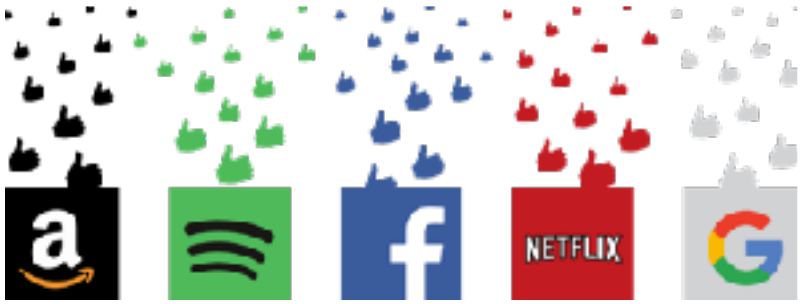
Structured Policy Learning: Towards Real-World Sequential Decision Making

Hoang M. Le

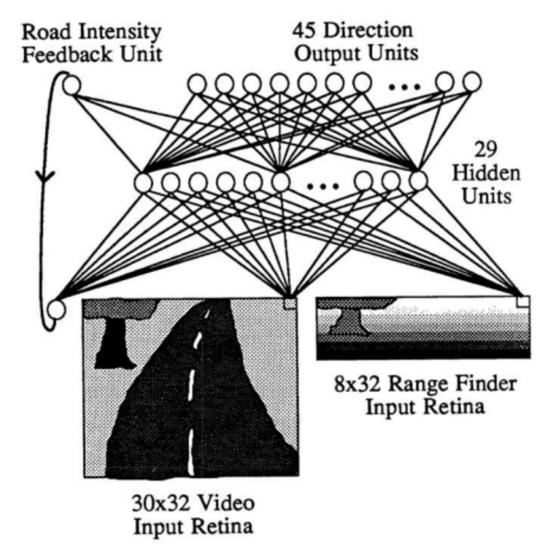
California Institute of Technology

Thesis Committee: Anima Anandkumar (Caltech & NVIDIA)
Hal Daumé III (Microsoft & UMD)
Adam Wierman (Chair, Caltech)
Yisong Yue (PhD advisor, Caltech)

Sequential decision making systems

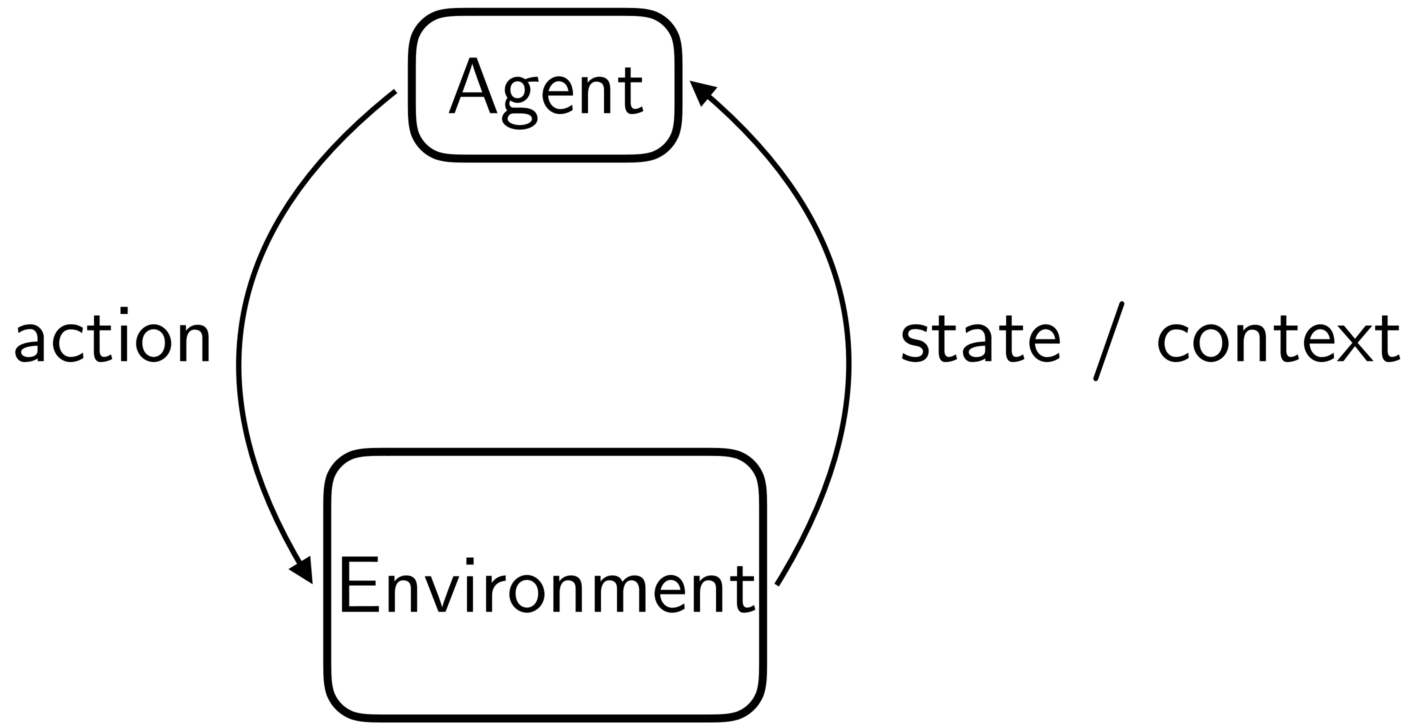


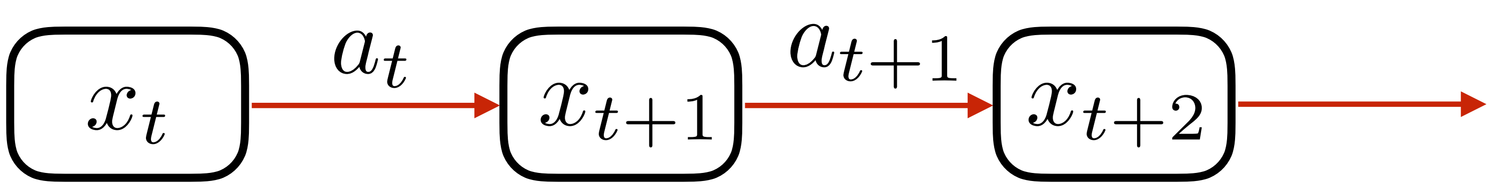
Machine learning for decision making



(ALVINN - Dean Pomerleau et al., 1989-1999)

Policy Learning





Policy $\pi : \mathrm{X} \mapsto \mathrm{A}$

Value function: Optimization objective to derive “optimal” policy

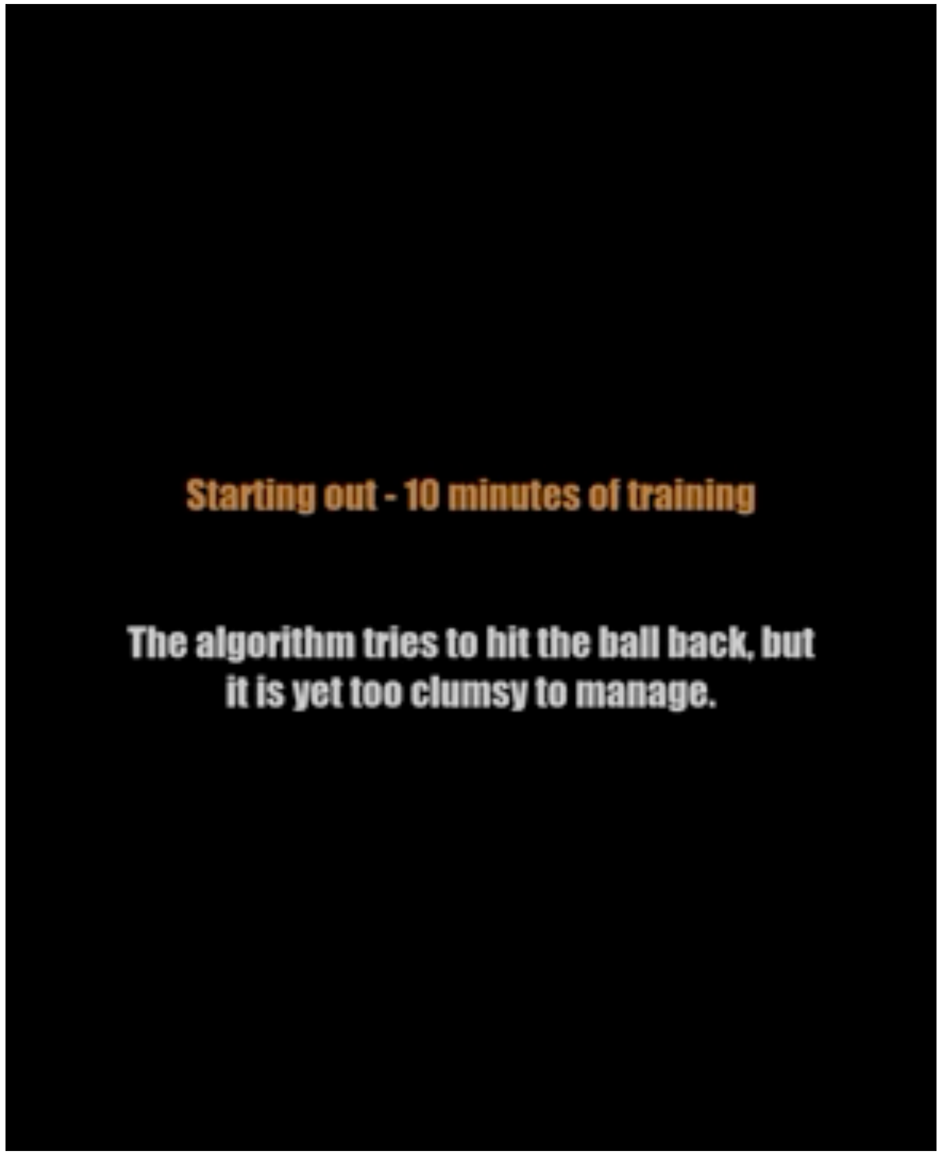
Model: Unknown Dynamics

Reinforcement learning (RL)

Exploration-based methods to minimize long term cost

Reinforcement learning (RL)

Exploration-based methods to minimize long term cost



Policy: $x =$ screen $\mapsto a =$ move

Value: total single-stage cost $C(\pi) = \mathbb{E}\left[\sum c(x,a)\right]$

Model: game engine (unknown)

Reinforcement learning

Success stories:

OpenAI Fiv

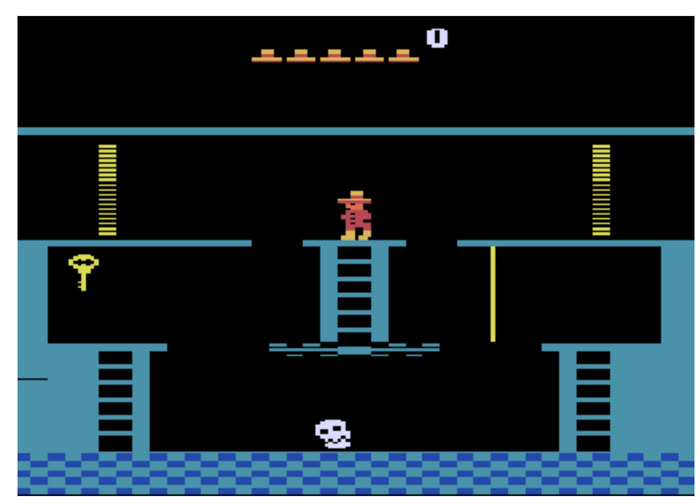
Silver et al., Scienc

Levine et al., IJRR 2017

Cautionary tales:

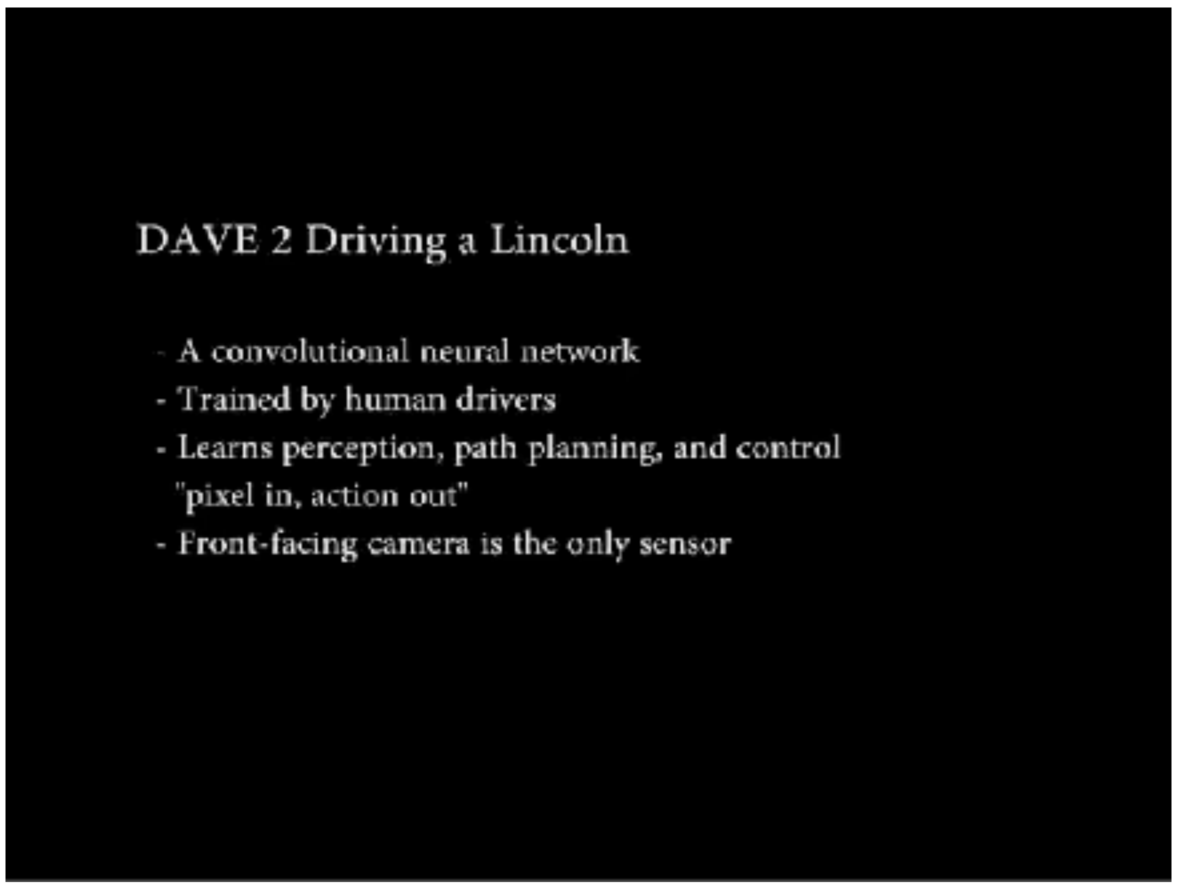
Imperfect cost and observations

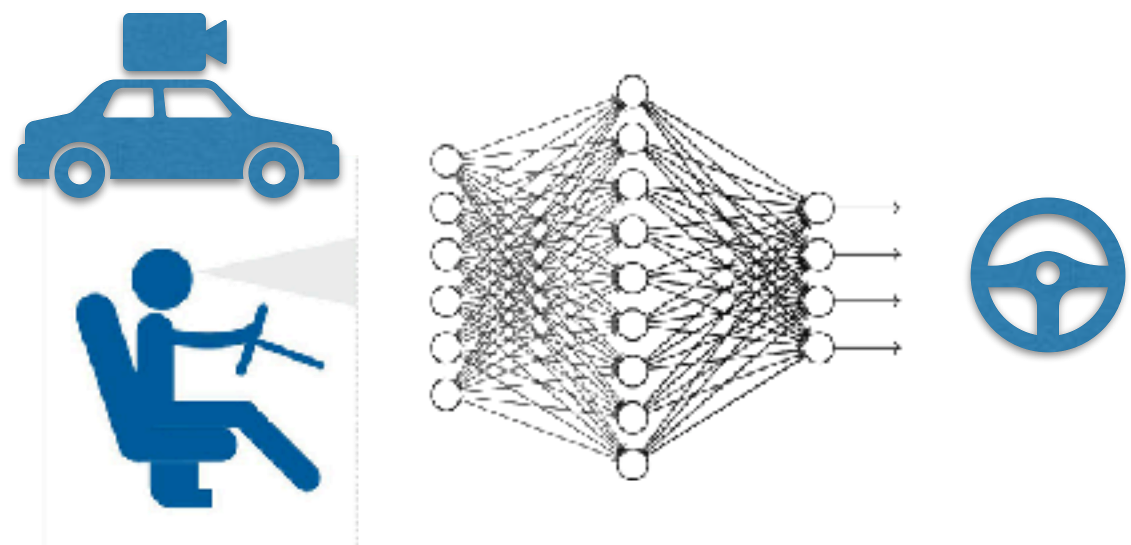
Inefficient exploration
brittle performance



Imitation learning (IL)

Expert-based methods to minimize long-term imitation loss
(Behavioral cloning, interactive imitation learning, inverse RL…)





Policy: x = camera images $\mapsto$ a = steering angle

Value: imitation loss w.r.t. expert $C(\pi) = \mathbb{E}\left[||\pi(x) - \pi^*(x)||\right]$

Model: traffic environment (unknown)

Imitation learning tutorial - ICML 2018

Yisong Yue

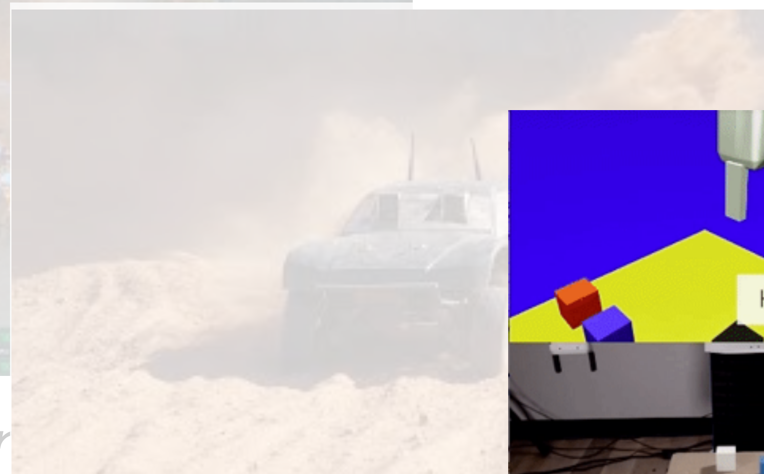
Hoang M. Le

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

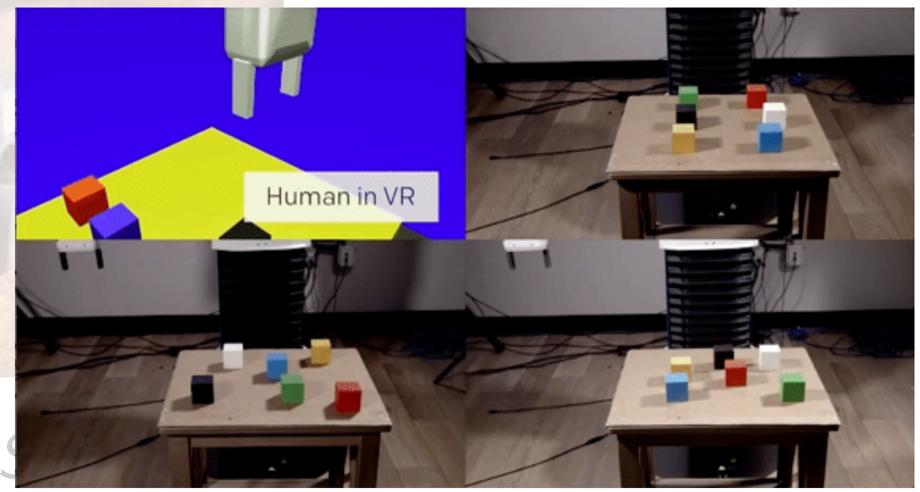
Imitation learning

Success stories:

AlphaStar



Pan et al., RSS

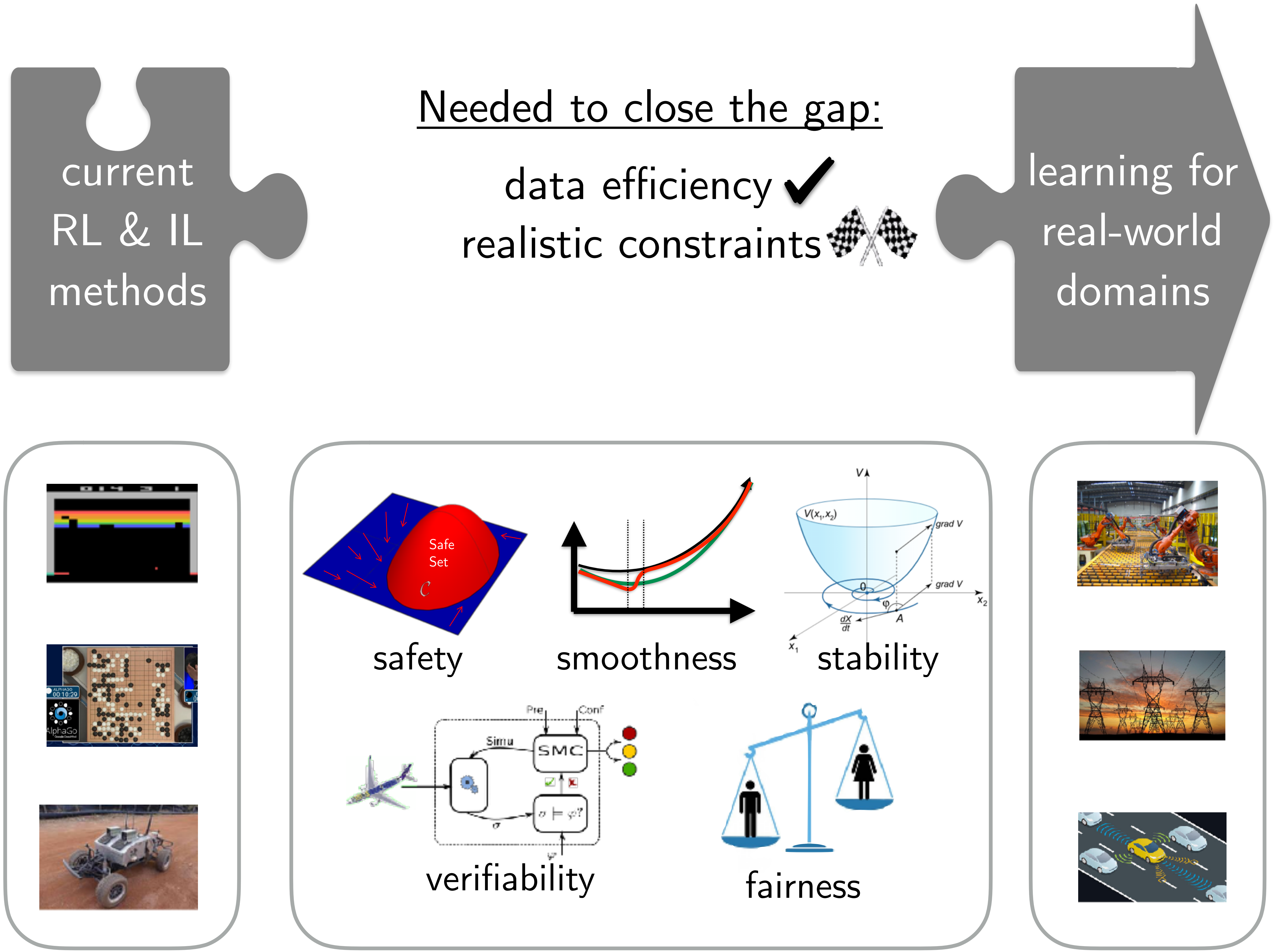


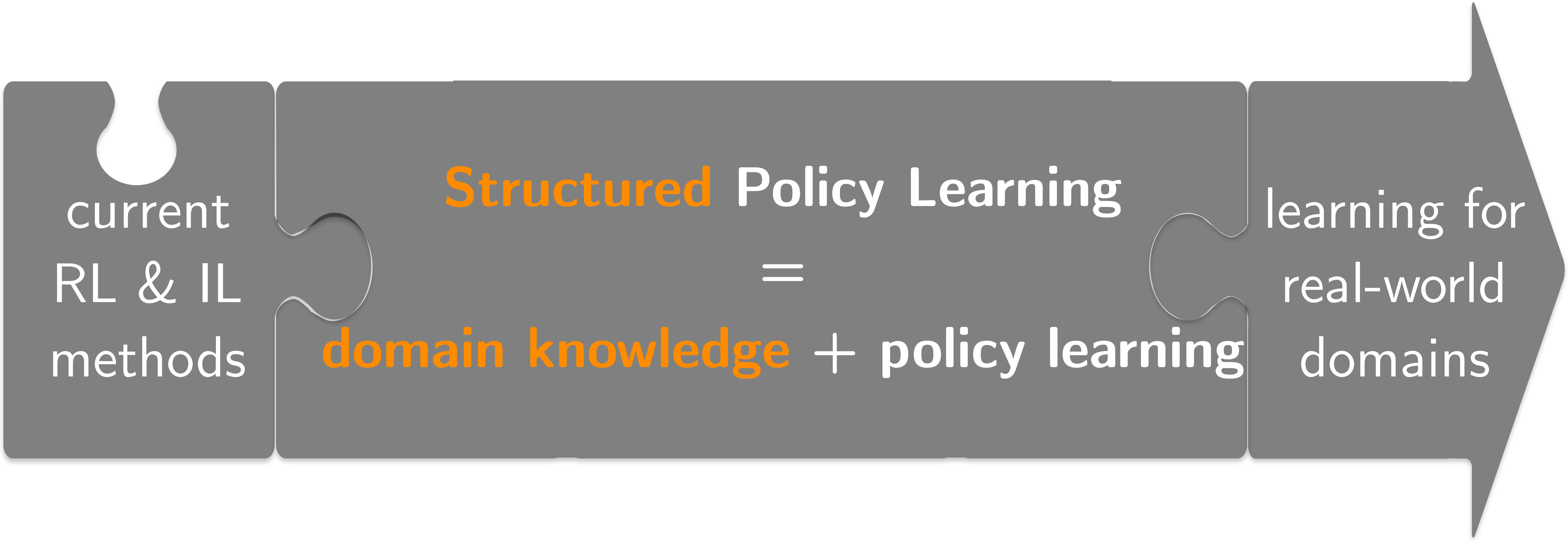
Duan et al., NeurIPS 2017

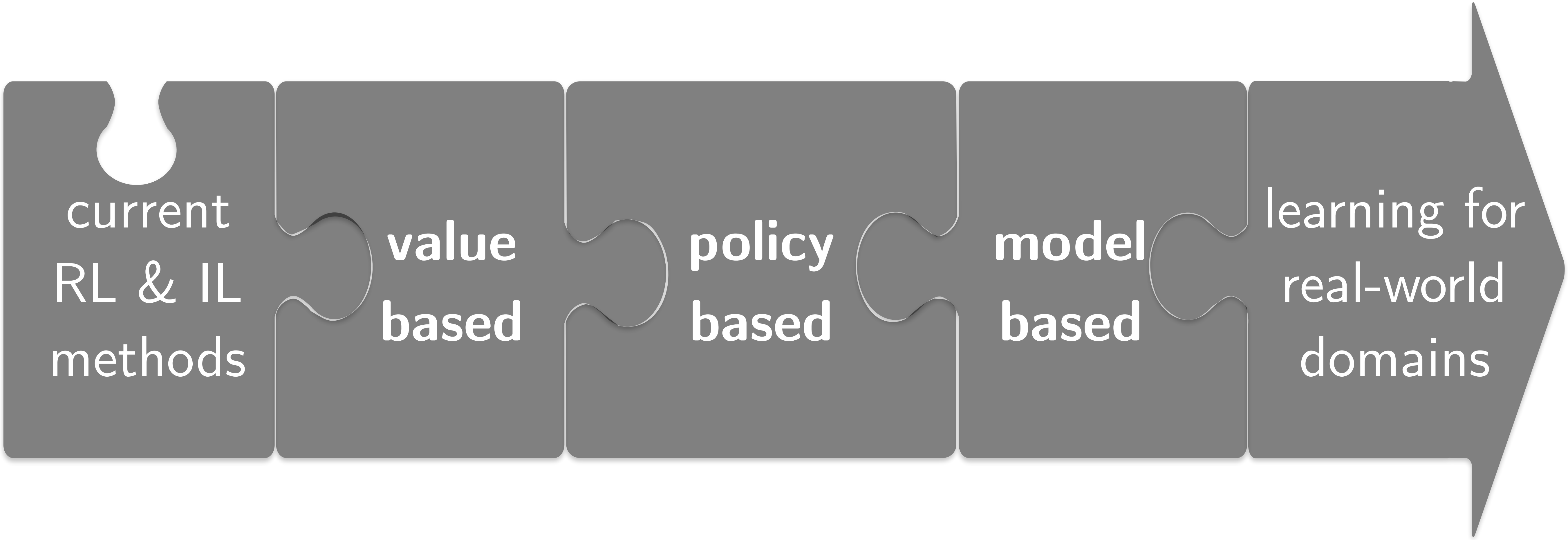
Cautionary tales:

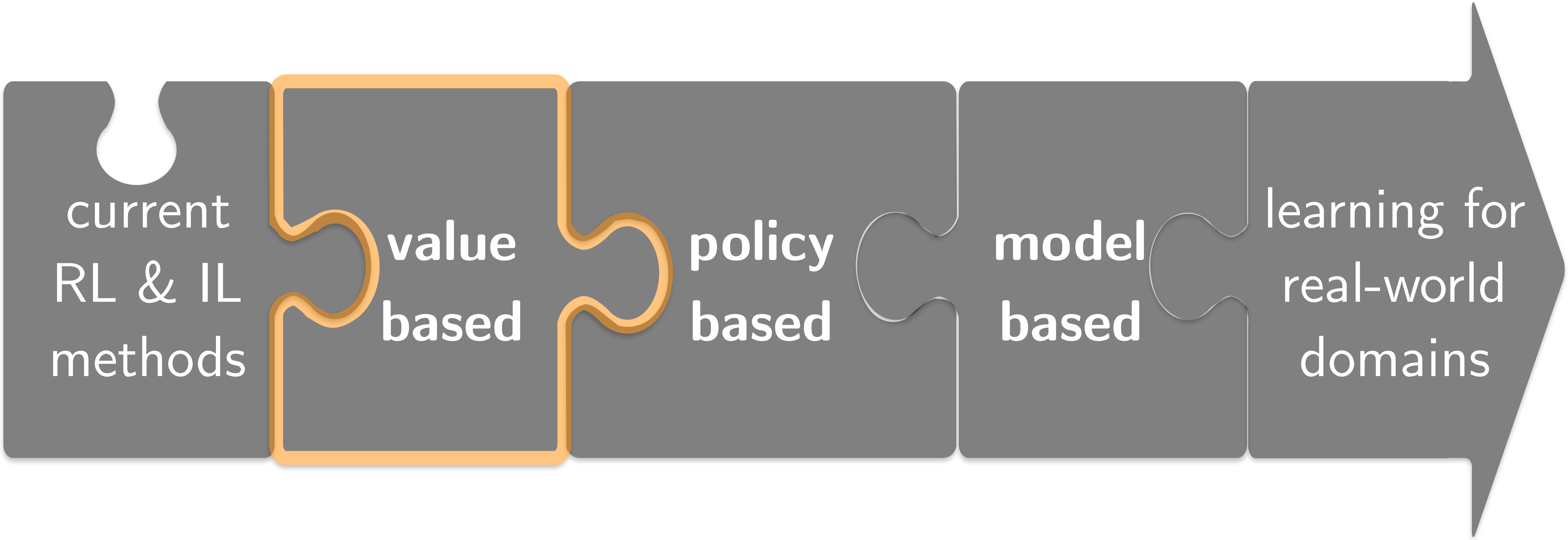
Expensive expert data

Sub-optimal expert


current
RL & IL
methods
Needed to close the gap:
data efficiency
realistic constraints
learning for
real-world
domains
Safe
Set
safety
smoothness
V
V(x1,x2)
grad V
grad V
0
x2
x1
A
dX
dt
stability
Pre
Conf
Simu
SMC
verifiability
fairness
AlphaGo


current
RL & IL
methods
Structured Policy Learning
=
domain knowledge + policy learning
learning for
real-world
domains


current RL & IL methods
value based
policy based
model based
learning for real-world domains


current RL & IL methods
value based
policy based
model based
learning for real-world domains

Why value-based

Usual RL objective: find π

$$\min_{\pi} \quad C(\pi) = \mathbb{E}\left[\sum c(\text{state}, \text{action})\right]$$

scalar cost objective

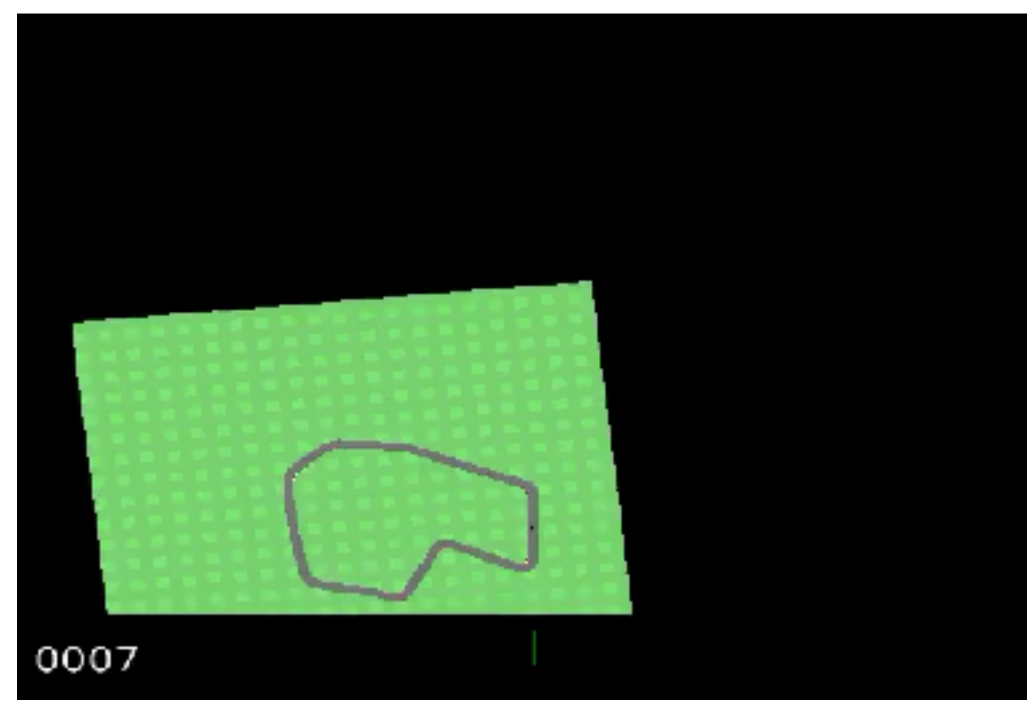
Reality: hard to define a single cost function

Multi-criteria value-based constraints

$$\min_{\pi} \quad \text{travel time}$$

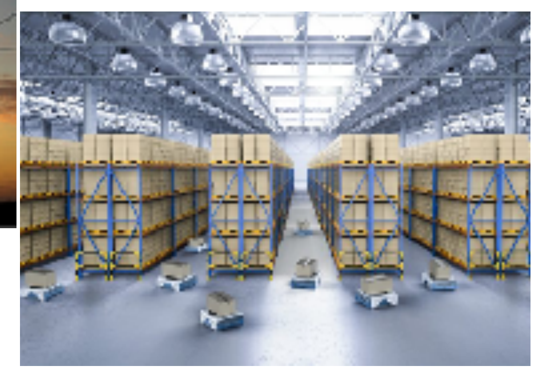
$$\text{s.t. lane centering}$$

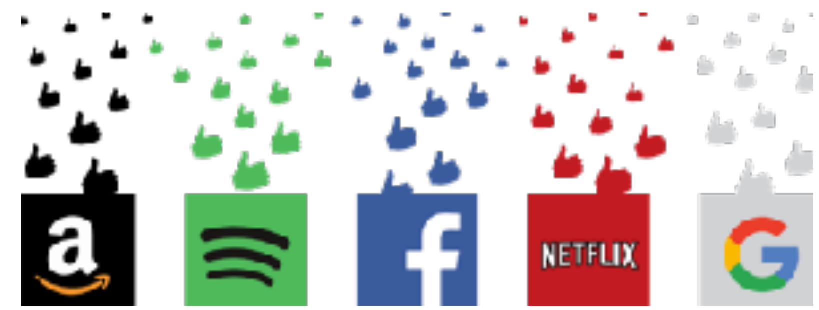
$$\text{smooth driving}$$



Online RL: changed cost objective $\Longrightarrow$ need to solve a fresh problem

off-policy with value-based constraints





π_{D} generates historical (sub-optimal) data

- **Learn better policy from data under multiple value-based constraints?**

Given: n tuples data set $\mathrm{D} = \{(\text{state}, \text{action}, \text{next state}, c, g)\} \sim \pi_{\mathrm{D}}$

Goal: find π

$$\begin{aligned} \min_{\pi} \quad & C(\pi) \\ \text{s.t.} \quad & G(\pi) \leq \tau \end{aligned}$$

m valued-based constraints

$$G(\pi) = \mathbb{E}\left[\sum g(\text{state}, \text{action})\right] \quad g = \begin{bmatrix} g_1 & g_2 & \ldots & g_m \end{bmatrix}^{\top}$$

Example:

Counterfactual & Safe policy learning $g(x) = \mathbf{1}\left[x = x_{avoid}\right]$

Lagrangian $$L(\pi, \lambda) = C(\pi) + \lambda^\top G(\pi)$$

(P) $$\min_{\pi} \max_{\lambda \geq 0} L(\pi, \lambda)$$

(D) $$\max_{\lambda \geq 0} \min_{\pi} L(\pi, \lambda)$$

Policy class convexification: Allow *randomized policies* to handle non-convex costs

Proposed Approach: Solving a repeated game between π and λ

Lagrangian $$L(\pi, \lambda) = C(\pi) + \lambda^\top G(\pi)$$

(P) $$\min_{\pi} \max_{\lambda \geq 0} L(\pi, \lambda)$$

(D) $$\max_{\lambda \geq 0} \min_{\pi} L(\pi, \lambda)$$

Algorithm (rough sketch)

Iteratively:

1: $\pi \leftarrow$ Best-response(λ) → batch RL w.r.t. $c + \lambda^\top g$

Lagrangian $$L(\pi, \lambda) = C(\pi) + \lambda^\top G(\pi)$$

$$(P) \qquad \min_{\pi} \max_{\lambda \geq 0} L(\pi, \lambda)$$

$$(D) \qquad \max_{\lambda \geq 0} \min_{\pi} L(\pi, \lambda)$$

Algorithm (rough sketch)

Iteratively:

1: $\pi \leftarrow$ Best-response(λ)

2: L_{max} = evaluate (D) fixing π

3: L_{min} = evaluate (P) fixing λ

4: if $L_{max} - L_{min} \leq \omega$:

5: stop

6: new $\lambda \leftarrow$ Online-algorithm(all previous π)

Regret $= O(\sqrt{T}) \implies$ convergence in $O(\frac{1}{\omega^2})$ iterations

Off-policy evaluation

Given $\mathrm{D} = \{(\text{state}, \text{action}, \text{next state}, c)\} \sim \pi_\mathrm{D}$ estimate $\widehat{C}(\pi) \approx C(\pi)$

Fitted Q Evaluation (simplified)

For K iterations:

Solve for $Q : (\text{state}, \text{action}) \mapsto y = c + Q_{prev}(\text{next state}, \pi(\text{next state}))$

Return value of Q_K

Guarantee for FQE

For $n = poly(\frac{1}{\epsilon}, \log\frac{1}{\delta}, \log K, \log m, \mathtt{dim}_\mathrm{F})$, *with probability* $1 - \delta$:

$$|C(\pi) - \widehat{C}(\pi)| \le O(\sqrt{\beta}\epsilon)$$

distribution shift coefficient of MDP

End-to-end Performance Guarantee

For $n = \textit{poly}(\frac{1}{\epsilon}, \log\frac{1}{\delta}, \log K, \log m, \mathtt{dim}_{\mathrm{F}})$, *with probability* $1-\delta$:

$$\text{C(returned policy)} - \text{C(optimal)} \leq O(\omega + \sqrt{\beta}\epsilon)$$

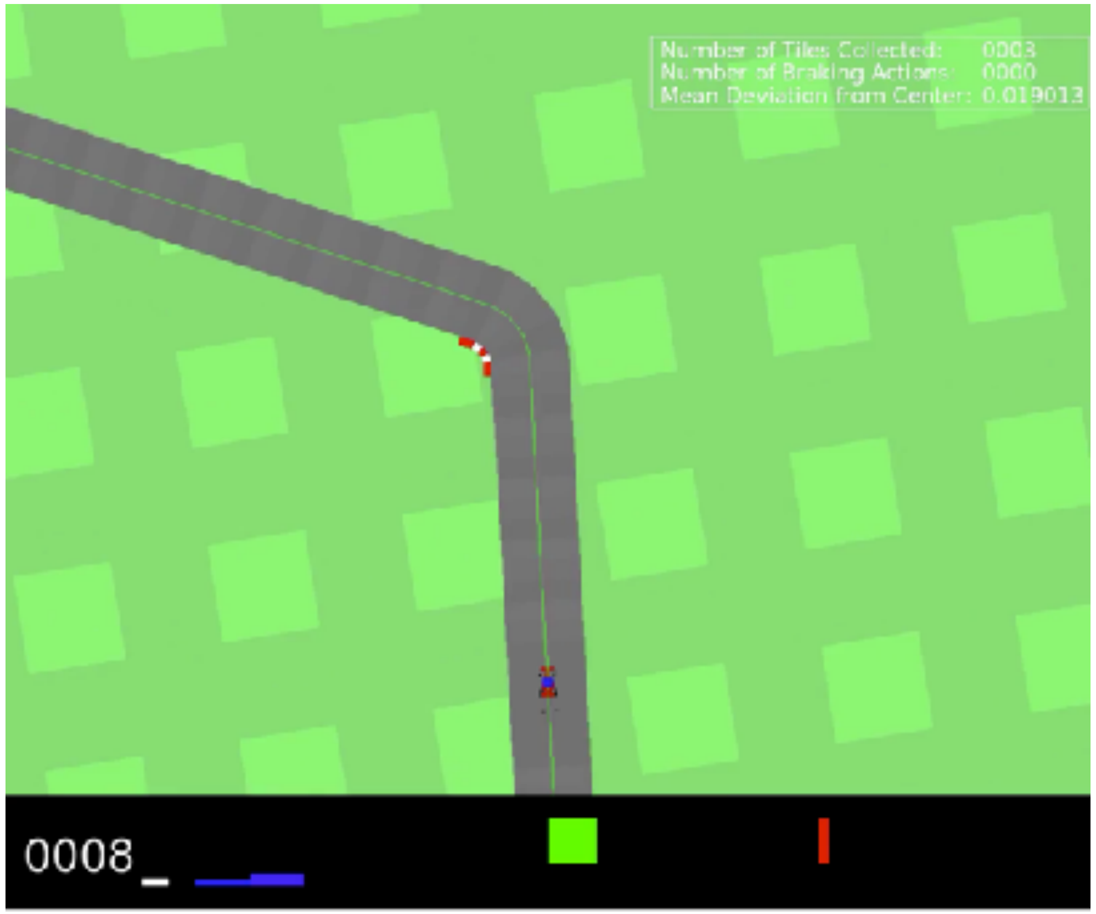
and

$$\text{constraint violation} \leq O(\omega + \sqrt{\beta}\epsilon)$$

stopping condition

minimize travel time

π_{D}



returned policy

Results:

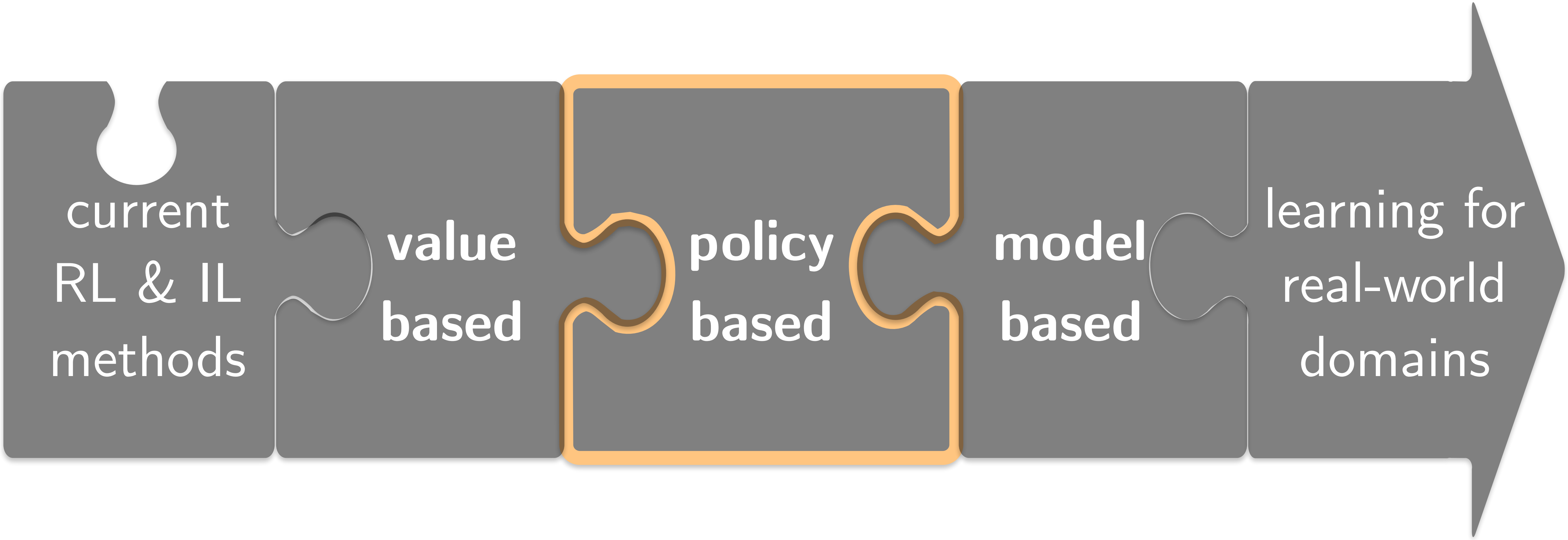
- both constraints satisfied
- travel time still matches online RL optimal

Learning with value-based constraints

- Value-based constraint specification: Flexible to encode domain knowledge

- Data efficiency from off-line policy learning and counterfactual cost function modification

- Extensive benchmarking of OPE: FQE among the best methods
  - *Empirical Study of Off-policy Policy Evaluation for Reinforcement Learning - VoloshinLeJiangYue -* ***(submitted)***


current RL & IL methods
value based
policy based
model based
learning for real-world domains

Why policy-based

- Encoding structure into policy class can be more natural
- Benefit: policy-based guarantee
- Example 1: symbolic verification of ***programs*** (& interpretable)

$$
\begin{aligned}
&\textbf{if } (\mathbf{obs}_{\texttt{TrackPos}}(0) < 0.001 \textbf{ and } \mathbf{obs}_{\texttt{TrackPos}}(0) > -0.001)\\
&\qquad \textbf{then } PID_{\texttt{rpm}}(0.44, 4.92, 0.89, 49.79)\\
&\qquad \textbf{else } PID_{\texttt{rpm}}(0.40, 4.92, 0.89, 49.79)
\end{aligned}
$$

"if the car is aligned with the axis of the track..."

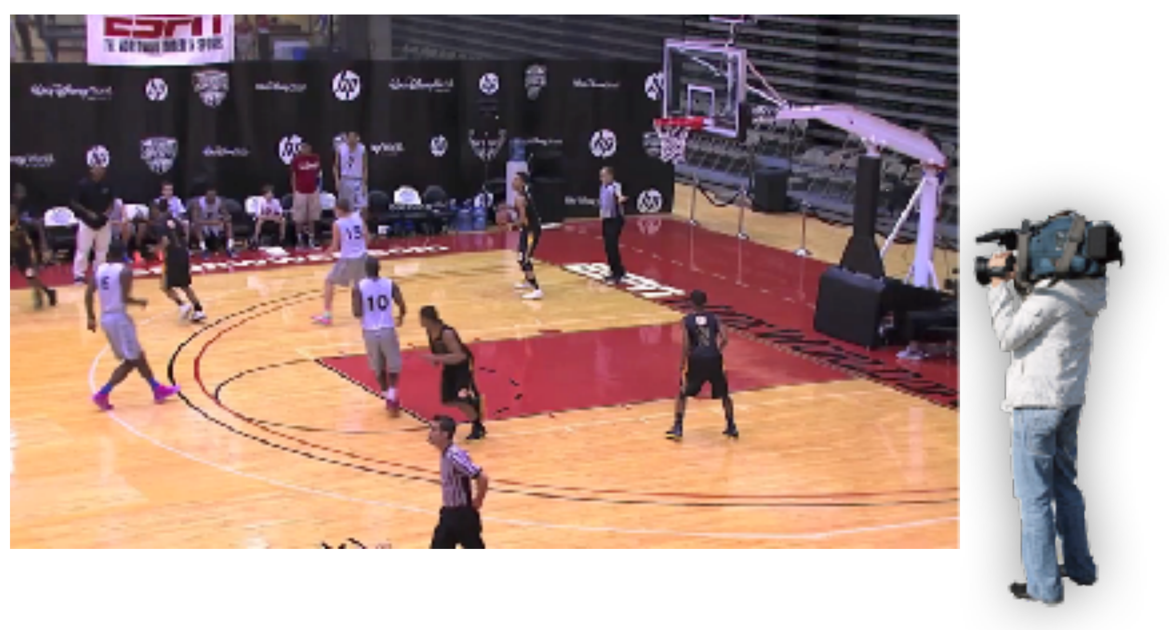
"then accelerate, otherwise slow down"

Why policy-based

- Encoding structure into policy class can be more natural

- Benefit: policy-based guarantee

- Example 2: smoothness guarantee



$\pi_\theta(x)$ is smooth, e.g., $L_\Pi < 1$

Integrate policy structure

- Neural policy class F : deep RL, IL
  - flexible, but unstable and does not satisfy desired property
- Programmatic policy class Π
  - less flexible, but certifiable

<u>Aside:</u>

regularization in supervised learning

$$\min_{\theta} L(\theta) + \lambda R(\theta)$$

prior knowledge on θ

Integrate policy structure

- Neural policy class F : deep RL, IL
  - flexible, but unstable and does not satisfy desired property
- Programmatic policy class Π
  - less flexible, but certifiable
- Hybrid representation (policy class regularization)

$$H \equiv \Pi \oplus F$$

$h \equiv \pi + \lambda f$ defined as $h(x) = \pi(x) + \lambda f(x)$

Programmatic reinforcement learning

- The program space Π
  - language (arithmetic, boolean, relational) over simple policies

- **Goal:** find the best program

$$\pi^* = \operatorname{argmin}_{\pi \in \Pi} C(\pi)$$

- Learning programmatic policies (program synthesis): highly structured nature of policy space

- **Approach:**

Building program structure into policy search via "lift-and-project"

Imitation-Projected Policy Gradient for Programmatic Reinforcement Learning
- ***Le****VermaYueChaudhuri* - ***NeurIPS 2019***

Imitation-projected policy gradient

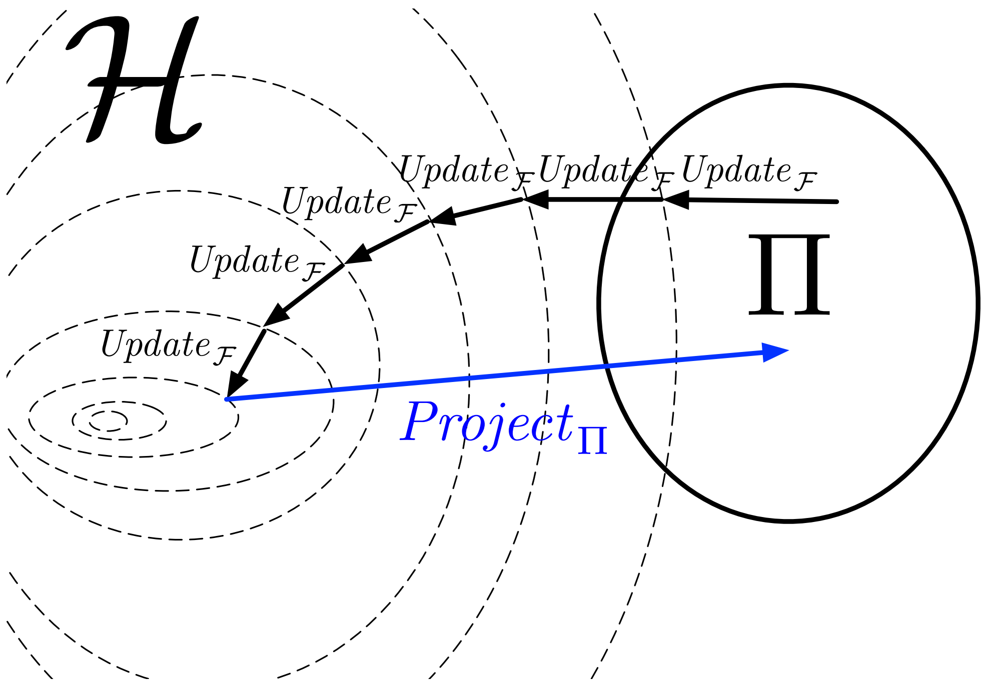
hybrid class: $\mathrm{H} \equiv \Pi \oplus \mathrm{F}$

each iteration: $h_t \leftarrow \text{UPDATE}_{\mathrm{F}}(\pi_{t-1})$

$\pi_t \leftarrow \text{PROJECT}_{\Pi}(h_t)$

UPDATE: $f \leftarrow f - \eta\lambda \nabla_{\mathrm{F}} C(\pi + \lambda f)$

$h \leftarrow \pi + \lambda f$



PROJECT: imitation learning

Approximate Mirror Descent

hybrid class: $\mathrm{H} \equiv \Pi \oplus \mathrm{F}$

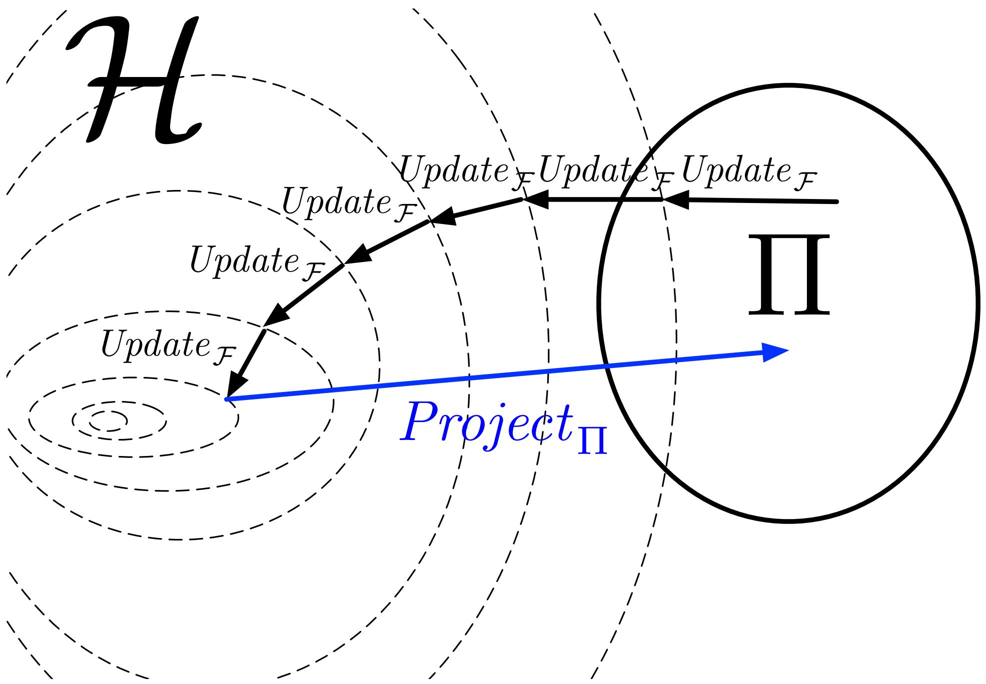
each iteration: $h_t \leftarrow \text{UPDATE}_{\mathrm{F}}(\pi_{t-1}) \approx \text{UPDATE}_{\mathrm{H}}(\pi_{t-1})$

$\pi_t \leftarrow \text{PROJECT}_{\Pi}(h_t) \approx \mathrm{argmin}_{\pi \in \Pi} ||\pi - h_t||^2$

UPDATE: $f \leftarrow f - \eta\lambda \nabla_{\mathrm{F}} C(\pi + \lambda f)$

$h \leftarrow \pi + \lambda f$

$\text{UPDATE}_{\mathrm{H}}(\pi_{t-1}) = \pi_{t-1} - \nabla_{\mathrm{H}} C(\pi_{t-1})$



Experiment

Experiment

Generalization: IPPG completed 12/20 unseen tracks, DDPG completed 3/20

	G-Track	E-Road	Aalborg	Ruudskogen	Alpine-2
G-Track	-	119 / Cr	Cr / Cr	Cr / Cr	Cr / Cr
E-Road	103 / 88	-	Cr / Cr	Cr / Cr	Cr / Cr
Aalborg	199 / 86	221 / 102	-	212 / Cr	214 / Cr
Ruudskogen	124 / Cr	127 / Cr	Cr / Cr	-	Cr / Cr
Alpine-2	210 / Cr	226 / Cr	176 / Cr	227 / Cr	-

“Programmatic” imitation learning

- The program space Π is regularized neural space:

$$\pi = \lambda f_\theta + (1-\lambda) g_w$$

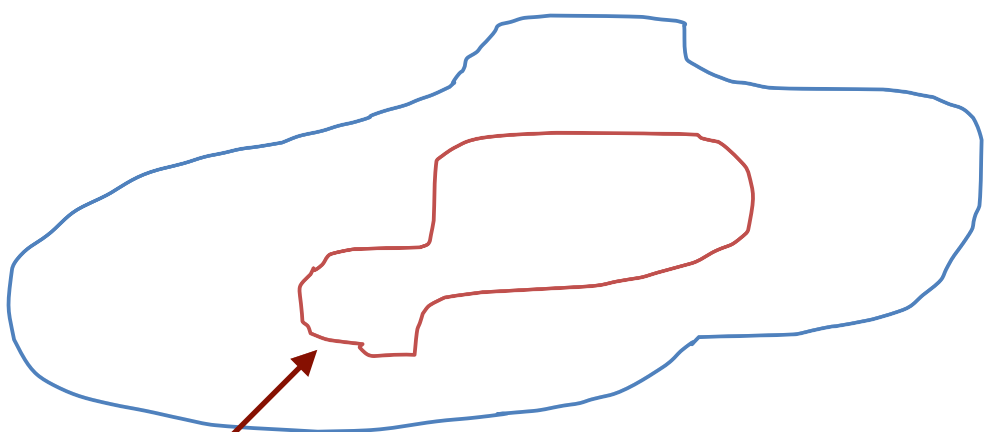
neural net policy

linear policy

- **Goal:** find the best smooth policy

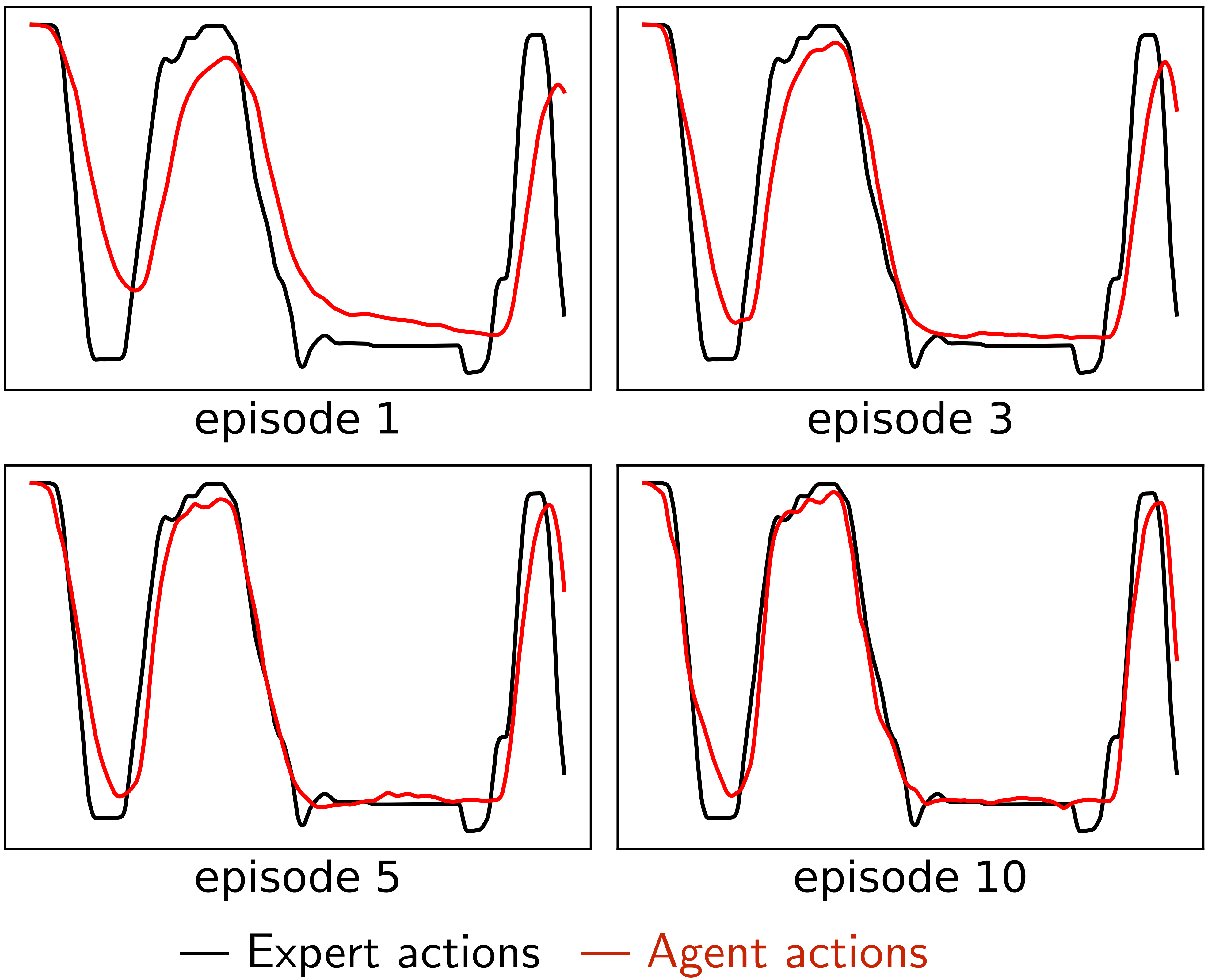
$$\pi^* = \operatorname{argmin}_{\pi \in \Pi} C(\pi)$$

- Friendly case: $\Pi \subset \mathscr{F}$
  - IL for both UPDATE and PROJECT
  - can choose learning rate independent on horizon to guarantee improvement

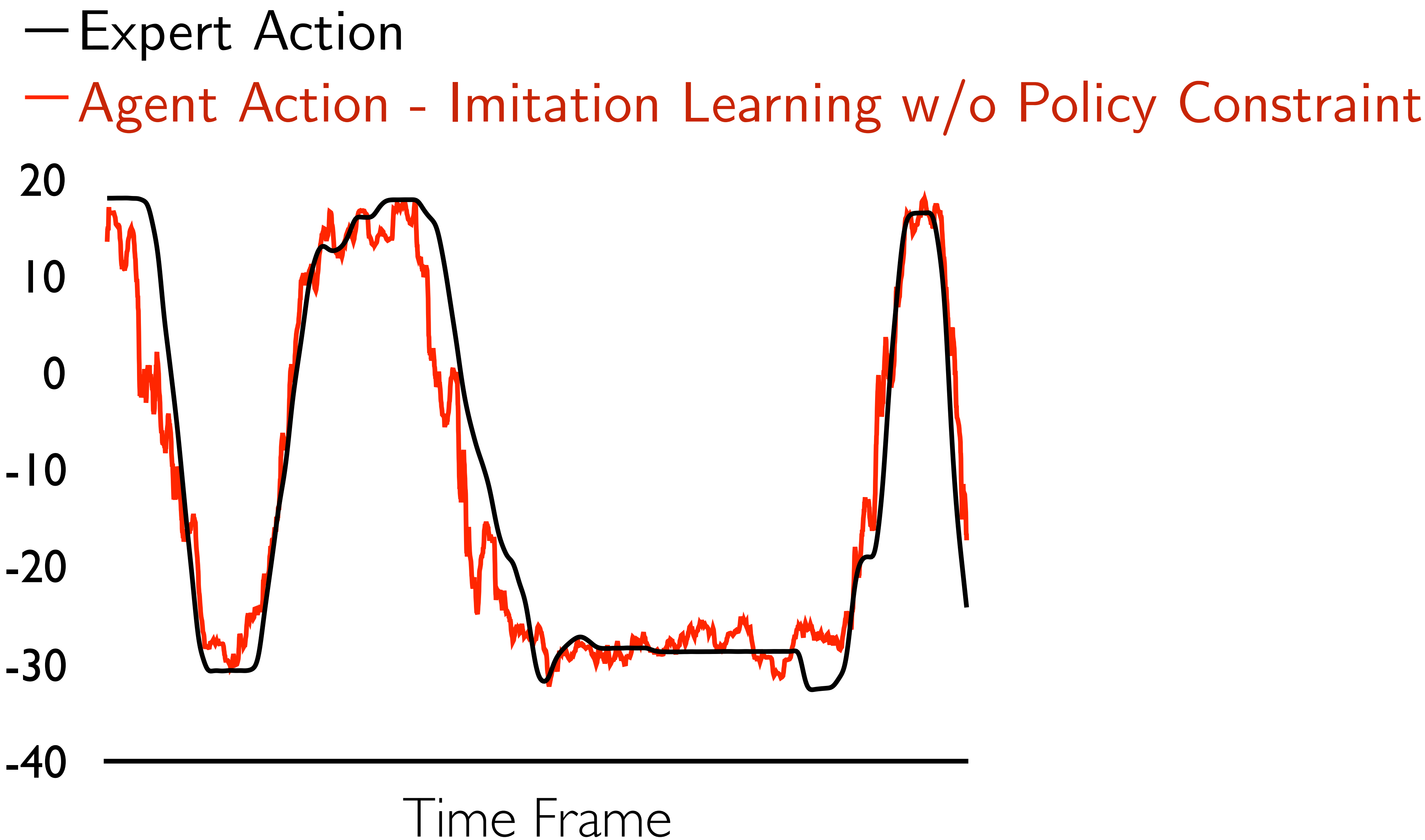


Smooth Imitation Learning for Online Sequence Prediction
- ***Le***_KangYueCarr_ - ***ICML 2016***

Learning progress



vs. standard IL



Application: automated camera

Post-hoc Smoothing

SIMILE

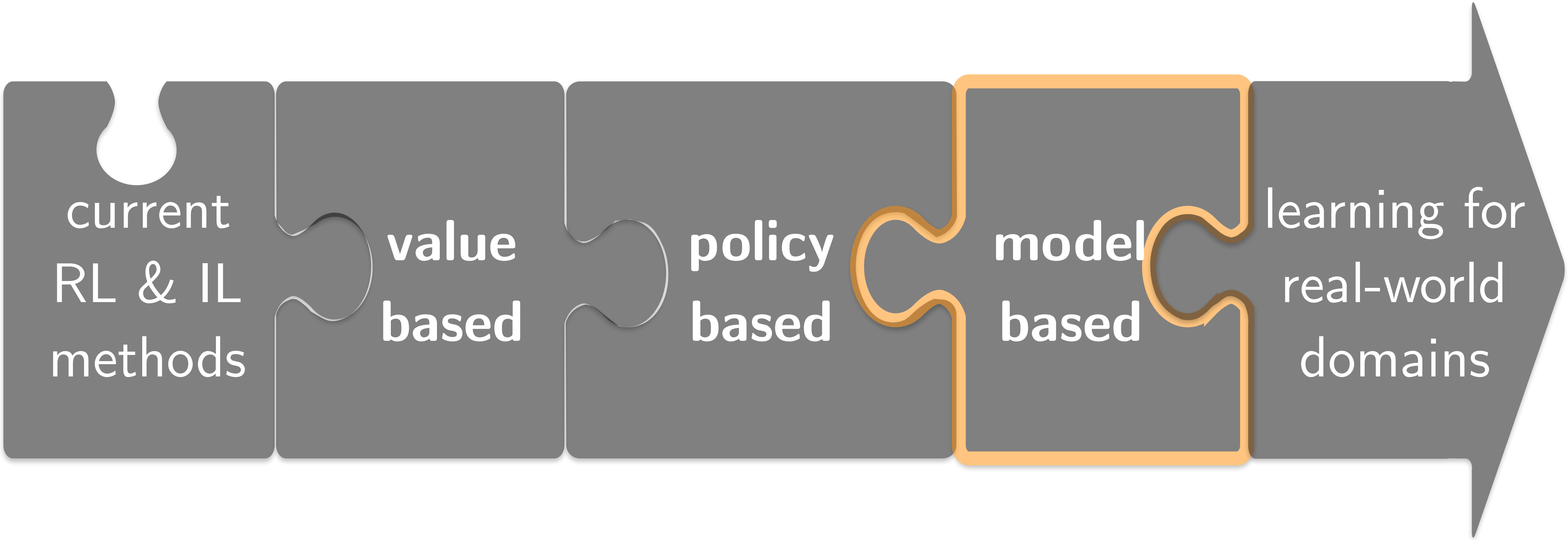
Learning Online Smooth Predictors for Real-time Camera Planning
- *Chen**Le**CarrYueLittle* - ***CVPR 2016 (Oral Presentation)***

Application: off-line video editing

Raw footage

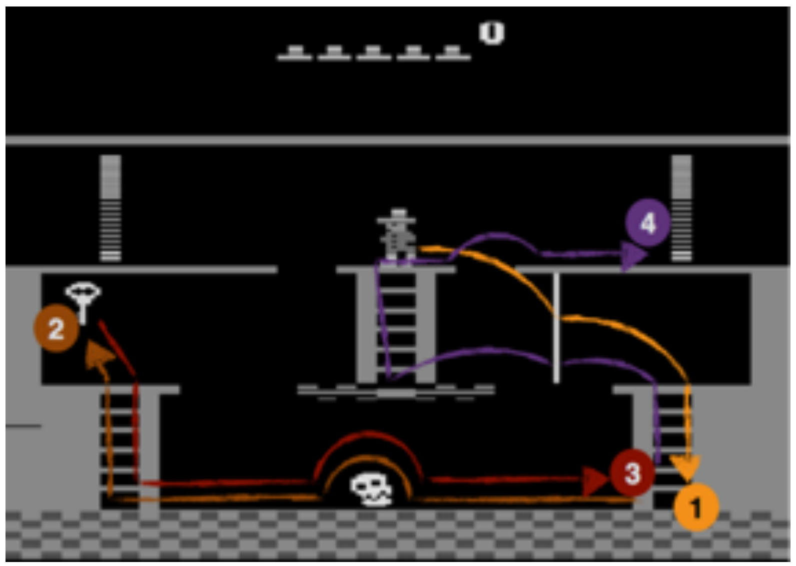
Footage edited by policy

(with Cendon and Yue @ Caltech)

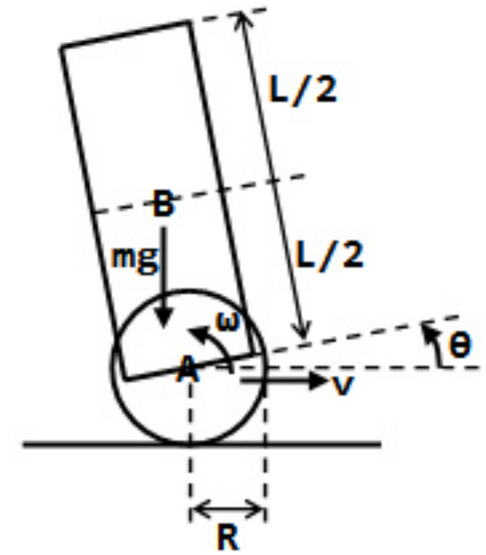

current RL & IL methods
value based
policy based
model based
learning for real-world domains

Why model-based

- Some knowledge about the environment can speed-up learning

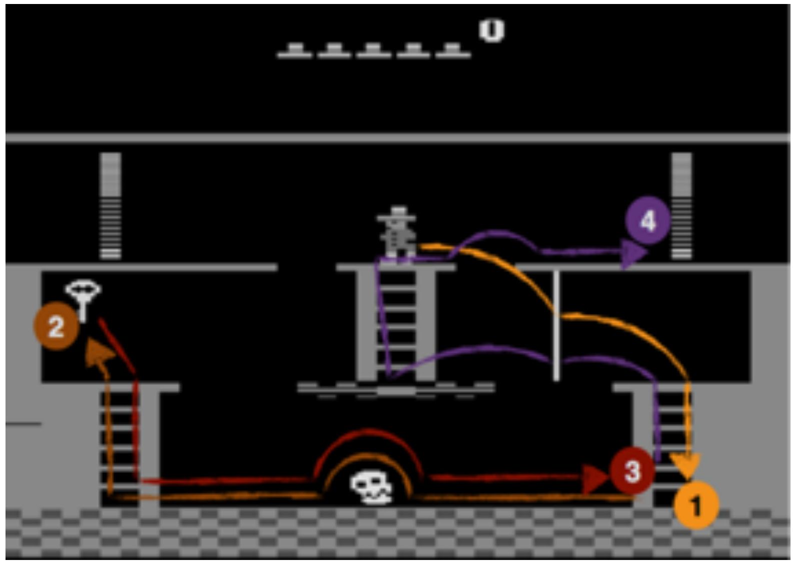


hierarchical structure

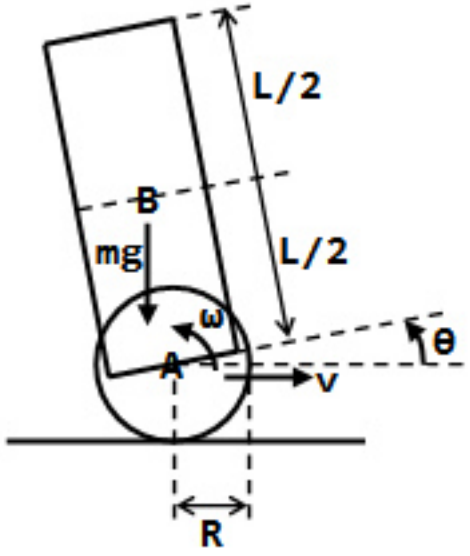


approximate model

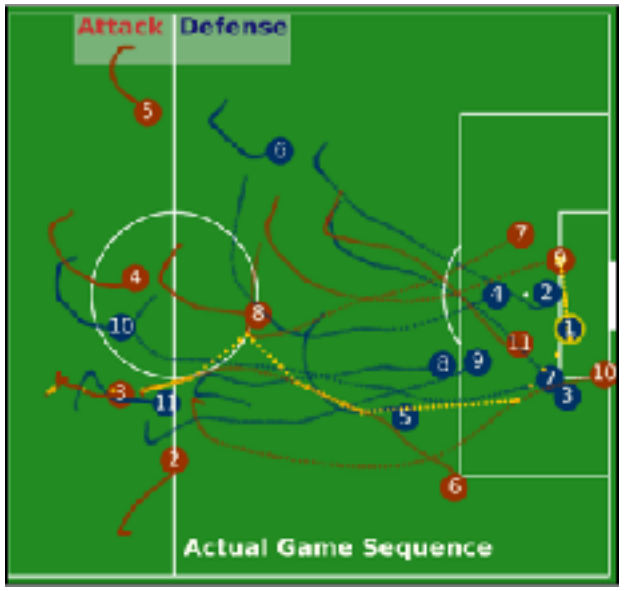
latent structure



hierarchical structure



approximate model

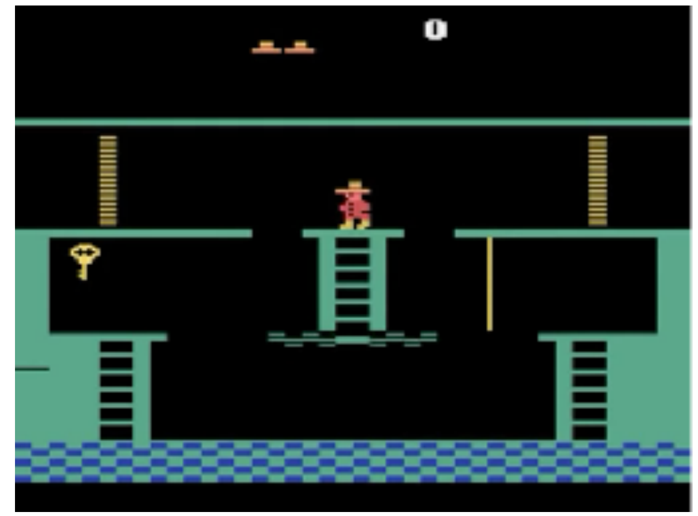


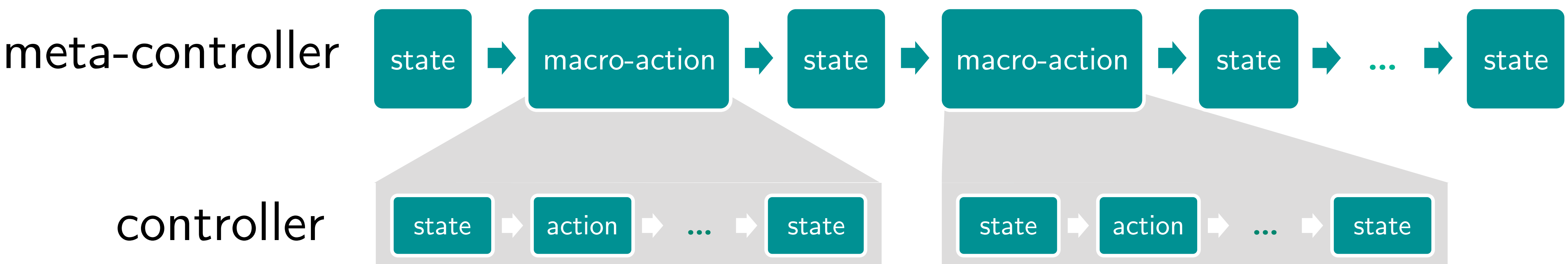
latent structure

Given domain hierarchical structure...

How can we improve data efficiency for imitation and reinforcement learning?

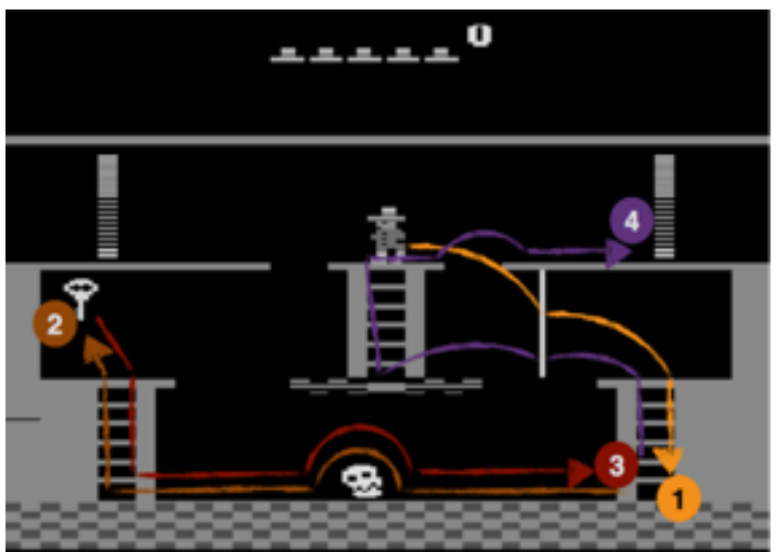
Hierarchical decision making





Alternative feedback mechanism more natural for domain experts?

High-level feedback

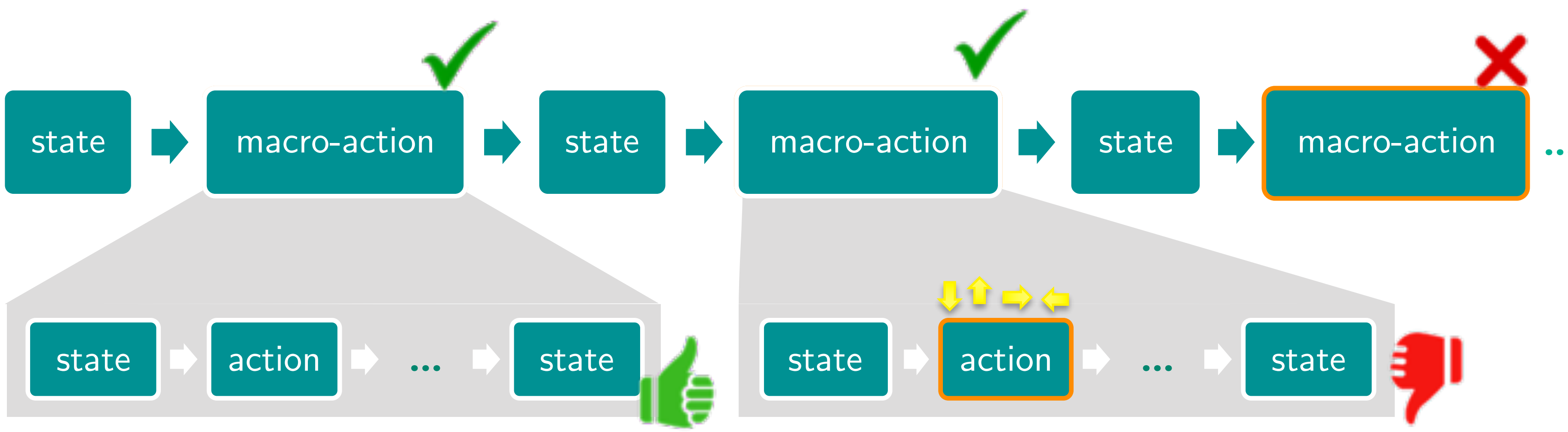


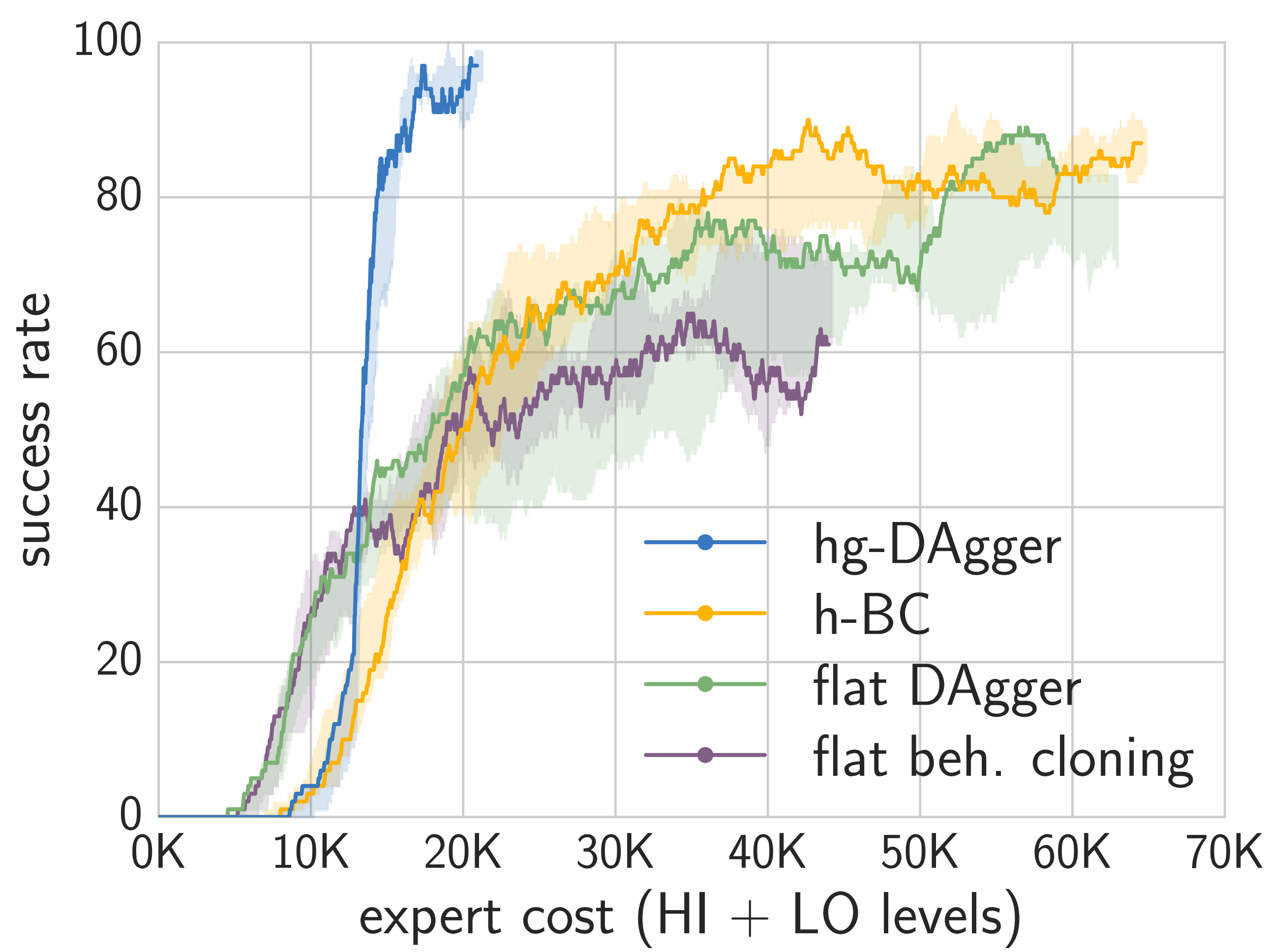
<u>Navigation instruction:</u>
Stair —> Get Key
—> Stair —> Open Door

Verify / "Lazy" Evaluation

Macro-action completed?

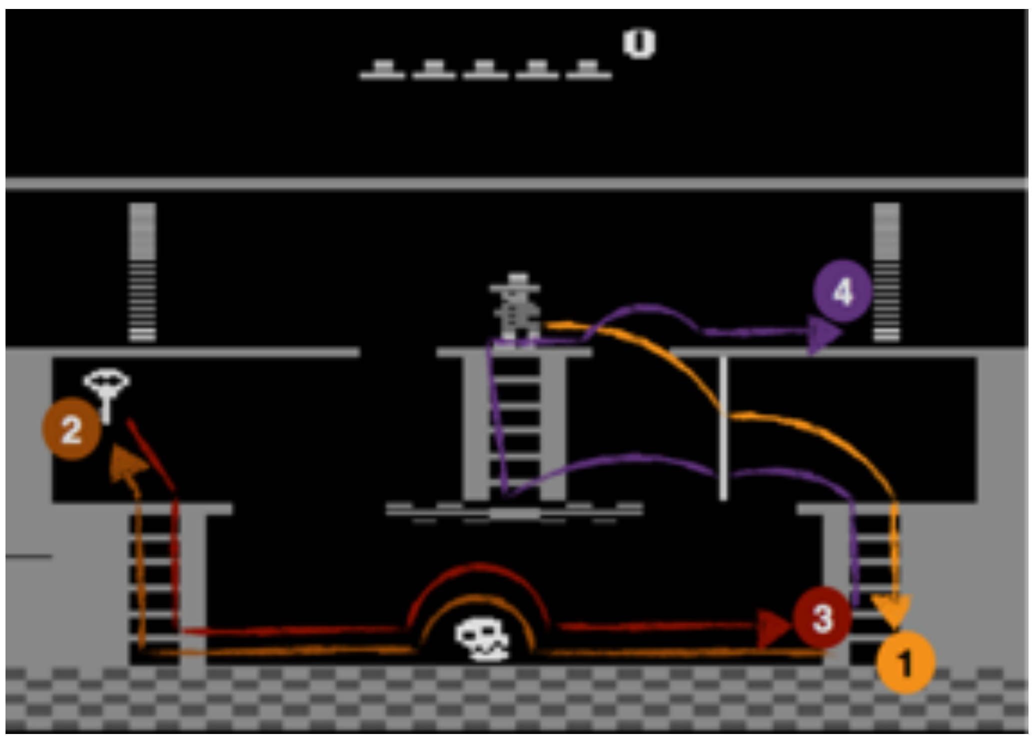
Hierarchical imitation learning

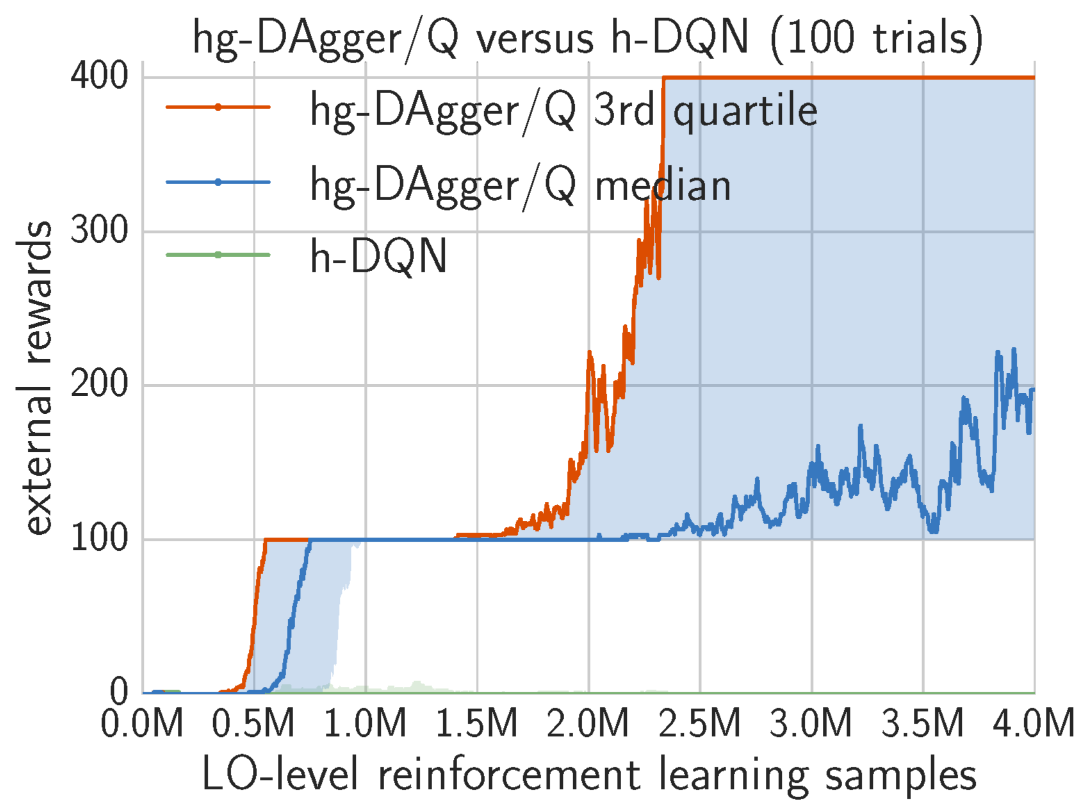




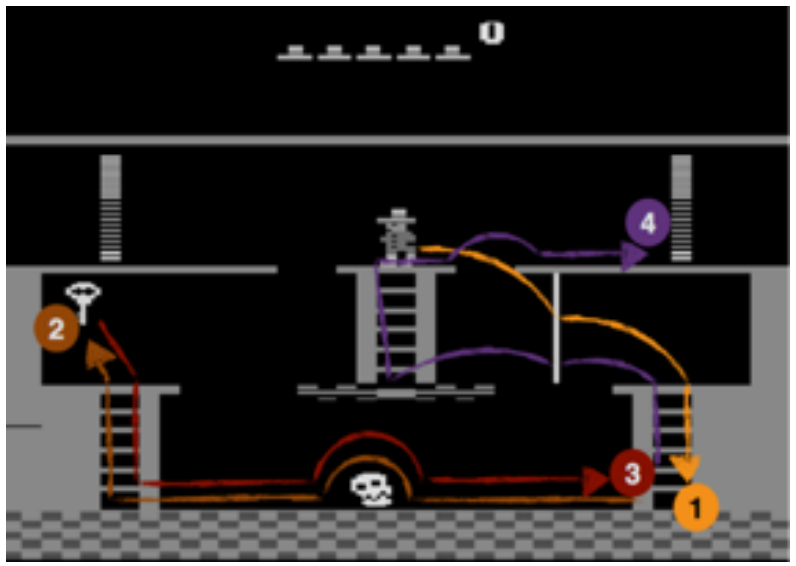
Hierarchical imitation and reinforcement learning

- IL for meta-controller (macro-actions)
- RL/IL for low-level policies

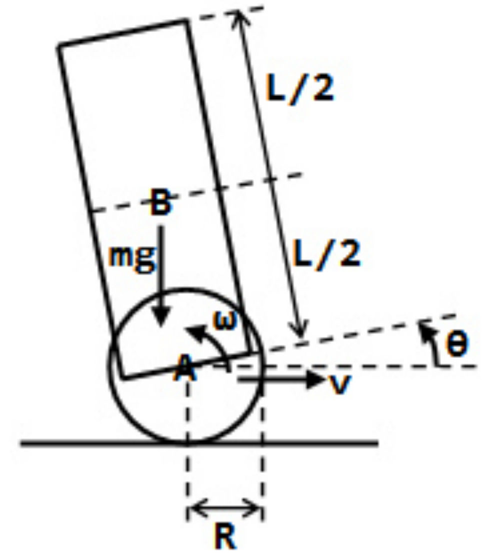




- More data-efficient than flat imitation learning
- Much faster learning than standard reinforcement learning



hierarchical structure

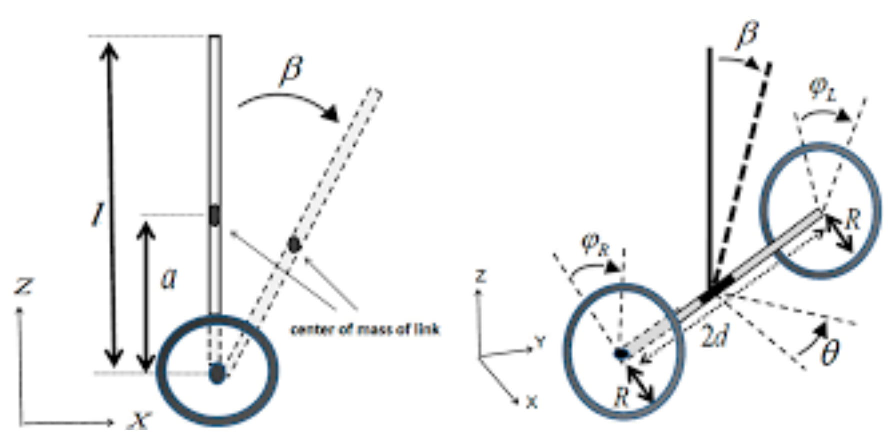


approximate model

latent structure

Approximate model

- Model-based RL: estimate model from data
- Robotics & Control: model from physics



$$\rho\ddot{w}_1 + \rho x_1\ddot{\theta} + EI\left(\frac{6w_1 - 4w_2 + w_3}{h^4}\right) - \rho w_1\dot{\theta}^2 + \eta I\left(\frac{6\dot{w}_1 - 4\dot{w}_2 + \dot{w}_3}{h^4}\right) = 0 \quad (24)$$

for $i = 2, 3, \ldots, n-2$

$$\rho\ddot{w}_i + \rho x_i\ddot{\theta} - \rho w_i\dot{\theta}^2 + EI\left(\frac{-4w_{i-1} + 6w_i - 4w_{i+1} + w_{i-2} + w_{i+2}}{h^4}\right) + \quad (25)$$

$$\eta I\left(\frac{-4\dot{w}_{i-1} + 6\dot{w}_i - 4\dot{w}_{i+1} + \dot{w}_{i-2} + \dot{w}_{i+2}}{h^4}\right) = 0$$

for $i = n-1$

$$\rho\ddot{w}_{n-1} + \rho x_{n-1}\ddot{\theta} - \rho w_{n-1}\dot{\theta}^2 + EI\left(\frac{-4w_{n-2} + 5w_{n-1} - 6w_n + w_{n-3}}{h^4}\right) + \quad (26)$$

$$\eta I\left(\frac{-4\dot{w}_{n-2} + 5\dot{w}_{n-1} - 6\dot{w}_n + \dot{w}_{n-3}}{h^4}\right) = 0$$

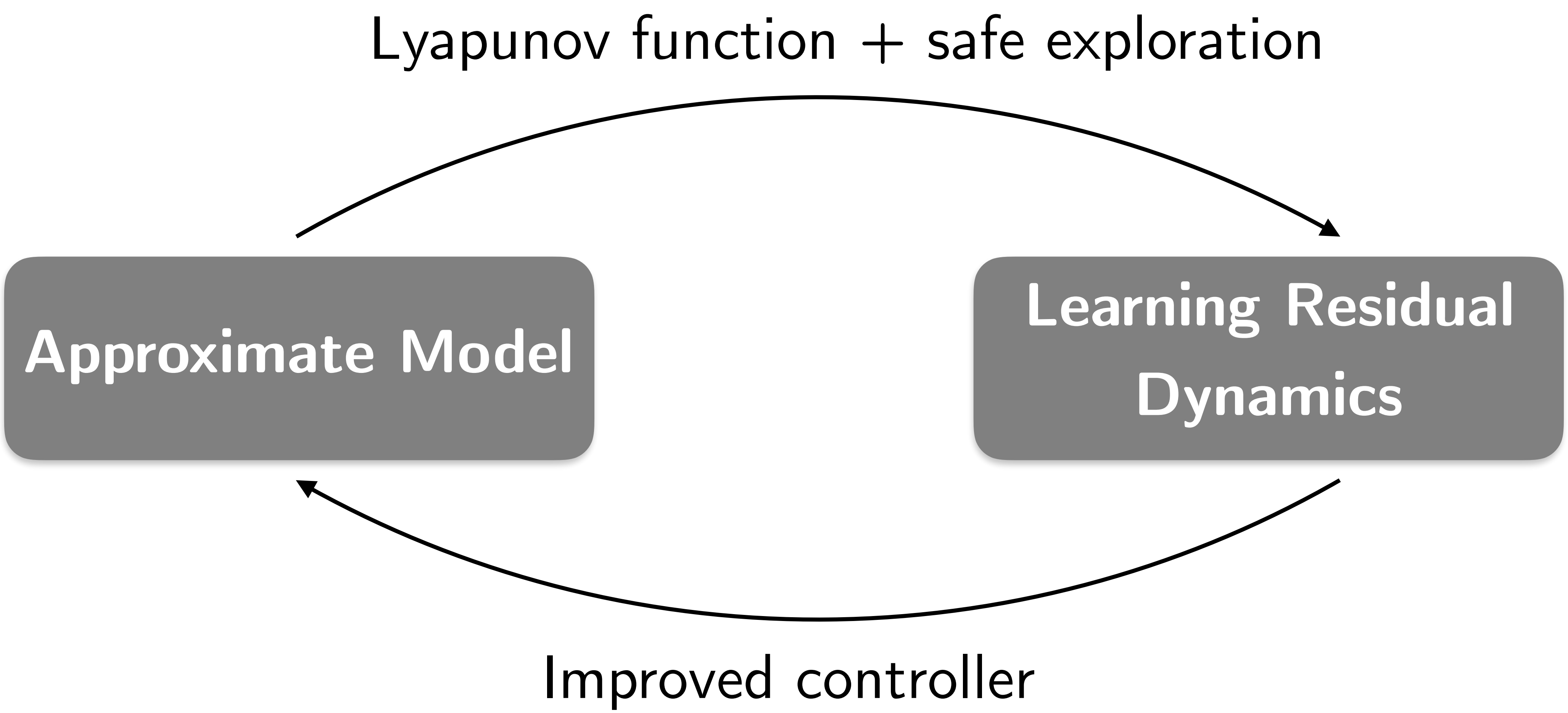
for $i = n$

$$\rho\ddot{w}_n + \rho x_n\ddot{\theta} - \rho w_n\dot{\theta}^2 + EI\left(\frac{-2w_{n-1} + 5w_n + w_{n-2}}{h^4}\right) + \quad (27)$$

$$\eta I\left(\frac{-2\dot{w}_{n-1} + 5\dot{w}_n + \dot{w}_{n-2}}{h^4}\right) = 0$$

- Reinforcement Learning + Control: how to integrate model-based control and learning-based methods ?

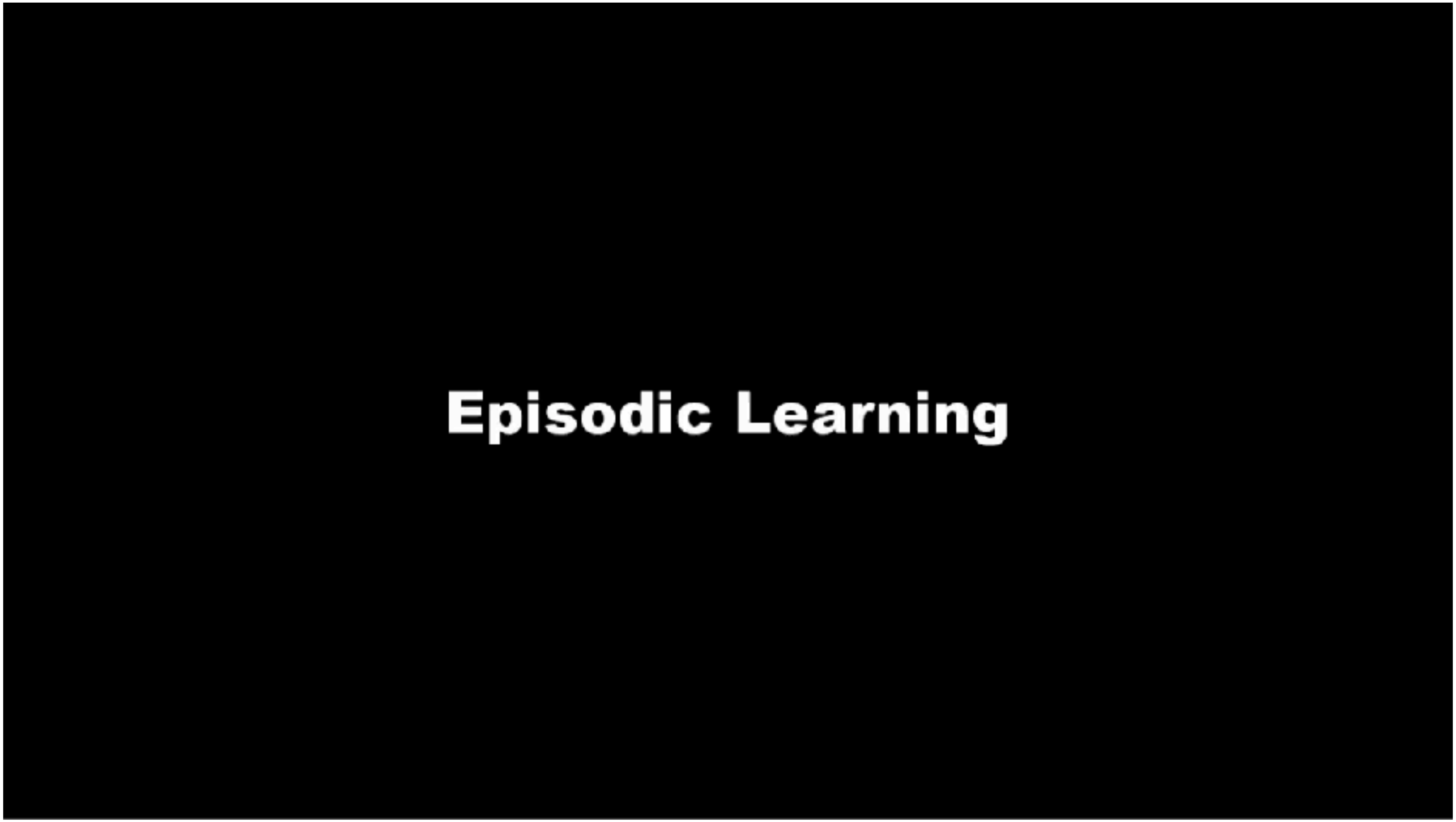
Learning + model-based control



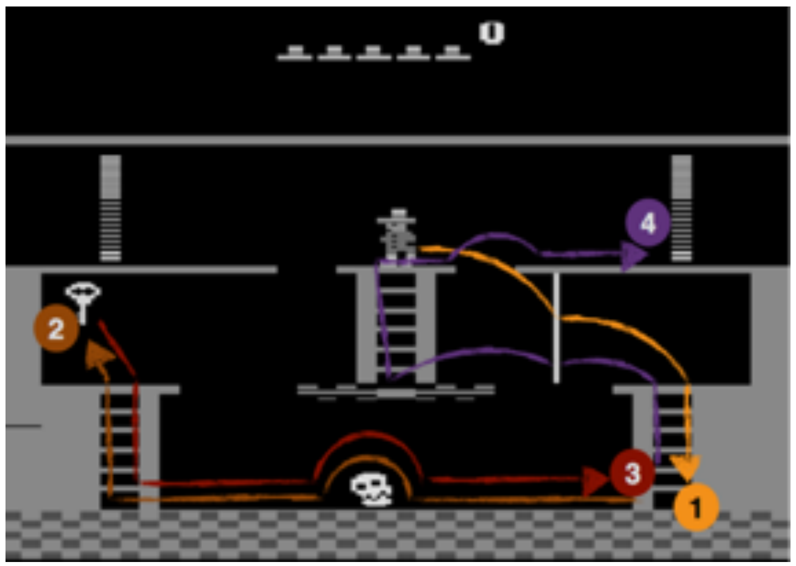
Episodic Learning with Control Lyapunov Functions for Uncertain Robotic Systems
*- Taylor*Dorobantu*__Le__YueAmes -* ***IROS 2019***

A Control Lyapunov Perspective on Episodic Learning via Projection to State Stability
*- Taylor*Dorobantu*Krisnamoothy__Le__YueAmes -* ***CDC 2019***

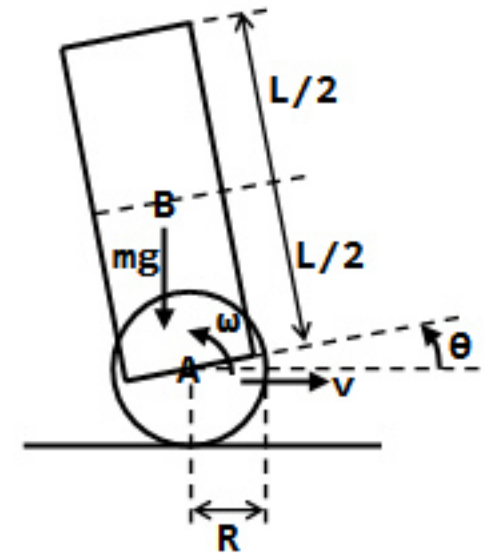
Learning + model-based control



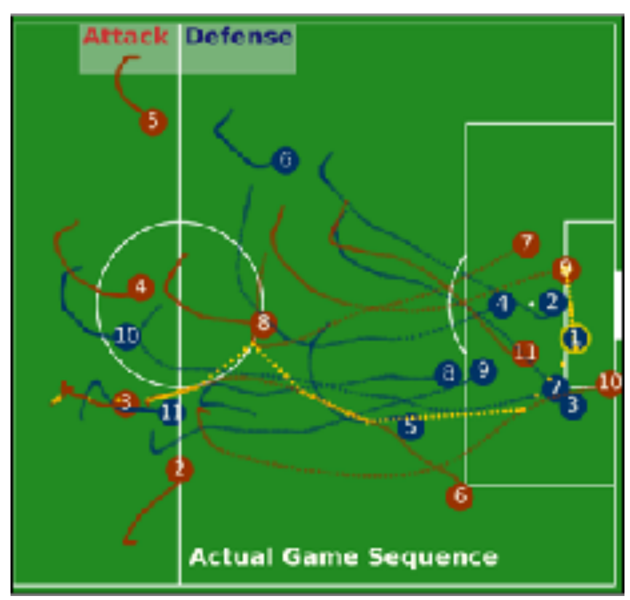
Episodic Learning with Control Lyapunov Functions for Uncertain Robotic Systems
*- Taylor*Dorobantu****Le***YueAmes -* ***IROS 2019***



hierarchical structure

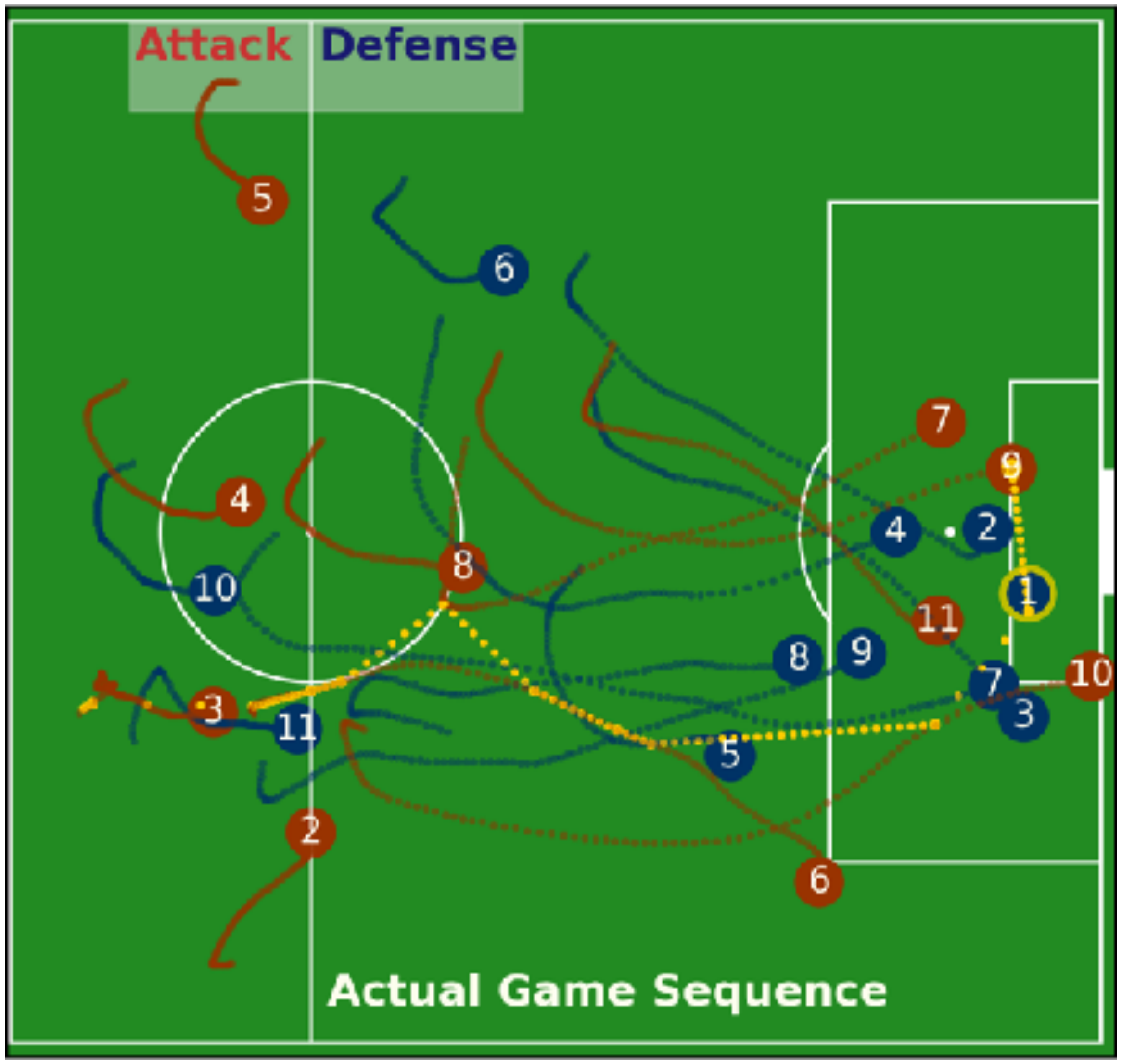


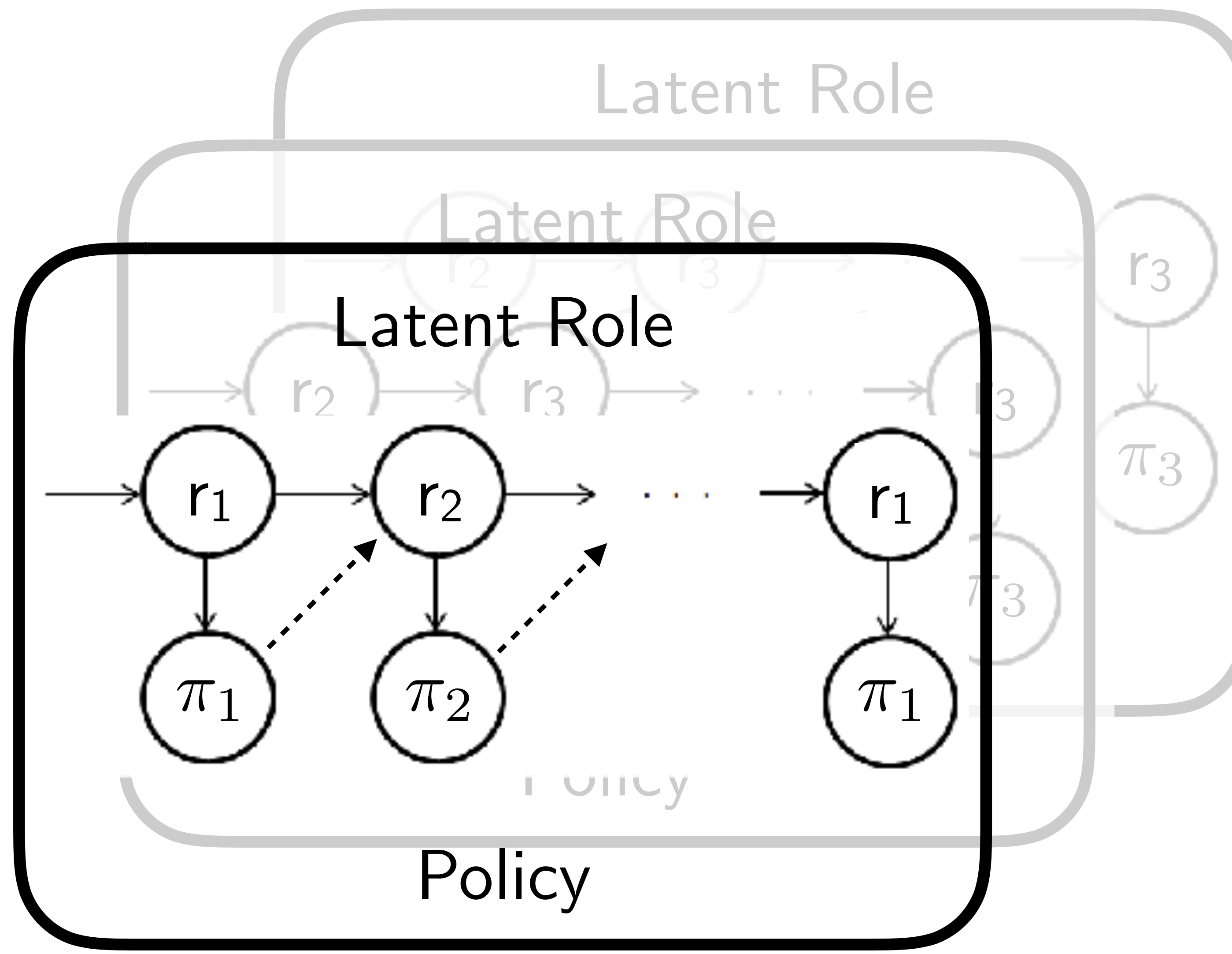
approximate model



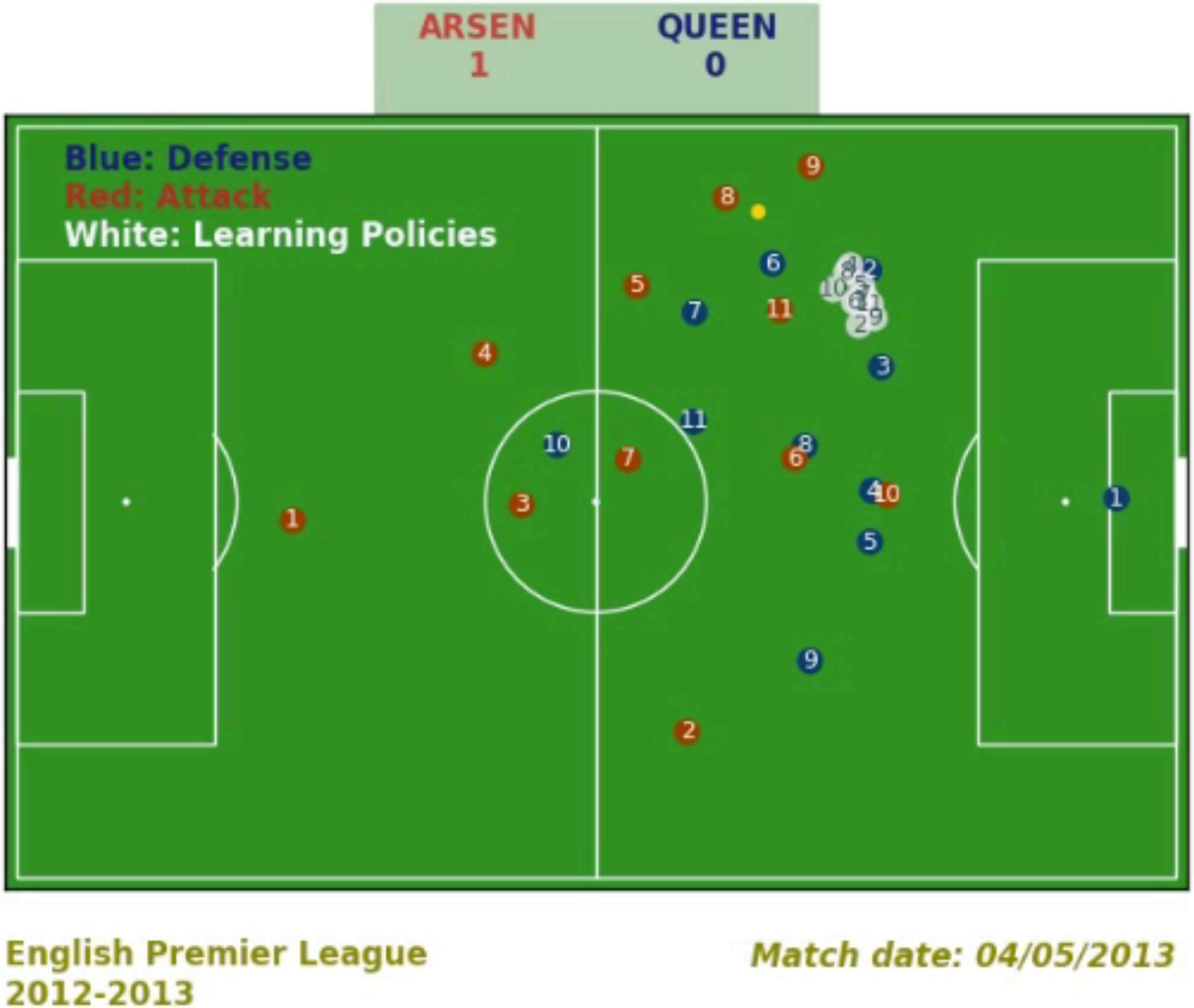
latent structure

Latent structure model

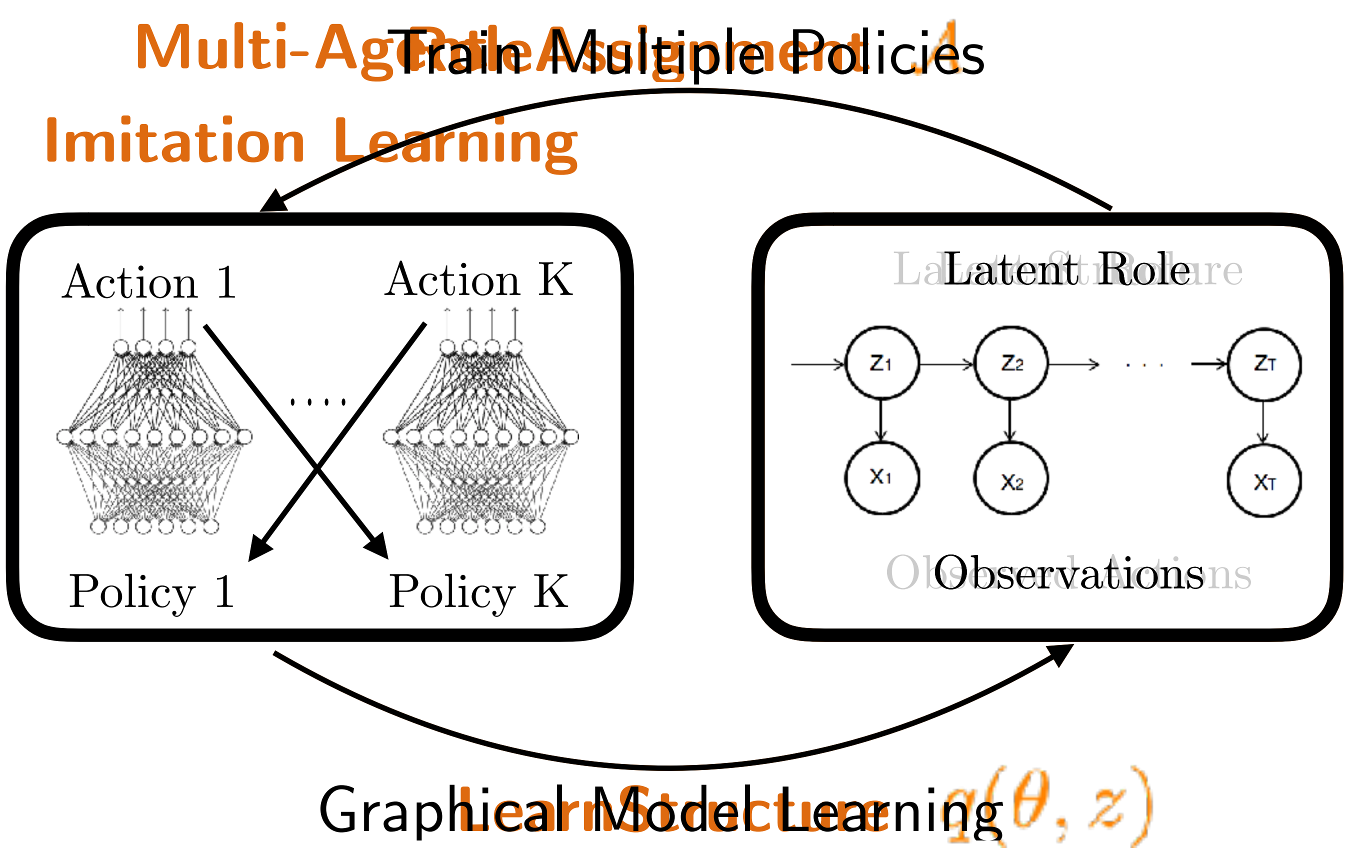




Policy learning w/o latent structure



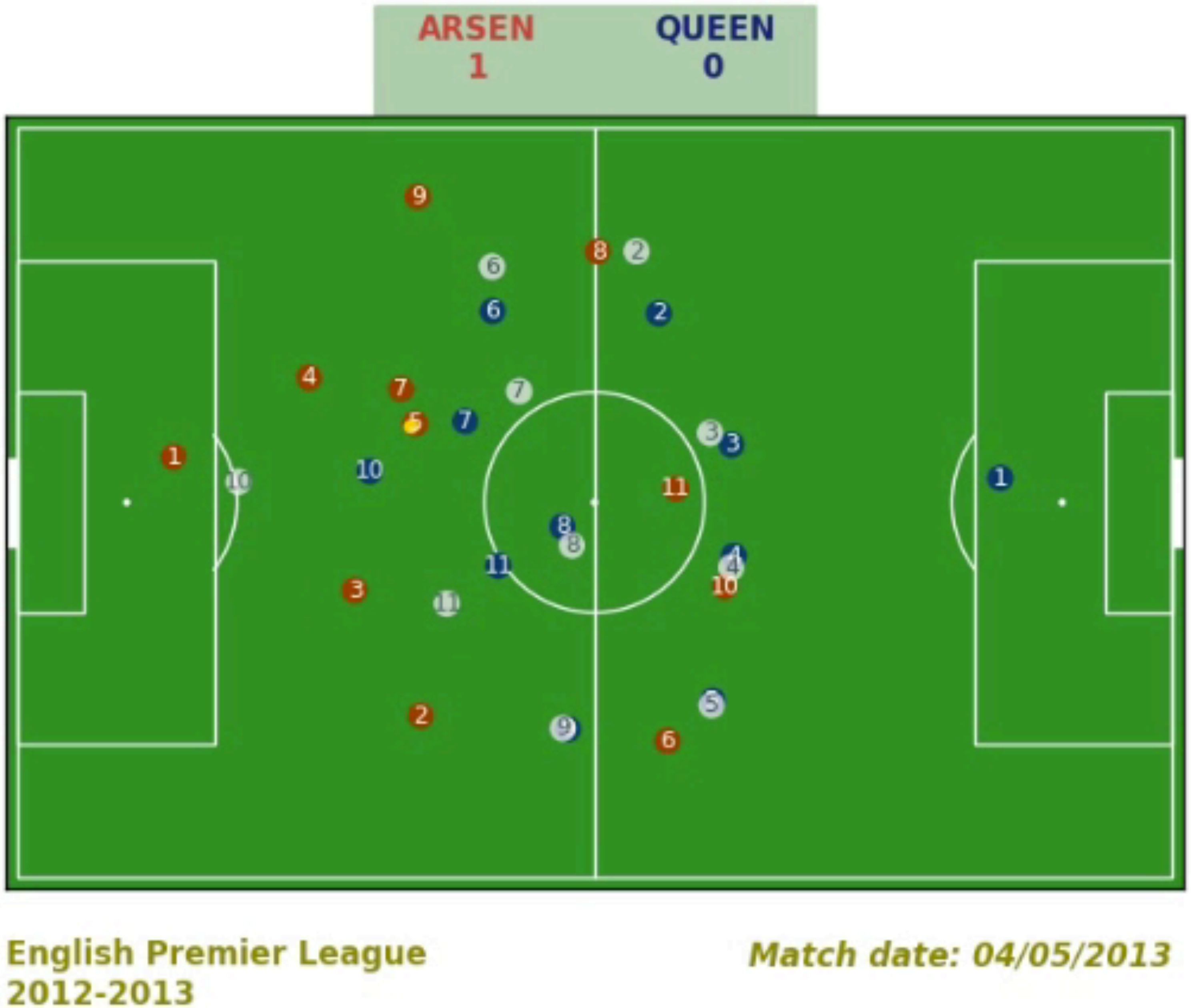
Policy + latent model learning



- Policy learning: reduction to single-agent imitation learning
- Latent structure: unsupervised (stochastic) variational inference

Coordinated Multi-agent Imitation Learning
*- **Le**YueCarrLucey - **ICML 2017***

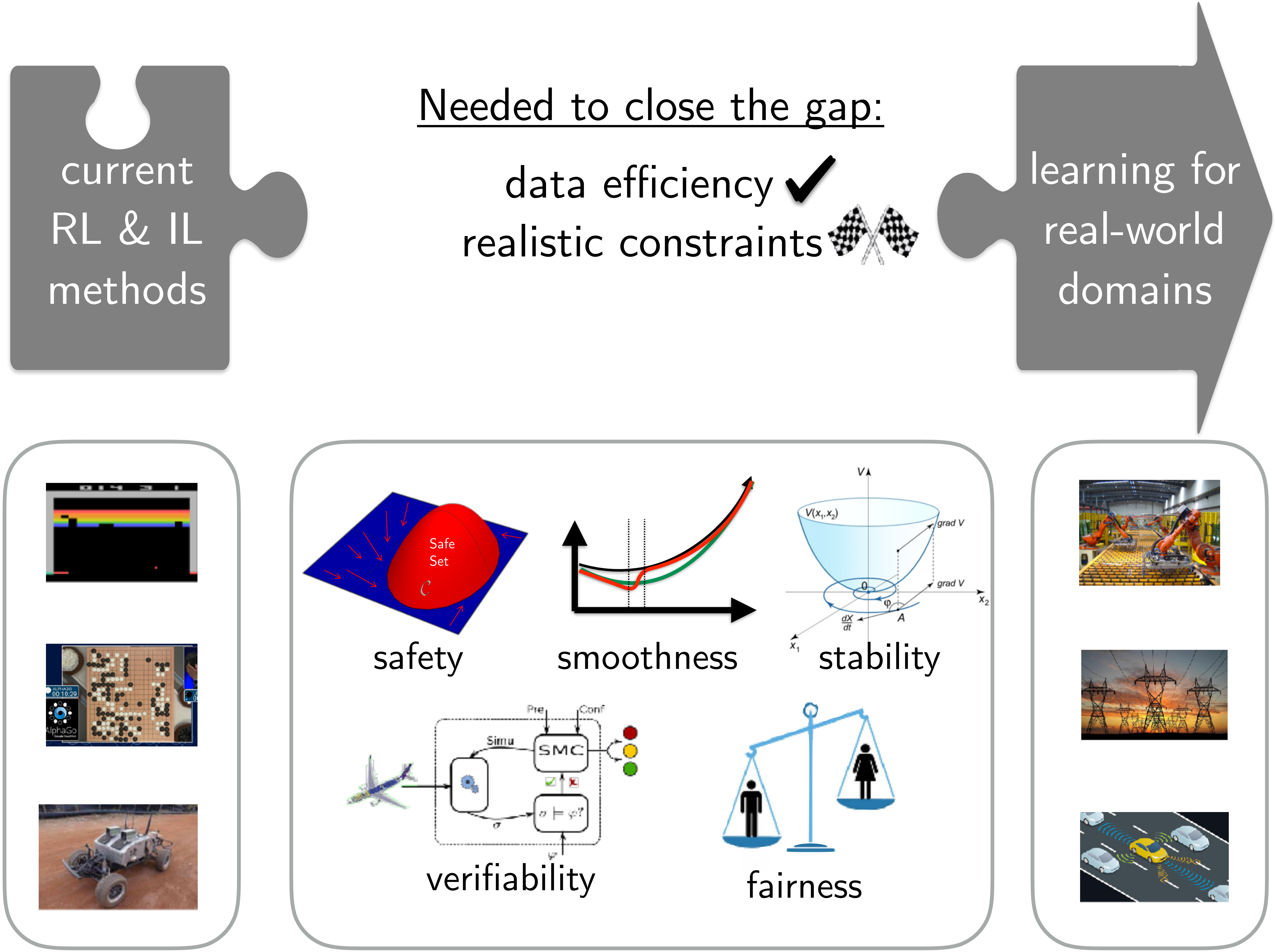
Result on behavior modeling

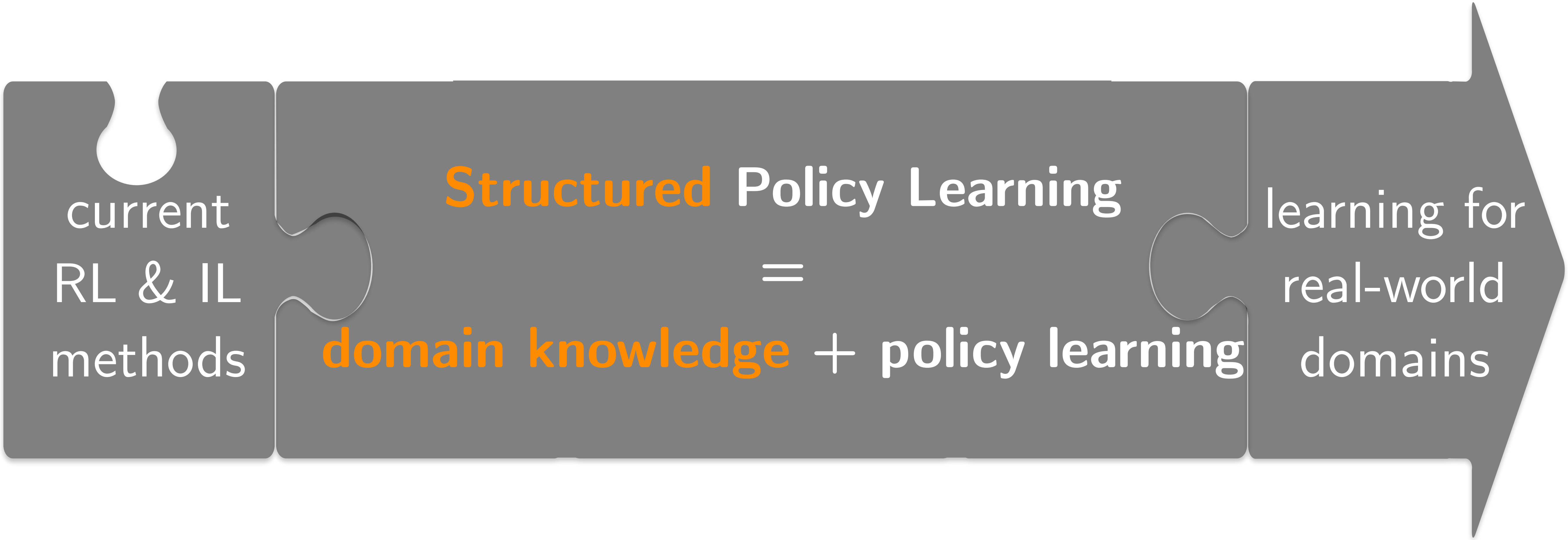


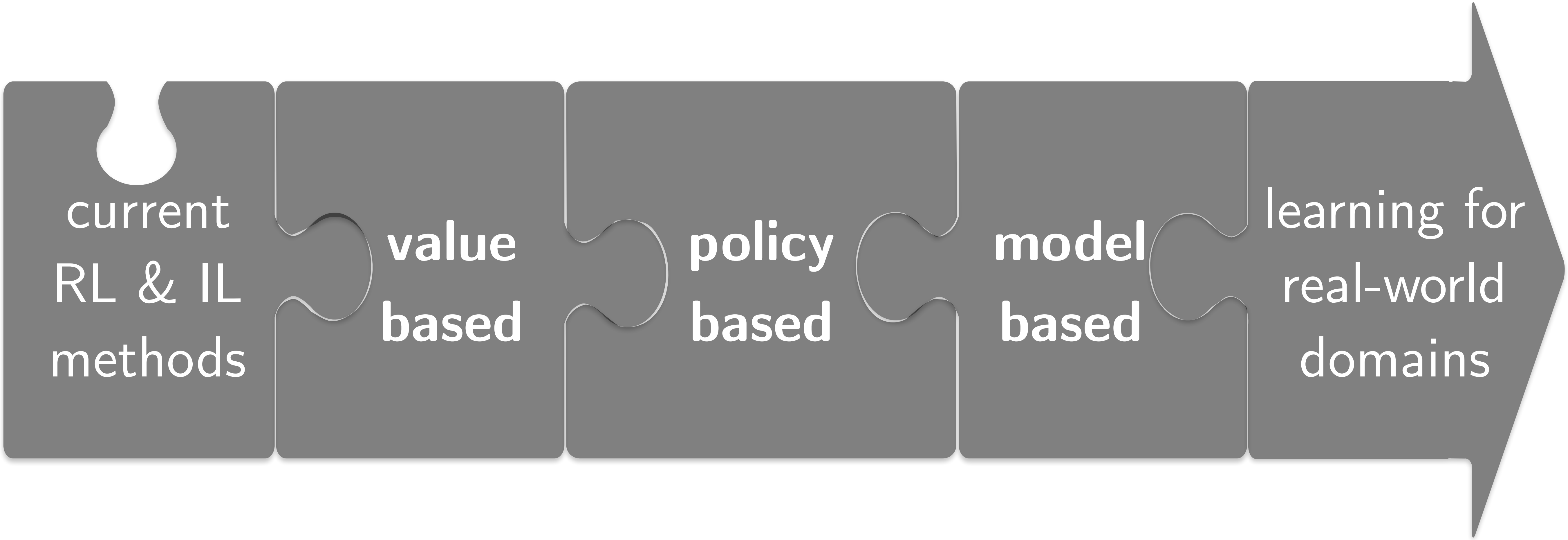
Combining latent structure with policy learning leads to better performance and data-efficiency

Data-Driven Ghosting using Deep Imitation Learning
*- **Le**CarrYueLucey - **SSAC 2017 (Best Paper Award - runner up)***

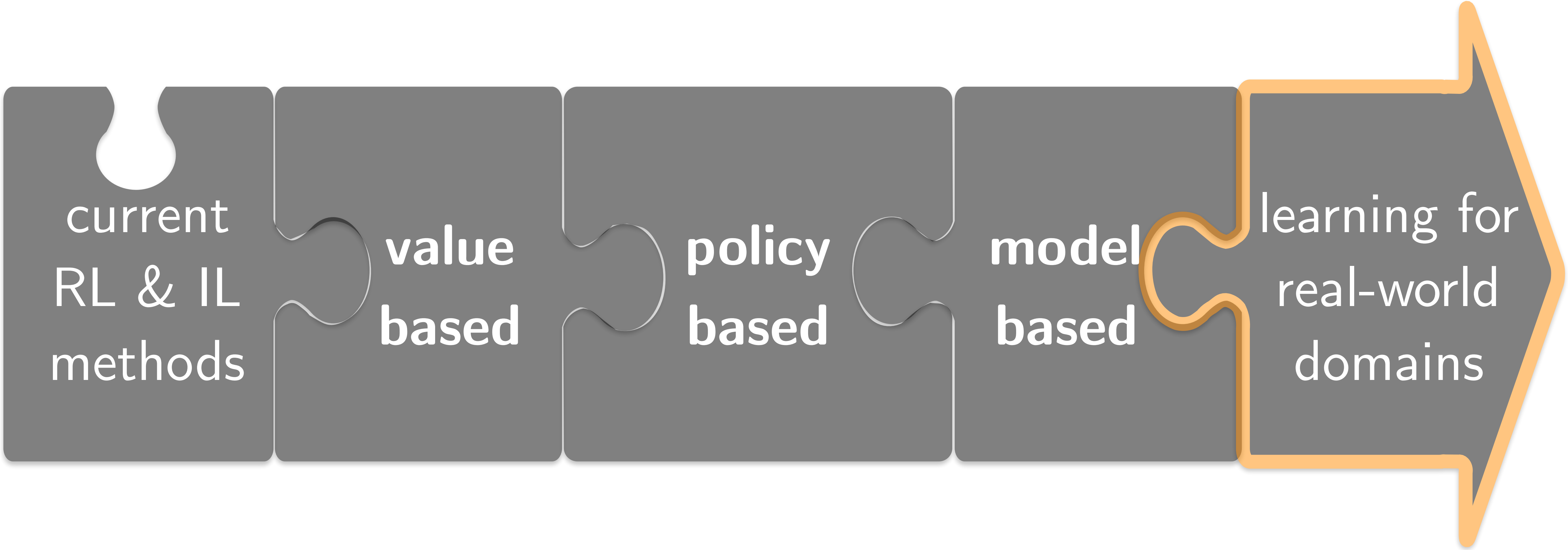
Data-Driven Ghosting
*- Carr**Le**Yue - US Patent App #15830710*


current
RL & IL
methods
Needed to close the gap:
data efficiency
realistic constraints
learning for
real-world
domains
Safe
Set
safety
smoothness
stability
verifiability
fairness


current RL & IL methods
Structured Policy Learning
=
domain knowledge + policy learning
learning for real-world domains



- Value-based: impose constraints on overall performance
- Policy-based: building structural constraints into policy class
- Model-based: exploiting partial knowledge of the model



- ✓ Generalization, unifying perspectives
- ✓ Realistic benchmarks
- ✓ Interfacing with other research areas

References

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

[2] *Batch Policy Learning under Constraints*
Hoang M. Le, Cameron Voloshin, Yisong Yue - ICML 2019

[3] *Empirical Study of Off-Policy Policy Evaluation for Reinforcement Learning*
Cameron Voloshin, **Hoang M. Le**, Nan Jiang, Yisong Yue - (under review)

[4] *A Control Lyapunov Perspective on Episodic Learning via Projection to State Stability*
Andrew J. Taylor, Victor Dorobantu, Meera Krishnamoothy, **Hoang M. Le**, Yisong Yue, Aaron Ames - CDC 2019

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

[6] *Hierarchical Imitation and Reinforcement Learning*
Hoang M. Le, Nan Jiang, Alekh Agarwal, Miro Dudík, Yisong Yue, Hal Daumé - ICML 2018

[7] *Coordinated Multi-Agent Imitation Learning*
Hoang M. Le, Yisong Yue, Peter Carr, Patrick Lucey

[8] *Data-Driven Ghosting using Deep Imitation Learning*
Hoang M. Le, Peter Carr, Yisong Yue, Patrick Lucey - SSAC 2017

[9] *Smooth Imitation Learning for Online Sequence Prediction*
Hoang M. Le, Andrew Kang, Yisong Yue, Peter Carr - ICML 2016

[10] *Learning Online Smooth Predictors for Real-time Camera Planning using Recurrent Decision Trees*
Jianhui Chen, **Hoang M. Le**, Peter Carr, Yisong Yue, James J. Little - CVPR 2016


Yisong Yue


Adam Wierman


Anima Anandkumar


Hal Daumé III


Alekh Agarwal


Miro Dudík


Nan Jiang


Peter Carr


Cameron Voloshin


Swarat Chaudhuri


Abhinav Verma


Luciana Cendon


Victor Dorobantu


Andrew Taylor


Patrick Lucey


Aaron Ames


Jim Little


Jimmy Chen


Andrew Kang