Many thanks to all three reviewers for appreciating our work and providing helpful comments and questions.

To Reviewer 1

1. *Why is the sigma-algebra formulation useful*: It allows us to compare the representation power of different classes of GNNs succinctly, precisely by comparing the σ-algebras generated by them. For example, if the σ-algebra generated by the class of type A GNNs is a sub-σ-algebra of that of type B GNNs, then we know that A is no more powerful than B.
2. *Sigma-algebra in the case of infinite feature space*: An extension to countably infinite space is straightforward, while for uncountable space we would need more technical results on σ-algebras. We therefore leave it for future efforts.
3. *Figure 1 & proofs:* The inclusions in Fig. 1 have been establish in the literature, which we referenced in Appendix C. We will add the references per arrow in the caption. We will also add more explanations and illustrations to the proofs.
4. *More thorough experiments*: We extended the numerical results to other standard datasets: collab, mutag, proteins and ptc, and obtained competitive performance as shown below. Moreover, we found that in the GIN paper's experiments, node degrees were added as node features for the social network datasets, which we did not add in the experiment reported in our paper. After adding them, Ring-GNN's performance on IMDB datasets also improved.

	imdbb	imdbm	collab	mutag	ptc	proteins
Ring-GNN	73.3±4.9	51.3±4.24	80.12±1.4	86.8±6.4	65.7±7.13	75.65±2.93
GIN	75.1±5.1	52.3±2.8	80.2±1.9	89.4±5.6	64.6±7.0	76.2±2.8
G-invariant	72.0±5.54	48.73±3.41	78.36±2.47	84.61±10	59.47±7.3	76.58±5.49

To Reviewer 2

1. *Adding SVD to Ring-GNN is unfair*: Indeed, SVD is quite powerful. We added it to the Ring-GNN model to produce a theoretically very expressive object without going to higher order tensors. However, the fact that the Ring-GNNs are strictly more expressive than order-2 G-invariant networks and spectral GNNs do not rely on adding the SVD. Moreover, the results in the table above show that Ring-GNN *without* SVD can achieve competitive performance on real datasets.
2. *Relation to Bloom-Reddy and Teh*: Their work provides a very nice and general theoretical framework that establishes equivalence between functional and probabilistic perspectives to symmetry via noise outsourcing in both general and particular settings. Our framework belongs to the functional perspective to symmetry (in particular $\mathbb{S}_{n_2}$-invariance), and an extension to the probabilistic perspective with ideas from Bloom-Reddy and Teh would be quite interesting. The concept of orbits also applies in our setting, and the concept of maximal invariants is related to our definition of GIso-discriminating. However, a key distinction is that being a maximal invariant is a property of functions, whereas we define GIso-discriminating to be a property of *classes* of functions. Our definition is arguably more suitable for studying the representation power of different GNN architectures, and moreover makes it possible to relate graph isomorphism testing to function approximation. Furthermore, our theoretical framework described in section 4 focuses on sigma-algebras generated by classes of GNN functions when they are not necessarily GIso-discriminating, allowing us to compare their representation powers to each other, which is another novel contribution.

To Reviewer 3

1. *Ring-GNN not guaranteed to be universal*: It is correct that Ring-GNN is not guaranteed to be universal. In fact we don't expect it to be, given that arbitrarily high order tensors are needed to universally approximate [13,17]. However, it is provably more powerful than order-2 G-invariant graph networks, to which Ring-GNN is an extension.
2. *Comparison with LanczosNet*: Indeed, LanczosNet is an interesting model that captures multi-scale information on graphs, which we will add to the references. But firstly, as the initial vector for the Lanczos algorithm is chosen at random, the output of the network is not invariant/equivariant, therefore it's a priori not suitable for graph isomorphism testing. Secondly, as the reviewer pointed out, Ring-GNN is able to express matrices of the form of $min(A^{2^J}, 1)$, which contains some information not present in A^{2^J}. For example, consider the two following graphs with 6 nodes and identical node features: G1 is a "hexagon" and G2 is two disconnected "triangles". If we propagate the node features under the adjacency or Laplacian, identical results would be obtained on the two graphs, regardless of the number of iterations. In the language of the LanczosNet paper, the two graphs have identical Krylov subspaces, and hence we suspect that LanczosNet may have trouble telling them apart. On the other hand, propagating the node features under $min(A^2, 1)$ leads to differences between the two graphs, making them distinguishable by Ring-GNNs.
3. *High computational complexity of Ring-GNN*: One computational bottleneck of Ring-GNN is the SVD, which is of theoretical interest but in practice it can be removed with comparative empirical performance (see table above).
4. *Theoretical analysis does not include node features*: Our analyses do apply to the case where nodes have features actually, though admittedly the notations might be slightly misleading. In our setup, $G \in \mathcal{X}^{n \times n}$ contains both node and edge features: the diagonal entries correspond to node features, and the off-diagonal ones correspond to edge features. If node features are d dimensional vectors, we can have $\mathcal{X}$ (compact) $\subseteq \mathbb{R}^d$, and the theoretical analyses still apply.
5. *Why Ring-GNN performs worse than GIN on IMDB datasets*: On one hand, we note that experimental performance does not strictly correlate with representation power. On the other, in our latest results, Ring-GNN outperformed GIN on the ptc dataset and had improved accuracy on the IMDB datasets, as explained in the "To Reviewer 1" section above.