

Formalismos de planificación para la animación comportamental de humanos virtuales*

Alicia Fornés Bisquerra, Miguel Lozano Ibáñez, Fernando Barber Miralles.

Departamento de Informática, Universidad de Valencia,
Dr. Moliner 50 (Burjassot), Valencia, España
Email: afornesb@yahoo.es, {miguel.lozano, fernando.barber}@uv.es

Resumen

En este artículo presentamos una comparación entre dos formalismos de planificación válidos, utilizados en la animación comportamental de personajes virtuales 3D inteligentes. El primero de ellos, las redes jerárquicas de tareas (Hierarchical Task Networks, HTNs) han sido introducidas con éxito por Cavazza en entornos de storytelling [1][2], el segundo, MinMin-HSP (Heuristic Search Planning), utiliza técnicas de planificación basadas en búsqueda heurística [3][4] y ha sido diseñado de forma que soporte los principales requerimientos dinámicos de estos entornos. Ambos sistemas inteligentes han sido integrados sobre el mismo escenario, extraído de la popular serie de TV *Friends*, donde la simulación visual llevará asociada un hilo narrativo o historia, que depende en gran medida del comportamiento demostrado por los personajes 3D en su entorno. Los resultados expuestos han sido obtenidos mediante la integración de estos sistemas autónomos en la plataforma gráfica propuesta por el motor de juegos Unreal Tournament, soportando la interacción necesaria mediante el uso de sockets UDP y generando distintas instancias o *storylines*, a partir del escenario propuesto.

Palabras clave: Actores virtuales, Simulaciones gráficas interactivas, Animación comportamental, Búsqueda heurística, Planificación de tareas.

1 Introducción

En los entornos dinámicos 3D se requiere una mayor interacción con los actores virtuales, donde las expectativas de calidad comportamental aún no se corresponden con la gran calidad gráfica conseguida. El necesario aumento de la calidad comportamental citada, vendrá de la mano de la integración de personajes inteligentes, dotados de las principales características de los seres vivos: percepción, locomoción, expresividad, comportamiento adaptativo, cognición, razonamiento, aprendizaje e inteligencia. En los últimos años, se han integrado distintas arquitecturas de agente para el soporte de estos requerimientos comportamentales. Así, Terzopoulos [5] ha desarrollado uno de los más conocidos ejemplos de vida artificial: los peces artificiales, agentes autónomos con características de un pez real, con su propio movimiento, percepción, comportamiento y aprendizaje. Estas técnicas han sido adaptadas a humanoides [6], con un sistema de navegación basado en visión sintética y percepción, además del estudio del tacto y audición virtual. Otro ejemplo de humano virtual es Jack [7], un personaje usado principalmente para el estudio de modelos biomecánicos, aunque integra un sistema de planificación completo para la animación de su comportamiento.

* Parcialmente subvencionado por el proyecto GVA-CTIDIB/2002/182

Quizás el formalismo más comúnmente utilizado en este ámbito sea las máquinas de estados finitos, pero sus claras limitaciones, como planes pre-compilados y en cuanto a la difusa representación del objetivo, requieren otro tipo de técnicas más flexibles y acordes con el dinamismo del entorno, como es el caso de los sistemas de planificación. En este dominio de animación comportamental, el objetivo propuesto para la generación de un comportamiento inteligente residirá en encontrar (usando algoritmos de búsqueda heurística en tiempo real en nuestro caso) una correcta secuencia de acciones que permitan al actor o personaje conseguir sus objetivos. Los planes así calculados poseerán ahora el valor añadido de su visualización en el entorno virtual, produciendo la animación inteligente del comportamiento buscada.

Para mantener tanto el interés como la inmersión del usuari@ en el entorno 3D, las aplicaciones 3D interactivas en tiempo real, como simulaciones dedicadas al entrenamiento, aplicaciones médicas o plataformas de ocio, requerirán de la integración de este tipo de personajes capaces de soportar un mayor número de situaciones “virtuales” con las que interactuar con los elementos de su entorno, incluyendo lógicamente al usuari@, tal y como proponen las metáforas en torno a la TV interactiva y *storytelling*, lo cual ha quedado claramente demostrado desde la aparición del popular juego *The Sims*.

En el punto siguiente repasaremos el marco de trabajo empleado, repasando los entornos de *storytelling*, donde además de la calidad visual de la simulación 3D se valorará el contenido narrativo de la historia generada. Los puntos 3 y 4 revisarán los formalismos objeto de la comparación citada, mientras que en el punto 5 expondremos los resultados obtenidos por el planificador basado en técnicas de búsqueda heurística introducido (MinMin-HSP), comparando éstos con los previamente obtenidos por el equipo de Cavaza[1][2] en trabajos anteriores a éste.

2 Sistemas interactivos

Los sistemas de generación de historias interactivas (Interactive Storytelling) se centran en el estudio de técnicas de animación de personajes inteligentes para generar escenarios donde los personajes crean la historia a través de la interacción de sus sistemas de comportamiento, normalmente basados en técnicas de planificación.

La idea principal del Storytelling [8] [9] es la de generar una historia como resultado de las acciones de los personajes o del usuario. La narración se crea por la múltiple interacción entre los agentes autónomos, debida a que los personajes interactúan en el mundo virtual y compiten por los recursos, ya sean objetos u otros agentes.

Los encuentros entre personajes y sus planes son arbitrarios, y tienen el potencial de crear situaciones de importancia narrativa, variando el guión argumental y desembocando en distintos finales para la historia.

El usuario es un espectador que puede explorar el entorno, interviniendo físicamente sobre los objetos (cogiendo, quitando o moviendo objetos) y pasando información a los personajes (dando instrucciones o aconsejándoles). De esta forma, en vez de verse inmerso en la historia, puede influenciarla.

Los principales requerimientos de los sistemas interactivos son:

- Entorno dinámico, que cambie según la influencia de los actores o usuario.
- Sistema que intercale planificación y ejecución, para que el agente se adapte a este entorno dinámico.

- Replanificación en caso de que las acciones fallen.
- Soporte de interactividad por parte del usuario.

Hay diversos factores que afectan al hilo argumental, y por tanto, hacen que la narración no sea predecible desde la perspectiva del usuario:

- Arbitraria localización espacial inicial de los actores, duración de las acciones y velocidad de los actores.
- Interacción entre planes de los actores debido a la competición por los recursos.
- Resultado arbitrario de algunas acciones terminales.
- El estado emocional de los personajes influye en la elección de sus acciones.
- La intervención del usuario.

A continuación se expondrán y compararán dos formalismos de planificación que pueden ser usados para la generación de historias interactivas: HTN (Hierarchical Task Network) y HSP (Heuristic Search Planning).

3 Redes jerárquicas de tareas (HTN)

Las redes jerárquicas de tareas son una de las técnicas más antiguas para proveer de conocimiento específico al dominio del problema a un sistema de planificación. Se fundamenta en la descomposición de tareas y su total ordenación.

Cada comportamiento del personaje se describirá mediante un HTN, descomponiéndose el objetivo (el nodo raíz) en subobjetivos independientes. Las capas más bajas muestran las diferentes formas de conseguir cada uno de estos subobjetivos. La búsqueda se realiza a través del grafo Y/O, donde cada rama representa la composición de acciones (ramas Y) o alternativas (ramas O), mientras que el nivel más bajo representa las acciones terminales (que ejecutará el agente).

Cavazza ha integrado satisfactoriamente las HTN [1] [2] usando una variación del algoritmo RTAO* (Real Time AO*) que intercala planificación y ejecución. De esta forma sólo se calcula una solución parcial, suficiente para la elección de la próxima acción a ejecutar por el personaje. En el campo de agentes pedagógicos, Elliott y Rickel también han usado satisfactoriamente este tipo de formalismos [10].

Para mostrar un ejemplo de aplicación, Cavazza se ha inspirado en la popular serie televisiva Friends, donde Ross intenta conseguir una cita con Rachel. Este comportamiento (ver figura 1) se define como un objetivo de alto nivel, que se desglosará en diversos subobjetivos (cada uno de ellos es una rama Y del grafo): Obtener información sobre ella (preferencias y gustos), ganarse su amistad, conseguir hablar con ella en privado y finalmente, pedirle una cita.

Cada uno de estos subobjetivos se puede conseguir de diversas formas (ramas O):

Para obtener información, se puede leer su diario, telefonar a su madre o bien pedir esta información a algún amigo. Para ganarse su afecto, Ross puede optar por regalarle un regalo (que le guste a ella) o bien ser amigable (piropeándola o ganándose la amistad de sus amigos). Para conseguir hablar con ella en privado hay toda una serie de alternativas, desde echar a los demás, atraer su atención o apartándola de los otros. Finalmente, puede optar por pedirle para salir en persona o bien, pedir a alguno de sus amigos que se lo pregunte en su lugar (Ross puede ser muy tímido).

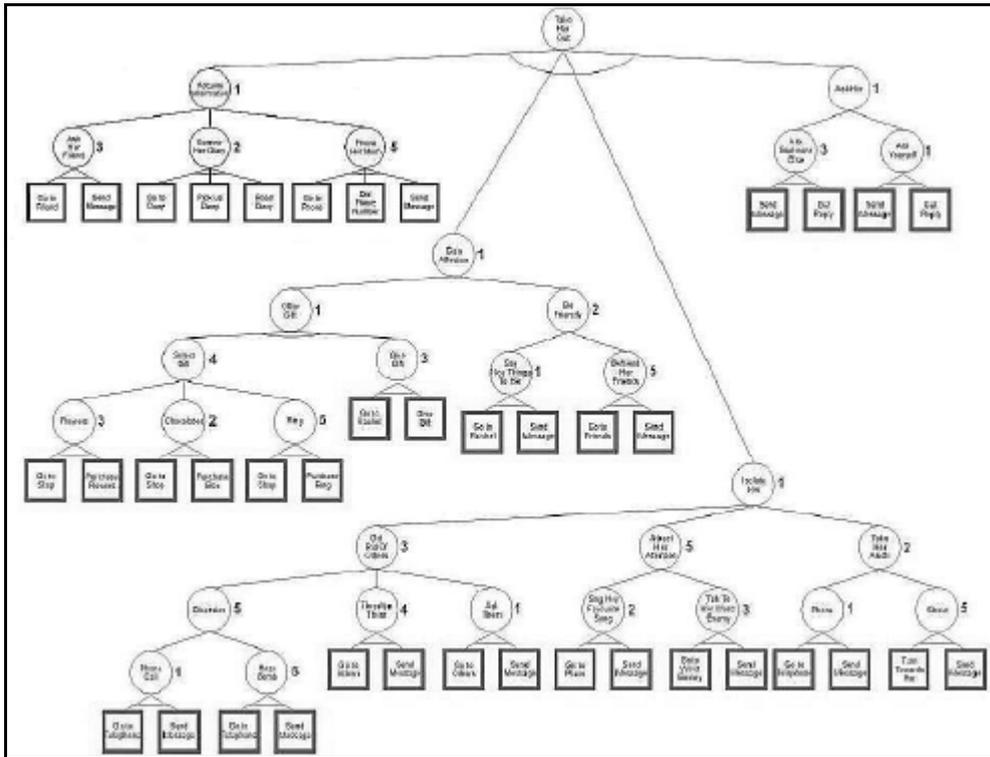


Figura 1. Representación HTN de Ross, extraída de [2]

El principal interés de la historia es saber si Ross lo conseguirá o no (según la elección de las acciones, el final variará).

La variabilidad de la narración y la interacción entre los diversos personajes (por la competición de recursos), hace que sean necesarios diversos mecanismos para controlar las diferentes situaciones:

- Fallo de las acciones y replanificación: Cada vez que una acción falle, se debe realizar una replanificación. En el ejemplo, si Ross no consigue telefonar a la madre de Rachel para pedirle información sobre las preferencias de su hija, Ross optará por leer su diario o pedirle esta información a algún amigo de ella.
- Razonamiento situado: Algunas situaciones inesperadas (o encuentros casuales entre agentes) no pueden ignorarse. El razonamiento situado está orientado a obtener un resultado específico ante una situación concreta, o a evitar un resultado indeseable. Por ejemplo, si al principio de la simulación (antes de obtener información sobre sus gustos) Ross se encuentra con Rachel de forma accidental, debe plantearse cómo actuar. Ross puede hablar con ella o esconderse de ella, pero no puede ignorarla. Tampoco puede seducirla porque aún no conoce sus preferencias.
- Reparación de acciones: Se refiere al restablecimiento de condiciones de ejecución para conseguir el mismo estado final que la acción original. Las condiciones de ejecución representan la información desconocida por el agente, y que varía de forma dinámica (son condiciones de observabilidad que sólo se conocerán en tiempo de ejecución). Un ejemplo sería la disponibilidad de un determinado objeto o factores externos (temperatura de una habitación...). La reparación de acciones puede ilustrarse en el ejemplo de Friends: si Ross va a la habitación de Rachel para leer su

diario, puede ocurrir que Rachel esté escribiendo en él. Ross puede optar por replanificar (buscando otra fuente de información) o reparar la acción, es decir, esperar (sin ser visto) a que Rachel acabe y abandone la habitación.

Tanto el razonamiento situado como la reparación de acciones son dos problemas aún no resueltos completamente. Las principales dificultades residen en el incremento de la interacción entre los agentes y las descripciones asociadas al razonamiento situado, por las cuales aún no hay claros principios metodológicos establecidos.

4 Planificadores de búsqueda heurística (HSP)

Las HTN permiten una buena representación del conjunto de posibles soluciones, pero su limitación en la generación de historias más complejas que permitan dependencias a largo plazo, provoca que se consideren otro tipo de técnicas. Si se introducen subtareas con dependencias, desaparecería el orden parcial de HTN, permitiendo la creación de nuevas narraciones. Por ejemplo, el hecho de que una acción falle, puede eliminar átomos previamente conseguidos, introduciendo de esta forma flexibilidad en la narración (si Ross es grosero, perderá el subobjetivo de ganarse su afecto, y por tanto deberá volver a conseguirlo).

Los planificadores de búsqueda heurística (HSP) han demostrado gran eficiencia en un gran número de problemas de planificación. Mediante su adaptación a los requerimientos narrativos de la generación de historias, es posible realizar animaciones de alto valor narrativo usando un objetivo simple como conductor de la narración, ofreciendo un alto potencial para posibilitar variaciones en la narración [3] [4].

HSP se caracteriza por una representación del dominio del problema, un algoritmo de búsqueda y una función heurística.

Para el dominio del problema se ha optado por una representación basada en STRIPS [11], donde cada personaje tendrá definido un conjunto de operadores correspondientes a su comportamiento, que soportarán dependencias a largo plazo. De esta forma, la descripción de cada operador estará formada por una lista de precondiciones y efectos, en forma de lista de borrar y lista de añadir. Los operadores enlazarán los estados con sus sucesores de acuerdo a las precondiciones de cada operador.

Cada estado del agente estará formado por átomos (Regalo, Simpatía, Manos vacías...) y fluents (Humor, Timidez, Educación...). Los fluents son funciones que varían en el tiempo y representan el estado interno del personaje.

La estructura de la representación del problema estará formada por:

- Estado inicial y final: Contiene la lista de átomos y fluents correspondientes al estado inicial y objetivo del personaje.
- Operadores (ver figura 2):
 - o Precondiciones: Contiene la lista de átomos, fluents y condiciones de ejecución que deben satisfacerse para que el operador sea aplicable.
 - o Efectos: Contiene la lista de átomos y fluents que se verán actualizados como resultado de la aplicación del operador. Habrá doble lista de átomos y fluents, correspondientes a los efectos producidos ante el éxito o fracaso de la aplicación del operador.
 - o Acciones: Contiene la lista de acciones terminales asociadas al operador.

OPERADOR		Telefonar a la madre de Rachel	
PRECONDICIONES		Átomos	Manos vacías
		Fluents	
		Condiciones de ejecución	Teléfono cerca, Teléfono libre
EFECTOS	Éxito	Átomos	Información (le gustan las flores, los bombones...)
		Fluents	Humor + 3
	Fracaso	Átomos	
		Fluents	Humor - 1
ACCIONES		Ir hasta el teléfono, Marcar número de teléfono, Mensaje a la madre	

OPERADOR		Decirle piropos	
PRECONDICIONES		Átomos	
		Fluents	Timidez < Y, Humor > Z
		Condiciones de ejecución	Ver Rachel, Rachel libre
EFECTOS	Éxito	Átomos	Simpatía
		Fluents	Afecto + 5, Humor + 3
	Fracaso	Átomos	
		Fluents	Afecto - 1, Humor - 2
ACCIONES		Ir hasta Rachel, Decir piropo	

Figura 2 – Ejemplo de operadores STRIPS

El algoritmo de búsqueda elegido ha sido el MinMin de R. Korf [12], que intercala planificación y ejecución. Básicamente, MinMin realiza una búsqueda hacia adelante desde el estado actual del agente, comprobando qué operadores pueden aplicarse. Para que un operador sea aplicable en un determinado estado, debe satisfacerse la lista de átomos y fluents de sus precondiciones. La lista de condiciones de ejecución no se evalúan en la búsqueda, sino que se evalúan en tiempo de ejecución (permitiendo el fallo de las acciones).

Debemos observar que se obtiene el conjunto de posibles estados futuros bajo la hipótesis de un mundo cerrado (si un átomo no se encuentra en la lista de átomos, se asume que su valor es falso).

Dado el alto coste computacional de esta búsqueda exhaustiva, cuando se llega a un cierto nivel de profundidad (horizonte de búsqueda) se usa la función heurística introducida por Bonet y Geffner para la planificación [13], que subestimaré la distancia de éstos al estado objetivo. Esta heurística consiste en ignorar las listas de borrado, expandiendo los hechos atómicos que pertenecen a las postcondiciones, hasta que se satisfagan todos los átomos del objetivo.

Como que el cálculo de la función heurística es el cuello de botella del algoritmo, se usan dos tipos de poda: Se descartan los planes de más longitud que el máximo establecido y los nodos que no producen nuevos efectos (los llamados nodos inútiles). Además, para evitar los ciclos y bucles infinitos, se realiza una poda en los planes muy largos, además de la introducción de átomos de control [4].

5 Resultados

Para la obtención de resultados se ha usado el UnReal Tournament, que provee un motor gráfico eficiente, ahorrando tiempo de desarrollo y permitiendo una fácil integración de la visualización e interacción con los objetos. Se usa C++ con una interfaz de sockets UDP y una API, el UnReal Script, que permite la definición de nuevas acciones a través de primitivas básicas.

Para realizar la comparación con HTN, se ha adaptado la descripción del HTN del ejemplo de Friends con 19 operadores STRIPS. Al principio MinMin realiza una búsqueda y extrae diversos planes, intentando satisfacer el de menor longitud: *Pedir información a algún amigo de Rachel, Comprarle un regalo, Pedir que les dejen solos, Decirle piropos, Gritar para que Rachel vaya a él, Darle el regalo y finalmente, pedirle una cita.*

A simple vista puede parecer que las historias creadas son cortas, pero esto depende de si los operadores aplicados tienen éxito. En la figura 3 se muestra cómo el fracaso en la aplicación de algunos operadores puede incluso obligar a conseguir subobjetivos alcanzados previamente:

Cuando Ross falla en el intento de ganarse la amistad de sus amigos, decide cambiar de estrategia, intentando satisfacer el subobjetivo de obtener información sobre sus preferencias. Luego, cuando no hay flores en la tienda, decide telefonar a su madre para pedir más información sobre su hija (la información de que le gustan las flores no le sirve dado que en la tienda no quedan). Como se puede observar, éste es un ejemplo de cómo un subobjetivo (información) se debe volver a conseguir.

Tras entregarle los bombones y piropoarla, Ross logra el subobjetivo de ganarse su afecto, pero luego, cuando intenta de forma grosera quedarse a solas con ella, pierde su afecto. Entonces, debe volver a ganarse su afecto, y finalmente, consigue que Mónica pregunte a Rachel si quiere salir con él.

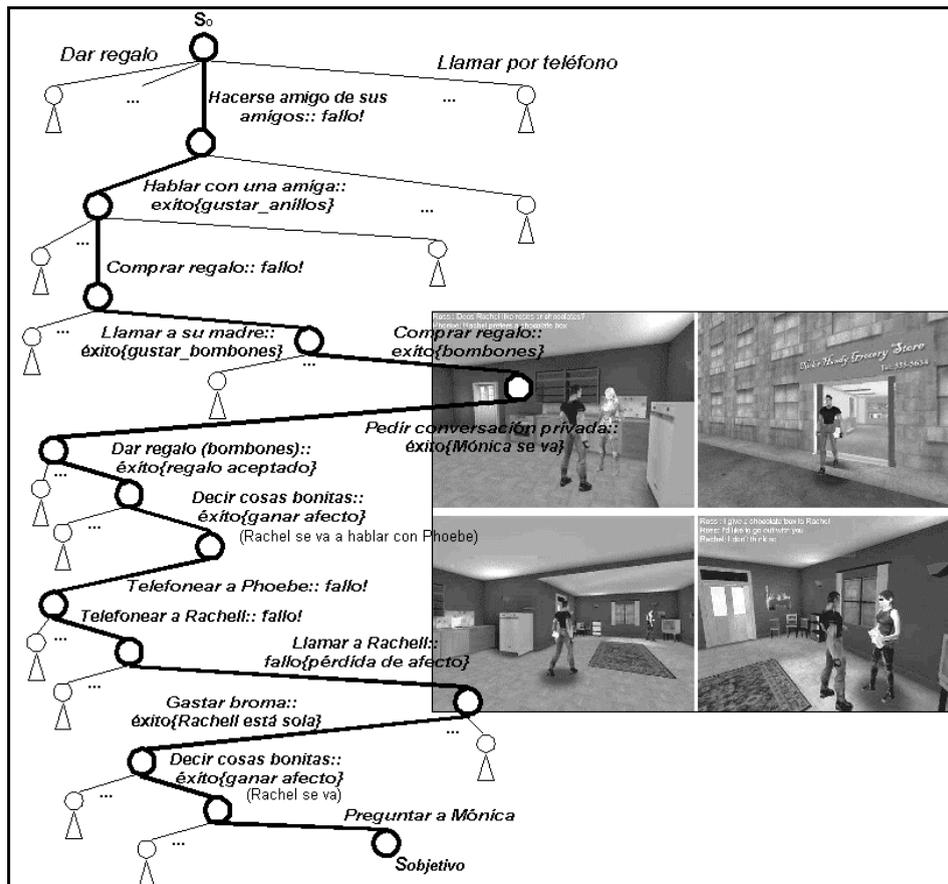


Figura 3 –Traza MinMin-HSP para el problema planteado

6 Conclusiones

En este artículo hemos analizado dos formalismos válidos para la animación comportamental de actores o personajes virtuales en el marco conocido como "Interactive Storytelling". Este dominio posee, además de fuertes requerimientos de representación, la necesidad de intercalar planificación y ejecución. En este sentido, HTN ofrece claros principios para el diseño de historias, a partir de una visión global del hilo argumental, motivada por la descomposición totalmente ordenada de las tareas que el personaje debe ir realizando. De esta forma se preserva una estructura básica temporal para las subobjetivos de alto nivel, mientras que la variabilidad vendrá asociada a las acciones de nivel más bajo.

La total descomposición y ordenación de tareas (HTNs) comporta serias limitaciones en cuanto al tipo de historias a simular. Por una parte, cuando una tarea se ha completado, no es posible deshacer sus efectos, mientras que por otra, cualquier narración simple normalmente conlleva dependencias entre las acciones contenidas en la misma. Por último, la replanificación se realiza en el nivel superior del HTN (intentando conseguir el mismo subobjetivo de otra forma), impidiendo un posible cambio de estrategia.

Mediante HSP hemos aumentado la complejidad del problema, dado que siempre se consideran la totalidad de operadores aplicables, lo que posibilitará los cambios de estrategia comentados. De esta forma, en vez de usar otro operador para conseguir el mismo subobjetivo (como ocurre en HTN), se puede optar por cambiar de estrategia e intentar satisfacer algún otro subobjetivo. Gracias a eso, HSP ofrece una mayor variabilidad en cuanto al número de historias generadas [14].

En dominios de Storytelling la variabilidad de la narración es un factor decisivo, el cual está íntimamente relacionado con la actual variabilidad de los personajes a la hora de la toma de sus decisiones, por lo que las técnicas de animación comportamental basadas en HSP han resultado ser una buena aproximación al problema.

Aunque ambos formalismos de planificación han resultado ser válidos en cuanto a los requerimientos temporales de este tipo de aplicaciones gráficas, HTN ofrece claros principios para el diseño de historias (visión global del hilo argumental), mientras que HSP ofrece mayor variabilidad en las historias generadas. La novedad e interés en este campo de investigación, junto con la cantidad de problemas aún sin resolver, como el razonamiento situado o la reparación de acciones, nos alientan a seguir en esta línea, con el objetivo de poder producir y visualizar distintas historias multi-personaje en Entornos Virtuales 3D.

Referencias

- [1] M. Cavazza, F. Charles, S. J. Mead, "Character-based interactive Storytelling", *Interactive Entertainment*, Julio/Agosto 2002, *IEEE Intelligent Systems*
- [2] M. Cavazza, F. Charles, S. J. Mead, "Interacting with virtual characters in Interactive Storytelling", *Proceedings of the Autonomous Agents Conf., AAMAS'02*. Bologna, Italia 2002.
- [3] M. Lozano, S. J. Mead, M. Cavazza, F. Charles, "Search-based planning for character

animation” 2ª Conferencia Internacional en aplicación y desarrollo de videojuegos. Enero 2003, Universidad de Hong Kong, HKSAR.

- [4] M. Lozano, S. J. Mead, M. Cavazza, F. Charles, “Search based planning: a method for character behaviour”, Proceedings of the 3rd on Intelligent Games & Simulation. Congreso GameOn-2002, Gran Bretaña.
- [5] D. Terzopoulos, “Artificial Life for Computer Graphics”, revista “Communications of the ACM”, agosto 1999, vol 42, num 8
- [6] T. F. Rabie, D. Terzopoulos “Active Perception in Virtual Humans”, Vision Interface (VI2000), Canadá
- [7] N. Badler, M. Palmer, T. Bindiganavale “Animation Control for real-time virtual humans”, “Communications of the ACM”, Agosto 1999, vol 42, num 8
- [8] Young, R.M. “Creating Interactive Narrative Structures: The Potential for AI Approaches”. AAAI Spring Symposium in Artificial Intelligence and Interactive Entertainment, AAAI Press, 2000.
- [9] Young, R.M. “An Overview of the Mimesis Architecture: Integrating Intelligent Narrative Control into an Existing Gaming Environment”. Working Notes of the AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment, AAAI Press 2001.
- [10] C. Elliott, J. Rickel, J. Lester, “Lifelike Pedagogical agents and affective computing: An explanatory synthesis”.
- [11] N. J. Nilsson, “Inteligencia artificial, una nueva síntesis”, capítulo 22, McGraw Hill. 2001.
- [12] R. Korf, “Real-time heuristic search”, Artificial Intelligence, 42:2-3, pp189-211. 1999
- [13] B. Bonet, H. Geffner, “Planning as heuristic search: new results” Proceedings of ECP'99, Durham, Inglaterra, 1999
- [14] F. Charles, M. Lozano, S. J. Mead, A. Fornés, M. Cavazza, “Planning formalisms and authoring in Interactive Storytelling”, TIDSE 2003