

Sistemas Multi-Agentes

Agencia

Dr. Alejandro Guerra-Hernández

Universidad Veracruzana

Instituto de Investigaciones en Inteligencia Artificial
Campus Sur, Calle Paseo Lote II, Sección Segunda No 112,
Nuevo Xalapa, Xalapa, Ver., México 91097

`aguerra@uv.mx`

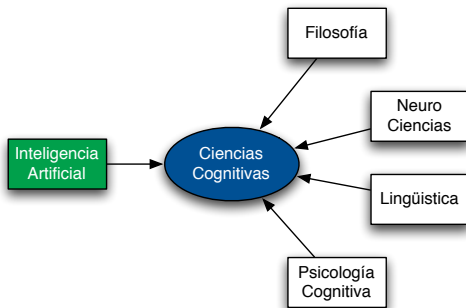
`https://www.uv.mx/personal/aguerra/`

Maestría en Inteligencia Artificial 2023



La IA como una Ciencia Cognitiva

- ▶ La Inteligencia Artificial (IA) tiene como objetivo el estudio de las **entidades inteligentes**.

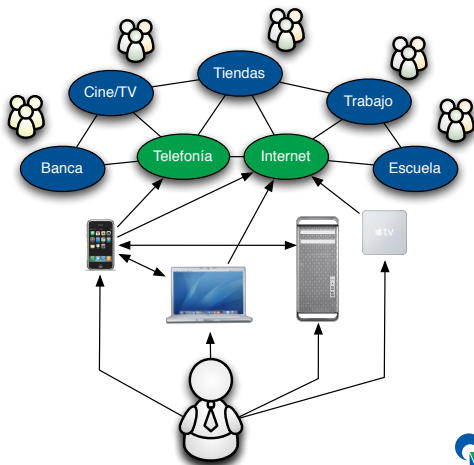


- ▶ *Modus operandi*: La construcción de **agentes racionales** [10].



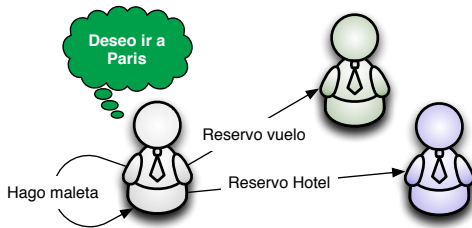
La IA como Ingeniería

- ▶ Ubicuidad
- ▶ Interconexión
- ▶ Inteligencia
- ▶ Delegación
- ▶ Homocentrismo



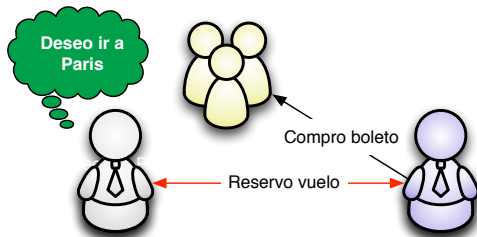
Agencia y Filosofía

- ▶ Desde Aristóteles [1], los filósofos usan el término **agente** para referirse a una entidad que **actúa** con un **propósito** dentro de un **contexto social**.



Agencia, Derecho y Economía

- ▶ Desde el derecho romano, un **agente** es la persona que actúa en beneficio de otra con un propósito específico, bajo **delegación** limitada de autoridad y responsabilidad.
- ▶ Esta noción se usa ampliamente en economía [8]



Agencia e Inteligencia Artificial

- ▶ Definición consensual de agente:
« Un agente es un **sistema computacional** capaz de **actuar** de manera **autónoma** para satisfacer sus **objetivos y metas**, mientras se encuentra **situado persistentemente** en su **medio ambiente**. »

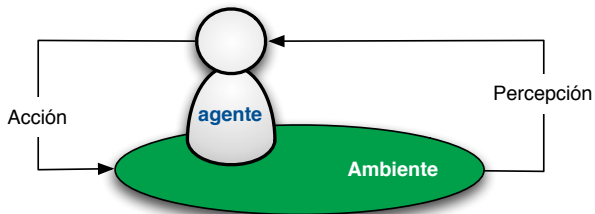
Wooldridge y Jennings [13], Russell y Norvig [10]

Ventajas de esta definición

- ▶ Considera las facultades cognitivas de los agentes al servicio de encontrar **cómo hacer lo correcto**.
- ▶ Considera **diferentes tipos de agente**.
- ▶ Considera **diferentes especificaciones sobre los sub-sistemas** que componen los agentes.

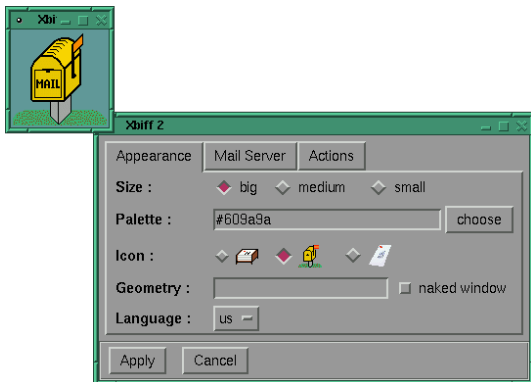
Abstracción de un agente

- ▶ Se centra en la **interacción** entre el agente y su ambiente.



Ejemplo de agente: xbiff

- ▶ Propósito: Avisarme si tengo correo nuevo.



Racionalidad: Do the right thing

- ▶ Un agente es **racional** si hace lo correcto.
- ▶ Una primera aproximación: Una acción correcta causa que el agente tenga un mayor **éxito**.
- ▶ Esto reduce el problema de la racionalidad a definir **cómo** se debe evaluar el “éxito” del agente.

Medida de desempeño

- ▶ Es el criterio usado para determinar el **éxito** de un agente.
- ▶ Es preferible que la medida sea **objetiva**, impuesta por alguna forma de **autoridad**.
- ▶ Bajo ninguna circunstancia, el agente puede **manipular** tal estándar.

Ejemplo de medida de desempeño

- ▶ La medida de desempeño para `xbiff` podría ser:

$$md(R) = \frac{\text{notificaciones}(R)}{\text{mensajesRecibidos}(R)} \quad (1)$$

Exito esperado

- ▶ La racionalidad de un agente se define, en términos más precisos, en relación con el **éxito esperado** dado lo que el agente ha **percibido**.
- ▶ No podemos exigir a un agente que tome en cuenta lo que no puede percibir, o haga lo que sus efectores no pueden hacer.

Factores de racionalidad

- ▶ La **medida de desempeño**.
- ▶ La **secuencia de percepciones** del agente.
- ▶ El **conocimiento** del agente.
- ▶ Las **acciones** del agente.

Agente ideal

- ▶ Es aquel que para toda secuencia de percepciones posible, selecciona y ejecuta una acción que se espera **maximice** la medida de desempeño, con base en la información que proveen su percepción y conocimiento sobre el ambiente.
- ▶ Es posible describir un agente por medio de un **mapeo** percepción-acción.
- ▶ Un mapeo ideal describe un agente ideal.

Ejemplo: mapeo ideal para `xbiff`

- ▶ En el caso de `xbiff` la función ideal es muy sencilla. Si `biff` avisa que llegó un mensaje nuevo, desplegar el ícono de mensaje nuevo; en cualquier otro caso, desplegar el ícono de mensajes leídos:

$$selAcc(Per \leftarrow check()) = \begin{cases} set() & \text{Si } Per = true \\ unset() & \text{En cualquier otro caso} \end{cases}$$

donde: `check()` es una acción de `xbiff` que regresa `true` si hay un mensaje nuevo en el buzón; `set()` es la acción que sube la bandera en la interfaz; y `unset()` la baja.

No todo el que actúa es inteligente

- ▶ Independientemente de la implementación usada para construir a `xbiff`, **no resulta natural** identificar a los daemons de UNIX como agentes; y menos aún como agentes inteligentes.
- ▶ Otro infame ejemplo: el apagador de luz.

Comportamiento autónomo y flexible

- ▶ **Reactividad.** Capacidad de percibir el medio ambiente y responder a tiempo a los cambios en él, a través de acciones.
- ▶ **Iniciativa.** Capacidad de exhibir un comportamiento orientado por sus metas, tomando la iniciativa para satisfacer sus objetivos de diseño (*pro-activeness*).
- ▶ **Sociabilidad.** Capacidad de interactuar con otros agentes, posiblemente tan complejos como los seres humanos, con miras a la satisfacción de sus objetivos.

Autonomía en detalle según Covrigaru y Lindsay [4] I

- ▶ Comportamiento **orientado por sus metas** y capacidad de **seleccionar** que meta va a procesar a cada instante.
- ▶ Activo en un **período** relativamente mayor al necesario para satisfacer sus metas.
- ▶ Es lo suficientemente **robusto** como para seguir siendo viable a pesar de los cambios en el ambiente.
- ▶ Interacción con su ambiente en la modalidad de **procesamiento de información**.
- ▶ **Variedad de respuestas**, incluyendo movimientos de adaptación fluidos; y su atención a los estímulos es **selectiva**.



Autonomía en detalle según Covrigaru y Lindsay [4] II

- ▶ Ninguna de sus funciones, acciones o decisiones, está totalmente **gobernada** por un agente externo.
- ▶ No necesita ser programado nuevamente por un **agente externo**.



Autonomía y sociedad

- ▶ En economía política se conocen como **bienes primarios**, a aquellos que son medios necesarios para que el agente tenga mayor éxito para satisfacer sus “intenciones” y avanzar en su plan de vida a largo término.
- ▶ Es necesaria una pluralidad de tales bienes para configurar lo que él llama **contexto de elección**. Esta pluralidad es posible únicamente si el agente tiene una relación cercana con su ambiente **social y cultural**.
- ▶ En el contexto de Inteligencia Artificial, Newell [9] ofrece argumentos similares.

Metas y autonomía

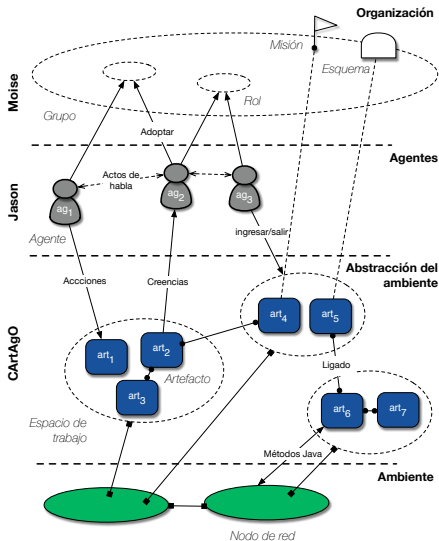
- ▶ Ser autónomo, depende no sólo de la habilidad para seleccionar metas u objetivos de entre un conjunto de ellos, ni de la habilidad de formularse nuevas metas, sino de tener el **tipo adecuado** de metas.
- ▶ Los agentes artificiales son usualmente diseñados para llevar a cabo tareas por nosotros, de forma que debemos **comunicarles** que es lo que esperamos que hagan.
- ▶ Las **metas** y las **funciones de utilidad** son dos maneras de indicarle a un agente lo que hacer, sin decirle cómo hacerlo.

¿Real o virtual?

- ▶ Por **ambiente**, entendemos el espacio donde un agente, o un grupo de ellos, se encuentra situado.
- ▶ Brooks [2] argumenta que el medio ambiente por excelencia es el **mundo real**, y en su propuesta todo agente toma una forma **robótica**.
- ▶ Etzioni [5] considera que no es necesario que los agentes tengan implementaciones robóticas porque los ambientes **virtuales**, como los sistemas operativos y el web, son igualmente válidos que el mundo real.



Tercer vía: Agentes, Artefactos y Organizaciones



Propiedades de los Artefactos

- ▶ Son **objetos reactivos** más no autónomos, ni proactivos.
- ▶ Encapsulan los **sensores** y **actuadores** a disposición de los agentes.
- ▶ Se pueden **conectar** unos con otros para obtener funcionalidades más complejas.
- ▶ Se organizan en **espacios de trabajo** que proveen una noción de sitio o localidad.
- ▶ Como **meta-modelo** pueden aproximar tanto ambientes reales, como virtuales.

Descripción PEAS

- ▶ Russell y Norvig [10] proponen una especificación informal de agentes, conocida como **PEAS** por sus siglas en inglés (*Performance, Environment, Actuators, Sensors*).

Agente	Desempeño	Ambiente	Actuadores	Sensores
taxi	seguridad, velocidad legal, comodidad ganancias	caminos, tráfico peatones, clientes	volante, acelerador freno, direccional claxón, pantalla	cámaras, sonar velocímetro, GPS acelerómetro, teclado



Percepción

Observable. Los sensores pueden percibir el estado **completo** del ambiente. **Ej.** Los juegos formales: ajedrez, damas, backgammon, etc.

Eficazmente observable. Los aspectos **relevantes** del ambiente pueden ser percibidos. La relevancia depende de la función de desempeño.

Parcialmente observable. Lo anterior no es posible. **Ej.** Algunos juegos de cartas como el poker, la calle donde manejamos.

Inobservable. No hay sensores para contender con su entorno de trabajo.



Número de Agentes e interacciones

- Monoagente.** Un agente resuelve la tarea en el entorno de trabajo, o lo hacen varios **sin interactuar**.
- Multiagente.** El caso contrario al anterior.
- Competitivo.** Un SMA con **interacciones negativas**. Ej. Los juegos con contrincante, el estacionamiento.
- Colaborativos.** Un SMA con **interacciones positivas**. Ej. El aprendizaje social.

Actuación

Determinista. El próximo estado depende exclusivamente del **estado actual** y de la **acción** que se ejecute. **Ej.** Los juegos formales.

Estocástico. Otros factores influyen en el próximo estado del ambiente. **Ej.** Los entornos parcialmente observables parecen estocásticos.

Incierto. Se dice de los ambientes estocásticos o parcialmente observables.

No determinista. Los actuadores se caracterizan en términos de sus **posibles resultados**, pero ninguna probabilidad es asociada a ellos.

Estratégico. El entorno solo depende de lo que haga el **contrincante**.

Desempeño

Episódico. El desempeño del agente y/o sus acciones, se evalúan en **rondas**. La calidad de una acción en una ronda, no depende de las rondas previas.

Secuencial. El caso contrario.

Cambio

Dinámico. El ambiente puede **cambiar mientras** el agente se encuentra deliberando.

Estático. El caso contrario.

Semi-dinámico. El ambiente no cambia, pero las acciones tienen **peor desempeño** conforme pasa el tiempo. **Ej.** Perforación petrolera.

Discretización

Discreto. Se considera un **número limitado de posibles estados**, distintos y claramente definidos.

Continuo. El caso contrario.

- ▶ Estas propiedades aplican a los **estados** de un ambiente, a la forma en que se registra el **tiempo**, y a las percepciones y acciones en general.

Representación

- Atómica.** Cada estado del ambiente es **indivisible** –No tiene estructura interna. **Ej.** Búsquedas, Modelos de Markov, Aprendizaje por refuerzo.
- Factorizada.** Los estados se definen en términos de un conjunto de **variables** o **atributos**, cada uno de los cuales toma un **valor**. **Ej.** Satisfacción de restricciones, lógica proposicional, planeación, redes bayesianas, aprendizaje supervisado.
- Estructurada.** El ambiente se concibe como compuesto por **objetos** que están **relacionados** entre ellos. **Ej.** Bases de datos relacionales, lógica de primer orden, modelos probabilistas de primer orden, aprendizaje basado en conocimiento y comprensión del lenguaje natural.



Diferentes ambientes

Ambiente	Observ.	Determ.	Episódico	Estático	Discreto	SMA
Crucigrama	si	si	no	si	si	mono
Ajedrez con reloj	si	estratégico	no	semi	si	multi
Backgammon	si	estocástico	no	si	si	multi
Poker	parcial	estocástico	no	si	si	multi
Tutor inglés	parcial	estocástico	no	no	si	multi
Brazo robótico	efectivo	estocástico	si	no	no	mono
Control refinería	parcial	estocástico	no	no	no	mono
Robot navegador	parcial	estocástico	no	no	no	mono
Análisis imágenes	si	si	si	semi	no	mono
Manejo de autos	parcial	estocástico	no	no	no	multi
Diagnóstico	parcial	estocástico	no	no	no	mono



Arquitectura abstracta de Wooldridge [12]

- ▶ El **ambiente** puede caracterizarse por medio de un conjunto finito de estados discretos posibles, definido como:

$$E = \{e, e', \dots\}$$

- ▶ La **competencia** de un agente, se define como el conjunto finito de acciones que éste puede ejecutar:

$$Ac = \{\alpha, \alpha', \dots\}$$

- ▶ Una **corrida** de un agente en un ambiente se define como una secuencia finita de estados y acciones intercalados:

$$r = e_0 \xrightarrow{\alpha_0} e_1 \xrightarrow{\alpha_1} e_2 \xrightarrow{\alpha_2} e_3 \xrightarrow{\alpha_3} \dots \xrightarrow{\alpha_{u-1}} e_u$$

Corridas

- ▶ Sea R el conjunto de todas las posibles secuencias finitas sobre E y Ac .
- ▶ Definimos R^{Ac} como el subconjunto de las corridas que terminan en una acción
- ▶ y R^E como el subconjunto de las corridas que terminan en un estado del ambiente.
- ▶ Para modelar el efecto de una acción en el ambiente, usamos una función de transición (Fagin [6]):

$$\tau : R^{Ac} \rightarrow \wp(E)$$

- ▶ Si $\tau(r) = \emptyset$ para todo $r \in R^{Ac}$, se dice que el sistema ha terminado su corrida.

Ambiente y Agentes

- ▶ Un **ambiente** se define como una tripleta $Env = \langle E, e_0, \tau \rangle$ donde E es el conjunto de los posibles estados del ambiente, $e_0 \in E$ es un estado inicial y τ es la función de transición de estados.
- ▶ Los **agentes** se modelan como funciones que mapean corridas que terminan en un estado del ambiente, a acciones:

$$Ag : R^E \rightarrow Ac$$



Sistema Agente

- ▶ Un **sistema agente** es una tupla conformada por un agente y un ambiente.
- ▶ El conjunto de **posibles corridas** del agente Ag en el ambiente Env se denota como $R(Ag, Env)$
- ▶ Una secuencia de la forma: $(e_0, \alpha_0, e_1, \alpha_1, e_2, \dots)$ representa una **corrida del agente** Ag en el ambiente Env ssi $Env = \langle E, e_0, \tau \rangle$; $\alpha_0 = Ag(e_0)$; y para $i > 0$:

$$e_i \in \tau((e_0, \alpha_0, \dots, \alpha_{i-1}))$$

y

$$\alpha_i = Ag((e_0, \alpha_0, \dots, e_i))$$



Programa de agente (mapeo ideal)

```
1: function agenteldeal(Per)
2:    $R^E \leftarrow \text{push}(\textit{Per}, R^{Ac})$ 
3:   acción  $\leftarrow \textit{Ag}(R^E)$ 
4:    $R^{Ac} \leftarrow \text{push}(\textit{acción}, R^E)$ 
5:   return  $R^{Ac}$ 
6: end function
```

- ▷ $Per \in E$ es una percepción.
- ▷ $R^E, R^{Ac} = \emptyset$ inicialmente.
- ▷ $Ag : R^E \rightarrow Ac$ es la función ideal.



Programa de ambiente

```
1: procedure Ambiente( $e, \tau, Ags, fin$ )
2:   repeat
3:     for all  $Ag \in Ags$  do
4:        $acción(Ag) \leftarrow agenteIdeal(percibir(Ag, e))$ 
5:     end for
6:      $e \leftarrow \tau(\bigcup_{Ag \in Ags} acción(Ag))$ 
7:   until  $fin(e)$ 
8: end procedure
```

▷ $e \in E$ estado inicial.

▷ τ función de transición del ambiente.

▷ fin es un predicado de fin de corrida.



Percepción y acción

- ▶ Sea Per un conjunto no vacío de percepciones, la función **percibir/2** se define como:

$$percibir : E \rightarrow Per$$

- ▶ La función **actuar/2** se define entonces como:

$$actuar : Per \rightarrow Ac$$

- ▶ Un **agente** puede definirse ahora como la tupla:

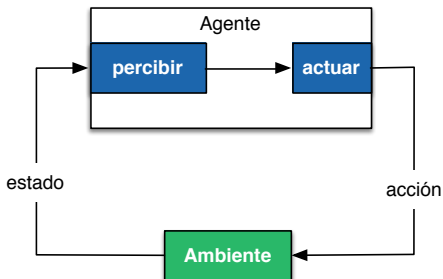
$$Ag = \langle percibir, actuar \rangle$$

Propiedades de la percepción

- ▶ Sean $e \in E$ y $e' \in E$, tal que $e \neq e'$ pero $\text{percibir}(e) = \text{percibir}(e')$. Desde el punto de vista del agente, e y e' son **indistinguibles**.
- ▶ Dados dos estados del ambiente $e, e' \in E$, $\text{percibir}(e) = \text{percibir}(e')$ será denotado como $e \sim e'$.
- ▶ El ambiente es accesible para el agente, si y sólo si $|E| = |\sim|$ y entonces se dice que el agente es **omnisciente**.
- ▶ Si $|\sim| = 1$ entonces se dice que el agente no tiene capacidad de percepción, es decir, el ambiente es percibido por el agente como si tuviera un **estado único**.

Agentes reactivos

- ▶ Los **agentes reactivos**, o reflex, seleccionan sus acciones basados en su percepción actual del ambiente, ignorando el resto de su historia perceptual.



Ejemplo: Xbiff

- ▶ Los estados posibles del ambiente, las acciones y la función de selección de acción para `xbiff`:

$$percibir \leftarrow check() \in \{true, false\}$$

$$actuar = \begin{cases} set() & \text{Si } percibir = true \\ unset & \text{En cualquier otro caso} \end{cases}$$

Reglas para Xbiff

1. If *percibir* = *true* then *set()*
2. If *true* then *unset()*.

Programa de agente reactivo

```
1: function Agente-Reactivo(e)
2:   estado ← percibir(e)
3:   regla ← selecciónAcción(estado, reglas)      ▷ reglas condición-acción.
4:   acción ← cons(regla)                       ▷ cons, el consecuente de la regla.
5:   return acción
6: end function
```



Limitaciones de los agentes reactivos

- ▶ Existen otras maneras de implementar agentes reactivos. Ej. La arquitectura subsumida de Brooks [3] o las redes de comportamiento de Maes [7].
- ▶ Todas comparten una **limitación formal**: Producen un comportamiento racional, sólo si la decisión correcta puede obtenerse a partir de la **percepción actual** del agente.
- ▶ Esto es, su comportamiento es correcto si, y sólo si, el ambiente es **observable** o **efectivamente observable**.

Estado interno

- ▶ La forma más natural de enfrentar un ambiente inaccesible es llevando un **registro** de lo percibido, de forma que el agente tenga acceso a lo que en cierto momento ya no puede percibir.
- ▶ Sea I el conjunto de estados internos posibles de un agente. Redefinimos la función **actuar** para mapear estados internos a acciones posibles:

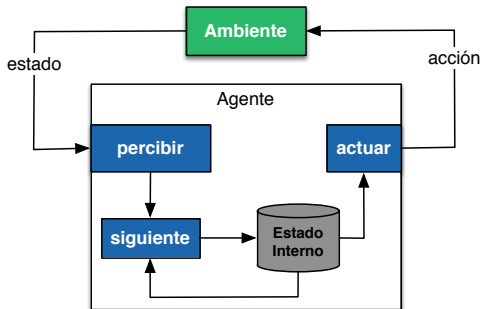
$$\text{actuar} : I \rightarrow Ac$$

- ▶ Una nueva función **siguiente/2**, mapea estados internos y percepciones a estados internos. Se usa para actualizar el estado interno del agente:

$$\text{siguiente} : I \times Per \rightarrow I$$

Agentes con estado interno

- ▶ Un agente con **estado interno** interactúa con su ambiente como se muestra.



Programa de agente con estado

```
1: function agente-Con-Estado(e)
2:    $p \leftarrow \text{percibir}(e)$ 
3:    $\text{estado} \leftarrow \text{siguiente}(\text{estado}, p)$ 
4:    $\text{regla} \leftarrow \text{selecciónAcción}(\text{estado}, \text{reglas})$ 
5:    $\text{acción} \leftarrow \text{cons}(\text{regla})$    ▷  $\text{cons}/1$  computa la consecuencia de una regla.
6:   return acción
7: end function
```



Enfoque IA tradicional

- ▶ El comportamiento racional puede obtenerse a partir de una **representación simbólica** del ambiente y el comportamiento deseado.
- ▶ El agente manipulará **sintácticamente** esta representación para actuar.
- ▶ Llevada al extremo, esta aproximación nos lleva a formular el estado de un agente como un conjunto **fórmulas lógicas** y la selección de acción como **demostración de teoremas** o **deducción lógica**.



Agentes e inferencia

- ▶ Sea L el conjunto de fórmulas bien formadas en la lógica de primer orden clásica.
- ▶ El conjunto de bases de conocimiento en L se define como $D = \wp(L)$, es decir, el conjunto de conjuntos de fbf en L . Los elementos de D se denotan Δ, Δ_1, \dots
- ▶ El estado interno del agente es siempre un miembro de D . El proceso de decisión del agente especifica mediante un conjunto de reglas de inferencia ρ .
- ▶ Escribimos $\Delta \vdash_{\rho} \psi$ si la fbf ψ puede ser validada en Δ .
- ▶ Definimos la función siguiente/2 del agente como:

$$\text{siguiente} : D \times \text{Per} \rightarrow D$$



Programa de agente lógico (inferencia)

```
1: function Selección-Acción( $\Delta : D, Ac$ )
2:   for all  $a \in Ac$  do
3:     if  $\Delta \vdash_{\rho} actuar(a)$  then
4:       return  $a$ 
5:     end if
6:   end for
7:   for all  $a \in Ac$  do
8:     if  $\Delta \not\vdash_{\rho} \neg actuar(a)$  then
9:       return  $a$ 
10:    end if
11:  end for
12:  return null
13: end function
```

- ▷ Δ base de conocimiento.
- ▷ Ac acciones.
- ▷ ρ reglas de inferencia.



Metas

- ▶ Las **metas** describen situaciones deseables para un agente, y se definen como cuerpos de conocimiento.
- ▶ Relacionadas con el concepto de **espacio de estados de un problema** compuesto por un estado inicial del ambiente, $e_0 \in E$; por un conjunto de operadores o acciones que el agente puede ejecutar para cambiar de estado; y un espacio de estados deseables.
- ▶ Implícita en la arquitectura del agente, está su **intención** de ejecutar las acciones que el **crea** le garantizan satisfacer cualquiera de sus metas.
- ▶ Esto se conoce en filosofía como **silogismo práctico**.



Las metas de un agente

- Especificación basada en **predicados**:

$$\Psi : R \rightarrow \{0, 1\}$$

- Una corrida $r \in R$ satisface la especificación ssi $\Psi(r) = 1$.
- Un **ambiente de tareas** se define entonces como el par $\langle Env, \Psi \rangle$.
- Dado un ambiente de tareas, la siguiente expresión:

$$R_{\Psi}(Ag, Env) = \{r | r \in R(Ag, Env) \wedge \Psi(r)\}$$

denota el conjunto de todas las corridas del agente Ag en el ambiente Env que satisfacen la tarea especificada por Ψ .



Metas y éxito

- ▶ Podemos expresar que un agente Ag tiene éxito en el ambiente de tareas $\langle Env, \Psi \rangle$ de dos maneras diferentes:
 - ▶ $\forall r \in R(Ag, Env)$ tenemos que $\Psi(r)$, lo que puede verse como una especificación pesimista de éxito, puesto que el agente tiene éxito únicamente si todas sus corridas satisfacen Ψ ;
 - ▶ $\exists r \in R(Ag, Env)$ tal que $\Psi(r)$, lo cual es una versión optimista de la definición de éxito, puesto que especifica que el agente tiene éxito si al menos una de sus corridas satisface Ψ .

Utilidad

- ▶ Una **utilidad** es un valor numérico que denota la bondad de un estado del ambiente.
- ▶ Implícitamente, un agente tiene la “intención” de alcanzar aquellos estados que **maximizan** su utilidad a largo término.
- ▶ La especificación de una tarea en este enfoque corresponde simplemente a una **función utilidad** $u : E \rightarrow \mathbb{R}$ la cual asocia valores reales a cada estado del ambiente.
- ▶ Por ejemplo, la utilidad para una corrida r de un **agente filtro de spam**, puede definirse como:

$$u(r) = \frac{SpamFiltrado(r)}{SpamRecibido(r)}$$



Agentes óptimos

- ▶ Si la función de utilidad u tiene algún **límite superior**, por ej., $\exists k \in \Re$ tal que $\forall r \in R. u(r) \leq k$, entonces es posible hablar de agentes que **maximizan** la utilidad esperada.
- ▶ Definamos $P(r|Ag, Env)$, es evidente que:

$$\sum_{r \in R(Ag, Env)} P(r|Ag, Env) = 1$$

- ▶ Entonces el **agente óptimo** Ag_{opt} entre el conjunto de agentes posibles AG en el ambiente Env está definido como:

$$Ag_{opt} = \arg \max_{Ag \in AG} \sum_{r \in R(Ag, Env)} u(r)P(r|Ag, Env)$$



Racionalidad acotada

- ▶ Los agentes enfrentan limitaciones temporales y tienen capacidades limitadas de deliberación, lo que sugiere una **racionalidad acotada**.
- ▶ Russell y Wefald [11] introducen el concepto de **agente óptimo acotado**, donde AG_m representa el conjunto de agentes que pueden ser implementados en una máquina m .



Ejercicios sugeridos

- ▶ Piense un poco en su tema de tesis y describa un agente y su medio ambiente en ese contexto. Base su descripción en el esquema PEAS.
- ▶ ¿Existe una medida de desempeño para su agente? ¿Puede formularla?
- ▶ ¿Es posible agrupar las percepciones y acciones de su agente en artefactos? Describa un ejemplo.
- ▶ ¿Qué arquitectura elegiría para su agente? Justifique la respuesta.



Referencias I

- [1] Aristóteles. "Prior Analytics No 391". En: *Loeb Classical Library*. Cambridge, MA, USA: Harvard University Press, 1960.
- [2] RA Brooks. *Cambrian Intelligence: the Early History of the New AI*. Cambridge, MA, USA: The MIT Press, 1999.
- [3] R Brooks. "A Robust Layered Control System for a Mobile Robot". En: *IEEE Journal of Robotics and Automation* 2.1 (1986), págs. 14-23.
- [4] AA Covrigaru y RK Lindsay. "Deterministic Autonomous Systems". En: *AI Magazine* 12.3 (1991), págs. 110-117.
- [5] O Etzioni. "Intelligence without Robots". En: *AI Magazine* 14.4 (1993).
- [6] R Fagin et al. *Reasoning about Knowledge*. Cambridge, MA, USA: The MIT Press, 1995.
- [7] P Maes. "A Spreading Activation Network for Action Selection". En: *Intelligent Autonomous Systems 2*. Amsterdam, the Netherlands: IOS Press, 1989, págs. 875-885.
- [8] W Muller-Freienfels. "Agency". En: *Encyclopedia Britannica*. Internet version. Encyclopedia Britannica, Inc., 1999.
- [9] A Newell. *Unified Theories of Cognition*. Cambridge, MA, USA: Harvard University Press, 1990.

Referencias II

- [10] SJ Russell y P Norvig. *Artificial Intelligence: A Modern Approach*. Third. Prentice Hall Series in Artificial Intelligence. USA: Prentice Hall, 2009.
- [11] SJ Russell y E Wefald. *Do the Right Thing: Studies in Limited Rationality*. Cambridge, MA, USA: The MIT Press, 1991.
- [12] M Wooldridge. *An Introduction to MultiAgent Systems*. 2nd. West Sussex, England: John Wiley & Sons, LTD, 2009.
- [13] M Wooldridge y N Jennings. "Intelligent Agents: Theory and practice". En: *The Knowledge Engineering Review* 10.2 (1995), págs. 115-152.