Notes on Markov Chains and Markov Decision Processes

Viviana Arrigoni
Internet of Things 2024-2025
Computer Science Department, Sapienza University of Rome

La probabilità che il sistema si trovi in uno stato Sj al tempo n+1 dipende solo dallo stato attuale e non dai precedenti (il futuro dipende solo dal presente e non dalla storia passata)

1 Markov Chains

A *discrete time stochastic process* is a sequence of random variables $X_0, X_1, \dots$ that evolve over discrete time steps. Each random variable X_n represents the state of a system at time n, and the transition to the next state is stochastic.

Definition 1.1. A *Discrete Time Markov Chain* or *Process* is a sequence of random variables $X_0, X_1, \dots$ such that $X_n \in S \ \forall n$, where S is a numerable or finite set of states, and:

$$\mathbb{P}[X_{n+1} = S_j | X_n = S_{i_n}, X_{n-1} = S_{i_{n-1}}, \dots, X_0 = S_{i_0}] = \mathbb{P}[X_{n+1} = S_j | X_n = S_{i_n}] \quad \forall n, m. \tag{1}$$

If Equation 1 holds, it defines the "memoryless" or "Markov" property of a process.

Definition 1.2. A D.T.M.C. is *homogeneous* or *has stationary transition probabilities* if the probability of transitioning from state s to state s' does not change over time. Formally:

PROBABILITA' DI TRANSIZIONE SONO COSTANTI NEL TEMPO

$$\mathbb{P}[X_{n+1} = s' | X_n = s] = \mathbb{P}[X_{n+m+1} = s' | X_{n+m} = s] \tag{2}$$

Definition 1.3. The *transition matrix* P of a H.D.T.M.C. is a $|S| \times |S|$ matrix such that its generic element $p_{i,j}$ represents the probability of transitioning from state i to state j, $p_{i,j} = \mathbb{P}[X_n = j | X_{n-1} = i]$.

$|S|$ NUMERO STATI

The transition matrix P must be stochastic, meaning that its rows sum to 1:

QUESTO ASSICURA CHE IL SISTEMA TRANSITI IN UNO STATO BEN PRECISO.

$$\sum_{j=1}^{|S|} p_{ij} = 1, \ \forall i. \tag{3}$$

For the remainder of this Section, we assume homogeneous discrete-time Markov Processes.

Example 1.1. Consider the following example. The state space is $S = \{\text{sunny S, rainy R, cloudy C}\}$. A possible transition matrix is:

$$P = \begin{array}{ccccc} & \text{S} & \text{R} & \text{C} & \\ \text{S} & 0.7 & 0.2 & 0.1 & = 1 \\ \text{R} & 0.4 & 0.5 & 0.1 & = 1 \\ \text{C} & 0.3 & 0.3 & 0.4 & = 1 \end{array} \tag{4}$$

We can also represent transitions as a *transition graph*, i.e., a bidirected weighted complete graph, as in Figure 1.

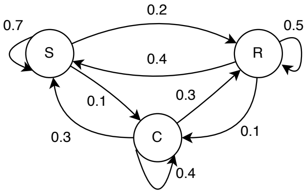


Figure 1: Transition Graph

SE VOGLIAMO CALCOLARE TALE PROBABILITA' CONSIDERANDO TUTTI I PERCORSI, DOVREMMO ELENCARLI TUTTI IMPRATICABILE.

Sometimes, we might be interested in knowing the probability of being in state s at time $n+1$, knowing that at time 0 we are in state s_0. If we try to compute such probability step by step, we should find all possible paths in the transition graph that begin in s_0 and end in state s with length $n+1$, which is impractical. However, another possible and more effective way exists. Let's build some intuition by observing $\mathbb{P}[X_{n+1}=s|X_0=s_0]$ when $n=1$, i.e., $\mathbb{P}[X_2=s|X_0=s_0]$. It holds that:

$$\mathbb{P}[X_2=s|X_0=s_0]=\sum_{s_k\in S}\mathbb{P}[X_2=s|X_1=s_k,X_0=s_0]\cdot\mathbb{P}[X_1=s_k|X_0=s_0]$$

DA S_K PASSIAMO AD S.

PASSIAMO DA UNO STATO INTERMEDIO $S_0 \to S_K$

$$=\sum_{s_k\in S}\mathbb{P}[X_2=s|X_1=s_k]\cdot\mathbb{P}[X_1=s_k|X_0=s_0] \quad (5)$$

Questa equazione esprime che la probabilità di essere in s al tempo 2, partendo da s0, è data dalla somma delle probabilità di passare per ogni stato intermedio sk, moltiplicata per la probabilità di transizione da sk a s.

$$=\sum_{s_k\in S}p_{s_k,s}\cdot p_{s_0,s_k}$$

SOMMIAMO TUTTE LE POSSIBILITA' SU TUTTI GLI STATI INTERMEDI

Let's consider the previous example, and let $s_0=$S and $s=$C. The last expression of Equation 5 is:

PROBABILITA' DI ANDARE IN CLOUDY IN 2 PASSI, PARTENDO DA SUNNY.

- p_{s_0,s_k} è la probabilità di andare dallo stato iniziale s_0 allo stato intermedio s_k.
- $p_{s_k,s}$ è la probabilità di andare dallo stato intermedio s_k allo stato finale s.
- s_k rappresenta tutti i possibili stati intermedi.
- La sommatoria $\sum_{s_k\in S}$ significa che dobbiamo sommare questi prodotti per **tutti** gli stati intermedi possibili s_k.

$$\sum_{s_k\in\{\mathrm{S,\ C,\ R}\}}p_{s_k,\mathrm{C}}\cdot p_{\mathrm{S},s_k}=p_{\mathrm{S,S}}p_{\mathrm{S,C}}+p_{\mathrm{S,C}}p_{\mathrm{C,C}}+p_{\mathrm{S,R}}p_{\mathrm{R,C}}=P_{S,*}P_{*,C} \quad (6)$$

PARTO DA SUNNY RIMANGO IN SUNNY ARRIVO IN C PARTO DA SUNNY VADO IN RAINY ARRIVO IN CLOUDY

Questa espressione compatta sign stiamo prendendo la riga corrispon S dalla matrice di transizione P, corrispondente a C da P, moltiplic valori e sommando i contributi.

Questa espressione compatta significa che stiamo prendendo la riga corrispondente a S dalla matrice di transizione P, e la colonna corrispondente a C da P, moltiplicando i valori e sommando i contributi.

that is, the multiplication of the row relative to state S by the column relative to state C in matrix P. The result is the element (S,C) of the matrix $P\cdot P=P^2$, i.e., $p^2_{S,C}$.
To build more intuition, let's consider now $n=2$. We want to compute $\mathbb{P}[X_3=s|X_0=s_0]$. We can see that:

PER CALCOLARE LA PROBABILITA' DI PASSARE DA UNO STATO s_0 AD UNO STATO s IN M PASSI SI USA LA POTENZA DELLA MATRICE P.

$$P[X_M=s \mid X_0=s_0]=p^{M}_{s_0,s}$$

ELEMENTO (s_0,s) DELLA MATRICE P^M

$$\begin{aligned}\mathbb{P}[X_3=s|X_0=s_0]&=\sum_{s_k\in S}\mathbb{P}[X_3=s|X_2=s_k]\cdot\mathbb{P}[X_2=s_k|X_0=s_0]\\ \text{(by Equation 5)}\quad&=\sum_{s_k\in S}\mathbb{P}[X_3=s|X_2=s_k]\cdot p^2_{s_0,s_k}\\ &=\sum_{s_k\in S}p_{s_k,s}\cdot p^2_{s_0,s_k}\\ \text{(by Equation 6)}\quad&=p^3_{s_0,s}.\end{aligned}$$

PROBABILITA' DI ANDARE DA s_0 AD s_K

Theorem 1.1. It holds that $\mathbb{P}[X_{n+1}=s|X_0=s_0]=p^{n+1}_{s_0,s}$, where $p^{n+1}_{i,j}$ is the generic (i,j) element of the power matrix P^{n+1}.

Proof. By induction. We have already seen that the thesis holds for $n=1$ and $n=2$. Assume that the thesis holds for n, i.e., $\mathbb{P}[X_n=s|X_0=s_0]=p^n_{s_0,s}$. It holds that:

$$\begin{aligned}\mathbb{P}[X_{n+1}=s|X_0=s_0]&=\sum_{s_k\in S}\mathbb{P}[X_{n+1}=s|X_n=s_k]\mathbb{P}[X_n=s_k|X_0=s_0]\\ \text{(induction hypothesis)}\quad&=\sum_{s_k\in S}\mathbb{P}[X_{n+1}=s|X_n=s_k]p^n_{s_0,s_k}\\ \text{(by homogeneity)}\quad&=\sum_{s_k\in S}p_{s_k,s}p^n_{s_0,s_k}=p^{n+1}_{s_0,s}\end{aligned}$$

Esempio: Se oggi piove (s'), la probabilità che tra 3 giorni ($m=3$) ci sia il sole (s) è $p^3_{s',s'}$ indipendentemente dal giorno della settimana.

□

Observation 1.1. *Theorem 1.1 holds in the following general case:* $\mathbb{P}[X_{n+m}=s|X_n=s']=p^m_{s',s}$

Although Markov's property greatly simplifies this computation, we still need to do several matrix multiplications. (**Question:** What is the time complexity for computing P^n?) Nevertheless, we can derive the **asymptotical behaviour** of the system. To do so, we first need to introduce some additional concepts.

Definition 1.4. A *stationary probability distribution* $\pi=(\pi_1,\ldots,\pi_{|S|})$ for a Markov Chain with transition matrix P is a probability distribution such that:

$$\pi=\pi P. \quad (7)$$

TUTTI GLI STATI COMUNICANO TRA DI LORO.

Definition 1.5. A Markov Chain is *irreducible* if it is always possible to transition from any state s to any state s' in a finite number of time steps. A Markov Chain is *reducible* if it has an *absorbing state* that can be reached from any state. An *absorbing state* s is such that $p_{s,s} = 1$, meaning that $p_{s,s'} = 0 \ \forall s' \neq s$.

Example 1.2. The Markov Chain described by the following transition graph is reducible, and state S is an absorbing state. Notice that in the transition graph, we have omitted the arcs with 0 weight.

S IN QUESTO CASO É UNO STATO ASSORBENTE DATO CHE NON HA ARCHI USCENTI

↓

LA CATENA DI MARKOV É RIDUCIBILE SE HA ALMENO 1 STATO ASSORBENTE → $p_{s,s}=1$ RAGGIUNGIBILE DA QUALSIASI ALTRO STATO.

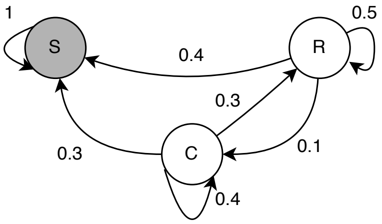
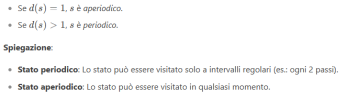


Figure 2: The transition graph of a reducible Markov Chain.

Definition 1.6. The *period* of a state s, $d(s)$, is defined as follows:



- Se $d(s) = 1$, s è *aperiodico.*
- Se $d(s) > 1$, s è *periodico.*

Spiegazione:

- **Stato periodico**: Lo stato può essere visitato solo a intervalli regolari (es.: ogni 2 passi).
- **Stato aperiodico**: Lo stato può essere visitato in qualsiasi momento.

gcd = MCD

$$d(s) = \gcd\{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}. \tag{8}$$

If $d(s) = 1$, then state s is *aperiodic*. If $d(s) > 1$, then state s is *periodic*. A Markov Chain is *periodic* if it has at least one periodic state.

Example 1.3. Consider the Markov Chain described by the following transition matrix:

La probabilità di 0.2 di rimanere nello stato 1 in un passaggio, essendo > 0, conferma che lo stato 1 è aperiodico.

$$P = \begin{array}{c c} & \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} 0.2 & 0.5 & 0.3 \\ 0.3 & 0.4 & 0.3 \\ 0.2 & 0.3 & 0.5 \end{pmatrix} \end{array}. \tag{9}$$

Is state 1 periodic? To answer this question, we first need to find the set of all natural numbers greater than 0 such that $p^n_{1,1} > 0$.
Notice that $1 \in \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$, since $p^1_{s,s} > 0$. Furthermore, gdc$\{1, n\} = 1 \ \forall n \in \mathbb{N}^+$, hence $d(s) = 1$, meaning that it is aperiodic.

Example 1.4. Consider the Markov Chain described by the following transition graph:

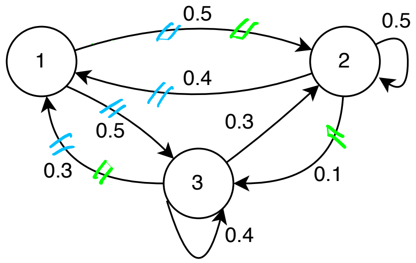


Figure 3: Transition graph

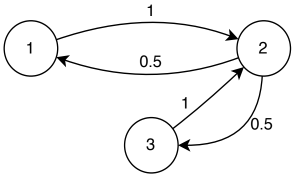
Is state 1 periodic? Notice that $1 \notin \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$, as $p_{1,1} = 0$. For $n = 2$, instead, there are two possible ways to reach state 1, starting from state 1: $X_0 = 1, X_1 = 2, X_2 = 1$ and $X_0 = 1, X_1 = 3, X_2 = 1$. Hence $2 \in \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$. It is easy to see that $3 \in \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$. Since gcd$\{2, 3, n\}$ =gcd$\{2, 3\} = 1 \ \forall n \in \mathbb{N}^+$, state 1 is aperiodic.

↓

MCD(2,3)=1 ↗

Example 1.5. Consider the Markov Chain described by the following transition matrix and graph:

$$P = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 \\ 0.5 & 0 & 0.5 \\ 0 & 1 & 0 \end{pmatrix} \end{array} \tag{10}$$



$1 \notin \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$

$2 \in$ ″

$3 \notin$ ″

$4 \in$ ″

MCD(4,2) = 2 $\Rightarrow$ STATE 1 IS PERIODIC.

Again, we wonder whether state 1 is periodic. Notice that $p_{1,1} = 0$, hence $1 \notin \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$. For $n = 2$, it holds that $p^2_{1,1} > 0$ (following the transition $X_0 = 1, X_2 = 2, X_3 = 1$), hence $2 \in \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$. Instead, $3 \notin \{n \in \mathbb{N}^+ : p^n_{s,s} > 0\}$. In fact, starting from state 1, we can possibly only go to state 2, from where we can either end up in 1 again or in 3. In both cases, there is no way that the next step is state 1. Notice that $p^n_{1,1} > 0$ if and only if $n = 2k$, $k \in \mathbb{N}, k > 0$, i.e., if and only if n is even. Then, $d(1) = 2$, and the Markov process is periodic.

Theorem 1.2 (The Fundamental Theorem of Markov Chains)**.** Let $X_0, X_1, \ldots$ be a Markov chain over a finite state space S, with transition matrix P. Suppose that the chain is irreducible and aperiodic. Then the following facts hold:

1. There exists a unique stationary distribution, $\pi = (\pi_1, \pi_2, \ldots)$ such that for any states i and j it holds that $\lim_{t\to\infty} \mathbb{P}[X_t = i | X_0 = j] = \pi_i$. Questo significa che, indipendentemente dallo stato iniziale j, la probabilità di trovarsi nello stato i converge a π_i per $t \to \infty$.
2. π is a left eigenvector of matrix P, with eigenvalue 1 ($\pi = \pi P$)

   - π è un **autovettore sinistro** di P con autovalore 1, cioè:

     $$\pi = \pi P.$$

   - In altre parole, π è invariante sotto l'azione della matrice P.

The theorem states that if we solve the linear system $\{P^T \pi^T = \pi^T, \sum_{i=1,\ldots,|S|} \pi_i = 1\}$, then we have a unique solution π that represents the long-term behaviour of the Markov process.

Example 1.6. Consider the Markov Chain described by the transition matrix of Example 1.3. It is easy to see that the process is irreducible and aperiodic. To find the stationary probability π, we solve the following linear system:

$$P = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 0.2 & 0.5 & 0.3 \\ 0.3 & 0.4 & 0.3 \\ 0.2 & 0.3 & 0.5 \end{pmatrix} \end{array}. \qquad \begin{cases} 0.2\pi_1 + 0.3\pi_2 + 0.2\pi_3 = \pi_1 \\ 0.5\pi_1 + 0.4\pi_2 + 0.3\pi_3 = \pi_2 \\ 0.3\pi_1 + 0.3\pi_2 + 0.5\pi_3 = \pi_3 \\ \pi_1 + \pi_2 + \pi_3 = 1 \end{cases} \tag{11}$$

which can be rewritten as:

$$\begin{cases} -0.8\pi_1 + 0.3\pi_2 + 0.2\pi_3 = 0 \\ 0.5\pi_1 - 0.6\pi_2 + 0.3\pi_3 = 0 \\ 0.3\pi_1 + 0.3\pi_2 - 0.5\pi_3 = 0 \\ \pi_1 + \pi_2 + \pi_3 = 1 \end{cases} \tag{12}$$

Notice that the linear system in this example has 4 equations and 3 variables, meaning that it is overdetermined. Nevertheless, we will always find a unique solution if the Markov Process is irreducible and aperiodic. In particular, the solution to this linear system is $\pi = (0.2386, 0.3864, 0.3750)$, which means that the $\sim 24\%$ of the time the system is in state s_1, the $\sim 38.5\%$ of the time the system is in state s_2, and the 37.5% of the time the system is in state s_3.

2 Markov Decision Processes

In a discrete-time Markov process, we just observe the system evolving over time following a probability distribution defined by the transition matrix P. In a Markov Decision Process (MDP), at each time step, being on a state s, we can **decide** to take an **action** that, with a certain probability, will take us to another state s' in the next time step. Our decision to take an action is driven by the achievement of a reward. Intuitively, a Markov decision process is a stochastic dynamic program for sequential decision-making with uncertain outcomes that satisfies the Markov's property.

Example 2.1. Consider an **environment** where a robot (i.e., the **agent**) moves through a room, which we can divide into square tiles. The robot can move only from one tile to an adjacent one, sharing one side, or it can stay in the same tile. Each tile represents a state of the system and is assigned a reward value. The robot can go up, down, left, right, or stay $\{U, D, L, R, S\}$. Unfortunately, the floor where the robot is moving is slippery, and sometimes, if it decides to take an action, for instance, to go up, it can slipper and end up in another tile. Consider the environment described in figure 4. There are nine states, all having zero reward except for state s_3, which is the tile where the robot has its recharging station, and for state s_6, where there is the elevator that takes the robot to another floor. Initially, the robot does not know anything about the environment and needs to explore it by moving through the room. From any state s, the robot takes an action with a certain probability. For instance, suppose that the robot is initially in state s_7 and moves up with probability 0.4 to reach s_4. With a 0.9 probability, it will indeed end up in state s_4, but with probability 0.1, it will slip and end up in state s_1.

SCARSA PROBABILITÀ DI SCIVOLARE ED ANDARE NELLO STATO s_1

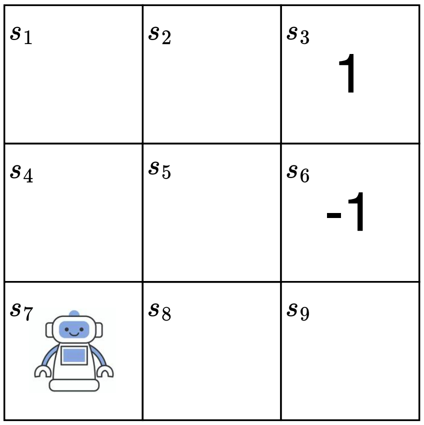


Figure 4: grid world

Definition 2.1. A Markov Decision Process (MDP) is a 4-tuple $(S, A, P, \mathcal{R})$, where:

- S is the set of all possible states. It can be finite, countable or continuous.
- A is the set of actions. We call A_s the set of actions that can be taken from state s. We call A_n the action taken at time n.
- $P^a_{s,s'} = \mathbb{P}[s'|s, a]$, the probability of transitioning from state s to state s' by taking action $a \in A$. Since we are talking about a **Markov** decision process, the probability of reaching a certain state in one step only depends on the current state, and not on previous ones, i.e., $\mathbb{P}[X_n = s'|X_{n-1} = s, \ldots, X_0 = s_0, A_n = a] = \mathbb{P}[X_n = s'|X_{n-1} = s, A_n = a]$.
- $\mathcal{R}$ is the expected reward function, i.e., the expected reward that we get if we are in a state a and make a decision a, $\mathcal{R}^a_s = \mathbb{E}[R_{n+1}|X_n = s, A_n = a]$. A reward R_s is associated with each state s.

A Markov Decision Process defines an environment. An agent is an entity that interacts with the environment and gets feedback from it. The goal of the agent is to find the best actions to take to maximize the achieved reward. Actions can be formally defined by a **policy** $\pi : A \times S \to [0, 1]$, $\boxed{\pi(a|s) = \mathbb{P}[A_t = a|S_t = s]}$. Fixed a state s, $\pi(a|s)$ is a probability distribution, i.e., $\boxed{\sum_{a\in A} \pi(a|s) = 1}$ $\forall s \in S$, $\pi(a|s) \geq 0 \, \forall a$. The policy completely defines the behaviour of an agent. A natural question is why we need the policy to be probabilistic, not deterministic. Consider the example 2.1. A good policy is the following (also depicted in Figure 5):

LA POLICY RESTITUISCE UNA PROBABILITÀ DI [0,1].

Probabilità che l'agente scelga l'azione a dato che si trova nello stato s al tempo t.

Le probabilità non possono essere negative

La somma delle probabilità di tutte le azioni possibili in s

- $\pi(R|s_1) = 1$, $\pi(a|s_1) = 0 \; \forall a \neq R$
- $\pi(R|s_2) = 1$, $\pi(a|s_2) = 0 \; \forall a \neq R$
- $\pi(S|s_3) = 1$, $\pi(a|s_3) = 0 \; \forall a \neq S$

Questo esempio rappresenta la policy deterministica in un processo decisionale di Markov dove ad ogni stato è associata un'azione specifica.

π(R|s₁)=1, π(a|s₁)=0 ∀a≠R
- (Nello stato s₁, l'agente sceglie sempre l'azione R, con probabilità 1)
- (tutte le altre azioni hanno probabilità 0)

- $\pi(U|s_4) = 1,\ \pi(a|s_4) = 0\ \forall a \neq U$
- $\pi(U|s_5) = 1,\ \pi(a|s_5) = 0\ \forall a \neq U$
- $\pi(U|s_6) = 1,\ \pi(a|s_6) = 0\ \forall a \neq U$
- $\pi(U|s_7) = 1,\ \pi(a|s_7) = 0\ \forall a \neq U$
- $\pi(U|s_8) = 1,\ \pi(a|s_8) = 0\ \forall a \neq U$
- $\pi(L|s_9) = 1,\ \pi(a|s_9) = 0\ \forall a \neq L$

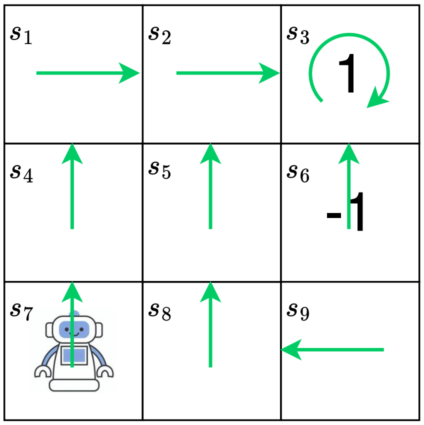


Figure 5: Policy example

Notice that this policy is deterministic, as we allow just one action in each state. Furthermore, this policy seems to be a good one because it drives the agent through the shortest paths to the maximal reward while avoiding the negative reward. Why would we need to lose the determinism and take an action with a positive probability that does not direct us through the best path to the reward? For instance, why would we choose a policy that could take the agent away from s_3, e.g., $\pi(S|s_3) = 0.5$, $\pi(L|s_3) = 0.5$, $\pi(a|s_3) = 0\ \forall a \neq S, L$? There are two main reasons to do that:

1. Exploration. As we observe Figure 4, we have a global view of the grid world, but the robot does not, and it has to move around and explore the environment before finding the best path to the reward.

2. Dynamism. Environments can be dynamic; for instance, the reward in state s_3 could change to -1 at a certain time step. The deterministic policy described above would get the agent stuck in s_3, losing its optimality.

In general, deterministic policies work well only when the environment is completely explored and is known to be static.

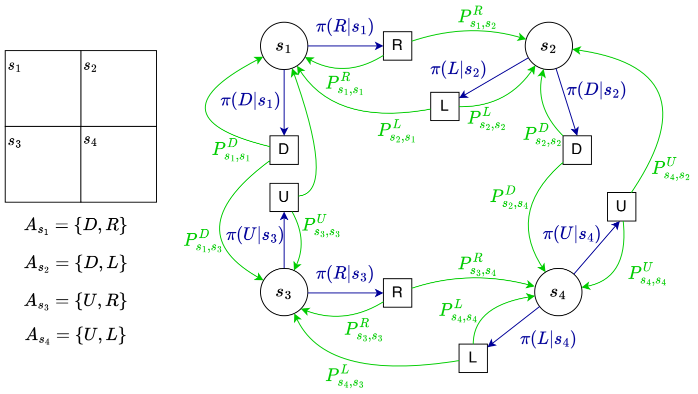
We can represent an MDP as a transition graph, similar to how we did for Markov chains. Nevertheless, the graph is much more complex since transitions have an additional layer of non-determinism (the set of possible actions). Consider the example of Figure 6. The environment is composed of four states $s_1, \dots, s_4$. The set of possible actions are Up, Down, Left, and Right. From each state, only adjacent states can be reached. Nevertheless, with some positive probability, when the agent decides to take a step towards a direction, it might get stuck in its current state. The transition graph and the transition matrix of the MDP are provided in Figure 6.

Definition 2.2. Given a policy π over a MDP $(S, A, P, \mathcal{R})$, we can define the *transition probabilities over the policy π to transit from state s to state s'* as:

$$P^{\pi}_{s,s'} = \sum_{a \in A_s} \pi(a|s) P^{a}_{s,s'}. \tag{13}$$

Similarly, we can define the *expected reward of being in state s under policy π* as:

$$\mathcal{R}^{\pi}_{s} = \mathbb{E}[R^{\pi}_{s}] = \sum_{a \in A_s} \pi(a|s) \mathcal{R}^{a}_{s}. \tag{14}$$



$$P = \begin{bmatrix}
p^U_{1,1} & p^U_{1,2} & p^U_{1,3} & p^U_{1,4} \\
p^D_{1,1} & p^D_{1,2} & p^D_{1,3} & p^D_{1,4} \\
p^L_{1,1} & p^L_{1,2} & p^L_{1,3} & p^L_{1,4} \\
p^R_{1,1} & p^R_{1,2} & p^R_{1,3} & p^R_{1,4} \\
p^U_{2,1} & p^U_{2,2} & p^U_{2,3} & p^U_{2,4} \\
p^D_{2,1} & p^D_{2,2} & p^D_{2,3} & p^D_{2,4} \\
p^L_{2,1} & p^L_{2,2} & p^L_{2,3} & p^L_{2,4} \\
p^R_{2,1} & p^R_{2,2} & p^R_{2,3} & p^R_{2,4} \\
p^U_{3,1} & p^U_{3,2} & p^U_{3,3} & p^U_{3,4} \\
p^D_{3,1} & p^D_{3,2} & p^D_{3,3} & p^D_{3,4} \\
p^L_{3,1} & p^L_{3,2} & p^L_{3,3} & p^L_{3,4} \\
p^R_{3,1} & p^R_{3,2} & p^R_{3,3} & p^R_{3,4} \\
p^U_{4,1} & p^U_{4,2} & p^U_{4,3} & p^U_{4,4} \\
p^D_{4,1} & p^D_{4,2} & p^D_{4,3} & p^D_{4,4} \\
p^L_{4,1} & p^L_{4,2} & p^L_{4,3} & p^L_{4,4} \\
p^R_{4,1} & p^R_{4,2} & p^R_{4,3} & p^R_{4,4}
\end{bmatrix}$$

Figure 6: Transition graph and transition matrix of a simple MDP.

What is the idea? The Markov Decision Process defines an environment. What we want to do is to *move* through this environment to maximize the reward by defining a policy. This means that we want to maximize the following function:

$$\sum_{t=0}^{T} R_t. \tag{15}$$

Where T is the experimental period. However, sometimes we are interested in infinite horizons, either because our system is inherently infinite or because we do not know T in advance. Therefore, we want to maximize the long-term rewards, i.e., the following quantity:

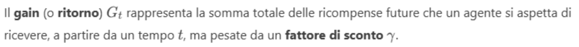
$$\sum_{t=0}^{\infty} R_t. \tag{16}$$

There is a problem in this formulation: any two policies that guarantee a positive reward R_t an infinite number of times are equivalent, i.e, they have the same reward, that is infinite! Consider the grid world example where one cell has reward 1 and another cell has reward 100, while all the other cells have reward 0. A policy that takes the agent to the cell with reward 1 and makes the agent stay there as much as possible will have $R_t = 1$ most of the times. If we simulate this environment in a infinite horizon, the sum of the reward is going to be infinite. Consider another policy that takes the agent to the cell with reward 100 and makes it stay there most of the time. Similarly, in a infinite horizon, the total reward is going to be infinite. Nevertheless, our intuition tells us that the second policy is better than the first one.

Definition 2.3. We call the *gain* or the *return* at time t the quantity:

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{17}$$

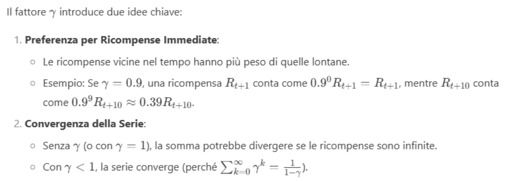
where $\gamma \in (0, 1]$ is the *discount factor*.


Il **gain** (o **ritorno**) G_t rappresenta la somma totale delle ricompense future che un agente si aspetta di ricevere, a partire da un tempo t, ma pesate da un **fattore di sconto** γ.

The discount factor is a discounting term that gives more weight to the reward of the first steps and less and less weight to the rewards of the time steps that are far in the future. By discounting the gain stepwise, we add some "greediness" to the computation.

Observation 2.1. *The gain has a recursive formulation:*

$$\begin{aligned} G_t &= R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \dots \\ &= R_{t+1} + \gamma G_{t+1} \end{aligned} \tag{18}$$


Il fattore γ introduce due idee chiave:

1. **Preferenza per Ricompense Immediate**:
   - Le ricompense vicine nel tempo hanno più peso di quelle lontane.
   - Esempio: Se $\gamma = 0.9$, una ricompensa R_{t+1} conta come $0.9^0 R_{t+1} = R_{t+1}$, mentre R_{t+10} conta come $0.9^9 R_{t+10} \approx 0.39 R_{t+10}$.
2. **Convergenza della Serie**:
   - Senza γ (o con $\gamma = 1$), la somma potrebbe divergere se le ricompense sono infinite.
   - Con $\gamma < 1$, la serie converge (perché $\sum_{k=0}^{\infty} \gamma^k = \frac{1}{1-\gamma}$).

Definition 2.4. The *state value function* $V_\pi(s)$ *of an MDP following policy* π is the expected return starting from state s and following π:

Misura quanto l'agente può accumulare in ricompense future se parte da s ed segue π (policy)

$$V_\pi = \mathbb{E}_\pi[G_t|S_t = s]. \tag{19}$$

Given a policy π, the state value function provides a measure of how good it is to be in state s.

Definition 2.5. The *action value function* $Q_\pi(s, a)$ *of an MDP following policy* π is the expected return starting from state s, taking an action s, and following π:

La **funzione di valore di azione** $Q_\pi(s, a)$ rappresenta il **ritorno atteso** partendo da s, prendendo l'azione a, e poi seguendo π:

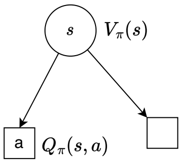
$$Q_\pi = \mathbb{E}_\pi[G_t|S_t = s, A_t = a]. \tag{20}$$

Given a policy π, the action value function says how good it is to take a particular action a from that s. The term $\mathbb{E}_\pi$ means that we are taking the expected value by sampling from policy π.

Observation 2.2. *Based on Observation 2.1, it holds that:*

$$V_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma V_\pi(S_{t+1}|S_t = s)] \tag{21}$$
$$Q_\pi(s, a) = \mathbb{E}_\pi[R_{t+1} + \gamma Q_\pi(S_{t+1}, A_{t+1}|S_t = s, A_t = a)] \tag{22}$$

Equations 21 and 22 are useful as they allow us to write the state value and the action value functions in terms of one another. Let's zoom in on the transition graph to state s, as in Figure 7.



Questo mostra come lo stato si suddivide in possibili azioni che si possono intraprendere.
Il valore dello stato Vπ(s) è determinato considerando il valore di tutte le azioni disponibili da quello stato, ponderato per la probabilità di scegliere ogni azione secondo la politica π:

Figure 7: Transition graph, from s to a.

As we are in state s, and we want to evaluate $V_\pi(s)$, we need to average all the possible action value functions for all possible actions that we can take from s. Formally:

$$V_\pi(s) = \sum_{a \in A} \pi(a|s) Q_\pi(s, a), \tag{23}$$

where, by definition, $V_\pi(s)$ is the expected return at state s, $\pi(a|s)$ is the (normalized) average number of the times the agent takes action a from state s (defined by the policy π), and, for each possible action a, $Q_\pi(s, a)$ is the expected return that the agent gets from taking action a from state s and following π. Analogously, we can write Q as a function of V. Consider the zoomed-in graph in Figure 8.

Il grafo mostra il processo decisionale di un agente che, trovandosi in uno stato s, esegue un'aione a per ottenere una ricompensa R e muoversi verso uno stato successivo.

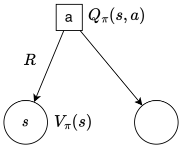


Figure 8: Transition graph, from a to s.

By taking a certain action a at a state s, we can find $Q_\pi(s, a)$ by computing the expected immediate reward of the action and adding the discounted expected reward from all possible states s' the agent can reach by taking action a. Formally:

$$Q_\pi(s, a) = \mathcal{R}_s^a + \gamma \sum_{s' \in S} P_{s,s'}^a V_\pi(s'). \tag{24}$$

sommatoria: per ogni stato che raggiungo eseguendo l'azione a.

If we put it all together, we can **define the state value and the action value functions recursively** (Figure 9), as follows:

$$V_\pi(s) = \sum_{a \in A} \pi(a|s)[\mathcal{R}_s^a + \gamma \sum_{s' \in S} P_{s,s'}^a V_\pi(s')] \tag{25}$$

il valore di uno stato dipende dalle azioni che possono essere eseguite in esso e dal valore che queste azioni generano.

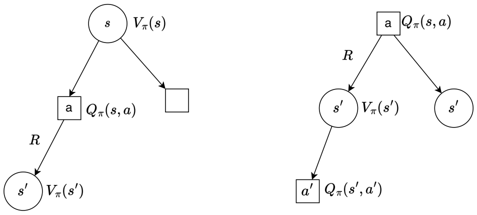


Figure 9: Graphical explanation of the Bellman expectation Equations

and:

$$Q_\pi(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in S} P_{s,s'}^a \sum_{a' \in A} \pi(a'|s') Q_\pi(s',a'). \tag{26}$$

These equations are known as the **Bellman Expectation Equations**, which are particularly useful as they provide a closed-form solution to find $V_\pi(s)$ and $Q_\pi(s,a)$. In fact, we can write $V_\pi(s)$ in matrix form as follows:

$$\begin{aligned}
V_\pi &= R^\pi + \gamma P^\pi V_\pi \\
I V_\pi &= I R^\pi + \gamma P^\pi V_\pi \\
(I - \gamma P^\pi) V_\pi &= R^\pi \\
V_\pi &= (I - \gamma P^\pi)^{-1} R^\pi
\end{aligned}$$

If $(I - \gamma P^\pi)$ is invertible. V_π is a $|S| \times 1$ vector with elements $(V_\pi(s_1), \ldots, V_\pi(s_{|S|})$. R^π is a $|S| \times 1$ vector with elements $(\mathcal{R}_{s_1}^\pi, \ldots, \mathcal{R}_{s_{|S|}}^\pi)$. P^π is a $|S| \times |S|$ matrix whose generic element is defined in Equation 13. The Equation $V_\pi = (I - \gamma P^\pi)^{-1} R^\pi$ can still be hard to solve. We can define the matrix Q_π with a similar approach (try to do it).

Recall that our goal is to find the policy that maximizes the expected reward. Such a policy is the **optimal policy**, usually denoted with π^*.

Definition 2.6. The optimal state value is the maximum value for a state s of $V(s)$, i.e., $V^*(s) = \max_\pi V_\pi(s)$. The optimal action value is the maximum value for a state s and an action a of $Q(s,a)$, i.e., $Q^*(s,a) = \max_\pi Q_\pi(s,a)$.

We say that a policy π is better than a policy π' if $V_\pi(s) \geq V_{\pi'}(s)$ $\forall s$. This relation defines a partial ordering of the policies; in this case, $\pi \geq \pi'$. In Markov decision processes, an optimal policy π^* always exists, and a policy π^* is optimal if $\pi \leq \pi^*$ $\forall \pi$. Furthermore, all optimal policies have the same maximum state value and action value. Notice that knowing Q^* means knowing the optimal policy. So, in order to find the optimal policy, it is enough to define it to deterministically choose always the optimal action:

$$\pi(a|s) = \begin{cases} 1 & \text{if } a = \arg\max_{a \in A} Q^*(s,a) \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

FACT: Such a policy is optimal, and a **deterministic optimal solution always exists!** With a similar approach used for defining the Bellman Expectation Equations, we can define the **Bellman Optimality Equations** as follows:

$$V^*(s) = \max_a \mathcal{R}_s^a + \gamma \sum_{s' \in S} P_{s,s'}^a V^*(s'), \tag{28}$$

$$Q^*(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in S} P_{s,s'}^a \max_{a'} Q^*(s',a'). \tag{29}$$

How do we solve these, and in particular Equation 29, in practice to get the optimal policy π^*? We were "kinda" able to solve the Bellman Expectation Equations, but the Bellman Optimality Equations are not linear, as there is a max function. This means that it is very difficult to optimize them, and there is no closed form in general. Common ways to get around this problem are to use dynamic programming and Q-learning.

3 Exercises

Exercise 3.1. Consider the Markov Chains described by the following matrices P_1 and P_2:

$$P_1 = \begin{pmatrix} 0 & 0.5 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0.5 \\ 0.5 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \qquad P_2 = \begin{pmatrix} 0 & 0.4 & 0.4 & 0.2 \\ 0.5 & 0 & 0 & 0.5 \\ 0.5 & 0 & 0 & 0.5 \\ 0.1 & 0.4 & 0.5 & 0 \end{pmatrix}$$

1. Draw the transition graphs.

2. Are they periodic? If yes, what is the period of their states?

Exercise 3.2. Consider the Markov Chain described by the following transition matrix:

$$P = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

1. Are there absorbing states? If so, what are they? Is it periodic? Why?

Exercise 3.3. Consider the Markov Chain described by the following transition matrix:

$$P = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

1. Are there absorbing states? If so, what are they? Is it periodic? Why?

Exercise 3.4. Consider the Markov Chain described by the following transition matrix:

$$P = \begin{pmatrix} 0.4 & 0.5 & 0.1 \\ 0.2 & 0.2 & 0.6 \\ 0.3 & 0.3 & 0.4 \end{pmatrix}$$

1. Does a unique stationary distribution exist? Why? If a stationary distribution exists, find it.