Scaling up and Stabilizing Differentiable Planning with Implicit Differentiation

ICLR 2023

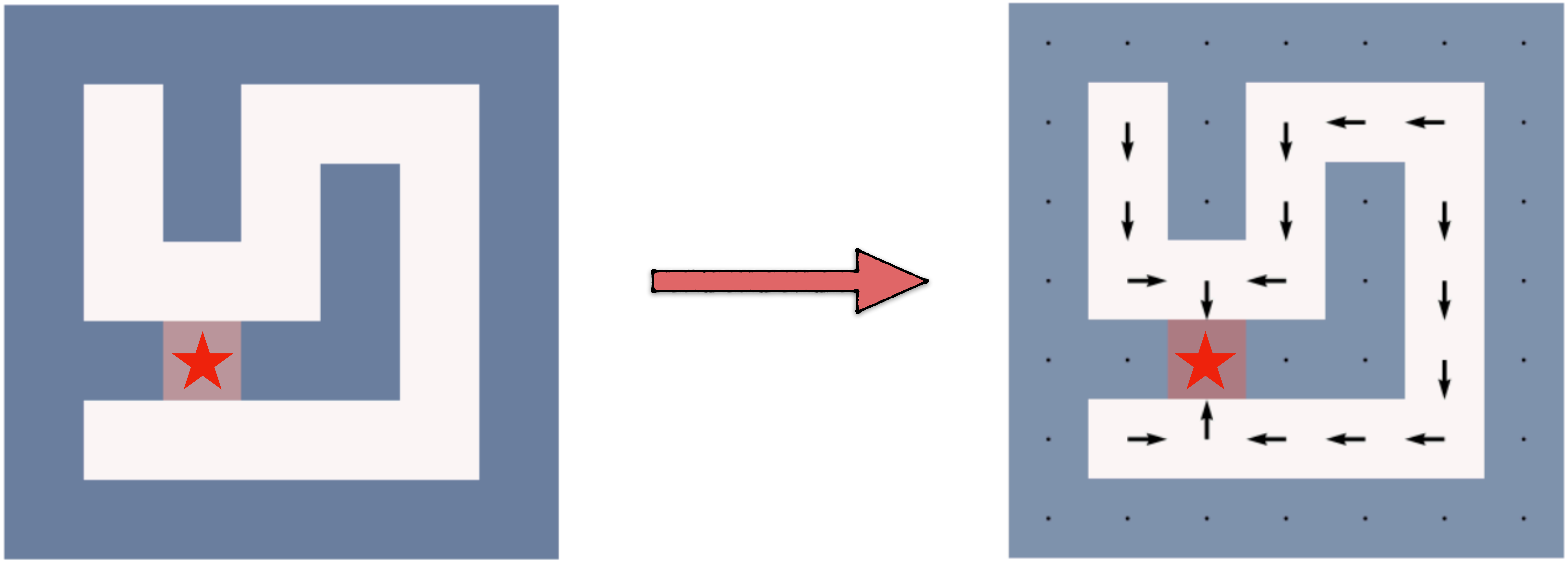
***Linfeng Zhao*[1]**, *Huazhe Xu*[2,3], *Lawson Wong*[1]

[1] Khoury College of Computer Sciences, Northeastern University

[2] Institute for Interdisciplinary Information Sciences, Tsinghua University

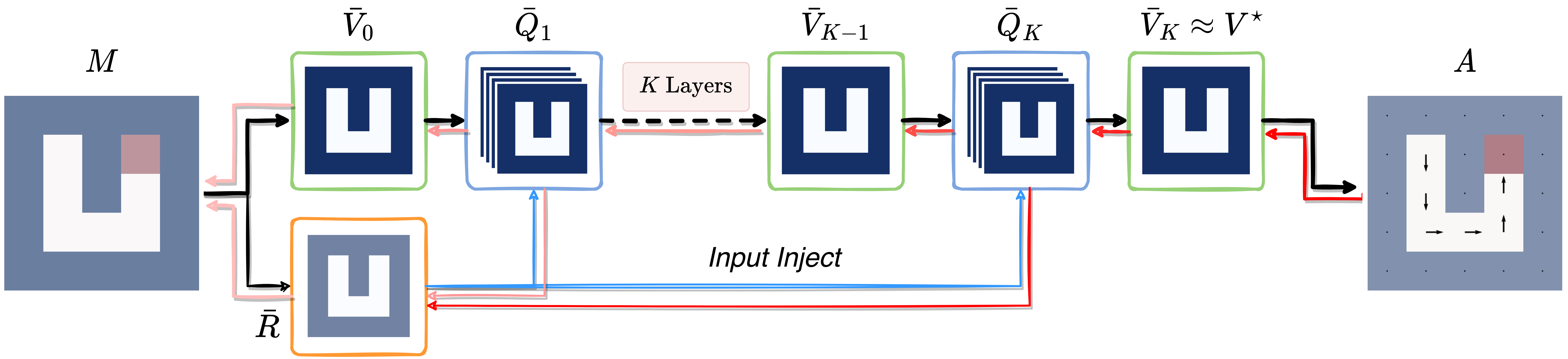
[3] Shanghai Qi Zhi Institute

Path Planning



Find shortest path / optimal actions to the goal location (red)

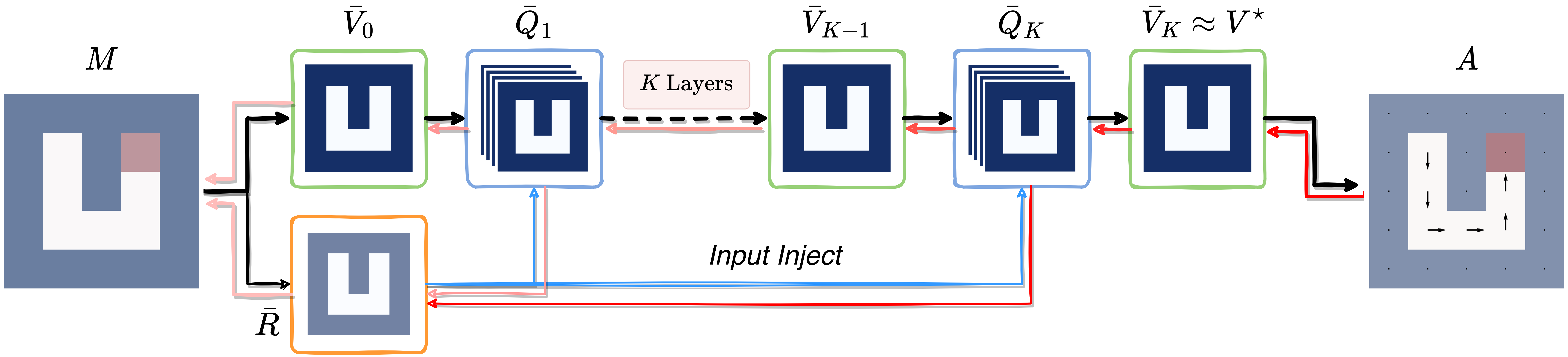
Background: Value Iteration Networks



- *Value Iteration Networks* implement Value Iteration by CNNs
- It iteratively applies Bellman operator and differentiates through multiple layers

Tamar et al. Value Iteration Networks. NIPS 2016.

Algorithmic Differentiation in VIN



Key issue:
Forward and Backward passes are coupled together

When the planning horizon is long, backpropagation is not scalable, stable, or efficient

Implicit Differentiation

- Bellman equation:

$$\boldsymbol{v}^\star = f(\boldsymbol{v}^\star, \boldsymbol{x})$$

- Differentiating both sides:

$$\frac{\partial \boldsymbol{v}^\star(\cdot)}{\partial(\cdot)} = \frac{\partial f(\boldsymbol{v}^\star(\cdot), \boldsymbol{x})}{\partial(\cdot)} = \frac{\partial f(\boldsymbol{v}^\star, \boldsymbol{x})}{\partial \boldsymbol{v}^\star} \frac{\boldsymbol{v}^\star(\cdot)}{\partial(\cdot)} + \frac{\partial f(\boldsymbol{v}^\star, \boldsymbol{x})}{\partial(\cdot)}$$

$$\frac{\partial \ell}{\partial(\cdot)} = \frac{\partial \ell}{\partial \boldsymbol{v}^\star} \frac{\partial \boldsymbol{v}^\star(\cdot)}{\partial(\cdot)} = \frac{\partial \ell}{\partial \boldsymbol{v}^\star} \left(I - \frac{\partial f(\boldsymbol{v}^\star, \boldsymbol{x})}{\partial \boldsymbol{v}^\star} \right)^{-1} \frac{\partial f(\boldsymbol{v}^\star, \boldsymbol{x})}{\partial(\cdot)}$$

- Solving backward fixed-point:

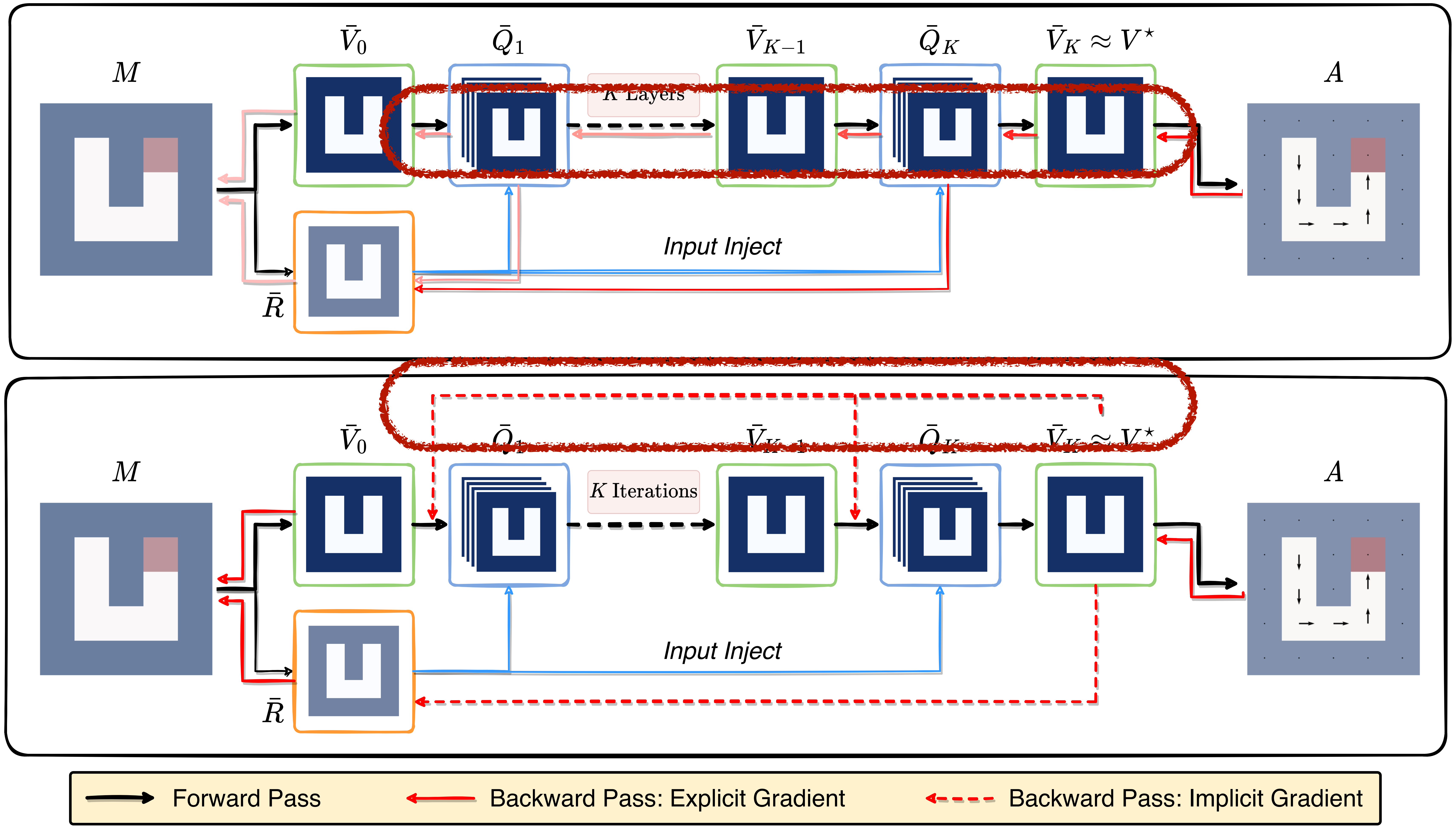
$$\boldsymbol{w}^\top \triangleq \frac{\partial \ell}{\partial \boldsymbol{v}^\star} \left(I - \frac{\partial f(\boldsymbol{v}^\star, \boldsymbol{x})}{\partial \boldsymbol{v}^\star} \right)^{-1}; \quad \boldsymbol{w}^\top = \boldsymbol{w}^\top \frac{\partial f(\boldsymbol{v}^\star, \boldsymbol{x})}{\partial \boldsymbol{v}^\star} + \frac{\partial \ell}{\partial \boldsymbol{v}^\star}.$$

Bai et al. Deep Equilibrium Models. 2019.
Nikishin et al. Control-Oriented Model-Based Reinforcement Learning with Implicit Differentiation. 2021.
Gehring et al. Understanding End-to-End Model-Based Reinforcement Learning Methods as Implicit Parameterization. 2021.

Method: Implicit Differentiable Planners

Algorithmic Differentiable Planner: (ADP)
VIN

Implicit Differentiable Planner: (IDP)
***ID**-VIN*



Comparison

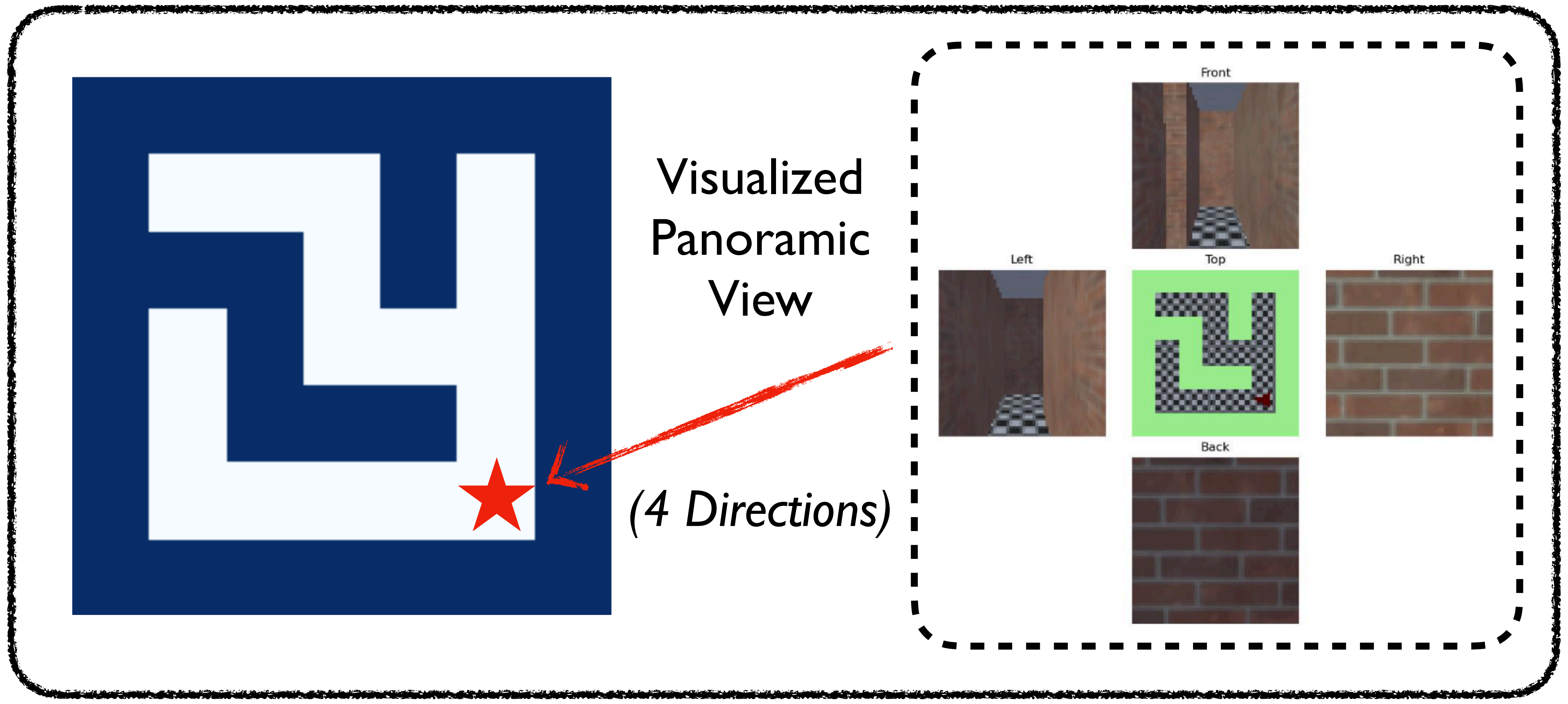
Algorithmic Differentiable Planners (ADP)

- ADPs (e.g., VIN) couple forward and backward passes
- Gradients may explode or vanish

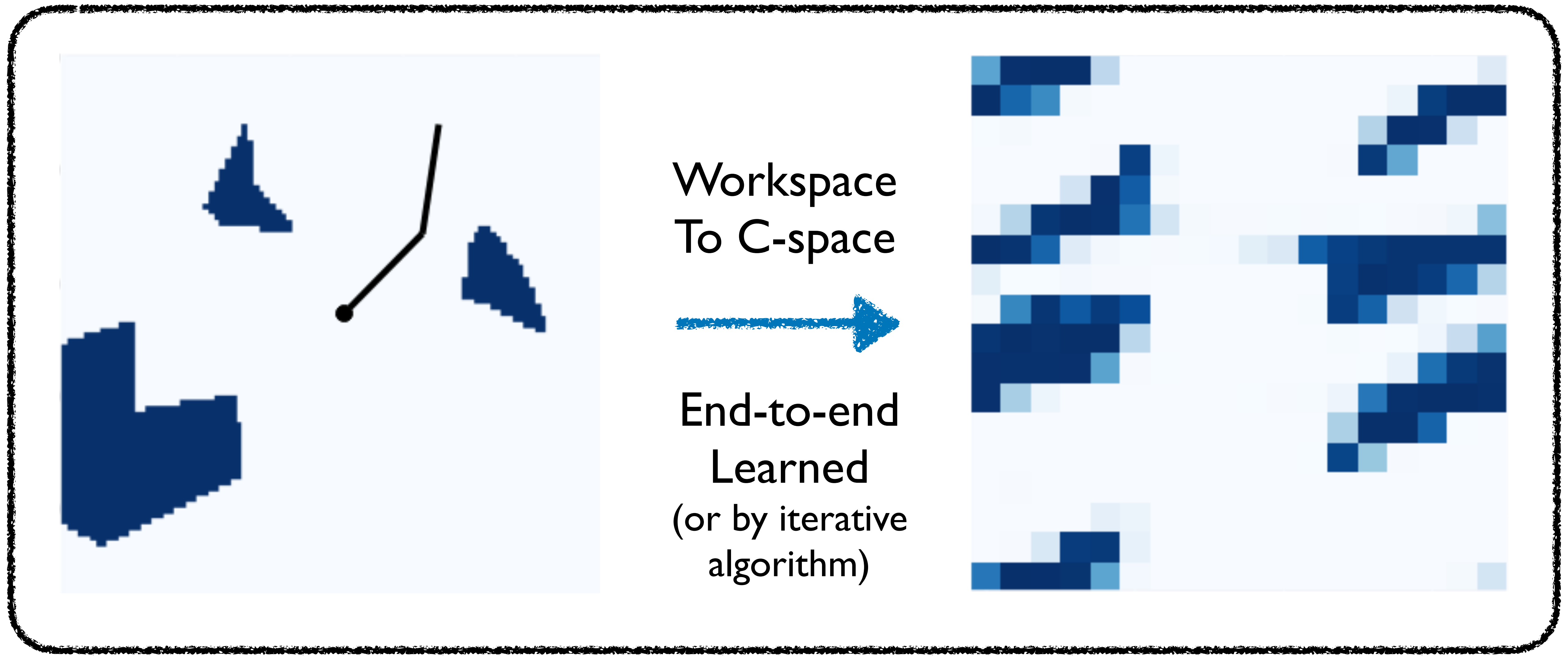
Implicit Differentiable Planners (IDP)

- Our IDPs (e.g., ID-VIN) *decouple* forward and backward passes
- Implicit differentiation is constant in forward planning horizon
- Allow training to scale up to larger maps and planning horizons with stable gradient computation

Experiment: Setup

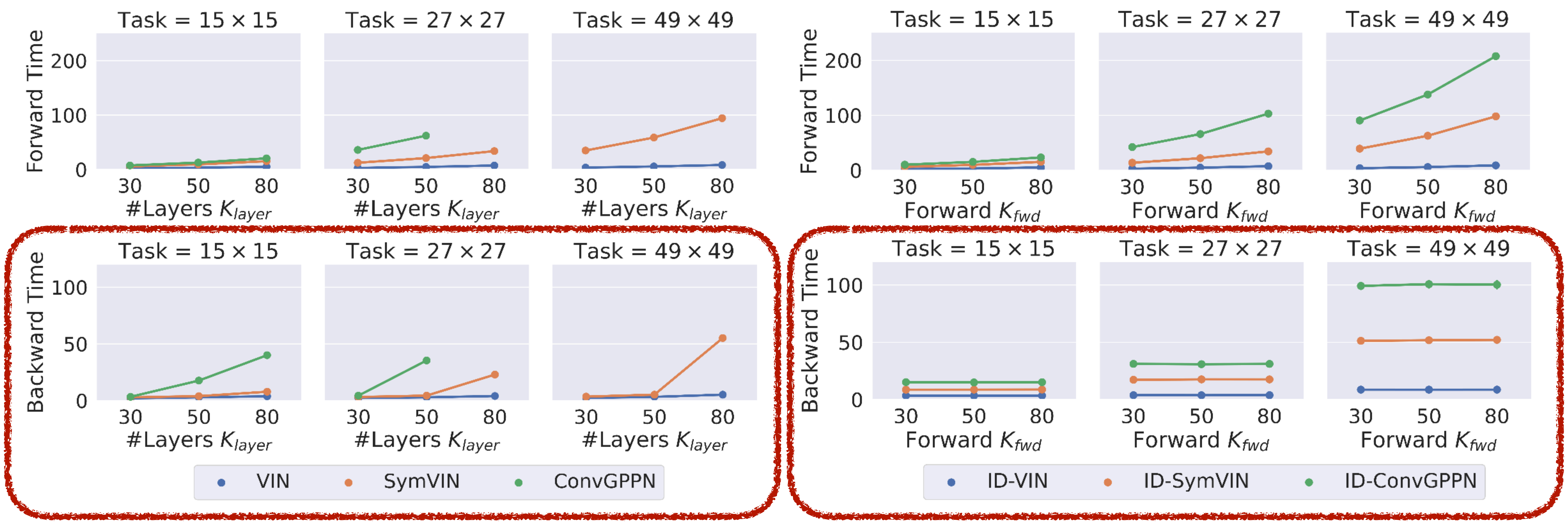


2D and Visual
Maze Navigation



2-DOF Manipulation
In Workspace and C-space

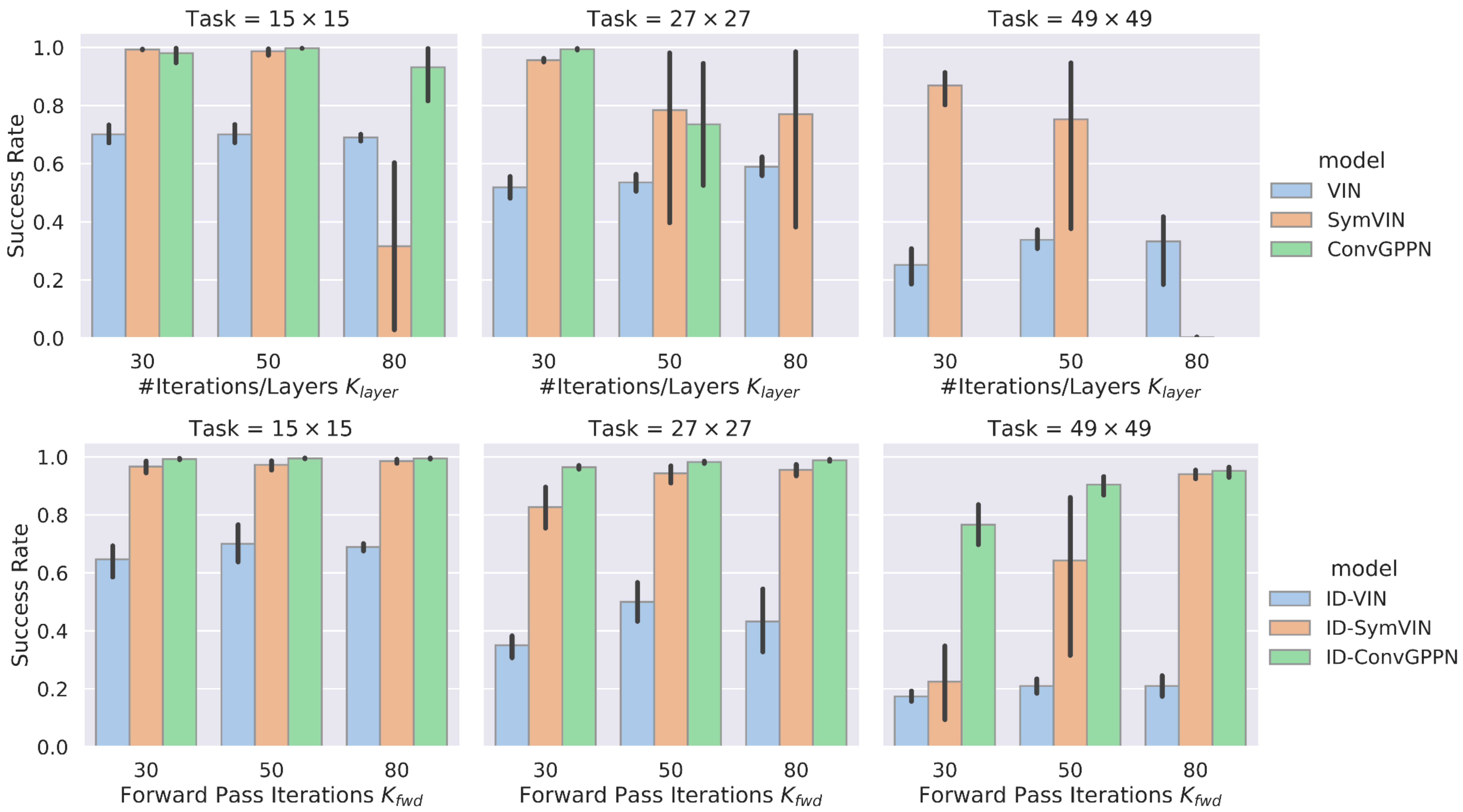
Results: Runtime on 2D Nav



Algorithmic Differentiable Planners

Implicit Differentiable Planners

Results: Success Rate



Summary of Contributions

- We apply implicit differentiation on VIN-based differentiable planning algorithms. This connects with deep equilibrium models (DEQ) (Bai et al., 2019).
- We propose a practical implicit planning pipeline and implement implicit version of VIN, as well as GPPN (Lee et al., 2018) and SymVIN (Zhao et al., 2022).
- We empirically study the convergence stability, scalability, and efficiency of the ADPs and proposed IDPs.

Thank you!

Check out our project website:

http://lfzhao.com/IDPlan

zhao.linf@northeastern.edu