

CS885 Reinforcement Learning

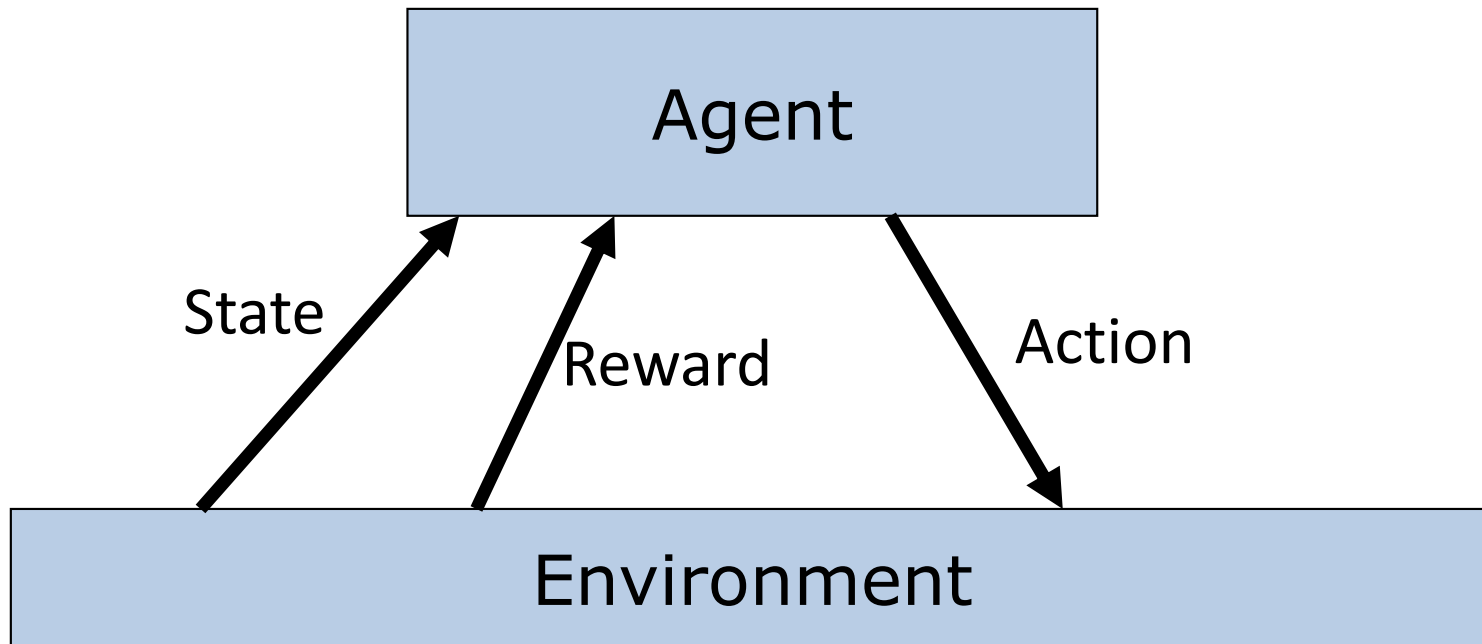
Lecture 1b: May 2, 2018

Markov Processes
[RusNor] Sec. 15.1

Outline

- Environment dynamics
- Stochastic processes
 - Markovian assumption
 - Stationary assumption

Recall: RL Problem



Goal: Learn to choose actions that maximize rewards

Unrolling the Problem

- Unrolling the control loop leads to a sequence of states, actions and rewards:

$$s_0, a_0, r_0, s_1, a_1, r_1, s_2, a_2, r_2, \dots$$

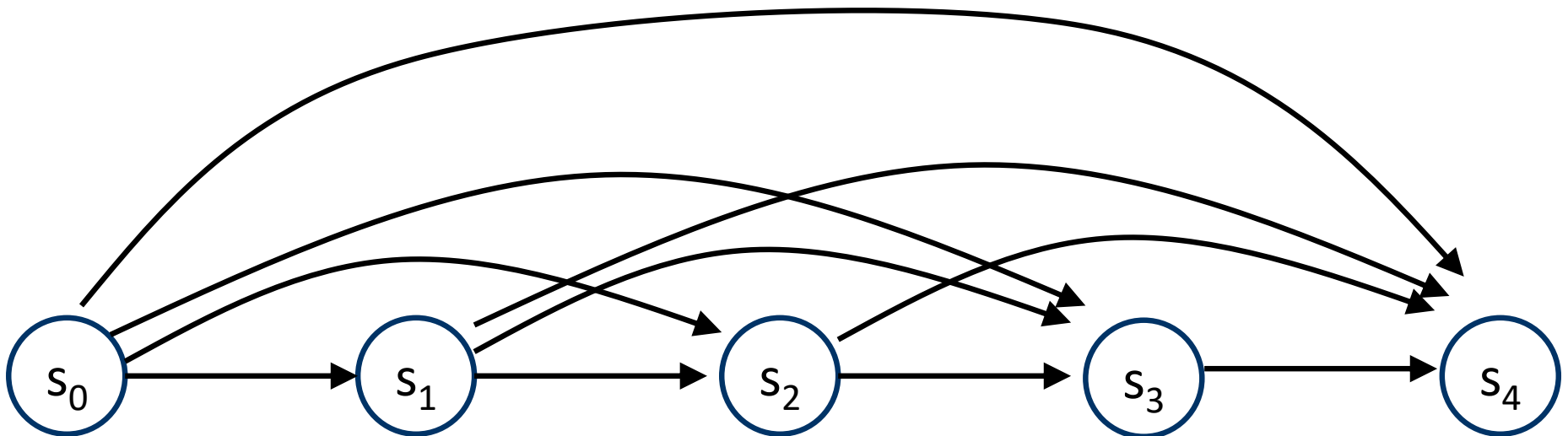
- This sequence forms a stochastic process (due to some uncertainty in the dynamics of the process)

Common Properties

- Processes are rarely arbitrary
- They often exhibit some structure
 - Laws of the process do not change
 - Short history sufficient to predict future
- **Example:** weather prediction
 - Same model can be used everyday to predict weather
 - Weather measurements of past few days sufficient to predict weather.

Stochastic Process

- Consider the sequence of states only
- Definition
 - Set of States: S
 - Stochastic dynamics: $\Pr(s_t | s_{t-1}, \dots, s_0)$



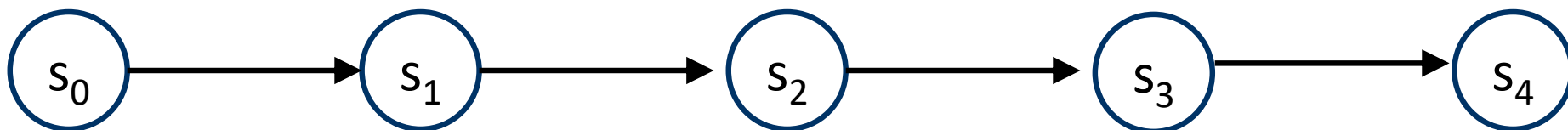
Stochastic Process

- Problem:
 - Infinitely large conditional distributions
- Solutions:
 - Stationary process: dynamics do not change over time
 - Markov assumption: current state depends only on a finite history of past states

K-order Markov Process

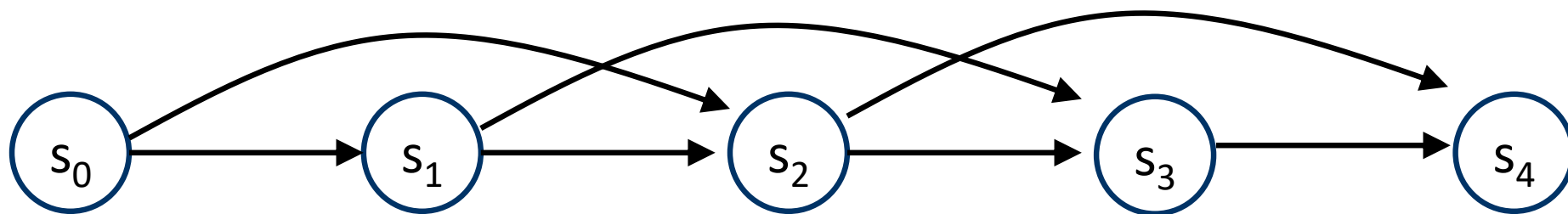
- Assumption: last k states sufficient
- First-order Markov Process

$$- \Pr(s_t | s_{t-1}, \dots, s_0) = \Pr(s_t | s_{t-1})$$



- Second-order Markov Process

$$- \Pr(s_t | s_{t-1}, \dots, s_0) = \Pr(s_t | s_{t-1}, s_{t-2})$$



Markov Process

- By default, a Markov Process refers to a

- First-order process

$$\Pr(s_t | s_{t-1}, s_{t-2}, \dots, s_0) = \Pr(s_t | s_{t-1}) \quad \forall t$$

- Stationary process

$$\Pr(s_t | s_{t-1}) = \Pr(s_{t'} | s_{t'-1}) \quad \forall t'$$

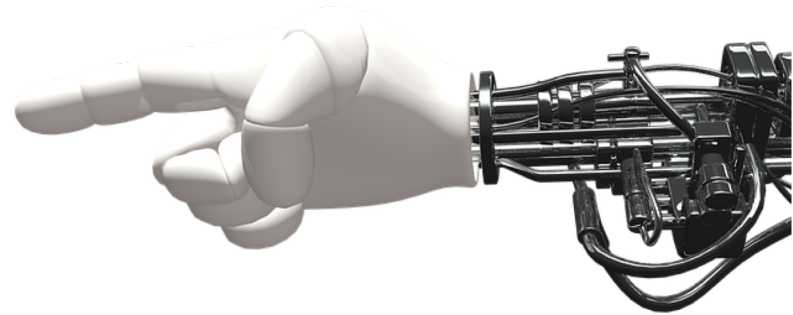
- **Advantage:** can specify the entire process with a single concise conditional distribution

$$\Pr(s' | s)$$

Examples

- Robotic control
 - **States:** $\langle x, y, z, \theta \rangle$
coordinates of joints
 - **Dynamics:** constant motion

- Inventory management
 - **States:** inventory level
 - **Dynamics:** constant (stochastic) demand



Non-Markovian and/or non-stationary processes

- What if the process is not Markovian and/or not stationary?
- Solution: add new state components until dynamics are Markovian and stationary
 - Robotics: the dynamics of $\langle x, y, z, \theta \rangle$ are not stationary when velocity varies...
 - Solution: add velocity to state description e.g. $\langle x, y, z, \theta, \dot{x}, \dot{y}, \dot{z}, \dot{\theta} \rangle$
 - If acceleration varies... then add acceleration to state
 - Where do we stop?

Markovian Stationary Process

- **Problem:** adding components to the state description to force a process to be Markovian and stationary may significantly **increase computational complexity**
- **Solution:** try to find the smallest state description that is self-sufficient (i.e., Markovian and stationary)

Inference in Markov processes

- Common task:
 - **Prediction:** $\Pr(s_{t+k}|s_t)$
- Computation:
 - $\Pr(s_{t+k}|s_t) = \sum_{s_{t+1}\dots s_{t+k-1}} \prod_{i=1}^k \Pr(s_{t+i}|s_{t+i-1})$
- Discrete states (matrix operations):
 - Let T be a $|S| \times |S|$ matrix representing $\Pr(s_{t+1}|s_t)$
 - Then $\Pr(s_{t+k}|s_t) = T^k$
 - Complexity: $O(k|S|^3)$

Decision Making

- Predictions by themselves are useless
- They are only useful when they will influence future decisions
- Hence the ultimate task is **decision making**
- How can we influence the process to visit desirable states?
 - Model: Markov **Decision** Process