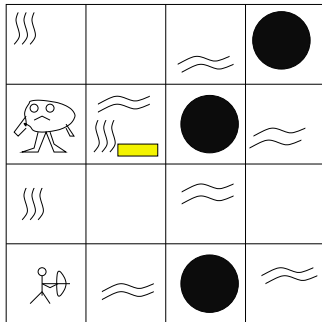


# El mundo de Wumpus:

---

El mundo de Wumpus:



↑  
Salida



# Descripción PAGE del agente cazador

---

Percepciones (restringidas a la casilla que ocupa):

- ▶ El agente percibe si en su casilla se encuentra el wumpus
- ▶ En los cuadros adyacentes al wumpus, percibe su hedor
- ▶ En los cuadros adyacentes a un pozo, percibe la brisa
- ▶ Donde está el oro, percibe su brillo
- ▶ Si avanza hasta un muro, percibe el choque
- ▶ Cuando mata al wumpus, percibe un grito

La percepción es representable por cinco símbolos  
(hedor,brisa,brillo,choque,grito)

# Descripción PAGE del agente cazador




---

- ▶ Acciones:
  - ▶ Avanzar,
  - ▶ Girar 90° grados a izq. o der.,
  - ▶ Lanzar una flecha (la flecha llega hasta el wumpus o a la pared), y
  - ▶ **salir** (si se encuentra en la casilla de salida)
- ▶ El agente muere si entra en un pozo, u ocupa una casilla en la que está un wumpus vivo
- ▶ Objetivo: encontrar el oro y volver a la salida lo más rápidamente posible (vivo, claro)

# Ejecución

---

Supongamos que el agente ha visitado ya las casillas (1,1), (1,2) y (2,1), encontrándose en esta última:

?	?	?	?
?	?	?	?
		?	?
			?

↑  
Salida

# Representación y funcionamiento

---

- ▶ La sucesión de percepciones ha sido (con respecto a (hedor,brisa,brillo,choque,grito)):  
(0,0,0,0,0), (0,1,0,0,0), (1,0,0,0,0)
- ▶ Si el agente cazador es un **agente basado en conocimiento**, su comportamiento estará determinado por la base de conocimiento de que dispone.
- ▶ La base de conocimiento contendrá un **conocimiento inicial**, pero su contenido irá variando con el funcionamiento del agente.
- ▶ El bucle de funcionamiento básico será:
  1. **Percibir** y añadir la información obtenida a la base de conocimiento.
  2. **Deducir** qué acción llevar a cabo, de acuerdo con el conocimiento disponible.
  3. **Llevar a cabo la acción seleccionada** y añadir esta información en la base de conocimiento.

## Representación y funcionamiento (II)

---

Tras visitar las casillas (1,1), (1,2) y (2,1) en la base de conocimiento tiene (entre otras) la siguientes fórmulas:

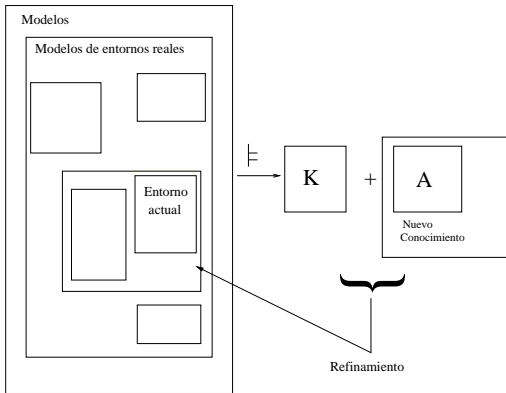
$$\underbrace{\begin{cases} \neg H_{11} & \neg B_{11} \\ \neg H_{21} & B_{21} \\ H_{12} & \neg B_{12} \end{cases}}_{\text{Percepciones}} + \underbrace{\begin{cases} \neg H_{11} \rightarrow \neg W_{11} \wedge \neg W_{12} \wedge \neg W_{21} \\ \neg H_{21} \rightarrow \neg W_{11} \wedge \neg W_{21} \wedge \neg W_{22} \wedge \neg W_{31} \\ \neg H_{12} \rightarrow \neg W_{11} \wedge \neg W_{12} \wedge \neg W_{22} \wedge \neg W_{13} \\ H_{12} \rightarrow W_{13} \vee W_{12} \vee W_{22} \vee W_{11} \end{cases}}_{\text{Conocimiento básico}}$$

¿Puede deducir el agente que Wumpus está en  $W_{13}$ ?

# Conocimiento y modelos posibles

Dada una base de conocimiento  $K$ , hay que considerar tres clases de modelos:

1. Los modelos lógicos de  $K$ .
2. Los modelos que representan entornos *reales*.
3. El modelo que representa al entorno en el que está el agente.



# Percepción nueva = refinamiento de la clase de modelos

---

- ▶ Cada nuevo conocimiento percibido refina el conjunto de modelos.
- ▶ Si los modelos del tipo (2) se reducen a uno, entonces tenemos conocimiento perfecto.
- ▶ Este proceso se puede *localizar* para una propiedad concreta:
  - ▶ Si con respecto a una propiedad concreta, sólo hay un modelo de tipo (2), entonces está determinada.



# Modelos posibles para $K$ después del segundo movimiento

---

El agente no sabe en cuál de estos mundos está:

?	?	?	?
?	?	?	?
		?	?
⊙	≈	●	?

↑  
Salida

?	?	?	?
?	?	?	?
	●	?	?
⊙	≈		?

↑  
Salida

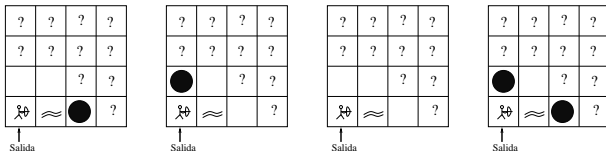
?	?	?	?
?	?	?	?
	●	?	?
⊙	≈	●	?

↑  
Salida

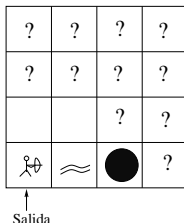
# Refinamiento

**Razonamiento:** refinar el conjunto de *mundos posibles*.

- Ahora sabe que en (2,2) no hay pozo: Los modelos de esta propiedad son:



- Refina su conocimiento: está en el mundo:



# Esquemas de razonamiento

---

- ▶ Esquema de razonamiento deductivo (algunas ideas):
  - ▶ Si *demuestra* que una casilla esta libre de peligro, entonces puede visitarla para percibir.
  - ▶ Si *demuestra* que  $W_{i,j}$ , entonces dispara.
  - ▶ Si *demuestra* que tiene un pozo la evita.
- ▶ Pero, ¿Qué hace si no demuestra que una casilla es segura?
  - ▶ Opción cauta: elegir las casillas *demostrablemente seguras*.  
Problema: puede quedarse bloqueado.
  - ▶ Opción atrevida: elegir, si es necesario, entre las casillas que **no** son *demostrablemente peligrosas*.  
Problema: puede morir.
- ▶ Problema: ¿Cómo planificar la búsqueda?
  - ▶ ¿Cuándo aplicar una opción atrevida o cauta?

# Ejemplo

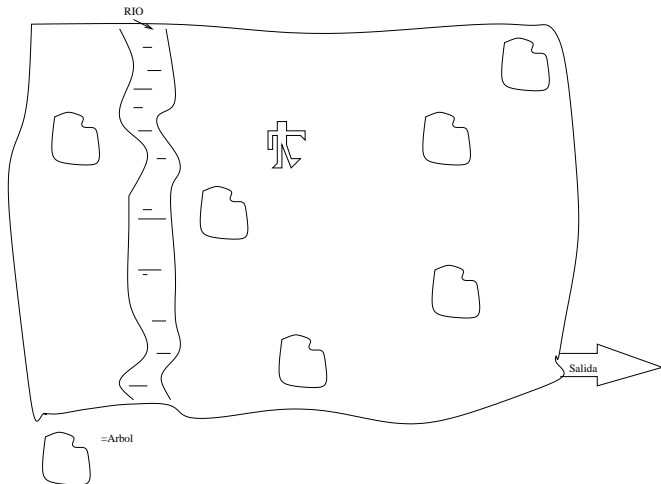
---

Supongamos que un robot está navegando por un parque, y debe llegar a la salida de éste.

- ▶ Percepciones: Su entorno.
- ▶ Acciones: Deplazarse (paso a paso).
- ▶ Objetivo: Alcanzar la salida.
- ▶ Entorno: Estático, discreto, determinista, episódico.
  - ▶ Entorno absoluto: Guarda en memoria el mapa completo del lugar (accesible).
  - ▶ Entorno local: Guarda la información sólo de su entorno físico más próximo (efectivamente accesible).
- ▶ Obstáculos: Río, árboles.

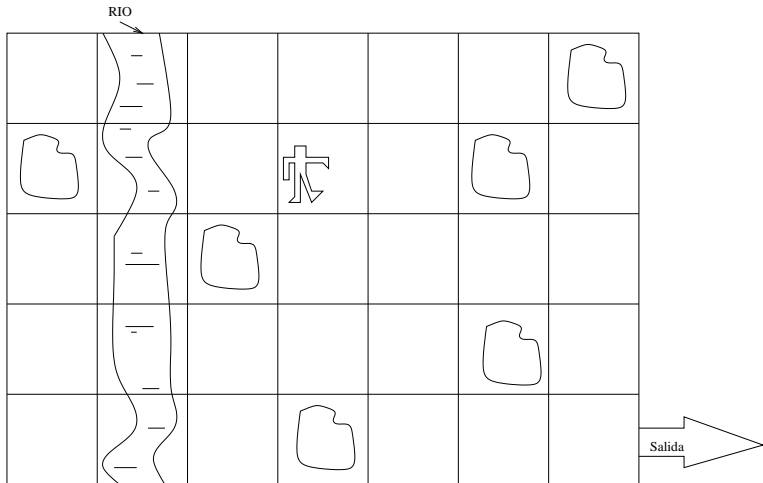
# Gráfico

---



# Representación del ejemplo

---



# Representando el ejemplo (I)

---

Variables:

- ▶  $A_{ij}$  = En la casilla  $(i, j)$  hay un árbol.
- ▶  $R_{ij}$  = Por la casilla  $(i, j)$  pasa el río.
- ▶  $D_{i,j}^t$  = En el momento  $t$  es posible desplazarse  $i$  casillas horizontales,  $j$  verticales ( $i, j \in \{-1, 0, 1\}$ ).
- ▶  $E_{ij}^t$  = El robot se encuentra en la casilla  $(i, j)$  en el momento  $t$ .

Base de conocimiento:

- ▶ Hechos:  $\underbrace{\{E_{44}^1\}}_{\text{Posición}} \cup \underbrace{\{\neg A_{11}, \dots, A_{41}, \dots, R_{21}, \dots\}}_{\text{Distribución de obstáculos}}$
- ▶ Condiciones: Para cada  $(i, j)$  y cada  $(k, h) \neq (i, j)$ :

$$\neg(E_{ij}^t \wedge E_{kh}^t)$$

## Representando el ejemplo (II)

---

Componente pensante:

- ▶ Acciones (reglas E-C-A):  $E_{34}^t \wedge \neg(R_{35} \vee A_{35}) \rightarrow D_{01}^t$
- ▶ Restricciones:  $\neg(A_{11} \wedge E_{11}^t)$
- ▶ Prohibiciones:  $E_{11}^t \rightarrow \neg D_{-1,-1}^t$
- ▶ Definiciones: La casilla  $(i,j)$  está libre ( $L_{ij}$ )

$$L_{ij}^t \leftrightarrow \neg(R_{ij} \vee E_{ij}^t \vee A_{ij})$$

- ▶ Creencia: por ejemplo,  $E_{11}^t \rightarrow L_{12}^t$  (¿falsa?)



## Representando el ejemplo (III)

---

- ▶ Resultado de la acción (mandato a los sensores; movimiento):
  - ▶  $D_{10}^t \wedge E_{22}^t \rightarrow E_{32}^{t+1}$  (¡No es disparable!).
  - ▶  $E_{34}^t \wedge D_{01}^t \rightarrow E_{35}^{t+1}$
- ▶ Estado:  $\{E_{ij}^t\}$  (entorno representado localmente).
- ▶ Estado:  $\{E_{ij}^t, A_{21}, \dots\}$  (entorno global).
- ▶ La componente reactiva está formada por reglas E-C-A.

## Representando el ejemplo (IV)

---

Componente racional:

- ▶ Objetivo:  $E_{71}^8$
- ▶ Planificación: Obtener un conjunto adecuado de reglas del tipo:

$$E_{ij}^t \rightarrow D_{kh}^t$$

para ejecutarlas ordenadamente.

Componente pro-activa: **Si**  $E_{34}^t$  **entonces** Objetivo :  $E_{44}^{t+1}$

Ejercicio: Elegir una base de conocimiento  $K$  relativamente pequeña y **natural** de tal modo que  $K \models E_{71}^8$ .

¿Se puede extraer de la prueba por tableros el camino que debe seguir el robot?

# Limitaciones de la lógica proposicional para el problema

---

- ▶ Grado de autonomía:

Se puede diseñar un conjunto de reglas para alcanzar la salida, si se puede, para ese parque, independientemente del sitio donde se encuentra el robot.

Si se cambia a otro parque, ya no sirve el sistema.

- ▶ Expresividad deficiente.

En el ejemplo anterior, el tamaño de la base de conocimiento es  $O(n)$  (siendo  $n$  el número de casillas), pues el entorno es accesible en su totalidad (con respecto al conocimiento del agente).

# Ventajas y Limitaciones

---

- ▶ La lógica proposicional es muy útil si no es importante el contenido de cada proposición, sólo es importante la estructura de la información
- ▶ En estos casos, muy manejable.
- ▶ Los problemas fundamentales son decidibles.
- ▶ Es una lógica Bivalente, pero se puede alterar la semántica para obtener lógicas multivaloradas.
- ▶ En algunos casos no es importante que sean deducibles todas las tautologías.

# Bibliografía

---

- ▶ M. Ben-Ari: *Mathematical Logic for Computer Science*. 2 ed. (2001), Capítulos 1 y 2.
- ▶ Russell & Norvig: *Inteligencia Artificial: Un enfoque moderno*. Capítulo 7. Disponible en la web:  
<http://aima.cs.berkeley.edu/newchap07.pdf>