Symbolic Distillation of Neural Networks

Word Count: xxx words

Yihao Liu; CRSid:yl2063

June 2025

Executive Summary

Background and Significance

Deriving closed-form analytic expressions for complex physical systems has long been a formidable challenge: deep learning yields powerful "black-box" models with high predictive accuracy, yet their internal representations offer little insight into underlying analytic laws. Modern Graph Neural Networks (GNNs) naturally mirror many-body interactions through inductive biases [1] such as permutation invariance [2], locality, and parameter sharing, providing a structured latent space suitable for interpretation. Concurrently, symbolic regression methods like PySR [3] apply evolutionary search to extract human-readable equations from data, balancing complexity against accuracy. We reproduce and extend the work [4], which integrates GNNs' physics-aligned message representations with symbolic regression. We aim to bridge high-dimensional "black-box" prediction and interpretable deep learning in scientific research with a reproducible pipeline.

Research Objectives and Methods Overview

We first generate datasets by simulating four canonical 2D many-body systems (mass–spring, inverse-distance, Coulomb, and inverse-square distance), each with disjoint trajectory splits for training and testing.

GNNs are well suited to these many-body dynamics because they treat particles as graph nodes and pairwise interactions as edges, passing learned messages that aggregate local physical effects. In our implementation, both nodes and edges are modeled by

MLPs: the node MLP embeds particle attributes (position, velocity, mass), while the edge MLP computes interaction messages from neighboring nodes. The edge MLP corresponds to physical forces between particles, while the node MLP takes the sum of incoming messages together with each particle's attributes and predicts its acceleration.

A theory from the original work [4] indicates that after model training the physical force information lie in a low-dimensional subspace, matching the true force law dimensionality, of the message activation of the edge model. In addition, the true force components form a linear transformation of the corresponding message components.

To investigate this compact representation, we apply and compare four variants of a one-step message-passing GNN:

- **Standard**: unrestricted message dimension.
- **Bottleneck**: forces messages into the same low dimension as the physical system.
- ℓ_1 **regularization**: encourages sparsity in the message components.
- **KL regularization**: aligns messages to a standard Gaussian distribution.

For each variant, we extract the highest two message channels by variance and first apply linear regression against the ground-truth force components to measure how closely messages linearly align to physical laws. We then run symbolic regression, by using PySR, on those channels to recover analytic expressions.

Two extensions further probe the encoding mechanism: one tests acceleration versus raw forces, and the other examines whether individual channels specialize in distinct force directions.

Main Results and Key Innovations

Linear Regression Analysis We evaluate how well the selected message channels from each GNN variant linearly align with true force components across four 2D many-body systems. Table 1 summarizes the results:

There is an overall trend,

$$\text{Bottleneck} > \ell_1 > \text{KL} > \text{Standard},$$

confirming that stronger constraints on message dimensionality yield better linear alignment with the true forces.

System	Standard	Bottleneck	ℓ_1	KL
Spring	0.343, 0.337	0.997, 0.998	0.833, 0.863	0.466, 0.487
Coulomb	0.011, 0.033	0.763, 0.162	0.443, 0.158	0.584, 0.350
Inverse-square	0.020, 0.043	0.668, 0.525	0.201, 0.206	0.163, 0.162
Inverse-distance	0.431, 0.425	0.409, 0.530	0.470, 0.508	0.391, 0.302

Table 1: Linear-combination fitting R^2 values for the two highest-variance message channels.

We provide Figure 1 and Figure 2 to illustrate the linear transformation in the mass–spring system for the Standard and Bottleneck GNN. The Bottleneck variant shows much stronger linear relationship than the Standard variant.

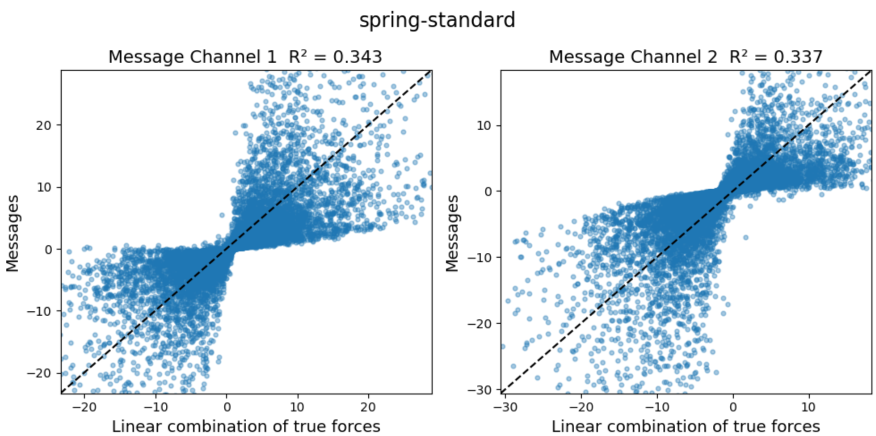

Figure 1: Spring system – Standard GNN variant. The dashed line shows $y = x$.

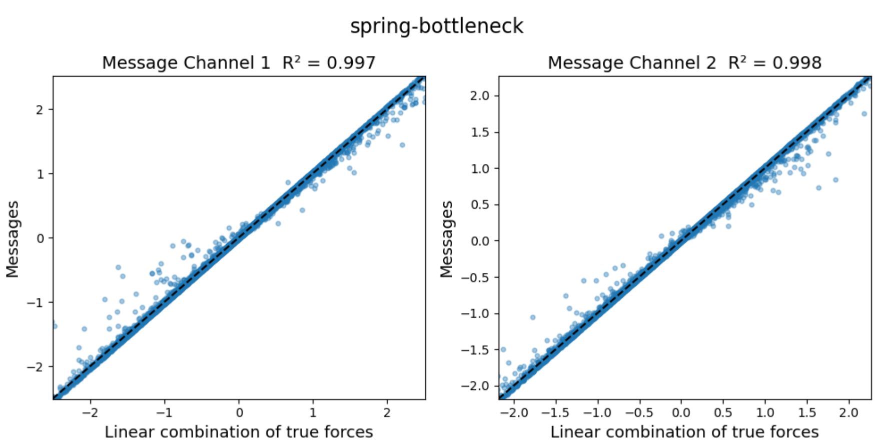

Figure 2: Spring system – Bottleneck GNN variant.

Symbolic Regression Analysis The symbolic regression outcomes, summarized in Table 2, largely echo the linear regression, except that the ordering of ℓ_1 versus KL differs.

Table 2: Symbolic regression outcome markers for each system and GNN variant.

System	**Standard**	**Bottleneck**	ℓ_1	**KL**
Spring	$\times$	$\checkmark$	$\circ\star$	$\star$
Coulomb	$\times$	$\checkmark$	$\times$	$\checkmark$
Inverse-square	$\times$	$\circ\star$	$\times$	$\circ\star$
Inverse-distance	$\times$	$\circ$	$\circ$	$\circ$

$\checkmark$ perfect discovery matching the true law.

$\times$ no valid formula recovered.

$\circ$ partial discovery (kernel found, but missing factors and/or dimensions).

$\star$ correct formula exists in the Pareto front but was not selected as "best" by PySR.

$\circ\star$ both partial and not top-ranked.

To illustrate the results of using PySR to perform symbolic regression, we display the analytic equation recovered from the Spring-Bottleneck case:

$$\phi_{\text{edge}} = \left(\tfrac{1.2640961}{r} - 1.2788521\right)\left(0.299832\,\Delta x + \Delta y\right) \tag{1}$$

Here $\Delta x = x_j - x_i$, $\Delta y = y_j - y_i$, and $r = \sqrt{\Delta x^2 + \Delta y^2}$. The small cross-term $0.299832\,\Delta y$ reflects a rotated basis in the two-dimensional space. By comparing with the standard Hooke's law,

$$\mathbf{F}_{ij} = -\,k\,(r_{ij} - 1)\,\frac{\mathbf{r}_{ij}}{r_{ij}}, \quad \mathbf{r}_{ij} = (x_i - x_j,\; y_i - y_j), \quad r_{ij} = \|\mathbf{r}_{ij}\|. \tag{2}$$

$$F_{ij,x} = -\,k\,(r_{ij} - 1)\,\frac{x_i - x_j}{r_{ij}}, \qquad F_{ij,y} = -\,k\,(r_{ij} - 1)\,\frac{y_i - y_j}{r_{ij}}, \tag{3}$$

we show that the near-perfect match confirms the rediscovery of the true Hooke's law directly into the message channels.

Extension i: Acceleration vs. Force Encoding In the inverse-distance system, none of the GNN variants recovered the necessary source node mass. We hypothesized that the edge MLP messages (in this case only) align more naturally with particle acceleration contribution than with raw force. We re-ran linear regression against ground-truth acceleration instead of force. The resulting R^2 scores jumped dramatically—for example, Bottleneck rose from 0.409 to 0.956. Crucially, this extension does not assert that GNNs always inherently learn acceleration over force, but demonstrates that PySR is reliable in extracting whatever physical law the model has learned.

Conclusion

Research Impact and Future Directions

References

[1] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. Dahl, A. Vaswani, K. Allen, C. Nash, V. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li, and R. Pascanu, "Relational inductive biases, deep learning, and graph networks," 2018.

[2] S. J. Prince, *Understanding Deep Learning.* The MIT Press, 2023.

[3] M. Cranmer, "Interpretable machine learning for science with pysr and symbolicregression.jl," 2023.

[4] M. Cranmer, A. Sanchez-Gonzalez, P. Battaglia, R. Xu, K. Cranmer, D. Spergel, and S. Ho, "Discovering symbolic models from deep learning with inductive biases," 2020.

Appendix: Declaration of AI generation tools

Declaration of AI generation tools for the report part is made here:

Although some of the advise is not accepted, ChatGPT was used to help this report:
1. The format of the mathematical calculations was helped by it to make the process in a publication quality.
2. The format of plots, tables and itemized expression was helped to make things in a publication quality.
3. It suggested some alternative wording and other academic language issues throughout the whole executive summary.
4. It provided proofreading for some of text, including the description of the methodology, results and the overall logic flow among paragraphs.