

Agentes Inteligentes

Alejandro Figueroa

Agentes y Ambientes

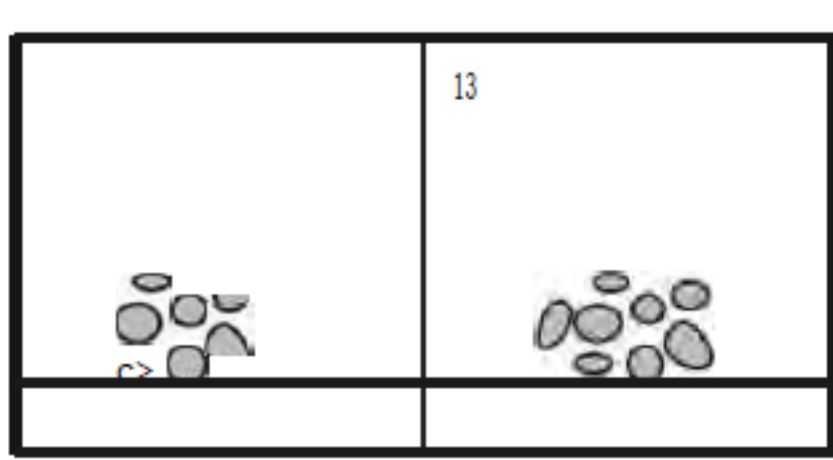
- Un agente es cualquier cosa que percibe el ambiente a través de algún sensor y actúa sobre ese ambiente mediante “actuadores”.
 - Un ser humano tiene ojo, oídos como sensores y manos, piernas y cuerdas vocales como “actuadores”.
 - Un agente robótico puede tener cámaras como sensores y motores como “actuadores”.
 - Un agente de software recibe el input del teclado, archivos, y tráfico de red y actúa sobre su ambiente desplegando por pantalla, escribiendo archivos, generando tráfico de red.
- A lo que el agente percibe en cierto instante se le llama “percept” (de acuerdo a su palabra en inglés).

Agentes y Ambientes

- Una secuencia de “percept” es todo lo que el agente ha percibido en la historia. En general, un agente escoge su acción en lo que percibe en cierto instante y/o el historial de sus percepts, pero no puede tomar una decisión en algo que no ha percibido.
- Para mapear los “percepts” a una acción se utiliza una función de agente.
 - Uno podría tabular una tabla para indicarle al agente lo que tiene que hacer en cada uno de los posibles percepts.
 - También puede usar un programa o una función matemática.

Agentes y Ambientes

- Para ilustrar esto, imaginemos un mundo imaginario donde tenemos una aspiradora y dos posibles lugares donde puede estar.



Los movimientos posibles de este agente: La aspiradora puede moverse a la izquierda, derecha, succionar el polvo o bien hacer nada.

Agentes y Ambientes

Percept sequence	Action
<i>[A, Clean]</i>	<i>Right</i>
<i>[A, Dirty]</i>	<i>Suck</i>
<i>[B, Clean]</i>	<i>Left</i>
<i>[13, Dirty]</i>	<i>Suck</i>
<i>[A, Clean], [A, Clean]</i>	<i>Right</i>
<i>[A, Clean], [A, Dirty]</i>	<i>Suck</i>
<i>⋮</i>	<i>⋮</i>
<i>[A, Clean], [A, Clean], [A, Clean]</i>	<i>Right</i>
<i>[A, Clean], [A, Clean], [A, Dirty]</i>	<i>Suck</i>
<i>⋮</i>	<i>⋮</i>

¿Cómo uno llena esta tabla?

Agentes y Ambientes

- Un agente racional es el que siempre hace lo correcto. Lo cual se mide considerando las consecuencias del actuar del agente.
 - Las percepciones del agente lo obligan a llevar a cabo una secuencia de acciones. La que provocan una secuencia de cambios en el ambiente. Si la secuencia es deseable, entonces el agente se ha desempeñado bien.
 - Hay una métrica de desempeño que mide cuán deseable es el efecto producido por una secuencia de acciones.
- La métrica depende de la tarea que se esté tratando de alcanzar.

Agentes y Ambientes

- Lo que es o no racional depende de las siguientes cuatro cosas:
 - La métrica de desempeño que define el criterio de éxito.
 - El conocimiento previo del agente del ambiente.
 - Las acciones que el agente puede desempeñar.
 - La secuencia histórica de percepts que tiene el agente.
- Lo que nos lleva a la siguiente definición de agente racional:
 - “Para cada secuencia posible de “percerpts”, un agente racional debería escoger la acción que maximiza su métrica de desempeño, dado la evidencia provista por la secuencia de percepts y cualquier conocimiento previo ue el agente tenga.

Agentes y Ambientes

- Hay una diferencia entre un agente racional y otro omnisciente:
 - Un agente omnisciente conoce el resultado real de sus acciones. Esto es imposible en la realidad.
 - La racionalidad maximiza el desempeño esperado, mientras la perfección maximiza el desempeño real.
- La definición de racionalidad no pide que el agente sea omnisciente.
- Por ejemplo, sería racional que un agente mirara para los dos lados al cruzar la calle. Mirar par uno sería irracional. El agente debería priorizar el mirar primero antes de cruzar porque eso maximiza su desempeño.

Agentes y Ambientes

- Esta definición de agente racional implica que no sólo el agente debe recolectar información de acuerdo a sus sensores, sino que también debe aprender tanto como sea posible de lo que percibe.
- Si un agente confía en el conocimiento previo dado por su diseñador, y no en sus propias percepciones, decimos que el agente carece de autonomía.
- Un agente racional debe ser autónomo.
- Otro concepto importante es el de “ambiente de tarea” que es el problema que el agente tiene que resolver.

Agentes y Ambientes

- Es necesario especificar, la métrica de desempeño, el ambiente, los “actuators” y los sensores.

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits	Roads, other traffic, pedestrians, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS , odometer, accelerometer, engine sensors, keyboard
PEAS description of the task environment for an automated taxi.				



Algunos objetivos pueden estar en conflicto, ergo el agente debe hacer un trade-off.

Agentes y Ambientes

- En el caso de un taxi, tenemos diferentes tipos de caminos. Los caminos contienen tráfico, peatones, animales, obras en la carretera, autos policía, hoyos, lomo de toro, etc.
- El taxi debe interactuar con pasajeros reales.
- El clima: escarcha, nieve, lluvia, etc.
- En el caso de un taxi de exportación, podría estar diseñado para manejar en UK o Chile.
- Los “actuadores” son el acelerador, el control sobre el manubrio y los frenos. Comunicarse con los pasajeros y con otros conductores educadamente.

Agentes y Ambientes

- Por el lado de los sensores:
 - Cámaras para ver el camino.
 - Sensores para detectar otros vehículos y obstáculos.
 - Medidores de bencina, velocidad, etc.
 - Control del GPS para no perderse.
 - Interfaz para comunicarse con el pasajero (teclado o pantalla).

Agentes y Ambientes

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, reduced costs	Patient, hospital, staff	Display of questions, tests, diagnoses, treatments, referrals	Keyboard entry of symptoms, findings, patient's answers
Satellite image analysis system	Correct image categorization	Downlink from orbiting satellite	Display of scene categorization	Color pixel arrays
Part-picking robot	Percentage of parts in correct bins	Conveyor belt with parts: bins	Jointed arm and hand	Camera, joint angle sensors
Refinery controller	Purity, yield, safety	Refinery, operators	Valves, pumps, beaters, displays	Temperature, pressure, chemical sensors
Interactive English tutor	Student's score on test	Set of students, testing agency	Display of exercises, suggestions, corrections	Keyboard entry

Agentes y Ambientes

- Completamente vs. Parcialmente observable
 - Si los sensores le dan acceso al estado completo del ambiente. En particular, a los aspectos relevante para su desempeño.
 - Un ambiente no sólo puede ser parcialmente observable producto de no tener todos los sensores necesarios, sino que también producto ruido o por fallas en la recolección de datos por los sensores.
- Nótese que hay variable que no son posible de observar como por ejemplo lo que piensan los otros taxistas.

Agentes y Ambientes

- Agentes únicos vs. Múltiples
 - Un ambiente de una agente resolviendo un puzzle es un ambiente de agente único. En cambio, en un juego de ajedrez tenemos un ambiente de múltiple agentes.
- Competitivo vs. Cooperativo
 - En un juego de ajedrez los agentes compiten, pero en cambio en el caso del taxi, los agente van a intentar evitar los choques. En este último caso el ambiente es parcialmente cooperativo.
- Determinista vs. Estocástico
 - Si el ambiente queda completamente determinado por el movimiento del agente o no. La mayoría de los ambientes son muy complejos por lo que son vistos como estocásticos.

Agentes y Ambientes

- Episódico vs. Secuencial
 - En el primero, la experiencia del agente es dividida en episodios atómicos, en el cual recibe un “percept” y ejecuta una acción. El próximo episodio no depende de las decisiones anteriores. La mayoría de las tareas de clasificación son episódicas. Por ejemplo, detectar elementos defectuosos en la línea de ensamblaje no depende del anterior, básicamente.
 - En un ambiente secuencial, la decisión actual podría afectar a las decisiones futuras. El ajedrez y la conducción de un taxi son secuenciales. Los ambientes secuenciales son más complejos que los futuros, ya que te obligan a pensar en más adelante.

Agentes y Ambientes

- Estático vs. Dinámico
 - Si el ambiente puede cambiar mientras un agente está deliberando, entonces se dice que es dinámico, de lo contrario es estático. Para el caso de los ambientes estáticos, los agentes son más fáciles de diseñar ya que no es necesario monitorearlos, en cambio los dinámicos están constantemente demandando acciones del agente.
- Discreto vs. Continuo
 - Esta dimensión se aplica al estado del ambiente, a la forma que el tiempo es manejado, y a los percepts y acciones del agente. Jugar al ajedrez involucra (salvo el reloj) variables discretas, sin embargo el conductor de taxi maneja variables continuas.

Agentes y Ambientes

- Known vs. Unknown
 - Se refiere al conocimiento del agente del ambiente. Si conoce todas las leyes que lo rigen. Entre más desconocido sea, más importante será la capacidad del agente de aprender y tomar buenas decisiones.
 - El conocimiento de las leyes tienen que ver con las reglas. No si el agente conoce el estado de los otros agentes por ejemplo.
- Como se puede apreciar el caso más difícil de lidiar es cuando se tiene parcialmente observable, multiagente, estocástico, secuencial, dinámico, continuo y desconocido.
 - Un conductor de taxi tiene casi todas estas dificultades, salvo que el ambiente es casi todo conocido.
 - Conducir en un país nuevo con una geografía nueva es mucho más difícil.

Agentes y Ambientes

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
Interactive. English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete

La Estructura de los Agentes

- Por un lado tenemos la función del agente, es decir el programa, y por otro lado la arquitectura, es decir el computador y sus sensores.

```
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,
```



El agente guarda una tabla con los percepts históricos. El historial le sirve para saber qué hacer en el futuro.

La Estructura de los Agentes

- En la práctica este método es muy poco útil:
 - Ningún agente va a tener el espacio en memoria para guardar esta tabla.
 - El diseñador no tendría tiempo para diseñar una tabla con todos los casos.
 - Ningún agente podría aprender las entradas correctas a esa tabla desde la experiencia.
 - Aún si el ambiente fuera muy simple, el diseñador del agente no tendría la guía para llenar la tabla.
- El desafío de la IA es escribir agente que hagan algo similar a esta tabla, pero de una manera más eficiente en términos de tamaño.

La Estructura de los Agentes

- Hay cuatro tipos de programas de agentes que abarcan casi todos los principios de los agentes inteligentes:
 - Simple reflex agents;
 - Model-based reflex agents;
 - Goal-based agents; and
 - Utility-based agents.

La Estructura de los Agentes: Simple-Reflex

- Es el tipo más simple.
- El agente selecciona sus acciones en base a la información que recibe en el momento. No usa la historia.
- Por ejemplo, el caso discutido de la aspiradora. El agente sólo percibe donde está y la mugre que hay. No tiene información de la mugre histórica.

```
function REFLEX-VACUUM-AGENT([location, status]) returns an action
  if status = Dirty then return Suck
  else if location = A then return Right
  else if location = B then return Left
```

La Estructura de los Agentes: Simple-Reflex

- Este tipo de agentes pueden estar inmersos en medio más complejos también. Por ejemplo, un agente que comienza el frenado del taxi cuando el auto que va adelante comienza a frenar.
- Los humanos tenemos muchas de estas respuestas aprendidas, como respirar o pestañar.
- Normalmente están compuestos por reglas del tipo if/then o situation/action.

```
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
```


La Estructura de los Agentes: Simple-Reflex

- Estos agentes son muy simples, pero poseen una inteligencia limitada.
- La decisión correcta sólo puede ser tomada si el ambiente es observable completamente. Sólo un poco de no-observabilidad puede ocasionar muchos problemas.

La Estructura de los Agentes: Model-Reflex

- El agente mantiene algún estado interno que de alguna forma modela la historia de los percepts y con esto refleja al menos en parte lo que no puede observar del estado actual.
 - Para el ejemplo del taxi, el estado interno no es muy caro. Básicamente se necesita almacenar el frame previo que permita detectar el cambio de luces traseras.
 - Para cambiar de pista, el taxi debe registrar y rastrear la posición de todos los autos cercanos.
- Para actualizar la información interna, se necesitan dos tipos de información:
 - ¿Cómo el mundo evoluciona independiente del agente?
 - ¿Cómo las acciones del agente afectan al mundo?

La Estructura de los Agentes: Model-Reflex

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

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

El cómo se almacenan los modelos en el agente depende de la arquitectura y del problema.

La Estructura de los Agentes: Goal-Reflex

- Muchas veces tener conocimiento suficiente del ambiente no es suficiente para decidir qué es lo que se debe hacer.
 - Por ejemplo, en una bifurcación, un taxi puede doblar a la derecha, izquierda o seguir derecho. La decisión correcta depende de lo que el taxi está intentando hacer. En este caso, el objetivo es el destino final del pasajero.
 - Hay veces que la gratificación viene de inmediato después de una acción. Otras veces viene después de una secuencia larga de decisiones.
 - Si bien este tipo de agente parece ser menos eficiente, es más flexible ya que el conocimiento utilizado para la toma de decisiones es explícito y puede ser modificado. Si comienza a llover puede adaptar su comportamiento para operar los frenos de manera más efectiva o cambiar a una mejor ruta para alcanzar el destino.

La Estructura de los Agentes: Utility Agent

- El objetivo no es el único parámetro necesario para generar un comportamiento de alta-calidad en la mayoría de los ambientes.
 - Por ejemplo, hay muchas formas de alcanzar un destino para un taxi. Algunas son más rápidas, seguras, otras son más confiables o baratas que otras.
 - Es necesario distinguir y comparar diferentes “estados del mundo” de acuerdo a cuan “feliz” hacen al agente.
 - Una función de utilidad es simplemente una internalización de cómo el agente mide el desempeño. Mediante la maximización de esta función, el agente escoge sus acciones.
 - La función de utilidad es una manera que tiene el agente de ser racional.

La Estructura de los Agentes: Utility Agent

- Los agentes basados en utilidad tienen ventajas en términos de flexibilidad y aprendizaje.
 - Este tipo de agente puede manejar objetivos que se contraponen, por ejemplo velocidad versus seguridad.
 - Puede ser que el agente esté tratando de alcanzar varios objetivos inciertos. La función de utilidad puede sopesar esas probabilidades y las importancias de los objetivos.
 - Técnicamente hablando, un agente racional basado en utilidad se mueve de tal forma que maximiza la utilidad esperada producto de sus acciones.
 - El mundo es incierto, estocástico y cambiante.
 - Este tipo de agentes no da una definición “global” de racionalidad: aquellos agentes que tienen el mejor desempeño.

La Estructura de los Agentes: Utility Agent

- ¿Es tan simple como diseñar agentes que maximicen la utilidad esperada?
 - No es tan simple. El agente debe modelar y rastrear su ambiente, lo cual involucra diversas áreas de investigación como percepción, representación, razonamiento, y aprendizaje.
 - Determinar el camino a la mejor utilidad esperada requiere algoritmos muy ingeniosos. La “racionalidad perfecta” es muchas veces inalcanzable debido a la complejidad computacional.

La Estructura de los Agentes: Agentes que Aprenden

- Un agente que aprende puede ser dividido en cuatro componentes conceptuales:
 - El elemento de aprendizaje que es el responsable de hacer las mejoras. Recoge las críticas de desempeño y ve cómo introducir las mejoras.
 - El elemento de desempeño que es el responsable de seleccionar las acciones a ejecutar.
 - El elemento crítico es necesario porque los percepts no proveen información del éxito del agente.
 - La otra componente es la generador de problemas, responsable de sugerir acciones que van a guiar a experiencias nuevas e informativas.

La Estructura de los Agentes: Agentes que Aprenden

- El punto es que si sólo nos guiamos por el elemento del desempeño, va a seguir ejecutando las mejores acciones dado lo que ya sabe. Pero si el agente está dispuesto a explorar un poco y ejecutar tal vez acciones sub-óptimas con la esperanza de mejorar su desempeño en el largo plazo.
- El objetivo del “generador del problemas” es sugerir acciones exploratorias. Por ejemplo, identificar ciertas áreas de comportamiento que necesitan mejora y sugerir experimentos, como probar los frenos en diferentes terrenos bajo diferentes condiciones.

La Estructura de los Agentes: Agentes que Aprenden

- El elemento encargado del aprendizaje puede aprender patrones del ambiente. También del resultado de sus acciones.
 - Por ejemplo, un agente taxista puede ver que ciertas formas de conducir llevan a que reciba menos propinas.
 - También podría aprender que ciertas rutas minimizan su propina en pro de otras.

Referencias

- “Artificial Intelligence: A Modern Approach”, 3rd Edition, Capítulo 2.