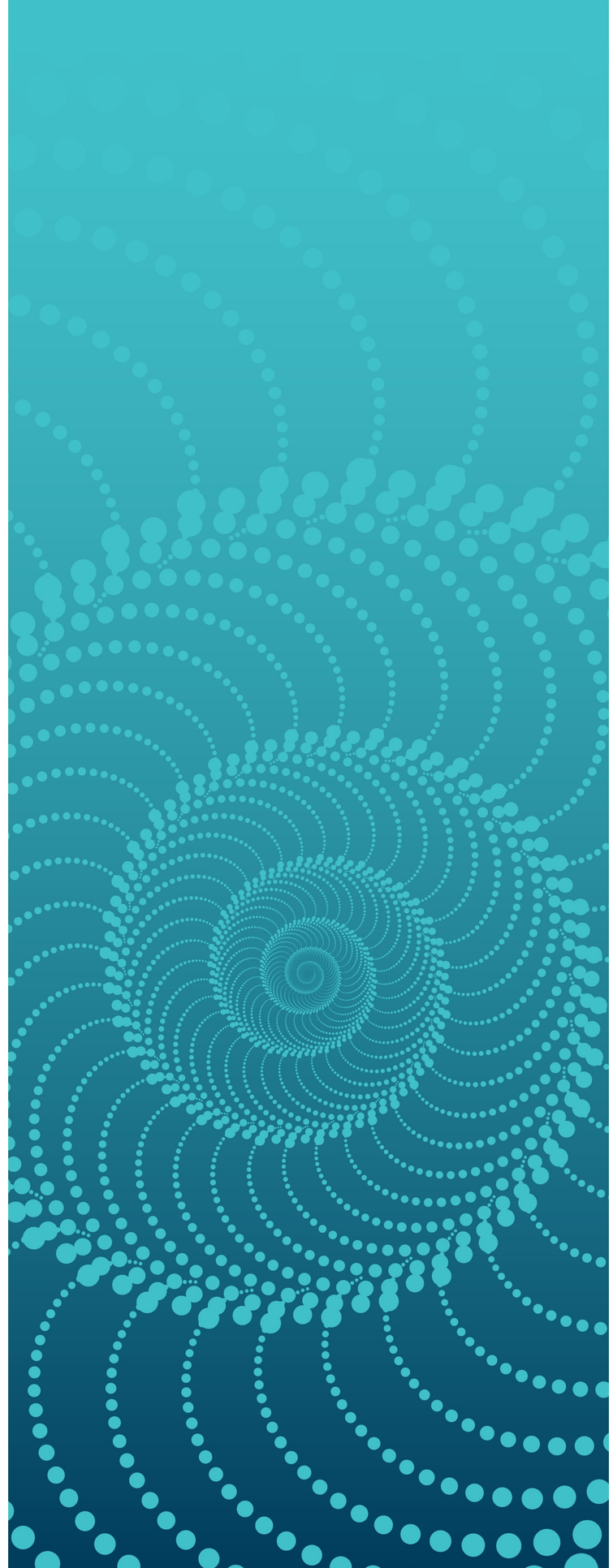
HSLU Lucerne University of Applied Sciences and Arts

Dynamic Programming Summary

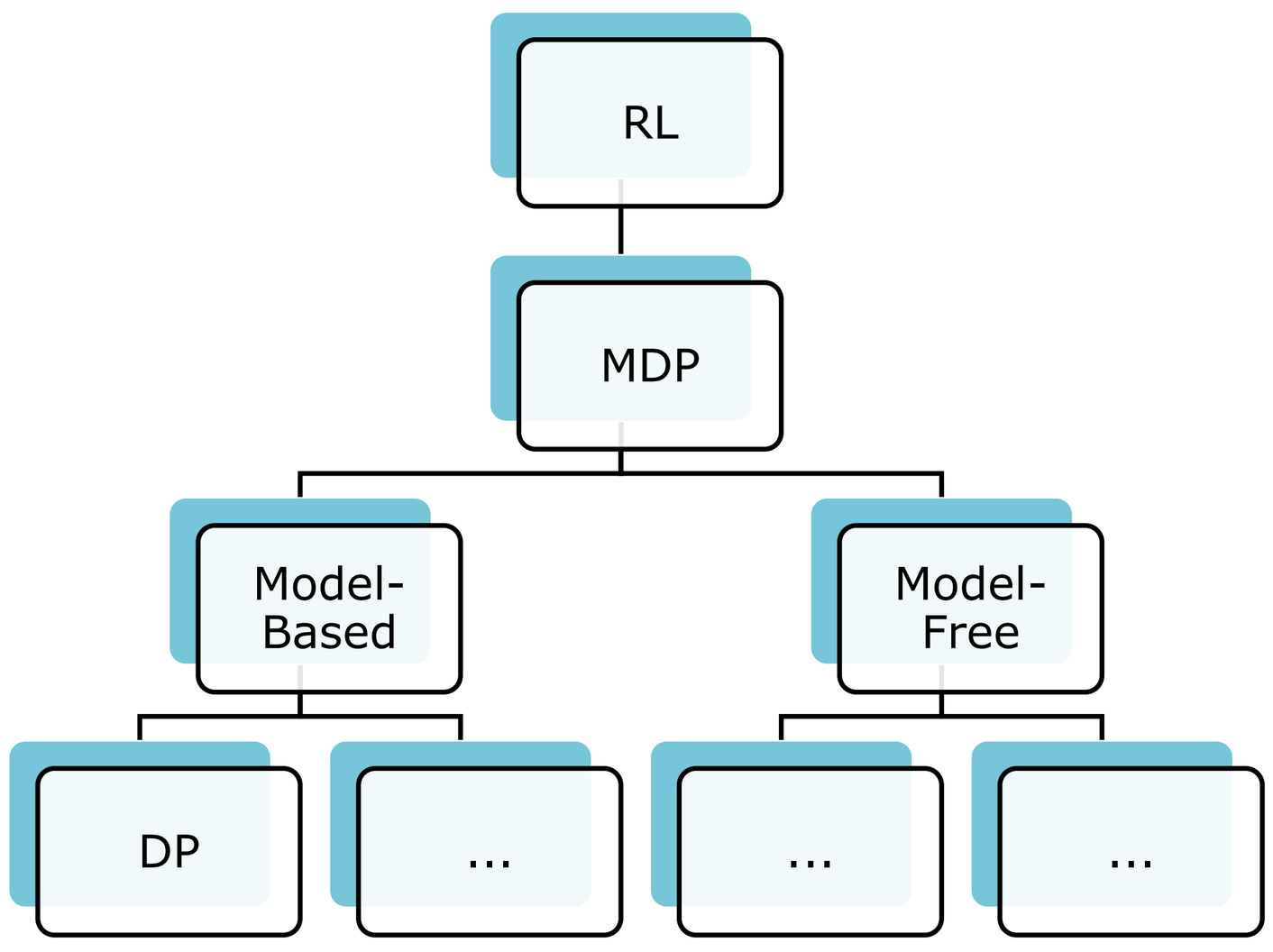
Reinforcement Learning
October 20, 2022

FH Zentralschweiz



Dynamic Programming in RL

Algorithms to compute optimal value functions and policies given a perfect **model** of the MDP



Policy Evaluation (Prediction)

- Iteratively calculates the value function of a given policy:

$$v_{k+1}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r \mid s, a) \left[r + \gamma v_k(s')\right]$$

Iterative Policy Evaluation, estimate v_π

Input: a policy π
Initialize:
  $V(s) \in \mathbb{R}$ arbitrarily, except $V(\text{terminal}) = 0$
Loop:
  $\Delta \leftarrow 0$
  Loop for each $s \in \mathcal{S}$:
    $v \leftarrow V(s)$
    $V(s) \leftarrow \sum_a \pi(a|s) \sum_{s',r} p(s', r \mid s, a) \left[r + \gamma V(s')\right]$
    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
  until $\Delta < \theta$

Policy Improvement

Calculate a new policy from a value function by using only greedy actions

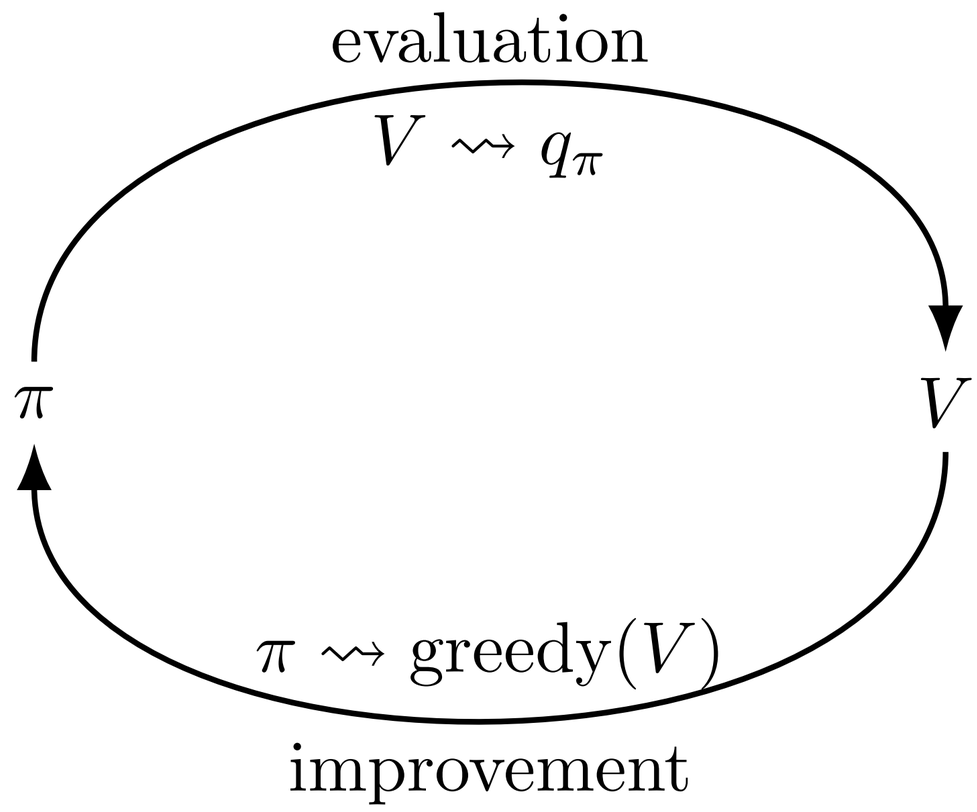
$$\begin{aligned}\pi'(s) &\doteq \operatorname*{argmax}_a q_\pi(s,a) \\ &= \operatorname{argmax} \sum_{s',r} p(s',r \mid s,a)\left[r + \gamma v_\pi(s')\right]\end{aligned}$$

Policy Iteration

Repeatedly calculate

- the value function from a policy, and then
- a better policy (greedy) from the value function

$$\pi_0 \xrightarrow{\text{E}} v_{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} v_{\pi_1} \xrightarrow{\text{I}} \pi_2 \xrightarrow{\text{E}} \cdots \xrightarrow{\text{I}} \pi_* \xrightarrow{\text{E}} v_*$$



Value Iteration

- Calculate value function after evaluating the policy ones (for all states)

$$v_{k+1}(s) = \max_a \sum_{s',r} p(s', r \mid s, a) \left[r + \gamma v_k(s') \right], \qquad \forall s \in \mathcal{S}$$

Value Iteration, estimate $\pi \approx \pi_*$

Initialize:
  $V(s) \in \mathbb{R}$ arbitrarily, except $V(\text{terminal}) = 0$
Loop:
  $\Delta \leftarrow 0$
  Loop for each $s \in \mathcal{S}$:
    $v \leftarrow V(s)$
    $V(s) \leftarrow \max_a \sum_{s',r} p(s', r \mid s, a) \left[r + \gamma V(s') \right]$
    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
  until $\Delta < \theta$
Output deterministic policy, such that
  $\pi(s) = \text{argmax}_a \sum_{s',r} p(s', r \mid s, a) \left[r + \gamma V(s') \right]$

HSLU Lucerne University of Applied Sciences and Arts

Reinforcement Learning

October 20, 2022

FH Zentralschweiz

HSLU Lucerne University of Applied Sciences and Arts

{{DN:Hierarchy|Organisation Bezeichnung Spez.EN|ID:32|Hierarchy:1}}
Research
Prof. Dr. Thomas Koller
Lecturer

Phone direct +41 41 757 68 32
thomas.koller@hslu.ch