University at Buffalo
Department of Computer Science and Engineering
School of Engineering and Applied Sciences

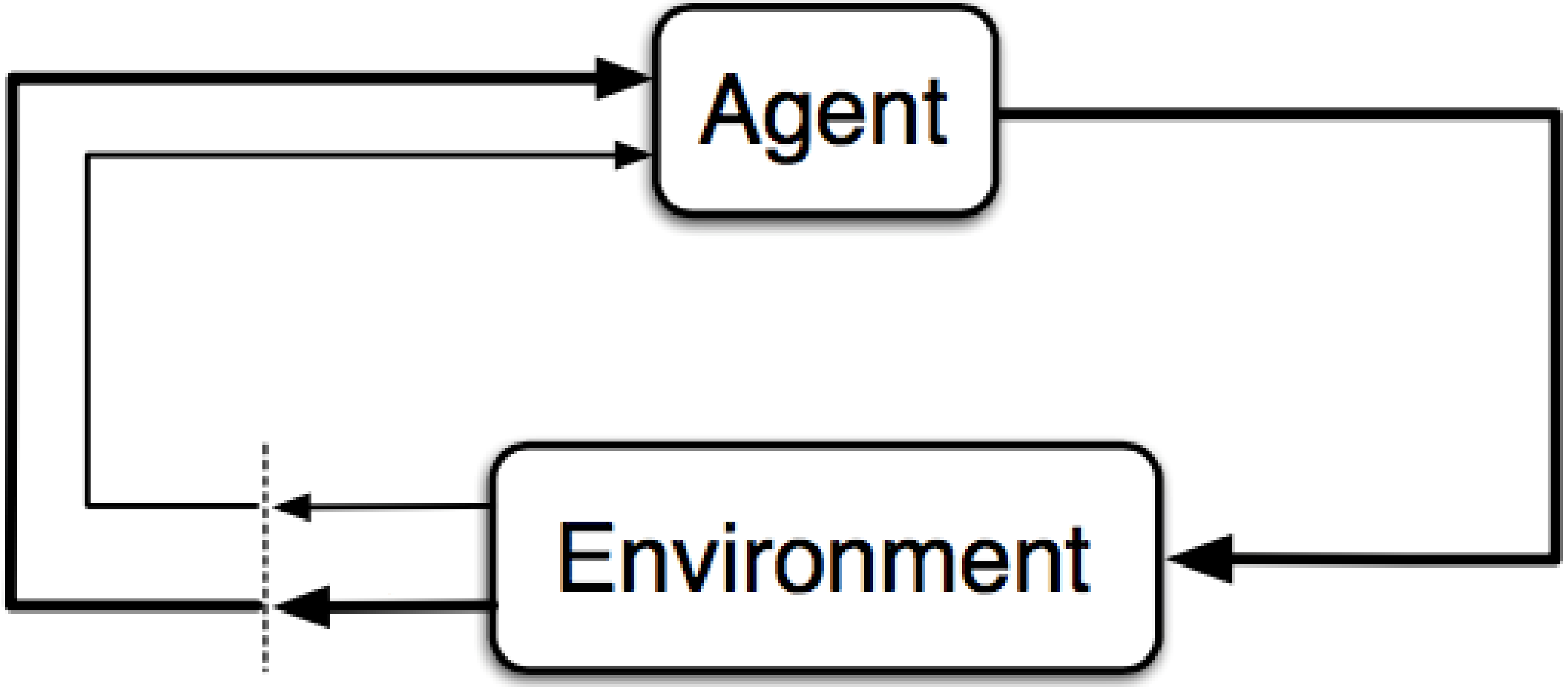
TABULAR METHODS OVERVIEW

Lecture 7.1

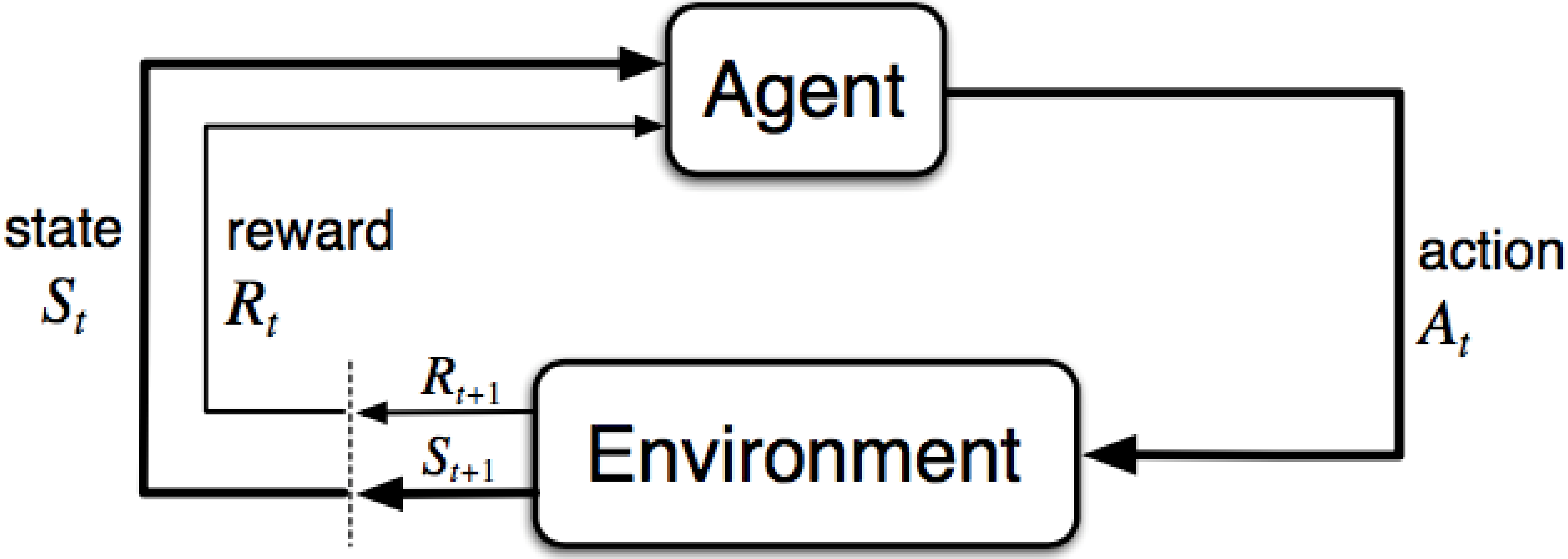
CSE4/510: Reinforcement Learning

September 17, 2019

MDP



MDP



TRUE / FALSE?

Markov Decision Process is defined as:

(s, a, O, P, \gamma)

MDP

FALSE

Markov Decision Process is defined as:

(s, a, O, P, r, \gamma)

Policy

1. Deterministic

2. Stochastic

A. $\pi(a|s) = \mathbb{P}_\pi[A = a|S = s]$

B. $\pi(s) = a$

Policy

ENVIRONMENT

Deterministic / Stochastic?

POLICY

Deterministic / Stochastic?

Agent

RL agents goal?

Types of value functions?

Value Functions

Types of value functions:

State value function describes the value of a state when following a policy. It is the expected return when starting from state s acting according to our policy π:

$$V^{\pi}(s) = \mathbb{E}_{\pi}[R_t | S_t = s]$$

Action value function tells us the value of taking an action a in state s when following a certain policy π. It is the expected return given the state and action under π:

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}[R_t | S_t = s, A_t = a]$$

Value Functions

V(s) can also be interpreted, as the cumulative future reward

Are we missing something?

Value Functions

V(s) can also be interpreted, as the expected cumulative future discounted reward

Dynamic Programming

1. Evaluate **A.** $V_\pi(s) = \mathbb{E}[r_{t+1} + \gamma r_{t+2} + \ldots | S_t = s]$

2. Improve **B.** $\pi' = greedy(V_\pi)$

Dynamic Programming

Evaluate $V_\pi(s) = \mathbb{E}[r_{t+1} + \gamma r_{t+2} + \ldots | S_t = s]$

Improve $\pi' = greedy(V_\pi)$

Dynamic Programming

Given a policy π

- **Evaluate** the policy π

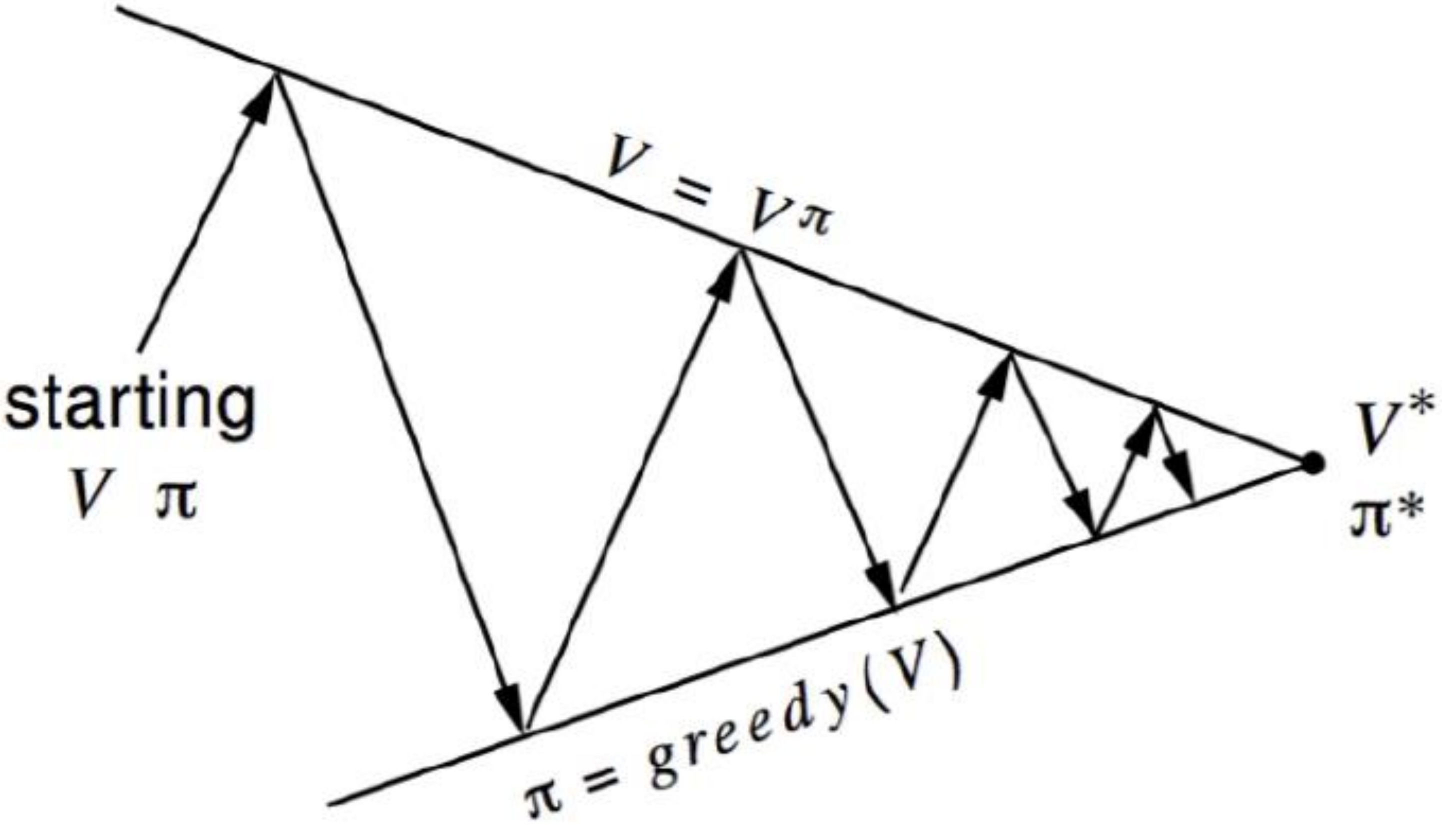
$$V_\pi(s) = \mathbb{E}[r_{t+1} + \gamma r_{t+2} + \ldots | S_t = s]$$

- **Improve** the policy by acting greedly with respect to v_π

$$\pi' = greedy(V_\pi)$$

$$\pi_0 \xrightarrow{E} V_{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} V_{\pi_1} \xrightarrow{I} \pi_2 \xrightarrow{E} V_{\pi_2} \xrightarrow{E} \ldots \xrightarrow{I} \pi_* \xrightarrow{E} V_*$$

Dynamic Programming



Dynamic Programming

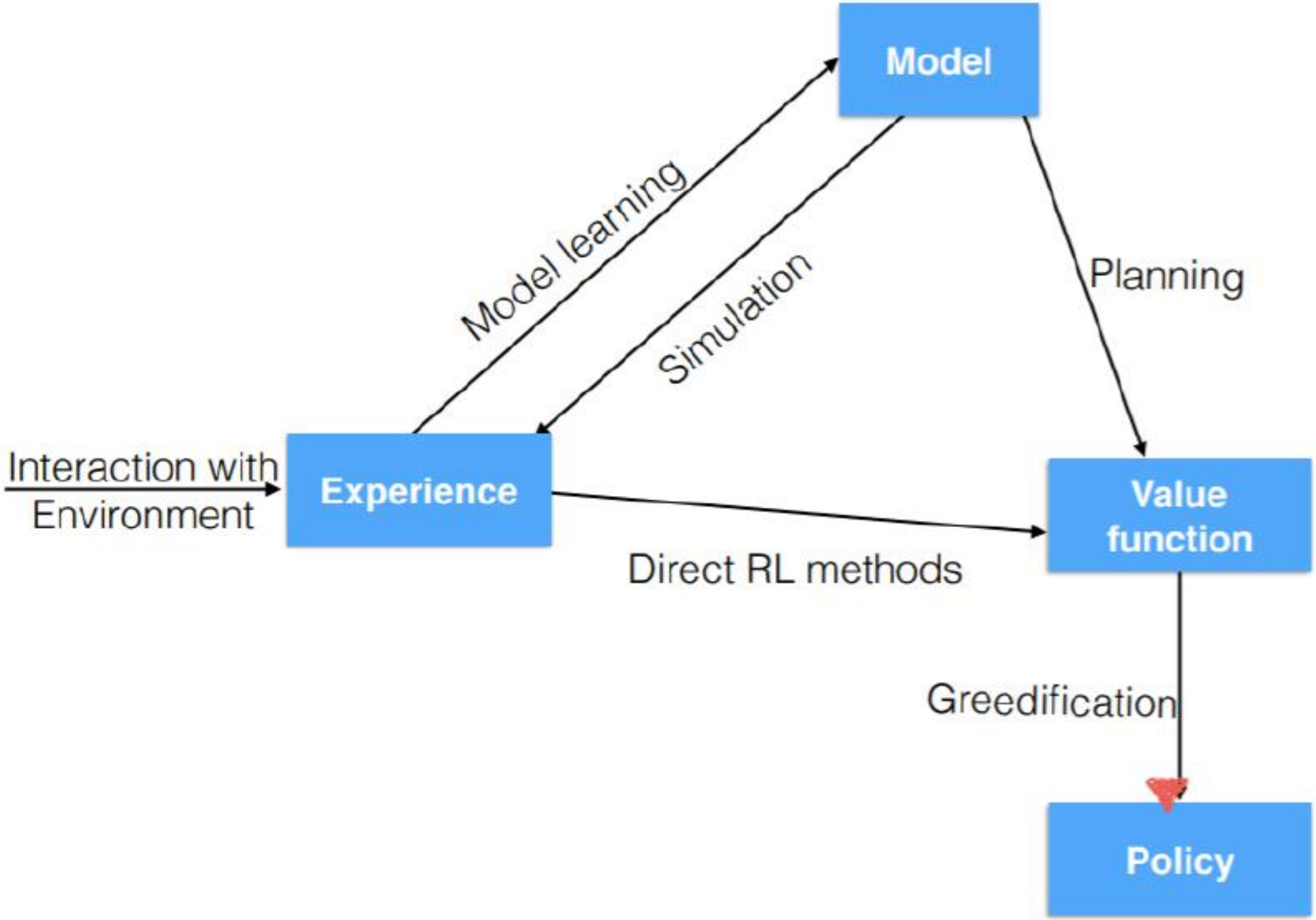
1. Distribution model

A. Produce a single outcome taken according to its probability of occurring

2. Sample model

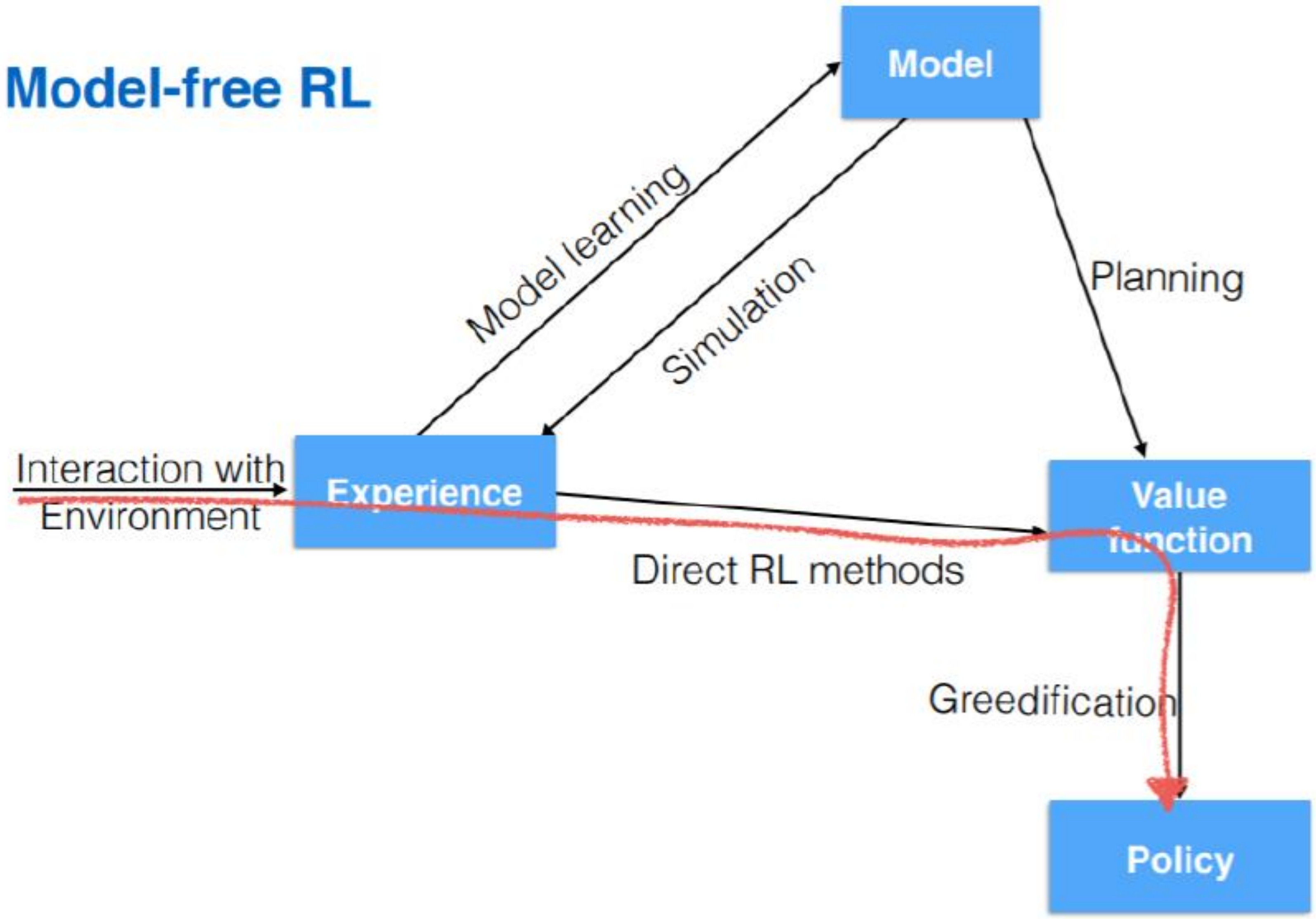
B. List all possible outcomes and their probabilities

Overview



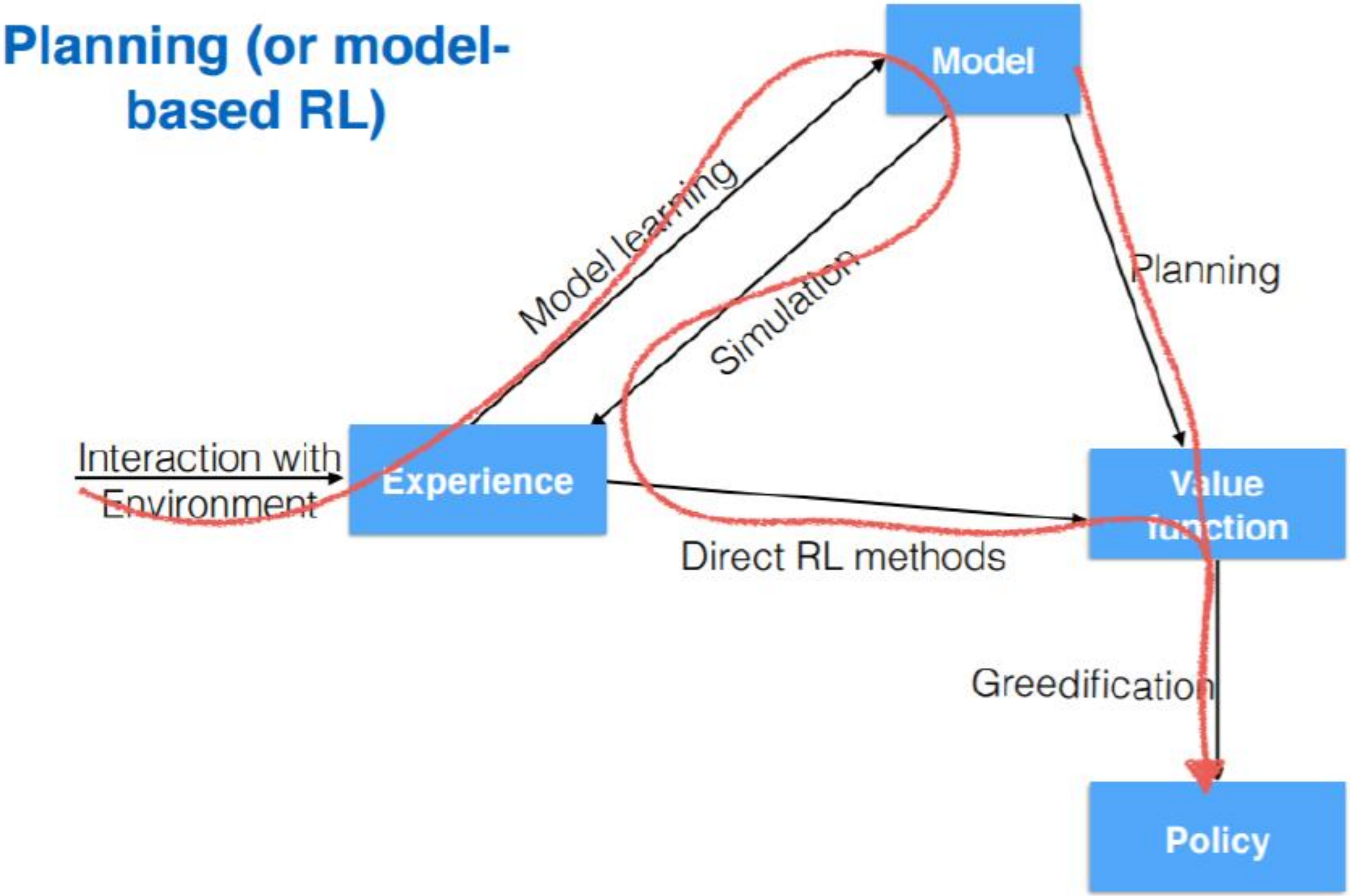
Model
Model learning
Simulation
Planning
Interaction with
Environment
Experience
Value
function
Direct RL methods
Greedification
Policy

Overview



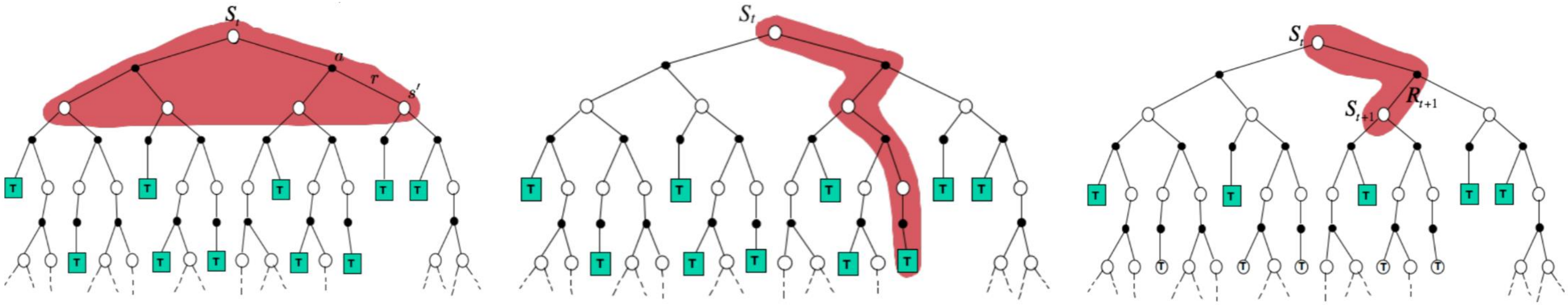
Model-free RL
Model
Model learning
Simulation
Planning
Interaction with Environment
Experience
Value function
Direct RL methods
Greedification
Policy

Overview


Planning (or model-based RL)
Model
Model learning
Simulation
Planning
Interaction with Environment
Experience
Value function
Direct RL methods
Greedification
Policy

Overview

MC / DP / TD ?



Bootstrapping

Bootstrapping

MC

DP

TD

Bootstrapping

Bootstrapping

MC

✓ DP

✓ TD

Sampling

Sampling

MC
DP
TD

Sampling

Sampling

- MC
- DP
- TD

Value Based RL

Dynamic Programming A. $V(S_t) \leftarrow V(S_t) + \alpha\left(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)\right)$

Monte Carlo B. $V(S_t) \leftarrow E_\pi\left[R_{t+1} + \gamma V(S_{t+1})\right] = \sum_a \pi(a|S_t) \sum_{s',r} p(s',r|S_t,a)[r + \gamma V(s')]$

Temporal Difference C. $V(S_t) \leftarrow V(S_t) + \alpha\left(G_t - V(S_t)\right)$

Value Based RL

Dynamic Programming $V(S_t) \leftarrow E_\pi\left[R_{t+1} + \gamma V(S_{t+1})\right] = \sum_a \pi(a|S_t) \sum_{s',r} p(s',r|S_t,a)[r + \gamma V(s')]$

Monte Carlo $V(S_t) \leftarrow V(S_t) + \alpha\left(G_t - V(S_t)\right)$

Temporal Difference $V(S_t) \leftarrow V(S_t) + \alpha\left(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)\right)$

Dynamic Programming

Policy Iteration (using iterative policy evaluation) for estimating $\pi \approx \pi_*$

1. Initialization
   $V(s) \in \mathbb{R}$ and $\pi(s) \in \mathcal{A}(s)$ arbitrarily for all $s \in \mathcal{S}$

2. Policy Evaluation
   Loop:
   - $\Delta \leftarrow 0$
   - Loop for each $s \in \mathcal{S}$:
     - $v \leftarrow V(s)$
     - $V(s) \leftarrow \sum_{s',r} p(s', r \,|\, s, \pi(s)) \big[r + \gamma V(s') \big]$
     - $\Delta \leftarrow \max(\Delta, |v - V(s)|)$

   until $\Delta < \theta$ (a small positive number determining the accuracy of estimation)

3. Policy Improvement
   policy-stable $\leftarrow$ *true*
   For each $s \in \mathcal{S}$:
   - *old-action* $\leftarrow \pi(s)$
   - $\pi(s) \leftarrow \arg\max_a \sum_{s',r} p(s', r \,|\, s, a) \big[r + \gamma V(s') \big]$
   - If *old-action* $\neq \pi(s)$, then *policy-stable* $\leftarrow$ *false*

   If *policy-stable*, then stop and return $V \approx v_*$ and $\pi \approx \pi_*$; else go to 2

TD / Monte Carlo ?