

MRP is a tuple  $(S, P, R, \gamma)$ , where  $S$  is the set of states  $P$  is 2-dimensional transition matrix  $R$  is a reward function defined as  $R = E[r_{t+1} | S_t = s]$ , that is collected upon leaving a the state  $s$ , i.e. at time step  $t + 1$   $\gamma$  is the discount factor  
**## Return** - Total discounted reward at from time step  $t$

$R_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots = \sum_{k=0}^n \gamma^k R_{t+k+1}$  -  $\gamma$  close to 0 leads to short-sighted evaluation (i.e. don't look into the future too much) -  $\gamma$  close to 1 leads to far-sighted evaluation This return is specific for some sample of states. It does not take into account all possible paths that can be taken from a given state

## Value

The aggregated return for a given state can be thought of as its 'value'

$$v(s) = E[R_t | S_t = s]$$

## Bellman Equation for MRP

$$\begin{aligned} v(s) &= E[R_t | S_t = s] \quad \&= E[r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots | S_t = s] \quad \&= \\ &= E[r_{t+1} + \gamma(r_{t+2} + \gamma(r_{t+3} + \dots)) | S_t = s] \quad \&= E[r_{t+1} + \gamma E[r_{t+1} | S_t = s]] \quad \&= \\ &= E[r_{t+1} + \gamma E[R_{t+1} | S_t] | S_t = s] \quad \&= E[r_{t+1} + \gamma v(s_{t+1}) | S_t = s] \end{aligned}$$

The transition from (1) to (2) uses the Tower rule and in essence the conditional in the inner expected value will be averaged out/canceled out so it's value does not matter. Therefore it is more useful to use the value of the next state  $s_{t+1}$

See proof: <https://chat.openai.com/share/77adb96c-1d17-4fca-8d88-91792567c997>

## Different Notation

### Sum notation (written out)

$$v(s) = R_s + \gamma \sum_{s'} P_{ss'} v(s')$$

**### Vector form**

$$v = R + \gamma P v$$

This can be analytically solved:  $v = (I - \gamma P)^{-1} R$

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

**## Matrix inversion** however has a complexity of  $O(n^3)$  and therefore works only for small MRPs

## Policy ( $\pi$ )

MRP helps us observe the state space and figure out the value of each state. In order to actually influence the environment and take actions a policy  $\pi$  is introduced.

A policy could either be deterministic and non-deterministic.