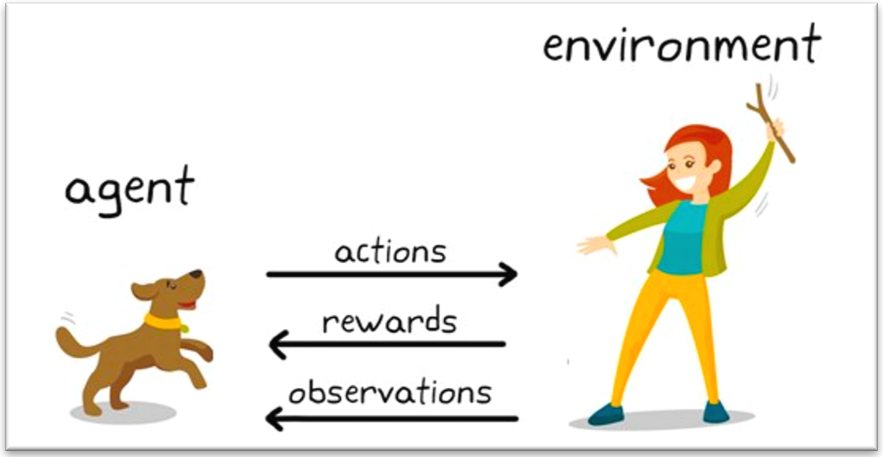
Contextual Decision Processes with low Bellman rank are PAC-Learnable

June 29, 2020

Hoe Sung Ryu

1. Preliminary



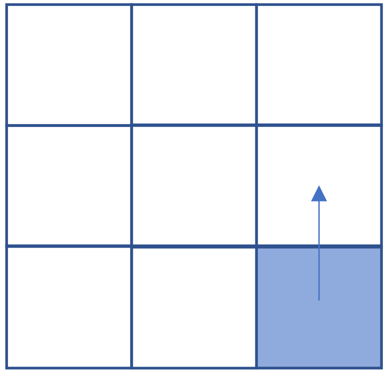
Problems

1. Exploration
2. Long term planning
3. Generalization

* Note that
Exploitation: make the best decision given current information
Exploration: gather more information

1. Preliminary

Markov Decision Processes (MDPs)



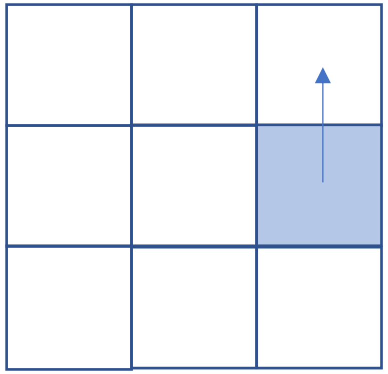
State: $\chi_1 \sim \Gamma_1$

Take action : a_1

Observe: $r(a_1)$

1. Preliminary

Markov Decision Processes (MDPs)



State: $\chi_2 \sim \Gamma_1(\chi_2, a_1)$

Take action : a_2

Observe: $r(a_2)$

Episodic: H steps in trajectory

Markovian:

- χ_h only depend on (χ_{h-1}, a_{h-1})
- r_h only depend on (χ_h, a_h)

Challenge: Allow large number of unique observations $\mathcal{X}$

1. Preliminary

Goal of Reinforcement Learning

Maximize long-term reward

$$\sum_{h=1}^{H} r_h(a_h)$$

1. Preliminary

Goal of Reinforcement Learning

Maximize long-term reward using policies that are mapping from state to actions

$$\sum_{h=1}^{H} r_h(\pi(x_h))$$

1. Preliminary

Existing results

Cardinality state space

Good: Learn ε −optimal policy using $poly(\,|\mathcal{X}|, A, H, \frac{1}{\varepsilon}\,)$

The ϵ-*greedy* algorithm continues to explore forever

- With probability $1 - \epsilon$ select $a = \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, \hat{Q}(a)$
- With probability ϵ select a random action

Bad: Small number of states necessary for learning

1. Preliminary

Lower Bound Exponential lower bound exists(Akshay Krishnamurthy, NIPS 2016)

There is an MDP with $|A|^H$ states

where finding an ε −optimal policy requires $\Omega(\frac{|A|^H}{\epsilon^2})$ trajectories.

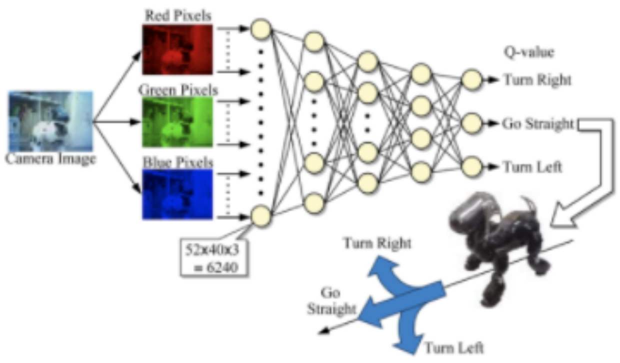
Too many "unique" states in real-wold task

=> Typically, done via value-function approximation

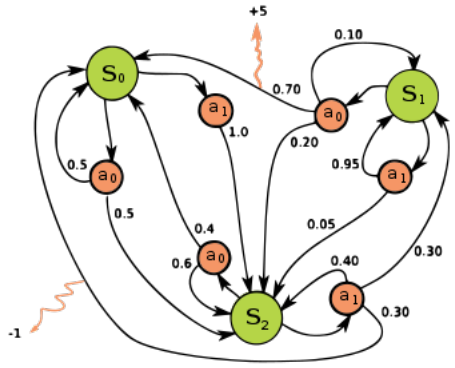
=> I.e. Using Deep Learning method generalizes across realted observations

1. Introduction

Empirical RL



Theory RL



Contextual Decision Processes

Develop Reinforcement learning approaches guaranteed to learn an optimal policy with a small number of samples despite rich observations

1. Introduction

Key ideas

- Introduce a new model: Contextual Decision processes(CDPs)

- New measure of the hardness of exploration: Bellman Rank & Bellman error Matrices

- Algorithm with sample complexity scaling with `Bellman Rank': Optimism Led Iterative Value-function Elimination(OLIVE)

1. Introduction

Assumption 2 (*Realizability*). We are given access to a class of predictors $\mathcal{F} \subseteq (\mathcal{X} \times \mathcal{A} \to [0,1])$ of size $|\mathcal{F}| = N$ and assume that $Q^\star = f^\star \in \mathcal{F}$. We identify each predictor f with a policy $\pi_f(x) \triangleq \mathrm{argmax}_a\, f(x,a)$. Observe that the optimal policy is $\pi_{f^\star}$ which satisfies $V(\pi_{f^\star}) = V^\star$.

https://papers.nips.cc/paper/6575-pac-reinforcement-learning-with-rich-observations.pdf

$\boxed{\text{Define new formulation: CDPs}}$

Average Bellman error

Optimism Led Iterative Value-function Elimination (OLIVE)

2. Contextual Decision Processes(CDPs)

Definitions 1. CDPs

Let, $\mathrm{H} \in \mathbb{N}$ denote a time Horizon , A is the action space and

$\mathcal{X}$ be a large state space of unbounded size and partitioned into subset $\chi_1, \chi_2 \dots \chi_H$

Then, finite horizon episodic CDP tuple is $(\mathcal{X}, \mathrm{A}, R, \mathrm{P})$ where R is reward and P is policy

2. Contextual Decision Processes(CDPs)

Definitions 2. Policy and Value of a policy

- Policy $\pi: \mathcal{X} \rightarrow A$ s.t. $a_h = \pi(x_h), \forall h \in H$

- Value function of a policy $V^\pi = \mathbb{E}[\sum_{h=1}^{H} r_h | a_{1:H} \sim \pi]$

  (where $a_{1:H \sim \pi}$ abbreviates for $a_1 = \pi(x_1), \ldots a_H = \pi(x_H)$)

3. Average Bellman error

Sketch of Algorithm

- Start with an initial guess $f_1 \in \mathrm{F}$
- According to f_1. Collect trajectories $(\mathcal{X}, \mathrm{A}, R, \mathrm{P})$

  s.t. $(\chi_1, a_1, r_1 \ \ldots, \chi_h, a_h, r_h)$ where $a_h = \pi_{f_1}(\chi_h)$
- Use trajectories to obtain better estimate $f_2 \in \mathrm{F}$
- Repeat

Define new formulation: CDPs

$\boxed{\text{Average Bellman error}}$

Optimism Led Iterative Value-function Elimination (OLIVE)

3. Average Bellman error

Definitions 3. Average Bellman error

Given a policy $\pi: \mathcal{X} \rightarrow A$ and a function $f: \mathcal{X}$ x $A \rightarrow [0,1]$ $then$,
the average Bellman error of f under π at level h is defined as

$$\varepsilon(f, \pi, h) = \mathbb{E}[f(x_h, a_h) - r_h - f(x_{h+1}, a_{h+1}) | \, a_{1:h-1} \sim \pi, a_{h:h+1} \sim \pi_f]$$

3. Average Bellman error

Definitions 4. (Bellman error is zero)

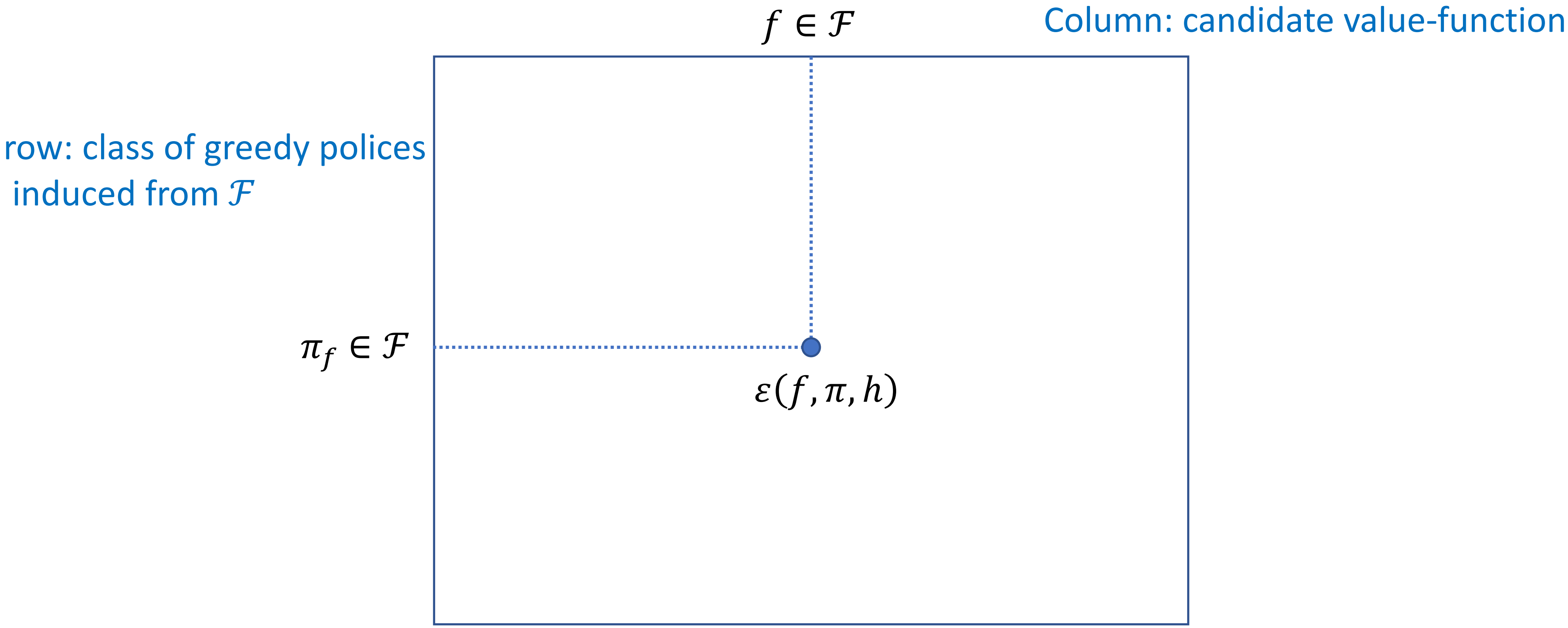
Given an (f, π, h) triple,

If f is optimal value function $in\ \mathcal{F}$ then $\varepsilon(f, \pi, h) = 0\ where, \forall\ \pi\ and\ h$

=> Finally, we can discover f that behaves as Q*,

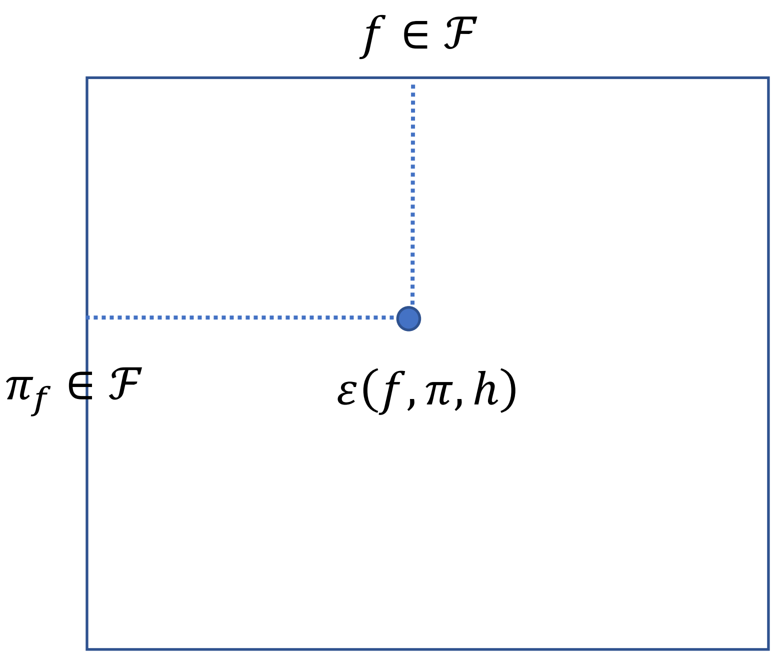
3. Average Bellman error

Bellman error Matrix



3. Average Bellman error

Bellman error Matrix



We do not know the basis, just its exitence.

Define new formulation: CDPs

Average Bellman error

Optimism Led Iterative Value-function Elimination (OLIVE)

4. Optimism Led Iterative Value-function Elimination(OLIVE)

Simplified Algorithm (OLIVE)

STEP 1. Generate trajectories using π_f

$f \in \mathcal{F}$

$\pi_f \in \mathcal{F}$

$\varepsilon(f, \pi, h) \neq 0$

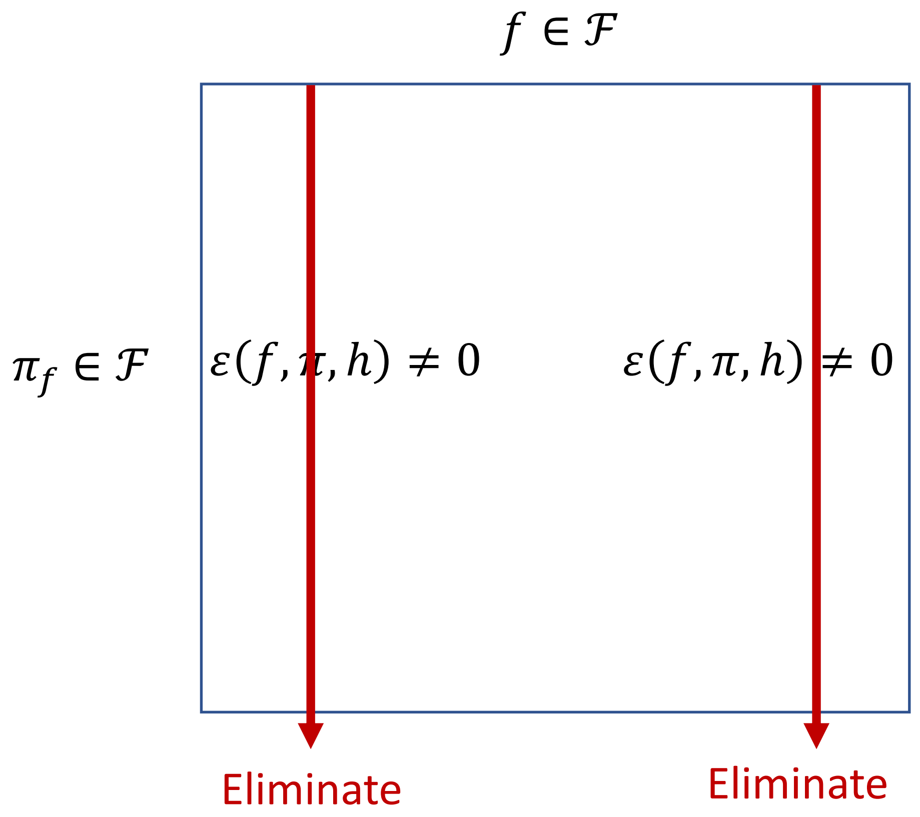
$\varepsilon(f, \pi, h) \neq 0$

4. Optimism Led Iterative Value-function Elimination(OLIVE)

Simplified Algorithm (OLIVE)

STEP 1. Generate trajectories using π_f

STEP 2. Eliminate all f with non-zero Bellan error



4. Optimism Led Iterative Value-function Elimination(OLIVE)

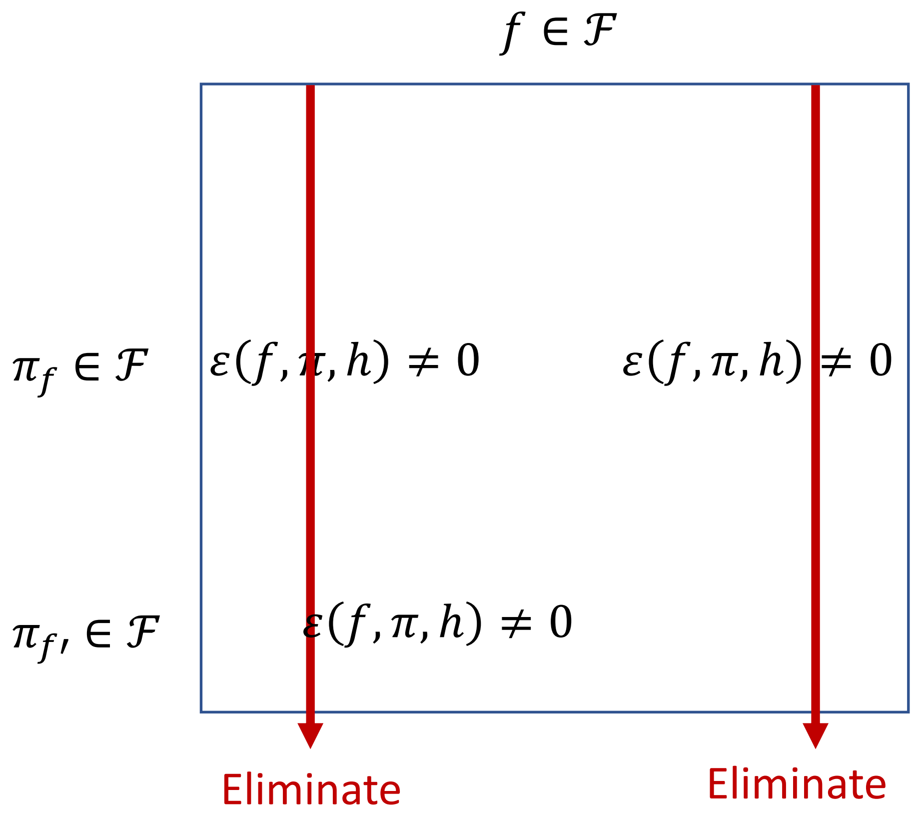
Simplified Algorithm (OLIVE)

STEP 1. Generate trajectories using π_f

STEP 2. Eliminate all f with non-zero Bellan error

STEP 3. Choose a new $\pi_{f'}$ optimistically:

is the maximizer of



4. Optimism Led Iterative Value-function Elimination(OLIVE)

Simplified Algorithm (OLIVE)

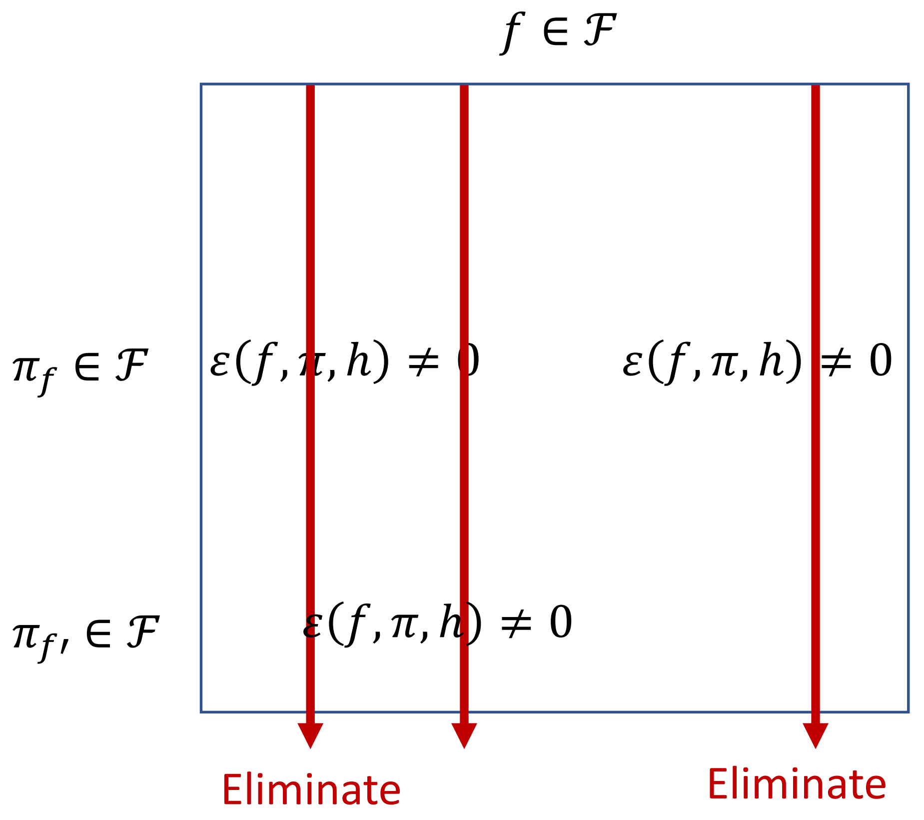
STEP 1. Generate trajectories using π_f

STEP 2. Eliminate all f with non-zero Bellan error

STEP 3. Choose a new $\pi_{f'}$ optimistically:

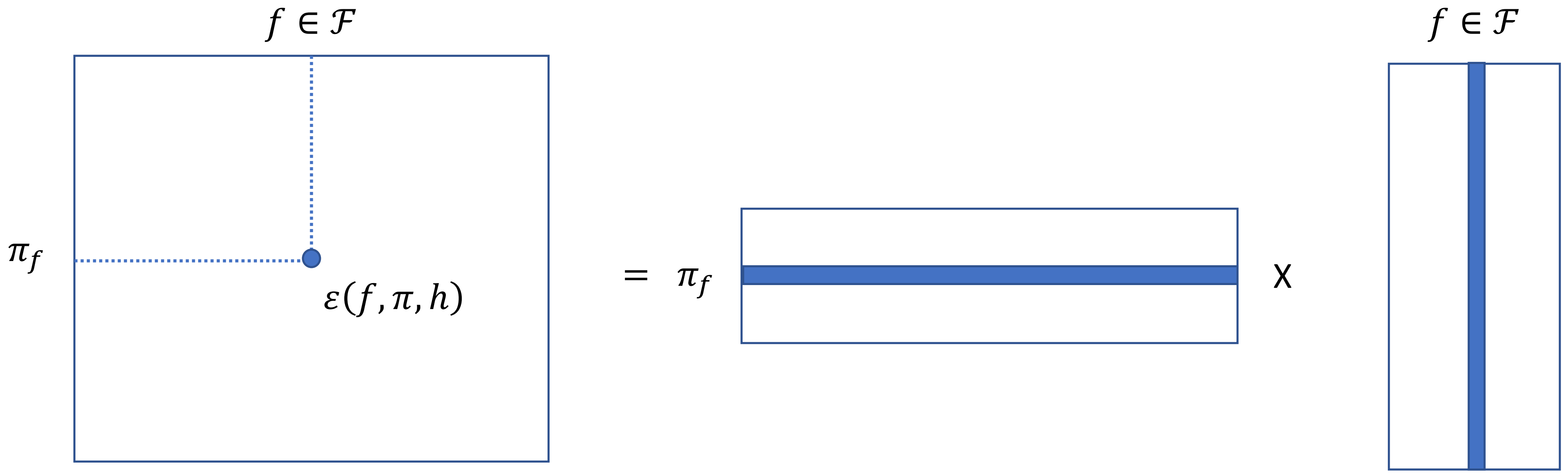
STEP4. Repeat until $V^{\pi *} - V^{\pi} \leq \varepsilon$

Maximize the value function V^{π}



4. Optimism Led Iterative Value-function Elimination(OLIVE)

Analysis of iteration complexity with Factored Matrix view



Note that $\mathrm{Rank(AB)} \leq \mathrm{min}(rank(A), rank(B))$
=> Rank will be less than $|\mathcal{F}|$

Q&A

Thank you