Learning (Approximately) Equivariant Networks via Constrained Optimization

Andrei Manolache[124]   **Luiz F.O. Chamon**[3]   **Mathias Niepert**[12]

[1]Computer Science Department, University of Stuttgart, Germany
[2]International Max Planck Research School for Intelligent Systems, Germany
[3]Department of Applied Mathematics, École Polytechnique de Paris, France
[4]Bitdefender, Romania
`{andrei.manolache, mathias.niepert}@ki.uni-stuttgart.de`
`luiz.chamon@polytechnique.edu`

Abstract

Equivariant neural networks are designed to respect symmetries through their architecture, boosting generalization and sample efficiency when those symmetries are present in the data distribution. Real-world data, however, often departs from perfect symmetry because of noise, structural variation, measurement bias, or other symmetry-breaking effects. Strictly equivariant models may struggle to fit the data, while unconstrained models lack a principled way to leverage partial symmetries. Even when the data is fully symmetric, enforcing equivariance can hurt training by limiting the model to a restricted region of the parameter space. Guided by homotopy principles, where an optimization problem is solved by gradually transforming a simpler problem into a complex one, we introduce *Adaptive Constrained Equivariance* (ACE), a constrained optimization approach that starts with a flexible, non-equivariant model and gradually reduces its deviation from equivariance. This gradual tightening smooths training early on and settles the model at a data-driven equilibrium, balancing between equivariance and non-equivariance. Across multiple architectures and tasks, our method consistently improves performance metrics, sample efficiency, and robustness to input perturbations compared with strictly equivariant models and heuristic equivariance relaxations.

1 Introduction

Equivariant neural networks (NNs) [1–4] leverage known symmetries to improve sample efficiency and generalization, and are widely used in computer vision [1, 3, 5–7], graph learning [8–11], and 3D structure modeling [12–17]. Despite their advantages, training these networks can be challenging. Indeed, their inherent symmetries give rise to complex loss landscapes [18] even when the model matches the underlying data symmetry [18, 19]. Additionally, real-world datasets frequently include noise, measurement bias, and other symmetry-breaking effects such as dynamical phase transitions or polar fluids [20–24] that undermine the assumptions of strictly equivariant models [25–27].

A well-established strategy for approaching difficult optimization problems is to use *homotopy* (or continuation), i.e., to first solve a simple, surrogate problem and then track its solutions through a sequence of progressively harder problems until reaching the original one [28, 29]. Introducing equivariance gradually can facilitate training by guiding the model through a series of less stringent intermediate optimization problems; however, implementations of this strategy based on modifying the training objective [18] or the underlying model [30] require domain-specific knowledge to design the intermediate stages and craft loss penalties, limiting practical usability.

This work, therefore, addresses the following question: *is there an automated, general-purpose strategy to navigate these intermediate problems that precludes manual tuning and that works effectively for various models and tasks?* To this end, we leverage the connection between homotopy approaches and dual methods in optimization, which both provide rigorous frameworks for solving constrained problems by addressing a sequence of relaxed problems with progressively stricter penalties. By casting the training process as a constrained optimization problem, we introduce *Adaptive Constrained Equivariance* (ACE), a principled framework for automatically relaxing/tightening equivariance in NNs. Our method balances downstream performance (e.g., accuracy) with equivariance, relaxing equivariance at initial stages while gradually transitioning to a strictly equivariant model. This transition occurs in an automated, data-driven fashion, eliminating manual tuning.

We summarize our contributions as follows:

1. **A general equivariant training framework.** We introduce ACE, a principled constrained optimization method to train equivariant models by automatically transitioning in a data-driven fashion from a flexible, non-equivariant model to one that respects symmetries without manually tuned penalties, weights, or schedules (Sec. 2).

2. **Accommodating partial equivariance.** Our data-driven method can identify symmetry-breaking effects in the data and automatically relax equivariance to mitigate their impact on downstream performance while exploiting the remaining partial equivariance (Sec. 2.2).

3. **Theoretical guarantees and empirical alignment.** We provide explicit bounds on the approximation error and equivariance level of the resulting model (Thm. A.1–A.2) and show how they are reflected in practice.

4. **Comprehensive empirical evaluation.** We conduct a systematic study across four representative domains to examine the impact of our algorithm on convergence, sample efficiency, and robustness to input degradation compared to strictly equivariant and relaxed alternatives.

2 Homotopic Equivariant Training

Homotopic architectures. Throughout this paper, we consider an architecture similar to [30] built by composing equivariant layers linearly perturbed by non-equivariant components. Formally, let

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

where $\gamma = (\gamma_1, \ldots, \gamma_L)$ are the *homotopic parameters*; $f^i : Z_{i-1} \to Z_i$ (and thus similarly for $f^{\text{eq},i}$ and $f^{\text{neq},i}$), with $Z_0 = X$ and $Z_L = Y$; and $f^{\text{eq},i}$ equivariant with respect to the actions ρ_i of a group G on Z_i, i.e., $f^{\text{eq},i}(\rho_{i-1}(g)z) = \rho_i(g)f^{\text{eq},i}(z)$ for all $g \in G$. Note that $\rho_0 = \rho_X$ and $\rho_L = \rho_Y$. We generally take $Z_i = \mathbb{R}^{k_i}$ and the group action ρ_i to be a map from G to the general linear group GL_{k_i} (i.e., the group of invertible $k_i \times k_i$ matrices). Immediately, the NN is equivariant for $\gamma = 0$, since it is a composition of equivariant functions f^{eq}. Otherwise, it is allowed to deviate from equivariance by an amount controlled by the magnitude of γ.

We address the challenges of equivariant training and previous relaxation methods, namely (i) the complex loss landscape induced by equivariant NNs ($\gamma = 0$); (ii) the manual tuning of penalties, weights, and/or homotopy schedules; (iii) the detection and mitigation of partial symmetries in the data. We use the insight that dual methods in constrained optimization are closely related to homotopy or simulated annealing in the sense that they rely on a sequence of progressively more penalized problems. This suggests that (i)–(ii) can be addressed in a principled way by formulating equivariant NN training as a constrained optimization problem (Sec. 2.1). We can then analyze the sensitivity of dual variables to detect the presence of symmetry violations in the data and relax the equivariance requirements on the final model [Sec. 2.2, (iii)]. We also characterize the approximation error and equivariance violations of the homotopic architecture (1) (Thm. A.1–A.2).

2.1 Strictly equivariant data: Equality constraints

We begin by considering the task of training a strictly equivariant NN, i.e.,

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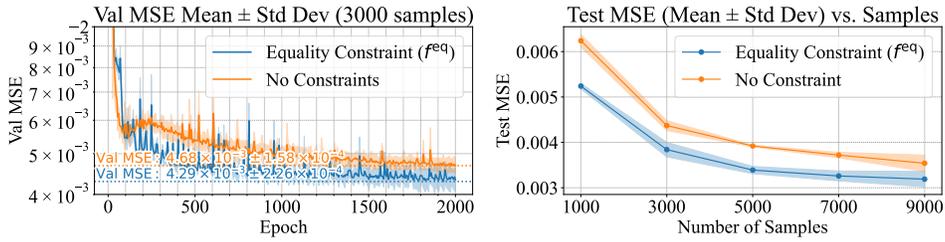\end{aligned} \tag{PI}$$

Figure 1: 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.

Indeed, only equivariant models $f_{\theta,0}$ are feasible for (PI), so that its solution must be equivariant. We can now consider its empirical dual problem

$$\underset{\lambda \in \mathbb{R}^L}{\text{maximize}} \; \underset{\theta, \gamma}{\min} \; \hat{L}(\theta, \gamma, \lambda) \triangleq \frac{1}{N} \sum_{n=1}^{N} \ell_0\big(f_{\theta,\gamma}(x_n), y_n\big) + \sum_{i=1}^{L} \lambda_i \gamma_i, \qquad \text{(DI)}$$

where (x_n, y_n) are i.i.d. samples from $\mathfrak{D}$, $\lambda \in \mathbb{R}^L$ collects the *dual variables* [31, 32] λ_i, and $\hat{L}$ is called the *empirical Lagrangian* [33, 34]. In general, solutions of (PI) and (DI) are not related due to (i) non-convexity, which hinders strong duality [35], and (ii) the empirical approximation of the expectation in (PI) by samples. Yet, for rich enough parametrizations $f_{\theta,\gamma}$, constrained learning theory shows that solutions of (DI) do generalize to those of (PI) [31–33]. In other words, (DI) approximates the solution of the equivariant NN training problem (PI) without constraining the model to be equivariant. More precisely, (DI) gives rise to a gradient descent (on θ, γ) and ascent (on λ) algorithm (Supplementary Alg. 1) that acts as an annealing mechanism adapted to the downstream task, taking into account the effect of enforcing equivariance on the objective function.

Naturally, the iterates $\gamma_i^{(t)}$ do not vanish in any finite number of iterations. In fact, the behavior of saddle-point algorithms such as Alg. 1 are intricate even in convex settings [36, 37] and its iterates may, even for small learning rates η_p, η_d, (i) asymptotically approach zero or (ii) display oscillations that do not subside. In case (i), if we stop Alg. 1 at iteration T and deploy its solution with $\gamma = 0$ (i.e., $f_{\theta^{(T)},0}$). This is not the case, however, when the γ_i oscillate without subsiding [case (ii)]. As we argue next, this effect can be used to identify and correct for symmetry deviations in the data.

2.2 Partially equivariant data: Resilient constraints

In some practical scenarios, the data used for training may only partially satisfy the equivariance relation due to the presence of noise, measurement bias, or other symmetry-breaking effects [20–27]. In such cases, the equivariance constraints in (PI) become too stringent. Alg. 1 will then oscillate as steps 4 and 5 push γ_i alternately closer and further from zero [case (ii)] or γ_i will settle, but λ_i will not vanish. These observations can then be used as evidences that the data does not fully adhere to imposed symmetries. This is not necessarily a sign that equivariance should be altogether abandoned, but that we can improve performance by relaxing it. While we could replace the equality in (PI) by $|\gamma_i| \leq u_i$ for $u_i \geq 0$, choosing a suitable set of $\{u_i\}$ is not trivial, since they must capture how much partial equivariance we expect or can tolerate at each layer. Setting the u_i too strictly degrades downstream performance (objective function ℓ_0), whereas setting them loosely interferes with the ability of the model to exploit whatever partial symmetry is in the data. Striking this balance manually is impractical, especially as the number of layers (and consequently γ_i) increases.

Instead, we use the outcomes of Alg. 1 to determine which constraints in (PI) to relax and by how much. We can then leverage the fact that stricter constraints lead to larger λ_i [31, 35] and relax each layer proportionally to their corresponding dual variable. In doing so, we reduce the number of $\gamma_i \neq 0$ as well as their magnitude, thus improving downstream performance (as measured by ℓ_0) while preserving equivariance as much as possible (Theorem A.2).

This trade-off rationale can be formalized and achieved automatically without repeatedly solving Alg. 1 by using *resilient constrained learning* [31, 32, 38]. Concretely, we replace (PI) by

$$\begin{aligned} \underset{\theta, \gamma, u}{\text{minimize}} \quad & \mathbb{E}_{(x,y) \sim \mathfrak{D}}\left[\ell_0\big(f_{\theta,\gamma}(x), y\big)\right] + \frac{1}{2}\|u\|^2 \\ \text{subject to} \quad & |\gamma_i| \leq u_i, \qquad \text{for } i = 1, \dots, L, \end{aligned} \qquad \text{(PII)}$$

where $u \in \mathbb{R}_+^L$ collects the *slacks* u_i. Notice that the slacks are not hyperparameters, but optimization variables that trade off between downstream performance, that improves by relaxing the constraints,

3

Figure 2: Evolution of γ in ACE equality constrained networks. **Left:** VN-DGCNN on ModelNet40; layers 0, 5, 11, 13 decay to zero, with deeper layers briefly oscillating, evidence that relaxing equivariance in later layers improves the final solution. **Right:** EGNO on CMU MoCap (Run); layers 0, 2, 4, 5 hover around zero without converging, showing that strict equivariance obtained from training with equality constraints is too restrictive.

and constraint violation (i.e., the level of equivariance). Using duality, (PII) lead to Supplementary Alg. 2. which balances the level of equivariance imposed on the model (u_i) and the relative difficulty of imposing this equivariance (λ_i) as opposed to improving downstream performance (J_0).

3 Experimental Setup and Results

In this section, we aim to answer the following two research questions:

RQ1. *Final Performance*: How does gradually imposing equivariance via ACE affect downstream task accuracy and sample efficiency across diverse domains and models?

RQ2. *Training Dynamics:* How do ACE constrained equivariance strategies, namely equality constrained optimization and partial equivariance enforced via resilient inequality constraints guide predictive performance when relaxing strict equivariance, and reveal where flexibility is most beneficial through their learned parameters?

To address **RQ1**, we begin with the N-Body simulations dataset [39]. In Figure 1 (left), we compare the validation MSE (mean±std) across 2000 epochs for the "vanilla" SEGNN and an equivariant projection f^{eq} obtained by training with ACE. Remarkably, f^{eq} trained with ACE converges more rapidly and attains a lower final MSE than the vanilla baseline, demonstrating both accelerated training dynamics and greater stability. Figure 1 (right) shows test MSE with respect to the training set size. Here, our f^{eq} trained on only 5000 samples matches (and slightly exceeds) the vanilla SEGNN's performance at 9000 samples, corresponding to a $\approx 44\%$ reduction in required data. To address **RQ2**, we evaluate predictive performance on the CMU Motion Capture (MoCap) "Run" and "Walk" datasets [40], where we examine the effect of training with both the equality constraint and the inequality constraint of resilient constrained learning. As shown in Supplementary Table 2, the equivariant projection f^{eq} performs similar to the vanilla EGNO. However, when using resilient inequality constraints we observe that the test MSE improves on both *Run* and *Walk* compared to the purely equivariant model and the ones trained with equality constraints. To better understand these effects, we inspect how the constrained parameters evolve during training. On ModelNet40, the modulation coefficients γ_k in the equality-constrained VN-DGCNN decay rapidly and reach zero within the first third of training (Figure 2, left). In the early layers, γ rapidly decays to zero; in the deeper layers, it oscillates around zero, indicating that the optimizer initially explores non-equivariant directions before settling on a strictly equivariant solution that aligns with the data symmetries. In contrast, the same coefficients in EGNO oscillate around 0 throughout training on the MoCap dataset, suggesting that strict equivariance may be overly restrictive for this task (Figure 2, right). We expand our empirical analysis and discuss theoretical alignment in Appendix A.4.

4 Conclusion

We introduced *Adaptive Constrained Equivariance* (ACE), a homotopy-inspired constrained equivariance training framework, that learns to enforce equivariance layer by layer via modulation coefficients γ. Theoretically, we show how our constraints can lead to equivariant models, deriving explicit bounds on the approximation error for fully equivariant models obtained via equality constraints and on the equivariance error for partially equivariant models obtained from inequality constraints. Across multiple datasets, ACE consistently improves accuracy, sample efficiency and convergence speed, while requiring no *ad hoc* schedules or domain-specific penalties. These findings suggest that symmetry can be treated as a "negotiable" inductive bias that is tightened when it helps generalization and relaxed when optimization requires flexibility, offering a practical middle ground between fully equivariant and unrestricted models.

4

References

[1] Taco S. Cohen and Max Welling. Group equivariant convolutional networks. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 2990–2999. JMLR.org, 2016. 1, 9

[2] Siamak Ravanbakhsh, Jeff Schneider, and Barnabás Póczos. Equivariance through parameter-sharing. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2892–2901. PMLR, 06–11 Aug 2017. URL `https://proceedings.mlr.press/v70/ravanbakhsh17a.html`.

[3] Taco S. Cohen and Max Welling. Steerable CNNs. In *International Conference on Learning Representations*, 2017. URL `https://openreview.net/forum?id=rJQKYt5ll`. 1

[4] Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2747–2755. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/kondor18a.html`. 1, 9

[5] Daniel E. Worrall, Stephan J. Garbin, Daniyar Turmukhambetov, and Gabriel J. Brostow. Harmonic networks: Deep translation and rotation equivariance. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7168–7177, 2016. URL `https://api.semanticscholar.org/CorpusID:206596746`. 1

[6] Risi Kondor, Zhen Lin, and Shubhendu Trivedi. Clebsch–gordan nets: a fully fourier space spherical convolutional neural network. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 10138–10147, Red Hook, NY, USA, 2018. Curran Associates Inc.

[7] Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=Hkbd5xZRb`. 1

[8] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. 1

[9] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, 2017. 15

[10] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[11] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. 1

[12] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 991–1001, 2017. 1, 9, 13, 14

[13] Nathaniel Thomas, Tess E. Smidt, Steven M. Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick F. Riley. Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. *ArXiv*, abs/1802.08219, 2018. URL `https://api.semanticscholar.org/CorpusID:3457605`. 9, 15

[14] J. Klicpera, F. Becker, and S. Günnemann. Gemnet: Universal directional graph neural networks for molecules. *CoRR*, 2021.

[15] Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gabor Csanyi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 11423–11436. Curran Associates, Inc., 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/4a36c3c51af11ed9f34615b81edb5bbc-Paper-Conference.pdf`.

[16] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[17] Viktor Zaverkin, Francesco Alesiani, Takashi Maruyama, Federico Errica, Henrik Christiansen, Makoto Takamoto, Nicolas Weber, and Mathias Niepert. Higher-rank irreducible cartesian tensors for equivariant message passing. In *Conference on Neural Information Processing Systems*, 2024. 1

[18] Ahmed Elhag, T. Rusch, Francesco Di Giovanni, and Michael Bronstein. Relaxed equivariance via multitask learning. 10 2024. doi: 10.48550/arXiv.2410.17878. 1, 9, 10, 11

[19] Mircea Petrache and Shubhendu Trivedi. Approximation-generalization trade-offs under (approximate) group equivariance. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=DnO6LTQ77U`. 1, 9, 10

[20] Nadav Dym and Haggai Maron. On the universality of rotation equivariant point cloud networks. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=6NFBvWlRXaG`. 1, 3, 10

[21] Chaitanya K. Joshi, Cristian Bodnar, Simon V. Mathis, Taco Cohen, and Pietro Liò. On the expressive power of geometric graph neural networks. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

[22] Yongjoo Baek, Yariv Kafri, and Vivien Lecomte. Dynamical symmetry breaking and phase transitions in driven diffusive systems. *Physical Review Letters*, 118(3), January 2017. ISSN 1079-7114. doi: 10.1103/physrevlett.118.030604. URL `http://dx.doi.org/10.1103/PhysRevLett.118.030604`. 9

[23] Simon A. Weidinger, Markus Heyl, Alessandro Silva, and Michael Knap. Dynamical quantum phase transitions in systems with continuous symmetry breaking. *Phys. Rev. B*, 96:134313, Oct 2017. doi: 10.1103/PhysRevB.96.134313. URL `https://link.aps.org/doi/10.1103/PhysRevB.96.134313`.

[24] Calum J. Gibb, Jordan Hobbs, Diana I. Nikolova, Thomas Raistrick, Stuart R. Berrow, Alenka Mertelj, Natan Osterman, Nerea Sebastián, Helen F. Gleeson, and Richard. J. Mandle. Spontaneous symmetry breaking in polar fluids. *Nature Communications*, 15(1), July 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-50230-2. URL `http://dx.doi.org/10.1038/s41467-024-50230-2`. 1, 9, 10

[25] Tess Smidt, Mario Geiger, and Benjamin Miller. Finding symmetry breaking order parameters with euclidean neural networks. *Physical Review Research*, 3, 01 2021. doi: 10.1103/PhysRevResearch.3.L012002. 1

[26] Rui Wang, Robin Walters, and Rose Yu. Approximately equivariant networks for imperfectly symmetric dynamics. 01 2022. doi: 10.48550/arXiv.2201.11969. 9, 10

[27] Rui Wang, Elyssa Hofgard, Han Gao, Robin Walters, and Tess E. Smidt. Discovering symmetry breaking in physical systems with relaxed group convolution. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024. 1, 3

[28] Andrew Blake and Andrew Zisserman. *Visual reconstruction*. MIT Press, Cambridge, MA, USA, 1987. ISBN 0262022710. 1

[29] Alan Yuille. Energy functions for early vision and analog networks. Technical report, USA, 1987. 1

[30] Stefanos Pertigkiozoglou, Evangelos Chatzipantazis, Shubhendu Trivedi, and Kostas Daniilidis. Improving equivariant model training via constraint relaxation. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 83497–83520. Curran Associates, Inc., 2024. URL `https://proceedings.neurips.cc/paper_files/paper/2024/file/98082e6b4b97ab7d3af1134a5013304e-Paper-Conference.pdf`. 1, 2, 9, 10, 11, 13, 14, 16

[31] Ignacio Hounie, Alejandro Ribeiro, and Luiz F. O. Chamon. Resilient constrained learning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc. 3

[32] L. F. O. Chamon, A. Amice, S. Paternain, and A. Ribeiro. Resilient control: Compromising to adapt. In *IEEE Control and Decision Conference*, 2020. 3

[33] L. F. O. Chamon and A. Ribeiro. Probably approximately correct constrained learning. In *Conference on Neural Information Processing Systems~(NeurIPS)*, 2020. 3

[34] L. F. O. Chamon, S. Paternain, M. Calvo-Fullana, and A. Ribeiro. Constrained learning with non-convex losses. *IEEE Trans. on Inf. Theory*, 69[3]:1739–1760, 2023. doi: 10.1109/TIT.2022.3187948. 3

[35] Dimitri P. Bertsekas. *Convex optimization theory*. Athenea Scientific, 2009. 3

[36] Angelia Nedić and Asuman Ozdaglar. Approximate primal solutions and rate analysis for dual subgradient methods. *SIAM Journal on Optimization*, 19(4):1757–1780, 2009. doi: 10.1137/070708111. URL `https://doi.org/10.1137/070708111`. 3

[37] Ashish Cherukuri, Enrique Mallada, and Jorge Cortes. Asymptotic convergence of constrained primal-dual dynamics. *Systems & Control Letters*, 87, 10 2015. doi: 10.1016/j.sysconle.2015.10.006. 3

[38] L. F. O. Chamon, S. Paternain, and A. Ribeiro. Counterfactual programming for optimal control. In *Learning for Dynamics and Control (L4DC)*, 2020. 3

[39] Thomas N. Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard S. Zemel. Neural relational inference for interacting systems. In *International Conference on Machine Learning*, pages 2693–2702, 2018. 4

[40] CMU. The cmu motion capture dataset, 2003. URL `https://mocap.cs.cmu.edu/`. 4

[41] Fabian B. Fuchs, Daniel E. Worrall, Volker Fischer, and Max Welling. Se(3)-transformers: 3d roto-translation equivariant attention networks. *CoRR*, abs/2006.10503, 2020. URL `https://arxiv.org/abs/2006.10503`. 9, 14, 15

[42] Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. *CoRR*, abs/2102.09844, 2021. URL `https://arxiv.org/abs/2102.09844`. 9, 14, 15

[43] Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling. Geometric and physical quantities improve e(3) equivariant message passing. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=_xwr8gOBeV1`. 9, 13, 14, 16

[44] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J. Guibas. Vector neurons: A general framework for so(3)-equivariant networks. In *ICCV*, pages 12180–12189. IEEE, 2021. ISBN 978-1-6654-2812-5. 9, 13, 14

[45] Minkai Xu, Jiaqi Han, Aaron Lou, Jean Kossaifi, Arvind Ramanathan, Kamyar Azizzadenesheli, Jure Leskovec, Stefano Ermon, and Anima Anandkumar. Equivariant graph neural operator for modeling 3d dynamics. *arXiv preprint arXiv:2401.11037*, 2024. 9, 14, 15

[46] Shuxiao Chen, Edgar Dobriban, and Jane H. Lee. A group-theoretic framework for data augmentation. *Journal of Machine Learning Research*, 21(245):1–71, 2020. URL `http://jmlr.org/papers/v21/20-163.html`. 9, 10

[47] Alberto Bietti and Julien Mairal. Group invariance, stability to deformations, and complexity of deep convolutional representations. *J. Mach. Learn. Res.*, 20(1):876–924, January 2019. ISSN 1532-4435.

[48] Akiyoshi Sannai and Masaaki Imaizumi. Improved generalization bound of group invariant / equivariant deep networks via quotient feature space. *Uncertainty in Machine Learning*, 2021. 9, 10

[49] Marc Finzi, Gregory Benton, and Andrew G Wilson. Residual pathway priors for soft equivariance constraints. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 30037–30049. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper_files/paper/2021/file/fc394e9935fbd62c8aedc372464e1965-Paper.pdf`. 9, 10

[50] Nate Gruver, Marc Anton Finzi, Micah Goldblum, and Andrew Gordon Wilson. The lie derivative for measuring learned equivariance. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=JL7Va5Vy15J`. 10

[51] Samuel E. Otto, Nicholas Zolman, J. Nathan Kutz, and Steven L. Brunton. A unified framework to enforce, discover, and promote symmetry in machine learning, 2024. URL `https://arxiv.org/abs/2311.00212`. 10

[52] Roger A. Horn and Charles R. Johnson. *Matrix analysis*. Cambridge University Press, 2018. 11

[53] L. Ruddigkeit, R. van Deursen, L. C. Blum, and J.-L. Reymond. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of Chemical Information and Modeling*, pages 2864–75, 2012. ACS. 13

[54] R. Ramakrishnan, O. Dral, P., M. Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 2014. Nature. 13

[55] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015. doi: 10.1109/CVPR.2015.7298801. 13

[56] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.*, 38(5), October 2019. ISSN 0730-0301. doi: 10.1145/3326362. URL `https://doi.org/10.1145/3326362`. 13

[57] Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: sampling configurations for multibody systems with symmetric energies, 2019. URL `https://arxiv.org/abs/1910.00753`. 14, 15

[58] Weitao Du, He Zhang, Yuanqi Du, Qi Meng, Wei Chen, Bin Shao, and Tie-Yan Liu. Equivariant vector field network for many-body system modeling. *CoRR*, abs/2110.14811, 2021. URL `https://arxiv.org/abs/2110.14811`. 14

[59] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon. Dynamic graph CNN for learning on point clouds. *CoRR*, 2018. 14

[60] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. 16

[61] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019. 16
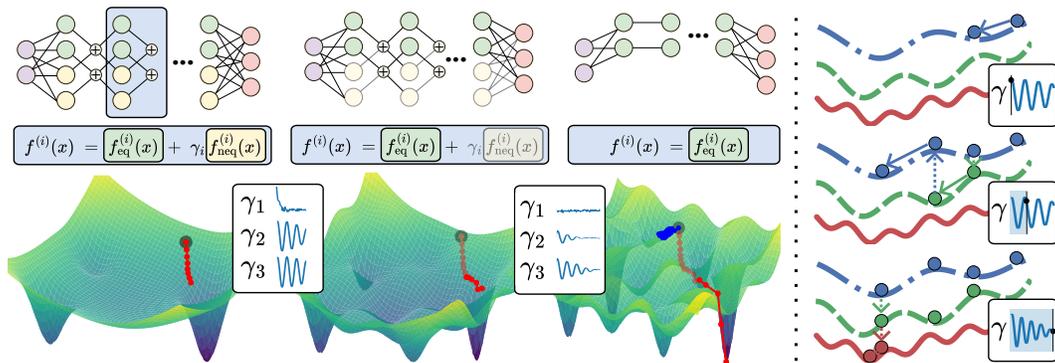
Figure 3: The proposed ACE homotopic optimization scheme at a glance. **Left:** The red trajectory illustrates the training of a relaxed, non-equivariant model that gradually becomes equivariant as the layer-wise coefficients γ_i decay. The blue trajectory illustrates a strictly equivariant network f_{eq} trained from the same initialization. **Right:** the coefficient γ allows the training problem to transition between easier (non-equivariant) and harder (equivariant) problems, facilitating convergence.

A   Appendix / supplemental material

We provide further discussion of related work, an extended empirical evaluation, and additional technical and theoretical details of our homotopic equivariant method. Additionally, Fig. 3 offers an intuitive illustration of our approach and contrasts standard training with the proposed homotopic optimization scheme.

A.1   Related Work

Equivariant NNs encode group symmetries directly in their architecture. In computer vision, for instance, equivariant convolutional NNs accomplish this by tying filter weights for different rotations and reflections [1–4]. Equivariant NNs have also been extended to 3D graphs and molecular systems, as is the case of SchNet, that operates directly on atomic coordinates while remaining E(3)-invariant [12], and Tensor Field Networks or SE(3) Transformers, that extend this to full roto-translation equivariance using spherical harmonics and attention, respectively [13, 41].

Scalable, parameter-efficient architectures that remain equivariant include EGNNs [42], SEG-NNs [43], and methods such as Vector Neurons [44], which lift scalars to 3D vectors, achieving exact SO(3) equivariance in common layers with minimal extra parameters. Moreover, recent works model 3D trajectories rather than static snapshots, such as EGNO, which uses SE(3) equivariant layers combined with Fourier temporal convolutions to predict spatial trajectories [45].

While exact equivariance can improve sample efficiency [46–48], it may render the optimization harder, particularly when the data violates symmetry (even if partially) due to noise or other symmetry-breaking phenomena [22–24]. This has motivated the use of models with *approximate* or *relaxed* equivariance based on architectures that soften weight-sharing, such as Residual Pathway Priors [49] that adds a non-equivariant branch alongside the equivariant one, or works such as Wang et al. [26], where weighted filter banks are used to interpolate between equivariant kernels. Recent works have formalized the trade-offs that appear due to these symmetry violations, establishing generalization bounds that specify when and how much equivariance should be relaxed [19].

Beyond architectural modifications, another line of work relaxes symmetry *during training*. REMUL [18] adds a penalty for equivariance violations to the training objective, adapting the penalty weights throughout training to balance accuracy and equivariance. This procedure does not guarantee that the final solution is equivariant or even the degree of symmetry achieved. An alternative approach modifies the underlying architecture by perturbing each equivariant layer and gradually reducing the perturbation during training according to a user-defined schedule [30]. This guided relaxation facilitates training and guarantees that the final solution is equivariant, but is sensitive to the schedule. Additional equivariance penalties based on Lie derivative are used to mitigate this issue, increasing the number of hyperparameters and restricting the generality of the method.

The method proposed in this paper addresses many of the challenges in these exact and relaxed equivariant NN training approaches, namely, (i) the complex loss landscape induced by equivariant NNs; (ii) the manual tuning of penalties, weights, and schedules; (iii) the need to account for partial symmetries in the data.

Algorithm 1 Strictly equivariant data	**Algorithm 2** Partially equivariant data
1: **Inputs**: $\eta_p, \eta_d > 0$, $\gamma^{(0)} = 1$, $\lambda^{(0)} = 0$	1: **Inputs**: $\eta_p, \eta_d > 0$, $\gamma^{(0)} = 1$, $\lambda^{(0)} = 0$
2: $J_0^{(t)} = \dfrac{1}{N} \sum\limits_{n=1}^{N} \ell_0\big(f_{\theta^{(t)}, \gamma^{(t)}}(x_n), y_n\big)$	2: $J_0^{(t)} = \dfrac{1}{N} \sum\limits_{n=1}^{N} \ell_0\big(f_{\theta^{(t)}, \gamma^{(t)}}(x_n), y_n\big)$
3: $\theta^{(t+1)} = \theta^{(t)} - \eta_p \nabla_\theta J_0^{(t)}$	3: $\theta^{(t+1)} = \theta^{(t)} - \eta_p \nabla_\theta J_0^{(t)}$
4: $\gamma_i^{(t+1)} = \gamma_i^{(t)} - \eta_p\left(\nabla_{\gamma_i} J_0^{(t)} + \lambda_i^{(t)}\right)$	4: $\gamma_i^{(t+1)} = \gamma_i^{(t)} - \eta_p\left(\nabla_{\gamma_i} J_0^{(t)} + \lambda_i^{(t)}\right)$
	5: $u_i^{(t+1)} = u_i^{(t)} + \eta_p(\rho u_i^{(t)} - \lambda_i^{(t)})$
5: $\lambda_i^{(t+1)} = \lambda_i^{(t)} + \eta_d \gamma_i^{(t)}$	6: $\lambda_i^{(t+1)} = \left[\lambda_i^{(t)} + \eta_d(|\gamma_i^{(t)}| - u_i^{(t)})\right]_+$

A.2 Training Equivariant Models

Equivariant NNs incorporate symmetries inherent to the data by ensuring that specific transformations of the input lead to predictable transformations of the output. Formally, let $f_\theta : X \to Y$ be a NN parameterized by $\theta \in \mathbb{R}^p$, the network weights. A NN is said to be *equivariant* to a group G with respect to actions ρ_X on X and ρ_Y on Y if

$$f_\theta\big(\rho_X(g)x\big) = \rho_Y(g)f_\theta(x), \quad \text{for all } x \in X \text{ and } g \in G. \tag{2}$$

Equivariant NNs improve sample efficiency by encoding transformations from a symmetry group G [46–48]. Strictly enforcing equivariance, however, limits the expressiveness of the model and create intricate loss landscapes, which can slow down optimization and hinder their training [19, 18], particularly in tasks involving partial or imperfect symmetries [20–24].

To temper these trade-offs, recent work relaxes symmetry during training by modifying either the optimization objective or the model itself [30, 18, 49, 26, 19]. Explicitly, let $f_{\theta,\gamma} : X \to Y$ be a NN now parameterized by both θ, the network weights, and $\gamma \in \mathbb{R}^L$, a set of *homotopic parameters* representing the model *non-equivariance*. The model is therefore equivariant for $\gamma = 0$, i.e., $f_{\theta,0}$ satisfies (2), but can behave arbitrarily if $|\gamma_i| > 0$ for any i. Consider training this NN by solving

$$\underset{\theta}{\text{minimize}} \ \alpha\left[\frac{1}{N} \sum_{n=1}^{N} \ell_0\big(f_{\theta,\gamma}(x_n), y_n\big)\right] + \beta\ell_{\text{eq}}\big(f_{\theta,\gamma}\big), \tag{3}$$

where (x_n, y_n) are samples from a data distribution $\mathfrak{D}$, ℓ_0 denotes the top-line objective function (typically, mean-squared error or cross-entropy), ℓ_{eq} denotes an equivariance metric, and $\alpha, \beta > 0$ are hyperparameters controlling their contributions to the training loss. Hence, traditional equivariant NN training amounts to using $\alpha = 1$ and $\beta = \gamma = 0$ in (3). In [18], a non-equivariant model ($\gamma = 1$) is trained using $\ell_{\text{eq}}(f) = \sum_n \mathbb{E}_{g_n}\big[\|f\big(\rho_X(g_n)x_n\big) - \rho_Y(g_n)f(x_n)\|^2\big]$ with g_n sampled uniformly from G. The values of α, β are adapted to balance the contributions of each loss, i.e., to balance accuracy and equivariance. In contrast, [30] uses an equivariance metric ℓ_{eq} based on Lie derivatives [50, 51], fix α, β, and manually decrease γ throughout training until it vanishes, i.e., until the model is equivariant.

While effective in particular instances, these approaches are based on sensitive hyperparameters, some of which are application-dependent. This is particularly critical in [18], where both performance and equivariance depend on the values of α, β. While [30] guarantees equivariance by driving γ to zero, its performance is substantially impacted by the rate at which it vanishes. It therefore continues to use ℓ_{eq} even though it is not needed to ensure equivariance. What is more, it is once again susceptible to symmetry-breaking effects in the data since it strictly imposes equivariance. As discussed in Section 2 and in the following Appendix A.3, we address these challenges by doing away with hand-designed penalties ($\beta = 0$) *and* manually-tuned schedules for γ.

A.3 Homotopic Equivariant Training

First, we recall our homotopic architecture:

$$f_{\theta,\gamma} = f_{\theta,\gamma}^L \circ \cdots \circ f_{\theta,\gamma}^1 \quad \text{with} \quad f_{\theta,\gamma}^i = f_\theta^{\mathrm{eq},i} + \gamma_i f_\theta^{\mathrm{neq},i}, \quad i = 1, \ldots, L, \tag{4}$$

Continuing the discussion in Section 2.1 and Section 2.2, we provide the learning algorithms for both strictly equivariant training (Alg. 1) and the partial equivariant variant (Alg. 2). By taking $\gamma_i^{(0)} = 1$, Alg. 1 starts by training a flexible, non-equivariant model (step 3). The gradient step on the homotopy parameters γ_i (step 4) then simultaneously seeks to reduce the objective function ℓ_0 and bring γ_i closer to zero, targeting the constraint in (PI). The strength of that bias toward zero (namely, λ_i) depends on the accumulated constraint violations (step 5): the longer γ_i is strictly positive (negative), the larger the magnitude of λ_i. Hence, the γ_i can increase to expand the flexibility of the model or decrease to exploit the symmetries in the data. The initial non-equivariant NN is therefore turned into an equivariant one by explicitly taking into account the effects this has on the downstream performance, instead of relying on hand-tuned, *ad hoc* penalties or schedules as in [18, 30].

If we stop Alg. 1 at iteration T and deploy its solution with $\gamma = 0$ (i.e., $f_{\theta^{(T)},0}$), we introduce an error that depends on the magnitude of $\gamma_i^{(T)}$. The following theorem bounds this approximation error for the architecture in Eq. (4) under mild assumptions on its components. In particular, note that Ass. 1 holds, e.g., for MLPs with ReLU activations and weight matrices of bounded spectral norms [52].

Assumption 1. The architecture in (4) is such that, for all θ and i, $f_\theta^{\mathrm{eq},i}$ and $f_\theta^{\mathrm{neq},i}$ are M-Lipschitz continuous and $f_\theta^{\mathrm{neq},i}$ is a bounded operator, i.e., $\|f_\theta^{\mathrm{neq},i}(x)\| \le B\|x\|$ for all $x \in X$.

Theorem A.1. *Consider $f_{\theta,\gamma}$ as in (4) satisfying Assumption 1. Then,*

$$\left\| f_{\theta,\gamma}(x) - f_{\theta,0}(x) \right\| \le \left[\sum_{k=0}^{L-1} |\gamma_{k+1}| \left(1 + \frac{1}{k} \sum_{j=1}^k |\gamma_j| \right)^k \right] BM^{L-1}\|x\|, \quad \text{for all } x \in X.$$

Proof. The proof proceeds by bounding the error layer by layer. Indeed, begin by defining

$$z_0 = x \quad \text{and} \quad z_i = f_{\theta,\gamma}^i(z_{i-1}) = f_\theta^{\mathrm{eq},i}(z_{i-1}) + \gamma_i f_\theta^{\mathrm{neq},i}(z_{i-1}), \text{ for } i = 1, \ldots, L, \tag{5}$$

so that $z_L = f_{\theta,\gamma}(x)$. Likewise, let

$$z_0^{\mathrm{eq}} = x \quad \text{and} \quad z_i^{\mathrm{eq}} = f_\theta^{\mathrm{eq},i}(z_{i-1}^{\mathrm{eq}}), \text{ for } i = 1, \ldots, L, \tag{6}$$

so that $z_L^{\mathrm{eq}} = f_{\theta,0}(x)$.

Proceeding, we use the triangle equality to obtain

$$\delta_i \triangleq \|z_i - z_i^{\mathrm{eq}}\| \le \left\| f_i^{\mathrm{eq}}(z_{i-1}) - f_i^{\mathrm{eq}}(z_{i-1}^{\mathrm{eq}}) \right\| + |\gamma_i|\left\| f_i^{\mathrm{neq}}(z_{i-1}) \right\| \tag{7}$$

We bound (7) using the fact that f_i^{eq} is M-Lipschitz continuous and that f_i^{neq} is a bounded operator (Ass. 1) to write

$$\left\| f_i^{\mathrm{eq}}(z_{i-1}) - f_i^{\mathrm{eq}}(z_{i-1}^{\mathrm{eq}}) \right\| \le M\left\| z_{i-1} - z_{i-1}^{\mathrm{eq}} \right\| = M\delta_{i-1}$$
$$|\gamma_i|\left\| f_i^{\mathrm{neq}}(z_{i-1}) \right\| \le |\gamma_i|B\left\| z_{i-1} \right\|.$$

Back in (7) we get

$$\delta_i \le M\delta_{i-1} + |\gamma_i|B\left\| z_{i-1} \right\|. \tag{8}$$

Noticing that the quantity of interest is in fact $\left\| f_{\theta,\gamma}(x) - f_{\theta,0}(x) \right\| = \|z_L - z_L^{\mathrm{eq}}\| = \delta_L$, we can use (8) recursively to obtain

$$\delta_L \le B \sum_{k=1}^L M^{L-k}|\gamma_k|\left\| z_{k-1} \right\|, \tag{9}$$

using the fact that $\delta_0 = 0$.

To proceed, note from Assumption 1 and the definition of the architecture in (1) that $f_{\theta,\gamma}^i$ is $M(1 + |\gamma_i|)$-Lipschitz and $z_i = f_{\theta,\gamma}^i \circ \cdots \circ f_{\theta,\gamma}^1$. Hence, we obtain

$$\|z_i\| \leq \prod_{j=1}^{i} \left[M(1 + |\gamma_j|) \right] \|x\|, \quad \text{for } i = 1, \ldots, L. \tag{10}$$

Using the fact that $z_0 = x$, (9) immediately becomes

$$\delta_L \leq BM^{L-1} \left[|\gamma_1| + \left(\sum_{k=2}^{L} |\gamma_k| \prod_{j=1}^{k-1} (1 + |\gamma_j|) \right) \right] \|x\|, \tag{11}$$

To obtain the bound in Theorem A.1, suffices it to use the relation between the arithmetic and the geometric mean (AM-GM inequality) to obtain

$$\prod_{j=1}^{k-1} (1 + |\gamma_j|) \leq \left(1 + \frac{1}{k-1} \sum_{\ell=1}^{k-1} |\gamma_\ell| \right)^{k-1}.$$

Rearranging the terms yields the desired result. $\qquad\square$

Theorem A.1 shows that as long as the $\{\gamma_i^{(T)}\}$ are small enough, the error incurred from setting them to zero is negligible. This is not the case, however, when the γ_i oscillate without subsiding (Section 2.2). This effect can be used to identify and correct for symmetry deviations in the data. To this end, we rely on the fact that γ provides a measure of non-equivariance, which we formalize next.

Theorem A.2. *Consider $f_{\theta,\gamma}$ as in (4) satisfying Assumption 1 and let $C = \max(B/M, 1)$. Then,*

$$\left\| \rho_Y(g) f_{\theta,\gamma}(x) - f_{\theta,\gamma}(\rho_X(g)x) \right\| \leq 2 \sum_{k=1}^{L} |\gamma_k| \left(1 + \frac{C}{L-1} \sum_{j \neq k} |\gamma_j| \right)^{L-1} B^2 M^{L-1} \|x\|,$$

for all $x \in X$ and $g \in G$.

Proof. We proceed once again in a layer-wise fashion using the definitions in (5) and

$$\tilde{z}_0 = \rho_X(g)x \quad \text{and} \quad \tilde{z}_i = f_{\theta,\gamma}^i(\tilde{z}_{i-1}) = f_\theta^{\text{eq},i}(\tilde{z}_{i-1}) + \gamma_i f_\theta^{\text{neq},i}(\tilde{z}_{i-1}), \text{ for } i = 1, \ldots, L.$$

Hence, we can write the equivariance error at the i-th layer as

$$\epsilon_i \triangleq \|\rho_i(g)z_i - \tilde{z}_i\| = \left\| \rho_i(g) f_\theta^{\text{eq},i}(z_{i-1}) + \gamma_i \rho_i(g) f_\theta^{\text{neq},i}(z_{i-1}) - f_\theta^{\text{eq},i}(\tilde{z}_{i-1}) - \gamma_i f_\theta^{\text{neq},i}(\tilde{z}_{i-1}) \right\|$$

Notice that, in this notation, we seek a bound on ϵ_L independent of $x \in X$ and G.

Proceeding layer-by-layer, use the equivariance of $f^{\text{eq},i}$ and the triangle inequality to bound ϵ_i by

$$\epsilon_i \leq \left\| f_\theta^{\text{eq},i}(\rho_{i-1}(g)z_{i-1}) - f_\theta^{\text{eq},i}(\tilde{z}_{i-1}) \right\| + |\gamma_i| \left\| \rho_i(g) f_\theta^{\text{neq},i}(z_{i-1}) - f_\theta^{\text{neq},i}(\tilde{z}_{i-1}) \right\| \tag{12}$$

Using the fact that $f^{\text{eq},i}$ is Lipschitz continuous and $f^{\text{neq},i}$ is a bounded operator (Assumption 1), we obtain

$$\left\| f_\theta^{\text{eq},i}(\rho_{i-1}(g)z_{i-1}) - f_\theta^{\text{eq},i}(\tilde{z}_{i-1}) \right\| \leq M \|\rho_{i-1}(g)z_{i-1} - \tilde{z}_{i-1}\| = M\epsilon_{i-1}$$

$$\left\| \rho_i(g) f_\theta^{\text{neq},i}(z_{i-1}) - f_\theta^{\text{neq},i}(\tilde{z}_{i-1}) \right\| \leq B(B \|z_{i-1}\| + \|\tilde{z}_{i-1}\|)$$

We can further expand the second bound using the fact that

$$\|\tilde{z}_{i-1}\| = \|\tilde{z}_{i-1} - \rho_{i-1}z_{i-1} + \rho_{i-1}z_{i-1}\| \leq \|\tilde{z}_{i-1} - \rho_{i-1}z_{i-1}\| + \|\rho_{i-1}z_{i-1}\| \leq \epsilon_{i-1} + B \|z_{i-1}\|.$$

When combined into (12) we obtain the bound

$$\epsilon_i \leq (M + |\gamma_i|B)\epsilon_{i-1} + 2|\gamma_i|B^2 \|z_{i-1}\|. \tag{13}$$

12

Applying (13) recursively, we obtain

$$\epsilon_L \leq 2B^2 \sum_{k=1}^{L} |\gamma_k| \left[\prod_{j=k+1}^{L} (M + |\gamma_j|B) \right] \|z_{k-1}\|, \tag{14}$$

where we used the fact that $\epsilon_0 = \|\rho_X(g)z - \tilde{z}_0\| = 0$.

To proceed, we once again use Assumption 1 and the definition of the architecture in (4) to obtain (10), so that (14) becomes

$$\epsilon_L \leq 2B^2 \sum_{k=1}^{L} |\gamma_k| \left[\prod_{j=k+1}^{L} (M + |\gamma_j|B) \right] \left[\prod_{j=1}^{k-1} \left[M(1 + |\gamma_j|) \right] \right] \|x\|.$$

To simplify the bound, we take $C = \max(B/M, 1)$, so that we can write

$$\epsilon_L \leq 2M^{L-1}B^2 \sum_{k=1}^{L} |\gamma_k| \left[\prod_{j=k+1}^{L} \left(1 + \frac{B}{M}|\gamma_j| \right) \right] \left[\prod_{j=1}^{k-1} (1 + |\gamma_j|) \right] \|x\|$$

$$\leq 2M^{L-1}B^2 \sum_{k=1}^{L} |\gamma_k| \left[\prod_{j \neq k} (1 + C|\gamma_j|) \right] \|x\|, \tag{15}$$

Using once again the relation between the arithmetic and the geometric mean (AM-GM inequality) yields

$$\epsilon_L \leq 2 \sum_{k=1}^{L} |\gamma_k| \left(1 + \frac{C}{L-1} \sum_{j \neq k} |\gamma_j| \right)^{L-1} B^2 M^{L-1} \|x\|.$$

$\square$

Note that the bounded assumption on the group action is mild and that in many cases of interest the group action is in fact isometric, i.e., $\|\rho_i(g)z\| = \|z\|$ (e.g., rotation, translation). Theorem A.2 shows that by adjusting the degree to which the equivariance constraint in (PI) is relaxed, we can control the extent to which the final solution remains equivariant.

A.4 Other Experimental Results

To further address **RQ1**, we report in Table 1 (left) the test MSE results ($\times 10^{-3}$) for various SEGNN variants and baselines. Our SEGNN trained with ACE, f^{eq}, outperforms all other baselines, including the heuristic-based relaxed equivariance scheme of Pertigkiozoglou et al. [30] and, after projection, matches the performance of a non-equivariant SEGNN that retains the same architecture $f^{\mathrm{eq}} + f^{\mathrm{neq}}$ but, without using ACE, relies on explicit $\mathsf{SO}(3)$ augmentations to learn equivariance.

For molecular property regression, we evaluate on the QM9 dataset [53, 54] using the invariant SchNet [12] and our ACE variant. Due to its strict invariance, SchNet may severely hinder convergence without any relaxation; indeed, as shown in Table 1 (right), incorporating ACE during training reduces the mean absolute error (MAE) across all quantum-chemical targets. To assess the impact of ACE on a more flexible architecture, we compare a standard SEGNN [43] against its ACE counterpart on QM9. As shown in Table 1 (right), the ACE SEGNN lowers the MAE on most targets; however, the overall performance remains largely comparable. Collectively, these findings show that SEGNN's E(3)-equivariance aligns well with the QM9 dataset, facilitating optimization. However, ACE can still provide an additional boost in performance for certain molecular properties.

On the ModelNet40 classification benchmark [55], we apply ACE to the VN-DGCNN architecture [56, 44]. Figure 4 (left) shows that our ACE equivariant model f^{eq} achieves gains of $+2.78$ and $+1.77$ percentage points for the class and instance accuracy metrics over normal training. The partially equivariant ACE variant $f^{\mathrm{eq}} + f^{\mathrm{neq}}$ achieves comparable improvements. Both models are obtained by checkpointing the same training runs.

To further address **RQ2**, we evaluate convergence under input degradation on ModelNet40 [55] by randomly dropping 0-85% of the points in the 3D point clouds of every shape. As shown in

Table 1: Error metrics for equality-constrained models. **Left:** N-body, equality-constrained SEGNN projections reach the lowest test MSE ($\times 10^{-3}$). ACE† represents models trained with equality constraints, $*$ are trained without ACE, but with $\mathsf{SO}(3)$ augmentations. **Right:** QM9, constraining SchNet and SEGNN reduces or matches MAE on most quantum-chemical targets.

Method	MSE $\downarrow$
Linear [42]	81.9
SE(3)-Tr. [41]	24.4
RF [57]	10.4
ClofNet [58]	6.5
EGNN [42]	7.1
EGNO [45]	5.4
SEGNN [43]	$5.6_{\pm 0.25}$
SEGNN$_{\text{Rel.}}$ [30]	$4.9_{\pm 0.18}$
SEGNN$^{*}_{f^{\text{eq}}}$	$5.7_{\pm 0.21}$
SEGNN$^{*}_{f^{\text{eq}}+f^{\text{neq}}}$	$\mathbf{3.8_{\pm 0.06}}$
SEGNN$^{\text{ACE}\dagger}_{f^{\text{eq}}+f^{\text{neq}}}$	$\mathbf{3.8_{\pm 0.16}}$
SEGNN$^{\text{ACE}\dagger}_{f^{\text{eq}}}$	$\mathbf{3.8_{\pm 0.17}}$

	SchNet [12] (MAE $\downarrow$)		SEGNN [43] (MAE $\downarrow$)	
Target	Base	ACE (f^{eq})	Base	ACE (f^{eq})
μ	$0.0511_{\pm 0.0011}$	$\mathbf{0.0393_{\pm 0.0007}}$	$0.0371_{\pm 0.0002}$	$\mathbf{0.0362_{\pm 0.0007}}$
α	$0.1236_{\pm 0.0037}$	$\mathbf{0.0952_{\pm 0.0042}}$	$0.0991_{\pm 0.0033}$	$\mathbf{0.0936_{\pm 0.0126}}$
$\varepsilon_{\text{homo}}$	$0.0500_{\pm 0.0006}$	$\mathbf{0.0443_{\pm 0.0006}}$	$0.0250_{\pm 0.0005}$	$\mathbf{0.0244_{\pm 0.0007}}$
$\varepsilon_{\text{lumo}}$	$0.0421_{\pm 0.0005}$	$\mathbf{0.0360_{\pm 0.0004}}$	$\mathbf{0.0244_{\pm 0.0003}}$	$0.0257_{\pm 0.0006}$
$\Delta\varepsilon$	$0.0758_{\pm 0.0008}$	$\mathbf{0.0686_{\pm 0.0011}}$	$\mathbf{0.0450_{\pm 0.0009}}$	$0.0505_{\pm 0.0091}$
$\langle R^2 \rangle$	$1.4486_{\pm 0.1225}$	$\mathbf{0.7535_{\pm 0.0426}}$	$1.2028_{\pm 0.0827}$	$\mathbf{1.1681_{\pm 0.1890}}$
ZPVE	$0.0029_{\pm 0.0003}$	$\mathbf{0.0018_{\pm 0.0000}}$	$0.0029_{\pm 0.0004}$	$\mathbf{0.0025_{\pm 0.0001}}$
U_0	$9.6429_{\pm 3.2673}$	$\mathbf{3.4039_{\pm 0.2872}}$	$2.6163_{\pm 0.4020}$	$\mathbf{2.5597_{\pm 0.7650}}$
U	$8.3412_{\pm 2.8914}$	$\mathbf{3.3540_{\pm 0.5006}}$	$2.5645_{\pm 0.5310}$	$\mathbf{2.4796_{\pm 0.1440}}$
H	$9.5835_{\pm 4.1029}$	$\mathbf{3.4403_{\pm 0.4403}}$	$2.7577_{\pm 0.7910}$	$\mathbf{2.4569_{\pm 0.5460}}$
G	$8.5405_{\pm 2.0311}$	$\mathbf{3.6473_{\pm 0.4100}}$	$2.7505_{\pm 0.6970}$	$\mathbf{2.5954_{\pm 1.0250}}$
c_v	$\mathbf{0.0379_{\pm 0.0003}}$	$0.0376_{\pm 0.0006}$	$\mathbf{0.0389_{\pm 0.0038}}$	$0.0403_{\pm 0.0050}$

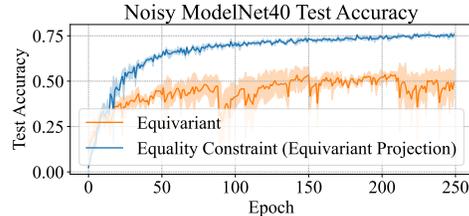| DGCNN Variant	Class Acc. $\uparrow$	Inst. Acc. $\uparrow$
DGCNN-VN [44, 59]	$0.7707_{\pm 0.0223}$	$0.8223_{\pm 0.0074}$
DGCNN-VN$^{\text{ACE}\dagger}_{f^{\text{eq}}+f^{\text{neq}}}$	$0.7919_{\pm 0.0156}$	$0.8392_{\pm 0.0115}$
DGCNN-VN$^{\text{ACE}\dagger}_{f^{\text{eq}}}$	$\mathbf{0.7985_{\pm 0.0232}}$	$\mathbf{0.8400_{\pm 0.0123}}$

Figure 4: Accuracy and robustness of normal and equality-constrained VN-DGCNNs. **Left:** On the clean ModelNet40, partially and purely equivariant constrained models (ACE†) outperform the baseline. **Right:** On noisy ModelNet40, training without ACE is unstable, while the equivariant model obtained with ACE (f^{eq}) yields a stable, high-accuracy solution.

Figure 4 (right), the strictly equivariant baseline fails to converge under this perturbation, whereas our equality-constrained projection f^{eq} converges quickly to a substantially higher test accuracy.

While the main text characterizes only the evolution of the modulation coefficients γ_i, it omits the corresponding dual variables λ_i, which serve as the Lagrange multipliers enforcing each equivariance constraint. Here, we present the full, detailed plots of γ_i and λ_i for ModelNet40 (Fig. 8 and Fig. 9) and CMU MoCap (Fig. 10 and Fig. 11), allowing simultaneous inspection of the symmetry relaxation and the accompanying optimization "pressure". By tracing λ_i alongside γ_i, one can observe how retained constraint violations accumulate multiplier weight, driving subsequent adjustments to γ_i and thereby revealing the continuous "negotiation" between inductive bias and training flexibility instantiated by our primal–dual algorithm.

Moreover, we further observe the learned slack values u_k in the resilient model for the case where the values of γ do not converge: as shown in Figure 7, early layers acquire larger slack, allowing flexibility where local structure dominates, while later layers remain nearly equivariant. As training progresses, the constraints tighten for all layers, increasingly penalizing non-equivariant solutions.

These dynamics demonstrate the ability of the proposed constrained learning framework to guide the model toward suitable, approximately equivariant behavior, stabilizing training under degradation, and improving performance over fully equivariant models.

A.5 Alignment with theoretical results

Finally, we analyze how the results from Theorem A.2 align with observed model behavior in the ACE resilient setting. We compute the equivariance deviation as $\mathbb{E}_{g \sim \mathsf{SO}(3)} \left[\left\| \rho_Y(g) f_{\theta,\gamma}(x) - f_{\theta,\gamma}\left(\rho_X(g)x \right) \right\| \right]$ (cf. Theorem A.2), where the expectation is approximated using five random samples $g \in \mathsf{SO}(3)$. As shown in Figure 6, the deviation from equivariance measured for the resilient model steadily decreases and approaches values near zero by the end of

14

Equivariant
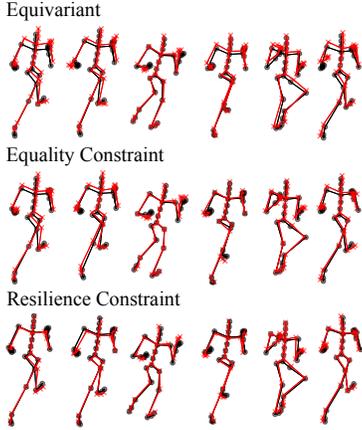
Equality Constraint

Resilience Constraint

Figure 5: Qualitative comparison on the CMU MoCap dataset (Subject #9, Run). **Top row:** Standard equivariant EGNO. **Middle row:** Partially equivariant EGNO ($f^{\text{eq}} + f^{\text{neq}}$) trained with an ACE equality constraint. **Bottom row:** Partially equivariant EGNO ($f^{\text{eq}} + f^{\text{neq}}$) trained with an ACE resilient inequality constraint, yielding significant improvements over both.
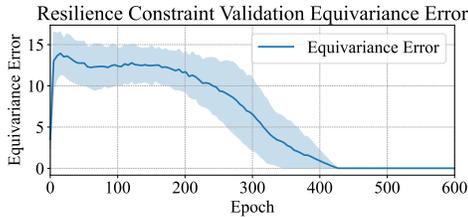
Table 2: Test MSE ($\times 10^{-2}$) on the CMU MoCap dataset for Subject #9 (Run) and Subject #35 (Walk). Results are shown for the standard equivariant models, EGNO models trained with an equality constraint (ACE†) and an EGNO model trained with an resilient inequality constraint (ACE‡). The partially equivariant EGNO ($f^{\text{eq}} + f^{\text{neq}}$) 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 [57]	$521.3_{\pm 2.3}$	$188.0_{\pm 1.9}$
TFN [13]	$56.6_{\pm 1.7}$	$32.0_{\pm 1.8}$
SE(3)-Tr.[41]	$61.2_{\pm 2.3}$	$31.5_{\pm 2.1}$
EGNN [42]	$50.9_{\pm 0.9}$	$28.7_{\pm 1.6}$
EGNO (report.) [45]	$33.9_{\pm 1.7}$	$8.1_{\pm 1.6}$
EGNO (reprod.) [45]	$35.3_{\pm 3.2}$	$8.5_{\pm 1.0}$
EGNO$^{\text{ACE}\dagger}_{f_{\text{eq}}}$	$35.3_{\pm 1.6}$	$7.9_{\pm 0.3}$
EGNO$^{\text{ACE}\dagger}_{f^{\text{eq}}+f^{\text{neq}}}$	$32.6_{\pm 1.6}$	$7.5_{\pm 0.3}$
EGNO$^{\text{ACE}\ddagger}_{f^{\text{eq}}+f^{\text{neq}}}$	$23.8_{\pm 1.5}$	$7.4_{\pm 0.2}$

Figure 6: 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.
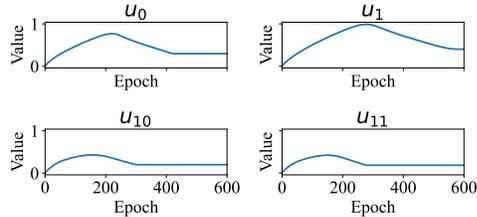

Figure 7: Adaptive constraint values u on the first two and last two layers of an EGNO model trained with ACE resilient inequality constraints on the CMU MoCap dataset (Subject #9, Run). Early layers allow large slacks that decrease over training, while later layers stay more tightly constrained.

training. This suggests that the equivariance constraint is effectively enforced in practice, and that partial equivariance can be preserved in a stable and interpretable way throughout training.

A.6 Limitations and Further Work

While our ACE framework is broadly applicable to a range of architectures, it does incur some overhead: during training we introduce extra (non-equivariant) MLP layers, which modestly slow down training (see Table 3). Addressing this, one avenue for future research is to "break" equivariance without adding any auxiliary parameters, e.g., by integrating constraint enforcement directly into existing layers or by designing specialized update rules that can break equivariance while preserving runtime efficiency.

Moreover, our experiments have focused on symmetry groups acting on point clouds, molecules, and motion trajectories. A natural extension is to apply our approach to other symmetry groups and data modalities, such as 2D image convolutions or 2D graph neural networks. Alongside this, co-designing network architectures that anticipate the dynamic tightening and loosening of equivariance constraints could produce models that converge even faster, require fewer samples, or require fewer extra parameters.

Table 3: Parameter counts and runtimes for the equivariant baselines versus models trained with ACE. We relax equivariance during training with extra linear or MLP layers, operations that PyTorch executes efficiently, therefore training and validation are only marginally slower than the baselines. At inference we drop these non-equivariant layers, yielding virtually identical runtimes to the pure equivariant models. All experiments were performed on a RTX A5000 GPU with an Intel i9-11900K CPU.

| Model	#Parameters (Train)	Train time (s/epoch)	Val time (s/epoch)
SEGNN - N-body (vanilla)	147,424	$1.368_{\pm0.004}$	$0.505_{\pm0.005}$
SEGNN - N-body (constrained)	937,960	$1.512_{\pm0.004}$	$0.527_{\pm0.004}$
SEGNN - QM9 (vanilla)	1,033,525	$118.403_{\pm0.108}$	$11.152_{\pm0.083}$
SEGNN - QM9 (constrained)	9,869,667	$129.039_{\pm0.043}$	$11.402_{\pm0.089}$
EGNO - MoCap (vanilla)	1,197,176	$0.702_{\pm0.007}$	$0.130_{\pm0.000}$
EGNO - MoCap (constrained)	1,304,762	$1.106_{\pm0.004}$	$0.130_{\pm0.000}$
DGCNN-VN - ModelNet40 (vanilla)	2,899,830	$57.826_{\pm0.500}$	$5.954_{\pm0.043}$
DGCNN-VN - ModelNet40 (constrained)	8,831,612	$81.760_{\pm0.169}$	$5.978_{\pm0.006}$
SchNet - QM9 (vanilla)	127,553	$10.389_{\pm0.152}$	$0.944_{\pm0.003}$
SchNet - QM9 (constrained)	153,543	$11.126_{\pm0.155}$	$0.947_{\pm0.002}$
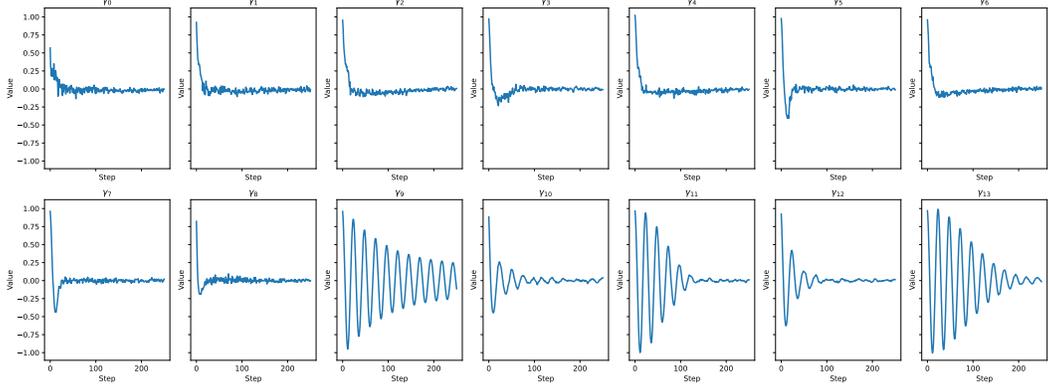
Figure 8: Full layer-wise γ_i values on ModelNet40. The ACE VN-DGCNN quickly drives the modulation coefficients γ_i of the first few layers towards 0, locking those layers into fully equivariant behaviour, while deeper layers exhibit brief oscillations before convergence.

Finally, the dual variables learned by our algorithm encode where and when symmetry is most beneficial; exploiting these variables for adaptive pruning or sparsification of non-equivariant components could also lead to more efficient models.

This work aims to investigate how training can be improved for equivariant neural networks, offering benefits in areas such as drug discovery and materials science. We do not foresee any negative societal consequences.

A.7 Additional Experimental Details

All experiments are implemented in PyTorch [60] using each model's official code base and default hyperparameters as a starting point, with the exception of the experiments with DGCNN, where we use the same setup as in Pertigkiozoglou et al. [30]. Because several repositories employed learning rates that proved sub-optimal, we performed a grid search for both the primal optimizer and, when applicable, the dual optimizer in our constrained formulation. After selecting the best values, we re-ran every configuration with multiple random seeds. A noteworthy mention is that the test results on the QM9 dataset reported in Table 1 use different splits for the training, validation and test datasets. For SchNet, we shuffle the dataset obtained from PyTorch Geometric [61] and split the dataset with a 80%/10%/10% train/validation/test ratio. For SEGNN, we use the official splits from [43]. Table 4 lists the learning rates explored and the number of seeds per dataset.
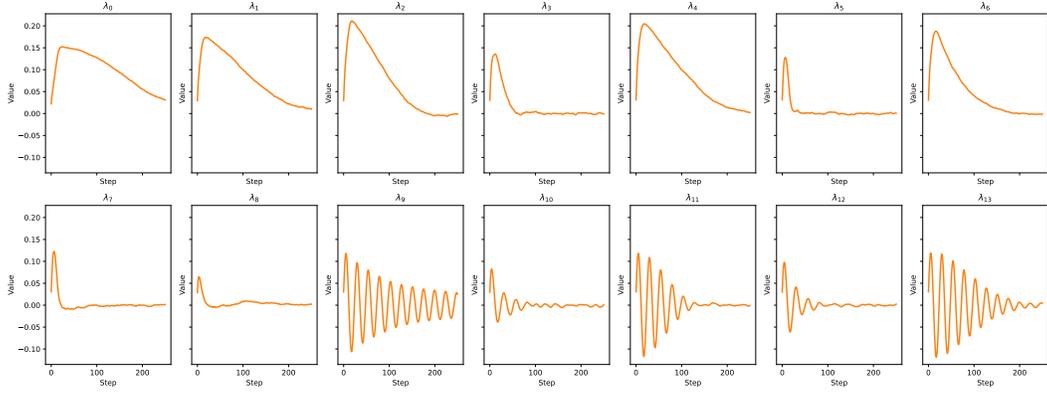
Figure 9: Dual variables λ_i on ModelNet40. The companion values of λ_i mirror the γ_i dynamics: layers whose γ_i are further away from zero accumulate larger λ_i, signaling higher constraint cost.
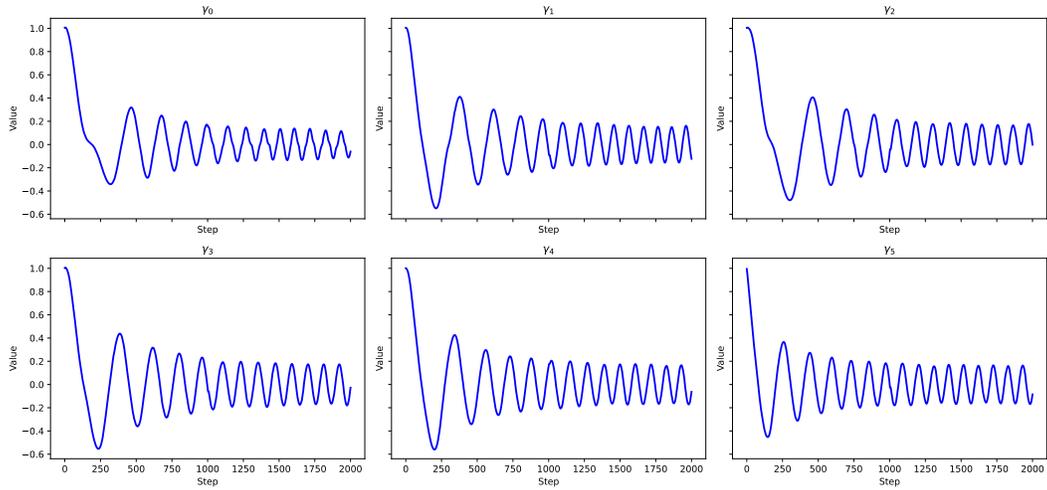

Figure 10: Layer-wise γ_i dynamics on CMU MoCap (Subject #9, Run) dataset. Rather than decaying to zero as on the ModelNet40 dataset, the modulation coefficients γ_i oscillate around zero for the full 2000-step training window, signaling that imposing strict equivariance is overly restrictive for the dataset and model.
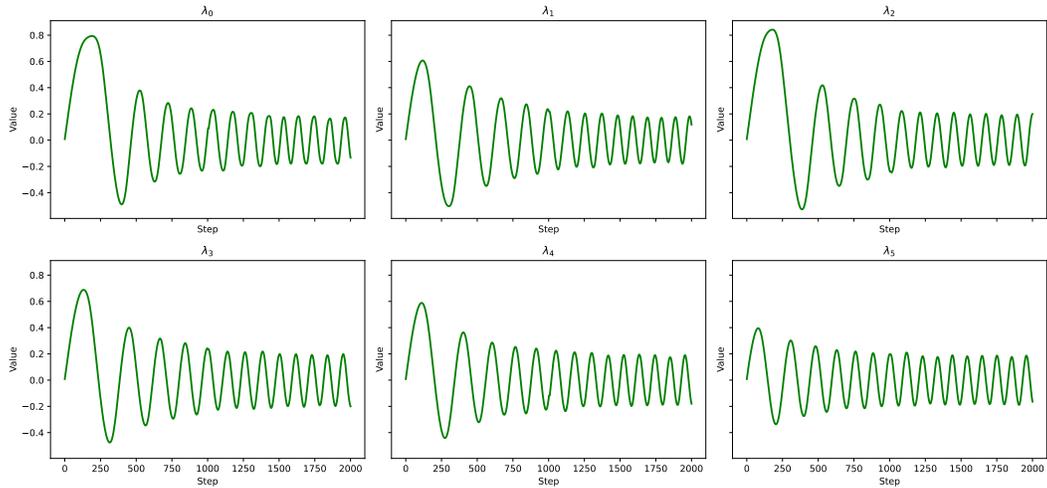

Figure 11: The dual variables λ_i on CMU MoCap (Subject #9, Run). Similar to the γ_i coefficients, the Lagrange multipliers λ_i oscillate near zero, but never converge.

We will make the code and instructions on how to reproduce the experiments publicly available upon acceptance.

Table 4: Learning rates and number of random seeds used for each dataset.

Dataset	Primal LR grid	Dual LR grid	Seeds
ModelNet40	$\{\,0.10,\ 0.20\,\}$	$\{\,1{\times}10^{-4},\ 1{\times}10^{-3}\,\}$	5
CMU MoCap	$\{\,5{\times}10^{-4},\ 1{\times}10^{-3}\,\}$	$\{\,5{\times}10^{-5},\ 2{\times}10^{-4},\ 1{\times}10^{-2}\,\}$	3
N-Body	$\{\,5{\times}10^{-4},\ 8{\times}10^{-4},\ 9{\times}10^{-4},\ 1{\times}10^{-3},\ 5{\times}10^{-3}\,\}$	$\{\,5{\times}10^{-5},\ 1{\times}10^{-4},\ 8{\times}10^{-4}\,\}$	5
QM9	$\{\,5{\times}10^{-5},\ 5{\times}10^{-4},\ 5{\times}10^{-3},\ 1{\times}10^{-2}\,\}$	$\{\,5{\times}10^{-4},\ 1{\times}10^{-2},\ 1{\times}10^{-1}\,\}$	5

All of the experiments were performed on internal clusters that contain a mix of Nvidia RTX A5000, RTX 4090, or A100 GPUs and Intel i9-11900K, AMD EPYC 7742, and AMD EPYC 7302 CPUs. All experiments consumed a total of approximately 1000 GPU hours, with the longest compute being consumed on the QM9 dataset. Runtimes and parameter counts are reported in Table 3.