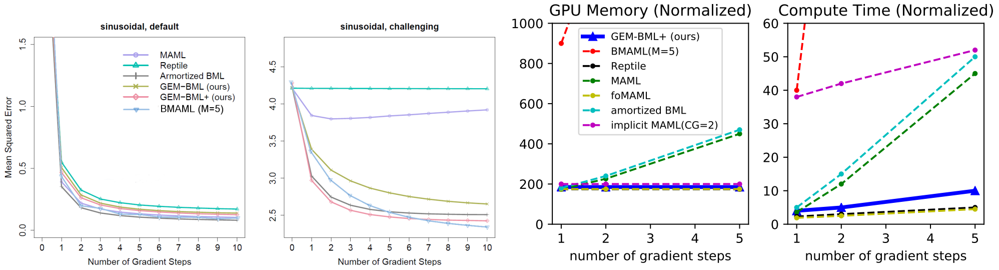

Contribution I: General Bayesian Framework Our work is motivated by a general question: what is the optimal solution of meta learning, or how can we extract information from known tasks to help the most in the new tasks? Our approach started with a generalized meta learning setting in 2.1, where data in meta-training phase is not artificially divided into training/testing as in previous work. Then we provided Theorem 1 as the foundation of our general Bayesian framework by claiming its optimality under certain metrics. We agree with Reviewer #4, that the mathematical techniques are easily derived, nonetheless, the theory on $L^{[2]}$ method has not been established in the field of meta learning to the best of our knowledge (e.g. empirical Bayes focuses on $L^{[1]}$ with fixed number of tasks). As shown, $L^{[2]}$ is essential for NN based models, especially in RL, which highlighted our contribution of GEM-BML+ algorithm.

Contribution II: EM Style Algorithm Our general Bayesian framework inspires EM style (or coordinate descent, as Reviewer # 4 pointed out) algorithms where meta-update only depends on the solution of each inner-update. This is in contrast to MAML style where meta-update depends on the process of each inner-update (path dependence). This is an important contribution because it brings huge potential on the improvement of efficiency, scalability and flexibility compare to MAML style algorithm (Line 37-50, Appendix B5). Side note on iMAML: this gradient-based meta-learning method is not within our framework though it shares the EM style. We didn't think it's important to include iMAML in performance comparison because its EM style update is by the effort of approximating MAML rather than a Bayesian approach. Also, it requires Hessian as second-order information and solving QP at every meta-update step (Line 303).

Contribution III: Summary and Future Work As recognized, we provided a nice summary on a variety of related work as in Table 1. However, one major contribution is that Table 1 can be extended with more columns (e.g. one more column of $L^{[2]} = E_{\tau \in \mathcal{T}} L(\Theta; D_\tau^{tr} \cup D_\tau^{val}) - L(\Theta; D_\tau^{tr})$) and more rows to a larger matrix with some blank elements (models) that haven't been explored before. For example, KL-Chaser Loss model in the right bottom of Table 1 hasn't been studied before. We leave the thorough study of all combinations to future work. This kind of recasting/summary is novel to the best of our knowledge. [E.Grant 2018 Recasting] provided a qualitative recasting of MAML to EB using the loose connection between early stopping and implicit prior, which is totally a different story from our work. Amortized BML directly goes into ELBO gradient and BMAML proposes "learn to infer". We hope this work not only provides clear overview of methods in meta learning, but also sheds light on future work.

Responses to specific questions

Reviewer # 1: a) We did compare with iMAML in Figure 2(see also the above Figure) b) If we assume independent distribution of neural weights (Line 203) between NN layers (which is the case in our experiments and almost all implementations of Bayesian NN), Line 4 of GEM-BML(+) can be written as the sum of different layers and therefore the parallelization will give exactly the same results as our experiment with a reduction in computational time. c) A large portion of "related work" is actually within the body(Section 2,3,4 and Appendix B6).

Reviewer # 2: For Figure 1(a), each color refers to a task and each point is regarding a set of model weights (NN weights) which has good performance on that task. This figure demonstrates that many good solutions exist for each task (local optimums of NN,Line 99). The dotted line area is the small neighboring zone A^* where each color has at least one point inside (Section 2.3 and Appendix B.2 provide further explanation).

Reviewer # 3: We choose Amorized BML as the benchmark of Bayesian meta-learning to compare(typo:ELBO=>Amortized BML) because its performance is comparable to all other previous Bayesian meta-learning algorithms. We actually included BMAML in a previous version of our paper(See the Figure above for both performance(left two,blue) and efficiency(right two,red)). The efficiency of BMAML is not within feasible range. As for PMAML, the code is not released and the algorithm is complex to reproduce.

Reviewer # 4: We realized that "GEM is the Bayesian version of Reptile" can mislead to a trivial impression on our work. We would like to point out that, our method was not motivated by "how can we improve from Reptile" (like from MAML to BMAML), but from a total different perspective. The connection with Reptile was a coincidence discovery (while we were working on the recast/summary). Lastly, Reptile can not be applied to RL (as stated by its author and confirmed in our experiments) while our method performs excellently.