

Agentes Inteligentes

Sistemas Inteligentes Distribuidos

Sergio Alvarez

Javier Vázquez

Bibliografía

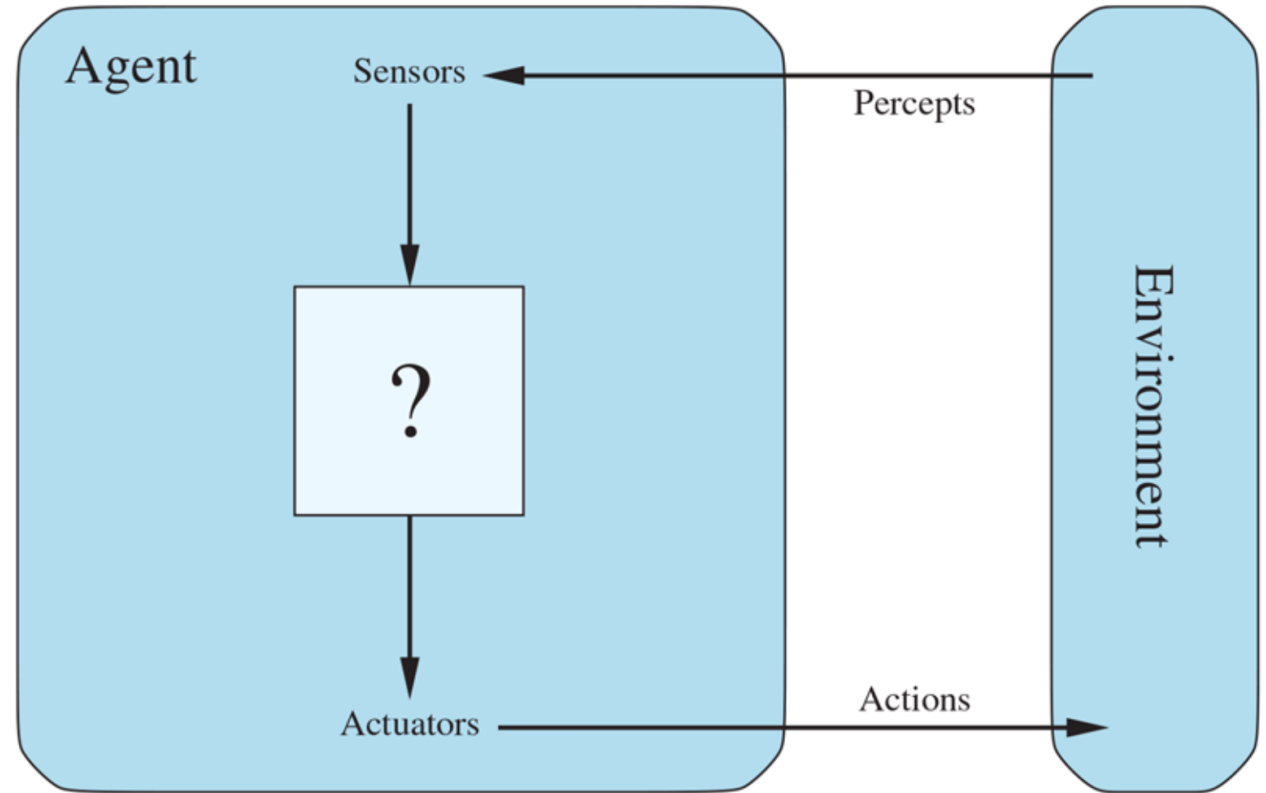
- *[AIMA] Artificial Intelligence: A Modern Approach* (Russell & Norvig), cap. 2

Agente, racionalidad

Agentes Inteligentes

Definición

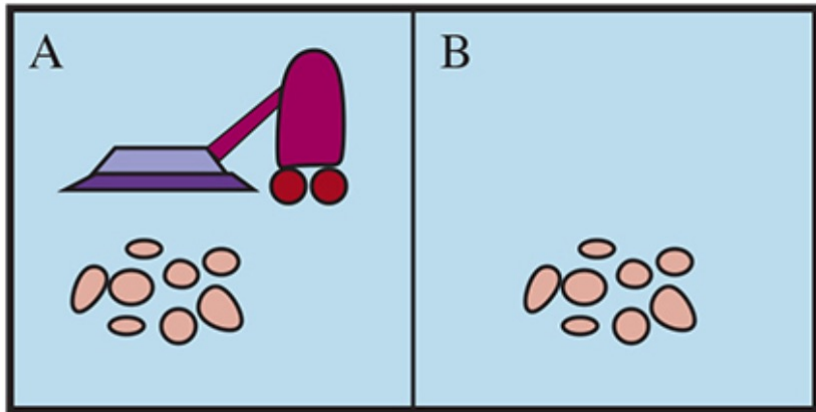
- Un agente es un sistema computacional capaz de actuar en un entorno, de manera autónoma, para cumplir los objetivos de diseño
- A través de *sensores*, el agente recibe *percepciones* del entorno
- El agente *actúa* en el entorno a través de sus *actuadores*



Agente inteligente como función

- En determinadas circunstancias, un agente debería ser capaz de interactuar con el entorno en tiempo real y de manera asíncrona
- Desde un punto de vista teórico, vamos a simplificar suponiendo que a cada percepción le corresponde una acción
 - Podría ser una acción nula por voluntad o por incapacidad de razonar
- Un agente, pues, puede interpretarse como una función que retorna:
 - Una acción para cada percepción, o bien
 - Una acción para la secuencia histórica de percepciones (si tiene *memoria*)

Agente inteligente como función



Percept sequence	Action
$[A, \textit{Clean}]$	<i>Right</i>
$[A, \textit{Dirty}]$	<i>Suck</i>
$[B, \textit{Clean}]$	<i>Left</i>
$[B, \textit{Dirty}]$	<i>Suck</i>
$[A, \textit{Clean}], [A, \textit{Clean}]$	<i>Right</i>
$[A, \textit{Clean}], [A, \textit{Dirty}]$	<i>Suck</i>
\vdots	\vdots
$[A, \textit{Clean}], [A, \textit{Clean}], [A, \textit{Clean}]$	<i>Right</i>
$[A, \textit{Clean}], [A, \textit{Clean}], [A, \textit{Dirty}]$	<i>Suck</i>
\vdots	\vdots

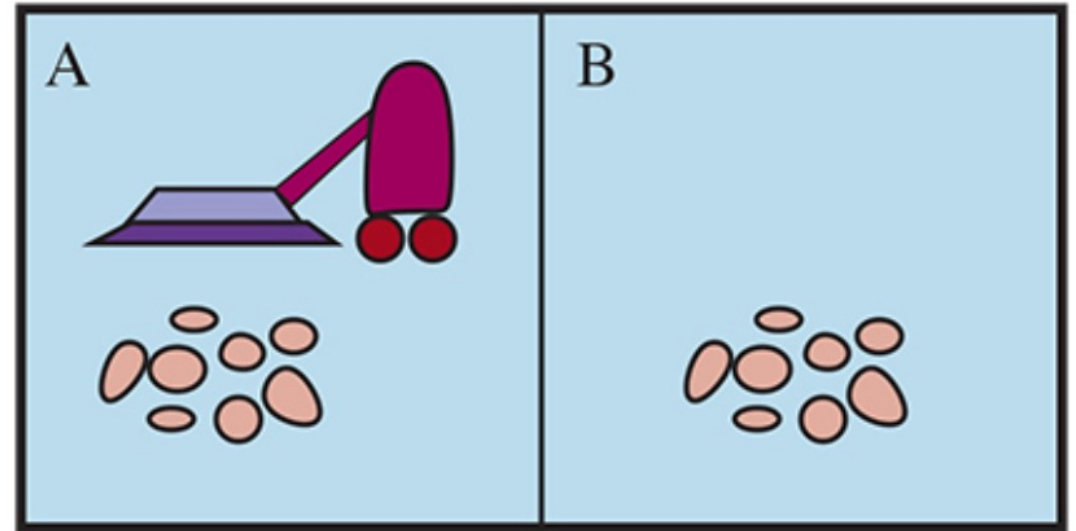
Ejemplo de posible *tabulación* de un agente aspiradora en un entorno con dos estancias

Racionalidad

- Cuando un agente hace lo correcto, diremos que es un *agente racional*
- En general, diremos que hay racionalidad cuando se cumplen los objetivos de diseño
 - En base a métricas de rendimiento que dependerán de estos objetivos
- A partir de cada posible secuencia de percepciones y, potencialmente, su conocimiento propio, un agente racional debería seleccionar la acción que maximice su métrica de rendimiento

Racionalidad

¿Qué métricas de rendimiento serían apropiadas para nuestro agente aspiradora?

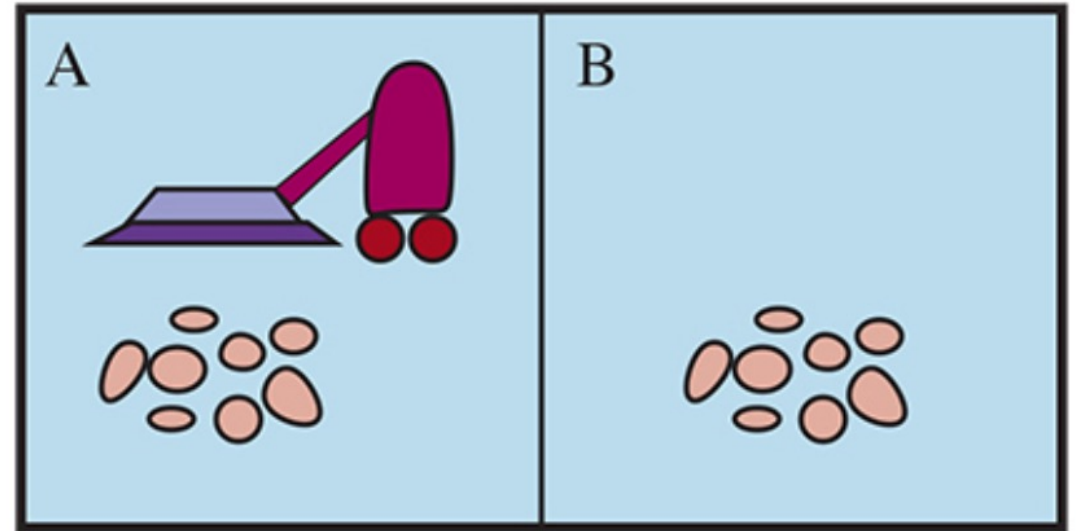


Racionalidad

¿Qué métricas de rendimiento serían apropiadas para nuestro agente aspiradora?

- ¿Número de veces que ha limpiado una estancia?
- ¿Limpieza de las estancias?

En general, es mejor definir métricas sobre el entorno que sobre las acciones



Racionalidad vs perfección

- La perfección implicaría maximizar el rendimiento real en base a la evaluación de la acción tomada, pero...
 - ... puede que el agente no sea omnisciente
 - ... puede que el agente esté en un entorno altamente cambiante, posiblemente con otros agentes
 - ... puede que el entorno sea imposible de modelar (*frame problem*)
- Racionalidad no significa perfección
 - La racionalidad busca maximizar el **rendimiento esperado**, en base al conocimiento interno del agente
- Veremos ejemplos de diseño de agente que priorizan esta definición de racionalidad
 - Y cómo adaptar estos diseños para permitir que su conocimiento se adapte

Entornos

Agentes Inteligentes

Definición de entorno

- Los entornos son los **problemas** para los cuales los agentes son **soluciones**
- Un entorno se define en base a la tarea objetivo a resolver
- PEAS: *Performance, Environment, Actuators, Sensors*

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, reduced costs	Patient, hospital, staff	Display of questions, tests, diagnoses, treatments	Touchscreen/voice entry of symptoms and findings
Satellite image analysis system	Correct categorization of objects, terrain	Orbiting satellite, downlink, weather	Display of scene categorization	High-resolution digital camera
Part-picking robot	Percentage of parts in correct bins	Conveyor belt with parts; bins	Jointed arm and hand	Camera, tactile and joint angle sensors
Refinery controller	Purity, yield, safety	Refinery, raw materials, operators	Valves, pumps, heaters, stirrers, displays	Temperature, pressure, flow, chemical sensors
Interactive English tutor	Student's score on test	Set of students, testing agency	Display of exercises, feedback, speech	Keyboard entry, voice

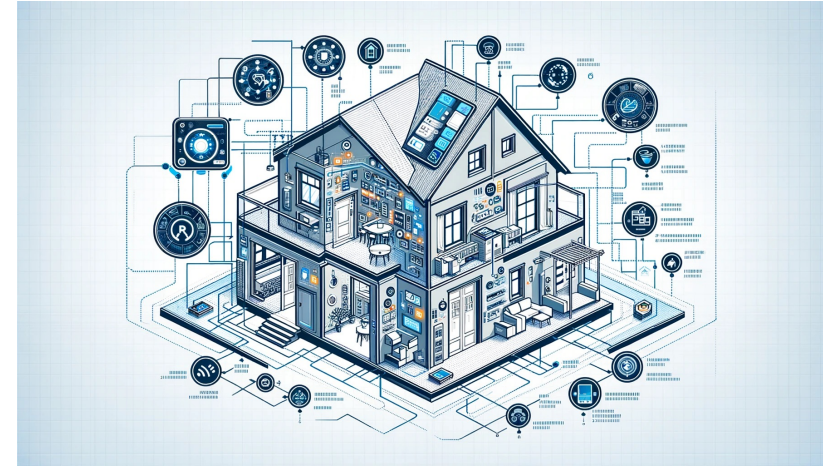
[fuente: AIMA 4th Edition, pág. 109]

Ejemplos de entorno



Photo (c) Guofei Aviation

Acción coordinada de drones



Creación propia (DALL-E)

Domótica



(c) The RoboCup Federation

Equipos deportivos de robots



Dominio p'ublico
(<https://www.pickpik.com>)

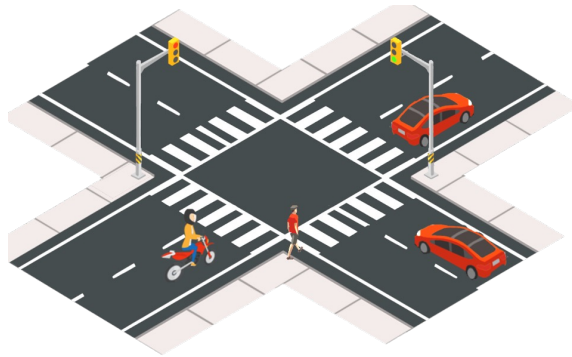
Detección de defectos en cadena de montaje

Ejemplos de entorno



Videojuegos

Imagen (c) 2019
Center for Human-Compatible AI
https://github.com/HumanCompatibleAI/overcooked_ai



Vehículos conectados

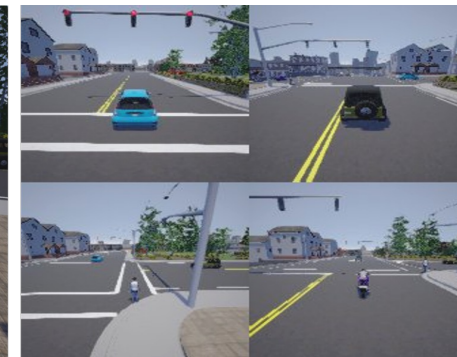


Imagen (c) 2019 Praveen Palanisamy
<https://github.com/praveen-palanisamy/macad-gym>

Public domain (<https://www.pexels.com>)



Mercados financieros

Ajedrez

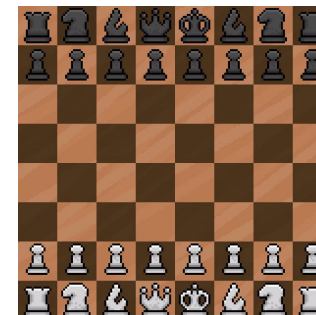


Imagen (c) The Farama Foundation 2024
<https://github.com/Farama-Foundation/PettingZoo>

Propiedades de un entorno

Propiedad	Menos complejidad	Más complejidad
Observabilidad	Totalmente observable	Parcialmente observable
Número de agentes	Un agente	Multiagente
Determinismo	Determinista	No determinista
Atomicidad	Episódico	Secuencial
Dinamicidad	Estático	Dinámico
Continuidad	Discreto	Continuo
Conocimiento	Conocido	Desconocido

Observabilidad

Totalmente observable: ajedrez

Parcialmente observable: vehículos
conectados

- Un entorno es **totalmente observable** si los sensores del agente permiten **acceso total al estado relevante** del entorno **en todo momento, sin ruido ni incertidumbre**
- La relevancia viene determinada por la métrica de rendimiento
- Un entorno totalmente observable permite simplificar la parte de la memoria interna dedicada al entorno
- Un caso especial de entornos parcialmente observables son los entornos no observables
 - Ejemplo: Kriegspiel chess

Número de agentes

Un agente: detección de defectos

Multiagente: mercado financiero

- Los agentes relevantes para un agente A son aquellos cuyo comportamiento *racional* puede tener influencia en el rendimiento de A
- El entorno es **multiagente** si hay como mínimo: **un agente A, y un agente relevante para A**
- Los entornos multiagente pueden ser cooperativos, competitivos, o una mezcla de ambos

Determinismo

Determinista: ajedrez

No determinista: equipo de drones

- Si es posible calcular el próximo estado del entorno únicamente a partir del estado actual y las acciones que están ejecutando los agentes, entonces el entorno es **determinista**
- La interpretación de esta definición es pragmática: es posible que teóricamente sea posible calcular el próximo estado, pero no sea viable técnica o prácticamente
 - En estos casos, el entorno es no determinista
 - Un entorno parcialmente observable es, por lo general, no determinista
- No determinista no es equivalente a estocástico
 - Si el modelo del entorno es estocástico con probabilidades conocidas, el cálculo del próximo estado se puede hacer en base a distribuciones de probabilidad
 - Si hay variables aleatorias en el modelo del entorno cuya distribución de probabilidad sea desconocida para el agente, el entorno es estocástico y no determinista

Atomicidad

Episódico: detección de defectos

Secuencial: mercado financiero

- En un entorno **episódico**, cada decisión del agente es **independiente de las percepciones y acciones pasadas**
- En un entorno secuencial, por lo tanto, hay que tener en cuenta no sólo la historia sino también las posibles consecuencias futuras de cada acción

Dinamicidad

Estático: ajedrez (sin tiempo)

Dinámico: ajedrez (con tiempo)

- En general, diremos que un entorno es **dinámico** cuando su estado puede **cambiar durante el tiempo que tarda el agente** en decidir su próxima acción
- Por lo tanto, si el entorno cambia, pero lo hace a una frecuencia inferior al ciclo de razonamiento del agente, o no cambia, es un entorno estático
- Un caso especial son aquellos entornos donde los elementos del entorno no cambian, pero el paso del tiempo impacta en el rendimiento del agente: son entornos semi-dinámicos

Continuidad

Discreto: mercado financiero

Dinámico: vehículo conectado

- Un entorno es **continuo** cuando cualquiera (basta sólo uno) de los siguientes elementos es **continuo**:
 - Estado del entorno
 - Gestión del tiempo
 - Espacio de percepciones y percepciones
 - Espacio de acciones y acciones

Conocimiento

Conocido: detección de defectos

Desconocido: videojuego (1ª vez)

- Si el agente conoce **las leyes que gobiernan el entorno y las métricas de rendimiento que debería optimizar**, entonces el entorno es **conocido**
- No hay una relación clara entre observabilidad y conocimiento
 - Un entorno puede ser conocido, pero parcialmente observable (juegos de cartas)
 - Un entorno puede ser desconocido, pero totalmente observable (un videojuego retro que no habíamos visto nunca)

Propiedades de un entorno

Propiedad	Menos complejidad	Más complejidad
Observabilidad	Totalmente observable	Parcialmente observable
Número de agentes	Un agente	Multiagente
Determinismo	Determinista	No determinista
Atomicidad	Episódico	Secuencial
Dinamicidad	Estático	Dinámico
Continuidad	Discreto	Continuo
Conocimiento	Conocido	Desconocido
	Entorno más sencillo	Entorno más complejo

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete

[fuente: AIMA 4th Edition, pág. 116]

Arquitectura de un agente

Agentes Inteligentes

Un agente "sencillo"

Percept sequence	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Clean], [A, Clean]	Right
[A, Clean], [A, Dirty]	Suck
⋮	⋮
[A, Clean], [A, Clean], [A, Clean]	Right
[A, Clean], [A, Clean], [A, Dirty]	Suck
⋮	⋮

¿Es este programa
suficiente?

¿Es
computacionalmente
viable?

function TABLE-DRIVEN-AGENT(*percept*) **returns** an action

persistent: *percepts*, a sequence, initially empty

table, a table of actions, indexed by percept sequences, initially fully specified

append *percept* to the end of *percepts*

action ← LOOKUP(*percepts*, *table*)

return *action*

Racionalidad limitada

- En un entorno dinámico, la utilidad de una acción varía con el tiempo
- Los procesos de toma de decisión cuestan tiempo y recursos
- **El comportamiento racional, cuando se toma en consideración el coste de la decisión, difiere del comportamiento *ideal***
- El trabajo principal de un agente es hacer las mejores decisiones tan pronto como sea posible, sin pretender buscar el óptimo teórico (*bounded optimality*)
- La arquitectura del agente ha de dotar de una estructura computacional que le permita disponer de opciones, y seleccionar entre ellas, para alcanzar esta “racionalidad limitada”

Russell, S. J. (1991). An architecture for bounded rationality. *ACM SIGART Bulletin*, 2(4), 146-150.

Estructura de un agente

- Nuestro objetivo: encontrar maneras de **diseñar** agentes que generen comportamiento racional de la manera más eficiente posible
- El agente define una función (secuencias de percepciones a acciones), pero ¿cómo se implementa esta función?

agente = infraestructura + programa

(Nota: en ALMA se define *agent = architecture + program*, pero nosotros consideraremos que tanto infraestructura como programa están formados de componentes que conforman una arquitectura)

- Los dos elementos son distintos e intercambiables pero interdependientes
 - Por ejemplo, el programa no debería intentar ejecutar acciones que no estén disponibles a través de los actuadores

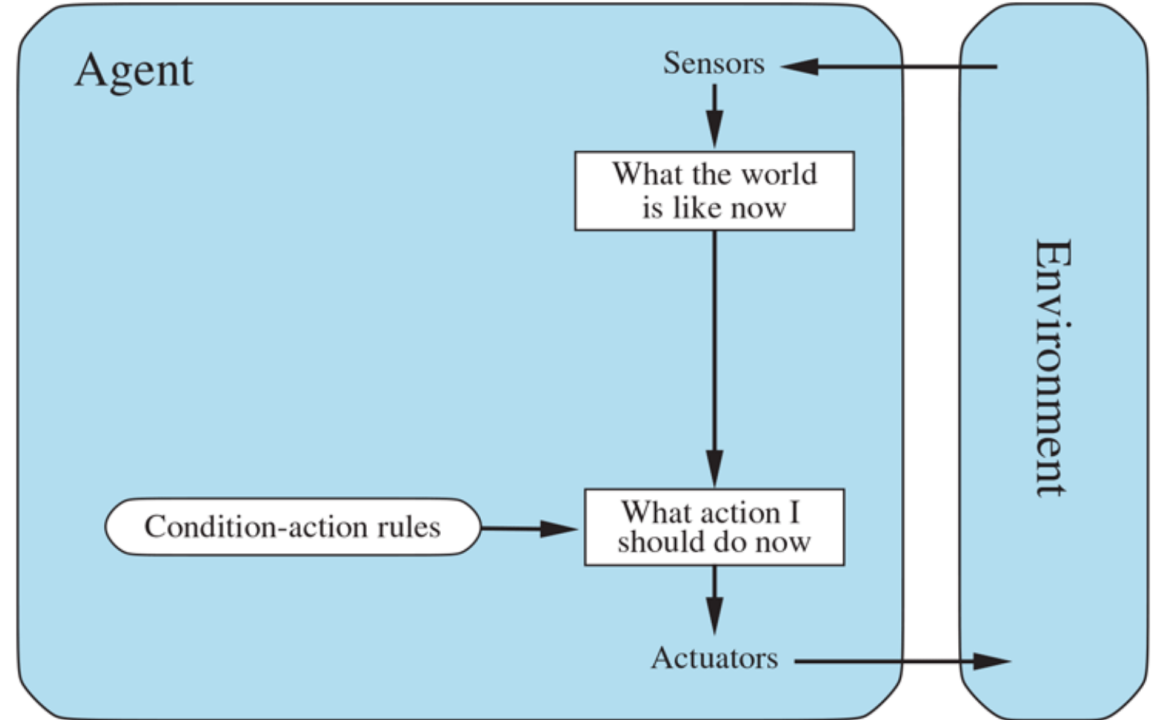
Arquitecturas de agente

- Una **arquitectura** de agente es una especificación de **componentes y la conexión entre los mismos**, diseñada con el objetivo de facilitar la generación de **comportamiento racional**
 - Las arquitecturas de agente genéricas son independientes de la tarea objetivo
 - Las arquitecturas facilitan, no resuelven: cada componente deberá ser implementado de manera específica para la tarea objetivo
- Veremos:
 - Agentes puramente reactivos
 - Agentes reactivos con estado
 - Agentes deliberativos guiados por objetivos
 - Agentes deliberativos guiados por utilidad
 - Agentes adaptativos

Agente puramente reactivo

- AIMA: *Simple reflex agents*
- No hay componente de memoria
- El modelo es típicamente un sistema de producción, pero podría ser e.g. una red neuronal

```
function SIMPLE-REFLEX-AGENT(percept) returns an action  
persistent: rules, a set of condition–action rules  
  
  state ← INTERPRET-INPUT(percept)  
  rule ← RULE-MATCH(state, rules)  
  action ← rule.ACTION  
  return action
```

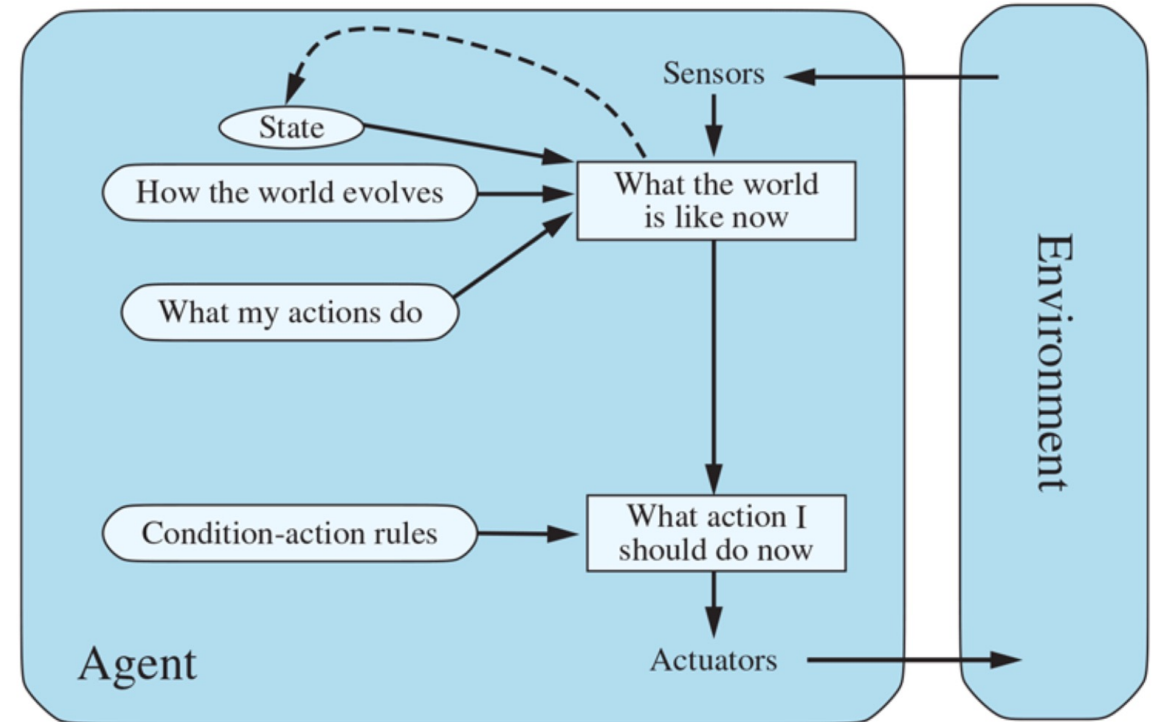


Agente reactivo con estado

- ALMA: *Model-based reflex agents*
- Se añade:
 - **Componente de memoria**
 - Modelo de actualización del estado interno (**revisión de creencias**)

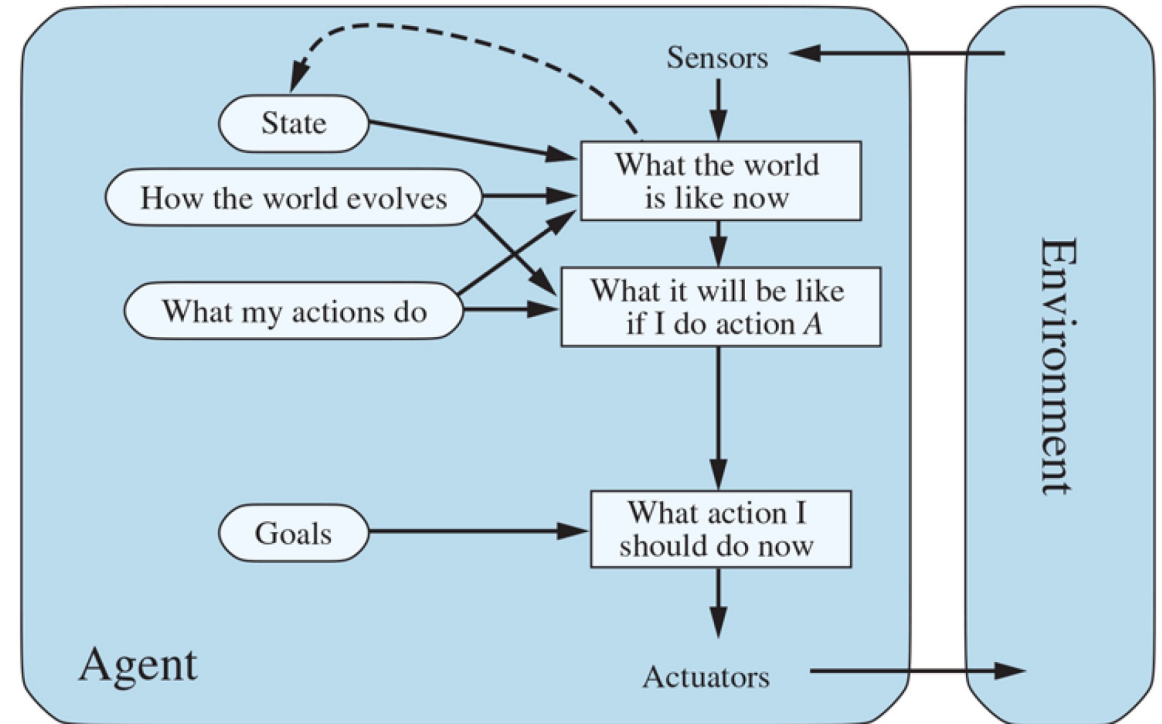
```
function MODEL-BASED-REFLEX-AGENT(percept) returns an action
  persistent: state, the agent's current conception of the world state
               transition_model, a description of how the next state depends on
                 the current state and action
               sensor_model, a description of how the current world state is reflected
                 in the agent's percepts
               rules, a set of condition-action rules
               action, the most recent action, initially none

  state ← UPDATE-STATE(state, action, percept, transition_model, sensor_model)
  rule ← RULE-MATCH(state, rules)
  action ← rule.ACTION
  return action
```



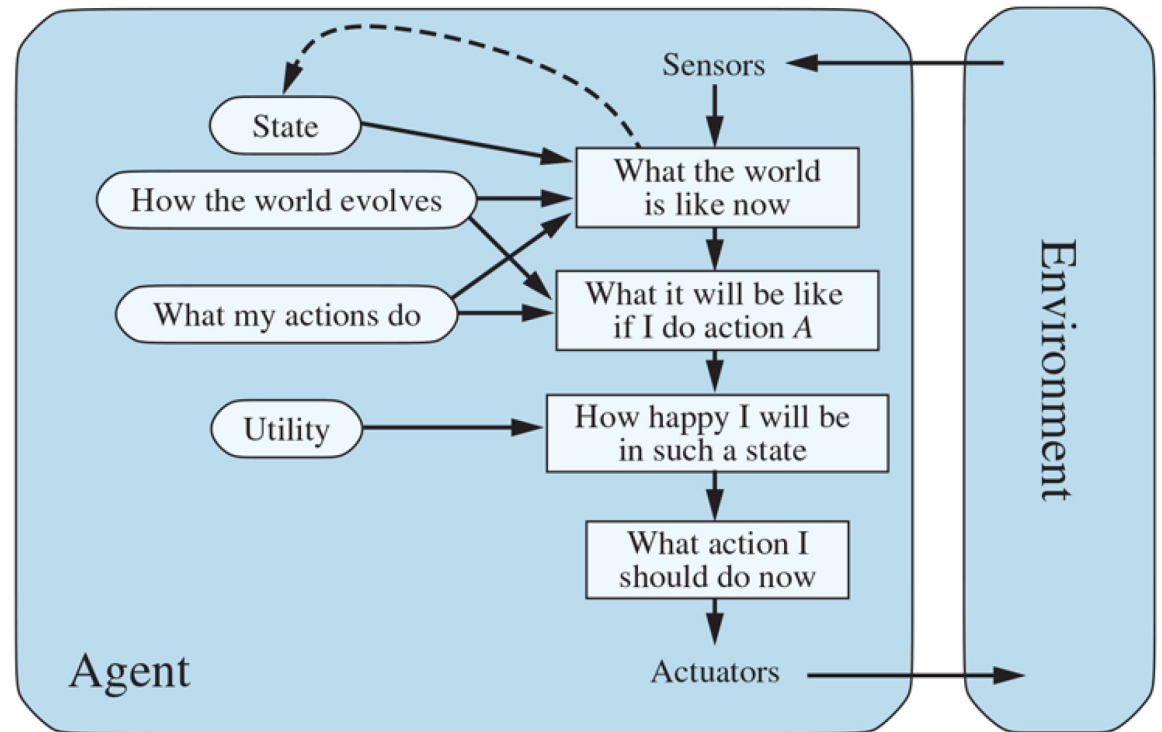
Agente deliberativo por objetivos

- AIMA: *Goal-based agents*
- Memoria, revisión de creencias
- Se añade:
 - Modelo del mundo para evaluar el efecto de las acciones (***world model***)
 - Modelo de **selección de acciones basada en objetivos simbólicos**
- Deliberación:
 - Selección de intenciones
 - Selección de planes



Agente deliberativo por utilidad

- ALMA: *Utility-based agents*
- Memoria, revisión de creencias
- Se añade:
 - Modelo del mundo para evaluar el efecto de las acciones (***world model***)
 - **Función de utilidad subsimbólica**
- Deliberación:
 - Ordenación de opciones por preferencia (utilidad)



Agentes adaptativos

- ALMA: *Learning agents*
- Se construyen a partir de un agente A (cualquier arquitectura)
- Se añade:
 - La métrica de rendimiento (**función de recompensa**)
 - **Módulo de evaluación** del rendimiento
 - **Programa para modificar A**
 - **Generador de configuraciones** del entorno
- El aprendizaje por refuerzo es un posible método para modificar A

