We thank the reviewers for their positive comments and constructive suggestions. We first address the key common concerns and then move to individual reviewers. The paper will be updated accordingly in the camera-ready version.

Comparison with prior works (R1, R2 and R4): Besides the novel theoretical contribution, our work distinguished itself from prior works on Bayesian sparse neural network by imposing a spike-and-slab prior with the Dirac spike function. Hence automatically, all posterior samples are from exact sparse DNN models. In contrast, Blundell et al. (ICML 2015) and Deng et al. (NIPS 2019) considered spike-and-slab priors with a Gaussian and Laplacian spike respectively, while Ghosh et al. (ICML 2018) chose to use the popular horseshoe shrinkage prior. These existing works actually yield posteriors over the dense DNN model space despite applying sparsity induced priors. In order to derive explicit sparse inference results, user has to additionally determine certain pruning rules on the posterior, which is similar to variational dropout (Molchanov et al. ICML 2017). Thus in our mind, they still belong to the category of Bayesian pruning methods. However, we do agree with the reviewers that additional experiments could be made for better comparison. For example, we conduct Horseshoe BNN under Simulation setting I: it achieves test RMSEs of 1.02 ± 0.01 (A) and 1.24 ± 0.03 (B), with neuron-wise sparsity ratio of 0.79 ± 0.19 (A) and 0.81 ± 0.09 (B). This performance is worse than our method. Note that more experiments will be added in the final version.

Extend current result to more complicated network models (R1, R2 and R3): Our developed theory should work as long as it is a regular DNN (i.e., networks only involve fully-connected layers). Unfortunately we are unable to perform super large-scale experiments due to limited computing resources at hand. Extending current results to more complicated networks (convolutional layer, residual network, etc.) is not trivial. Conceptually, it requires design of structured sparsity (e.g., group sparsity in Neklyudov et al. NIPS 2017) to serve the purpose of faster prediction. Theoretically, it requires deeper understanding of the expressive ability (i.e. approximation error) and capacity (i.e., packing or covering number) of the network model space. By intuition, we conjecture that the generalized theory should still hold, but it will be a future work to provide rigorous theoretical support. To illustrate the practical value of our method for complex tasks, as a preliminary experiment, we apply a 2-Conv-2-FC network on Fashion-MNIST. The testing accuracy is 90.07% with 60% sparsity (connection-wise), where the baseline by dense model is 90.65%.

R2: Please refer to common concerns for major questions, and below is our response to minor questions:
- To induce structured group sparsity, say in CNN modeling, we can let all weights in the same filter share the same binary indicator, such that the whole filter is turned on/off simultaneously. The theoretical correctness of this approach is left as our future work.
- The reviewer is correct. Molchanov et al. (ICML 2017) is a Bayesian pruning method, and the word "frequentist" on Line 112/113 is a typo that should be removed.
- The five datasets are chosen for numerical study since they have fairly larger sample size.
- Throughout the paper, we always report the standard deviation. The "standard error" mentioned in the supplementary material is a typo.

R3: Our results can be extended to different variation distribution families as long as they contain a Dirac component at zero (although the technical details might differ), and the choice of normal slab distribution in this paper is merely for technical simplicity. Without such a Dirac component, the variational posterior can no longer automatically induce sparse inferences. Therefore, those cases are not under our consideration.

The authors don't possess cutting-edge infrastructure, thus reporting the CPU/GPU time of our implementation is not very meaningful. Relatively, under same computational environment, our method takes roughly twice more time than the frequentist counterparts, which is majorly due to the reason that there are more parameters to optimize in our algorithm. Developing more computational efficient algorithm could be a future direction.

R4: The proposed choice of λ does encourage posterior sparsity, but it doesn't play a dominating role on how sparse the posterior would be. Instead, given the value of λ, it is the likelihood of the data that helps to adaptively choose the data-dependent optimal posterior sparsity level. λ plays as a regularization, and we provide theoretical suggestion such that it doesn't mask the importance of the data likelihood. In contrast, pruning methods such as AGP and LOT require user-input knowledge to explicitly determine the sparsity level of the result. In our implementation of AGP and LOT, we tried grid search for the sparstiy levels ranging from 95% to 5%, and the level that yields the best testing accuracy is chosen and reported.

For minor mistakes,
- Thanks for pointing out the typo and the outdated references. In Line 18, the sum should be iterated over $\gamma \in \Gamma^T$.
- Condition 4.4 is not required by Lemma 4.1 since we prove for cases either $\lim n(r_n^* + \xi_n^*) = \infty$ or $\lim n(r_n^* + \xi_n^*) \neq \infty$.
- For the question raised, as has been described in Line 214/215, the posterior sparsity is measured by $\sum_{i=1}^{T} \phi_i / T$ and an individual connection (of the VB point estimator) is considered inactive if $\phi_i < 0.5$.