Learning (Approximately) Equivariant Networks via Constrained Optimization

Andrei Manolache[1,2,3], Luiz F.O. Chamon[4], Mathias Niepert[1,2]

1: Computer Science Department, University of Stuttgart, Germany

2: International Max Planck Research School for Intelligent Systems, Germany

3: Bitdefender, Romania

4: Department of Applied Mathematics, École Polytechnique de Paris, France

Equivariant functions transform data in a predictable way:

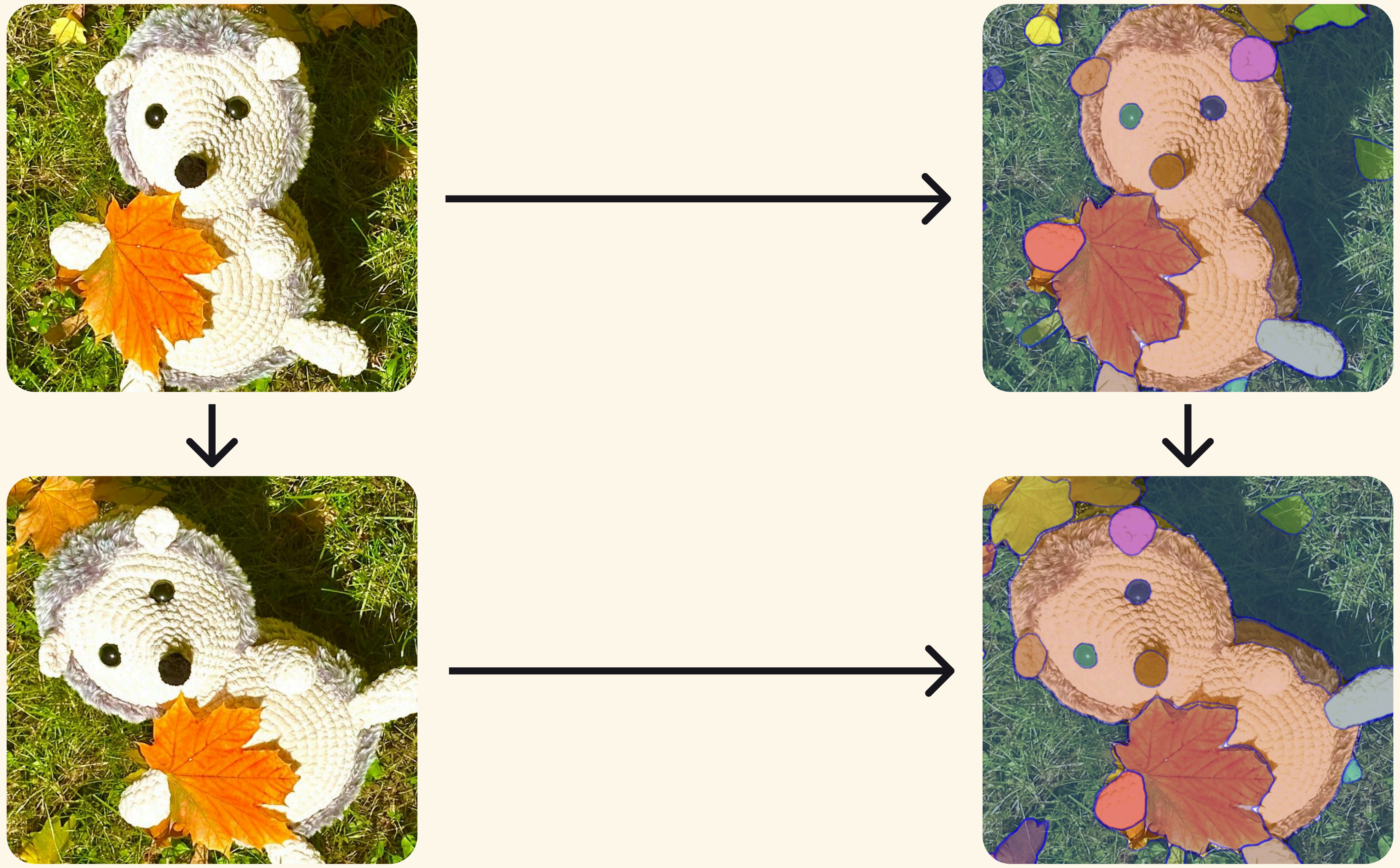
$$f_\theta\big(\rho_X(g)\,x\big) \;=\; \rho_Y(g)\,f_\theta(x) \qquad \forall\, g \in G.$$

Equivariant functions transform data in a predictable way:

$$f_\theta(\rho_X(g)\,x) = \rho_Y(g)\,f_\theta(x) \qquad \forall\, g \in G.$$

Equivariant functions transform data in a predictable way:

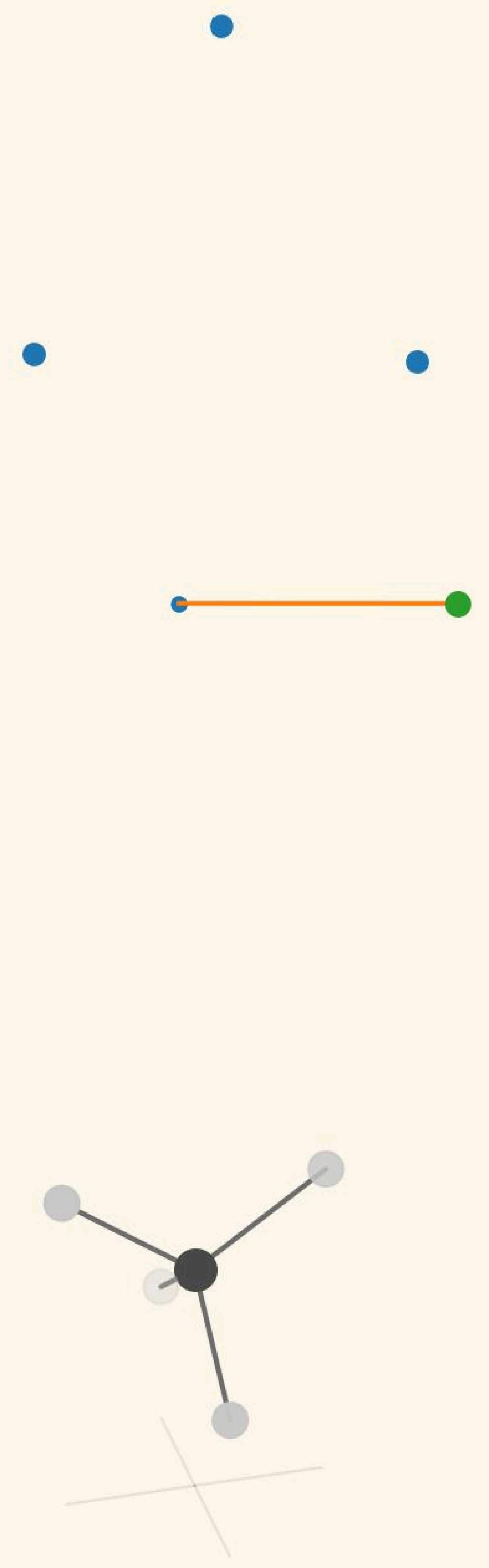
$$f_\theta\big(\rho_X(g)\,x\big) \;=\; \rho_Y(g)\,f_\theta(x) \qquad \forall\, g \in G.$$



Invariance is a special case: set $\rho_Y(g) = I$, *then* $f_\theta\big(\rho_X(g)\,x\big) \;=\; f_\theta(x) \qquad \forall\, g \in G.$

It’s useful to explicitly model symmetries:

- Better sample efficiency and lower parameter count
  - Hypothesis space is restricted
- Respects physics and improves interpretability
- Makes model robust to transformations



There’s a few catches:

- Empirically harder to optimize [1, 2]
- Misalignment risk
  - Symmetry-breaking phenomena
  - Noise, bias, missing structure in data
- Performance-equivariance tradeoff [3]
- Some layers break equivariance
  - Some CNNs are not translation equivariant! [4]

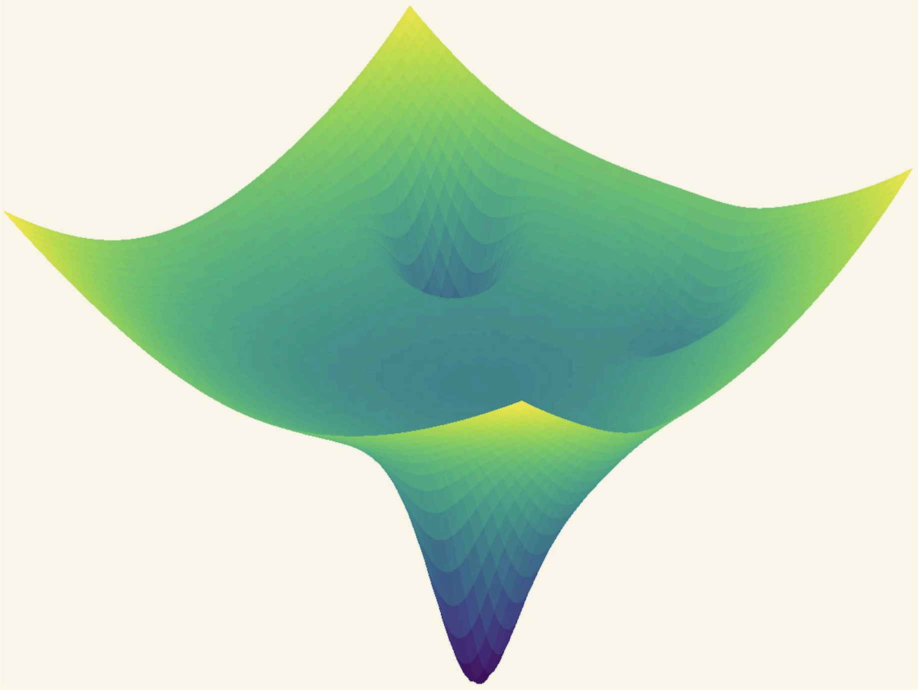


Fig. 1: Loss landscape, no equivariance

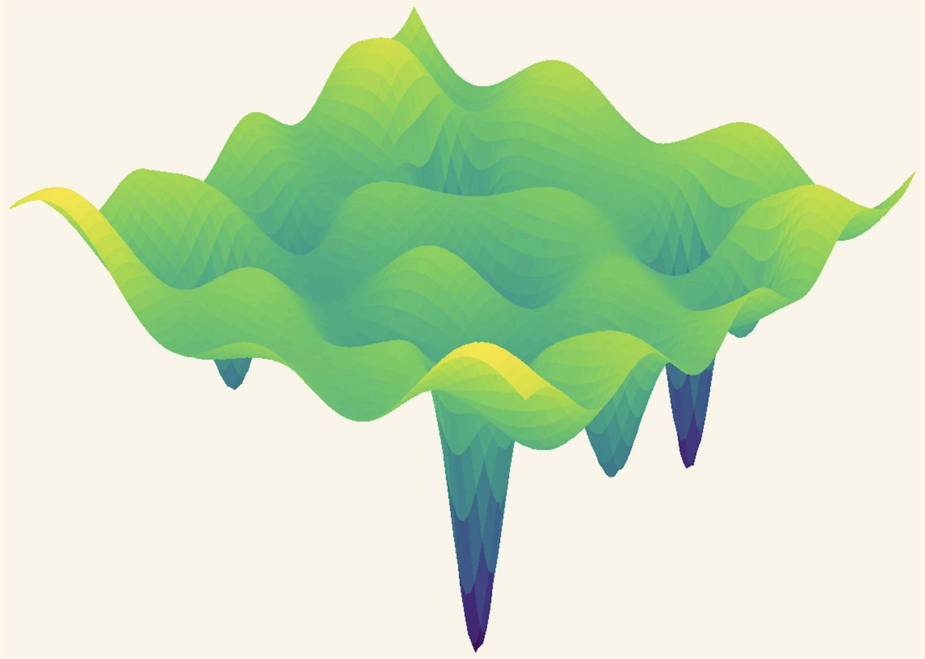


Fig. 2: Loss landscape, more equivariance

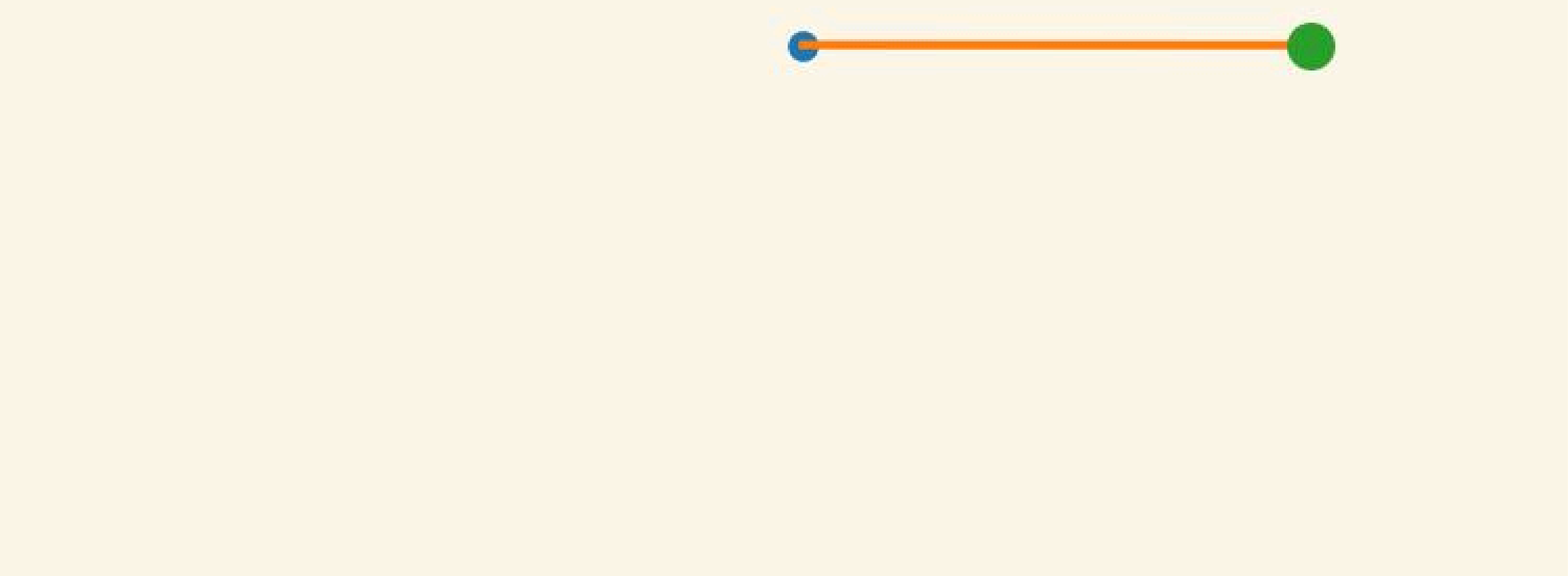


Fig. 3: Dampened pendulum breaks time symmetry

[1]: Elhag et al, “Relaxed Equivariance via Multitask Learning”, arXiv ‘24
[2]: Pertigkiozoglou et al, “Improving Equivariant Model Training via Constraint Relaxation”, NeurIPS ‘24
[3]: Petrache & Trivedi, “Approximation-Generalization Trade-offs under (Approximate) Group Equivariance”, NeurIPS ‘23
[4]: Azulay & Weiss, “Why do deep convolutional networks generalize so poorly to small image transformations?”, JMLR ‘19

WHY EQUIVARIANCE?

There's a few catches:

- Empirically harder to optimize [1, 2]
- Misalignment risk
    - Symmetry-breaking phenomena
    - Noise, bias, missing structure in data
- Performance-equivariance tradeoff [3]
- Some layers break equivariance
    - Some CNNs are not translation equivariant! [4]

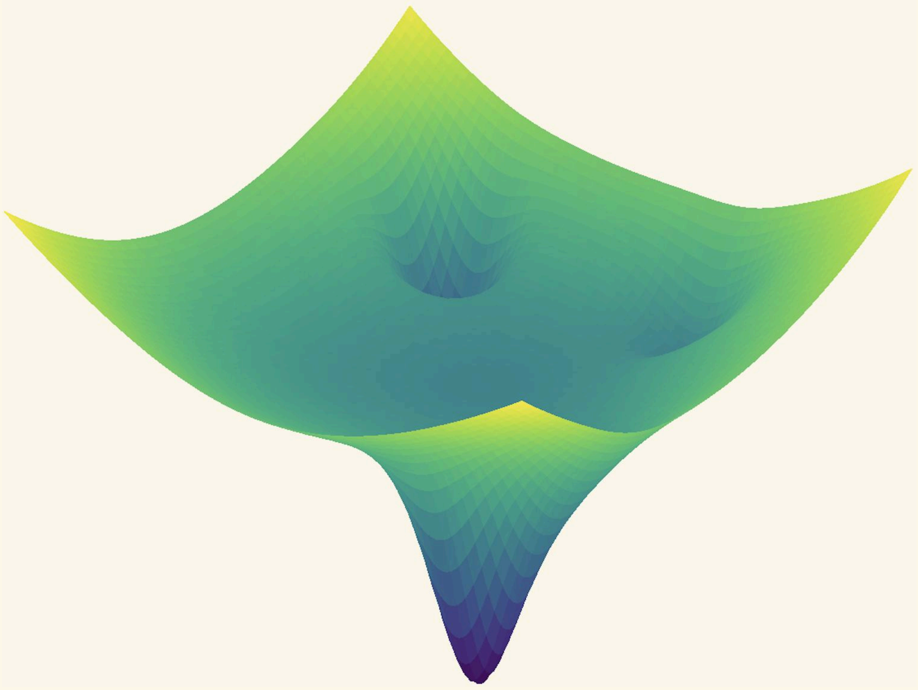


Fig. 1: Loss landscape, no equivariance

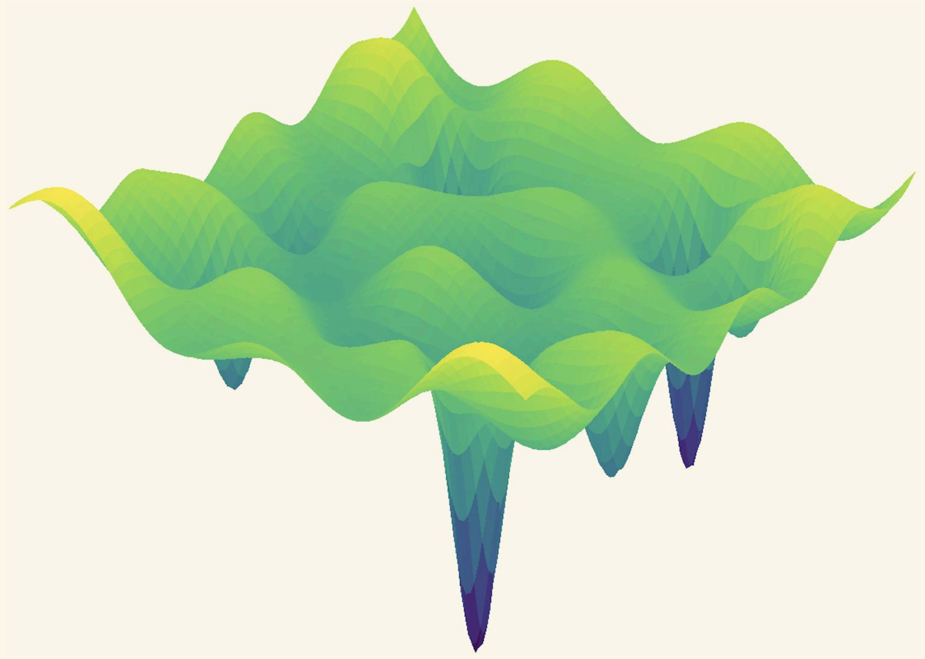


Fig. 2: Loss landscape, more equivariance

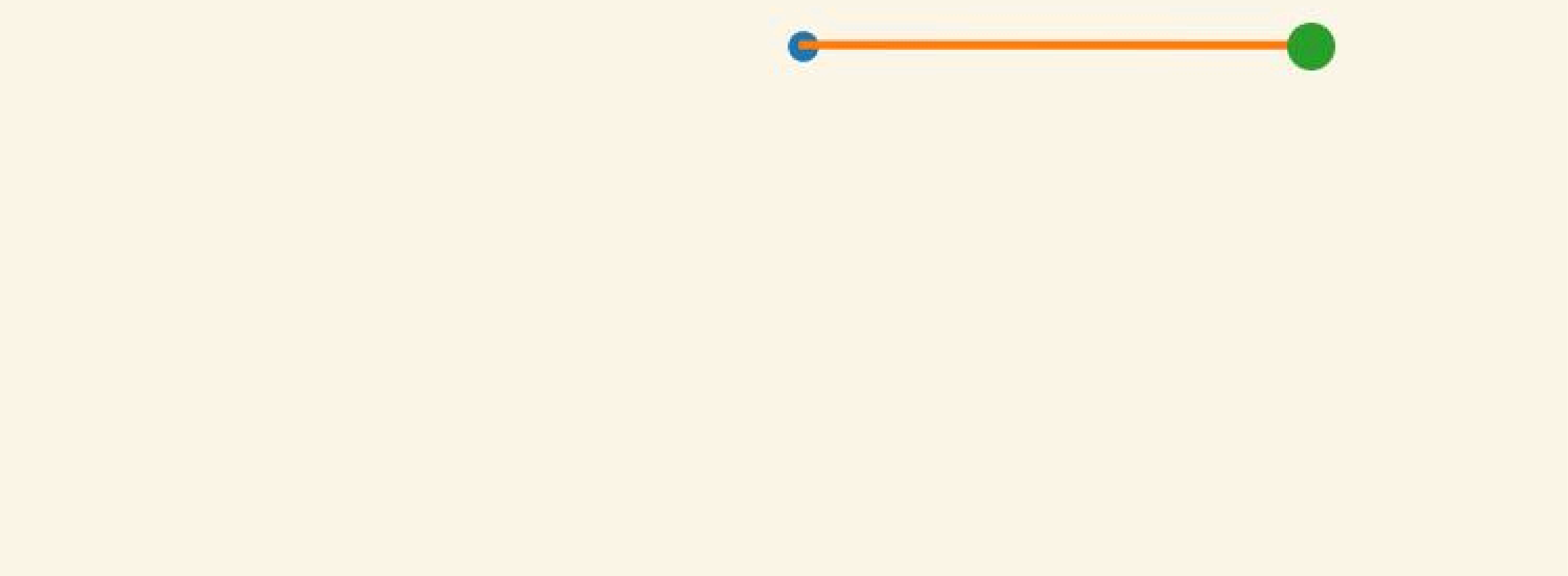
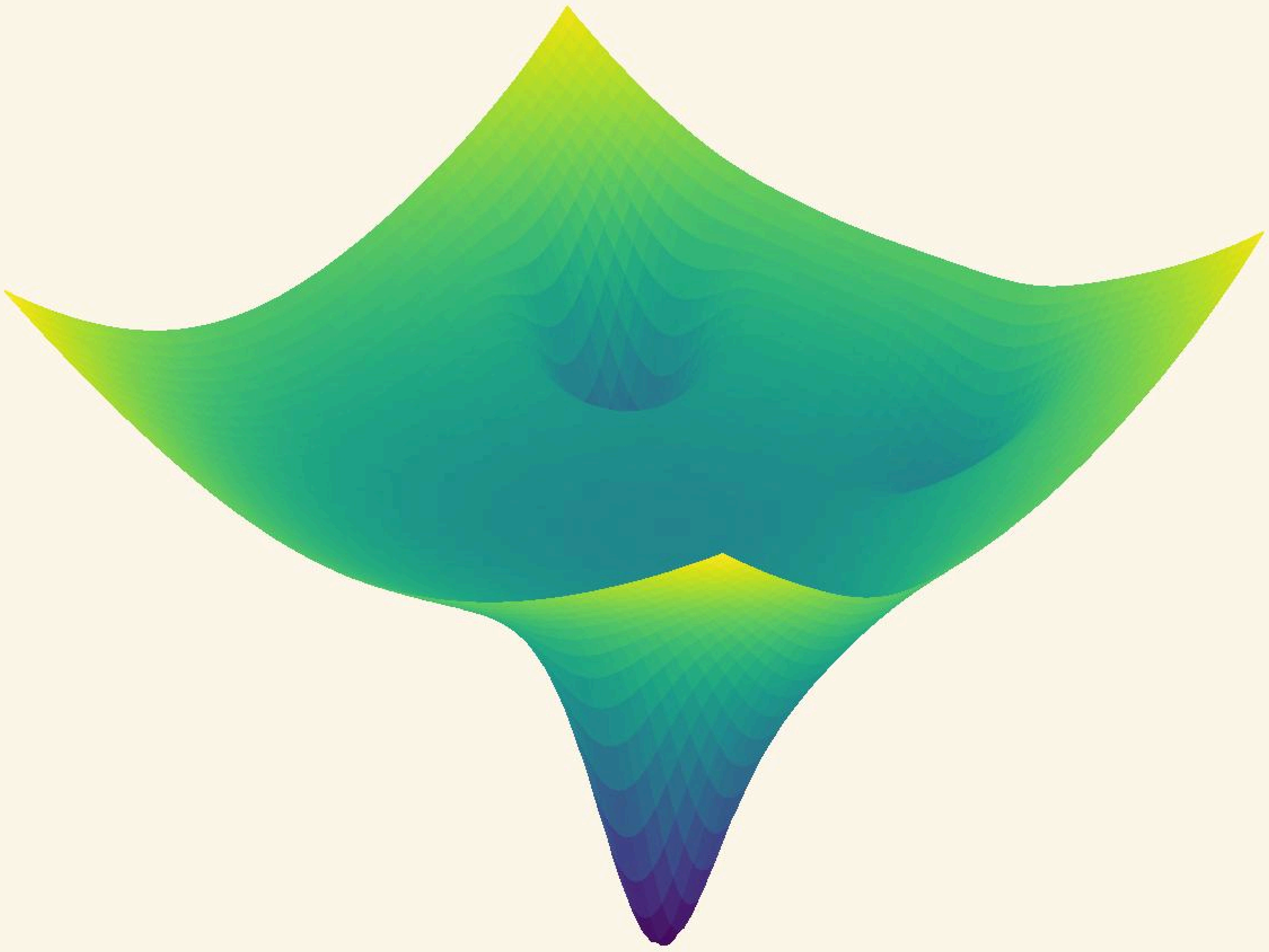


Fig. 3: Dampened pendulum breaks time symmetry

[1]: Elhag et al, "Relaxed Equivariance via Multitask Learning", arXiv '24
[2]: Pertigkiozoglou et al, "Improving Equivariant Model Training via Constraint Relaxation", NeurIPS '24
[3]: Petrache & Trivedi, "Approximation-Generalization Trade-offs under (Approximate) Group Equivariance", NeurIPS '23
[4]: Azulay & Weiss, "Why do deep convolutional networks generalize so poorly to small image transformations?", JMLR '19

Idea:
Start non-equivariant, enforce symmetries over time



Equivariant model:

$$f_\theta\big(\rho_X(g)\,x\big) \;=\; \rho_Y(g)\,f_\theta(x) \qquad \forall\, g \in G.$$

Homotopic architecture:

$$f_{\theta,\gamma} = f^L_{\theta,\gamma} \circ \cdots \circ f^1_{\theta,\gamma} \quad \text{with} \quad f^i_{\theta,\gamma} = f^{\mathrm{eq},i}_\theta + \gamma_i f^{\mathrm{neq},i}_\theta, \quad i = 1, \ldots, L$$

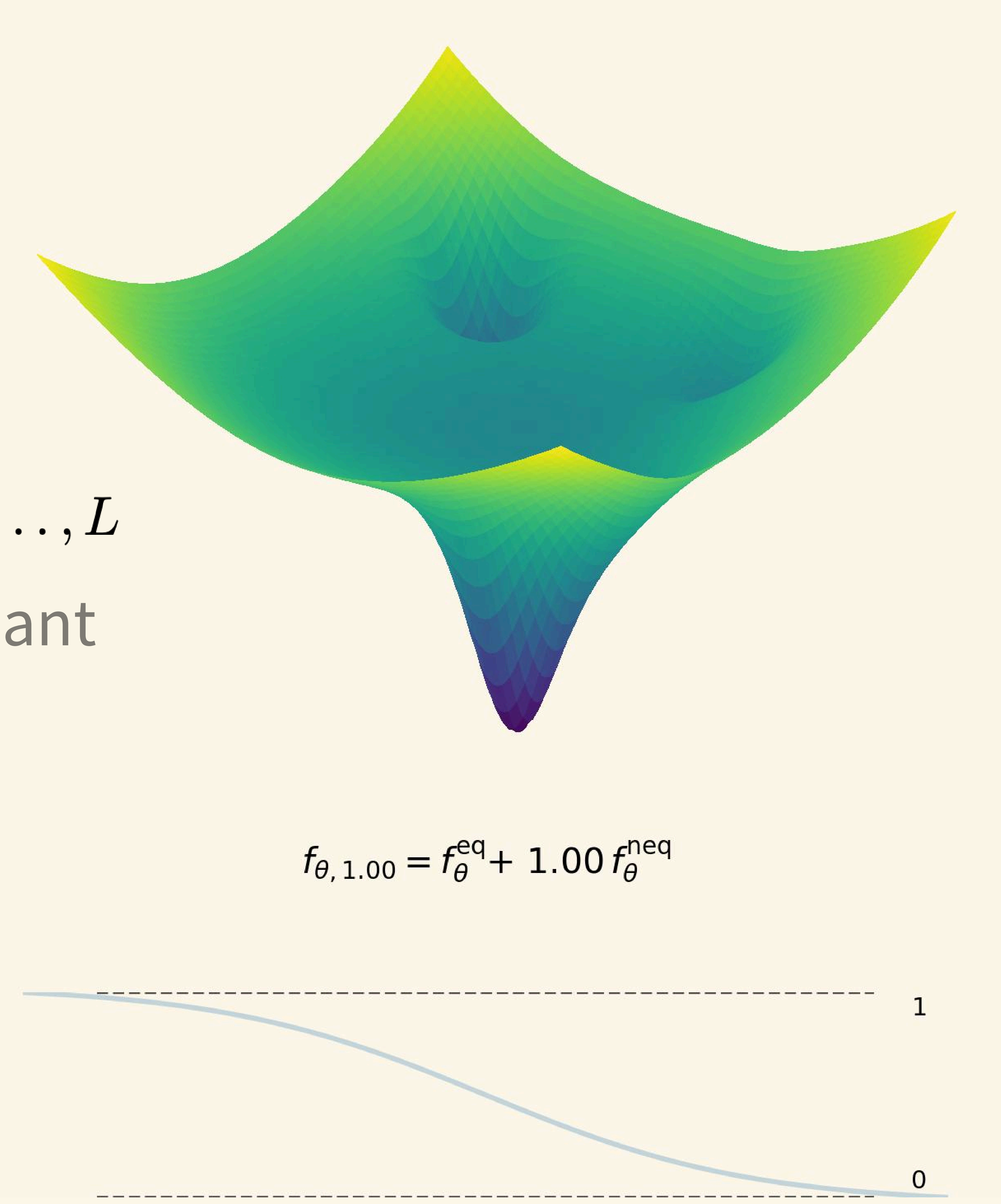
Equivariant model:

$$f_\theta(\rho_X(g)\,x) \;=\; \rho_Y(g)\,f_\theta(x) \qquad \forall\, g \in G.$$

Homotopic architecture:

$$f_{\theta,\gamma} = f^L_{\theta,\gamma} \circ \cdots \circ f^1_{\theta,\gamma} \quad \text{with} \quad f^i_{\theta,\gamma} = f^{\text{eq},i}_\theta + \gamma_i f^{\text{neq},i}_\theta, \quad i = 1, \ldots, L$$

$f_{\theta,0}$ is entirely equivariant, $f_{\theta,1}$ is the “least” equivariant



How do we control γ ?

Our approach: Adaptive Constrained Equivariance (ACE)

Our approach: Adaptive Constrained Equivariance (ACE)

- Write equivariance condition as an optimization constraint:

$$\begin{aligned} \underset{\theta,\gamma}{\text{minimize}} \quad & \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\ell_0\big(f_{\theta,\gamma}(x),y\big)\right] \\ \text{subject to} \quad & \gamma_i = 0, \quad \text{for } i = 1,\dots,L. \end{aligned}$$

Our approach: Adaptive Constrained Equivariance (ACE)

- Write equivariance condition as an optimization constraint:

$$\begin{aligned} \underset{\theta,\gamma}{\text{minimize}} \quad & \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\ell_0\big(f_{\theta,\gamma}(x), y\big)\right] \\ \text{subject to} \quad & \gamma_i = 0, \quad \text{for } i = 1, \ldots, L. \end{aligned}$$

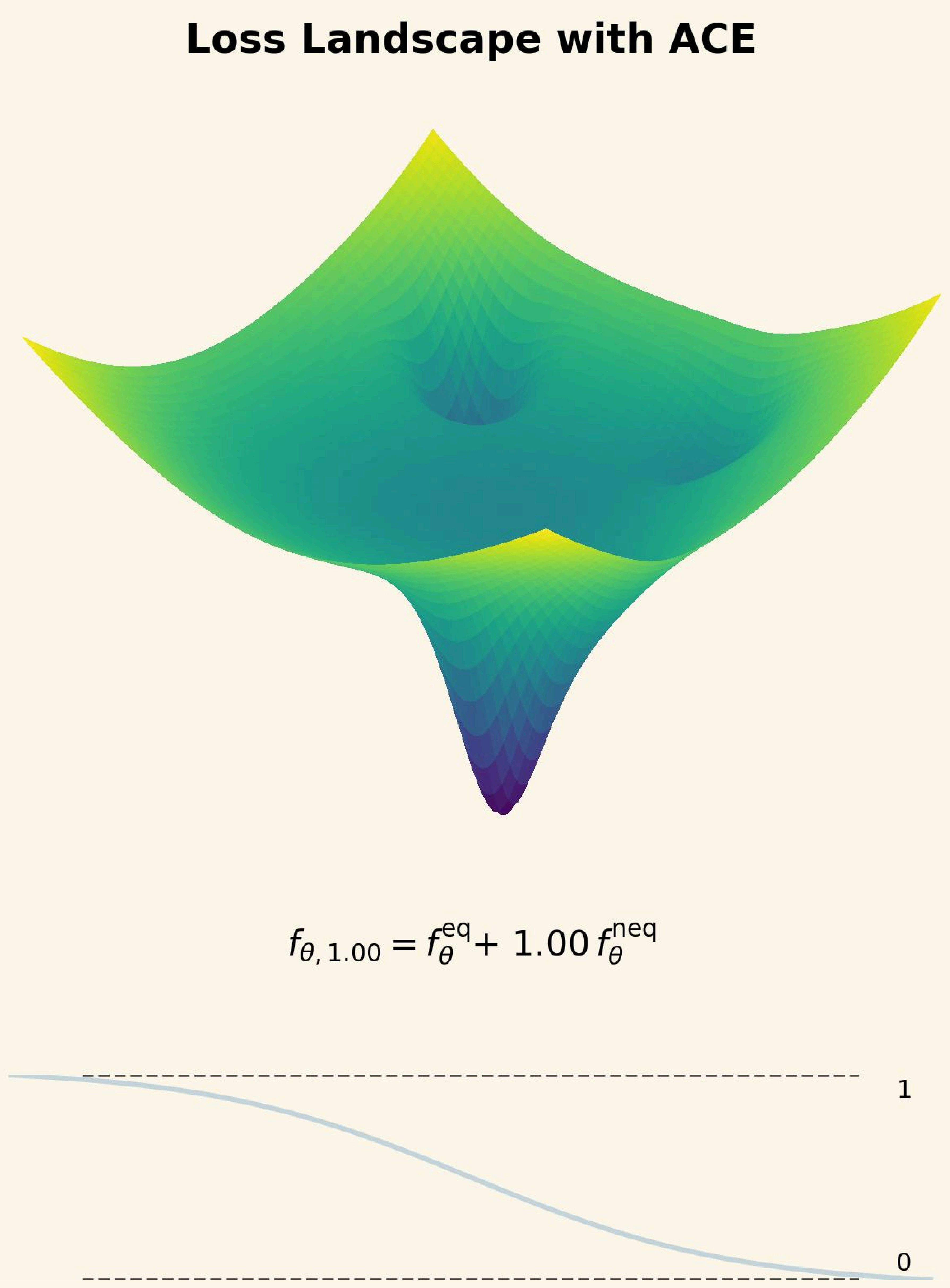
- We optimize the empirical dual problem:

$$\max_{\lambda\in\mathbb{R}^L} \min_{\theta,\gamma} \hat{L}(\theta,\gamma,\lambda) \triangleq \frac{1}{N}\sum_{n=1}^{N} \ell_0(f_{\theta,\gamma}(x_n), y_n) + \sum_{i=1}^{L} \lambda_i\gamma_i$$

Learning algorithm acts as an “annealing mechanism”:

1. Start with a flexible, non-equivariant model
2. Gradient descent seeks to reduce both the loss and solve the constraint
3. Slack magnitude (gradient ascent) depends on the constraint violation

Hence, gamma can either increase to expand flexibility, or decrease to exploit the symmetries in the data.



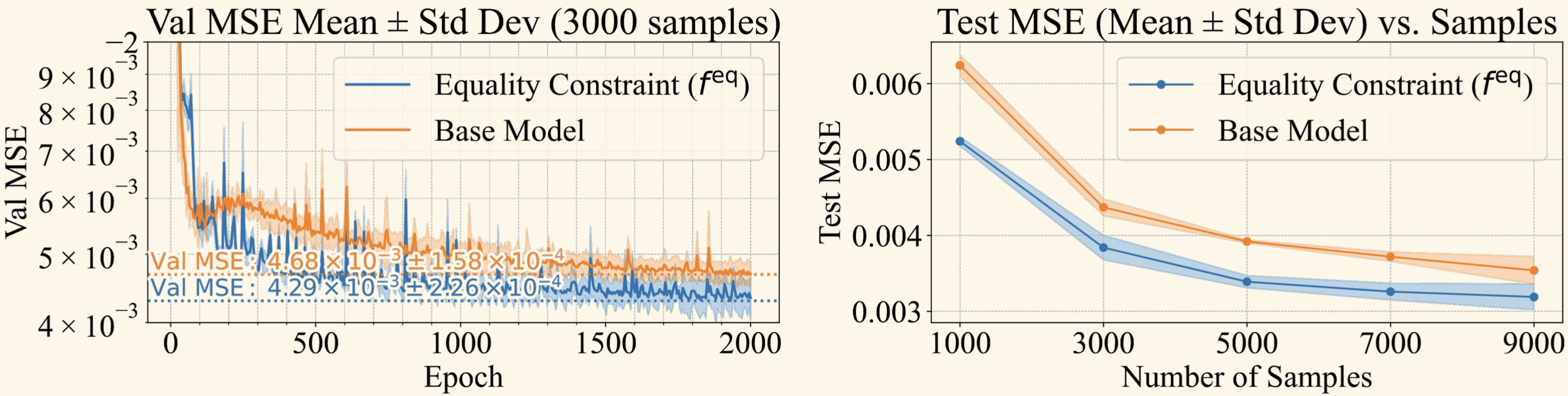
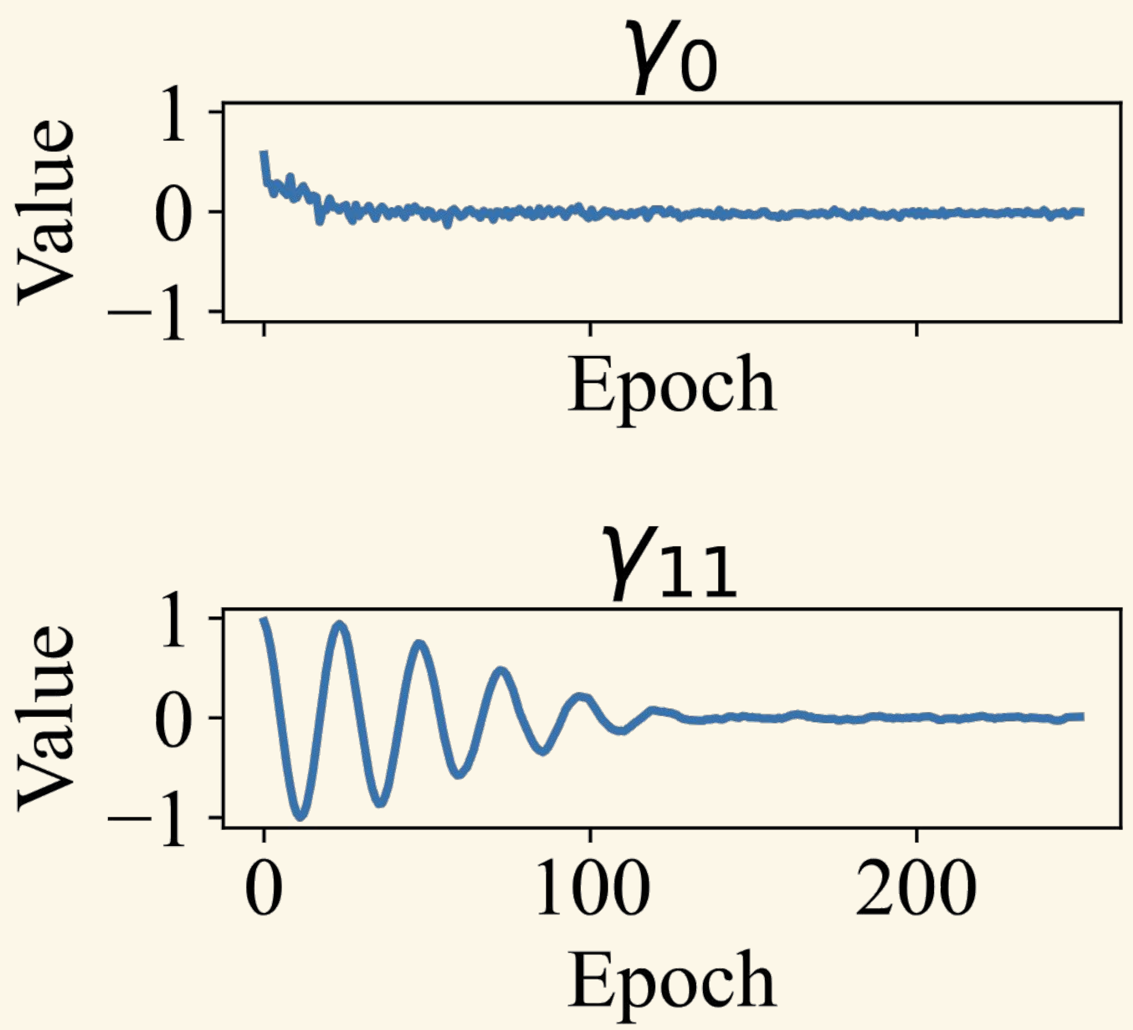
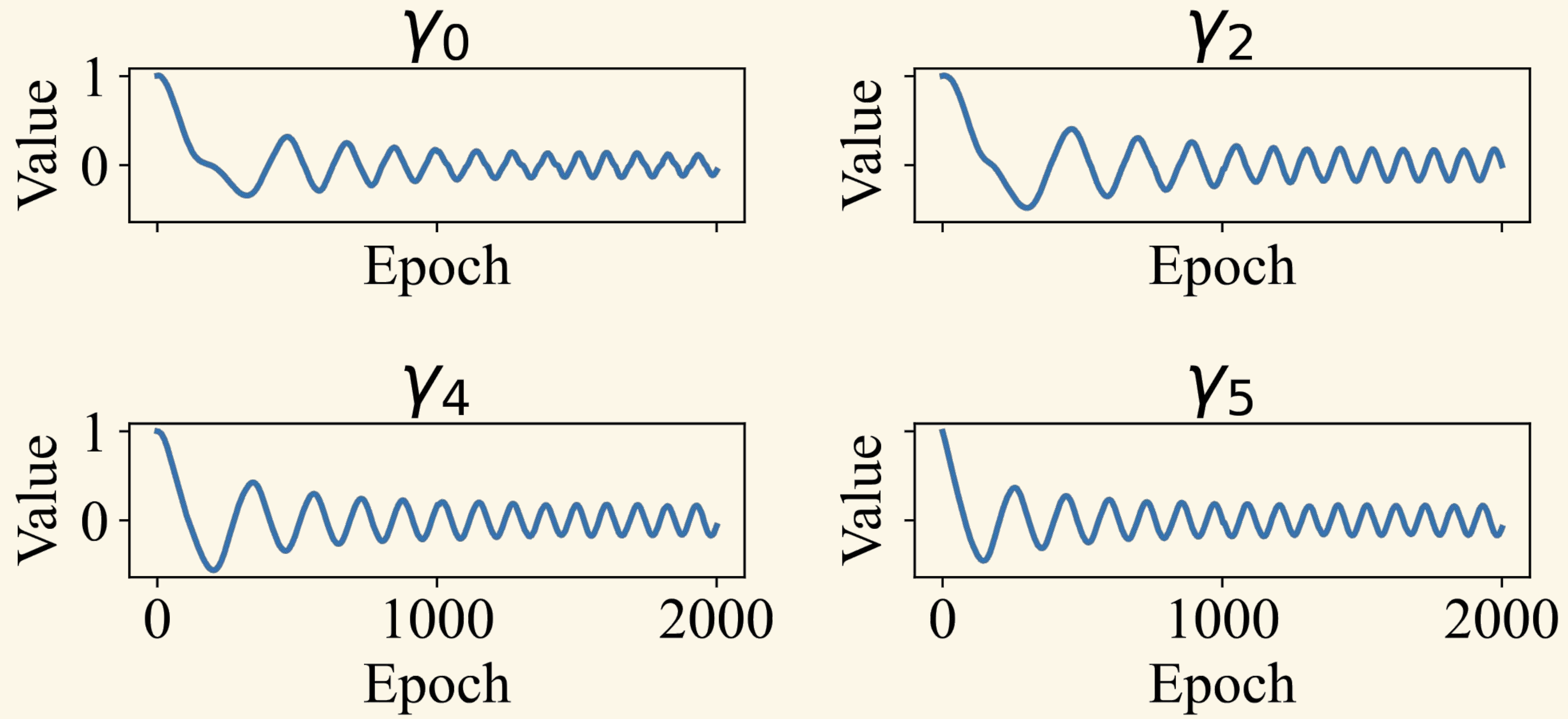


Figure 2: SEGNN trained with ACE equality constraints compared with the normal SEGNN on the N-Body dataset. **Left:** Validation MSE over 2000 epochs. **Right:** Test MSE versus training set size.

What if we want partial equivariance?

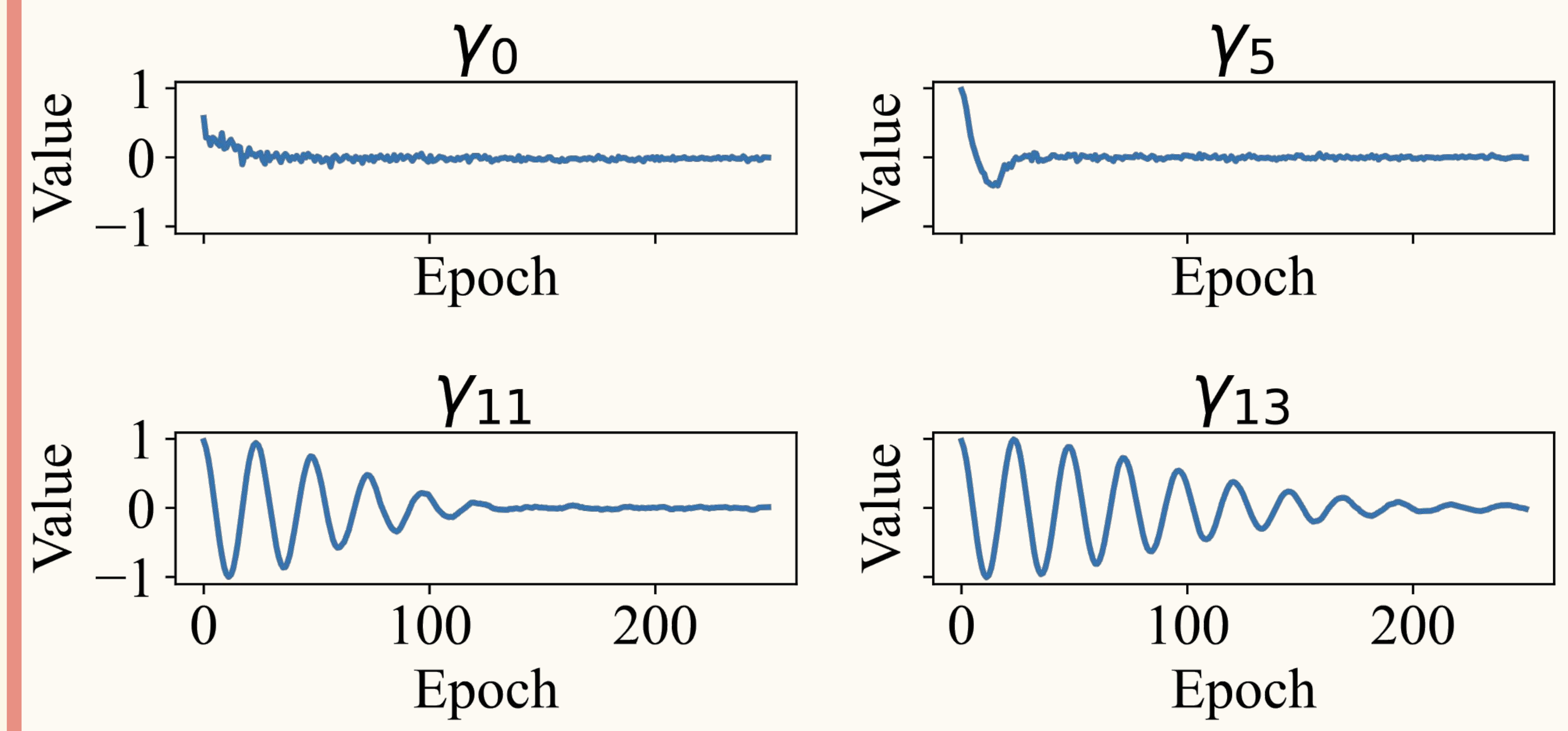


VN-DGCNN / ModelNet40

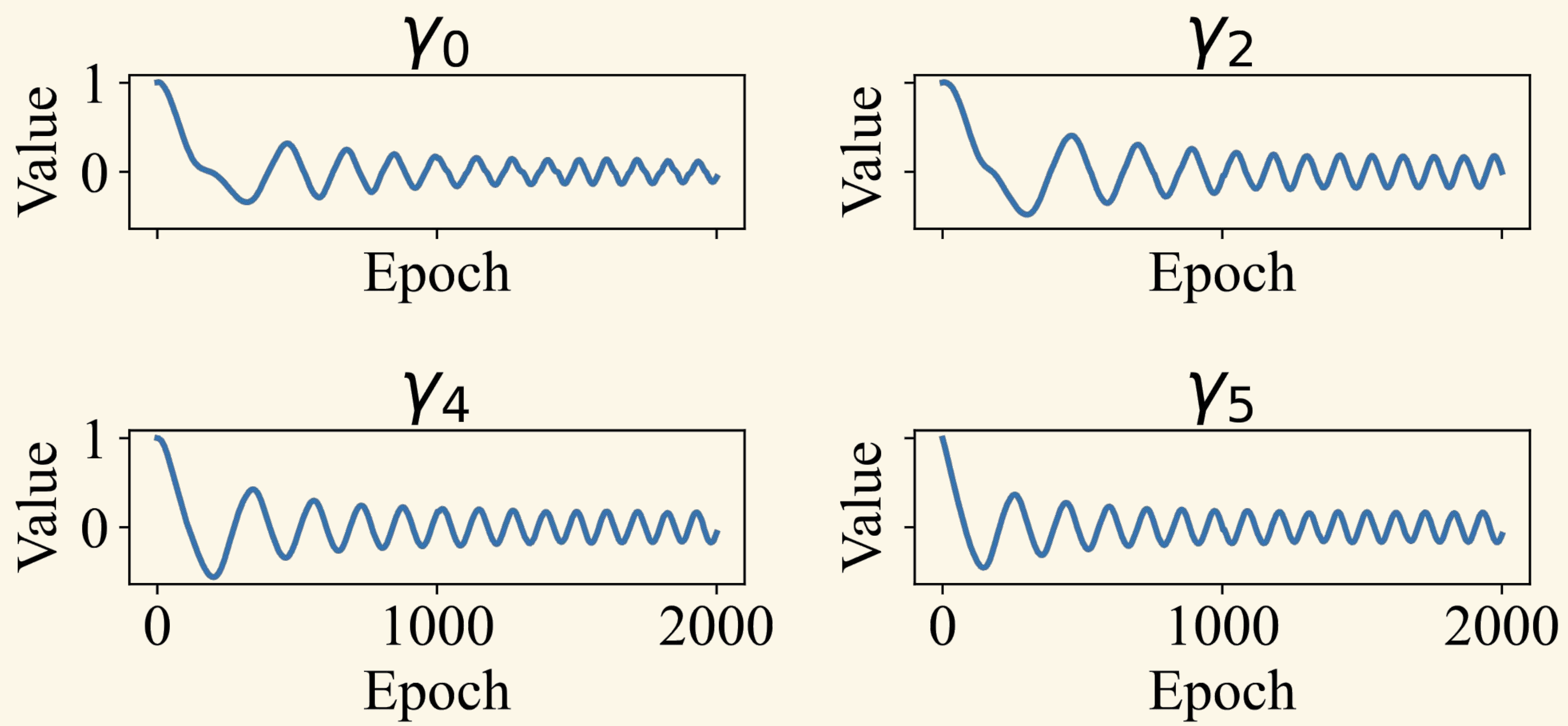


EGNO / CMU MoCap

What if we want partial equivariance?

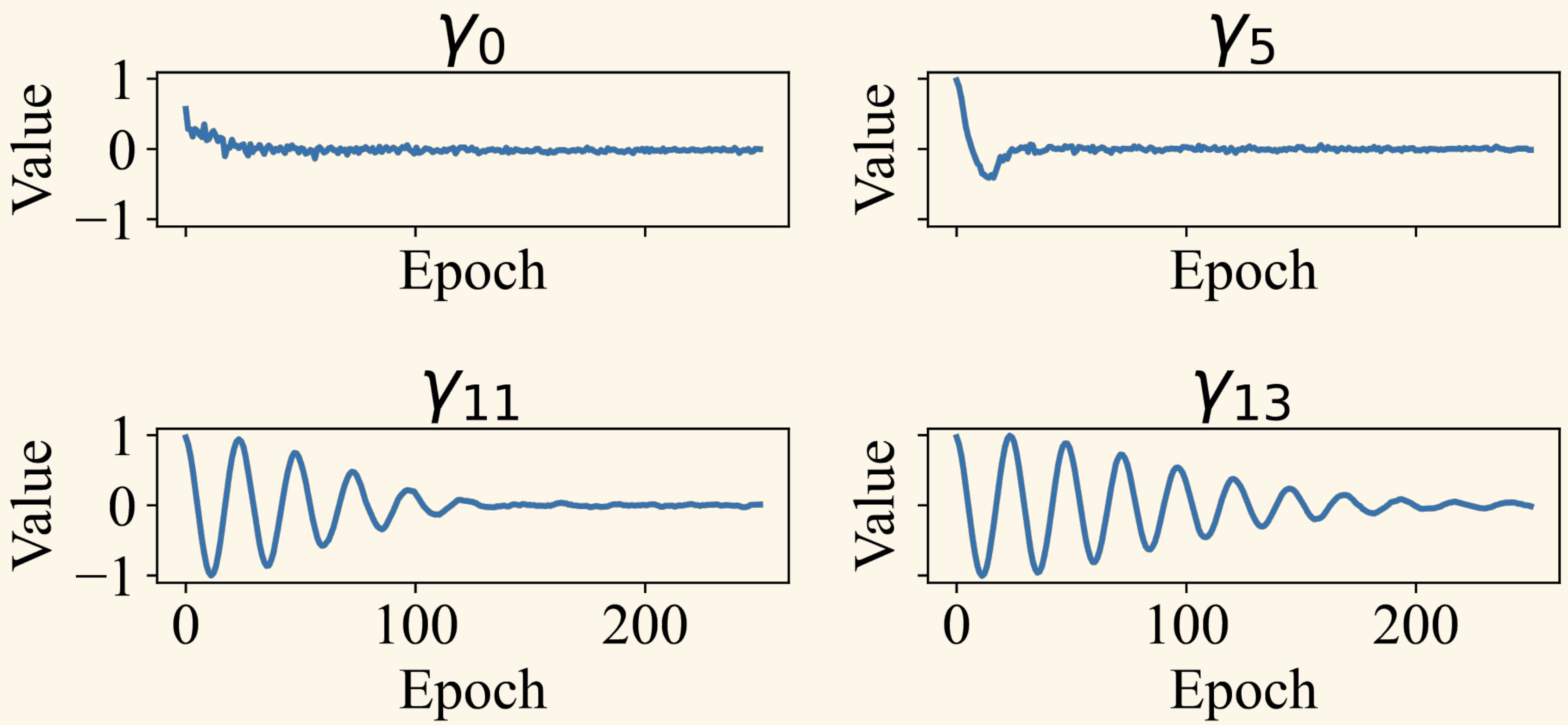


VN-DGCNN / ModelNet40

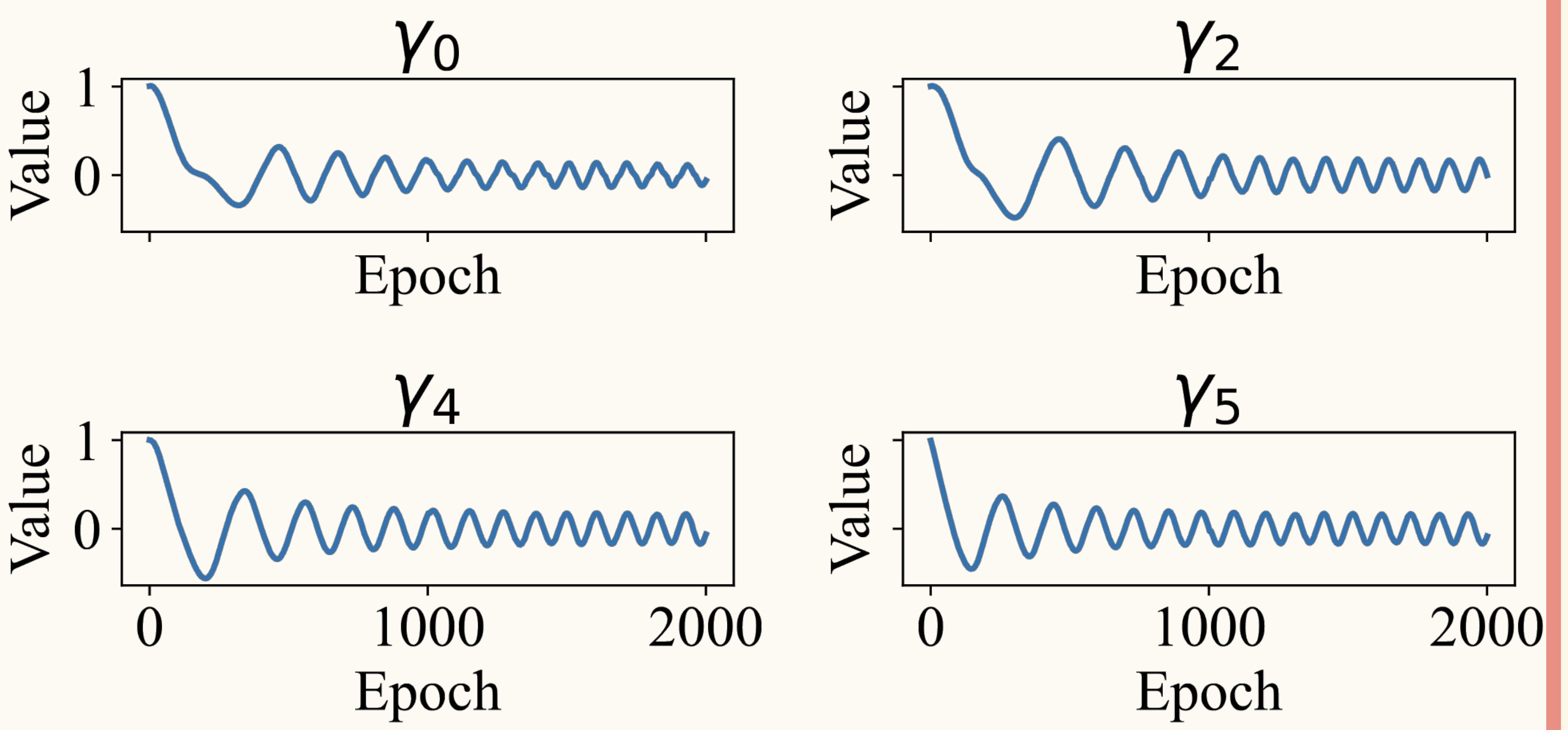


EGNO / CMU MoCap

What if we want partial equivariance?



VN-DGCNN / ModelNet40



EGNO / CMU MoCap

Resilient Constrained Learning [6]:

$$\begin{aligned} &\underset{\theta,\gamma,u}{\text{minimize}}\; \mathbb{E}_{(x,y)\sim\mathcal{D}}\Big[\ell_0\big(f_{\theta,\gamma}(x), y\big)\Big] + \frac{\rho}{2}\,\|u\|^2 \\ &\text{subject to } |\gamma_i| \le u_i, \quad \text{for } i = 1,\ldots,L. \end{aligned}$$

[6]: Hounie et al, "Resilient Constrained Learning", NeurIPS '23

Theorem (Simplified). *The equivariance error of our homotopic architecture grows at most linearly in the largest homotopy parameter* $\bar{\gamma}$*:*

$$\left\|\rho_Y(g) f_{\theta,\gamma}(x) - f_{\theta,\gamma}(\rho_X(g)x)\right\| \;\leq\; \mathcal{O}(\bar{\gamma}), \quad \forall g \in G.$$

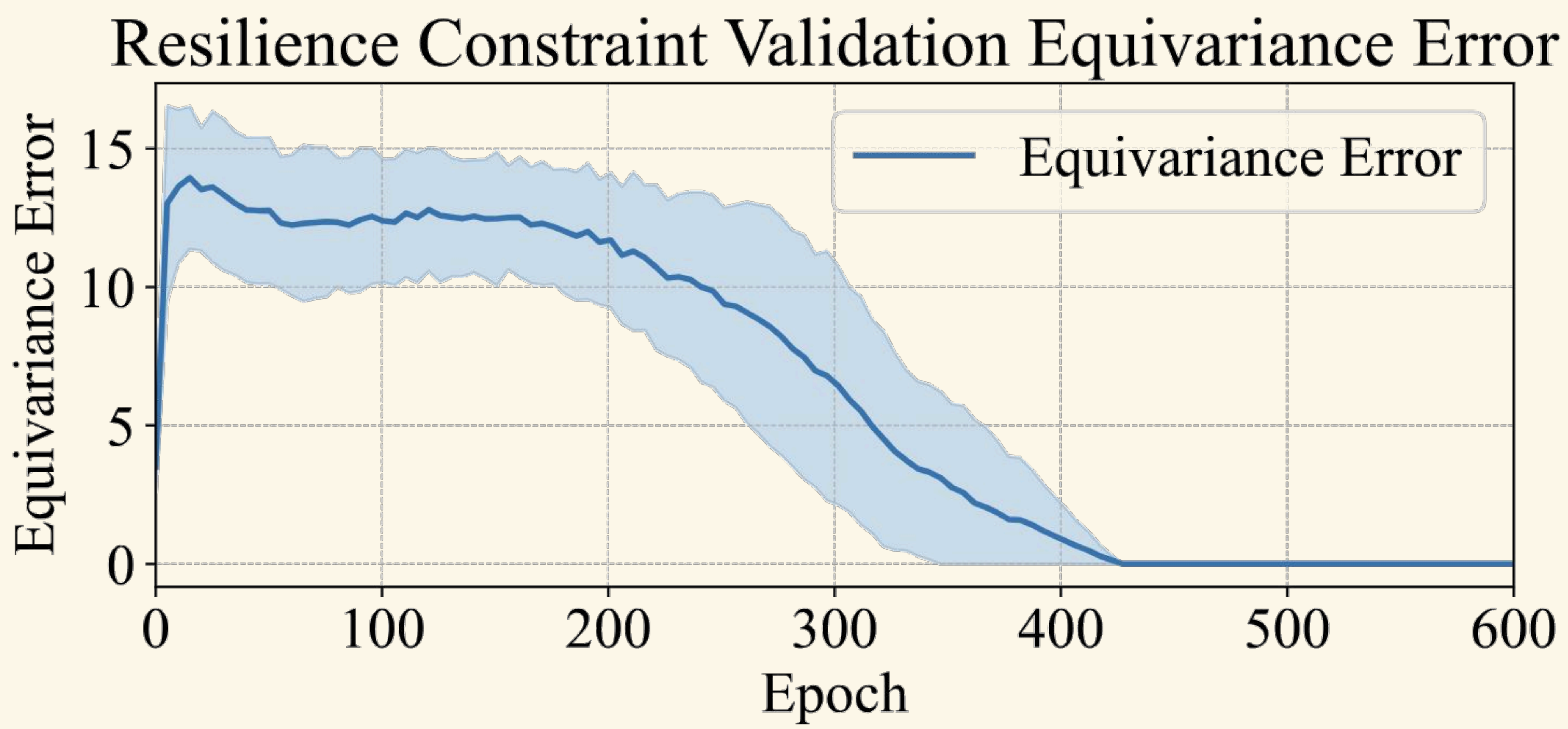


Figure 4: Validation equivariance error during training of a partially equivariant EGNO model with an ACE resilient inequality constraint on the CMU MoCap dataset (Subject #9, Run). The equivariance error decreases and approaches values near zero.

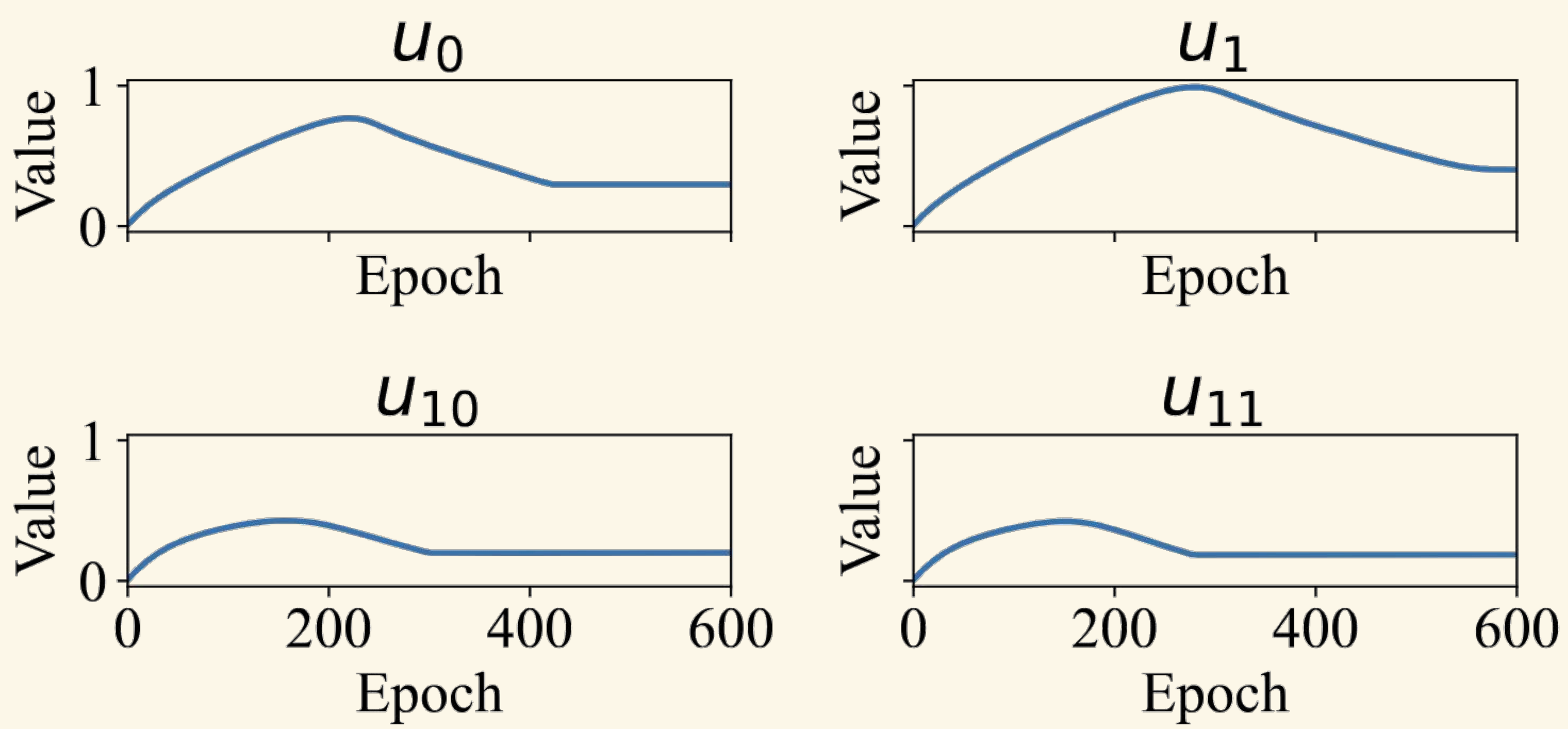


Figure 5: Adaptive constraint values u on the first two and last two layers of an EGNO model trained with ACE resilient inequality constraints on the CMU MoCap dataset (Subject #9, Run). Early layers allow large slacks that decrease over training, while later layers stay more tightly constrained.

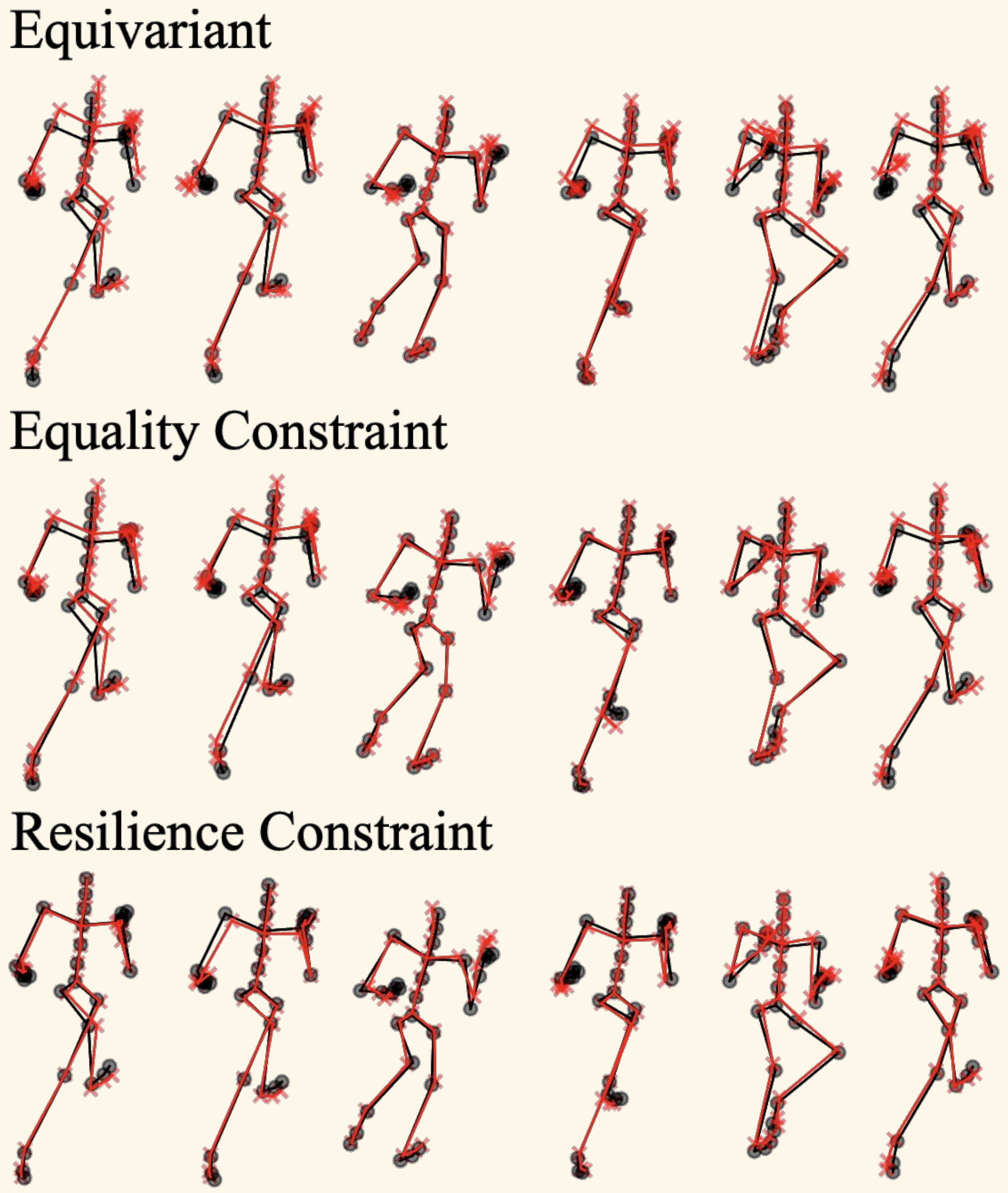


Figure 3: Qualitative comparison on the CMU MoCap dataset (Subject #9, Run). **Top row:** Standard equivariant EGNO. **Middle row:** Partially equivariant EGNO ($f^{\mathrm{eq}} + f^{\mathrm{neq}}$) trained with an ACE equality constraint. **Bottom row:** Partially equivariant EGNO ($f^{\mathrm{eq}} + f^{\mathrm{neq}}$) trained with an ACE resilient inequality constraint, yielding significant improvements over both.

Table 1: Test MSE ($\times 10^{-2}$) on the CMU MoCap dataset for Subject #9 (Run) and Subject #35 (Walk). Results are shown for the standard equivariant models, EGNO models trained with an equality constraint (ACE†) and an EGNO model trained with an resilient inequality constraint (ACE‡). The partially equivariant EGNO ($f^{\mathrm{eq}} + f^{\mathrm{neq}}$) outperforms both the standard and equality-constrained projection f^{eq} models, with the adaptive-resilience variant achieving the lowest error on both subjects.

Model	MSE↓ (Run)	MSE↓ (Walk)
MPNN [9]	66.4$\pm$2.2	36.1$\pm$1.5
RF [62]	521.3$\pm$2.3	188.0$\pm$1.9
TFN [13]	56.6$\pm$1.7	32.0$\pm$1.8
SE(3)-Tr.[42]	61.2$\pm$2.3	31.5$\pm$2.1
EGNN [43]	50.9$\pm$0.9	28.7$\pm$1.6
EGNO (report.) [46]	**33.9**$\pm$**1.7**	8.1$\pm$1.6
EGNO (reprod.) [46]	**35.3**$\pm$3.2	8.5$\pm$1.0
$\mathrm{EGNO}^{\mathrm{ACE}\dagger}_{f^{\mathrm{eq}}}$	**35.3**$\pm$1.6	**7.9**$\pm$**0.3**
$\mathrm{EGNO}^{\mathrm{ACE}\dagger}_{f^{\mathrm{eq}}+f^{\mathrm{neq}}}$	**32.6**$\pm$1.6	**7.5**$\pm$**0.3**
$\mathrm{EGNO}^{\mathrm{ACE}\ddagger}_{f^{\mathrm{eq}}+f^{\mathrm{neq}}}$	**23.8**$\pm$1.5	**7.4**$\pm$**0.2**

Conclusions:

1. Constrained Equivariance (ACE) can lead to better strictly equivariant models in a principled way.

Conclusions:

1. Constrained Equivariance (ACE) can lead to better strictly equivariant models in a principled way.
2. We can detect equivariance misalignment for free.

Conclusions:

1. Constrained Equivariance (ACE) can lead to better strictly equivariant models in a principled way.
2. We can detect equivariance misalignment for free.
3. We can relax equivariance by using resilient constrained learning

Conclusions:

1. Constrained Equivariance (ACE) can lead to better strictly equivariant models in a principled way.
2. We can detect equivariance misalignment for free.
3. We can relax equivariance by using resilient constrained learning

Open Questions:

1. Can we use constrained optimization to obtain (partially) equivariant models from non-equivariant architectures?

Conclusions:

1. Constrained Equivariance (ACE) can lead to better strictly equivariant models in a principled way.
2. We can detect equivariance misalignment for free.
3. We can relax equivariance by using resilient constrained learning

Open Questions:

1. Can we use constrained optimization to obtain (partially) equivariant models from non-equivariant architectures?
2. Is there a better way to add “lottery tickets”? Is there any other interesting properties we can obtain by using ACE?


Thank you!
Questions?