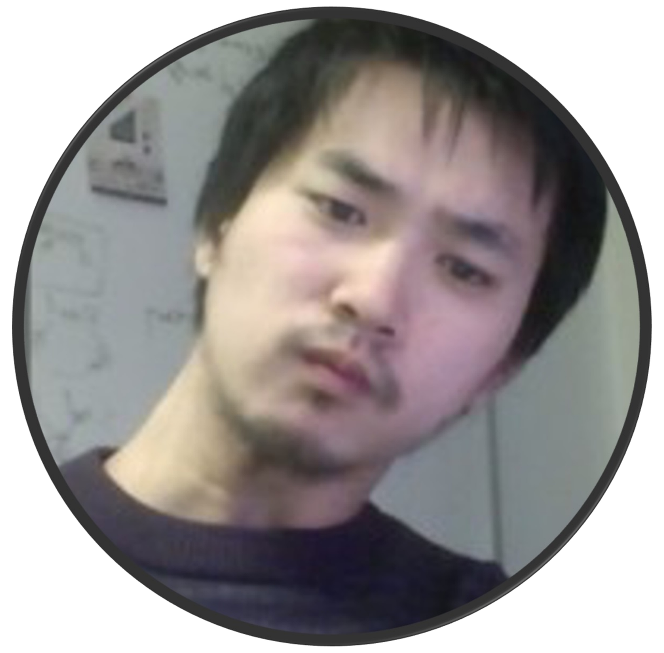
Differentiable Robotics

Boris Ivanovic, Peter Karkus | American Control Conference 2023

NVIDIA Autonomous Vehicle Research Group

Members









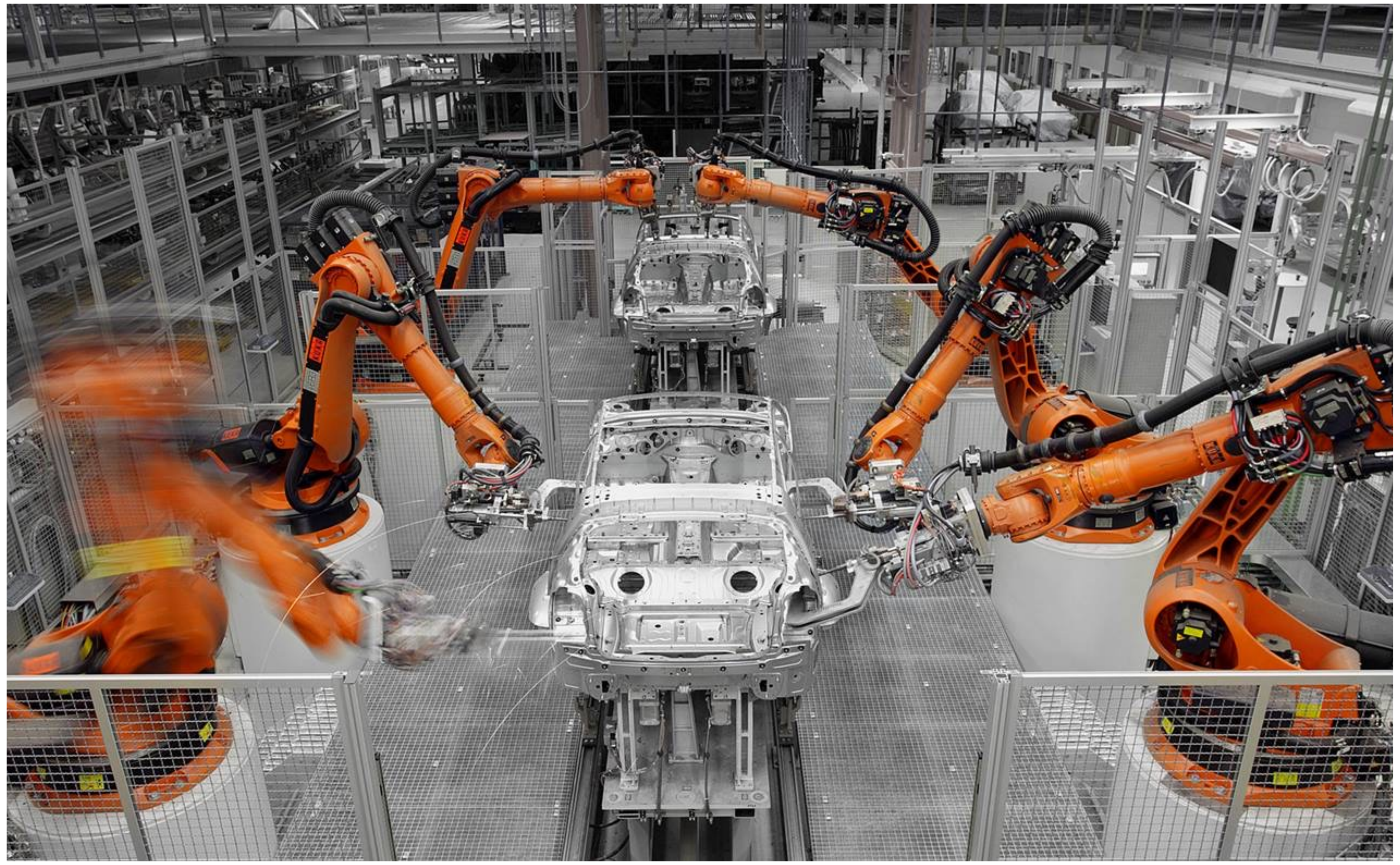


Faculty Scientists

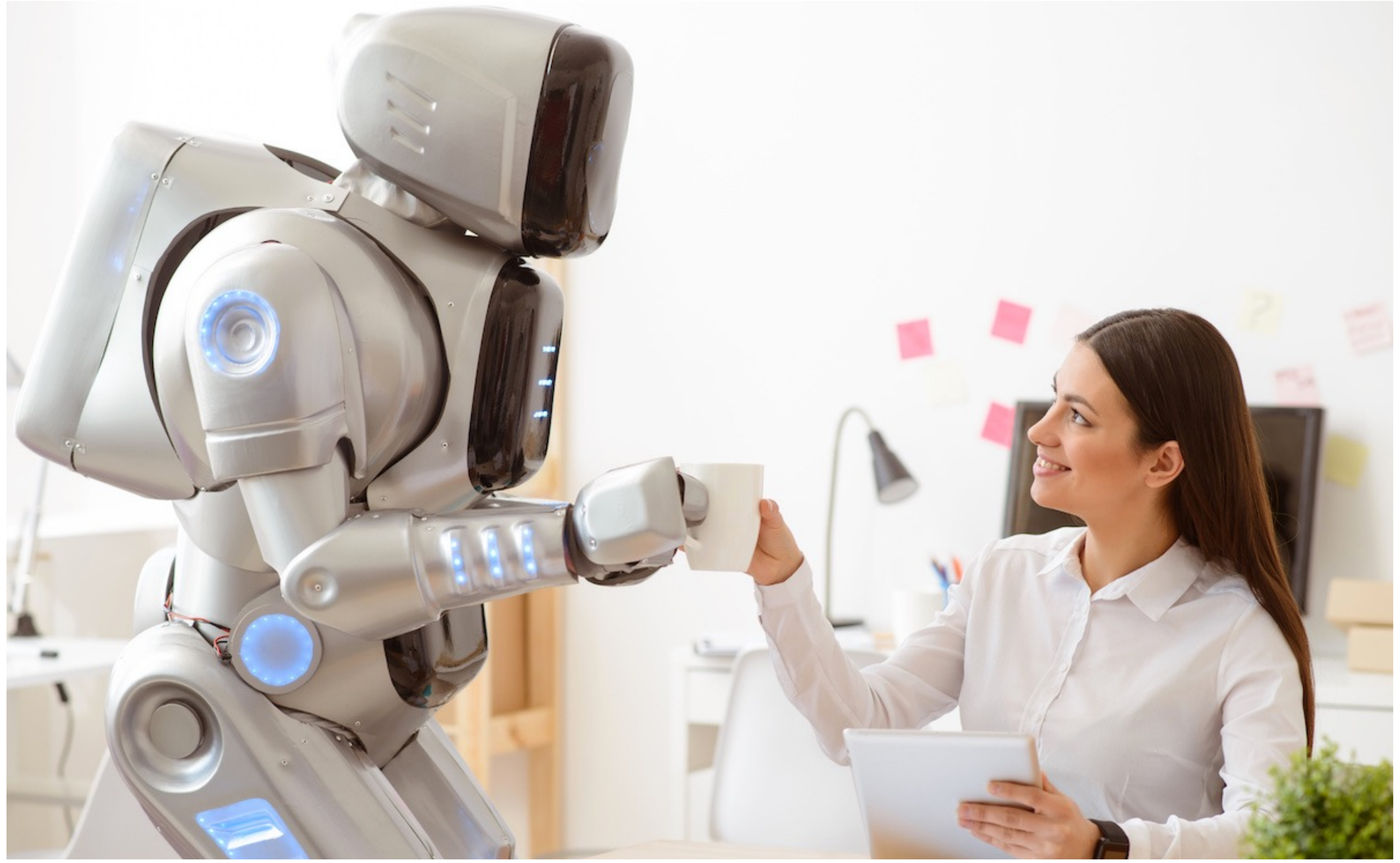

Consultant

For more information about the Autonomous Vehicle Research Group:
https://research.nvidia.com/labs/avg/

From control to human-level robot intelligence



Robots today

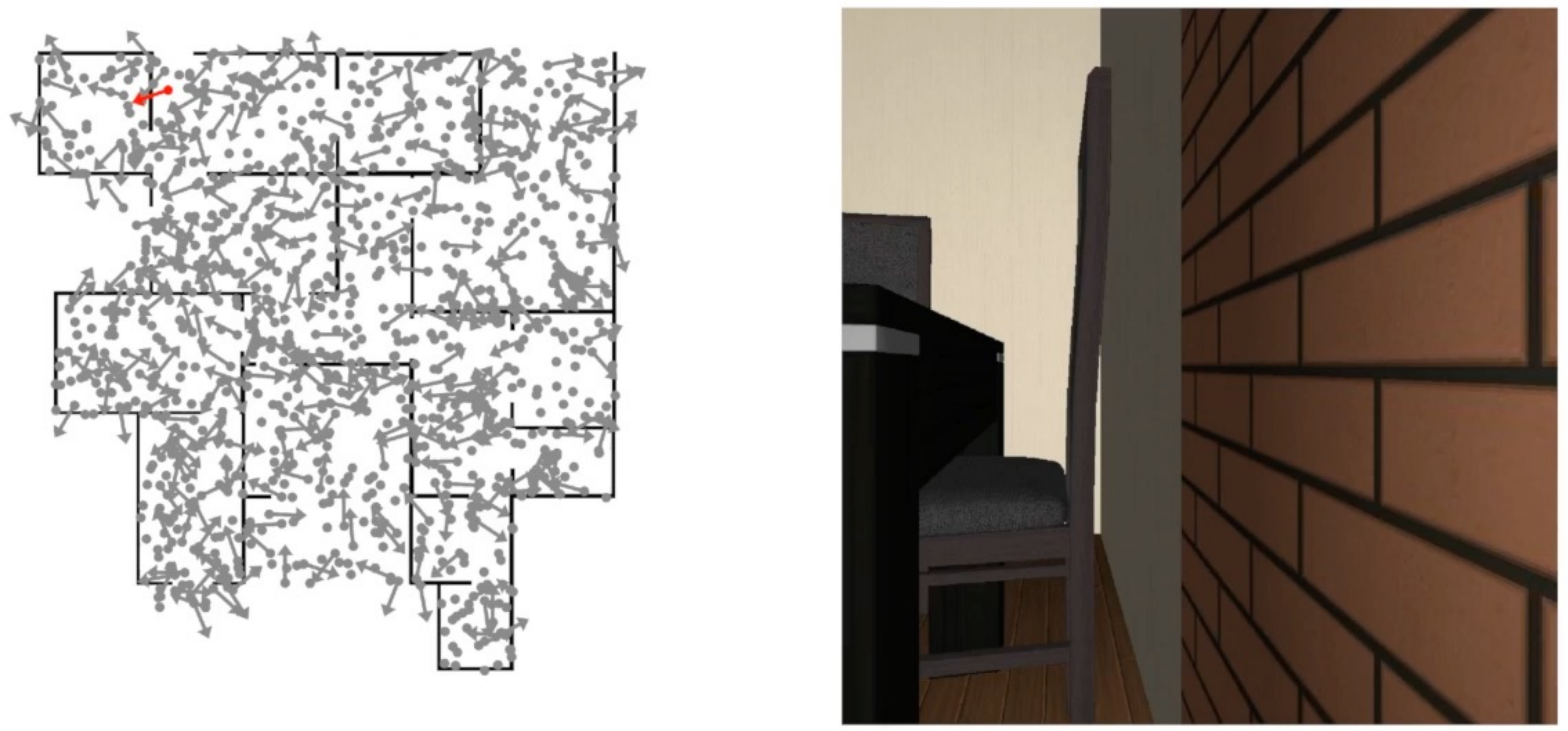


Robots tomorrow

Differentiable Algorithm Networks (DANs)

a general architecture for designing scalable and compositional robot learning systems.

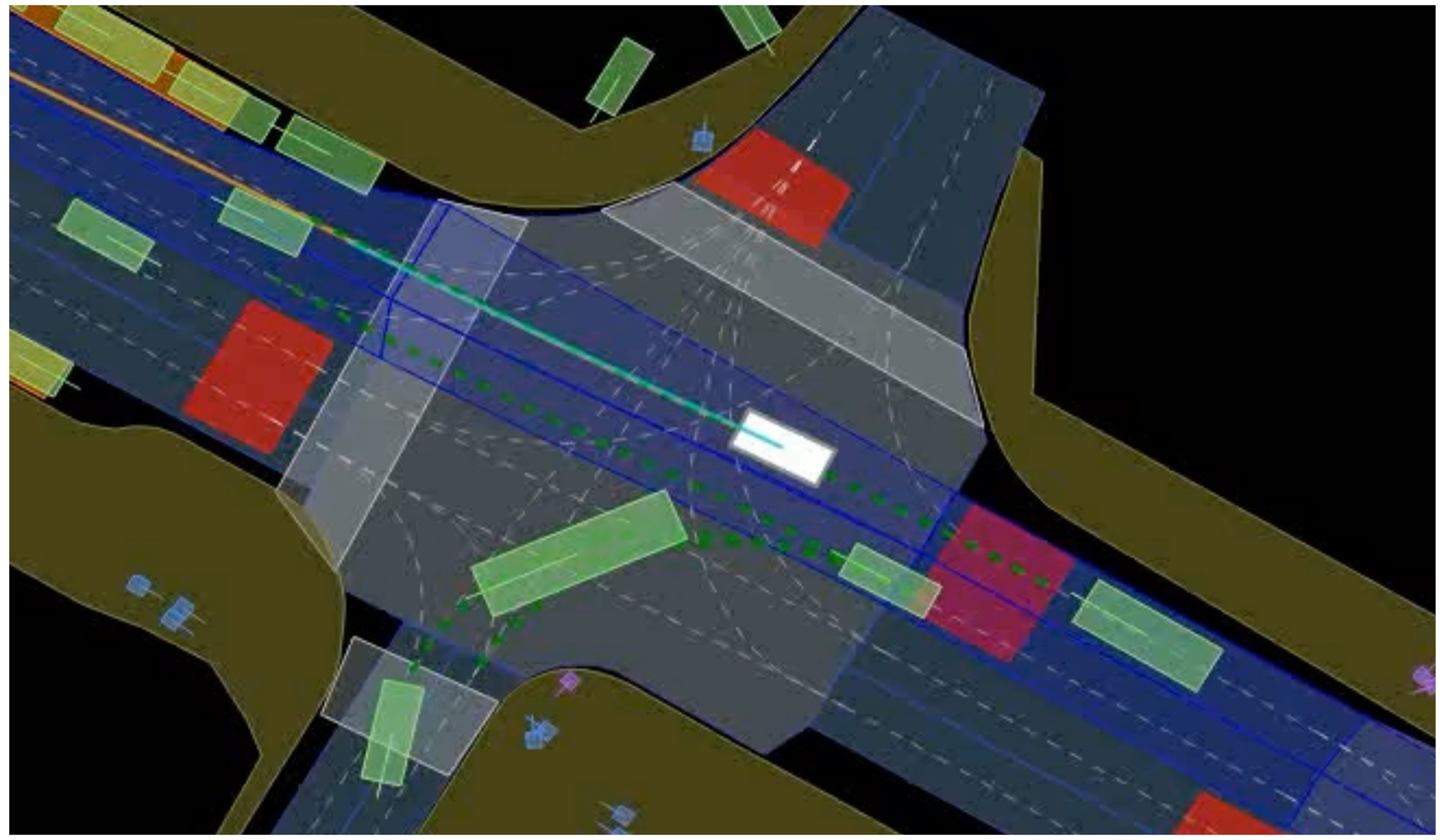
Applications



Visual localization

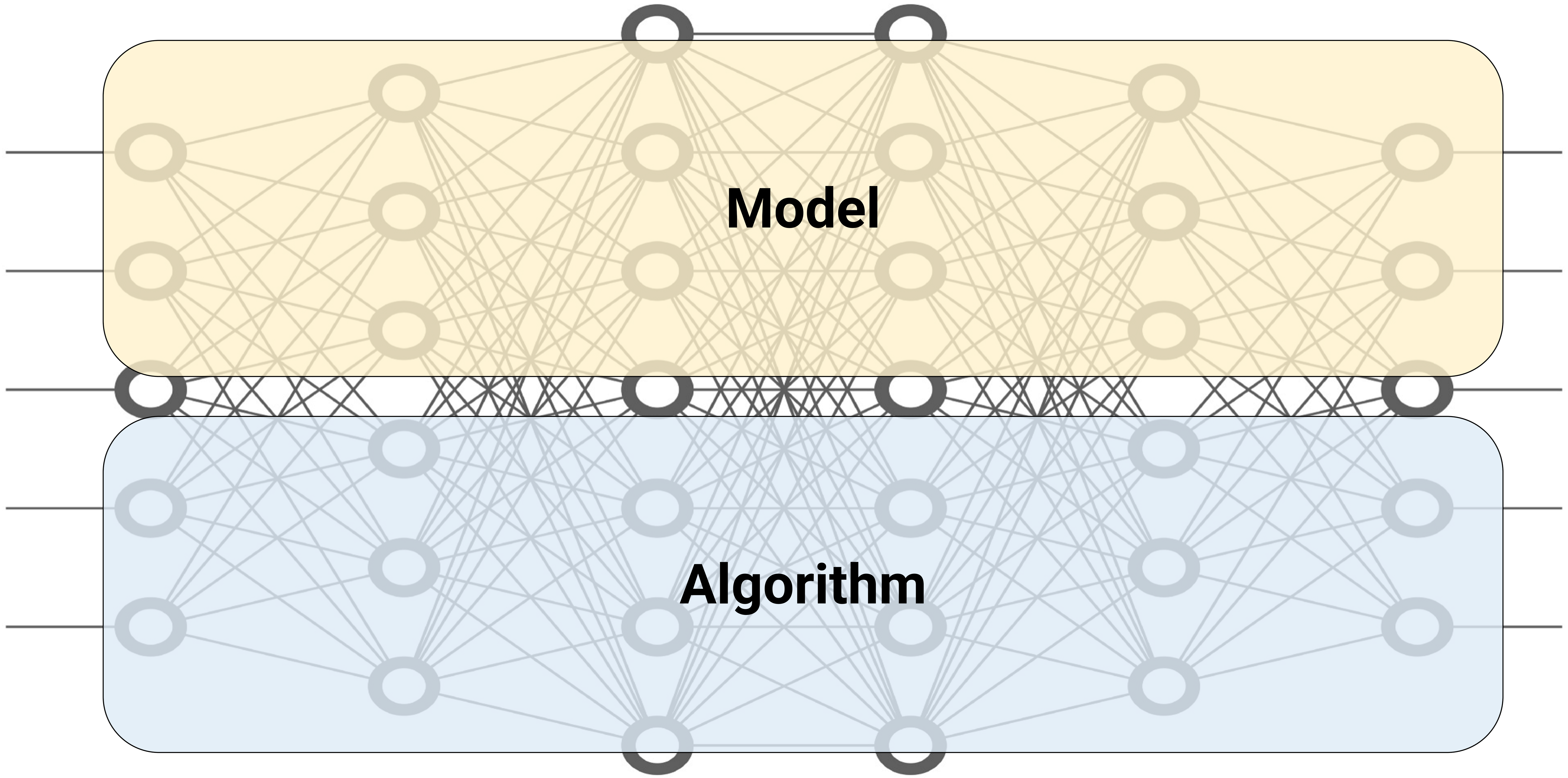
SLAM

Navigation



Autonomous driving

Idea: encode algorithms in neural networks



Differentiable Algorithm Network = Computation graph

Neural network = Computation graph

Example: value iteration

MDP:

$S, A, T(s,a,s'), R(s,a)$

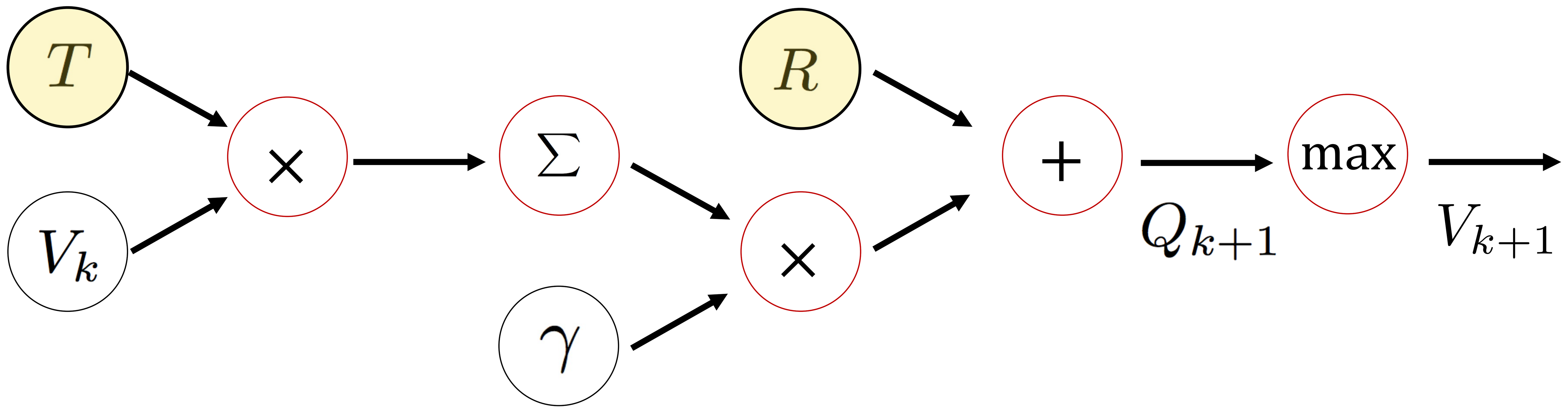
ValueIteration:

Repeat for $k = 1 : K$

$$Q_{k+1}(s,a) = \boxed{R(s,a)} + \gamma \sum_{s' \in S} \boxed{T(s,a,s')} V_k(s')$$

$$V_k(s) = \max_a Q_k(s,a)$$

$$a_t = \operatorname*{argmax}_a Q_K(s_t, a)$$



Example: value iteration

MDP:
$S, A, T(s,a,s'), R(s,a)$

ValueIteration:

Repeat for $k = 1 : K$

$$Q_{k+1}(s,a) = R(s,a) + \gamma \sum_{s' \in S} T(s,a,s') V_k(s')$$

$$V_k(s) = \max_a Q_k(s,a)$$

$$a_t = \operatorname*{argmax}_a Q_K(s_t, a)$$

Example: value iteration

MDP:
$S, A, T(s,a,s'), R(s,a)$

ValueIteration:

Repeat for $k = 1 : K$

$$Q_{k+1}(s,a) = R(s,a) + \gamma \sum_{s' \in S} T(s,a,s') V_k(s')$$

$$V_k(s) = \max_a Q_k(s,a)$$

$$a_t = \operatorname*{argmax}_a Q_K(s_t, a)$$

Example: value iteration

MDP:
$S, A, T(s,a,s'), R(s,a)$

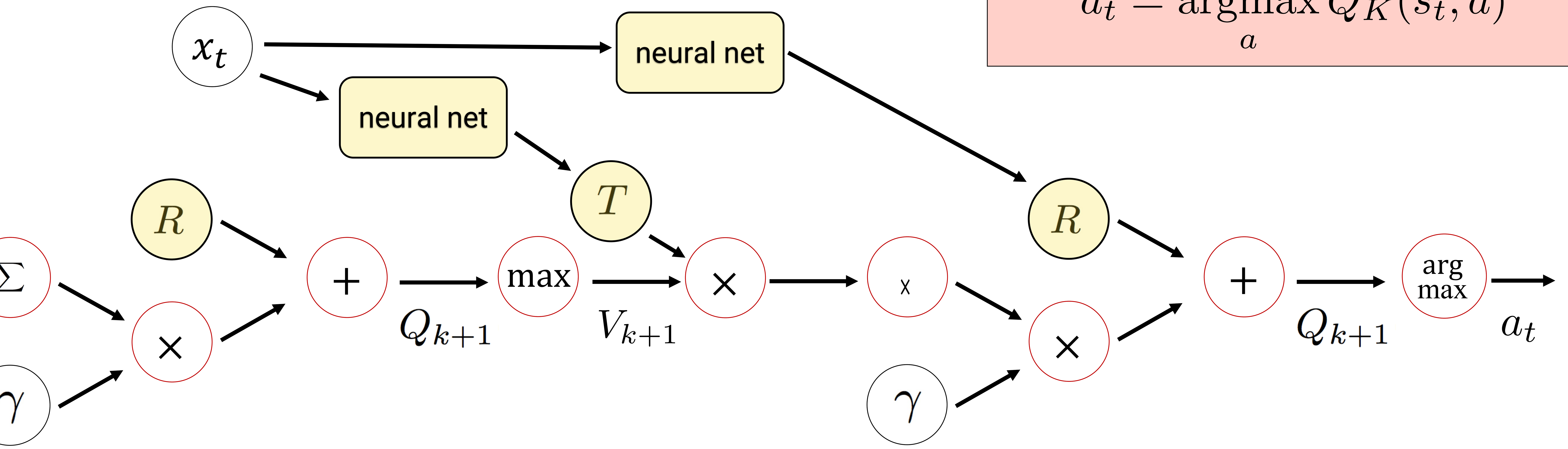
ValueIteration:

Repeat for $k = 1 : K$

$$Q_{k+1}(s,a) = R(s,a) + \gamma \sum_{s' \in S} T(s,a,s') V_k(s')$$

$$V_k(s) = \max_a Q_k(s,a)$$

$$a_t = \operatorname*{argmax}_a Q_K(s_t, a)$$



Technical challenges

Non-differentiable operations

- Continuous relaxation
- Sampling
  - reparameterization trick
  - Gumbel-softmax
  - straight-through estimation
  - importance sampling
- Implicit gradients (e.g., ODEnet)
- Score-function estimation
  (stochastic computation graphs)

Large computation graph

- Efficient parameterization
- Scale algorithm at test time
- Implicit gradients

State Estimation

Kalman filter
- *Haarnoja et al. 2016*
- *Kloss et al. 2018*

Histogram filter
- *Jonschkowski et al. 2016*

Particle filter
- *Jonschkowski et al. 2018*
- ***Karkus et al. 2018***
- *Wen et al. 2020*

Mapping
- *Gupta et al. 2017*
- ***Karkus et al. 2020***

SLAM
- *Jatavallabhula et al. 2019*
- ***Karkus et al. 2021***

Planning

Value Iteration
- *Tamar et al. 2016*
- *Shankar et al. 2016*
- *Gupta et al. 2017*
- *Lee et al. 2018*

MCTS
- *Guez et al. 2018*

QMDP planner
- ***Karkus et al. 2017***

Breadth-first search
- *Oh et al. 2017*
- *Fraquhar et al. 2017*

A* search
- *Yonetani et al. 2020*

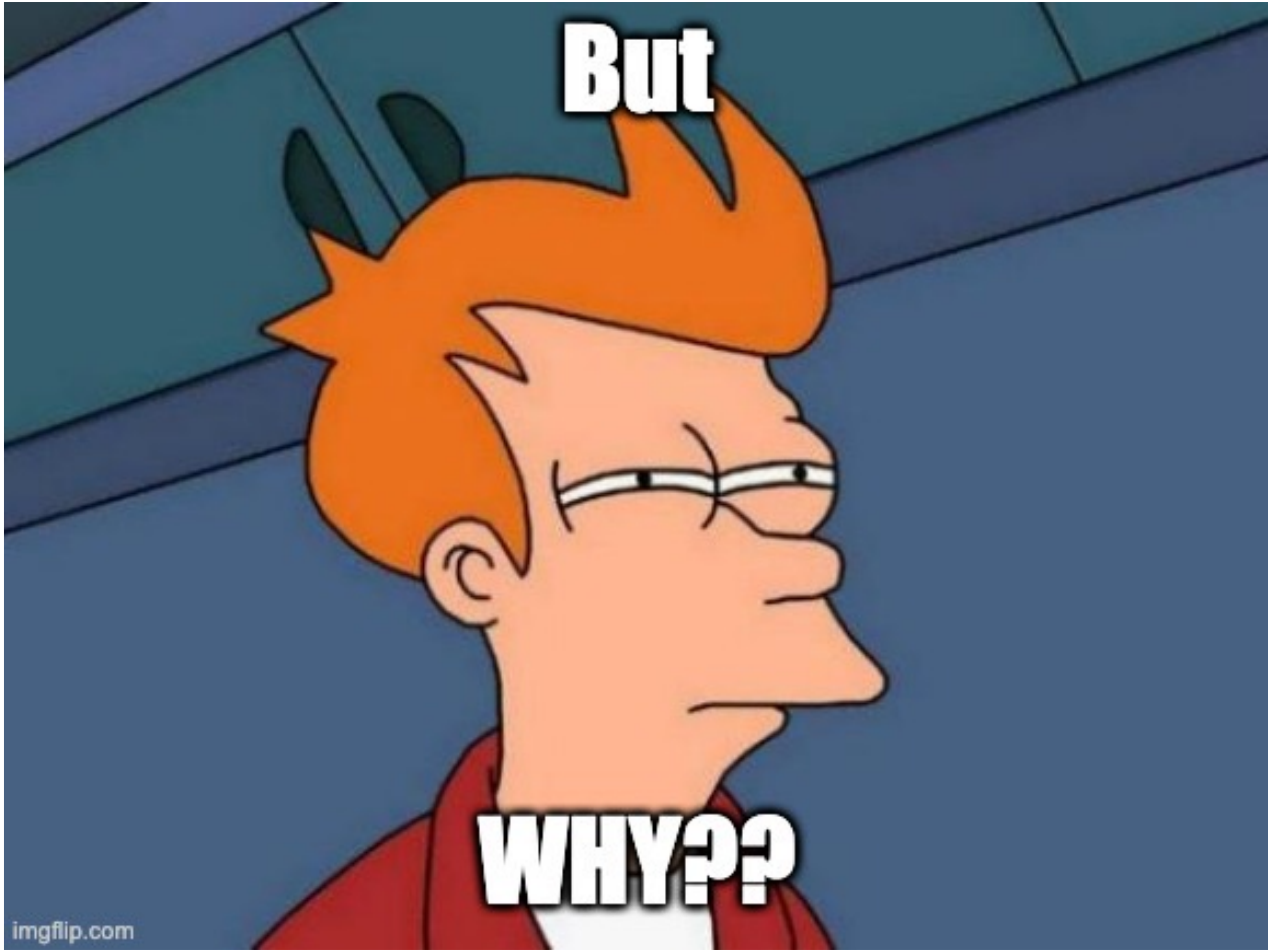
Control

Model-predictive control
- *Amos et al. 2018*
- *East et al. 2020*

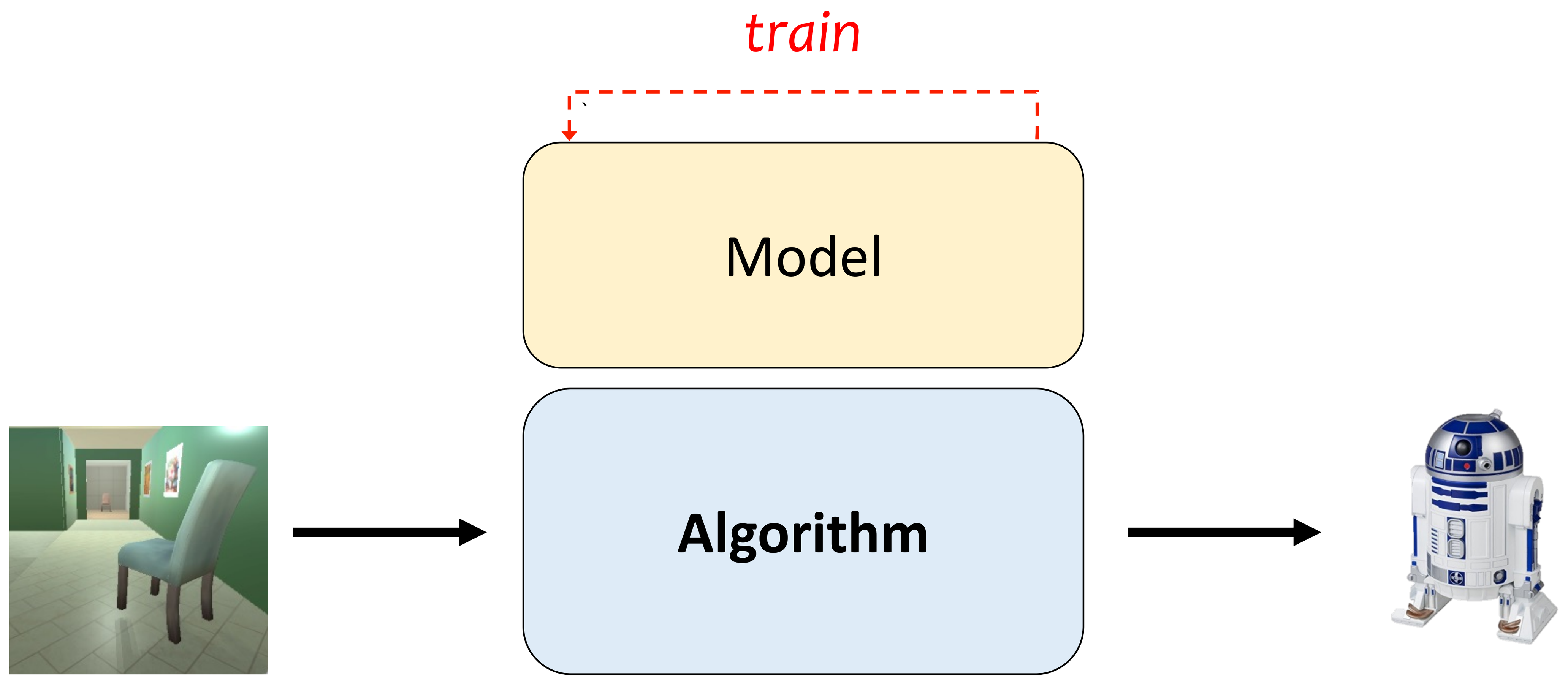
Path-integral optimal control
- *Okada et al. 2017*

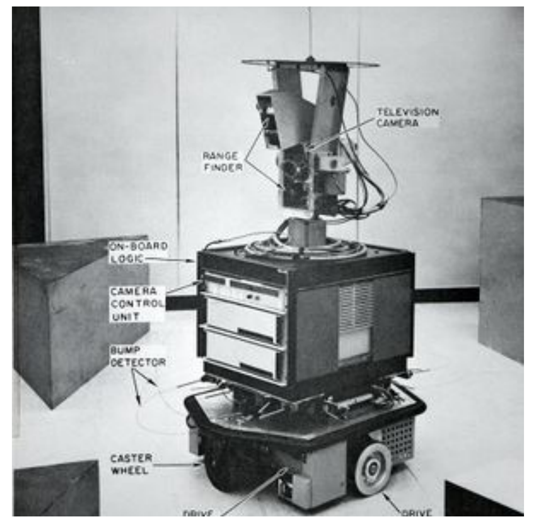
ODE solver
- *Chen et al. 2018*
- *Zhong et al. 2020*

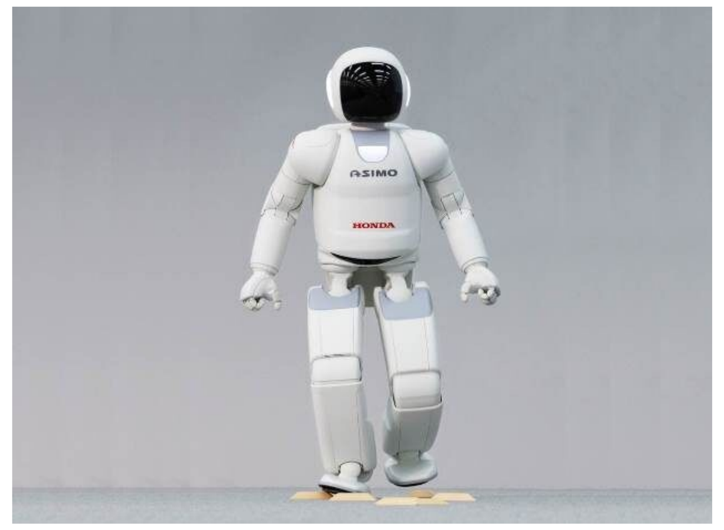
Convex optimization
- *Amos et al. 2017*
- *Agrawal et al. 2018*


But
WHY??
imgflip.com

Model-based

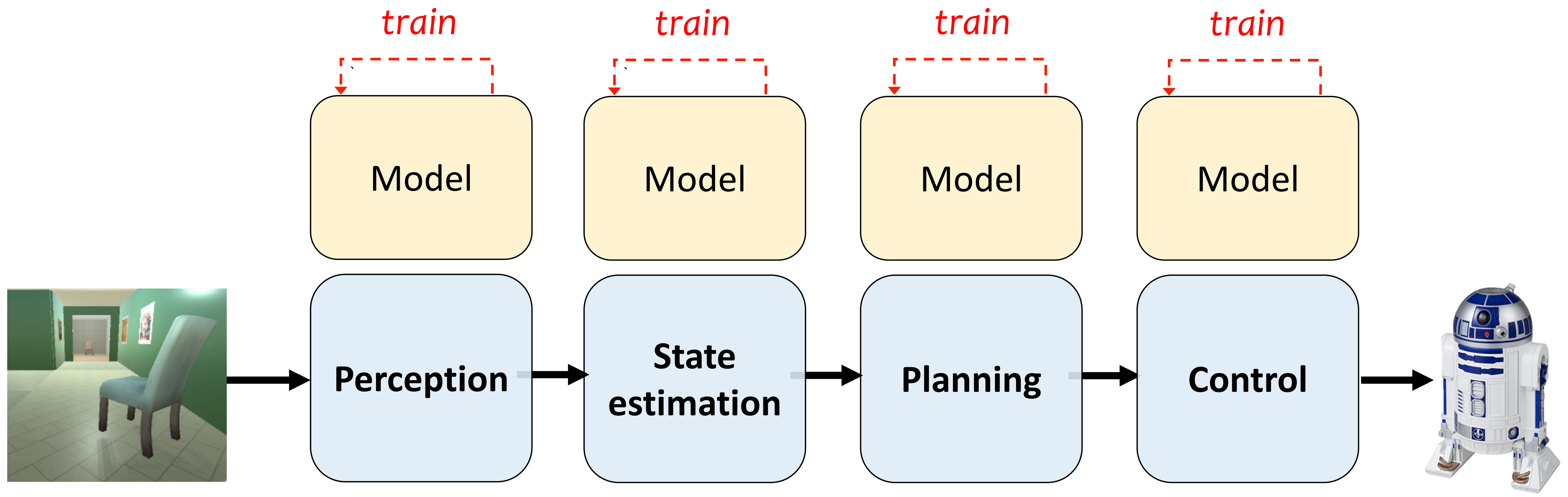




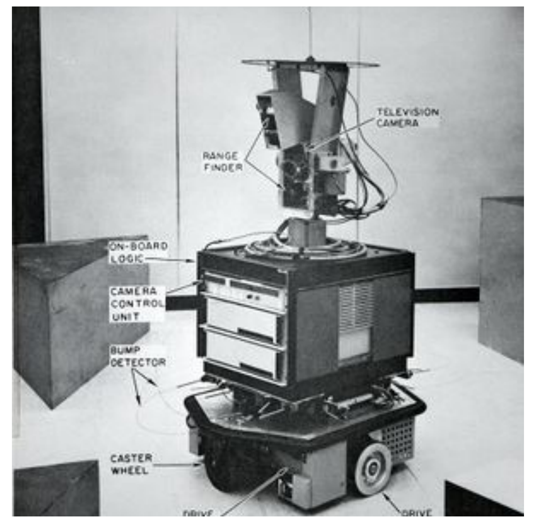


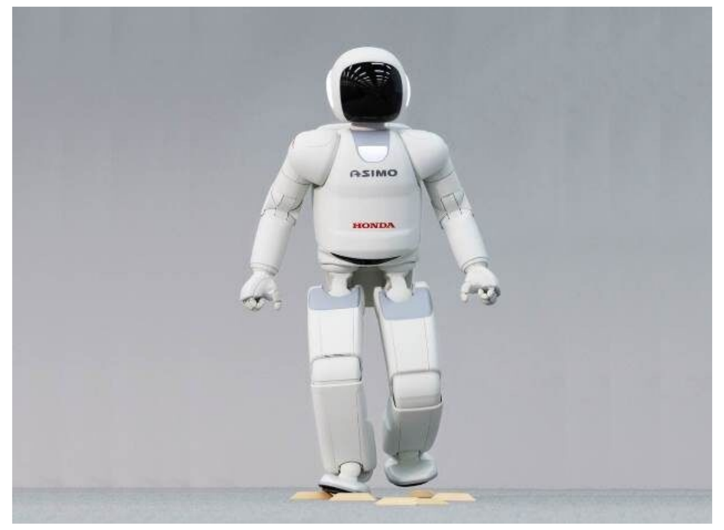
Model-free

Model-based

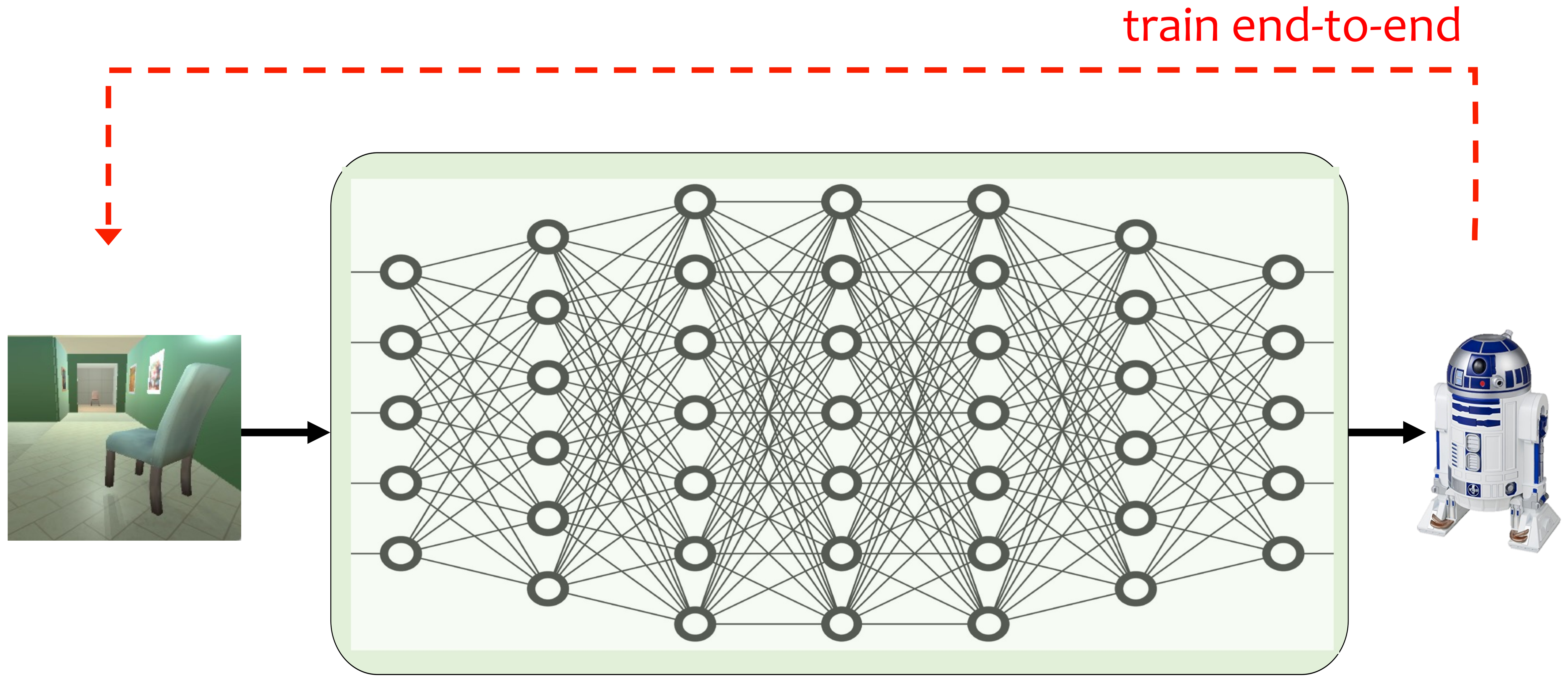


Composable and *interpretable* but
Brittle because of unavoidable approximations.

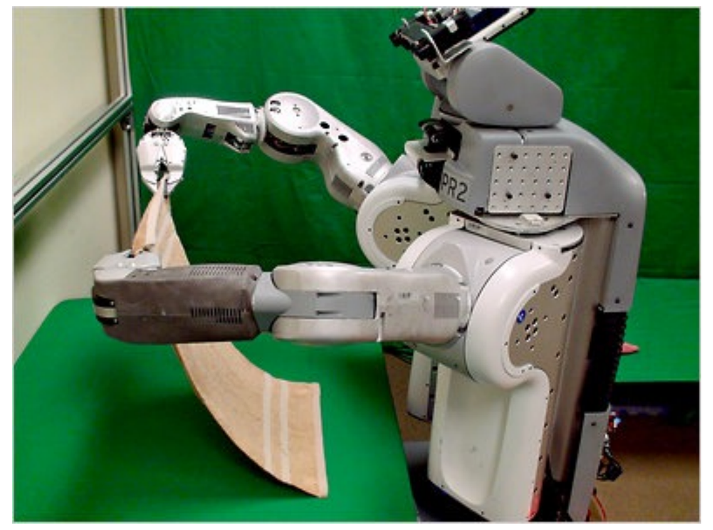
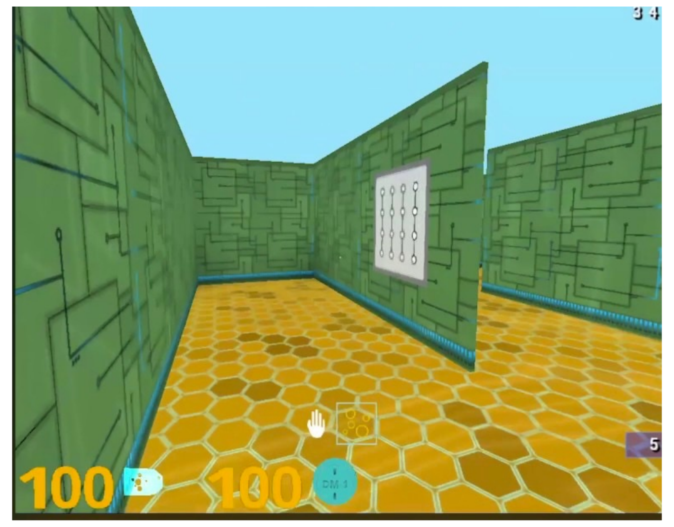
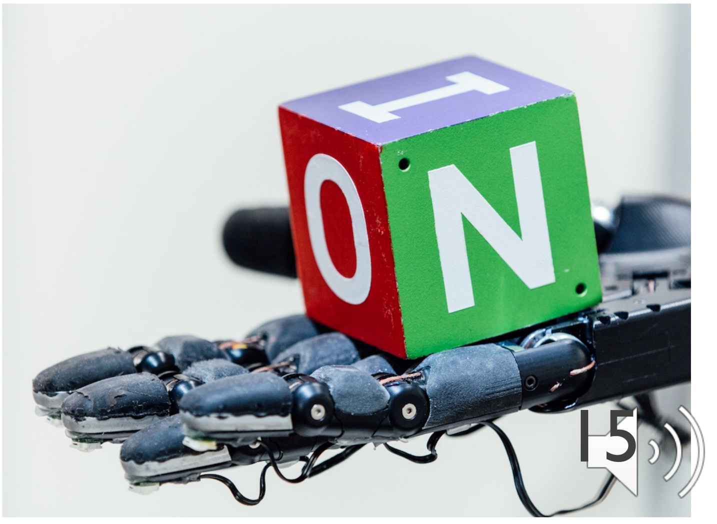


Model-free



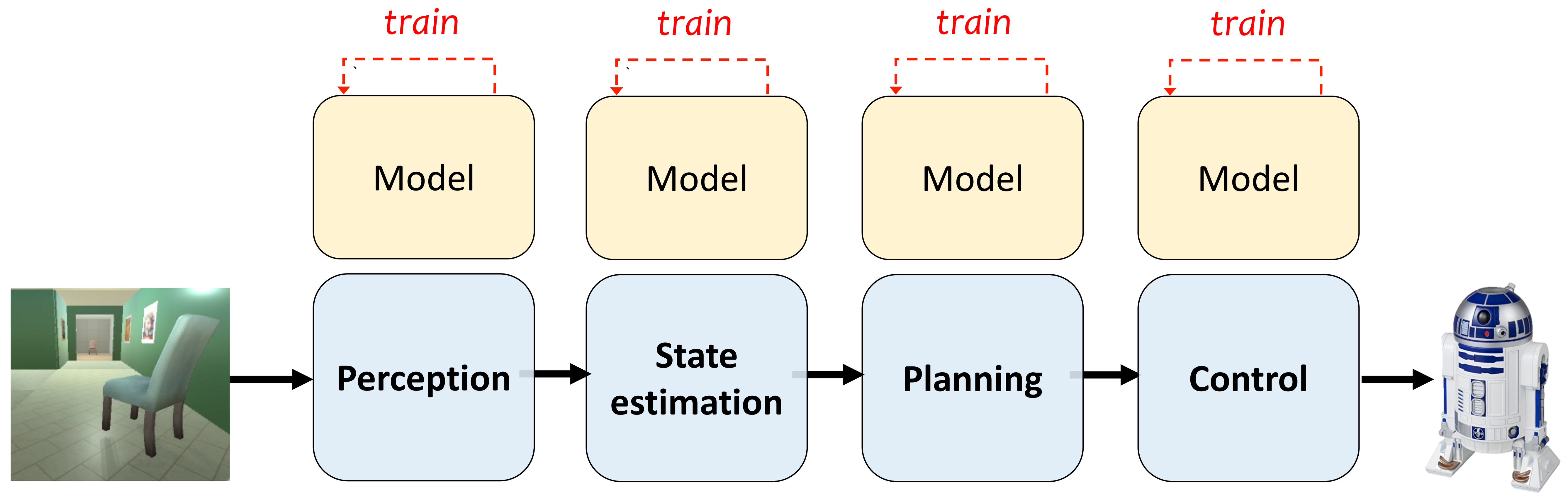
End-to-end learnable but
Lacks prior for compositional generalization, and
interpretability for safety guarantees

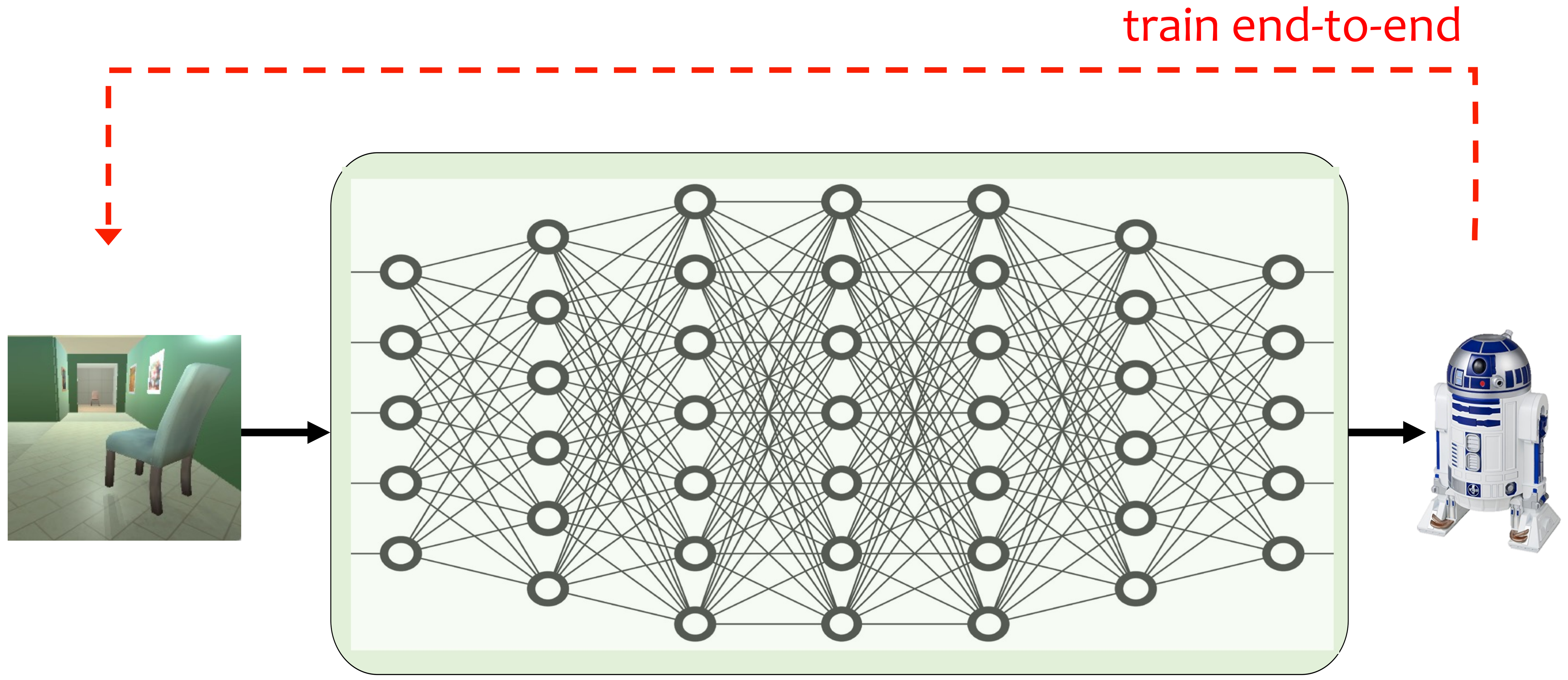





Modular / Model-based

+

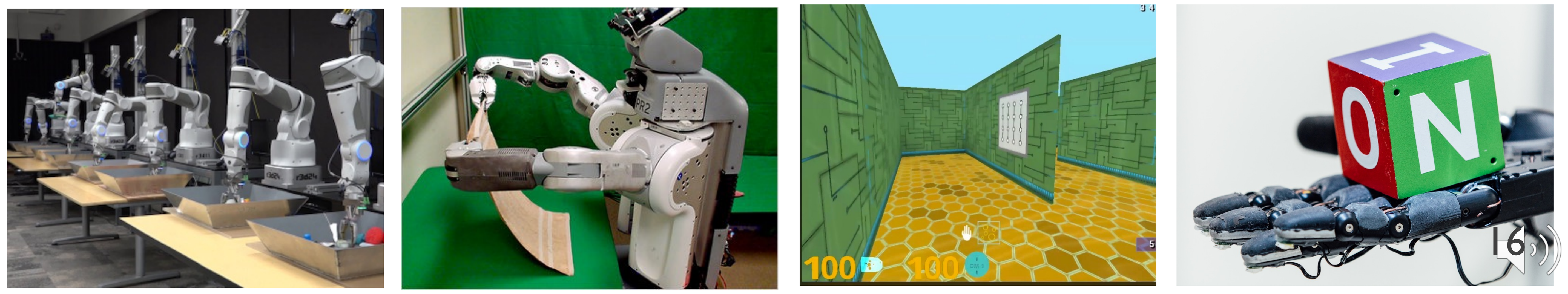
End-to-end / Model-free



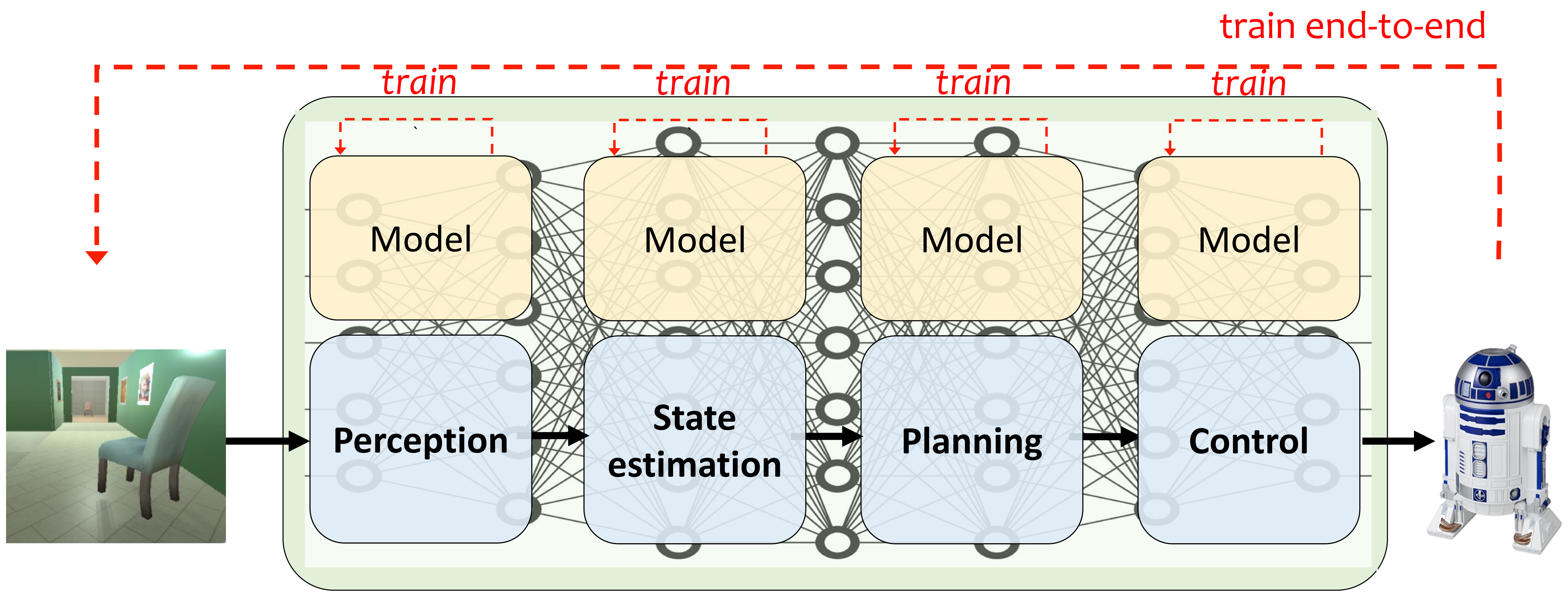


Composable but
Brittle because of unavoidable approximations

End-to-end learnable but
Lacks prior for compositional generalization and
interpretability for safety guarantees



Modular + End-to-end

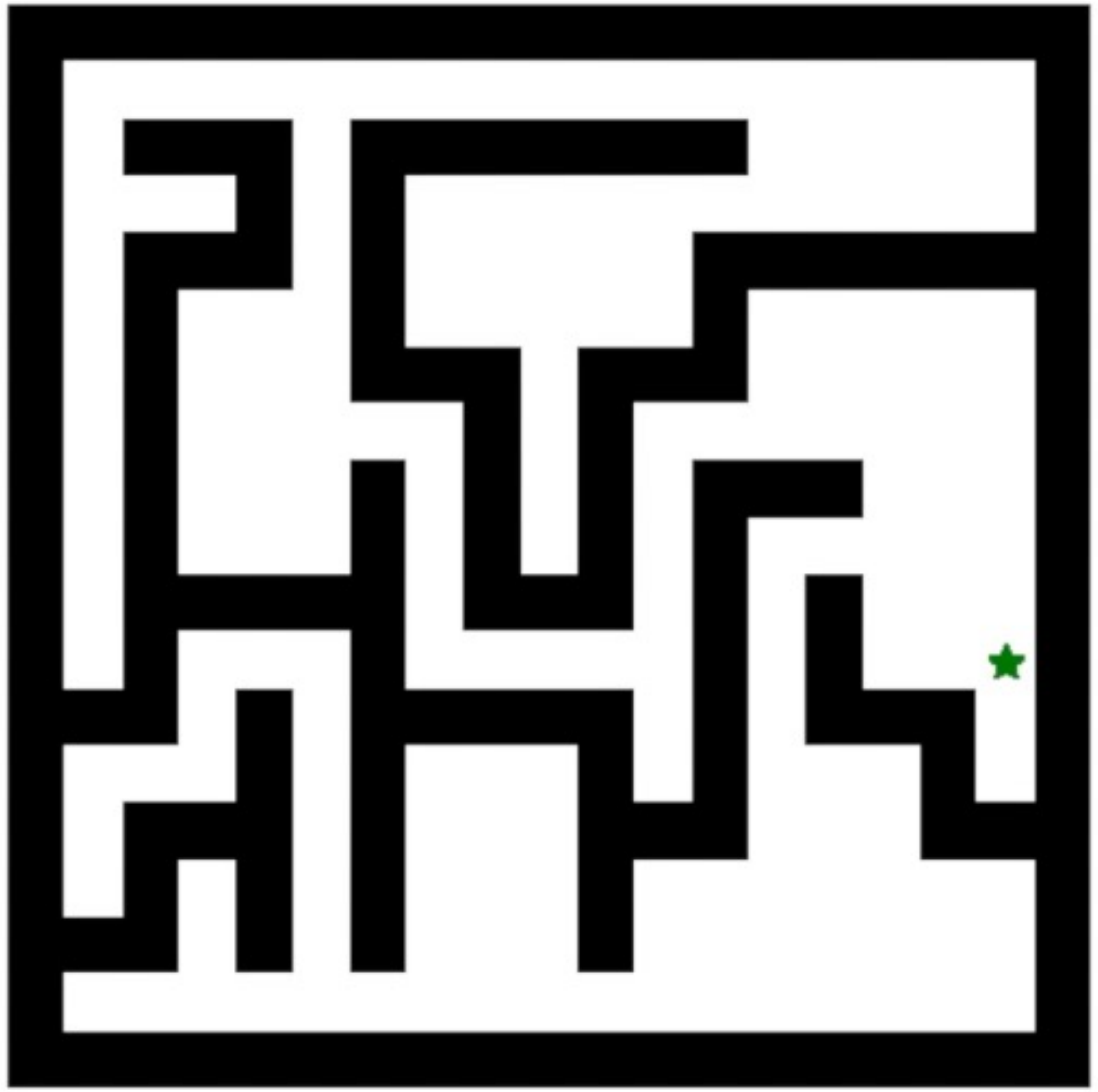


Differentiable Algorithm Network (DAN)

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019

Case study: visual navigation

Map & Goal



Observation

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019

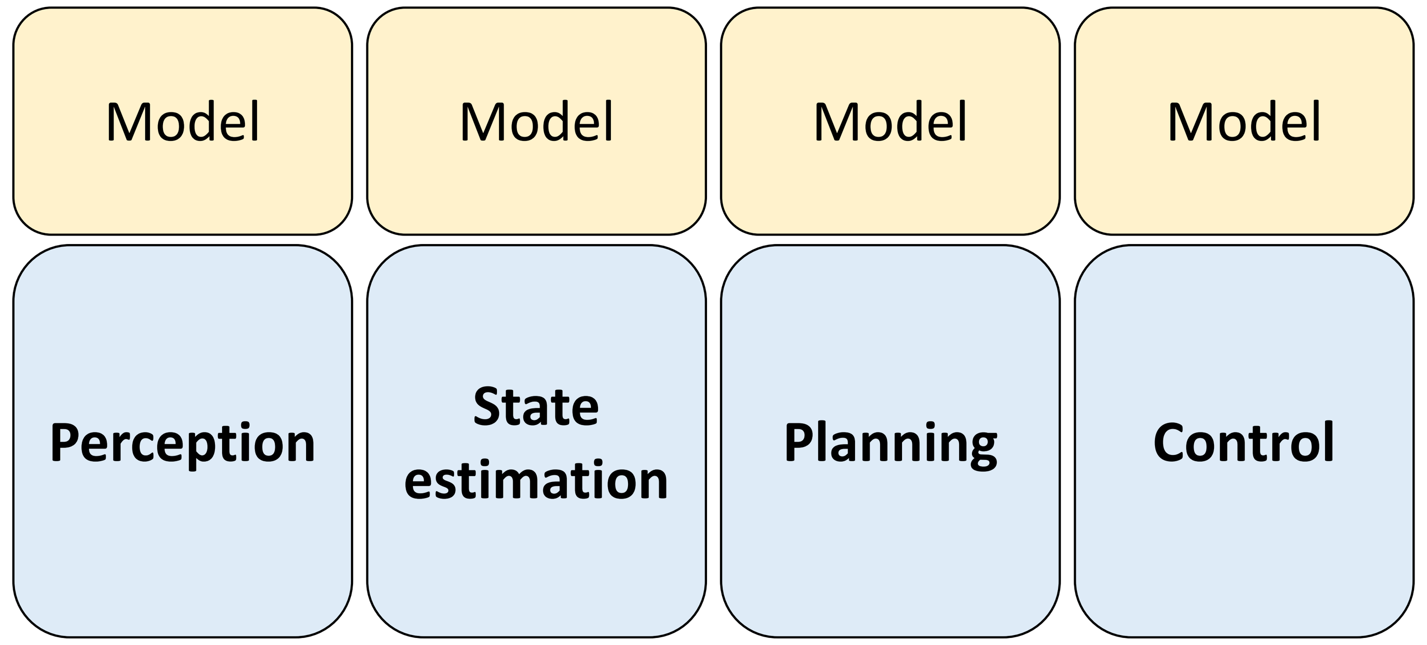
Case study: visual navigation

Train

10k expert demos

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019

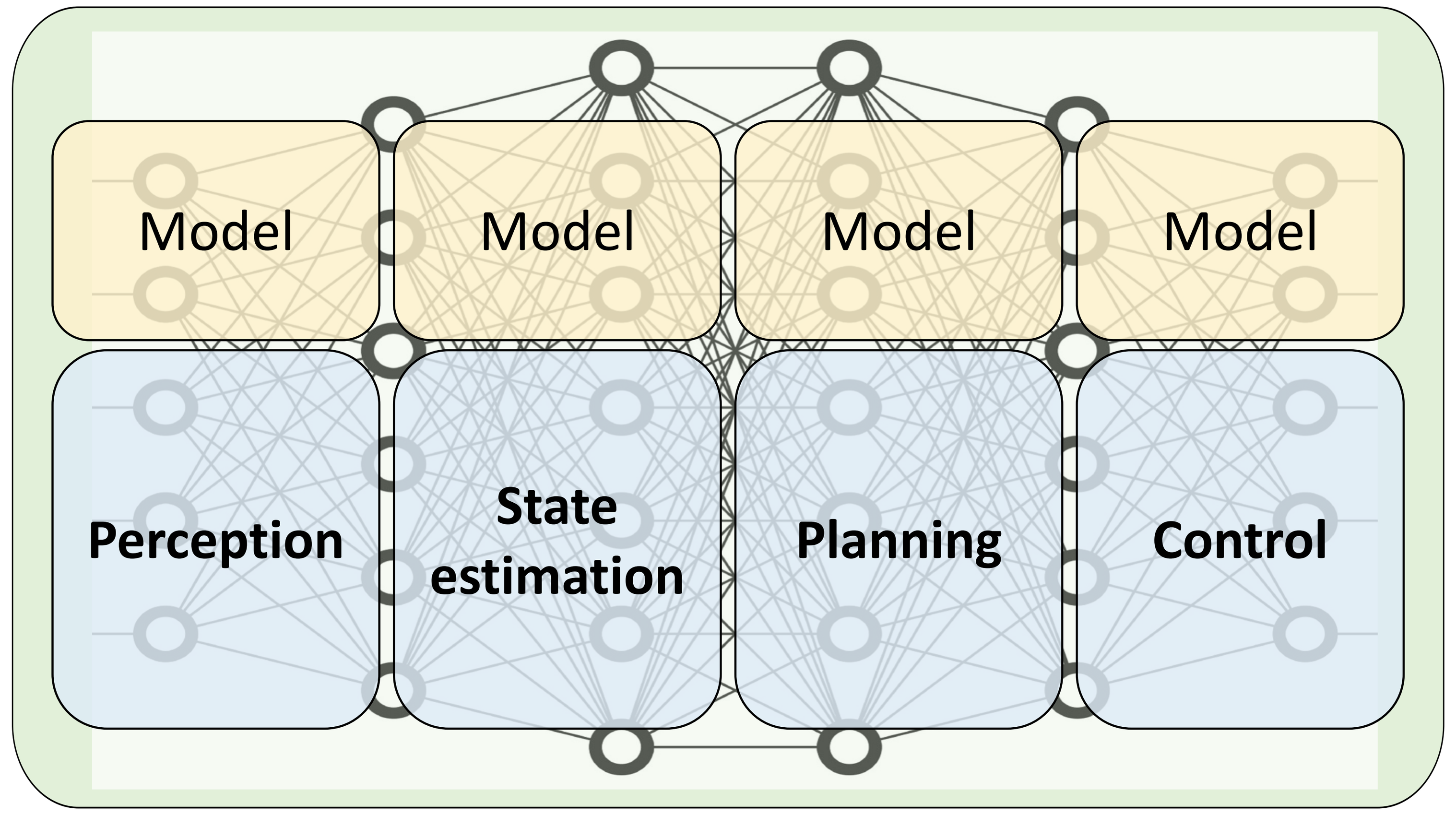
Modular



76.6%

success rate

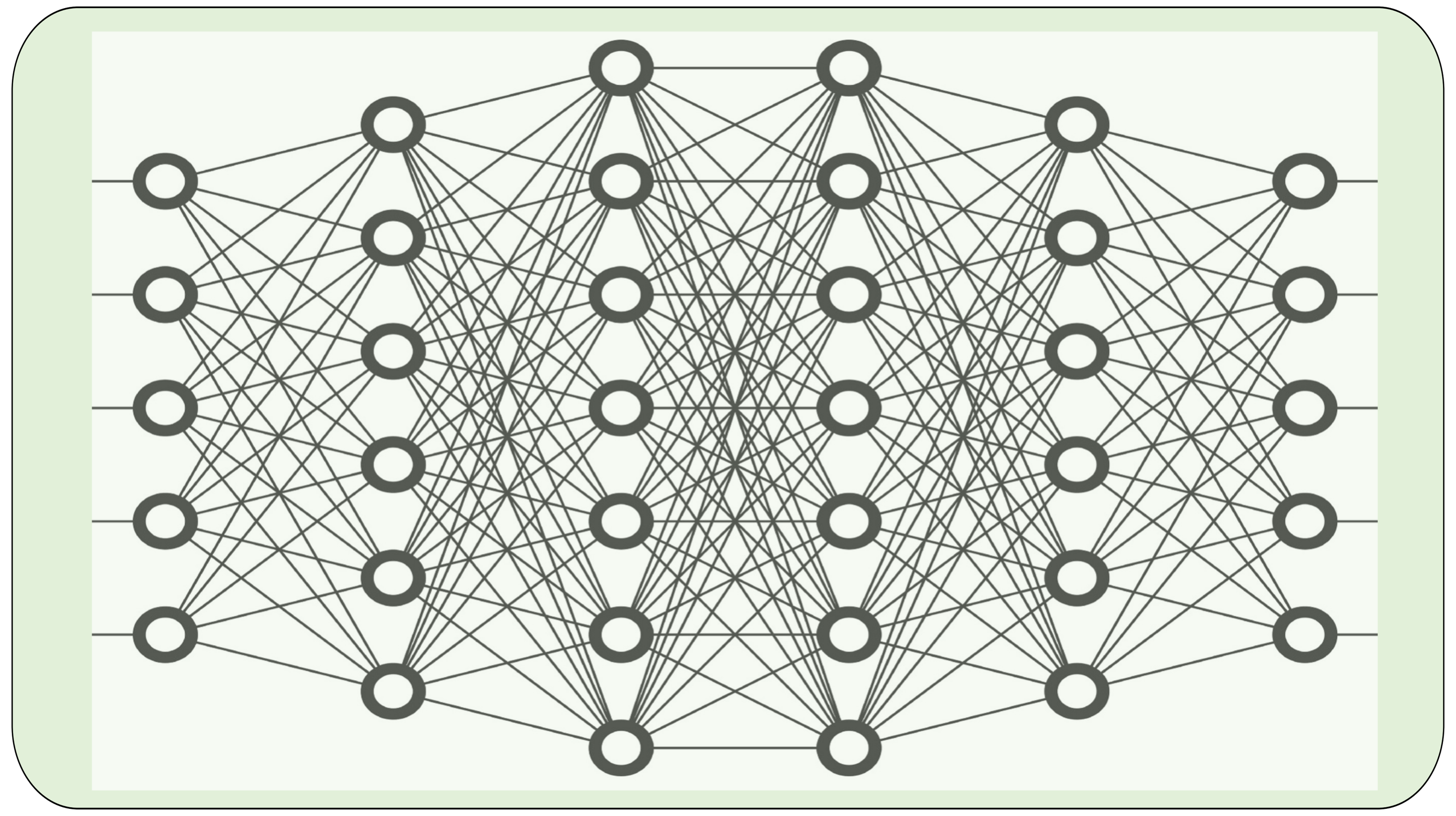
DAN (ours)



99.8%

success rate

End-to-end



38.4%

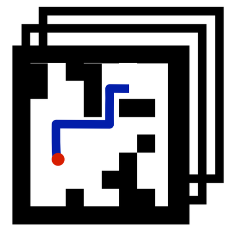
success rate

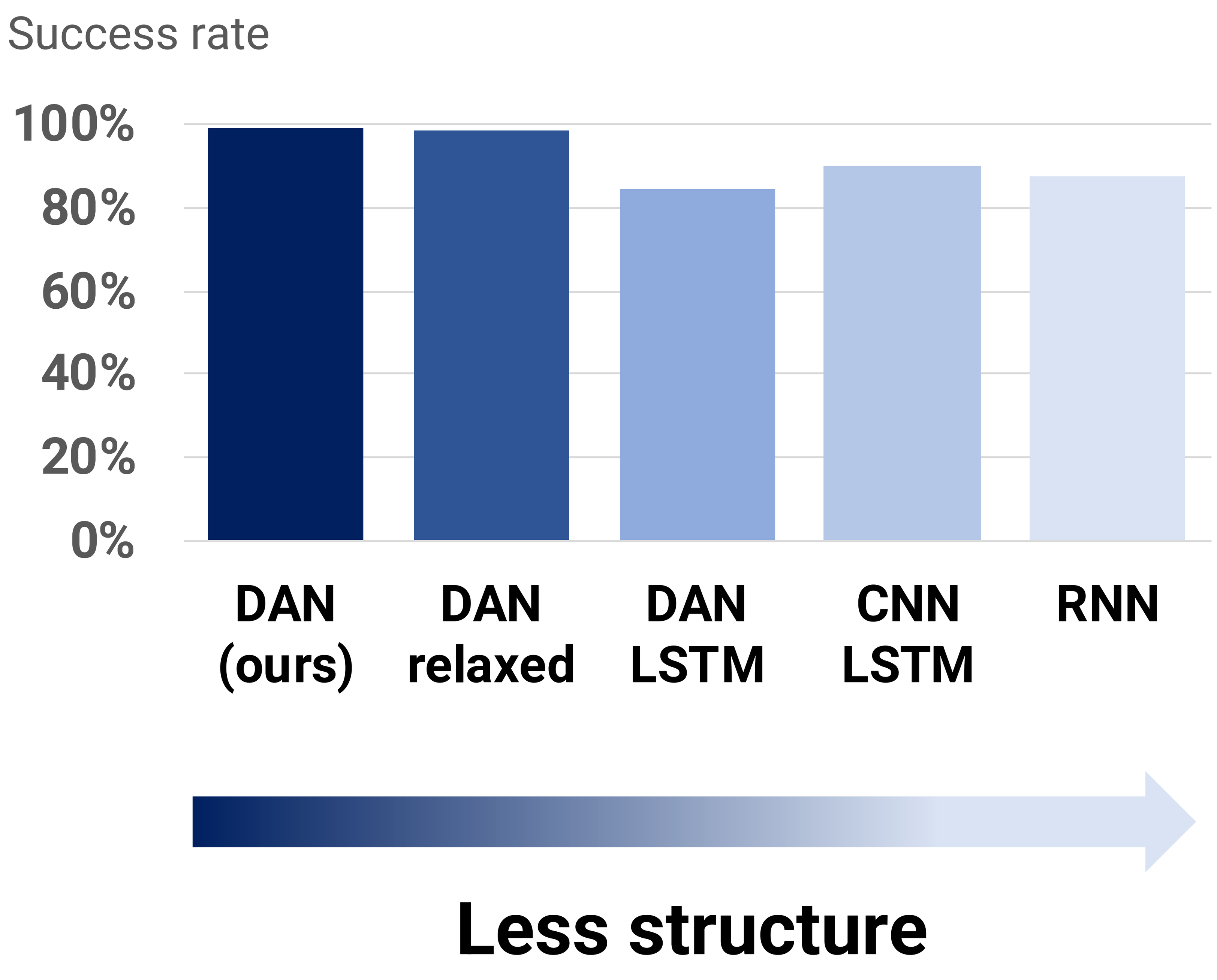
Structured prior

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019

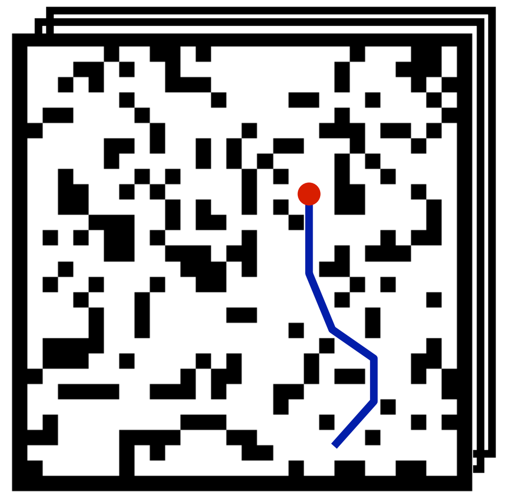
Structure priors help to generalize

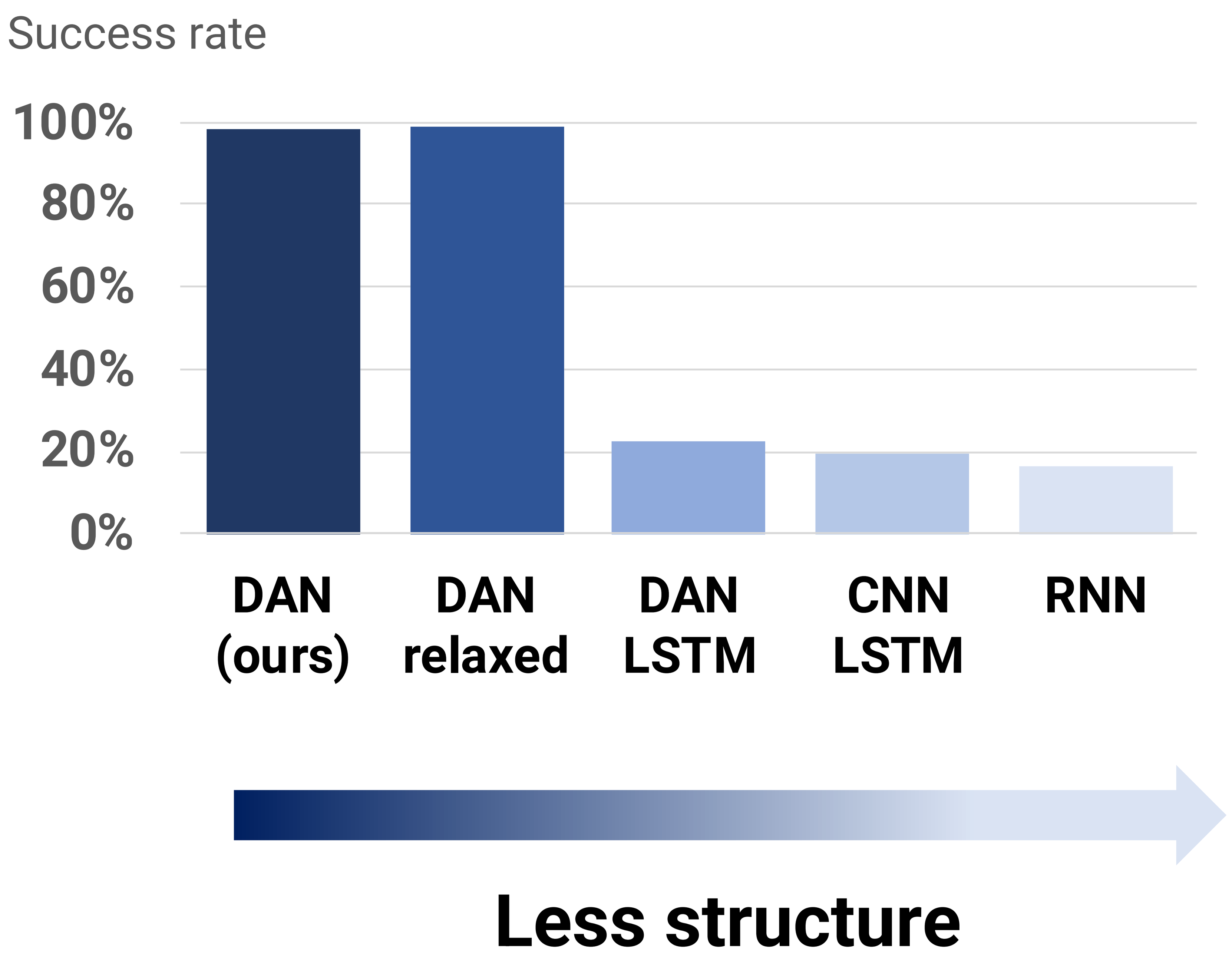
10x10



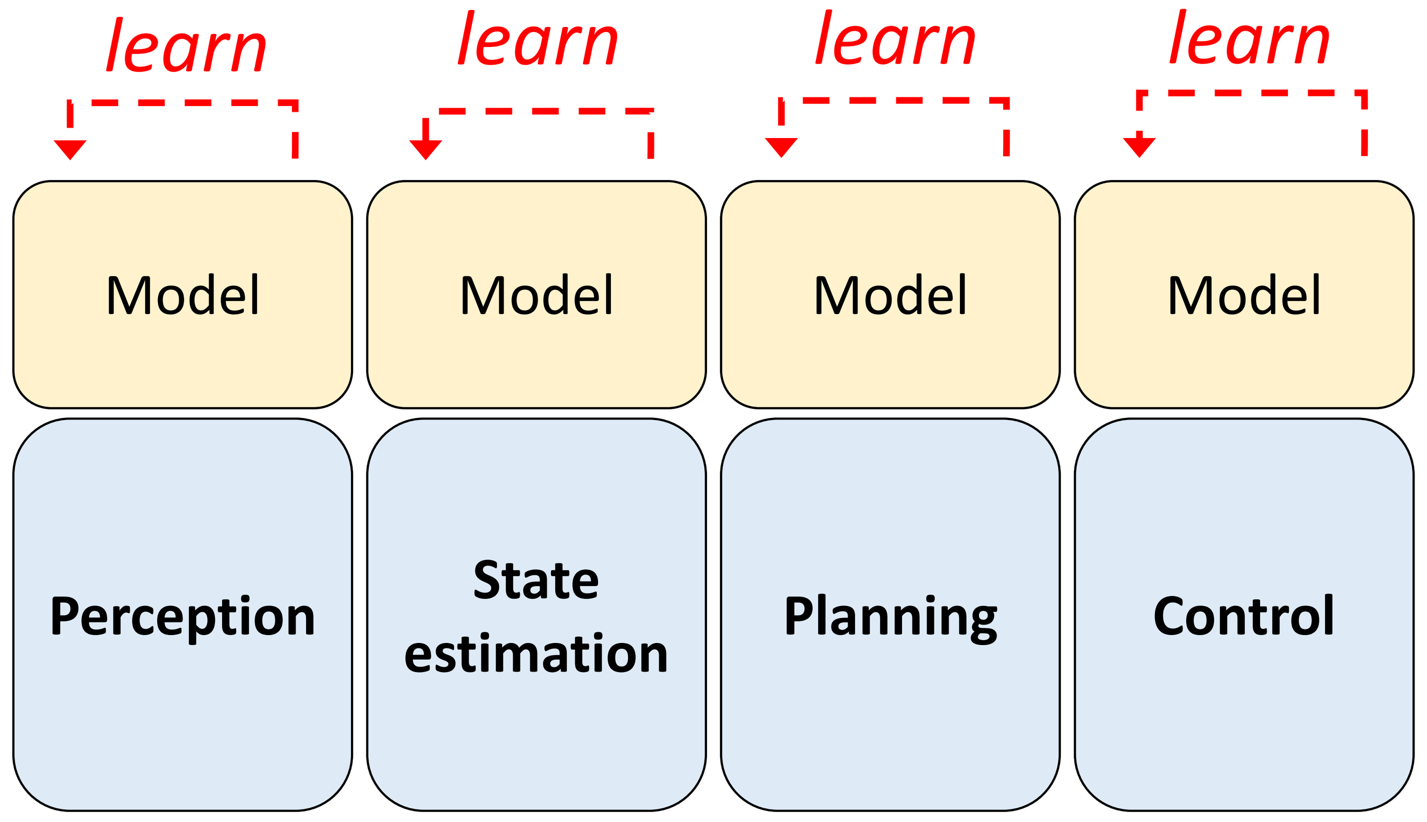


30x30





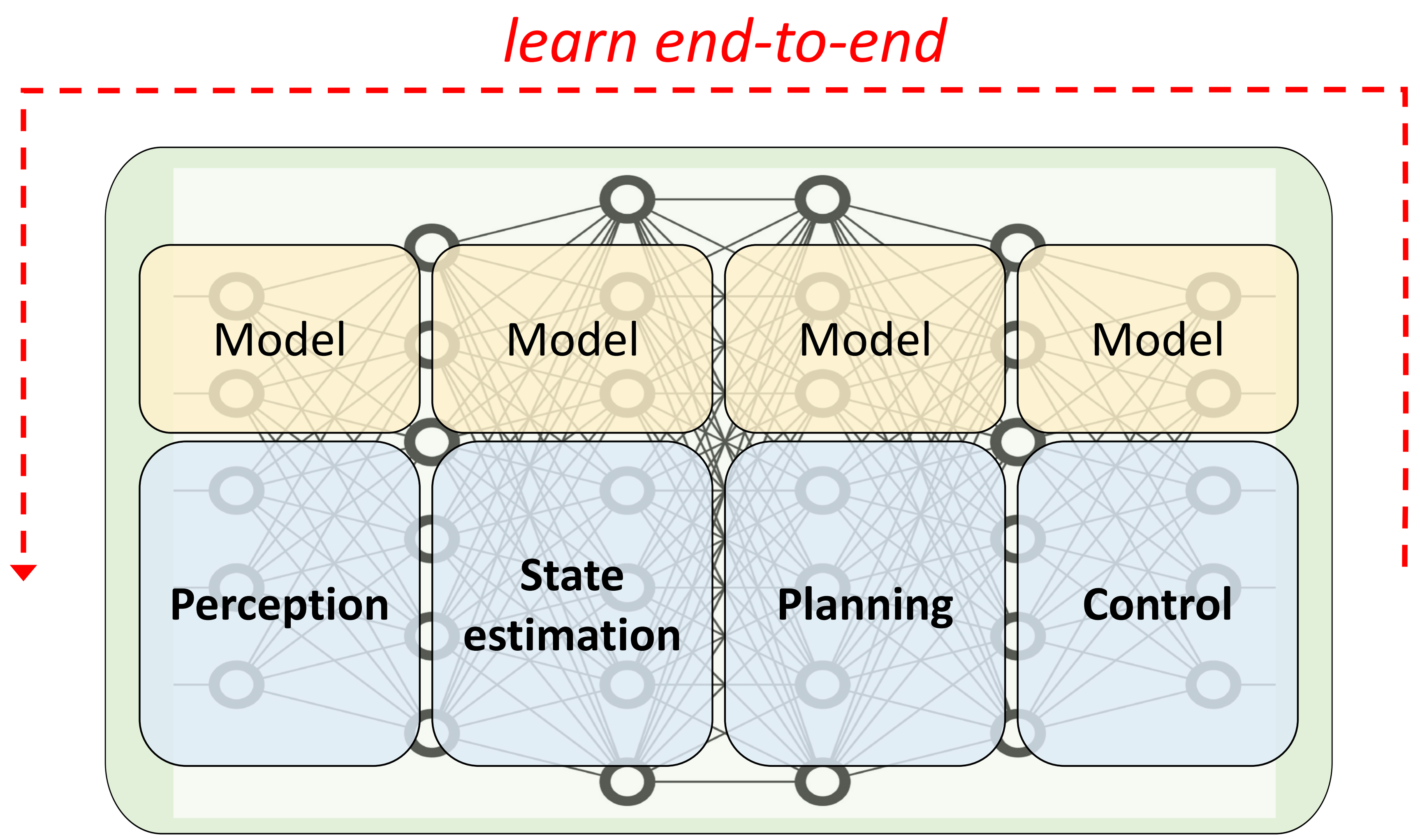
Modular



76.6%

success rate

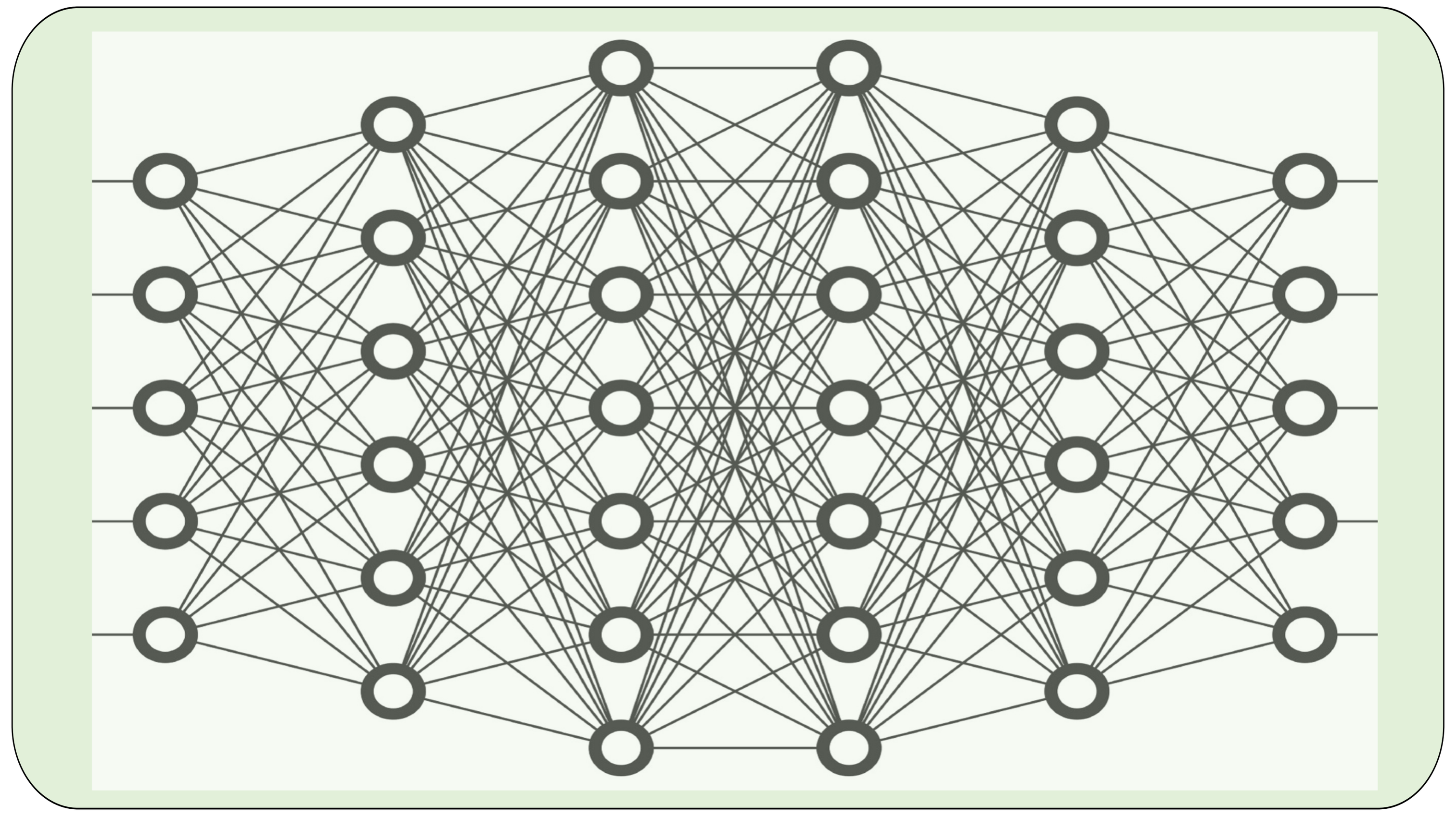
DAN (ours)



99.8%

success rate

End-to-end



38.4%

success rate

Task-oriented learning

Structure prior

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019

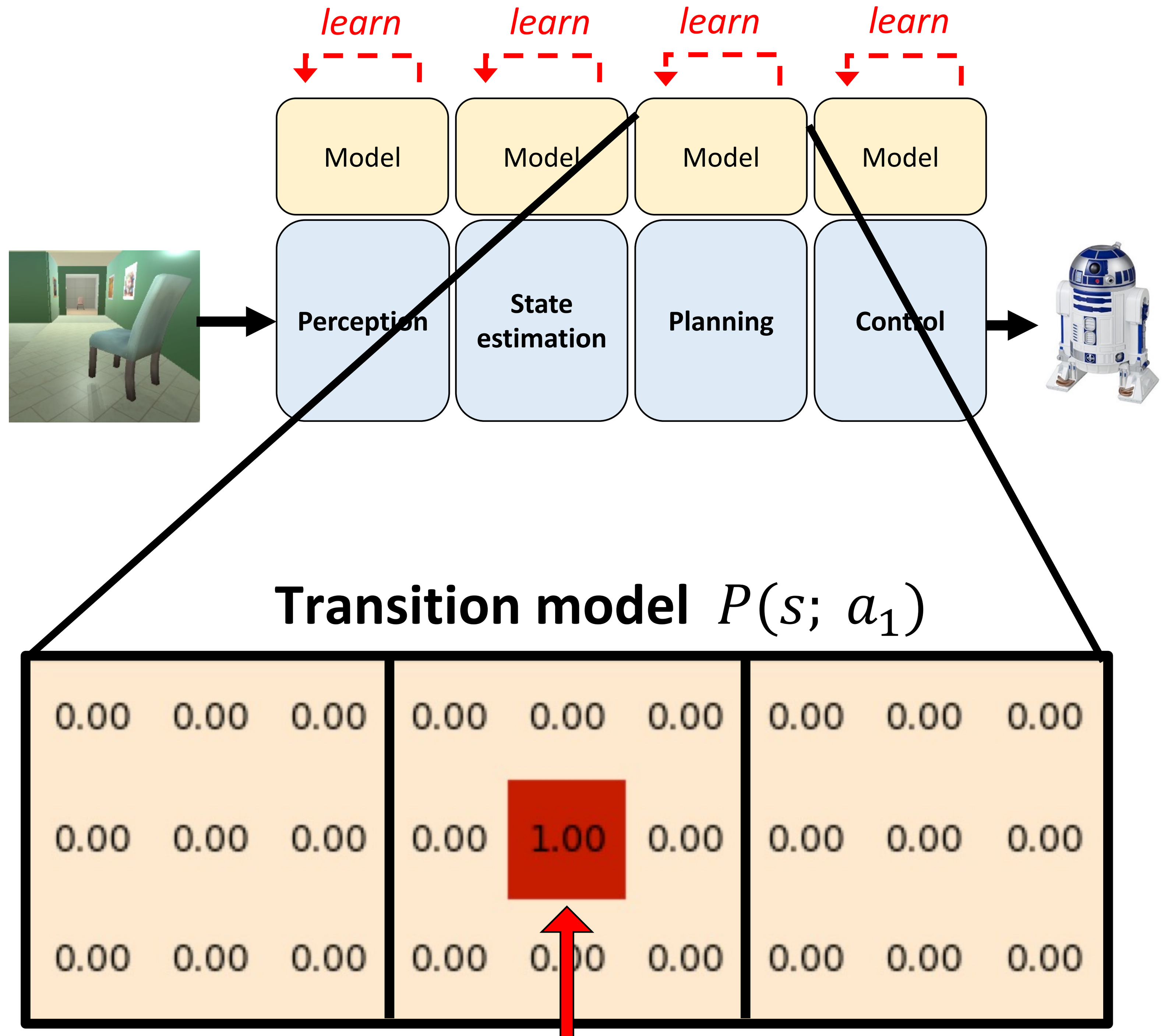
A “wrong” model can fix a “wrong” algorithm

➢ Short planning horizon

Model-based

59.0%

success rate

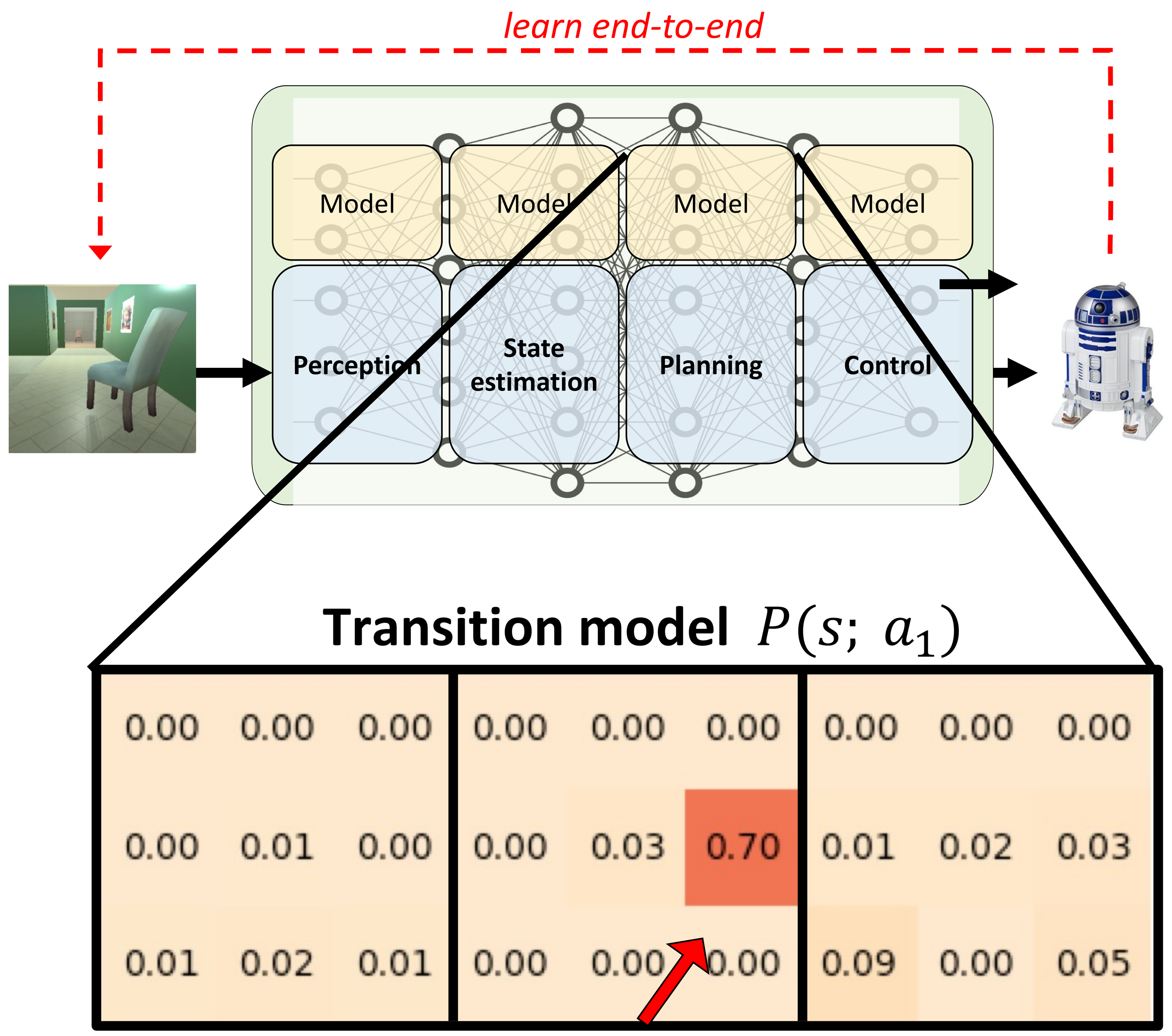


P(**Turn right**)

DAN (ours)

97.8%

success rate



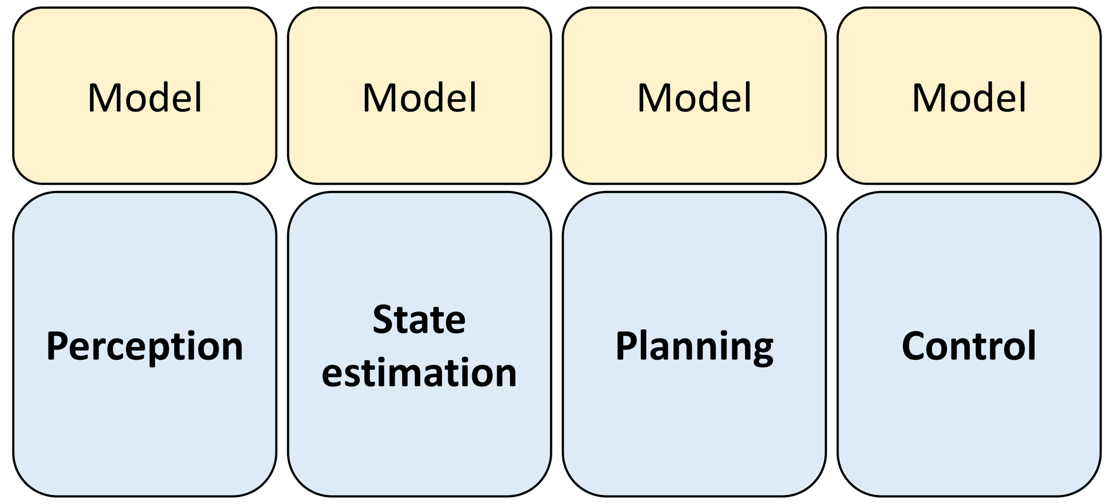
P(**Turn right AND move forward**)

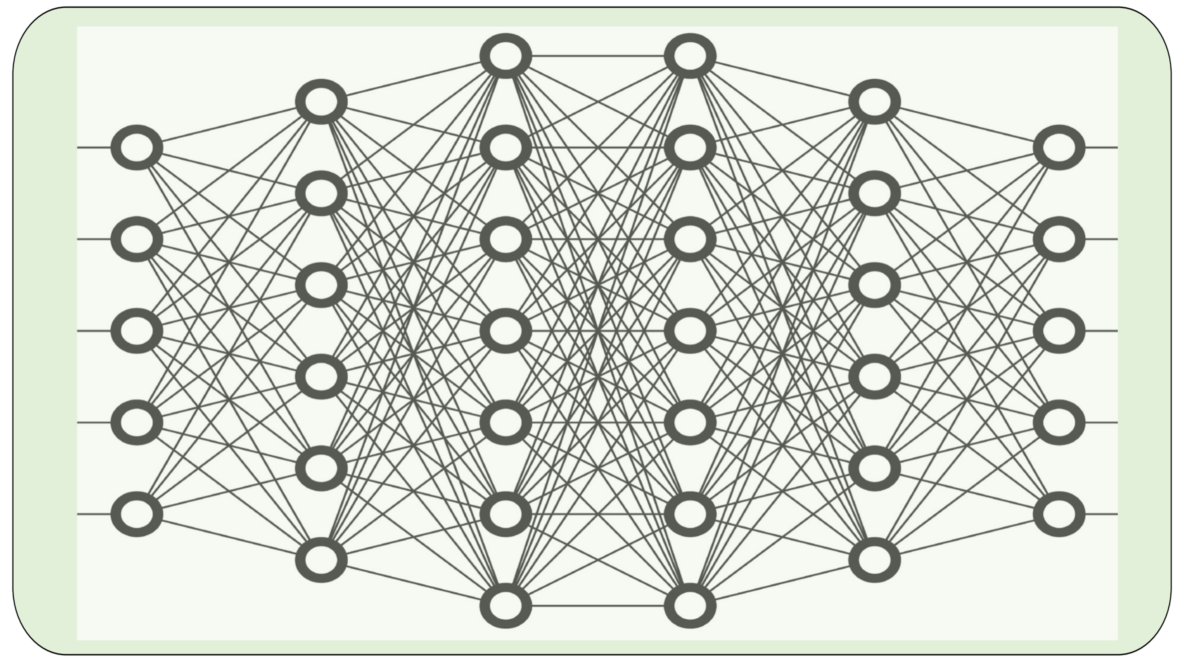
A “wrong” model can fix a “wrong” algorithm

- Short planning horizon? Learn macro actions!
- Transition model doesn’t capture obstacles? Learn to inflate collision penalties!
- Cannot plan for information gathering? Learn to reward it!
- Perception mistakes lead to bad actions? Learn to communicate confidence!

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019





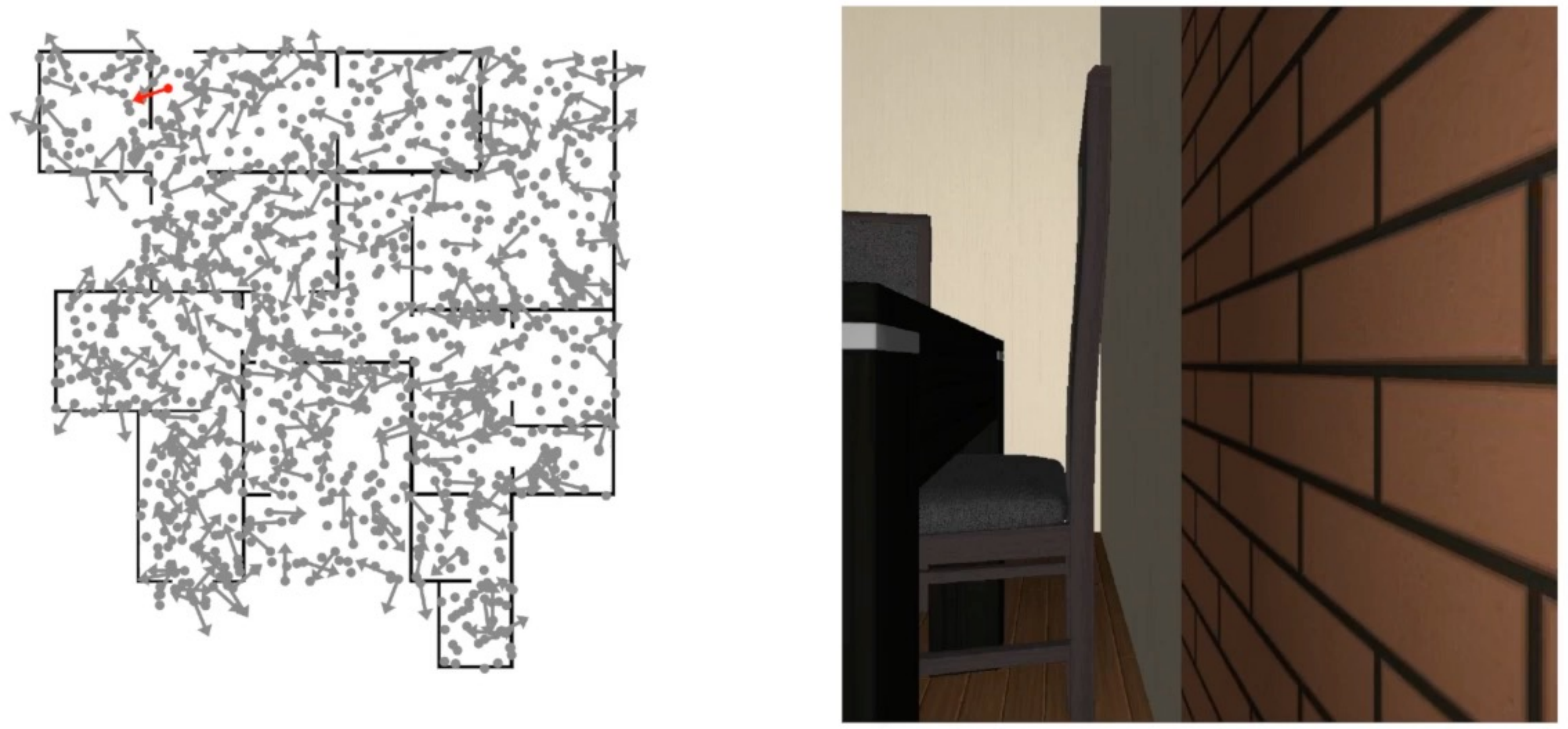
Model-based algorithm structure

DAN

Model-free neural network

P. Karkus, X. Ma, D. Hsu, L.P. Kaelbling, W.S. Lee, T. Lozano-Perez. *Differentiable Algorithm Networks for Composable Robot Learning*. RSS 2019

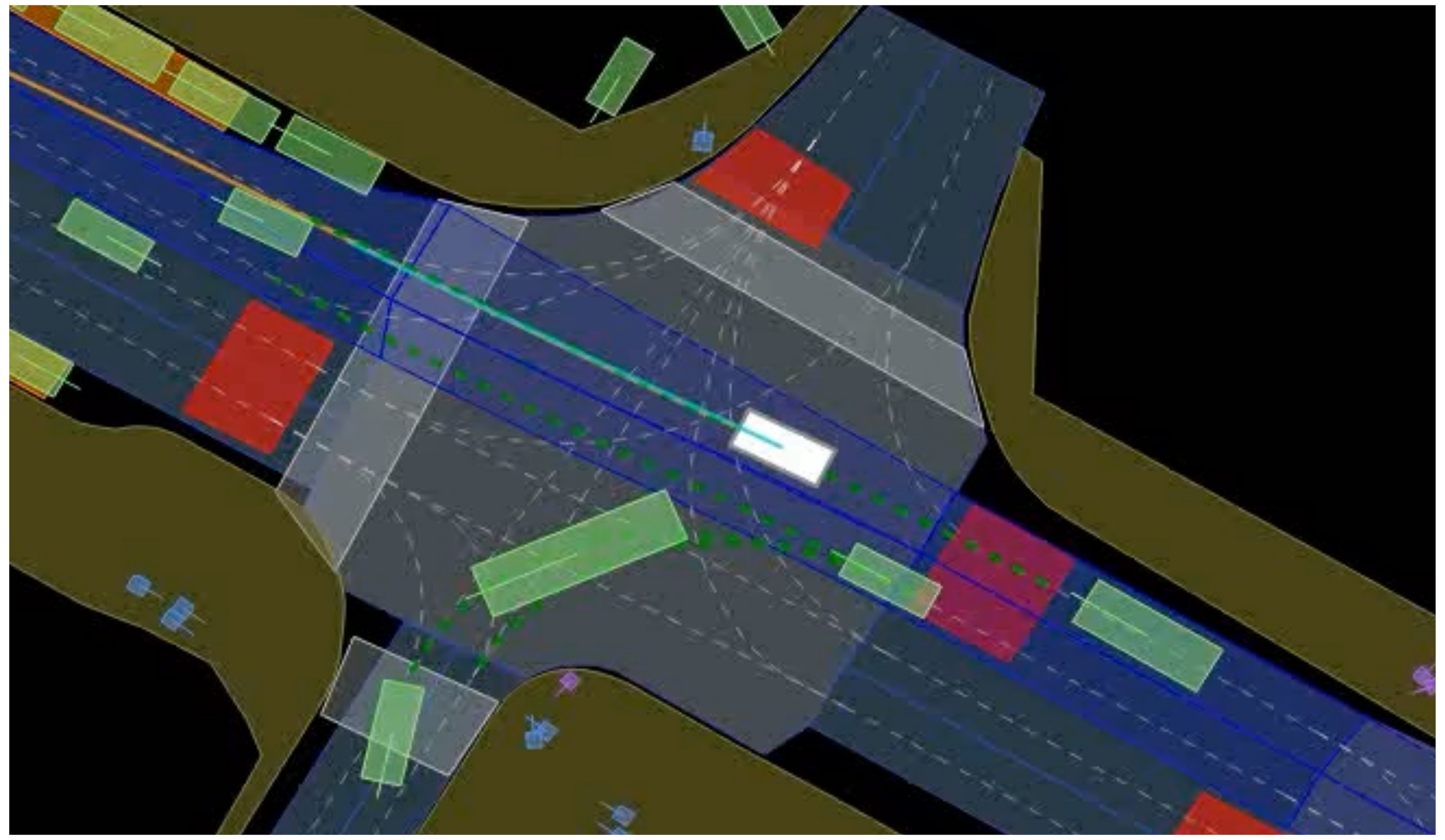
Applications



Visual localization

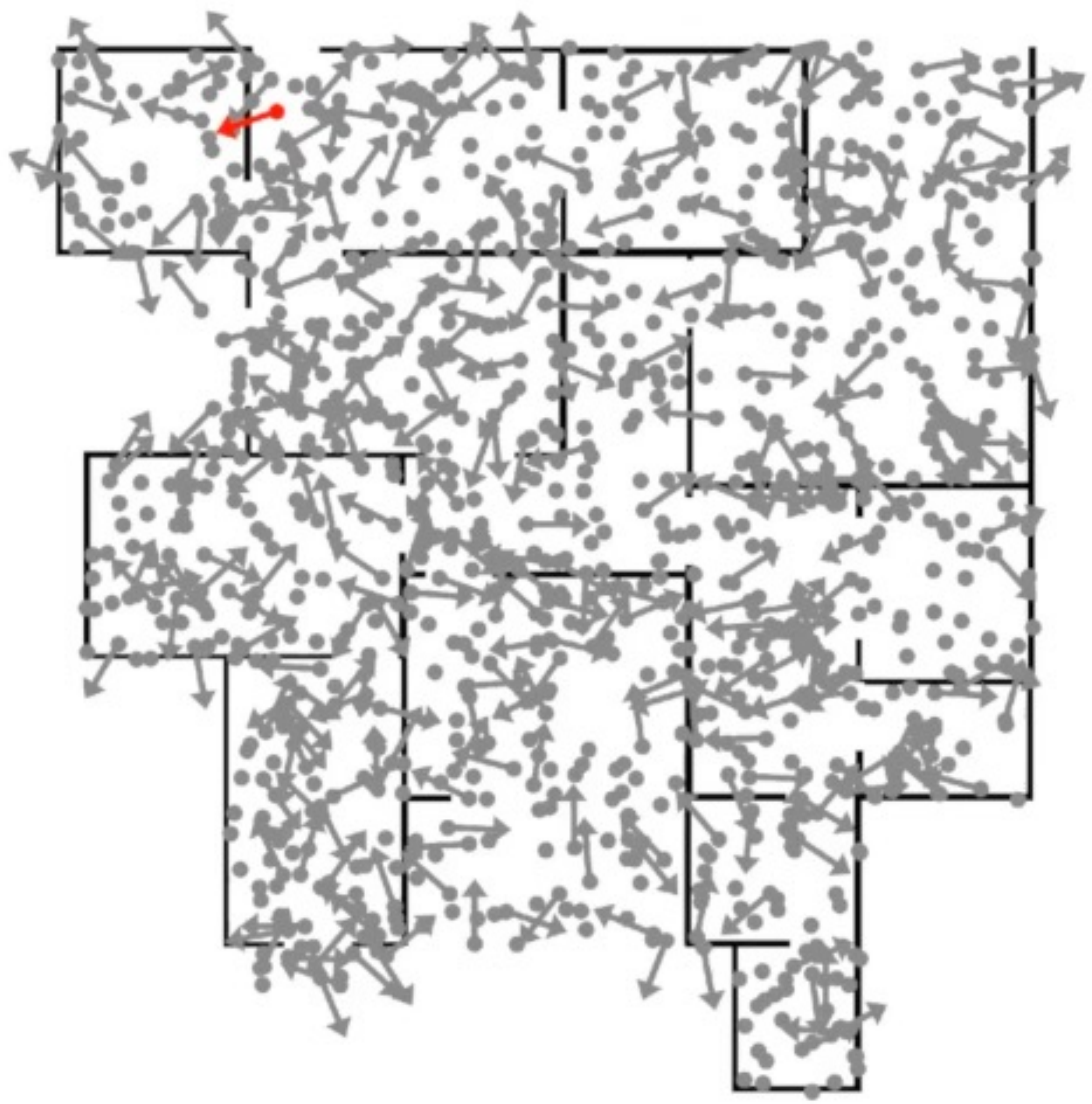
SLAM

Navigation



Autonomous driving

Particle Filter Network



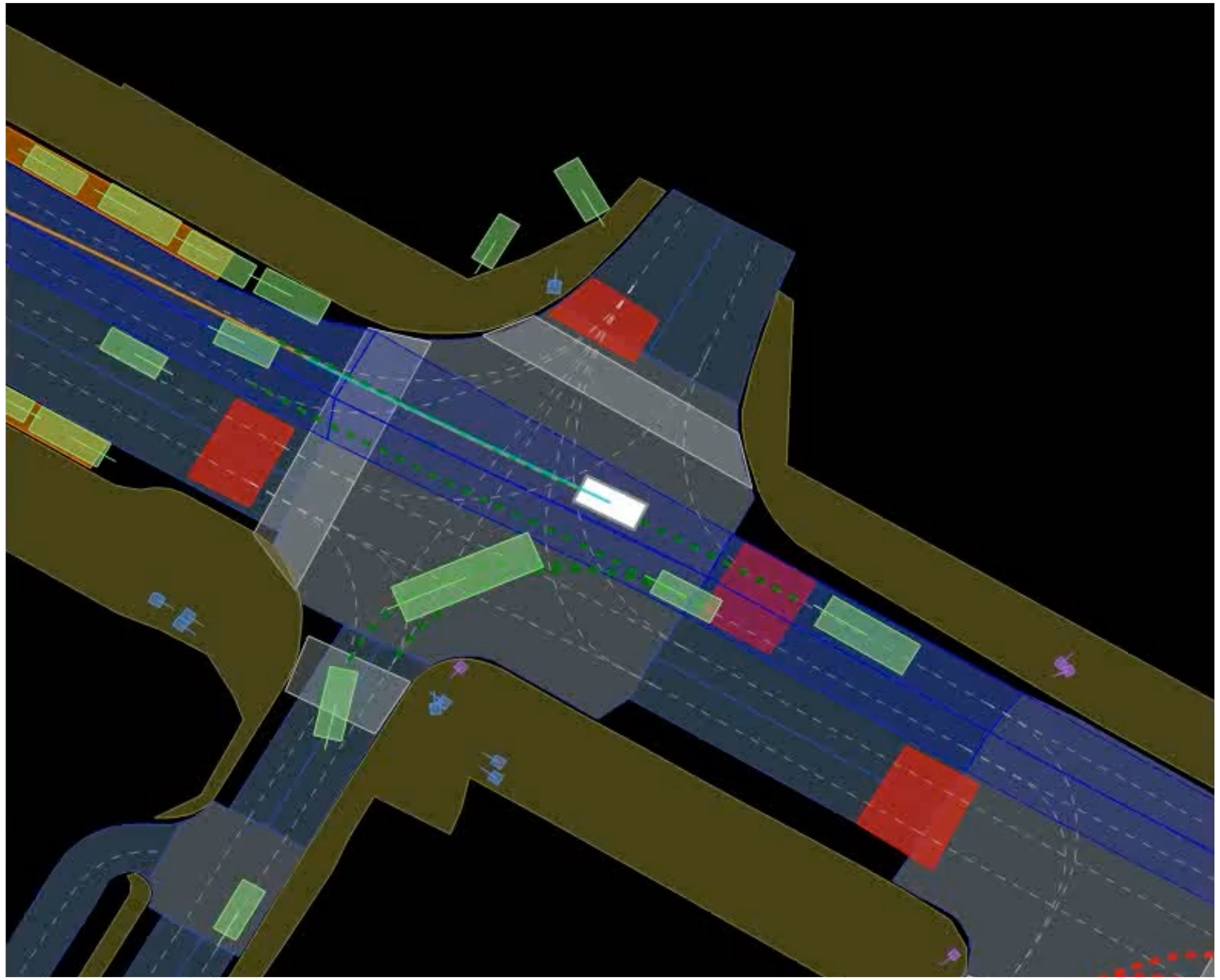
P. Karkus, D. Hsu, W.S. Lee. Particle Filter Networks with Application to Visual Localization. CoRL 2018

Differentiable SLAM-net

P. Karkus, S. Cai, D. Hsu. *Differentiable SLAM-net: Learning Particle SLAM for Visual Navigation.* CVPR 2021

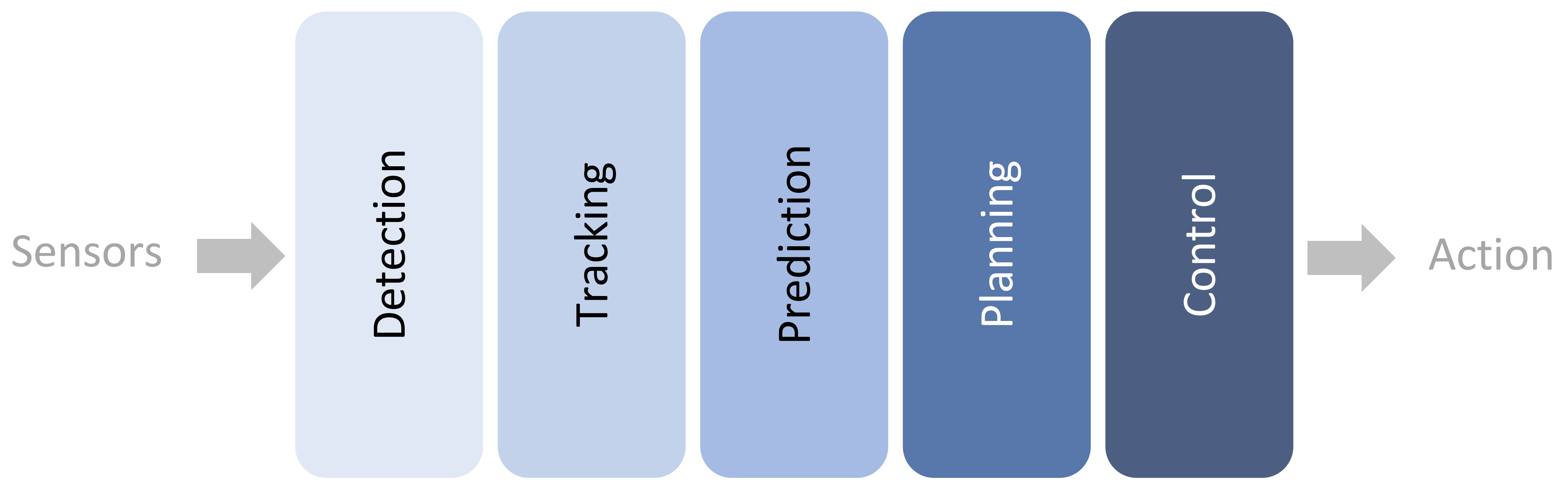
Spot navigation demo

Autonomous driving



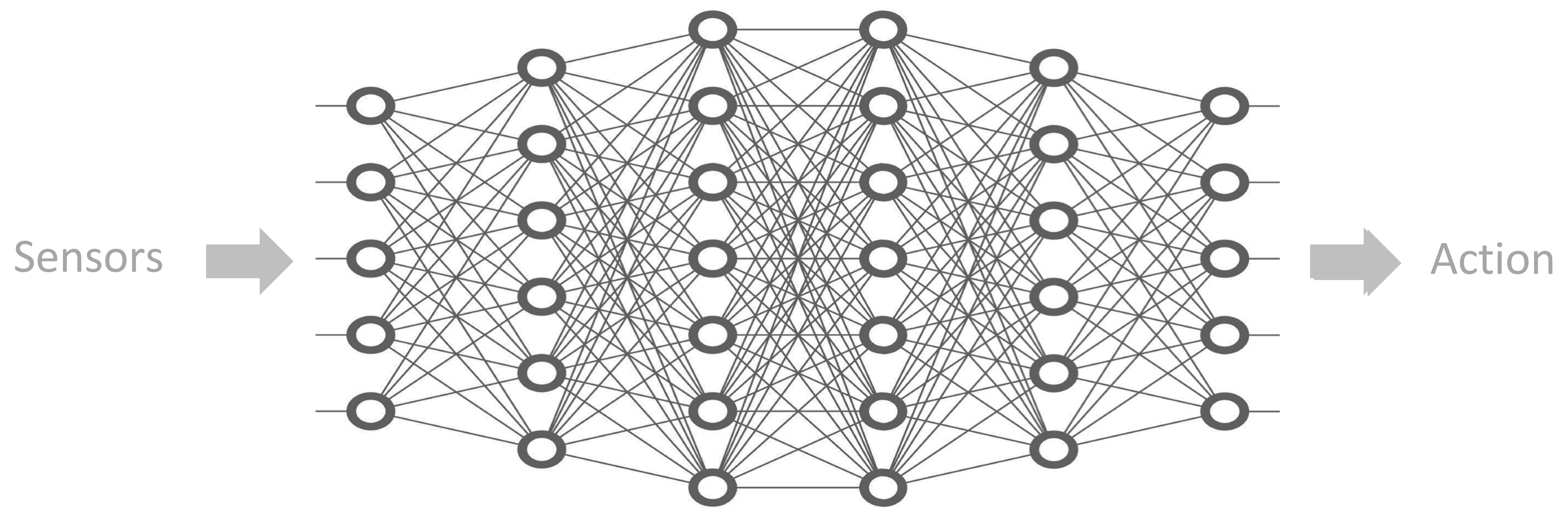
Differentiable & Modular AV Stacks

Motivation



Modular architecture

?



End-to-end architecture

Differentiable & Modular AV Stacks

Key idea

Modular architecture

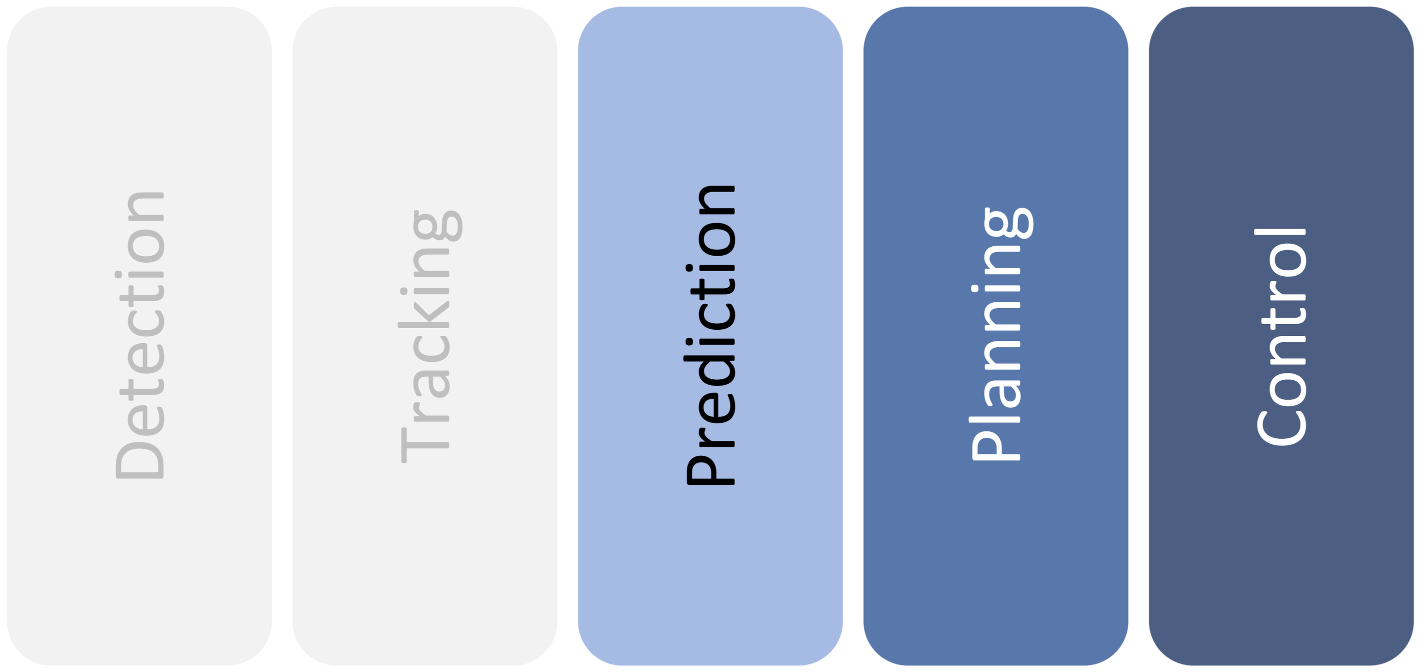
End-to-end architecture

Differentiable & modular stack

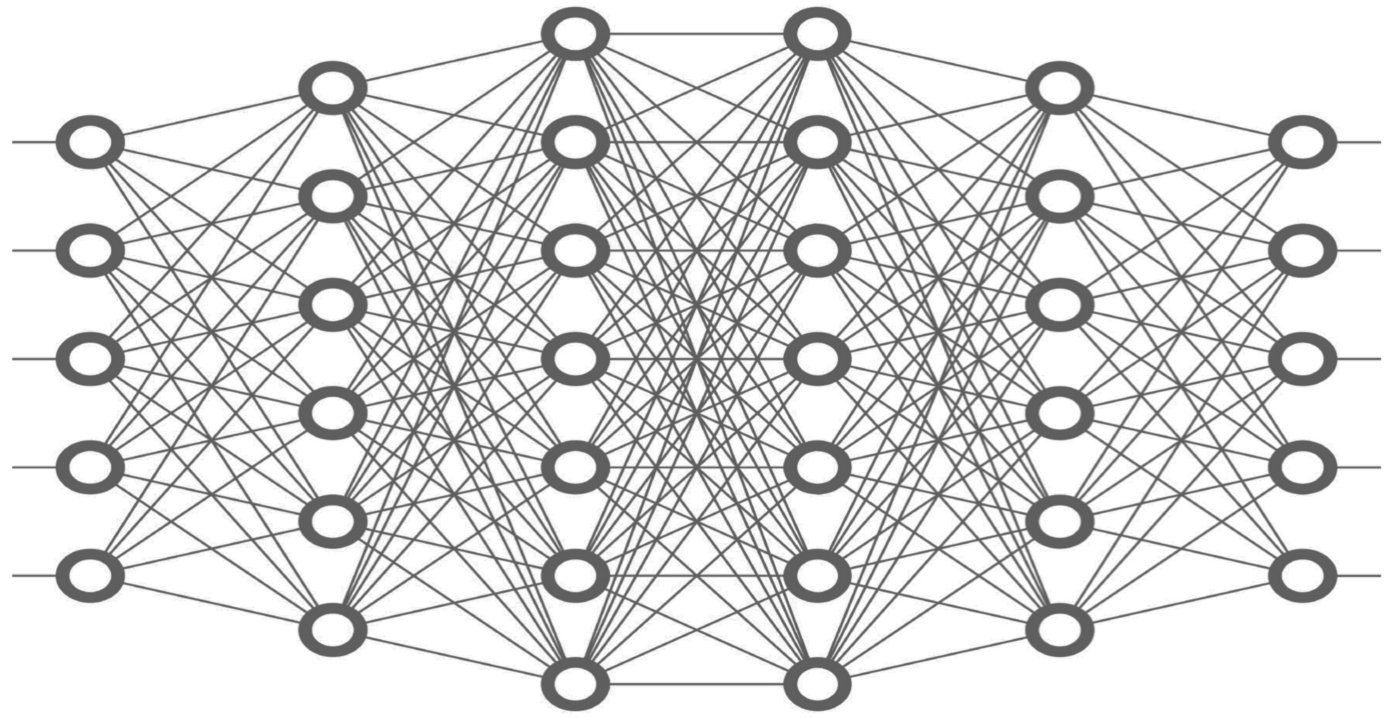
Idea: use a differentiable computation graph to represent modules and train them end-to-end

Differentiable & Modular AV Stacks

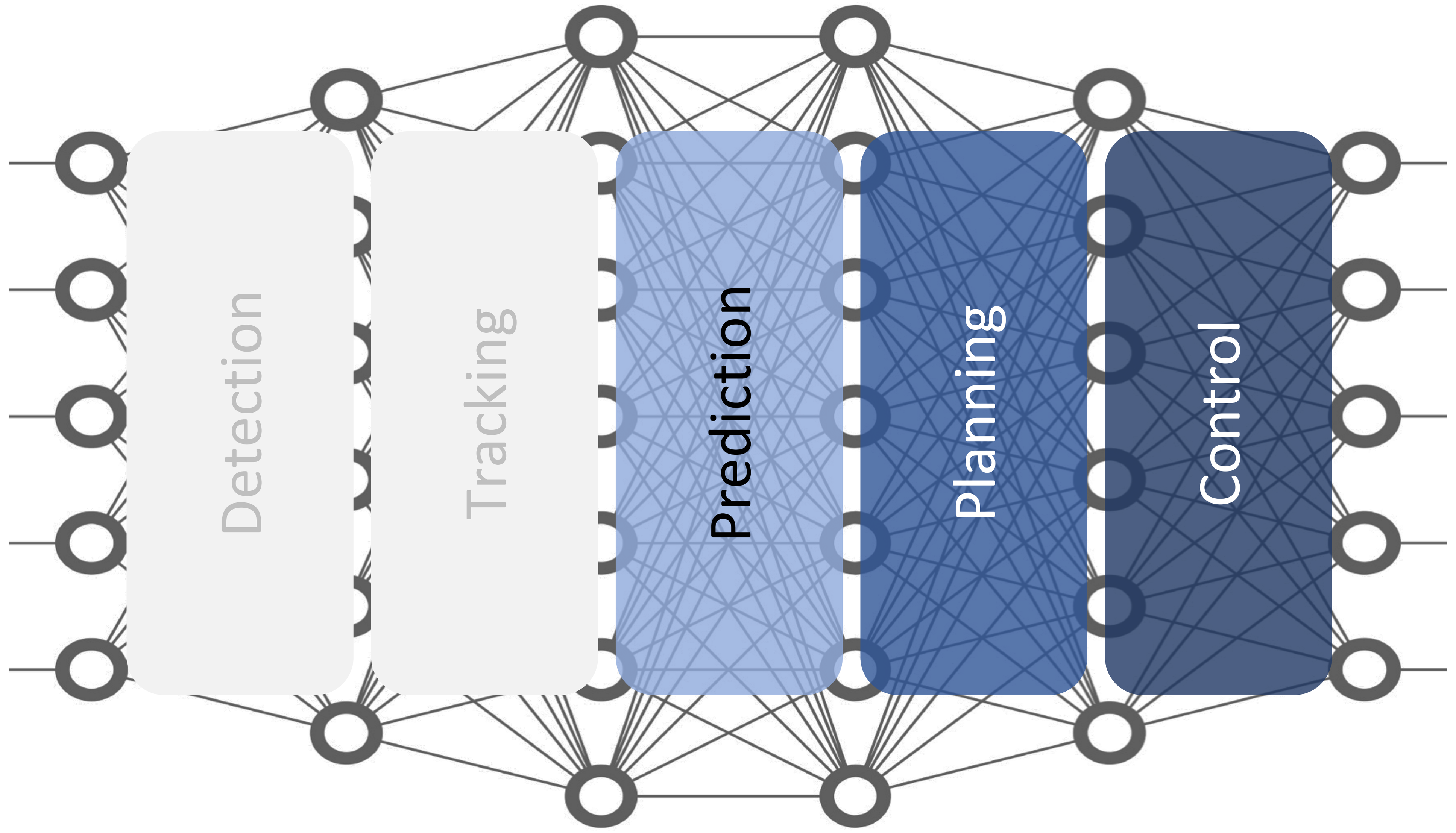
Example



Modular architecture



End-to-end architecture



Differentiable & modular stack

P. Karkus, B. Ivanovic, S. Mannor, M. Pavone. DiffStack: A Differentiable and Modular Control Stack for Autonomous Vehicles, CoRL 2022

DIFFSTACK: DIFFERENTIABLE PREDICTION-PLANNING-CONTROL

DIFFSTACK: DIFFERENTIABLE PREDICTION-PLANNING-CONTROL

Prediction
Trajectron++ model
(Salzmann et al. 2020)

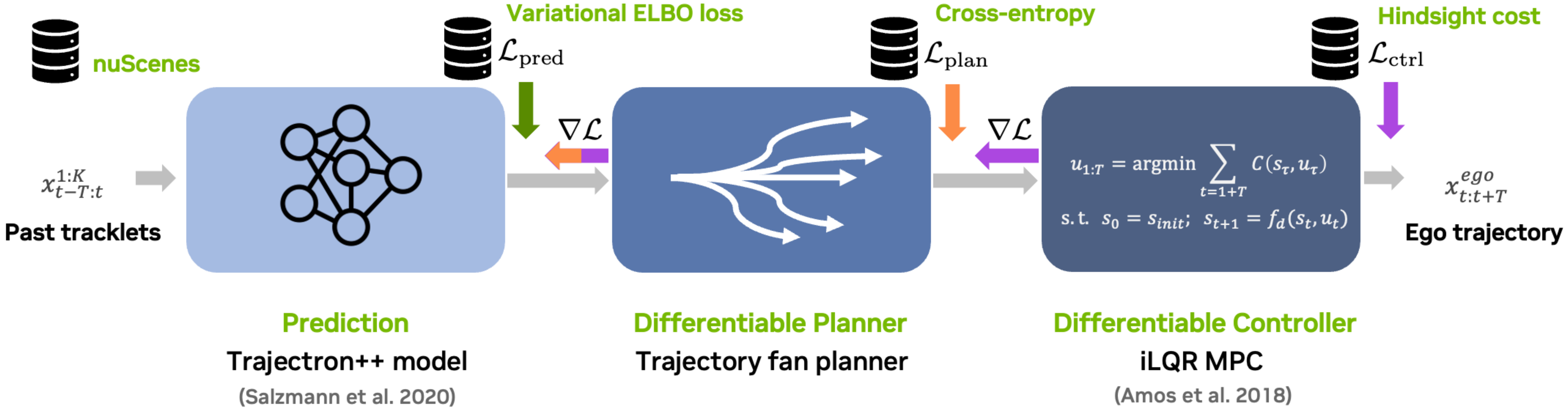
Differentiable Planner
Trajectory fan planner

Differentiable Controller
iLQR MPC
(Amos et al. 2018)

P. Karkus, B. Ivanovic, S. Mannor, M. Pavone. DiffStack: A Differentiable and Modular Control Stack for Autonomous Vehicles, CoRL 2022

Differentiable & Modular AV Stacks

Quantitative results

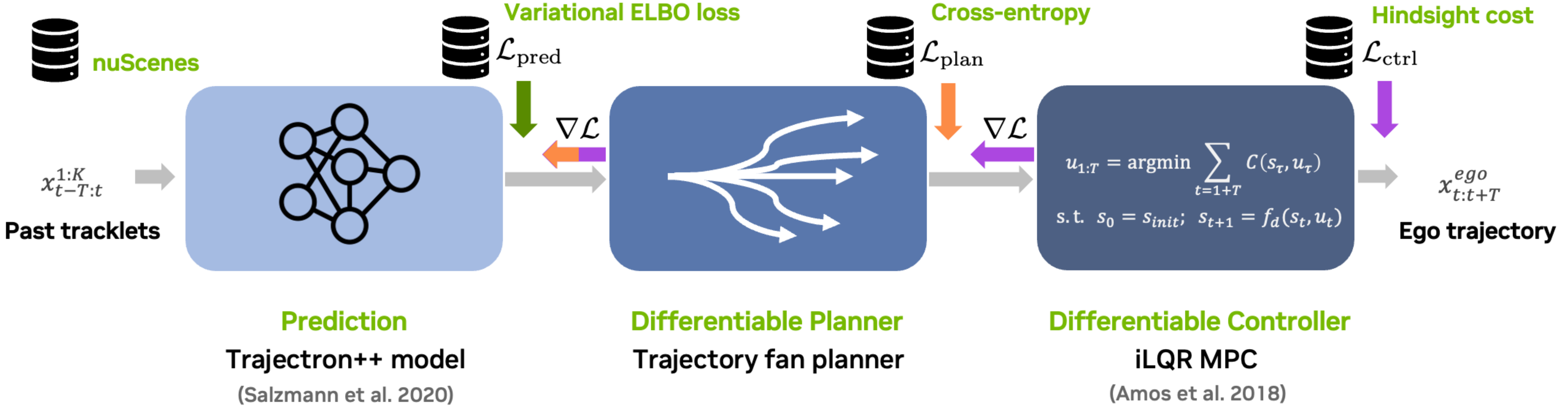


Standard modular stack		
Differentiable stack (ours)		

P. Karkus, B. Ivanovic, S. Mannor, M. Pavone. DiffStack: A Differentiable and Modular Control Stack for Autonomous Vehicles, CoRL 2022

Differentiable & Modular AV Stacks

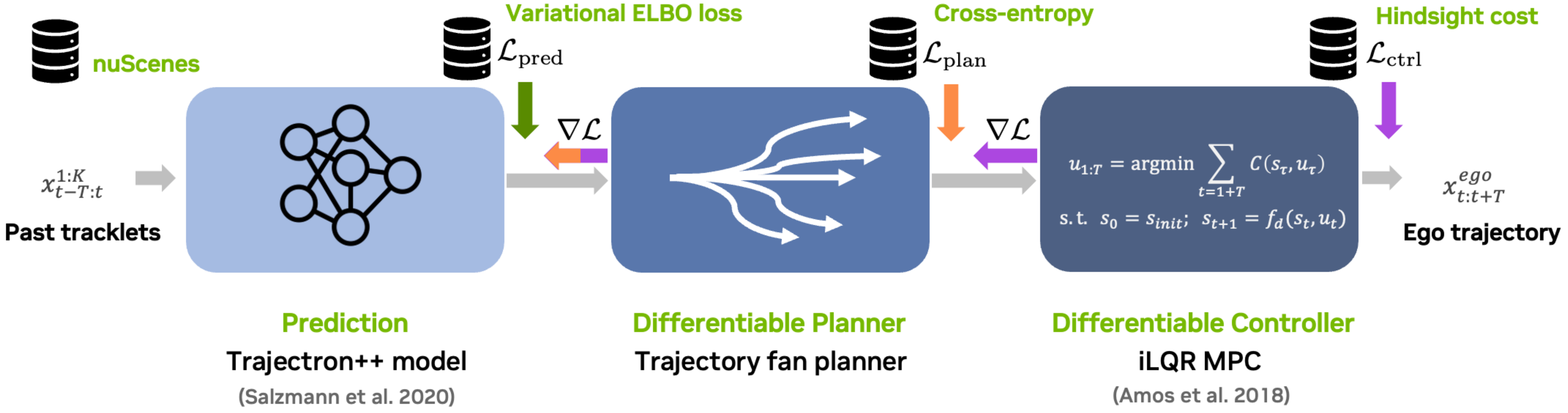
Quantitative results



	Perception performance ADE (m) ↓	
Standard modular stack	1.32 ± 0.06	
Differentiable stack (ours)	1.27 ± 0.07	

Differentiable & Modular AV Stacks

Quantitative results



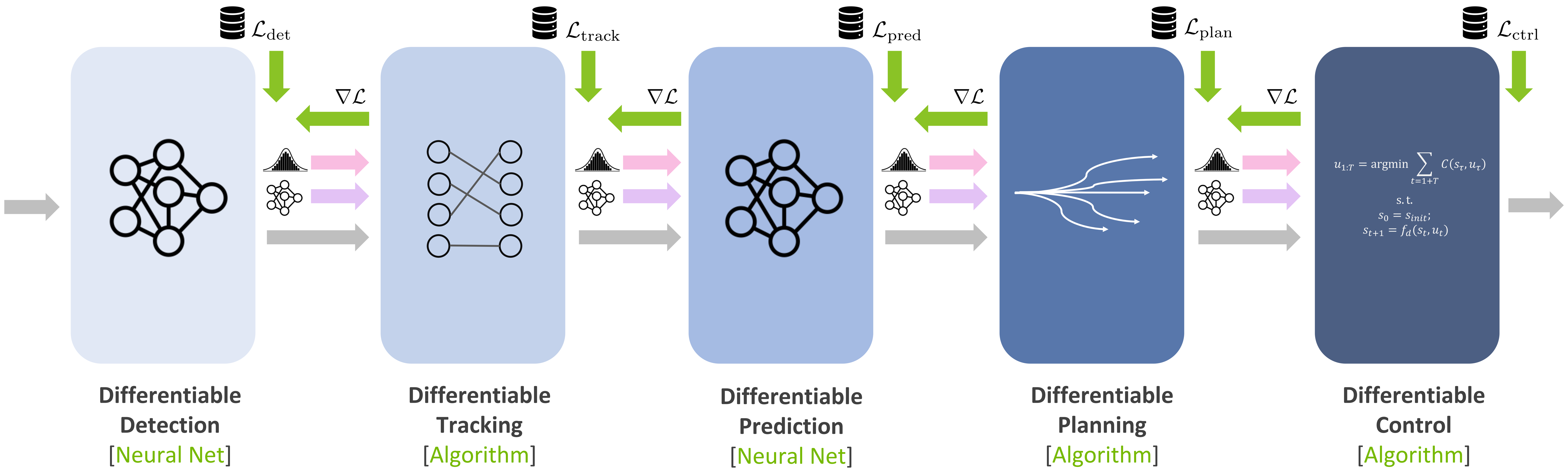
	Perception performance ADE (m) ↓	Cost difference from theoretical best ↓
Standard modular stack	1.32 ± 0.06	29% ± 2%
Differentiable stack (ours)	1.27 ± 0.07	**17%** ± 2%

Upper bound on cost: planning with no predictions
Lower bound on cost: planning with ground truth predictions

P. Karkus, B. Ivanovic, S. Mannor, M. Pavone. DiffStack: A Differentiable and Modular Control Stack for Autonomous Vehicles, CoRL 2022

Differentiable & Modular AV Stacks

Vision



$\nabla\mathcal{L}$ Joint training

Learned interfaces

Uncertainty propagation

Adversarial robustness & safety

Task-oriented metrics

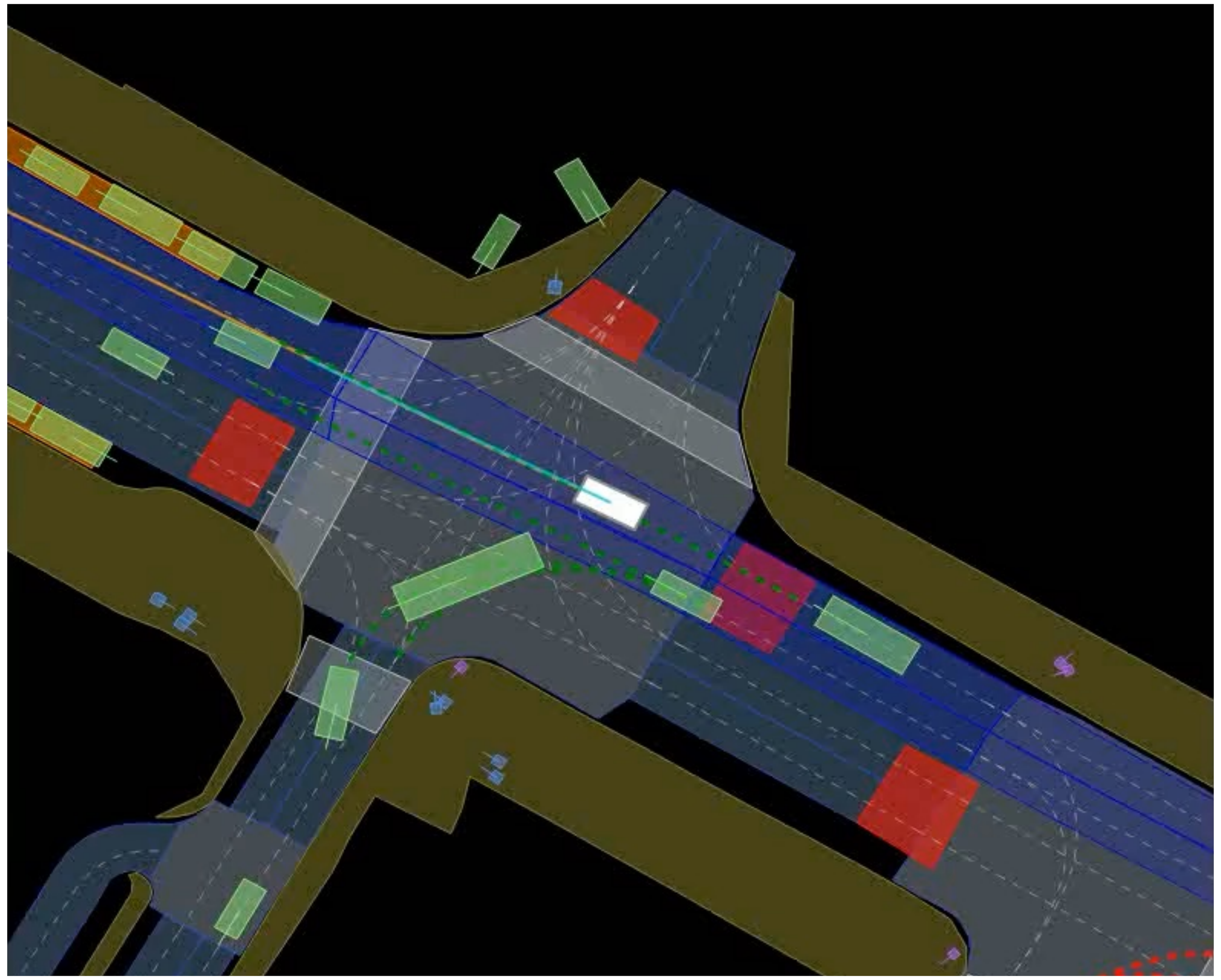
Simulation technology

Differentiable & Modular AV Stacks

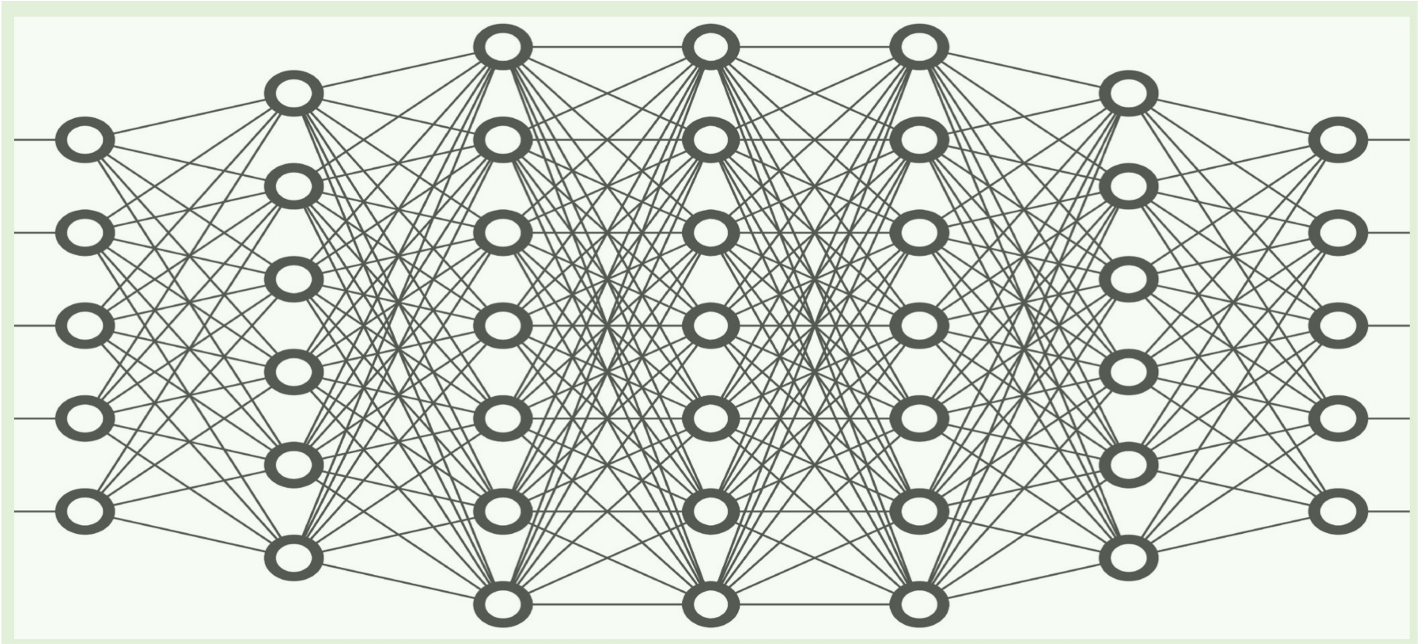
Ongoing work

CARLA Driving Challenge

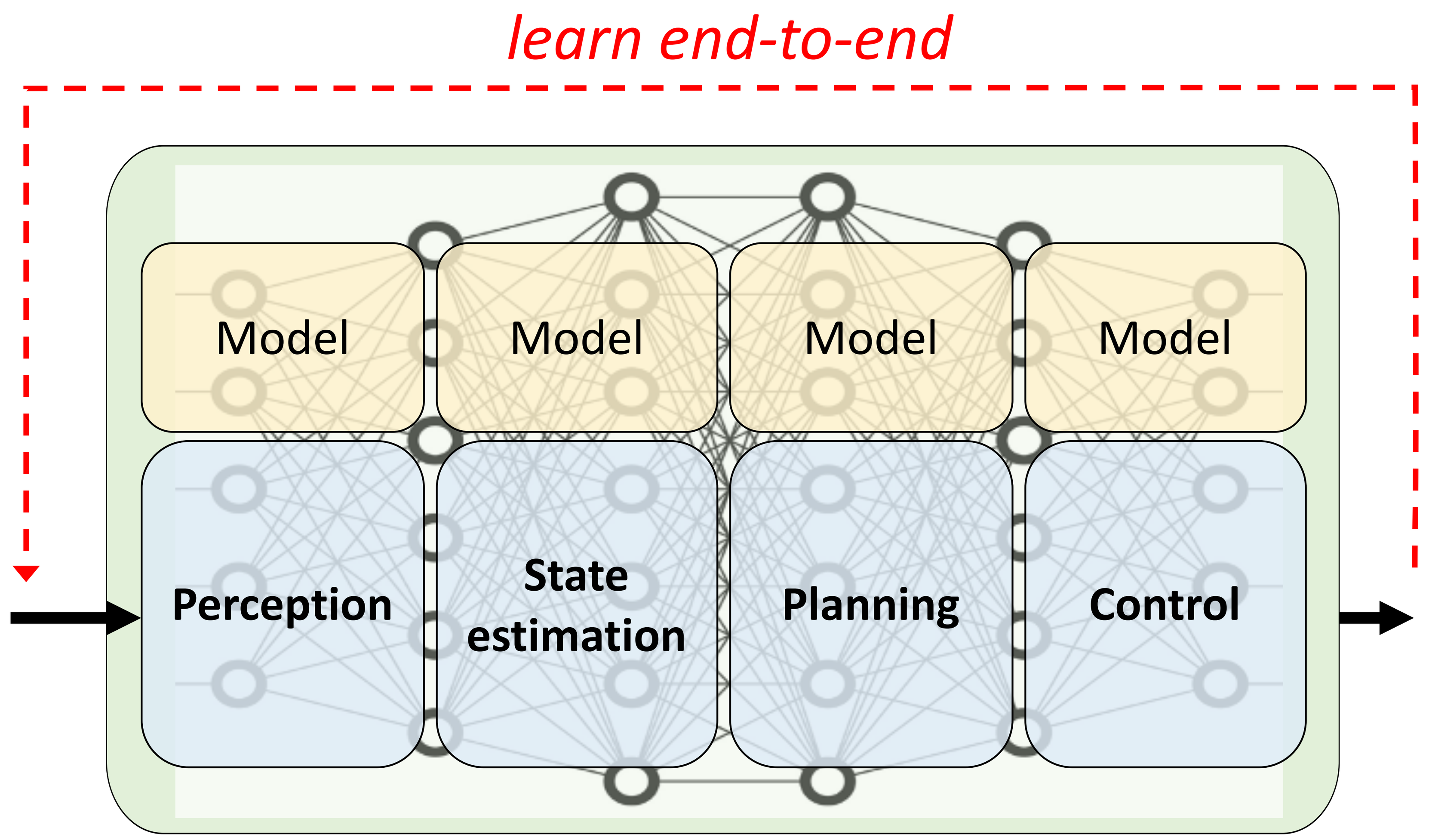
nuPlan Driving Challenge



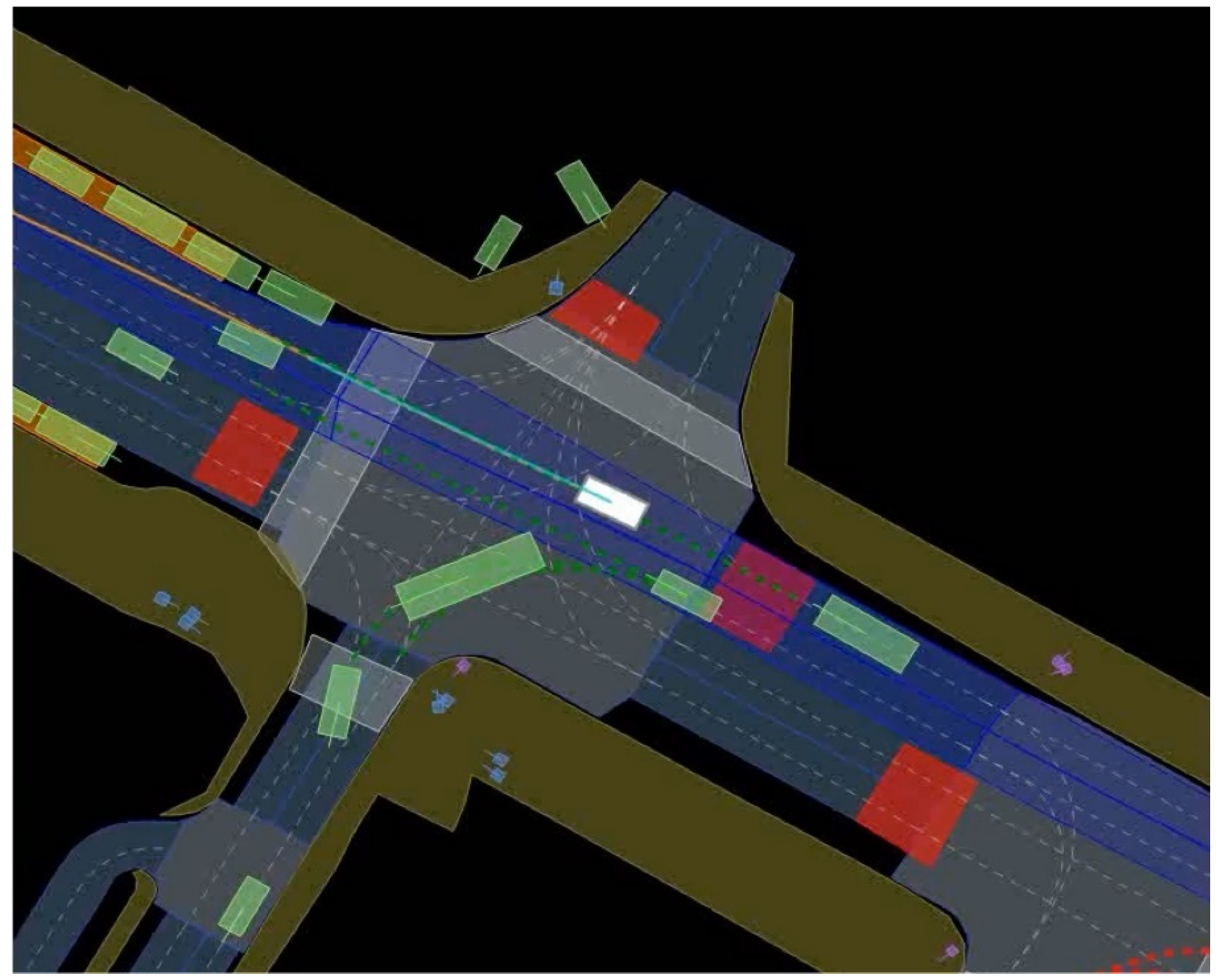
Summary



Algorithms can be built into neural networks



DAN =
Modular + Task-oriented



Applications