Improving Adversarial Robustness of Deep Equilibrium Models with Explicit Regulations Along the Neural Dynamics

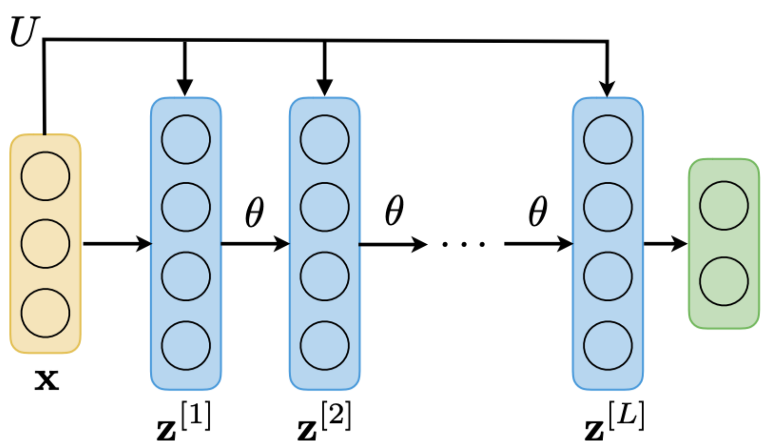
Zonghan Yang, Peng Li, Tianyu Pang, Yang Liu

Tsinghua University & Sea AI Lab




Deep Equilibrium Models (DEQs)

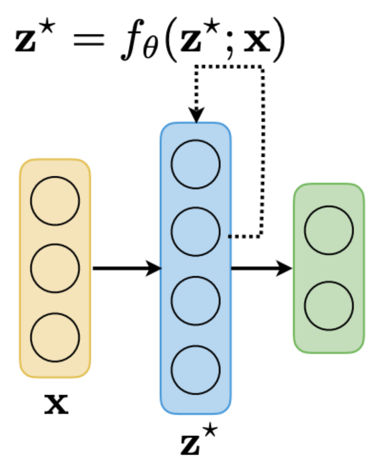
- Replace layer-wise propagation in conventional neural networks with fixed-point iteration



Weight-tied input-injected layer:

$$\mathbf{z}^{[i+1]} = f_\theta(\mathbf{z}^{[i]}; \mathbf{x}) = \sigma(W\mathbf{z}^{[i]} + U\mathbf{x} + b)$$

(just a simple example)



Deep Equilibrium (DEQ) Models:

Solve $\mathbf{z}^\star = f_\theta(\mathbf{z}^\star; \mathbf{x})$ (forward)

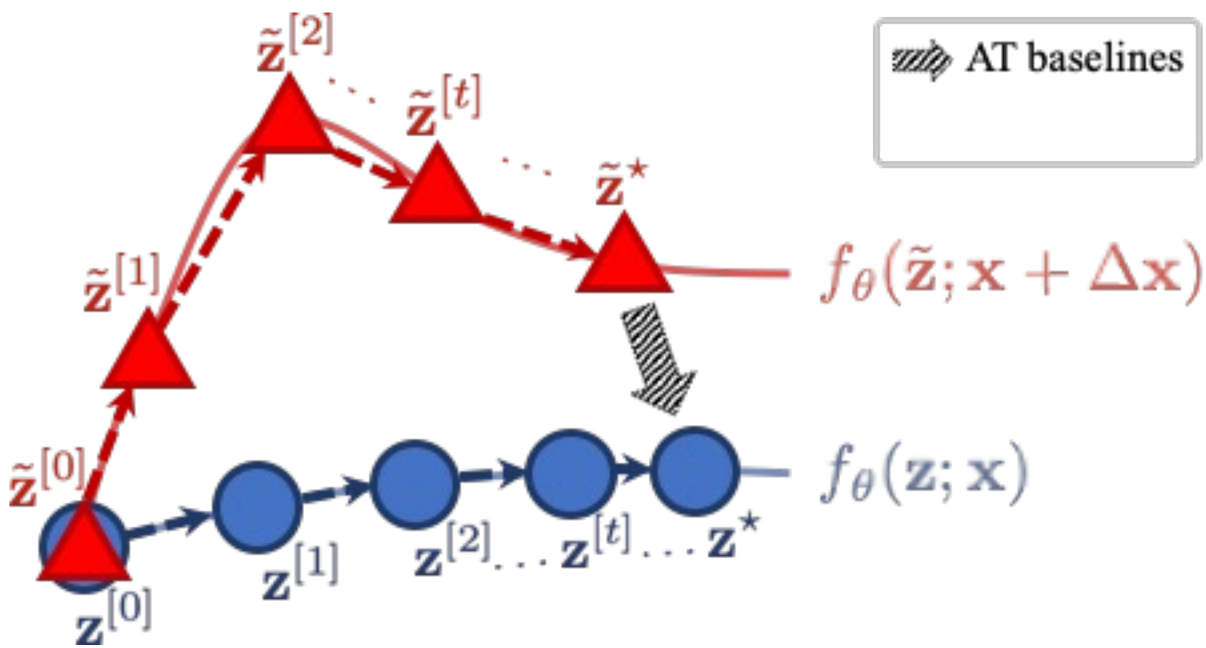
$$\frac{\partial \ell}{\partial (\cdot)} = -\frac{\partial \ell}{\partial \mathbf{z}^\star}\left(I - \frac{\partial f_\theta}{\partial \mathbf{z}^\star}\right)^{-1}\frac{\partial f_\theta}{\partial (\cdot)}$$ (backward)

- The nature of neural dynamics $\{\mathbf{z}^{[t]}\}$ in DEQ models

Figures from the poster of Bai et al., (2019)

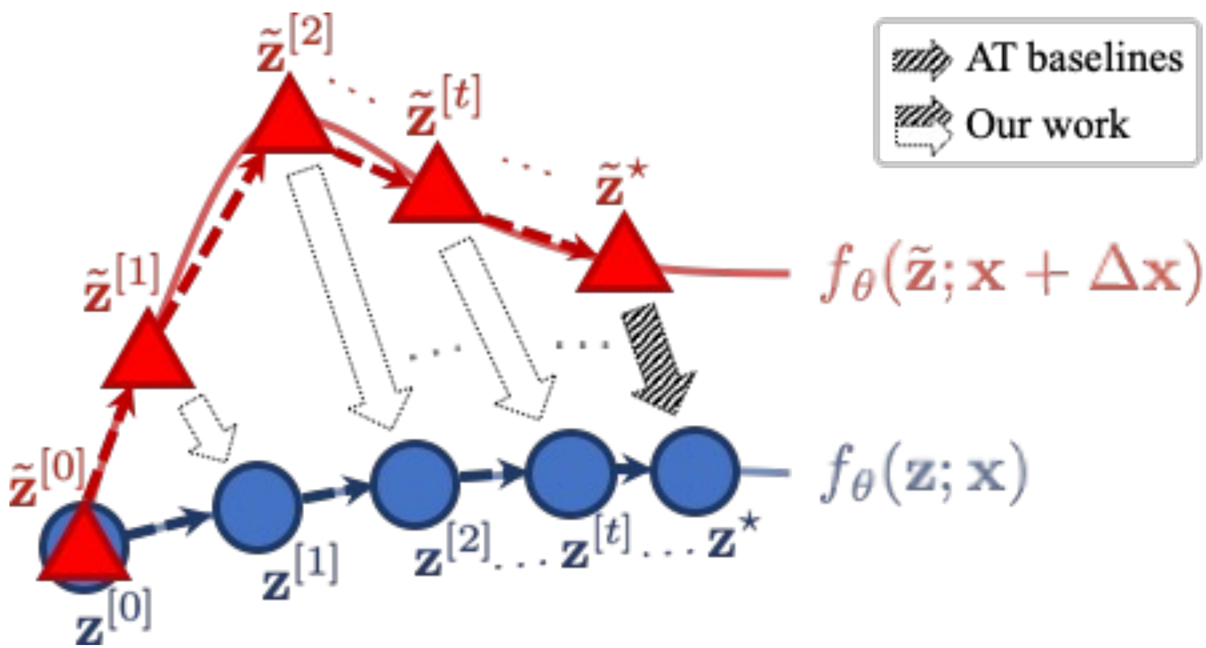
Robustness of DEQs

- Certificated robustness requires careful parameterization;
- Adversarial training for DEQs (Gurumurthy et al, 2021; Yang et al., 2022) shows inferior robustness performance compared with deep network counterparts



Robustness of DEQs

- Certificated robustness requires careful parameterization;
- Adversarial training for DEQs (Gurumurthy et al, 2021; Yang et al., 2022) shows inferior robustness performance compared with deep network counterparts
- Ours: AT + explicit regulations along the neural dynamics



The Deviation of Intermediate $\mathbf{z}^{[t]}$

- Assume that a clean input $\mathbf{x}$ induces $\{\mathbf{z}^{[t]}\}$, while a perturbed input $\mathbf{x} + \Delta\mathbf{x}$ induces $\{\tilde{\mathbf{z}}^{[t]}\}$. Since

$$\mathbf{z}^{[t+1]} = f_\theta(\mathbf{z}^{[t]}; \mathbf{x}), \quad \tilde{\mathbf{z}}^{[t+1]} = f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x} + \Delta\mathbf{x})$$

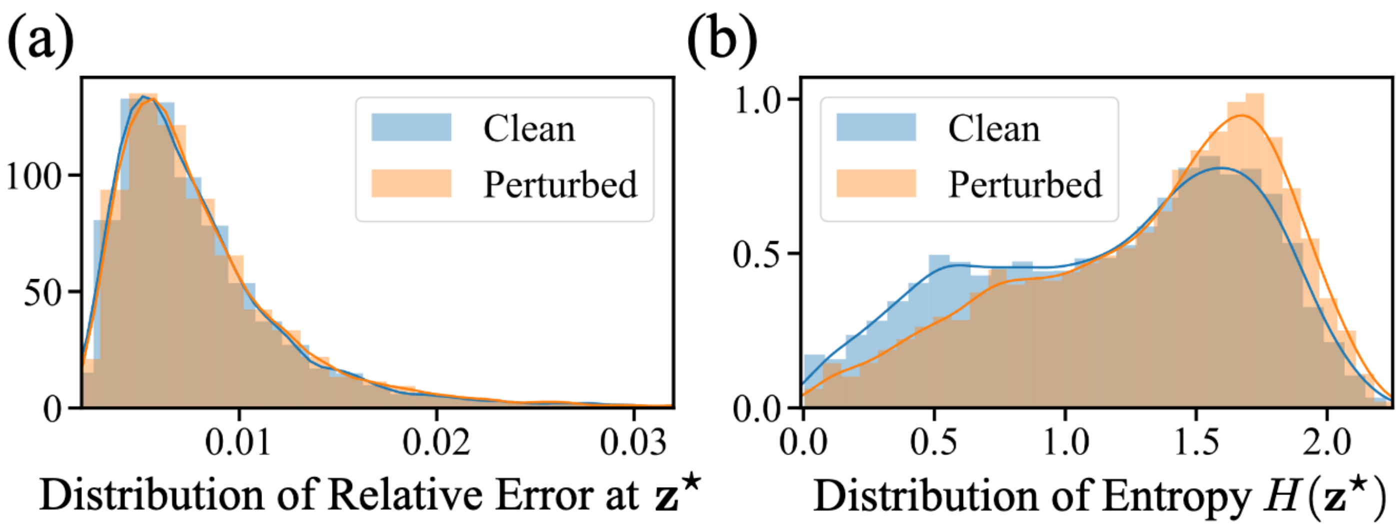
We have

$$\begin{aligned}
&\|\tilde{\mathbf{z}}^{[t+1]} - \mathbf{z}^{[t+1]}\| = \|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x} + \Delta\mathbf{x}) - f_\theta(\mathbf{z}^{[t]}; \mathbf{x})\| \\
&= \|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}+\Delta\mathbf{x}) - f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}) \; + \; f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}) - f_\theta(\mathbf{z}^{[t]}; \mathbf{x})\| \\
&\leq \underbrace{\|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}+\Delta\mathbf{x}) - f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x})\|}_{\text{Perturbation from } \mathbf{x}} + \underbrace{\|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}) - f_\theta(\mathbf{z}^{[t]}; \mathbf{x})\|}_{\text{Accumulation in } \mathbf{z}}.
\end{aligned}$$

- The deviation of neural dynamics is caused by (i) input perturbation and (ii) error accumulation

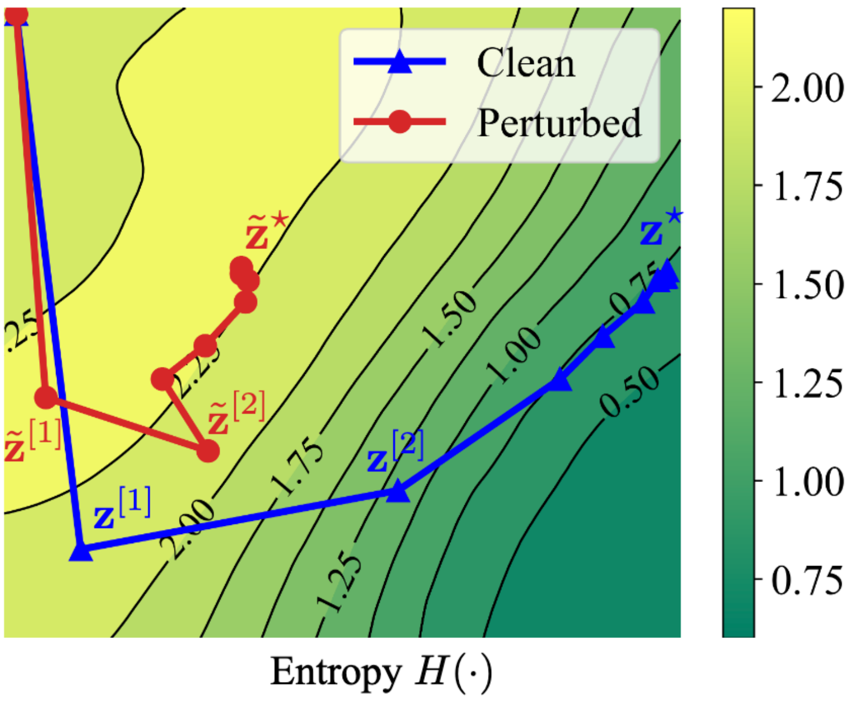
Input Entropy Reduction

- How to reduce the effect of input perturbation?
- Observation: perturbed inputs yield higher prediction entropy, although converging similarly
  - Drawing the distribution:



Input Entropy Reduction

- How to reduce the effect of input perturbation?
- Observation: perturbed inputs yield higher prediction entropy, although converging similarly
  - An exampled visualization along the neural dynamics:



Input Entropy Reduction

- How to reduce the effect of input perturbation?
- Observation: perturbed inputs yield higher prediction entropy, although converging similarly
- During inference, update the input *along the neural dynamics* by minimizing the prediction entropy

$$
\begin{aligned}
\min_{\mathbf{u}^{[1]},\cdots,\mathbf{u}^{[N]}} \quad & H(\mathbf{z}^{[N]}), \\
\text{s.t.} \quad & \mathbf{z}^{[t+1]} = \mathrm{Solve}\Big(\mathbf{z} = f_\theta(\mathbf{z}; \mathbf{x} + \mathbf{u}^{[t]});\ \mathbf{z}^{[\le t]}\Big), \\
& \mathbf{u}^{[t]} \in [-\epsilon, \epsilon]^l
\end{aligned}
$$

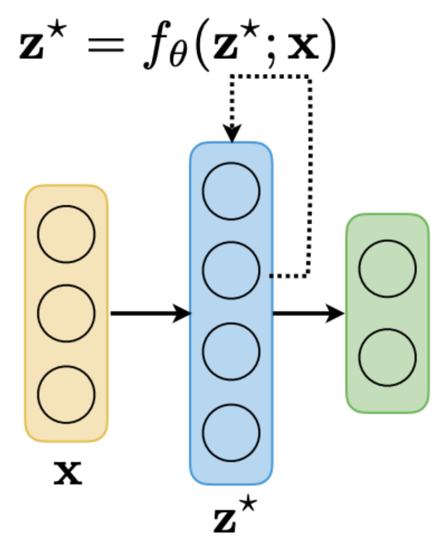
- $\{\mathbf{u}^{[t]}\}$ can be solved in the manner of iterative PGD

Adv. Loss from Random $\mathbf{z}^{[t]}$

- How to reduce the effect of error accumulation?

$$
\begin{aligned}
&\|\tilde{\mathbf{z}}^{[t+1]} - \mathbf{z}^{[t+1]}\| = \|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x} + \Delta\mathbf{x}) - f_\theta(\mathbf{z}^{[t]}; \mathbf{x})\| \\
&= \|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}+\Delta\mathbf{x}) - f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}) \;+\; f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}) - f_\theta(\mathbf{z}^{[t]}; \mathbf{x})\| \\
&\leq \underbrace{\|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}+\Delta\mathbf{x}) - f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x})\|}_{\text{Perturbation from } \mathbf{x}} + \underbrace{\|f_\theta(\tilde{\mathbf{z}}^{[t]}; \mathbf{x}) - f_\theta(\mathbf{z}^{[t]}; \mathbf{x})\|}_{\text{Accumulation in } \mathbf{z}}.
\end{aligned}
$$

- Observation: All intermediate $\mathbf{z}^{[t]}$s can be used to calculate the loss function
- Use random intermediate $\mathbf{z}^{[t]}$ to calculate adversarial loss to impose explicit regulations



Results and Future Work

- Higher adversarial robustness on CIFAR-10 compared to strong deep network baselines trained with AT

Architecture	Method	Clean	PGD	AA	All
ResNet-18	Pang et al. (2021)	**81.47**	-	49.14	49.14
DEQ-Large	Yang et al. (2022)	74.92	50.46	50.33	50.33
	+ Input Entropy Reduction	73.80	51.41	50.52	50.52
	+ Loss from Random $\mathbf{z}^{[t]}$	77.64	51.10	49.64	49.64
	+ Both	78.89	**55.18**	**51.50**	**51.50**

- Future work
  - Evaluation on larger benchmarks
  - Continue to exploit the structural properties of DEQs to design tailored adversarial defenses

Paper

Code