RL for Operations
Day 2: Nonparametric RL, Exogenous MDPs

Sean Sinclair, Sid Banerjee, Christina Yu
Cornell University

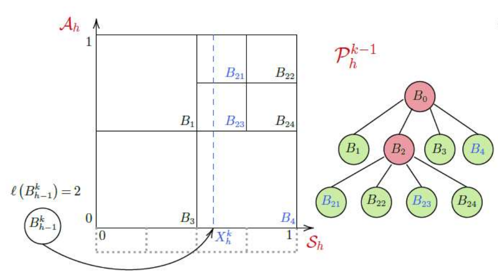


Figure 0 Illustrating the state-action partitioning scheme

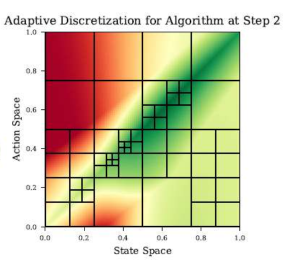
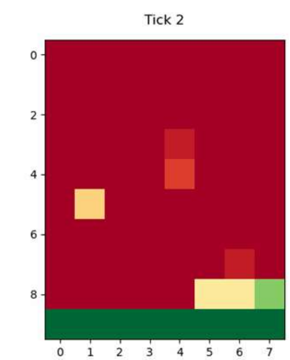


Figure 0 Partitioning in practice



RL for Operations
Day 2: Nonparametric RL, Exogenous MDPs

Sean Sinclair, Sid Banerjee, Christina Yu
Cornell University

Plan for Today

Nonparametric RL

- "Nonparametric" function approximation
- Strong guarantees across:
  Sample complexity, *space complexity*, *storage complexity*

Hindsight Learning

- Exogenous MDPs as model for OR problems
- Use of *Hindsight Planning* oracle for algorithm design
- Empirical results in VM allocation with Microsoft Azure

Tree-Partitions

- Implement tree-based adaptive discretization from nonparametric RL algorithms
- Use ORSuite to test on "continuous Ambulance routing"

Hindsight Planning for Exo-MDPs

- Use ORSuite model for revenue management and pricing (an example of an Exo-MDP)
- Implement Bayes Selector
- Use ORSuite to run simulations to compare performance against tabular algorithms

Plan for Today

Nonparametric RL

- "Nonparametric" function approximation
- Strong guarantees across:
  Sample complexity, *space complexity*, *storage complexity*

Hindsight Learning

- Exogenous MDPs as model for OR problems
- Use of *Hindsight Planning* oracle for algorithm design
- Empirical results in VM allocation with Microsoft Azure

Tree-Partitions

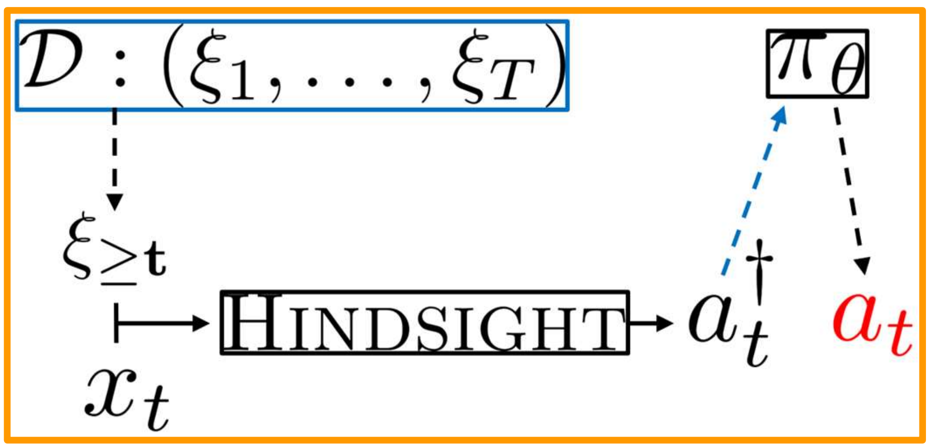
- Implement tree-based adaptive discretization from nonparametric RL algorithms
- Use ORSuite to test on "continuous Ambulance routing"

Hindsight Planning for Exo-MDPs

- Use ORSuite model for revenue management and pricing (an example of an Exo-MDP)
- Implement Bayes Selector
- Use ORSuite to run simulations to compare performance against tabular algorithms

RL in MDPs with Exogenous Inputs

Sean Sinclair,
Cornell University



VM Allocation

Requests arrive with corresponding lifetimes and sizes

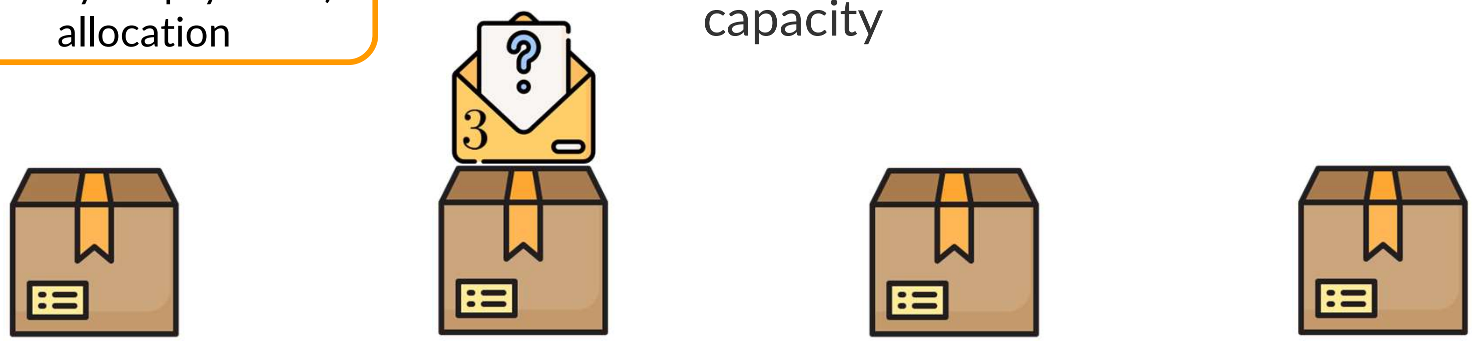
Set of Physical Machines

Replays traces from Azure Public Trace Dataset

VM Allocation

This is called a HEN (healthy empty node) allocation

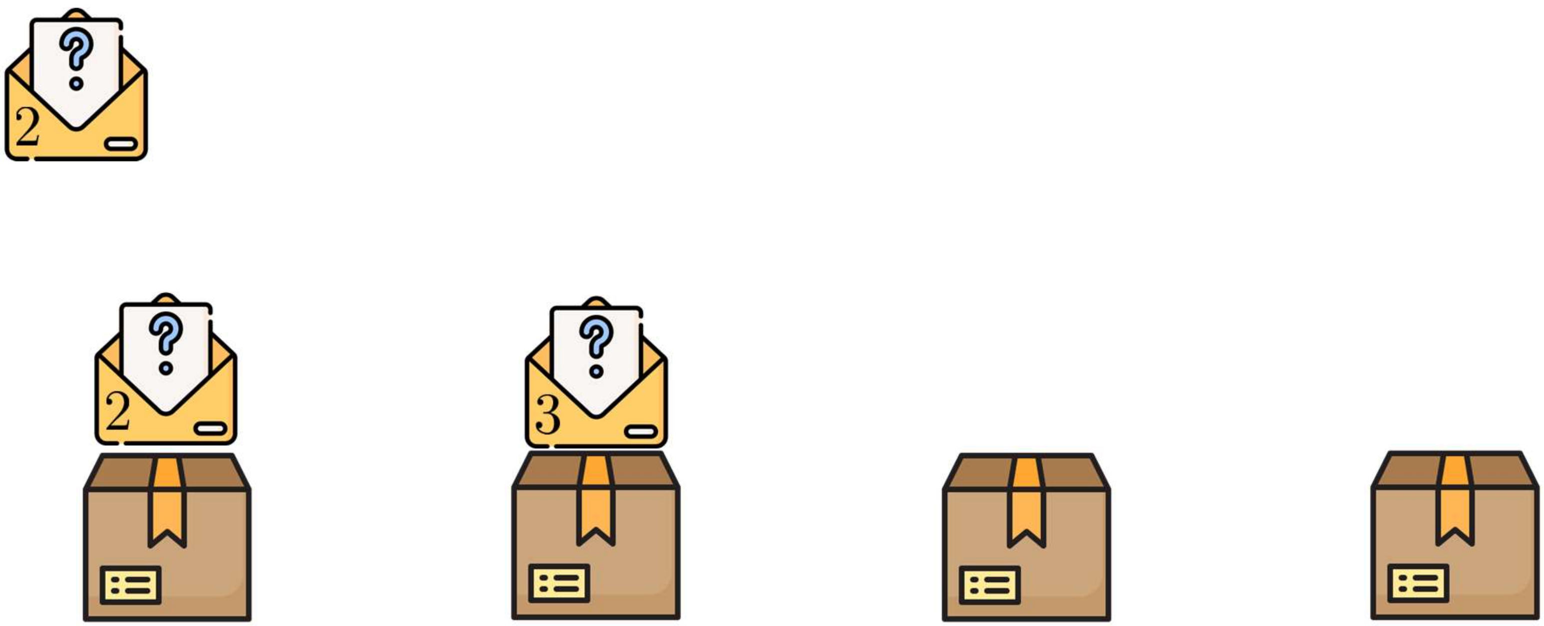
Decide which machine to allocate based on current capacity



Set of Physical Machines

Replays traces from Azure Public Trace Dataset

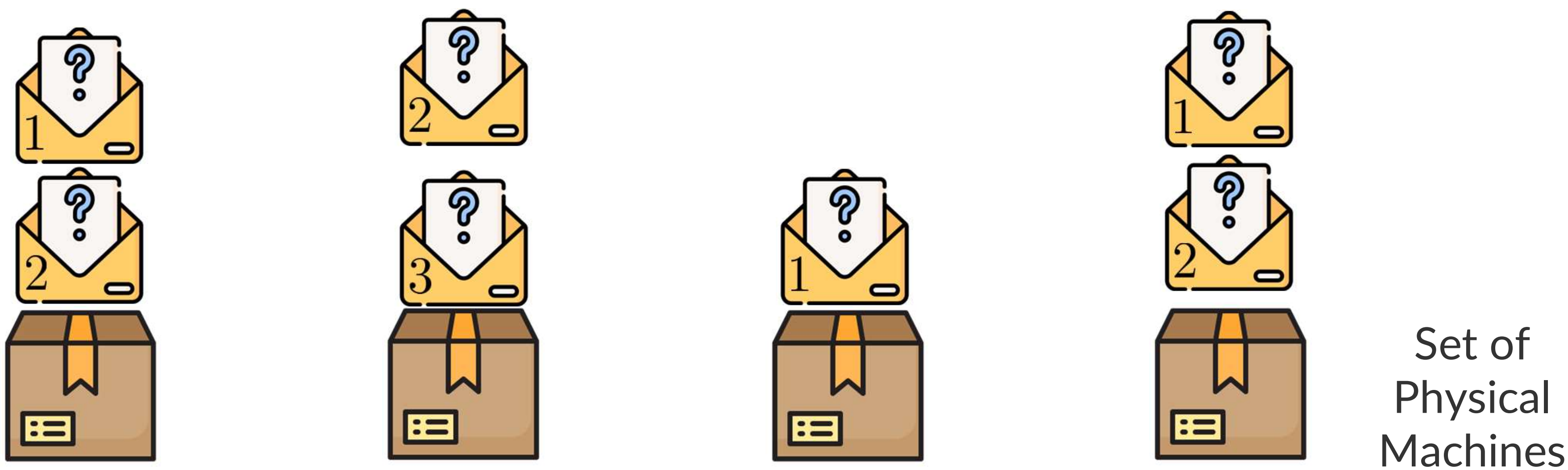
VM Allocation



Set of Physical Machines

Replays traces from Azure Public Trace Dataset

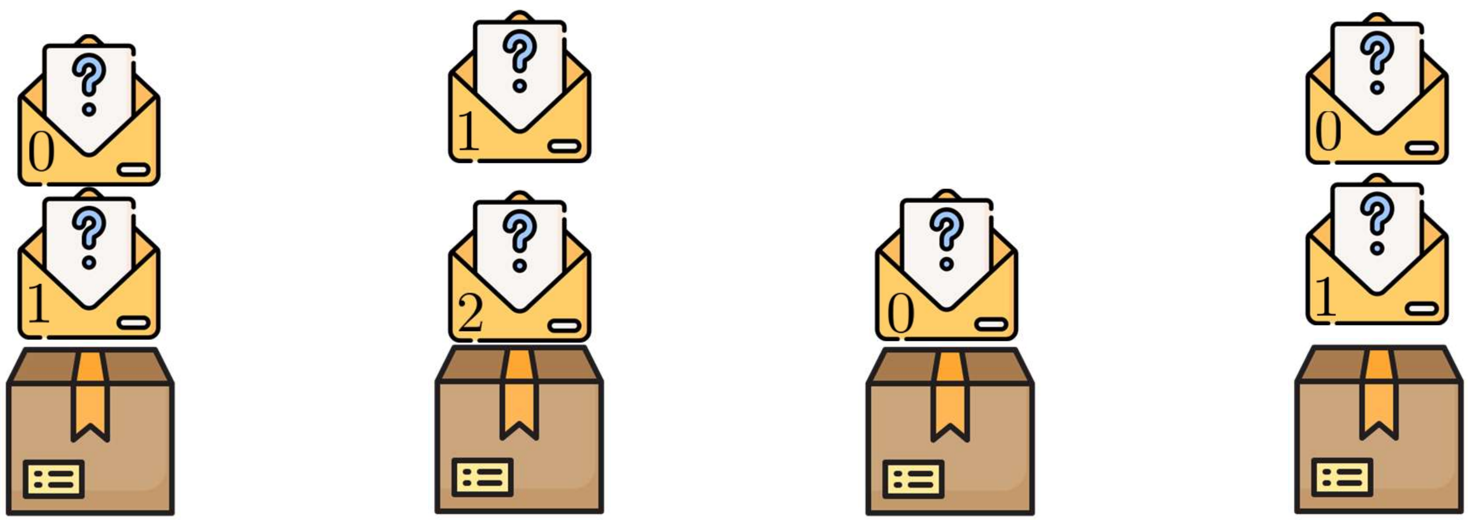
VM Allocation



Replays traces from Azure Public Trace Dataset

VM Allocation

‘Time’ passes and expired VMs
leave system, and process repeats

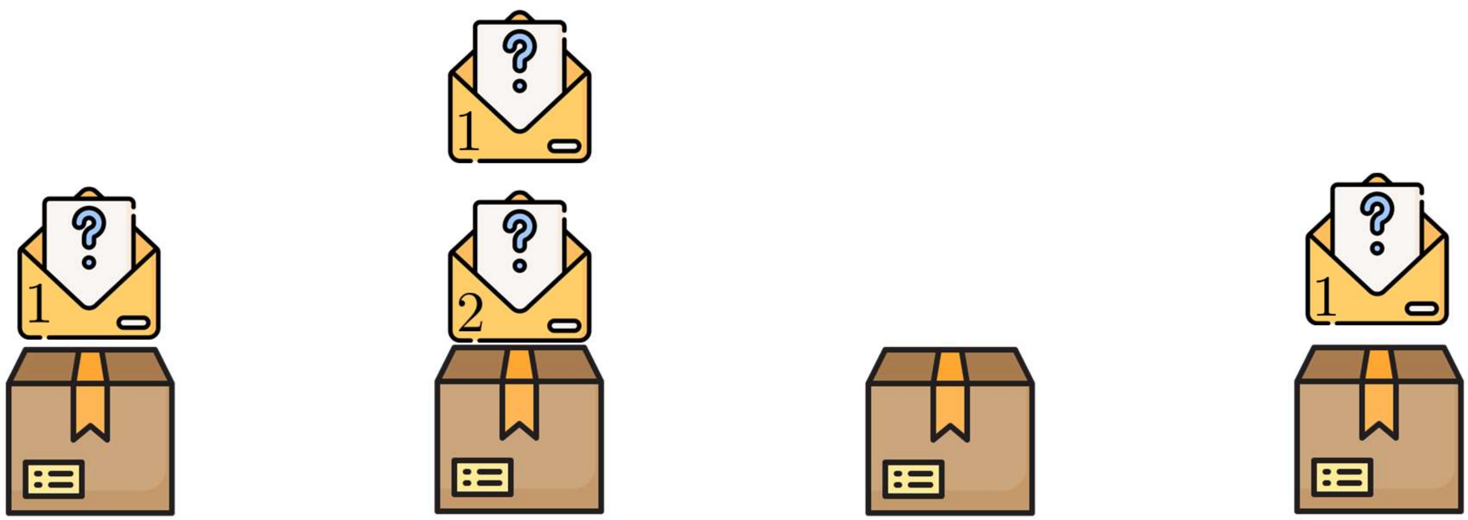


Set of
Physical
Machines

Replays traces from Azure Public Trace Dataset

VM Allocation

‘Time’ passes and expired VMs
leave system, and process repeats



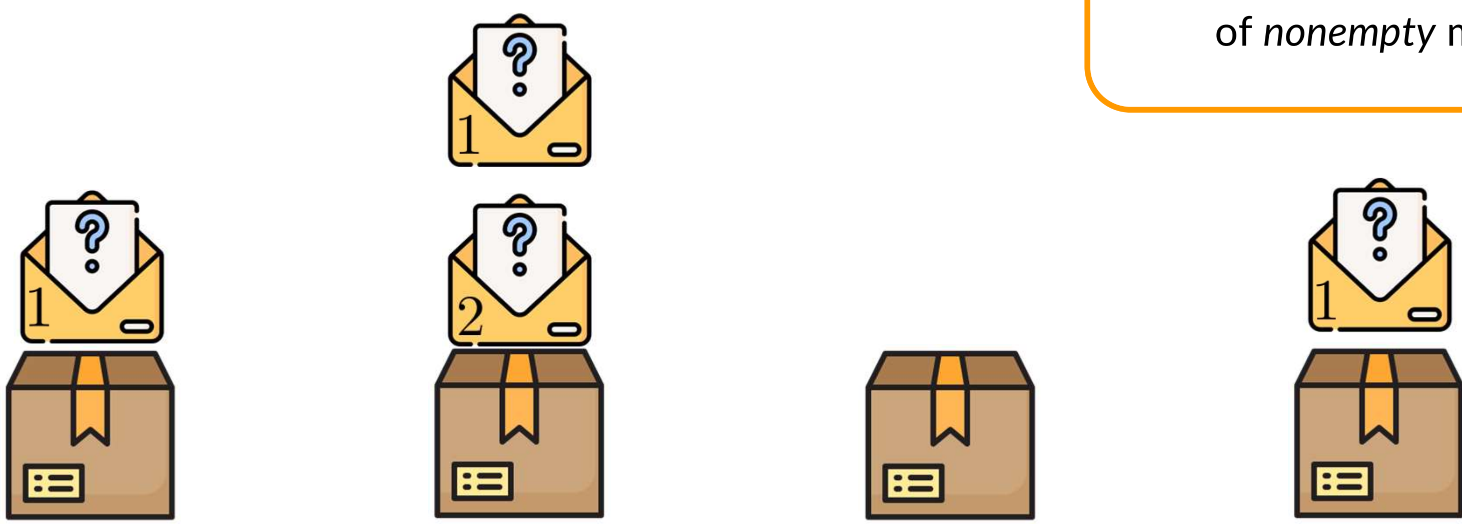
Set of
Physical
Machines

Replays traces from Azure Public Trace Dataset

VM Allocation

'Time' passes and expired VMs leave system, and process repeats

Hope to achieve good packing, i.e. used resources over capacity of *nonempty* machines

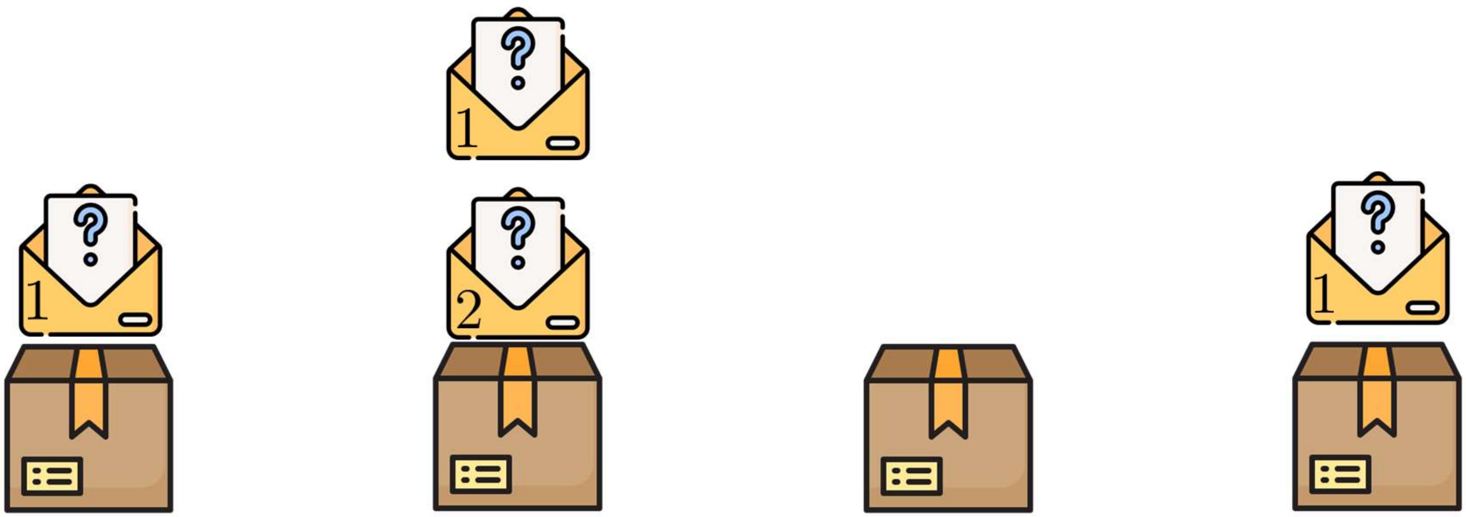


Set of Physical Machines

Replays traces from Azure Public Trace Dataset

VM Allocation

Exogenous demand governs state transition and rewards



Set of Physical Machines

Replays traces from Azure Public Trace Dataset

Structures

Real applications have additional structure:

- "Low Rank" Q values
- "Low Rank" Reward and Transitions
- Function Approximation

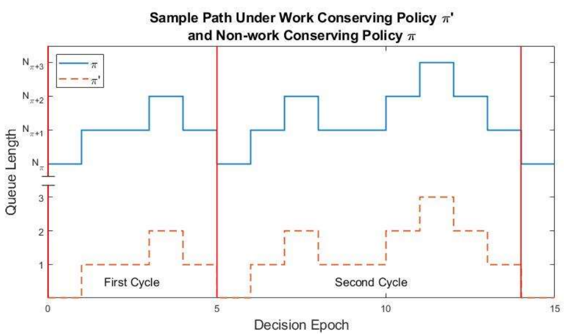
Geometric Interpretations

Difficult to verify, hard to understand in OR models

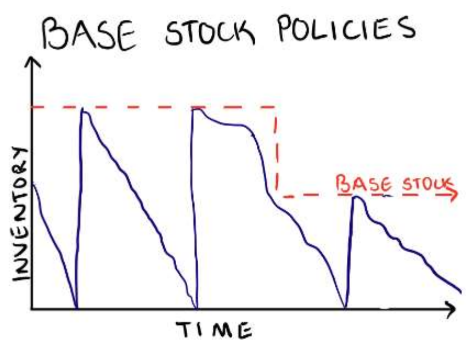
Additional structure in OR arises through modelling

Exogeneity

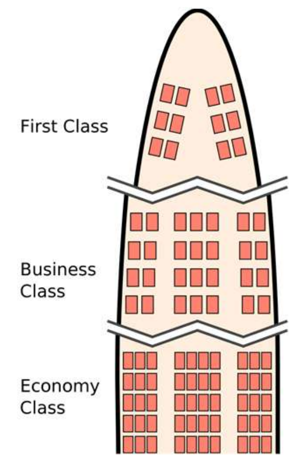
Exogenous demand governs state transition and rewards



Stochastic Networks
(patient arrivals)

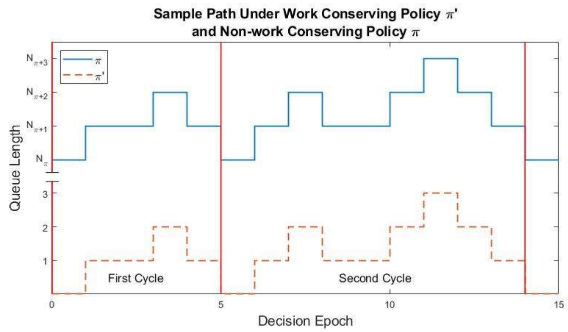


Inventory Control
(demand)

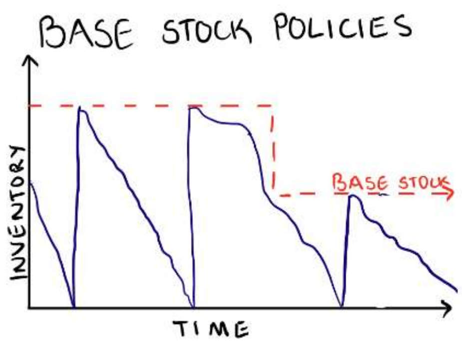


Revenue Management
(fare class)

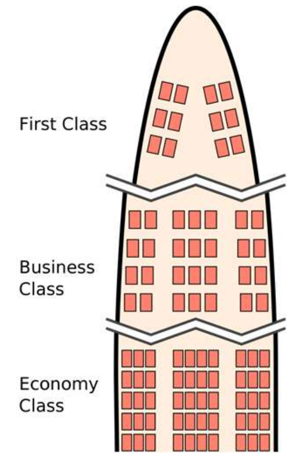
Exogeneity



Stochastic Networks
(patient arrivals)



Inventory Control
(demand)



Revenue Management
(fare class)

How can we design data-driven algorithms for RL with exogenous uncertainty?

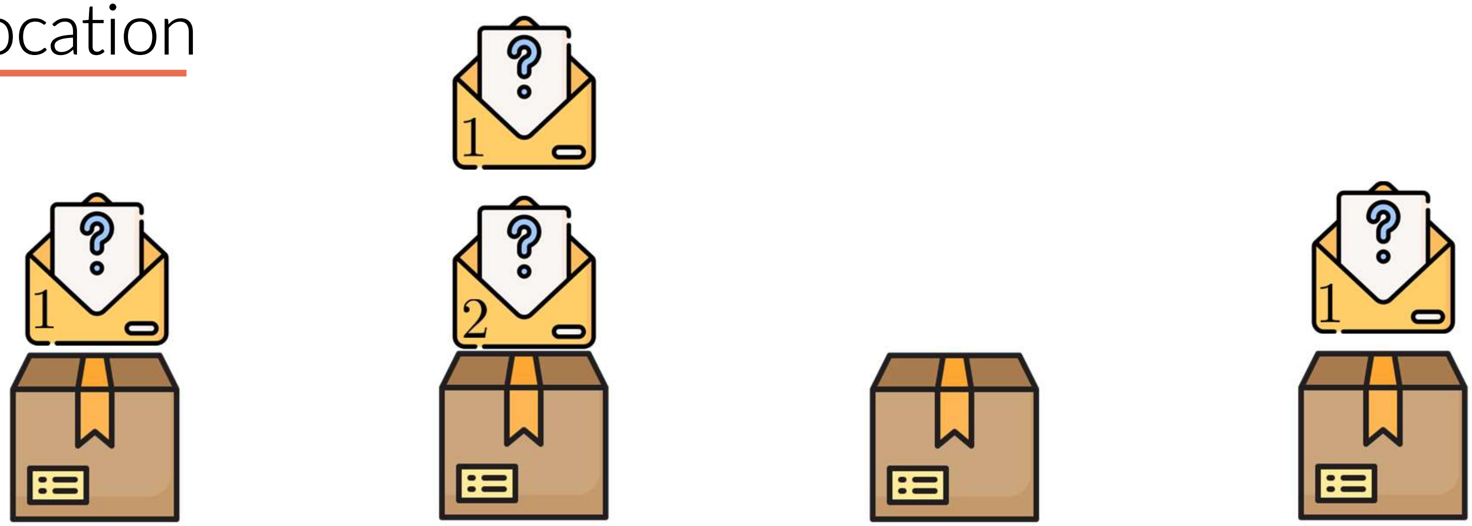
Recall................

Maybe a better model....

Exogenous MDP

- Unknown distribution over exogenous inputs (i.e. arrivals)
- Known reward and transition as function of exogenous trace
- Access to historical data of exogenous inputs

VM Allocation



Set of Physical Machines

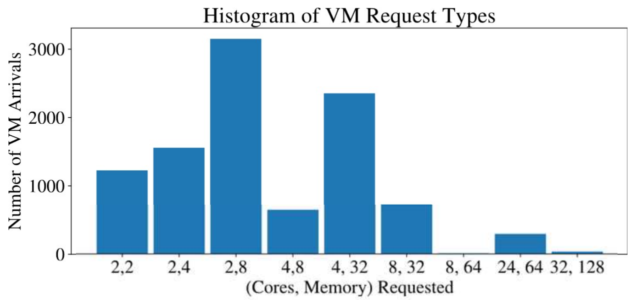
Replays traces from Azure Public Trace Dataset

Solve for optimal sequence of actions in hindsight, use as training signal

Three Tools

Historical Data

Sequence of exogenous variable traces



Planning Oracles

Solve for hindsight optimal action sequence

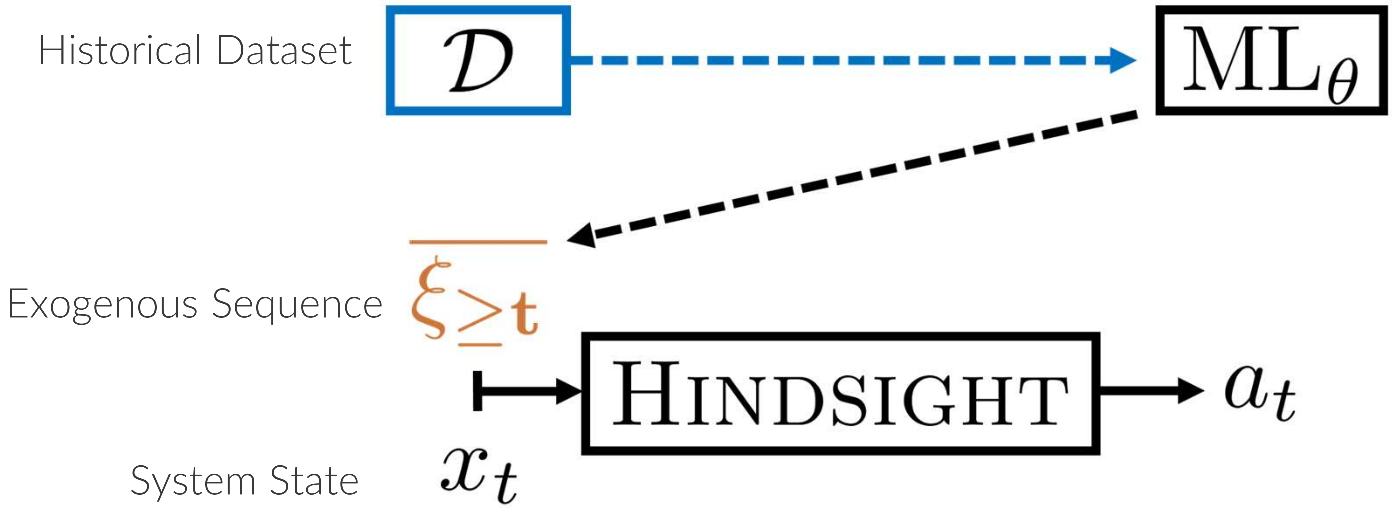
GUROBI OPTIMIZATION

Simulators

High fidelity simulator of business logic

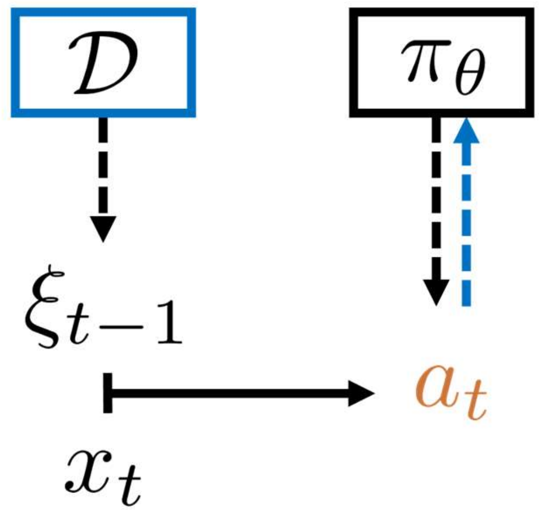
MARO

ML Forecast



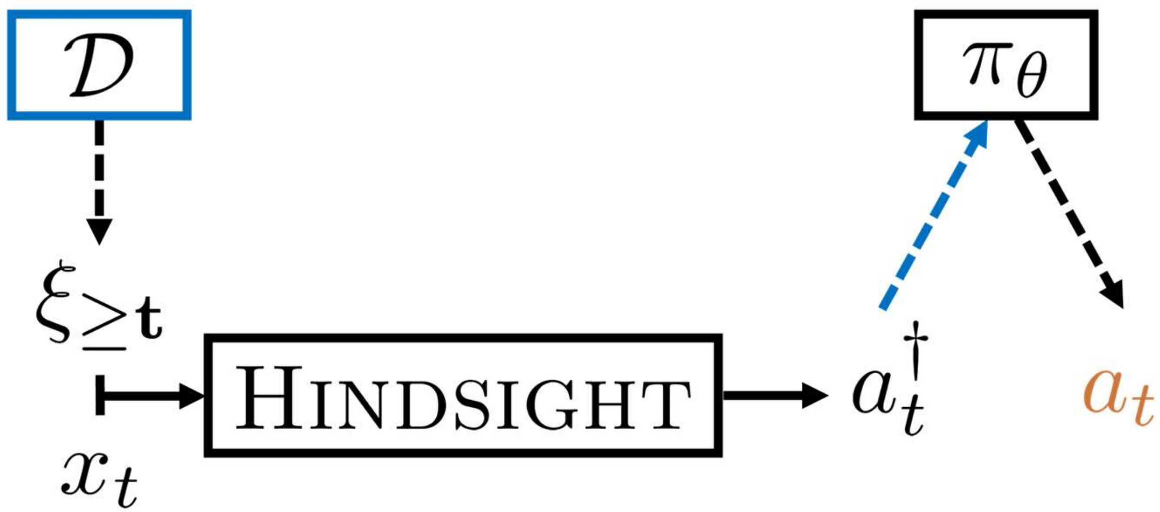
Algorithm Design	Hindsight Data	Planning Oracles	Simulators
ML Forecasting	✓	✓	✗

Tabula RL



Algorithm Design	Hindsight Data	Planning Oracles	Simulators
ML Forecasting	✓	✗	✗
Tabula RL	✓	✗	✓

Hindsight RL



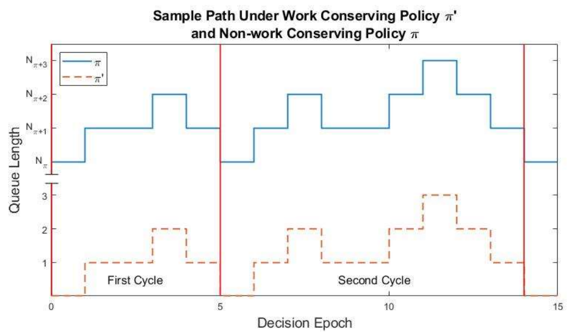
Algorithm Design	Hindsight Data	Planning Oracles	Simulators
ML Forecasting	✓	✓	✗
Tabula RL	✓	✗	✓
Hindsight Learning	✓	✓	✓

Exogenous MDP | Hindsight Planning | Reduction to Experts

Understand algorithmic approaches with hindsight planning, reduction to experts, and supervised learning, for Exogenous MDPs.

Exogenous MDP

Core Feature: Retrospective Planning + Search



M/M/1 Queue with Removable Server

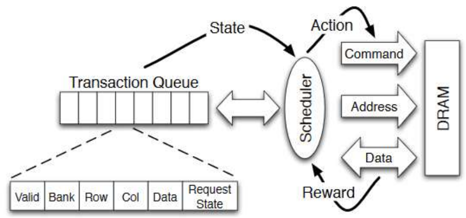
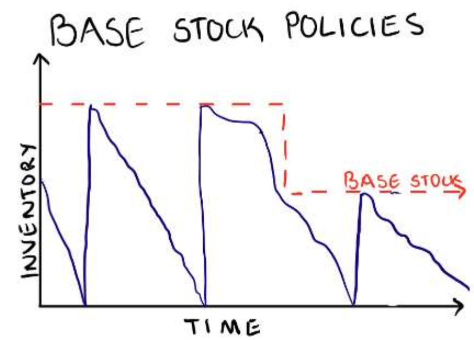


Figure 4: High-level overview of an RL-based scheduler.

Dynamic Memory Controllers



Inventory Control

Exogenous randomness, 'offline' policy calculatable by arrival demand sequence

Exogenous MDP

$$S = X \times \Xi^T$$

State Space Decomposition

'Endogenous' (System State) and 'Exogenous' (Arrival State)

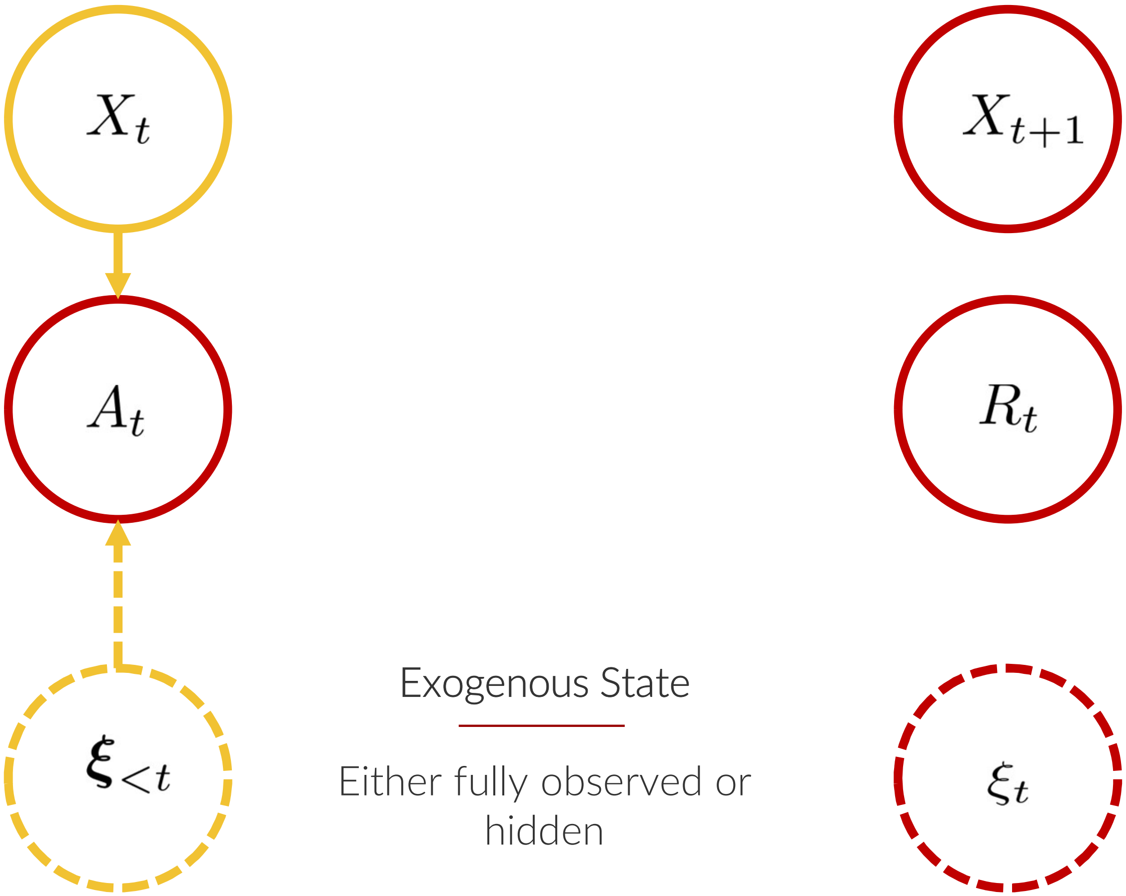
System State (X) - Capacity

Exogenous State (Ξ) – Arrival

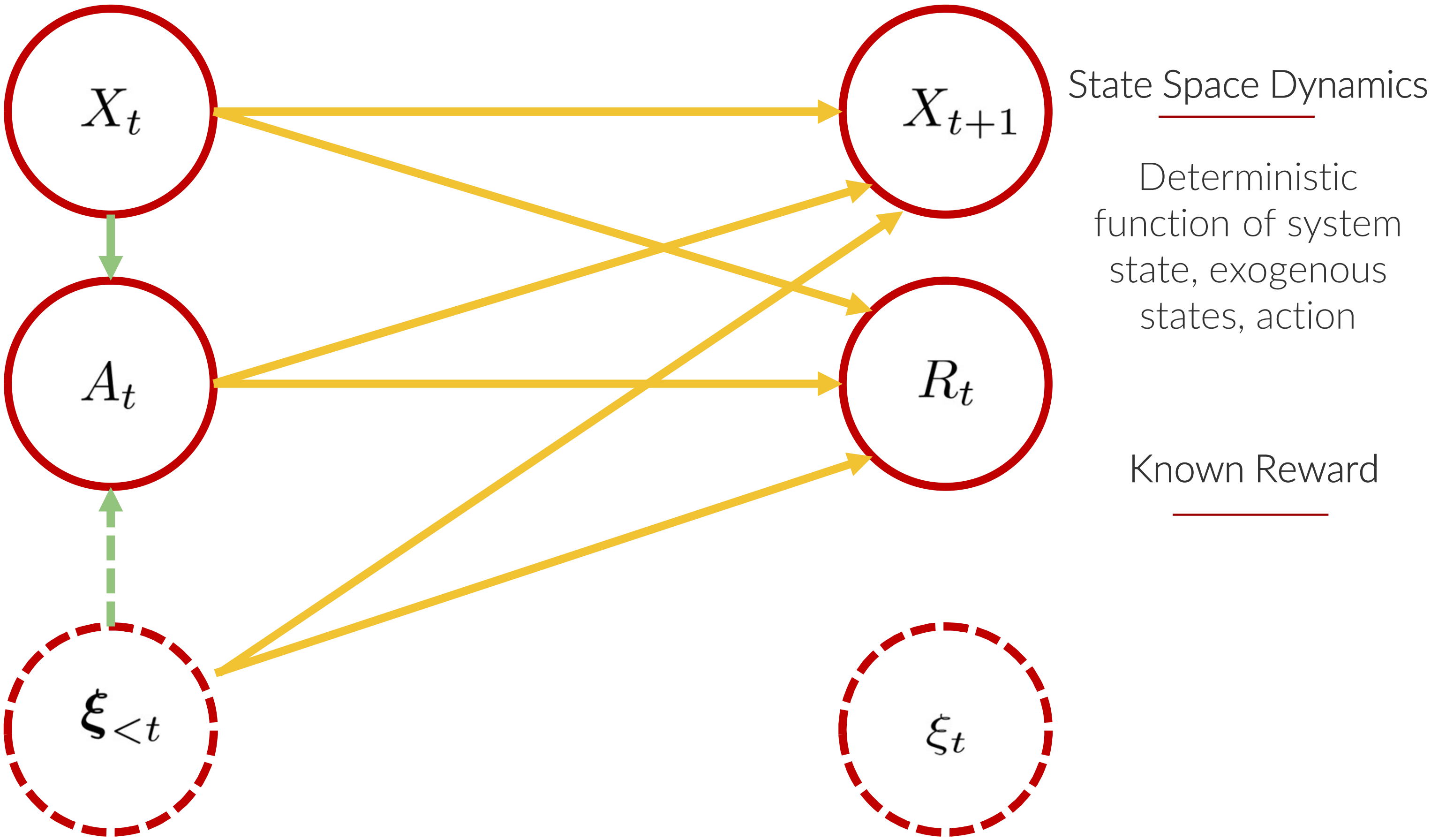
(correlated with prior exogeneity, independent of system state)

Exogenous MDP

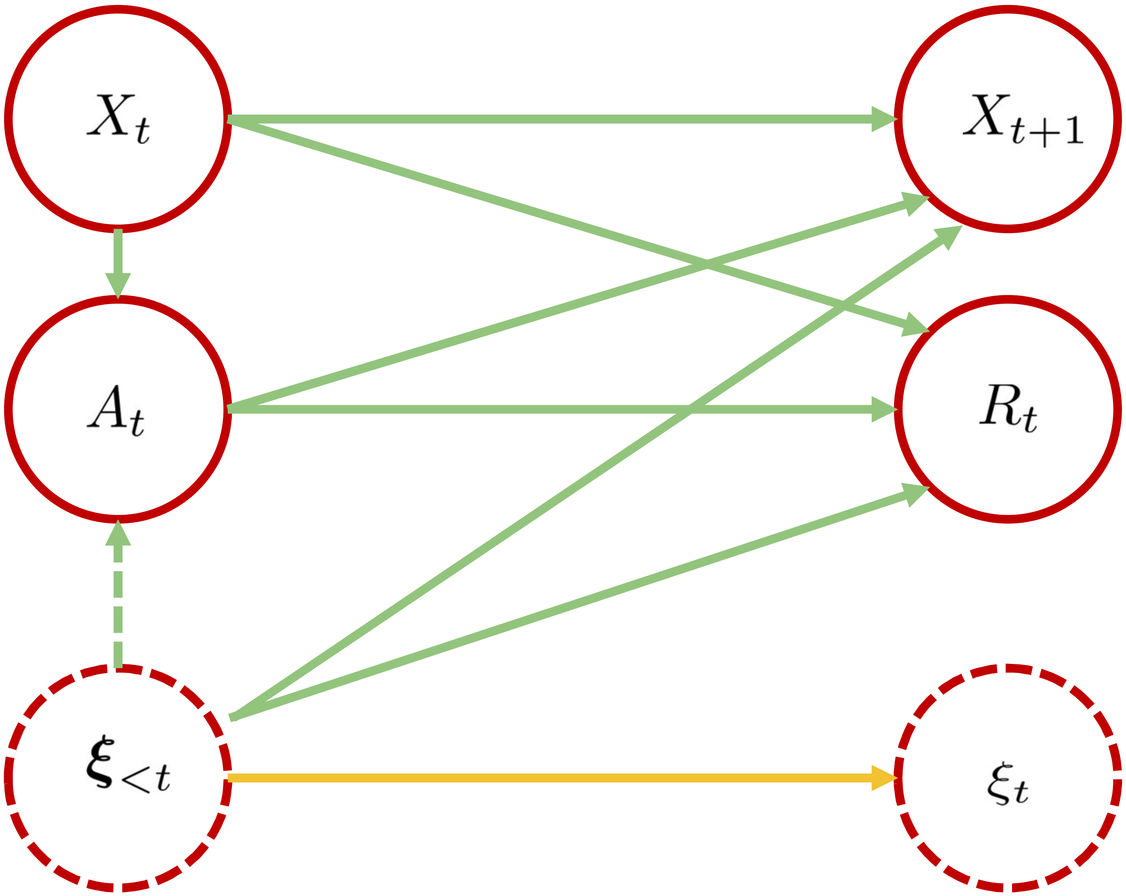
$$\boldsymbol{\xi}_{<t} = \{\xi_1, \ldots, \xi_{t-1}\}$$



Exogenous MDP



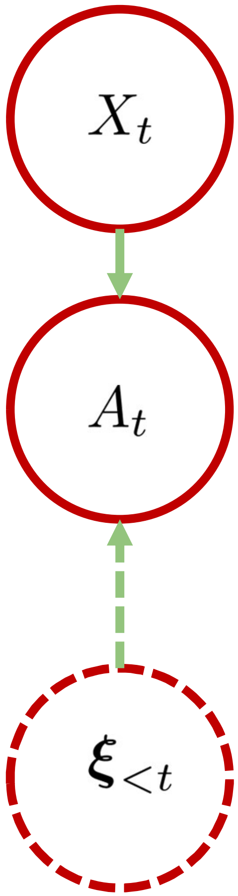
Exogenous MDP



Exogenous Transitions

Exogenous states evolve by independent process with unknown dynamics

Exogenous MDP

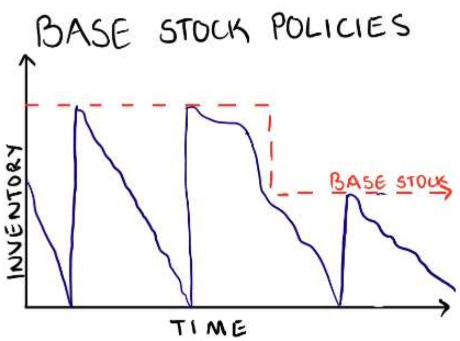


Exogenous Transitions

Exogenous states evolve by independent process with unknown dynamics

Can reduce horizon dependence on exogenous trace if IID, etc.

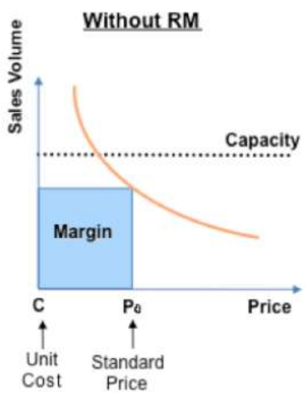
Exogenous MDP



Inventory Control

System state: Inventory levels and outstanding orders

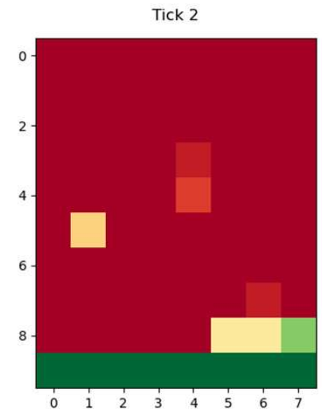
Exogenous state:
Current period demand



Online Revenue Management

System state: Inventory

Exogenous state:
Current period demand type

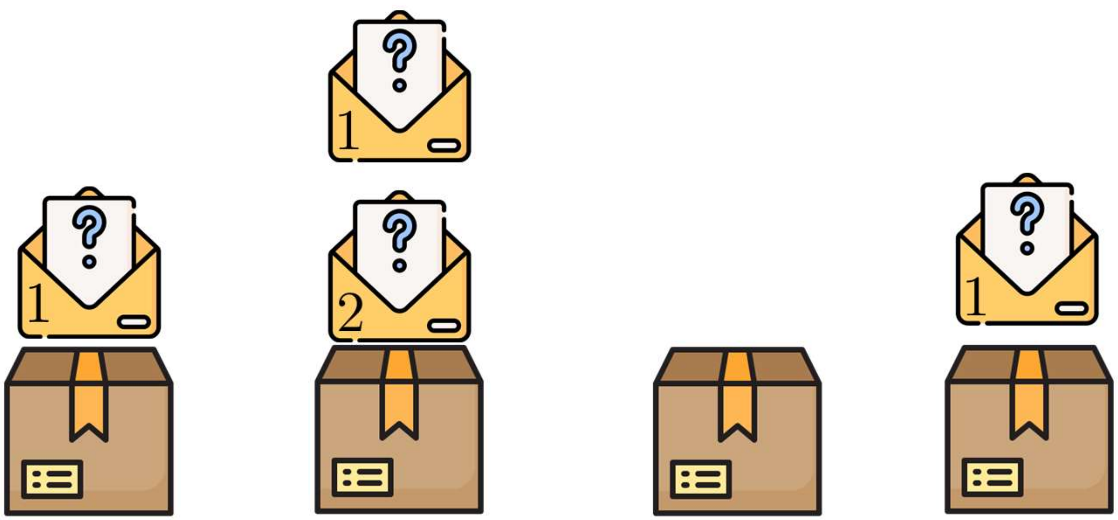


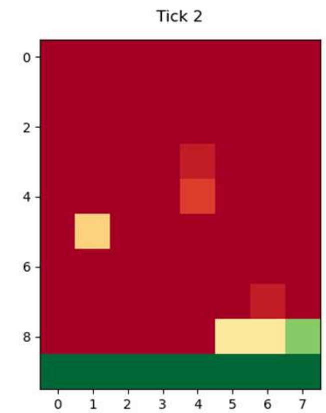
VM Allocation

System state: Current capacity for each physical machine

Exogenous state:
Current period VM arrival type

Exogenous MDP





VM Allocation

System state: Current capacity for each physical machine

Exogenous state:
Current period VM arrival type

Exogenous MDP

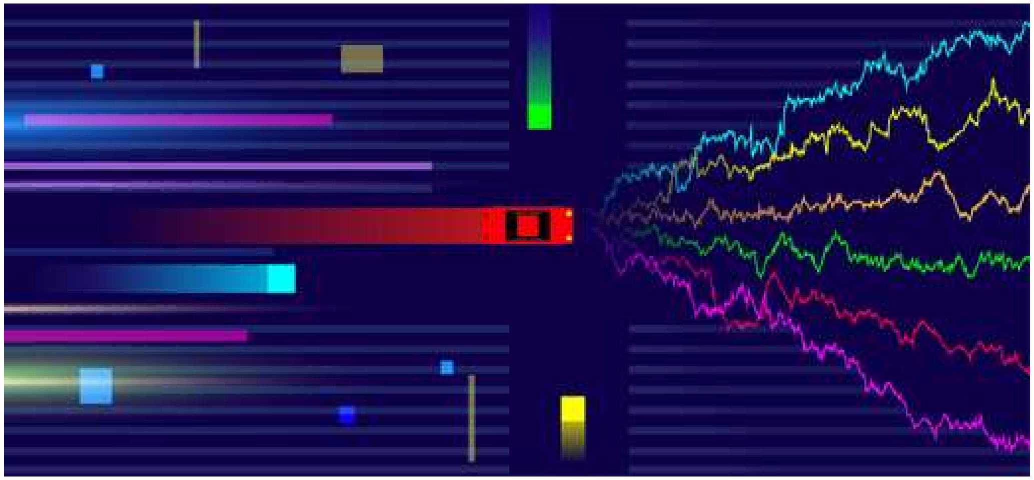
Related to original MDP model with different "information structure"

Exo-MDP	Typical MDP
Known structure of dynamics and rewards	Unknown structure of dynamics and rewards
Unknown distribution on exogenous inputs	Known distribution on exogenous inputs (uniform)

Can do typical "inverse uniform" to go between model, assumes access to "exogenous randomness" in typical MDP

Exogenous MDP

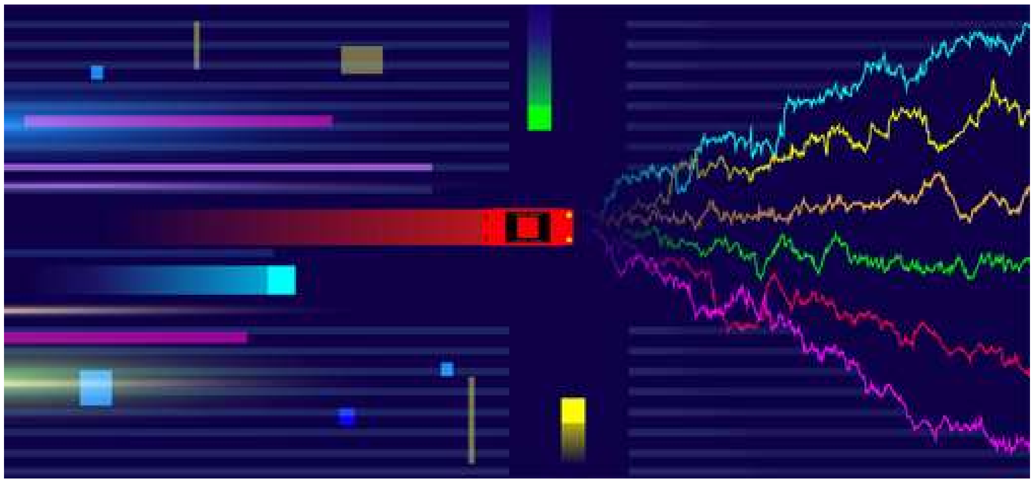
Typically value function has complicated probabilistic structure....



Planning for all possible futures

Exogenous MDP

Typically value function has complicated probabilistic structure….



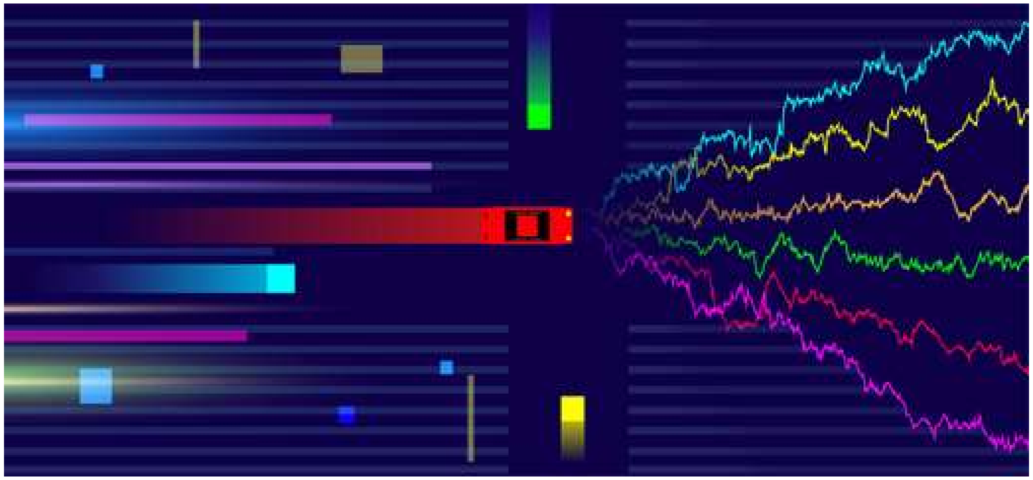
Each trajectory dictated solely by exogenous inputs

Exogenous MDP

"Simulation" Q function:

$$Q_t^\pi(s, a, \boldsymbol{\xi}_{\geq t}) = \mathbb{E}[\sum_{\tau \geq t} r(s_\tau, a_\tau, \xi_\tau) | s_t = s, a_t = a]$$

$$V_t^\pi(s, \boldsymbol{\xi}_{\geq t}) = \sum_a \pi(a|s) Q_t^\pi(s, a, \boldsymbol{\xi}_{\geq t}).$$



$Q_t^\pi(s, a, \boldsymbol{\xi}^1)$

$Q_t^\pi(s, a, \boldsymbol{\xi}^2)$

$Q_t^\pi(s, a, \boldsymbol{\xi}^N)$

Exogenous MDP

"Simulation" Q function:

$$Q_t^\pi(s, a, \boldsymbol{\xi}_{\geq t}) = \mathbb{E}[\sum_{\tau \geq t} r(s_\tau, a_\tau, \xi_\tau) | s_t = s, a_t = a]$$

$$V_t^\pi(s, \boldsymbol{\xi}_{\geq t}) = \sum_a \pi(a|s) Q_t^\pi(s, a, \boldsymbol{\xi}_{\geq t}).$$

Given fixed exogenous inputs, can simulate reward of any policy where only randomness is randomness in policy

$$Q_t^\pi(s, a) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[Q_t^\pi(s, a, \boldsymbol{\xi}_{\geq t})]$$

$$V_t^\pi(s) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[V_t^\pi(s, \boldsymbol{\xi}_{\geq t})].$$

Exogenous MDP

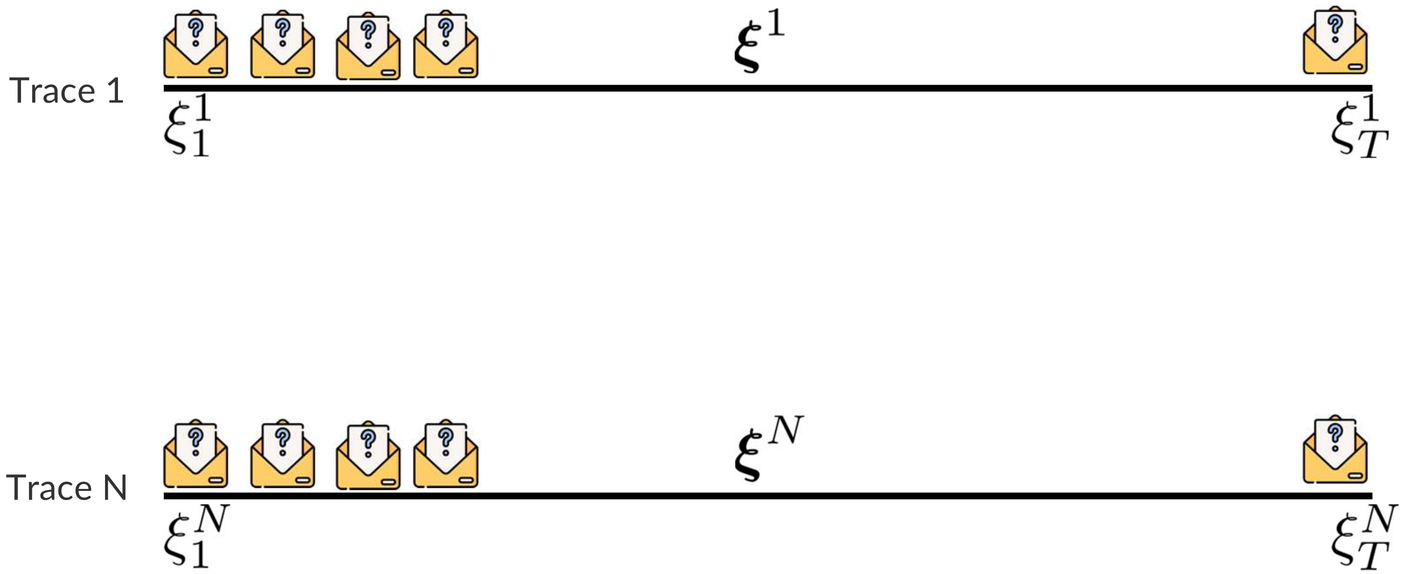
Assume access to a dataset $\mathcal{D} = \{\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^N\}$

Goal: Minimize regret: $\textsc{Regret}(\pi) = V_1^*(s_0) - V_1^\pi(s_0)$

Exogenous MDP

Assume access to a dataset $\mathcal{D} = \{\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^N\}$

Goal: Minimize regret: $\textsc{Regret}(\pi) = V_1^*(s_0) - V_1^\pi(s_0)$



Recall................

Maybe a better model....

Exogenous MDP

- Unknown distribution over exogenous inputs (i.e. arrivals)
- Known reward and transition as function of exogenous trace
- Access to historical data of exogenous inputs

Related Work: Variance Reduction

Typical RL algorithms use the following update step:

$$\nabla J(\theta) = \mathbb{E}[\nabla_\theta \log(\pi_\theta(a \mid s))(Q^{\pi_\theta}(s,a) - b(s))]$$
$$s = (x, \xi_{<t})$$

Current policy

Q estimates

State dependent baseline

Not exploiting known optimal structure as function of future exogenous trace

Exogenous dependent baseline

Can also use baseline depending on entire sequence of exogenous process

$$\nabla J(\theta) = \mathbb{E}[\nabla_\theta \log(\pi_\theta(a \mid s))(Q^{\pi_\theta}(s,a) - b(s, \boldsymbol{\xi}))]$$
$$s = (x, \xi_{<t})$$

[Mao2018, Mesnard2020]

Related Work: Value Decomposition

Application of the Bellman relationship shows that

$$V(x, \xi_{<t}) = \max_a r(x, \xi_{<t}, a) + \gamma \mathbb{E}[V(x', \xi \leq t)]$$

But if the reward function decomposes via

$$r(x, \xi, a) = r_1(x, \xi, a) + r_2(\xi)$$

Bellman equations also decompose into:

$$V^{exo}(\xi) = r_2(\xi) + \gamma \mathbb{E}[V^{exo}(\xi')]$$

$$V^{endo}(x, \xi, a) = r_1(x, \xi, a) + \gamma \mathbb{E}[V^{endo}(x', \xi')]$$

[Dietterich2018, Bray2019]

Related Work: Value Decomposition

Bellman equations also decompose into:

$$V^{exo}(\xi) = r_2(\xi) + \gamma \mathbb{E}[V^{exo}(\xi')]$$

$$V^{endo}(x, \xi, a) = r_1(x, \xi, a) + \gamma \mathbb{E}[V^{endo}(x', \xi')]$$

Linearity doesn't hold for all OR models (and VM allocation metrics)

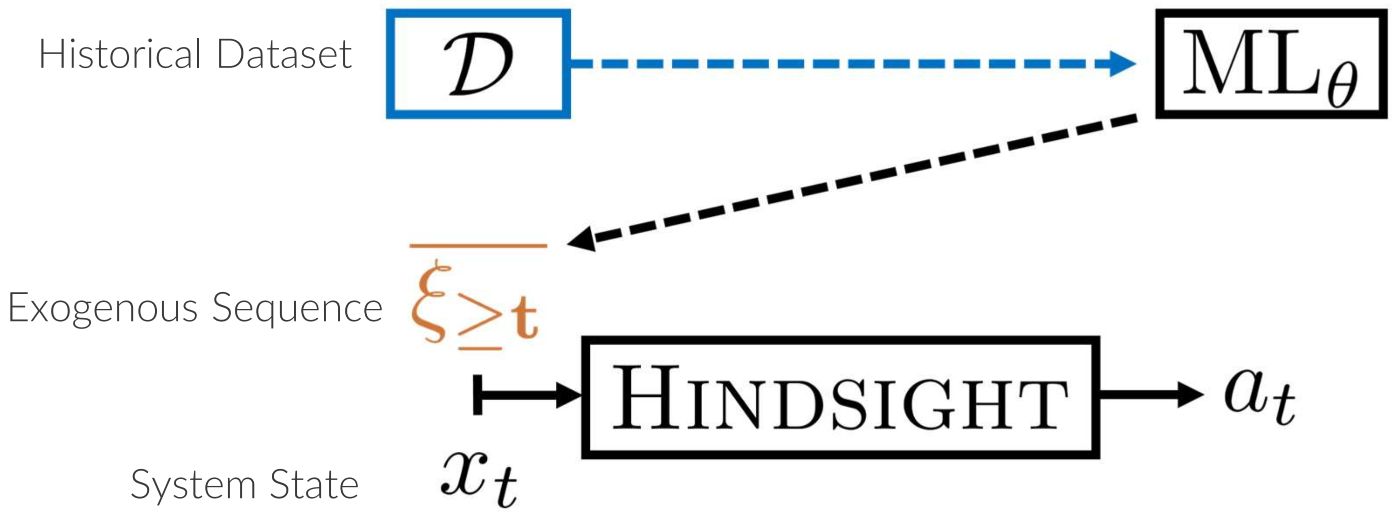
- Dimensionality reduction in eliminating spurious exogenous variables
- Reduces statistical and computational complexity for value iteration based techniques

[Dietterich2018, Bray2019]

Algorithm Design

Algorithm Design	Hindsight Data	Planning Oracles	Simulators
ML Forecasting	✓	✓	✗
Tabula RL	✓	✗	✓
Hindsight Learning	✓	✓	✓

Algorithm Design



Algorithm Design	Hindsight Data	Planning Oracles	Simulators
ML Forecasting	✓	✓	✗
Tabula RL	✓	✗	✓
Hindsight Learning	✓	✓	✓

ML Forecast

Only source of uncertainty is the distribution of exogenous Markov chain (Ξ)

Why not estimate P_Ξ, plug in estimate into Bellman equations, and solve for optimal policy in modified MDP?

Estimated exogenous distribution

Known transitions

$$\overline{Q}_h^*(s,a) = r_h(s,a) + \mathbb{E}_{\xi_t \sim \overline{P_\Xi(\cdot|\boldsymbol{\xi}_{<t})}}[\max_{a'} \overline{Q}_{h+1}^*(f(s,a,\xi_t),a')]$$

$$\overline{\pi}_h^*(s) = \operatorname*{argmax}_a \overline{Q}_h^*(s,a)$$

ML Forecast

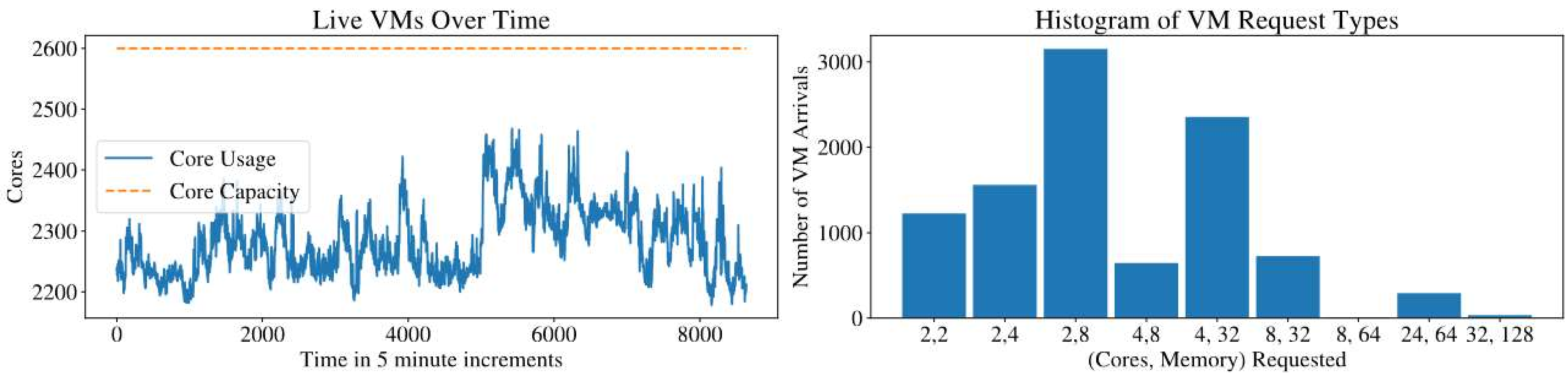
Only source of uncertainty is the distribution of exogenous Markov chain ($\boldsymbol{\Xi}$)

Why not estimate $\boldsymbol{P}_{\boldsymbol{\Xi}}$, plug in estimate into Bellman equations, and solve for optimal policy in modified MDP?

- Full planning is computationally costly
- Statistical complexity necessarily scales with difficulty in estimating distribution of Markov chain

ML Forecast

- Full planning is computationally costly
- Statistical complexity necessarily scales with difficulty in estimating distribution of Markov chain



ML Forecast

Efficient computationally in settings with strong predictors for exogenous distribution

If the exogenous process is IID, and use $\overline{P_{\Xi}}$ as empirical distribution, then with high probability

$$\textsc{Regret}(\pi) \leq 2T^2 \sqrt{\frac{\log(2|\Xi|\delta)}{N}}$$

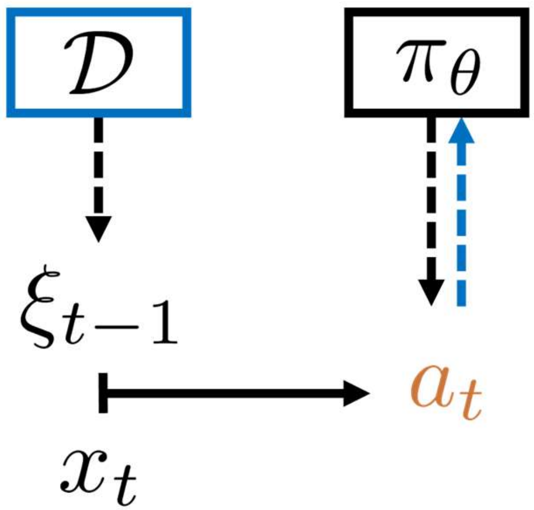
But under correlation, error scales *linearly* with approximation error

ML Forecast

Efficient computationally in settings with strong predictors for exogenous distribution

Failed miserably in VM allocation

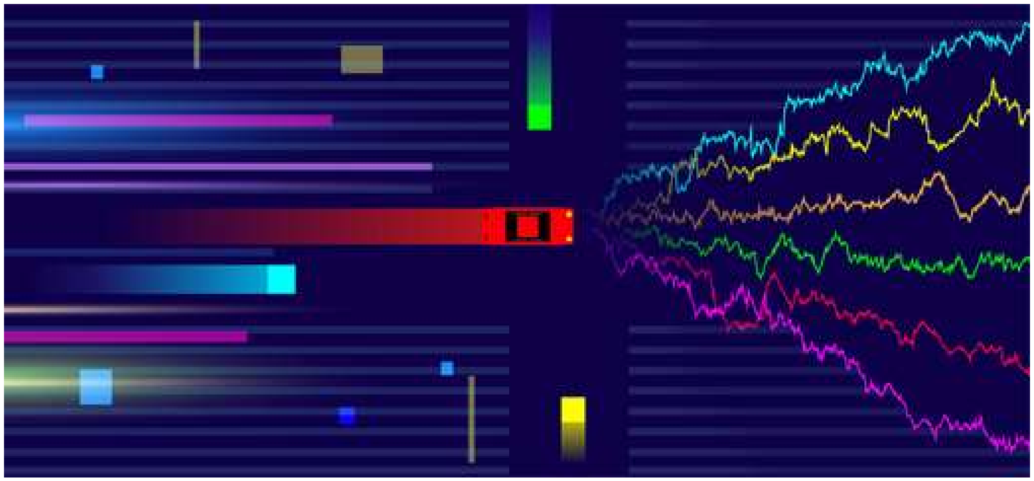
Tabula RL



Algorithm Design	Hindsight Data	Planning Oracles	Simulators
ML Forecasting	✓	✓	✗
Tabula RL	✓	✗	✓

Tabula RL

"Simulation" Q function:



$$Q_t^\pi(s, a, \boldsymbol{\xi}^1)$$

$$Q_t^\pi(s, a, \boldsymbol{\xi}^2)$$

$$Q_t^\pi(s, a, \boldsymbol{\xi}^N)$$

Given historical dataset, can simulate value of any policy on given trace:

- Unbiased for true value
- No distribution shift
- Reduces to supervised learning

Tabula RL

Given historical dataset, can simulate value of any policy on given trace:

- Unbiased for true value
- No distribution shift
- Reduces to supervised learning

Converges to optimal policy in empirical MDP where exogenous dynamics replaced with empirical on dataset

$$\overline{P}_{\Xi} = \frac{1}{N} \sum_i \delta_{\boldsymbol{\xi}^i}$$

Tabula RL

Can maximize empirical return directly, similar to ERM strategy of supervised learning

$$\overline{\pi} = \arg\max_{\pi \in \Pi} \overline{\mathbb{E}}[V_1^{\pi}(s_0, \boldsymbol{\xi})]$$

Policy Class

Simulated value

Empirical over historical dataset

With high probability

$$\textsc{Regret}(\overline{\pi}) \leq T\sqrt{\frac{2\log(2|\Pi|/\delta)}{N}}$$

Tabula RL

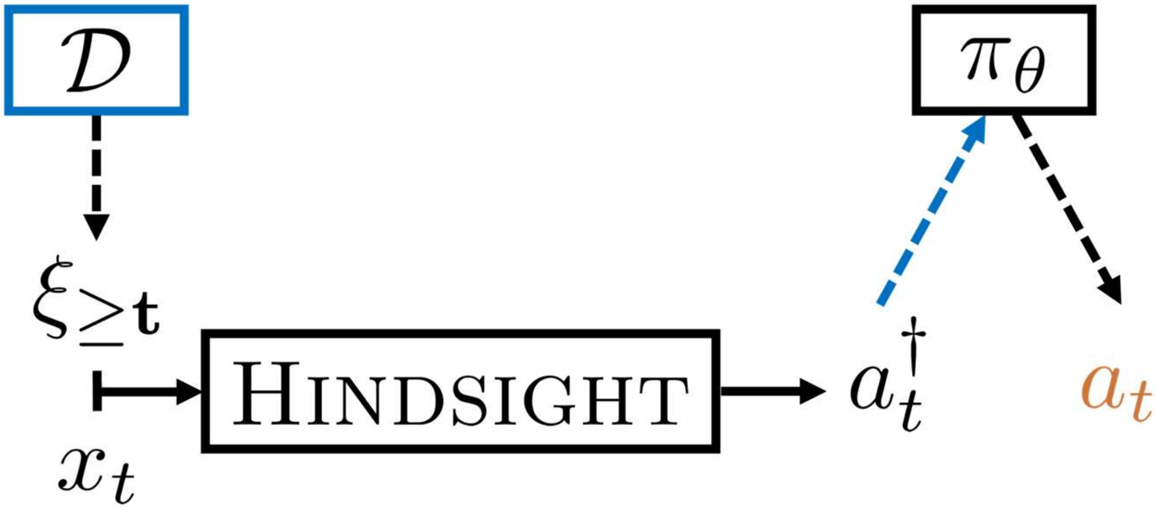
With high probability

$$\textsc{Regret}(\overline{\pi}) \leq T\sqrt{\frac{2\log(2|\Pi|/\delta)}{N}}$$

Proper dependence on time horizon

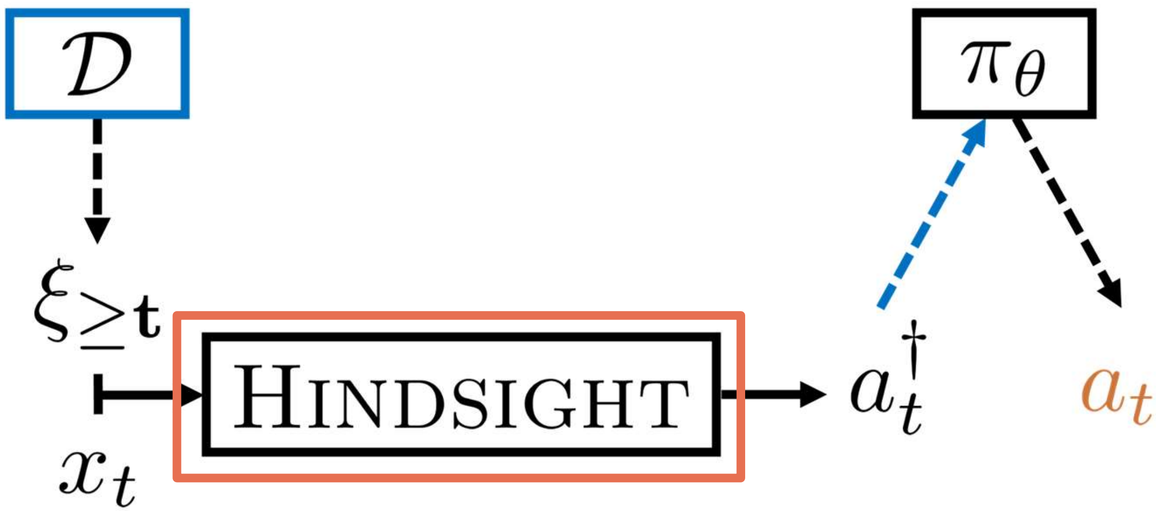
Convergence to optimal policy is idealized assumptions, hides optimization issues when studying statistical guarantees

Hindsight Planning



Algorithm Design	Hindsight Data	Planning Oracles	Simulators
ML Forecasting	✓	✓	✗
Tabula RL	✓	✗	✓
Hindsight Learning	✓	✓	✓

Hindsight Planning



Hindsight Planning

Solve for hindsight optimal sequence of actions for fixed exogenous trace

Given any trace $\boldsymbol{\xi}_{\geq t} = (\xi_t, \ldots, \xi_T)$ and state $s = (x_t, \boldsymbol{\xi}_{<t})$ we can solve:

$$\begin{aligned} \max_{a_t, \ldots, a_T} \ & \textstyle\sum_{\tau=t}^{T} r(s_\tau, a_\tau, \xi_\tau) \\ \text{s.t.} \ & x_{\tau+1} = f(s_\tau, a_\tau, \xi_\tau), \ \text{for } \tau = t, \ldots, T \\ & s_\tau = (x_\tau, \boldsymbol{\xi}_{<\tau}), \ \text{for } \tau = t, \ldots, T. \end{aligned}$$

Denote objective value as $\textsc{Hindsight}(t, \boldsymbol{\xi}_{\geq t}, s)$.

Hindsight Planning

Given a fixed exogenous trace, the optimal policy can be solved via a (combinatorial) optimization problem

$$\textsc{Hindsight}(t, \boldsymbol{\xi}_{\geq t}, s) = \max_{a_t, \ldots, a_T} \sum_{\tau=t}^{T} r(s_\tau, a_\tau, \xi_\tau)$$

$$\text{s.t. } x_{\tau+1} = f(s_\tau, a_\tau, \xi_\tau), \text{ for } \tau = t, \ldots, T$$

$$s_\tau = (x_\tau, \boldsymbol{\xi}_{<\tau}), \text{ for } \tau = t, \ldots, T.$$

$$= V_t^\dagger(s, \boldsymbol{\xi}_{\geq t})$$

Whatever cumulative reward function of interest

Capacity constraints on physical machines

"Optimistic Value Function"

[Vera2018,2019]

Related Work: Variance Reduction

Exogenous dependent baseline

Can also use baseline depending on entire sequence of exogenous process

$$\nabla J(\theta) = \mathbb{E}[\nabla_\theta \log(\pi_\theta(a \mid s))(Q^{\pi_\theta}(s,a) - b(s, \boldsymbol{\xi}))]$$
$$s = (x, \xi_{<t})$$

Not exploiting known optimal structure as function of future exogenous trace

$$b(s, \boldsymbol{\xi}) = V_t^{\dagger}(s, \boldsymbol{\xi}_{\geq t})$$

Failed miserably in VM allocation

[Mao2018, Mesnard2020]

Bayes Selector

Given a fixed exogenous trace, the optimal policy can be solved via a (combinatorial) optimization problem

$$\textsc{Hindsight}(t, \boldsymbol{\xi}_{\geq t}, s) = \max_{a_t,\ldots,a_T} \sum_{\tau=t}^{T} r(s_\tau, a_\tau, \xi_\tau)$$
$$\text{s.t. } x_{\tau+1} = f(s_\tau, a_\tau, \xi_\tau), \text{ for } \tau = t, \ldots, T$$
$$s_\tau = (x_\tau, \boldsymbol{\xi}_{<\tau}), \text{ for } \tau = t, \ldots, T.$$

Can develop an online policy:

- In current state, solve optimization problem replacing unknown trace with historical traces
- Execute policy by averaging over decisions aggregated for current exogenous state

Requires frequent ***online*** resolves of an IP

[Vera2018,2019]

Bayes Selector

Solving for optimal non-anticipatory policy is hard, focus on a surrogate

$$\pi_t^\dagger(s) = \operatorname*{argmax}_{a \in \mathcal{A}} Q_t^\dagger(s,a)$$
$$Q_t^\dagger(s,a) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[r(s,a,\xi_t) + \textsc{Hindsight}(t+1, \boldsymbol{\xi}_{>t}, f(s,a,\xi_t))]$$
$$V_t^\dagger(s) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[\textsc{Hindsight}(t, \boldsymbol{\xi}_{\geq t}, s)]$$

Pick actions “optimal on average” over exogenous traces

Bayes Selector

Solving for optimal non-anticipatory policy is hard, focus on a surrogate

$$\pi_t^\dagger(s) = \operatorname*{argmax}_{a \in \mathcal{A}} Q_t^\dagger(s, a)$$
$$Q_t^\dagger(s, a) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[r(s, a, \xi_t) + \textsc{Hindsight}(t + 1, \boldsymbol{\xi}_{>t}, f(s, a, \xi_t))]$$
$$V_t^\dagger(s) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[\textsc{Hindsight}(t, \boldsymbol{\xi}_{\geq t}, s)]$$

Theorem: In many OR problems of interest, we have that:

$$\textsc{Regret}(\pi^\dagger) \leq O(1)$$

Bin packing, knapsack, revenue management, ...

Bayes Selector

Theorem: In many OR problems of interest, we have that:

$$\textsc{Regret}(\pi^\dagger) \leq O(1)$$

Bin packing, knapsack, revenue management, ...

Requires:

- Exchangeability of actions
- Lipschitzness of value function and exogenous sequence

Bayes Selector

Using techniques from "Compensated Coupling" show:

Theorem: In an arbitrary Exo-MDP we have that:

$$\textsc{Regret}(\pi^\dagger) \leq \sum_{t=1}^{T} \mathbb{E}_{S_t \sim \Pr_t^{\pi^\dagger}}[\Delta_t^\dagger(S_t)] \text{ where}$$

$$\Delta_t^\dagger(s) = Q_t^\dagger(s, \pi^\dagger(s)) - Q_t^\star(s, \pi^\dagger(s)) + Q_t^\star(s, \pi^\star(s)) - Q_t^\dagger(s, \pi^\star(s))$$

True regardless of underlying problem, issue potentially scales with time horizon for arbitrary problem

Bayes Selector

$$\pi_t^\dagger(s) = \operatorname*{argmax}_{a\in\mathcal{A}} Q_t^\dagger(s,a)$$
$$Q_t^\dagger(s,a) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[r(s,a,\xi_t) + \textsc{Hindsight}(t+1, \boldsymbol{\xi}_{>t}, f(s,a,\xi_t))]$$
$$V_t^\dagger(s) = \mathbb{E}_{\boldsymbol{\xi}_{\geq t}}[\textsc{Hindsight}(t, \boldsymbol{\xi}_{\geq t}, s)]$$

Cons:
- Requires frequent online resolves of an IP (infeasible for large-scale system applications)
- Generalization to unobserved states, non-robust to outliers in exogenous traces

Pros:
- Easier to learn from data (statistically)
- Allows for imitation learning reduction for computational gains

Reduction to Experts

Iteratively use these values of $Q_t^\dagger(s, a, \boldsymbol{\xi}_{>t})$ collected across a trajectory from the actor from current parameters, update policy using either:

Q Network Distillation

$$Q_\theta = \operatorname{argmin} \sum_{x,u,a} (Q_\theta(x,u,a) - Q^\dagger(s,a,\boldsymbol{\xi}_{>t}))^2$$

$$\pi_\theta(x,u) = \operatorname{argmax} Q_\theta(x,u,a)$$

Depends on underlying "realizability"

Mean Actor Critic

$$\nabla J(\theta) = \mathbb{E}[\sum_a \nabla_\theta \pi_\theta(a \mid x,u) Q^\dagger(s,a,\boldsymbol{\xi}_{>t})]$$

Reduction to Experts

Theorem: In an arbitrary Exo-MDP we have that:

$$\textsc{Regret}(\pi^\dagger) \leq \sum_{t=1}^{T} \mathbb{E}_{S_t \sim \Pr_t^{\pi^\dagger}}[\Delta_t^\dagger(S_t)] + 2T\sqrt{\frac{2\log(2|\Pi|/\delta)}{N}} + o(1)$$

Hindsight Bias

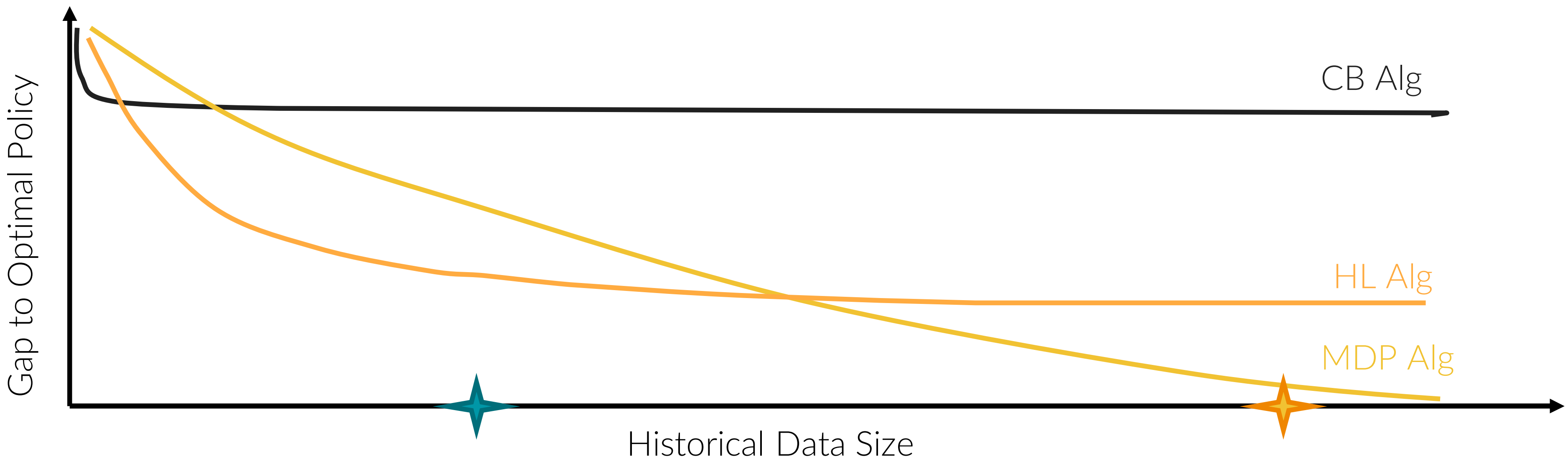
SAA

Imitation Learning regret (typically 1/N)

Reduction to Experts

Theorem: In an arbitrary Exo-MDP we have that:

$$\textsc{Regret}(\pi^\dagger) \leq \sum_{t=1}^{T} \mathbb{E}_{S_t \sim \Pr_t^{\pi^\dagger}} [\Delta_t^\dagger(S_t)] + 2T\sqrt{\frac{2\log(2|\Pi|/\delta)}{N}} + o(1)$$



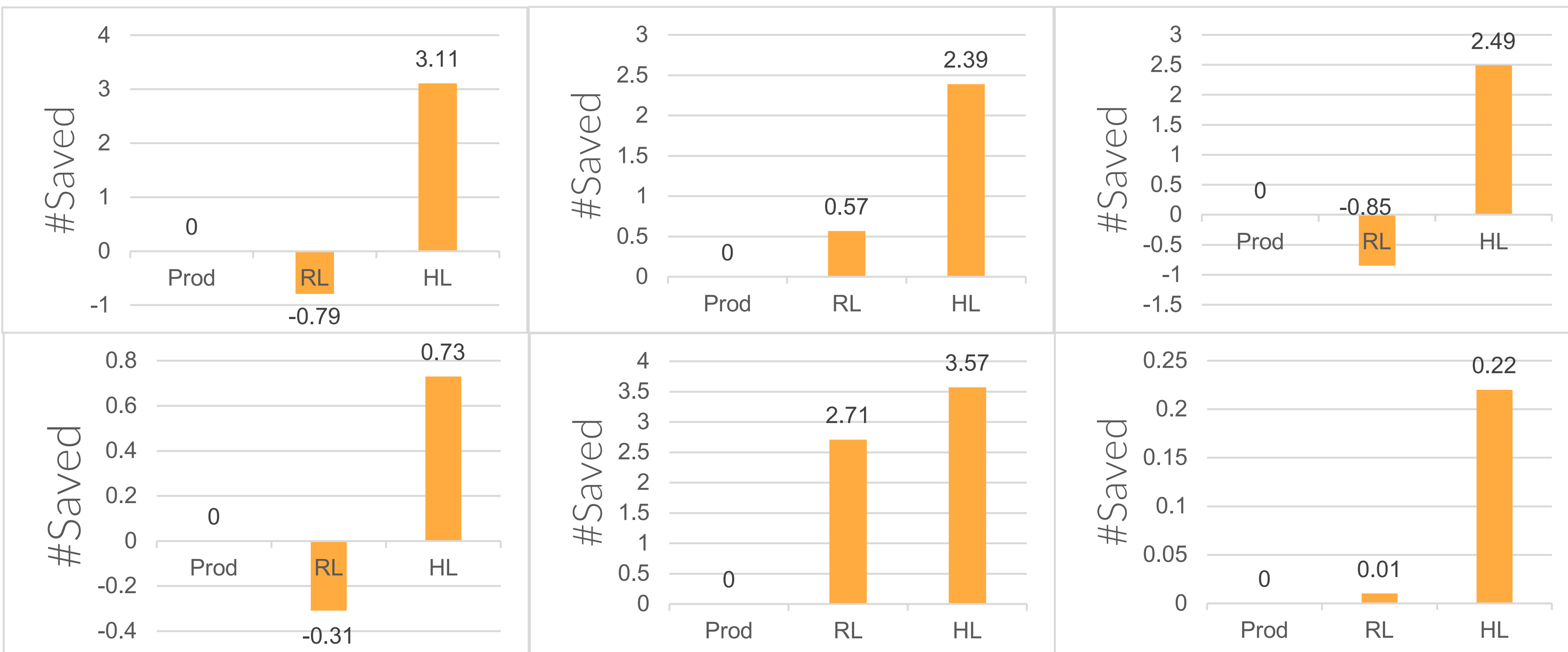
Simulation Results

- Trained via 25 day trace, tested on independent 1 day VM trace
- Compare algorithm vs “best fit” production heuristic

Algorithm	PMs Saved
Performance Upper Bound	4.95
DQN	-0.64
MAC	-0.51
Guided DQN	3.71
Guided MAC	4.33

Simulation Results

Results on Microsoft Azure high-fidelity simulators



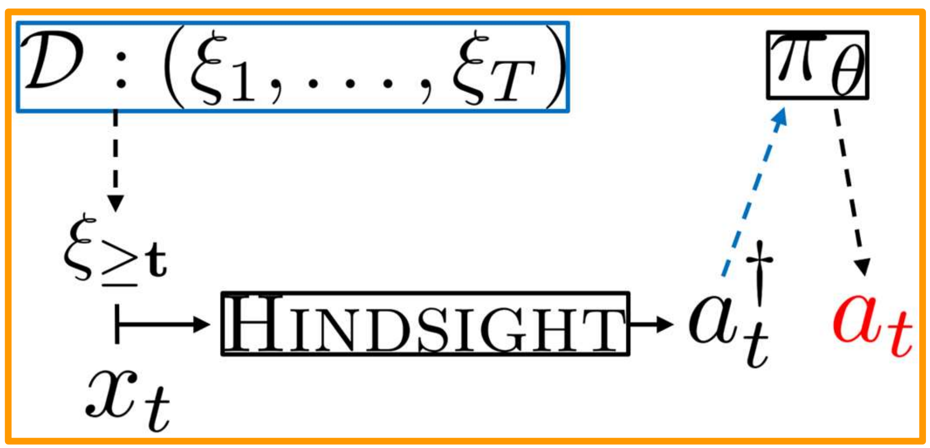
Open Questions

- Show explicit computational + theoretical improvements over Sim2Real RL
- Formalize reduction to imitation learning (i.e. computational + statistical improvements under strongly convex rewards, etc)
- Implement and test in other domains (caching, etc)

RL for Operations, 2022

RL in MDPs with Exogenous Inputs

Sean Sinclair,
Cornell University



Plan for Today

Nonparametric RL

- "Nonparametric" function approximation
- Strong guarantees across:
  Sample complexity, *space complexity*, *storage complexity*

Hindsight Learning

- Exogenous MDPs as model for OR problems
- Use of *Hindsight Planning* oracle for algorithm design
- Empirical results in VM allocation with Microsoft Azure

Tree-Partitions

- Implement tree-based adaptive discretization from nonparametric RL algorithms
- Use ORSuite to test on "continuous Ambulance routing"

Hindsight Planning for Exo-MDPs

- Use ORSuite model for revenue management and pricing (an example of an Exo-MDP)
- Implement Bayes Selector
- Use ORSuite to run simulations to compare performance against tabular algorithms

References

[Sinclair2022] Sean R. Sinclair, Felipe Frujeri, Ching-An Cheng, Adith Swaminathan. “Hindsight Learning for MDPs with Exogenous Inputs.“ *Under Review*, 2022.

[Mao2018] Hongzi Mao et al. “Variance Reduction for Reinforcement Learning in Input-Driven Environments”. *ICLR*, 2019.

[Mesnard2020] Thomas Mesnard et al. “Counterfactual Credit Assignment in Model-Free Reinforcement Learning.” *ICML*, 2021.

[Dietterich2018] Thomas Dietterich, George Trimponias, Zhitang Chen. “Discovering and Removing Exogenous State Variables and Rewards for Reinforcement Learning.” *ICML*, 2018.

[Bray2019] Robert L. Bray. “Markov Decision Processes with Exogenous Variables.” *Management Science*, 2019.

References

[Vera2018] Alberto Vera, Siddhartha Banerjee. "The Bayesian Prophet: A Low-Regret Framework for Online Decision Making." *SIGMETRICS*, 2019.
[Vera2019] Alberto Vera, Siddhartha Banerjee, Itai Gurvich. "Online Allocation and Pricing: Constant Regret via Bellman Inequalities". *Operations Research*, 2020.

Email srs429@cornell.edu for more extensive references + lit review summary