Caltech

Policy Learning with Certifiable Guarantees

Yisong Yue

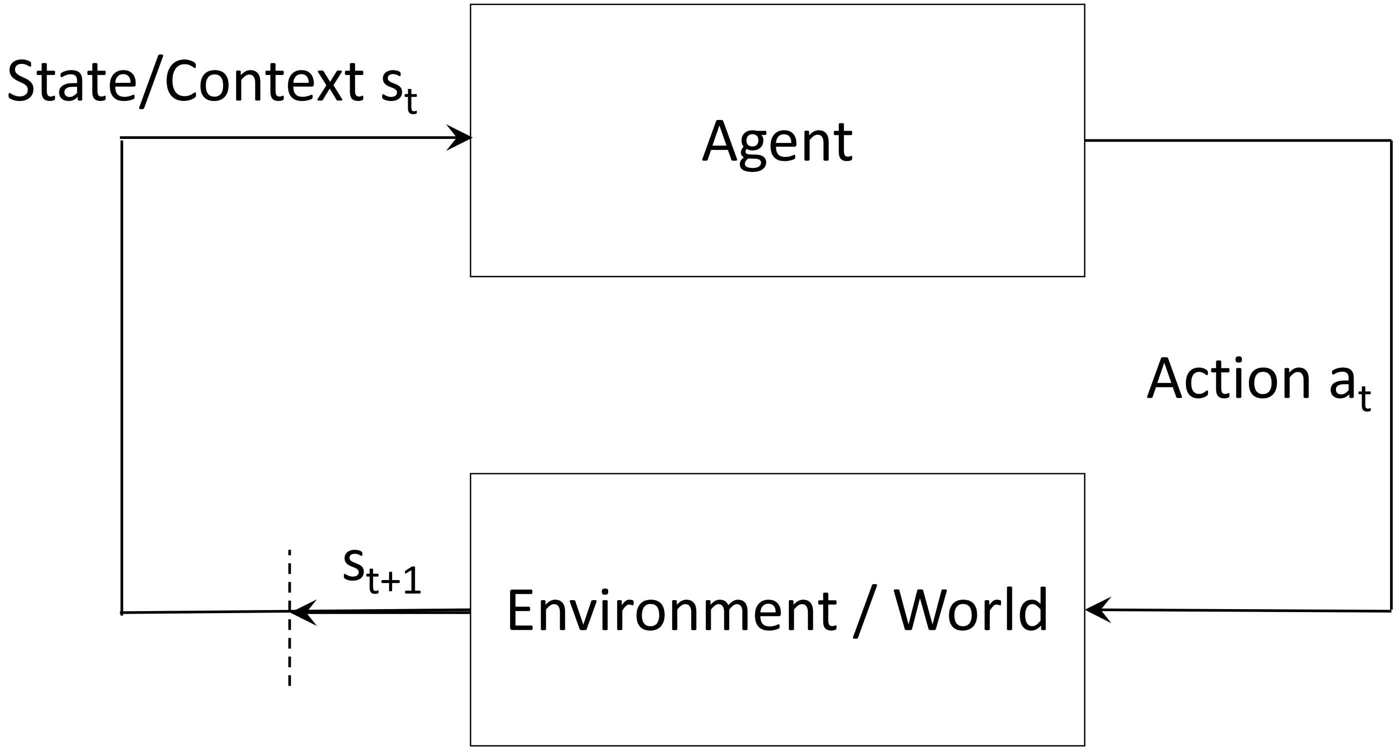
Policy Learning (Reinforcement & Imitation)

Goal: Find “Optimal” Policy

Imitation Learning:
Optimize imitation loss

Reinforcement Learning:
Optimize environmental reward

Learning-based Approach for Sequential Decision Making

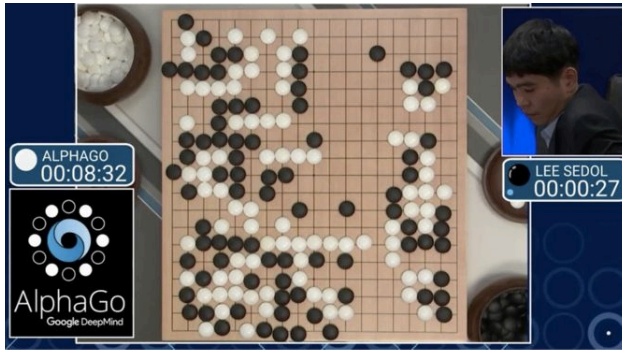


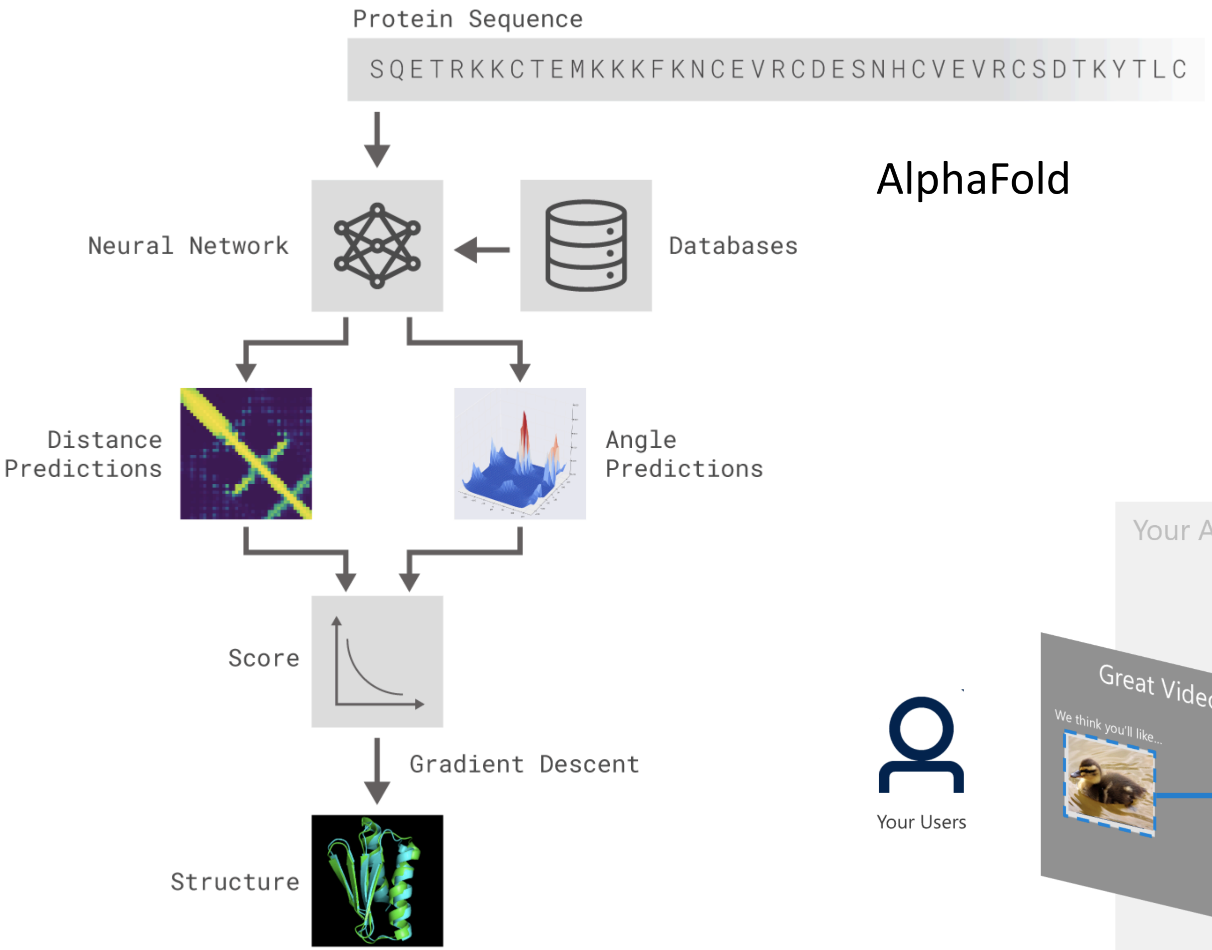
Imitation Learning Tutorial

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

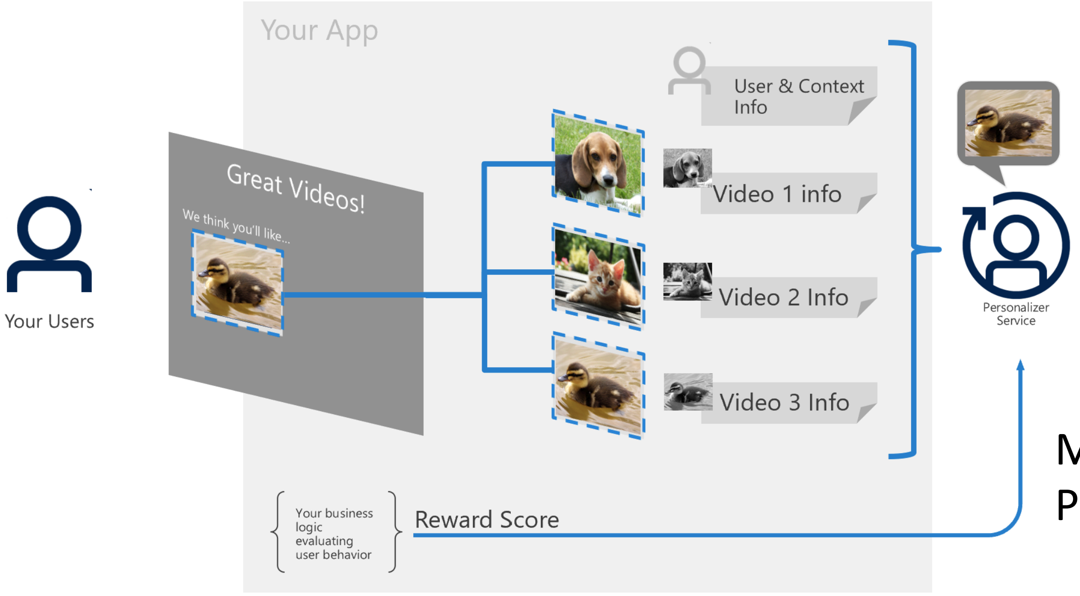
	Yisong Yue	Hoang M. Le
		
	yyue@caltech.edu	hmle@caltech.edu
	@YisongYue	@HoangMinhLe
	yisongyue.com	hoangle.info

Many Exciting Success Stories





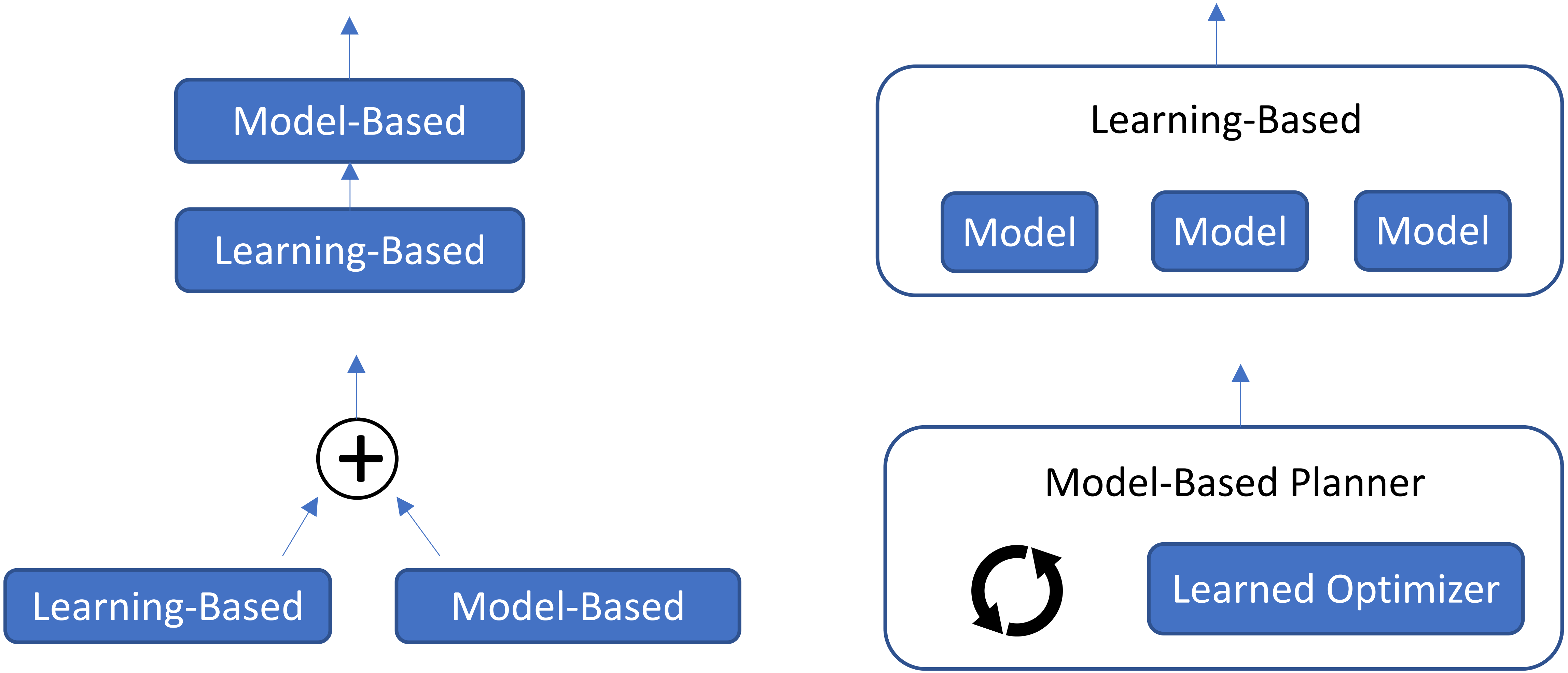
AlphaFold



Microsoft Azure Personalizer

" I want to use deep learning to optimize the design, manufacturing and operation of our aircrafts. But I need some guarantees. " -- Aerospace Director

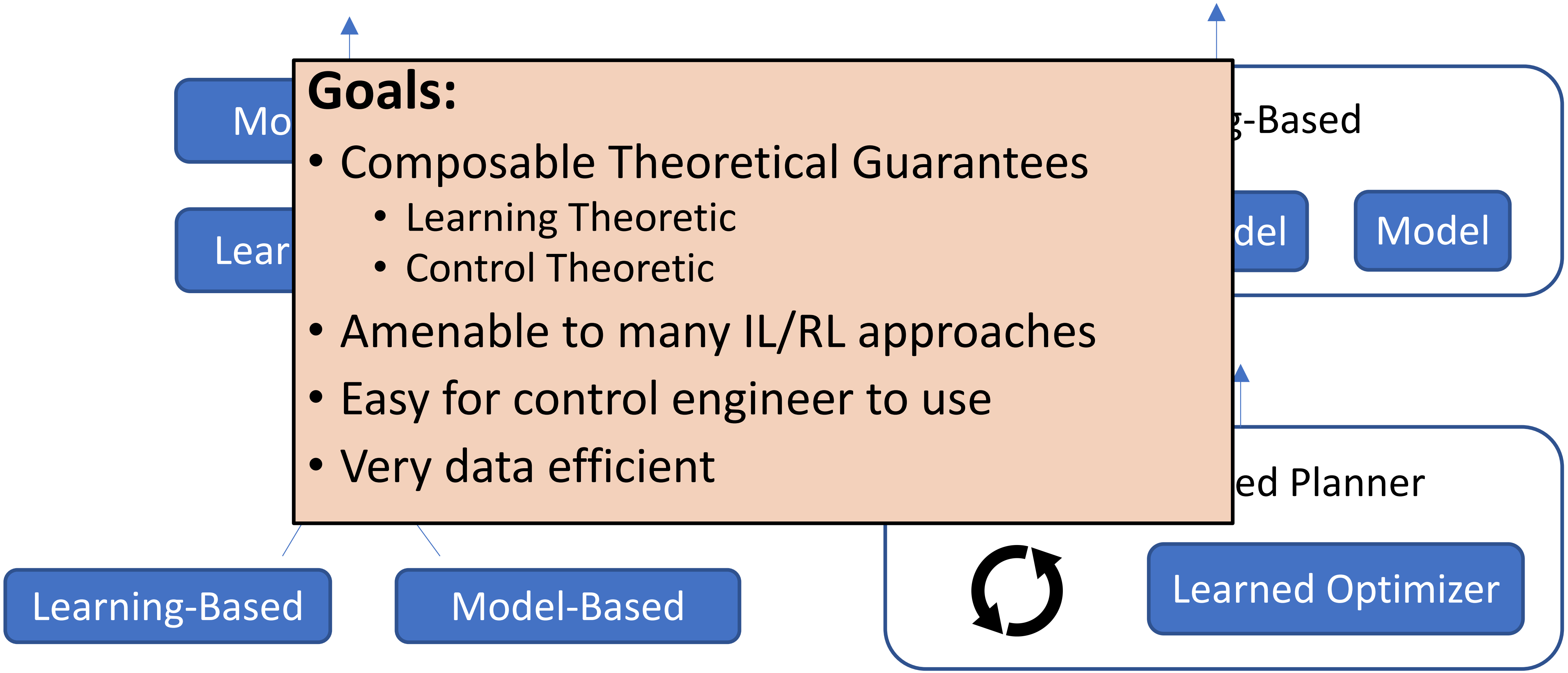
Blending Models/Rules & Black-Box Learning



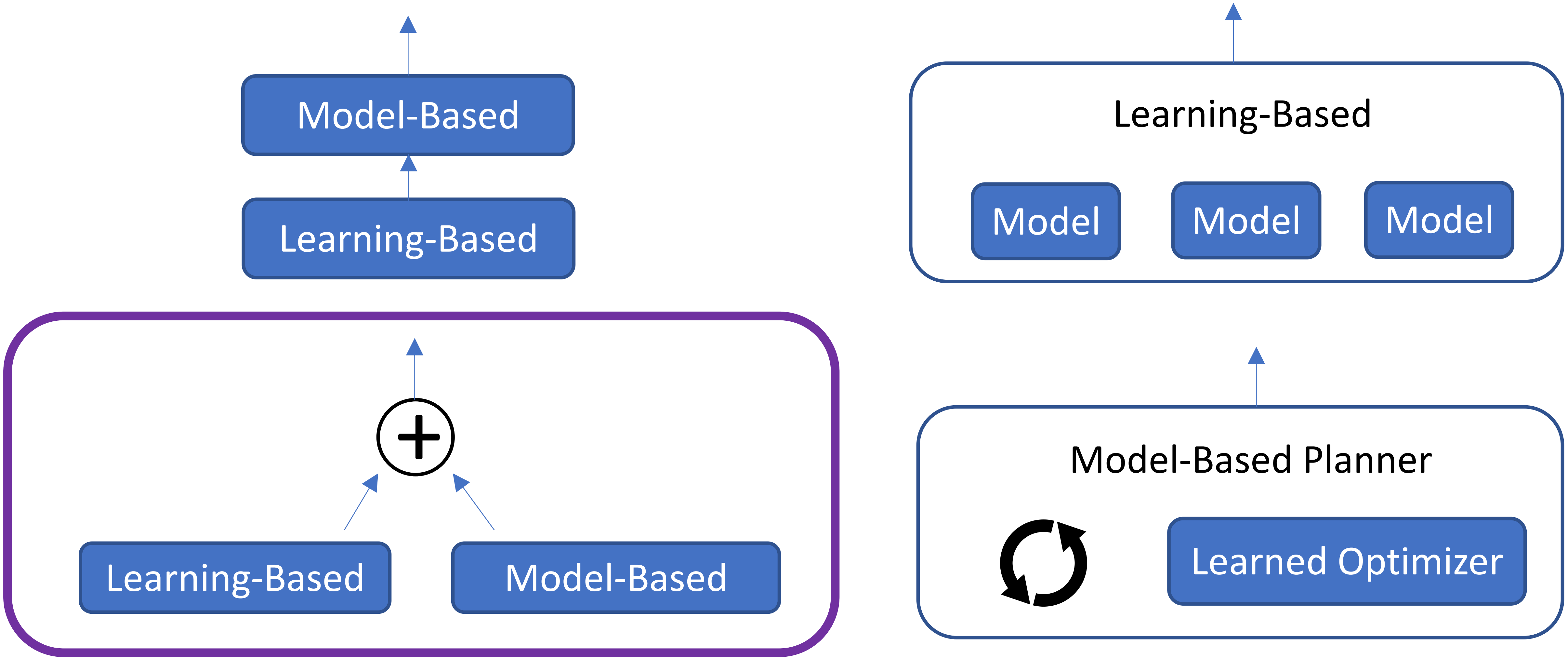
Blending Models/Rules & Black-Box Learning

Goals:

- Composable Theoretical Guarantees
  - Learning Theoretic
  - Control Theoretic
- Amenable to many IL/RL approaches
- Easy for control engineer to use
- Very data efficient



Blending Models/Rules & Black-Box Learning



Starting Point

- Model-Based/Free
- On/Off Policy
- Imitation/Reinforcement
- Optimal Control

Standard IL/RL Objective

$$\arg\min_h L(h)$$

$$s.t.$$

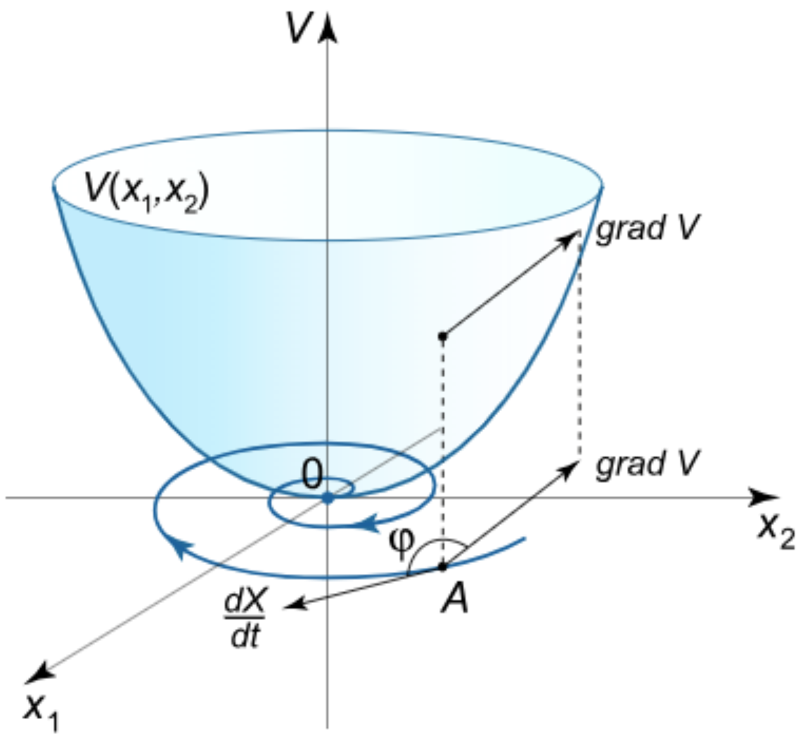
$$R(h) < \kappa$$

Side Constraint

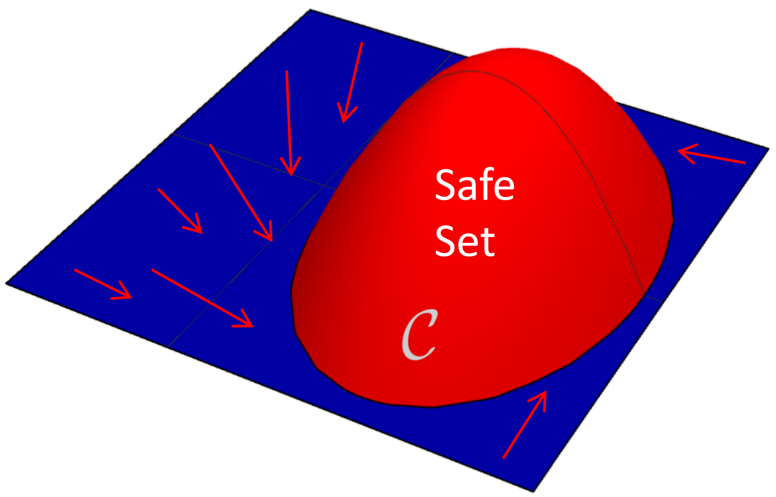
What can R encode?

Side Guarantees

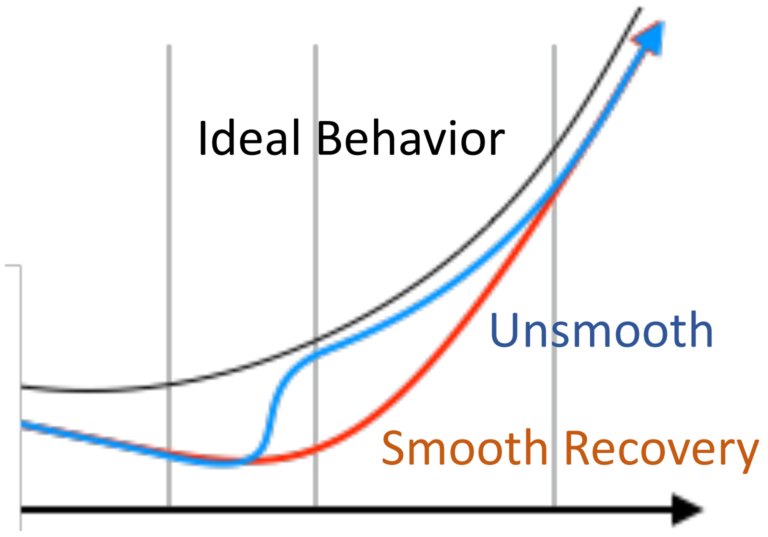
Possibly Others:
- Fairness
- Low-risk
- Temporal logic
- Etc…



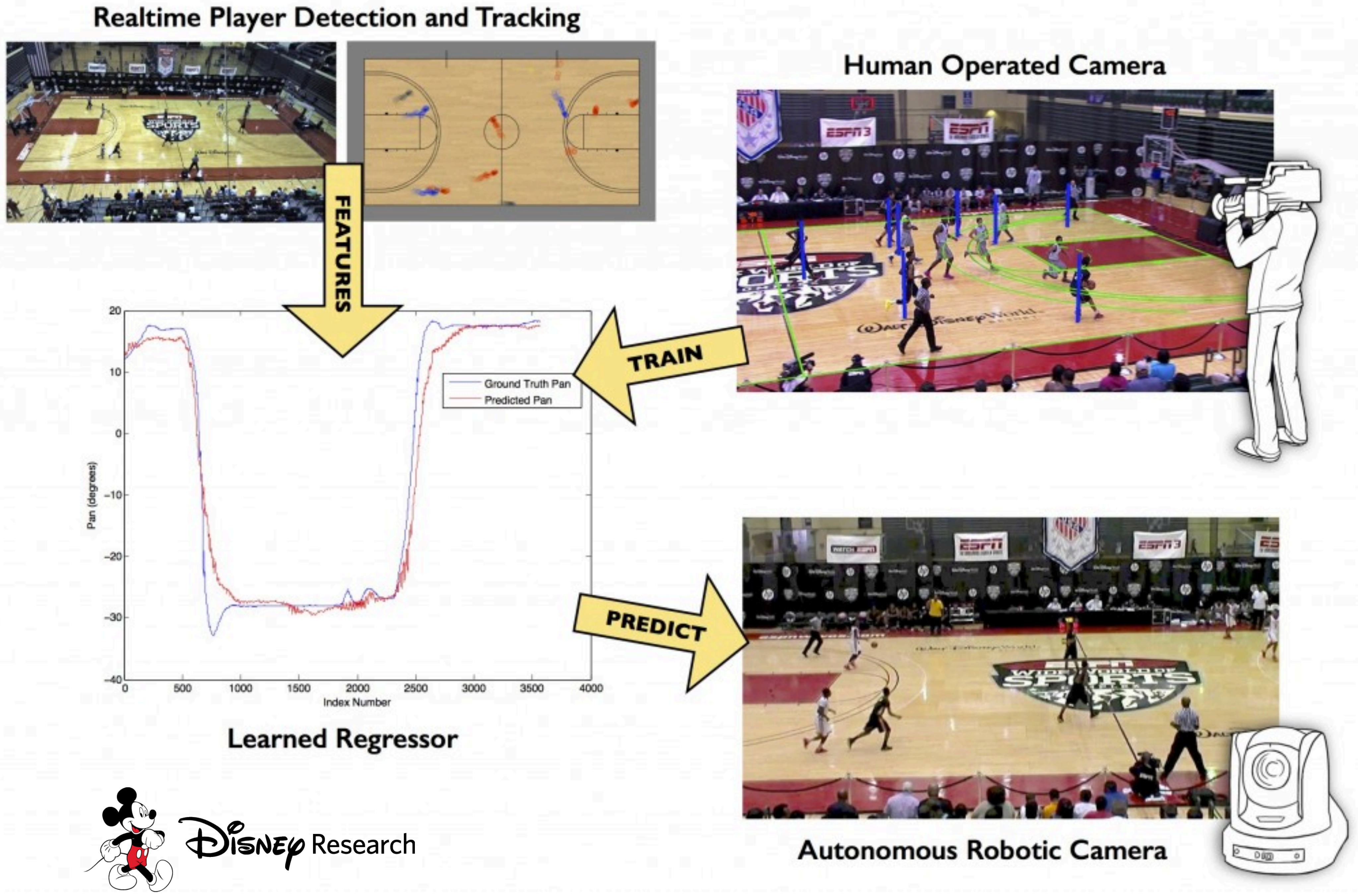
Stability



Safety

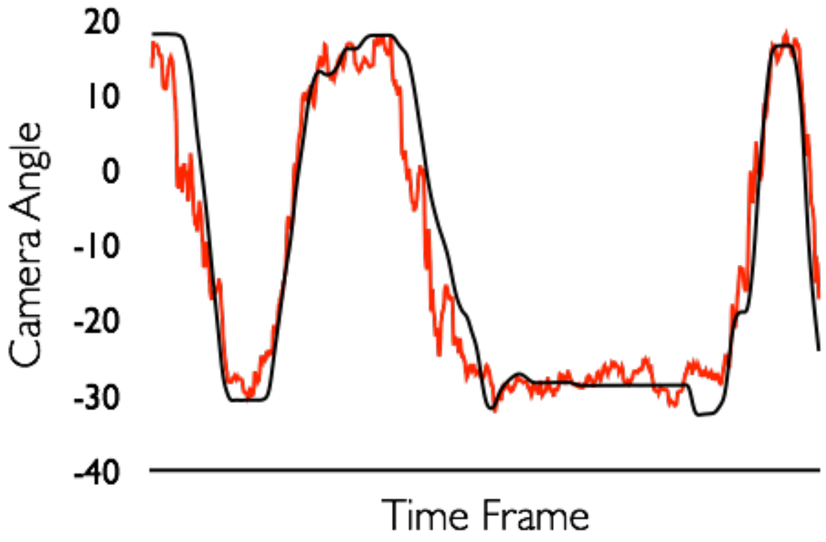


Smoothness

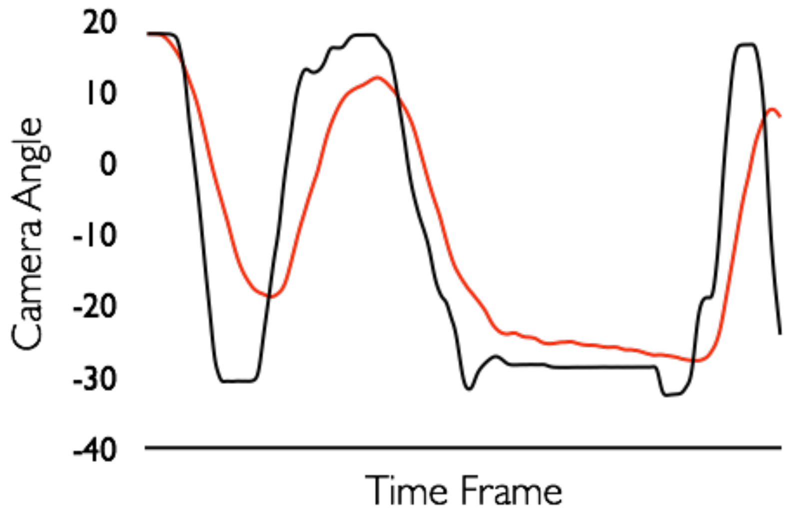

Realtime Player Detection and Tracking
Human Operated Camera
FEATURES
TRAIN
PREDICT
Ground Truth Pan
Predicted Pan
Pan (degrees)
Index Number
20
10
0
−10
−20
−30
−40
0
500
1000
1500
2000
2500
3000
3500
4000
Learned Regressor
Autonomous Robotic Camera
Disney Research

Naïve Approach

- Supervised learning of demonstration data
  - Train predictor per frame
  - Predict per frame



In practice, 2-step smoothing:



Starting Point

IL/RL Objective

$$\underset{h}{argmin} L(h) \quad s.t. \quad R(h) < \kappa$$

$$\Leftrightarrow$$

$$\underset{h}{argmin} L(h) + \lambda R(h)$$

Smooth Dynamics

What is R?

Regularize to Function Class
(h is "close to" some g)

$$\begin{aligned} & argmin_h L(h) \\ s.t. & \\ & \exists g \in G: ||h - g||^2 < \kappa \end{aligned}$$

$$\Leftrightarrow$$

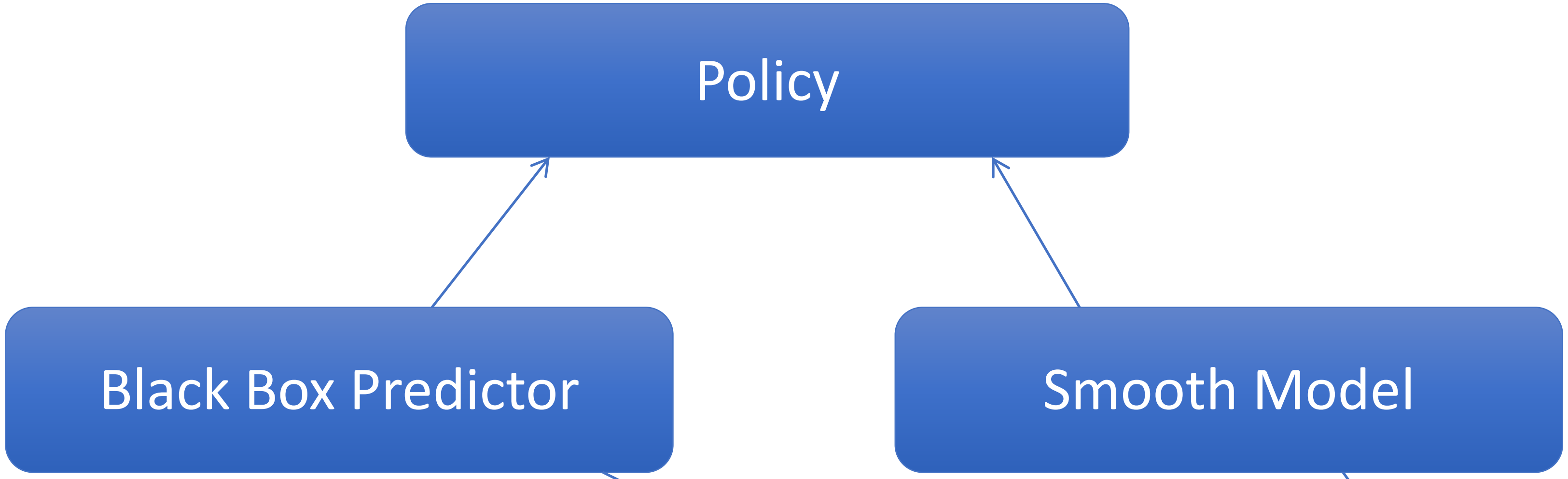
$$argmin_{h,g} L(h) + \lambda ||h - g||^2$$

Model-Based Controllers
(provably smooth)

Intractable?

Smooth Policy Class (solution concept)

Hoang Le



$$\mathrm{argmin}_{h=(f,g)} L(h) \quad \text{s.t.} \quad h(s) = \mathrm{argmin}_{a'} (f(s) - a')^2 + \lambda (g(s) - a')^2$$

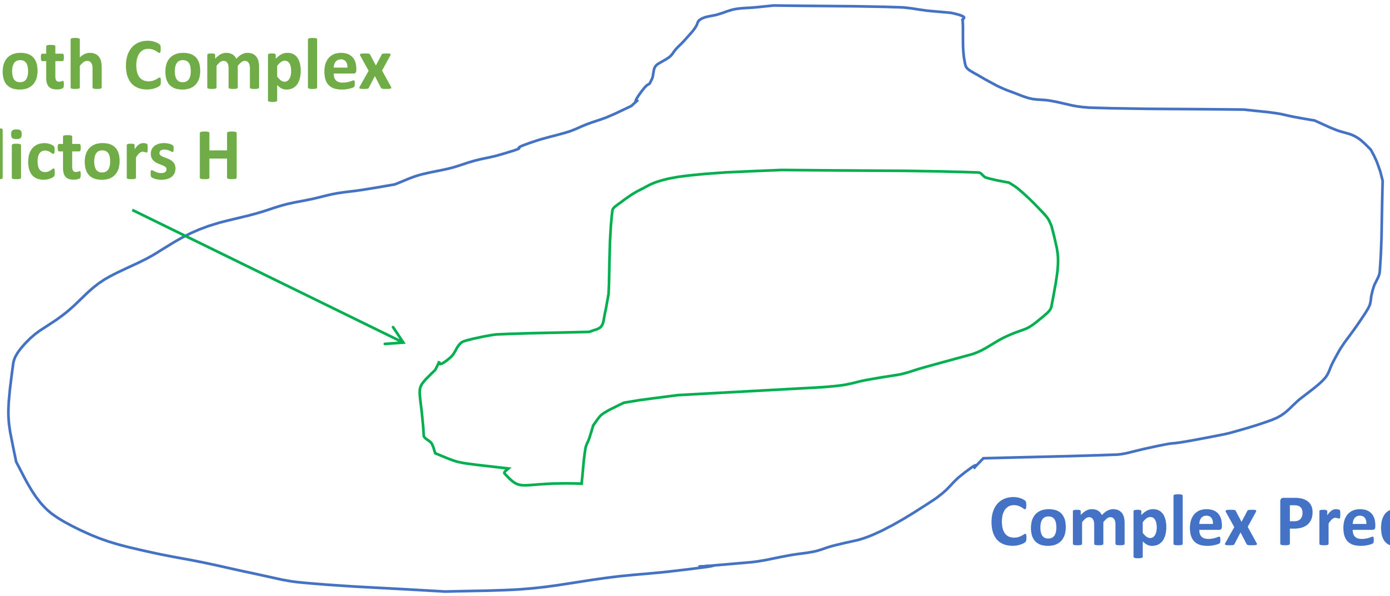
$$= \frac{f(s) + \lambda g(s)}{1 + \lambda}$$

Smooth Imitation Learning for Online Sequence Prediction
Hoang Le, Andrew Kang, Yisong Yue, Peter Carr. ICML 2016

Test-Time Functional Regularization

Hoang Le

Smooth Complex Predictors H



Complex Predictors F

$$\mathrm{argmin}_{h=(f,g)} L(h) \quad \text{s.t.} \quad h(s) = \mathrm{argmin}_{a'} (f(s) - a')^2 + \lambda (g(s) - a')^2$$

$$= \frac{f(s) + \lambda g(s)}{1 + \lambda}$$

Smooth Imitation Learning for Online Sequence Prediction
Hoang Le, Andrew Kang, Yisong Yue, Peter Carr. ICML 2016

Basic Algorithmic Recipe

$$argmin_{h=(f,g)} L(h) \quad \text{s.t.} \quad h(s) = argmin_{a'} (f(s) - a')^2 + \lambda (g(s) - a')^2$$
$$= \frac{f(s) + \lambda g(s)}{1 + \lambda}$$

1. Initialize g
2. Hold g fixed, train f using standard policy learning
3. Hold h fixed, estimate better g to characterize h
4. Repeat from Step 1

Basic Algorithmic Recipe

$argmin_{h=(f,g)} L(h)$ ... $(s) - a')^2$

Theoretical Questions:

- Does having g help with learning?
- Can we preserve properties of g?
- Can we leverage existing work as subroutines?

Practical Questions

- Is it easy for a practitioner to use?

1. Initialize g
2. Hold g fixed, train f using standard policy learning
3. Hold h fixed, estimate better g to characterize h
4. Repeat from Step 1

Smooth Imitation Learning for Online Sequence Prediction
Hoang Le, Andrew Kang, Yisong Yue, Peter Carr. ICML 2016

Summary of Theoretical Guarantees

$$argmin_{h=(f,g)} L(h) \quad \text{s.t.} \quad h(s) = argmin_{a'} (f(s) - a')^2 + \lambda (g(s) - a')^2$$
$$= \frac{f(s)+\lambda g(s)}{1+\lambda}$$

- By construction: h "close" to g
  - Certifications on g => (relaxed) certifications on h

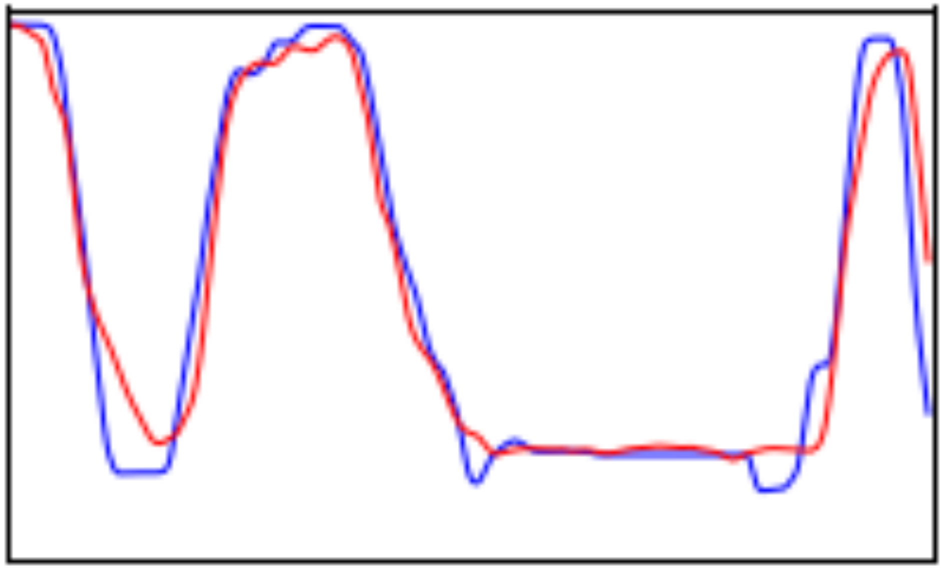
Run-time regularization

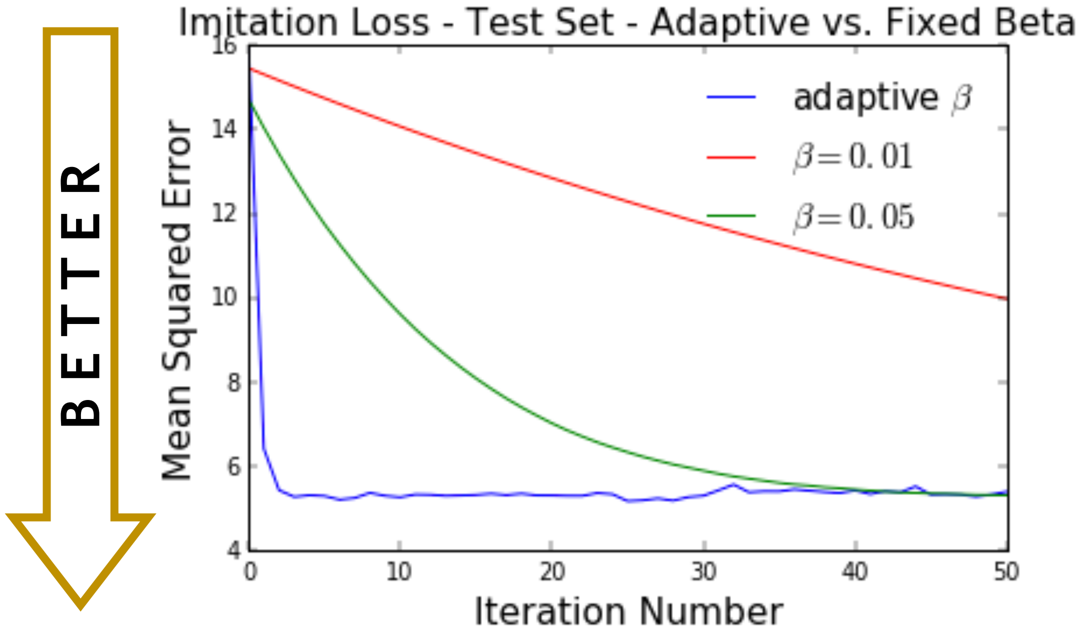
E.g., "smoothness"

- Compatible with many forms of IL/RL
  - Can be exponentially faster than prior work (SEARN)

Adaptive Step Size
Exploits Lipschitz

Our Results



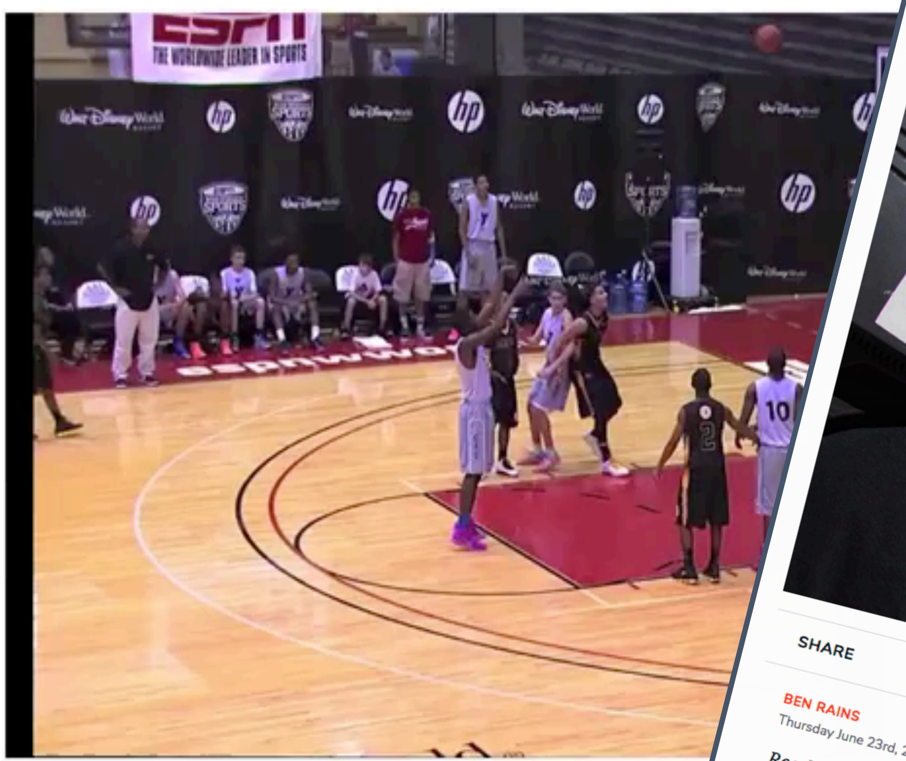


Provably Smooth Predictions
(G = linear autoregressors)

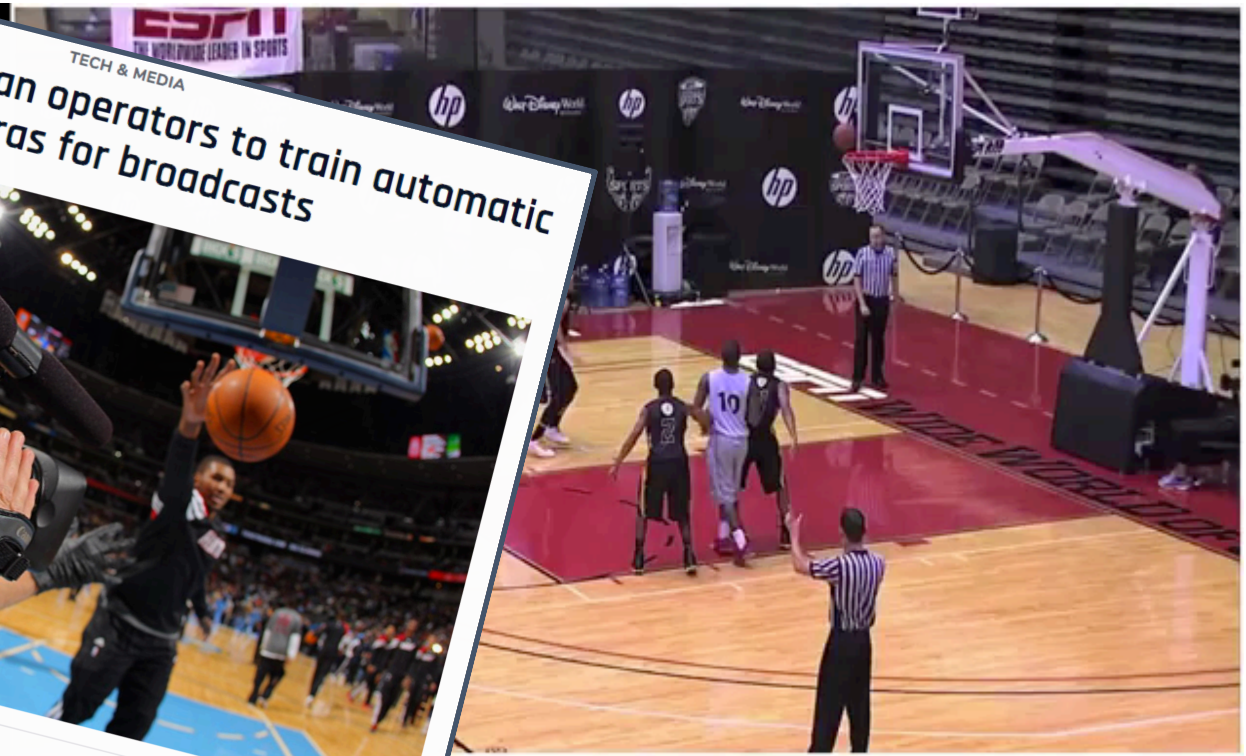
Provably Faster Learning
(Natural Policy Updates)

Smooth Imitation Learning for Online Sequence Prediction
Hoang Le, Andrew Kang, Yisong Yue, Peter Carr. ICML 2016

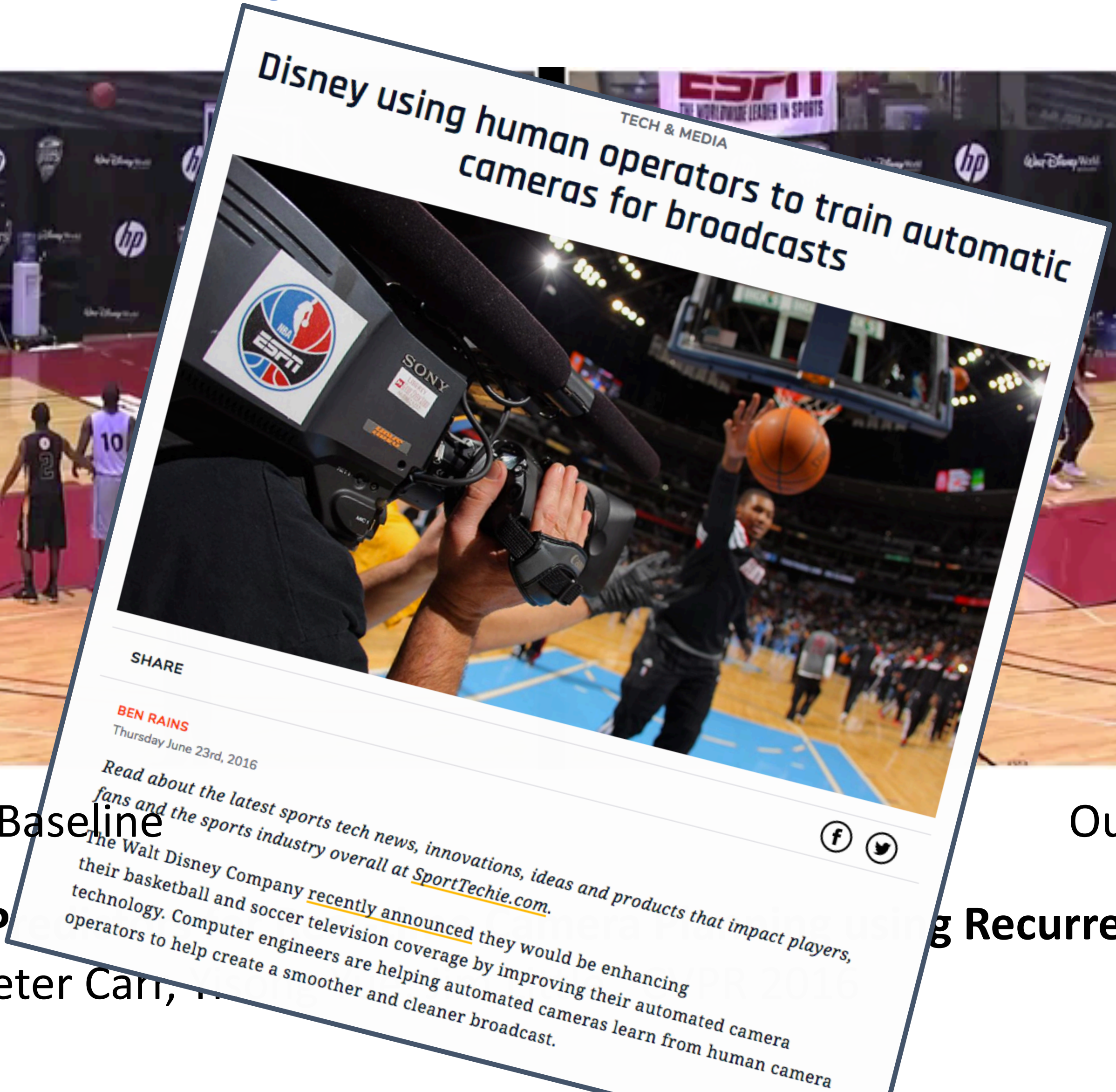
Qualitative Comparison



2-Step Baseline



Our Approach



TECH & MEDIA

Disney using human operators to train automatic cameras for broadcasts

SHARE

BEN RAINS
Thursday June 23rd, 2016

Read about the latest sports tech news, innovations, ideas and products that impact players, fans and the sports industry overall at SportTechie.com.

The Walt Disney Company recently announced they would be enhancing their basketball and soccer television coverage by improving their automated camera technology. Computer engineers are helping automated cameras learn from human camera operators to help create a smoother and cleaner broadcast.

Learning Online Smooth P... g Recurrent Decision Trees

Jianhui Chen, Hoang Le, Peter Carr, ...

Generalized Control Regularization

Richard
Cheng

$$h(s) = \frac{f(s)+\lambda g(s)}{1+\lambda}$$

- f is black box learning
- g is “control prior”   (e.g., H-infinity controller)

- Learn f using policy gradient using any standard RL method

Control Regularization for Reduced Variance Reinforcement Learning
Richard Cheng, Abhinav Verma, Gabor Orosz, Swarat Chaudhuri, Yisong Yue, Joel Burdick. ICML 2019

Generalized Control Regularization

Richard Cheng

$$h(s) = \frac{f(s)+\lambda g(s)}{1+\lambda}$$

- Theorem (informal):
  - Variance of policy gradient decreases by factor of: $\left(\frac{1}{1+\lambda}\right)^2$
  - Bias converges to: $D_{TV}(h^*, g)$

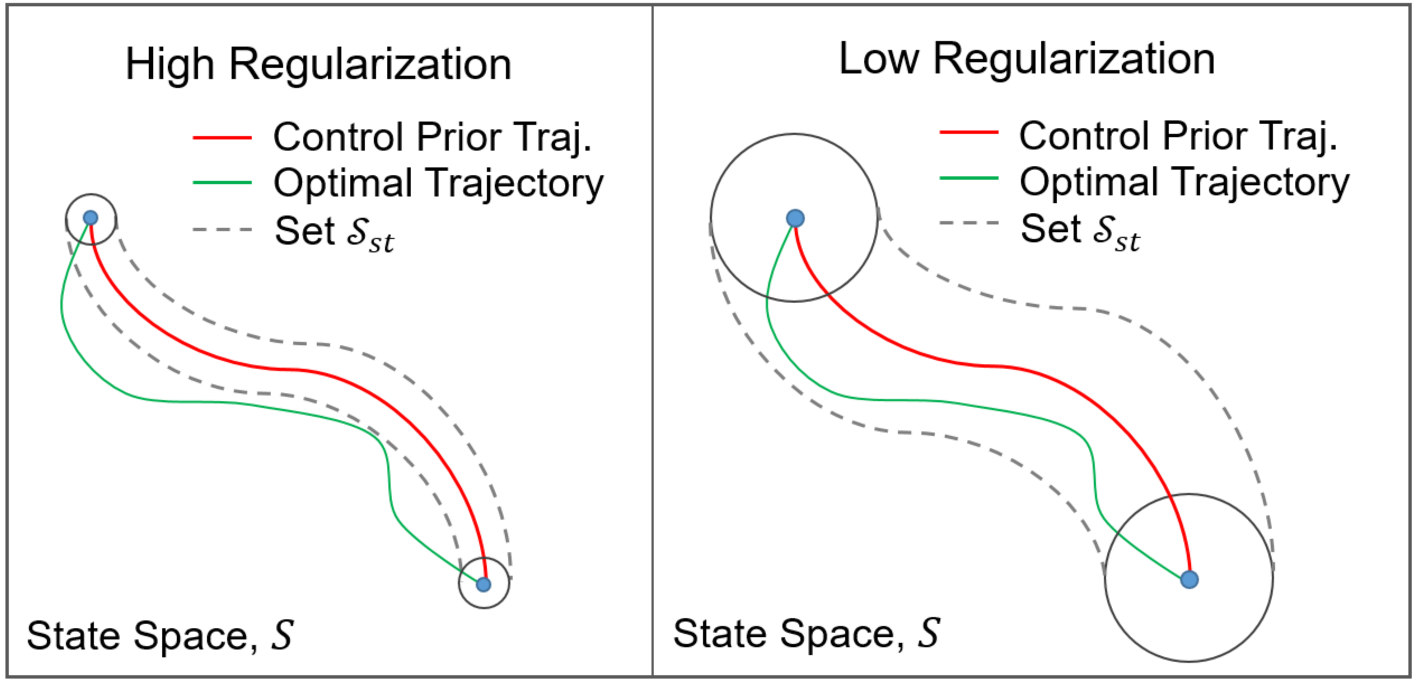
Implies much faster learning!

Control Regularization for Reduced Variance Reinforcement Learning
Richard Cheng, Abhinav Verma, Gabor Orosz, Swarat Chaudhuri, Yisong Yue, Joel Burdick. ICML 2019

Generalized Control Regularization

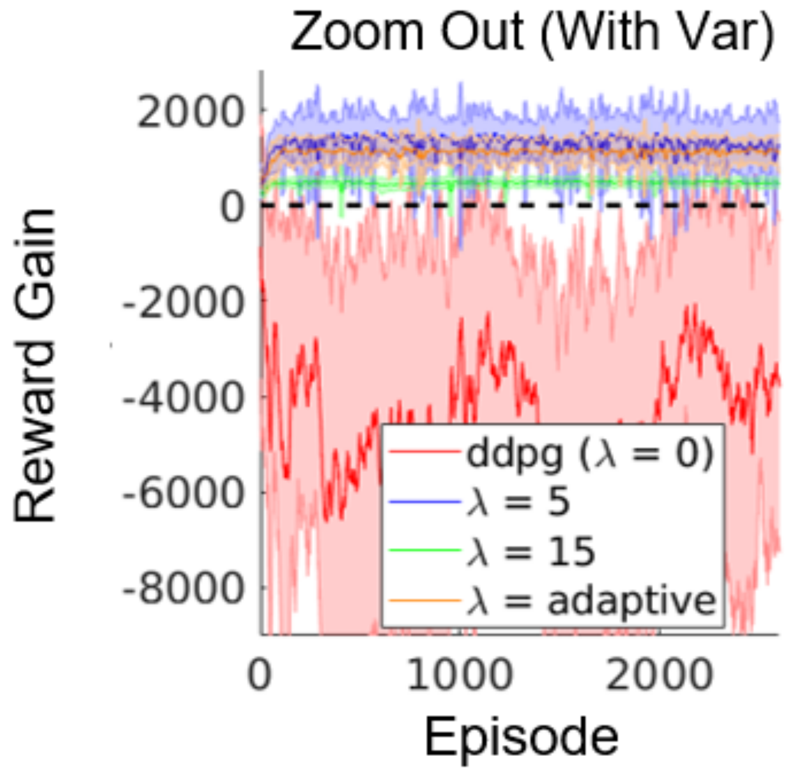
Richard Cheng

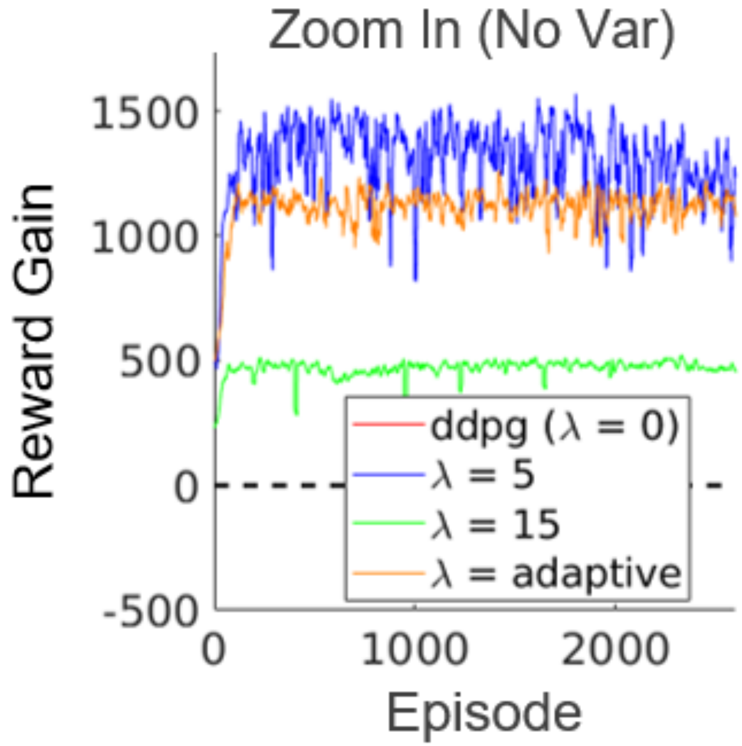
- (Relaxed) Lyapunov stability bounds:

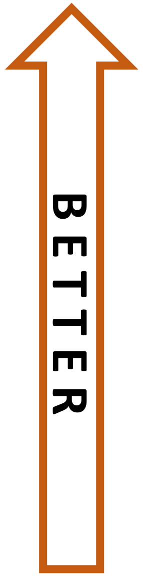


Control Regularization for Reduced Variance Reinforcement Learning
Richard Cheng, Abhinav Verma, Gabor Orosz, Swarat Chaudhuri, Yisong Yue, Joel Burdick. ICML 2019

Generalized Control Regularization







Richard Cheng

Control Regularization for Reduced Variance Reinforcement Learning
Richard Cheng, Abhinav Verma, Gabor Orosz, Swarat Chaudhuri, Yisong Yue, Joel Burdick. ICML 2019


Ready
3 4 5 6 7
120 180 240

Improving Control Prior?

Abhinav
Verma

Hoang
Le

Recall Algorithmic Recipe:

1. Initialize g
2. Hold g fixed, train f using standard policy learning
3. Hold h fixed, estimate better g to characterize h
4. Repeat from Step 1

How to synthesize g?

Imitation-Projected Policy Gradient for Programmatic Reinforcement Learning
Abhinav Verma, Hoang Le, Yisong Yue, Swarat Chaudhuri. NeurIPS 2019

Aside: Batch Learning

Hoang Le

- Suppose learning on historical data ("off-policy")
- How to ensure that constraint is satisfied (with high probability)?

$$\begin{aligned} &\operatorname{argmin}_h L(h) \\ &s.t. \\ &R(h) < \delta \end{aligned} \quad \Longrightarrow \quad \operatorname{argmin}_h \max_\lambda L(h) + \lambda(R(h) - \delta)$$

- Convert learning into 2-player game on Lagrangian
  - h player plays best response
  - λ player plays no-regret online learning
- **PAC-guarantees on constraint satisfaction** ←

Satisfying constraints in training set
→
ε-satisfaction in test set W.P. 1-δ

Batch Policy Learning under Constraints
Hoang Le, Cameron Voloshin, Yisong Yue. ICML 2019

Summary: Functional Regularization

Equivalence Between Regularization & Constrained Learning

IL/RL Objective

$$argmin_h L(h) \quad s.t. \quad R(h) < \kappa \iff argmin_h L(h) + \lambda R(h)$$

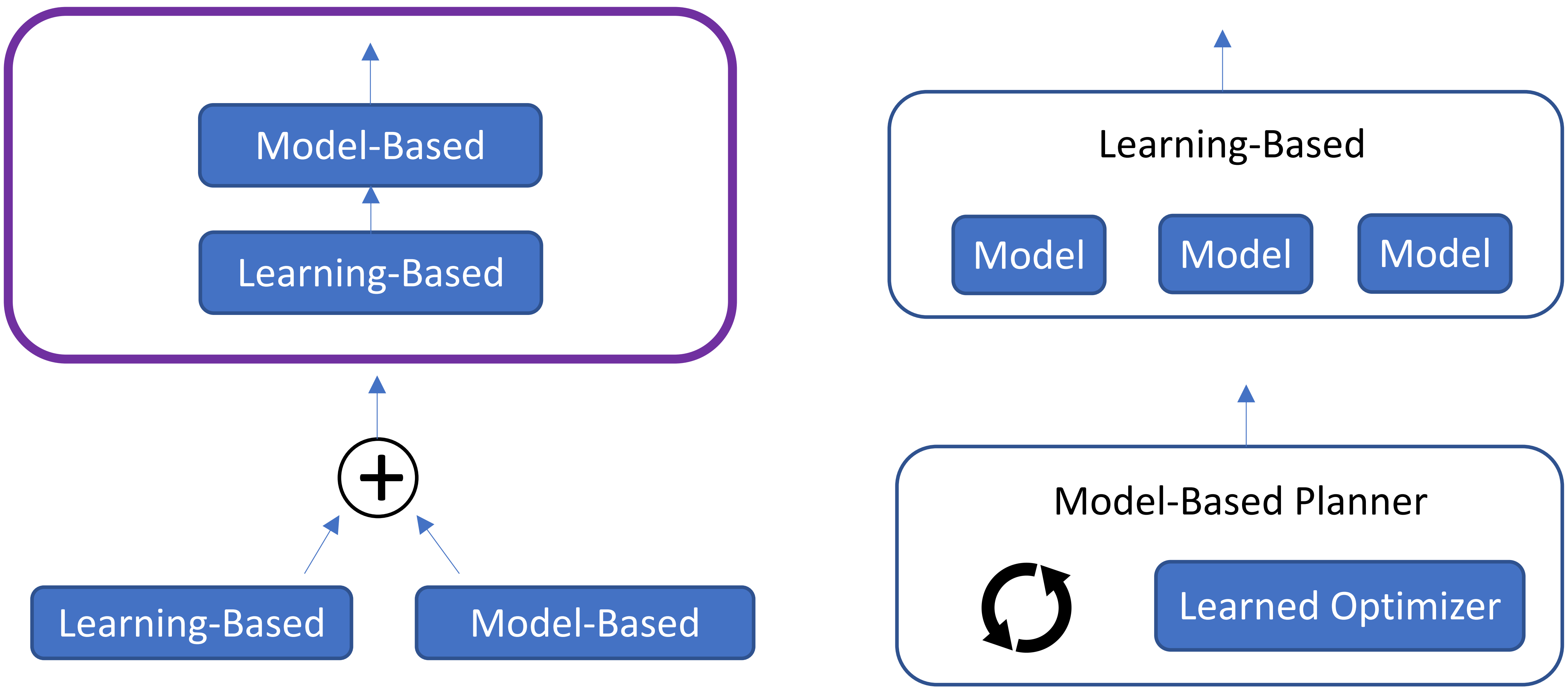
Side Guarantees

Hybrid Policy Solution Concept

$$h(s) = argmin_{a'} (f(s) - a')^2 + \lambda (g(s) - a')^2$$

$$= \frac{f(s) + \lambda g(s)}{1 + \lambda}$$

Blending Models/Rules & Black-Box Learning



Model-Based Control

New State

Current Action (aka control input)

$$s_{t+1} = F(s_t, u_t) + \epsilon$$

Current State

Unmodeled Disturbance / Error

(Value Iteration is also contraction mapping)

Robust Control (fancy contraction mappings)

- Stability guarantees (e.g., Lyapunov)
- Precision/optimality depends on error

Learning Residual Dynamics

F = nominal dynamics
$\tilde{F}$ = learned dynamics

New State

Current Action (aka control input)

$$s_{t+1} = F(s_t, u_t) + \tilde{F}(s_t, u_t) + \epsilon$$

Current State

Unmodeled Disturbance / Error

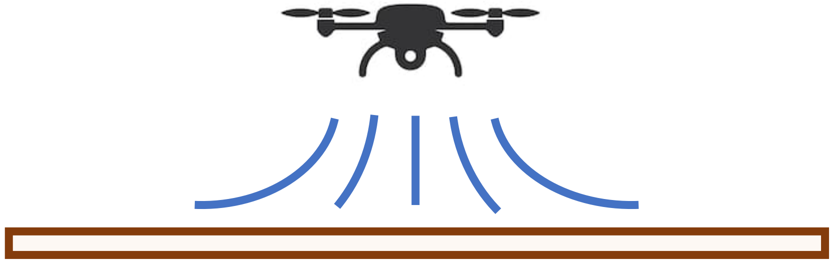
Leverage robust control (fancy contraction mappings)

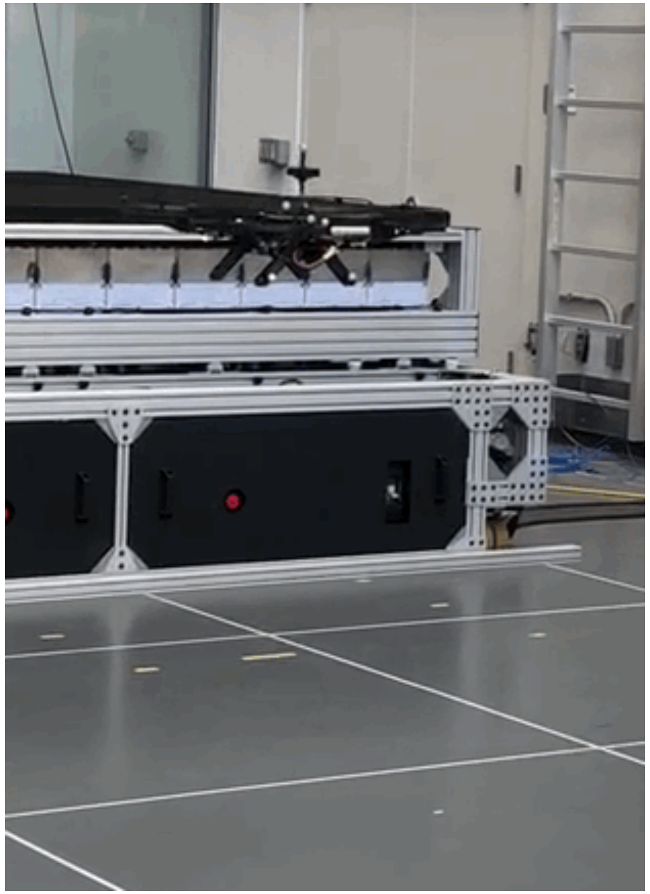
- Preserve stability (even using deep learning)
- Requires $\tilde{F}$ Lipschitz & bounded error

Stable Drone Landing

Guanya
Shi

Ground effect





Neural Lander: Stable Drone Landing Control using Learned Dynamics
Guanya Shi, Xichen Shi, Michael O'Connell, Rose Yu, Kamyar Azizzadenesheli, Anima Anandkumar, Yisong Yue, Soon-Jo Chung. ICRA 2019

Control System Formulation

Learn the Residual

- Dynamics:

$$\begin{cases} \dot{\mathbf{p}} = \mathbf{v}, & m\dot{\mathbf{v}} = m\mathbf{g} + R\mathbf{f}_u + \mathbf{f}_a \\ \dot{R} = RS(\boldsymbol{\omega}), & J\dot{\boldsymbol{\omega}} = J\boldsymbol{\omega} \times \boldsymbol{\omega} + \boldsymbol{\tau}_u + \boldsymbol{\tau}_a \end{cases}$$

- Control:

$$\begin{cases} \mathbf{f}_u = [0, 0, T]^\top \\ \boldsymbol{\tau}_u = [\tau_x, \tau_y, \tau_z]^\top \\ \begin{bmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} c_T & c_T & c_T & c_T \\ 0 & c_T l_{\text{arm}} & 0 & -c_T l_{\text{arm}} \\ -c_T l_{\text{arm}} & 0 & c_T l_{\text{arm}} & 0 \\ -c_Q & c_Q & -c_Q & c_Q \end{bmatrix} \begin{bmatrix} n_1^2 \\ n_2^2 \\ n_3^2 \\ n_4^2 \end{bmatrix} \end{cases}$$

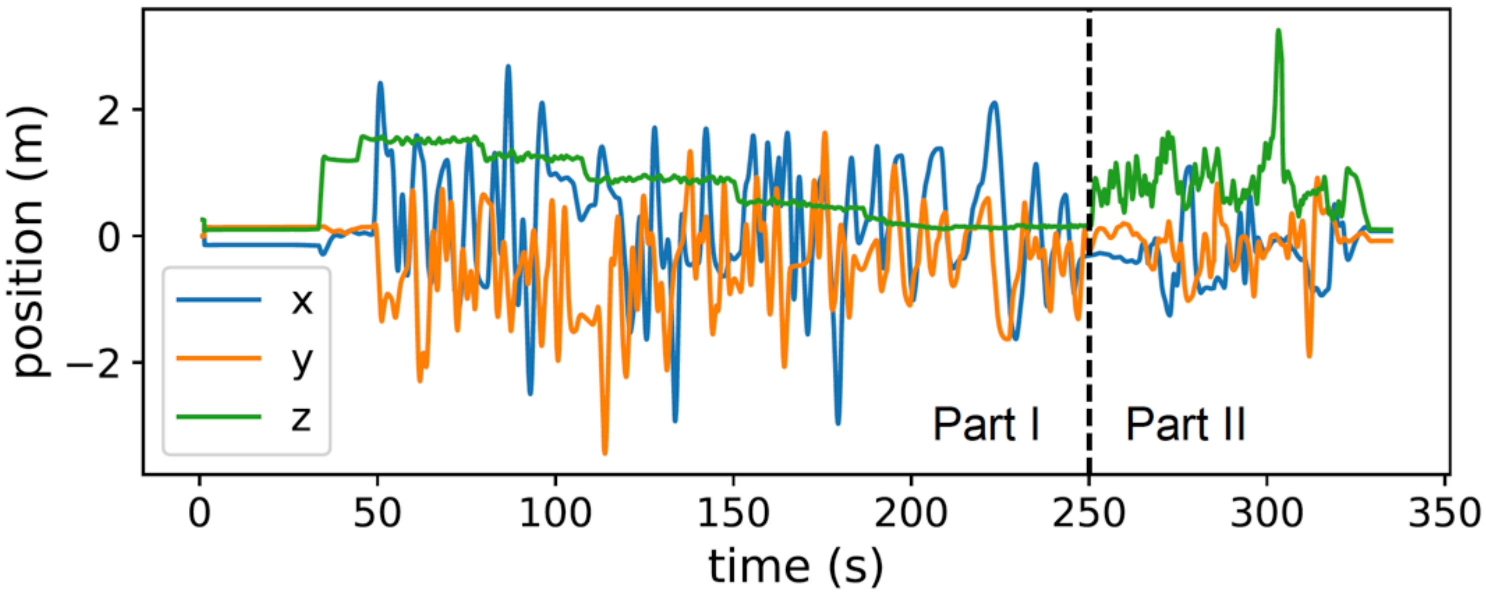
- Unknown forces & moments:

$$\begin{cases} \mathbf{f}_a = [f_{a,x}, f_{a,y}, f_{a,z}]^\top \\ \boldsymbol{\tau}_a = [\tau_{a,x}, \tau_{a,y}, \tau_{a,z}]^\top \end{cases}$$

Learn the Residual

Data Collection (Manual Exploration)

Current Research:
Safe Exploration

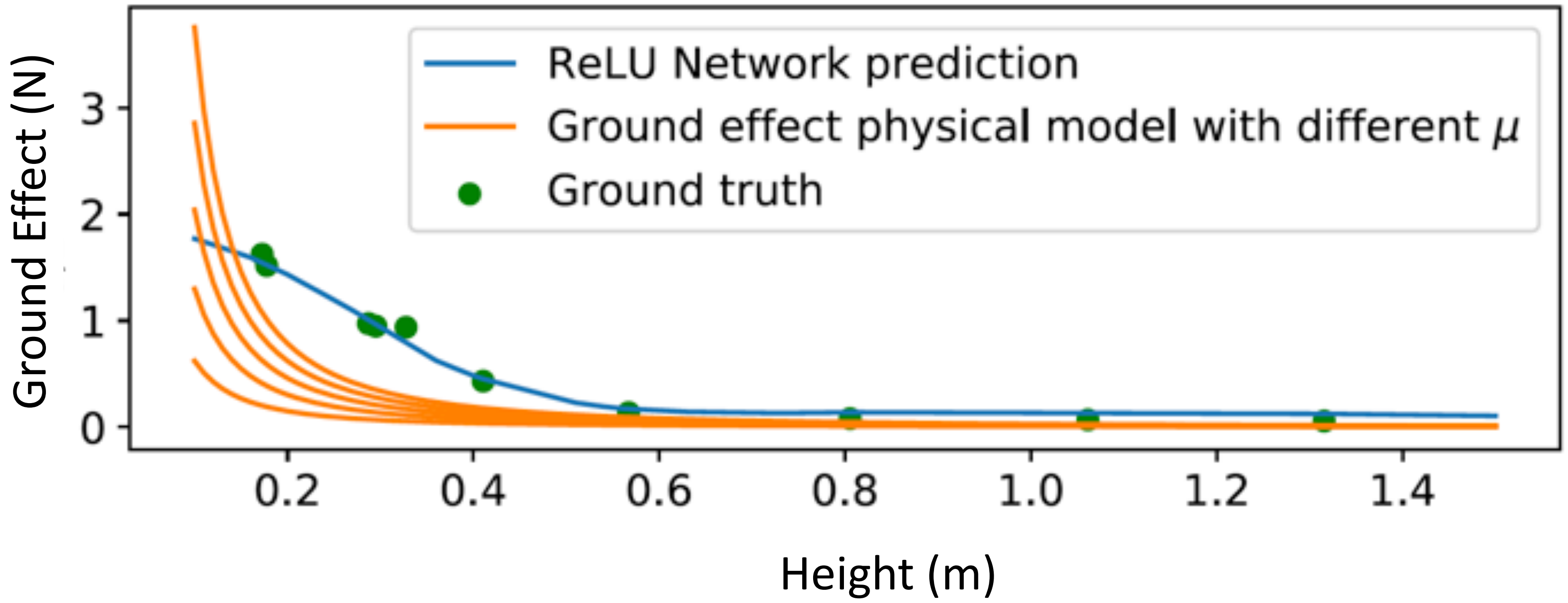


Ensures $\widetilde{F}$ is Lipshitz
[Bartlett et al., NeurIPS 2017]
[Miyato et al., ICLR 2018]

Spectral-Normalized 4-Layer Feed-Forward

- Learn ground effect: $\tilde{F}(s,u) \rightarrow \mathbf{f}_a = [f_{a,x}, f_{a,y}, f_{a,z}]^\top$
- (s,u): height, velocity, attitude and four control inputs

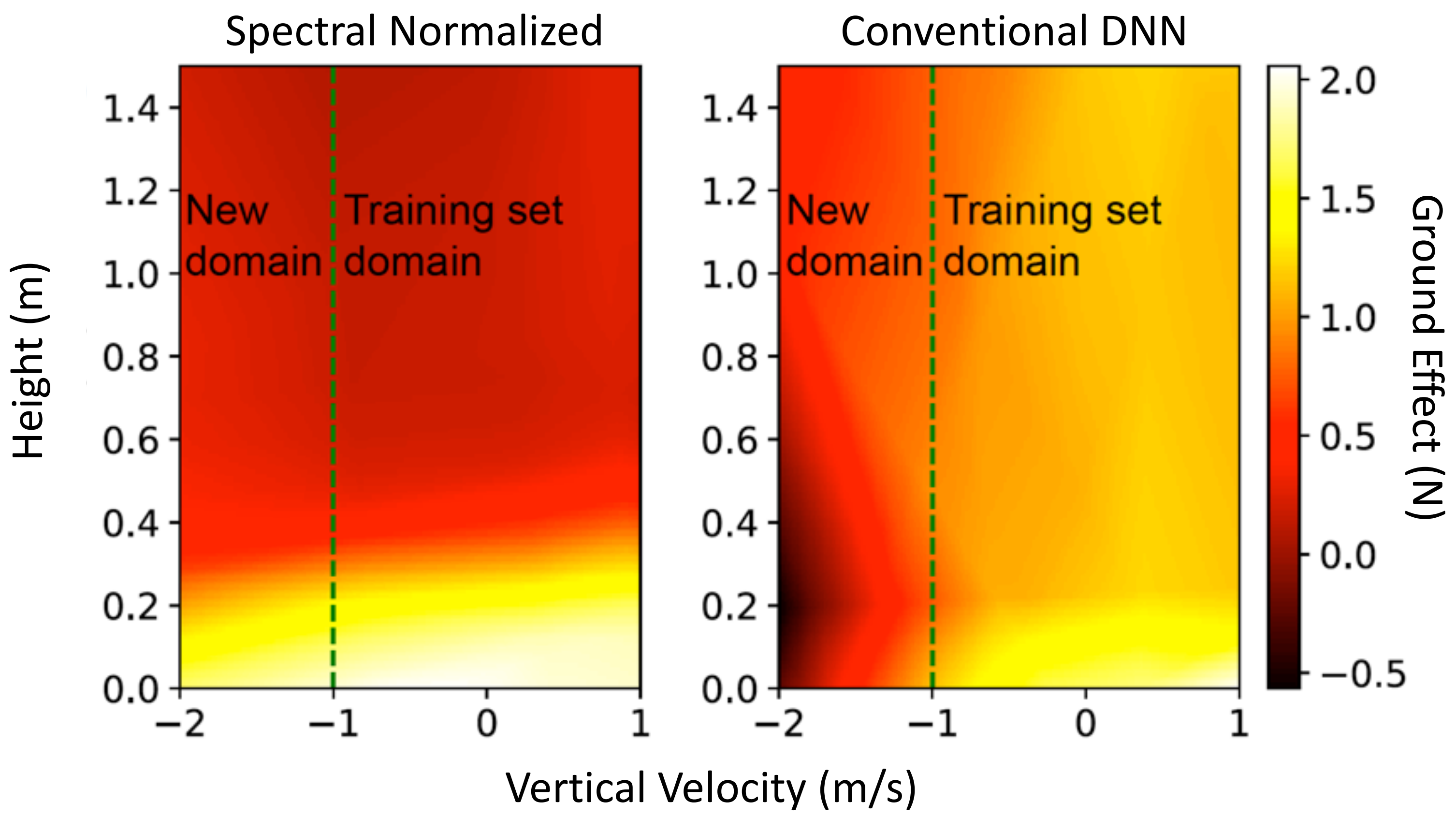
Prediction Results



Neural Lander: Stable Drone Landing Control using Learned Dynamics
Guanya Shi, Xichen Shi, Michael O'Connell, Rose Yu, Kamyar Azizzadenesheli, Anima Anandkumar, Yisong Yue, Soon-Jo Chung. ICRA 2019.

Current Research:
Quantifying Extrapolation

Prediction Results



Neural Lander: Stable Drone Landing Control using Learned Dynamics
Guanya Shi, Xichen Shi, Michael O'Connell, Rose Yu, Kamyar Azizzadenesheli, Anima Anandkumar, Yisong Yue, Soon-Jo Chung. ICRA 2019.

Controller Design (simplified)

Guanya Shi

- Nonlinear Feedback Linearization:

$$u_{nominal} = K_s \eta \qquad \eta = \begin{bmatrix} p - p^* \\ v - v^* \end{bmatrix}$$

Desired Trajectory
(tracking error)

Feedback Linearization (PD control)

- Cancel out ground effect $\tilde{F}(s, u_{old})$: $u = u_{nominal} + u_{residual}$

Requires Lipschitz & small time delay

Controller Design (simplified)

Guanya Shi

- Nonlinear Feedback Linearization:

$$u_{nominal} = K_s \eta \qquad \eta = \begin{bmatrix} p - p^* \\ v - v^* \end{bmatrix}$$

Desired Trajectory
(tracking error)

Feedback Linearization (PD control)

- Cancel out ground effect $\tilde{F}(s, u_{old})$: $u = u_{nominal} + u_{residual}$

(time delay)

Requires Lipschitz & small time delay

Controller Design (simplified)


Guanya Shi

- Nonlinear Feedback Linearization:

$$u_{nominal} = K_s \eta \qquad \eta = \begin{bmatrix} p - p^* \\ v - v^* \end{bmatrix}$$

Desired Trajectory
(tracking error)

Stability Guarantee:
(simplified)

Time delay

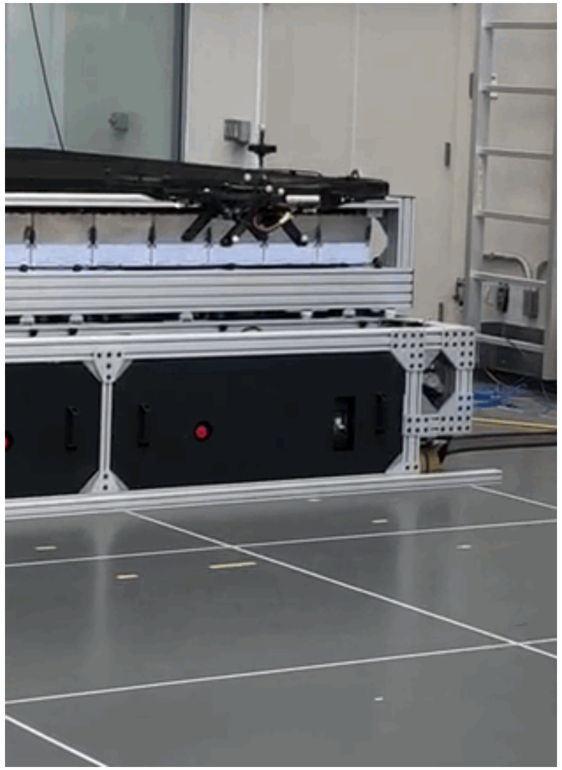
Unmodeled disturbance

$$\|\eta(\mathrm{t})\| \leq \|\eta(0)\| \exp\left\{\frac{\lambda_{min}(K) - \tilde{L}\rho}{C} t\right\} + \frac{\epsilon}{\lambda_{min}(K) - \tilde{L}\rho}$$

Lipschitz of NN

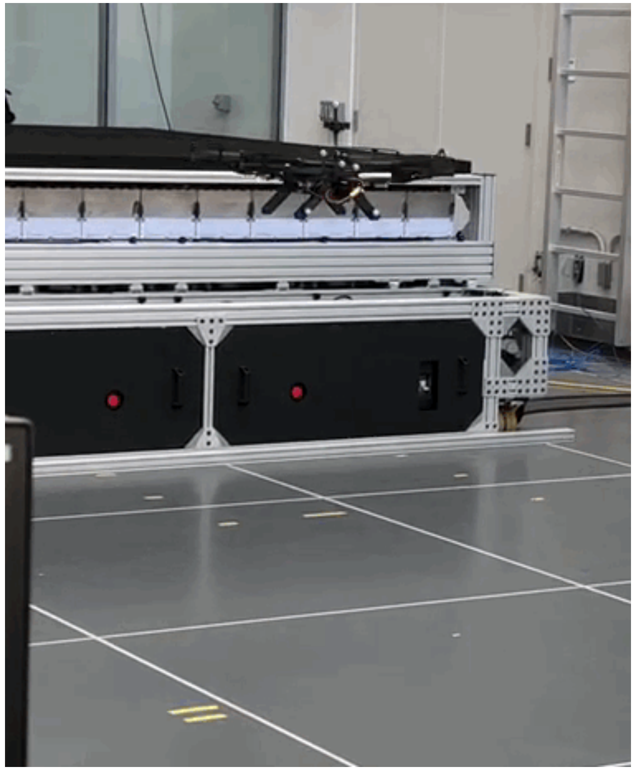
$$\Longrightarrow \|\eta(\mathrm{t})\| \to \frac{\epsilon}{\lambda_{min}(K) - \tilde{L}\rho}$$ Exponentially fast

Robust Landing Control



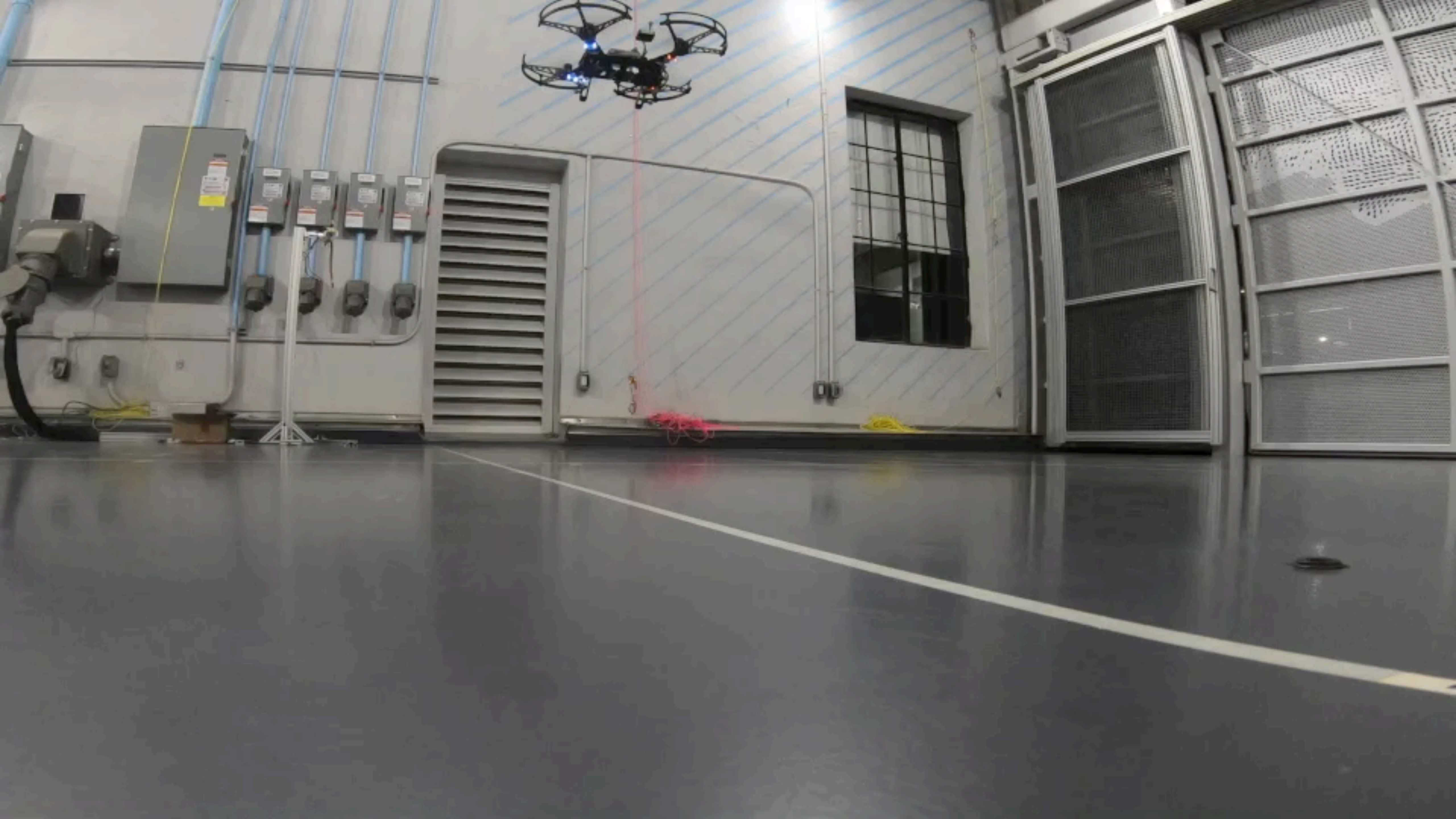
PD

PID



Neural-Lander (PD+Fa)

https://www.youtube.com/watch?v=C_K8MkC_SSQ



Aside: Learning Control Lyapunov Functions

- CLFs encode low-dimensional projection of dynamics
  - DOF of action space rather than state space
  - Can be easier to learn than full dimensional dynamics

- How to learn CLF for controller design?
- How to analyze stability under model uncertainty?

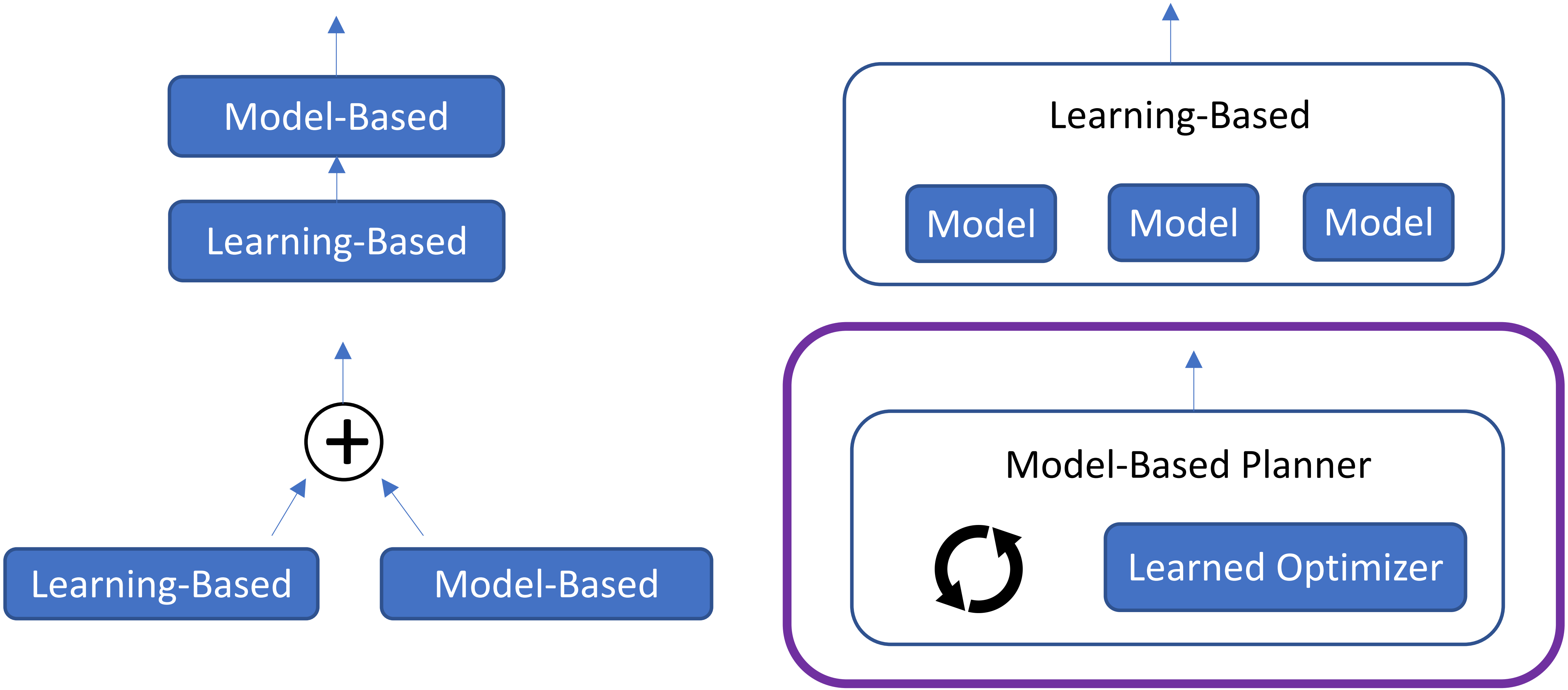
Andrew Taylor

Victor Dorobantu

Episodic Learning with Control Lyapunov Functions for Uncertain Robotic Systems
Andrew J. Taylor, Victor D. Dorobantu, Hoang M. Le, Yisong Yue, Aaron D. Ames. IROS 2019.

A Control Lyapunov Perspective on Episodic Learning via Projection to State Stability
Andrew J. Taylor, Victor D. Dorobantu, Meera Krishnamoorthy, Hoang M. Le, Yisong Yue, Aaron D. Ames. CDC 2019.

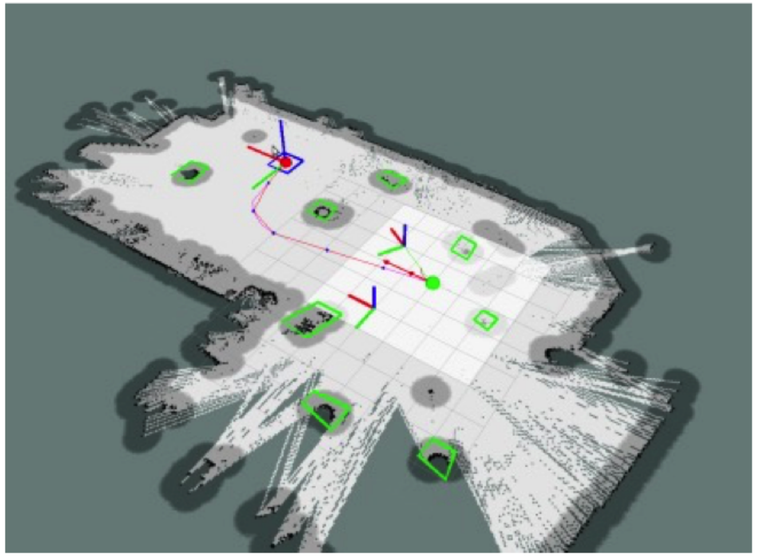
Blending Models/Rules & Black-Box Learning



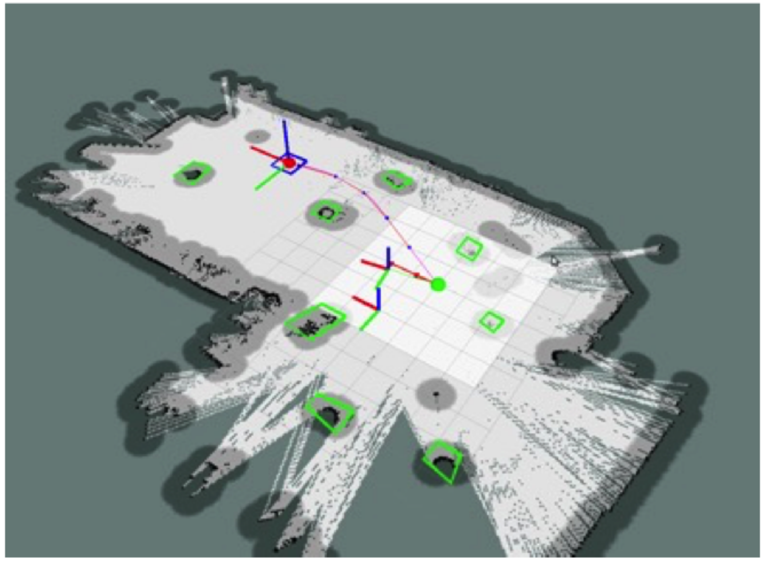
Ravi Lanka

Motivating Example:
Risk-Aware Planning

Jialin Song



Low Risk



High Risk

- Compiled as mixed integer program
- Challenging optimization problem

Model-Based Planning

- Environment Model is Given
- Design global plan (aka trajectory)
- Satisfy global constraints
  - Previous topics only ensured local constraints
  - E.g., Lyapunov stability, smoothness
- **NP-Hard optimization problem!**

Optimization as Sequential Decision Making

- 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 Problem
  - Builds upon modern RL/IL
- Theoretical Analysis/Guidance
- Interesting Algorithms

Example #1: Learning to Search (Discrete)

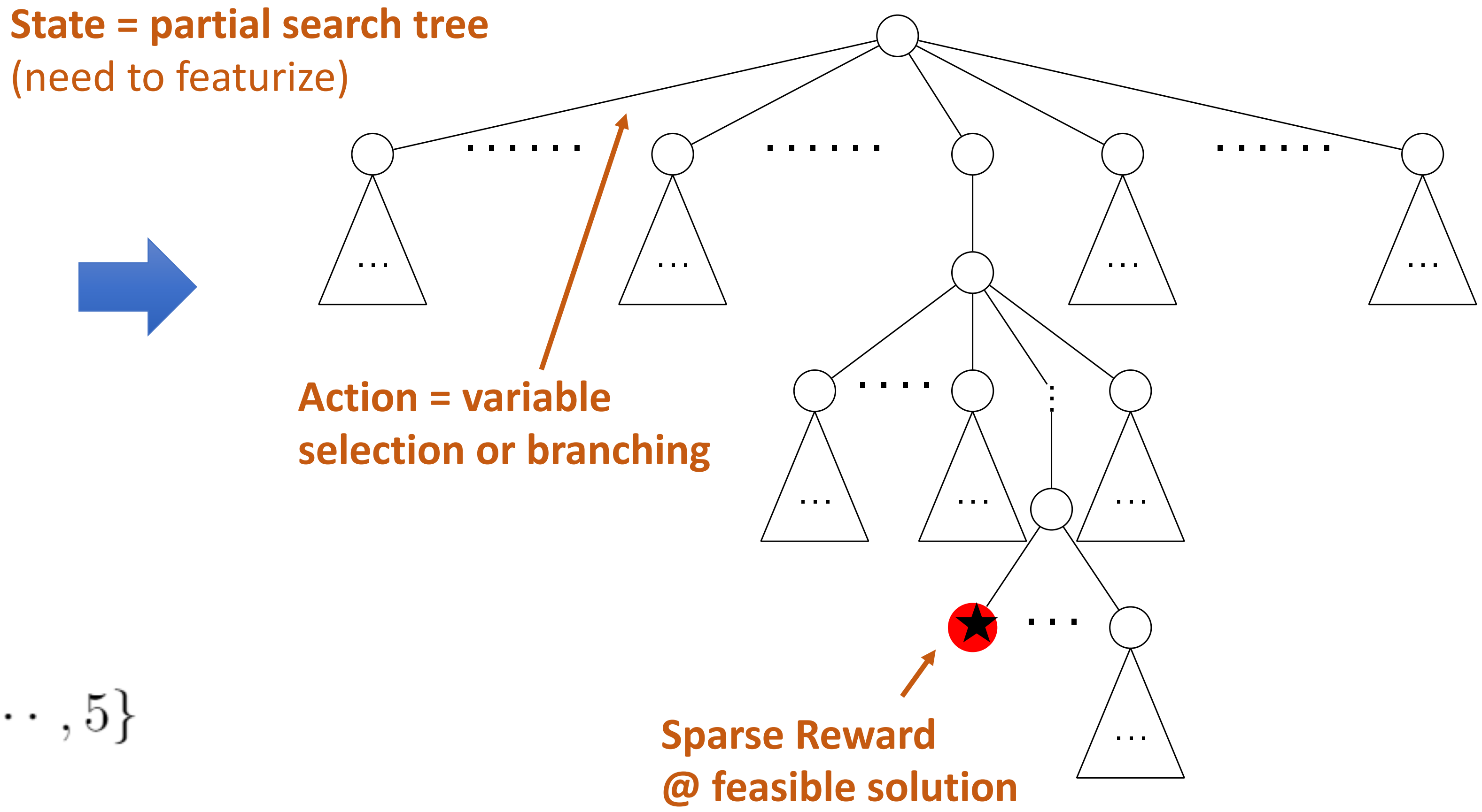
Integer Program

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

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 Bound)



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

Example #2: Learning Greedy Algorithms (discrete)

Contextual Submodular Maximization:

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

Submodular Utility

Context / Environment

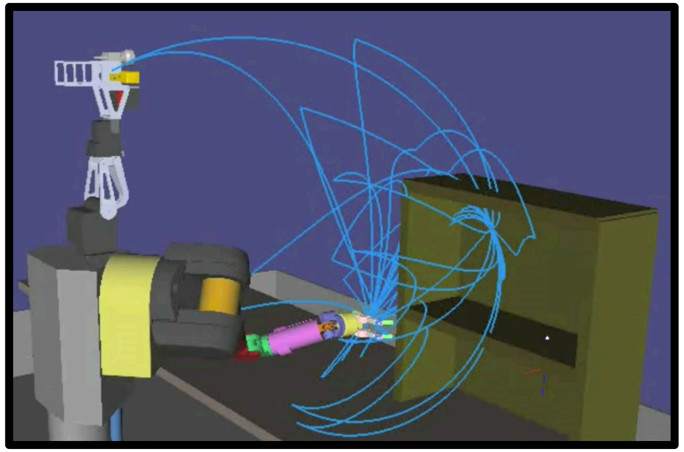
Selected Elements

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

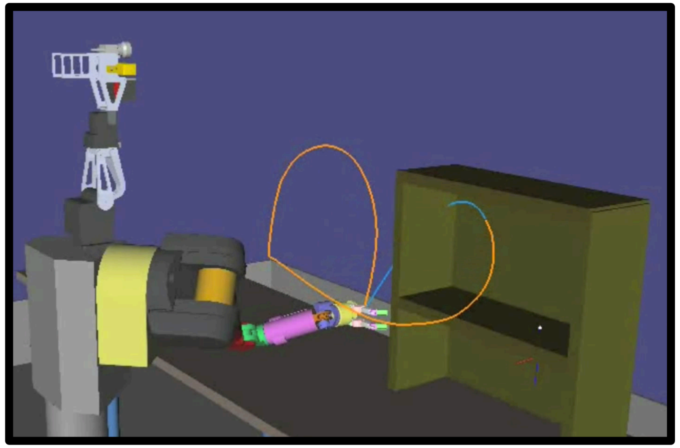
Not Available at Test Time

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

State s = (Ψ, x)



Dictionary of Trajectories



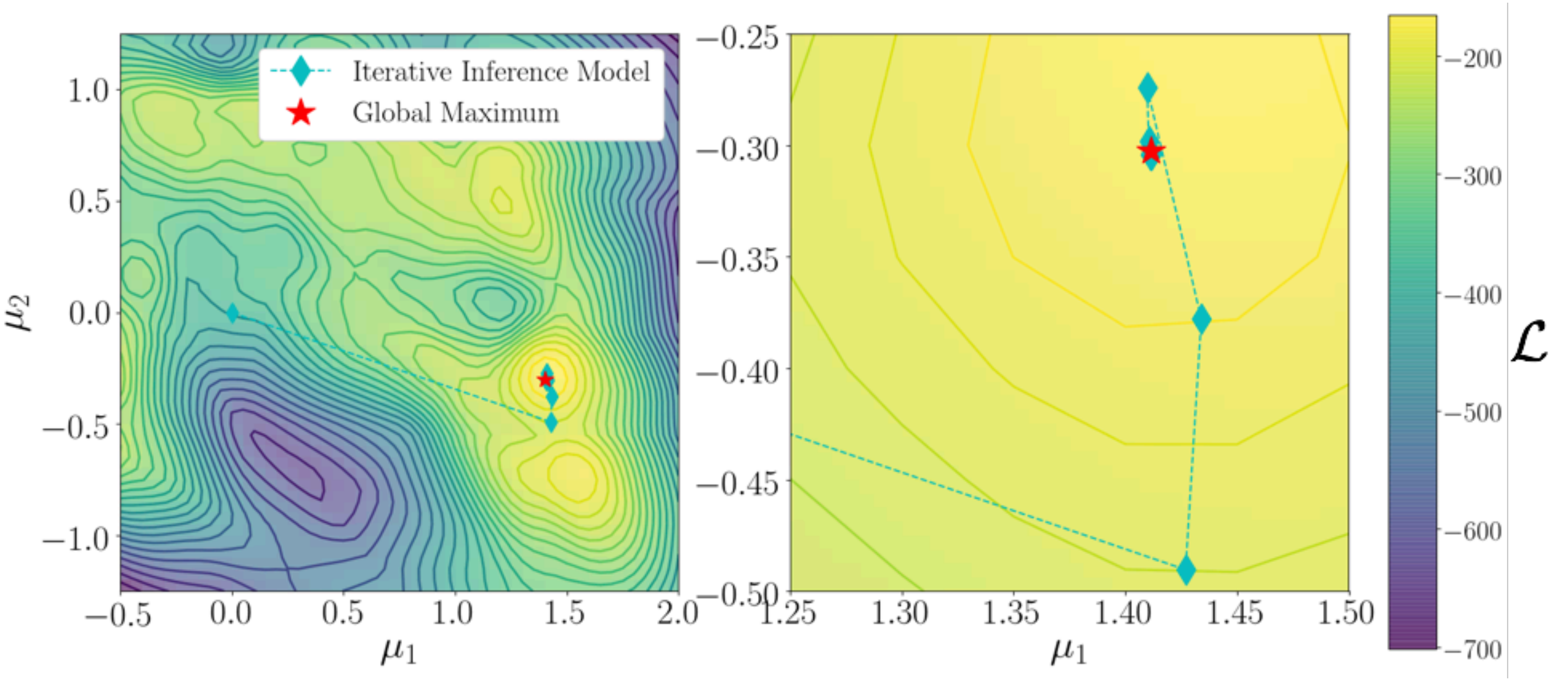
Select Diverse Set

Learning Policies for Contextual Submodular Prediction S. Ross, R. Zhou, Y. Yue, D. Dey, J.A. Bagnell. ICML 2013

Example #3: Iterative Amortized Inference (continuous)

Gradient Descent Style Updates:

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



Useful for Accelerating Variational Inference

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

Optimization as Sequential Decision Making

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

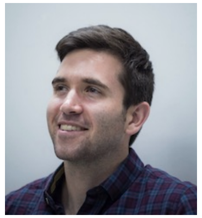
- Discrete Optimization (Greedy), Dense Rewards
- **Learning Policies for Contextual Submodular Prediction** [ICML 2013]

Learning to Infer

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


Jialin Song


Stephane Ross


Joe Marino

Optimization as Sequential Decision Making

Learning to Search

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

Jialin Song

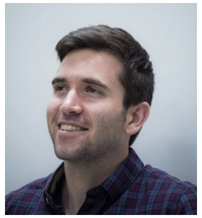
Contextual Submodular Maximization

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

Stephane Ross

Learning to Infer

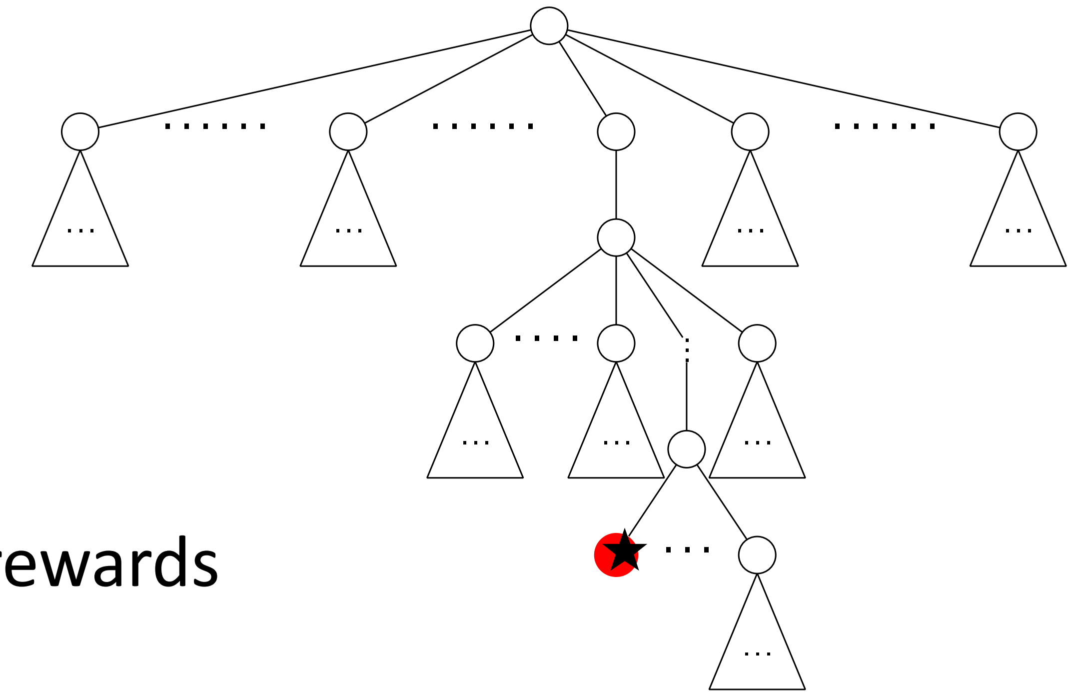
- Continuous Optimization (Gradient-style), Dense Rewards
- **Iterative Amortized Inference** [ICML 2018]
- **A General Method for Amortizing Variational Filtering** [NeurIPS 2018]



Joe Marino

Learning to Optimize for Tree Search

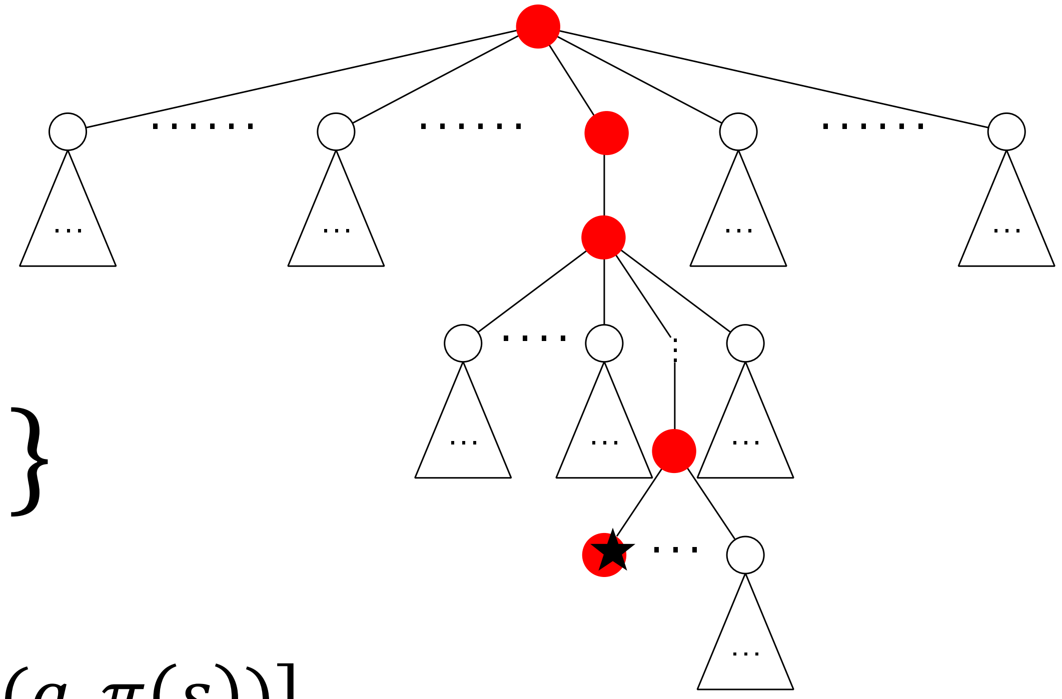
- Idea #1: Treat as Standard RL
- Randomly explore for high rewards
  - **Very hard exploration problem!**
- Issues: massive state space & sparse rewards



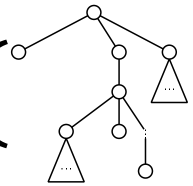
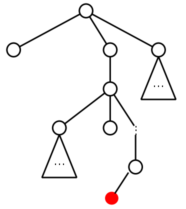
Learning to Optimize for Tree Search

- Idea #2: Treat as Standard IL

- Convert to Supervised Learning
  - Assume access to solved instances



"Demonstration Data"

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

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

Retrospective Imitation

Jialin Song

Ravi Lanka

- 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,…,K

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

Connections to Curriculum Learning & Transfer Learning

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

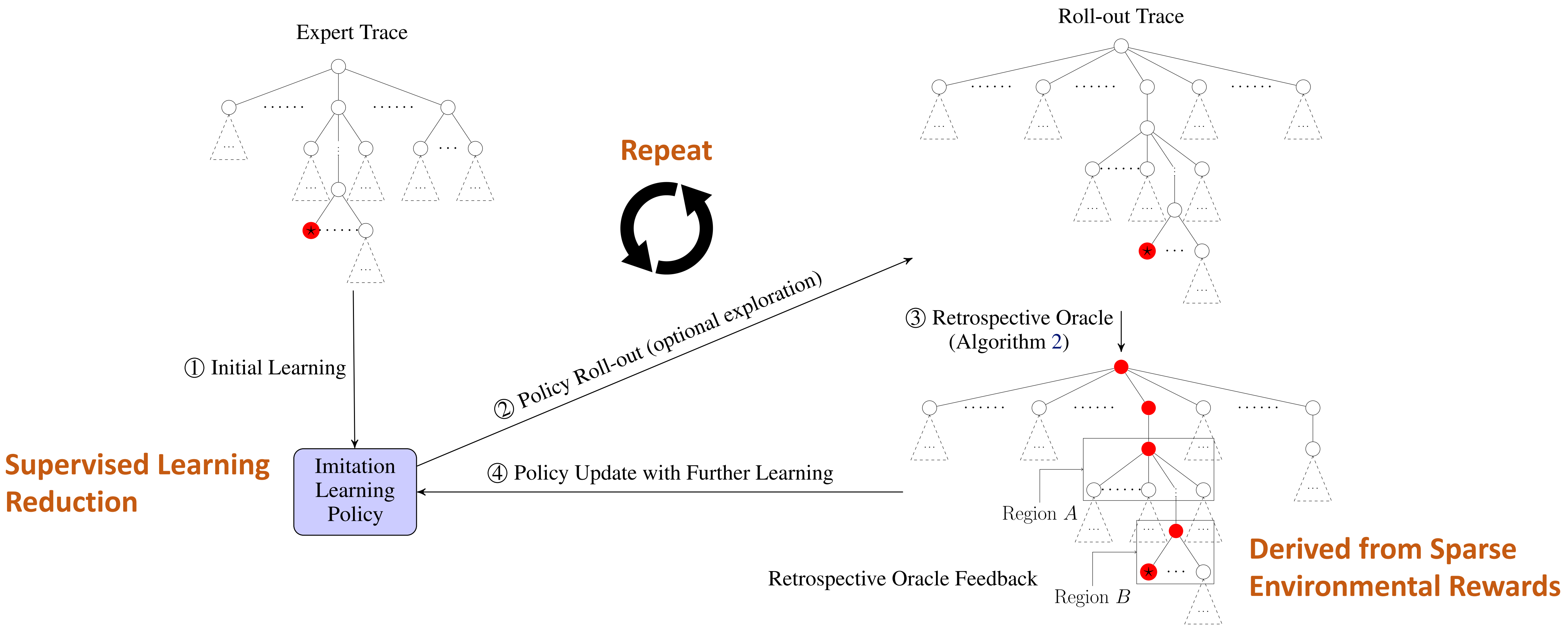
Retrospective Imitation

- 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 Environmental Rewards

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)

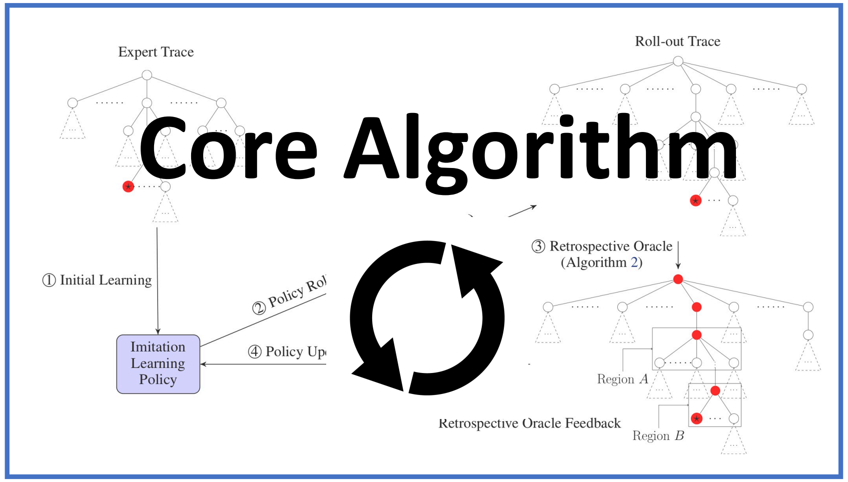
Initialize k=1

Initialize Gurobi/SCIP/CPlex

Problem Difficulty k

Base Solver

Instances & Demonstrations

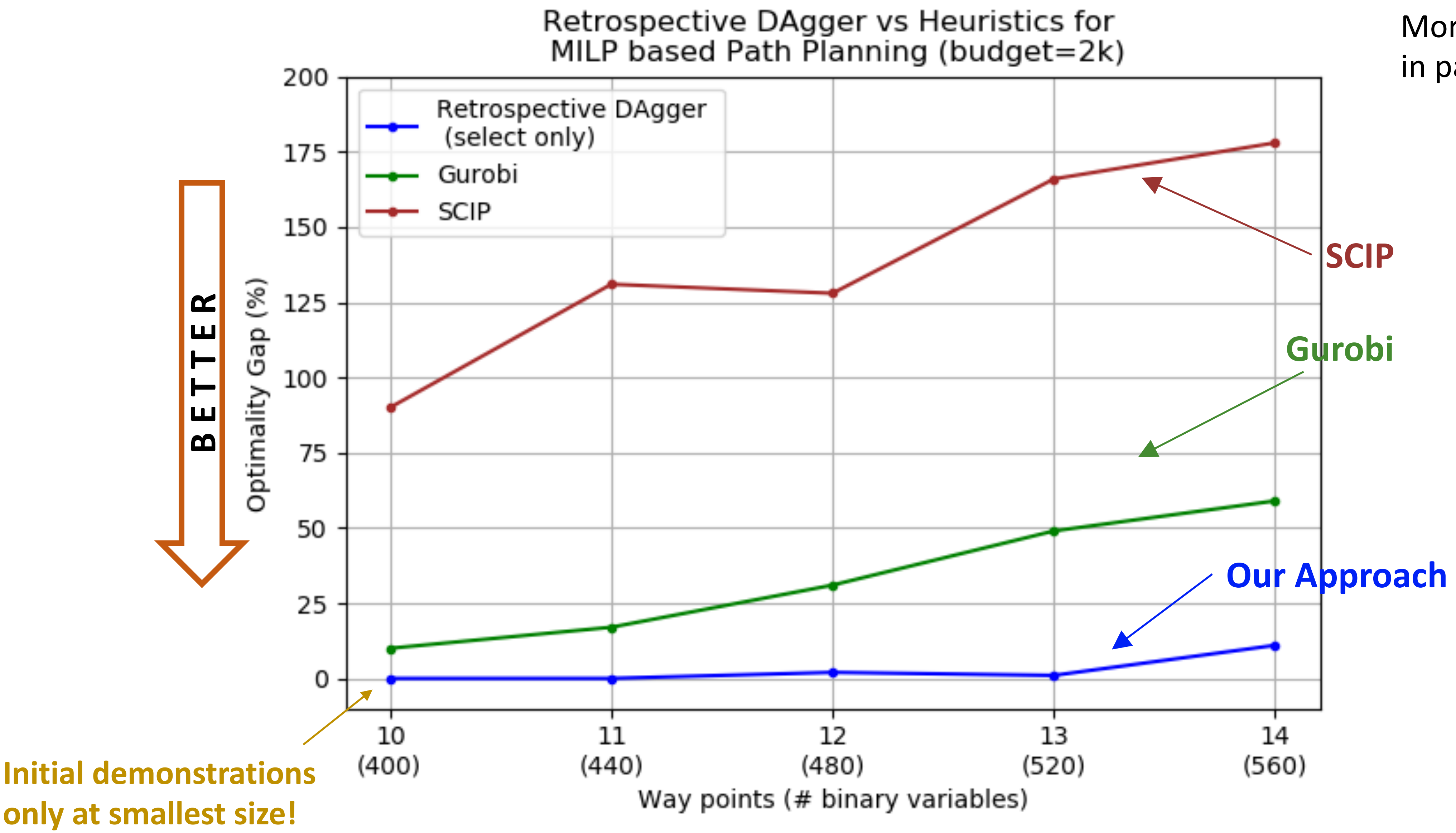


k=k+1

Use trained π

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

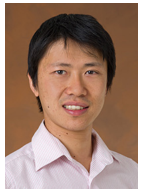
More experiments in paper



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

Ongoing: Integration with ENav

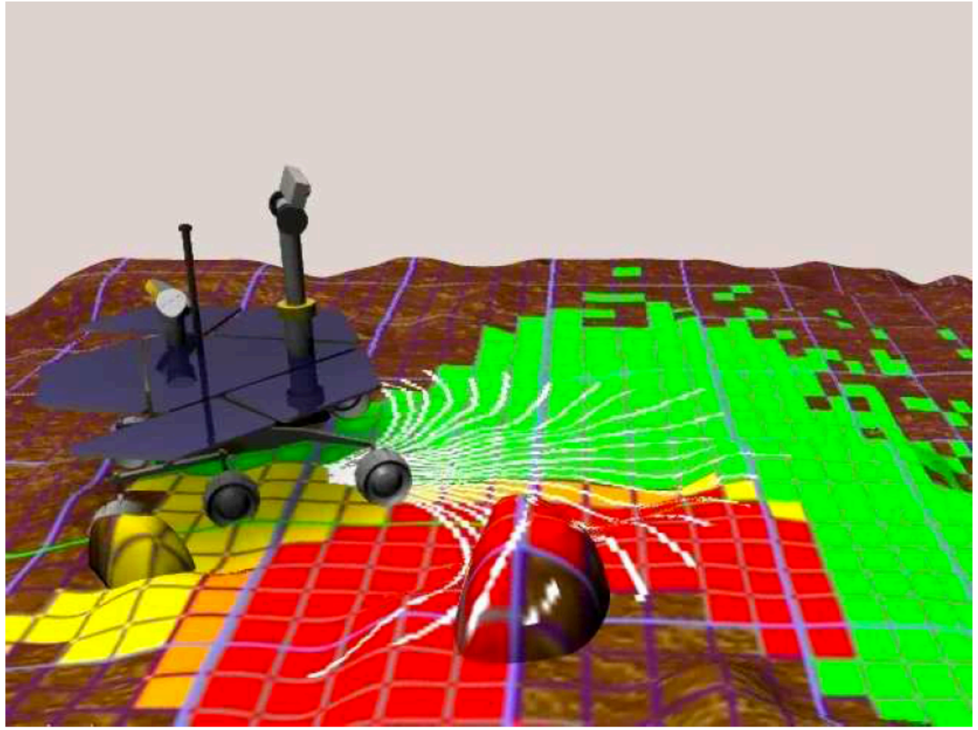
Ravi Lanka

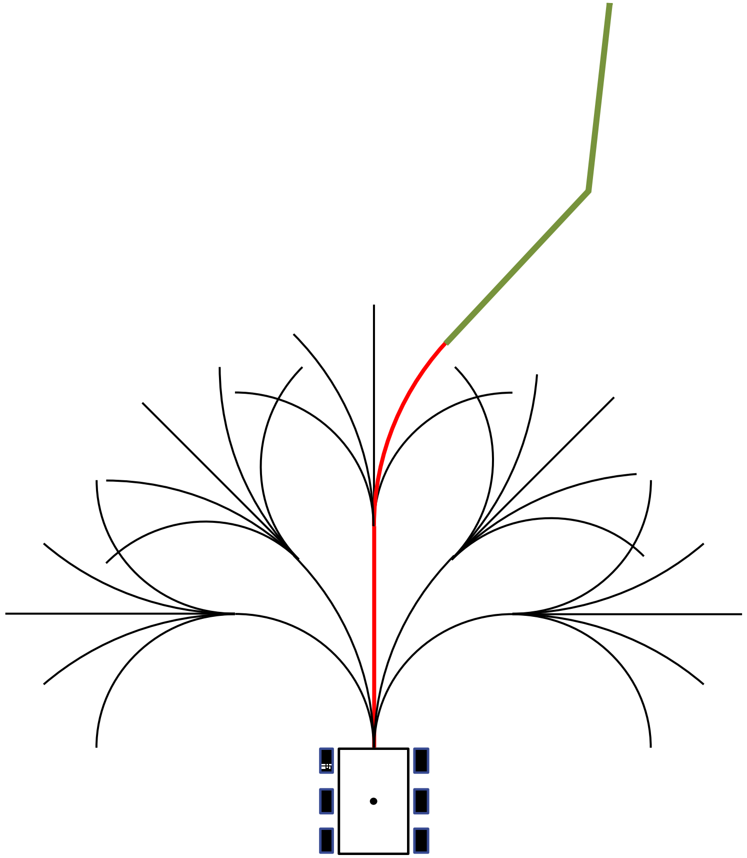


Hiro Ono

Olivier Toupet

Neil Abcouwer





Ongoing: Additive Manufacturing

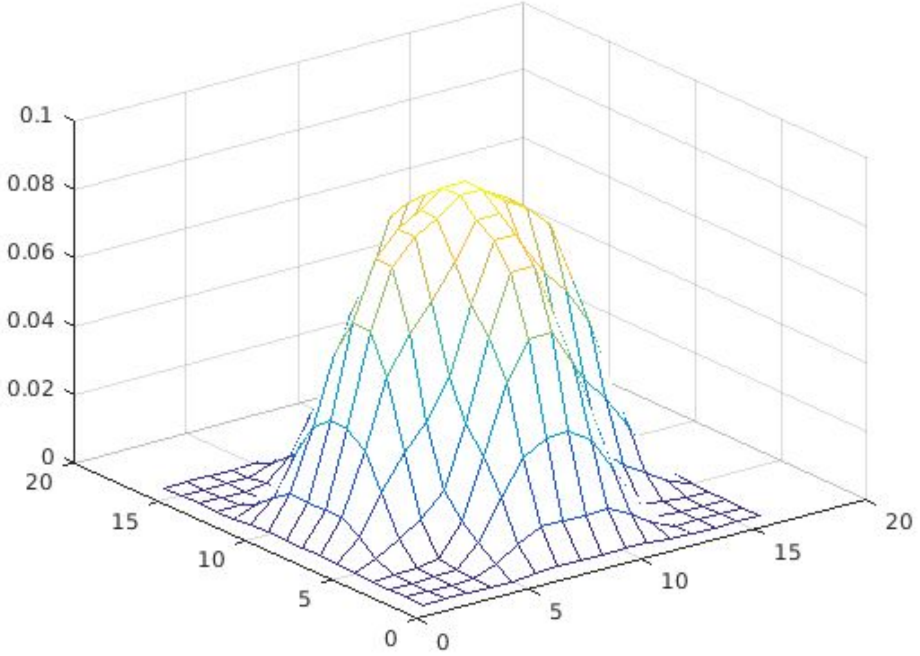

Stephanie Ding

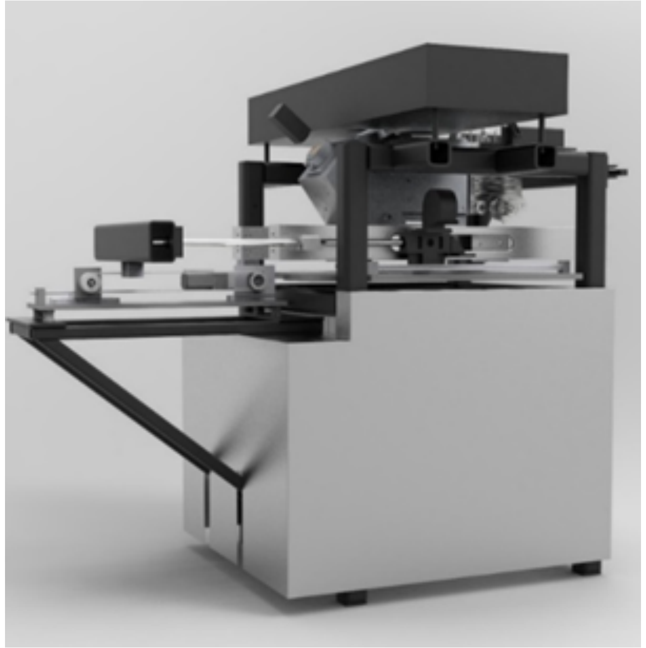

Jialin Song


Uduak Inyang-Udoh


Sandipan Mishra

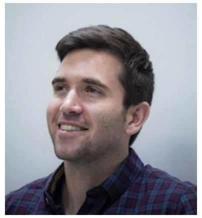
- Planning for 3D Inkjet Droplet Printing



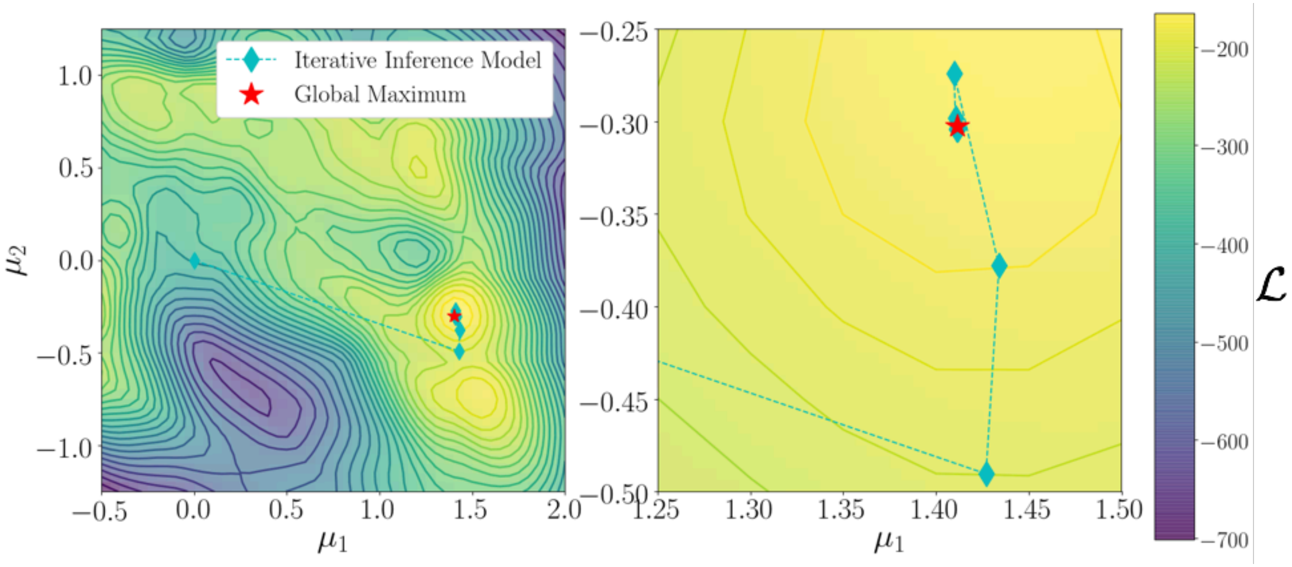


Rensselaer TEXAS The University of Texas at Austin

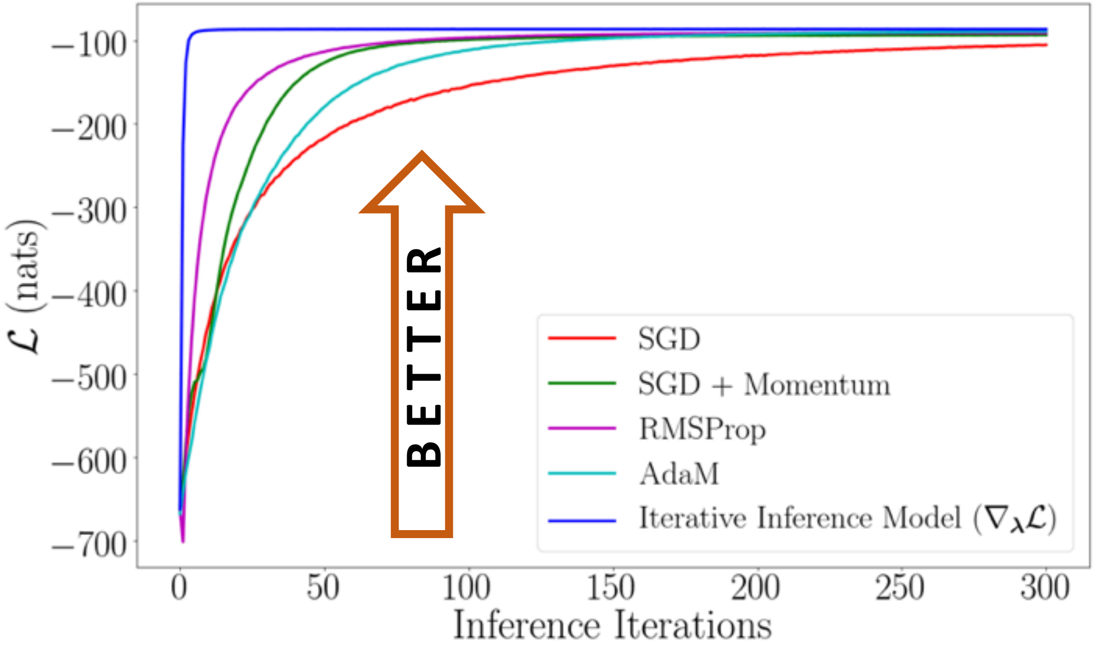
Iterative Amortized Inference
(for Deep Probabilistic Models)



Joe Marino



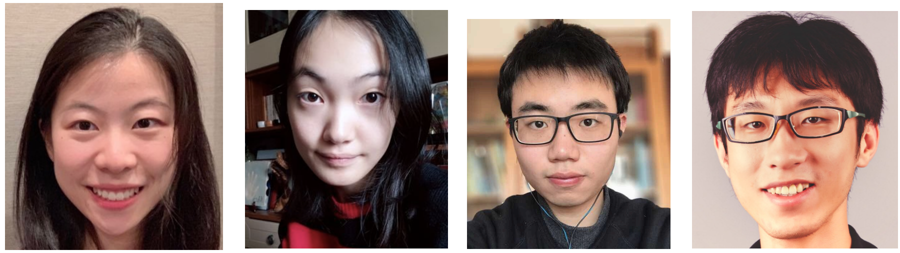
Related to “Learning to Learn” [Andychowicz et al., 2016]



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

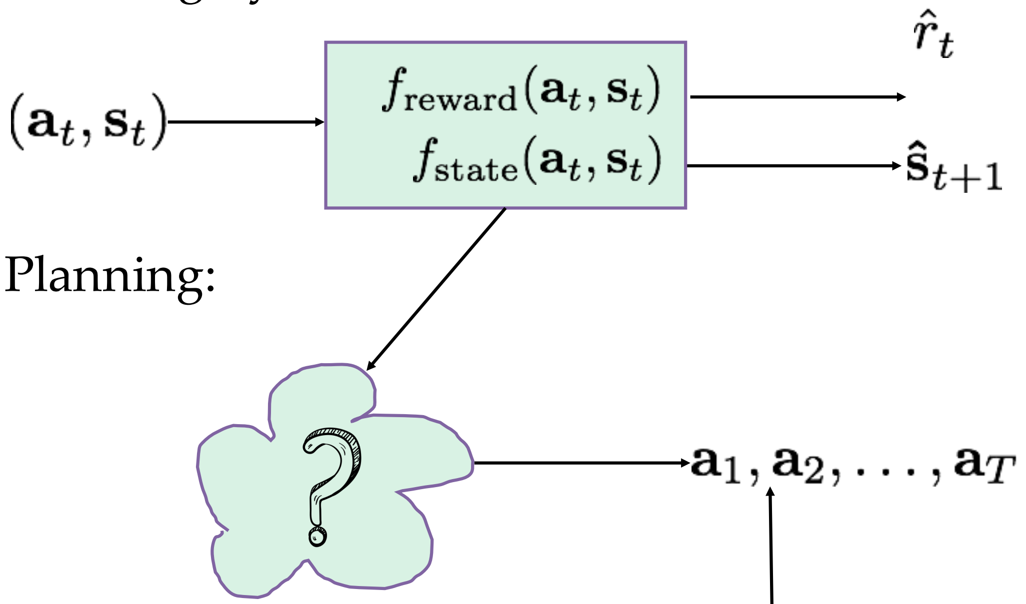
A General Framework for Amortizing Variational Filtering, Joe Marino et al, NeurIPS 2018

Ongoing: Amortized Planning



Yujia Huang | Sophie Dai | Hao Liu | Tongxin Li

Learning dynamics:

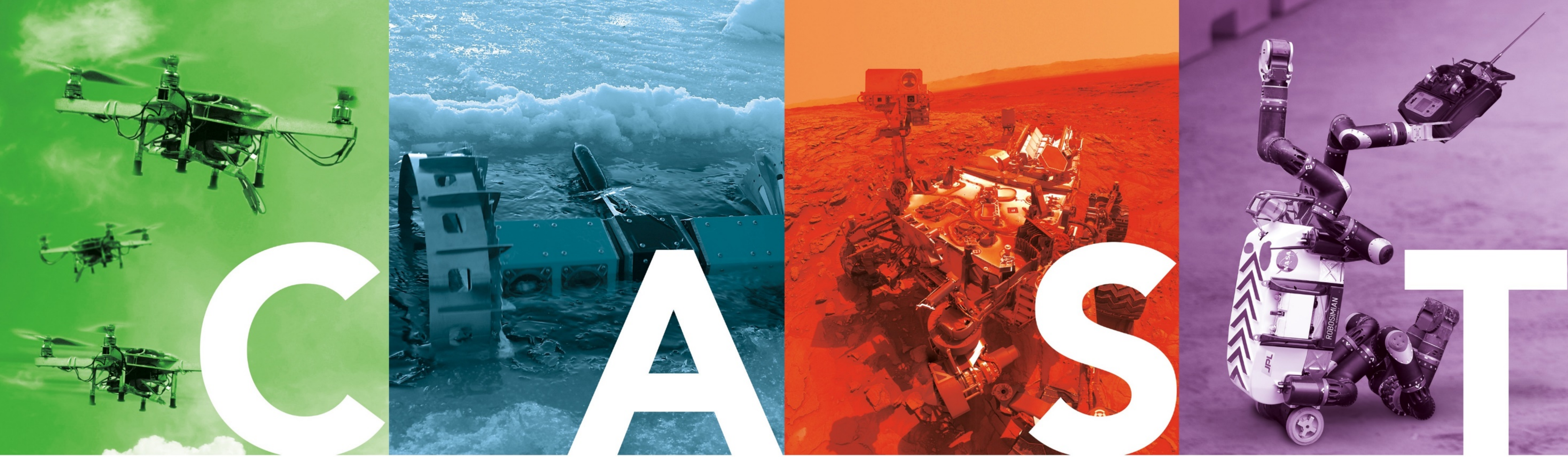


Optimize:

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

Baseline: Gradient-based Planning

Can use (offline) training to amortize?



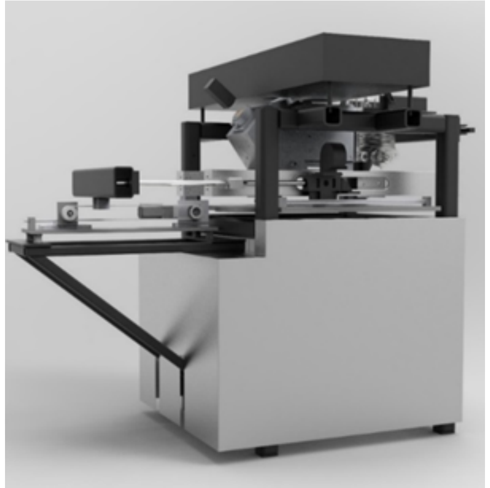
Center for Autonomous Systems and Technologies

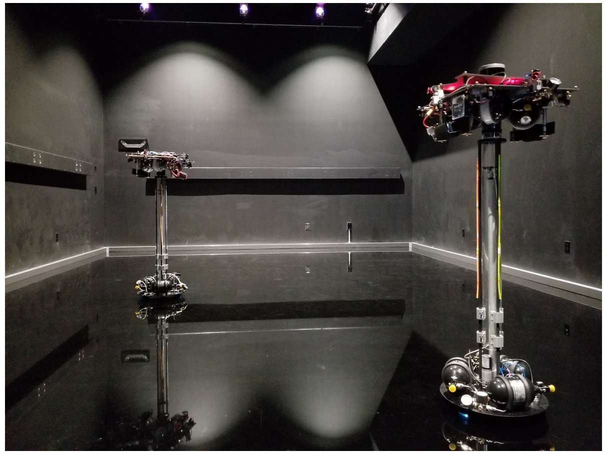
A New Vision for Autonomy

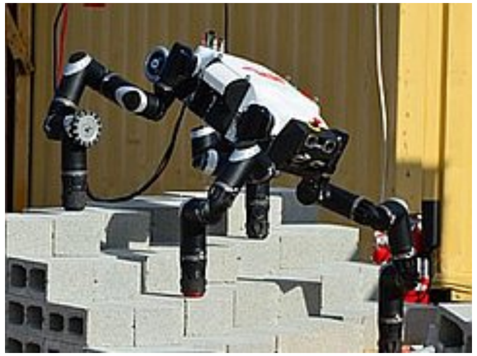
Caltech

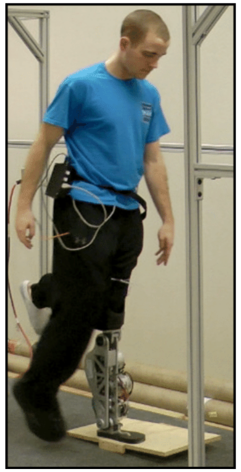
http://cast.caltech.edu

Autonomous Dynamic Robots







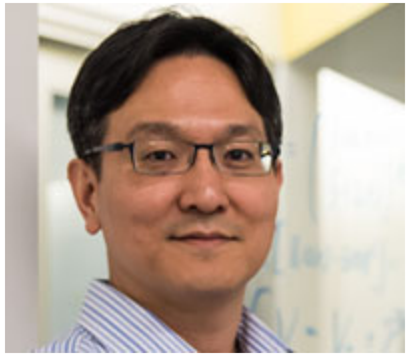


http://cast.caltech.edu

Postdoc Openings!

(applications due January)

Mory Gharib



Soon-Jo Chung

Aaron Ames

Anima Anandkumar

Yisong Yue

Joel Burdick

Katie Bouman

Pietro Perona

Takeaways

- Control methods => analytic guarantees

  (side guarantees)

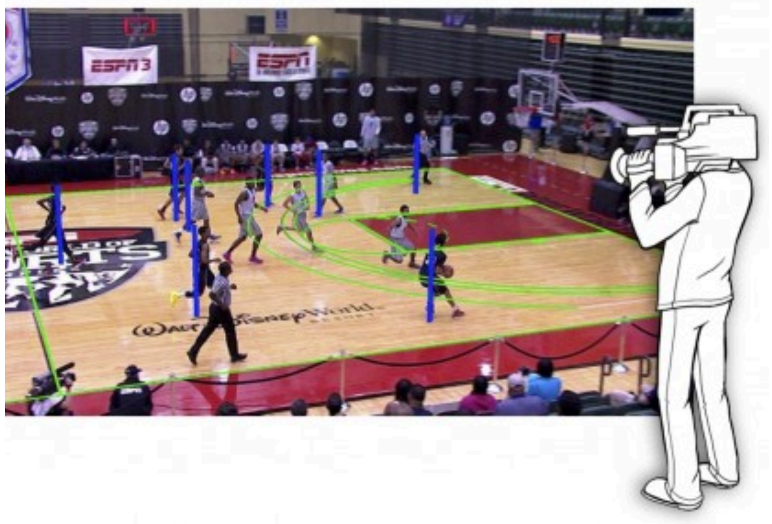
- Blend w/ learning => improve precision/flexibility

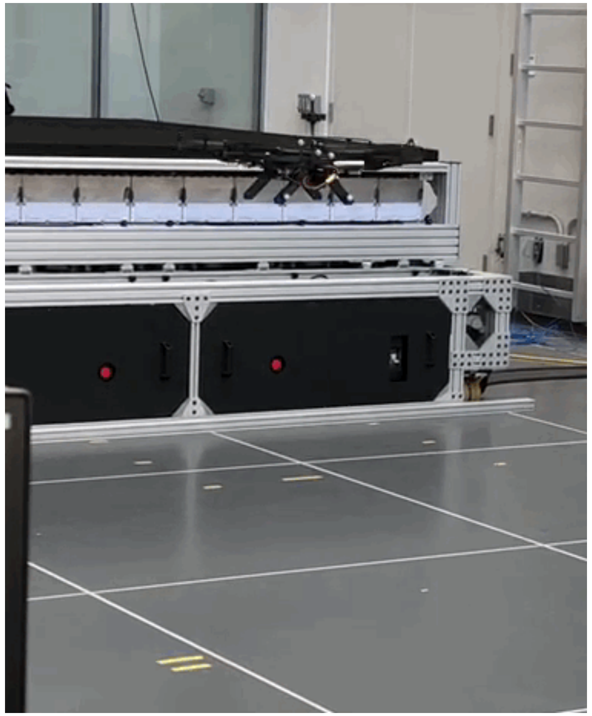
- Preserve side guarantees (possibly relaxed)

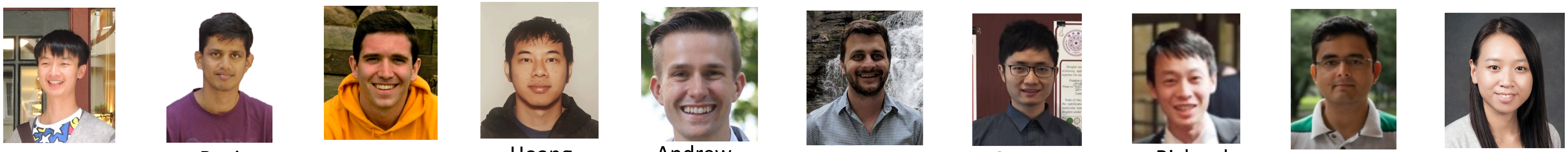
- Sometimes interpret as functional regularization

  (speeds up learning)

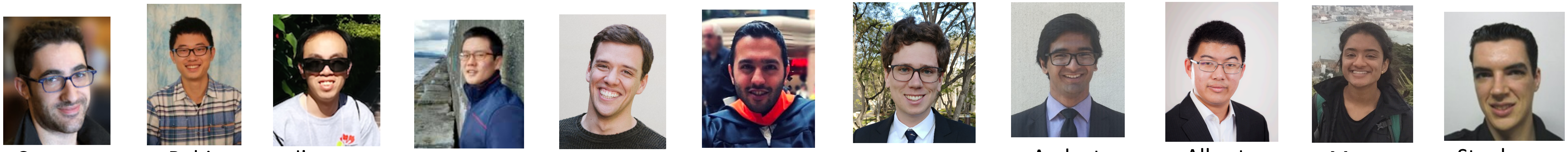
- Also: combinatorial planning as policy learning



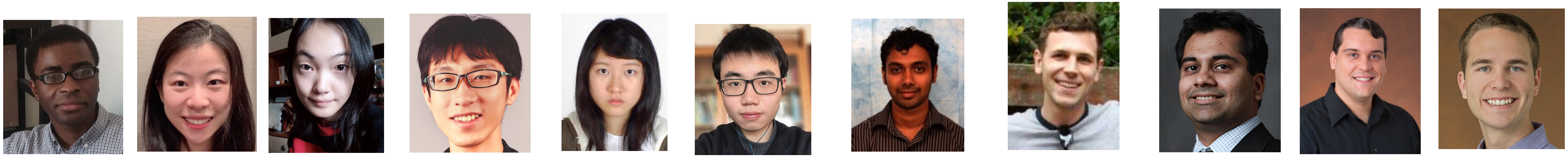




Jialin Song | Ravi Lanka | Joe Marino | Hoang Le | Andrew Taylor | Victor Dorobantu | Guanya Shi | Richard Cheng | Abhinav Verma | Angie Liu



Cameron Voloshin | Robin Zhou | Jimmy Chen | Andrew Kang | Milan Cvitkovic | Kamyar Azizzadenesheli | Michael O'Connell | Aadyot Bhatnagar | Albert Zhao | Meera Krishnamoorthy | Stephane Ross



Uduak Inyang-Udoh | Yujia Huang | Sophie Dai | Tongxin Li | Stephanie Ding | Hao Liu | Debadeepta Dey | Peter Carr | Sandipan Mishra | Olivier Toupet | Neil Abcouwer

Anima Anandkumar | Soon-Jo Chung | Aaron Ames | Joel Burdick | Gabor Orosz | Swarat Chaudhuri | Stephan Mandt | Drew Bagnell | Ufuk Topcu | Hiro Ono | Jim Little

References

Smooth Imitation Learning for Online Sequence Prediction, Hoang Le, et al., ICML 2016

Control Regularization for` Reduced Variance Reinforcement Learning, Richard Cheng et al. ICML 2019

Batch Policy Learning under Constraints, Hoang Le, et al. ICML 2019

Learning Smooth Online Predictors for Real-Time Camera Planning using Recurrent Decision Trees, Jianhui Chen, et al., CVPR 2016

Imitation-Projected Policy Gradient for Programmatic Reinforcement Learning, Abhinav Verma, Hoang Le, et al., NeurIPS 2019

Neural Lander: Stable Drone Landing Control using Learned Dynamics, Guanya Shi, et al., ICRA 2019

Robust Regression for Safe Exploration in Control, Angie Liu, Guanya Shi, et al., arxiv

Episodic Learning with Control Lyapunov Functions for Uncertain Robotic Systems, Andrew Taylor, Victor Dorobantu, et al., IROS 2019

A Control Lyapunov Perspective on Episodic Learning via Projection to State Stability, Andrew Taylor, Victor Dorobantu, et al., CDC 2019

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 2013

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

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

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

https://github.com/rcheng805/CORE-RL

https://sites.google.com/view/constrained-batch-policy-learn/

https://github.com/vdorobantu/lyapy

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

https://github.com/joelouismarino/iterative_inference

https://github.com/joelouismarino/amortized-variational-filtering