Agentes guiados por utilidad

Sistemas Inteligentes Distribuidos

Sergio Alvarez
Javier Vázquez

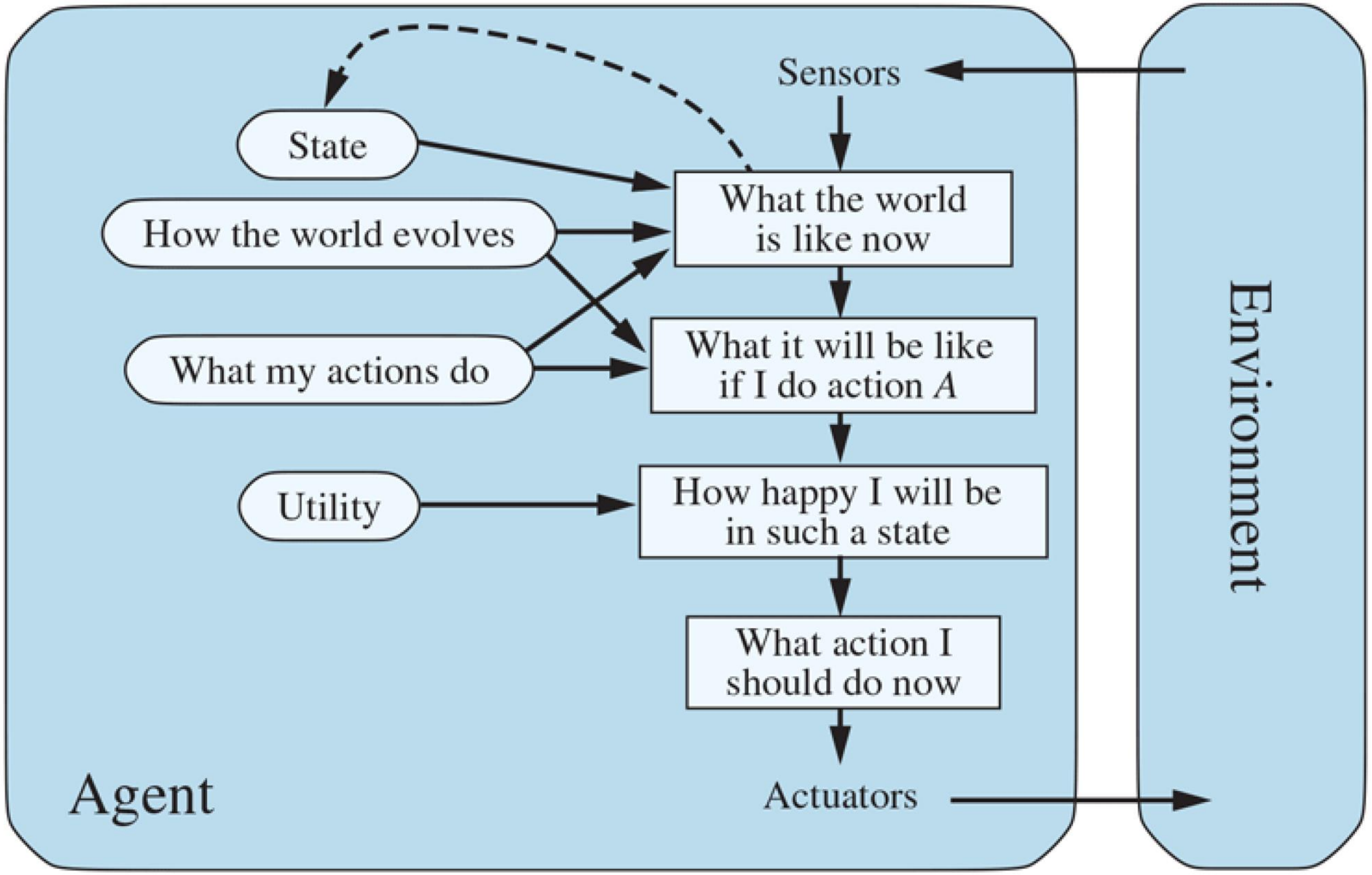
Bibliografía

- *Artificial intelligence: a modern approach* (Russell & Norvig), cap. 2, 16
- *Reinforcement Learning: An Introduction* (Sutton & Barto), cap. 3, 4
- *Multi-Agent Reinforcement Learning* (Albrecht et al.), cap. 2

Utilidad

Agentes guiados por utilidad

Agente deliberativo por utilidad



Objetivos vs utilidad

- Dependiendo del entorno, es posible que guiar por objetivos no sea lo más adecuado
  - ¿Qué ocurre si tenemos diversas maneras válidas de cumplir con los objetivos? ¿Son todas igual de racionales?
    - La única distinción que tenemos es *valido* o *no válido* según el formalismo lógico que escojamos
  - ¿Cómo gestionamos entornos parcialmente observables o no deterministas?
    - Si aumentamos la complejidad de la representación de los objetivos, podemos aumentar la complejidad del razonamiento
- Alternativa: función de utilidad
  - Representación subsimbólica en lugar de simbólica

Utilidad y recompensa

Agentes guiados por utilidad

Función de utilidad

- Internalización de una métrica numérica de rendimiento
  - Idealmente, alineada con una optimización de la racionalidad
- Pros:
  - Más flexibilidad, e.g. gestión de la incertidumbre
  - Más adaptable
  - Permite diferenciar entre posibilidades válidas
- Contras
  - Partimos de la suposición de que es posible reducir la racionalidad a un valor numérico
  - ¿De dónde sale esta función de utilidad? Normalmente no pensamos en términos de una función de este tipo
  - Según como se modele la utilidad, es posible que no haya transparencia en la toma de decisiones
  - No hay una única manera de modelar utilidades

Utilidad por estado

- Una función de utilidad (o también función de valor) se define como:

$$\mathcal{U}: S \to \mathbb{R}$$

  que asocia un número real a cada estado del entorno
- Sin embargo, cuál es el valor de una *ejecución*…
  - ¿es la utilidad máxima de un estado dentro de una ejecución?
  - ¿es la suma de utilidades de todos los estados de una ejecución?
  - ¿es la media aritmética de las utilidades de todos los estados de una ejecución?
- **Problema:** es difícil especificar una visión a largo plazo cuando asignamos utilidades a estados individuales
  - Veremos más adelante el concepto de factor de descuento para abordar esto
- **Problema:** no tenemos en cuenta las acciones tomadas por el agente

Utilidad por ejecución

- Otra posibilidad: asignar una utilidad no a estados individuales, sino a ejecuciones (secuencias de *pasos*)

$$\mathcal{U}: S \times \cdots \times S \rightarrow \mathbb{R}$$

- Este enfoque implementa una visión a largo plazo
- **Problema:** hemos de tener en cuenta la probabilidad de la ocurrencia de los estados
- ¿Cómo podemos combinar estas dos visiones, estado vs ejecución?

Señal de recompensa

- Señal (o función) que guía al agente basándose en:
  - el estado en el que estaba,
  - la acción que ha tomado y
  - el estado al que llega

$$\mathcal{R}: S \times A \times S \to \mathbb{R}$$

- La función de utilidad se puede calcular a partir de esta señal
  - Habitualmente: **el valor esperado de la utilidad es una función sobre la suma de recompensas recibidas durante la ejecución**
- ¿De dónde viene esta señal? Generalmente, del entorno

La hipótesis de recompensa

That all of ***what we mean by goals and purposes*** *can be well thought of as* ***maximization of the expected value of the cumulative sum of a received scalar signal*** *(reward).*

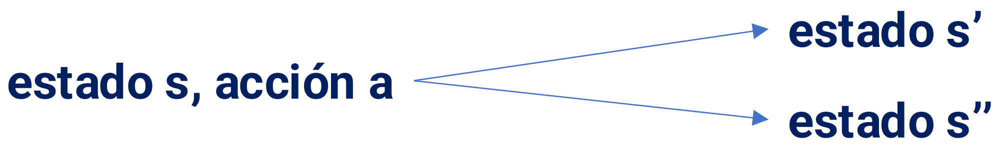
The reward hypothesis, Richard Sutton
http://incompleteideas.net/rlai.cs.ualberta.ca/RLAI/rewardhypothesis.html

Problemas de decisión secuencial

Agentes guiados por utilidad

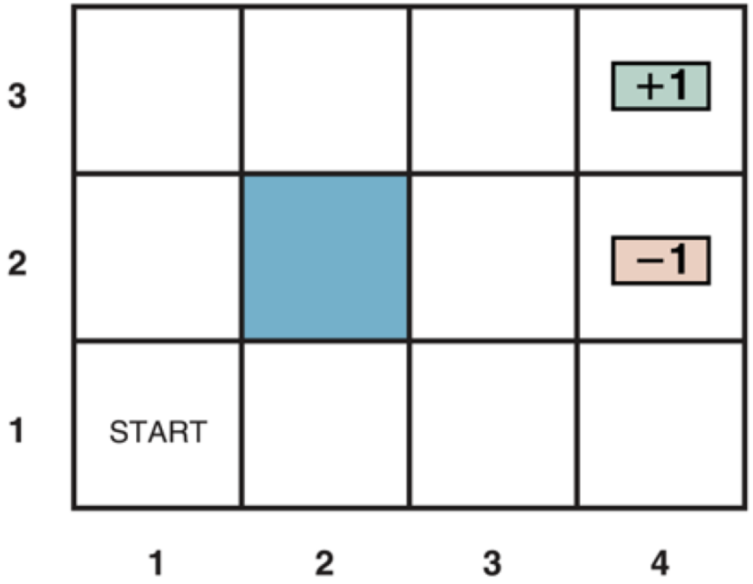
Problemas de decisión secuencial

- Queremos modelar el proceso de razonamiento que permite a un agente tomar decisiones en un entorno
- Vamos a suponer (por ahora) que…
    - Tenemos acceso a una señal de recompensa
    - El entorno es observable
    - Las acciones del agente sobre el entorno tienen un efecto aleatorio pero conocido: el entorno es determinista y estocástico

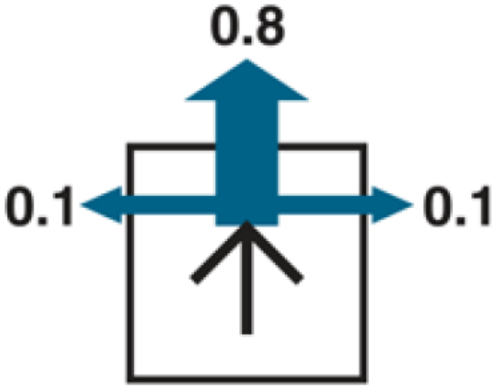


Problemas de decisión secuencial

- Ejemplo: Grid World 4x3 (AIMA)



La acción **moverse** tiene efecto estocástico: 80% de moverse en la dirección intencionada, 0% si la dirección es una pared
Recompensa = -0.04 tras moverse, excepto en los estados finales marcados



- ¿Qué estrategia debería seguir el agente?

Problemas de decisión secuencial

- Ejemplo: dados
  - CS221 - Stanford
- Para cada turno $t = 1, 2, \ldots$
  - Escoge: **jugar** o **parar**
  - Efecto de **jugar:**
    - Recibes 4 euros
    - Lanzas un dado de 6 caras
      - Si el resultado es 1 o 2, se acaba el juego
      - Si el resultado es otro, se avanza al siguiente turno
  - Efecto de **parar:**
    - Recibes 10 euros
    - Se acaba el juego

Formalización

- Ambos ejemplos tienen elementos comunes
- Queremos ser capaces de formalizar, de manera abstracta, cualquier problema de este tipo
- ¿Nos sirven los métodos que hemos visto hasta ahora?
  - Algoritmos de búsqueda
  - Deliberación por objetivos

Formalización: MDPs

Un proceso de decisión de Markov (MDP) se define como una tupla $\langle \boldsymbol{S}, \boldsymbol{A}, \boldsymbol{\mathcal{R}}, \boldsymbol{\mathcal{T}}, \boldsymbol{\mu}, \boldsymbol{\gamma} \rangle$ tal que:

- S es el conjunto de estados, del cual $\bar{S} \subset S$ es el conjunto de estados finales
- A es el conjunto de acciones
- $\mathcal{R}: S \times A \times S \to \mathbb{R}$ es la función de recompensa, normalmente formulada como $r(s, a, s')$
- $\mathcal{T}: S \times A \times S \to [0,1]$ es la función de probabilidad de transición entre estados, normalmente formulada como $p(s'|s, a)$
- $\mu: S \to [0,1]$ es la función de distribución del estado inicial
- $\gamma \in [0,1]$ es el factor de descuento

Formalización: MDPs

- La función de transición $\mathcal{T}$ cumple:

$$\forall s \in S \setminus \bar{S}, \forall a \in A: \sum_{s' \in S} p(s'|s,a) = 1$$

$$\forall s, s' \in \bar{S}, \forall a \in A: \, p(s'|s,a) = 0$$

- La función de distribución del estado inicial μ cumple:

$$\sum_{s \in S} \mu(s) = 1$$

$$\forall s \in \bar{S}: \mu(s) = 0$$

Formalización: MDPs

- Es posible convertir problemas de búsqueda en MDPs
  - Si suponemos que un algoritmo de búsqueda ramifica en base a una función de sucesores $Sucesor(s, a)$ entonces:

$$p(s'|s,a) = \begin{cases} 1 & \text{si } s' = Sucesor(s,a) \\ 0 & \text{en cualquier otro caso} \end{cases}$$

  - También habría que reconvertir la función de coste en $\mathcal{R}$, maximizando en lugar de minimizando
- ¿Podemos usar el formalismo de MDPs para encontrar estrategias óptimas para un agente?
  - ¿Nos hace falta tener un histórico de las decisiones pasadas tomadas por el agente?

Suposición de Markov

- El futuro es independiente del pasado
- **El próximo estado y el siguiente valor de recompensa son dependientes, únicamente, del estado actual y de la siguiente acción tomada por el agente**
- Esta suposición permite simplificar el modelo y el cálculo de estrategias óptimas a partir de él
- Si la historia o el estado interno del agente es relevante, hay que modelar el espacio de estados de manera acorde
- Si el entorno no es totalmente observable: POMDPs

Volviendo al ejemplo de los dados…

- Para cada turno $t = 1, 2, \ldots$
  - Escoge: **jugar** o **parar**
  - Efecto de **jugar:**
    - Recibes 4 euros
    - Lanzas un dado de 6 caras
      - Si el resultado es 1 o 2, se acaba el juego
      - Si el resultado es otro, se avanza al siguiente turno
  - Efecto de **parar:**
    - Recibes 10 euros
    - Se acaba el juego

Vamos a modelar este juego como un MDP

MDP: juego de los dados

- Para cada turno $t = 1, 2, \ldots$
  - Escoge: **jugar** o **parar**
  - Efecto de **jugar:**
    - Recibes 4 euros
    - Lanzas un dado de 6 caras
      - Si el resultado es 1 o 2, se acaba el juego
      - Si el resultado es otro, se avanza al siguiente turno
  - Efecto de **parar:**
    - Recibes 10 euros
    - Se acaba el juego

$$\mathrm{S} = \{\mathrm{jugando}, \mathrm{fin}\}, \bar{\mathrm{S}} = \{\mathrm{fin}\}, \mu(\mathrm{jugando}) = 1$$
$$\mathrm{A} = \{\mathrm{jugar}, \mathrm{parar}\}$$

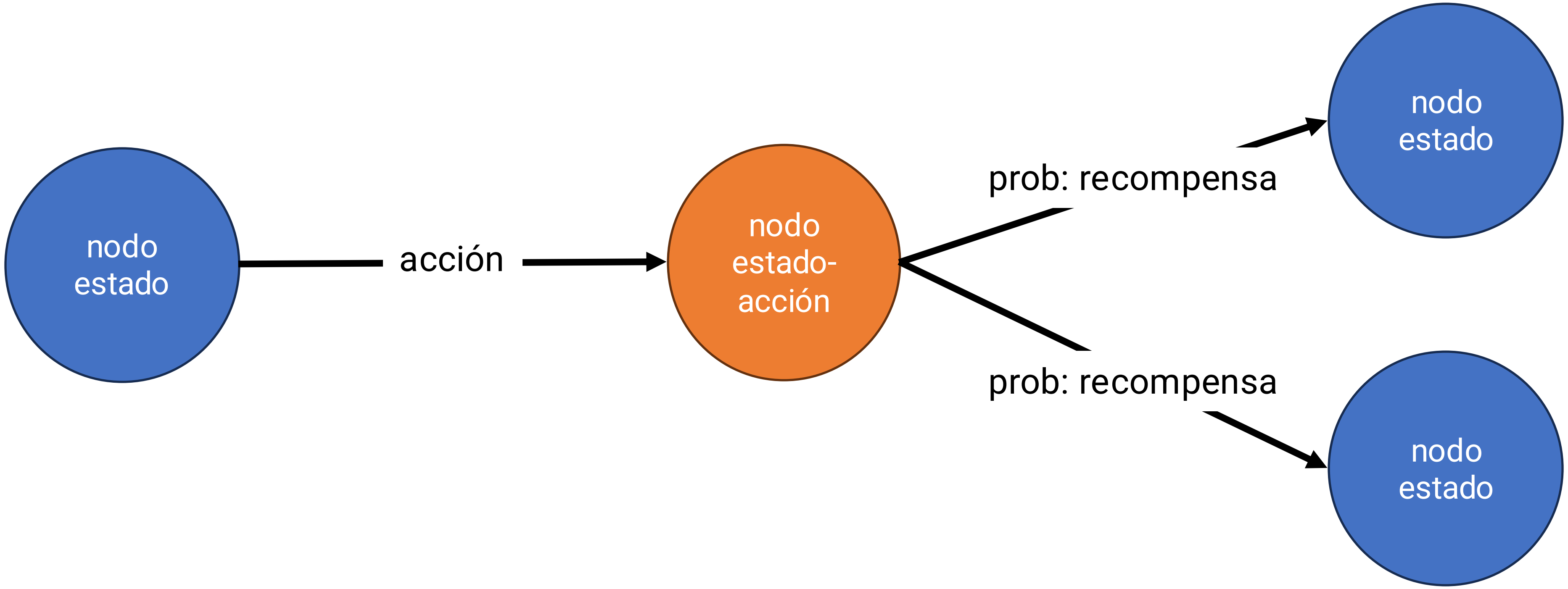
$$\mathrm{p}(\boldsymbol{s} = \mathrm{jugando} \mid \mathbf{s} = \mathrm{jugando}, \mathbf{a} = \mathrm{jugar}) = \frac{2}{3}$$

$$\mathrm{p}(\mathrm{fin} \mid \mathrm{jugando}, \mathrm{jugar}) = \frac{1}{3}$$

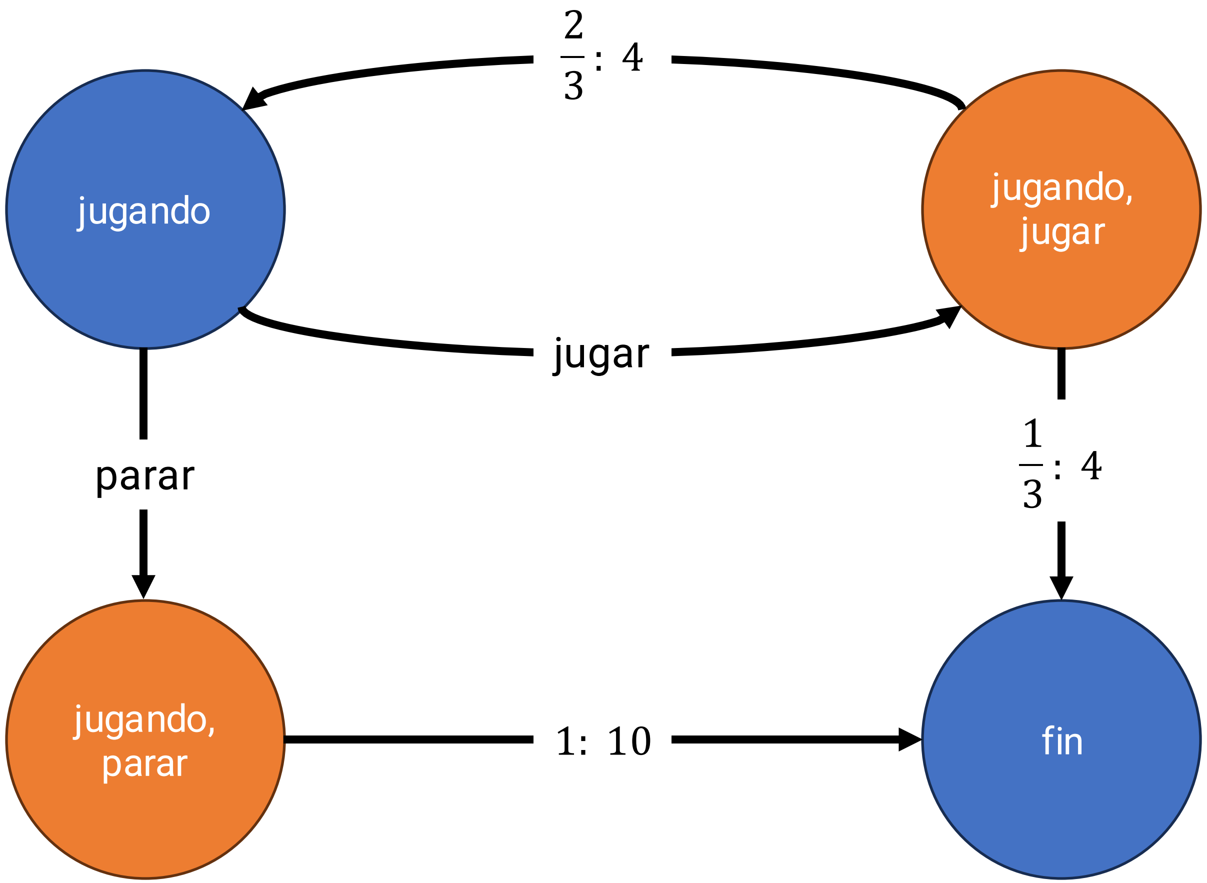
$$\mathrm{p}(\mathrm{fin} \mid \mathrm{jugando}, \mathrm{parar}) = 1$$

$$\mathcal{R}(\mathbf{s} = \mathrm{jugando}, \mathbf{a} = \mathrm{jugar}, \mathbf{s}' = \mathrm{jugando}) = 4$$
$$\mathcal{R}(\mathrm{jugando}, \mathrm{jugar}, \mathrm{fin}) = 4$$
$$\mathcal{R}(\mathrm{jugando}, \mathrm{parar}, \mathrm{fin}\ ) = 10$$

MDPs como grafos



MDP: juego de los dados



Políticas y trayectorias

Agentes guiados por utilidad

{salvarez,jvazquez}@cs.upc.edu

Política

- Una solución a un MDP se llama **política** y se define como:

$$\pi: A \times S \to [0,1]$$

- Una política es una asignación, para cada par acción $a \in A$, estado $s \in S \backslash \bar{S}$, de una probabilidad $\pi(a|s)$:
  - **$\mathbf{1}$ si $\boldsymbol{a}$ es la opción escogida** cuando el agente está en el estado s,
  - **$\mathbf{0}$ en cualquier otro caso**, i.e. para todas las acciones no escogidas en el estado s

Política

- Por ejemplo, para el juego de los dados, las dos únicas políticas posibles son:
  - $\pi(\text{jugar} \mid \text{jugando}) = 1$ y 0 en el resto de los casos, o
  - $\pi(\text{parar} \mid \text{jugando}) = 1$ y 0 en el resto de los casos
  - Es irrelevante asignar $\pi(? \mid \text{fin})$ porque es estado final
- ¿Tendría sentido tener una política que sea dinámica, es decir, que elija una acción u otra dependiendo del turno t?
- ¿Tendría sentido tener una política que asigne probabilidades en el rango $(0,1)$? Es decir, permitiendo más de una acción escogida en un estado
- Repasad los conceptos vistos hasta ahora para responder a estas preguntas

Política

- Queremos saber qué estados son buenos o malos para que la política tome buenas decisiones
- Para ello, nos vendría bien saber cuán bueno es un estado…
  - …pero mirar la recompensa inmediata no nos sirve: el futuro sigue siendo relevante
- Hay que mirar las recompensas de las **trayectorias que parten de un estado**

Trayectoria

- Dado un MDP, una política nos da **trayectorias aleatorias**
- La **utilidad** de una política se define como la **suma (*descontada*) de las recompensas** de la trayectoria
  - Por lo tanto, podemos formular la utilidad como una **variable aleatoria**
- El **valor** de una política en un estado concreto es la **utilidad esperada** en ese estado
- Una trayectoria se expresa concatenando estados y acciones:

$$[s_0, a_0, s_1, a_1, s_2, a_2, \dots]$$

- Cada trayectoria además va asociada a una secuencia de recompensas:

$$[r_0, r_1, r_2, \dots] = [\mathcal{R}(s_0, a_0, s_1), \mathcal{R}(s_1, a_1, s_2), \mathcal{R}(s_2, a_2, s_3), \dots]$$

Trayectorias del juego de los dados

Suponiendo que la utilidad es la suma de recompensas…

Trayectoria ($\pi(\text{jugando}) = \text{parar}$)	**Recompensas**	**Utilidad**
[jugando, parar, fin]	[10]	10

horizonte infinito

Trayectoria ($\pi(\text{jugando}) = \text{jugar}$)	**Recompensas**	**Utilidad**
[jugando, jugar, fin]	[4]	4
[jugando, jugar, jugando, jugar, fin]	[4, 4]	8
[jugando, jugar, jugando, jugar, jugando, jugar, fin]	[4, 4, 4]	12
[jugando, jugar, jugando, jugar, jugando, jugar, jugando, jugar, fin]	[4, 4, 4, 4]	16
…	…	…

El factor de descuento

- El elemento γ del MDP representa la **preferencia del agente en favor de las recompensas inmediatas con respecto de las recompensas futuras**
  - A medida que γ se acerca a 0, las recompensas futuras se consideran insignificantes
  - A medida γ se acerca a 1, las recompensas futuras tienden a ser tan importantes como las inmediatas
  - Cuando $\gamma = 1$, hablamos de recompensas puramente aditivas

El factor de descuento

- La fórmula de la utilidad para una trayectoria, teniendo en cuenta el factor de descuento, queda así:

$$\mathcal{U}([s_0, a_0, s_1, a_1, a_2, \dots]) = r_0 + \gamma r_1 + \gamma^2 r_2 + \ \dots = \sum_{\boldsymbol{t=0}}^{\infty} \boldsymbol{\gamma^t r_t} = \sum_{t=0}^{\infty} \gamma^t \, \mathcal{R}(s_t, a_t, s_{t+1})$$

- Teniendo en cuenta que, sea cual sea la política, cada trayectoria puede ser aleatoria (por μ y $\mathcal{T}$), el objetivo del agente es maximizar el valor esperado de la utilidad:

$$\mathbb{E}_\pi[r_0 + \gamma r_1 + \gamma^2 r_2 + \ \dots] = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{\boldsymbol{t=0}}^{\infty} \boldsymbol{\gamma^t r_t}\right] = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \, \mathcal{R}(s_t, a_t, s_{t+1})\right]$$

$\pi(a|s)$ $p(s'|s,a)$

Dos variables aleatorias

¿Por qué definir un factor de descuento?

- Para reflejar la tendencia humana a **tener más en cuenta las ganancias a corto plazo**
- Razón *económica*: las ganancias inmediatas pueden suponer un mayor margen de ahorro e **inversión**
- Si la función de transición no es perfecta, es preferible quedarse con las **recompensas más accesibles** por si las recompensas futuras son inalcanzables
- Razón *pragmática*: **hace desaparecer el problema del horizonte infinito**
- La preferencia entre trayectorias es estable, por lo que **un factor de descuento no modificará esta preferencia** (a menos que el entorno sea dinámico)

Garantías de convergencia

- Un MDP tiene una utilidad finita cuando se cumple al menos una de las siguientes condiciones:
  - El factor de descuento cumple $\gamma < 1$
    - Aplicación de la suma de progresiones geométricas infinitas

$$\mathcal{U}([s_0, a_0, s_1, a_1, \dots]) \leq \sum_{t=0}^{\infty} \gamma^t \mathcal{R}_{max} = \frac{\mathcal{R}_{max}}{1-\gamma}$$

  - El entorno define estados finales y se garantiza que uno de ellos se visita en cada trayectoria
  - No hay ciclos en el grafo generado a partir del MDP

Utilidad con descuento

Trayectoria ($\pi(\mathbf{jugando}) = \mathbf{jugar}$)	$\mathcal{R}(s, a, s')$	$\gamma = 0$	$\gamma = 0.5$	$\gamma = 1$
[jugando, jugar, fin]	[4]	4	4	4
[jugando, jugar, jugando, jugar, fin]	[4, 4]	4	6	8
[jugando, jugar, jugando, jugar, jugando, jugar, fin]	[4, 4, 4]	4	7	12
[jugando, jugar, jugando, jugar, jugando, jugar, jugando, jugar, fin]	[4, 4, 4, 4]	4	7.5	16
…	…	…	…	…

Utilidad con descuento

Trayectoria ($\pi(\text{jugando}) = \text{jugar}$)	$\mathcal{R}(s, a, s')$	$\gamma = 0$	$\gamma = 0.5$	$\gamma = 1$
[jugando, jugar, fin]	[4]	4	4	4
[jugando, jugar, jugando, jugar, fin]	[4, 4]	4	6	8
[jugando, jugar, jugando, jugar, jugando, jugar, fin]	[4, 4, 4]	4	7	12
[jugando, jugar, jugando, jugar, jugando, jugar, jugando, jugar, fin]	[4, 4, 4, 4]	4	7.5	16
…	…	…	…	…

Esta es la función que buscábamos: la que nos da el valor de un estado (V^{π})

Evaluación de política

Agentes guiados por utilidad

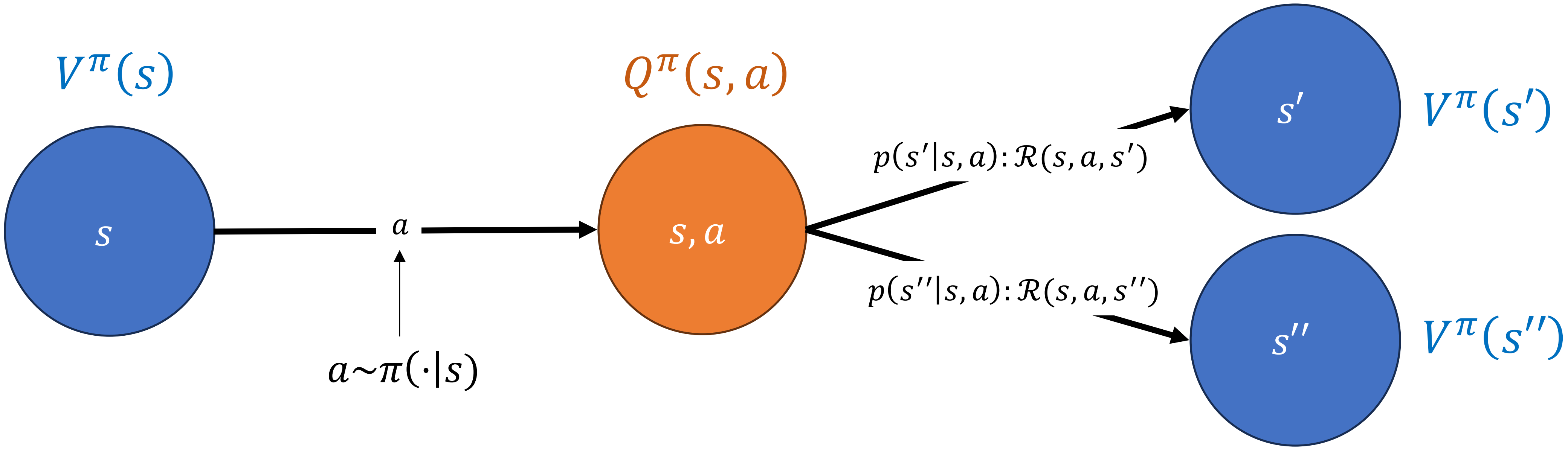
{salvarez,jvazquez}@cs.upc.edu

Evaluación de política

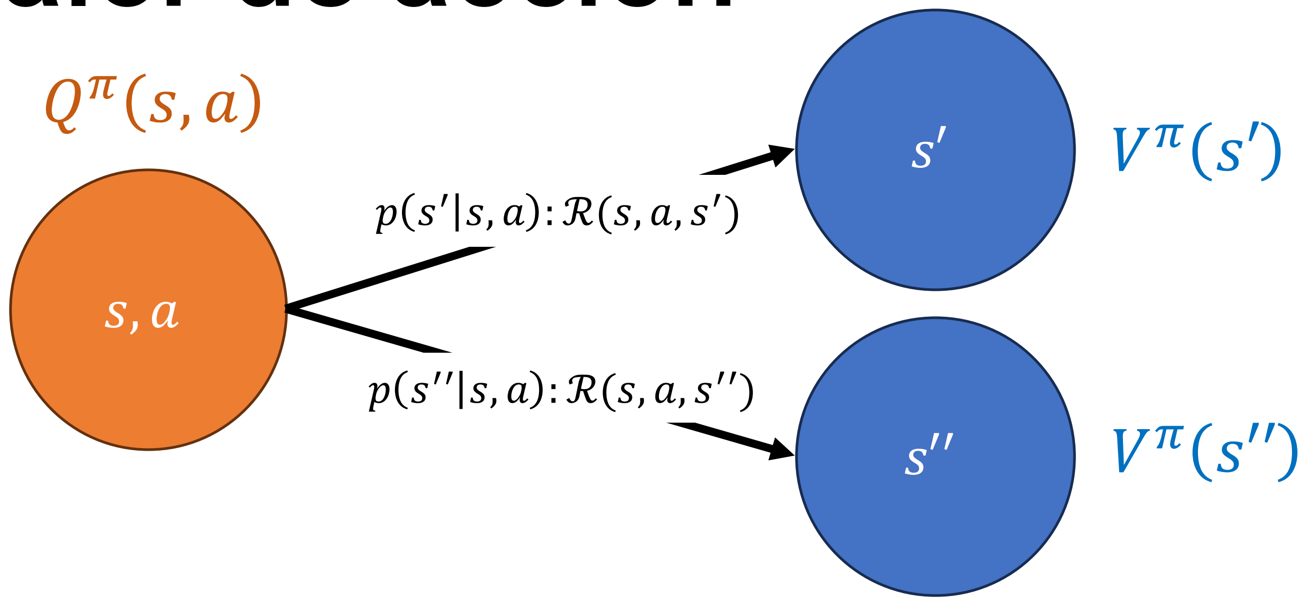
- Si se dan las siguientes premisas:
  - Tenemos acceso a una señal de recompensa
  - El entorno es observable
  - Las acciones del agente sobre el entorno tienen un efecto aleatorio pero conocido: el entorno es determinista y estocástico
  - **Hay garantía de convergencia**
  - **Conocemos la política del agente**
- Entonces podemos aplicar el **método de evaluación de política** para calcular la **utilidad esperada en cada estado**

Valor de estado, valor de acción

- **Valor de un estado** bajo una política: $\boldsymbol{V^{\pi}(s)}$
  - Utilidad esperada al seguir la política π desde el estado s
- **Valor de una acción en un estado** bajo una política: $\boldsymbol{Q^{\pi}(s,a)}$
  - Utilidad esperada al seguir la política π tomando la acción a desde el estado s

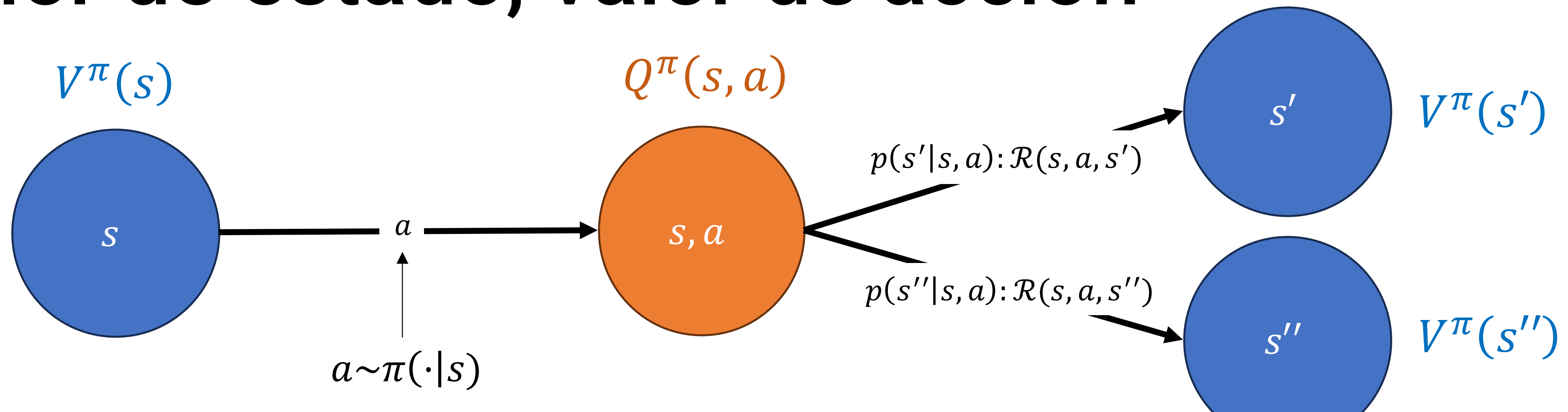


Valor de estado, valor de acción



$$Q^{\pi}(s,a) = \sum_{s' \in S} p(s'|s,a)[\mathcal{R}(s,a,s') + \gamma V^{\pi}(s')]$$

Valor de estado, valor de acción



$$Q^\pi(s,a) = \sum_{s' \in S} p(s'|s,a)[\mathcal{R}(s,a,s') + \gamma V^\pi(s')]$$

$$V^\pi(s) = \begin{cases} 0 & \text{si } s \in \bar{S} \\ Q^\pi(s,a) & \text{en cualquier otro caso, donde } a \sim \pi(\cdot\,|s) \end{cases}$$

Ecuación de Bellman

$$V^{\pi}(s) = \mathbb{E}_{\pi}[\mathcal{U}_t | s_t = s] = \mathbb{E}_{\pi}[r_t + \gamma \mathcal{U}_{t+1} | s_t = s] =$$

$$= \sum_{a \in A} \pi(a|s) \sum_{s' \in S} p(s'|s,a) \left[\mathcal{R}(s,a,s') + \gamma \mathbb{E}_{\pi}[\mathcal{U}_{t+1} | s_{t+1} = s']\right] =$$

$$= \sum_{a \in A} \underbrace{\pi(a|s)}_{\substack{\text{1 sólo para} \\ a \sim \pi(\cdot\,|s)}} \underbrace{\sum_{s' \in S} p(s'|s,a) \left[\mathcal{R}(s,a,s') + \gamma V^{\pi}(s')\right]}_{Q^{\pi}(s,a)}$$

Valor de estado: juego de los dados

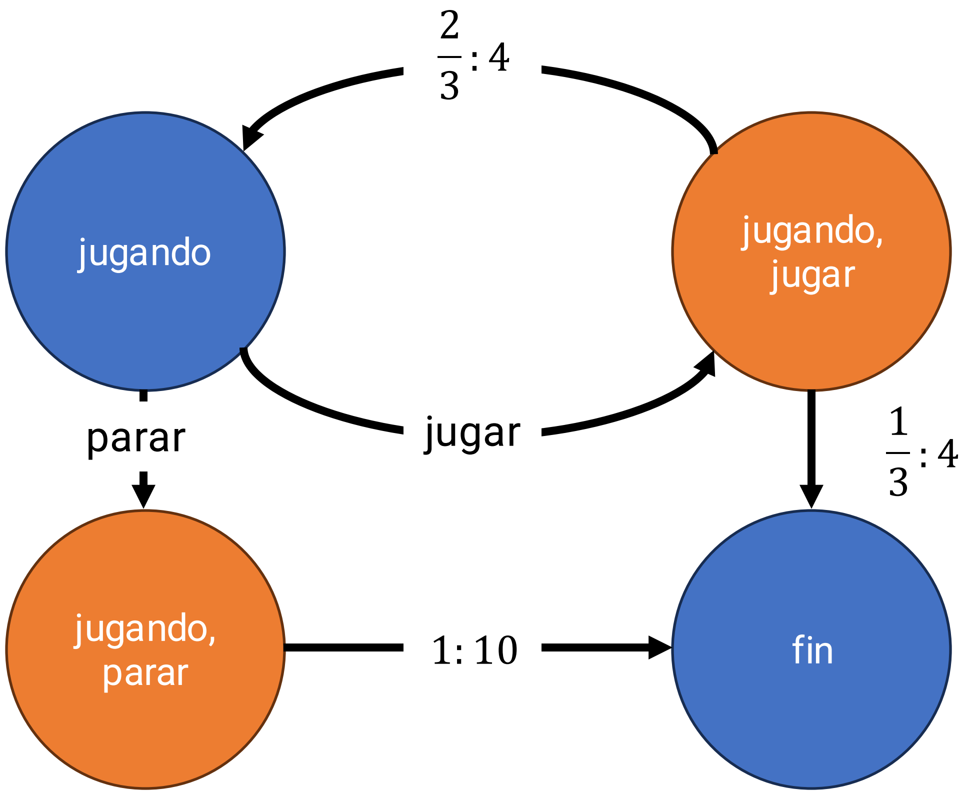
Escogemos π tal que $\pi(\text{jugando}) = \text{jugar}$

Asumimos $\gamma = 1$

$V^{\pi}(\text{fin}) = 0$

$V^{\pi}(\text{jugando}) = \frac{1}{3}\left(4 + V^{\pi}(\text{fin})\right) + \frac{2}{3}\left(4 + V^{\pi}(\text{jugando})\right)$

¿Podemos calcular $V^{\pi}(\text{jugando})$ en este caso?



Algoritmo: evaluación de política

- Programación dinámica
  - Inicializamos con valores arbitrarios e iteramos, actualizando los valores, hasta que converjan
- Pseudocódigo:
  - Inicializar $V_0^{\pi}(s) \leftarrow 0, \forall s \in S$
  - Para cada iteración $t = 1, \dots, t_{MAX}$
    - Para cada estado $s \in S$:

$$V_t^{\pi}(s) \leftarrow \underbrace{\sum_{s' \in S} p(s'|s,a)[\mathcal{R}(s,\pi(s),s') + \gamma V_{t-1}^{\pi}(s')]}_{Q_{t-1}^{\pi}(s,\pi(s))}$$

Algoritmo: evaluación de política

- Condición de convergencia, definimos ϵ:

$$\max_{s \in S} |V_t^\pi(s) - V_{t-1}^\pi(s)| \leq \epsilon$$

Si $\epsilon = 0$, la convergencia es total

- Únicamente es necesario guardar dos iteraciones: la actual y la anterior
- Complejidad: $O(t_{MAX}S^2)$

Iteración de valor

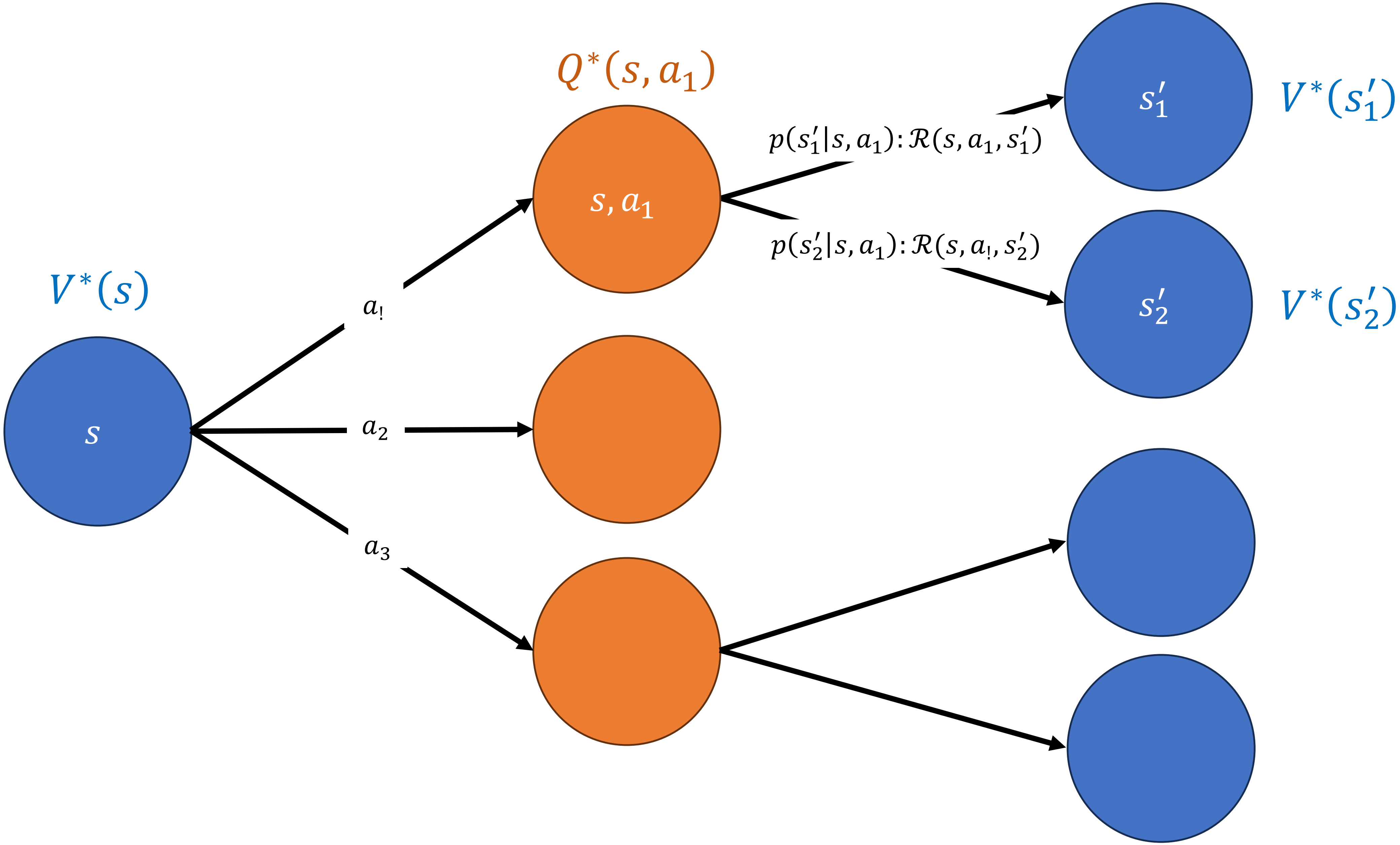
Agentes guiados por utilidad

Iteración de valor

- Si se dan las siguientes premisas:
  - Tenemos acceso a una señal de recompensa
  - El entorno es observable
  - Las acciones del agente sobre el entorno tienen un efecto aleatorio pero conocido: el entorno es determinista y estocástico
  - Hay garantía de convergencia
  - **NO conocemos la política del agente**
- En este caso podemos aplicar el método de **iteración de valor** para encontrar la **política óptima**

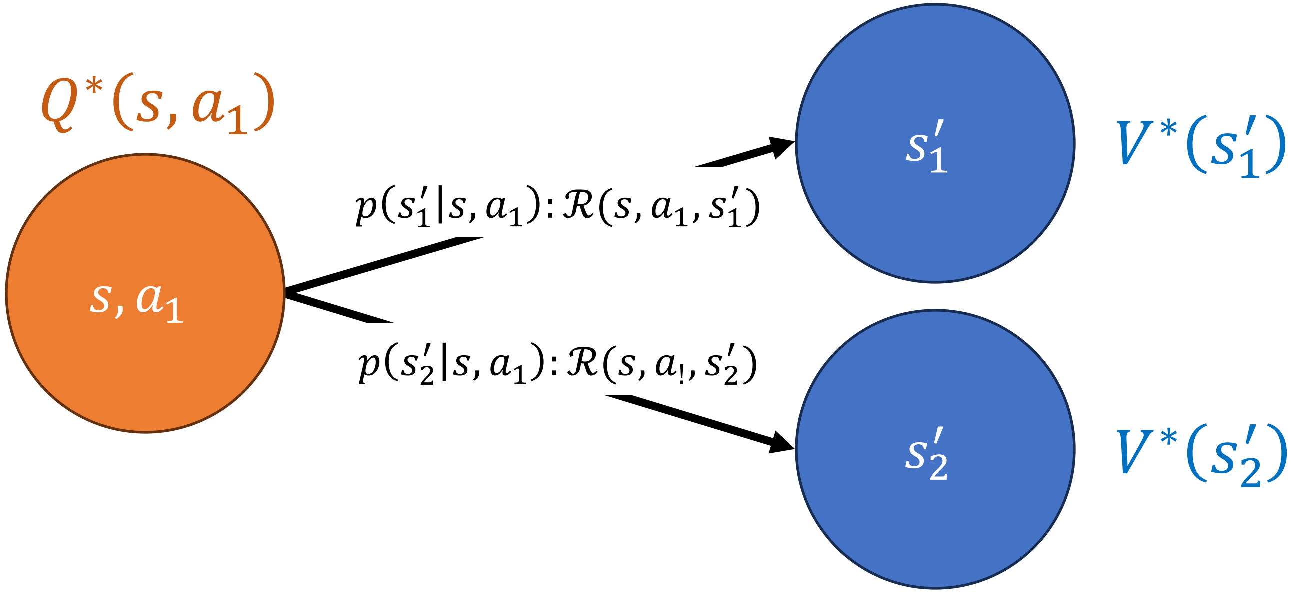
Valor óptimo

- No tenemos una política, así que queremos encontrar la óptima
- Llamaremos $V^*(s)$ a la función que retorna el valor máximo posible para un estado s
- Llamaremos $Q^*(s, a)$ a la función que retorna el valor máximo posible para una acción a en un estado s


$Q^*(s, a_1)$
s, a_1
$V^*(s'_1)$
s'_1
$p(s'_1|s, a_1)\colon \mathcal{R}(s, a_1, s'_1)$
$p(s'_2|s, a_1)\colon \mathcal{R}(s, a_!, s'_2)$
$V^*(s)$
$a_!$
s'_2
$V^*(s'_2)$
s
a_2
a_3

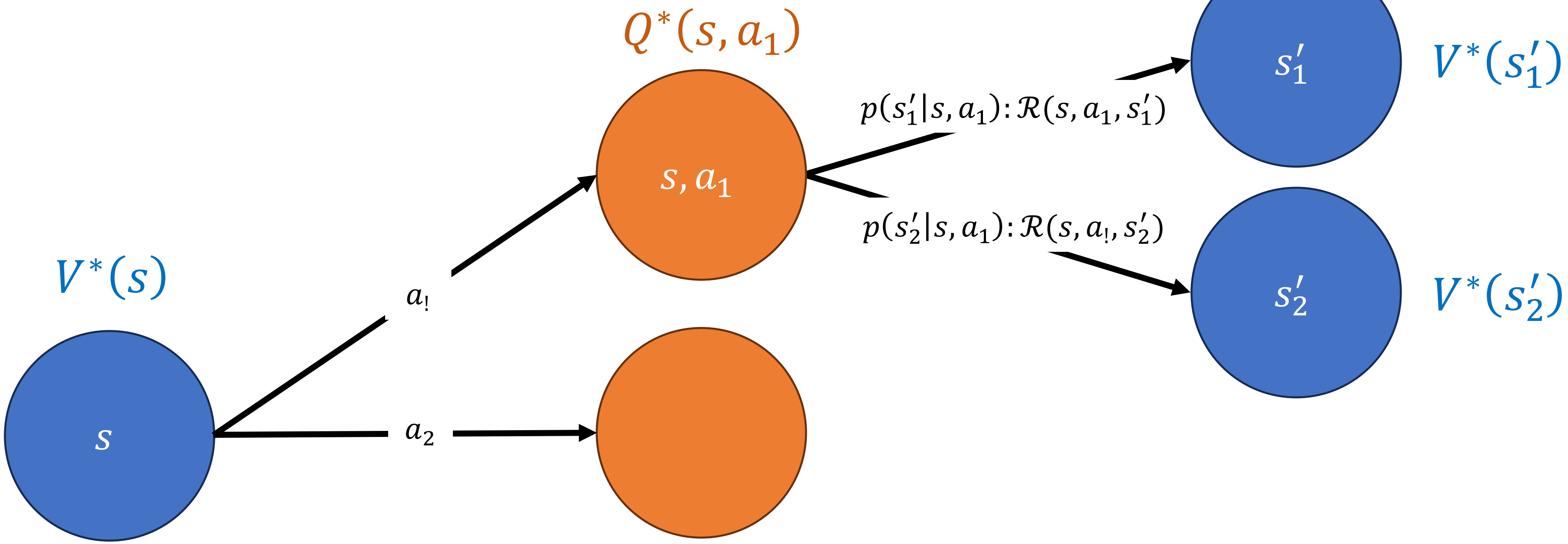
Valor acción-estado óptimo

$$Q^*(s,a) = \sum_{s' \in S} p(s'|s,a)[\mathcal{R}(s,a,s') + \gamma V^*(s')]$$



Valor estado óptimo

$$V^*(s) = \max_{a \in A} Q^*(s,a)$$



Política óptima

La política óptima se puede obtener simplemente escogiendo la acción $a \in A$ que maximiza la función $Q^*(s, a)$ para el estado s:

$$\pi^*(a|s) = \begin{cases} 1 & \text{si } a = \arg\max_{a' \in A} Q^*(s, a') \\ 0 & \text{en cualquier otro caso} \end{cases}$$

Esta fórmula es equivalente, por substitución, a:

$$\pi^*(a|s) = \begin{cases} 1 & \text{si } a = \arg\max_{a' \in A} \sum_{s' \in S} p(s'|s, a')[\mathcal{R}(s, a', s') + \gamma V^*(s')] \\ 0 & \text{en cualquier otro caso} \end{cases}$$

Algoritmo: iteración de valor

- De nuevo, programación dinámica
  - Inicializamos con valores arbitrarios e iteramos, actualizando los valores, hasta que converjan
- Pseudocódigo:
  - Inicializar $V_0^*(s) \leftarrow 0, \forall s \in S$
  - Para cada iteración $t = 1, \dots, t_{MAX}$
    - Para cada estado $s \in S$:

$$V_t^*(s) \leftarrow \max_{a \in A} \underbrace{\sum_{s' \in S} p(s'|s,a)[\mathcal{R}(s,a,s') + \gamma V_{t-1}^*(s')]}_{Q_{t-1}^*(s,a)}$$

Algoritmo: iteración de valor

- Convergencia:
  - Como en evaluación de política:
$$\max_{s \in S} \left| V_t^*(s) - V_{t-1}^*(s) \right| \leq \epsilon$$
  - Iteración de valor garantiza el óptimo si:
    - $\gamma < 1$, o bien
    - el grafo resultante del MDP es acíclico
- Complejidad: $O(t_{MAX} S^2 A)$

Recapitulando

- Si tenemos la tarea objetivo formulada como un MDP y además tenemos una política
  - **Evaluación de política** nos permite obtener las funciones de valor y por lo tanto **la utilidad esperada**
- Si tenemos la tarea objetivo formulada como un MDP
  - **Iteración de valor** nos permite obtener **la política óptima**
- ¿Y si no tenemos observabilidad total?
  - Partially Observable Markov Decision Processes (POMDPs)
- ¿Y si no tenemos un MDP?
  - **Aprendizaje por refuerzo** (siguiente tema)

POMDPs (breve introducción)

Agentes guiados por utilidad

POMDPs

- Queremos modelar el proceso de razonamiento que permite a un agente tomar decisiones en un entorno *parcialmente observable*: *Partially observable Markov decision process*
- Vamos a suponer que…
  - Tenemos acceso a una señal de recompensa
  - El entorno es *parcialmente* observable
  - El entorno puede ser *dinámico* y/o *no determinista*

observación s’ → **creencia b, acción a** → **observación s’**
creencia b, acción a → **observación s’’**

Formalización: POMDPs

Un POMDP se define como una tupla $\langle \boldsymbol{S}, \boldsymbol{A}, \boldsymbol{\mathcal{R}}, \boldsymbol{\mathcal{T}}, \boldsymbol{\mu}, \boldsymbol{\gamma}, \boldsymbol{\Omega}, \boldsymbol{\mathcal{O}}, \boldsymbol{\mathcal{B}} \rangle$ tal que:

$S, A, \mathcal{T}, \mu, \gamma$ son los mismos elementos que en un MDP

$\mathcal{R}: S \times A \to \mathbb{R}$ es la función de recompensa, definida aquí sólo sobre estado y acción

Ω es el conjunto (finito) de observaciones

$\mathcal{O}: A \times S \to \Delta\Omega$ es una función de observación, tal que $\mathcal{O}(o|a, s)$ denota la probabilidad de observar o cuando el agente toma la acción a y sucede una transición a s

$\mathcal{B}$ es una función de probabilidad sobre estados a partir de secuencias de acciones y observaciones: $\mathcal{B}(s_t) = Pr(s_t = s|s_0, a_1, o_1, a_2, o_2, \ldots, a_{t-1}, o_{t-1})$

Entonces, la política se define de esta manera:

$$\boldsymbol{\pi}: \boldsymbol{\Omega} \times \boldsymbol{\Omega} \times \cdots \times \boldsymbol{\Omega} \to \boldsymbol{A}$$

De observaciones a creencias

- La probabilidad de una observación se puede calcular a partir de $\mathcal{T}$ iterando por estados posibles

$$Pr(o|a,b) = \sum_{s'} Pr(o|a,s',b)\, Pr(s'|a,b)$$

$$= \sum_{s'} \mathcal{O}(o|a,s')\, Pr(s'|a,b) = \sum_{s'} \mathcal{O}(o|a,s') \sum_{s} \mathcal{B}(s) p(s'|s,a)$$

- A partir de $\mathcal{B}$, se puede calcular la creencia usando la regla de Bayes:

$$\mathcal{B}'(s_t) = Pr(s'|b,a,o) = \frac{\mathcal{O}(o|a,s') \sum_{s\in S} \mathcal{B}(s) p(s'|s,a)}{Pr(o|a,b)}$$

De POMDPs a MDPs

- Un POMDP se puede reducir a un MDP con la función de transición $\mathcal{T}'$:

$$\mathcal{T}'(b, a, b') = \sum_{o \in \Omega} \Pr(b'|o, a, b) \Pr(o|a, b)$$

$$= \sum_{o \in \Omega} \Pr(b'|o, a, b) \sum_{s \in S} \mathcal{O}(o|a, s') \sum_{s \in S} \mathcal{B}(s) p(s'|s, a)$$

y la función de recompensa $\mathcal{R}'$:

$$\mathcal{R}(b, a) = \sum_{s \in S} \mathcal{B}(s) \mathcal{R}(s, a)$$