



INTELIGENCIA ARTIFICIAL

Proyecto N° 2

Agentes y Búsqueda

Segundo Cuatrimestre de 2007

Ubicada al norte del reino de Ardos, se halla una comarca con ríos profundos, puentes, montañas y bosques encantados. En algún lugar de la comarca se erige un castillo en el que vive el rey ardano. Dispersos por la comarca se ubican ejércitos del reino y ejércitos enemigos.

Un mensajero de un reino aliado, montado en su caballo, debe llegar al castillo lo antes posible para entregarle una importante carta al rey. El mensajero posee un mapa de la comarca que le servirá para encontrar un camino que lo conduzca a su destino. Además, en caso de necesitar atravesar territorio donde se ubica un ejército enemigo, lleva consigo un uniforme del bando contrario para disfrazarse y pasar desapercibido.

El propósito del proyecto es modelar el problema considerando que el mensajero es un agente que se mueve en la comarca (su entorno) y su objetivo es llegar al castillo en la menor cantidad de tiempo posible. Antes de lanzarse en su travesía, el agente empleará métodos de búsqueda (vistos en la materia) para hallar un camino en el mapa que conduzca hacia el castillo.

Para el presente proyecto deberá implementar:

- **en Prolog**, dos agentes mensajeros, uno que emplee búsqueda (ciega) en profundidad para hallar en camino en el mapa y el otro búsqueda (heurística) A^* . Además de los agentes, se deberá implementar un simulador de entorno (comarca) para interactuar con los mismos.
- **en Java**, una interfaz gráfica que permita (mínimamente) ingresar una configuración de la comarca (incluyendo la ubicación del castillo, ejércitos del reino y ejércitos enemigos), seleccionar un agente (DFS o A^*), definir la posición inicial del mensajero y correr la simulación mostrando apropiadamente los resultados de la misma.

La comarca (entorno)

La comarca será representada mediante una grilla de n filas (numeradas de 0 a $n - 1$) por m columnas (también numeradas de 0 a $m - 1$), como se muestra en la figura 1. Cada celda de la grilla puede contener alternativamente solo pasto, un bosque, montañas, o agua. Las celdas de agua pueden disponerse una al lado de la otra conformando ríos (o lagos).

Sobre una celda de agua puede haber un puente. Los puentes permiten cruzar una celda de agua desde una orilla (celda) de tierra a otra y estos **no** pueden ser alineados para conformar puentes de mayor longitud. Un puente puede estar sano, deteriorado o roto, y en este último caso el puente no puede ser cruzado. Por otro lado, sobre una celda de pasto o montaña puede haber un ejército del reino o enemigo. Por último, una de las celdas de pasto contendrá el castillo del rey.

Observación: Para evitar que el agente se desplace fuera de la grilla, se asume que las celdas del borde son siempre celdas de bosque. En otras palabras, la comarca siempre estará rodeada de bosques.

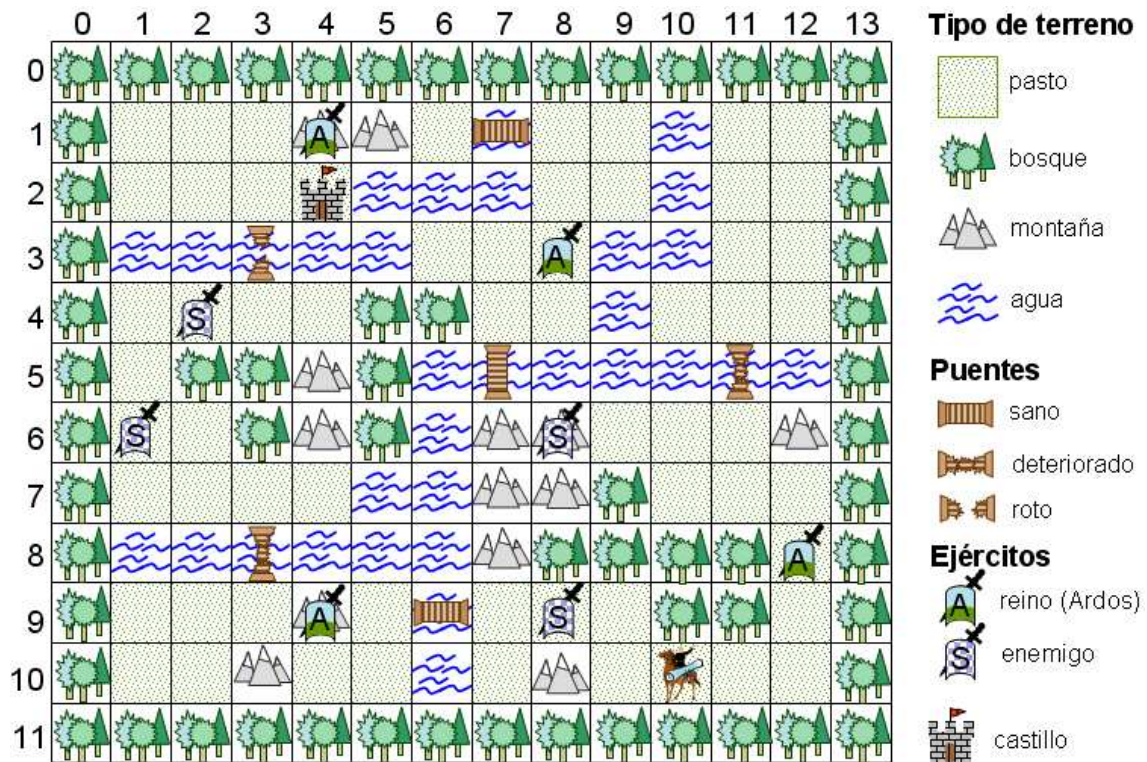


Figura 1: Comarca

Convenciones de Representación (en PROLOG)

En primer lugar, especificaremos como hacer referencia a una posición o celda de la grilla. Utilizaremos el par $[F, C]$ (lista de dos elementos) para hacer referencia a la celda de la grilla ubicada en la fila F y columna C .

Finalmente, la configuración de la comarca será representada mediante hechos de la forma $celda(Pos, Terreno, Contenido)$, donde Pos representa una posición en la grilla, $Terreno \in \{pasto, agua, mont, bosque\}$ y $Contenido \in \{puente(Estado), ejercito(Bando), castillo, -\}$, donde a su vez $Estado \in \{sano, det, roto\}$ y $Bando \in \{reino, enemigo\}$. Debe recordarse que los puentes solo pueden ubicarse en celdas de agua, los ejércitos sólo en celdas de pasto o montaña y el castillo solo en una celda de pasto. El '-' se emplea para aquellas celdas que no contienen ni puente, ni ejército, ni el castillo.

Percepciones y Acciones

El agente es capaz de realizar una serie de acciones y cada vez que ejecuta una acción percibe algún estímulo por parte del entorno. En cada instante de la simulación el agente se encuentra en una posición/celda determinada. En todo momento a lo largo de su travesía el agente puede saber su ubicación actual y percibir lo que hay en las celdas que lo rodean. Es decir, en cada ciclo de la simulación, el agente percibe del entorno su posición actual (coordenadas de la celda en la que se encuentra), además de una descripción del contenido las 4 posiciones adyacentes.

El agente puede avanzar hacia cualquiera de los cuatro puntos cardinales (norte, sur, este y oeste) y puede hacerlo a tres diferentes velocidades: *al galope*, *al trote* y *a paso de hombre*. Además, el agente puede colocarse el disfraz (para poder avanzar sobre una celda donde se

ubica un ejército enemigo) o quitárselo (para poder avanzar sobre una celda donde se ubica un ejército del reino, sin ser confundido con el enemigo).

Efectos de las acciones

El agente no puede atravesar una celda de agua, salvo que dicha celda tenga un puente. Si el agente avanza sobre una celda de agua sin puente entonces muere ahogado. De la misma forma, si el agente avanza sobre una celda de agua con un puente roto, entonces cae al agua y muere ahogado. Tampoco puede avanzar sobre un bosque, ya que estos están encantados, y de hacerlo también moriría. Para los restantes tipos de terreno, el agente puede avanzar sobre ellos, pero debe regular su velocidad para atravesarlos con éxito, como se especifica a continuación. El agente puede avanzar sobre una celda de pasto a cualquier velocidad. Sin embargo, el agente puede avanzar a lo sumo al trote sobre un puente sano y apenas a paso de hombre sobre un puente deteriorado o sobre una celda de montañas. Si el agente fuera más rápido de lo indicado en cada caso, entonces se resbalaría y moriría. Finalmente, independientemente de la velocidad de avance, si el agente avanza sin disfraz sobre una celda con un ejército enemigo, o disfrazado sobre una celda con un ejército del reino, entonces morirá.

Convenciones de Representación

Se deberán utilizar los siguientes términos PROLOG para representar las percepciones:

- `perc(PosAct, AlNorte, AlSur, AlEste, AlOeste)`, donde `PosAct` representa la posición actual del agente, y `AlNorte`, `AlSur`, `AlEste` y `AlOeste` describen el contenido de las celdas adyacentes a la posición actual. Concretamente, estas descripciones son listas de la forma `[Pos, Terreno, Contenido]`, donde `Pos` es la posición de la celda adyacente, `Terreno` $\in \{\text{pasto}, \text{agua}, \text{mont}, \text{bosque}\}$ y `Contenido` $\in \{\text{puente(Estado)}, \text{ejercito(Bando)}, \text{castillo}, -\}$, donde a su vez `Estado` $\in \{\text{sano}, \text{det}, \text{roto}\}$ y `Bando` $\in \{\text{reino}, \text{enemigo}\}$.
- `fin_simulacion(Razon)`, donde `Razon` $\in \{\text{muerto}, \text{trompetas}, \text{abandono}\}$. Esta percepción indica al agente que terminó la simulación, ya sea porque murió, llegó al castillo (trompetas) o decidió abandonar (ver acciones).

Se deberán utilizar los siguientes términos PROLOG para representar las acciones:

- `avanzar(Direccion, Velocidad)`, donde `Direccion` $\in \{\text{norte}, \text{sur}, \text{este}, \text{oeste}\}$ y `Velocidad` $\in \{\text{al_galope}, \text{al_trote}, \text{a_paso_hombre}\}$.
- `poner_disfraz` y `quitar_disfraz`, efectuadas por el agente para colocarse y quitarse el disfraz, respectivamente.
- `abandonar`, una acción que puede realizar el agente para abandonar su travesía hacia el castillo.

El mensajero (agente) y búsqueda en el mapa

Apenas comenzada la simulación, y antes de efectuar su primera acción, el agente efectuará una búsqueda en el mapa para hallar un camino (secuencia de celdas) que lo conduzca al castillo. El mapa, que forma parte del estado interno del agente, será representado mediante una grilla con las mismas dimensiones que la grilla representando la comarca, e indica el tipo de

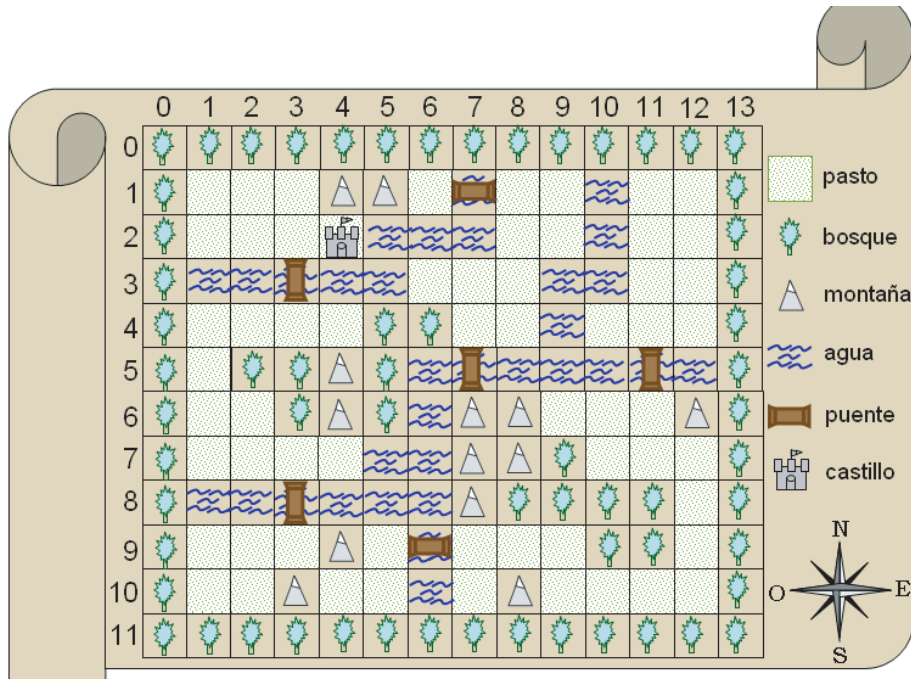


Figura 2: Mapa de la Comarca

terreno de cada celda de la comarca (pasto, agua, montaña o bosque), la posición del castillo y hasta la ubicación de los puentes (ver figura 2). No obstante, como es de esperarse, el mapa no indica el estado de los puentes, ni la ubicación de los ejércitos. Frente a la falta de información acerca de los puentes, el agente asumirá (de forma imprudente) que todos están sanos, y en consecuencia, el camino hallado por el agente mediante búsqueda no siempre será viable en la práctica. Concretamente, si el camino hallado en el mapa atraviesa un puente que en realidad se encuentra roto, entonces el agente no podrá llegar al castillo por dicho camino.

Luego de haber obtenido un camino (tentativo) hacia el castillo mediante búsqueda, el agente comenzará a ejecutar acciones sobre el entorno para efectivamente seguir dicho camino a través de la comarca. De esta forma, para decidir la próxima acción a realizar en cada ciclo de la simulación, el agente debe tener en cuenta:

- las coordenadas de la celda a la que debe moverse a continuación de acuerdo al camino hallado en el mapa.
- la presencia o ausencia de un ejército en dicha celda, y en el caso de haber uno, considerar si debe cambiarse o no de atuendo.
- por último, si cuenta con la ropa apropiada (o no hay ejército), entonces debe avanzar sobre la celda. Para ello tendrá en cuenta el tipo de terreno o estado del puente en dicha celda, percibido del entorno. En base a esta información el agente deberá determinar la velocidad de avance óptima (lo más veloz posible sin resbalar). En caso de percibir un puente roto, el agente abandonará la simulación (acción abandonar).

Representación del mapa en Prolog

El mapa será representado mediante hechos de la forma `mapa(Pos, Terreno, Contenido)`, donde `Pos` representa una posición en la comarca, `Terreno` $\in \{\text{pasto}, \text{agua}, \text{mont}, \text{bosque}\}$ y `Contenido` $\in \{\text{puente}, \text{castillo}, -\}$.

Acerca de la búsqueda A^*

El agente que implemente la búsqueda heurística A^* deberá considerar que el costo de avanzar sobre una celda de pasto es de 1 minuto, ya que sabe que puede hacerlo al galope, el costo de avanzar sobre un puente es de 2 minutos ya que debe pasarlos al trote y el costo de avanzar sobre una celda de montañas es de 3 minutos, ya que debe hacerlo a paso de hombre. Recordar que el agente no sabe el estado de los puentes a partir del mapa, y por ello asume que todos los puentes están sanos. Como heurística debe considerarse la *distancia de Manhattan* entre la posición considerada y la posición del castillo.

Requerimientos para la Interfaz

La interfaz debe presentar al usuario una representación gráfica de la comarca al estilo de la mostrada en la figura 1, permitiéndole definir (y en lo posible modificar) la disposición de ríos, montañas, bosques, puentes (y el estado de los mismos), tropas del reino y enemigas, y la ubicación del castillo. Además, la interfaz debe presentar una representación gráfica del mapa de la comarca, al estilo de la mostrada en la figura 2. El mapa deberá ser generado automáticamente a partir de la configuración de la comarca ingresada. Finalmente, debe brindar la posibilidad de seleccionar el tipo de agente (DFS o A^*), definir su ubicación inicial e iniciar la simulación.

La interfaz deberá mostrar claramente los siguientes datos asociados a la simulación:

- la frontera y el nodo seleccionado en cada paso de la búsqueda efectuada por el agente, además de la solución hallada (camino al castillo) y el costo de la misma.

Observación: por claridad, mostrar solo la posición asociada a cada nodo.

- la secuencia de acciones efectuadas por el agente mensajero durante la simulación.

Importante: para facilitar la prueba del proyecto, la interfaz debería permitir, luego de haber ingresado una configuración de la comarca, efectuar varias pruebas distintas sobre dicha configuración (variando el agente y la posición inicial del mismo).

Otras convenciones y requerimientos

- **Predicados PROLOG a implementar.** Deberán desarrollarse una serie de predicados para implementar los agentes y el simulador de entorno. El agente que realice búsqueda en profundidad deberá implementarse mediante un predicado `dfs/2`. El agente que realice búsqueda A^* deberá implementarse mediante un predicado `astar/2`. Para ambos predicados, el primer argumento representará una percepción recibida por parte del entorno y el segundo la próxima acción que el agente desea realizar. Deberá implementarse un predicado `go/2` que inicie la simulación de la interacción entre un agente y el entorno. El primer argumento representará la posición inicial del mensajero y el segundo, el nombre del agente. Por ejemplo:

`?-go([10,10],dfs).`

comenzará una simulación con el agente implementado por el predicado `dfs/2`, ubicado en la posición `[10, 10]`.

- **Conocimiento del agente.** La única información que posee el agente es la obtenida a través de percepciones recibidas del entorno. No puede (ni debe) obtener información de otra manera. Por lo tanto, LA IMPLEMENTACIÓN DEBE RESPETAR ESTA RESTRICCIÓN.

- **Orden de Generación de Vecinos.** En las búsquedas efectuadas sobre el mapa, se asume que los nodos vecinos de un nodo N se generan siguiendo el orden norte, este, sur, oeste. Es decir, el nodo correspondiente a la posición que se halla al norte de (la posición asociada a) N queda como hijo izquierdo de N en el árbol de búsqueda.

Condiciones de Entrega

Para el presente proyecto se establecen las siguientes condiciones de entrega:

1. Aparte de los programas implementados, se deberá realizar un informe completo que explique los programas realizados y reseñe las decisiones de diseño e implementación tomadas, así como posibles restricciones en el uso de los programas. El informe debe incluir ejemplos de corridas del programa y también un listado de los principales programas implementados. Finalmente debe incluir una sección que explique detalladamente los pasos requeridos para ejecutar la interfaz suministrada.
2. Las comisiones pueden estar conformadas por hasta 2 integrantes (recordar que éstos deben estar previamente registrados con la cátedra).
3. La fecha límite de entrega del presente proyecto es el día lunes 22 de Octubre de 2007 en el horario de clase. Los proyectos entregados fuera de término recibirán una penalización en su calificación, la cual será proporcional al retraso incurrido.
4. Deberá presentarse un folio plástico CERRADO (no entregar carpetas) conteniendo los siguientes elementos:
 - El informe impreso, que deberá estar encabezado por una carátula identificando claramente a los integrantes de la comisión.
 - Un disquete o cdrom conteniendo:
 - a) Una carpeta **Prolog**, que contenga el código PROLOG de los predicados implementados. En un archivo “comarca.pl” incluir todos los predicados que implementan el entorno. En un archivo “agentedfs.pl” incluir todos los predicados que implementan al agente dfs. En un archivo “agenteastar.pl” incluir todos los predicados que implementan el agente astar.
 - b) Una carpeta **Java**, que contenga el código JAVA de la interfaz implementada.
 - c) Una carpeta **Test**, que contenga todos los archivos requeridos para ejecutar la interfaz implementada, entre ellos el archivo ejecutable de la interfaz (.jar).
5. EN LA EVALUACIÓN DEL PROYECTO SE TENDRÁ EN CUENTA QUE LA IMPLEMENTACIÓN RESPETE TODAS LAS CONVENCIONES ESTABLECIDAS. SE RECOMIENDA PRESTAR ATENCIÓN A DICHAS CONVENCIONES.