

Identificación del Trabajo	
Área:	Análisis de señales, modelados y simulación
Categoría:	Alumno
Regional:	Santa Fe

## Implementación de un agente basado en objetivos mediante estrategias de búsqueda y cálculo situacional

---

**Gómez José Ignacio, Hechim Pablo Nicolás, Paciuk Sebastián Federico**

*Universidad Tecnológica Nacional (Lavalse 610, Santa Fe), Facultad Regional Santa Fe, UTN.*

*E-mail de contacto: [nicolashechim@gmail.com](mailto:nicolashechim@gmail.com)*

*Este trabajo ha sido realizado bajo la dirección de la Dra. Milagros Gutiérrez y el Dr. Jorge Roa, en el marco de la asignatura "Inteligencia Artificial" de 5to año de Ingeniería en Sistemas de Información.*

### Resumen

---

La inteligencia artificial es una rama de las ciencias de la computación que estudia cómo agentes inteligentes se desenvuelven en un ambiente determinado para lograr conseguir un objetivo específico.

El presente trabajo se basa en una consigna de la cátedra Inteligencia Artificial, que consiste en el diseño e implementación de un agente inteligente para el control de un dron que tiene por objetivo encontrar la posición exacta de los victimarios en un sector específico de la ciudad de Santa Fe. Para alcanzarlo, el dron puede utilizar técnicas de cálculo situacional y búsqueda - informadas, en el caso de  $A^*$ , o no informadas en el caso de amplitud, profundidad y costo uniforme -.

Se estudian las diferencias más importantes entre estos algoritmos de resolución de problemas, al mismo tiempo que se diseña y construye un agente que brinde respuesta a la situación planteada.

**Palabras Claves:** Inteligencia Artificial; Agente inteligente; Búsqueda; Cálculo situacional

---

### 1. Introducción.

Un agente es un sistema que está situado en algún ambiente, y que es capaz de realizar acciones autónomas en ese ambiente, para cumplir sus objetivos de diseño <sup>[1]</sup>. Para que un agente sea inteligente debe ser proactivo, reactivo y tener habilidad social<sup>[2]</sup>. Aplica técnicas de Inteligencia Artificial (IA) para definir su comportamiento y establecer las acciones que puede llevar a cabo, de acuerdo a la circunstancia en la que se encuentra.

Se considera como ambiente a un sector de la ciudad de Santa Fe donde el agente dron (vehículo capaz de volar y de ser comandado a distancia) percibe señales emitidas por pequeños dispositivos que son disparados sobre personas por un sistema de alertas en un ambiente nano-radio. Estas alertas son accionadas como consecuencia de un hecho ilícito y ponen al agente en búsqueda del criminal que ha cometido el hecho, debiendo éste distinguir entre víctimas y victimarios para así encontrar al victimario en cuestión. Para ello, contará con una cámara que le permitirá identificar si una persona que tiene frente a él es víctima o victimario.

El agente se mueve en tres niveles de altitud: alto, medio y bajo. En el nivel alto y medio, el dron se mueve por encima de cualquier edificio, mientras que en el nivel bajo deberá moverse por encima de las calles, yendo de una esquina a la otra. Podrá conocer su ubicación (latitud,longitud) y altura por medio de un GPS.

Además, sólo podrá conocer la ubicación exacta de una persona cuando se encuentre en el nivel bajo, ya que de otro modo distingue sólo la potencia de señales provenientes de una determinada zona (cuyo alcance se incrementa en el nivel alto). Para percibir las señales, el dron utiliza un tercer sensor, que funciona a través de una antena que percibe las señales de los nano-radios.

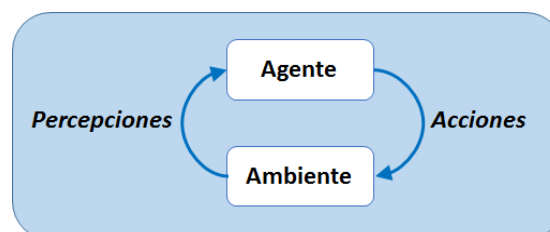
Para los niveles alto y medio, el dron sólo podrá utilizar una estrategia de búsqueda, mientras que para el nivel bajo puede escoger entre búsqueda o cálculo situacional. La diferencia entre estas dos estrategias radica en que el cálculo situacional utiliza un agente basado en conocimiento, que actúa según ciertas reglas que le permiten razonar, y de esa forma reaccionar a sus percepciones hasta poder llegar a su objetivo. Por otro lado, búsqueda es un método exploratorio, donde el agente puede realizar una determinada cantidad de acciones, y en base a las distintas secuencias de acciones elige aquella que le permite llegar a su objetivo; el criterio de elección depende de cuál algoritmo se utilice (amplitud, profundidad, costo uniforme o A\*).

## **2. Metodología.**

### **2.1. Técnicas de búsqueda.**

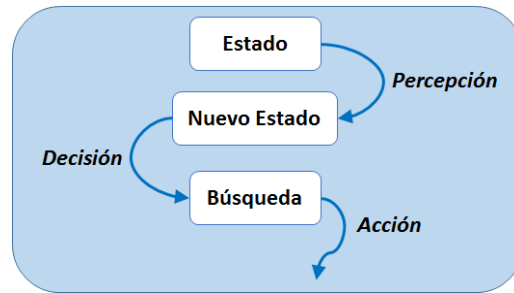
El agente percibirá el entorno a través de su cámara, GPS y antena, y decidirá qué acciones pertinentes debe llevar a cabo para distinguir y encontrar a un victimario.

El ambiente del agente será lo que el dron percibe y a quien responde, es decir, el ambiente le proveerá señales y será el dron quien le devuelva una acción, para seleccionar la acción, el agente utilizará una estrategia de búsqueda que le permite explorar los diferentes estados alcanzados al ejecutar acciones. Luego, si el agente encuentra un camino desde el estado inicial al estado objetivo, entonces la solución consistirá en seleccionar la primera acción de este camino. Se ilustra lo expresado en la Figura 1.



**Figura 1.** Interacción Agente – Ambiente

Se necesita que el agente realice un proceso de búsqueda cada vez que perciba señales del ambiente. Este proceso se representa en la Figura 2.

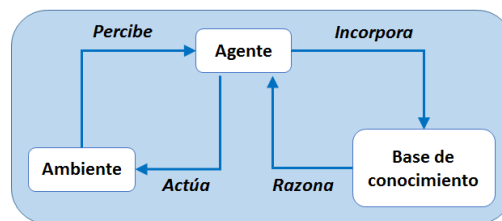


**Figura 2.** Proceso de decisión de acción

## 2.2. Técnica de cálculo situacional.

El agente percibirá el entorno a través de su cámara y GPS, y decidirá qué acciones pertinentes debe llevar a cabo para distinguir y encontrar un victimario. Notar que la percepción de antena que posee en la técnica de búsqueda no se utiliza, debido a que para cálculo situacional el dron se mueve únicamente en el nivel bajo.

El ambiente del agente será lo que el dron percibe. Incorporará conocimiento a partir de sus percepciones, y ejecutará una acción racional a partir de su base de conocimiento que modificará el ambiente. Se necesita que realice un proceso de cálculo situacional cada vez que perciba señales del ambiente. El comportamiento se ilustra en la Figura 3.



**Figura 3.** Interacción Agente - Ambiente

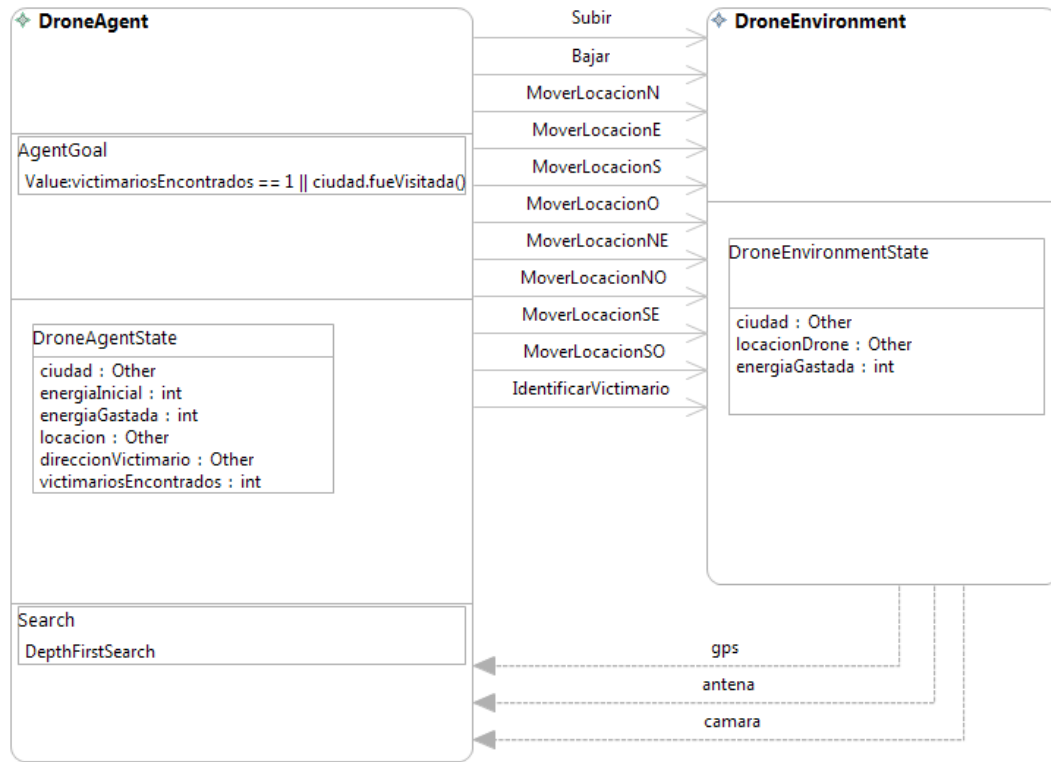
## 2.2. Marco de implementación.

Como marco de implementación para la resolución del problema, se utilizó lenguaje de programación JAVA SE 1.6, utilizando la herramienta Eclipse Galileo como entorno de desarrollo. Se utilizó un framework que permite realizar el modelo conceptual de un agente y, a partir de éste, generar las clases Java que representan el ambiente, el agente, y toda clase involucrada en su interacción (percepciones, acciones, entre otros). Dicho framework recibe el nombre de FAIA <sup>[3]</sup>. Por otro lado, se utiliza la herramienta IDEM-IA <sup>[4]</sup>, que permite la representación gráfica del modelo agente-ambiente; además, al estar integrado a FAIA, permite generar las clases del modelo directamente del diagrama.

Para la etapa de cálculo situacional se emplea un lenguaje de programación lógica: SWI-Prolog; mediante el mismo se representa la base de conocimiento del agente, y el conjunto de reglas que le permiten resolver el problema de buscar un victimario en el nivel bajo.

### 2.3. Modelado del agente y su ambiente.

En IDEM-IA se modela el estado del agente, las operaciones que este puede realizar, el estado del ambiente, las percepciones del agente y la función objetivo definida. La figura 4 muestra el modelo conceptual del agente definido para resolver el problema.



**Figura 4.** Diagrama IDEM-IA para el agente cuando éste utiliza búsqueda

Como se observa, se define DroneAgent y DroneEnvironment como el agente y su ambiente respectivamente.

Puede observarse que el drone deberá finalizar su búsqueda cuando haya encontrado al victimario o bien cuando haya recorrido toda la “ciudad” (que representa el mapa con los lugares que puede recorrer) y no lo haya encontrado.

Al utilizar cálculo situacional, esta situación no resulta necesaria ya que el drone contará con reglas que le permiten llegar al objetivo aún cuando no conoce el lugar preciso en que se encuentra el victimario. Recordemos, además, que en el cálculo situacional se utilizará sólo en el nivel bajo, por lo cual el drone ya habrá percibido las señales con la antena, y no es necesario percibir otra vez.

#### 2.4.1. Estado del agente.

*ciudad:* Estructura de datos que representa el mapa donde el drone puede moverse, y sólo posee los datos que éste conoce.

*energiaInicial:* Representa las unidades de energía inicial que dispone el drone.

*energíaGastada*: Representa el consumo total de energía acumulado.

*locación*: Representa la posición actual del drone dentro del mapa (ciudad).

*direcciónVictimario*: Cuando el drone se encuentra en el nivel bajo, indica si hay un victimario en alguna de las posiciones cardinales respecto a la posición actual del drone. En otros niveles existe, pero carece de valor utilizarlo.

*victimariosEncontrados*: Indica la cantidad de victimarios encontrados por el drone.

#### **2.4.2. Estado del ambiente.**

*ciudad*: Estructura de datos que representa el mapa donde el drone puede moverse. Contempla las intensidades, cantidades y ubicaciones de víctimas y victimarios.

*locaciónDrone*: Indica la ubicación actual del drone.

*energíaGastada*: Indica la energía gastada hasta el momento por el drone.

#### **2.4.3. Percepciones del drone.**

*GPS*: Permite al drone saber cuál es su ubicación actual.

*Antena*: Permite al drone identificar unívocamente las señales de los nano – radios. Dichas señales son percibidas con distinta intensidad según el nivel de altura en el que se encuentre el drone.

*Cámara*: Permite al drone identificar a un victimario en el nivel de altitud “bajo”. Le muestra en línea recta todas las calles desde la posición actual hasta el límite del cuadrante en el cual se encuentra.

#### **2.4.4. Operadores del drone.**

*Subir*: Acción que permite al drone desplazarse desde el nivel de altitud “bajo” a “medio” y desde “medio” a “alto”.

*Bajar*: Acción que permite al drone desplazarse desde el nivel de altitud “alto” a “medio” y desde “medio” a “bajo”.

*MoverLocación<DirecciónCardinal>*: Acción que permite al drone desplazarse hacia un cuadrante, subcuadrante o esquina adyacente.

*Identificar*: Acción que permite al drone diferenciar víctimas de victimarios cuando se encuentra en una esquina.

#### **2.4.5. Prueba de meta - Estrategias de Búsqueda.**

Se considera que se alcanzó el objetivo si se encuentra un victimario o se recorre toda la ciudad (o todas las esquinas que tienen señal):

```
victimariosEncontrados == 1 || ciudad.fueVisitada() == TRUE
```

Vale la pena destacar que la condición en que se considera la ciudad visitada, es introducida para dos casos: el primero es el caso en que el victimario haya salido del rango alcanzable por el drone y, por lo tanto, no exista victimario en las zonas que éste vigila. El segundo de los casos y tal vez más importante, es que en la etapa de búsqueda en la que el drone “piensa” cuál es la mejor acción a utilizar a continuación, el drone puede que no

conozca la posición del victimario; inicialmente todo lo que el drone percibe son señales de distintas intensidades según la zona y en la etapa de búsqueda no sabrá cuál es el lugar donde se halla el victimario, y por consiguiente no podrá encontrarlo.

#### 2.4.6. Heurística.

La heurística se utiliza en las estrategias de búsqueda A\*, y aquí se plantea como el cociente entre la energía disponible del drone y la intensidad percibida en su locación actual. Si el drone se encuentra en el nivel de altitud “bajo” y no hay victimario en ninguna de las direcciones posibles a tomar en línea recta, se sumará un “plus condicional” que actuará como penalización en la relación definida. En la Figura 5 se expresa la heurística definida.

$$h(n) = \frac{\text{energía disponible}}{1 + \text{intensidad percibida}} + \text{plus condicional}$$

**Figura 5.** Función Heurística para Búsqueda A\*.

#### 2.4.7. Definición de reglas diagnósticas.

Estas reglas se utilizan en cálculo situacional y permiten determinar propiedades del ambiente a través de las percepciones. Se definieron tres tipos de reglas diagnósticas, una que determina si hay un victimario en la esquina actual (a través de la Cámara), la otra si hay un victimario en alguna dirección cardinal (a través de la Cámara) y por último la posición actual del drone (a través del GPS).

Reglas expresadas en SWI-Prolog:

1. hayVictimario(Esq,S):-  
perception(\_,\_,\_,\_,\_,\_,\_,\_,1,Xpos,Ypos),esquina(Esq,Xpos,Ypos),actualSituation(S).
2. direccionVictimario(N,NE,E,SE,Su,SO,O,NO,PosAct,S):-  
perception(N,NE,E,SE,Su,SO,O,NO,PosAct,\_,\_),actualSituation(S).
3. posicion(Xpos,Ypos,S):-  
perception(\_,\_,\_,\_,\_,\_,\_,\_,Xpos,Ypos),actualSituation(S).

#### 2.4.8. Valoración de acciones

En cálculo situacional se valoran las acciones, y aquí se establecieron cuatro valoraciones:

*Excelente:* Se considera así a la acción que permite que DroneAgent identifique al victimario. Esto se da cuando se encuentra en una esquina con victimario.

*Muy buena:* Se considera así a la acción que permite que DroneAgent se desplace en una dirección en la que hay un victimario.

*Buena:* Se considera así a la acción que permite que DroneAgent se desplace hacia una esquina con personas pero sin victimario en esa dirección.

*Regular:* Se considera así a la acción que permite que DroneAgent se desplace hacia una esquina vacía y sin victimario en esa dirección.

#### **2.4.9. Axiomas de posibilidad**

Estos axiomas se utilizan en cálculo situacional y definen cuándo se puede ejecutar una acción. Se tienen dos tipos de axiomas:

*identificarVictimario*: si el drone se encuentra en una esquina donde hay un victimario, es posible llevar a cabo esta acción.

*moverLocacion*: si el drone quiere moverse en una dirección en la que hay una esquina adyacente y cuenta con la energía necesaria que insume este movimiento, es posible llevar a cabo esta acción.

#### **2.4.10. Axiomas de estado sucesor**

Estos axiomas se utilizan en cálculo situacional y permiten agregar nuevos conocimientos a partir de las acciones que ejecuta el drone. Se tienen los siguientes:

1. Para mantener de situación en situación el valor de la variable “hayVictimario” en una esquina.

2. Para aumentar de situación en situación la variable “cantidadVictimarios” si se identificó un victimario en la última acción.

3. Para mantener de situación en situación la variable “cantidadVictimarios” si no se identificó un victimario en la última acción.

4. Para cambiar de posición si el drone se mueve.

5. Para mantener la posición si el drone no se mueve.

#### **2.4.11. Prueba de meta - Estrategia de Cálculo Situacional**

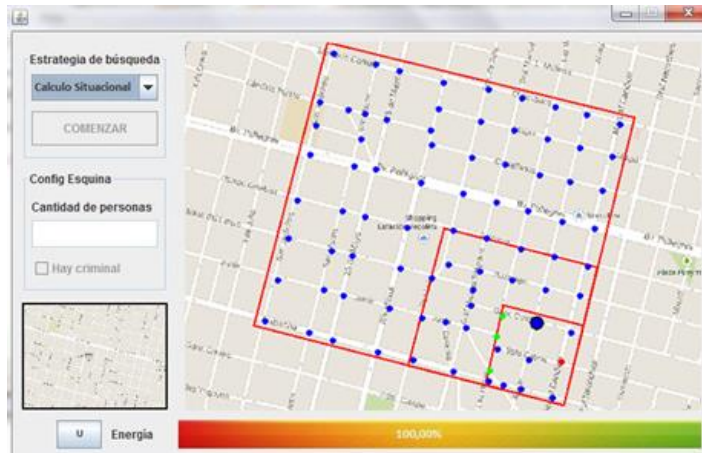
Se considera que se alcanzó el objetivo si se encuentra un victimario:

`goalReached(S):- cantidadVictimarios(N,S), N > 0.`

#### **2.5. Interfaz de usuario**

El usuario puede elegir la estrategia de búsqueda que quiera utilizar: cálculo situacional, A\*, profundidad o anchura. Cada esquina es configurable, puede elegirse la cantidad de personas que estén y determinar si habrá o no un victimario en ella. También se exhibe una barra que representa la cantidad de energía disponible del drone.

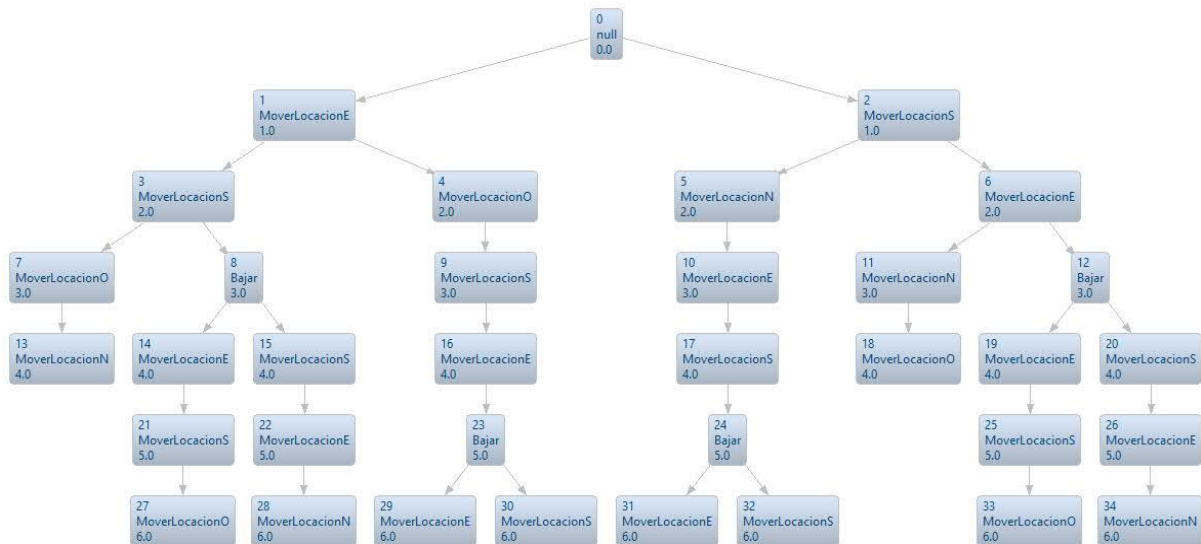
A continuación, en la Figura 6 se muestra un ejemplo de un escenario inicializado donde la posición inicial del drone se representa con el círculo azul más grande; las esquinas sin señal con puntos azules; las esquinas con señal con puntos verdes; las esquinas con señal y victimarios con puntos rojos. En el escenario hay un gran cuadrado de color rojo que indica los límites dentro de los cuales podrá moverse el drone. Para indicar el nivel de altitud “alto” se divide el cuadrado más grande en cuatro zonas. Luego para el nivel de altitud “medio” se divide cada zona en cuatro generando un total de dieciséis zonas. Y por último, el nivel de altitud “bajo”.



**Figura 6.** Interfaz de usuario con un escenario inicializado

### 3. Resultados

#### 3.1. Búsqueda

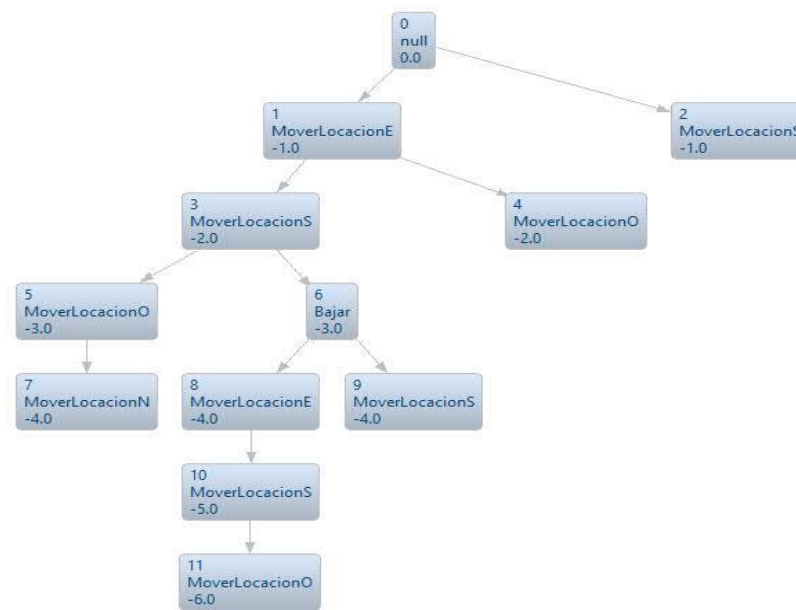


**Figura 7.** Estructura del árbol de búsqueda generado con la estrategia de amplitud.

Se logra que con todas las estrategias de búsqueda utilizadas, estas son: profundidad, amplitud, costo uniforme y A\*, el dron llegue a su objetivo. En la figura 7 puede observarse el árbol de búsqueda inicial generado por la estrategia de búsqueda en amplitud para una determinada situación. Puede verse cómo los operadores se ejecutan de izquierda a derecha en el orden en que son seteados en el agente: MoverLocacionN, MoverLocacionE, MoverLocacionS, MoverLocacionO, Bajar, Subir. Vemos además en la estructura especial del árbol dado que los nodos se expanden en orden horizontal nivel a nivel.

Con búsqueda en profundidad el escenario genera un árbol inicial como el de la figura 8 y expande verticalmente analizando primero el nodo más profundo que queda por expandir.





**Figura 8.** Estructura del árbol de búsqueda usando estrategia de profundidad.

Con las estrategias de costo uniforme y de A\*, la búsqueda se restringe; en el caso de costo, sólo expandirá aquellos nodos que queden por expandir cuyo costo acumulado sea menor que el menor costo hallado hasta el momento. En el caso de A\*, además del costo considera un valor heurístico que estima el costo que falta para llegar al objetivo.

### 3.2. Cálculo Situacional

El escenario de prueba utilizado corresponde al de la Figura 6. Seguimos las acciones tomadas por el drone para apreciar cómo el drone responde a la evaluación de acciones propuesta: inicialmente puede ir hacia el E, S y O. Según el orden de las reglas de estado sucesor, debería ir al E. Sin embargo, elige ir hacia el O porque en esa esquina hay señal y esta acción está considerada como una buena acción, mejor que moverse hacia una esquina sin personas que se considera acción regular. En la situación siguiente elige ir al S por la misma razón; la otra opción es ir al E pero no hay personas. Luego va hacia el S de nuevo; ¿por qué ejecuta esta acción si ir a una esquina sin señal es “peor” que ir a una que sí tiene, como la esquina al N de su posición actual? Porque el drone tiene una regla que le indica que no puede volver a pasar por una esquina ya visitada. Entonces, su único camino es hacia el S. Aquí el drone decide en base a su conocimiento ir hacia el Este ya que las reglas le indican que esa es la mejor acción a tomar. Como ha percibido a través de la cámara que hacia el E encontrará un victimario en alguna de las esquinas de esa dirección, decide aventurarse en ese camino. Ejecuta esta acción considerada muy buena y mejor que ir al S (acción buena). Así es como primero se mueve al E y luego, por el mismo motivo, al E nuevamente. Aquí, percibe que en su posición actual hay un victimario. Luego, identifica al victimario, dado que esta es una acción considerada excelente.

En la implementación de este tipo de estrategia se obtuvieron mejores resultados que con búsqueda, ya que el proceso de unificación de predicados es más veloz que armar árboles sucesivos para tomar una decisión.

#### **4. Discusión**

Como en el nivel bajo del mapa existen muchas conexiones entre las esquinas dentro de una determinada zona, encontramos que resulta más conveniente utilizar cálculo situacional en ese nivel, dado que el drone se maneja según ciertas reglas que le permiten inferir en qué lugar se encuentra (y en qué lugar no) el victimario. Esto permite llegar rápido a una respuesta; en contraposición, búsqueda genera un árbol de búsqueda muy grande ya que explora muchos caminos y combinaciones de movimientos entre las esquinas. Ambos permiten encontrar la solución a un mismo problema, pero cálculo situacional permite hacerlo en forma más eficiente (computacionalmente hablando). Por otro lado, búsqueda resulta muy buena para utilizar en los niveles medio y alto, ya que en estos niveles el drone debe desplazarse entre las diferentes zonas del mapa, explorando y percibiendo en qué zonas existen señales y cuál es su intensidad. Es decir, en aquellos niveles en que resulta necesario y beneficioso explorar, búsqueda resulta más útil que cálculo situacional.

#### **5. Conclusión**

El uso de IDEMIA y FAIA para el modelado como para la implementación ha facilitado el desarrollo permitiendo al equipo de trabajo centrarse en los aspectos relevantes del diseño del comportamiento del agente. En tanto, el drone logra alcanzar el objetivo para los escenarios propuestos con las técnicas de búsqueda y cálculo situacional. En cuanto a las implementaciones, la estrategia de búsqueda forma un árbol en cada situación y en cada nodo se fija cuál es la mejor opción para ir a cualquier nodo adyacente. En cálculo situacional, el agente percibe, decide y acciona según lo que crea más conveniente, almacenando la información de la situación actual en una base de conocimiento. En otras palabras, con búsqueda en cada paso que el drone da, debe recalcular de cero cuál es el mejor camino a la solución y en cálculo situacional el drone puede ir almacenando cierta información que le resulta útil para llegar más rápido al objetivo. Otra diferencia importante que encontramos, es que en búsqueda el drone explora; ejecuta todos los operadores que puede y decide cuál es la secuencia de acciones que lo lleva a su objetivo, y ejecuta la primera de ellas; en cálculo situacional, el drone elige la próxima acción a ejecutar basado en la situación actual, los conocimientos previos, sus reglas de operación y un ranking de las posibles acciones a ejecutar. Teniendo en cuenta esto y el tiempo de implementación de las técnicas, nos resultó más conveniente utilizar cálculo situacional.

#### **Bibliografía**

- [1] Russell, S. & Norvig, P. (2010). Artificial Intelligence: A Modern Approach. Prentice-Hall, 3rd edition.
- [2] Nilsson, N. J. (1998) Artificial Intelligence: A New Synthesis. Morgan Kaufmann ed.
- [3] Roa, J., Gutiérrez, M. y Stegmayer, G. (2009) FAIA: Framework para la enseñanza de agentes en IA. IE Comunicaciones: Revista Iberoamericana de informática educativa. No 7/8, ISSN 1699-4574, Pp. 43-56.
- [4] Santana, W., Roa, J., Gutiérrez, M., Stegmayer, G. (2012). IDEM-IA: Un entorno de desarrollo integrado para el modelado de agentes inteligentes. IE Comunicaciones: Revista Iberoamericana de informática educativa. Nro 13, ISSN 1699-4574, Pp. 43-56.