

Las bases del aprendizaje por refuerzo

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Metas del aprendizaje por refuerzo

- Conseguir que un agente aprenda a tomar decisiones **secuenciales** de manera **óptima**
- Ejemplo: Queremos que el agente aprenda a jugar ajedrez para jugar contra un ser humano. Nota: En el ajedrez las jugadas se dan una a una

Toma decisiones para mover
las piezas de color negro

Agente



Persona

Metas del aprendizaje por refuerzo

- Es **secuencial** porque...

Las jugadas se da una a una



- Es **óptimo** porque...

Jugada tras jugada, el agente debe saber cuál es la pieza más adecuada para mover

Metas del aprendizaje por refuerzo

- Se debe tener en cuenta 2 situaciones, donde el agente:

a) Sacrifica una ficha propia/No toma una ficha contraria

	Corto plazo	Largo plazo
Parece	👎	
En realidad es		👍

Parece ser una mala decisión, pero no lo es
OJO: No siempre se cumple, es una posibilidad

Metas del aprendizaje por refuerzo

- Se debe tener en cuenta 2 situaciones, donde el agente:
 - a) No sacrifica una ficha propia/Toma una ficha contraria
 - b) No sacrifica una ficha propia/Toma una ficha contraria

	Corto plazo	Largo plazo
Parece		
En realidad es		

Parece ser una buena decisión, pero no lo es
OJO: No siempre se cumple, es una posibilidad

Metas del aprendizaje por refuerzo

El agente debe tener un balance, para “idear” la mejor estrategia y ganar el juego

Recompensas a
largo plazo



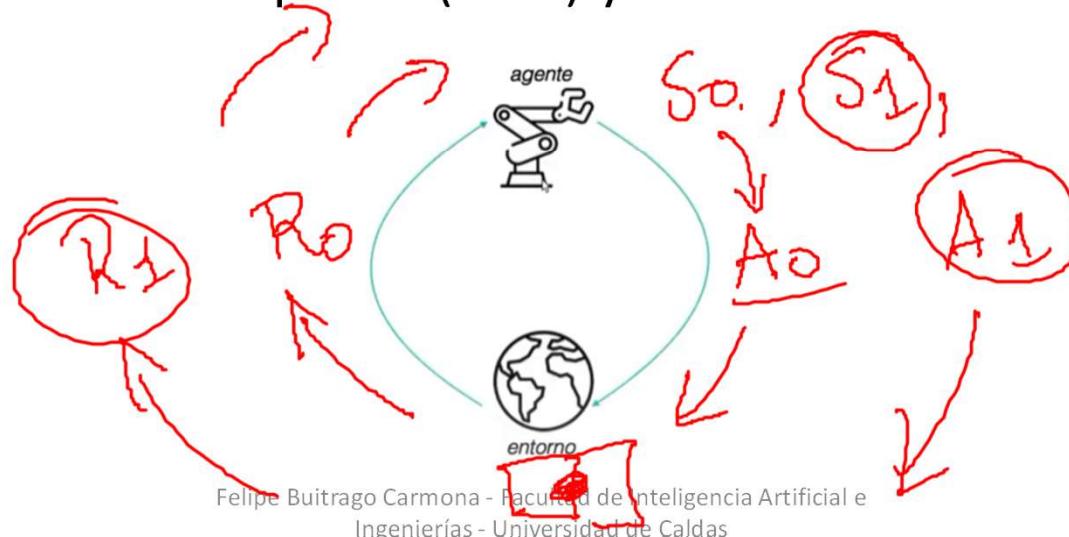
Recompensas a
corto plazo

Recompensas y estados

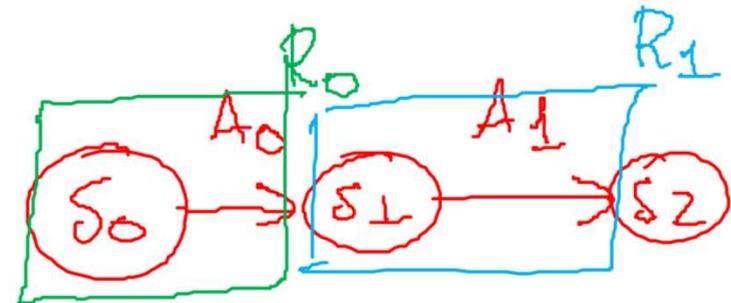
Markov Decision Process



- El agente se retroalimenta del entorno
- En cada iteración, el **entorno** le devuelve una recompensa **positiva** o **negativa**, dependiendo de la acción ejecutada por el **agente**
- El agente recibe: recompensa (+ o -) y un nuevo estado por cada iteración



Recompensas y estados



Se presentan las siguientes variables:

- Instantes de tiempo: $t = 0, 1, 2, 3, \dots$
- Estado en un tiempo dado: S_t ✓
- Acción tomada por el agente en un tiempo dado: A_t
- Recompensa obtenida por el agente: R_t

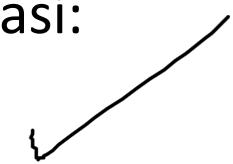
Recompensas y estados

- **Espacio de estados**

El conjunto de todos los posibles estados se representa así:

$$S = \{S_0, S_1, S_2, S_3, \dots, S_m\}$$

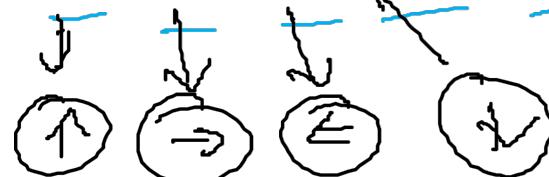
S_0	S_1	S_2	S_3
			S_n



- **Espacio de acciones**

El conjunto de todas las posibles acciones se representa así:

$$A = \{A_0, A_1, A_2, A_3, \dots, A_n\} \rightarrow$$



Recompensas y estados

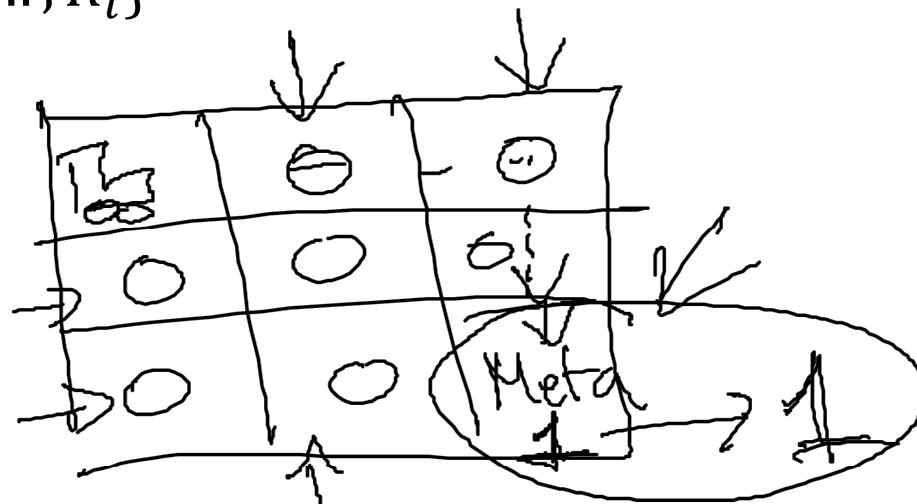
$$\underline{MDP} = \{S, A, R, P\}$$

- **Función de recompensa o recompensa**

El conjunto de todas las posibles recompensas se representa así:

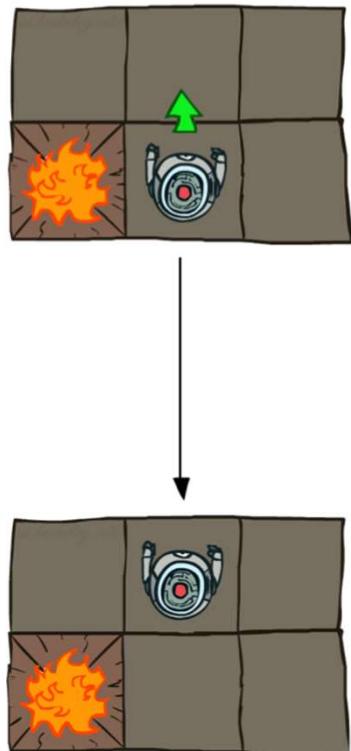
$$R = \{R_0, R_1, R_2, R_3, \dots, R_l\}$$

$$\left\{ \begin{array}{c} \downarrow \\ \emptyset, 1 \end{array} \right.$$

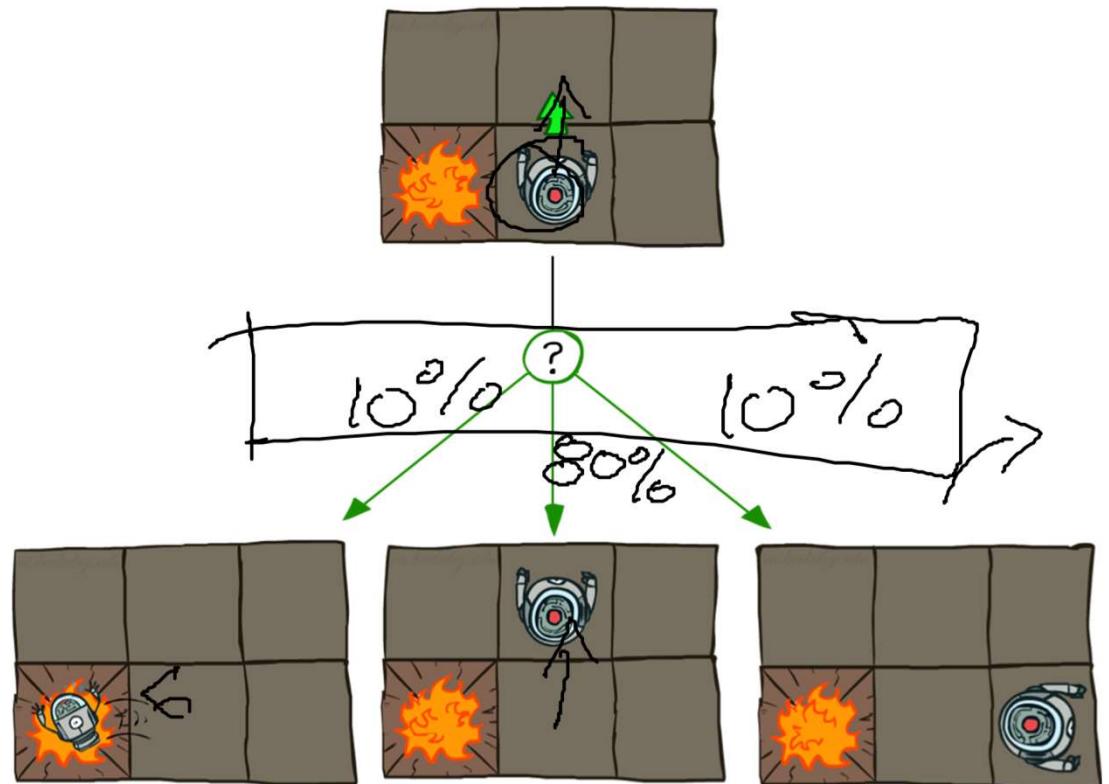


Grid World Actions

Deterministic Grid World



Stochastic Grid World

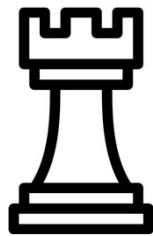


Componente aleatorio

- Se debe tener en cuenta que hay un componente de aleatoriedad y no es determinístico, es decir, no hay un 100% de certeza sobre lo que va a ocurrir
- Ejemplo: Agente ataca al rey del oponente, entonces, viéndolo desde la probabilidad:
 - A) El oponente protege al rey con la torre ($70\% \rightarrow 0.7$)
 - B) El oponente protege al rey con el caballo ($25\% \rightarrow 0.25$)
 - C) El oponente protege al rey con el peón ($5\% \rightarrow 0.05$)

Componente aleatorio

- Lo más **PROBABLE** es que según el contexto, el oponente mueva la torre, pero **PUEDE** ser que mueva el caballo o el peón



- Dicha incertidumbre de tener varias posibilidades de movida, hace que haya un **componente aleatorio**

Componente aleatorio

MDPF S, A, R, P

- Cuando ejecutamos una acción, hay un cambio de estado con un grado de aleatoriedad, la función que permite calcular cómo se pasa de un estado a otro teniendo en cuenta esa aleatoriedad, se conoce como **función de transición**:
- Función de transición

$$p(S'|S, a) = P(S_t = S' | S_{t-1} = S, A_{t-1} = a)$$

Función de transición

$$p(S'|S, a)$$

$$P(S_t = S' | S_{t-1} = S, A_{t-1} = a)$$

Donde:

→ $p(S'|S, a)$ Función de transición ✓

→ S' Estado futuro, estado de la siguiente jugada

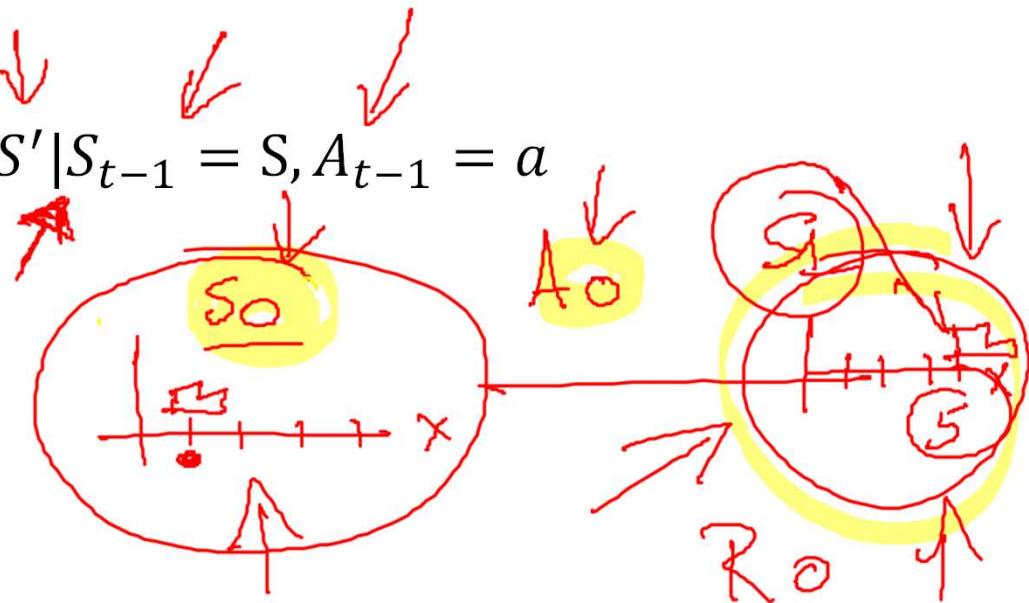
S : Estado actual —

a : Acción ejecutada actual —

S_t : Estado en un tiempo dado t

S_{t-1} : Estado anterior en un tiempo dato t

A_{t-1} : Acción anterior a la ejecutada en un tiempo dato t



Función de transición

La traducción de la función anterior sería:



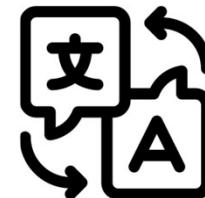
$$p(S'|S, a) = P(S_t = S'|S_{t-1} = S, A_{t-1} = a)$$

IZQUIERDA: Cuál será probabilidad del estado futuro (en el siguiente movimiento), **dados** un estado actual y la acción ejecutada actual.

Ejemplo: Cuál sería la probabilidad de que el oponente mueva la torre, dado el estado actual del juego y que el agente está atacando al rey

Función de transición

La traducción de la función anterior sería:



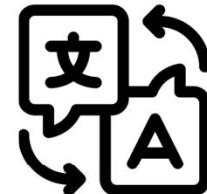
$$p(S'|S, a) = P(S_t = S'|S_{t-1} = S, A_{t-1} = a)$$

DERECHA: La probabilidad (se mide entre 0 → suceso no ocurre y 1 → certeza total de que ocurre) de que en el instante de tiempo t, nuestra configuración del juego, termine estando en el estado futuro, teniendo en cuenta que en el estado de tiempo anterior, se tenía un estado S y la acción anterior, se tenía una acción a.

Función de transición

$$MDP = \{S, A, R, p\}$$

La traducción de la función anterior sería:



$$\rightarrow p(S'|S, a) = P(S_t = S' | S_{t-1} = S, A_{t-1} = a)$$

En resumen...

La función de transición describe la probabilidad de que el agente pase del estado actual S al estado futuro S' como consecuencia de tomar la acción a .

Proceso de decisión de Márkov Tablero unidimensional determinístico

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Proceso de decisión de Márkov

Tablero unidimensional determinístico

- Es un conjunto - notación matemática que contiene todos los **componentes** del problema de aprendizaje por refuerzo:
 - a) El espacio de estados (S)
 - b) El espacio de acciones (A)
 - c) La recompensa (R)
 - d) La función de transición (p)

Proceso de decisión de Márkov

Tablero unidimensional determinístico

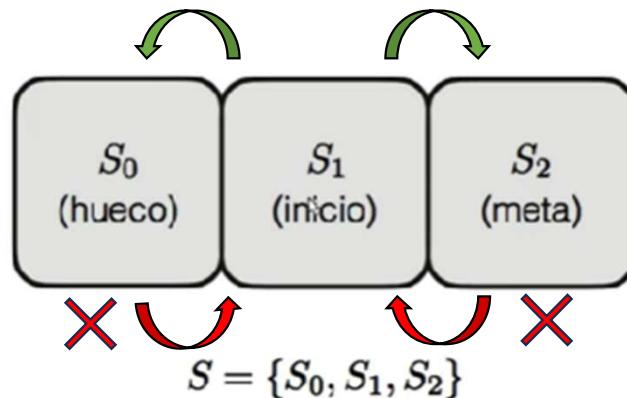
- Agente que se desplaza a través de un tablero unidimensional
- Tablero unidimensional: 1 única fila
- Limita la movilidad del agente
- Agente se mueve en una sola dirección, hacia la izquierda o derecha



Proceso de decisión de Márkov Tablero unidimensional determinístico

Espacio de estados → Define las reglas del juego

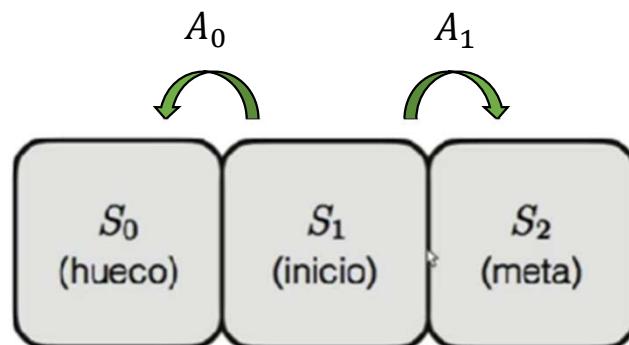
- Posee 3 casillas para 3 diferentes estados
- El agente sólo podrá moverse si está en el estado inicial, si está en los extremos NO



Proceso de decisión de Márkov Tablero unidimensional determinístico

Espacio de acciones → Movimientos que puede ejecutar el agente

- A_0 : Movimiento hacia la izquierda
- A_1 : Movimiento hacia la derecha



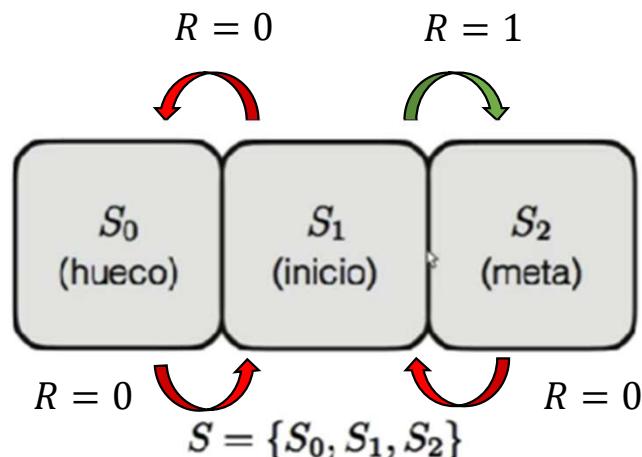
$$S = \{S_0, S_1, S_2\}$$

Felipe Buitrago Campona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Proceso de decisión de Márkov Tablero unidimensional determinístico

Espacio de recompensas → Movimiento que lo lleve a la meta

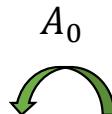
- $R = 1$: S_1 a S_2
- $R = 0$: S_1 a S_0 , S_0 a S_1 & S_2 a S_1



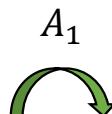
Proceso de decisión de Márkov Tablero unidimensional determinístico

Función de transición → Problema determinístico → No hay componente de aleatoriedad

- Si el agente ejecuta la acción A_0 , con una certeza del 100% se va a mover hacia la izquierda → **No hay componente de aleatoriedad**



- Si el agente ejecuta la acción A_1 , con una certeza del 100% se va a mover hacia la derecha → **No hay componente de aleatoriedad**



Proceso de decisión de Márkov

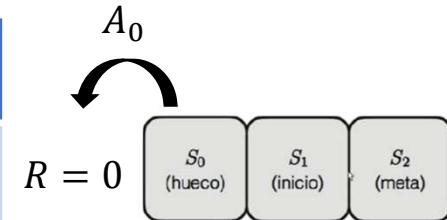
Tablero unidimensional determinístico

Función de transición

- ① Estado actual
 - ② Acción actual
 - ③ Estado siguiente
 - ④ Función de transición
 - ⑤ Recompensa

Proceso de decisión de Márkov Tablero unidimensional determinístico

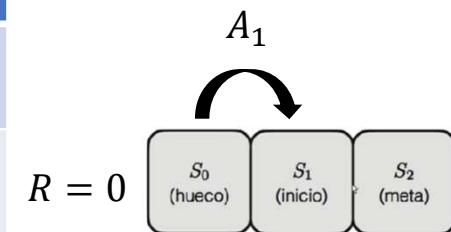
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0



Proceso de decisión de Márkov

Tablero unidimensional determinístico

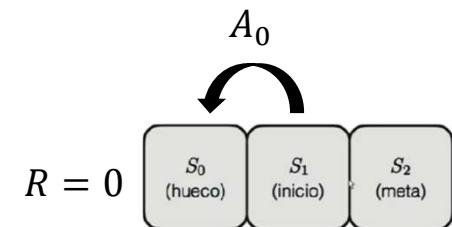
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0



Proceso de decisión de Márkov

Tablero unidimensional determinístico

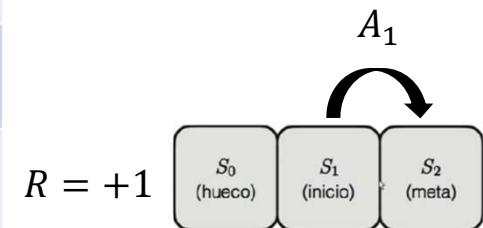
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	1	0



Proceso de decisión de Márkov

Tablero unidimensional determinístico

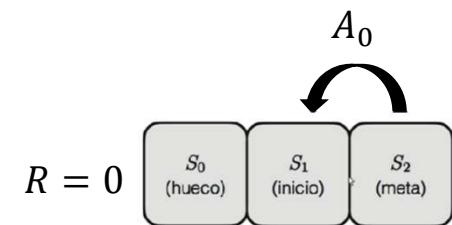
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	1	0
S_1	A_1	S_2	1	+1



Proceso de decisión de Márkov

Tablero unidimensional determinístico

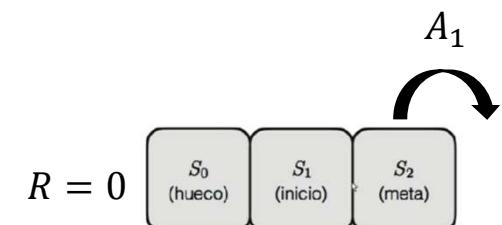
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	1	0
S_1	A_1	S_2	1	+1
S_2	A_0	S_2	1	0



Proceso de decisión de Márkov

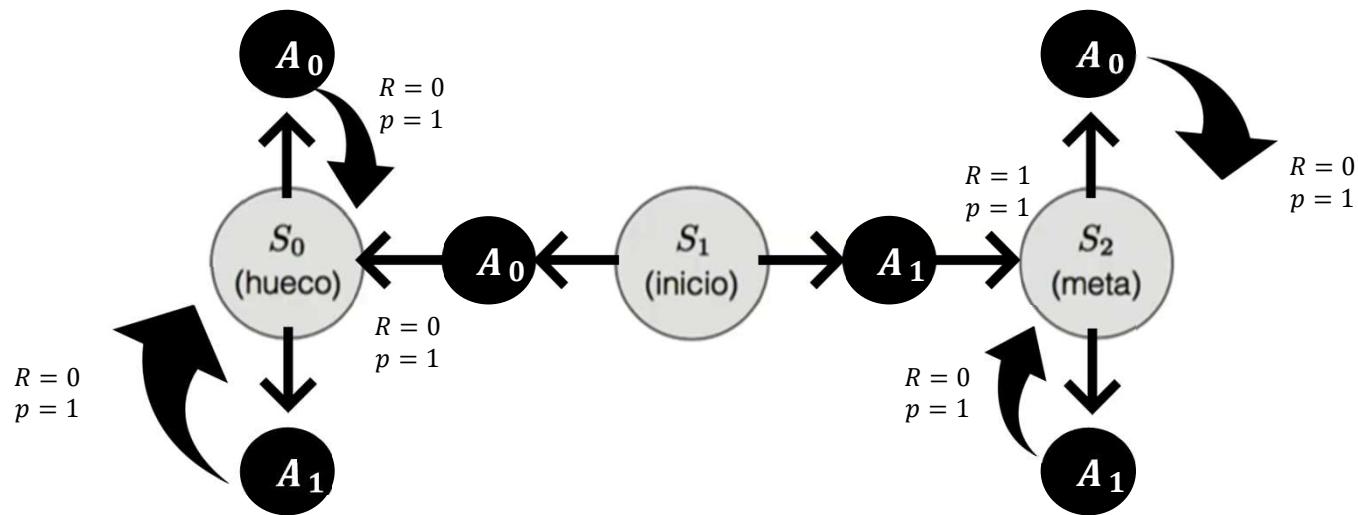
Tablero unidimensional determinístico

$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	1	0
S_1	A_1	S_2	1	+1
S_2	A_0	S_2	1	0
S_2	A_1	S_2	1	0



Proceso de decisión de Márkov

Tablero unidimensional determinístico-grafo



Sólo hay un cambio de estado desde que el agente esté en el inicio.

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

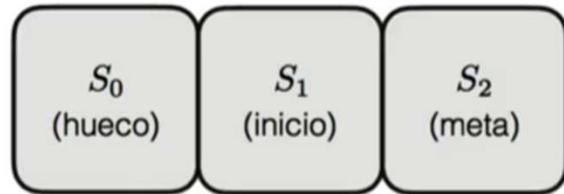
Proceso de decisión de Márkov Tablero unidimensional estocástico

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Proceso de decisión de Márkov

Tablero unidimensional estocástico

- Para este caso se preservan las reglas:
 - Hay 3 estados: hueco, inicio, meta
 - Hay 2 acciones: A_0 hacia la izquierda, A_1 hacia la derecha
 - Hay 2 recompensas: 1 para el pase del inicio a meta, 0 para cualquier otro caso



$$S = \{S_0, S_1, S_2\}$$

$$A = \{A_0, A_1\}$$

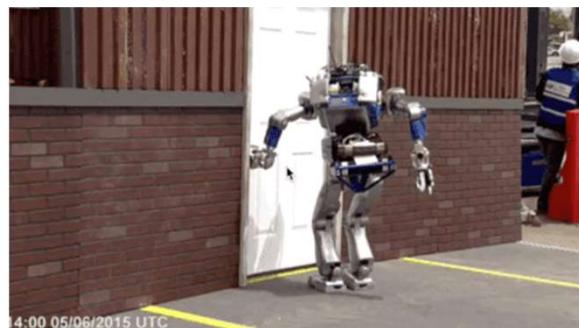
$$R = \{+1, 0\}$$

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Proceso de decisión de Márkov

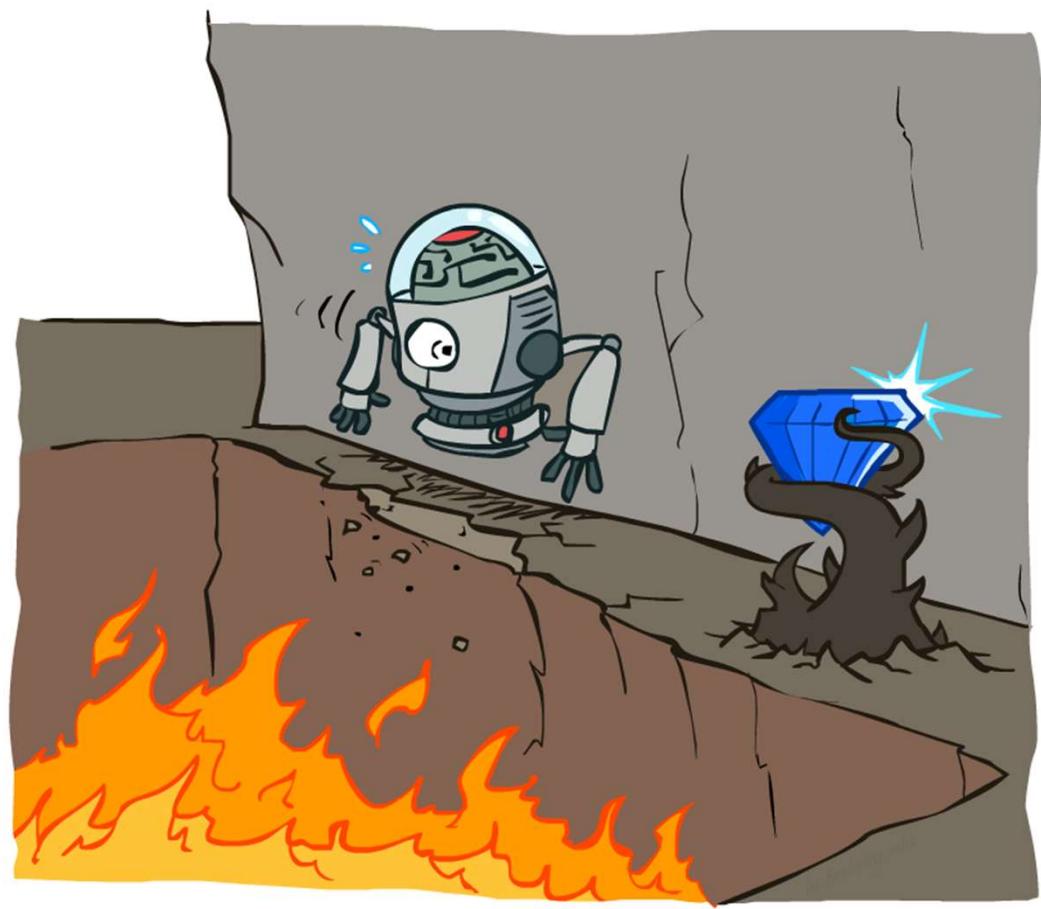
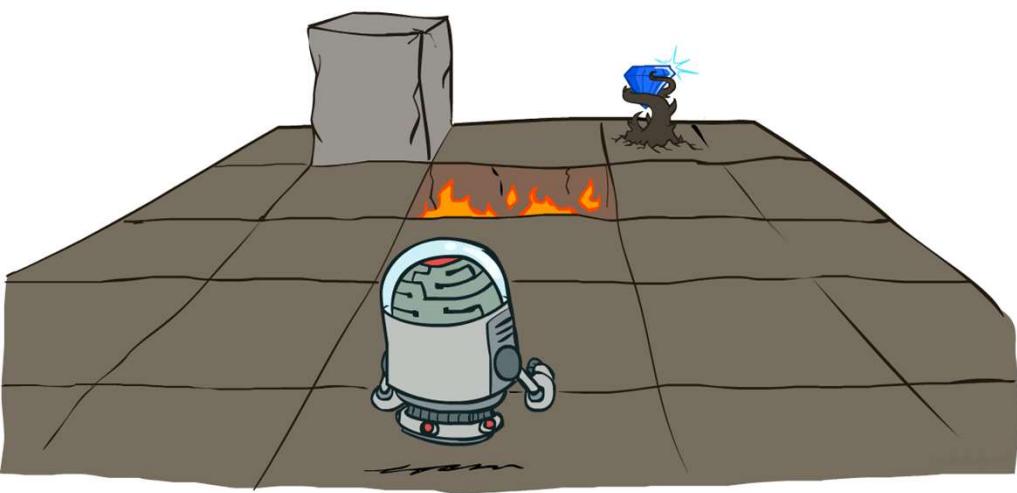
Tablero unidimensional estocástico

- En este caso, se añade un componente aleatorio
- En las aplicaciones reales, siempre habrá una **incertidumbre**
- Ejemplo: En robótica la vibración impide que un robot agarre con su pinza un objeto, o el objeto está húmedo. Sin embargo, la mayoría de las veces lo logrará (no siempre).

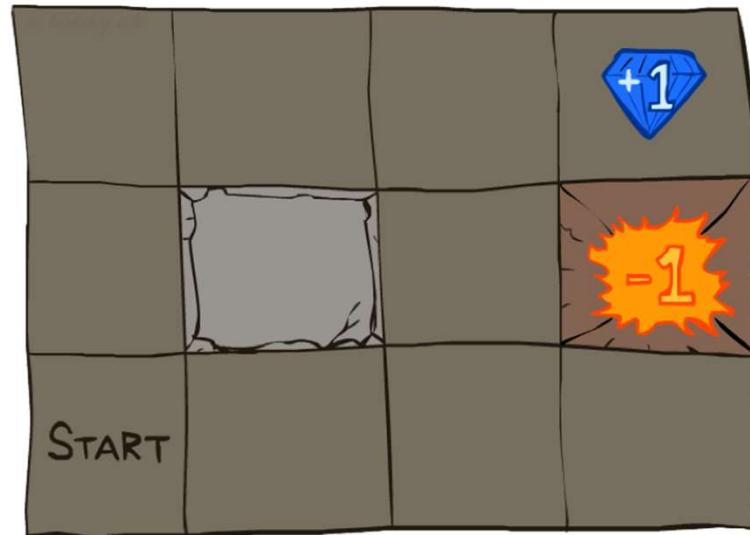


Componentes estocásticos:

- Vibración del ambiente
- Objeto húmedo
- Robot con pinzas desajustadas



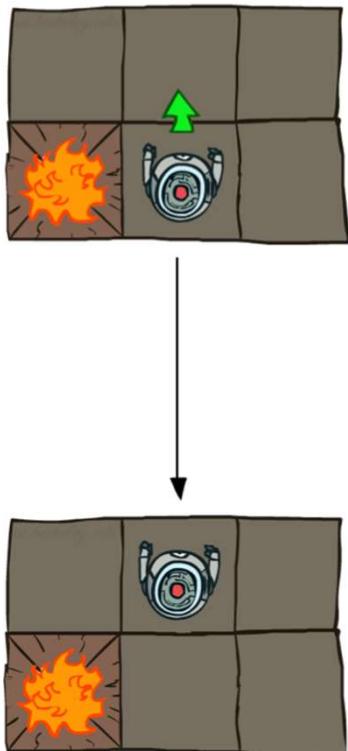
Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas



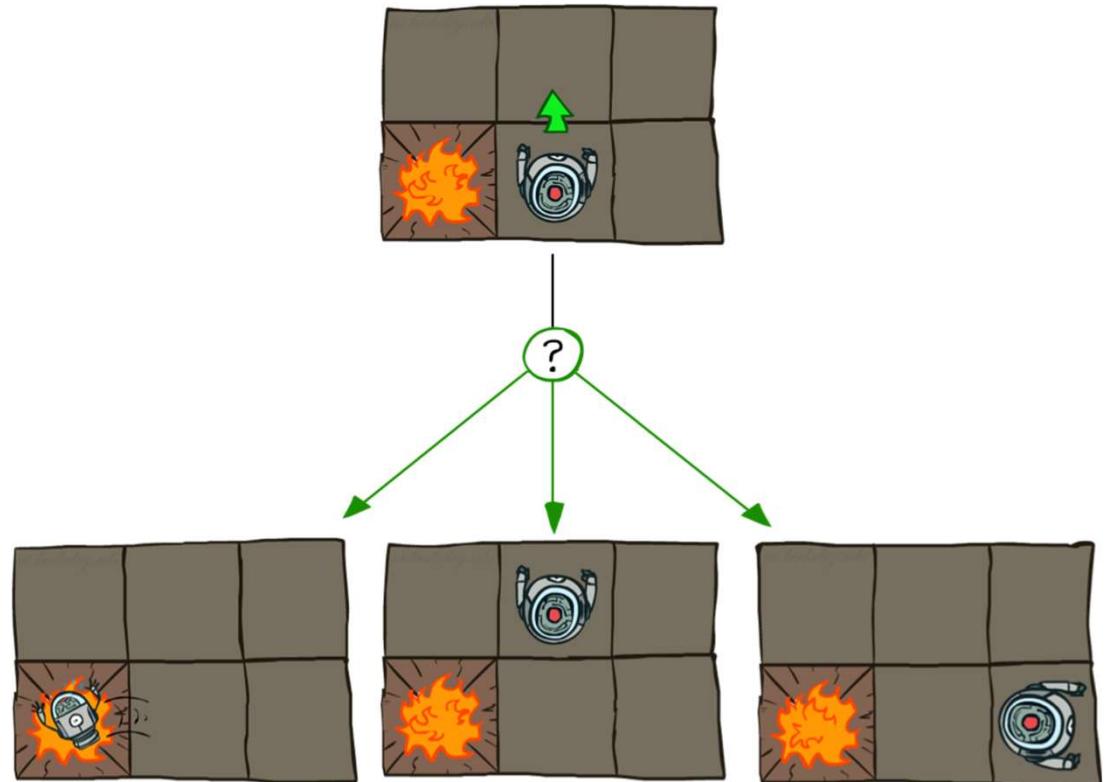
Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Grid World Actions

Deterministic Grid World

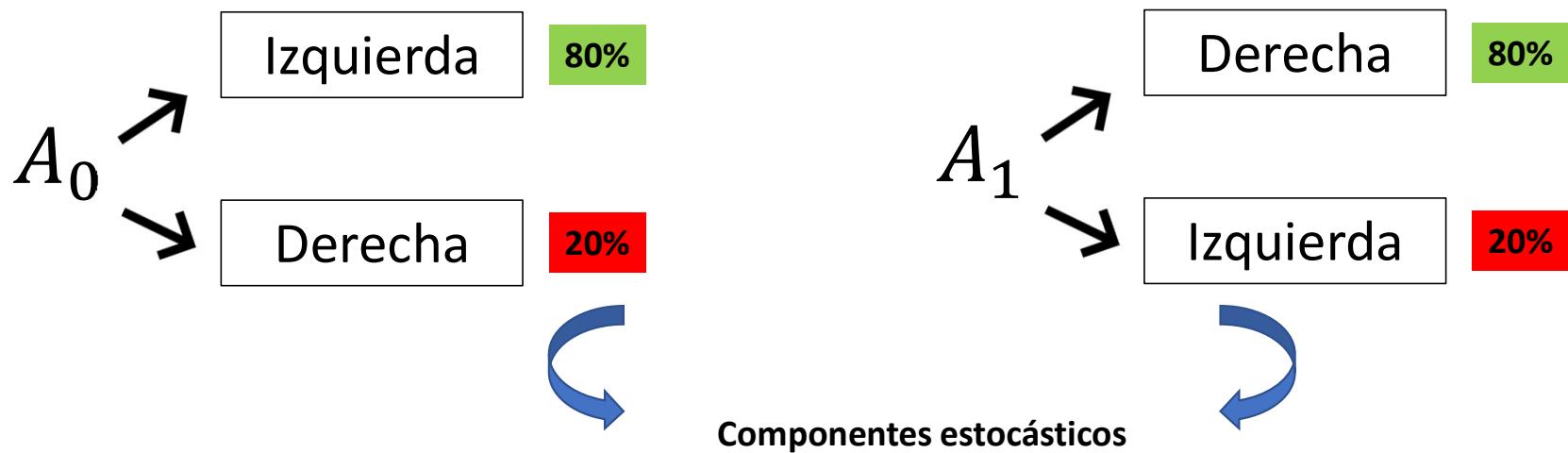


Stochastic Grid World



Proceso de decisión de Márkov

Tablero unidimensional estocástico



Proceso de decisión de Márkov

Tablero unidimensional estocástico

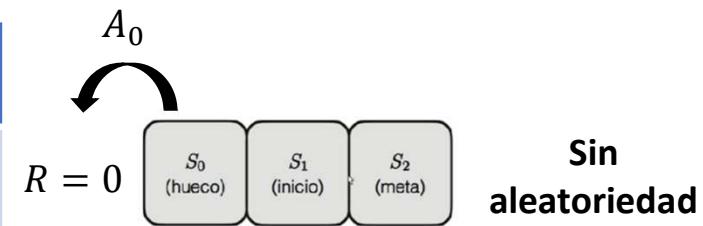
Función de transición

- ① Estado actual
 - ② Acción actual
 - ③ Estado siguiente
 - ④ Función de transición
 - ⑤ Recompensa

Proceso de decisión de Márkov

Tablero unidimensional estocástico

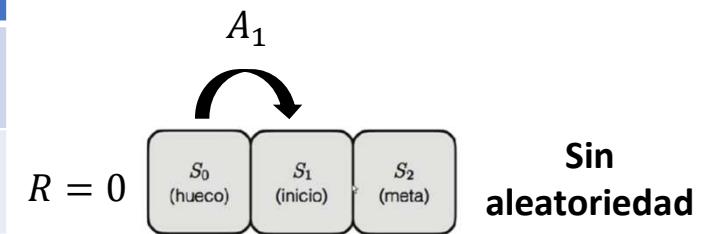
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0



Proceso de decisión de Márkov

Tablero unidimensional estocástico

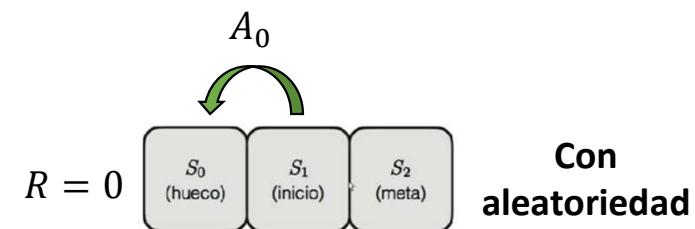
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0



Proceso de decisión de Márkov

Tablero unidimensional estocástico

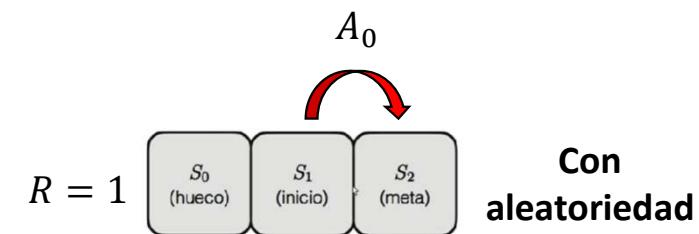
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	0.8	0



Proceso de decisión de Márkov

Tablero unidimensional estocástico

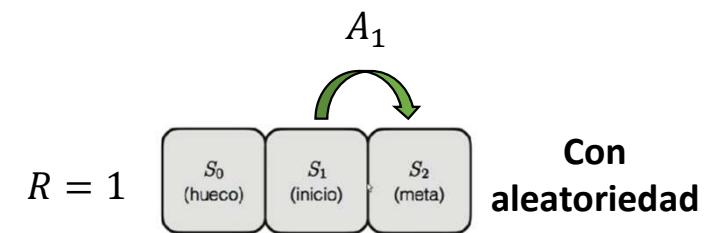
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	0.8	0
S_1	A_0	S_2	0.2	1



Proceso de decisión de Márkov

Tablero unidimensional estocástico

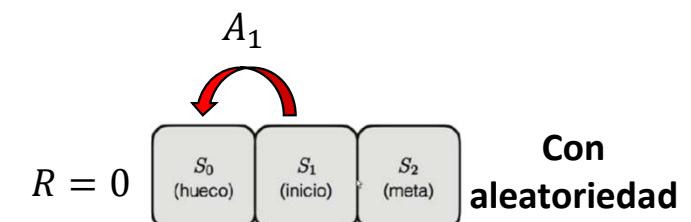
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	0.8	0
S_1	A_0	S_2	0.2	1
S_1	A_1	S_2	0.8	1



Proceso de decisión de Márkov

Tablero unidimensional estocástico

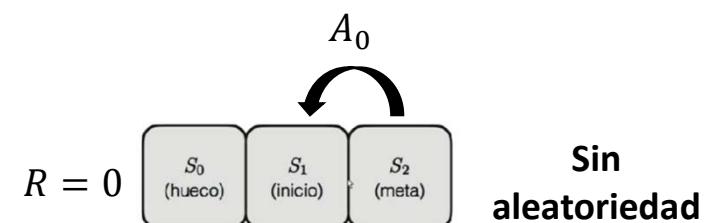
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	0.8	0
S_1	A_0	S_2	0.2	1
S_1	A_1	S_2	0.8	1
S_1	A_1	S_0	0.2	0



Proceso de decisión de Márkov

Tablero unidimensional estocástico

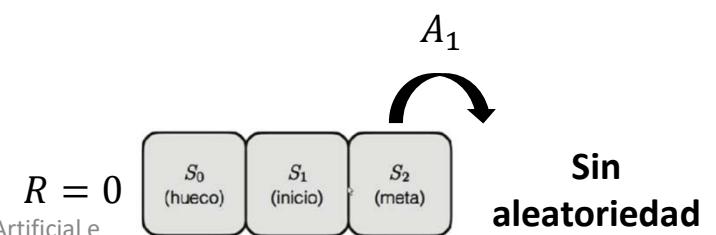
$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	0.8	0
S_1	A_0	S_2	0.2	0
S_1	A_1	S_2	0.8	1
S_1	A_1	S_0	0.2	0
S_2	A_0	S_2	1	0



Proceso de decisión de Márkov

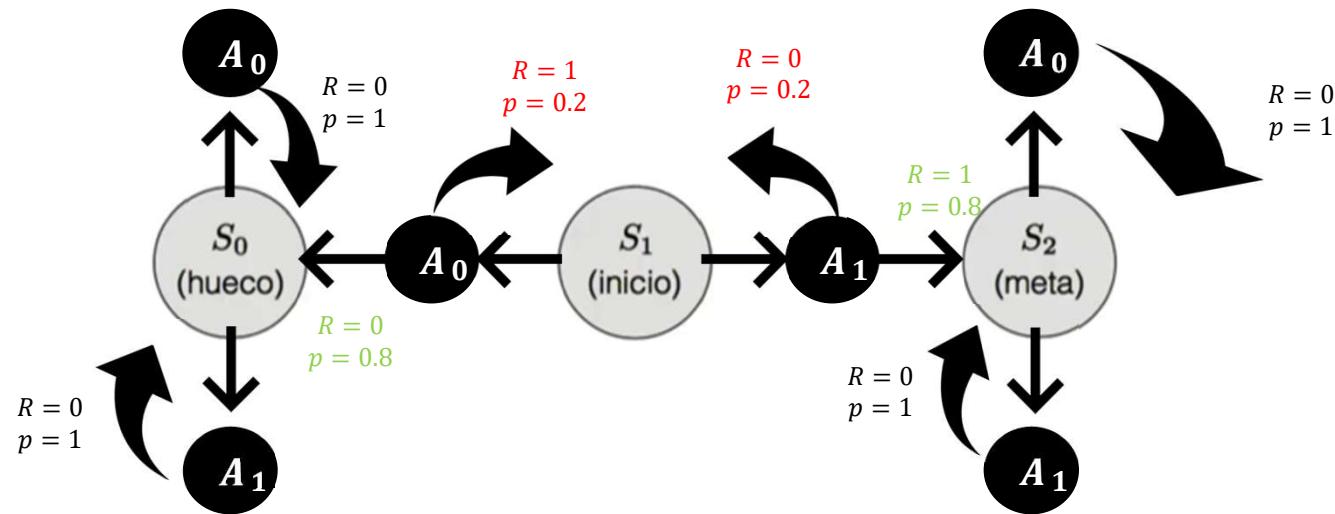
Tablero unidimensional estocástico

$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	A_0	S_0	1	0
S_0	A_1	S_0	1	0
S_1	A_0	S_0	0.8	0
S_1	A_0	S_2	0.2	1
S_1	A_1	S_2	0.8	1
S_1	A_1	S_0	0.2	0
S_2	A_0	S_2	1	0
S_2	A_1	S_2	1	0



Proceso de decisión de Márkov

Tablero unidimensional estocástico-grafo



Sólo hay un cambio de estado desde que el agente esté en el inicio.

Tablero bidimensional estocástico

Entorno y reglas del juego

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

Entorno y reglas del juego

- Tablero cuadrado 4x4
- Casillas enumeradas del 0 al 15
- El agente siempre estará ubicado desde el inicio
- Aparece el concepto **episodio**: serie de decisiones tomadas por el agente desde el inicio hasta la meta.



Tablero bidimensional estocástico

Entorno y reglas del juego

- Puede haber 2,3,4,5... episodios, en caso de que el agente no llegue a la meta
- Si el agente llega a la meta, tiene una recompensa de 1
- Si el agente llega a otro punto que no sea la meta, tiene una recompensa de 0



Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

Entorno y reglas del juego

- Casillas rojas son huecos: 5, 7, 11, 12 son estados terminales. El agente pierde el juego
- Hay componente estocástico según la opción que emprenda el agente.

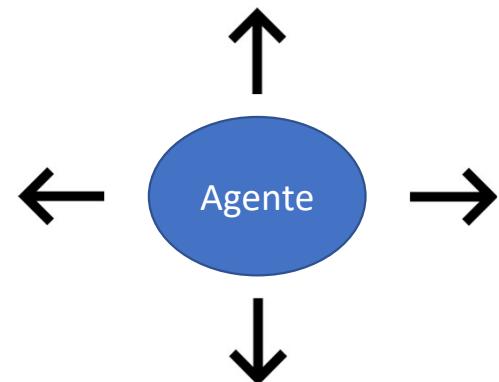


Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

Entorno y reglas del juego

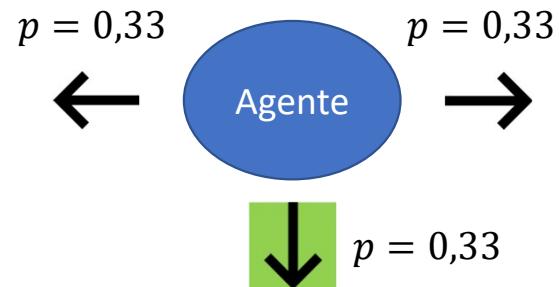
Movimientos del agente:



- La dirección escogida por el agente, tiene una probabilidad $p=0,33$
- El 0,66 restante, se distribuye equitativamente en las direcciones ortogonales del movimiento

Tablero bidimensional estocástico Entorno y reglas del juego

a) Hacia abajo



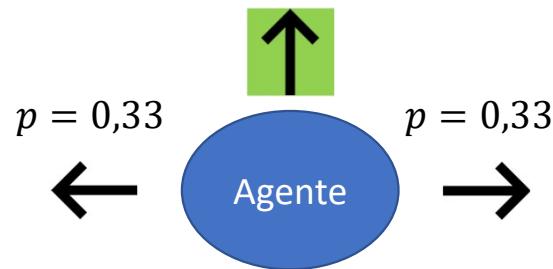
- Derecha e izquierda: ortogonales
- Derecha e izquierda: componente estocástico

Tablero bidimensional estocástico

Entorno y reglas del juego

a) Hacia arriba

$$p = 0,33$$

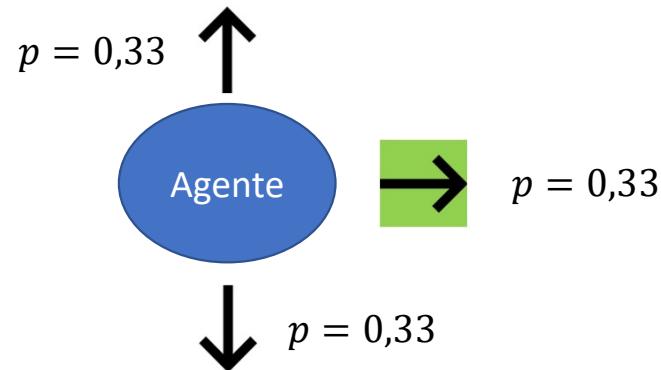


- Derecha e izquierda: ortogonales
- Derecha e izquierda: componente estocástico

Tablero bidimensional estocástico

Entorno y reglas del juego

a) Hacia la derecha

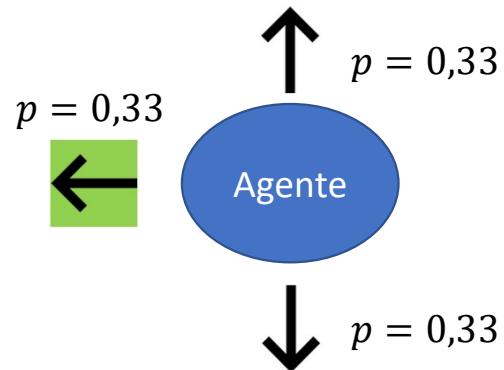


- Derecha e izquierda: ortogonales
- Derecha e izquierda: componente estocástico

Tablero bidimensional estocástico

Entorno y reglas del juego

a) Hacia la izquierda



- Derecha e izquierda: ortogonales
- Derecha e izquierda: componente estocástico

Tablero bidimensional estocástico Estados y la propiedad de Márkov

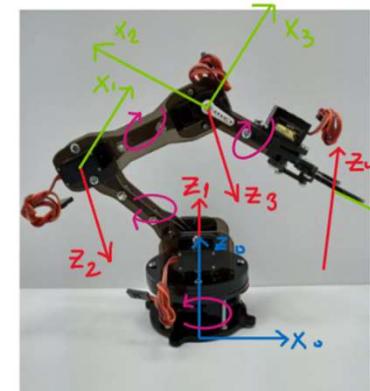
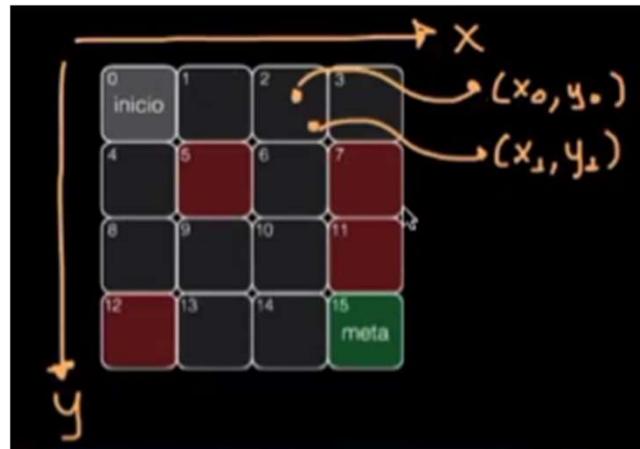
- Cada casilla es un estado: $S = \{S_0, S_1, S_2, S_3, \dots, S_{15}\}$
- Espacio de estados **discreto**, cada estado está representado por un número entero: 0, 1, 2...
- Espacio de estados **finito**: 16



Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico Estados y la propiedad de Márkov

- ¿Cómo podría ser continuo e infinito el espacio de estados?



- En un sistema de coordenadas (x_0, y_0) y (x_1, y_1) están en la misma casilla pero distanciadas levemente

Tablero bidimensional estocástico

Propiedad de Markov: Intuición

- Si el agente está en la casilla 2, se puede mover hacia la 1, 3 o 6. Por tanto no resulta relevante para saber el estado futuro (1, 3 o 6) conocer donde estaba antes de llegar a la casilla 2
- Lo único que resulta relevante es saber el **estado actual** y las **posibles acciones**. No es relevante saber el historial de estados o acciones pasadas



Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

Propiedad de Márkov: Notación matemática

$$P(S_{t+1}|S_t, A_t) = P(S_{t+1}|S_t, A_t, S_{t-1}, A_{t-1}, S_{t-2}, A_{t-2}, \dots)$$

$$P(S_{t+1}|S_t, A_t) = P(S_{t+1}|S_t, A_t, S_{t-1}, A_{t-1}, S_{t-2}, A_{t-2}, \dots)$$

Lado izquierdo: Cuál es la probabilidad de alcanzar el estado futuro, partiendo del estado y la acción presente.

Tablero bidimensional estocástico

Propiedad de Márkov: Notación matemática

$$P(S_{t+1}|S_t, A_t) = P(S_{t+1}|S_t, A_t, S_{t-1}, A_{t-1}, S_{t-2}, A_{t-2}, \dots)$$

$$P(S_{t+1}|S_t, A_t) = P(S_{t+1}|S_t, A_t, S_{t-1}, A_{t-1}, S_{t-2}, A_{t-2}, \dots)$$

Lado derecho: Probabilidad de alcanzar el estado futuro, teniendo en cuenta todo el historial de los pares estado-acción, en todos los instantes de tiempo anteriores.

Tablero bidimensional estocástico

Propiedad de Márkov: Notación matemática

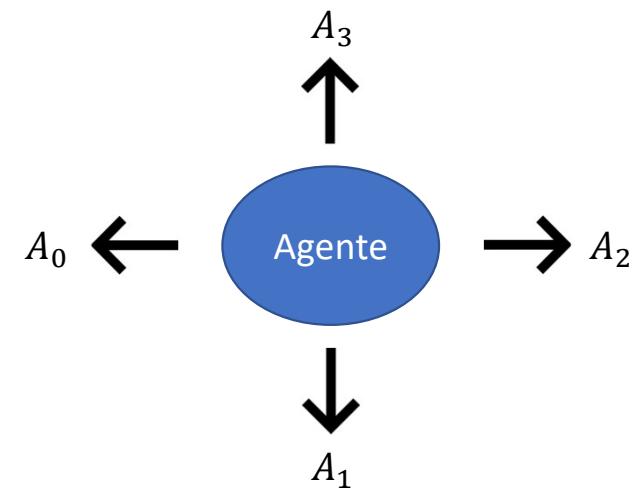
$$P(S_{t+1}|S_t, A_t) = P(S_{t+1}|S_t, A_t, S_{t-1}, A_{t-1}, S_{t-2}, A_{t-2}, \dots)$$

¿Qué nos dice la propiedad, entonces...?

- Como lo indica el lado izquierdo de la notación matemática, no será necesario conocer el historial (lado derecho de la notación matemática), **pues sólo se está teniendo en cuenta el estado y acción presente, para mirar la posibilidad de un estado futuro**

Tablero bidimensional estocástico

Las acciones en nuestro juego



- **Espacio de acciones:** $A = \{A_0, A_1, A_2, A_3\}$
- **Finito y discreto (no hay puntos intermedios de movimiento)**

Tablero bidimensional estocástico

Otro tipos de acciones

- La cabrilla de un carro autónomo puede girar con un ángulo Θ cualquiera y tiene ∞ posibilidades



- **Espacio de acciones:** ∞
- Infinito y discreto (hay infinitos ángulos enteros y decimales)

Tablero bidimensional estocástico

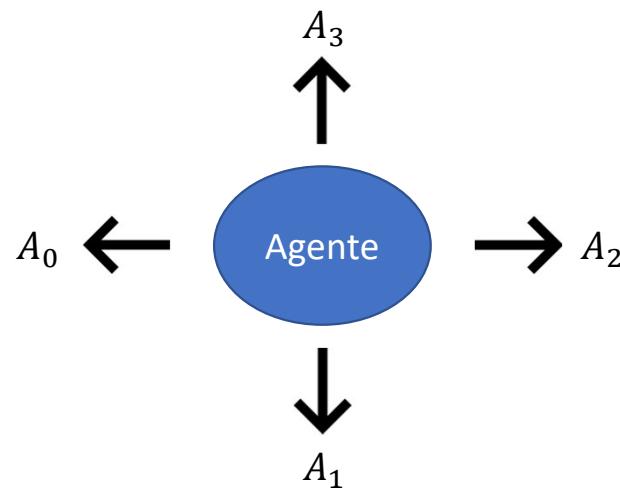
Otro tipos de acciones

- En el ejemplo del tablero bidimensional hay:
 - 1 variable por 1 acción
 - Cuando el agente ejecuta 1 acción, se puede mover en 1 sola dirección: arriba, abajo, derecha o izquierda
- En el ejemplo del carro autónomo y en otros muchos ejemplos hay:
 - n variables por 1 acción
 - Acción: girar a la derecha
 - Variables: girar el timón, acelerar, frenar (involucra varias variables para cumplir con dicha acción)

Tablero bidimensional estocástico

Función de transición y recompensa

- Recordar que la recompensa cuando el agente llegue a la casilla 15 (meta) es 1
- Recordar que la recompensa en cualquiera otra casilla diferente a la 15 (meta) es 0



Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

Tabla de transición

- Se diligenciará una tabla del estilo de las anteriores pero...
 - a) Tener en cuenta que son 16 estados diferentes
 - b) La tabla de transición va a tener 152 filas (imposible de plasmar)
 - c) Se van a mirar casos particulares

Tablero bidimensional estocástico

Tabla de transición

- Casos particulares

a) Casilla de inicio



Tablero bidimensional estocástico

Tabla de transición

- Casos particulares

b) Huecos del tablero



Tablero bidimensional estocástico

Tabla de transición

- Casos particulares

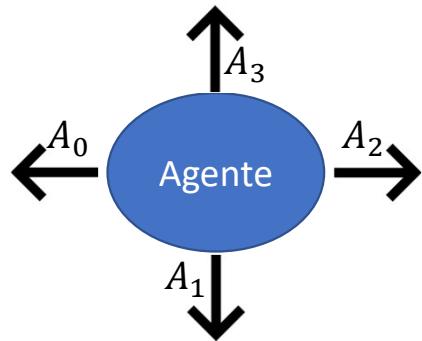
c) Casilla 14 que permite llevar al agente a la meta



Tablero bidimensional estocástico

Caso 1:

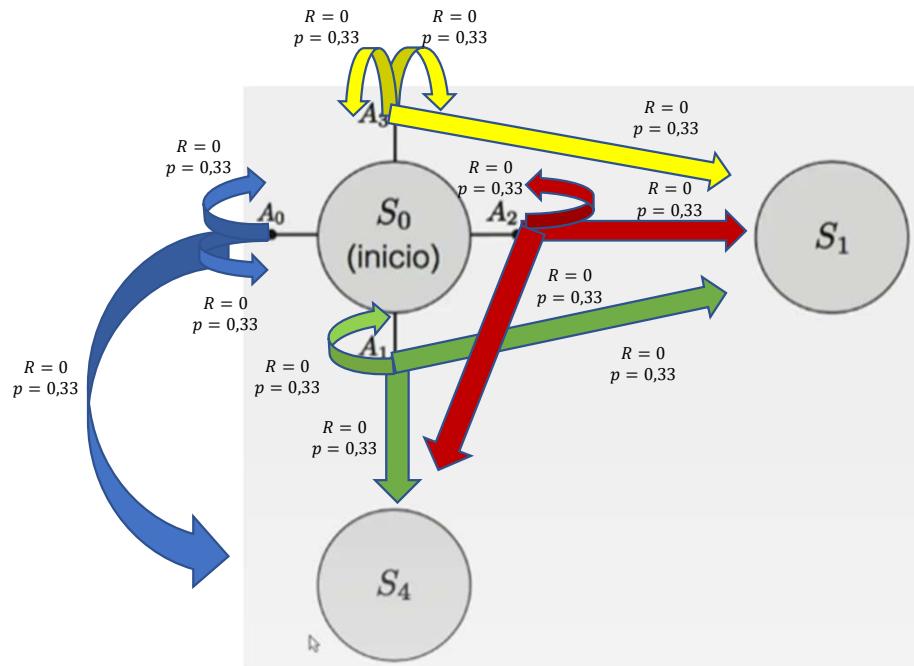
a) Casilla de inicio



$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_0	$A_0 \leftarrow$	S_0	0.33	0
S_0	$A_0 \uparrow$	S_0	0.33	1
S_0	$A_0 \downarrow$	S_4	0.33	0
S_0	$A_1 \downarrow$	S_4	0.33	0
S_0	$A_1 \leftarrow$	S_0	0.33	1
S_0	$A_1 \rightarrow$	S_1	0.33	0
S_0	$A_2 \rightarrow$	S_1	0.33	0
S_0	$A_2 \uparrow$	S_0	0.33	1
S_0	$A_2 \downarrow$	S_4	0.33	0
S_0	$A_3 \uparrow$	S_0	0.33	0
S_0	$A_3 \leftarrow$	S_0	0.33	1
S_0	$A_3 \rightarrow$	S_1	0.33	0

Tablero bidimensional estocástico

Grafo para el caso 1



Tablero bidimensional estocástico

Explicación tablas de transición

¿Por qué la tabla posee 12 filas?

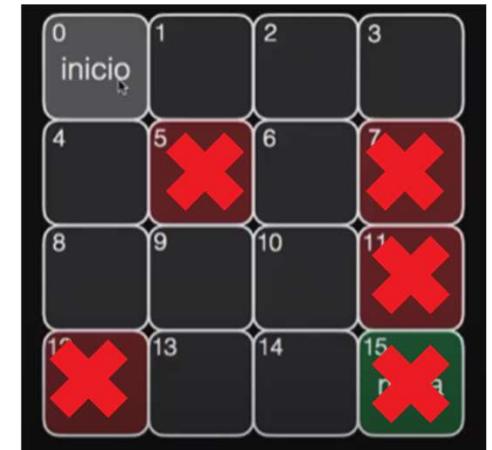
- Hay 4 posibles acciones A_0, A_1, A_2, A_3

¿Cuáles casillas cumplen lo anterior?

- 0, 1, 2, 3, 4, 6, 8, 9, 10, 13, 14

¿Cuántas filas salen en total?

$$12 \text{ (filas por caso)} \times 11 \text{ (casillas que cumplen el caso)} = \mathbf{132}$$



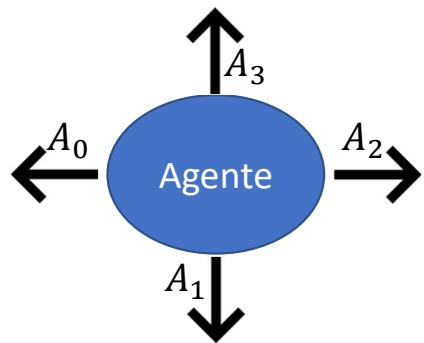
Tablero bidimensional estocástico

Caso 2

a) Huecos (caso terminal)



$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_5	$A_0 \leftarrow$	S_5	1	0
S_0	$A_1 \downarrow$	S_5	1	0
S_0	$A_2 \rightarrow$	S_5	1	0
S_0	$A_3 \uparrow$	S_5	1	0

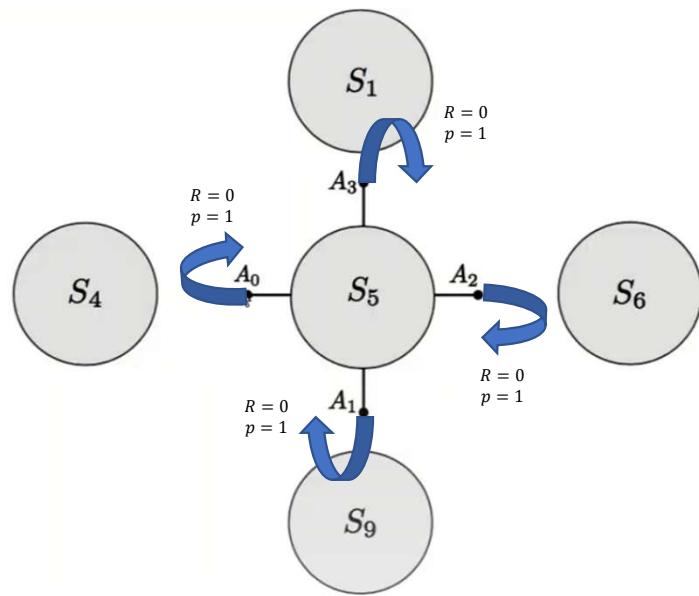


No hay componente estocástico, pues
independientemente de la acción, ya se está en un
hueco y no hay movimiento

Salen

Tablero bidimensional estocástico

Grafo para el caso 2



Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

Explicación tablas de transición

¿Por qué la tabla posee 4 filas?

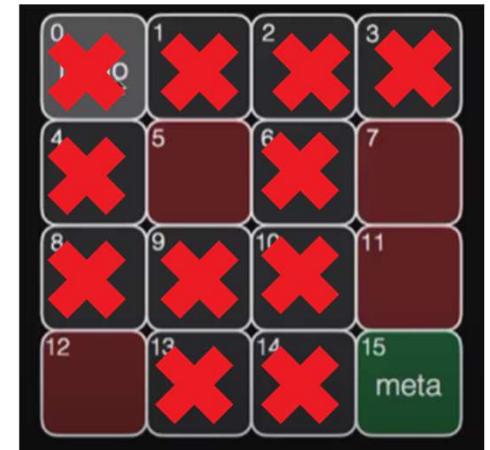
- Hay 4 posibles acciones A_0, A_1, A_2, A_3

¿Cuáles casillas cumplen lo anterior?

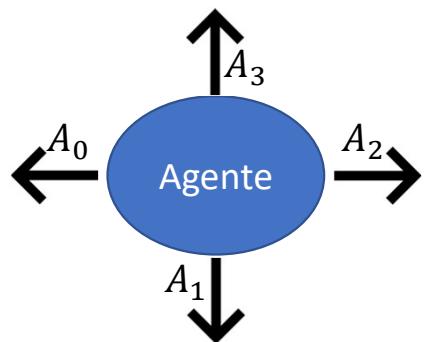
- 5, 7, 11, 12, 15

¿Cuántas filas salen en total?

$$4 \text{ (filas por caso)} \times 5 \text{ (casillas que cumplen el caso)} = 20$$



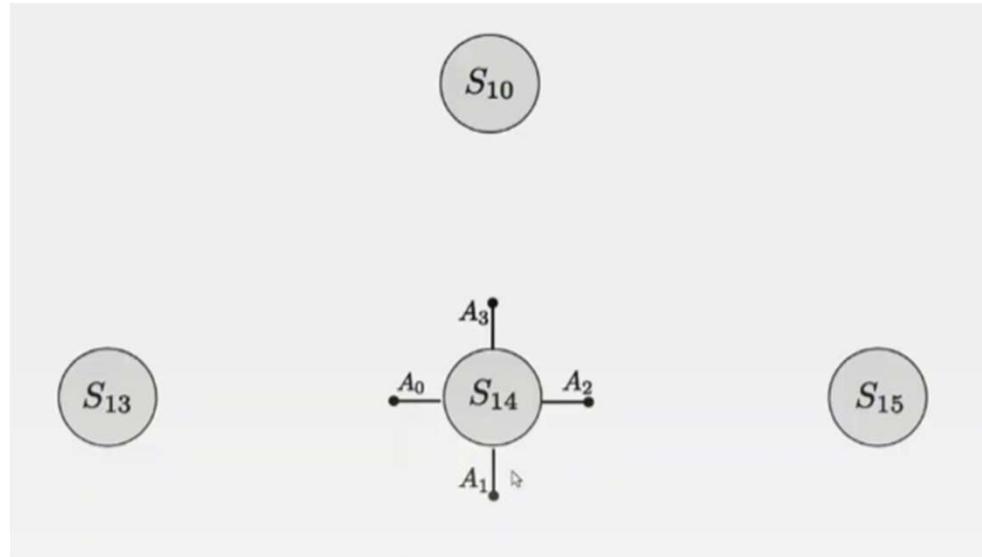
Tablero bidimensional estocástico



$s_{t-1}(s)$	$A_{t-1}(a)$	$S_t(s')$	$p(s' s, a)$	R_t
S_{14}	$A_0 \leftarrow$	S_{13}	0.33	0
S_{14}	$A_0 \uparrow$	S_{10}	0.33	1
S_{14}	$A_0 \downarrow$	S_{14}	0.33	0
S_{14}	$A_1 \downarrow$	S_{14}	0.33	0
S_{14}	$A_1 \rightarrow$	S_{15}	0.33	1
S_{14}	$A_1 \leftarrow$	S_{13}	0.33	0
S_{14}	$A_2 \rightarrow$	S_{15}	0.33	1
S_{14}	$A_2 \uparrow$	S_{10}	0.33	1
S_{14}	$A_2 \downarrow$	S_{14}	0.33	0
S_{14}	$A_3 \uparrow$	S_{10}	0.33	0
S_{14}	$A_3 \leftarrow$	S_{13}	0.33	0
S_{14}	$A_3 \rightarrow$	S_{15}	0.33	1

Tablero bidimensional estocástico

Grafo para el caso 3



Realizar grafo para el caso 3 -> casilla 14

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

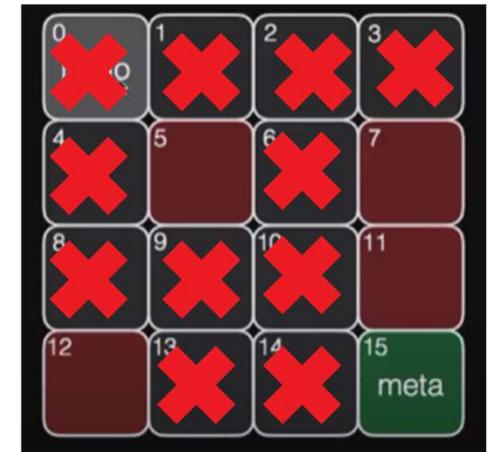
Explicación tablas de transición

¿Por qué la tabla posee 12 filas?

- Hay 4 posibles acciones A_0, A_1, A_2, A_3

¿Cuántas filas salen en total?

$$12 \text{ (filas del caso)} \times 1 \text{ (casillas que cumplen el caso)} = \mathbf{12}$$



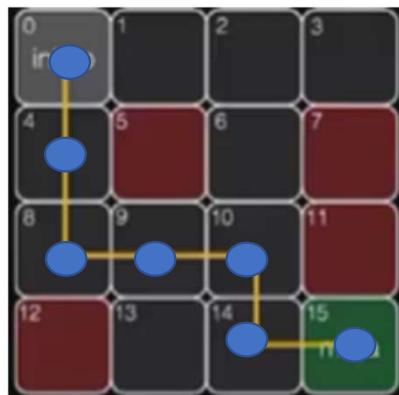
Tablero bidimensional estocástico

Horizonte

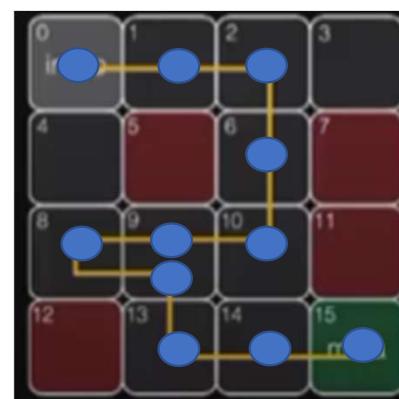
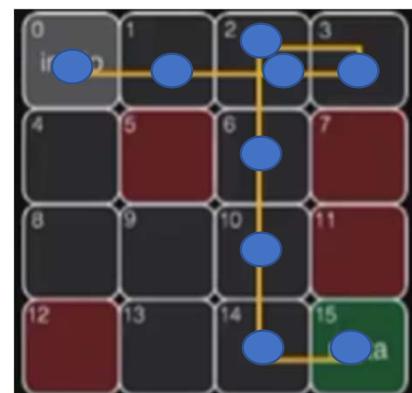
- El agente debería ganar el juego en la menor cantidad de jugadas, es decir, en el menor tiempo posible
- Entonces en el proceso de decisión de Márkov se añade explícitamente la variable tiempo, usando el concepto **horizonte**
- **Horizonte:** Duración de la interacción agente-entorno

Tablero bidimensional estocástico

Tipos de horizonte

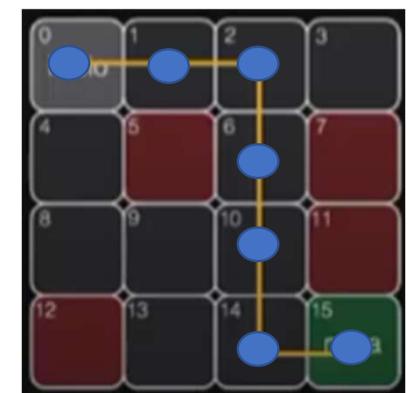


8 movimientos
Horizonte=8



10 movimientos
Horizonte=10

6 movimientos
Horizonte=6



- En todos los casos el agente cumple con el objetivo
- Pero hay opciones más óptimas que otras, caso 1 y 3

Tablero bidimensional estocástico

Tipos de horizonte

Puede ser finito o infinito

- Horizonte finito: Número de jugadas límite, tiempo límite para ganar (ya sea discreto o continuo)



Tablero bidimensional estocástico

Tipos de horizonte

Puede ser finito o infinito

- Horizonte infinito: Sin número de jugadas límite, ni tiempo límite para ganar. Así la cantidad de casillas sea finito, puede quedarse probando combinaciones indefinidamente.



Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Tablero bidimensional estocástico

Tipos de horizonte

El horizonte infinito tiene una subcategoría, el **indefinido**

- **Horizonte infinito indefinido:** Cabe destacar que el horizonte es infinito porque el agente se puede demorar y hacer un número de jugadas indefinido. **PERO** si cae en un hueco, allí terminaría su juego, pero no sabemos cuántas jugadas se requieren para que eso ocurra.



Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Descuento

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

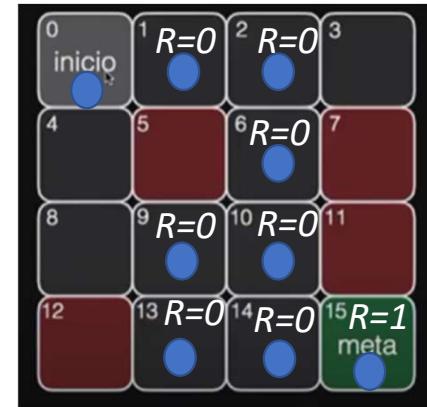
Tablero bidimensional estocástico

Descuento

- Ruta 1

Recompensa total= $0+0+0+0+0+0+\mathbf{1}=1$

Horizonte= 8



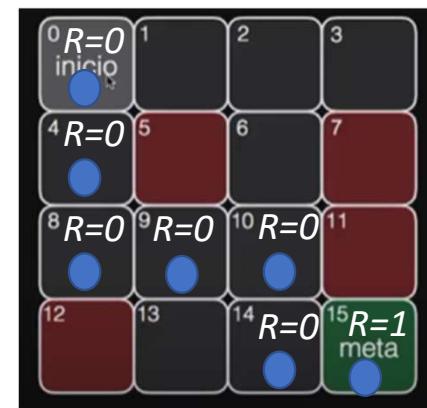
Tablero bidimensional estocástico

Descuento

- Ruta 2

Recompensa total= $0+0+0+0+0+\mathbf{1}=1$

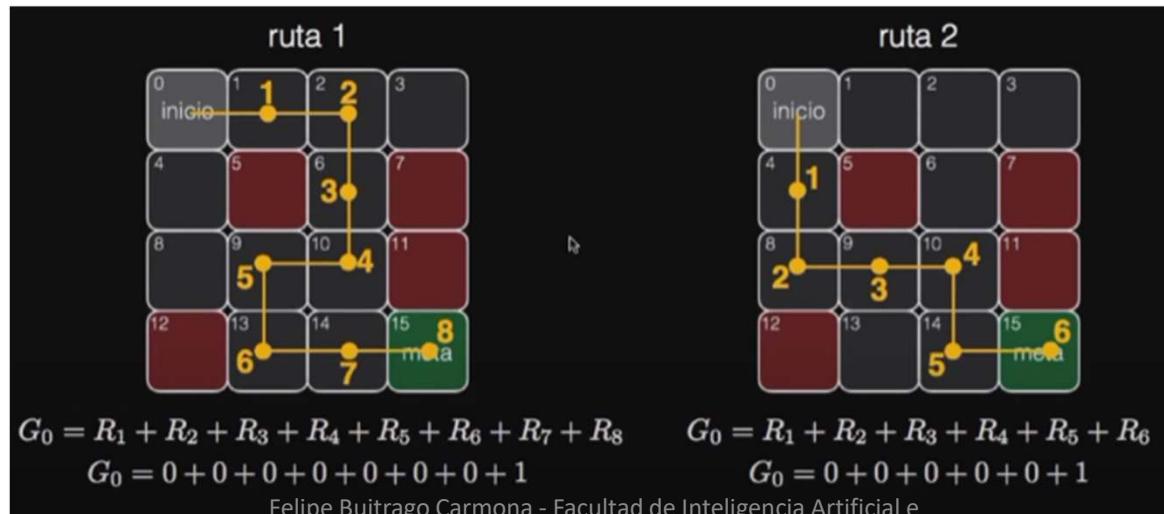
Horizonte= 6



Tablero bidimensional estocástico

Descuento

- El **retorno** es la recompensa total, es decir, la suma de las recompensas individuales representado como G_0



Tablero bidimensional estocástico

Descuento

- Ruta 2 es la más **ÓPTIMA**, requiere menos jugadas para llegar a la meta
- Sin embargo, el agente NO sabe y no tiene manera (hasta este punto) de saber cuál es la ruta más **ÓPTIMA**, para él es lo mismo porque en ruta 1 y ruta 2, obtiene una **RECOMPENSA** de 1
- **El descuento permite llegar al agente a la meta de manera óptima**

Tablero bidimensional estocástico

Descuento: Importancia de las recompensas en el tiempo

1. ¿Cuál es el problema?

Ruta 1 se tarda más en alcanzar la recompensa que la Ruta 2

2. ¿Cómo atacarlo?

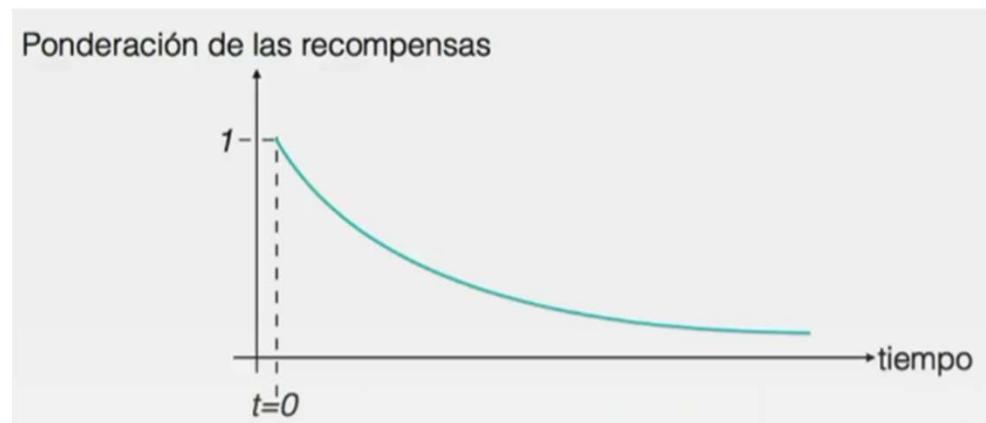
Concepto del **descuento**, entre más tiempo pase, las recompensas van a tener **menos peso**.

3. ¿Cómo sería?

Ruta 1 debería obtener una **recompensa menor** a la Ruta 2, por haberse tardado más

Tablero bidimensional estocástico

Descuento: Ponderación de las recompensas

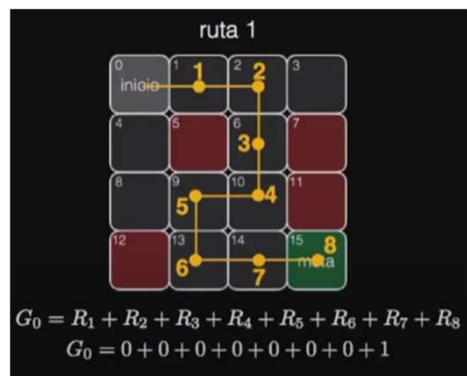


- A medida que va transcurriendo el tiempo, la ponderación de las recompensas va aumentando
- A **mayor** tiempo, **menor** ponderación, por ende, **menor** recompensa
- A **menor** tiempo, **mayor** ponderación, por ende, **mayor** recompensa

Tablero bidimensional estocástico

Descuento: Ponderación de las recompensas

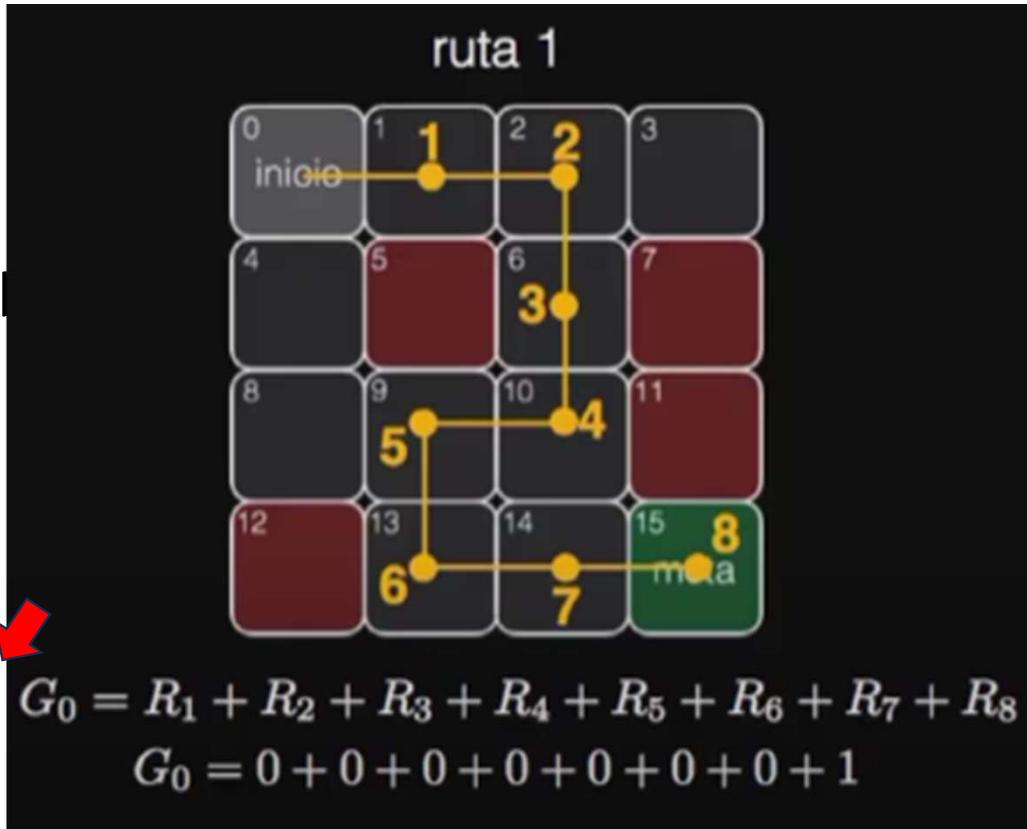
- γ : **Factor de descuento**
- $0 < \gamma < 1$: No puede ser 1, porque a medida que pase el tiempo, la ponderación sería la misma en todos los instantes de tiempo.
- $\gamma = 0,99$ (el valor que usualmente se le da)



En este caso, casa una de las 8 recompensas, está ponderada por un factor de 1, todas están multiplicadas por 1. Pues no hay un factor de descuento hasta ese punto

La idea con el factor de descuento, es ir penalizando.

Tablero Descuento



compensas

9^x , a medida que
1 y por tanto, el
on el **número de**

nalizada por la

1 será **menor**
de movimientos.

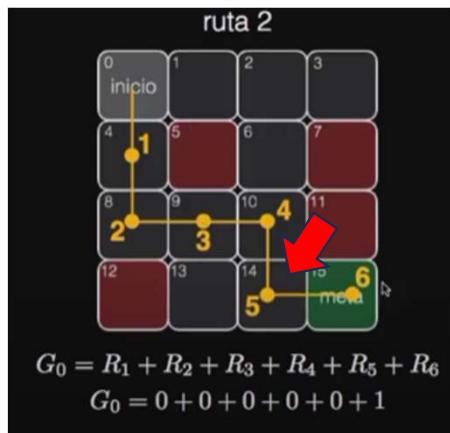
Implementando el factor de descuento $\gamma=0,99$:

$$G_0 = (0,99)^0 * 0 + (0,99)^1 * 0 + (0,99)^2 * 0 + (0,99)^3 * 0 + (0,99)^4 * 0 + (0,99)^5 * 0 + (0,99)^6 * 0 + (0,99)^7 * 1$$

$$G_0 = 0,932$$

Tablero bidimensional estocástico

Descuento: Ponderación de las recompensas



- Si tenemos un **factor de ponderación** 0.99^x , a medida que haya 1 nuevo movimiento, x aumenta en 1 y por tanto, el factor de ponderación va **disminuyendo con el número de jugadas**.
- Cada jugada consecutiva, se va viendo penalizada por la “demora” de la anterior.
- Notamos que, la recompensa para la ruta 2 será **mayor** que para la ruta 1, dado el número **menor** de movimientos.

Implementando el factor de descuento $\gamma=0,99$:

$$G_0 = (0,99)^0 * 0 + (0,99)^1 * 0 + (0,99)^2 * 0 + (0,99)^3 * 0 + (0,99)^4 * 0 + (0,99)^5 * 1$$

$$G_0 = 0,951$$

Tablero bidimensional estocástico

Descuento: Notación matemática

- **Retorno sin descuento (SIN penalidades):**

$$G_t = R_{t+1} + R_{t+2} + R_{t+3} + \dots$$

G_t : Retorno sin descuento

R_{t+x} : Recompensa individual en el instante de tiempo posterior, sin descuento

Tablero bidimensional estocástico

Descuento: Notación matemática

- **Retorno con descuento (CON penalidades):**

$$G_t = \gamma^0 * R_{t+1} + \gamma^1 * R_{t+2} + \gamma^2 * R_{t+3} + \gamma^3 * R_{t+4} + \dots$$

γ : Factor de descuento

R_{t+x} : Recompensa individual en el instante de tiempo posterior

∞ : Horizonte infinito (cantidad infinita de pasos), sino fuera así, se cambia el límite inferior de la sumatoria

k : Exponente para la elevación de r

$$G_t = \sum_{k=0}^{\infty} \gamma^k * R_{t+k+1} \quad \text{Ecuación generalizada}$$

La formulación del problema: Los procesos de decisión de Markov

- Elementos presentes en la interacción agente-entorno
 $\{S, A, p, R, H, \gamma\}$
- S : Espacio de estados
- A : Espacio de acciones
- p : Función de transición
- R : Recompensa (ya sea premio o castigo)
- H : Horizonte (duración de interacción agente-entorno)
- γ : Descuento

El objetivo del agente

Definición del aprendizaje por refuerzo, **RECORDAR**:

- Lograr que un agente aprenda a tomar decisiones de manera **óptima**

$$G_t = \sum_{k=0}^{\infty} \gamma^k * R_{t+k+1}$$

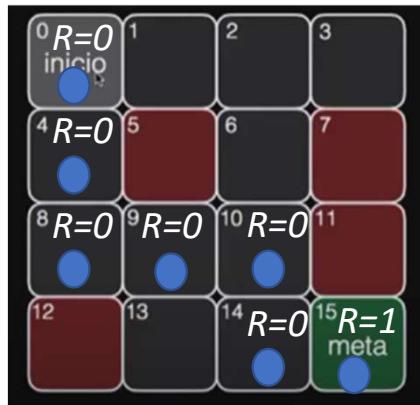
- Siendo G_t el retorno, se debe obtener el más alto posible, a esto se refiere la definición con **óptimo**

Objetivo del agente:

- Encontrar una **secuencia de acciones** que **maximice** el retorno

Toma de decisiones de manera óptima

Retomando un ejemplo anterior...



**EJEMPLO IDEAL
SIN COMPONENTE ESTOCÁSTICO**

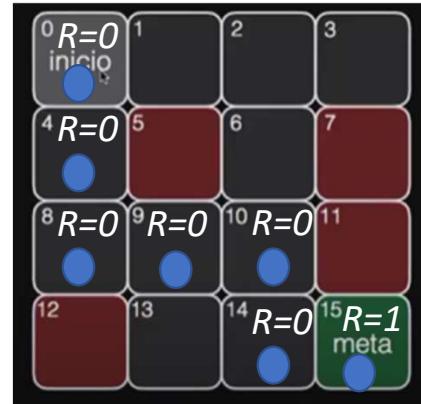
$$\gamma = 0,99$$

$$G_0 = (0,99)^0 * 0 + (0,99)^1 * 0 + (0,99)^2 * 0 + (0,99)^3 * 0 + (0,99)^4 * 0 + (0,99)^5 * 1 = 0,95$$

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Toma de decisiones de manera óptima

Retomando un ejemplo anterior...



CON COMPONENTE ESTOCÁSTICO

- Hay un $p = 33\%$ de que pase de la casilla 0 a la 4 (como se desea) ↓
- Hay un $p = 33\%$ de que pase de la casilla 0 a la 1 (como NO se desea) →
- Hay un $p = 33\%$ de que pase de la casilla 0 a la 0 (como NO se desea) ←

Toma de decisiones de manera óptima

En general para cualquier ruta...



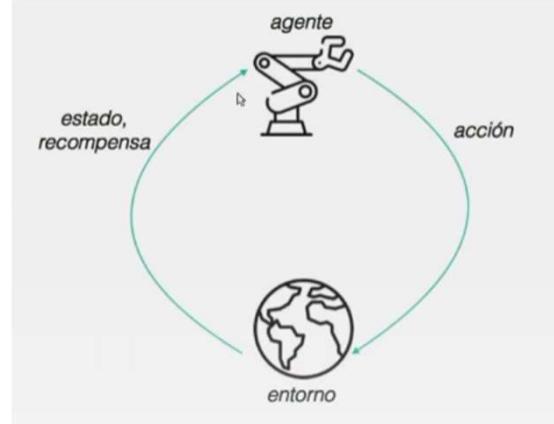
CON COMPONENTE ESTOCÁSTICO

- En general, lo mismo aplica para las casillas 0, 1, 2, 6, 10, 14 de la ruta. El movimiento se puede dar en **direcciones ortogonales** a la del movimiento deseado por el agente

La política

Toma de decisiones óptimas por parte del agente

Recordando la interacción agente-entorno... Hasta que el agente aprenda a desenvolverse en un entorno dado por repetición y refuerzo.



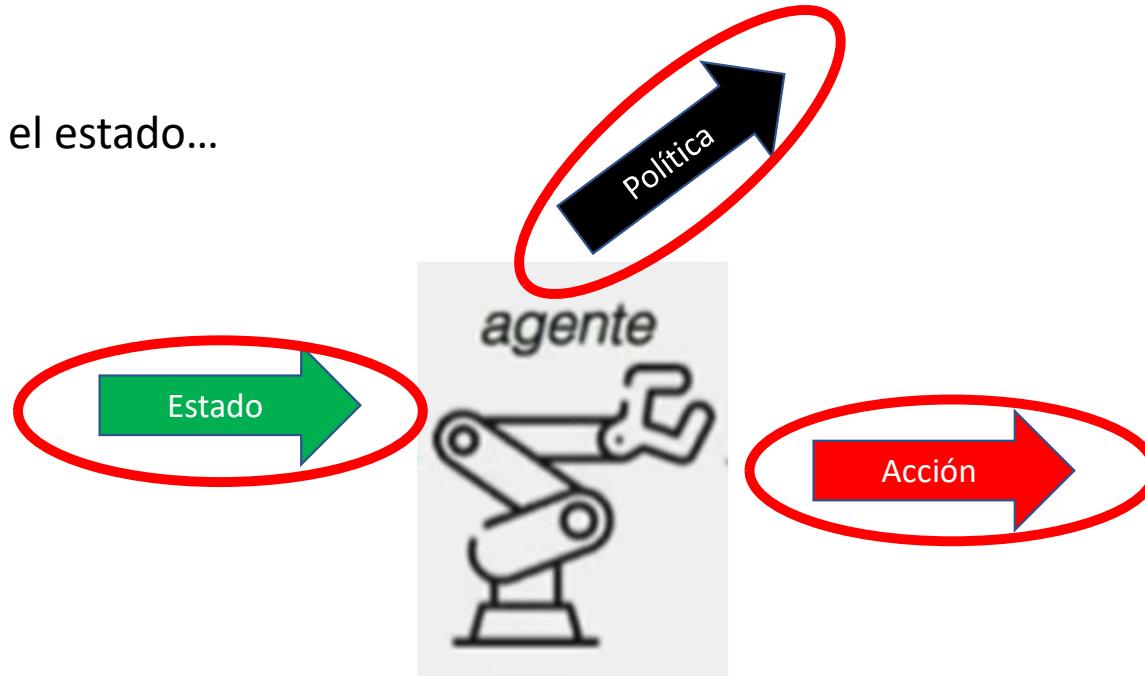
La política De estados a acciones

Enfocándonos en el agente...



La política De estados a acciones

Enfocándonos en el estado...



Lo que le permite el agente mapear de estados a acciones es el **cerebro** de ese agente, conocido como **política** (π)

Política determinista

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

La política Políticas determinísticas

Las políticas determinísticas ocurre en entornos determinísticos, no hay componente de aleatoriedad

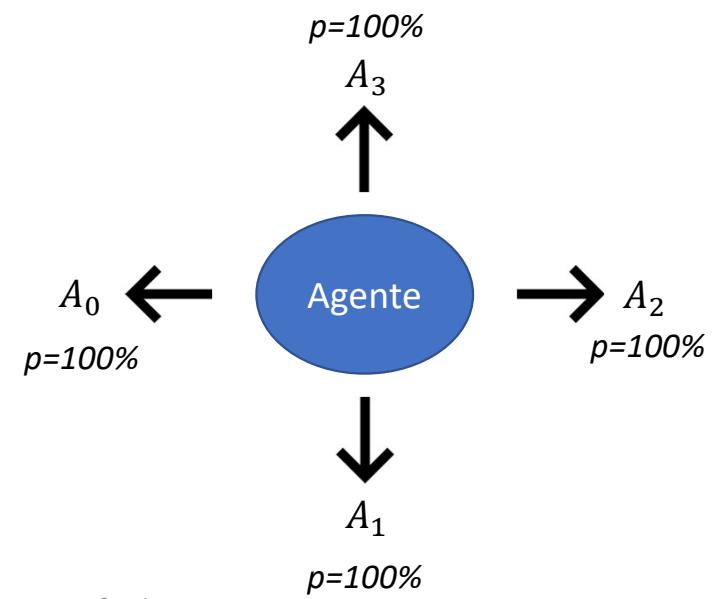
$$a = \pi(s)$$

- a : acción
- $\pi(s)$: *política*

La política

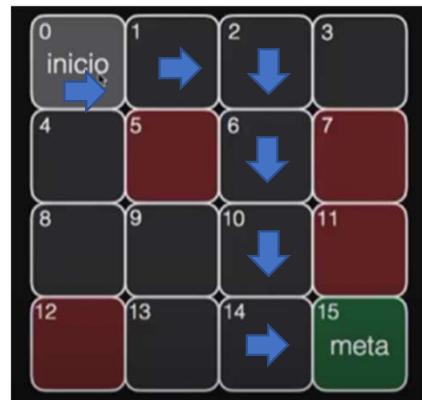
Políticas determinísticas

Las políticas determinísticas ocurre en entornos determinísticos, no hay aleatoriedad



La política Políticas determinísticas

Las políticas determinísticas ocurre en entornos determinísticos, no hay aleatoriedad



$$\pi(S_0) = A_2$$

$$\pi(S_1) = A_2$$

$$\pi(S_2) = A_1$$

$$\pi(S_3) = A_1$$

$$\pi(S_4) = A_1$$

$$\pi(S_5) = A_2$$

Política estocástica

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

La política Políticas estocásticas

Dado que no tenemos certeza hacia donde se va a ejecutar la acción que emprenda el agente, la política estocástica, se representará como una probabilidad condicional...

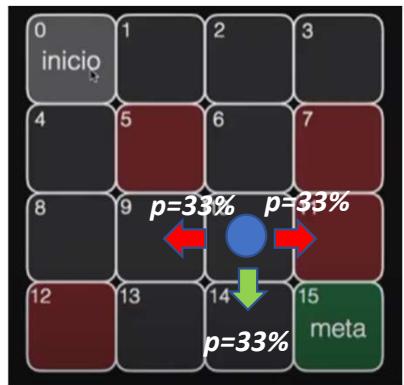
$$\pi(a | s) = P[A_t = a | S_t = s]$$

Traducción de la ecuación: La política estocástica es la probabilidad de que el agente ejecute en el instante de tiempo t, la acción a, teniendo en cuenta el estado S (la condición)

La política

Políticas estocásticas

Teniendo en cuenta las reglas del juego estocásticas, donde por ejemplo:



Recordando que...

Dirección verde: Movimiento deseado por el agente

Dirección roja: Movimiento posible dado el componente estocástico

La política Políticas estocásticas

¿Cómo sería la política estocástica para el estado S_{10} ?



$$\Leftrightarrow \pi(A_0 | S_{10}) = 0,02$$

$$\Downarrow \pi(A_1 | S_{10}) = 0,75$$

$$\Rightarrow \pi(A_2 | S_{10}) = 0,05$$

$$\Uparrow \pi(A_3 | S_{10}) = 0,08$$

- La suma de la distribución de las probabilidades debe ser 1
- Ejemplo de distribución de probabilidades

- La política estocástica no va a indicar una sola acción a ejecutar partiendo de un estado determinado, sino que nos entrega una **distribución de probabilidades**
- No hay una indicación certera de la acción a emprender → **política estocástica**

Función Estado Valor

Cuantificar que tan bueno es un estado

Función estado-valor

Hay una limitación en la política...

- Decide SOLAMENTE la siguiente acción a tomar
- No permite **cuantificar** la bondad del estado y acción a tomar por el agente, es decir, si es bueno o malo

Estas limitaciones son arregladas por...

- La función estado-valor, permiten **cuantificar** la bondad de un estado o acción en particular.

Función estado-valor

El agente recibe:

- Estado
- Recompensa

Pero la política según su notación matemática, sólo recibe:

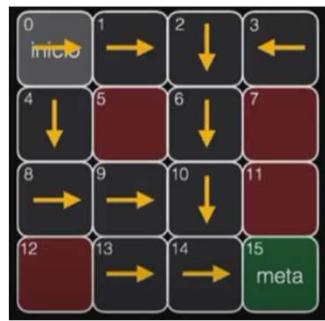
- El estado como entrada, se olvida de la recompensa

Función estado-valor

Comparando políticas...

- **Recordemos que...** La política le indica al agente la secuencia de acciones que debe realizar en un entorno

Política 1
 π_1



VS

Política 2
 π_2



- Sería muy “intuitivo” pensar que π_1 es mejor que π_2 , por la lógica de sus direcciones para llegar a la meta. Pero en este punto es muy subjetivo afirmar eso.
- Por lo anterior, es mejor poder cuantificar la bondad de las políticas (medir numéricamente)

Función estado-valor Comparando estados...

¿Qué estado será mejor para el agente?

Estado 13
 S_{13}



VS

Estado 14
 S_{14}



- Sería muy “intuitivo” pensar que S_{14} es mejor que S_{13} , porque está a 1 paso de la meta. Pero en este punto es muy subjetivo afirmar eso.
- Por lo anterior, es mejor poder cuantificar la bondad de los estados (medir numéricamente)
- Hay un componente estocástico, el hecho de que esté en S_{14} no quiere decir que va a llegar directo a S_{15}

Función estado-valor Retorno esperado

Para entender el concepto de **Retorno esperado**, se debe partir de 2 condiciones, teniendo en cuenta que se tiene:

Hay 1 proceso estocástico + El agente tiene 1 política

Condición 1) Como existe una política, no habrá secuencias de acciones arbitrarias

Condición 2) Como el entorno es estocástico, en cada estado aparecerán múltiples alternativas de movimiento

Función estado-valor Retorno esperado

Partiendo de Condición 1) y Condición 2):

El **retorno esperado**, es el promedio de todos los posibles retornos obtenidos, después de analizar todas las posibles secuencias de acciones

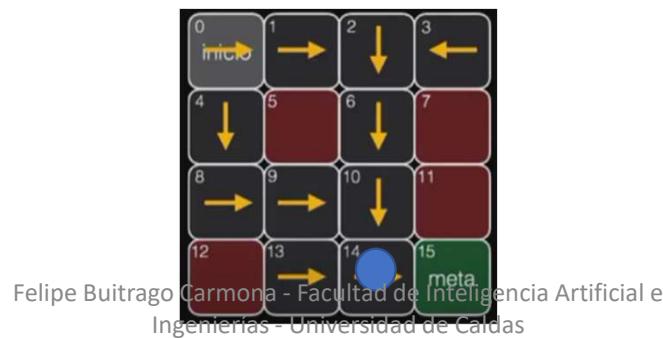
¿Por qué el término “esperado”?

Vamos a tener múltiples trayectorias, y no sólo 1, el retorno no será la suma de las recompensas individuales de 1 sola trayectoria. Por ello se promediará

El valor de un estado

El **valor de un estado** es el **retorno esperado** que se obtendrá si el agente partiera del estado S , y ejecuta la secuencia de acciones propuesta por la política π

Ejemplo: Estamos en S_{14} , y queremos saber QUÉ TAN BUENO es ese estado (lo que queríamos cuantificar anteriormente)



El valor de un estado



El agente se encuentra en S_{14} , y de acuerdo a su política, se le indica que se mueva a la derecha para alcanzar S_{15} :

- $S_{14} \rightarrow S_{15}$: 33% (movimiento esperado)
- $S_{14} \rightarrow S_{10}$: 33% (por componente estocástico)
- $S_{14} \rightarrow S_{14}$: 33% (por componente estocástico)

Asignando valores a las trayectorias, por temas de exemplificar:

- Valor para $S_{14} \rightarrow S_{15} = 10$
- Valor para $S_{14} \rightarrow S_{10} = 9$
- Valor para $S_{14} \rightarrow S_{14} = 2$

$$\text{Valor estado} = \frac{10 + 9 + 2}{3} = 7$$

Función estado-valor

Permite calcular el **valor de cada estado** (o retorno esperado), siguiendo la política, para **cada uno de los estados** que hacen parte del entorno (tablero, contexto, etc)



Ejemplo: Si nos paramos en S_2

1. metemos ese estado en la función estado-valor
2. La función arroja el valor obtenido para ese estado en particular

Utilidad de la función estado-valor

En términos generales, nos ayuda a identificar cuáles son los estados que funcionan **mejor** para el objetivo del agente y cuáles no funcionan **tan bien**

Notación matemática de la función estado-valor

Puntaje de un estado (Gt) . Retorno

$$v_{\pi}(s) = E_{\pi} \left[\sum_{k=0}^{\infty} \lambda^k R_{t+k+1} | S_t = s \right]$$

$v_{\pi}(s)$: Función estado-valor, con base en la política que dirige las acciones del agente, tomando como parámetro de entrada el estado a evaluar.

E_{π} : Valor esperado teniendo en cuenta la política en un estado dado (valor esperado se refiere al promedio)

$\sum_{k=0}^{\infty} \lambda^k R_{t+k+1}$: Retorno de la trayectoria

S_t : Estado en el instante de tiempo t

Notación matemática de la función estado-valor

$$v_{\pi}(s) = E_{\pi} \left[\sum_{k=0}^{\infty} \lambda^k R_{t+k+1} | S_t = s \right]$$

Traducción de la ecuación: La función estado valor, es igual al promedio del retorno que se obtendría si el agente partiera del estado s , tomando todas las posibles trayectorias y retornos que se puedan obtener partiendo del estado s

Notación matemática de la función estado-valor

$$v_{\pi}(s) = E_{\pi} \left[\sum_{k=0}^{\infty} \lambda^k R_{t+k+1} | S_t = s \right]$$

Traducción de la ecuación: La función estado valor, es igual al promedio del retorno que se obtendría si el agente partiera del estado s , tomando todas las posibles trayectorias y retornos que se puedan obtener partiendo del estado s

Felipe Buitrago Carmona - Facultad de Inteligencia Artificial e
Ingenierías - Universidad de Caldas

Función Estado Acción

Cuantificar que tan buena es una acción

Función acción-valor

- Posee una filosofía similar a la de la función estado-valor
- En la función estado-valor se le asignaba una puntuación a un estado dado, dependiendo de la casilla donde se encontrara el agente
- En la función acción-valor, no sólo se tiene en cuenta el estado en el que está el agente, sino el movimiento que ejecuta

Política 1
 π_1



VS

Política 2
 π_2



Política 1: agente en la casilla 9 a la derecha Política 2: agente en la casilla 9 abajo

Función acción-valor

- Al igual que en la función estado-valor, no es posible determinar solamente observando el tablero, cuál de las 2 políticas tiene el mejor movimiento para la casilla 9

Política 1
 π_1



Política 2
 π_2



- Debe haber algo que permita cuantificar la bondad de cada acción en un escenario dado, pues pueden haber múltiples políticas diferentes que dicten un comportamiento distinto para la casilla 9.

Función acción-valor o función Q (quality)

Permite calcular el **retorno esperado** que obtendría el agente al tomar la decisión a , estando en el estado s y siguiendo la política π

Política 1
 π_1



Política 2
 π_2



Ejemplo: Partiendo de la casilla 9, la función acción-valor analiza:

- El estado del agente (casilla en la que se encuentra)
- Acción que va a ejecutar (dirección de la flecha)

Función acción-valor o función Q (quality)

- Con los dos elementos anteriores (estado y acción), genera una puntuación
- Y da un valor de 0,82 (retorno esperado) para la política 1 y 0,75 (retorno esperado) para la política 2
- Por tanto, la política que en ese caso funcionaría mejor, es la 1 dado que tiene mejor retorno esperado.

Política 1

$$\pi_1 \\ 0,82$$



/ VS \

Política 2

$$\pi_2 \\ 0,75$$



Notación matemática de la función acción-valor

$$q_{\pi}(s, a) = E_{\pi} \left[\sum_{k=0}^{\infty} \lambda^k R_{t+k+1} | S_t = s, A_t = a \right]$$

$q_{\pi}(s, a)$: Función acción-valor, que arroja el valor de la acción, con base en la política que dirige las acciones del agente, tomando como parámetro de entrada el estado y la acción a evaluar.

E_{π} : Valor esperado teniendo en cuenta la política en un estado dado (valor esperado se refiere al promedio)

$\sum_{k=0}^{\infty} \lambda^k R_{t+k+1}$: Retorno de la trayectoria

$S_t = s, A_t = a$: Estado en el instante de tiempo t y acción en el instante de tiempo t

Notación matemática de la función acción-valor

$$q_{\pi}(s, a) = E_{\pi} \left[\sum_{k=0}^{\infty} \lambda^k R_{t+k+1} | S_t = s, A_t = a \right]$$

Traducción de la ecuación: La función acción-valor, es igual al promedio del retorno que se obtendría si el agente partiera del estado s y ejecutando la acción a, tomando todas las posibles trayectorias y retornos que se puedan obtener partiendo del estado s y acción a.

Ecuaciones para las funciones estado-valor y acción-valor

Función estado-valor

$$v_{\pi}(s) = E_{\pi}[G_t | S_t = s]$$

Función acción-valor

$$q_{\pi}(s, a) = E_{\pi}[G_t | S_t = s, A_t = a]$$

Las 2 funciones anteriores presentan limitaciones, pues:

- Mencionan la política, retorno, estado y acción. PERO, no lo es de manera **explícita**

¿Por qué es importante que sea explícito?

- Es fundamental para resolver dichas funciones a través de **algoritmos** que se vayan a desarrollar (aprendizaje por refuerzo)

Ecuación de Bellman (explícita) para la función estado-valor

- Richard Bellman: Realiza el primer algoritmo de “programación dinámica” del aprendizaje por refuerzo
- Realiza la transición de la función original estado-valor a la ecuación de Bellman

$$v_{\pi}(s) = E_{\pi}[G_t | S_t = s]$$



$$v_{\pi}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a) [r + \gamma v_{\pi}(s')]$$

Ecuación de Bellman (explícita) para la función estado-valor

$$v_{\pi}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a) [r + \gamma v_{\pi}(s')]$$

$\pi(a|s)$: Política

$p(s',r|s,a)$: Densidad de probabilidad conjunta, mide la probabilidad de que... si el agente se encuentra en el estado s y ejecuta la acción a, cuál es la probabilidad de que llegue a un estado s' y obtenga una recompensa r

r : recompensa de pasar de un estado s a un estado s' (después de todas las interacciones agente-entorno)

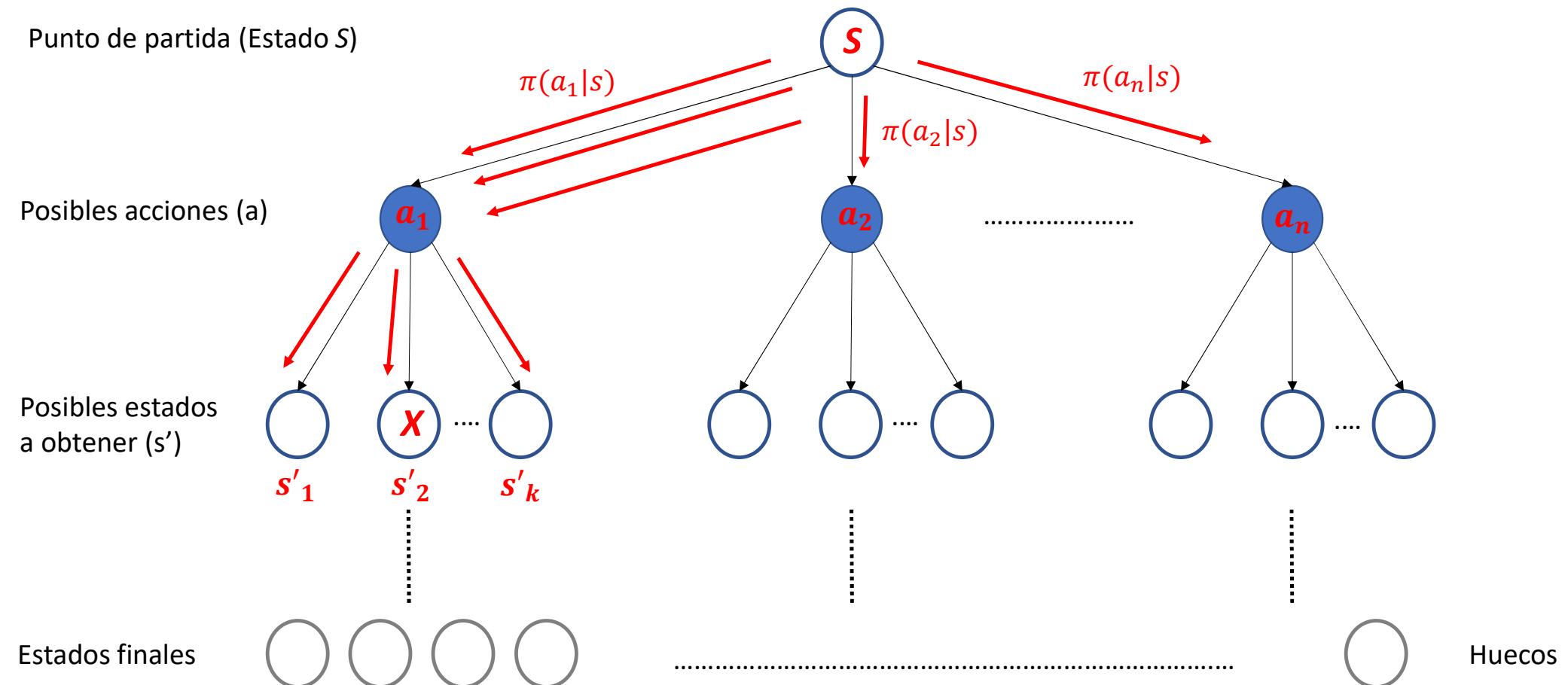
$\gamma v_{\pi}(s')$: Valor del estado en el siguiente estado, teniendo en cuenta una penalidad con el factor de descuento

Ecuación de Bellman (explícita) para la función estado-valor

Cosas por resaltar de la ecuación de Bellman para estado-valor...

- a) Relaciona el estado actual (s) con todos los posibles estados siguientes (s')
- b) Aparecen 3 sumatorias (una de 1 índice y otra de 2). La primera es sobre las acciones, la segunda es sobre los estados siguientes y la tercera es sobre las posibles recompensas
- c) En la ecuación de Bellman, se encuentran todos los elementos del proceso de decisión de Márkov explícitamente.

Ecuación de Bellman para la función estado-valor: Interpretación



Ecuación de Bellman (explícita) para la función acción-valor

- Richard Bellman: Realiza el primer algoritmo de “programación dinámica” del aprendizaje por refuerzo
- Realiza la transición de la función original acción-valor a la ecuación de Bellman

$$q_{\pi}(s, a) = E_{\pi}[G_t | S_t = s, A_t = a]$$



$$q_{\pi}(s, a) = \sum_{s', r} p(s', r | s, a) [r + \gamma v_{\pi}(s')]$$

Ecuación de Bellman (explícita) para la función acción-valor

$$q_{\pi}(s, a) = \sum_{s', r} p(s', r | s, a) [r + \gamma v_{\pi}(s')]$$

$p(s', r | s, a)$: Densidad de probabilidad conjunta, mide la probabilidad de que... si el agente se encuentra en el estado s y ejecuta la acción a , cuál es la probabilidad de que llegue a un estado s' y obtenga una recompensa r

r : recompensa de pasar de un estado s a un estado s' (después de todas las interacciones agente-entorno)

$\gamma v_{\pi}(s')$: Valor del estado en el siguiente estado, teniendo en cuenta una penalidad con el factor de descuento

Ecuación de Bellman (explícita) para la función acción-valor

Cosas por resaltar de la ecuación de Bellman para acción-valor...

- a) Esta ecuación de Bellman para acción-valor no es tan compleja
- b) Teniendo en cuenta el diagrama para la función estado-valor, ya no hay una gran cantidad de acciones por tomar, YA SE PARTE DE 1 SOLA Y DEFINIDA
- c) Dado lo anterior, desaparece la primera sumatoria que tiene la función estado-valor, pues tenemos una acción de partida