

Capítulo 3

Modelos de Comportamiento

Independientemente del diseño del mundo virtual en que habitan, los AVAs poseen Módulos de Comportamiento que les permiten reconocer el entorno a su alrededor y ofrecer respuestas acorde a los estímulos, tanto externos como internos, que reciben. Dicho modulo de comportamiento es el que controla toda las reacciones del agente, y su diseño determinará sus posibilidades de conducta dentro del mundo que habita.

En este capítulo se discuten las técnicas, y la manera en que éstas se integran para dar 'vida' y, si se quiere, hasta personalidad a los agentes virtuales autónomos; a la vez que se hace un repaso por una pequeña parte de la bibliografía dedicada al tema.

La Sección 3.1 comenta algunos marcos de simulación del comportamiento, en los que se incluye todo el proceso de toma de decisiones del agente, desde la información que toma del mundo virtual, hasta la respuesta física del agente. La Sección 3.2 presenta formalmente el paradigma de acción selección, ilustrándolo con algunos ejemplos de trabajos realizados.

3.1 Simulación del Comportamiento

Los Agentes Virtuales Autónomos, y en particular los HVAs, reúnen un conjunto de técnicas que, interrelacionadas adecuadamente, le permiten al AVA un comportamiento lo mas realista posible. Tales técnicas son muy diversas (en cuanto a su propósito se refiere), y pueden variar desde algoritmos de búsqueda de caminos, hasta mecanismos que intentan emular el proceso de razonamiento humano. Dichas técnicas se conjugan para formar lo que muchos autores llaman el Sistema de Comportamiento; sin embargo, y como ocurre en este trabajo y muchas otras obras, para evitar confusiones se empleará el término Marco de Simulación del Comportamiento para definir la arquitectura bajo la cual se integran todas esas técnicas.

Como se ha comentado anteriormente, el trabajo de Craig Reynolds [REYN87] sobre bandadas de aves, marca una nueva etapa en la animación de actores virtuales y sus posibles aplicaciones. Sin embargo, el modelo de comportamiento de sus criaturas es muy básico, restringiendo sus acciones a unos pocos patrones de conducta establecidos por el animador; pero introduce un componente imprescindible en los Marcos de Simulación del Comportamiento actuales, la percepción.

En 1999, Millar, Hanna, y Kealy [MILL99] propusieron un Marco de Simulación del Comportamiento general, que reunía las principales características de los modelos de animación del comportamiento conocidos hasta la fecha, con el propósito de poder producir un amplio rango de animaciones sin necesidad de modificar el código fuente. Tanto sus componentes como sus interacciones aparecen ilustrados en la Figura 3.1. La Componente Ambiente almacena información sobre el mundo virtual, así como la posición, apariencia y estados internos de todos los agentes que habitan en él. El Modulo de Percepción simula cualquier sentido requerido por el animador, no siendo necesario limitarse a la visión, ya que sentidos como el oído y el olfato podrían ser de gran utilidad para lograr comportamientos mas creíbles en determinadas animaciones. Por lo tanto, la componente de percepción proporciona una forma común de especificar las limitaciones sensoriales de los agentes en el mundo virtual. El Modulo de Comportamiento representa el núcleo del sistema y controla el resto de las componentes. Este se encuentra formado a su vez por 4 módulos principales: las variables de estado, la base de reglas, el modulo de memoria, y el modulo de movimiento; todos ellos con el propósito de ofrecer una respuesta conductual a lo que el agente ve o siente. Finalmente, el Modulo Motor se usa para limitar las acciones del actor virtual, proveyendo un conjunto de primitivas de movimiento que el modulo de comportamiento puede invocar.

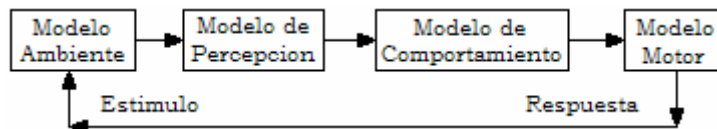


Figura 3.1. Marco de Simulación del Comportamiento propuesto por Millar, Hanna, y Kealy [MILL99]

El enfoque de Millar, Hanna, y Kealy, nos ofrece una visión global de la estructura de los sistemas de animación del comportamiento para AVAs; sin embargo, con la intención de profundizar un poco mas en este tópico (y en las técnicas involucradas), se presenta una breve revisión de algunos Marcos de Simulación del Comportamiento introducidos en los últimos años.

3.1.1 Sistema Clasificador Jerárquico

Jean-Ives Donnart y Jean-Arcady Meyer [DONN94] introducen una arquitectura para la simulación del comportamiento de animales virtuales autónomos (animats, como los autores lo llaman) motivados por grados de satisfacción. Esta arquitectura, basada en lo que denominan un sistema clasificador jerárquico, esta organizada en 5 módulos que habilitan a un animat a moverse por su entorno desde un punto a otro evitando obstáculos y reconociendo objetos. Cada animat esta equipado con 3 sensores de proximidad que lo mantienen informado de la presencia de objetos al frente, 90° a su izquierda, y 90° a su derecha; obteniendo su posición.

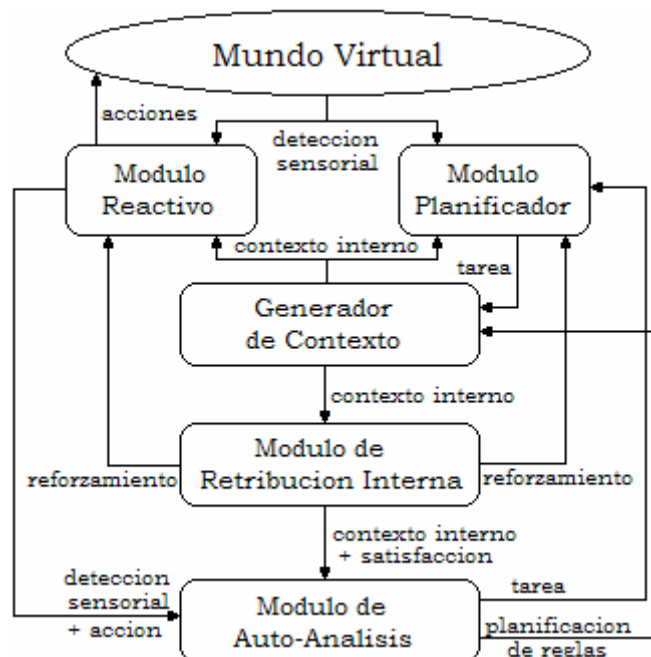


Figura 3.2. Sistema Clasificador Jerárquico [DONN94]

La Figura 3.2 muestra la estructura del sistema. El Modulo Reactivo consta de una serie de reglas (clasificadores) que le permiten al animat reaccionar a la información sensorial que percibe del ambiente, del mismo modo que a su contexto interno producido por el Generador de Contexto. Cada regla consta de un par de condiciones (información sensorial y dirección del objetivo actual) asociada a una de 3 posibles acciones (correspondiendo a uno de los tres posibles movimientos del animat: seguir recto, girar a la derecha, o girar a la izquierda). Cada vez que una regla es activada, una variable interna asociada a ella es reforzada para medir la consecuencia de la correspondiente acción. El Modulo Planificador consta de otra serie de reglas que permiten establecer la secuencia de tareas del animat. A cada regla de este modulo se le asocian dos variables internas (una local y otra global) para determinar la probabilidad de ser activada, y actualizar su número de reglas dinámicamente, incorporando reglas proporcionadas por el Modulo de Auto-Análisis, o removiendo las propias de poco uso. El Generador de Contexto consiste esencialmente de una pila de tareas, donde la del tope representa la tarea actual. Nuevas tareas pueden ser añadidas por el Modulo Planificador, o por el de Auto-Análisis, cuando un obstáculo es detectado. El Generador de Contexto también incluye un algoritmo que transforma la tarea actual en un objetivo, proporcionándole la dirección del mismo al Modulo Reactivo. En el caso de una tarea colocada por el Modulo Planificador, un objetivo corresponde a las coordenadas del punto a ser alcanzado; mientras que en las tareas especificadas por el Modulo de Auto-Análisis, el objetivo describe las coordenadas de proyección de la posición actual del animat en la línea recta a ser cruzada. Una funcionalidad emergente del Generador de Contexto es que habilita al animat a rodear los obstáculos encontrados. El Modulo de Retribución Interna se encarga de actualizar las variables internas de las reglas, mediante un proceso de reforzamiento, y el cual es determinado por el grado de satisfacción de aplicar dicha regla, así como su estimación de éxito. Y finalmente el Modulo de Auto-Análisis, además de caracterizar los obstáculos, es utilizado para detectar recursivamente las posibles posiciones alcanzables mediante un mecanismo de satisfacción para completar la acción. La Figura 3.3(a) muestra un ratón (animat) intentando alcanzar un trozo de queso, y la Figura 3.3(b) su habilidad para rodear un obstáculo basado en las posiciones que lo motivan más (con las que logra mayor satisfacción).

Esta arquitectura le permite a un animat desplazarse por el entorno en función del grado de satisfacción que cada posible acción (paso) le proporciona, mientras intenta alcanzar un objetivo. Si bien, se ha utilizado la arquitectura para definir la ruta que hará el animat hacia su objetivo, no debe confundirse con un algoritmo buscador de caminos, ya que esta

arquitectura puede ser empleada para desarrollar otros tipos de conductas motivadas (por ejemplo, en [SEVI04]).

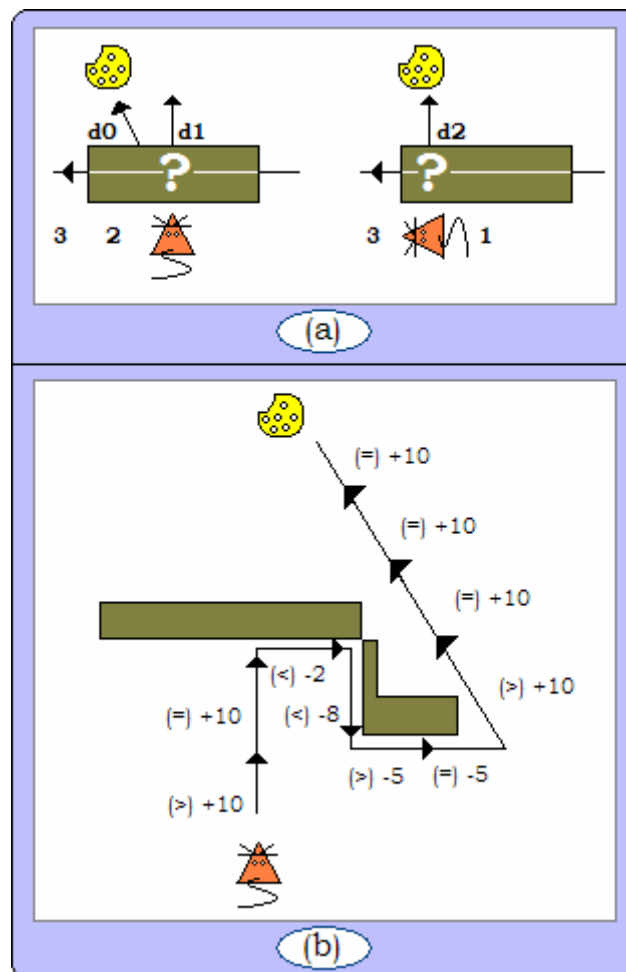


Figura 3.3. Desarrollo de un animat [DONN94]

3.1.2 Locomoción, Percepción y Comportamiento

Un trabajo que marcó etapa en cuanto al realismo visual, no solo en la apariencia física, sino también en el comportamiento autónomo de sus criaturas virtuales, fue el presentado por Terzopoulos y colegas [TERZ94]; donde se recrea un mundo acuático poblado por diversos tipos de peces (depredadores y presas). En el trabajo se presta atención a todos los detalles para conseguir el máximo realismo. El modelado de los peces virtuales se basa en fotografías de los peces reales, construyéndose modelos geométricos 3D de varias especies usando superficies NURBS, y utilizando las fotos como texturas para los mismos. Además, el esqueleto de los peces esta basado en

física, por lo que emplean formulas mecánicas para que sus aristas funcionen como muelles, proporcionándoles flexibilidad corporal; y su sistema de locomoción sigue las reglas hidrodinámicas de movimiento para el nado (ver Figura 3.4).

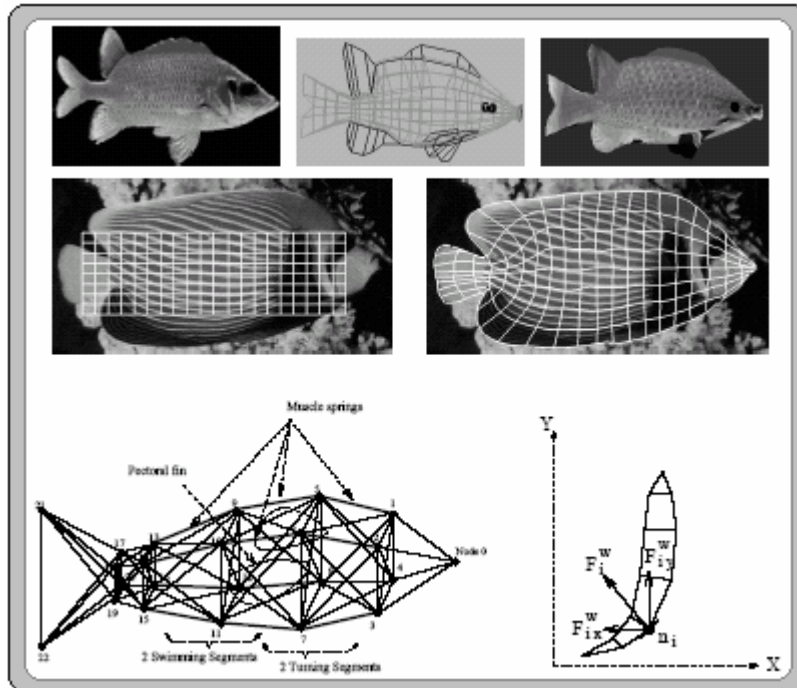


Figura 3.4. Modelado Biomecánico de un Pez Virtual [TERZ94]
© MIT Press

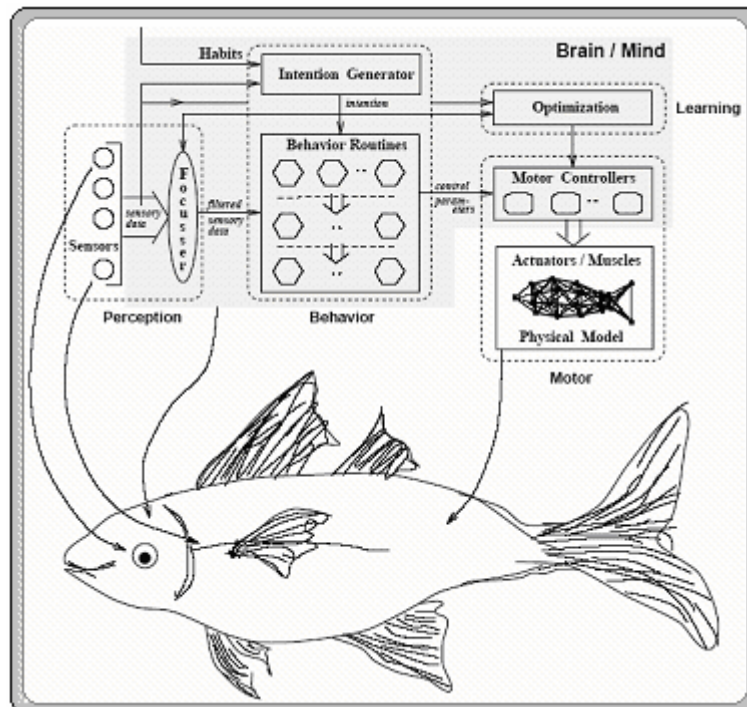


Figura 3.5. Sistema de Simulación del Comportamiento del Pez Virtual [TERZ94] © MIT Press

Sin embargo, el núcleo del trabajo lo constituye su Marco de Simulación de Comportamiento, mostrado en la Figura 3.5.

Veamos como funciona este Marco de Simulación del Comportamiento. El Sistema de Percepción esta compuesto por dos tipos de sensores, el primero es el tradicional sentido de la vista, con un ángulo de visión de 300 grados y un rango limitado, siendo sensible a obstrucciones por la presencia de obstáculos, y a los niveles de iluminación (o brillo). El segundo tipo de sensores le proporciona al pez información sobre los cambios de temperatura del agua virtual en el centro de su cuerpo. Por otro lado, la personalidad de cada pez esta determinada por un conjunto de parámetros de hábitos que definen lo que le gusta o no (claridad, oscuridad, frío, calor, andar en cardume, si es macho o hembra).

El sistema de comportamiento del pez es ejecutado continuamente dentro de un ciclo de simulación, en el que en cada instante de tiempo el Generador de Intenciones emite una intención (objetivo) basada en los hábitos del pez, su estado mental, y la información sensorial entrante; entonces elige y ejecuta una rutina de comportamiento, para luego activar el correspondiente controlador de movimiento. Cada pez cuenta con 3 variables de estado mental, que miden su nivel de hambre, libido, y temor, modeladas por funciones que consideran, por ejemplo, la velocidad de digestión y la hora de la última comida (para el caso del hambre). El Generador de Intenciones es el responsable del comportamiento dirigido a objetivos del pez. Si se percibe peligro inmediato de colisión, la intención generada será evitar la colisión. El Generador de Intenciones también incluye una pequeña área de memoria de corto término para hacer persistente una intención, y así evitar vacilaciones, pasando de un objetivo a otro. El Enfocador solo se activa en casos particulares, cuando se necesita prestar especial atención a algo (por ejemplo a un obstáculo para evitarlo), obteniendo información adicional como la posición del objeto mas cercano. Las Rutinas de Comportamiento estan formadas por un sistema basado en reglas, clasificadas según su propósito: evitar obstáculos estáticos, evitar peces, comer, aparearse, vagar, escapar, viajar en cardume, y abandonar un lugar; y que intentan satisfacer la intención del pez, eligiendo el apropiado controlador de movimiento. El Modulo de Aprendizaje fue desarrollado para conseguir los patrones de movimiento mas eficientes para el desplazamiento del pez, según las leyes hidrodinámicas; y una vez aprendidas se usan para la animación del nado del pez. La Figura 3.6 muestra una escena de la animación lograda con este modelo.



Figura 3.6. Escena de una Simulación del Pez Virtual
[TERZ94] © MIT Press

3.1.3 Comportamientos Complejos

Como es de esperar, los comportamientos más complejos o elaborados para AVAs se encuentran en los humanos virtuales autónomos (HVAs).

Uno de los centros de investigación que mas ha dedicado esfuerzos y recursos al estudio y desarrollo en esta área, es el Laboratorio de Computación Gráfica del Instituto de Tecnología de Suiza (como se comentara en la Sección 1.1.3), fundado y dirigido por Daniel Thalmann. La gran cantidad e importancia de las aportaciones debidas a los trabajos de Thalmann, hace que no sea de extrañar que se comenten algunos de ellos para ilustrar los desarrollos en arquitecturas de simulación del comportamiento para humanos virtuales autónomos.

3.1.3.1 Motivado por Emociones

En 1998, Thalmann y colegas presentan un Sistema de Simulación del Comportamiento para actores autónomos motivados por emociones [BECH98]. La Figura 3.7 muestra el sencillo Marco de Simulación de Comportamiento y el ciclo de animación para el HVA.

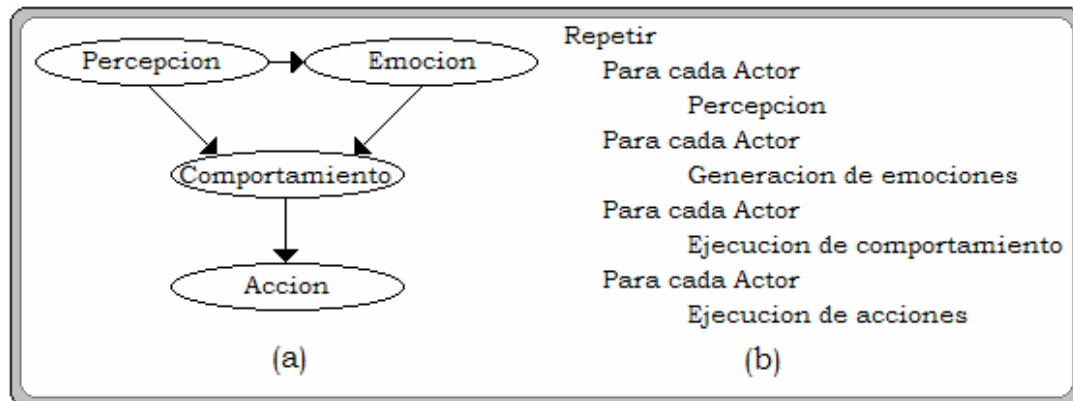


Figura 3.7. (a) Marco de Simulación. (b) Ciclo de Simulación

En dicha arquitectura, la percepción está dividida en tres categorías: percepción de objetos y actores, de acciones de los actores, y de eventos. Los eventos están asociados a las acciones que ejecutan los actores sobre los objetos, y a su vez se clasifican en tres tipos: eventos deseables, eventos que hacen felices a otros actores, y potenciales eventos (los cuales pueden o no ocurrir). El resultado de la percepción es almacenado en tres listas, una de objetos, otra de acciones, y una última de eventos. Las emociones sentidas por un actor son causadas por la percepción de objetos, acciones y/o eventos; y están caracterizadas por condiciones para cada emoción, y variables que miden su intensidad. Cada emoción es evaluada en dos pasos. Un “potencial emocional” es calculado a partir de un conjunto de elementos que contribuyen a la emoción; si el potencial sobrepasa un umbral dado, entonces su valor es usado para calcular la intensidad de la emoción. Dicho umbral corresponde a la mínima intensidad del potencial para que un actor sienta la emoción. Si la condición que caracteriza la emoción es satisfecha, el potencial emocional es calculado a partir de la información de la percepción. El módulo de comportamiento es el responsable de la selección y ejecución del comportamiento que exhibe el actor como consecuencia de la percepción y sus emociones. El módulo controla la ejecución de los comportamientos, y la transición entre comportamientos. El modelo de comportamiento descompone un comportamiento en comportamientos más simples, los cuales a su vez pueden ser descompuestos en otros comportamientos. Cada nivel de esta jerarquía de descomposición contiene uno o más comportamientos, los cuales son ejecutados secuencialmente o concurrentemente. Los comportamientos controlan la ejecución de acciones a través del módulo de acción.

El trabajo incluye varios ejemplos en los que se consideraron emociones como: agrado, desagrado, gozo, pena, admiración, oprobio, resentimiento, tristeza, temor. Cada una de ellas asociada a lo que le da origen, y al gesto facial correspondiente.

3.1.3.2 Motivado por Creencias

En el año 2000, Ángela Caicedo y Daniel Thalmann presentan un Marco de Simulación del Comportamiento mas elaborado para construir humanos virtuales con conductas orientadas a objetivos, presentando un proceso de toma de decisiones compuesto de un planificador de objetivos, manejo dinámico de creencias, objetivos que evolucionan, estados internos, y niveles de confianza [CAIC00].

Para que los humanos virtuales tengan una conducta convincente, los autores utilizan un Modulo de Comportamiento (Unidad de Razonamiento), a la que llaman IVA (Intelligent Virtual Agent), que es conectada al humano virtual para que funcione como su 'cerebro' (ver Figura 3.8).

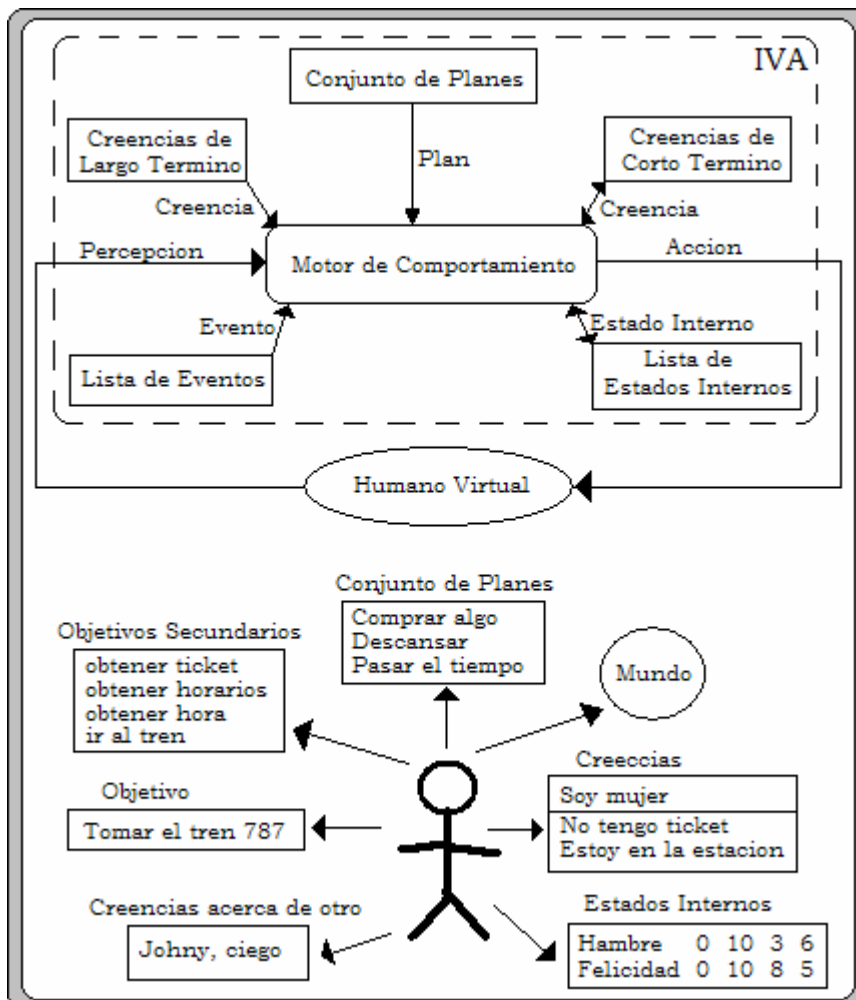


Figura 3.8. Unidad de Razonamiento para HVA [CAIC00]

El IVA esta basado en una arquitectura BDI (beliefs, desires, intentions) de creencias, deseos, e intenciones descrita en [RAO91], que los autores han

extendido para lograr que el humano virtual tenga la habilidad de actuar por sí mismo en un ambiente dinámico, confiando en sus creencias, estados internos, el estado actual del mundo que lo rodea, y las suposiciones sobre otros agentes. Las principales extensiones realizadas fueron: (a) categorización de las creencias con el propósito de simular un esquema simple de memoria de corto y largo término; (b) incorporación de estados internos para los agentes para simular emociones y estados mentales como temor, timidez, etc; (c) niveles de confianza en otros agentes, que evoluciona con el tiempo; y (d) incorporación de emociones y estructura de planes.

El IVA tiene un objetivo principal (el que intenta lograr en un momento dado) y varios subobjetivos (conjunto de planes que le permiten alcanzar el objetivo principal). Mediante un conjunto de planes compitiendo entre ellos, que especifican una secuencia de acciones, el IVA busca alcanzar su objetivo principal. A cada plan se asocia un conjunto de estados internos, un conjunto de precondiciones que deben satisfacerse, y una lista de los efectos de ejecutar el plan; toda esa información es usada para determinar la viabilidad del plan.

La Máquina de Comportamiento (MC) es el eje central del IVA, siendo donde se resuelve el paradigma de acción-selección (los paradigmas de acción-selección se comentan en la Sección 3.2), o sea, donde se eligen las acciones a realizar para los diferentes eventos que ocurren en el entorno del agente. La Máquina de Comportamiento chequea primero, en la lista de eventos pendientes, los eventos activados en instantes específicos de tiempo. Los eventos seleccionados son almacenados en la memoria de corto término, aunque hay casos en los que se almacena en la de largo término. La MC también verifica la percepción del ambiente para ver si coincide con su estado de creencias, y hacer las actualizaciones necesarias; entonces elige el plan que mejor se adecua a los factores externos e internos del agente.

El IVA forma parte de un Módulo Inteligente que los autores llaman IntelMod, y que constituye el sistema de animación en sí. La Figura 3.9 muestra dicho módulo.

Aunque los IVAs son autónomos, el IntelMod le permite al usuario interactuar con el humano virtual y, eventualmente, animarlo; además de proporcionarle al IVA la información sobre los eventos y sus relaciones con otros agentes. La Figura 3.10 muestra un humano virtual reaccionando a diversos eventos ocurridos en su entorno.

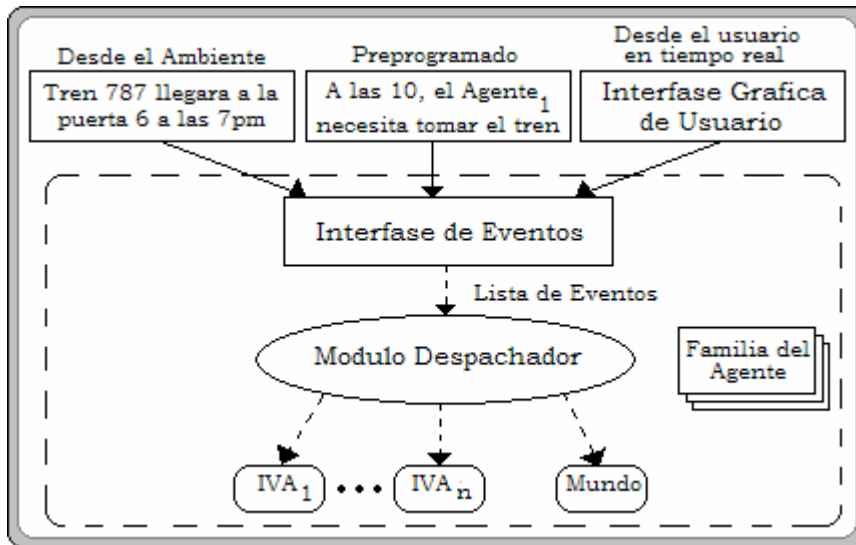


Figura 3.9. IntelMod [CAIC00]

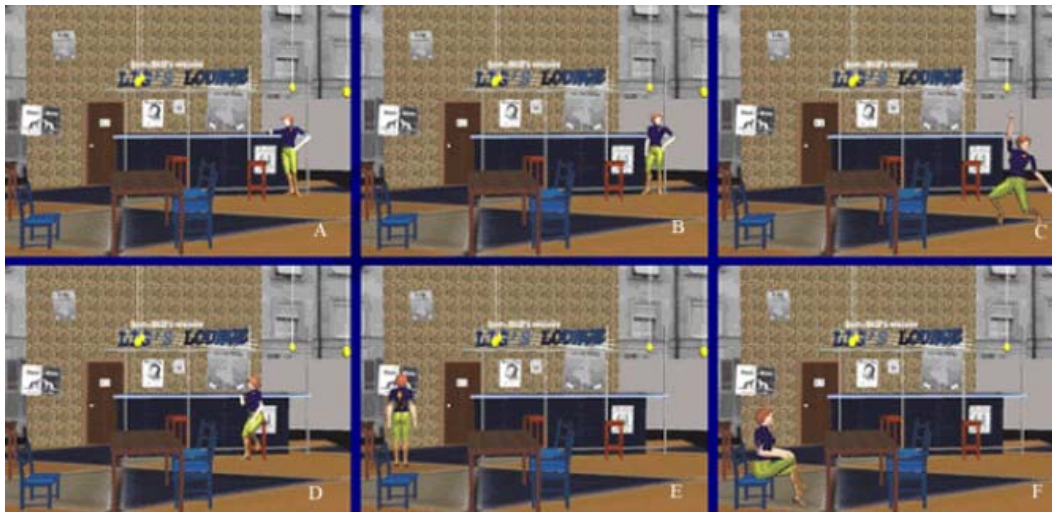


Figura 3.10. Humano Virtual (a) esperando a que le sirvan una bebida, (b) aburriéndose de esperar, (c) bailando al encenderse la música, (d) descansando, (e) viendo carteles, (f) esperando.
[CAIC00] © 3IA International Conference.

3.1.3.3 Separando la Mente del Cuerpo

Un año después, en el 2001, Jean-Sébastien Monzani, Angela Caicedo y Daniel Thalmann [THAL01], mejoran el sistema de animación, sustituyendo el IntelMod por el ACE (discutido en la Sección 2.3.1.1), el cual reúne diferentes comandos para controlar la simulación, como la creación y animación de objetos 3D, humanos virtuales y objetos inteligentes.

El ACE permite la separación entre el ‘cuerpo’ y el ‘cerebro’ del humano virtual, usando los términos ‘niveles bajo y alto’ para referirse a los elementos físicos (que incluyen el cuerpo, y las animaciones básicas como locomoción e interacción con objetos) y al comportamiento, respectivamente. La Figura 3.11 ilustra los niveles de separación, y la comunicación entre ellos.

Los humanos virtuales encapsulan varios motores de movimiento y expresiones faciales; pudiendo reproducir animaciones pregrabadas para caminar, o usar cinemática inversa para una mejor interacción con los objetos. Además, pueden percibir visualmente su entorno.

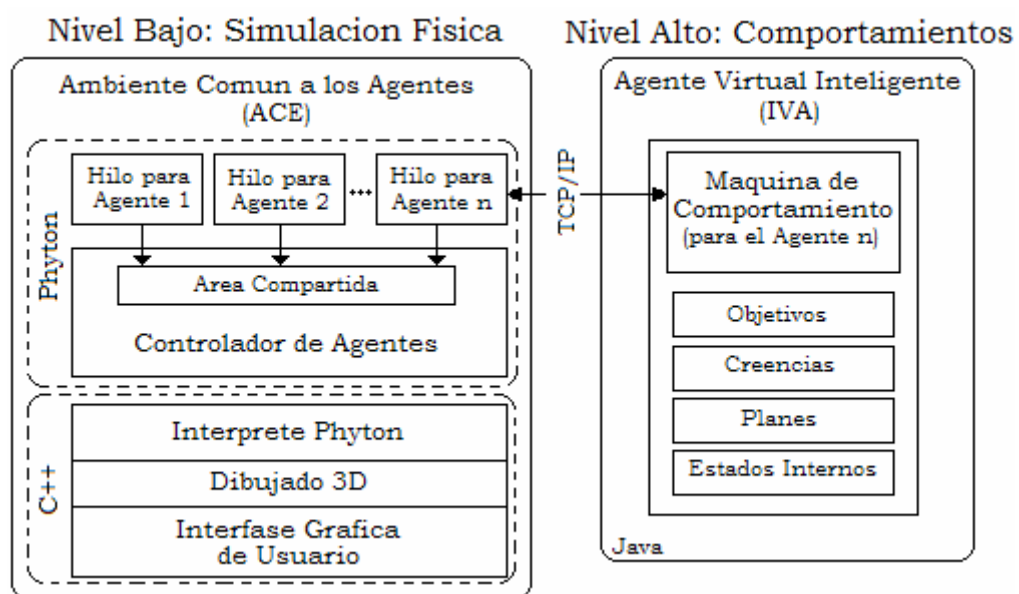


Figura 3.11. Marco de Simulación del Comportamiento propuesto por Thalmann y Caicedo [THAL01]

En el ACE, cada agente es ejecutado en un proceso (o hilo de ejecución) separado, para que pueda interactuar independientemente con el mundo 3D. Al igual que en su trabajo anterior, el IVA mantiene la arquitectura de creencias, deseos e intenciones, de manera que cada agente maneja las suyas propias, y toma decisiones por si mismo. Mientras que a un nivel bajo los agentes tienen que compartir un área común y son manejados por el Controlador de Agentes, lo mismo no es necesario para el IVA. La comunicación entre los niveles bajo y alto se hace con TCP/IP para permitir la implementación en entornos distribuidos, reduciendo el costo de CPU de la maquina encargada de dibujar la escena.

Finalmente, en el 2004, Etienne de Kevin y Daniel Thalmann [THAL04] incorporan al IVA varios Sistemas Clasificadores Jerárquicos (ver Sección 3.1.1) para probar una implementación del modelo motivacional de acción-selección aplicado a humanos virtuales. Los criterios de prueba están

basados en los requerimientos de Tyrrell [TYRR93] para el diseño de mecanismos de acción-selección.

Para las pruebas del modelo se creó una simulación de un humano virtual habitando un entorno complejo (un apartamento en el que puede ‘vivir’ autónomamente), percibiendo su entorno y satisfaciendo algunas motivaciones en lugares específicos. Se definieron un número fijo de motivaciones en total 10: hambre, sed, usar el inodoro, descansar, dormir, bañarse, leer, jugar, regar las plantas, y ver la televisión; cada una asociada a un lugar específico del apartamento en que se puede satisfacer, y a una acción determinada para satisfacerla.

3.2 Paradigma de Acción-Selección

El problema de acción-selección es un tópico abierto aun, ampliamente discutido en las ciencias cognitivas [TYRR92, TYRR93] por su enfoque multidisciplinario, y estudiado por etólogos [BLUM94] (aunque sus modelos tienden a ser conceptuales y difíciles de implementar).

El problema de acción-selección se centra básicamente en la elección del comportamiento; o lo que es lo mismo, elegir de dentro de un conjunto de acciones motoras disponibles ‘la mas apropiada’, según el conjunto de necesidades internas y estímulos externos percibidos por el agente. En esquemas motivacionales, los cuales usan medidas de satisfacción (ver sección 3.1.1.), ‘la mas apropiada’ acción se refiere a aquella que maximiza la medida de satisfacción del agente.

En las secciones anteriores se han comentado algunos mecanismos de acción selección, como parte de los marcos de simulación del comportamiento discutidos previamente. Sin embargo, puede resultar conveniente mencionar brevemente algunos trabajos dedicados a estos mecanismos.

Para ahondar mas en todo lo que concierne a este tópico, se recomienda [LEVI96].

3.2.1 MASM

MASM (Maes’Action Selection Mechanism) es un mecanismo de acción-selección, propuesto por Patti Maes [MAES89, MAES90, MAES91] y revisado

y evaluado por Toby Tyrrell [TYRR94], el cual combina ideas de inteligencia artificial (procesos distribuidos), vida artificial (propiedades emergentes), y etología (motivaciones y deseos alternativos). MASM es una red no jerárquica, distribuida (Figura 3.12). Las dos entradas de alimentación de la red la constituyen los sensores (que obtienen los estímulos externos producidos en el entorno), y de los objetivos/motivaciones (derivado de los estímulos generados internamente). Por otra parte, cada nodo tiene un conjunto de condiciones lógicas sobre el ambiente que debe satisfacer para poder ser ejecutable, o sea, ser una alternativa válida de acción. En los trabajos iniciales de Maes [MAES89, MAES90], los objetivos o motivaciones podían estar solo activados o desactivados; pero en su siguiente trabajo [MAES91] los objetivos o motivaciones permiten valores continuos.

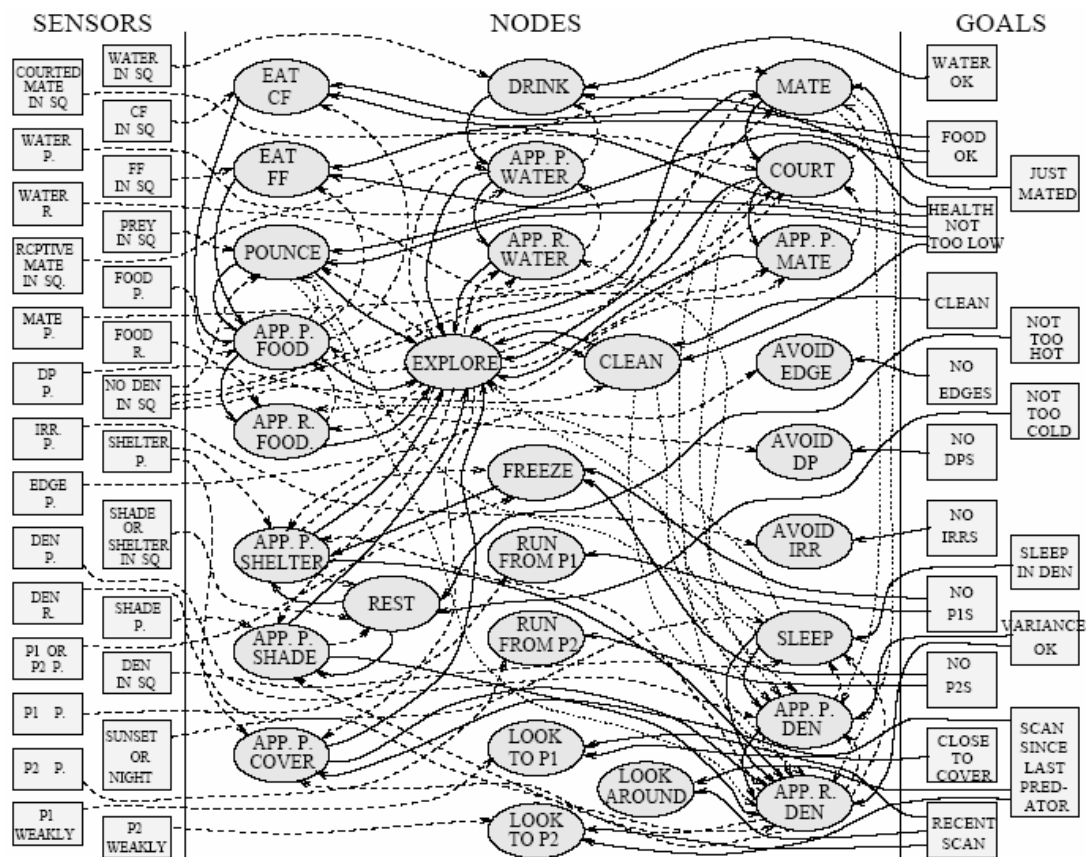


Figura 3.12. Diagrama de una Implementación del Mecanismo de Acción Selección de Maes [TYRR94] © MIT Press

3.2.2 IDA

IDA (Intelligent Distribution Agent) es un software para agentes 'concientes' desarrollado por la marina de los Estados Unidos (ver [NEGA00]). Su arquitectura, mostrada en la Figura 3.13, comprime diferentes módulos,

cada uno dedicado a un proceso cognitivo particular, tales como: percepción, aprendizaje, decisión, memoria asociativa, 'conciencia', emociones, deliberación, y generación de lenguaje. Su mecanismo computacional incorpora variantes y/o extensiones de la red de comportamiento de Maes (previamente descrita). IDA maneja un conjunto de restricciones de las necesidades de la marina, y las políticas y preferencias de los marineros.

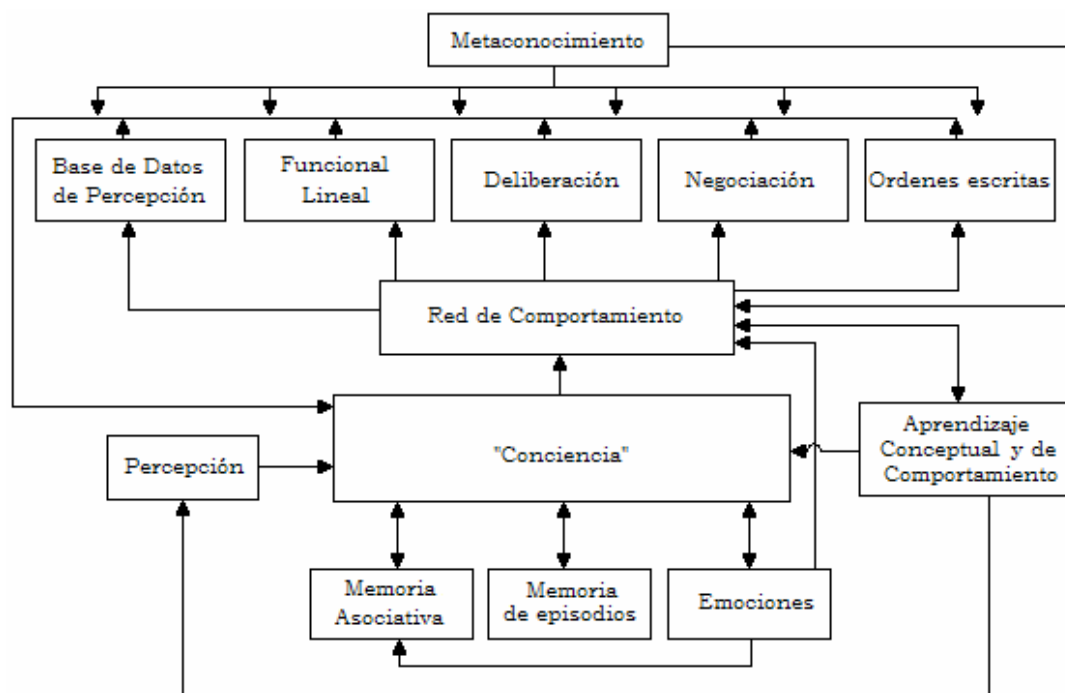


Figura 3.13. Estructura de IDA [NEGA00]

3.2.3 APOC

APOC [SCHE02] es una arquitectura para mecanismos de acción selección que provee una variedad de capacidades para la especificación de arquitecturas de agentes complejos; consta de componentes computacionales heterogéneos, los cuales poseen cualquiera de los cuatro siguientes enlaces entre ellos: (1) Activación, (2) Prioridad, (3) Observación, y (4) Componente enlace; de allí su nombre, APOC. Todos los mecanismos de acción selección estándar pueden ser implementados con esta arquitectura, además de proporcionar herramientas para manejar los recursos y modificar el mecanismo de acción selección en tiempo de corrida. La Figura 3.14 muestra un sencillo ejemplo de una implementación de esta arquitectura.

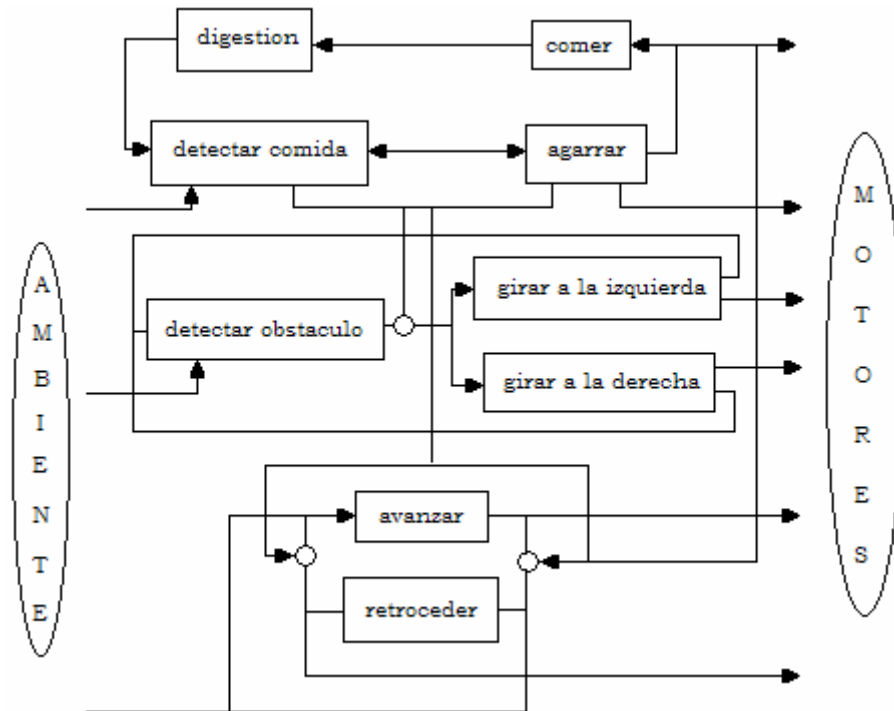


Figura 3.14. Ejemplo de una Implementación con APOC [SCHE02]