Cylindrical Lattice Embedding for Program Induction

Jinseo Shim
Gwangju Institute of Science and Technology (GIST)
jinseo5892@gm.gist.ac.kr

Abstract

This study is grounded in prior work on program induction framework with a structured latent program space, called Program Lattice Auto Encoder(PLAE). It preserves compositional structure by training an encoder where programs and their compositions correspond to integer linear combinations of program bases, forming a discrete *program lattice* that captures the geometric structure of compositional reasoning. Based on it, this paper proposes a novel extension of the PLAE aimed at improving generalization and efficiency by choosing a cylindrical lattice latent space instead of plane, which can represent invariant programs. The core hypothesis is that only isometric transformations conserve compositional properties of lattice structure and therefore developable surfaces such as a cylinder or cone are permissible as embedding space. Moreover, through demonstrating a contradiction of lattice on conical manifolds, it conclude that only cylinder is a possible embedding manifold for lattice structure.

Introduction&Background

A central goal of Artificial General Intelligence (AGI) is to develop models that exhibit human-like reasoning and generalization. The Abstraction and Reasoning Corpus(ARC) (Chollet 2019) is a crucial testbed for this goal, challenging models to infer and apply abstract rules from a few examples rather than relying on memorization. Neuro-symbolic methods, which blend neural and symbolic reasoning, have emerged as a leading strategy for solving ARC such as DreamCoder(Ellis et al. 2020). This hybrid approach is motivated by the distinct weaknesses of its alternatives: purely symbolic systems often struggle with the combinatorial explosion of their search space, while purely neural methods fail to generalize to novel, out-of-distribution tasks.

Recent work has sought to bridge the gap between purely symbolic and neural methods. A notable example is the Latent Program Network (LPN)(Macfarlane and Bonnet 2025), which learns a distribution over programs in a continuous latent space, enabling efficient gradient-based search at test time. LPN's key innovation is the fusion of neural network scalability with the structured reasoning of symbolic AI. The model generalizes not by simple stochastic sampling but by actively searching for the latent program that best explains a new task. However, LPN has several key limitations that motivate our work. Its primary drawback is a lack of interpretability as it operates on implicit, continuous representations rather than explicit, human-readable programs. Furthermore, its search process is not a true program synthesis; instead, it samples from a program distribution, and a decoder executes the program in a single step. This single-step execution limits the model's ability to perform multi-step compositional reasoning and overloads the decoder, which must be trained to handle programs of vastly different complexities. This contributes to poor skill-acquisition efficiency, which requires extensive training in large externally generated datasets.

Prior Work by Applicant In contrast, our ongoing work on the Program Lattice Auto Encoder (PLAE)(Park et al. 2025) embeds observations, such as input-output pairs, as vectors within a structured latent space, called a *program lattice*. Within this lattice, foundational programs act as basis vectors, allowing any composed program on a lattice point to be expressed as a linear combination of these bases. This reframes program induction into a three-step geometric process: 1) inferring the vector in the lattice space that corresponds to the task data, 2) identifying the closest lattice point to that vector, and 3) decomposing the resulting lattice point vector using the program bases. This framework enhances both interpretability and computational efficiency by capturing the geometric nature of compositional reasoning in a discrete space. However, the current PLAE model has a key limitation: its reliance on a fixed, pre-defined lattice structure makes it suboptimal for certain problem domains. Specifically, standard vector addition cannot model invariant operations—actions that should return to their starting point after a set number of applications, like a full rotation. This "invariant problem" forces the model to generate longer program sequences, leading to computational inefficiency.

Research Focus To overcome these key limitations of PLAE, this research statement focus on finding alternative manifolds of flat lattice embedding space to represent invariant operations while preserving lattice compositionality. Consequently, cylindrical embedding space was selected among iso-metric transformed manifolds by excluding manifold which cannot represent invariant program.

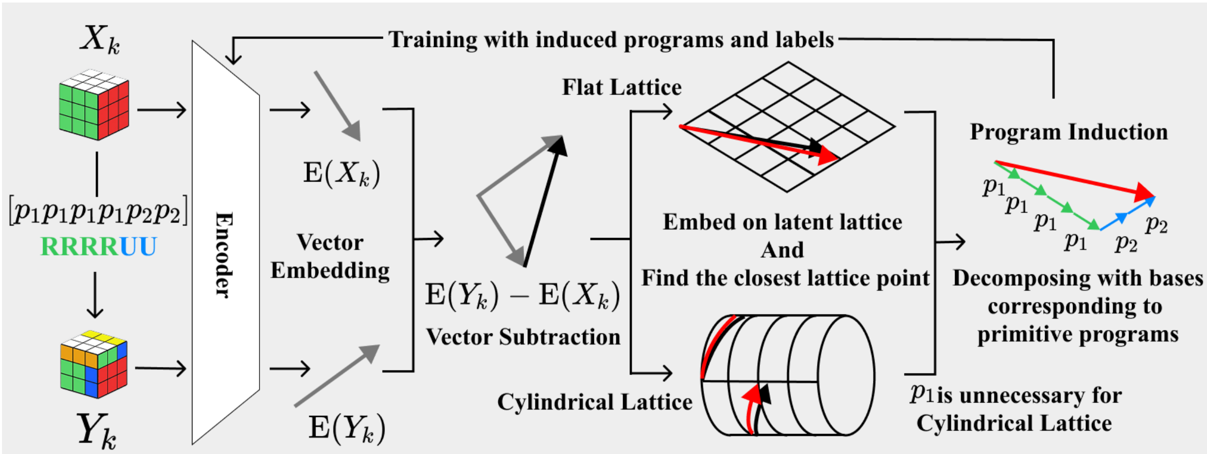

Figure 1: Program Induction process of Program Lattice Auto Encoder(above) and extended approach(blow): An encoder embeds input and output pair as vector on lattice and find closest lattice point. Finally vector was decomposed with bases corresponding to primitive programs.

Approach

To address the limitations of fixed flat lattices in modeling invariant operations, we propose mapping the latent program space onto a cylindrical manifold. Unlike a standard Euclidean plane where vector addition is unbounded, a cylindrical embedding introduces a periodic boundary condition along one dimension. This topological feature allows the model to naturally represent invariant operations—such as a 360° rotation—as closed loops that return to the initial state, rather than as infinite displacements in a flat space.

Geometric Justification and Definition The choice of the cylinder is not arbitrary but dictated by the geometric requirements of the *program lattice*. To preserve the compositionality of the lattice (where programs are linear combinations of basis vectors), the embedding manifold must be isometric to the flat Euclidean plane. This restricts our candidate spaces to developable surfaces—surfaces with zero Gaussian curvature—such as cylinders and cones. As illustrated in Figure 1 (bottom), the cylindrical manifold allows us to wrap the axis corresponding to the invariant operation. On this manifold, a sequence of programs that should result in an identity operation (e.g., four 90° rotations) forms a trajectory that encircles the cylinder and connects back to the starting lattice point. This resolves the "invariant problem" by matching the geometric path with the logical outcome, thereby preventing the generation of unnecessarily long program sequences.

Uniqueness of the Cylinder While other developable surfaces such as cones also preserve local isometry, they fail to support the global structure required for lattice embedding. On a conical manifold, straight lines from a flat plane map to spirals rather than closed loops due to the singularity at the apex. Consequently, a lattice structure cannot maintain its regular periodicity on a cone without distortion. Therefore, the cylinder is the unique manifold that simultaneously satisfies two critical conditions: 1) it preserves the metric tensor of the flat lattice (isometry), ensuring vector addition remains valid, and 2) it possesses the correct topology to map linear invariant sequences to closed loops.

Experimental Plan and Expected Findings

The proposed approach will be evaluated through a series of tests on both a controlled synthetic environment(such as Rubik's cube) and ARC datasets. The goal is to demonstrate that the cylindrical-based latent space improves computational efficiency by comparing the sample efficiency against other compositional reasoning models including PLAE.

If the cylindrical manifold demonstrates superior performance to a flat lattice as an embedding space, this would serve as strong evidence that a cylindrical geometry is a more suitable choice for abstracting data containing invariant operations. Furthermore, a significant performance gap would show that applying invariant operations play a crucial role in a model's overall efficiency. Beyond this, if the model can learn and successfully identify an appropriate radius for the cylinder when it is set as a trainable parameter, it would demonstrate the feasibility of a framework in which the model autonomously configures its embedding space based on the specific properties of the data.

Conclusion

This paper validates which embedding space is appropriate for representing invariant program on lattice, from the perspective of isometric transformation and geometrical structure of lattice. In conclusion, only the cylinder can satisfy the compositional relation of the lattice points without destroying the structural characteristics of the plane lattice. Furthermore, setting the radius of cylinder as trainable parameters provides an excellent starting point for a model that refines its latent space based on the dataset which connects to meta-learning and geometric deep learning on future research.

References

Chollet, F. 2019. On the Measure of Intelligence. arXiv:1911.01547.

Ellis, K.; Wong, C.; Nye, M.; Sable-Meyer, M.; Cary, L.; Morales, L.; Hewitt, L.; Solar-Lezama, A.; and Tenenbaum, J. B. 2020. DreamCoder: Growing generalizable, interpretable knowledge with wake-sleep Bayesian program learning. arXiv:2006.08381.

Macfarlane, M. V.; and Bonnet, C. 2025. Searching Latent Program Spaces. arXiv:2411.08706.

Park, J.; Park, J.; Shim, J.; Kim, S.; Vennemann, P.; and Kim, S. 2025. Composed Program Induction with Latent Program Lattice. In *NeurIPS 2025 Workshop on Symmetry and Geometry in Neural Representations*.