Learning to Optimize as Policy Learning

Yisong Yue

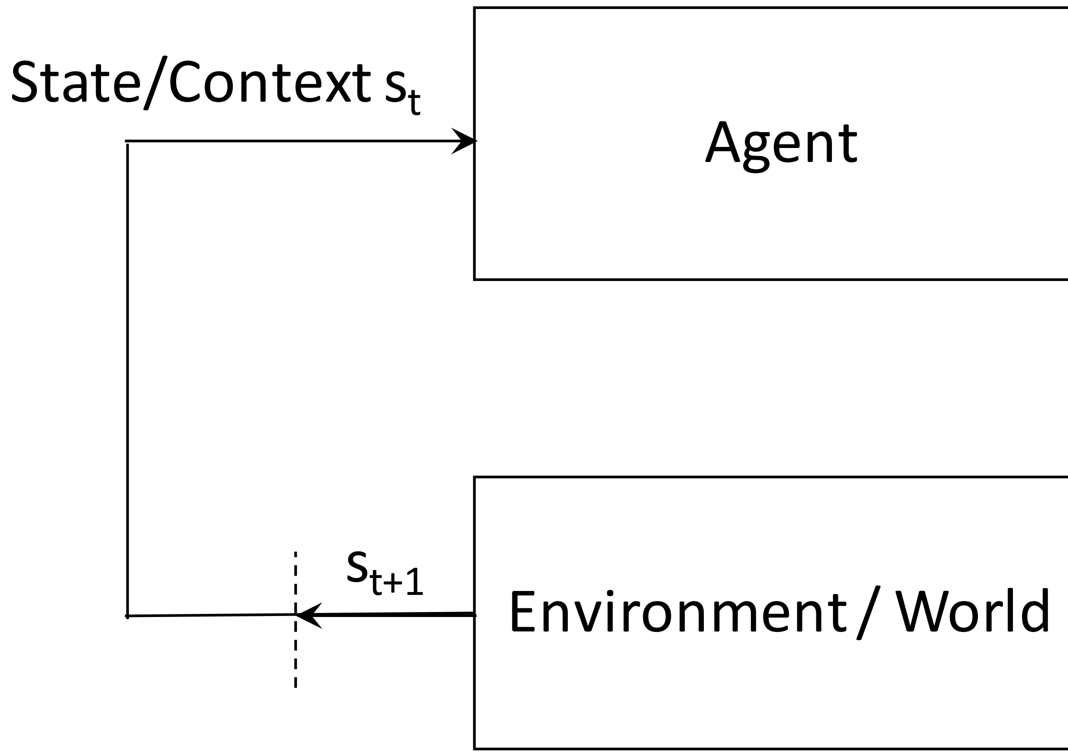
Policy Learning (Reinforcement & Imitation)

Goal: Find "Optimal" Policy

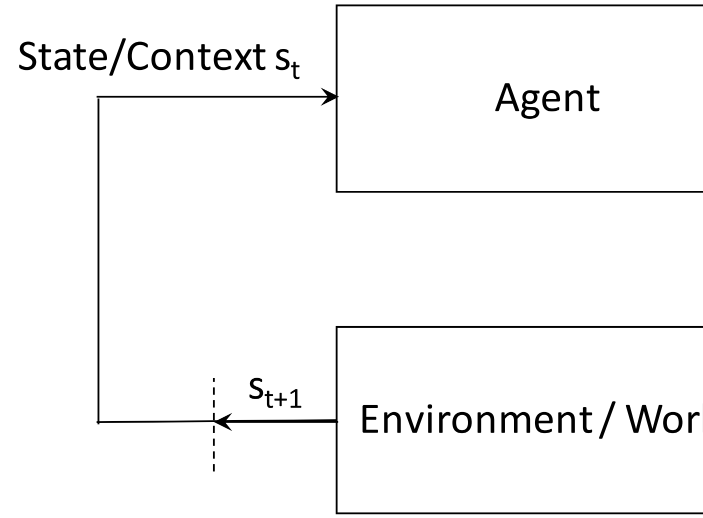
Imitation Learning:
Optimize imitation loss

Reinforcement Learning:
Optimize environmental reward

Learning-based Approach for Sequential Decision Making



Basic Formulation



(Typically a Neural Net)

- Policy: $\pi(s) \rightarrow P(a)$

State

Action

- Roll-out: $\tau = \langle s_0, a_0, s_1, a_1, s_2, \ldots \rangle$

(aka trace or trajectory)

Transition Function: P(s' | s,a)

- Objective: $\sum_i r(s_i, a_i)$

Optimization as Sequential Decision Mak

- Many Solvers are Sequential
  - Tree-Search
  - Greedy
  - Gradient Descent

- Can view solver as “agent” or “policy”
  - State = intermediate solution
  - Find a state with high reward (solution)
  - **Learn better local decision making**

- Formalize Learning
  - Builds upon mo
- Theoretical Analysi
- Interesting Algorith

Example #1: Learning to Search (Discrete)

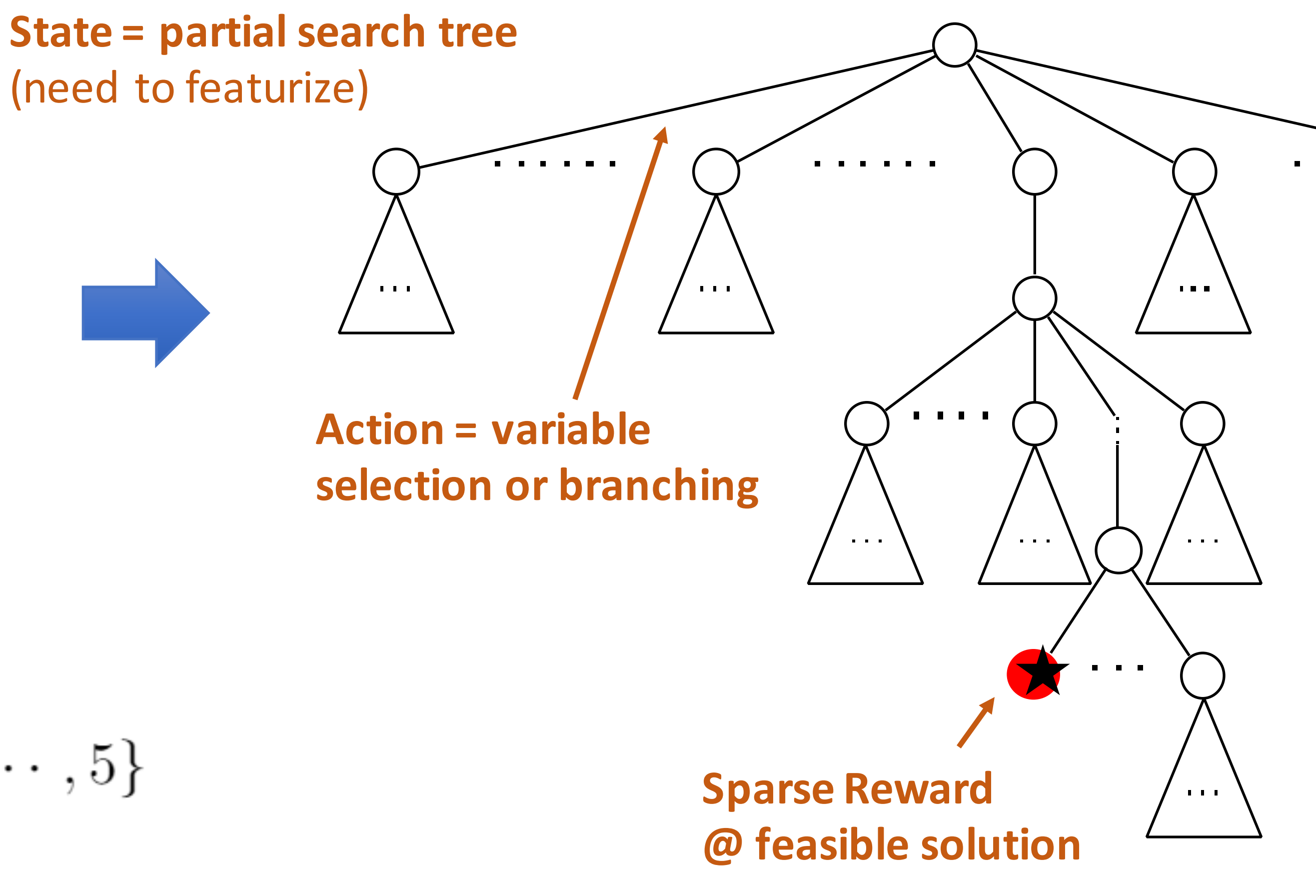
Integer Program

$$\max \; -\sum_{i=1}^{5} x_i,$$

subject to:

$$x_1 + x_2 \geq 1,$$
$$x_2 + x_3 \geq 1,$$
$$x_3 + x_4 \geq 1,$$
$$x_3 + x_5 \geq 1,$$
$$x_4 + x_5 \geq 1,$$
$$x_i \in \{0, 1\}, \forall i \in \{1, \cdots, 5\}$$

Tree-Search (Branch and Bou



[He et al., 2014][Khalil et al., 2016] [Song et al., arXiv]

Example #1: Learning to Search (Discrete)

Integer Program

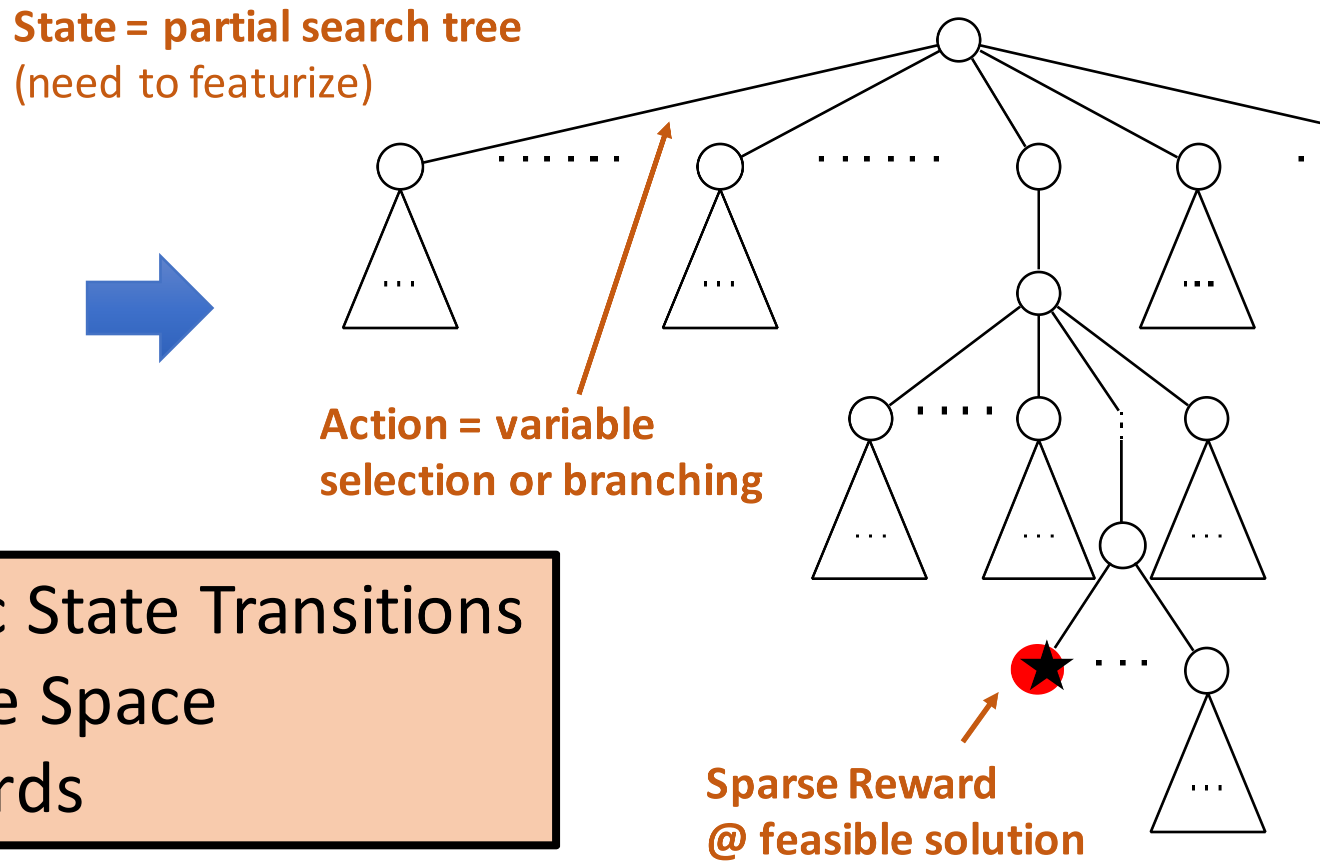
$$\max -\sum_{i=1}^{5} x_i,$$

subject to:

$$x_1 + x_2 \geq 1,$$
$$x_2 + x_3 \geq 1,$$
$$x_3 + x_4 \geq 1,$$

- Deterministic State Transitions
- Massive State Space
- Sparse Rewards

Tree-Search (Branch and Bou



[He et al., 2014][Khalil et al., 2016] [Song et al., arXiv]

Example #2: Learning Greedy Algorithms (discret

Contextual Submodular Maximization:

$$\underset{\Psi:|\Psi|\leq B}{argmax}\, F_x(\Psi)$$

Submodul

Context / Environment

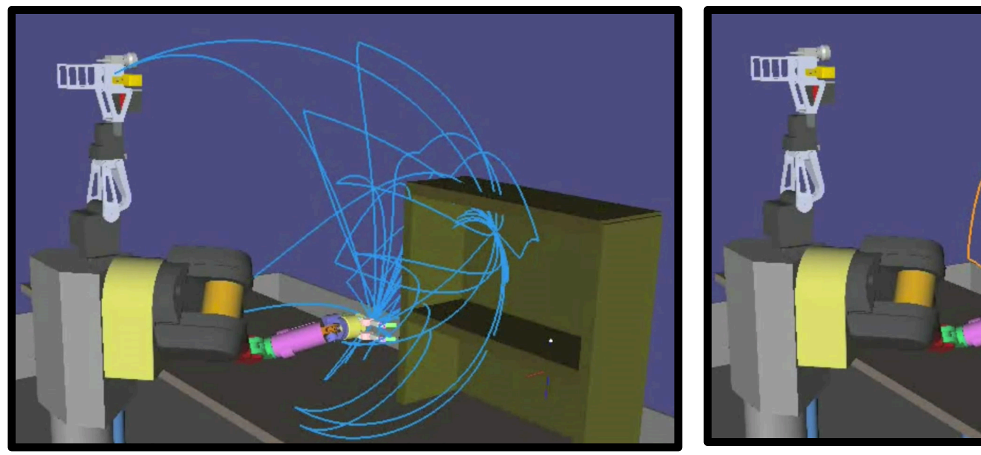
Selec

- Greedy Sequential Selection:
  - $\Psi \leftarrow \Psi \oplus \underset{a}{\operatorname{argmax}} F_x(\Psi \oplus a)$

Not Available at Test Time

- Train policy to mimic greedy:
  - $\pi(s) \rightarrow a$

State s = (Ψ, x)



Dictionary of Trajectories

Select

Learning Policies for Contextual Submodular Prediction S. Ross, R. Zhou, Y. Yue, D. Dey, J.A. Bagnell. ICM

Example #2: Learning Greedy Algorithms (discret

Submodul

Contextual Submodular Maximization:

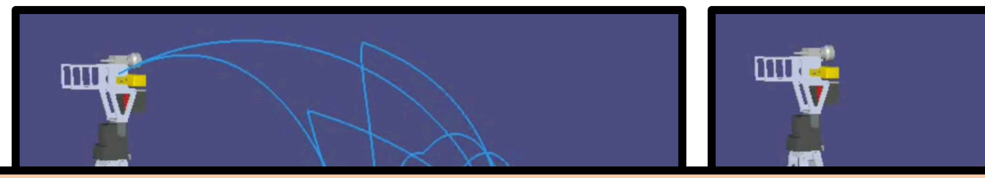
$$\underset{\Psi:|\Psi|\leq B}{argmax}\, F_x(\Psi)$$

Selec

Context / Environment

- Greedy Sequential Selection:
  - $\Psi \leftarrow \Psi \oplus \underset{a}{\operatorname{argmax}} F_x(\Psi \oplus a)$

Not Available at Test Time



- Train policy to mimic greedy:
  - $\pi(s) \rightarrow a$

State s = (Ψ, x)

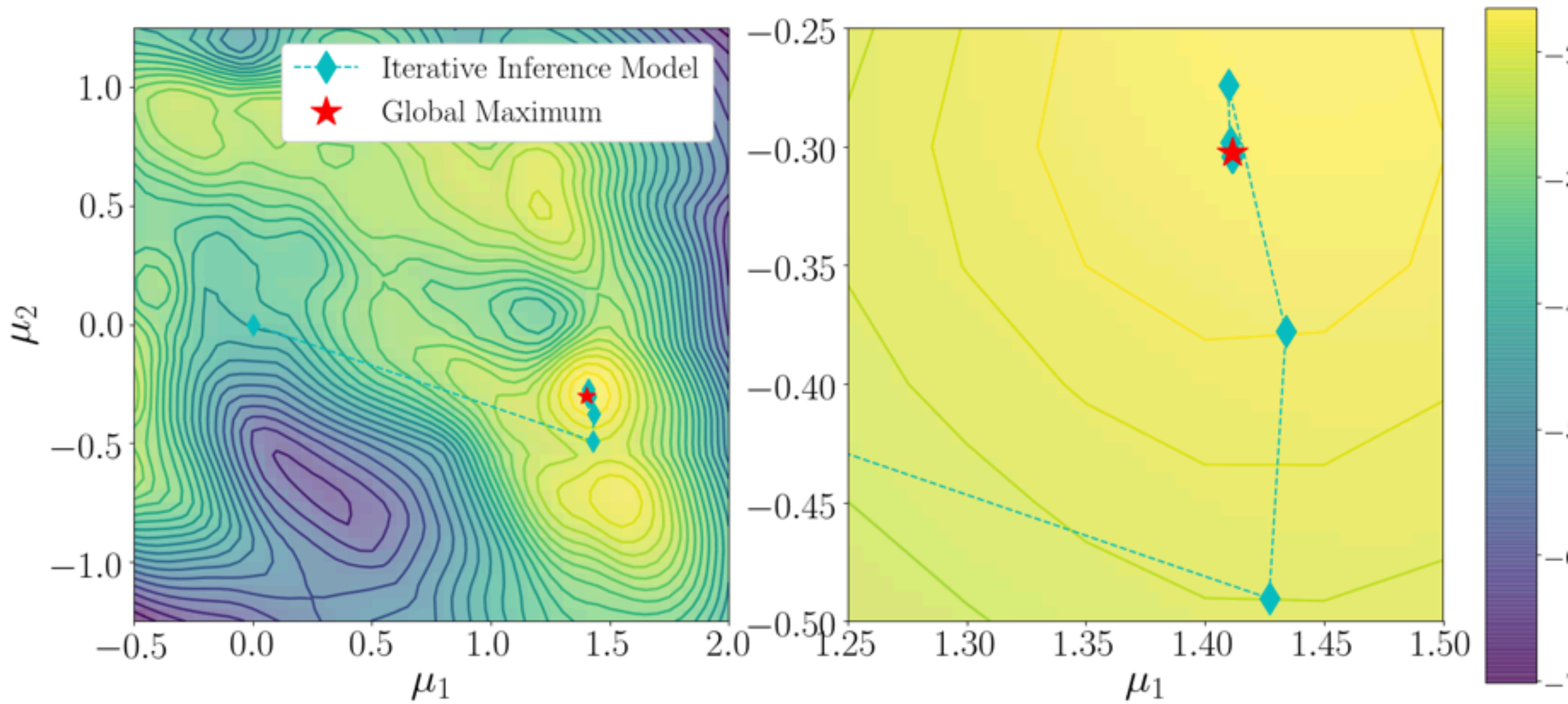
- Deterministic State Transitions
- Massive State Space
- Dense Rewards
- Note: Not Learning Submodula

Learning Policies for Contextual Submodular Prediction S. Ross, R. Zhou, Y. Yue, D. Dey, J.A. Bagnell. ICM

Example #3: Iterative Amortized Inference (conti

Gradient Descent Style Updates:

- State = description of problem & curr
- Action = next point



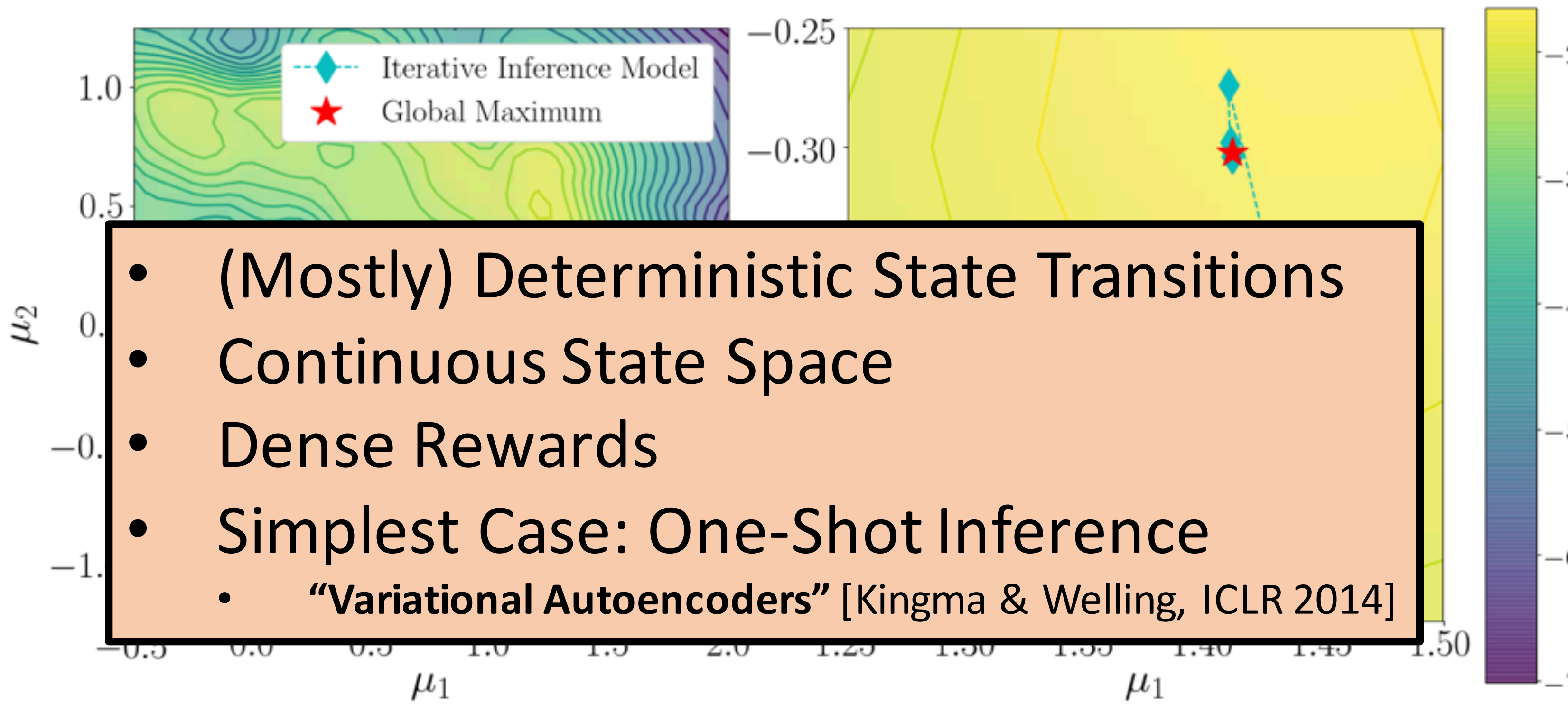
Useful for Accelerating Variational Inference

Iterative Amortized Inference, Joe Marino, Yisong Yue, Stephan Mandt. ICML 2018

Example #3: Iterative Amortized Inference (conti

Gradient Descent Style Updates:

- State = description of problem & curr
- Action = next point



- (Mostly) Deterministic State Transitions
- Continuous State Space
- Dense Rewards
- Simplest Case: One-Shot Inference
  - **“Variational Autoencoders”** [Kingma & Welling, ICLR 2014]

Useful for Accelerating Variational Inference

Iterative Amortized Inference, Joe Marino, Yisong Yue, Stephan Mandt. ICML 2018

Optimization as Sequential Decision Mak

Learning to Search

- Discrete Optimization (Tree Search), Sparse Rewards
- **Learning to Search via Retrospective Imitation** [arXiv]
- **Co-training for Policy Learning** [UAI 2019]

Contextual Submodular Maximization

- Discrete Optimization (Greedy), Dense Rewards
- **Learning Policies for Contextual Submodular Prediction** [ICML 2013]

Learning to Infer

- Continuous Optimization (Gradient-style), Dense Rewards
- **Iterative Amortized Inference** [ICML 2018]
- **A General Method for Amortizing Variational Filtering** [NeurIPS 2018]

Jialin

Stepha

Joe M

Optimization as Sequential Decision Mak

Learning to Search

- Discrete Optimization (Tree Search), Sparse Rewards
- **Learning to Search via Retrospective Imitation** [arXiv]
- **Co-training for Policy Learning** [UAI 2019]

Jialin

Contextual Submodular Maximization

- Discrete Optimization (Greedy), Dense Rewards
- **Learning Policies for Contextual Submodular Prediction** [ICML 2013]

Stepha

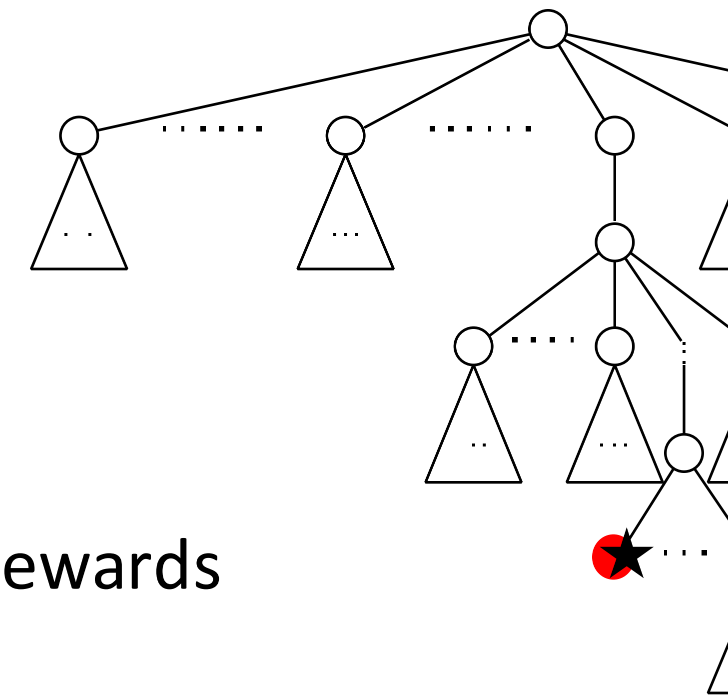
Learning to Infer

- Continuous Optimization (Gradient-style), Dense Rewards
- **Iterative Amortized Inference** [ICML 2018]
- **A General Method for Amortizing Variational Filtering** [NeurIPS 2018]

Joe M

Learning to Optimize for Tree Search

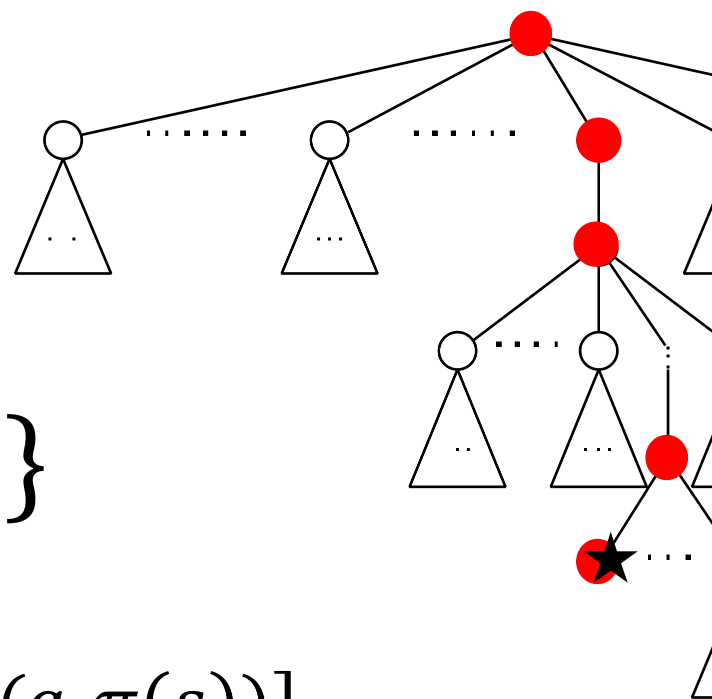
- Idea #1: Treat as Standard RL
- Randomly explore for high rewards
  - **Very hard exploration problem!**
- Issues: massive state space & sparse rewards



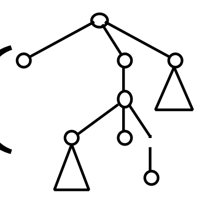
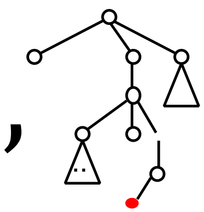
Learning to Optimize for Tree Search

- Idea #2: Treat as Standard IL

- Convert to Supervised Learning
  - Assume access to solved instances



"Demonstration Data"

- Training Data: $D_0 = \{($  ,  $)\}$

- Basic IL: $\underbrace{\operatorname*{argmin}_{\pi \in \Pi} L_{D_0}(\pi) \equiv E_{(s,a)\sim D_0}[\ell(a, \pi(s))]}_{\text{Behavioral Cloning}}$

Challenges w/ Imitation Learning

- Issues with Behavioral Cloning
  - Minimize L_{D_0} … implications?
  - If π makes a mistake early, subsequent state distribution $\approx D_0$??
  - Some extensions to Interactive IL [He et al., NeurIPS 2014]

Our Approach is also Interactive IL

- Demonstrations not Available on Large Problems
  - How to (formally) bootstrap from smaller problems?
  - Bridging the gap between IL & RL

Our Approach gives one solution

Retrospective Imitation

Jialin
Song

- Given:
  - Family of Distributions of Search problems
    - Family is parameterized by size/difficulty
  - Solved Instances on the Smallest/Easiest Instances
    - “Demonstrations”

Difficulty levels: k=1,…,

- Goal:
  - Interactive IL approach
  - Can Scale up from Smallest/Easiest Instances
  - Formal Guarantees

Connections to Curricul
& Transfer Learning

Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

Retrospective Imitation

- Two-Stage Algorithm

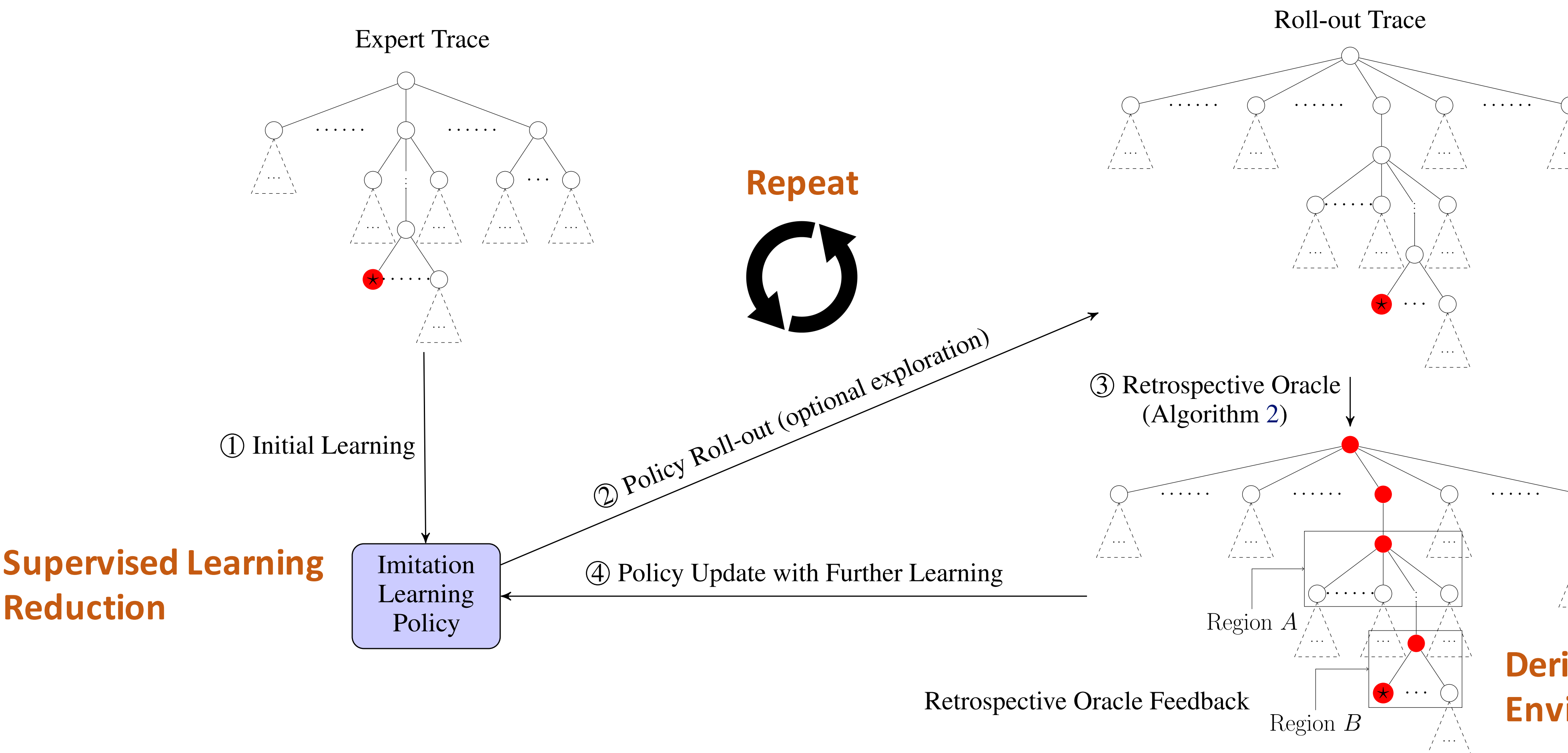
- Core Algorithm
  - Fixed problem difficulty
  - Reductions to Supervised Learning

- Full Algorithm w/ Scaling Up
  - Uses Core Algorithm as Subroutine

Interactive IL w/ Sparse Environme

Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

Retrospective Imitation (Core Algorithm)



Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

Retrospective Imitation (Full Algorithm)

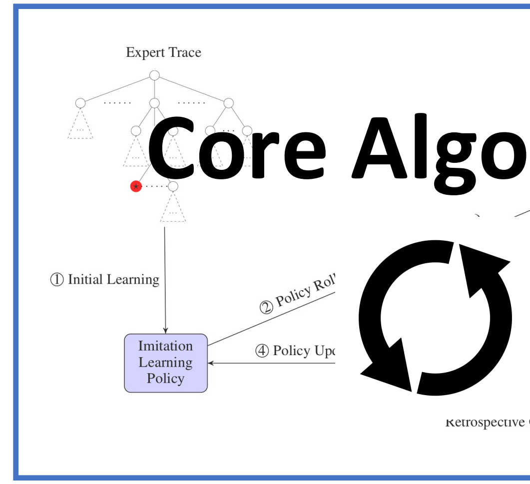
Initialize k=1

Initialize Gurobi/SCIP/CPlex

Problem Difficulty k

Base Solver

Instances & Demonstrations

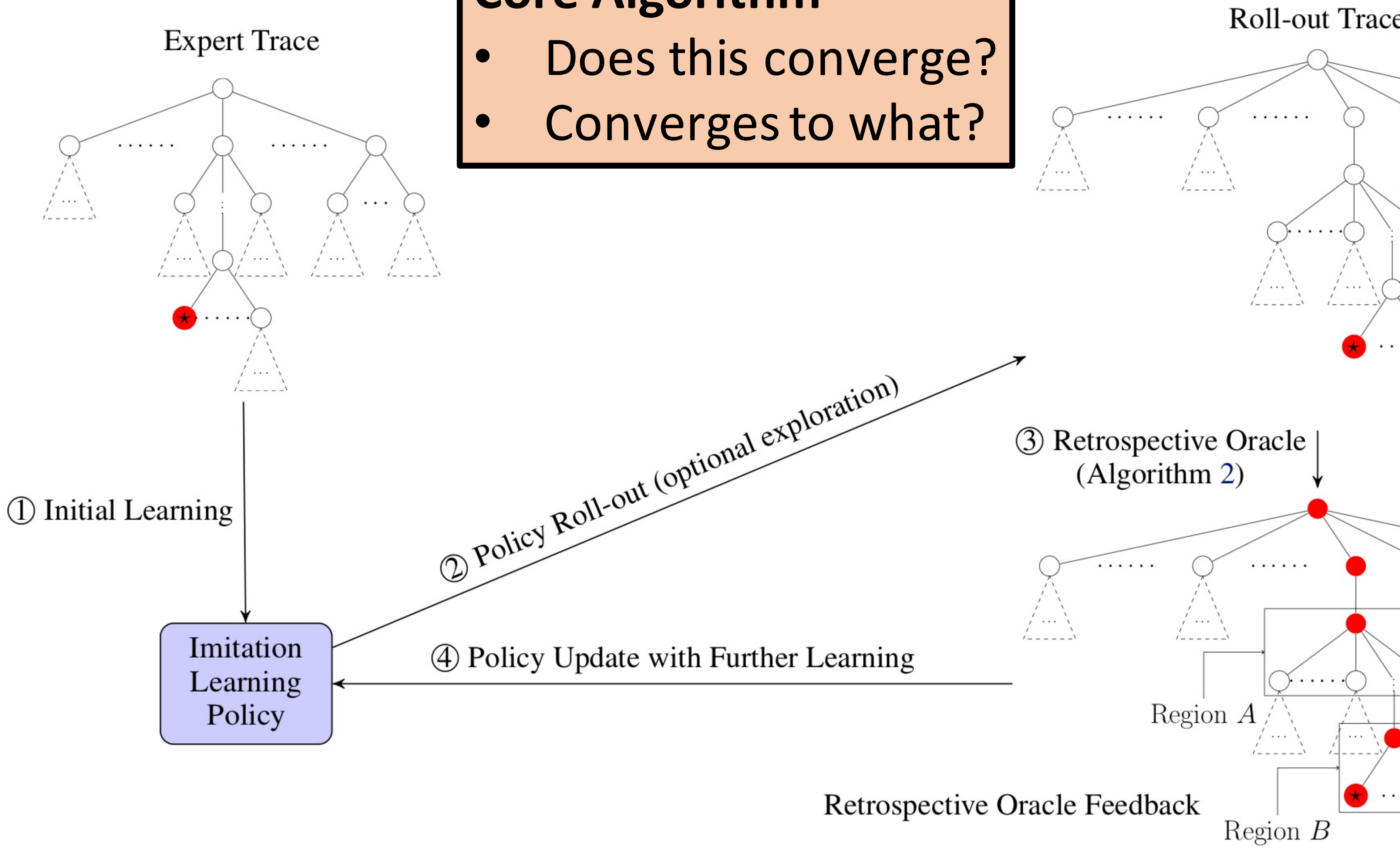


k=k+1
Use trained π

Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

Core Algorithm

- Does this converge?
- Converges to what?



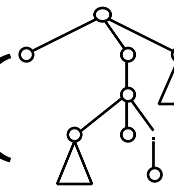
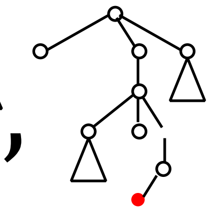
Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

Imitation Learning Tutorial (ICML 2018)

https://sites.google.com/view/icml2018-imitation-learning/

Yisong Yue	**Hoang M. Le**
	
yyue@caltech.edu	hmle@caltech.edu
@YisongYue	@HoangMinhLe
yisongyue.com	hoangle.info

Issues w/ Distribution Drift & Imitation S

- Demonstrations from initial Solver: $D_0 = \{($  ,  $)\}$

"correct" decision in this stat

Which input states?
Correct relative to what?

- Supervised learning: $\underbrace{\operatorname*{argmin}_{\pi \in \Pi} L_{D_0}(\pi) \equiv E_{(s,a)\sim D_0}[\ell(a, \pi(s))]}$

Oracle call to TensorFlow/PyTorch/etc…

If π achieves low error on D_0, so what?

Interactive Imitation Learning (Core Alg)

- First popularized by [Daume et al., 2009] [Ross et al., 2011]

- Basic idea:

i=i+1

  - Train $\pi_{i-1} = \underset{\pi \in \Pi}{\operatorname{argmin}} L_{D_{i-1}}(\pi)$ **Supervised Learning**
  - Roll-out π_{i-1}, collect traces $\{\tau\}$ **Run on instances**
  - Demonstrator converts $\{\tau\}$ into per-state feedback: $\widehat{D}_i$ **Depends on**
  - $D_i = \widehat{D}_i \cup D_{i-1}$ **Data aggregation**

Search-based Structured Prediction, Daume, Langford, Marcu, Machine Learning Journal 2009
A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning, Ross, Gordon, Bagn

Interactive Imitation

- First popularized by [Daume et al

- Basic idea:
  - Train $\pi_{i-1} = \underset{\pi \in \Pi}{\operatorname{argmin}} L_{D_{i-1}}(\pi)$

i=i+1

  - Roll-out π_{i-1}, collect traces $\{\tau\}$ **Run on instances**
  - Demonstrator converts $\{\tau\}$ into per-state feedback: $\widehat{D}_i$ **Depends on**
  - $D_i = \widehat{D}_i \cup D_{i-1}$ **Data aggregation**

Learns to Correct its Own Mis

Convergence Guarantees:

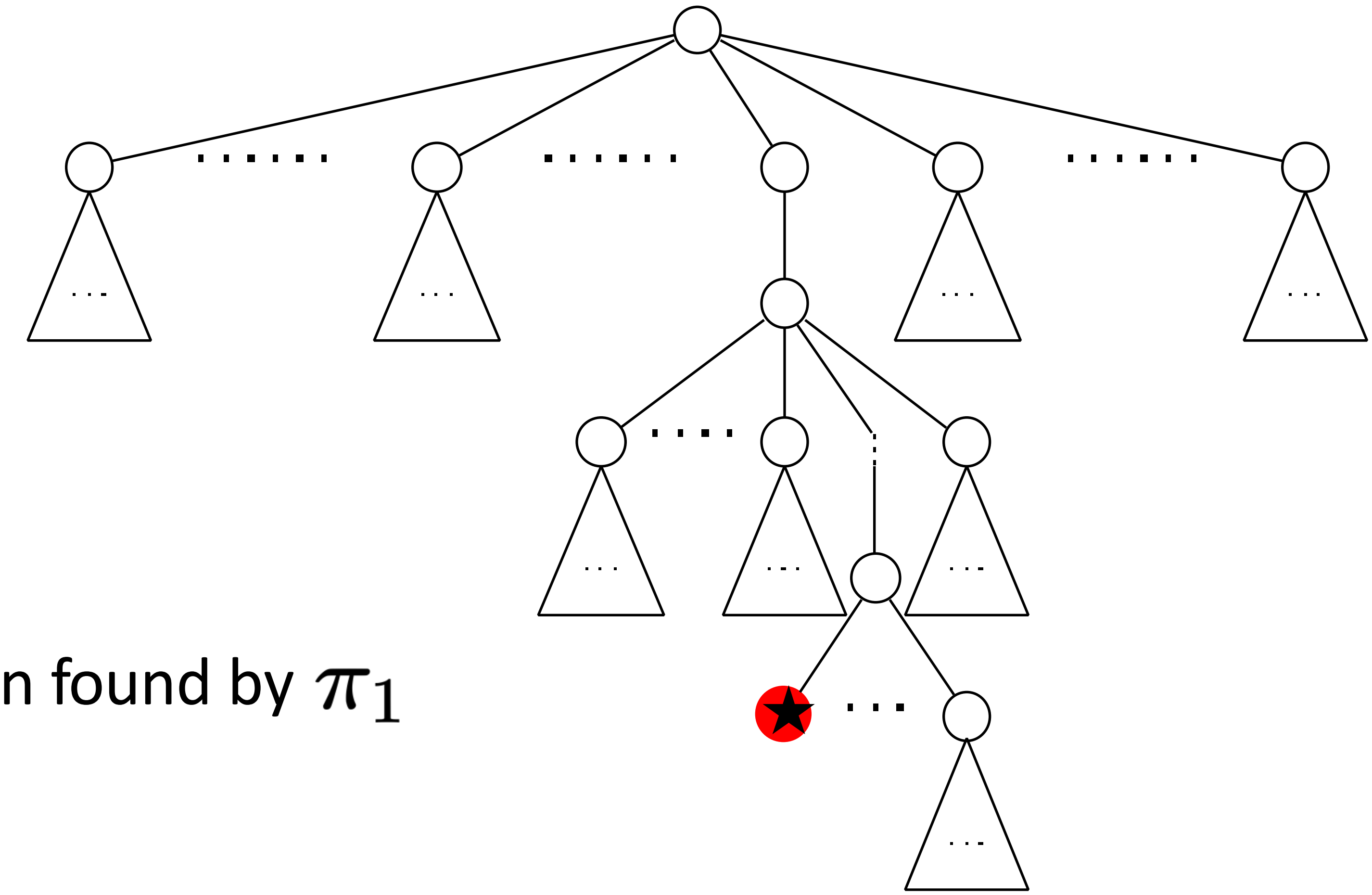
- $\sum_{i=0}^{M} L_{D_i}(\pi_i) \rightarrow \min_{\pi \in \Pi} \sum$
- Follow-the-Leader argume
- Also studied in [He et al., NeurI

Requires defining "correct"

- Retrospective Oracle

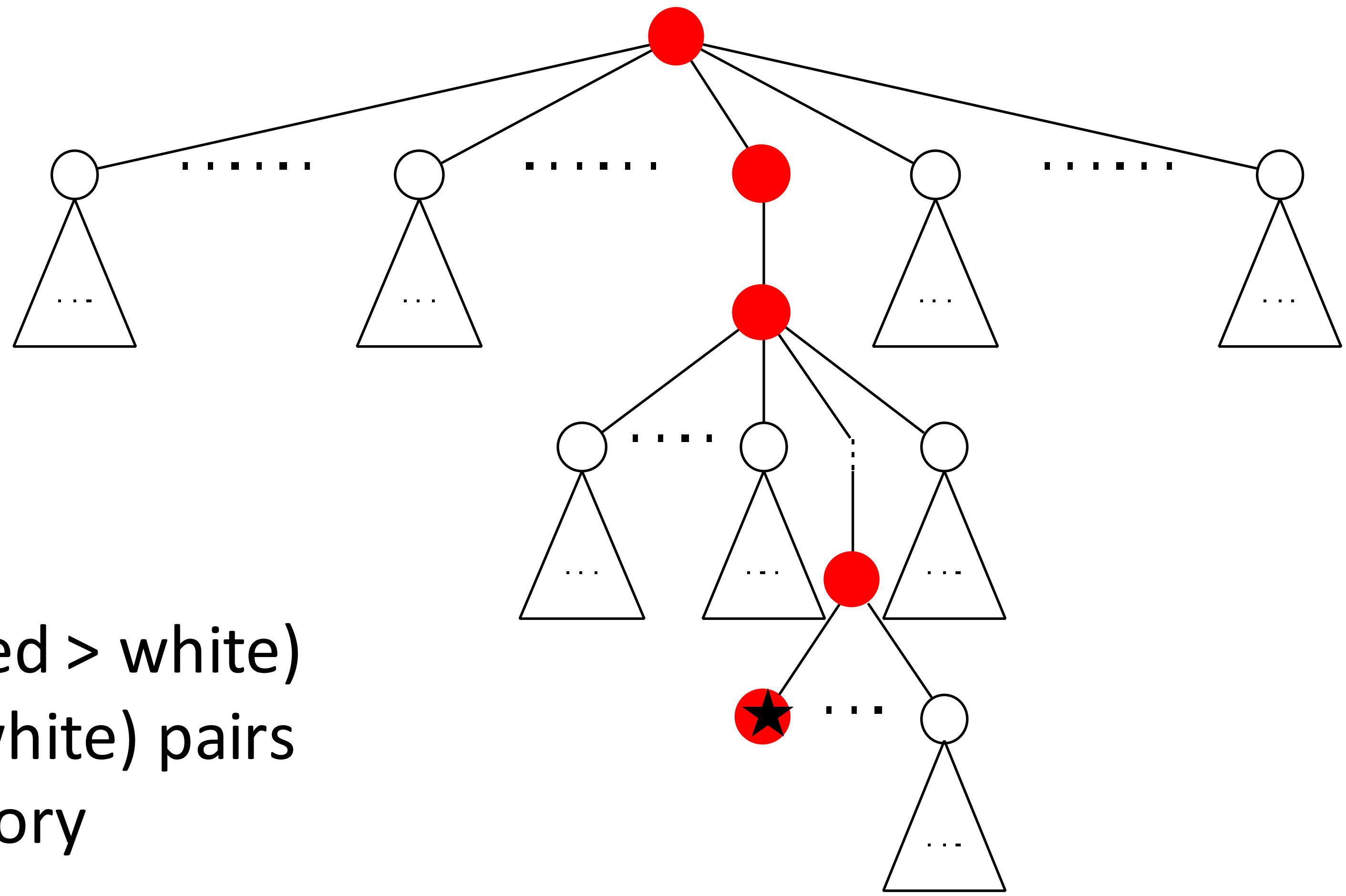
Search-based Structured Prediction, Daume, Langford, Marcu, Machine Learning Journal 2009
A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning, Ross, Gordon, Bagr

π_1 Policy Rollout



★: best solution found by π_1

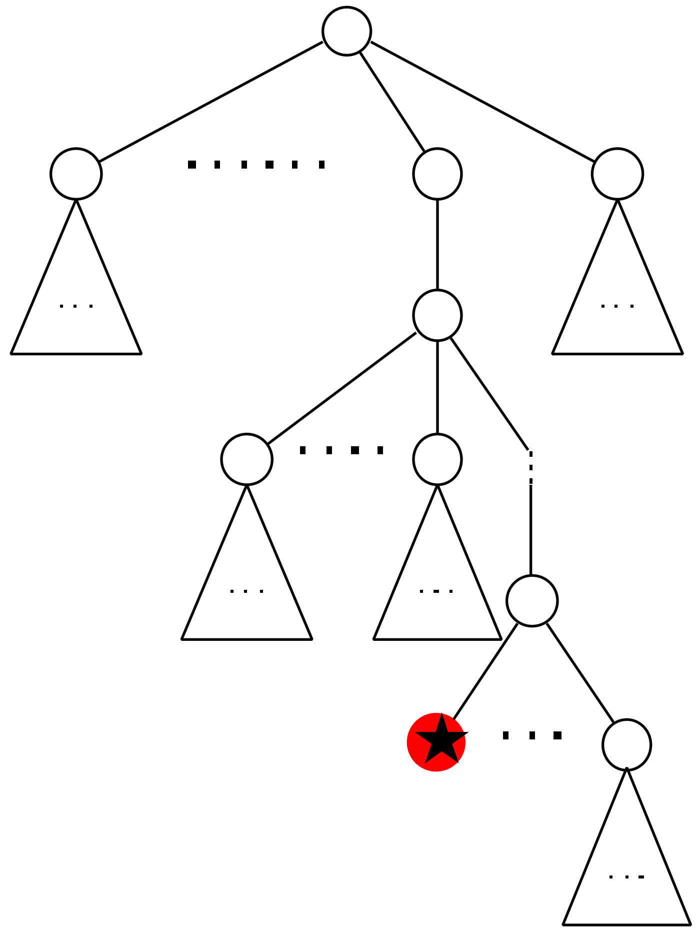
Retrospective Oracle Feedback



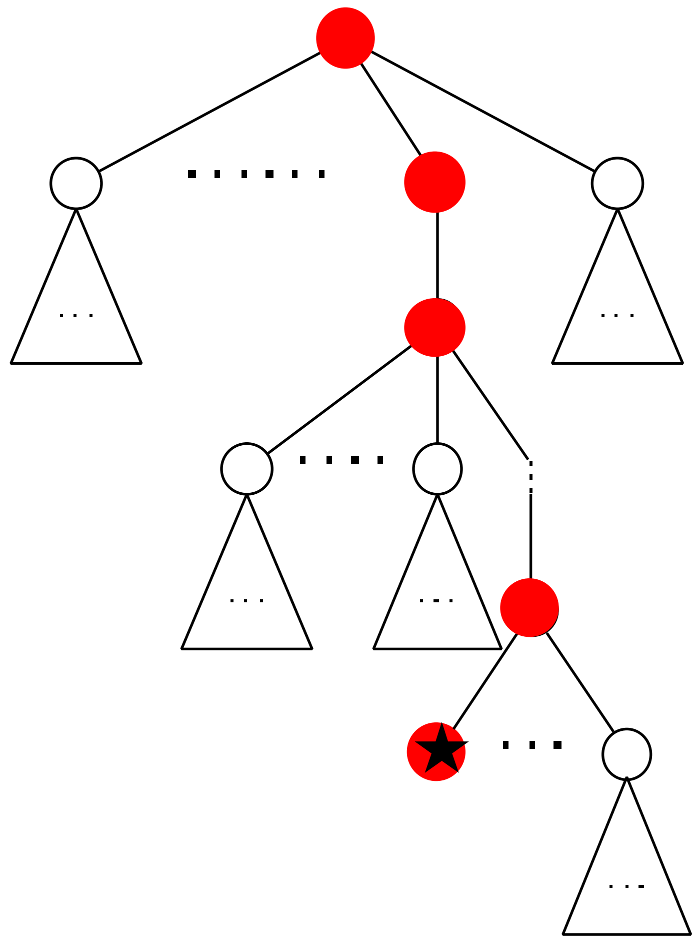
Feedback: (red > white)
for all (red, white) pairs
in the trajectory

Retrospect

π_2 Policy Rollout

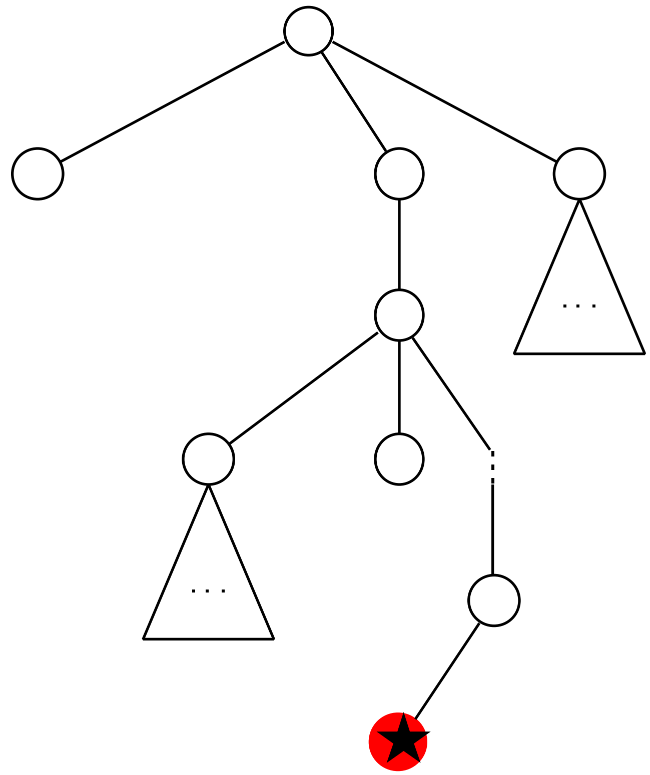


Retrospective Oracle Feedback



Feedback: (red > white)
for all (red, white) pairs
in the trajectory

π_3 Policy Rollout



Core Algorithm Summary

- Sequence of Learning Reductions
- Leverages Retrospective Oracle to Define "Correct"
  - Relies on sparse environmental rewards
- Converges to near-optimal policy in class
  - Offloads computational challenges to Supervised Learning Oracle
- For supervised learning error ε:

$$\boxed{\text{Expected Search Length} = \frac{H^*}{1-2\varepsilon}}$$

Optimal Search Le
(typically # intege

Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

Guarantees for Full Algorithm

- Run π^k on problems of difficulty k+1
  - Initial demonstrations for the harder problem instances

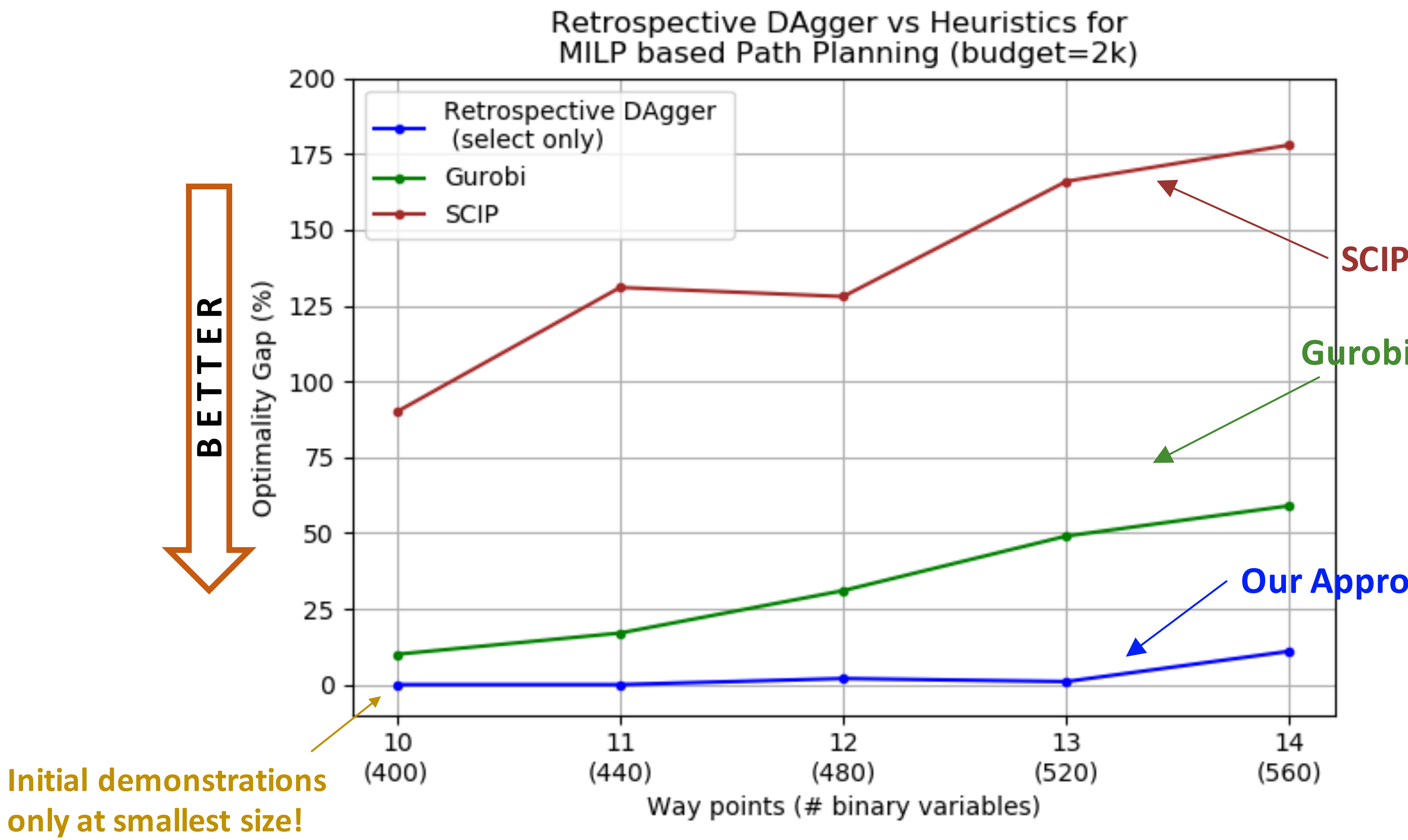
- **Suppose:** we could have run external solver on harder instanc

Gurobi/SCIP/CPlex/Et

- **Suppose:** search trace includes feasible solution of external so

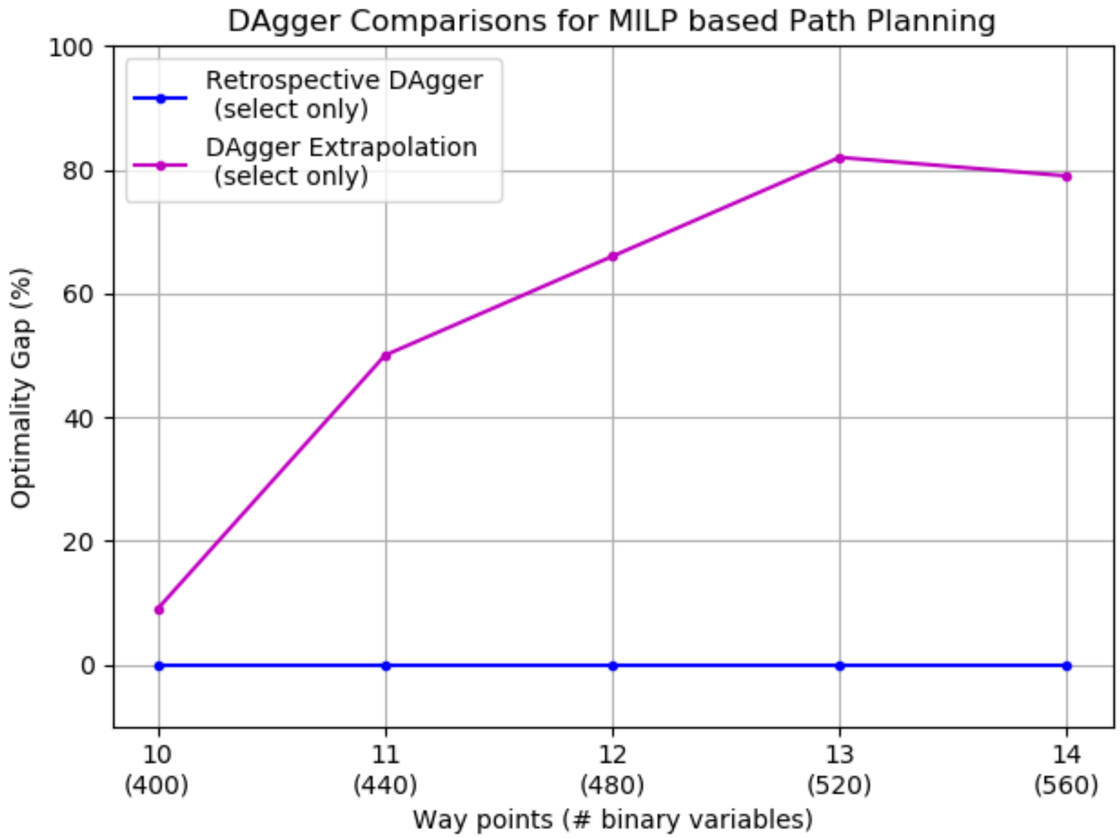
- Then π^k is as good as using original external solver!
  - (might take longer to converge)

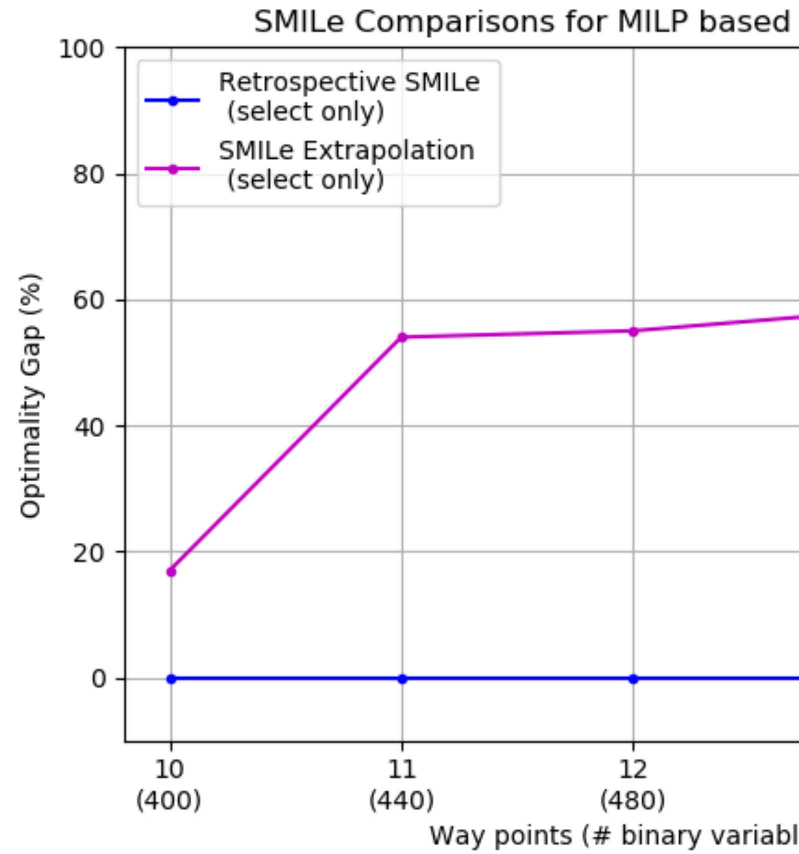
Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv



Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

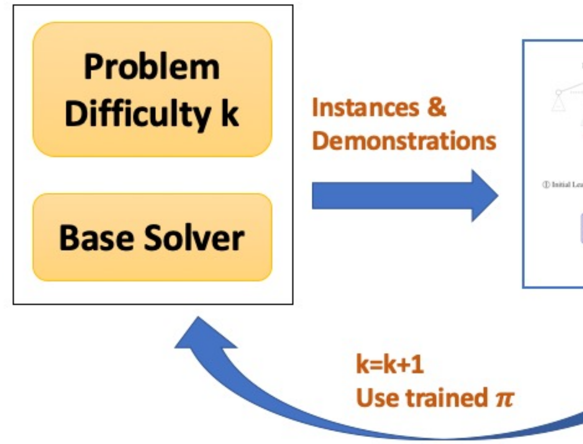
Comparisons w/ Conventional IL





Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv

Retrospective Imitation



- Two-Stage Algorithm
  - Leverages Supervised Learning Oracle

- Initial demonstrations on small problems

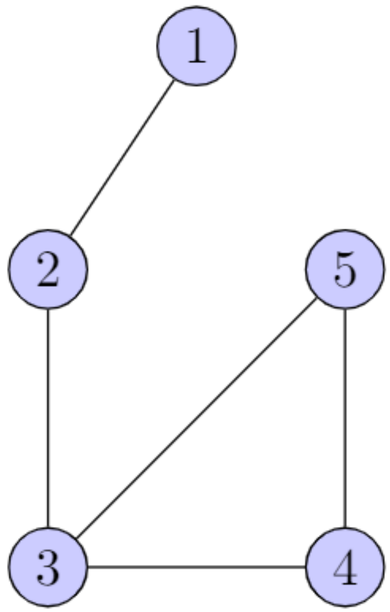
- Exploits sparse environmental reward
  - “Retrospective Oracle”

- Iteratively scale up to harder problems

Co-Training for Policy Learning
(Multiple Views)

Jialin Song

Example: Minimum Vertex Cover



Graph View

[Khalil et al., 2017]

$$\max \; -\sum_{i=1}^{5} x_i,$$

subject to:

$$x_1 + x_2 \geq 1,$$
$$x_2 + x_3 \geq 1,$$
$$x_3 + x_4 \geq 1,$$
$$x_3 + x_5 \geq 1,$$
$$x_4 + x_5 \geq 1,$$
$$x_i \in \{0, 1\}, \forall i \in \{1, \cdots, 5\}$$

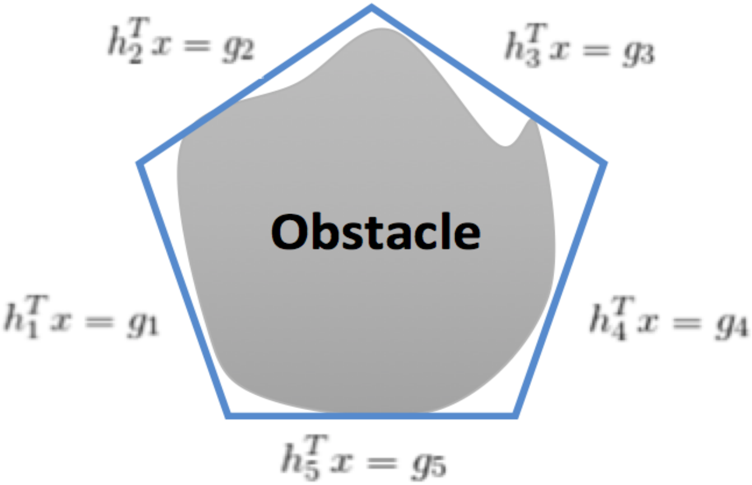
Integer Program View
(Branch & Bound View)

[He et al., 2014]

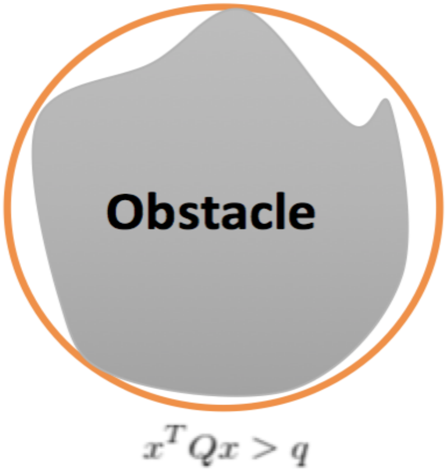
Co-Training for Policy Learning
(Multiple Views)

Jialin Song

Example: Different Types of Integer Programs



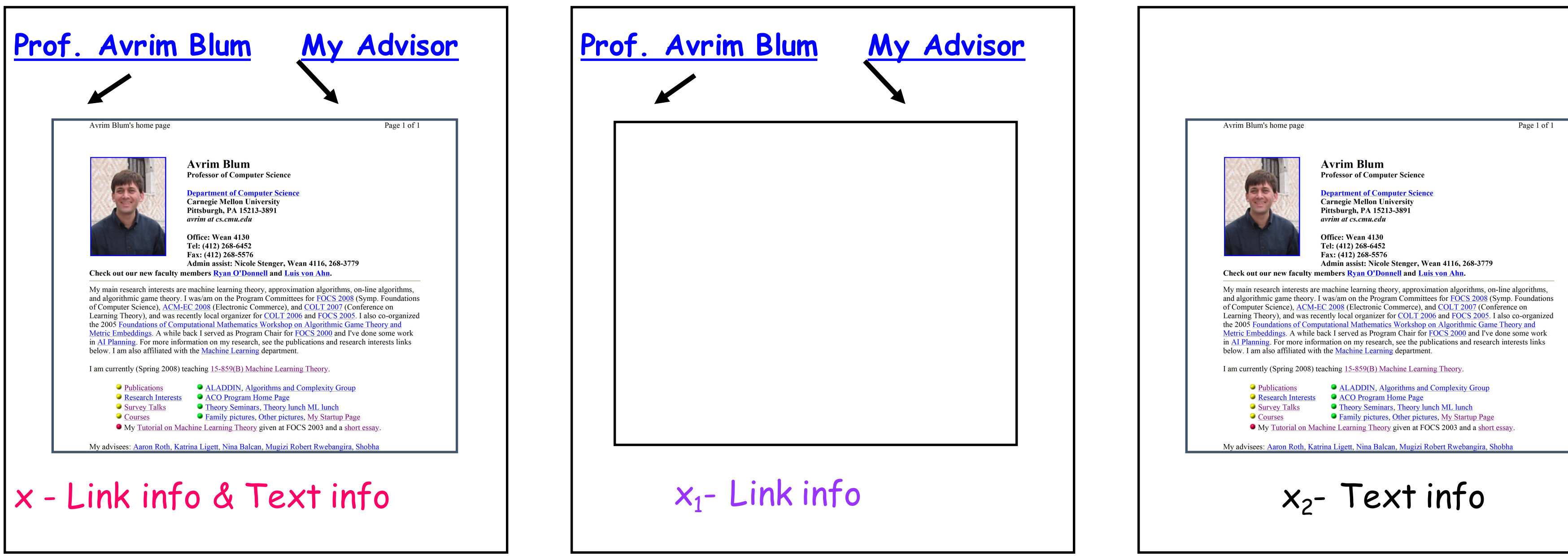
ILP



QCQP

Co-Training [Blum & Mitchell, 1998]

- Many learning problems have different sources of information
- Webpage Classification: Words vs Hyperlinks



(Taken from Nina Balcan's slides)

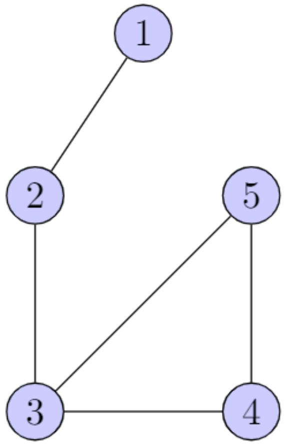
What’s Different about Policy Co-Training

- Sequential Decisions vs 1-Shot Decisions

- (Sparse) Environmental Feedback
  - Can collect more “labels”

(Not always applicable)

- Different Action Spaces
  - Graph vs Branch-and-Bound



max −

subject t

$x_1 + x_2$

$x_2 + x_3$

$x_3 + x_4$

$x_3 + x_5$

$x_4 + x_5$

$x_i \in \{0$

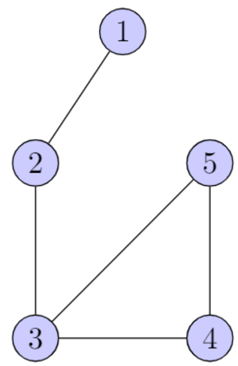
Co-training for Policy Learning, Jialin Song, Ravi Lanka, et al., UAI 2019

[1] “Learning combinatorial optimization algorithms over grap
[2] “Learning to Search in Branch and Bound Algorithms” [He e
[3] “Learning to Search via Retrospective Imitation” [Song et a

Intuition

MVC Instance

E.g., [1]



E.g., [2,3]

$$\max \; -\sum_{i=1}^{5} x_i,$$

subject to:

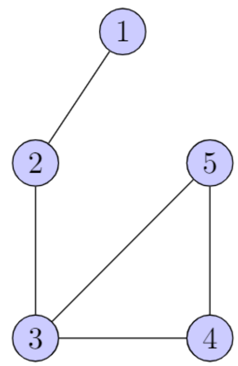
$$x_1 + x_2 \geq 1,$$
$$x_2 + x_3 \geq 1,$$
$$x_3 + x_4 \geq 1,$$
$$x_3 + x_5 \geq 1,$$
$$x_4 + x_5 \geq 1,$$
$$x_i \in \{0, 1\}, \forall i \in \{1, \cdots, 5\}$$

[1] “Learning combinatorial optimization algorithms over grap
[2] “Learning to Search in Branch and Bound Algorithms” [He e
[3] “Learning to Search via Retrospective Imitation” [Song et a

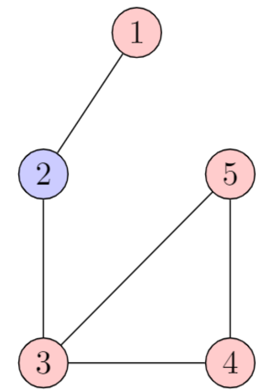
Intuition

MVC Instance

E.g., [1]



π^1



E.g., [2,3]

$$\max \; -\sum_{i=1}^{5} x_i,$$

subject to:

$$x_1 + x_2 \geq 1,$$
$$x_2 + x_3 \geq 1,$$
$$x_3 + x_4 \geq 1,$$
$$x_3 + x_5 \geq 1,$$
$$x_4 + x_5 \geq 1,$$
$$x_i \in \{0, 1\}, \forall i \in \{1, \cdots, 5\}$$

π^2

$$x_1 = 0$$
$$x_2 = 1$$
$$x_3 = 1$$
$$x_4 = 1$$
$$x_5 = 0$$

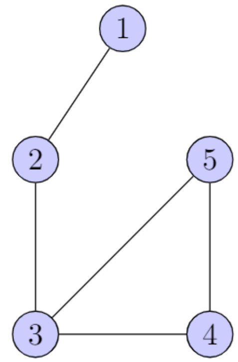
Bette

[1] “Learning combinatorial optimization algorithms over grap
[2] “Learning to Search in Branch and Bound Algorithms” [He e
[3] “Learning to Search via Retrospective Imitation” [Song et a

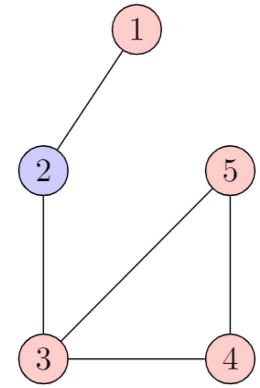
Intuition

MVC Instance

E.g., [1]



π^1



Demonstra

E.g., [2,3]

$$\max -\sum_{i=1}^{5} x_i,$$

subject to:

$$x_1 + x_2 \geq 1,$$
$$x_2 + x_3 \geq 1,$$
$$x_3 + x_4 \geq 1,$$
$$x_3 + x_5 \geq 1,$$
$$x_4 + x_5 \geq 1,$$
$$x_i \in \{0,1\}, \forall i \in \{1, \cdots, 5\}$$

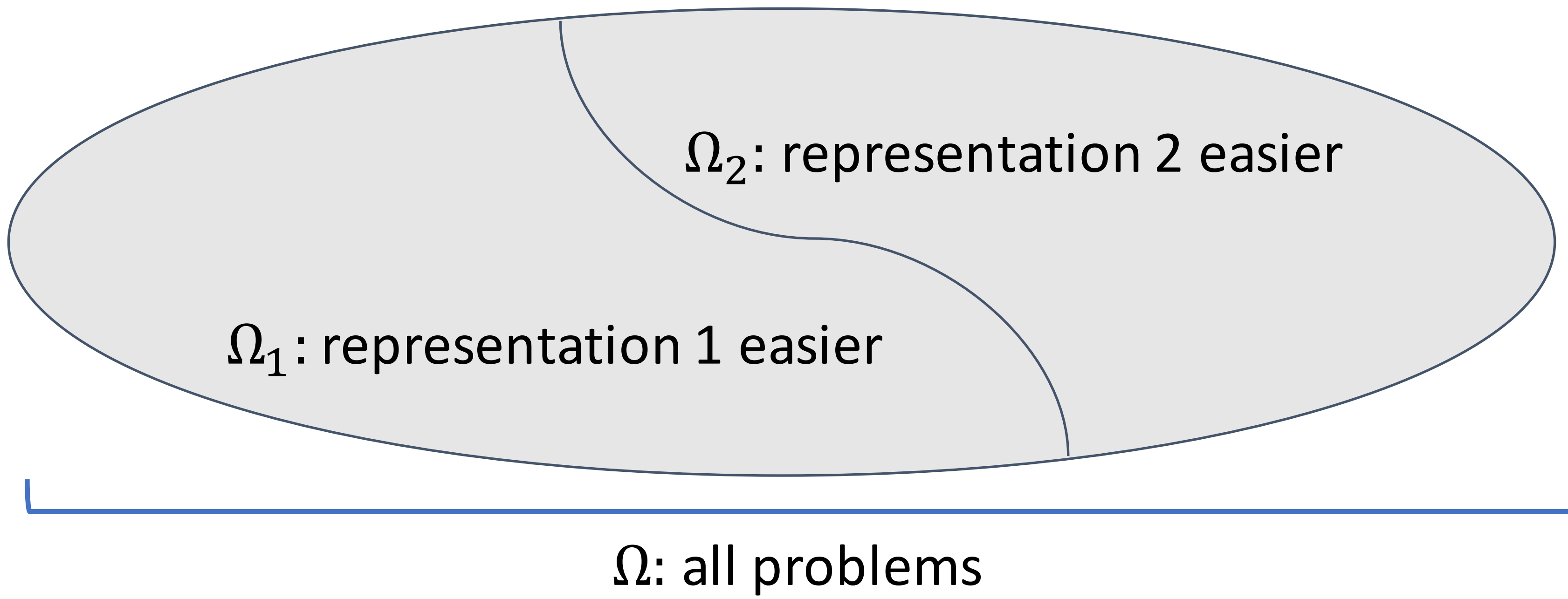
π^2

$$x_1 = 0$$
$$x_2 = 1$$
$$x_3 = 1$$
$$x_4 = 1$$
$$x_5 = 0$$

Bette

Theoretical Insight

- Different representations differ in hardness
- Goal: quantify improvement

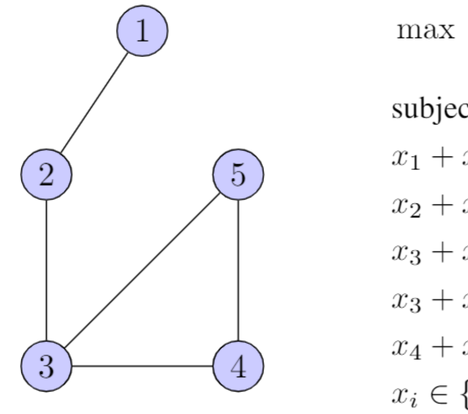


Co-training for Policy Learning, Jialin Song, Ravi Lanka, et al., UAI 2019

(Towards) a Theory of Policy Co-Training

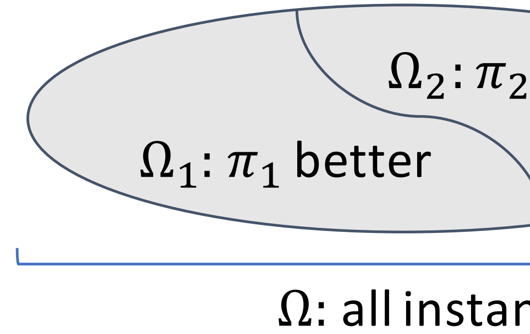
- Two MDP "views": M^1 & M^2
  - $f^{1\to 2}(\tau^1) \implies \tau^2$ (and vice versa)

    "Trajectory" / "Rollout"
  - Realizing τ^1 on M^1 $\Leftrightarrow$ realizing τ^2 on M^2



- **Question:** when does having two views/policies help?
  - Policy Improvement (next slide)
    - Builds upon [Kang et al., ICML 2018]
  - Optimality Gap for Shared Action Spaces (in paper)
    - Builds upon [DasGupta et al., NeurIPS 2002]

Policy Improvement Bound



Standard for Policy Gradient

1-step suboptimality of π^1 on Ω

KL Divergence of π^1 vs π'^1 on Ω

JS Divergence of π^2 vs π^1 on Ω_2

War

1-step suboptimal

$$J(\pi'^1) \geq J_{\pi^1}(\pi'^1) - \frac{2\gamma\left(\alpha_\Omega^1 \varepsilon_\Omega^1 + 4\beta_{\Omega_2}^2 \varepsilon_{\Omega_2}^2\right)}{(1-\gamma)^2} + \delta_{\Omega_2}^2$$

Performance of new policy (either RL or IL)

Approximation by sampling from π^1

Discount

Performance Gap of π^2 ov
$J(\pi^2|M{\sim}\Omega_2) - J(\pi^1|M{\sim}\Omega$

Want to Maximize

Builds upon theoretical results from [Kang et al., ICML 2018]

Policy Improvement Bound (Summary)

$$J(\pi'^1) \geq J_{\pi^1}(\pi'^1) - \frac{2\gamma\left(\alpha_\Omega^1 \varepsilon_\Omega^1 + 4\beta_{\Omega_2}^2 \varepsilon_{\Omega_2}^2\right)}{(1-\gamma)^2} + \delta_{\Omega}^{2}$$

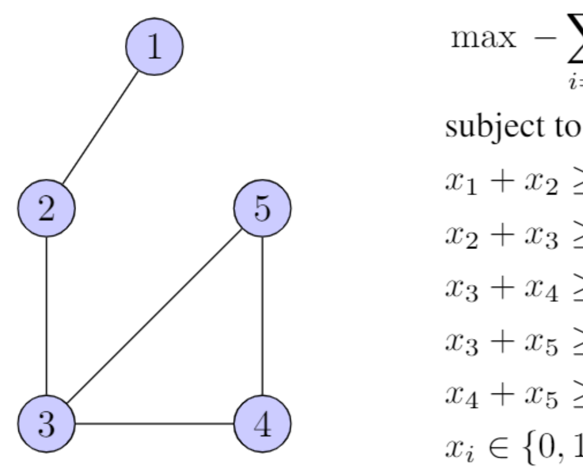
- Minimizing $\beta_{\Omega_2}^2$ → low disagreement between π^2 vs π^1
- Maximizing $\delta_{\Omega_2}^2$ → high performance gap π^2 over π^1 on some

CoPiEr Algorithm (Co-training for Policy Learning)

Sample $M \sim \Omega$

Rollout

Run $\pi^1 \rightarrow \tau^1$ Run π



M^1

Exchange **(only showing 1 version)**

If π^1 better: $\tau'^2 = f^{1\rightarrow 2}(\tau^1)$, $\tau'^1 = \emptyset$

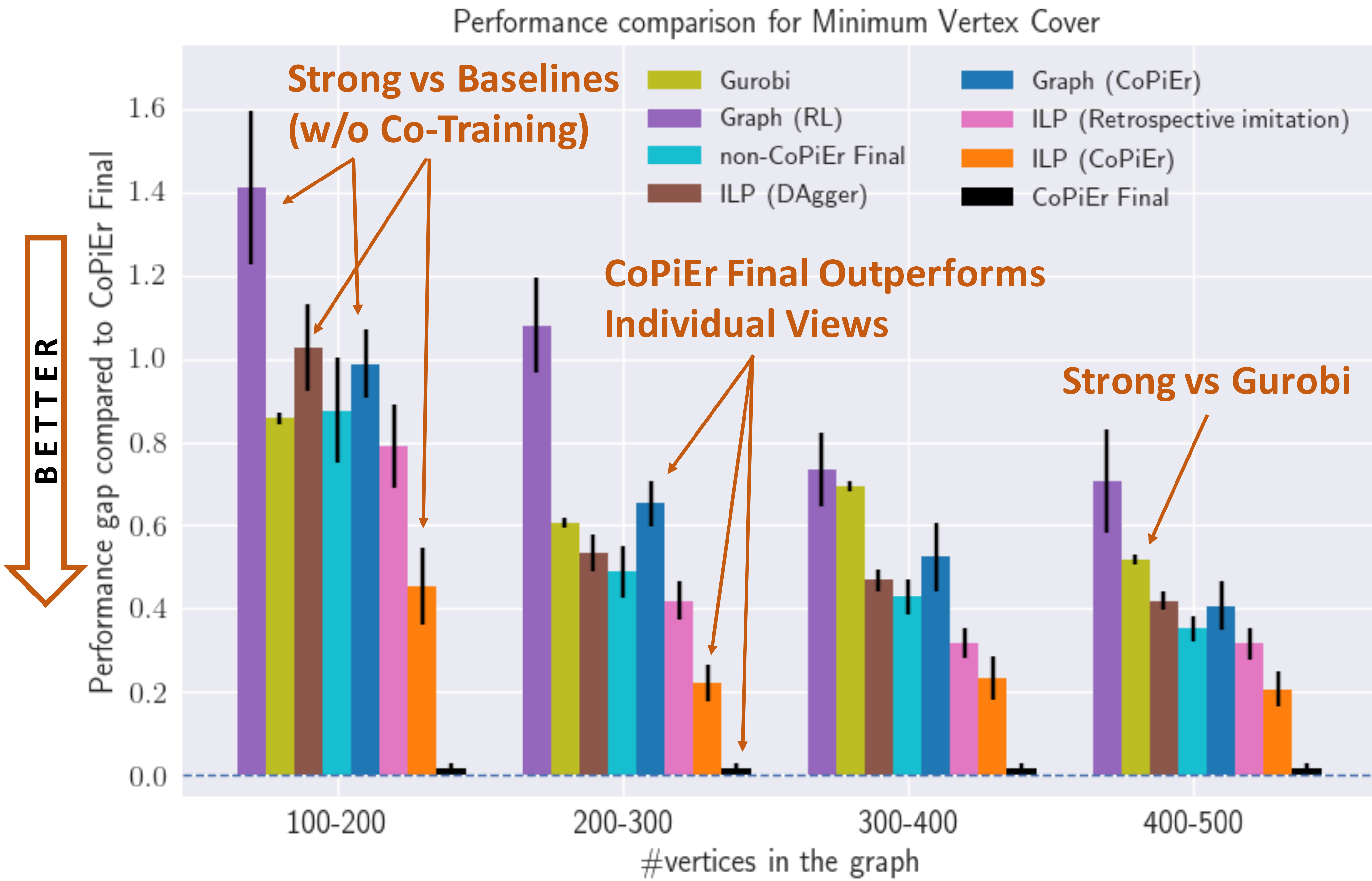
If π^2 better: $\tau'^1 = f^{2\rightarrow 1}(\tau^2)$, $\tau'^2 = \emptyset$

Update **(only showing 1 view)**

Augmented Obj: $\tilde{J}(\pi') = J_\pi(\pi') - \lambda L(\pi', \tau')$

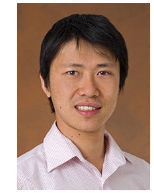
Take gradient step

Co-training for Policy Learning, Jialin Song, Ravi Lanka, et al., UAI 2019


Performance comparison for Minimum Vertex Cover
Strong vs Baselines
(w/o Co-Training)
Gurobi
Graph (RL)
non-CoPiEr Final
ILP (DAgger)
Graph (CoPiEr)
ILP (Retrospective imitation)
ILP (CoPiEr)
CoPiEr Final
CoPiEr Final Outperforms
Individual Views
Strong vs Gurobi
BETTER
Performance gap compared to CoPiEr Final
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
100-200
200-300
300-400
400-500
#vertices in the graph

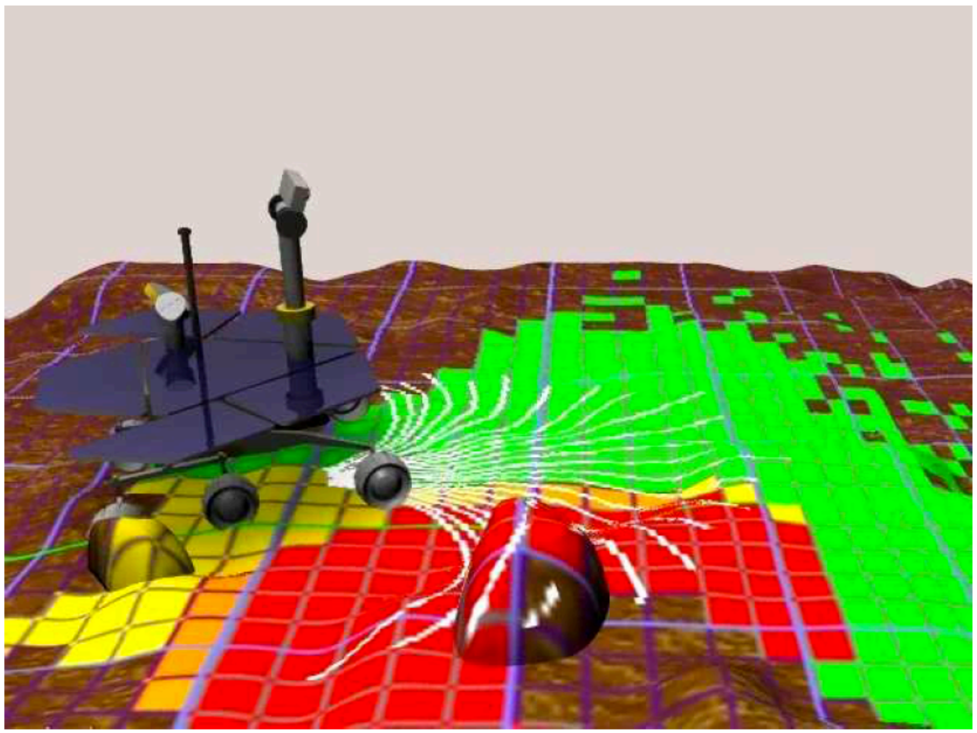
Ongoing: Integration with ENav

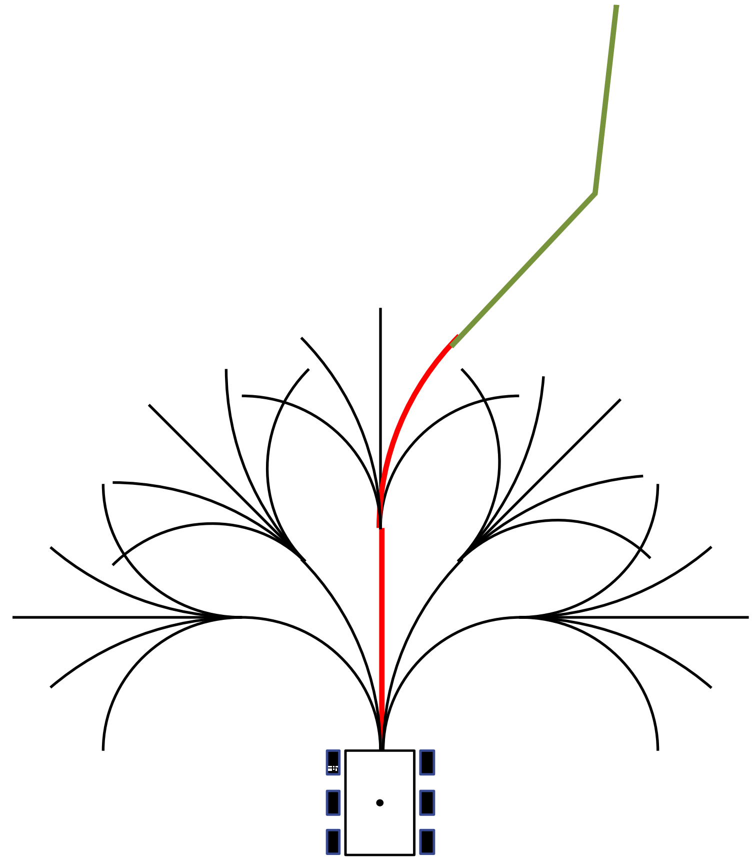
Ravi
Lanka



Hiro
Ono

O
To



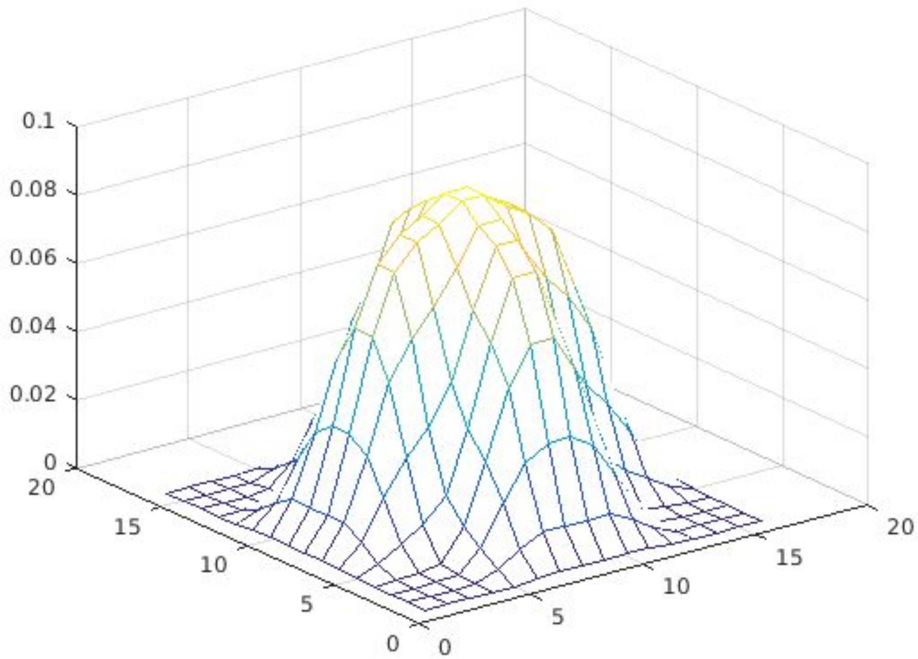


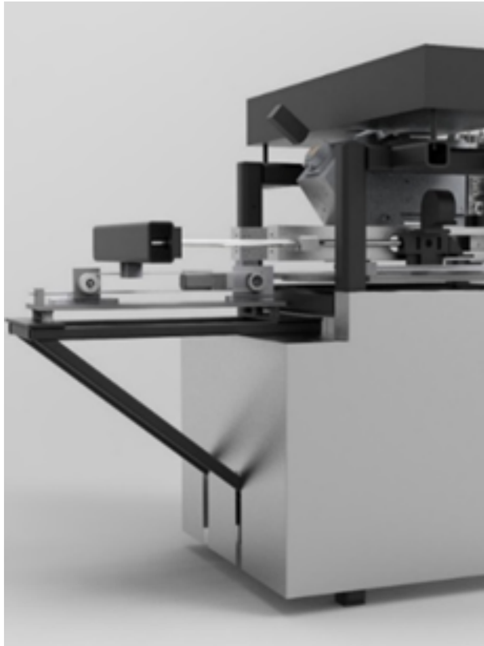
Ongoing: Additive Manufacturing

Stephanie Ding

Jialin Song

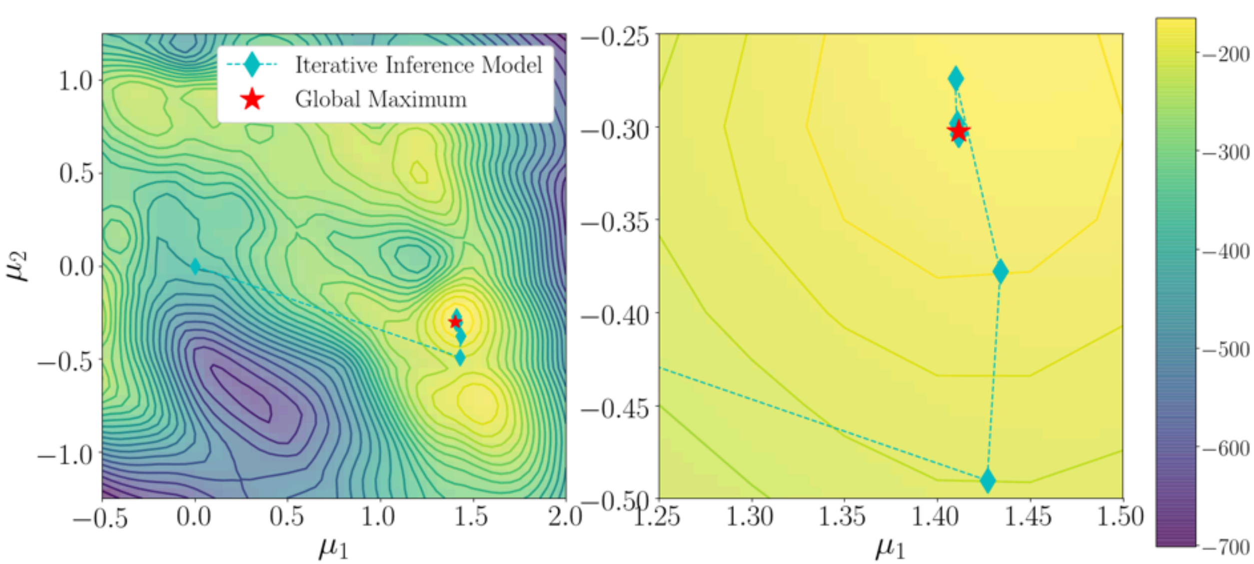
- Planning for 3D Inkjet Droplet Printing



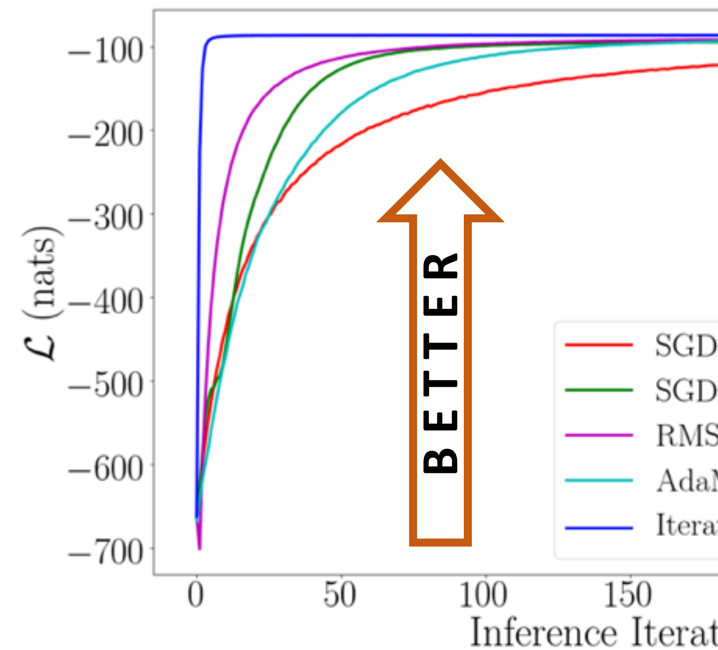


Rensselaer

Iterative Amortized Inference
(for Deep Probabilistic Models)



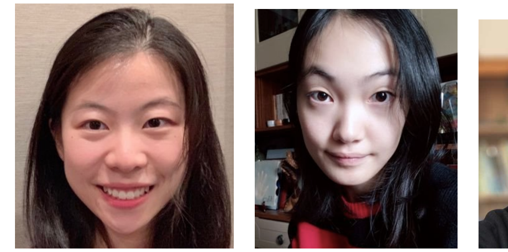
Related to "Learning to Learn" [Andychowicz et al., 2016]



Iterative Amortized Inference, Joe Marino et al., ICML 2018

A General Framework for Amortizing Variational Filtering, Joe Marino et al, NeurIPS 2018

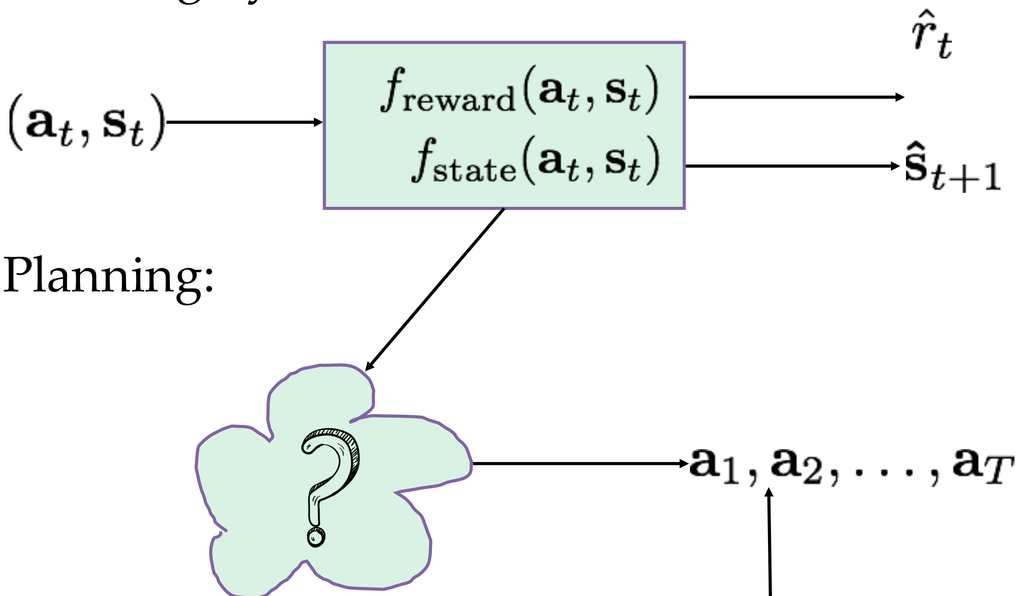
Ongoing: Amortized Planning



Yujia Huang

Sophie Dai

Learning dynamics:



Optimize:

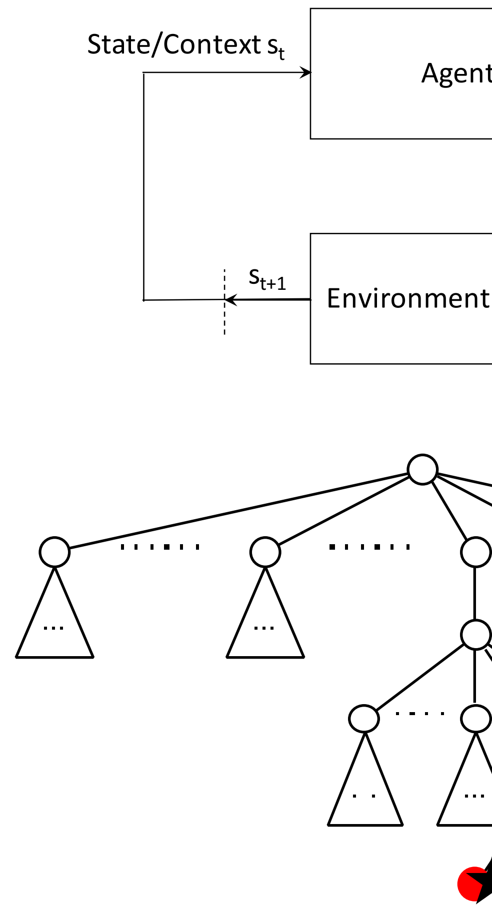
$$\max_{\mathbf{a}_1,\ldots,\mathbf{a}_T} \sum_{t=1}^{T} f_{\text{reward}}\left(f_{\text{state}}(\hat{\mathbf{s}}_{t-1}, \mathbf{a}_{t-1}), \mathbf{a}_t\right)$$

Baseline: Gradient-based P

Can use (offline) training to

Learning to Optimize as Policy Learning

- Optimization as Sequential Decision Making
- Formulate New Learning Problems
  - Builds upon RL/IL
- Interesting Algorithms
  - Theoretical Analysis/Guidance
  - Good Empirical Performance

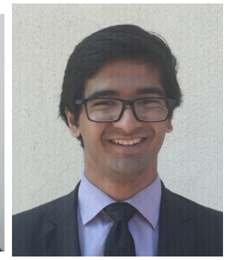


Jialin Song

Ravi Lanka

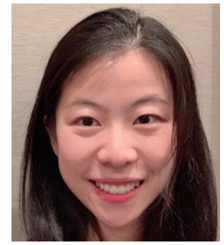
Joe Marino

Stephane Ross

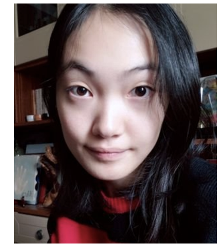


Aadyot Bhatnagar

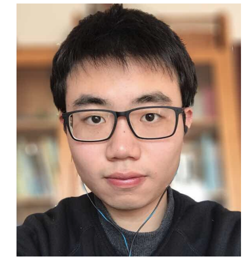
Albert Zhao



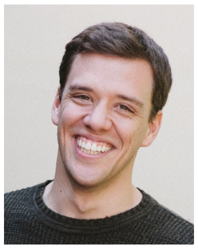
Yujia Huang



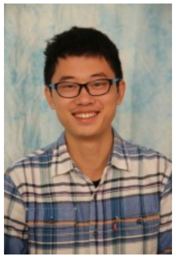
Sophie Dai



Hao Liu



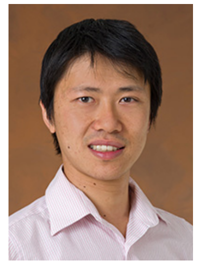
Milan Cvitkovic



Robin Zhou

Debadeepta Dey

Stephan Mandt



Hiro Ono

Drew Bagnell

Uduak Inyang-Udoh

Sandipan Mishra

Olivier Toupet

A

Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, et al., arXiv
Co-Training for Policy Learning, Jialin Song, Ravi Lanka, et al., UAI 2019
Learning Policies for Contextual Submodular Optimization, Stephane Ross et al., ICML
Iterative Amortized Inference, Joe Marino et al., ICML 2018
A General Framework for Amortizing Variational Filtering, Joe Marino et al, NeurIPS 20

https://github.com/ravi-lanka-4/CoPiEr
https://github.com/joelouismarino/iterative_inference