We would like to thank the five reviewers for their feedback. We provides point-by-point response to their concerns. We first discuss a few common concerns shared by **Reviewer 50** and **Reviewer 90** on the performance of MISSO in he numerical section:

• • **Comparison with MC-ADAM (Figure 2):** In order to avoid the confusion in the convergence of MISSO in the Resnet-18 example, we will extend the number of training epochs to 200. 200 epochs will stabilize the ELBO for MISSO, reaching similar values as MC-Adam. While both methods reach the same local minima, we acknowledge that in this particular example MC-Adam does reach an $\epsilon-$stationary points in fewer iterations than MISSO, probably suffering from a high variance, hence the twists and turns in the objective function. At this stage, we are not sure if it is possible to generalize a class of problems where baselines can beat MISSO in terms of convergence speed.

Reviewer 5: We thank the reviewer for valuable comments and reference. We would like to make the following clarification regarding the stationary measure we use:

– **Stationary Measure:** In Theorem 1, both rates in (16) and (17) are given with respect to first, the second order moment of the gap between the (potentially nonsmooth) objective function and the surrogates, and to secondly, the negative part of the directional derivative. For (16), the existence of the gradient of the gap is ensured by assumption H2 (adding the nonsmooth components of the objective function in the surrogates is an easy trick to ensure its existence). For (17), since our problem is constrained, we must have recourse to the directional derivative of the objective function, and more precisely to the convergence of its negative part to 0 as a characterization of stationarity.
Techniques in [Shamir, 2020] are useful when the suboptimality condition is derived on a nonsmooth quantity, which is not the case here.

Reviewer 9: We thank the reviewer for the useful comments. Our point-to-point response is as follows:

– **Exact Surrogate Minimization:** The advantage of our method is that the surrogate function is user designed. hence, its choice can be made so that the maximization step can be done in closed form at each iteration. In our numerical examples, we use quadratic surrogate of the objective function, leading to its exact minimization via a gradient step. The updates are detailed in the supplementary material for completeness, see for instance section B.3, equation (42) for the logistic regression example.

– **Proof Techniques:** The technicality of our proofs partly relies on mixing results from the MCMC literature, such as the Bracketing number and the control of the Monte Carlo noise, see references [25;26;27], with stochastic nonconvex optimization convergence bounds proofs techniques, generally used for gradient descent type of algorithms. The handling of both sampling and optimization procedure into a single theoretical framework makes the derivation of our bound challenging. While Theorem 2 is somewhat related to the convergence theorems in [Mairal, 2015], the finite-time analysis for such *incremental* and *doubly stochastic* method is not common in the literature.

Reviewer 35: We thank the reviewer for valuable comments. We clarify the following point on the experiments:

– **Numerical Experiments:** Thank you for the pointer. We plan on extending the number of epochs for all methods. With 200 epochs, MISSO does converge and reaches similar local minima as MC-Adam, while MC-Adagrad stays stuck in a worse local minima, as per our runs. We will plot a stabilized curve for MISSO in the revised paper.

Reviewer 50: We thank the reviewer for the valuable comments and typos. We would like to emphasize on the nature of our contribution, given some runs where MISSO does not display a particular edge over baselines:

– **Nature of the contribution:** We want to stress on the generality of our incremental optimization framework, which tackles a *constrained*, *non-convex* and *non-smooth* optimization problem. The main contribution of this paper is to propose and analyze a **unifying framework** for a large class of optimization algorithms which includes many well-known but not so well-studied algorithms. The major idea here is to relax the class of surrogate functions used in MISO [Mairal, 2015] and to allow for intractable surrogate that can only be evaluated by Monte-Carlo approximations. We provide a general algorithm and global convergence rate analysis under mild assumptions on the model and show that two examples, MLE for latent data models and Variational Inference, are its special instances. Working at the crossroads of *Optimization* and *Sampling* constitutes what we believe to be the novelty and the technicality of our theoretical results.

Reviewer 90: We thank the reviewer for the interest in our contribution. We also precise the following:

– **Comparison with Monte Carlo variants of baselines:** Given the generality of the MISSO framework, as recalled at the beginning of this rebuttal, the comparison with Monte Carlo variants of other baselines will have to be on a case by case study. As MISSO encompasses several methods such as MCEM or Variational Inference, their comparable baselines are usually different. Those competitors also highly depend on the type of model that is being trained. Here again, MISSO aims at tackling a large collection of latent variable models. Yet, when this latent variable model is a multilayered network or a plain linear mixed model, the most performing baselines will be of different nature. Hence, we argue that MISSO enjoys from its broad scope of possible algorithms for a large class of models, while displaying strong convergence guarantees.
From a practical point of view, the simplicity of its update in Algorithm 2 is enjoyable compared to various baselines such as Adam where several (possibly computationally heavy) estimations are calculated. Besides, to the best of our knowledge, the Monte Carlo variants of those baselines have not been studied. The convergence of relatively complex algorithms such Adagrad or Adam, while adding this layer of stochasticity, is far from easy to obtain.