We thank the reviewers for giving positive and insightful evaluations of our paper. We will adjust the camera-ready version to improve clarity of explanations and address all other comments. Specific responses are given below.

Related work. Thank you for introducing us to two new symbolic regression (SR) papers: Peterson (2020), and Sahoo et al. (2018). We will discuss these in our paper. Note that our work is slightly orthogonal to these approaches: SR algorithms like these and Eureqa, dCGP, and [34] are techniques to search an equation space with few input variables in the raw symbolic regression setting. Our paper gives a way of extending any of these to high-dimensional problems, by factorizing the problem into low-dimensional sub-problems corresponding to a neural network's structure. Consider attempting to fit the equivalent of both the edge model and node model, simultaneously, with any SR approach. In a typical setup, the number of possible functional forms for these under a complexity upper bound of 10 tokens, and only one latent variable, is approximately 10^9 equations each, but explored simultaneously, there are $(10^9)^2 = 10^{18}$ possibilities. However, by using our strategy to learn these models with a neural network first, and fit symbolic regression to the edge and node model independently, we go from 10^{18} to 2×10^9 considered equations. This speedup grows for multiple latents. Future work could replace Eureqa inside our framework with more sophisticated SR backends.

Comparison of SR packages. We will discuss a comparison of these SR packages, hyperparameters (fixed in Eureqa), and why we chose Eureqa—we realize now that many at NeurIPS would find this useful. Furthermore, we will demonstrate a comparison of our approach against a pure SR on one of our case studies. We have previously performed this experiment and found that Eureqa performed very poorly without our framework.

Udrescu & Tegmark (2019) [34]. This Eureqa alternative is optimized for rediscovering existing equations by, e.g., requiring as input the equation's constants, which allows for dimensional analysis (a trick to recall equations in physics). This approach does not seem applicable for discovering new equations so we chose Eureqa. Their description "recursively break hard problems into simpler ones with fewer variables" refers to a common search strategy in SR to determine feature importance (related to feature selection in decision tree learning), dissimilar to our framework.

Other architectures. Our procedure is not restricted to GNNs. However, the inductive bias of the architecture determines the structure of the recovered equation. GNNs with sum-pool give the form $y_i = f(x_i, \sum_j g(x_i, x_j))$. We focused on GNNs since this equation matches our problems, yet one can apply our framework to alternate inductive biases. We will add a discussion on additional example architectures.

Other domains. Regarding tests for alternate domains, symbolic priors might work best for problems where we know simple analytic equations already predict accurately: physics, chemistry, engineering, etc., whereas, e.g., biology and behavioral sciences lack strong analytic models. Perhaps there exists a deep unknown reason for this, or maybe one could use our framework to discover new equations in these domains. Regardless, we hope that our paper will stimulate future research to apply our framework to new architectures and datasets.

Lottery ticket hypothesis. We thank the reviewers for making this connection to the bottleneck model results. We will incorporate this in the discussion.

Generalization on simulation data. We did not test generalization performance on our simple n-body datasets because our framework recovered the ground truth equation. Thus, since Newton's law generalizes to any number of bodies, this equation will give a loss of zero on any dataset with the same force law. However, in the dark matter experiment, we also demonstrate better generalization (Section 4.3 - Symbolic generalization) than the trained neural network, despite this simulation being entirely unlike a simple analytic equation. This simulation integrates the complex and noisy (chaotic) dynamics of dark matter for a million CPU hours. Yet the analytic equation obtained via our framework, describing a pattern in the output dark matter dataset, generalizes significantly better (error=0.0892) than the same neural network it was extracted from (error=0.142). The question: "Where and why does a symbolic prior improve generalization?" presents a very intriguing problem and motivates future research.

Metrics. We will update Table 1 to use R^2 as a metric instead of mean square error to make results more intuitive.

Quantifying SR. As suggested, we will create a table showing success/failure for each force law SR reconstruction.

Table 2. "Formula" represents the recovered analytic expression to predict overdensity. These formulas contain a variable "e_i", which represents a sum of some scalar function over nearby dark matter, given in the adjacent column.

Generative models. We have so far not experimented integrating generative models in our framework, but this sounds like a very interesting approach, and we look forward to exploring this direction in future work.

Broader impact. We are also unaware of potential misuses, but will discuss any suggestions.

Global models. We studied GNNs with an edge model and node model (Newtonian, dark matter), a global model (Hamiltonian), but not all three models together. One could approach such a GNN with the same technique: sparsify latents, apply SR to each model, compose.