Reinforcement Learning

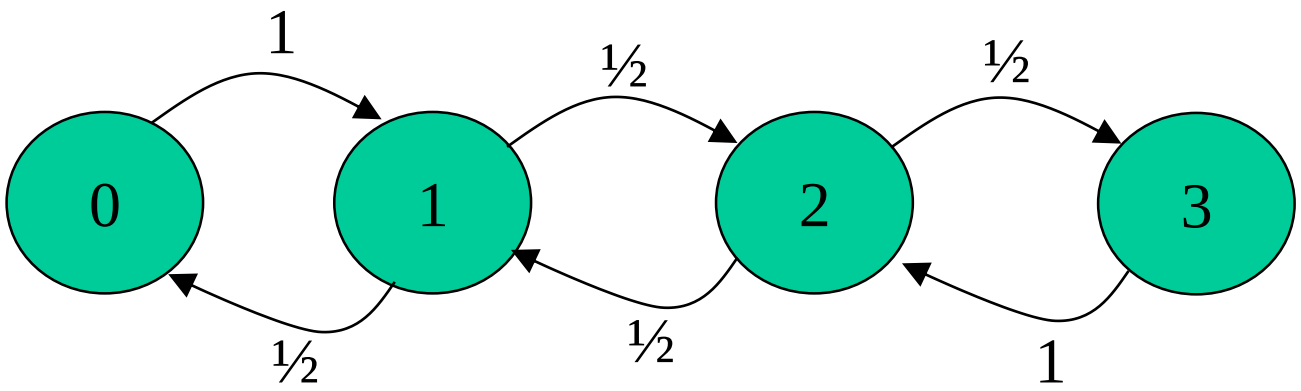
Lecture 3:
Markov Decision Processes

Yishay Mansour, Tel-Aviv University

Lecture 3: outline

- Markov Chain
  - Definition
  - Basic Properties
- Markov Decision Process
  - Definition
  - Return function
- Finite Horizon
  - Shortest paths
  - Dynamic Programing

Markov Chain



MC puzzle

- Consider the following gambling game:
  - You start with X_0 dollars
  - At time t
    - With prob ¼ you have $X_t = 4X_{t-1}$
    - With prob ¾ you have $X_t = X_{t-1}/4$
- What will happen in the long run?
  - Do you want to play the game?

Markov Chain: Overview

- Stochastic process:
  - $X_0, \dots, X_t$ are random variable
    - $\Pr[X_t = j | X_{t-1} = i, \dots, X_0]$
- Markov chain
    - $\Pr[X_t = j | X_{t-1} = i, \dots, X_0] = \Pr[X_t = j | X_{t-1} = i]$
  - Time homogeneous
    - $\Pr[X_t = j | X_{t-1} = i] = \Pr[X_1 = j | X_0 = i] = p_{i,j}$
    - Transition Matrix $P = (p_{i,j})$
      - $p_{i,j} \geq 0 \,; \sum_j p_{i,j} = 1 \; \forall i$

Markov Chain: Prob trajectory

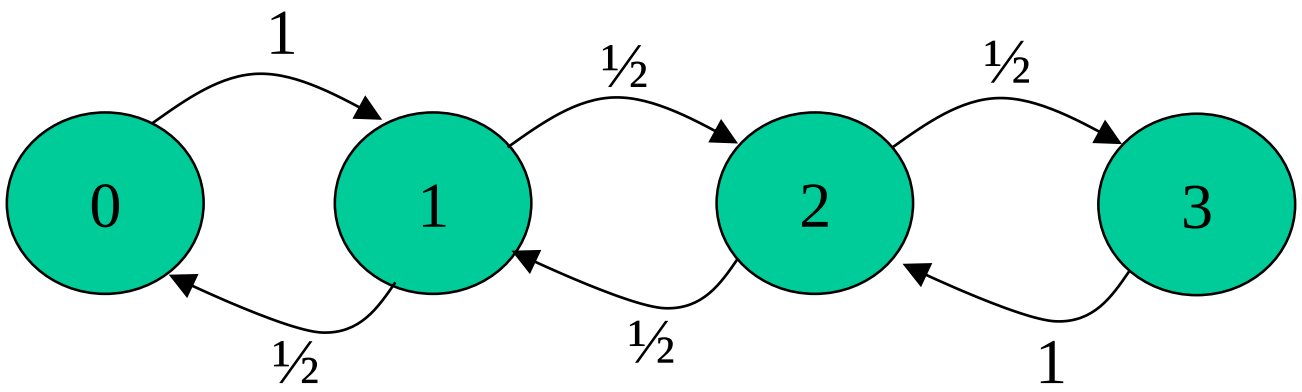
- Probability of trajectory
  - Given a distribution p_0 for X_0:
  - $\Pr[X_0 = i_0, \ldots, X_t = i_t] = p_0(i_0)p_{i_0,i_1} \cdots p_{i_{t-1},i_t}$
- Transition probability of m-step
  - $\Pr[X_m = j | X_0 = i] = p_{i,j}^{(m)}$
  - $p_{i,j}^{(m)} = [P^m]_{i,j}$
    - Proof by induction

MC: structural properties

- We like to define "reachability"
  - The ability to eventually move from i to j.
  - State j is **reachable** from i, $(i \to j)$
    - $p_{i,j}^{(m)} > 0$ for some $m \geq 1$
  - Transitivity
    - If $(i \to j)$ and $(j \to k)$ then $(i \to k)$

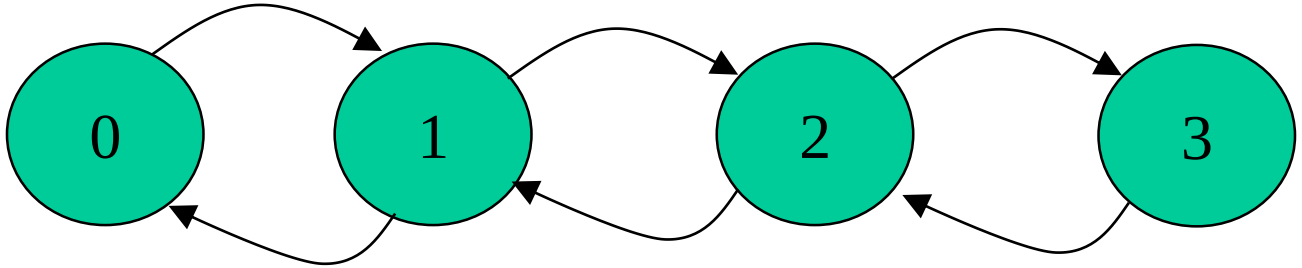
MC: structural properties

- Reachability ($0 \rightarrow 3$)



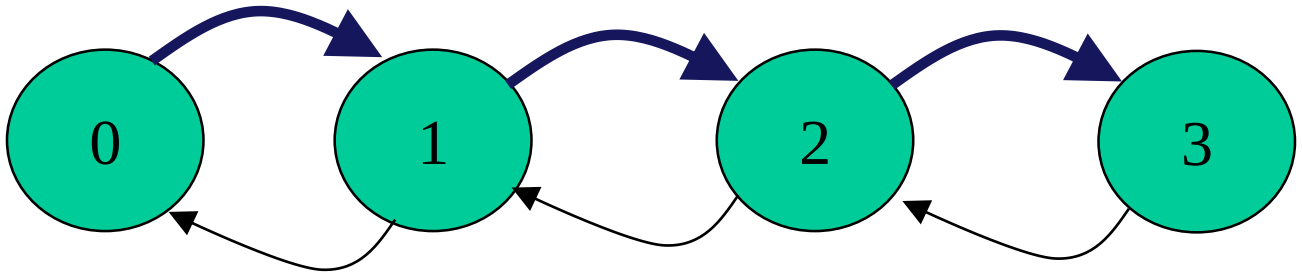
MC: structural properties

- Reachability ($0 \rightarrow 3$)
  - Drop probabilities
    - Assume they are non-zero
  - Test if 3 is reachable from 0



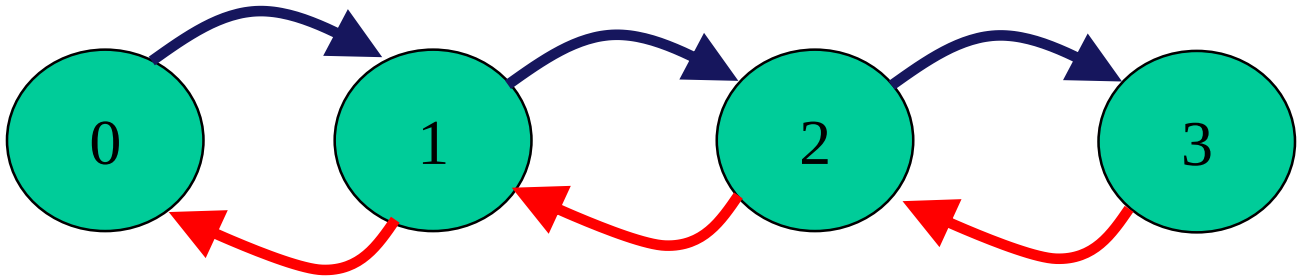
MC: structural properties

- Reachability ($0 \rightarrow 3$)
  - Drop probabilities
    - Assume they are non-zero
  - Test if 3 is reachable from 0



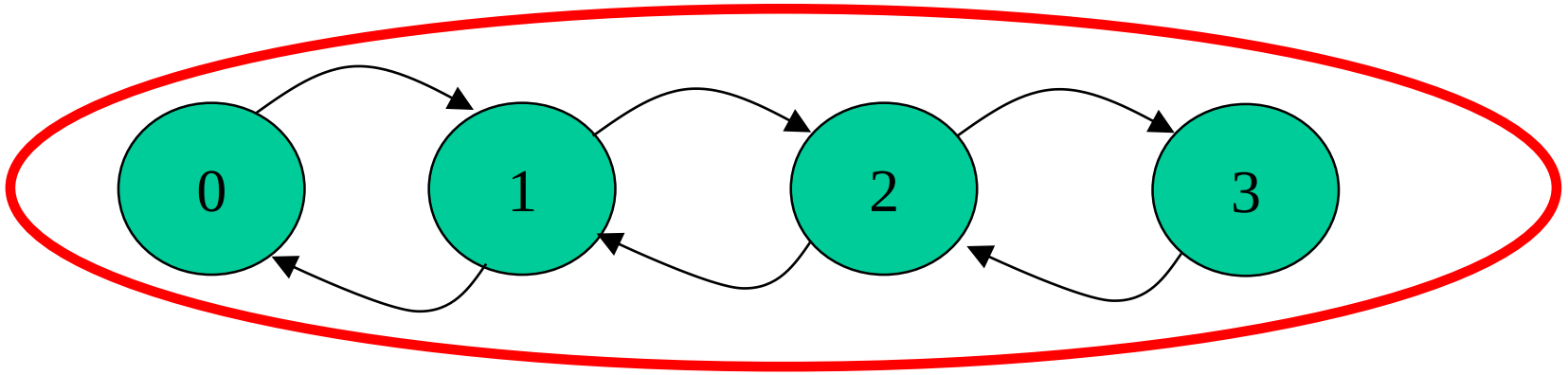
MC: structural properties

- states i and j are **communicating** if
  - $(i \rightarrow j)$ and $(j \rightarrow i)$
  - i is connected to j and j to i.



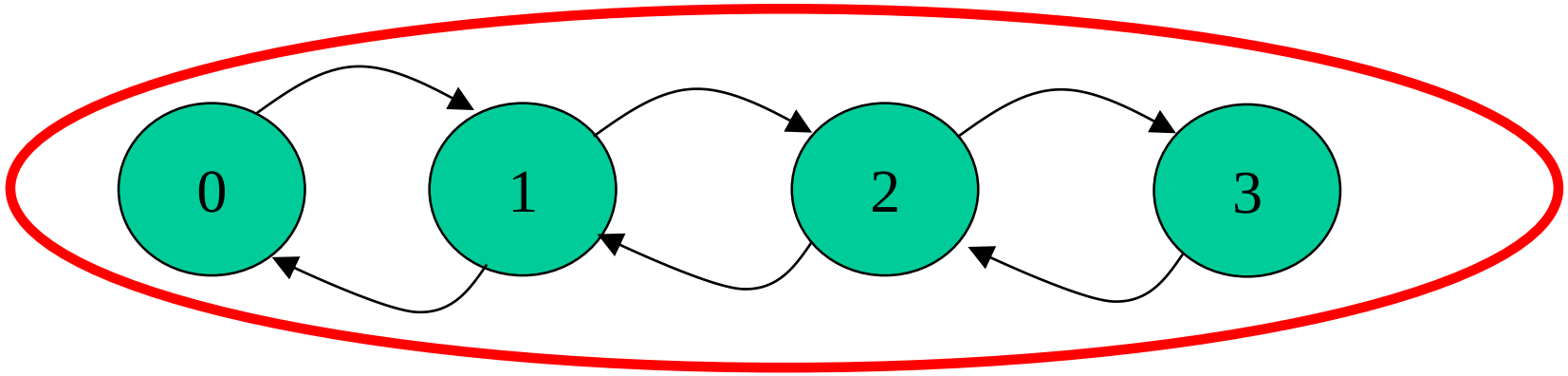
MC: structural properties

- **Communicating class** (or class)
  - Maximal set of states that are communicating
  - Strongly connected component



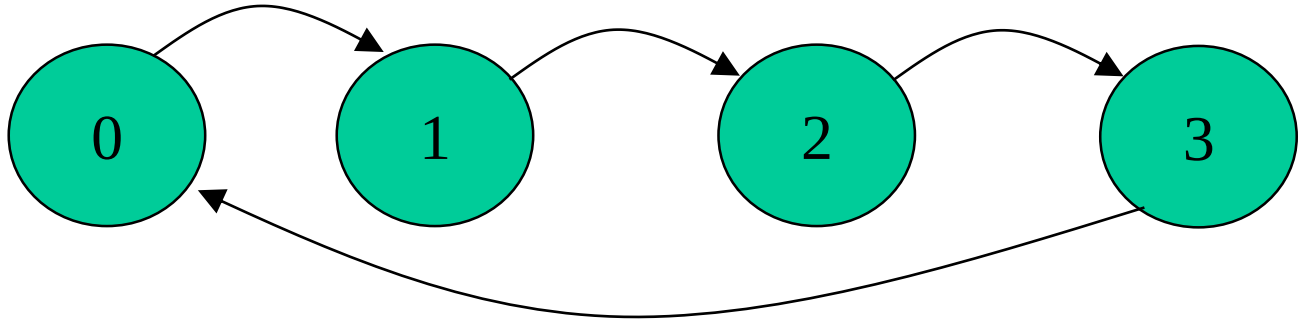
MC: structural properties

- Markov Chain is **irreducible** if all states are in a single communicating class
  - Strongly connected graph



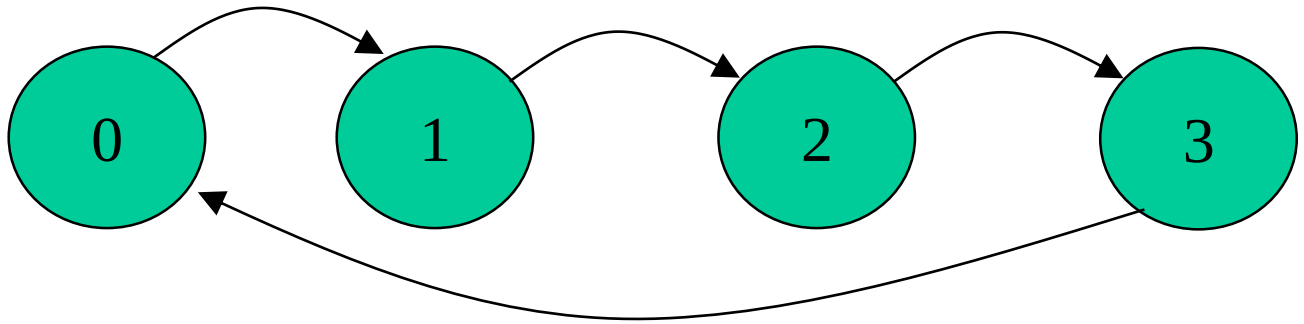
MC: periodicity

- State i has period d_i:
  - $d_i = \gcd\left\{m: p_{i,i}^{(m)} > 0\right\}$
    - Implication: If $X_t = i$ then at any time $t' > t$ s.t. $t' - t \bmod d_i \neq 0$ we have $X_{t'} \neq i$



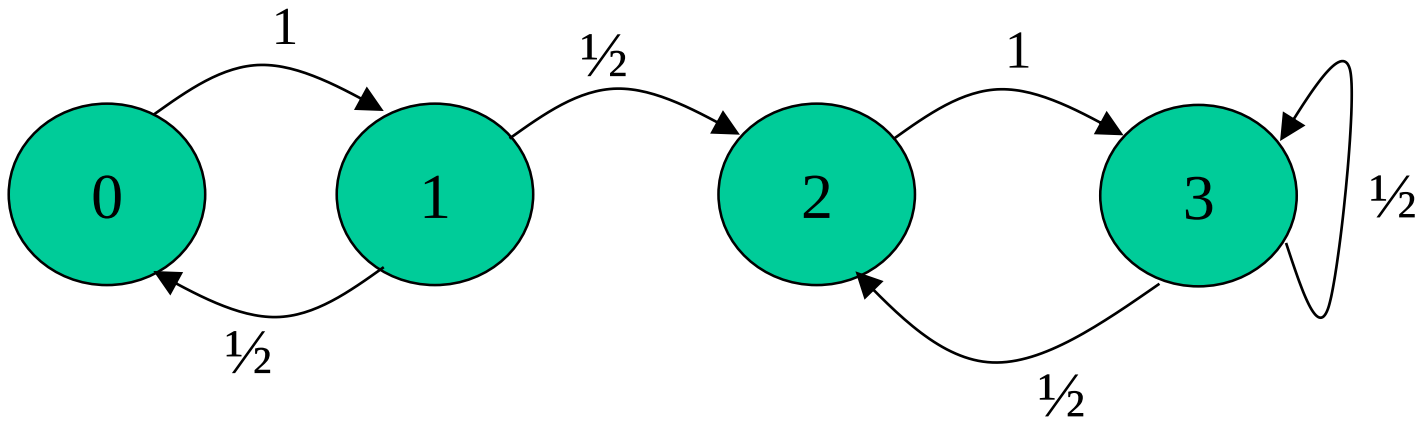
MC: aperiodic

- State i is aperiodic $if\ d_i = 1$
- A Markov Chain is **aperiodic** if every state is aperiodic



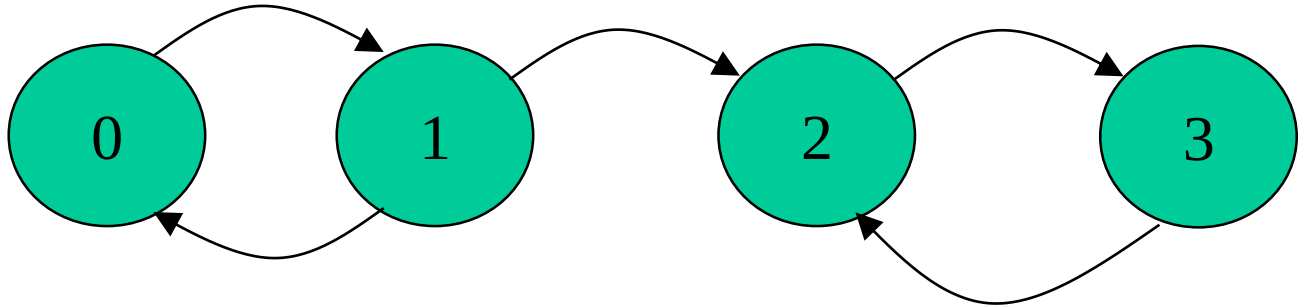
MC: recurrence

- State i is **recurrent** if
  - $\Pr[\exists t: X_t = i \mid X_0 = i] = 1$
  - Otherwise state i is **transient**



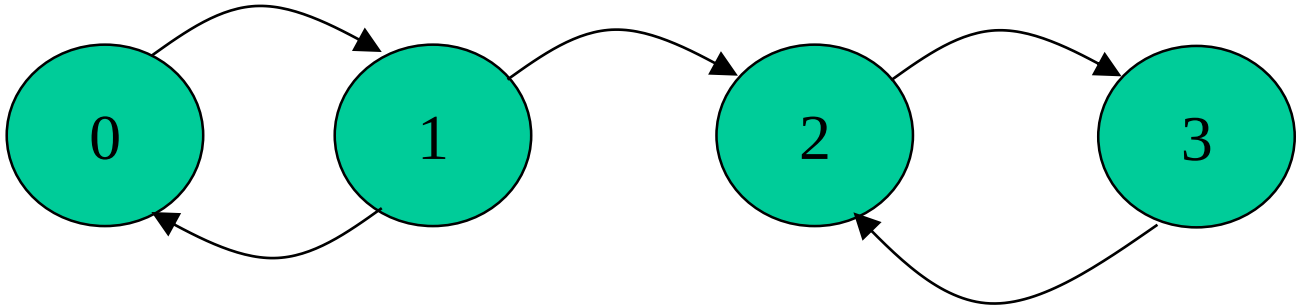
MC: recurrence

- For state i, with prob 1, if
  - **Recurrent:** it will appear infinitely often
  - **Transient:** it will appear only finite number
  - Equivalently, transient is $\sum_{m \geq 1} p_{i,i}^{(m)} < \infty$



MC: recurrence

- Recurrence in a class property
  - Either all states are recurrent or all transient

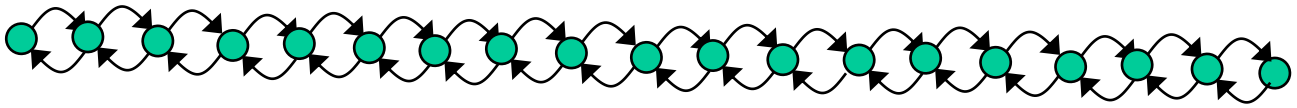


MC: recurrence and return time

- Return time:
  - T_i number of steps until we return to state i
- State i is **recurrent** iff $T_i < \infty$ w.p. 1
  - **Positive recurrent** $E[T_i] < \infty$
  - **Null recurrent** $E[T_i] = \infty$
    - We can have $T_i < \infty$ w.p. 1 and $E[T_i] = \infty$

MC: recurrence Examples

- Random walk: $X_t = X_{t-1} + Z, Z \in \{+1, -1\}$
  - Unbiased $\Pr[Z = +1] = \Pr[Z = -1] = \frac{1}{2}$
    - With prob 1 return to *origin*
    - Mean return time infinite
    - Null recurrent!



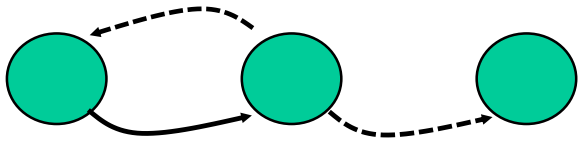
MC: recurrence Examples

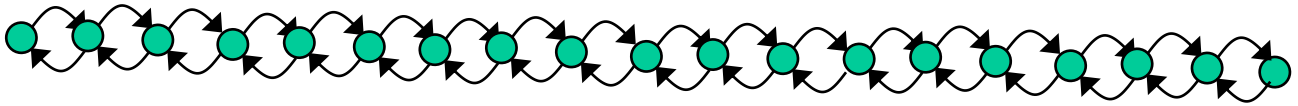
- Random walk:
  - $X_t = X_{t-1} + Z, Z \in \{+1, -1\}$
  - $\Pr[Z = +1] = \Pr[Z = -1] = \frac{1}{2}$
- Analysis:
  - Event T^1
    - $X_1 = X_0 + 1$
    - $X_{T_1} = X_0$

- $E[T] = 1 + E[T^1]$

$$= 1 + 1 + \frac{1}{2} 2E[T^1]$$

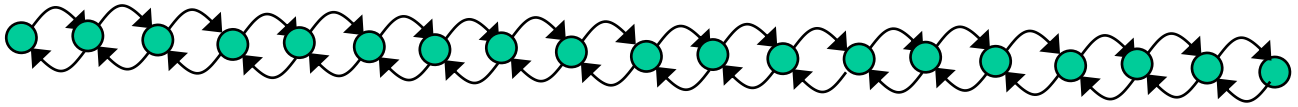
- $E[T^1] = \infty$





MC: recurrence Examples

- Random jumps:
  - With prob 1/3 $X_t = X_{t-1} + 1$
  - With prob 2/3 $X_t = 0$
    - Mean return time finite
    - Positive recurrent!
    - Shown in recitation



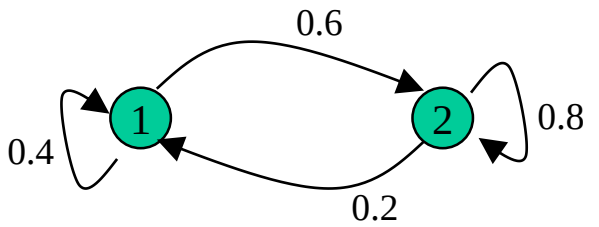
Lecture 3: outline

- Markov Chain
  - Definition
  - Basic Properties
- Markov Decision Process
  - Definition
  - Return function
- Finite Horizon
  - Shortest paths
  - Dynamic Programing

MC: Steady state distribution

- Steady state distribution:
  - $\mu^\top P = \mu$
  - $\mu_j = \sum_i \mu_i p_{i,j}$
  - If $X_t \sim \mu$ then $X_{t+1} \sim \mu$

- Example



- Matrix: $\begin{bmatrix} 0.4 & 0.6 \\ 0.2 & 0.8 \end{bmatrix}$

- Steady state: $[0.25 \;\; 0.75]$

MC: steady state

- P is row stochastic
- Eigenvalues:
  - $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
  - $Pz = z$
  - $z_1 = \vec{1}, \lambda_1 = 1$
  - $1 \geq \lambda_i$
    - $\|Pz\|_\infty \leq \|z\|_\infty$
  - Same eigenvalues for
    - $zP = z$

- Convergence to steady state
  - P^m eigenvalues
    $\lambda_1^m \geq \lambda_2^m \geq \cdots \geq \lambda_n^m \geq 0$
  - $x = \lambda_1 u_1 + \cdots + \lambda_n u_n$
  - $x^\top P^m = \sum_i \lambda_i^m u_i$
- Rate depends on
  - λ_2/λ_1

MC: Steady state distribution

- Theorem: For an irreducible aperiodic Markov Chain over finite state space the following holds
  - All states are *positive recurrent*
  - There exists a *unique stationary distribution* μ^*
  - *Convergence* to the stationary distribution in the limit
    - $\lim_{t\to\infty} p_{i,j}^{(t)} = \mu_j \quad \forall j, i$
  - *Ergodicity*: for any finite *f:*
    - $\lim_{t\to\infty} \sum_{s=0}^{t-1} \frac{1}{t} f(X_s) = \sum_i \mu_i f(i)$

MC: Steady state and return time

- Assume an irreducible aperiodic finite space Markov Chain
  - All states are positive recurrent
    - $E[T_i] = M_i < \infty$
    - For some state j, $E[T_j] \leq n$
  - Steady state distribution:
    - $\mu_i = \frac{1}{E[T_i]}$
      - – Intuitively, on average, state i appears every M_i steps.

MC: Countable state space

- Theorem: Let (X_t) be an irreducible aperiodic Markov chain over a countable space. Then **all** states are either:
  - Positive recurrent
  - Null recurrent
  - Transient

MC: Reversible

- Suppose exists μ s.t.
  - $\mu_i p_{ij} = \mu_j p_{ji} \quad \forall i, j \in X$
  - Then μ is a steady state
    - $\sum_i \mu_i \; p_{ij} \; = \sum_i \mu_j \; p_{ji} \; = \mu_j$
  - Called *detailed balance equations*

MC: Example Queue

- Dynamics:
  - $X_{t+1} = (X_t + A_t - S_t)^+$
  - $A_t \sim Br(p);\ \ S_t \sim Br(q)$
  - $\lambda = p(1-q); \eta = q(1-p); \rho = \lambda/\eta$
- Detailed balance equations
  - $\mu_i \lambda = \mu_{i+1} \eta$
- Solution with $\sum_i \mu_i = 1$, iff $\rho < 1$
  - $\mu_i = \mu_0 \rho^i;\ \mu_0 = 1 - \rho$

Lecture 3: outline

- Markov Chain
  - Definition
  - Basic Properties
- Markov Decision Process
  - Definition
  - Return function
- Finite Horizon
  - Shortest paths
  - Dynamic Programing

Controlled Markov Chains

- Adding actions (not rewards yet)
  - Finite state space S, where $S_t \subset S$
  - Finite action space A,
    - where $A_t(s) \subset A, s_t \in S_t$
  - Time horizon:
    - $\mathbb{T} = \{0, \dots, T-1\}$, finite horizon
    - $\mathbb{T} = \{0, 1, 2, \dots\}$, infinite horizon

Controlled Markov Chains

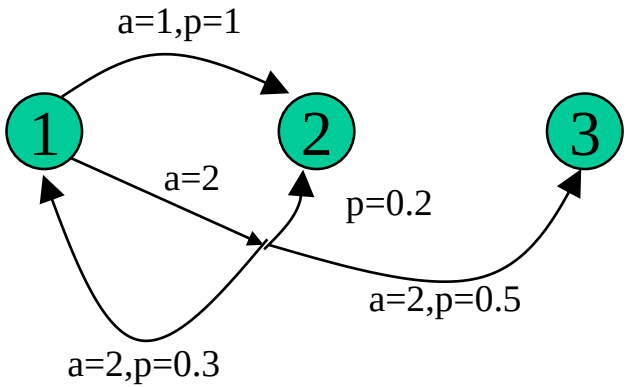
- State transition probability $p_t(\cdot\,|s,a)$:
  - $\Pr[s_{t+1} = s'|s_t = s, a_t = a] = p_t(s'|s,a)$
    - Clearly $p_t(s'|s,a) \geq 0$ & $\sum_{s' \in S_{t+1}} p_t(s'|s,a) = 1$
  - Markov property implicit!
- Stationary (time-invariant) models:
  - $S = S_t\,; A = A_t\,; p(\cdot\,|s,a) = p_t(\cdot\,|s,a) \quad \forall t$

CMC: Graphical notation

- Arrows labeled by
  - Action
  - Probability
- Example

$$p(s' = 2|s = 1, a = 1) = 1$$

$$p(s' = 1|s = 1, a = 2) = 0.3$$
$$p(s' = 2|s = 1, a = 2) = 0.2$$
$$p(s' = 3|s = 1, a = 2) = 0.5$$



Control Policies

- SD: Stationary Deterministic
  - $\pi: S \to A$
- MD: Markov Deterministic
  - $\pi: S \times T \to A$
- HD: History Deterministic
  - $\pi: \mathbb{H} \to A$
- SS: Stationary Stochastic
  - $\pi: S \to \Delta(A)$
- MS: Markov Stochastic
  - $\pi: S \times T \to \Delta(A)$
- HS: History Stochastic
  - $\pi: \mathbb{H} \to \Delta(A)$

Induced Stochastic Process

- Given:
  - Initial State Distribution $p_0(s_0)$
  - Policy $HS\ \pi$
- Induces distribution over trajectories
  - $h_T = (s_0, a_0, \ldots, s_{T-1}, a_{T-1}, s_T)$
- Probability of a trajectory:
  - $\Pr[h_T] = p_0(s_0) \prod_{t=0}^{T-1} p_t(s_{t+1}|s_t, a_t)\pi_t(a_t|h_t)$

Induced Stochastic Process

- Derivation
- Trajectory
  - $h_t = (s_0, a_0, \ldots, s_{t-1}, a_{t-1}, s_t)$
- Probability
  - $\Pr[h_{t+1}] = \Pr[h_t, a_t, s_{t+1}]$
    $= \Pr[s_{t+1}|h_t, a_t] \Pr[a_t|h_t] \Pr[h_t]$
    $= p_t(s_{t+1}|s_t, a_t)\pi_t(a_t|h_t) \Pr[h_t]$

Markov Control Policy

- Markov policy induces a Markov Chain:
  - $\Pr[s_{t+1} = s' | s_t = s] = \sum_{a \in A} p_t(s'|s,a)\pi_t(a|s)$

- Derivation:

$$\Pr[s_{t+1} = s' | s_t = s] = E_{h_t} \Pr[s_{t+1} = s' | s_t = s, h_t]$$

$$= E_{h_t} \sum_{a \in A_t} \Pr[s_{t+1} = s' | s_t = s, h_t, a]\pi_t(a|s)$$

$$= \sum_{a \in A} p_t(s'|s,a)\pi_t(a|s)$$

Remarks

- For most non-learning optimization problems, Markov policy are sufficient
  - Previous lecture: finite horizon DDP
- Full Observability is implicit
  - Partial Observation is the alternative
    - Subject of a future lecture

Performance Criteria: Finite Horizon

- Rewards:
  - $E[R_t] = r_t(s_t, a_t)\ \forall t \leq T-1$
  - $E[R_T] = r_T(s_T)$
- Finite Horizon:
  - $E[\sum_{t=0}^{T} R_t] = \sum_{t=0}^{T-1} r_t(s_t, a_t) + r_T(s_T)$
- Expected return
  - $J_T^{\pi}(s) = E^{\pi}[\sum_{t=0}^{T} R_t | s_0 = s] = E^{\pi,s}[\sum_{t=0}^{T} R_t]$

Remarks

- Extended rewards: $\mathrm{E}[R_t] = \tilde{r}_t\,(s_t, a_t, s_{t+1})$
  - $r_t(s,a) = E[R_t | s_t = s, a_t = a]$
  - Same expected rewards!
- Expectation sufficient for planning
  - $r_t(s,a) = E[R_t | s_t = s, a_t = a]$
- Risk sensitive criteria:
  - Involve variance
  - $\frac{1}{\lambda} \log E\left[e^{\lambda \cdot return}\right]$

Performance Criteria: Infinite Horizon

- Discounted return
  - Discount factor $0 < \gamma < 1$
  - Return:

$$V_\gamma^\pi(s) = E^\pi[\sum_{t=0}^{\infty} \gamma^t \, r(s_t, a_t) \,|s_0 = s]$$

- If $r(s,a) \in [0,1]$ then
  - $0 \leq V_\gamma^\pi(s) \leq \frac{1}{1-\gamma}$
  - All reward at $t \geq \log_\gamma \varepsilon = \frac{\log 1/\epsilon}{\log 1/\gamma}$ contribute at most $\frac{\varepsilon}{1-\gamma}$

Performance Criteria: Infinite Horizon

- Average return
  - $V_{av}^{\pi}(s) = \lim_{T \to \infty} \inf E^{\pi}\left[\frac{1}{T}\sum_{t=0}^{T-1} r(s_t, a_t) | s_0 = s\right]$
  - Very weak dependence on start state
    - Assuming irreducibility

Performance Criteria: Stochastic Shortest Path

- Set of goal states $S_G \subset S$
- Terminates when goal state reached
  - $\tau = \inf \{t \geq 0 \,|s_t \in S_G\}$
- Expected cost
  - $J^{\pi}_{ssp}(s) = E^{\pi,s}[\sum_{t=0}^{\tau-1} r(s_t, a_t) + r_G(s_\tau)]$
- Need to assume that termination happens with probability 1.

Sufficiency of Markov Policies

- In all performance criteria (shown)
  - Return is linear in $E[r_t(s_t, a_t)]$
  - If policies π and π' induce same marginal
    - $E[r_t(s_t, a_t)]$
  - Then they have the same (linear) return
- Previous Lecture (DDP):
  - History-dependent to Markov
  - Stochastic to Deterministic

Lecture 3: outline

- Markov Chain
  - Definition
  - Basic Properties
- Markov Decision Process
  - Definition
  - Return function
- Finite Horizon
  - Shortest paths
  - Dynamic Programing

Finite Horizon: Dynamic Prog

- Finite Horizon return
  - $V_T^{\pi}(s) = E^{\pi, s_0}[\sum_{t=0}^{T-1} r_t(s_t, a_t) + r_T(s_T)]$

- Optimal policy
  - $V_T^*(s_0) = V_T^{\pi^*}(s_0) = \max_{\pi \in HS} V_T^{\pi}(s_0)$

Finite Horizon: Dynamic Programming

- Principle of optimality
  - **The tail of an optimal policy is optimal for the "tail" problem**
    - *Does not hold for "prefix" !*
- Need to be re-applied for each setting
  - Finite Horizon:
    - For any state s' such that $\Pr[s_t = s'] > 0$
    - $\pi^*_{t:T} = (\pi^*_t, \ldots, \pi^*_T)$ is optimal for $[t, T]$ starting at s'
      - Namely, $V^{\pi^*}_{T-t}(s')$ is optimal.

Dynamic Prog: Policy Evaluation

- Fix a policy $\pi = (\pi_0, \ldots, \pi_{T-1})$ in MD
  - $V_k^\pi(s) = E^\pi[\sum_{t=k}^{T} R_t \mid s_k = s]$
    - $V_0^\pi(s_0) \triangleq J_T^\pi(s_0)$
- Lemma (value iteration)
  - For $k = T-1, \ldots, 0$, set $V_T^\pi(s) = r_T(s)$
  - $V_k^\pi(s) = r_k(s,a) + \sum_{s' \in S_{k+1}} p_k(s'|s,a) V_{k+1}^\pi(s')$
    - Where $a = \pi_k(s)$ and $s \in S_k$

- Proof: by backward induction on k.
  - Basis $k=T$, hold by the initialization.
  - Assume it holds for $k+1$ show for k:
  - $V_k^\pi(s) = E^\pi\left(R_k + \sum_{t=k+1}^{T} R_t \,|s_k = s, a_k = \pi_k(s)\right)$

$$= E^\pi\,(R_k|\, s_k = s, a_k = \pi_k(s)) +$$

$$E^\pi[E^\pi\left(\sum_{t=k+1}^{T} R_t \,\middle|s_k = s, s_{k+1}\right)]$$

$$= r_k\left(s, \pi_k(s)\right) + E^\pi[V_{k+1}^\pi(s_{k+1})|s_{k+1}]$$

$$= r_k(s,a) + \sum_{s' \in S_{k+1}} p_k(s'|s,a) V_{k+1}^\pi(s')$$

  - Where $a = \pi_k(s)$

- QED

Dynamic Programming : OPT Policy

- Optimal value function for $k \geq 0$
  - $V_k(s) = \max_{\pi^k} E^{\pi^k}[\sum_{t=k}^{T} R_t \mid s_k = s]$ for $s \in S_k$
    - where $\pi^k = (\pi_k, \ldots, \pi_{T-1})$
  - By definition $V_0(s_0) = J^*(s_0)$
  - Note that π^k is an arbitrary strategy for $[k, T]$

Finite Horizon Dynamic Programming

- Theorem: the following holds:
- Backward recursion
  - Set $V_T(s) = r_T(s)$ for $s \in S_T$
  - For $k = T - 1, \ldots, 0$
    - $V_k(s) = \max_{a \in A_k} (r_k(s,a) + \sum_{s' \in S_{k+1}} p_k(s'|s,a) V_{k+1}(s'))$
      - where $s \in S_k$

Finite Horizon Dynamic Programming

- Theorem (continue)
- Optimal policy:
  - Any policy π^* that satisfies

$$\pi_t^*(s) = \arg\max_{a \in A_t} \left(r_t(s,a) + \sum_{s' \in S_{t+1}} p_t(s'|s,a) V_{t+1}^{\pi}(s') \right)$$

    - where $s \in S_t$
  - Furthermore, π^* maximizes $J^{\pi}(s_0)$ for any $s_0 \in S_0$

- Proof: part *(i) computing* V_t
- Backward induction (again)
  - Base: $t = T$, follows from the initialization
  - Induction step
    - Suppose $V_{k+1}(s)$ is the optimal value function
    - Show that $V_k(s) = W_k(s)$ where
      - $W_k(s) = \max_{a \in A_k}(r_k(s,a) + \sum_{s' \in S_{k+1}} p_k(s'|s,a) V_{k+1}(s'))$
    - First show $V_k(s) \geq W_k(s)$ then $V_k(s) \leq W_k(s)$

- Show $V_k(s) \geq W_k(s)$
  - Find a policy π^k s.t. $V_k^{\pi^k}(s) = W_k(s)$
  - For state $s \in S_k$ select action $\bar{a} = a_k$ in

    $\bar{a} \in \arg\max_{a \in A_k}(r_k(s,a) + \sum_{s' \in S_{k+1}} p_k(s'|s,a) V_{k+1}(s'))$
  - After observing s_{k+1} continue with $\pi^{*,k+1}$
    - Guarantees $V_{k+1}^{\pi^{*,k+1}}(s) = V_{k+1}(s)$
  - This gives:

    $V_k^{\pi^k}(s) = r_k(s,\bar{a}) + \sum_{s' \in S_{k+1}} p_k(s'|s,\bar{a}) V_{k+1}^{\pi^{k+1}}(s')$

    $= r_k(s,\bar{a}) + \sum_{s' \in S_{k+1}} p_k(s'|s,\bar{a}) V_{k+1}(s') = W_k(s)$

- Show $V_k(s) \leq W_k(s)$
  - Enough to show for any "tail policy" π^k
  - Fix $\pi^k = (\pi_k, \dots, \pi_{T-1})$
    - Policy π^k at time $t \geq k$ selects action $a_t \sim \pi_t(\cdot \,|h_{k:t})$
      - Where $h_{k:t} = (s_k, a_k, \dots, s_t)$
    - Policy $(\pi^k|s, a)$ is π^{k+1} after $s_k = s, a_k = a$
    - Value function
    - $V_k^{\pi^k}(s) = \sum_{a \in A_k} \pi_k(a|s) \left\{ r_k(s, a) + \sum_{s' \in S_{k+1}} p_k(s'|s, a) V_{k+1}^{(\pi^k|s,a)}(s') \right\}$

- Since V_{k+1} is optimal for all $s \in S_k$:
  - $V_k^{\pi^k}(s) \leq \sum_{a \in A_k} \pi_k(a|s) \left\{ r_k(s,a) + \sum_{s' \in S_{k+1}} p_k(s'|s,a) V_{k+1}(s') \right\}$
  - $\leq \max_{a \in A_k} \left\{ r_k(s,a) + \sum_{s' \in S_{k+1}} p_k(s'|s,a) V_{k+1}(s') \right\} = W_k(s)$
  - Showed: $V_k^{\pi^k}(s) \leq W_k(s)$

- Part *(ii) defining* π^*
  - We are taking the actions that maximize V_k
  - By backward induction
    - π^* achieves $V_k^{\pi^*} = V_k$
    - Proof similar to what we saw ….

- QED

Q function

- Definition
  - $Q_k(s,a) \triangleq r_k(s,a) + \sum_{s' \in S_k} p_k(s'|s,a) V_k(s')$
- Previous Theorem:
  - $V_k(s) = \max_{a \in A_k} Q_k(s,a)$
  - $\pi_k^*(s) \in \arg\max_{a \in A_k} Q_k(s,a)$

Finite Horizon: Summary

- Finite Horizon Optimal value function
  - Computed backward
  - Dynamic programing equation
  - Bellman equation

MC puzzle:

- Ok, what did you decide?
- Consider the following gambling game:
  - You start with X_0 dollars
  - At time t
    - With prob ¼ you have $X_t = 4X_{t-1}$
    - With prob ¾ you have $X_t = X_{t-1}/4$
- What will happen in the long run?
  - Do you want to play the game?

MC Puzzle

- Expectation:
  - $E[X_t|X_{t-1}] = \frac{1}{4}4X_{t-1} + \frac{3}{4}\frac{X_{t-1}}{4} = \frac{19}{16}X_{t-1}$
  - $E[X_t] \to \infty$
- High probability?

$$E[\log X_t\,|X_{t-1}] = \frac{1}{4}(2 + \log X_{t-1}) + \frac{3}{4}(-2 + \log X_{t-1})$$

$$= -1 + \log X_{t-1}$$

  - $E[\log X_t] \to -\infty$

Lecture 3: outline

- Markov Chain
  - Definition
  - Basic Properties
- Markov Decision Process
  - Definition
  - Return function
- Finite Horizon
  - Shortest paths
  - Dynamic Programing