## Inteligencia Artificial

### Proyecto – Etapa 2

# *Desarrollo de Agentes Inteligentes*

# *Modelo de Agente BDI*

# Índice General:

# Introducción y Objetivos

El presente proyecto se presenta en el contexto de un juego de rol virtual (I.e., implementado en software).

Concretamente, el juego se desarrolla en un mundo virtual recreando una ciudad, con llanuras, bosques, ríos y montañas. El propósito de los agentes es apoderarse de la mayor cantidad posible de reliquias, y podrán conseguirlo ya sea explorando la ciudad para descubrirlos y levantarlos del suelo. La ciudad cuenta también con edificaciones. Principalmente se destacan las posadas, donde los personajes pueden descansar y reponerse de sus travesías ya que si están demasiado débiles, o incluso a desmayarse por falta de energía y perder todo lo que traen.

En el marco de este juego, el objetivo general del presente proyecto es el diseño e implementación de un agente inteligente para el control de un personaje jugador.

**El modelo de Agente BDI**

Un agente BDI es un tipo de agente racional que presenta actitudes mentales, concretamente

Creencias (Beliefs), Deseos (Desires), e Intenciones (Intentions). Este modelo tiene cierta base

filosófica, basándose en una teoría del razonamiento practico humano expuesta por Michael

Bratman ([Bra99]).

**Creencias**

Representan el estado de información del agente, es decir, su conocimiento sobre el entorno

(sobre si mismo y sobre otros agentes). Las creencias del agente pueden ir cambiando a través

del tiempo en base a sus percepciones, y como lo indica el termino creencias, \_estas pueden no

ser realmente verdaderas en el mundo actual, simplemente reflejan lo que el agente cree que

vale actualmente de acuerdo a lo percibido. Normalmente, esta información será guardada en

una base de datos (base de conocimiento), tanto mediante hechos (creencias primitivas) como

mediante reglas de inferencia, permitiendo encadenamiento hacia adelante para inferir nuevo

conocimiento (creencias derivadas).

**Deseos**

Representan el estado de motivación del agente, esto es, los objetivos o situaciones que el agente quisiera lograr o alcanzar. Ejemplos de deseos son: llegar a un cierto destino, tomarse vacaciones, o hacerse rico. Ciertos deseos del agente pueden estar en conflicto (es decir, no ser simultáneamente alcanzables), por ejemplo, ahorrar dinero y viajar, o ir a una fiesta y descansar.

**Intenciones**

Representan el estado deliberativo del agente, esto es, lo que el agente ha elegido hacer.

Usualmente las intenciones son un conjunto (consistente) de deseos que el agente se comprometió a alcanzar. Por simplicidad, para este proyecto se asumirá que el agente cuenta a lo sumo con una intención en cada momento.

**Planes**

Los planes son secuencias de acciones que un agente puede realizar para alcanzar una o más de sus intenciones (para este proyecto, la única intención actual del agente). Los planes pueden incluir otros planes, por ejemplo: un plan para conducir puede incluir un plan para encontrar las llaves del coche. Esto refleja que en el modelo de Bratman, los planes están inicialmente concebidos de forma parcial, entrando en detalle mientras su ejecución progresa.

**Actualización de Creencias**

El subprograma Actualizar Creencias se encarga de actualizar las creencias del agente en base a lo percibido y sus creencias previas. Este subprograma fue desarrollado para la primera etapa de este proyecto.

**Reconsideración de Intención actual**

Ciertas situaciones pueden motivar en el agente deseos de alta prioridad. Un deseo de alta prioridad amerita abandonar de inmediato lo que se está haciendo (intención actual) y comprometer esfuerzos en alcanzarlo. Por ejemplo, si el agente está peleando con otro y tras sufrir un daño considerable queda con muy poca energia, entonces descansar en una posada será un deseo de alta prioridad, que justica abandonar una búsqueda de reliquia. Luego, si el agente detecta un deseo de alta prioridad, entonces se establecerá dicho deseo como intención actual.

En caso de no existir deseo de alta prioridad, las situaciones especificadas a continuación conducirán al agente a razonar acerca de sus deseos y seleccionar una nueva intención a perseguir.

* No existe intención actual. Esto solo ocurre en el primer ciclo de ejecución del agente.
* La intención actual fue lograda.
* El plan para alcanzar la intención actual fue ejecutado por completo. Idealmente esto implicará que la intención fue lograda, quedando contemplado en el caso anterior. Sin embargo, esto no siempre será así, dado que el plan encontrado para alcanzar la intención podrá no ser correcto, o simplemente podrá no asegurar el logro de la intención. Por ejemplo, para anotar tres puntos en un partido de basket, puede seguirse el siguiente plan: dirigirse a una posición detrás de a la línea de triple, y lanzar en dirección al aro. Luego de ejecutar el plan, no necesariamente se habrá logrado la intención de anotar tres puntos.
* El plan para alcanzar la intención actual se tornó no factible. Por ejemplo, para alcanzar la intención de obtener cierta reliquia se cuenta con el siguiente plan: ir hasta la posición donde se encuentra la reliquia, y levantarla. Si mientras el agente se dirige a la posición de la reliquia percibe que otro agente la tomó, entonces el plan se torna no factible.

Luego, en cualquiera de estas cuatro situaciones el agente obtendrá el conjunto de deseos actuales según sus creencias, y seleccionara uno de los deseos del conjunto como próxima intención a perseguir.

Al establecer una nueva intención como intención actual (ya sea de alta prioridad o no) se establece también como plan actual aquel conteniendo a la nueva intención como única acción (de alto nivel). Como será explicado posteriormente al describir el planeamiento, las intenciones pueden considerarse como acciones de alto nivel, esto es, no ejecutables directamente por el agente (por ej., obtener una reliquia dada o saquear a un agente), en contraposición con las acciones primitivas, o directamente ejecutables (por ej., moverse a un nodo adyacente, pick\_up).

Finalmente, si nada de lo anterior ocurre, el agente continuará con la intención y plan actuales.

**Planeamiento y ejecución de planes**

De acuerdo a esta arquitectura, el agente modela internamente su capacidad de actuar sobre el mundo mediante el concepto de acciones con diferentes niveles de abstracción. En el extremo inferior, de menor nivel abstracción, se encuentran las acciones primitivas (como girar al norte o levantar un objeto dado), que el agente puede ejecutar de manera directa. En el extremo superior, de mayor nivel abstracción, se encuentran las intenciones del agente, consideradas acciones de alto nivel (por ejemplo, descansar u obtener un reliquia T dado), que a diferencia de las acciones primitivas, el agente no puede ejecutar de manera directa. Entre estos dos extremos pueden considerarse distintas acciones con niveles de abstracción intermedios. En adelante hablaremos de acción de alto nivel para referirnos a una acción no primitiva, y plan de alto nivel para referirnos a un plan que involucra acciones de alto nivel.

Bajo esta concepción de intenciones como acciones de alto nivel, un plan de alto nivel para alcanzar una intención I podrá ser aquel que tiene a I como única acción (de hecho, este es el caso del plan establecido al momento de seleccionar una nueva intención). Sin embargo, dado que el agente no puede ejecutar acciones de alto nivel en forma directa, si la próxima acción A1 del plan actual [A1; A2; :::; An] es de alto nivel, se necesitará hallar una secuencia de acciones de menor nivel [A1;1;A1;2; :::;A1:k], o (sub)plan, cuya ejecución equivalga al efecto de la acción de alto nivel A1 (planificación). Esa secuencia de acciones de menor nivel reemplazará a la acción de alto nivel en el plan original, obteniendo así un plan refinado, de menor nivel, [A1;1;A1;2;:::;A1:k;A2;:::;An]. El proceso se repite hasta que la próxima acción del plan sea primitiva, en cuyo caso se consume del plan y se ejecuta. A modo de ejemplo, supongamos que se establece obtener reliquia T como nueva intención, y por lo tanto el plan [obtener reliquia T] como plan actual. La próxima acción del plan actual, obtener reliquia T, es de alto nivel, y mediante planificación se obtiene un sub-plan equivalente: [ir a posición de T, levantar T]. Este subplan reemplaza a obtener reliquia T en el plan original, resultando en [ir a posición de T, levantar T]. Dado que la próxima acción del plan refinado (ir a posición de T) sigue siendo de alto nivel, entonces se planifica, obteniendo un sub-plan equivalente: moverse al nodo 104,

moverse al nodo 107, moverse al nodo 112 ]. Este sub-plan reemplaza a ir a posición de T en el plan [ir a posición de T, levantar T], resultando en [moverse al nodo 104, moverse al nodo

107, moverse al nodo 112, levantar T]. Finalmente la siguiente acción del plan actual (moverse

al nodo 104 ) es primitiva, de manera que se consume del plan, y se ejecuta.

**Implementación del Agente**

Se debe extender la implementación del agente modelo dotándolo de comportamiento racional considerable, de manera que se desenvuelva con éxito en el juego.

**Definición de los deseos considerados por el agente**

Los deseos que se consideraron en esta etapa fueron los siguientes:

* **Obtener Reliquias**

Agente desea poseer todas las reliquias posibles del suelo.

* **Obtener Pociones**

Agente desea poseer todas las pociones posibles del suelo que le permitirá obtener objetos ubicados dentro de tumbas que están cerradas.

* **Obtener Reliquias de Tumbas**

Agente desea poseer todas las reliquias que encierran todas las tumbas, siempre y cuando el agente tenga una poción en su poder para abrirla.

* **Explorar territorio desconocido**

El agente no desea quedarse quieto, por lo cual se mantiene en movimiento.

* **Moverse aleatoriamente**
* **Descansar (Deseo de alta prioridad):**

Si el nivel de energía del agente es menor a 100, éste querrá encontrar una posada para descansar.

**Requerimientos de comportamiento**

**Requerimiento 1:** recolección eficiente de todos los tesoros, es decir, tanto los que se encuentran en el suelo como los encerrados en tumbas.

Para este requerimiento el agente necesita desear:

* Obtener Reliquias [ desire (get([relic, Rname]) ] y Obtener Pociones [ desire(get([potion, Pname]) ]:

Necesario para lograr el comportamiento de juntar todos los tesoros y pociones tirados en el suelo. Para que se generen dichos deseos como precondición el agente debe saber la posición (At) de la reliquia/poción.

* Obtener Reliquias de Tumbas [ desire (get([grave, Gname]) ]:

Necesario para lograr que el agente abra tumbas, luego una vez abierta las reliquias que encerraba caen en el suelo y se activa el deseo (Obtener Reliquias). Para que se genere este deseo como precondición el agente debe saber que la tumba que va a abrir tiene reliquias y además que tiene al menos una poción en su poder.

Selección de la intención:

select\_intention(get(Obj), 'es el objeto más cercano de los que deseo obtener', Desires).

El agente va a seleccionar esta intención siempre y cuando pueda ya que es la que está más arriba en el código, es decir es la más alta en el orden de prioridad.

Para comenzar se obtienen las posiciones de todos los objetos que el agente desea ya sea tirados en el suelo o de las tumbas y se realiza una búsqueda para seleccionar como intención el más cercano en términos de costo.

Una vez que se encontró el más cercano se chequea si la energía del agente menos el costo de ir a obtener el deseo es mayor a 50.

**Plan:**

Se utiliza dos planes diferentes que depende si es de tipo tumba o de tipo reliquia/poción, los dos planes de alto nivel tienen en común que usan la acción de alto nivel goto(PosDest) que es necesaria para que el agente se mueva a una posición destino dada (PosDest).

Luego, cuando el agente está en la posición destino si es de tipo tumba(grave), se utiliza la primitiva cast\_spell/1 para abrirla.

Caso contrario, si es de tipo reliquia o poción, se utiliza la primitiva pickup/1 para agarrarla.

**Intención lograda:**

La acción de alto nivel utilizada goto(Pos) se logra cuando el agente se encuentra en Pos y get(Obj) se logra cuando el agente tiene el objeto que deseaba obtener.

**Requerimiento 2:** exploración completa del territorio

Para este requerimiento el agente necesita desear:

* Explorar territorio desconocido [ desire(goto(NodoId)) ]:

Para que el agente desee explorar como precondición debe tener más de 150 de energía y debe conocer un nodo con al menos un adyacente que no conozca.

Selección de la intención:

De todos los nodos desconocidos que el agente desea explorar se selecciona como intención explorar el más cercano.

Plan:

La intención de explorar no utiliza plan de alto nivel ya que usa directamente goto(NodoId) que solo hace una búsqueda para devolver un plan de las primitivas move/1.

**Intención lograda:**

Se logra cuando el agente se encuentra en Pos.

**Requerimiento 3:** administración segura de la energía y recarga:

Para este requerimiento el agente necesita desear:

* Descansar [desire(rest)]:

No le pusimos precondición el agente siempre va a desear descansar.

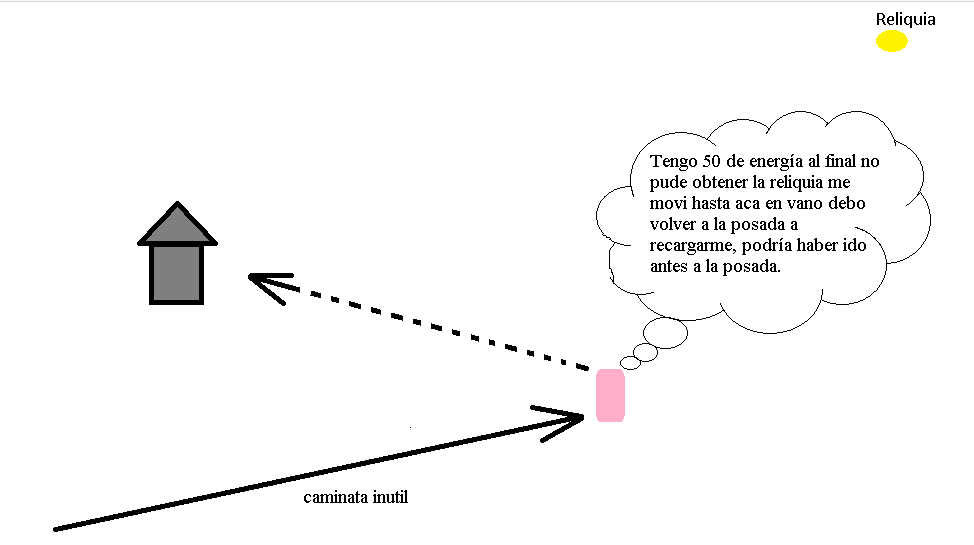
Además, esta como deseo de alta prioridad que tiene como precondición que es agente tenga menos de 50 de energía, y al ser de alta prioridad va a abortar cualquier plan que se esté ejecutando en ese momento.

Selección de la intención:

Si el deseo es de alta prioridad directamente se ejecuta no se selecciona.

Por otra parte, si no es de alta prioridad el deseo se va a seleccionar solo si el agente tiene una energía por debajo de 170 y no se logró seleccionar la intención de obtener tesoros/abrir tumbas porque puede pasar (además de que no hayan más tesoros) que la energía del agente menos el costo de ir a obtener el deseo de obtener el tesoro sea menor a 50.

Este control es necesario porque el plan de la intención de obtener tesoros/abrir tumbas puede ser abortado por el deseo de alta prioridad de descasar, que es seleccionado cuando el agente tiene menos de 50 de energía y no queremos que pase el siguiente escenario:



Plan:

Primeramente, se hace una búsqueda para encontrar la posición “Pos” de la posada que está más cerca del agente, luego el plan consiste en ir a dicha posición utilizando goto(Pos) y quedarse en la posada hasta que se logra la intención rest utilizando

*planify(stay, [null\_action , stay]) que*

Permite al agente permaner en la posada para que aumente su nivel de energía hasta que se logra la intención achieved(rest).

**Intención lograda:**

Se logra cuando la energía del agente es la máxima posible.

Aclaraciones:

Se modificó levemente el método buscar\_plan\_desplazamiento/3 ya que ahora tiene un parámetro más **buscar\_plan\_desplazamiento(+Metas, - Plan, -Destino, -Costo)** ,dicho parámetro de salida es el **costo** de energía que el agente va a gastar en moverse hasta la meta más cercana.Este parámetro se utiliza cuando se chequea vidaAgente – costo > 50 en la selección de intención de obtener tesoros / abrir tumbas.

Existe otro deseo que es que el agente se mueva aleatoriamente.