará la gráfica genética (GG), de mucha

ayuda en la organización del conocimiento conceptual y procedimental. Finalmente, los recursos educacionales que permitirán considerar aspectos pedagógicos, en la construcción del sistema.

IV. 2. 1 Análisis cognitivo de tareas (ACT)

De acuerdo a Castañeda [CASS93] un ACT está representado por una evaluación basada en descripciones claras del conocimiento semántico, procedimental y estratégico del experto o del estudiante; en nuestro caso como se trata de la modelización de la experticia que tiene un sujeto en el desarrollo de cierta habilidad, nos referimos al experto. El ACT es un análisis recursivo de tareas, que se realiza con el fin de esclarecer el proceso psicológico involucrado en la construcción cognitiva del desarrollo de las habilidades; de forma recursiva se divide la tarea en subtarear cada vez más específicas, para así conocer con mayor precisión los elementos que la componen. A estas subtarear se les agregan los procesos mentales que subyacen a cada una. Los procesos mentales están constituidos por estructuras que interactúan, y las estrategias involucradas para lograr cada paso de la tarea. En Castañeda [CASS93], Redding [REDR89] y Redding [REDR92], se da una explicación más profunda del ACT.

Al producto del ACT se le conoce como modelo cognitivo; donde se enfatiza la cognición, se analiza la pericia, se evalúa el conocimiento considerando su interrelación con todo el desarrollo del trabajo en cuestión, y se consideran los modelos mentales.

Durante el desarrollo de este proyecto se consideró de gran importancia establecer una diferencia entre estrategia y táctica. Por estrategia nos referimos a la toma de decisiones en la resolución de problemas a largo plazo. Por táctica nos referiremos a la toma de decisiones en la resolución de problemas a corto plazo, estas últimas se encuentran subordinadas a la estrategia.

De acuerdo a Ryder & Redding [RYDJ93], es recomendable un ACT en tareas que:

1. Involucran un alto grado de complejidad del problema y toma de decisiones.
2. Requieren una carga de trabajo fuerte o mucha atención por parte del individuo.
3. Subyace el desarrollo de habilidades, por ejemplo, que requieren mucha práctica para lograr la pericia.
4. Requieren grandes cantidades de información para ser asimilada durante la instrucción.
5. Los expertos consideran difíciles durante la verbalización o demostración a través de otras acciones, para inferir el proceso cognitivo.

6. Conllevan una considerable cantidad de dudas individuales de acuerdo al número de estrategias cognitivas disponibles.

El dominio de Makatsiná se encuentra ubicado en el punto número 3.

El ACT, contempla puntos de interés en el desarrollo de los SEIs como son:

- Un ACT resalta la naturaleza compleja de la interacción entre procesos cognitivos, estructuras y estrategias que constituyen una tarea determinada, y por extensión, la enseñanza de esa tarea.
- Del punto anterior se desprende la necesidad de identificar los conceptos y/o habilidades subyacentes a la ejecución de una tarea y encontrar caminos efectivos para su enseñanza.
- Permite identificar algún concepto subyacente, que sea suficiente para llegar al éxito en la enseñanza. Ya que cada tarea requiere de varias habilidades que deben ser combinadas, la ausencia de cualquiera de ellas conduce a la incapacidad del aprendizaje de esa TC.
- No existen reglas para construir situaciones que conduzcan al aprendizaje mágico de una TC.
- Aún con un ACT bien elaborado y apropiado, la capacidad de identificación de componentes para crear tácticas de enseñanza, lleva al descubrimiento de la utilización de enseñanza estratégica para resolver esa TC.
- Se cree que las diferencias de los estados de competencia entre los novatos y los expertos están representadas por enlaces creados en la realización de una tarea; donde éstas han sido aprendidas con el tiempo y reafirmadas con la práctica. Una forma sugerida para encontrarlas es jerarquizar por niveles de dificultad, la realización de esa tarea.
- Es necesario considerar los cambios de los diferentes estados en el desarrollo intelectual de las personas durante el aprendizaje de una tarea, así como la historia de los cambios de esos estados, y con que se cuenta, (habilidades y conocimiento previo) en el instante de comenzar el aprendizaje.

Existe una corriente de investigación que ha estado tratando de formalizar, desde el punto de vista matemático, los análisis de tareas; tal es el caso del trabajo realizado por Lee [LEEJ97], donde se propone un método para formalizar y verificar las tareas de los sistemas basados en el conocimiento; su trabajo tiene dos componentes:

1. Un modelo de especificación que describe las propiedades estáticas del sistema.
2. Un proceso de especificación que caracteriza las propiedades dinámicas del sistema.

El modelo de especificación cuenta con:

- Un modelo de los objetos del dominio.
- Otro referente a los estados de solución del problema.

Las propiedades dinámicas del sistema están caracterizadas por:

El uso de la transición de estados. Se describe explícitamente la funcionalidad de una tarea.

- La especificación de secuencias de tareas y las interacciones entre ellas.
- Para lograrlo utiliza las expresiones de estados de tareas creadas por Lee.

Estos trabajos desde el punto de vista de la inteligencia artificial, añaden nuevos atractivos a este tipo de análisis importados de la psicología cognitiva. A continuación se presenta la forma de obtener el ACT de acuerdo a tres autores: Castañeda [CASS93] y Ryder & Redding [RYDJ93].

IV. 2. 1. 1 Obtención del ACT de acuerdo a Castañeda

Aceptada la importancia del ACT, se dará una breve explicación de los puntos básicos para su obtención. En esta tabla se encuentran los datos del dominio de Makatsiná. (ver la Tabla IV.1).

La columnas de la tabla serán:

- Pasos del Desarrollo.
- Contenido de los pasos.
- Formas de Evaluación.
- Tipo de Representación.
- Complejidad de los Procesos que subyacen a la evaluación.

Y la información de estas columnas se analizará en cada una de las etapas del desarrollo de la tarea cognitiva en cuestión.

IV. 2. 1. 1. 1 Pasos del desarrollo

Se tiene que empezar por conocer los pasos requeridos para el desarrollo de la tarea. Este procedimiento permitirá descubrir los diferentes conjuntos de conocimiento que interactuarán para darle vida al módulo TUTOR, para lograrlo se utiliza el modelo mental [WILM96] con el que trabaja el experto durante la solución del problema. Basándose en este modelo se extraerán los pasos del procedimiento (ver la Figura IV.1).

IV. 2. 1. 1. 2 Contenido de los pasos

En esta columna se indica el tipo de conocimiento que va a contener cada uno de los pasos y que se clasifica en: factual, conceptual, procedimental, estratégico y táctico. Esta columna juega un papel importante dado que su obtención se puede centrar en conocimiento declarativo o procedimental, según sea el caso, un tutor de conocimiento o de habilidades. Esta columna ayudará a encontrar la mejor representación del conocimiento dentro del sistema.

IV. 2. 1. 1. 3 Formas de evaluación

Poder conocer cómo se va a evaluar una determinada actividad dentro del proceso tutorial, conduce a establecer la competencia de los diferentes niveles de abstracción. Esta parte está representada por la comunicación entre los diferentes conjuntos de conocimiento, en caso de que exista; si no existe la evaluación, será diferente.

IV. 2. 1. 1. 4 Tipo de representación

Su función es tratar de organizar el conocimiento del problema. Este conocimiento posteriormente será detallado en la columna de contenido de los pasos, aquí vale la pena aclarar que la psicología cognitiva cuenta con su propia definición. Las estructuras son consideradas como componentes estáticos del sistema (entidades funcionales) y los procesos son componentes dinámicos que se refieren a la actividad del sistema cognitivo (pensar, solucionar problemas, etc.).

Lo anterior nos auxilia durante la construcción del tipo de planes instruccionales y sus tácticas instruccionales (acciones pedagógicas) en las que se verá involucrado el alumno, para aprender un determinado conocimiento o habilidad.

IV. 2. 1. 1. 5 Complejidad de los procesos que subyacen a la ejecución

En esta columna se clasificará la complejidad en: **discriminación (simple y múltiple)**, subproceso del aprendizaje que permite diferenciar entre clases de cosas. **Generalización**, subproceso que permite establecer con precisión el rango en el que lo diferenciado es aplicable. **Categorización conceptual (vertical, horizontal)**, subproceso gracias al cual establecemos clases entre las cosas que nos rodean y somos capaces de organizar categóricamente nuestro mundo. **Integración del conocimiento en micro y macro estructuras**, proceso constructivo gracias al cual continuamente estructuramos nuestro conocimiento, con la finalidad de integrar conocimientos provenientes del material estudiado con la representación personal, derivada de la experiencia particular. **Solución de problemas (baja y alta complejidad)**, procesos gracias a los cuales se operan los conocimientos y

habilidades de una manera original, para solucionar situaciones novedosas, o no conocidas por el alumno.

Lo anterior ayuda a encontrar los elementos esenciales para poder estructurar las didácticas instruccionales. Estas didácticas permiten al alumno adquirir el conocimiento o la habilidad necesaria para el desarrollo de una determinada tarea.

IV. 2. 1. 2 El ACT de acuerdo a Ryder & Redding (un complemento)

El ACT se complementa con el análisis recursivo de la TC en cuestión; este análisis recursivo presenta tres estados: el estado de análisis inicial (orientación) dedicado al entendimiento del trabajo de forma global y de los componentes de experticia que componen este trabajo, así como los componentes de la experticia de estados subsecuentes. El estado intermedio (análisis básico) dedicado al análisis de desarrollo del contenido de los componentes. El estado final (cómo adquirir las habilidades), está dedicado al análisis progresivo del cambio de novatos a expertos [ELIR90]. Los tres análisis se llevan al nivel de profundidad deseado o el necesario para soportar el diseño de la enseñanza del dominio.

El análisis progresivo de estos estados se realizará en base a tres componentes que representan la experticia (conocimiento, habilidades y modelos mentales). Este análisis arrojará nuevos datos de importancia, así como restricciones. Uno de los datos es conocer la existencia o carencia de un flujo de datos entre tareas, en el caso de multitareas. En los siguientes párrafos se explicará cada componente.

IV. 2. 1. 2. 1 Conocimiento conceptual

Este concepto ya había sido tratado en el ACT por Castañeda [CASS93] y está representado por la columna de contenido de los pasos; aquí ahondaremos en su definición. Este componente incluye además del tipo de conocimiento del dominio, sus interrelaciones, así como, las reglas y procedimientos que complementen el trabajo de forma verbal.

IV. 2. 1. 2. 2 Habilidades

Este componente incluye todos los tipos de conocimiento procedimental, así como las habilidades que se requieren para el trabajo a ser enseñado. También incluye el tipo de aprendizaje asociado y las estrategias de desarrollo. Este componente es importante en el sentido de que será el que de pie a iteraciones; que implican una mayor descomposición de las habilidades de forma jerárquica. Así la habilidad estratégica contiene a la táctica y las procedimentales pueden descomponerse en otras más específicas de acuerdo al paso a ser iterado.

En el ACT, propuesto por Ryder et al. [RYDJ93] se hace hincapié en realizar este análisis considerando al trabajo como un todo, donde estas habilidades pueden o no corresponder con tareas individuales, o ser parte de una sola tarea.

El componente de habilidades es analizado, descomponiendo las tareas involucradas en una secuencia de pasos. En caso en que el procedimiento sea cognitivo, se centra el procedimiento en los procesos mentales.

Dado que Makatsiná, es un tutor de habilidades, este complemento resulta de vital importancia. A continuación se menciona la taxonomía de habilidades propuesta por Ryder et al. [RYDJ93]:

- Reconocimiento de patrones: se refiere a la identificación y clasificación de información que proviene de los sentidos (ejemplo la visión).
- Estratégico: se refiere a las decisiones tomadas en la resolución de problemas. Incluye procesamiento central y trata primeramente con los datos verbales y/o estímulos no predecibles.
- Motor Grueso: se refiere a los movimientos musculares, donde los requerimientos de decisión son mínimos. Se ejecutan en respuesta a una situación relativa a un estímulo estático (sentidos musculares y excitación del oído interno: kinesthetic).
- Motor Perceptual: se refiere a movimientos continuos, o donde el control de los movimientos depende de entradas perceptuales dinámicas. Incluye componentes perceptuales y motores.
- Procedimentales: se refieren a la secuencia de acciones motoras o cognitivas en situaciones predecible. Tienen baja demanda cognitiva, en el caso de acciones motoras.
- Interactivas: incluyen habilidades inter-personales como en el caso de comunicación, persuasión y supervisión.
- Habilidades integradas y de tiempo compartido: se refiere a la integración de varias habilidades en la realización de una sola tarea y a la atención que se produce cuando se cambia de tarea en un procedimiento multi-tarea. Esta basada en estudios de condiciones de transferencia de tarea sencilla a tarea doble. Este tipo de estudio es importante en trabajos con alta carga de condiciones y existe la necesidad de coordinar tareas y habilidades.

Como ya se había mencionado, debido a la diferencia entre estrategia y táctica se propone otro tipo de habilidad.

- Tácticas: se ubican en un nivel subordinado al estratégico y representan la toma de decisiones en la resolución de problemas de forma inmediata. La estrategia involucra el orden en el que se utilizarán las tácticas. La etapa táctica se inicia una vez decidida la estrategia.

Makatsiná se encuentra situado en habilidades integradas y conlleva habilidades estratégicas, tácticas y procedimentales, en acciones cognitivas.

Dependiendo de esta clasificación es necesario que se tome en cuenta diferentes:

- condiciones para promover la adquisición de habilidades (*i.e. tutor tipo entrenador-deportivo*).
- métodos de prueba (*i.e. esconder las características no explícitas del problema*).
- técnicas de enseñanza automática (*i.e. enseñanza reactiva*).
- técnicas de análisis (*i.e. diferentes tipos de ACT*)

IV. 2. 1. 2. 3 Modelos mentales

En algunas áreas cognitivas es posible formular teorías de competencia, que especifiquen: qué tiene que ser calculado, cuándo, y por qué; posteriormente en base a estas teorías desarrollar un algoritmo que lo represente. A esta área de estudio se le conoce como la teoría de competencia y se realiza en base a los modelos mentales.

La parte esencial de los modelos mentales radica en su característica funcional. Son definidos como abstracciones funcionales con respecto al trabajo o trabajos que proporcionan un marco deductivo, para la solución de problemas [RYDJ93]. Un modelo mental es distinto de otras formas de representación de conocimiento, como lo son las redes semánticas [LAUA95] debido a que no cuentan con un mecanismo de interpretación para saber si una afirmación es o no verdadera.

El aspecto básico de esta doctrina radica en asumir que las mayoría de las inferencias realizadas diariamente dependen de conclusiones espontáneas y razonamientos que no siempre llevan a una conclusión válida.

Los modelos mentales emergen como un intento de dar sentido a las inferencias, de forma implícita y explícita. Johnson-Laired mencionado en Wild [WILM96] argumenta que un modelo mental permite explicitar aquellos objetos o entidades, así como propiedades y relaciones, que son relevantes para potenciar las acciones. De esta forma cada entidad es representada por las propiedades de sus componentes. Las relaciones entre estas entidades están representadas por las relaciones entre estos componentes. Los modelos mentales son

importantes debido a que representan objetos, estados de asuntos, secuencias de eventos, la forma en que el mundo se encuentra y las acciones sociales y psicológicas de la vida diaria.

Existen dos grandes tipos de modelos mentales los perceptuales y los conceptuales. En cuanto a los perceptuales como su nombre lo indica dependen de los sentidos de percepción humanos. Para estos modelos mentales son importantes dos aspectos: la forma del mundo y la forma en como está el que lo percibe. En cuanto a los conceptuales representan aspectos abstractos. Ambos pueden reflejar: una posible situación, una situación imaginaria o una situación verdadera.

De acuerdo a Jhonson-Laird [JHOP88]: a) un modelo mental, representa una posibilidad, que contiene lo que es común a diferentes caminos en que la posibilidad puede ocurrir. Los novatos no tienden a asumir que cada modelo mental es probable a menos que hay evidencia de lo contrario, b) los modelos mentales representan de forma explícita aquello que es verdadero, pero no lo que es falso. Esta característica lleva a los razonadores novatos a cometer errores sistemáticos.

Un modelo conceptual necesita un mecanismo para su revisión de forma recursiva y hacer uso de operadores de conexión (conjunciones y disyunciones) que permitan determinar la existencia o no de un elemento y poder concluir la existencia o no de las relaciones a las que pertenecen dichos elementos. El uso de las disyunciones debe utilizarse siempre y cuando las condiciones de verdadero sean elementales, para evitar las explosiones combinatorias.

Jonshon-Laird [JHOP83] propone 5 restricciones para los posibles modelos mentales:

1. El principio de computabilidad: los modelos mentales y los mecanismos para su construcción e interpretación deben ser computables.
2. El principio de ser infinito: un modelo mental debe ser finito en tamaño y no puede representar de forma directa un dominio infinito.
3. El principio del constructivismo: un modelo mental esta construido en base a un conjunto de elementos que tienen un arreglo en una estructura, y que representan un estado de asuntos.
4. El principio de economía de modelos: la descripción de un solo estado de asuntos es representada por un solo modelo mental, aún cuando la descripción sea incompleta o indeterminada.
5. Los modelos mentales pueden representar situaciones no derministicas, si y solo si, su uso no es computacionalmente intratable; como un crecimiento exponencial de combinaciones.

En El Alami [ELAM98](1) se presenta un trabajo donde los modelos mentales del experto y del novato juegan un papel importante en la resolución de problemas y la forma en que el novato aprende.

Figura IV. 1 - Modelo Mental para la Solución de Estructuras Triangulares por el Método de los Nodos

¿Es estructura triangular (isostática y estable)? *entonces*

paso

Repite

utiliza reglas de simplificación del trabajo de barras

Hasta-que ninguna regla de simplificación pueda aplicarse

paso 2

Repite

Sin necesidad de calcular reacciones obtener la mayor cantidad de resultados

Hasta-que se necesiten las reacciones

paso

Si se necesita *entonces*

Calcula reacciones

fin-si

paso 4

Repite

Calcula el resto de fuerzas axiales

Hasta-que conocer todas las fuerzas en las barras

paso 5

sino

no tiene solución por este método

fin-si

paso 1

En el dominio específico de esa tarea contiene e integra:

- Conocimiento conceptual con respecto a la situación de un sistema.
- Conocimiento procedimental sobre cómo usar el sistema o cómo actuar en una situación.
- Habilidades de toma de decisiones para poder razonar sobre el sistema o situación.
- Conocimiento estratégico sobre cuándo y cómo habilidades de procedimientos diferentes y de toma de decisiones pueden ser utilizadas, y cómo interactúan o se relacionan los componentes de las tareas. También se considerará el conocimiento táctico.

Los modelos mentales son importantes porque mantienen la conciencia con respecto a la situación global del trabajo, y se pueden encontrar inferencias en el dominio de la tarea, de aquí su importancia en TC complejas.

En el caso específico del dominio de solución de estructuras triangulares. Para generar el modelo mental se partió de la representación mental del problema que tiene el experto durante la solución de éste. Este modelo mental se utilizará en el análisis cognitivo de tareas.

Durante el ACT no se obtiene el modelo mental exacto de un experto, pero si un conjunto de características que pueden ser utilizadas como un marco para la enseñanza del dominio.

Un modelo mental es representado como instrucciones, y puede evidenciar clara y exitosamente las restricciones y características de la tarea; con frecuencia está representado en varios niveles de abstracción, haciendo explícitos principios importantes y relaciones conceptuales que de otra forma serían difíciles de entender.

El formato de un modelo puede ser gráfico como mapas, diagramas de Venn, diagramas de flujo, diagramas estructurados, pseudocódigo, etc. En la Figura IV.1 se muestra el modelo mental del dominio de Makatsiná, utilizando pseudocódigo.

Construir modelos mentales es difícil y consume mucho tiempo, básicamente debido a que el experto desconoce su modelo mental. Por otro lado una vez que se cuenta con el modelo mental, hay que desarrollar un proceso de razonamiento que lo utilice. Realizar un buen ACT es útil para conocer el conocimiento y las habilidades necesarias en TC complejas.

Por todo lo anterior se puede concluir que la psicología dentro de la rama del procesamiento de información, resulta una herramienta muy útil en el desarrollo de los SEIs, pues permite descubrir los elementos medulares en su construcción como son: 1) los diferentes planes instruccionales enlazados a sus tácticas instruccionales, 2) llegar a las abstracciones deseadas en función de la jerarquización de dificultad de una tarea o en función del reconocimiento de diferentes habilidades para la realización de esa tarea, 3) conocer los diferentes puntos críticos dentro de la estrategia general y 4) conocer la competencia de los diferentes niveles de dificultad o de las diferentes habilidades según sea el caso.

IV. 2. 1. 3 La Gráfica genética (GG)

La gráfica genética (GG), es una herramienta para representar el conocimiento que nació de la inspiración de Ira Goldstein [GOLI76] y [GOLI79]. Basada en la epistemología genética de Jean Piaget [GINH86]. En una descripción general esta GG nos muestra el conocimiento (de cualquier tipo) agrupado en islas y enlaces para relacionarlas. Estos enlaces pueden ser de orden o de inclusión, como en el caso de las jerarquías anidadas de Gagné [GAGE85].

También se puede registrar la historia y el estilo de aprendizaje de un estudiante, tomando en cuenta las islas visitadas y la predilección de enlaces utilizados en el recorrido del estudiante durante el aprendizaje. En nuestro caso sólo desarrollamos las propiedades de representación del conocimiento y su tipo de enlace, así como el orden de presentación de las habilidades.

Los enlaces que se utilizan en esta gráfica son y pueden ser ampliados según las necesidades del dominio a modelar. Así lo hacen Bretch & Jones [BREB88], Fernández [FERI89], Laureano [LAUA97](1) y Ramírez [RAMJ94]. A continuación se presenta el significado de los utilizados durante el desarrollo de este ejemplo que se observan en la Figura IV.7:

1. PreCond : este enlace implica un orden de precedencia *antes que*.
2. PostCond: este enlace implica un orden de posterioridad, un conocimiento al que se puede acceder *después de* cubrir el conocimiento al que esta enlazado.
3. Anlg: *analogía*; cuando dos islas están enlazadas por este tipo de enlace, existe una correspondencia de las constantes de una isla a las constantes de la otra isla.
4. Comp: *componente*; este enlace implica que un conocimiento o habilidad está *compuesto por* otro componente.
5. Clase: la *Clase* implica la existencia de una jerarquía conceptual o de habilidades.
6. SubClase: la *SubClase* implica la existencia de niveles de granularidad en la definición de abstracciones conceptuales o de habilidades.
7. EsUn: El enlace *EsUn*, representa la definición de un componente específico de acuerdo al dominio en cuestión.

Los enlaces entre las islas, además de indicar el orden de ejecución de la tarea, explicitan las relaciones, y los datos de entradas y salidas que habrá entre las islas y los diferentes niveles de abstracción, (pueden o no existir) que representan la ejecución del SEI. Esta parte también esta relacionada con la obtención de: los puntos críticos de la estrategia general y de la competencia de los diferentes niveles (en términos de enseñanza).

IV. 3. Metodología

En las siguientes secciones se describirán cómo se encontraron las habilidades que integran esta TC, así como la información que no aparece de forma explícita en el problema y es necesario detectar, para la solución de éste.

IV. 3. 1 El dominio de Makatsiná

El dominio que se eligió es un dominio de habilidades situado en el campo de la mecánica; en un micromundo llamado análisis de estructuras triangulares por el método de los nodos, discutido en Laureano [LAUA92](2). Este dominio conlleva un alto grado de dificultad en la habilidad para resolver las estructuras triangulares, a diferencia de los conocimientos básicos necesarios, que en general son simples: plano cartesiano, estática, vectores, física básica.

IV. 3. 1. 1 Mecánica

La mecánica puede definirse como la ciencia que describe y predice las condiciones de reposo o movimiento de los cuerpos bajo la acción de fuerzas, y esta dividida en tres partes: mecánica de los cuerpos rígidos, mecánica de los cuerpos deformables y mecánica de los fluidos; a su vez la mecánica de los cuerpos rígidos se subdivide en estática y dinámica [BEEF80].

La mecánica es la base de la mayor parte de las ciencias de la ingeniería y es prerequisite indispensable para su estudio. Sin embargo no tiene el empirismo encontrado en muchas ciencias de la ingeniería. Por su rigor y por el énfasis que hace en el razonamiento deductivo, se parece a las matemáticas, sin embargo, no es una ciencia abstracta o pura, sino, una ciencia aplicada.

El diseño de una estructura se divide en dos partes. La primera corresponde a la determinación de los esfuerzos de los miembros de la estructura, de acuerdo a las fuerzas externas a las que esta sometida la estructura. La segunda parte consiste en el diseño de los miembros de la estructura, de acuerdo a los esfuerzos obtenidos en la primera parte. La primera parte se conoce como análisis estructural y la segunda como diseño estructural.

El dominio de Makatsiná se encuentra en la estática y pertenece al análisis estructural. La estructura elegida es una estructura triangular plana y el método de análisis elegido es el método de los nodos [HUAT80] y [BEEF80].

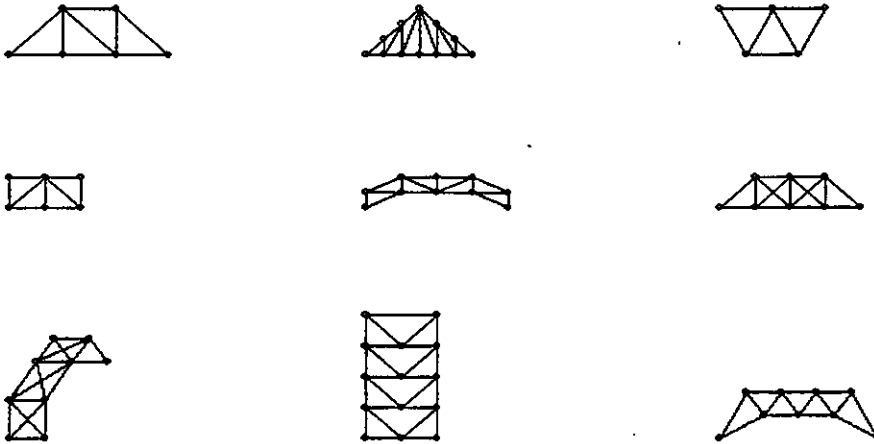
IV. 3. 1. 2 Estructuras triangulares

Los miembros de las estructuras triangulares son rectos y están conectados en sus extremos. Se asume que donde se conectan forman un nodo. Las fuerzas externas actúan en estos nodos en forma de fuerzas concentradas. Los miembros solamente están sujetos a fuerzas axiales, por lo que se induce a un estado uniforme de tensión o compresión axial.

La forma básica de estas estructuras es un triángulo de ahí su nombre. Formado por tres miembros conectados a través de sus nodos. Si se agregan a esta conexión otros dos

miembros que se conectan a dos de los nodos pertenecientes al primer triángulo formarán otro triángulo. Las estructuras construidas de esta forma son internamente rígidas (Figura IV.2).

Figura IV. 2 - Tipos comunes de estructuras triangulares



IV. 3. 1. 3 Análisis por el método de los nodos

El miembro de una armadura está sujeto sólo a fuerzas axiales de tensión o compresión. En general, existe un equilibrio entre las fuerzas internas de los miembros de una estructura y las fuerzas externas que actúan sobre el nodo en cuestión. Se deben satisfacer dos ecuaciones de equilibrio que son $\sum F_x = 0$ y $\sum F_y = 0$.

De las anteriores ecuaciones se derivan las condiciones generales de isostaticidad⁵ que son:

- $NE > NI$; hiperretaticidad
- $NE = NI$; isostaticidad
- $NE < NI$; hipostática

Esto es, para una estructura plana con N nodos, existe un sistema de $2N$ ecuaciones de equilibrio. Y las ecuaciones⁶ en una estructura triangular tomarían la siguiente forma:

⁵ Donde NE representa el número de ecuaciones y NI representa el número de incógnitas.

⁶ Donde N representa el número de nodos, NB el número de barras y NR el número de reacciones.

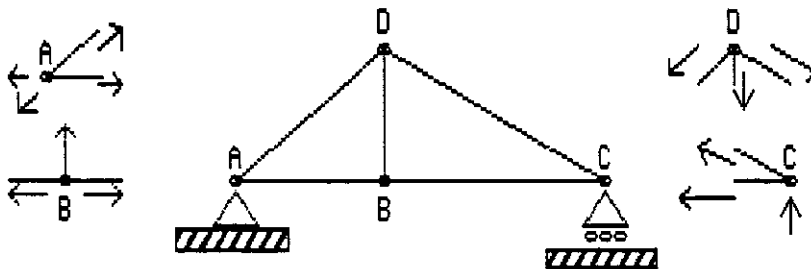
$$2N = NB + NR$$

Esto es para una estructura estáticamente determinada, el número independiente de ecuaciones de equilibrio es igual al número de reacciones desconocidas más el número de fuerzas axiales internas desconocidas. De aquí que se puedan determinar las ecuaciones para la obtención de las fuerzas internas de los miembros de una estructura triangular: 1) dibujando los diagramas de cuerpo libre de cada uno de los nodos, 2) escribiendo las ecuaciones de equilibrio para cada nodo y finalmente 3) resolviendo el sistema linear resultante (Figura IV.3).

IV. 3. 1. 4 Un error común

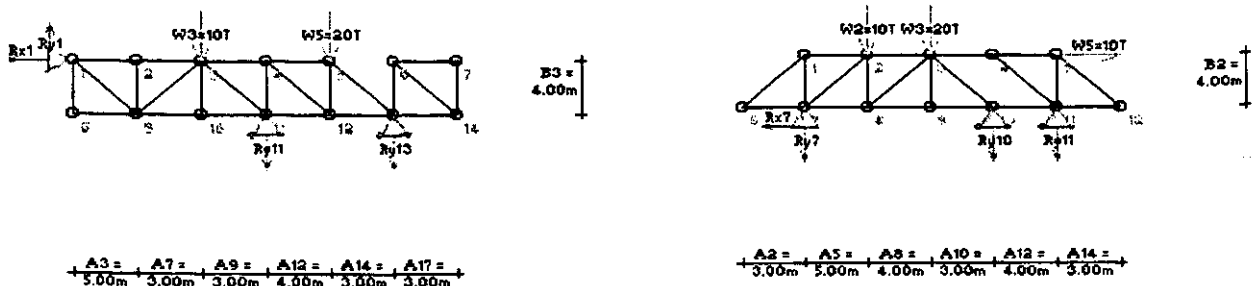
Durante el análisis de la estructura un error común consiste en pensar en una armadura como si esta fuera: 1) estáticamente determinada o 2) indeterminada, sin considerar la relación que existe entre nodos y reacciones.

Figura IV. 3 - Diagrama de cuerpo libre de los nodos



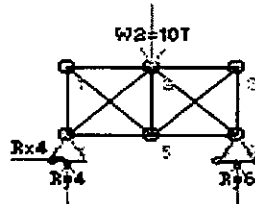
Otro error común es la idea generalizada de que si una armadura tiene más de tres reacciones es hiperestática. Los estudiantes no consideran necesaria la verificación de las premisas que dan verdadero o falso a esta conclusión. Las estructuras de la Figura IV.4 tienen más de tres reacciones.

Figura IV. 4 - Estructuras con más de tres reacciones



Las estructuras de la Figura IV.4 son isostáticas, pero externamente indeterminadas, por lo que pueden ser analizadas por el método de los nodos. Por otro lado existen estructuras con tres reacciones pero, hiperestáticas, tal es el caso de la estructura de la Figura IV.5.

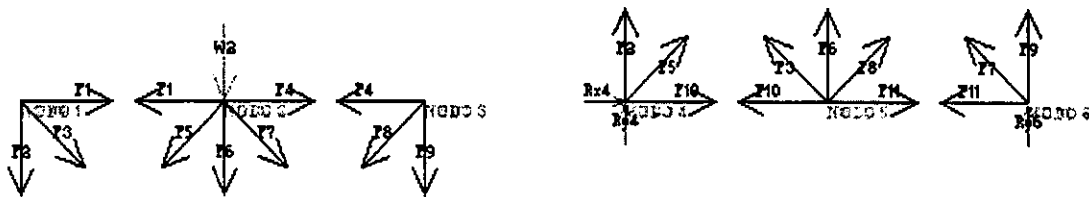
Figura IV. 5 - Estructura con tres reacciones



En el caso de la Figura IV.5 la estructura es hiperestática y no puede ser analizada por el método de los nodos, pero es externamente determinada; de aquí que las reacciones se puedan determinar.

Obteniendo los diagramas de cuerpo libre de los nodos de la estructura de la Figura IV.5 se observan más de dos fuerzas desconocidas por nodo, estos se indican en la Figura IV.6.

Figura IV. 6 - Diagrama de Cuerpo Libre de los Nodos



Este error es considerado como grave e implica una falta de conceptualización, que conlleva la incapacidad de aplicar el método.

El problema radicaba en estar conscientes de ¿cuántas habilidades?, son necesarias en el desarrollo de esta TC.

La adquisición de habilidades está postulada como la inclusión de tres estados de acuerdo a Anderson, Greeno, Kline & Neves [ANDJ81] y Rasmussen [RASJ86]:

1. El aprendizaje verbal del conocimiento en forma de reglas elementales.
2. Durante el aprendizaje de la habilidad, los componentes de la tarea se combinan y se automatizan gradualmente.

3. En el nivel más alto de experticia se desarrolla la automaticidad; ya no se requiere la atención para desarrollar tareas rutinarias.

Durante todo el proceso de análisis y diseño del sistema se consultó a un profesor experimentado en la impartición de la materia (su consulta se mantuvo en todo momento). El representa el experto en el dominio modelado.

Se evaluaron los errores más comunes en el desarrollo de la TC análisis de estructuras triangulares con estudiantes; observando su desempeño tanto en clase, como en los exámenes, y de esta forma se clasificaron los errores.

IV. 3. 2 Por qué un solo nivel de competencia

Ya hablamos en el Capítulo III (sección III.5.6.1, III.5.10 y III.6.2) del significado de los distintos niveles conductuales y sus características.

En el caso de los SEIs solamente estamos percibiendo una forma de representación del entorno que para procesos cognitivos hasta ahora, es la única conocida, dicho en otras palabras, aún no nos podemos asomar a la mente y percibir otras formas de representación.

Esta forma de representación es el desarrollo del estudiante a través de la interfaz, esto es, nuestro entorno es un entorno simulado. En base a él se definirán parámetros y de acuerdo a sus valores se proporcionará la reacción del sistema. Estos parámetros serán los errores. Aunque puede existir la granularidad en la revisión de errores no hay que confundirla con el nivel de competencia, que siempre será uno, *enseñar*.

En Makatsiná sólo contamos con un nivel que comparte el mismo lenguaje y la misma representación del conocimiento y en ese nivel se encuentra la conducta enseñar. Esta se implementa en base a subexpertos cuya subconducta es necesaria para desarrollar la conducta de enseñar. La revisión de errores por parte de los subtutores es independiente (mecanismo de arbitraje de exclusión y jerárquico (sección III.5.6.1.1)). Otro aspecto importante es que el modelo tutorial elegido es el tipo entrenador deportivo (coach), tomado del paradigma de Burton & Brown [BURR82]. Estas subconductas están representadas por subtutores.

Kaelbling mencionó el hecho respecto a la necesidad de que conductas complejas dependían de las condiciones del mundo, y proponía un mecanismo general que sintetizara la información de diferentes sensores de información acerca del mundo; este punto de vista también lo comparte la arquitectura de García-Alegre, Bustos & Guinea [GARM95], para conductas complejas. En un SEI, este requerimiento queda representado por el modelo del estudiante. En el caso del esquema de la acción producida por cada uno de estos subtutores se manejará la arquitectura básica de Brooks de percepción-acción.

Tabla IV.1 - Especificación del ACT de la tarea análisis de estructuras triangulares tomando en cuenta las habilidades

Pasos de desarrollo	Contenido de los pasos	Formas de evaluación	Tipo de representación	Complejidad de los procesos que subyacen a la ejecución
P.1 Saber si la estructura es isostática y estable	Estratégico y Procedimental	Problemas propuestos y MiniExámenes	Reglas sobre estructuras	Discriminación simple y múltiple
P.2 Utilizar las reglas de simplificación visual	Táctico y Procedimental	Problemas propuestos y MiniExámenes	Reglas sobre procesos	Generalización
P.3 Calcular fuerzas sin reacciones explícitas	Táctico y Procedimental	Problemas propuestos y MiniExámenes	Reglas sobre procesos	Generalización
P.4 Calcular reacciones	Táctico y Procedimental	Problemas propuestos y MiniExámenes	Reglas sobre procesos	Discriminación simple y múltiple
P.5 Calcular el resto de las fuerzas	Táctico y Procedimental	Problemas propuestos y Estrategias	Reglas sobre procesos	Generalización

IV. 3. 3 Número de habilidades

Para conocer el número de habilidades (subtutores), en los que se dividió nuestro problema nos ayudamos con el modelo mental del experto (Figura IV.1); éste encuentra distintas formas de representación en: a) los ACTs (Tabla IV.1 y Tabla IV.2), y b) la división conceptual del dominio; que se logra a través de la GG (Figura IV.7).

En primer lugar se encuentra la representación del modelo mental de la tarea cognitiva que se muestra en la Figura VI.1. Basándonos en este modelo, se indican los pasos del desarrollo de dicha tarea cognitiva, necesarios para encontrar los demás componentes del ACT. Estos datos se encuentran sintetizados en la Tabla IV.1. Se hace notar que en la columna de contenido de los pasos, todos ellos constan de procedimientos y estrategias lo que conduce a pensar en habilidades.

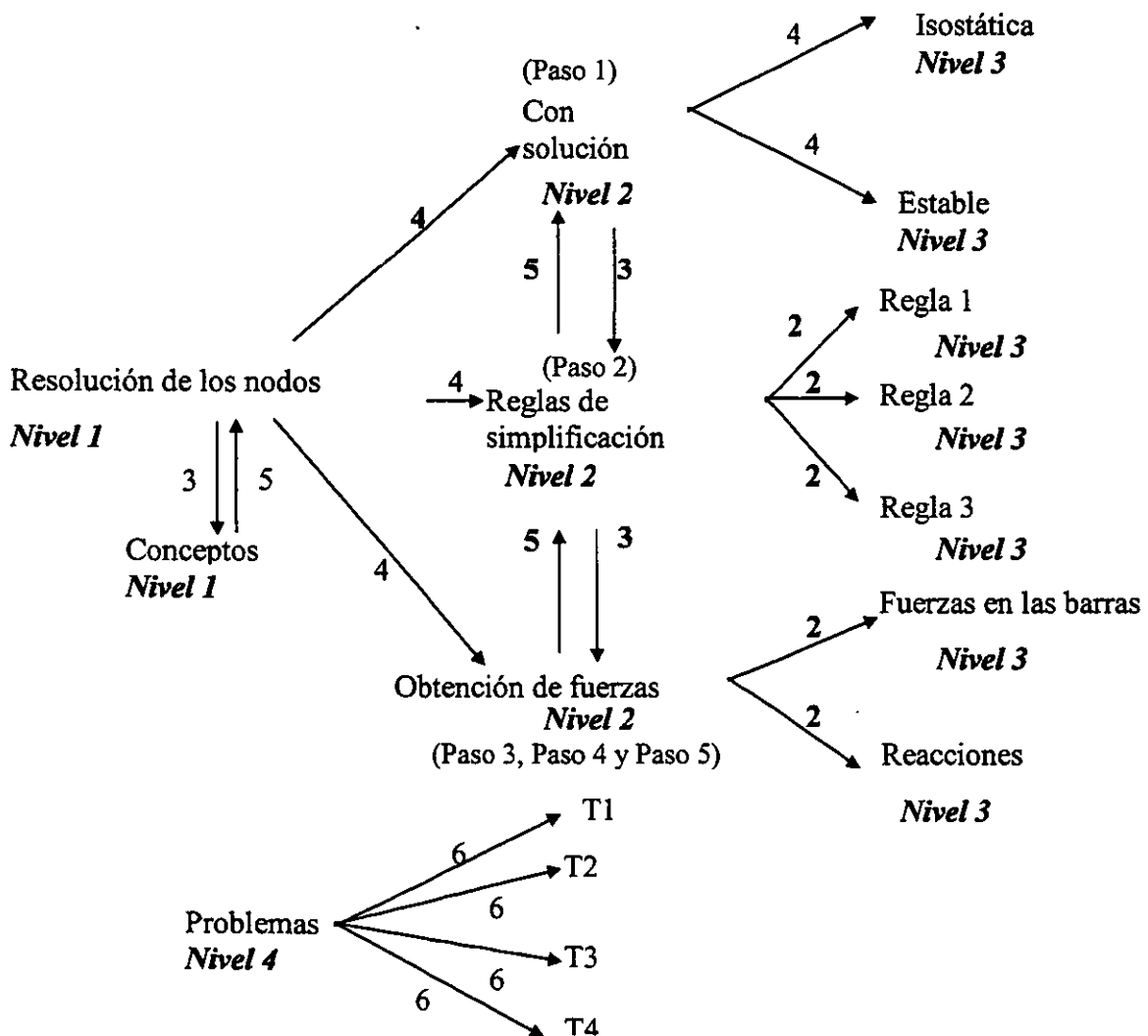
En segundo lugar se complementa este ACT con la propuesta de Ryder et al. [RYDJ93]. De acuerdo a estos últimos dos autores: a) el estado de orientación esta representado por la Tabla IV.1 y la Tabla IV.2, b) el estado de desarrollo de los componentes queda explícito en

las explicaciones verbales de obtención del método (punto IV.3.1.3), y c) el tercer estado sobre la adquisición de habilidades esta representado por la Figura IV.7.

En este análisis se va a encontrar el tipo de habilidades basándose en el modelo mental de la Figura IV.1 y en la taxonomía de habilidades propuesta por los autores. El tipo de conocimiento conceptual, se logra asociando cada paso del modelo mental con su explicación verbal. Los tipos de habilidades asociadas con los pasos del desarrollo se encuentran en la Tabla IV.2. En este caso no existe jerarquía de habilidades.

Figura IV. 7 - Diagrama conceptual del dominio (basándonos en la GG)

1 Clase	4 CompuestoPor	T1 1° SubTut	T4 4° Generales
2 SubClase	5 DespuésDe	T2 2° SubTut	
3 AntesQue	6 EsUn	T3 3° SubTut	



IV. 3. 4 Habilidades que se detectan

Del análisis de la Tabla IV.1 y la Tabla IV.2 obtenemos la primera aproximación al número de habilidades necesarias en el desarrollo de la tarea cognitiva en cuestión. Se detectan un máximo de cuatro habilidades que se encuentran representadas en los pasos ubicados dentro del modelo mental y estas son:

La primera: detectar las características físicas para saber si se puede o no utilizar el método (estratégica). **(Paso 1)**

La segunda: conocer y tener la capacidad para aplicar las reglas de simplificación de barras (reconocimiento de patrones, táctica y procedimental). **(Paso 2)**

En la tercera: se plantea conocer cuándo y cómo obtener las reacciones (táctica y procedimental). **(Paso 3 y Paso 4)**

En la cuarta: se necesita conocer la mecánica para calcular el resto de las fuerzas en las barras (táctica y procedimental). **(Paso 5)**

Estas habilidades se encuentran representadas en el ACT de la Tabla IV.2.

Tabla IV. 2 - ACT de la Solución de Estructuras Triangulares (basándonos en Ryder & Redding)

Habilidades	Conocimiento conceptual	Pasos del modelo mental
Estratégica	Estática	Paso 1
Reconocimiento de patrones, Táctica y Procedimental	Física elemental	Paso 2
Táctica y Procedimental	Estática	Paso 3
Táctica y Procedimental	Estática	Paso 4
Táctica y Procedimental	Estática	Paso 5
Estratégica	Estática	Paso 1

En tercer lugar se utiliza la GG para representar el conocimiento que estamos tratando, ésta se muestra en la Figura IV.7. El objetivo es poder encontrar las clases y subclases del conocimiento conceptual, lo que nos permitirá clasificar los pasos de acuerdo al conocimiento utilizado.

Observando el diagrama se puede concluir que: a) las reglas: 1, 2, y 3 son subclases de la habilidad reglas de simplificación, b) las fuerzas en las barras y las reacciones son una subclase de la habilidad obtención de fuerzas (en ambos casos representado por el enlace número 2). Por lo tanto cada clase, queda representada por una habilidad, que será monitorizada mediante un subtutor.

Para saber si se puede o no emplear el método, se necesitan dos conceptos (isostática y estable). Estos conceptos están inmersos en la habilidad de detectar las características físicas para saber, si se puede o no emplear el método (representado por el enlace número 4). Así que será imprescindible un subtutor que enseñe cómo detectar estas características.

Del anterior análisis se desprende la necesidad de 3 subtutores que desempeñarán el proceso de monitorización durante la sesión del usuario.

Por otro lado también se encuentra representado el dominio de los problemas (T1, T2, T3) que utilizará cada subtutor para comprobar su habilidad y el grupo formado por T4, que representa el conjunto total de problemas al que nos encaminamos cuando se inicia la sesión, (representados por el enlace no. 6). Los componentes básicos son un prerrequisito para aprender la habilidad global, representan los conocimientos de: plano cartesiano, estática, vectores y física básica, que no son tratados por este tutor.

Ahora bien existen otros dos enlaces, que implican el orden de acceso a las habilidades estos son 3 y 5, y no son complementarios. En el caso de *AntesQue*, implica que es una habilidad que se tiene que dominar antes de entrar a otra isla, pero no implica un orden, representa un prerrequisito, sin embargo, el hecho de incluir el enlace *DespuésDe*, obliga un orden de acceso. En el caso particular del aprendizaje de la habilidad de solución de estructuras triangulares por el método de los nodos es imprescindible integrar las habilidades en ese orden ya que el hecho de ser bueno en cualquiera de ellas o en todas no conlleva necesariamente al éxito del desarrollo de la habilidad global.

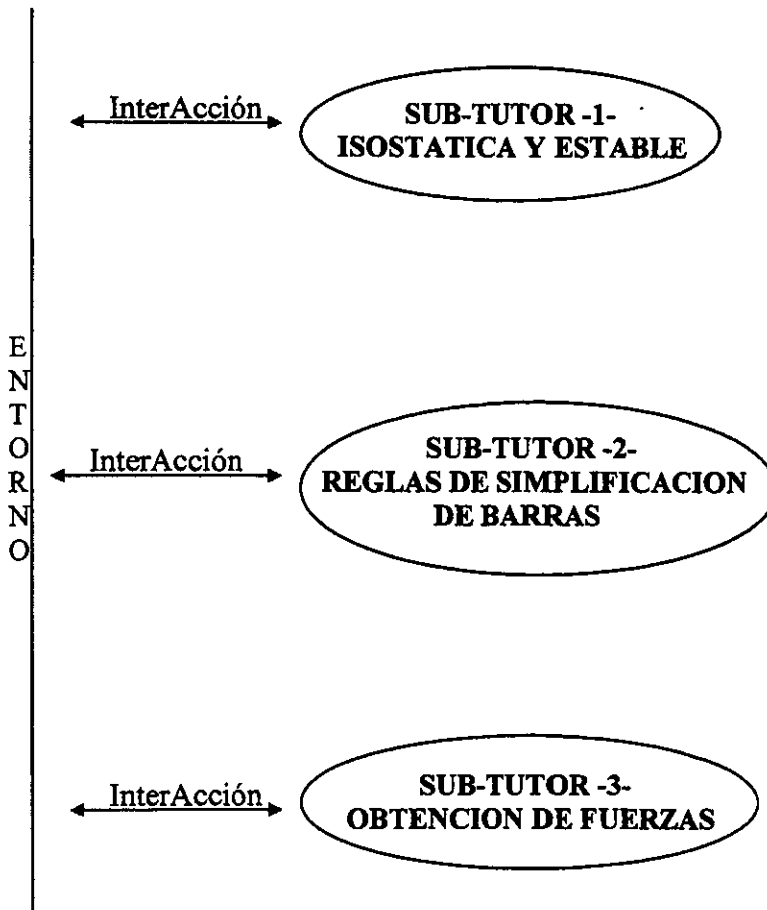
En este caso los niveles representan la abstracción conceptual donde: a) el nivel 1 esta representado por los conocimientos previos antes de abordar el procedimiento de análisis de estructuras por el método de los nodos, b) el nivel 2 esta representado por las habilidades necesarias para lograr la experticia en el procedimiento, y c) el nivel 3 esta representado por los componentes necesarios para lograr esas habilidades; en este caso pueden ser procedimientos (enlace 2) o conceptos (enlace 4). Hay que hacer notar que todas las habilidades se encuentran en el mismo nivel que es el 2, esto se debe a las explicaciones anteriores sobre el significado y caracterización de ellos en los SEIs (sección IV.3.2).

IV. 4 Reflexiones sobre el dominio cognitivo y su enseñanza

En las siguientes secciones explicaremos: 1) cómo basándonos en la necesidad de éstos tres subtutores, se implementó la reactividad del módulo tutorial, y 2) cómo se obtuvieron las características del entorno. Recordemos que de los principios reactivos se desprende la necesidad de obtener las características dinámicas del entorno a ser percibidas. Lo anterior para lograr la reacción del sistema y con ello la funcionalidad emergente.

Al analizar los SEIs el entorno cognitivo hace meditar en otros aspectos importantes en esta área. En los siguientes párrafos trataremos la importancia de la representación del dominio, en nuestro caso habilidades. Todo ello para lograr un proceso de percepción efectivo.

Figura IV. 8 - Reactividad del Módulo Tutorial



IV. 4. 1 Experto Vs. Novato

Mucho se ha hablado de la diferencia que hay en la forma de resolver los problemas que tienen los expertos y los novatos. Durante la última década la IA y la psicología cognitiva han investigado un conjunto de aspectos fundamentales relacionados con el tema, como son: 1) la cantidad de conocimiento que puede ser organizado, y 2) cómo esta organización puede jugar un papel importante para el reconocimiento eficiente en la aplicación de este conocimiento. Lo anterior durante la solución de un problema.

Ambas ramas han llegado a la conclusión de que la solución experta de un problema, depende básicamente de contar con el conocimiento apropiado del dominio específico y no de habilidades intelectuales extraordinarias.

Por otro lado estudios sobre estas diferencias han llegado a conclusiones de que no solamente influye el hecho obvio de que los expertos saben más, sino que también existe una diferencia cualitativa en la organización del conocimiento, así como en la forma en que éste es usado [ELIR90].

En el campo de la mecánica de estructuras, que es el que nos ocupa, existen dos diferencias básicas entre la forma que tienen de resolver los problemas los expertos y los novatos.

IV. 4. 1. 1 La primera diferencia

Los novatos comienzan la solución proponiendo un camino para ir resolviendo poco a poco las incógnitas y las ecuaciones que las resuelven, mismas que serán ocupadas en la siguiente ecuación propuesta. Esta forma hacia-adelante de resolver la realizan iterativamente hasta que quedan resueltas todas las incógnitas o hasta que se dan cuenta que ese camino no es válido, en cuyo caso habrán perdido mucho tiempo.

Los expertos también realizan una solución hacia-adelante. A partir de las variables conocidas generan el sistema o conjunto de ecuaciones que dejan al descubierto incógnitas que se vuelven conocidas. La diferencia básica es que los novatos no son capaces de encontrar esos aspectos que facilitan la tarea y aseguran un camino sin contratiempos.

La distinta forma de trabajo entre unos y otros, hace pensar que existen diferencias cualitativas en el proceso de inferencia. Estas diferencias cualitativas generan información que no se encuentra de forma explícita en el problema y que es fundamental para la solución de éste. Este tipo de información representa en robótica reactiva, las características dinámicas del entorno, que permitirán que el sistema reaccione. Para lograrlo, hay que analizar los errores. Cuando algún estudiante intenta resolver un problema, los errores que comete se deben a no saber identificar detalles que se encuentran de forma implícita. Conocer los

principios de la física general ayudarían al usuario. Estos principios llevan a concluir: 1) cuándo una estructura es o no estable, 2) cuándo es hiperestática, o 3) la capacidad de aplicación de las reglas de simplificación en la geometría de la estructura. Así que, son estos aspectos los que nos interesa que el estudiante aprenda a reconocer. La caracterización de los errores nos dio la pauta para generar las características dinámicas del entorno cognitivo; éstos se encuentran identificados en la Tabla IV.3.

IV. 4. 1. 2 La segunda diferencia

La segunda diferencia está representada por el contenido y organización del conocimiento para resolver el problema. Los novatos clasifican el problema en base a aspectos superficiales, con lo que tienen una representación improvisada. Los expertos los clasifican en base a aspectos más profundos, esto es, tienen un esquema mental más fuerte que les permite centrarse en los principios abstractos de la física y en las restricciones, para aplicar estos principios de acuerdo con su representación conceptual.

Uno de nuestros objetivos al dividir el módulo tutorial era poder representar los aspectos profundos, basándonos en la información que no se encuentra de forma explícita en el problema, y que el experto tiene, y el novato no, y de esta forma, volverlos explícitos al usuario.

IV. 4. 2 Representación del conocimiento

Se necesita un modelo de proceso tutorial que cuente con una representación del conocimiento basada en las unidades de conocimiento. Estas unidades representarán las habilidades. El proceso de percepción se hará a través de estas unidades durante la solución del problema.

El objetivo que se persigue en Makatsiná es que el usuario adquiriera el conocimiento procedimental y táctico de la habilidad para resolver un problema, lo cual en términos de enseñanza es equivalente a crear patrones de especificación que indiquen qué acciones particulares deben tomarse en consideración, y bajo qué conjunto de condiciones particulares del problema hay que llevar a cabo esas acciones determinadas. Esto en el campo de la robótica se traduce en la capacidad de reaccionar ante determinadas características del mundo, las cuales serán sintetizadas por los diferentes agentes, de acuerdo a sus funciones específicas. Lo anterior está relacionado con encontrar las características dinámicas del entorno que disparen la acción de los agentes.

Tabla IV. 3 - Errores críticos para controlar la intervención de los SUB-TUTORES (obtenidos a partir del desarrollo global de la TC)

SubTutores	Isostática y Estable	Simplificación de Barras	Obtención de Fuerzas
Error -1-	No se detecta hiperestaticidad y existe G	Que no haya ejecutado esta opción G	Que se ubique en un apoyo incorrecto L
Error -2-	No se detecta inestabilidad y existe G	Que falten barras por Regla 1 L	Que interprete mal la convención de signos L
Error -3-	Se detecta hiperestaticidad y no existe G	Que falten barras por Regla 2 L	Proyecciones mal usadas S
Error -4-	Se detecta inestabilidad y no existe G	Que falten barras por Regla 3 L	Mala interpretación del efecto de compresión y/o tensión G
Error -5-		Que no exista coincidencia en alguno de los casos o en todos A	El signo en la ecuación esta incorrecto S
Error -6-			Variables mal substituidas S
Error -7-	Tipo de Errores: G: Grave L : Leve S : Superficial A: Aborto		Que no se utilice la opción de transmisión de fuerzas por equilibrio G
Error -8-			Cuando la suma de momentos la usa y no la necesita L
Error -9-			Cuando la suma de momentos la necesita y no la realiza L
Error -10-			Cuando la ecuación no tiene nada que ver con la correcta A

Anderson, et al. [ANDJ81] propusieron una forma para la generalización de patrones de representación del conocimiento. Consiste en comparar aspectos del problema y de las soluciones; se crean unidades de conocimiento donde se asocian aspectos comunes del

problema, esto es, se divide la tarea en pequeños bloques que marquen aspectos fundamentales de la tarea a realizar. Estos bloques representan aspectos que se desea aprenda a ubicar el usuario. Dentro de cada bloque también se desea que aprenda a descubrir los aspectos implícitos de los problemas, que hay que aprender a observar. De esta manera se logra la adquisición de la habilidad.

En nuestra arquitectura MultiAgente las unidades de conocimiento que provienen del patrón general (modelo mental), estarán representadas por los agentes (subtutores), y los aspectos implícitos tendrán su representación en cada conjunto de errores, percibidos por cada agente y representados en la Tabla IV.3.

Finalmente, basándonos en todo este análisis del dominio, la propuesta concreta de la arquitectura reactiva del módulo tutor quedará representada en la Figura IV.8. Cada subtutor representa una habilidad y estas serán vigiladas de forma independiente. Debido al mecanismo de arbitraje elegido y a la forma de diseño de la conducta básica del agente reactivo (Ver Sección III.5.6.1 y Sección III.6.3).

Cada uno de ellos tendrá su percepción propia del mundo y cada uno sintetizará las características de ese mundo que le interesen (Sección III.2.1.5), para así construir su representación interna y actuar en consecuencia.

IV. 5 Aspectos cognitivos en la instrucción

De acuerdo a Castañeda [CASS94] es importante cuando uno elabora material de enseñanza en software, tener en cuenta los siguientes aspectos.

IV. 5. 1 La didáctica del conocimiento procedimental

- El dominio de Makatsiná tiene un contenido altamente abstracto; de aquí que sea óptimo para ser tratado por estrategias imaginales.
- Una vez que el estudiante es capaz de referir conocimiento declarativo, es posible utilizar esta base de conocimiento para el desarrollo de procedimientos de identificación de los diversos patrones perceptuales, motores y simbólicos necesarios para el aprendizaje específico de habilidades; así el estudiante podrá aprender diversas secuencias de acciones que lo capacitarán para resolver problemas.
- Los novatos dependen de procesos controlados que los hace lentos. Su ejecución es propensa al error porque el conocimiento declarativo que manejan es incompleto o está equivocado o la interpretación del conocimiento declarativo ha sobrecargado su

capacidad de memoria de trabajo o porque los procesos interpretativos fueron acelerados más allá de su capacidad.

- Las habilidades integradas necesitan de un procedimiento instruccional con mucha práctica y retroalimentación, para dar la oportunidad de que cada ensayo genere producciones relacionadas, y activadas simultáneamente, en la memoria a corto plazo, con lo que se aumenta la probabilidad de composición.
- El aporte de la computadora como elemento de ayuda a la composición es innegable, así lo demuestra el trabajo de Sueppes, Jerman y Brain (1968) referido en [CASS94], donde se demuestran los efectos de la instrucción auxiliada por computadora sobre la aritmética; se considera que la mejoría de aprendizaje de los estudiantes se debe a que pudieron practicar un gran número de procedimientos compuestos, con lo que tuvieron más capacidad cognitiva en la memoria de trabajo, para dedicarla a la solución de problemas, además de vigilar su desarrollo.
- El aprendizaje de la automatización de procesos y de habilidades cognitivas se ve altamente beneficiado por el uso de la computadora, debido al uso del tiempo, colorido, animación, repetición, registro y almacenamiento, cualidades que contribuyen a la presentación de estímulos instruccionales, en la presentación de mecanismos de generalización, discriminación e igualación a la instancia.
- La computadora puede ser usada como un apoyo de memoria externa dispuesta a facilitar el procesamiento humano, dada su capacidad para actualizar la información, aliviando la capacidad de la memoria humana de trabajo, debido a las sobrecargas impuestas por el material. Estos casos se presentan cuando el aprendizaje tiene que ver con el conocimiento procedimental.

Todo lo anterior fue considerado cuando se realizó el análisis y diseño de Makatsiná.

Capítulo V

Formalización de los agentes de Makatsiná

V. 1 Introducción

V. 1. 1 Modelos formales

A lo largo de este capítulo* se expondrá el modelo utilizado para formalizar los agentes de Makatsiná. La especificación formal de un sistema tiene como objetivo principal la formalización de la arquitectura y sus operaciones. Los beneficios obtenidos de la especificación serían: 1) un mejor entendimiento de la descripción del sistema, y 2) un acercamiento a la integración de la teoría de agentes y la práctica; intentando crear un puente entre ambas.

De acuerdo a Luck y d'Inverno [LUCM95](2) un entorno formal debe satisfacer las siguientes características:

- Debe ser legible y proveer de forma precisa y no ambigua significados para conceptos y términos.
- Debe permitir diseños alternativos de modelos y sistemas particulares para poder ser explícitamente presentados, evaluados y comparados.

* Los contenidos presentados en este capítulo amplían las ideas aportadas en las publicaciones referenciadas como: [LAUA97] (3) y [LAUA98] (3).

- Debe estar lo suficientemente estructurado para proveer una base para próximos nuevos desarrollos o conceptos más refinados.

De acuerdo a Julián, Carrasco y Botti [JULV99] existe un hueco entre el empleo de lenguajes de especificación formal (Z, VDM o CSP) para la especificación formal de agentes y el vocabulario informal empleado por los investigadores que implementan este tipo de sistemas. En este campo las teorías de agentes son en ocasiones abstractas y no se encuentran vinculadas con modelos computacionales concretos.

El lenguaje Z provee un formalismo accesible y sencillo, aunque esta limitado en la capacidad de expresar actitudes mentales y aspectos temporales. Sin embargo, la notación Z viene siendo empleada por diferentes investigadores dentro del campo de los sistemas MultiAgente [LUCM95](1), [LUCM95](2) [GOOR93] y [JULV99].

De acuerdo al planteamiento anterior es necesario establecer vínculos de unión entre los formalismos empleados para la especificación de agentes y aplicaciones reales de agentes. En este capítulo se propone un modelo formal de agentes basado en las investigaciones de Luck y d'Inverno [LUCM95](1); donde éste es vinculado a una aplicación a través de los ideogramas producto de la filosofía de diseño e implementación para sistemas interactivos propuesta por Nigay y Coutaz [NIGL91]. Lo anterior con el fin de lograr un acercamiento mayor entre la conceptualización de los agentes y la implementación de los mismos.

Para abordar este capítulo, empezaremos por dar un conjunto de definiciones y conceptos necesarios para unificar el vocabulario y las ideas que giran en torno a un tema tan nuevo como es el de las arquitecturas MultiAgente; posteriormente en base a ello, definiremos los agentes de Makatsiná.

En el congreso First International Conference on Autonomous Agents llevado a cabo en febrero de 1997, en Marina del Rey California, se hizo un esfuerzo por clasificar los diferentes tipos de agentes autónomos utilizados en diferentes áreas de aplicación con diferentes propósitos. Estos son: asistentes expertos, robots autónomos, softbots y agentes sintéticos [HAYB97].

V. 1. 2 Asistentes expertos

Son agentes de software, que proporcionan asistencia a los humanos en procesos que involucran decisiones complejas u otro tipo de tareas que procesen conocimiento como: monitorización médica, control industrial, administración de negocios, procesos de manufactura que involucren procesos computacionales, administración de tráfico aéreo, etc.

Este tipo de agentes conllevan trabajo relacionado con el aprendizaje, representación del conocimiento, planeación y razonamiento con recursos acotados, todo dentro de un marco práctico.

V. 1. 3 Robots autónomos

Estos agentes tienen la característica de una interacción muy fuerte a través de sus perceptores con el entorno y en base a estas percepciones actuar en dicho entorno.

Son utilizados para medios agresivos al hombre e interactúan también con otros objetos y con otros robots, su característica principal es que son ubicados en el mundo real, con todas las sorpresas y continuidad que ésto implica.

Estos dos tipos de agentes llegan con fuerza para la investigación de la IA, debido a que son capaces de operar en mundos reales y complejos y su construcción es esencialmente sencilla.

V. 1. 4 Softbots

Son definidos como programas que interactúan con entornos de software considerados como reales, tal es el caso de un sistema operativo de PC, la red Internet o el World Wide Web. Sus perceptores observan aspectos de este entorno y sus efectores son capaces de alterarlo directamente y comunicarse con otros agentes.

V. 1. 5 Agentes sintéticos

Operan en entornos simulados como es el mundo virtual, MUDS o los juegos de vídeo sus cualidades radican en la credibilidad y personalidad más que en la inteligencia o la experticia y pueden tener papeles en sistemas interactivos para entrenamiento, arte o educación.

Los agentes de Makatsina se encuentran situados en la clasificación de asistentes expertos, dado que su papel es de monitor, esto es, tienen que vigilar que el estudiante utilice correctamente la habilidad a su cargo.

V. 2 Definiciones

Tomaremos como base las definiciones dadas por Luck and d' Inverno [LUCM95], en las que se llega a definir: objetos, agentes y entorno, utilizando el lenguaje Z del que se presupone un conocimiento básico [SPIJ88].

V. 2. 1 Definiciones básicas

Acción: es un evento discreto que cambia el estado del entorno y no necesariamente es observado.

Atributo: es un aspecto que se puede percibir, pertenecen al mundo y son las únicas características manifestables, potencialmente accesibles por una computadora.

Objeto: es una entidad que contiene un conjunto de acciones y un conjunto de atributos.

CapazDe: P Acciones

Atributos: P Atributos

dónde P : representa el valor del dominio.

Entorno: es un conjunto de atributos que describe todos los aspectos del mundo.

$Entorno == P$ Atributos

Meta: está representada por un estado, compuesto por un conjunto de asuntos que se lleva a cabo en el entorno.

$Meta == P$ Atributos

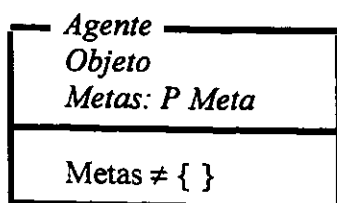
Interacción: se le llama a lo que sucede, cuando un conjunto de acciones son llevadas a cabo en el entorno; los efectos de una interacción en el entorno, son determinados al aplicar la función sobre la definición de sus axiomas, bajo las cuales se encontraba el entorno en curso y de las acciones llevadas a cabo.

EfectoDeInteracción: $Entorno \rightarrow P$ Acción α Entorno

donde: \rightarrow representa la correspondencia en un sentido de un dominio hacia otro.

α representa la correspondencia parcial en un sentido, de un dominio, hacia otro.

Agente: es una instanciación de un objeto, que tiene asociadas una o un conjunto de metas.



Un *objeto* puede convertirse en *agente* al servir a los propósitos de otro agente o a los propósitos de el mismo; cuando su actuación termina, nuevamente se convierte en objeto, es

el caso de la calabaza en el cuento de la Cenicienta; antes de ser utilizada por el Hada Madrina era tan solo un *objeto* se encontraba en el campo, sin ningún propósito específico, sin embargo, en el instante que tuvo un objetivo, asignado por otro *agente* en este caso, el Hada Madrina, que le asignó ser carroza para Cenicienta, se convirtió en *agente* y su meta duró hasta las 24 horas de aquella memorable noche, donde después volvió a ser objeto.

Hay que tener en cuenta que *la meta de un agente* puede existir de forma explícita o implícita, por lo que el comportamiento de un agente puede ser pasivo o activo.

Pasivo: sus metas son impuestas o asignadas, como en el caso de la calabaza.

Activo: se refiere al caso en que un agente es capaz de cambiar el estado del entorno; desarrollando acciones que satisfagan sus metas, como es el caso de un robot, un sistema experto o un SEI, compuesto por agentes.

Hasta aquí hemos definido las características principales de un objeto y un agente ahora pasaremos a definir los aspectos que intervienen cuando un agente entra en acción.

Un *agente* inmerso en un entorno cuenta con un conjunto de percepciones a las que tiene acceso, siendo estas los posibles atributos que el agente puede percibir del mundo, esta percepción a su vez depende de las capacidades del agente y del estado en curso de ese mundo.

Es importante mencionar que un agente no puede percibir todos los atributos del entorno, por lo que basa sus acciones en un subconjunto de estas, esta parte esta relacionada con la representación parcial del mundo propuesta por Brooks [BROR91](1) (Sección III.2.1.5) y comentada ampliamente en el capítulo de Agentes Reactivos. También existe la posibilidad de que *un agente no tenga la característica de percibir*, pero no por ello deja de ser un agente.

A continuación se definirán las funciones que permiten a un agente actuar, teniendo en consideración que el cuerpo de un agente se encuentra encapsulado y dividido en dos partes. La primera representa las funciones que puede realizar y solo a través de ellas se podrá invocar su participación dentro del sistema y la segunda representa la implementación de éstas y permanece oculta. Empezaremos por el concepto de vista.

Vista: es la percepción que un agente tiene del entorno.

V. 2. 2 Definiciones de los esquemas de un Agente

PercepciónDeUnAgente

AccionesQuePercibe: P Acción

Lo que puede hacer en base a sus capacidades.

LoQuePuedePercibir: Entorno \rightarrow P Acción α Entorno

Son los atributos potencialmente disponibles, basándose en sus capacidades perceptoras.

LoQuePercibirá: P Meta \rightarrow Entorno \rightarrow Vista

Representa los atributos que se perciben *en un instante dado*, de acuerdo a sus metas.

En el otro lado de la balanza se encuentra la acción producida por el agente donde en correspondencia directa con la meta o metas de dicho agente, existe una *función de selección-acción* la cual depende de; las metas, del estado del entorno en curso y de las percepciones en un instante dado. En el caso de no existir percepciones la función de selección-acción, solamente depende del entorno. Todo esto queda especificado en:

LaAcciónDeUnAgente

AccionesDeUnAgente: P Meta \rightarrow Vista \rightarrow Entorno \rightarrow P Acción

Asegura que la función de selección-acción regresa un conjunto de acciones que se encuentran dentro de la competencia del agente.

También introduciremos un esquema multi-agente, producto de una filosofía de Ingeniería de Software, que nos ayudará a comprender mejor la organización de los agentes de Makatsiná.

V. 3 Ingeniería de software

Los SEI pueden ser clasificados como sistemas interactivos. Dentro del proyecto de investigación Espirit BRA 3066 AMODEUS, se diseñó una arquitectura MultiAgente.

Uno de los propósitos de esta investigación era proporcionar una guía efectiva para el desarrollo de sistemas interactivos, así como representar al sistema en un marco de diferentes

niveles de abstracciones con el fin de lograr una organización paralela MultiAgente, [NIGL93] y [NIGL94].

V. 3. 1 Arquitectura MultiAgente en software

Las implicaciones de esta arquitectura MultiAgente en términos de ingeniería de software son:

- Un agente es una unidad modular.
- Un agente es una unidad computacional.
- Un agente mantiene un estado local.
- En un instante dado un agente puede escuchar varios eventos o producir varios eventos.

Dentro de las ventajas de estos agentes, está el hecho de que soportan el desarrollo oportunístico del usuario final, así como múltiples entradas y salidas paralelas, en este punto es necesario aclarar que las especificaciones externas son un punto primordial para poder llegar a la división de estos sistemas en agentes.

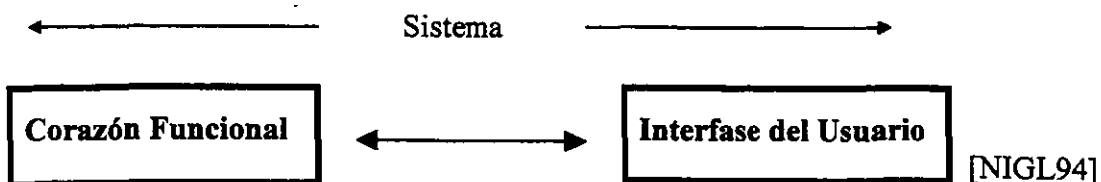
V. 3. 2 Organización de un sistema interactivo

La arquitectura MultiAgente fue desarrollada tomando como base la arquitectura para un sistema interactivo, observada en la Figura V.1.

El corazón funcional representa la implementación de los conceptos dependientes del dominio.

La interfase del usuario representa la cara del sistema.

Figura V. 1 - Arquitectura para un Sistema Interactivo



V. 3. 3 Modelo MultiAgente

Del modelo anterior nace el modelo MultiAgente y su anatomía consta de tres partes básicamente:

- Una presentación (P).
- Una abstracción (A).
- Un control de diálogo (CD).

La presentación representa la parte del agente que es vista por el mundo; está relacionada con alguna técnica de presentación.

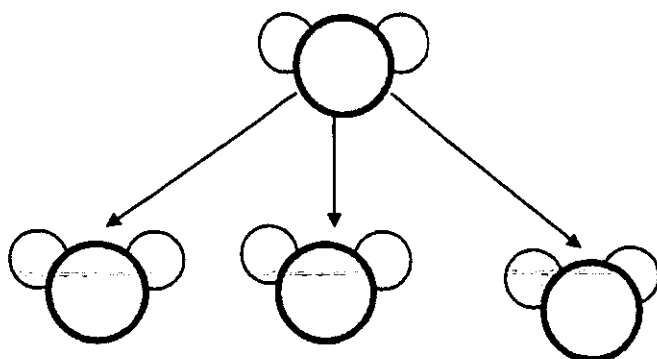
La abstracción representa el estado local del agente; es la parte donde se encuentran los objetos conceptuales para poder tener acceso al dominio; en ella se encuentra implementada la competencia del agente.

En cuanto a la parte llamada *controlador de diálogo*, es donde se lleva a cabo la coordinación entre la abstracción y la presentación y la coordinación con otros agentes en caso de ser necesaria; cabe mencionar que pueden existir agentes sólo con la parte de presentación o sólo con la parte de abstracción o con ninguna de éstas.

Este tipo de arquitectura además soporta la jerarquía como se observa en la Figura V.2, que es útil porque puede expresar:

- La continuidad en los niveles de abstracción.
- Algunas relaciones entre agentes.
- Coordinación de subagentes por un agente de nivel superior.

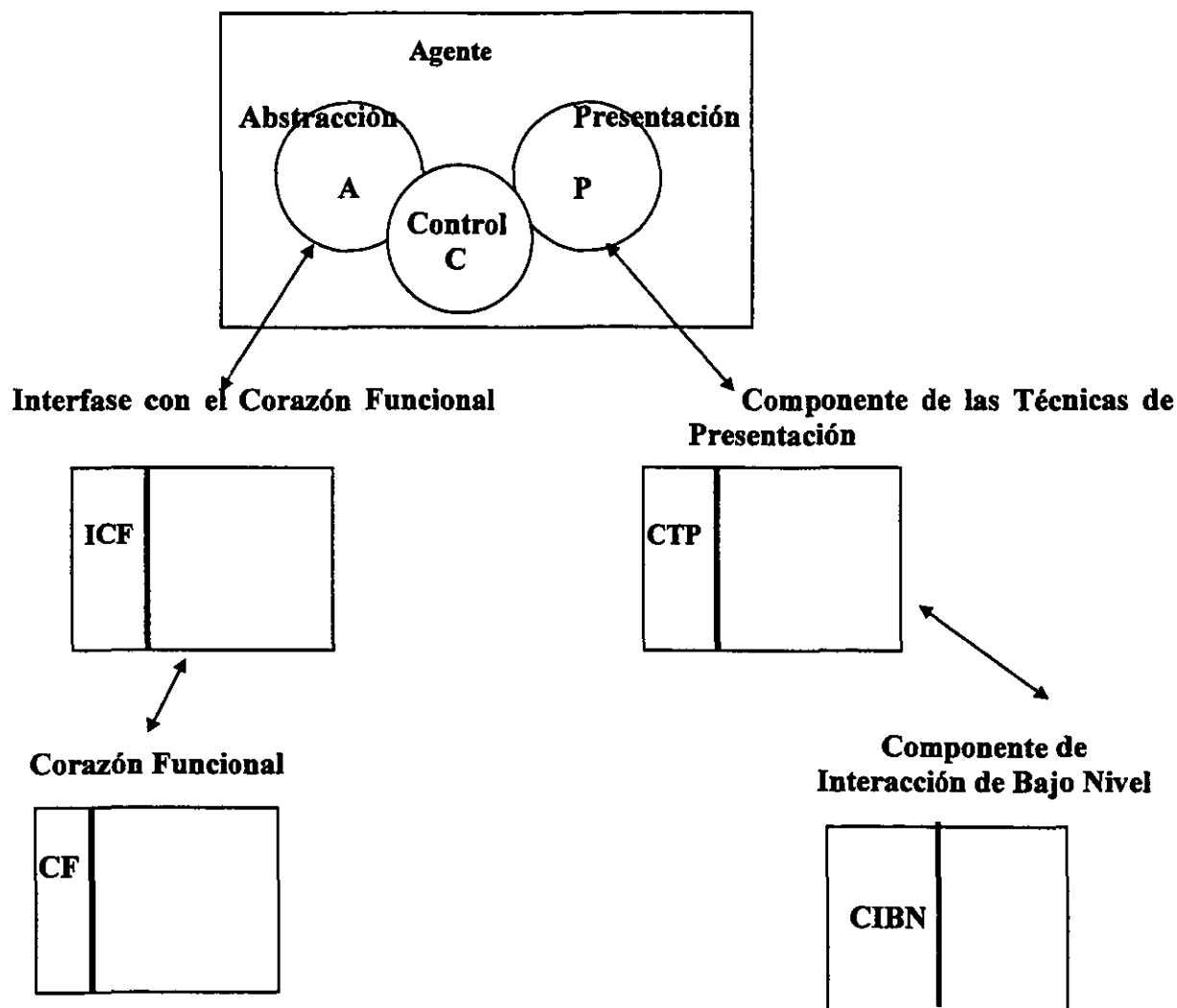
Figura V. 2 - Jerarquía de Agentes



Ahora bien cada parte de la abstracción de un agente consta de un componente que representa la *interfase con el corazón funcional (ICF)*; ésta será el puente entre lo que el

agente debe hacer con el dominio (los objetos conceptuales) y el *corazón funcional* (CF). El CF está constituido por los elementos del dominio. Por otro lado en la contraparte; la *presentación* consta de un componente que contiene las *técnicas utilizadas en la presentación* (CTP) del sistema hacia el exterior, y éste a su vez consta de un componente que será el receptor directo de los *elementos en el mas bajo nivel de abstracción* (CIBN), esto es: mouse, teclado, sensores, motores, ruedas, etc. Todo esto se observa en la Figura V.3.

Figura V. 3 - Anatomía de un Agente

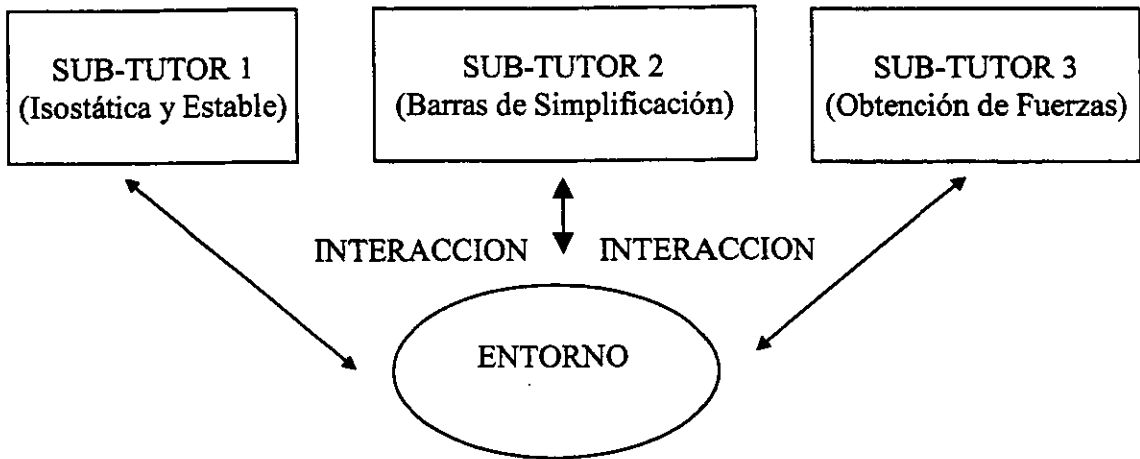


V. 4 Agentes de Makatsiná

Contamos con tres agentes independientes, esto es, no se comunican entre ellos. La tarea a ser desarrollada por estos agentes como ya se ha mencionado es una tarea jerárquica (Sección III.5.6.1), con lo que tenemos resuelto el problema de intervención durante el desarrollo de la sesión. Los agentes revisarán la tarea en determinados puntos críticos (ubicados por el experto) y lo harán de forma ordenada debido a la jerarquía mencionada. En caso de que alguno de ellos encuentre una falla entrará en acción todo su mecanismo interno. Las

características particulares de cada agente están basadas en el diagrama de reactividad de la Figura V.4.

Figura V. 4 - Reactividad del Módulo Tutorial



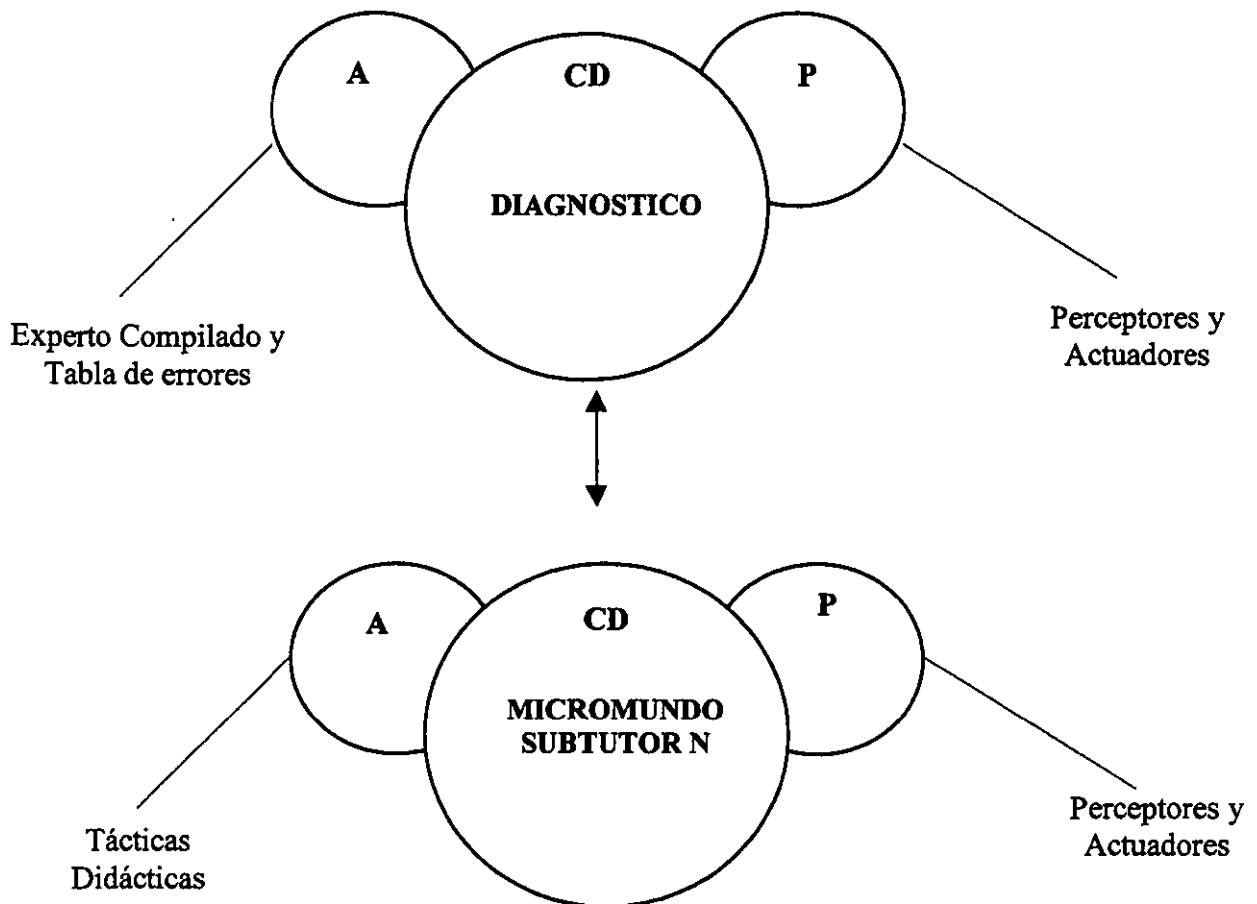
Recordemos que en Makatsiná se cuenta con un tutor tipo entrenador y que la intervención tutorial se realizó con seguimiento de asuntos de acuerdo a Burton & Brown [BURR82]. Dado que el comportamiento de nuestros agentes está dividido en dos subagentes; uno que es el diagnóstico, y su objetivo es observar el entorno (que consiste en el desarrollo del estudiante). A partir de esta observación se obtiene evidencia en base a las capacidades perceptuales del agente. Para saber si el estudiante usa, o no usa, o usa de forma incorrecta, la habilidad monitorizada y controlada de forma exclusiva por ese agente. De sus observaciones se detecta el o los errores cometidos y entonces entra en acción el otro subagente, representado por un MicroMundo cuya tarea consiste en crear un entorno que ayude al estudiante a aclarar sus dudas, todo mediante la utilización de las tácticas didácticas que guiarán su intervención hasta que el alumno sea devuelto al entorno principal, donde se cometió el error.

La intervención del MicroMundo consiste en presentar una explicación seguida de ejemplos parciales que muestran el uso correcto de la habilidad; posteriormente se realizará un mini-test, debido a que Makatsiná al final de la sesión emitirá un diagnóstico. Es necesario mencionar que debido al enfoque de entrenador, Makatsiná, cuenta con la capacidad en todo momento de mostrar el uso de la tarea por un experto, antes de ser invocado el MicroMundo.

En el nivel de abstracción más general un agente de Makatsiná está representado por la Figura V.5. En dicha figura, la abstracción está representada por la tabla de errores y las tácticas didácticas, por el otro lado la presentación está simbolizada por los medidores y

actuadores que son todos los mecanismos de percepción a través de la interfase y todos los mecanismos de intervención tutorial, tales como: animación, texto, sonido, etc.

Figura V. 5 - Agente Global Del Sistema



El modelo funcional de estos agentes, se encuentra representado por la Figura V.6 y por la Figura V.7, donde el funcionamiento global de un agente visto como un subtutor, se encuentra dividido en dos subagentes que representan la PercepciónDeUnAgente y la AcciónDeUnAgente visualizado en dichas figuras, respectivamente.

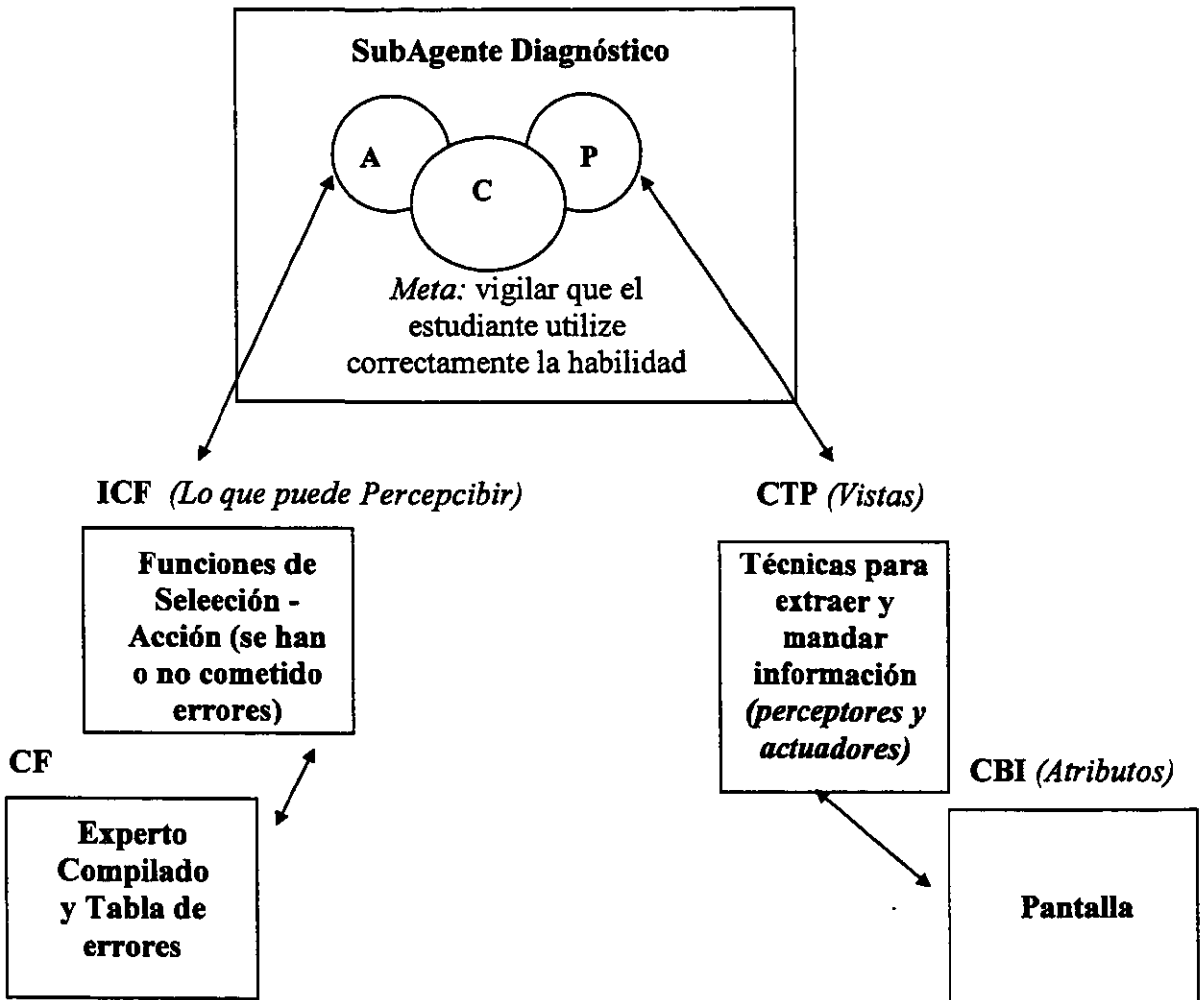
V. 4. 1 Percepción de un agente

En la Figura V.6 se muestra la parte relativa a la percepción. En ella se encuentran los atributos que es capaz de percibir de acuerdo a sus capacidades y en un instante dado, donde el entorno en curso se encontrará en un determinado estado. A nivel de acciones lo anterior se traduce en la revisión, que por parte de ese agente, tendrá lugar en el desarrollo de la tarea cognitiva del alumno. En los agentes las capacidades de percepción están representadas por los errores que le competen a él. En caso de error, este agente lo detectará y se lo comunicará al agente MicroMundo.

V. 4. 2 La Acción de un agente

La Figura V.7 muestra la parte relativa a la acción de dicho agente, lo que se traduce en que de acuerdo a la meta que es tratar el error producido por el estudiante se elegirá una táctica remedial la cual conlleva una serie de acciones didácticas que tendrán efecto en el entorno representado por la pantalla.

Figura V. 6 - Modelo Funcional (Parte -1-)

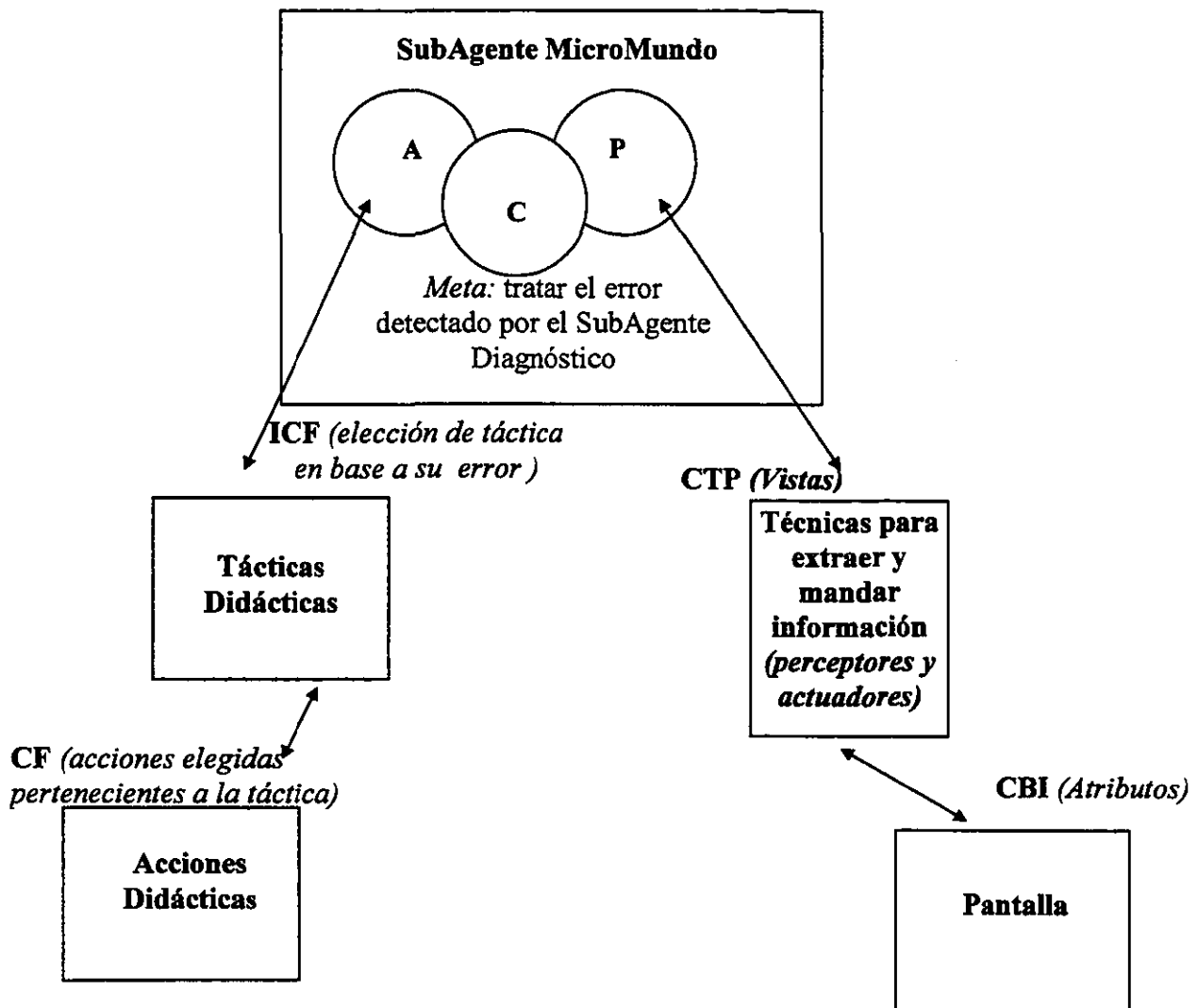


V. 4. 3 Definición de los esquemas de los agentes de Makatsiná

En base a estas explicaciones definiremos los esquemas de cada agente de Makatsiná, compuesto por los subagentes mencionados, donde de forma recursiva cada uno de ellos a su vez contará con percepción y acción en un grano más fino.

La definición de los esquemas de los diferentes agentes será la misma, lo único que cambiará serán los errores a tratar (ver Tabla V.2) y el orden de prioridad de revisión del entorno, dada la jerarquía del mecanismo de arbitraje (Sección III.5.6.1).

Figura V. 7 - Modelo Funcional (Parte -2-)



V. 4. 3. 1 Anatomía del agente subtutor

De acuerdo a lo anterior tenemos dividido el funcionamiento de un agente (subtutor), en dos subagentes, dónde la meta del agente MicroMundo es impuesta por el agente Diagnóstico. La habilidad que manejan los diferentes subtutores se encuentra en la Tabla V.1.

Tabla V. 1 - Habilidades que revisan los Subtutores

AGENTE	HABILIDAD
SubTutor 1	Isostática y Estable
SubTutor 2	Simplificación de Barras
SubTutor 3	Obtención de Fuerzas

V. 4. 3. 1. 1 SubAgente diagnóstico

Meta: Manejar la habilidad correctamente y en el orden indicado de prioridad.

ClaseDeMeta: Activa.

- ***Percepción DeUn Agente (Diagnóstico)***

AccionesQuePercibe: Revisar a través de sus perceptores o sea, la pantalla.

LoQuePuedePercibir: Errores que le competen.

LoQuePercibirá: Errores cometidos en un *instante dado*.

- ***LaAcciónDeUnAgente (Diagnóstico)***

AccionesDeUnAgente: Activar al agente MicroMundo en caso de error, en caso contrario continuar con la labor de centinela durante el desarrollo.

V. 4. 3. 1. 2 SubAgente MicroMundo

Meta: Tratar el error detectado a través de una táctica didáctica remedial.

ClaseDeMeta: Pasiva.

- ***PercepciónDeUnAgente (MicroMundo)***

AccionesQuePercibe: Expone a través de sus actuadores la táctica remedial involucrada con el tipo de error.

LoQuePuedePercibir: Conocer los resultados del MiniTest, practicado al estudiante.

LoQuePercibirá: Conocer si el estudiante ha comprendido, o no, el concepto erróneo a través de los resultados del MiniTest.

- ***LaAcciónDeUnAgente (MicroMundo)***

AccionesDeUnAgente: Regresar al entorno original una vez tratado el error y presentado el mini-test o en caso de cometer error de aborto, sacarlo del sistema, con las recomendaciones pertinentes.

De esta forma se implementa el circuito de interacción global del sistema compuesto por tres agentes y el entorno. Representado por la interfase que es el receptáculo del desarrollo del estudiante.

Tabla V. 2 - Errores Críticos para Controlar la Intervención de los SUBTUTORES (obtenidos a partir del desarrollo global de la TC)

SubTutores	Isostática y Estable	Simplificación de Barras	Obtención de Fuerzas
Error -1-	No se detecta hiperestaticidad y existe G	Que no haya ejecutado esta opción G	Que se ubique en un apoyo incorrecto L
Error -2-	No se detecta inestabilidad y existe G	Que falten barras por Regla 1 L	Que interprete mal la convención de signos L
Error -3-	Se detecta hiperestaticidad y no existe G	Que falten barras por Regla 2 L	Proyecciones mal usadas S
Error -4-	Se detecta inestabilidad y no existe G	Que falten barras por Regla 3 L	Mala interpretación del efecto de compresión y/o tensión G
Error -5-		Que no exista coincidencia en alguno de los casos o en todos A	El signo en la ecuación esta incorrecto S
Error -6-			Variables mal substituidas S
Error -7-	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> <p>Tipo de Errores: G: Grave L : Leve S : Superficial A: Aborto</p> </div>		Que no se utilice la opción de transmisión de fuerzas por equilibrio G
Error -8-			Cuando la suma de momentos la usa y no la necesita L
Error -9-			Cuando la suma de momentos la necesita y no la realiza L
Error -10-			Cuando la ecuación no tiene nada que ver con la correcta A

V. 5 Tratamiento de los errores

Los errores fueron obtenidos a partir del desarrollo de la tarea vista de forma global. Una vez que fueron detectados los agentes y sus correspondientes habilidades, se realizó un reparto entre éstos. No debemos olvidar que Makatsiná se ubica en un nivel de enseñanza donde lo que interesa es que el estudiante aprenda el uso de una determinada estrategia, de aquí el tratamiento de los errores a nivel didáctico.

V. 5. 1 Clasificación de los errores

En Makatsiná contamos con 4 tipos de errores que se pueden observar en la Tabla V.2 y son:

- Grave
- Leve
- Superficial
- Aborto

Los errores graves, implican una falta importante de conceptualización, lo que conlleva al fracaso de la aplicación del método de resolución.

Los errores leves, implican falta de atención global, más que falta de conocimiento, esto es, cuentan con el conocimiento específico, incluso pueden haberlo utilizado previamente, sin embargo, por alguna falta de atención se desorientan y no completan parte del proceso.

Los errores superficiales, son aquellos errores donde se detectó más falta de atención que de conocimiento, como puede ser, un signo incorrecto en una ecuación o una ecuación de proyección en el plano cartesiano mal construida.

Los errores de aborto, implican una falta total de conocimiento, con lo cual se considera imposible proseguir, recordemos que este tutor es de entrenamiento y se presupone un conocimiento inicial del método de resolución, visto en clase.

Capítulo VI

Ejemplos del desarrollo de Makatsiná

VI. 1 Introducción

A lo largo de este capítulo* se describirá el funcionamiento de Makatsiná; éste es concebido como un tutor de diagnóstico creado desde un principio poniendo énfasis en su parte de reactiva, esto es, su nivel conductual inmediato. Al finalizar la sesión informa sobre los puntos donde el alumno ha fallado. Para lograrlo cuenta con una representación del estudiante no exhaustiva y distribuida en los tres agentes. De acuerdo a la filosofía reactiva ampliamente tratada en el Capítulo III (Sección III.1.2 y III.1.3).

Por otro lado en Makatsiná se enseña un determinado procedimiento, de aquí que cuente con una intervención tutorial tipo entrenador 'coach'; donde en caso de cometer más de una vez el mismo error el tutor le mostrará el desarrollo correcto a partir de cualquier punto del ejercicio. Además cuenta con la opción de autorresolución donde se le muestra al alumno la solución de cualquier estructura paso a paso, en este modo el alumno no puede intervenir.

Recordemos que Makatsiná funciona en base a *tres agentes*; considerados como asistentes expertos, debido a que son tres expertos en las subhabilidades que componen el procedimiento de resolución de estructuras triangulares por el método de los nodos. Con base

* Los contenidos presentados en este capítulo amplían las ideas aportadas en la publicación referenciada como: [LAUA98] (1).

en lo anterior se implementó un mecanismo de interacción sencillo, que constituye la parte reactiva de la enseñanza.

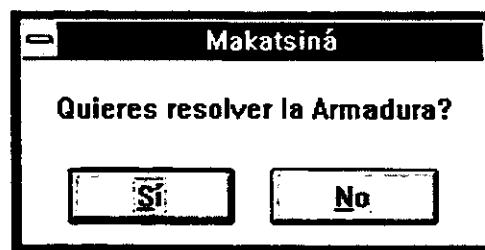
La intervención de estos subtutores es invisible a los ojos del alumno, mientras éstos no sean despertados el usuario puede equivocarse y corregir, regresar en los pasos, comenzar de nuevo o simplemente continuar. Cuando los subtutores son invocados, éstos se *asoman* a revisar el desarrollo del usuario y es cuando; sí el alumno ha cometido algún error comenzará la sesión de enseñanza y corrección en alguno de los micromundos pertenecientes a los subtutores (Ver Sección III.5.6).

Los subtutores son invocados cada vez que se inserta un paso nuevo dentro de la resolución, aunque también pueden invocarse, por errores directos, esto es, que no involucren pasos, i.e. hiperstaticidad, inestabilidad o simetría (Ver secciones III.5.6.1.1 y III.5.6.1.2). Para seguir estos ejemplos vale la pena tener en mente la tabla de errores que se encuentra en el Capítulo V. Tomando en consideración lo anterior se mostrarán los mecanismos de interacción de los distintos subtutores (asistentes expertos) en base a los errores cometidos durante el desarrollo de los ejemplos, así como el diagnóstico al finalizar cualquier sesión.

Se tomarán en cuenta los diferentes tipos de pantalla, con el fin de explicar el desarrollo de los ejemplos. Se dará una breve descripción de las diferentes áreas que la constituyen, así como de la forma de inicio.

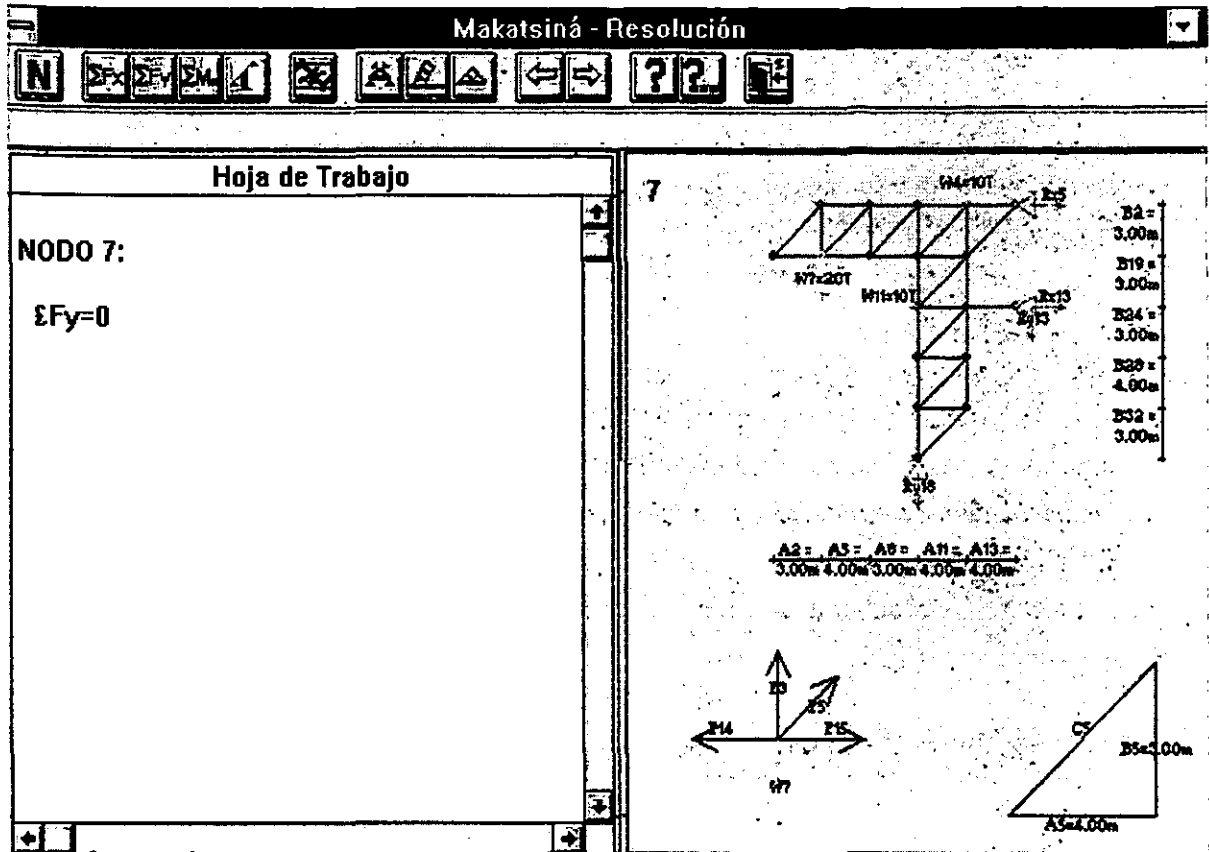
Cuando ejecutamos Makatsiná desde su ideograma o desde el administrador de archivos aparece la ventana de la Figura VI.1. La pregunta implica si se desea o no *el modo de autoresolución*, al cual se entraría al contestar de forma negativa.

Figura VI. 1



El sistema cuenta con la pantalla de la Figura VI.2 dividida en dos áreas principalmente, siendo una el área de trabajo del alumno y la otra donde aparece la estructura con sus características físicas. Se elaboró una interfaz lo más parecida a la forma manual en que es resuelto el problema, en ella las barras cambian por los colores convencionales (rojo y azul), cuando están resueltas. También se manejan colores en los nodos seleccionados para resolver, éstos son diferentes a los que aún no están resueltos.

Figura VI. 2



En la Figura VI.2 también observamos que se cuenta con una barra de herramientas, cuyos botones serán los que guiarán el desarrollo del ejercicio.

Botón de Nueva Estructura



Este botón permite al usuario permanecer en el sistema solicitando nuevas estructuras hasta que desee salir, acción que tendrá que combinar con *el botón de Salir*.

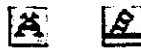
Botones de Sumatorias



Estos botones se utilizan cuando se desea hacer una *sumatoria de fuerzas sobre el eje X*, *sumatoria de fuerzas sobre el eje Y*, o *sumatoria de momentos*. En todos los casos al seleccionar una de estas opciones se genera al final de la *ventana de edición* alguno de los siguientes símbolos según sea el caso:

$$\begin{aligned}\Sigma F_x &= 0 \\ \Sigma F_y &= 0 \\ \Sigma M_o &= 0\end{aligned}$$

Botones de hiperestaticidad e inestabilidad



Estos botones deberán presionarse si el usuario detecta que la estructura no puede resolverse puesto que se trata de una estructura hiperestática o inestable. Estos son los errores directos ya referidos anteriormente (Ver secciones III.5.6.1.1 y III.5.6.1.2).

Botón de simetría



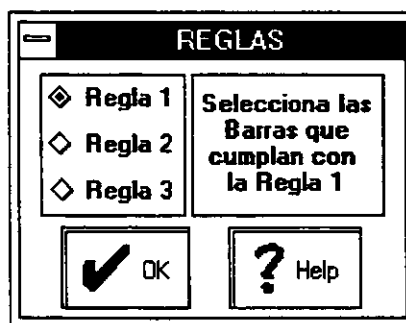
Existen estructuras cuya geometría es simétrica, esto es, que basta resolver la mitad de la estructura y la otra mitad reflejará los mismos valores. *Si el usuario detecta que la estructura es simétrica, deberá indicarlo por medio de este botón antes de comenzar los cálculos.* A continuación puede comenzar a resolver la estructura. Si resuelve una variable que tenga un valor *espejo*, el programa resolverá automáticamente el otro valor, con el objeto de aprovechar la simetría de la estructura.

Botón de reglas de simplificación



Cuando el usuario considere que se deben utilizar *reglas de simplificación* deberá presionar este botón, y aparecerá una caja de diálogo de reglas de simplificación ver la Figura VI. 3. Aquí se le pide al usuario que seleccione la regla de simplificación que desee utilizar. Una vez hecho esto podrá comenzar a marcar con el ratón las barras o reacciones que considere que cumplen con la regla mencionada.

Figura VI. 3



Botón de transmisión de fuerzas por equilibrio



Existen casos en los que *se puede transmitir una fuerza por equilibrio*; para evitar hacer la sumatoria de fuerzas sobre un nodo. Para aprovechar esta característica se hace uso del *botón*

de *paso por equilibrio*, normalmente se encuentra deshabilitado. Para habilitar el botón hay que seleccionar con el ratón una de las barras que ya se encuentran resueltas, el valor de ésta será transmitido por equilibrio a otra u otras barras. La acción anterior *habilitará el botón de paso por equilibrio*. El sistema mandará un mensaje al usuario indicando que puede comenzar a seleccionar todas las barras que considere que son transmitidas por equilibrio a partir de ésta. Hay que considerar el orden en que es transmitida la fuerza. Para terminar las selecciones es necesario volver a presionar el *botón de Paso Por Equilibrio*; y así deshabilitar el modo.

Botón de Retroceso



Este botón sirve en los dos modos: de *Resolución* y *AutoResolución*, se usa para deshacer el último paso que se ha realizado.

Botón de Avance



Para avanzar un paso en la *AutoResolución*, el usuario deberá presionar *el botón de avance, sólo sirve para este modo*.

Botón de Salida



El usuario podrá salir en cualquier momento del programa con la tecla de salida. Después de ésta breve, pero necesaria explicación pasemos a mostrar los ejemplos de Makatsiná. Para profundizar en el uso del sistema consultar [LAUA00](2).

VI. 2 Ejemplo 1 en el modo de resolución

En la Figura VI.4 se encuentra una estructura donde elegimos el nodo 3 para comenzar el cálculo, cuyas características físicas aparecen detalladas en el triángulo de abajo del área correspondiente de la pantalla. Enseguida oprimimos el botón para la obtención de sumatoria de fuerzas en el eje X y colocamos la ecuación que la resuelve.

Una vez resuelta la ecuación el sistema pregunta por la interpretación del signo, para saber si trabaja a compresión o tensión. En este instante entran los subtutores a revisar el trabajo. En este caso la estructura es hiperestática, y no tiene solución por este método. Cuando se detecta el error se lleva al usuario al micromundo correspondiente para repasar los conceptos y realizar un test, antes de volver a intentarlo, lo anterior se observa en la Figura VI. 5, la Figura VI.6, la Figura VI.7 y la Figura VI.8.

Si vuelve a cometer el mismo error Makatsiná le dará la solución. En este caso la estructura es hiperestática. Manda el mensaje que se observa en la Figura VI.9. Al final le diría que tiene que repasar los conceptos de hiperestaticidad e isostaticidad con lo que terminaría la interacción referente a ese ejemplo.

Figura VI. 4

Makatsiná - Resolución

Hoja de Trabajo

NODO 3:
 $\Sigma F_x = 0$
 $f_4 = w_3$
 $F_4 - 10 = 0$

INTERPRETACIÓN DEL SIGNO

Variable: **F4** Valor: **10.00**

Signo:

- Tensión
- Compresión
- Cero

OK Help

Diagrama de la estructura:

- Dimensiones: $A_3 = 4.00m$, $A_6 = 4.00m$, $A_5 = 3.00m$
- Reacciones: R_1 , R_2 , R_3
- Dimensiones de los triángulos: $A_5 = 4.00m$, $A_5 = 3.00m$

Figura VI. 5

TUTOR

Sabías que las "condiciones generales de isostaticidad" son:

- $NE > NI \Rightarrow$ Hiperestaticidad
- $NE = NI \Rightarrow$ Isostaticidad
- $NE < NI \Rightarrow$ Hipostática

donde:

- NE: Número de Ecuaciones
- NI: Número de Incógnitas

Estas, relacionadas a una armadura se convierten en:

$$2NN = (NB + NR)$$

donde

- 2NN representa a NE de las fórmulas generales
- NN: Número de Nodos
- NB: Número de Barras
- NR: Número de Reacciones

Figura VI. 6

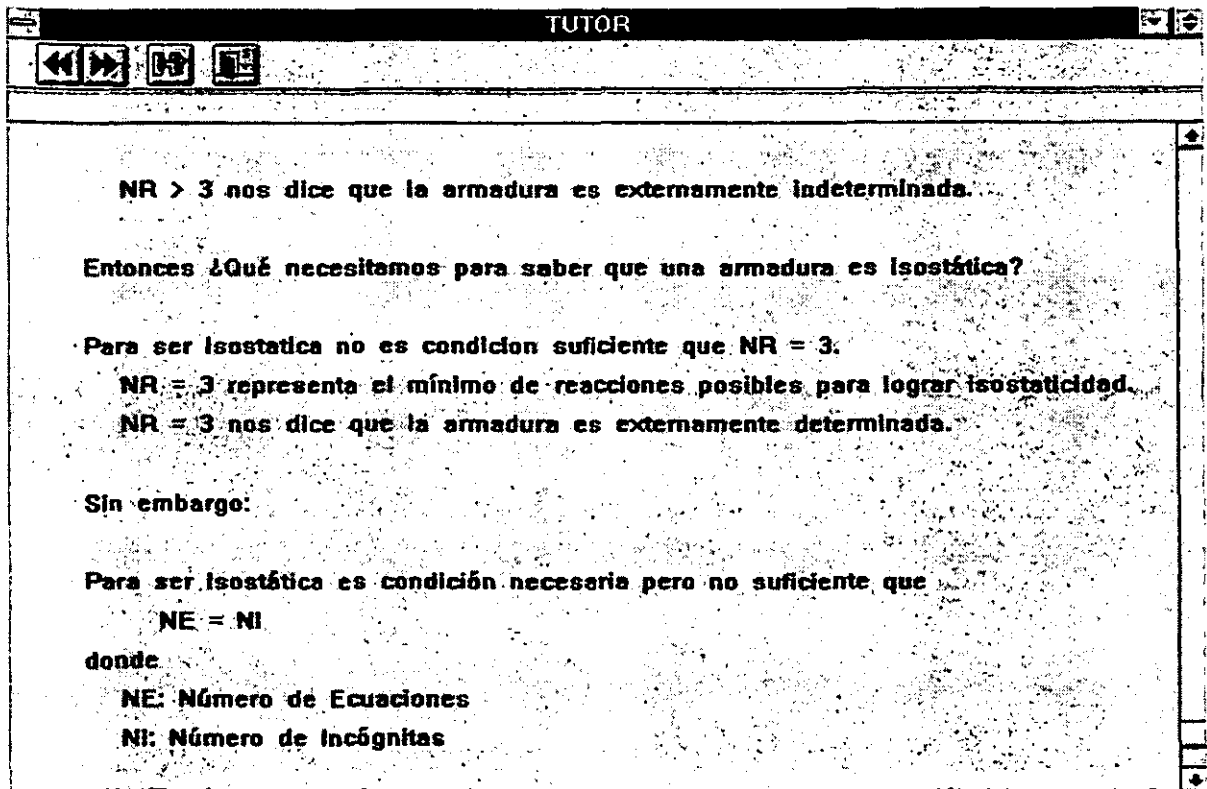


Figura VI. 7

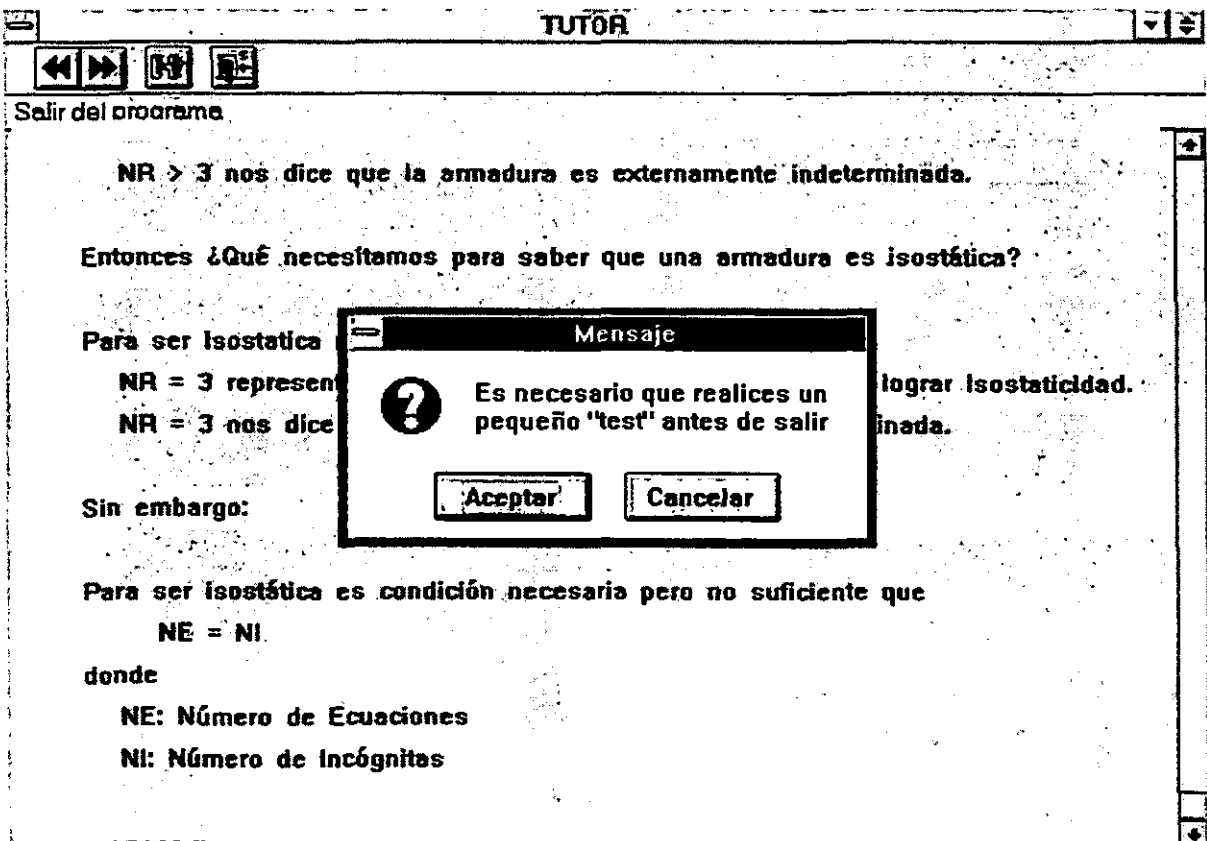


Figura VI. 8

TUTOR

Determine la isostaticidad y estabilidad de las siguientes armaduras:

Isostática, estable y externamente determinada
 Inestable
 Hiperestática, estable y externamente determinada
 Isostática, estable y externamente indeterminada
 Ninguna

"YA"

Figura VI. 9

Makatsiná - Resolución

Hoja de Trabajo

NODO 3:

$\sum F_x = 0$
 $f_4 = w_3$
 $F_4 - 10 = 0$
 $F_4 = 10.00$ COMPRESION

Error

La armadura en curso
ES HIPERESTÁTICA

Aceptar

VI. 3 Ejemplo 2 en el modo de resolución

En la Figura VI.10 se observa una estructura, ésta es muy parecida a la anterior, sólo cambia un tipo de apoyo con lo que se convierte en estática y estable, y en consecuencia podemos aplicar el método. Vale la pena comentar que muchas de las estructuras se encuentran trampeadas a propósito. Con el fin de que los estudiantes puedan captar los detalles que existen debajo de las características superficiales; y así poder utilizar con éxito el método.

Figura VI. 10

Hay que hacer notar que en ambos ejemplos se ha planteado mal la sumatoria de fuerzas en el eje X, no obstante en el primer caso el error de tratar de aplicar el método a una estructura hiperestática tiene una jerarquía superior. Así que al entrar a revisar los subtutores, el subtutor -1- tiene prioridad para marcar primero el error (Sección III.5.6.1.1). En el segundo caso antes de empezar a resolver las barras planteando ecuaciones existe la posibilidad de utilizar las reglas de simplificación lo que constituye un gran ahorro en el procedimiento, a menos, que no se conozcan. Lo anterior implica que se podría resolver toda la estructura a través del planteamiento de ecuaciones. Sin embargo, no hay que olvidar que *Makatsina* está reforzando el aprendizaje de una estrategia de solución vista en clase; es ahí donde se enseña el uso de las reglas de simplificación.

Se repite el procedimiento y en el instante en que los subtutores entran a revisar, descubren que no han utilizado las reglas de simplificación. Hay casos en los que éstas no son aplicables, pero éste no lo es; así que le avisa al alumno y lo lleva al Micromundo del subtutor -2-, como se observa en la Figura VI. 11.

En los subtutores se utilizó hipertexto, con lo que el cambio de color en el texto implica que se puede profundizar en algún tema, tal es el caso de los enunciados de las tres reglas, esto se observa en la Figura VI.12 y Figura VI.13.

Figura VI. 11

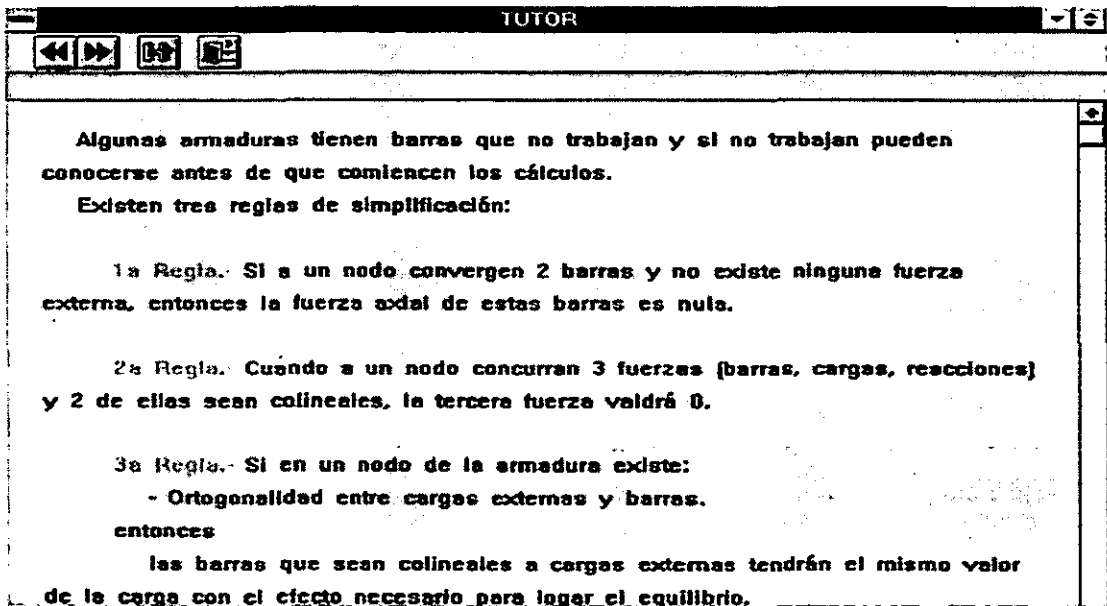


Figura VI. 12

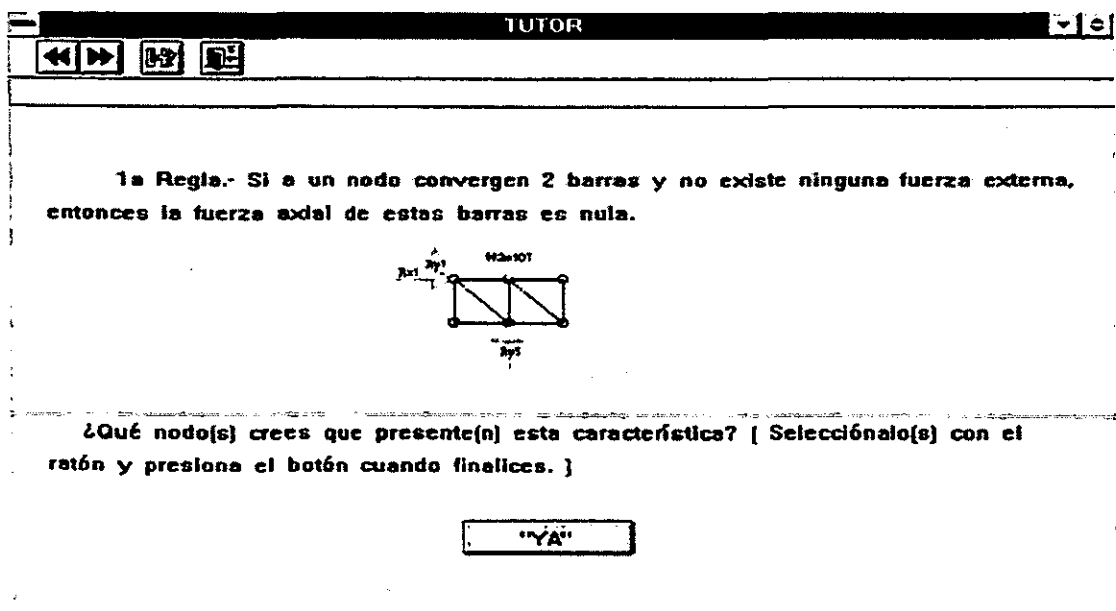
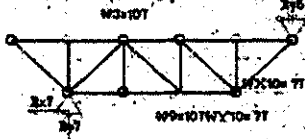


Figura VI. 13

TUTOR

2a Regla.- Cuando a un nodo concurren 3 fuerzas (barras, cargas, reacciones) y 2 de ellas sean colineales, la tercera fuerza valdrá 0.



¿Qué nodo(s) crees que presente(n) esta característica? (Seleccionalo(s) con el ratón y presiona el botón cuando finalices.)

"YA"

Es necesario hacer notar que en cada una de estas explicaciones y preguntas aparecerá una estructura diferente. El tutor sabe cuales son las estructuras ad-hoc para cada subtutor; de aquí la capacidad de la elección idónea.

Figura VI. 14

TUTOR

Salir del programa

Algunas armaduras tienen barras que no trabajan y si no trabajan pueden conocerse antes de que comiencen los cálculos.

Existen tres reglas de simplificación:

1a Regla.- Si a un nodo concurren 3 fuerzas (barras, cargas, reacciones) y ninguna fuerza externa, entonces la fuerza que no trabaja valdrá 0.

2a Regla.- Cuando a un nodo concurren 3 fuerzas (barras, cargas, reacciones) y 2 de ellas sean colineales, la tercera fuerza valdrá 0.

3a Regla.- Si en un nodo de la armadura existe:

- Ortogonalidad entre cargas externas y barras.

entonces las barras que sean colineales a cargas externas tendrán el mismo valor de la carga con el efecto necesario para lograr el equilibrio.

Mensaje

? Es necesario que realices un pequeño "test" antes de salir

Antes de salir de cada subtutor es necesario que el usuario realice un pequeño test, ver Figura VI.14, éste sirve para el diagnóstico final, no le dice la calificación, ésta no es importante, pero sí le dice cuales han sido sus fallas. Internamente el tutor cuenta con el término aprobado o reprobado, y éste se utiliza en el diagnóstico final. Cada uno de los tests fué elaborado por el experto; utilizando preguntas capciosas y con trampa. Se escogen estructuras triangulares del stock general de forma aleatoria y las características de la estructura por las que se pregunta en los tests no siempre son las mismas, esto es, el número de barras de rojo, verde o amarillo se eligen en base a números aleatorios. Por lo tanto, la selección no se realiza siguiendo algún patrón. Lo que significa que podríamos estar hablando de barras rojas, verdes o amarillas que no pudieran ser simplificadas por ninguna de las reglas. En la Figura VI.15 se observa un ejemplo.

Figura VI. 15

TUTOR

⏪ ⏩ 🏠 🖨
⏪ ⏩

Contesta a las siguientes preguntas:

a) La barra ROJA se simplifica por: Regla 1 Regla 2 Regla 3

b) La barra VERDE se simplifica por: Regla 1 Regla 2 Regla 3

c) La barra AMARILLA se simplifica por: Regla 1 Regla 2 Regla 3

"YA"

Como ya se mencionó Makatsiná reforzará el aprendizaje de una técnica vista en clase. La evaluación final consistirá en un diagnóstico; por lo que su interés radica en que el estudiante avance mientras no cometa errores de aborto, ya sea porque conoce el método o porque Makatsiná lo va guiando. El último caso se presenta cuando el estudiante es reincidente en ciertos errores. Por ejemplo si una vez visitado el micromundo; no vuelve a utilizar las reglas de simplificación, aparecería la pantalla de la Figura VI.16 y se guiaría el uso de éstas como se observa en la Figura VI.17 y la Figura VI.18.

Finalmente una vez aprendidos los conocimientos del subtutor de reglas de simplificación continuamos el camino, vigilados siempre por los tres subtutores. Sin embargo en la última etapa de la resolución de la estructura sólo se podrán cometer errores pertenecientes al subtutor -3-; debido a que se ha avanzado en el proceso de resolución y este se encuentra en la etapa final y debido también a los mecanismos de arbitraje (Sección III.5.6.1.1). En esta etapa de solución de estructuras una de las tácticas es retrasar la obtención de las reacciones por sumatoria de momentos lo más posible, hasta que sea imposible continuar sin el conocimiento de las magnitudes y direcciones de estas fuerzas, ésto lo observamos en la Figura VI.19 y la Figura VI.20.

Figura VI. 16

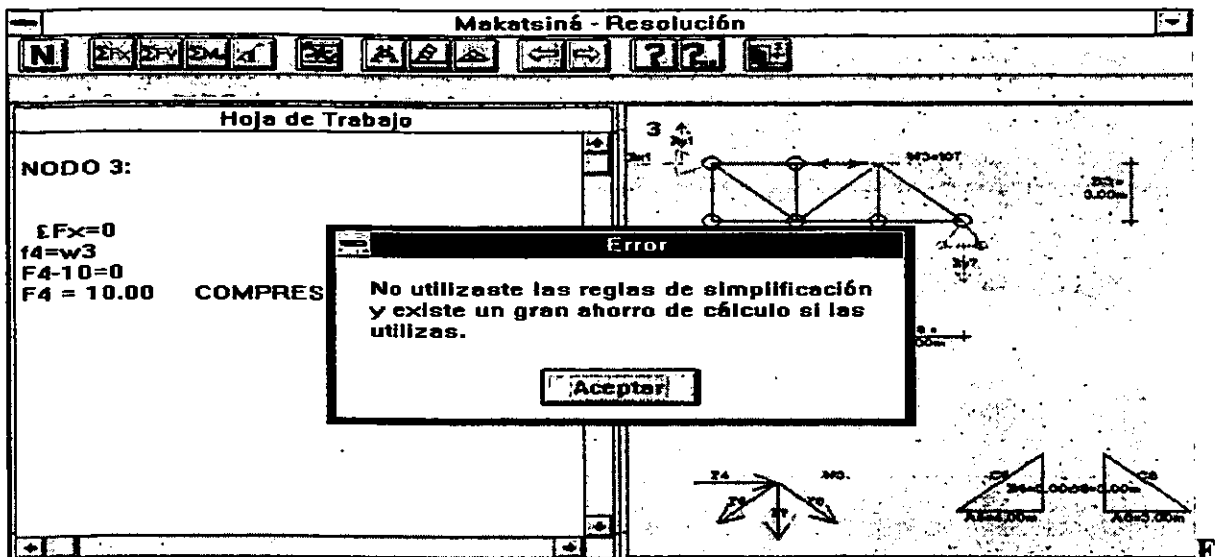


Figura VI. 17

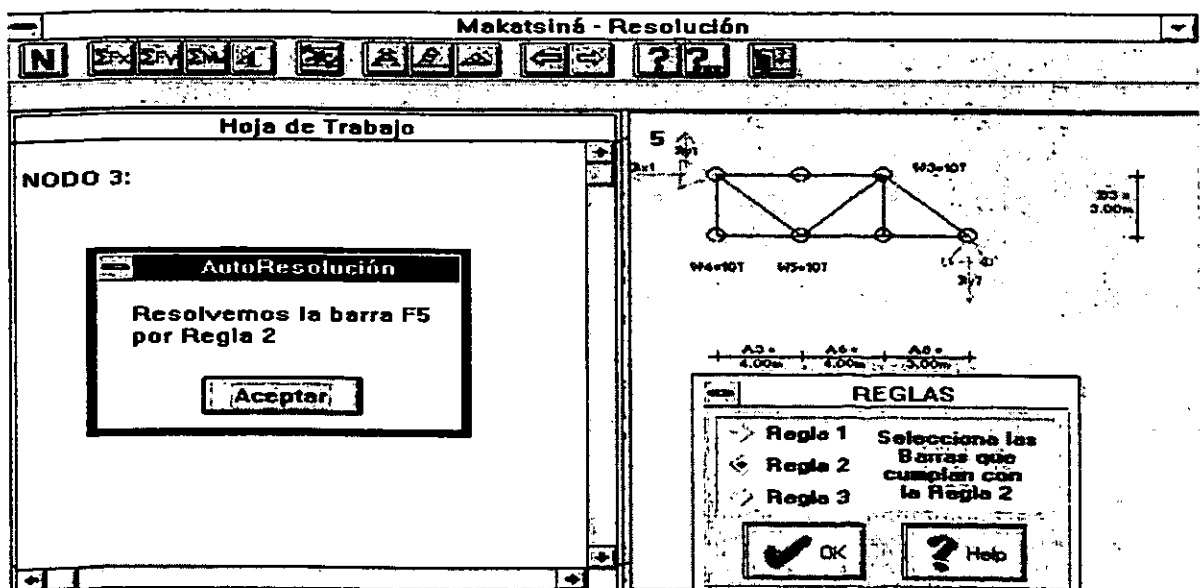


Figura VI. 18

Makatsiná - Resolución

Hoja de Trabajo

NODO 3:

$f4=w3$

Error

Falta simplificar barras por regla 2:

"Cuando a un nodo concurren 3 fuerzas (barras, cargas, reacciones), y 2 de ellas sean colineales, la 3a fuerza valdrá cero"

Aceptar

$A3 = 4.00m$ $A6 = 4.00m$ $A8 = 3.00m$
 $B3 = 3.00m$

Figura VI. 19

Makatsiná - Resolución

Hoja de Trabajo

Σ Por tercera regla de simplificación:

$+F2-W4=0$
 $+F2-10=0$
 $F2 = 10.00$ TENSION

NODO 4:

$ΣFy=0$
 $f3=w4$
 $F3-10=0$
 $F3 = 10.00$ COMPRESION

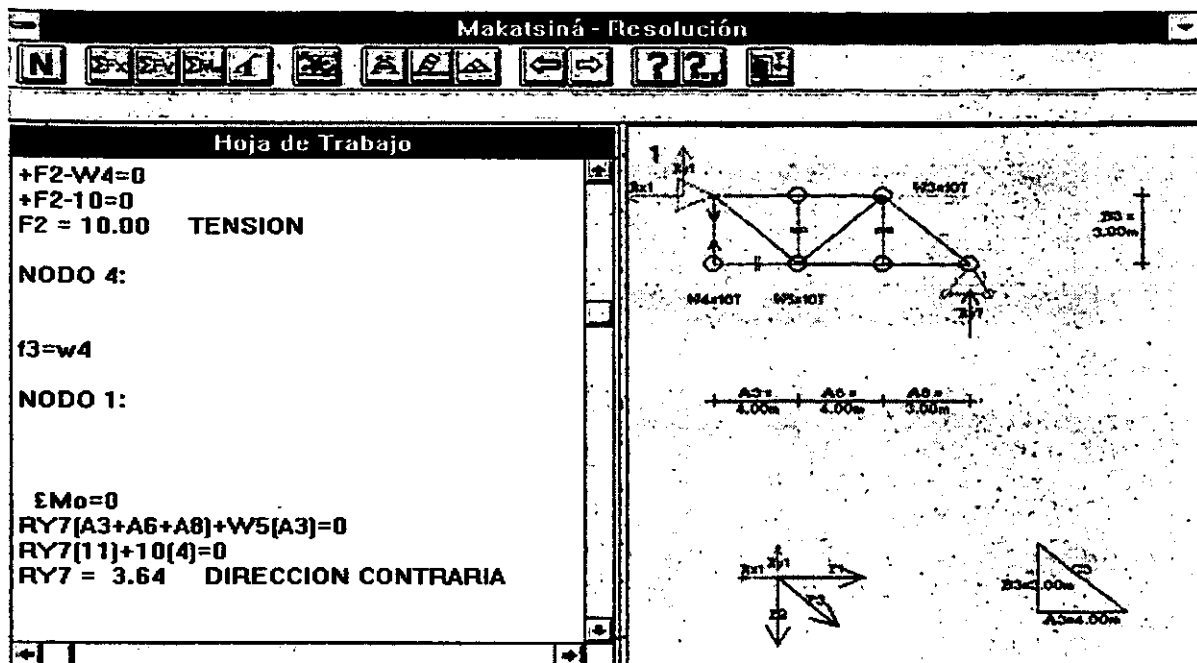
$A3 = 4.00m$ $A6 = 4.00m$ $A8 = 3.00m$
 $B3 = 3.00m$

Error

Necesitas realizar sumatoria de momentos para continuar.

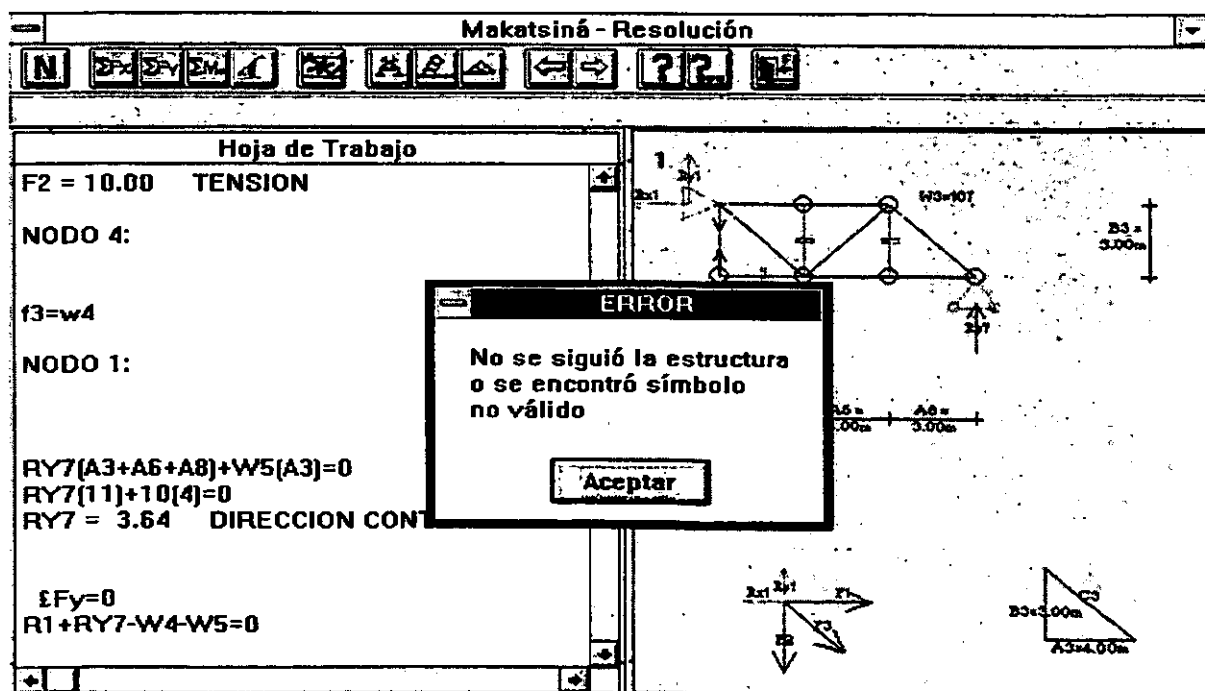
Aceptar

Figura VI. 20



Durante el resto del desarrollo se estarán revisando los errores pertenecientes al subtutor de obtención de fuerzas, aunque en realidad el desarrollo esta bajo la revisión constante de los tres. Dentro de esta etapa también existen errores superficiales como los de sustitución incorrecta en la ecuación, o signos impares como el caso de la Figura VI.21; donde el error que se ha cometido es escribir R1 en lugar de RY1 en la Sumatoria de fuerzas en el eje Y.

Figura VI. 21



El estudiante continua el desarrollo del ejercicio, esto se observa en la Figura VI.22, la Figura VI.23, la Figura VI.24 y la Figura VI.25, para finalmente terminar la sesión.

Figura VI. 22

Makatsiná - Resolución

Hoja de Trabajo

NODO 1:
 $R_{Y7}(A_3+A_6+A_8)+W_5(A_3)=0$
 $R_{Y7}(11)+10(4)=0$
 $R_{Y7} = 3.64$ DIRECCION CONTRARIA

NODO 7:
 $\sum F_x=0$
 $-F_{11}+F_8(A_8/C_8)=0$
 $-F_{11}+5.139(3/4.24)=0$
 $F_{11} = 3.64$ TENSION

Figura VI. 23

Makatsiná - Resolución

Transmisión de Fuerzas Por Equilibrio

Hoja de Trabajo

$R_{Y7} = 3.64$ DIRECCION CONTRARIA

NODO 7:
 $-F_{11}+F_8(A_8/C_8)=0$
 $-F_{11}+5.139(3/4.24)=0$
 $F_{11} = 3.64$ TENSION

FOR EQUILIBRIO

Selecciona la(s) barra(s) a la(s)cual(es) le es transmitida ésta.

Aceptar

Figura VI. 24

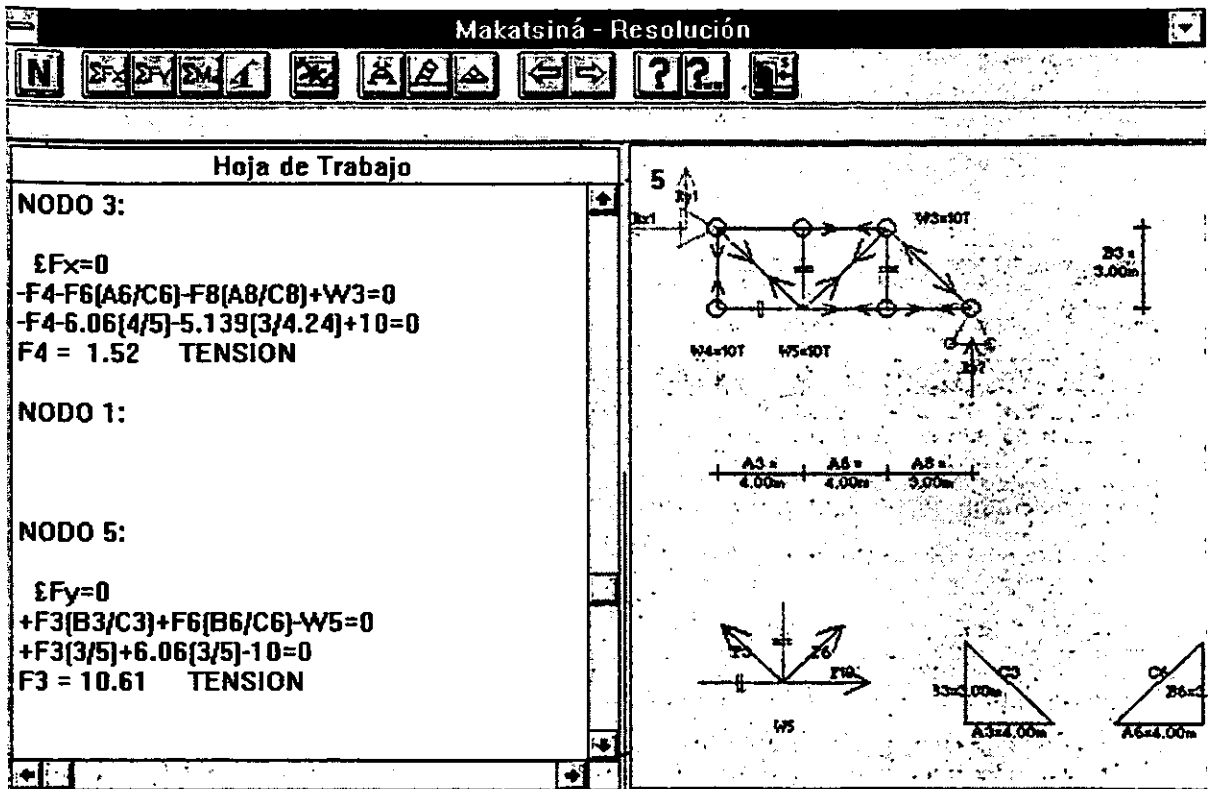
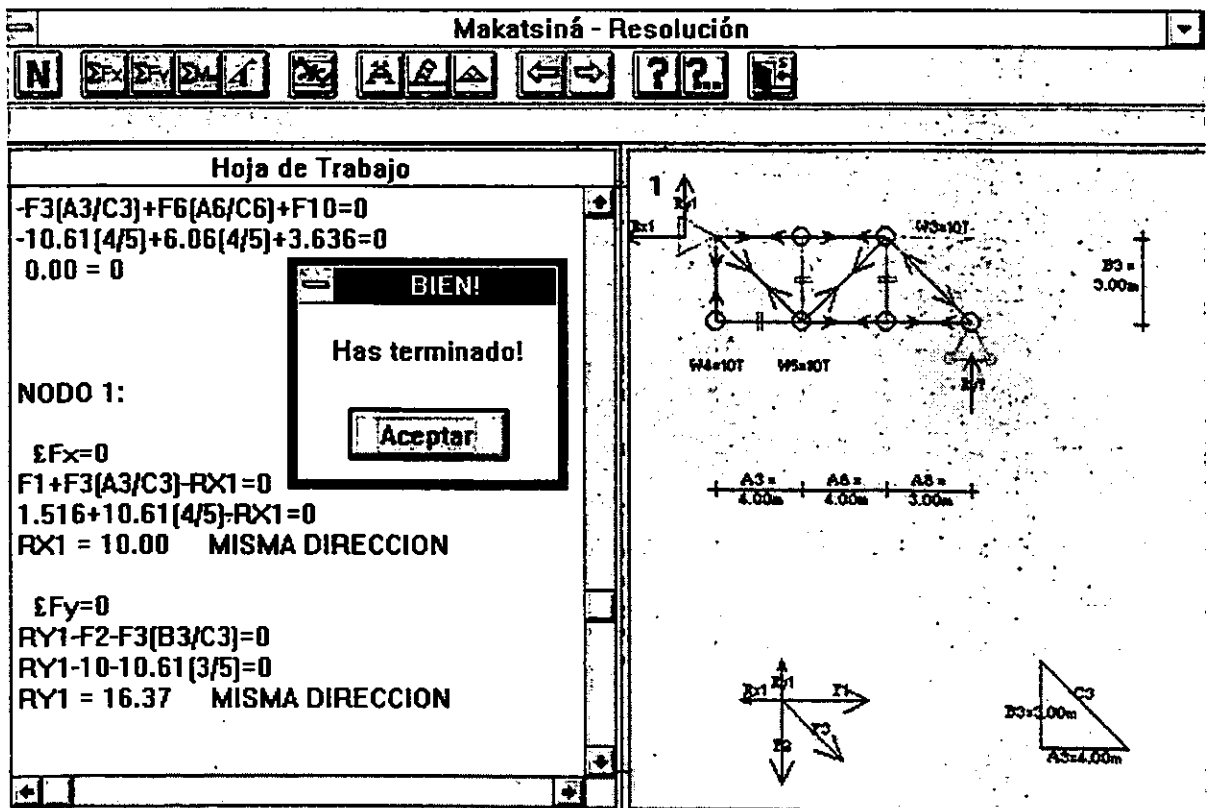


Figura VI. 25



En la Figura VI.26 y la Figura VI.27 se muestra el resultado del diagnóstico final, para que el estudiante sepa que es lo que le falla y debe estudiar, cabe mencionar que se cometieron todos los errores posibles, de aquí el resultado nada halagador, pero en general un estudiante con tantos errores no podría llegar a un feliz termino.

Figura VI. 26

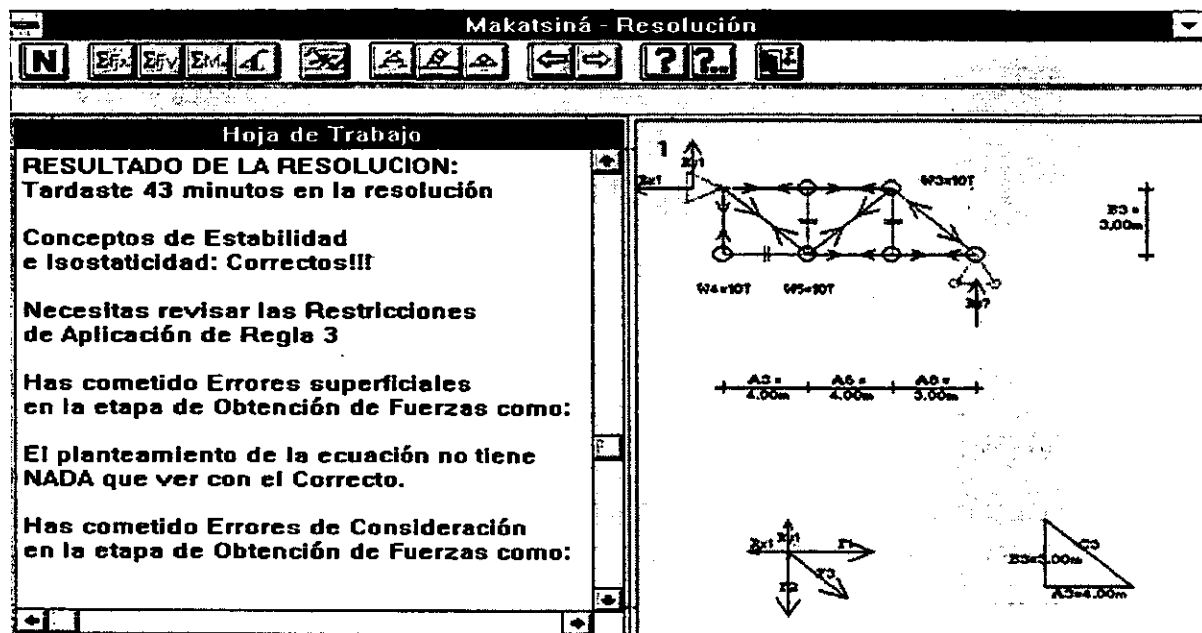
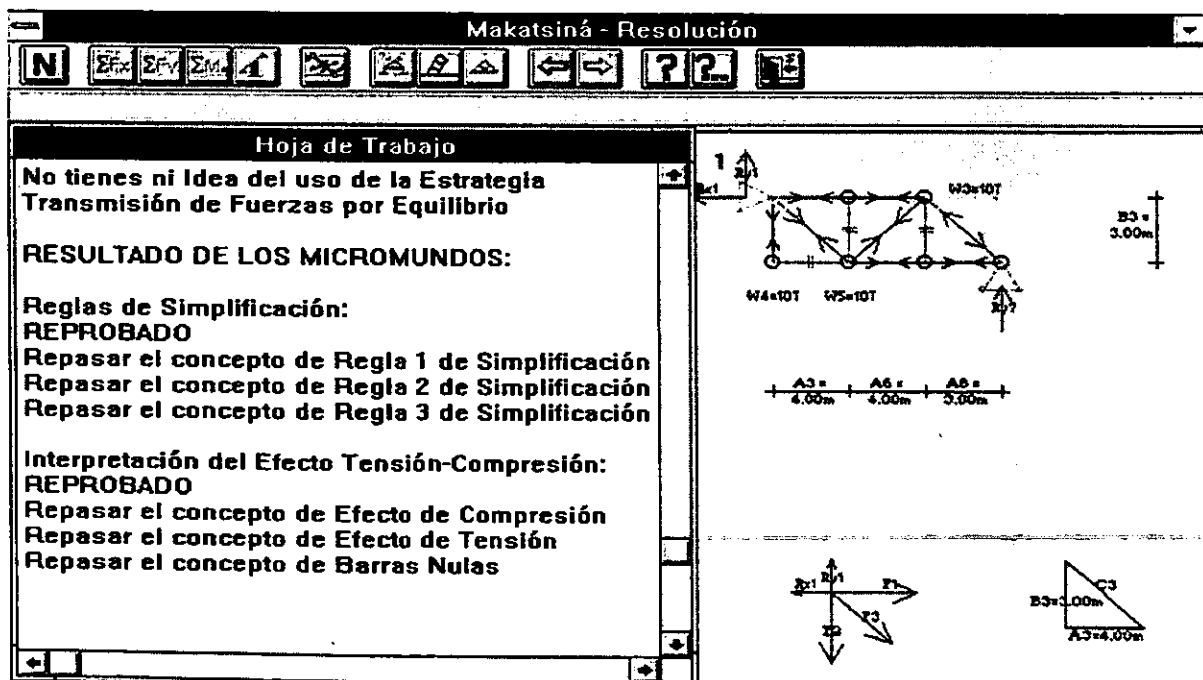


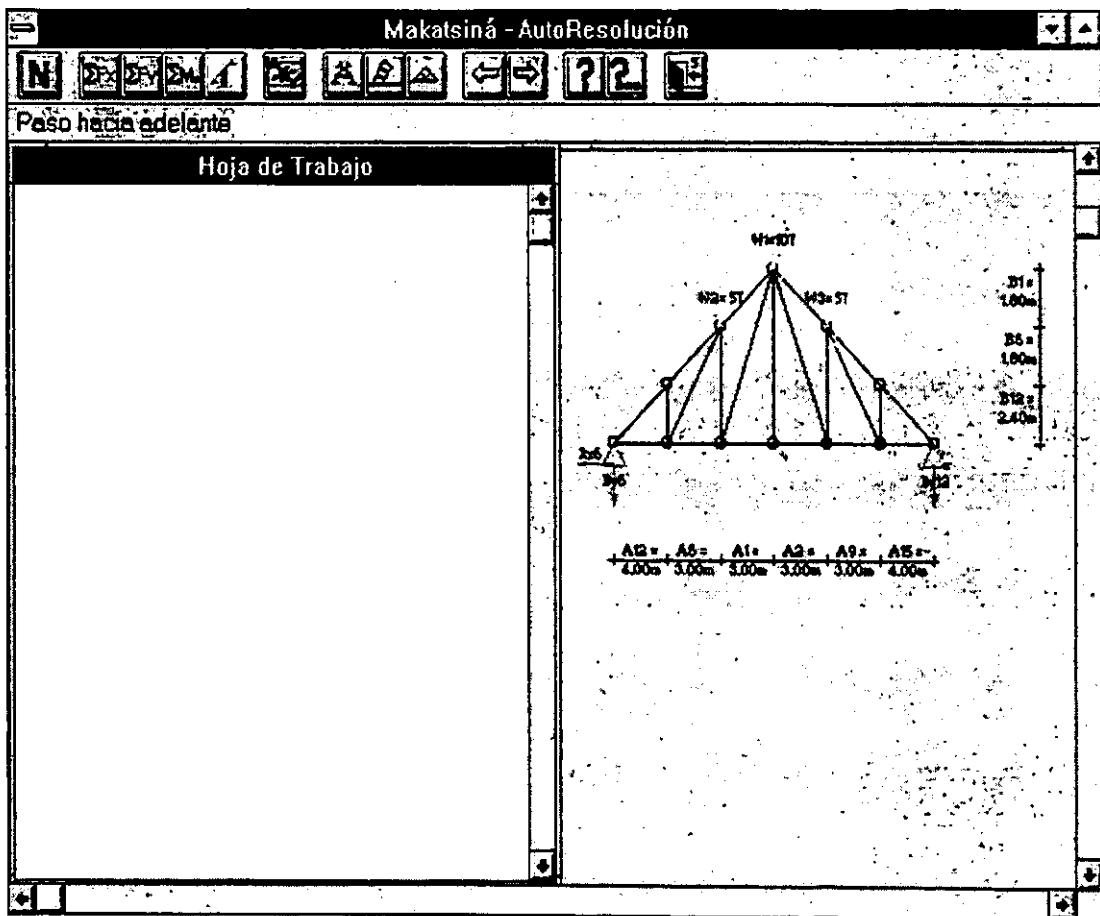
Figura VI. 27



VI. 4 Ejemplo 3 en el modo de autoresolución

En el modo de *AutoResolución* se mostrará un ejemplo de estructura triangular cuya geometría es simétrica, esto es, basta resolver la mitad de la estructura y la otra mitad reflejará los mismos valores. Si el usuario detecta que la estructura es simétrica, deberá indicarlo por medio de un botón antes de comenzar los cálculos. A continuación puede comenzar a resolverla; cada vez que resuelva una variable, el programa resolverá automáticamente el otro valor *espejo*. De esta forma se aprovecha la simetría de la estructura, en la Figura VI.28 y la Figura VI.29 se observa el inicio de este modo con una estructura simétrica.

Figura VI. 28



En la Figura VI.30 y la Figura VI.31 se muestran pantallas pertenecientes a la secuencia de resolución de este ejemplo y en la Figura VI.32 y la Figura VI.33 se muestra la resolución finalizada.

Figura VI. 29

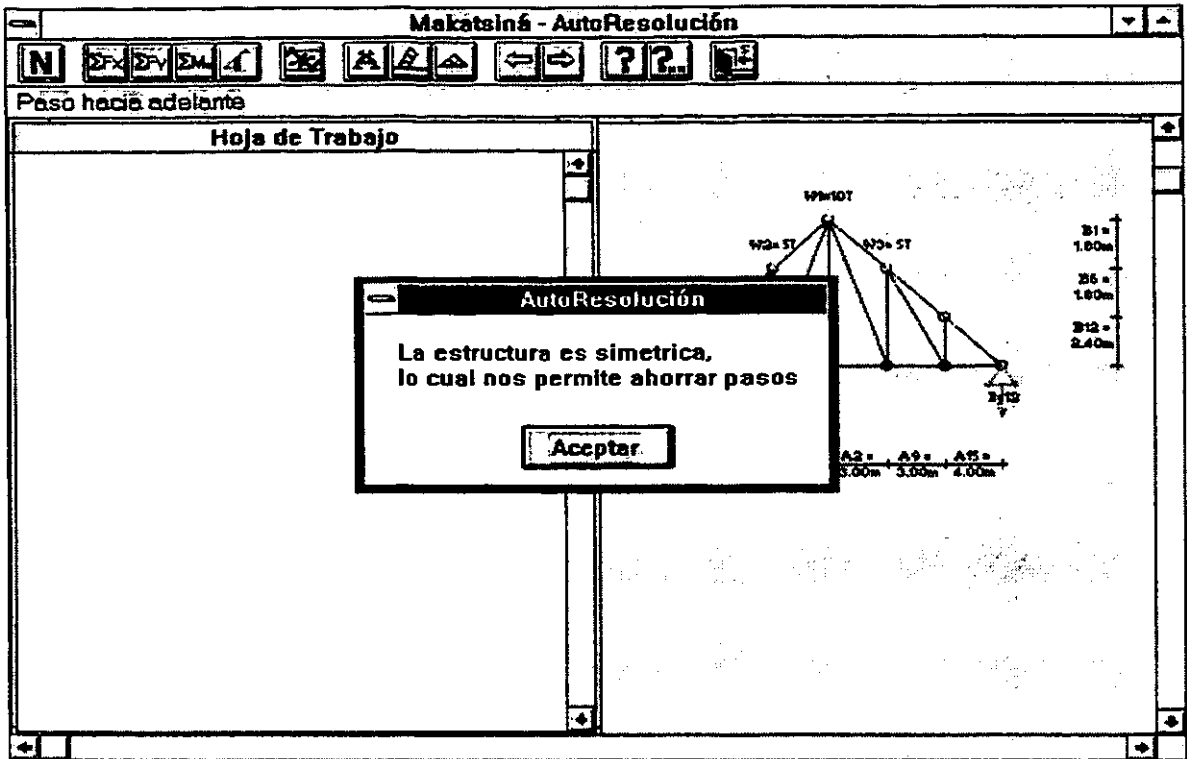


Figura VI. 30

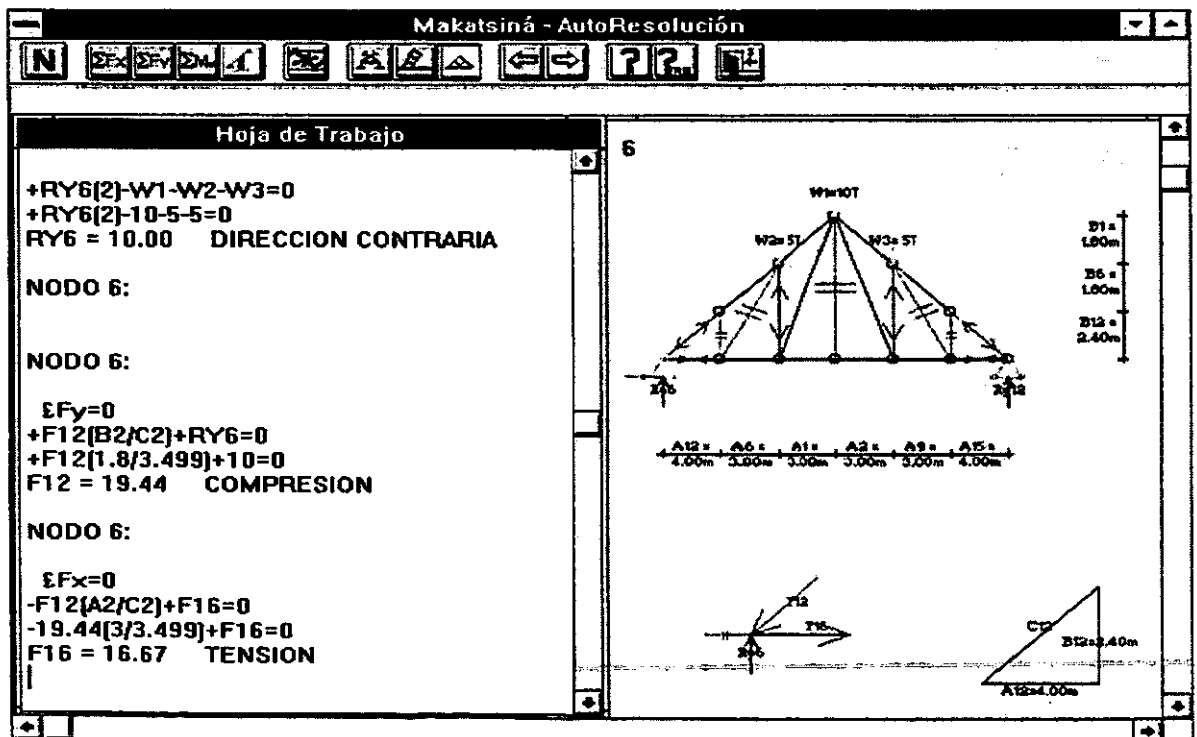
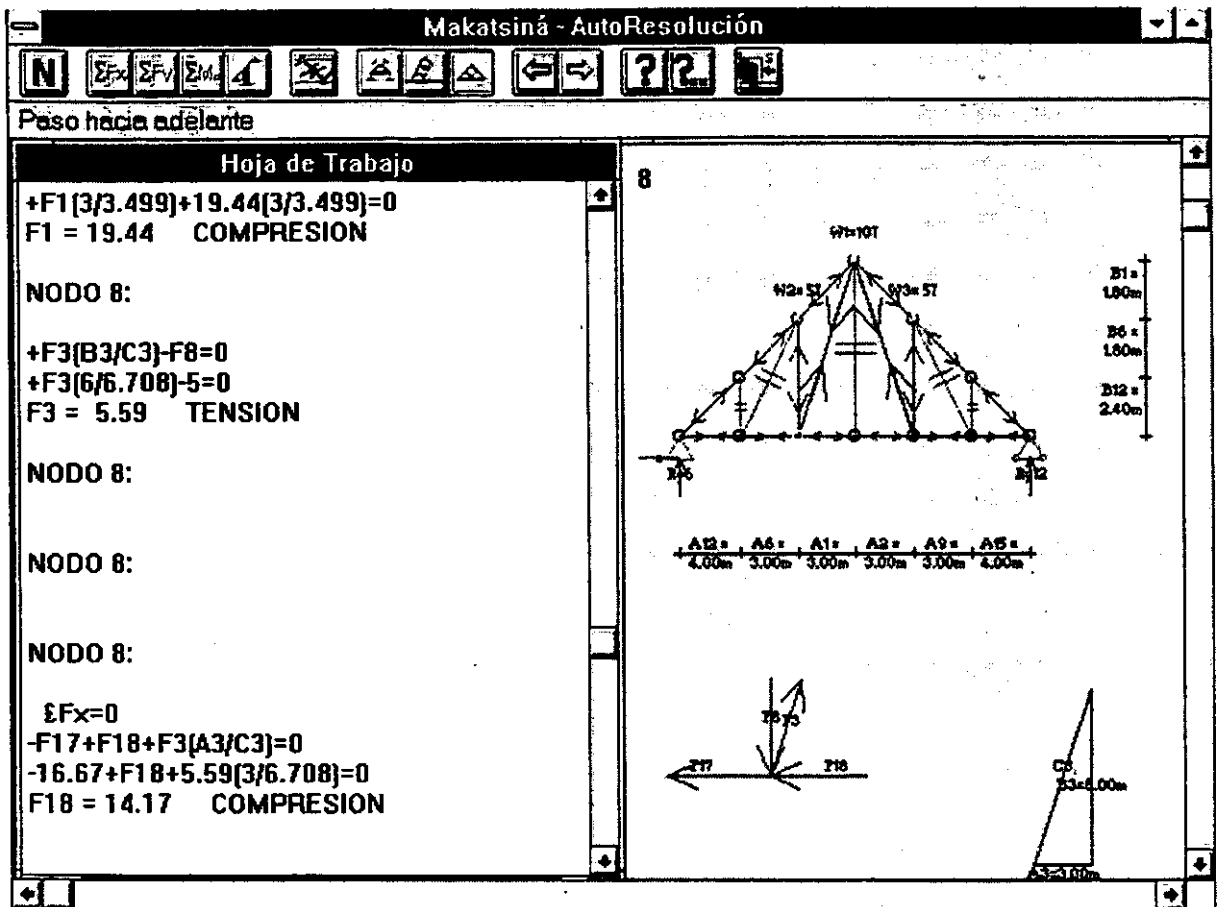


Figura VI. 33



VI. 5 Programación de Makatsiná

La programación está realizada en su totalidad sobre Borland C++ 4.5, utilizando OWL 2.5. Se realizó un *análisis y diseño basado en los paradigmas basado en objetos, orientado a objetos y tipos abstractos de datos*, todos estos se tratan con profundidad en Laureano y Ortiz [LAUA96](1) y se utilizó la biblioteca de clases Object Windows Library (OWL), [BORL94] cuyo objetivo es facilitar la programación sobre ambiente Windows. Los diagramas usados durante el análisis corresponden a la notación del método de Booch descrita en Booch [BOOG91] y Oktaba [OKTH93](1).

Todo lo referente a la programación de Makatsiná se detalla ampliamente en el Reporte de Investigación *Paquete computacional Makatsiná sistema tutorial inteligente para la resolución de estructuras triangulares por el método de los nodos*; cuya referencia es [LAUA00](2).

Capítulo VII

Conclusiones y trabajos futuros

VII. 1 Resumen

En este trabajo se han descrito diferentes aspectos asociados con el desarrollo de Makatsiná.

Una de las principales motivaciones era poder lograr de forma efectiva la enseñanza de una sola estrategia a través de la descomposición de habilidades. Con el fin de lograr más detalle tanto en las técnicas y estrategias tutoriales, como en el diagnóstico de errores. Lo anterior se logró aplicando la filosofía reactiva a los SEIs, no aplicada hasta el momento. Se consiguieron las siguientes características funcionales en el sistema de enseñanza Makatsiná, mismas que se resumen en los siguientes puntos:

- Cuenta con expertos en habilidades que son parte de una de mayor complejidad.
- Logra un mayor detalle en el manejo de errores.
- La enseñanza de la habilidad integrada queda inmersa en el ciclo de interacción estudiante-enseñanza. Lo que permite lograr la reacción a la falta o mal uso de cualquiera de las subhabilidades.
- El uso de los agentes reactivos permite una mayor velocidad de respuesta; con lo que se logra un ahorro en el tiempo de diagnóstico de errores.

- La ventaja de crecimiento en base a agentes; donde estos últimos pueden necesitar todos o algunos de los agentes de Makatsiná, para implementar la enseñanza de otra habilidad integrada.

VII. 2 Resultados y Aportaciones

Dentro de las principales aportaciones de este trabajo está la instrumentación lograda durante el análisis y diseño para implementar una arquitectura MultiAgente en un SEI, donde, además estos agentes tienen como característica principal, ser reactivos. Los puntos claves de la aportación son:

- La TC de análisis de estructuras triangulares por el método de los nodos, hasta el momento no había sido analizada con técnicas de modelos mentales y de ACT.
- La integración durante el análisis y diseño del dominio de dos estilos de ACT con uno de los usos de la GG, así como el modelo mental de la TC. Para encontrar el número y las habilidades subyacentes que integran a la habilidad en cuestión.
- Se usan los errores como características del entorno, capaces de hacer reaccionar a los agentes.
- Se amplió la taxonomía de habilidades propuesta por Ryder & Redding [RYDJ93] de acuerdo a las características de nuestro problema.
- Se creó una nueva arquitectura de SEI; donde los elementos de éste quedan inmersos en el diseño de los agentes, dentro de la arquitectura MultiAgente.
- A través de la experiencia de Makatsiná se concluye que un mecanismo de interacción sencillo en un SEI puede dar resultados interesantes.
- Se usa el ACT no sólo para esclarecer el proceso de enseñanza-aprendizaje, sino como un instrumento para lograr el acercamiento hacia una arquitectura reactiva.
- Se muestran los beneficios del ACT dentro del campo de análisis y diseño de agentes reactivos.
- Se propone un mecanismo de arbitraje para sistemas de enseñanza inteligentes reactivos. Basada en prioridades. Estas últimas pueden establecerse por la importancia de los errores durante el proceso de diagnóstico de los mismos; dentro del contexto global de solución del problema, o por el orden de enseñanza de las subhabilidades que involucran habilidades de mayor complejidad; en cuyo caso la jerarquía de la importancia de los errores quedará supeditada al orden de aprendizaje de las subhabilidades.

- Se propone una metodología para el análisis y diseño de agentes reactivos basándonos en un modelo cognitivo.
- Se propone una metodología para formalizar agentes; como consecuencia de haber utilizado el formalismo de agentes propuesto por: Luck y d'Inverno [LUCM95] y una forma de organización de agentes propuesta por Nigay y Coutaz [NIGL91]. Con lo que se garantiza completitud y corrección del sistema.

[ALOC96] recomienda que una evaluación imprescindible a un software educativo es la realizada por expertos en el área, más que la realizada en los grupos formados por los estudiantes, sin embargo, ésta última no se menosprecia, pueden ser complementarias, por lo que se ha reservado para trabajos futuros.

Makatsiná ha sido utilizado durante dos semanas por un grupo formado por tres expertos en análisis estructural, quienes además también son profesores, ellos lo han probado de forma independiente con el propósito de revisar el conocimiento incluido en el método de solución, además de probar las técnicas utilizadas por los MicroMundos, una vez detectados los errores. Posteriormente los expertos se reunieron para discutir sus resultados; finalmente llenaron un cuestionario de forma separada con preguntas concernientes a detalles en la operación del sistema. El resultado es que la calidad del conocimiento incluido en el método de los nodos es bueno, los errores significativos han sido incluidos en el sistema y las tácticas incluidas para su corrección son apropiadas. El sistema es interactivo, bueno y adecuado para el uso de los estudiantes.

VII. 3 Trabajos Futuros

VII. 3. 1 La robustez conductual y perceptual de Makatsiná

En Makatsiná no se da ningún tipo. En cuanto a la perceptual, no contamos con muchos sensores extrayendo el mismo tipo de información y no tenemos una división física de los subtutores, de aquí, que si falla uno en realidad supondría que fallaría todo el PC y por lo tanto, todos los subtutores.

Podríamos pensar en una división física al pensar que los subtutores estuvieran ubicados en 'Transputer' [BANJ93], [GARD90] y [GARD92]. Esto supondría un paso para lograr las dos robusteces, ya que podríamos tener a varios subtutores revisando los mismos errores.

Con respecto a la robustez conductual la podríamos lograr al tener varios agentes trabajando un mismo aspecto del problema, con el mismo método o con métodos diferentes; además en este tipo de sistemas se podría con esta duplicación subsanar incertidumbres en la

interpretación de los errores. Otra forma sería creando una granularidad mas fina en la revisión de un mismo error, cuando éstos pertenecieran a la clase de graves.

Otra forma sería crear agentes tipo 'default', que entrarán en acción en el instante en que fallen algunos o por tiempo o generalizando los errores para que pudiera actuar con varios tipos de éstos.

VII. 3. 2 Diagnóstico de errores

Ya se mencionó con anterioridad que el manejo de errores en un detalle fino, es un camino hacia la robustez conductual. Este camino además permite una mejor adaptación del sistema al entorno (estudiante). En Makatsiná se propone desarrollar un módulo de diagnóstico de errores más fino, respetando la arquitectura actual. Con el fin de profundizar el detalle del error y lograr con ello una mejor adaptación del sistema al estudiante.

VII. 3. 3 El impacto pedagógico de Makatsiná

En la carrera de Ingeniería Civil, las estructuras son a todas las ramas posibles de especialidad lo que el ballet clásico es a cualquier tipo de baile, esto es, el que sabe estructuras lo único que tiene que hacer es pegar la especialidad que desee y será bueno, por esta razón más del 60% de la carrera son estructuras, de aquí que nos interese conocer el impacto de este SEI en la capacidad de los estudiantes, es decir, mejora o no su capacidad de desarrollo de forma global.

Se realizarán evaluaciones de tipo pedagógico, donde, lo que nos interesará será conocer:

- La utilidad en la materia para la que fue creado de exprofeso, para lo cual se evaluará su uso en varios grupos piloto.
- La utilidad en cuanto a la capacidad de aprendizaje de otro tipo de dominio que no pertenezca al área de análisis y diseño de estructuras, esto es, *¿ enseña a pensar como ingeniero ?*.

VII. 3. 4 La generalización de la técnica propuesta de análisis y diseño del SEI MultiAgente

Se mencionó que una de las principales aportaciones de este trabajo consiste en la propuesta de la instrumentación lograda durante el análisis y diseño para implementar una arquitectura MultiAgente en un SEI; donde además estos agentes tienen como característica principal ser reactivos.

Se está trabajando actualmente en la abstracción de esta técnica para su aplicación a diferentes sistemas inteligentes tomando como base a los SEIs y a los robots autónomos, en los trabajos de Laureano y García-Alegre ([LAUA97](2) y [LAUA98](5)) se han realizado la primeras aproximaciones y se reportan resultados de momento prometedores.

VII. 3. 5 La generalización de la arquitectura de software para construir un SEI MultiAgente

Makatsiná desde el punto de vista de software no fue construido pensando en que pudiera ser utilizado como un entorno para la generación de SEIs al que se le pudieran adaptar diferentes tipos de dominio y en consecuencia diferentes expertos compilados con sus diferentes tipos de errores; lo anterior queda claro cuando se hojea el Anexo Programación de Makatsiná, sin embargo, su arquitectura en el nivel de abstracción más general, podría ser utilizada para la implementación de un entorno para la generación de SEIs en el que se generalizarían: *la clase subtutor* y *la clase micromundo* como parámetro del anterior y se contaría con funciones genéricas que permitieran el paso de errores como parámetros, así mismo se pensaría en una base de datos que permitiría construir el archivo del experto compilado, con lo cual se podrían manejar distintos tipos de dominio; aún así se adaptaría sólo a cierto tipo de dominios que cumplieran con las restricciones siempre presentes en cualquier entorno para la generación de SEIs.

La experiencia de Makatsiná permite vislumbrar la posibilidad de poder trabajar en un ambiente genérico; el cuál nos proporcionaría un grado de amplitud en el uso de dominios. No hay que perder de vista que este tipo de sistemas generales repercute indiscutiblemente en la riqueza de detalles de los sistemas que son construidos de forma ad-hoc al dominio; como es el caso de Makatsiná donde se cuenta con detalles visuales de gráficas, colores y animación. Así como el proceso de diagnóstico de errores; donde éstos tienen que ser tratados de forma general. De aquí el uso bastante frecuente de los entornos para la generación de SEIs como prototipos más que como constructores de sistemas. En Arruarte [ARRA98] se aborda la construcción de un entorno para la generación de un SEI; donde se explica con detalle las características de estos sistemas, sus bondades y sus limitaciones.

REFERENCIAS

- [AGRP87] Agre, P, y D. Chapman. *Pengi: An Implementation of a Theory of Activity*. Proceedings of the Sixth National Conference on Artificial Intelligence. AAAI-87. Morgan Kaufmann, Los Altos, California, 1987.
- [ALOC94] Alonso, C. y D. Gallego. *Los Estilos de Aprendizaje*. Ediciones Mensajero. Bilbao, España 1994.
- [ALOC96] Alonso, C. (Doctora en Pedagogía), *Correspondencia Privada*. Universidad de Educación a Distancia (UNED). Madrid, España - septiembre de 1996.
- [ANDJ81] Anderson J., J.G. Greeno, P.J. Kline, & D. Neves. M. *Acquisition of Problem Solving Skills and their Acquisition*. (Ed.) J. R. Anderson. In Cognitive Skill and their acquisition, pp. 191-230, Hillsdale, N.J: Erlbaum, 1981.
- [ANDJ85] Anderson, J. R. and B.J. Reiser. *The Lisp Tutor* . BYTE, Vol.10, pp. 159-175, 1985.
- [ANDJ88] Anderson, J. *The Expert Module (Capítulo II)*. In Foundations of Intelligent Tutoring Systems. Eds. Martha C. Polson and J. Jeffrey Richardson. Lea eds. Hove & London 1988.
- [ARRA94] Arruarte, A. y I. Fernández. *Tratamiento del Dominio en Herramientas de Ayuda a la Construcción de Sistemas Tutores Inteligentes: Una propuesta de diseño*. Memorias del Congreso IBERAMIA 94, pp. 213-227. Caracas Venezuela.
- [ARRA98] Arruarte, A. *Fundamentos y diseño de IRIS: un entorno para la generación de Sistemas de Enseñanza Inteligentes*. Tesis Doctoral (Tercer Ciclo) de la Universidad del País Vasco, San Sebastián 1998.
- [ARRF99] Arriaga F., M. El Alami, and A. Laureano. *Multi-Agent Simulation for Natural Systems*. En International Conference on Modelling and Simulation (MS' 99). 6-8 de mayo de 1999. Philadelphia, Pennsylvania..
- [ATKJ99] Atkinson-Abutridy, J.A. y J. Carrasco- León. *Un modelo Evolutivo de Redes de Comportamiento para Agentes Autónomos que utilizan mecanismos de Emergencia*. En Revista Iberoamericana de Inteligencia Artificial (eds.) Asociación Española Para la Inteligencia Artificial (AEPIA) No. 8, Otoño/99, pp. 59- 67. 1999.
- [BAJM91] Bajo, M. T. y J. Cañas. *Ciencia Cognitiva*. Editorial Debate. Madrid, España 1991.

- [BANJ93] Banâtre, J.P. *Langages pour le Parallélisme*. Le Courrier du CNRS No. 80, pp. 34. Meudon, Francia. 1993.
- [BARA76] Barr, A., M. Beard and R.C. Atkinson. *The Computer as a Tutorial Laboratory: The Stanford BIP Project*. International Journal of Man Machine Studies. Vol. 8, pp 567-596. 1976.
- [BEEF80] Beer, F. y E. Johnston. *Mecánica Vectorial para Ingenieros TOMO I*. Mc.Graw Hill 1980.
- [BERJ90] Beer, R.D. *Intelligence as Adaptive Behavior: An Experiment in Computational Neuroethology*. Academic Press, Inc. 1990.
- [BERJ93] Bermond, J.C. *Theorie des Graphes*. Le Courrier du CNRS No. 80. Febrero. Meudon, Francia. 1993.
- [BLOB71] Bloom, B. *Taxonomía de los Objetivos de la Educación*. Editorial ATENEO. 1971.
- [BOOG91] Booch, G. *Object Oriented Design with Applications*. The Benjamin Cummings, Inc. 1991.
- [BORL94] Manuales de *Borland Object Windows Programmer's Guide and Reference Guide*. 1994.
- [BORM93] Borillo, M. *De l'informatique à la Cognition*. Le Courrier du CNRS No. 80, pp. 74. Meudon, Francia. 1993.
- [BRAI92] Bratko, I. *Machine Learning and Qualitative Modeling in Medical Applications (Capítulo 2)*. In Artificial Intelligence and Intelligent Tutoring Systems. Editors Danny Kopec and R. Brent Thompson. ELLIS HORWOOD. England 1992.
- [BRAJ97] Bradshaw, J. *Software Agents*. AAAI Press/The MIT Press. 1997.
- [BREB88] Bretch, B. and M. Jones. *Student Models: The Genetic Graph Approach*. Man Machine Studies. Vol. 28, pp. 483-504.1988.
- [BROJ75] Brown, J. S., R.R. Burton and A. G. Bell. *SOPHIE, a Step Towards a Reactive Learning Environment*. Int. J. Man Machine Studies. Vol. 7, pp. 675-696.
- [BROJ82] Brown, J. S., R.R. Burton and J. deKleer. *Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II AND III*. In D. Sleeman & J.S. Brown (Eds.). Intelligent Tutoring Systems, pp. 227-282. New York: Academic Press.
- [BROR86] Brooks, R. *A Robust Layered Control System for a Mobile Robot*. IEEEJ. Robotics and Automation, RA-2 (1), 1986.
- [BROR91](1) Brooks, R. *Intelligence Without Representation*. Artificial Intelligence, 47. pp. 139-159. 1991.

- [BROR91](2) Brooks, R. *Intelligence Without Reason*. Memo no.1293, MIT Artificial Intelligence Laboratory. April 1991.
- [BROR93] Brooks, R. *Elephants Don't Play Chess*. Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back. The MIT Press Cambridge, Massachusetts London, pp. 3-15. England, 1993.
- [BURR82] Burton, R. & J. Brown. *An Investigation of Computer Coaching for Informal Learning Activities*. In D. Sleeman & J. S. Brown (Eds.) *Intelligent Tutoring Systems*, pp. 79-98. New York Academic Press, 1982.
- [BURR88] Burton, R. *The Environment Module of Intelligent Tutoring Systems (Capítulo V)*. In *Foundations of Intelligent Tutoring Systems*. Eds. Martha C. Polson and J. Jeffrey Richardson. Lea eds. Hove & London 1988.
- [CAÑJ99] Cañas, J.M. y M.C. García-Alegre. *Modulated agents for autonomous robot piloting*. Memorias de la VIII Conferencia de la Asociación Española para la Inteligencia Artificial. CAEPIA '99. Vol I, pp. 98-106. Murcia, noviembre, 1999.
- [CASS93] Castañeda, S. *Procesos Cognitivos y Educación Médica*. Facultad de Medicina-UNAM. Serie Seminarios No.1, 1993.
- [CASS94] Castañeda, S. *Cátedra José Gómez Robleda*. UNIDAD 3 y UNIDAD 4. Facultad de Psicología - UNAM 1993.
- [CARJ70] Carbonell, J. *AI in CAI: An Artificial Intelligence Approach to Computer Assisted Instruction*. IEEE transactions on Man Machine Systems, Mms. 11 (4) pp. 190-202, December 1970.
- [CARS83] Card, S. K., T.P. Moran & A. Newell. *The Psychology of Human Computer Interaction*. Hillsdale, NJ: Erlbaum, 1983.
- [CLAW82] Clancy, W.J. *GUIDON*. In the Handbook of Artificial Intelligence. Ed. A. Barr, E.A. Fegenbaum, Los Altos California, pp. 267-278, 1982.
- [CLAW84] Clancy, W.J., R. Letsinger. *NEOMYCIN: Reconfiguring a Rule - Based Expert System for Application to Teaching*. Eds. Mass, Addison - Wesley. 1984.
- [CLAW86](1) Clancy, W.J. *The science and engineering of qualitative models*. KSL Tech. Rep., pp. 86-27, Stanford University. 1986.
- [CLAW86](2) Clancy, W.J. *Qualitative student models*. Annual Rev. Computer Science, Vol. 1, pp. 381-450. 1986.
- [ELAM97] El Alami, M., Arriaga, F., Ugena, A. *SIMUL : A Simulation Environment for Interdisciplinary Project Work*. En Proceedings International Conference on Project Work, pp. 53-60, Roskilde University. Dinamarca. 1997.
- [ELAM98](1) El Alami, M. *Entorno Inteligente de Simulación para el Aprendizaje*. Tesis Doctoral (Tercer Ciclo) de la Universidad Politécnica de Madrid. 1998.

- [ELAM98](2) El Alami, M. F. de Arriaga, A. Ugena. *Multi-Agent Simulation as an Aid for Decision Making and Learning*. En Proceedings 2nd KFUPM Workshop on Information & Computer Science, pp. 121-130. Dhahran. Saudi Arabia. 1998.
- [ELIR90] Elio, R. and P. Scharf. *Modeling Novice-to-Expert Shifts in Problem-Solving Strategy and Knowledge Organization*. Cognitive Science 14, 579-639. 1990.
- [ELOJ95] Elorriaga, J., I. Fernández y J. Gutiérrez. *Sistemas Tutores Inteligentes y Aprendizaje Automático*. Revista Informática Automática. Vol. 28-4/1995, pp 3-18.- Madrid, España.
- [FERI89] Fernández, I. *Estrategias de Enseñanza en un Sistema Inteligente de Enseñanza Asistida por Ordenador*. Tesis Doctoral (Tercer Ciclo) de la Universidad del País Vasco, San Sebastián 1989.
- [FERJ92] Ferber J. and A. Drogoul. *Using Reactive Multi-Agent System Simulation and Problem Solving*. In Distributed Artificial Intelligence Theory and Praxis.- Computer and Information Science Vol. 5. Eds. Nicholas M. Avouris and Les Gasser. Kluwmer Academic Publishers.1992.
- [FERJ93] Ferber, J. *L'intelligence Artificielle Distribuée*. Le Courier du CNRS No. 80, pp. 87-88. Febrero. Meudon, Francia. 1993.
- [FERJ94] Ferber, J. *Simulating with reactive agents*. In Hillebrand, E. and Stender. J. (eds.), Many Agent Simulation and Artificial Life IOS Press, 8-28.
- [FLEB96] Fletcher, B. and Harris, S. *Development of a Virtual Environment Based Training System for ROV Pilots*. En Proceedings 'Oceans'96 MTS/IEEE '. 1996.
- [GAGE85] Gagné, E. *The Cognitive Psychology of school Learning*. Boston, MA: Little Brown & Company. 1985.
- [GARA93] García, A. *Agentes Computacionales Cognitivos Reactivos en la Modelación de la Conducta Visuomotora de la Rana*. Tesis de Grado de Maestría en Ciencias de la Computación. En el Instituto de Investigación en Matemáticas Aplicadas y Sistemas/ Universidad Nacional Autónoma de México. 1993.
- [GARM93]. García-Alegre, M. A. Ribeiro, J. Gasos y J. Salido. *Optimization of fuzzy behavior-based robots navigation in partially known industrial environments*. (Ed.) RezaLangari, John Yen and John Painter. Proceedings of The Third International Conference on INDUSTRIAL FUZZY CONTROL & INTELLIGENT SYSTEMS. IFIS '93, Houston, TEXAS. USA.
- [GARM95] García-Alegre M.C., P. Bustos and D. Guinea. *Complex Behavior Generation on Autonomous Robots: A Case Study*. Proceedings of the IEEE International Conference on Systems, Man & Cybernetics, pp. 1729-1734 Vol. II. Vancouver, British, Columbia, Canada. octubre 1995.

- [GARM97] García-Alegre, M.C. and F. Recio. *Basic Agents for Visual/Motor Coordination of a Mobile Robot*. Proceedings of First International Conference on Autonomous Agents, pp. 429-434. ACM Press. Marina del Rey California. February 5-8, 1997.
- [GARD90] García-Nocetti D.F., P. J. Fleming, H.A. Thompson, H.A. *Implementation of a Transputer-Based Flight Controller*. IEE Proceedings Part D, Vol. 137, No. 3, Special Issue on Control Theory and Applications, pp. 130-136. 1990.
- [GARD92] García-Nocetti, D.F., P.J. Fleming. *Parallel Processing in Digital Control*. Advances in Industrial Control, Springer-Verlag London Ltd (ISBN 3-540-19728-1), 1992.
- [GENM78] Genesereth, M. *An Automated User Consultant for MACSYMA*. Ph. D. thesis, Harvard University, 1978.
- [GHAM93] Ghallab, M. *Le Robot Qui Décide Seul*. Le Courrier du CNRS No. 80, pp. 80-81. Febrero. Meudon, Francia. 1993.
- [GINH86] Ginsburg, H. y S. Opper. *Piaget y la Teoría del Desarrollo Intelectual*. Prentice Hall 1986.
- [GIRJ92] Girard, J., G. Gauthier, S. Levesque. *Une architecture multiagent*. En Memorias Second International Conference, ITS'92. Lectures Notes in Computer Science No. 608. Springer-Verlag, pp. 172- 182.
- [GOLI76] Goldstein, I. *The Computer as a Coach: An Athletic Paradigm for Intellectual Education*. Memo no. 389, December 1976, MIT Artificial Intelligence Laboratory.
- [GOLI79] Goldstein, I. *The Genetic Graph: A Representation for the Evolution of Procedural Knowledge*. Man Machine Studies. Vol. 11, pp. 51-77.1977.
- [GOOR93] Goodwin, R. *Formalizing properties of agents*. Technical Report CMU-CS-93-159, School of Computer Science, Carnegie-Mellon, University, Pittsburgh, PA, 1993.
- [GOTS89] Gott, S.P. *Apprenticeship instruction for real - world tasks: The coordination of procedures, mental models and strategies*. In E.Z. Rothkopf (Ed.), En Revista Research in Education, Vol. 15, (pp. 97-169). Washington, D.C. American Educational Research Association. 1989.
- [GUTJ94](1) Gutiérrez, J., A. Díaz de Ilarraza, I. Fernández, J.A. Elorriaga, y J.A. Vadillo. *Tratamiento de las Interacciones del Alumno en un Sistema Tutor para Entrenamiento*. Memorias del Congreso de IBERAMIA 94, pp. 200-212. Caracas Venezuela.
- [GUTJ94](2) Gutiérrez, J. *INTZA: un Sistema Tutor Inteligente para Entrenamiento en Entornos Industriales*. Tesis Doctoral (Tercer Ciclo) de la Universidad del País Vasco, San Sebastián 1994.
- [HALH88] Half, H.. *Curriculum and Instruction In Automated Tutors (Capítulo IV)*. In Foundations of Intelligent Tutoring Systems. Eds. Martha C. Polson and J. Jeffrey Richardson. Lea eds. Hove & London. 1988.

- [HAYB79] Hayes-Roth, B., F. Hayes-Roth, S. Rosenschein, S. Cammarata. **Modeling Planning as Incremental Opportunistic Process**. Actas del 6º IJCAL, pp. 375-383. Tokio 1979.
- [HAYB97] Hayes-Roth, B. **Introducción**. Proceedings of First International Conference on Autonomous Agents, pp. 5-8. ACM Press. Marina del Rey California. February. 1997.
- [HUAT80] Huang, T.C. **Mecánica para Ingenieros-TOMO I**. Representación y Servicios de Ingeniería S.A. 1980.
- [JOHNP83] Johnson-Laird, P. N. **Mental Models**. Cambridge: Cambridge University Press. Cambridge, Mass.:Harvard University Press.1983.
- [JOHNP88] Johnson-Laird, P. N. **The Computer and the Mind: An Introduction to Cognitive Science**. Editorial Fontana. 1988.
- [JULV99] Julian, V. y C. Carrasco, V. Botti. **Formalización y Traducción a un modelo ejecutable de las Entidades de un Agente ARTIS**. Memorias de la VIII Conferencia de la Asociación Española para la Inteligencia Artificial. CAEPIA '99. Vol I, pp. 144-153. Murcia, noviembre, 1999.
- [KAEL87] Kaelbling, L. **An Architecture for Intelligent Reactive Systems**. Eds. Michael P. Georgeff and Amy L.Lansky. Reasoning about Actions and Plans, pp. 395-410. Morgan Kauffmann Publishers. 1987.
- [KAEL93] Kaelbling, L. and S.J. Rosenschein. **Action and Planning in Embedded Actions**. Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, pp. 35-48 The MIT Press Cambridge, Massachusetts London, England. 1993.
- [LAUA92](1) Laureano, A. **MICRARMS: Un Sistema de Enseñanza Basado en el Conocimiento de Análisis de Armaduras**. Tesis de Grado de Maestría en Ciencias de la Computación. En el Instituto de Investigación en Matemáticas Aplicadas y Sistemas/ Universidad Nacional Autónoma de México. 1992.
- [LAUA92](2) Laureano, A. **Introducción a la Mecánica Estructural por Medio de la Animación por Computadora**. Reporte de investigación No. 286. Departamento de Sistemas - Universidad Autónoma Metropolitana - Azcapotzalco, 1992.
- [LAUA93](1) Laureano, A. **Tecnologías Orientadas a Objetos**. Revista EN LINEA Universidad Autónoma Metropolitana. Vol. 1, Número 2, pp. 26-30. Azcapotzalco, julio de 1993.
- [LAUA93](2) Laureano, A. **Multimedios y Cognición**. Revista Perfiles Educativos. Número 62, pp. 38-41. Centro de Investigaciones y Servicios Educativos. Universidad Nacional Autónoma de México.1993.
- [LAUA94] Laureano, A. **Inteligencia Artificial e Inducción**. Revista EN LINEA Universidad Autónoma Metropolitana. Vol. 1, Número 4, pp. 7-12. Azcapotzalco, marzo de 1994.

[LAUA95] Laureano, A. *Herramientas para la Representación del Conocimiento en Sistemas con Inteligencia Artificial*. Revista EN LINEA Universidad Autónoma Metropolitana. Vol. 1 Número 5, pp. 14-22. Azcapotzalco, mayo de 1995.

[LAUA96](1) Laureano, A. y T. Ortiz. *Introducción a la Programación Orientada a Objetos: Un enfoque con tipos abstractos de datos y estructuras de datos*. libro editado por la Universidad Autónoma Metropolitana - Azcapotzalco. México, 1996.

[LAUA96](2) Laureano, A. y J. Negrete. *Una Arquitectura Reactiva en Sistemas de Enseñanza Inteligentes*. Memorias de Informática Educativa. Seccio-1/3softw-1/2/Word/Arqreac.doc. UNED-Madrid 1996.

[LAUA96](3) Laureano, A. y F. De Arriaga. *Diseño del Dominio en Sistemas de Enseñanza Inteligentes: basado en la reactividad de los agentes*. Memorias de II Jornadas Nacionales de Innovación en las Enseñanzas de las Ingenierías. Vol. II, pp. 987-999. Instituto de Ciencias de la Educación - Universidad Politécnica de Madrid, diciembre de 1996.

[LAUA96](4) Laureano, A. *Un Marco de Referencia para el Desarrollo Integral de los Sistemas Tutores Inteligentes*. En la Revista de Enseñanza y Tecnología de la Asociación para el Desarrollo de la Informática Educativa, Num. 6, diciembre de 1996, pp. 14-23. ESPAÑA.

[LAUA97](1) Laureano, A. *Multiagentes en el Diseño de un Sistema de Enseñanza Inteligente*. Revista Perfiles Educativos, pp. 23-33 No. 75, enero-marzo. Centro de Investigaciones y Servicios Educativos-Universidad Nacional Autónoma de México, 1997.

[LAUA97](2) Laureano, A. y M.C. García-Alegre. *Agentes Autónomos, Robots y Sistemas de Enseñanza Inteligentes*. Informe Técnico Sistemas IAI-CSIC 07/97. Departamento de Sistemas en el Instituto de Automática Industrial - Consejo Superior de Investigaciones Científicas. Madrid, ESPAÑA. 1997.

[LAUA97](3) Laureano, A. and F. de Arriaga. *Un Sistema MultiAgente Reactivo en los Sistemas de Enseñanza Inteligentes*. Revista Informática y Automática. Vol.30. Num. 4, pp 71-84. España. 1997.

[LAUA98](1) Laureano, A. and F. de Arriaga. *An Intelligent Tutoring System To Teach An Integrated Skill*. En Proceedings 2nd KFUPM Workshop on Information & Computer Science (WICCS'98), pp. 71-84. Daharam, Saudi Arabia. 1998.

[LAUA98](2) Laureano, A. *Los Sistemas Reactivos: Un Nuevo Acercamiento a la Inteligencia Artificial Distribuida*. Revista NOVATICA. No. 132, pp. 51-55. 1998.

[LAUA98](3) Laureano, A. and F. de Arriaga. *Multi-Agent Architecture for Intelligent Tutoring Systems*. En Interactive Learning environments. Vol.6, No. 3, pp. 225-250. SWETS & ZEITLINGER. 1998.

[LAUA98](4) Laureano, A. and F. de Arriaga. *El Análisis Cognitivo de Tareas: Una herramienta para modelar la conducta de los Sistemas de Enseñanza Inteligentes*. En Revista Latina de Pensamiento y Lenguaje. Número Monográfico, 2B, del Volumen 4.

Número dedicado a difundir avances tecnológicos sobre Cognición, Educación y Evaluación. 1998.

[LAUA98](5) Laureano, A. y M.C. García-Alegre. Informe Técnico, Sistemas IAI-CSIC, 11/98. *El Análisis Cognitivo de tareas en el Análisis de Conductas para Robots*. Departamento de Sistemas en el Instituto de Automática Industrial - Consejo Superior de Investigaciones Científicas. Madrid, ESPAÑA.

[LAUA99] Laureano, A., F. de Arriaga and M. El Alami. *Learning Environment Characterization by Means of Mental Models*. En International Conference on Computers and Advanced Technology in Education (CATE' 99), 6-8 de mayo de 1999, (Ed.) M.H. HAMZA, pp. 138-143. ISBN 0-88986-243-5. Philadelphia, Pennsylvania.

[LAUA00](1) Laureano, A. and F. de Arriaga. *Reactive Agent Design for Intelligent Tutoring Systems*. En Cybernetics and Systems (an International Journal). TAYLOR AND FRANCIS. enero 2000.

[LAUA00](2) Laureano, A. y S. Arriaga. *Paquete computacional Makatsiná sistema tutorial inteligente para la resolución de estructuras triangulares por el método de los nodos*. Reporte de Investigación. Departamento de Sistemas - Universidad Autónoma Metropolitana - Azcapotzalco, 2000. (por aparecer).

[LEEJ97] Lee, J. *Task Structures as a Basis for Modeling Knowledge - Based System*. International Journal of Intelligent Systems, pp. 167-190. Vol. 12, Number 3, March 1997.

[LESJ97] Lester, J.C. y B.A. Stone. *Increasing Believability in Animated Pedagogical Agents*. En Memorias Autonomous Agents 97, pp. 16-21. Marina del Rey California USA. ISBN: 0-89791-877-0/97/02.

[LUCM95](1). Luck, M. and M. d'Inverno. *A Formal Framework for Agency and Autonomy*. 1st. International conference on Multi-Agent Systems, pp. 254-260. AAAI Press. San Fransisco, California , junio de 1995.

[LUCM95](2). Luck, M. and M. d'Inverno. *Structuring a Z Specification to Provide a Formal Framework for Autonomous Agent Systems*. ZUM'95, LNCS, 967, pp. 47-62. Springer Verlag, Heidelberg, 1995.

[MAEP93](1) Maes, P. *Designing Autonomous Agents*. Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, pp 1-2. The MIT Press Cambridge, Massachusetts London, England, 1993.

[MAEP93](2). Maes, P. *Situated Agents Can Have Goals*. Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, pp. 49-70. The MIT Press Cambridge, Massachusetts London, England, 1993.

[METK93] Metz, K. and D. Hammer. *Learning Physics in a Computer Microworld: In What sense a World?*. Interactive Learning Environments. Vol. 3, Num. 1, 1993.

[MILD76] Mills, D. *Cognition and Instruction (Capítulo 3)*. "Task Analysis in Instructional Design: Some Cases from Mathematics". 1976.

- [MICD92] Michie, D. and M. Bain. *Machine Acquisition of Concepts from Sample Data (Capítulo 1)*. In Artificial Intelligence and Intelligent Tutoring Systems. Editors Danny Kopec and R. Brent Thompson. ELLIS HORWOOD. England 1992.
- [MINM87] Minsky, M. *La Sociedad de la Mente*. Editorial Galápagos. España 1987.
- [MOLJ96] Molina, J. y V. Matellán. *Robots Autónomos: Arquitecturas y Control*. Conferencia en la Universidad Politécnica de Madrid. Ciclo de Robótica, noviembre 1996. Revista Burán 7 de la rama de estudiantes del IEEE, No. 1, abril 1996.
- [MULJ91] Muller, J. *The Design of Intelligent Agents a Layered Approach*. Springer. 1991.
- [MULJ98] Muller, J. *Architecture and applications of intelligent agents: A survey*. The knowledge Engineering Review, Vol. 13: 353-380. 1998.
- [MULP93] Mullins, P. and S. Treu. *A Task-Based Cognitive Model for User - Network Interaction: Defining a Task Taxonomy to Guide the Interface Designer*. Interacting with Computers vol. 5 no. 2 (1993) 139-166.
- [NEGJ92] Negrete, J. *De la Filosofía a la Inteligencia Artificial*. Grupo Noriega Editores 1992.
- [NEGJ94] Negrete J. and A. García-Vega. *Adaptación estructural evolutiva de los robots reactivos*. Memorias del IV Congreso Iberoamericano de Inteligencia Artificial. IBERAMIA94. Caracas. Venezuela 1994.
- [NEGJ96] Negrete, J. *Emergencia de Comportamiento Social Cooperativo Entre agentes Reactivos de un Robot por Reforzamiento Retardado de Origen Propioceptivo*. Memorias del Congreso Iberoamericano de Inteligencia Artificial. Iberamia96. 484-491. Ed. Limusa. México. Noviembre, 1996.
- [NEGJ97](1) Negrete, J. y J. Díaz. *Nuevo Enfoque para la Implementación de Conductas Sociales entre Robots Multi-agente*. En Memorias del Primer Encuentro de Computación ENC97-Aprendizaje 77-82. Edición SMIA-SMCC. Querétaro, México. 1997.
- [NEGJ97](2) Negrete, J. Díaz. *Emergency of Hierarchy among Neurons of a Robot*. En Memorias International Biometric Society Network for Central América, the Caribbean, México, Colombia and Venezuela. 78-80. Xalapa. Ver. Mexico. 1997.
- [NEHP92] Néhémie, P. *A systemic approach for student modelling in a multi-agent aided learning environment. Une architecture multiagent*. En Memorias Second International Conference, ITS'92. Lectures Notes in Computer Science No. 608. Springer-Verlag, pp. 475-482.
- [NIGL91] Nigay, L. and J. Coutaz. *Building User Interfaces: Organizing Software Agents*. A Cookbook for Organizing Software Agents. ESPIRIT Basic Research Action 3066 AMODEUS (Assimilating Models of Designers, Users and Systems) 1991.

- [NIGL93] Nigay, L. and J. Coutaz. *A Design Space For Multimodal Systems: Concurrent Processing and Data Fusion*. INTERCHI '93, Conference Proceedings, pp. 172-178. Abril 1993.
- [NIGL94] Nigay, L. *Software Design and Implementation of Interactive Systems: A case study of Multimodal Interfaces*. Memorias de la Quinta Escuela Internacional de Invierno, Zacatecas. MEXICO 1994.
- [NILN89] Nilsson, N. *From Formal Systems to Practical Systems*. Proceedings of the Rochester Planning Workshop. Eds. J. Tenenberget. al. University of Rochester, New York. 1989.
- [NWAH96] Nwana, H. *Software agents: an overview*. The knowledge Engineering Review. Vol. 11:3, 205-244, 1996.
- [OKTH93] (1) Oktaba, H. *Diseño Orientado a Objetos: Método de Booch*. Notas de la Maestría en Ciencias de la Computación de la UACP y P del CCH, con sede en el IIMAS-UNAM, marzo de 1993. Ciudad Universitaria, MEXICO.
- [OKTH93] (2) Oktaba, H. *Análisis y Diseño Orientado a Objetos*. En Revista Soluciones Avanzadas, Año 1, No.4 julio-agosto 1993.
- [OKTH93] (3) Oktaba, H. y G. Quintanilla. *Abstracción de Datos en Lenguajes Orientados a Objetos*. En Revista Soluciones Avanzadas, Año 1, No.5 septiembre-octubre 1993.
- [PEAD86] Peachy, D. y G. McCalla. *Using planning techniques in intelligent tutoring systems*. En International Journal of Man-Machine Studies, Vol. 24, pp. 77-98. 1986.
- [RAMJ94] Ramirez, J. *La Teoría de las Gráficas*. Revista EN LINEA Universidad Autónoma Metropolitana, Vol. 1, Número 4, pp. 3-4. Traducción comentada del artículo original [BERJ93], Azcapotzalco, marzo 1994.
- [RASJ86] Rasmussen, J. *Information Processing Human-Machine Interaction An approach to Cognitive Engineering*. NORTH-HOLLAND, 1986.
- [REDR89] Redding, R.E. *Perspectives on cognitive task analysis: The state of the state of the art*. In 33rd. Annual Proceedings of the Human Factors Society. pp. 1348-1352. Santa Monica California. Human Factors. 1989.
- [REDR92] Redding, R.E. *A standard procedure for conducting cognitive task analysis*. ERIC. Documentation Reproduction Service .No. DE 340-847. 1992.
- [RICJ89] Rickel, J. *Intelligent Computer Aided Instruction: A Survey Organized Around Systems Components*. IEEE Transactions on Systems, Man, and Cybernetics. Vol. 19, No.1, January - February, 1989.
- [RICJ97] Rickel, J. y L. Jhonson. *Integrating Pedagogical Capabilities in a Virtual Environment Agent*. En Memorias Autonomous Agents 97, pp. 30-38. Marina del Rey California USA. ISBN: 0-89791-877-0/97/02.

- [RICE79] Rich, E.A. *User Modeling Via Stereotypes*. Cognitive Science Vol.3, pp. 329-354, 1979.
- [ROBG81] Robertson, G., A. Newel & K. Ramakrishna. *The ZOG Approach to Man - machine communication*. International Journal of Man Machine Studies. Vol. 14 pp 461-488. 1981.
- [ROSS87] Rosenschein, S. and L. Kaelbling. *The Synthesis of Digital Machines with Provable Epistemic Properties*. Eds. J.F. Halpern, Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning about Knowledge. Morgan Kaufmann, Los Altos, California, 1987.
- [RYDJ93] Ryder, J. M. and R. Redding. *Integrating Cognitive Task Analysis into Instructional Systems Development*. Educational Technology Research & Development. (ET&RD). Vol. 41, No. 2, pp 75-96.- 1993.
- [SANJ88] Sanborn, J. and J. Hendler. *A Model of Reaction for Planning in Dynamic Environments*. Int. Journal AI in Engineering (1988), special Issue on Planning.
- [SCHM87] Schoppers, M. *Universal Plans for Reactive Robots in Unpredictable Environments*. Proceedings of IJCAI-87, Milan Italy 1987.
- [SHOE76] Shortliffe, E.H. *Computer-based medical consultations: MYCIN*. New York: American Elsevier.
- [SPIJ88] Spivey, J.M. *Understanding Z: A Specification Language and its Formal Semantics*. Cambridge University Press. Great Britain, 1988.
- [SLED82] Sleeman, D. and J.S. Brown. *Intelligent Tutoring Systems: An Overview*. Academic Press, pp. 1-11, 1982.
- [SLED85] Sleeman, D. *UMFE: A User Modelling Front-End Subsystem*. International Journal of Man-Machine Studies, Vol. 23 pp. 437-472, 1985.
- [SOML93] Sombé, L. *Informations incomplètes, Informations Contradictoires*. Le Courrier du CNRS No. 80, pp. 68-69. Meudon, Francia. 1993.
- [TORC93] Torra, C. *From Geometric Motion Planning to Neural Motor Control in Robotics*. AICOM. Vol 6. Nr. 1, pp.3-17. March 1993.
- [URRM91] Urretavizcaya, M. *Sistema para la Detección, Diagnóstico y Corrección de Errores en un Sistema Tutor Inteligente*. Tesis Doctoral (Tercer Ciclo) de la Universidad del País Vasco, San Sebastián 1991.
- [VALK92] Valavanis, K and G. Saridis. *Intelligent Robotics Systems Theory, Design and Applications*. Kluwer Academic Publishers. 1992.
- [VANK88] VanLhen, K. *Student Modeling (Capítulo III)*. In Foundations of Intelligent Tutoring Systems. Eds. Martha C. Polson and J. Jeffrey Richardson. Lea eds. Hove & London 1988.

- [WENE87] Wenger, E. *Artificial intelligence and tutoring systems: Computational approaches to the communication of knowledge*. Los Altos, CA: Morgan Kaufmann, 1987.
- [WILM96] Wild, M. *Mental models and computer modelling*. Journal of Computer Assisted Learning. Vol. 12, pp. 10-21. 1996.
- [WILR83] Wilensky, R. *Planning and Understanding. A computational approach to human reasoning*. Addison Wesley. 1983.
- [WOOB84] Woolf B. and D. McDonald. *Building a Computer Tutor: Design Issues*. Computer IEEE, pp. 61-73, September 1984.
- [WOOB92] Woolf B. P. *Towards a Computational Model of Tutoring*. (Eds.) Jones, M. & Winne, P: Adaptive Learning Environments. Foundations and Frontiers, NATO ASI Series, pp. 209-231. Springer Verlag. 1992.
- [WOOD71] Woods, P. M. & J.R. Hartley. *Some Learning Models for Arithmetic Task and their Use in Computer - Based Learning*. British Journal for Education Psychology. Vol. 41, pp. 35-48, 1971.
- [WOOP71] Woods, P. M. & J. R. Hartley. *Some Learning Models for Arithmetic Task and Their Use in Computer - Based Learning*. British Journal for Education Psychology. Vol. 41, pp. 35-48.