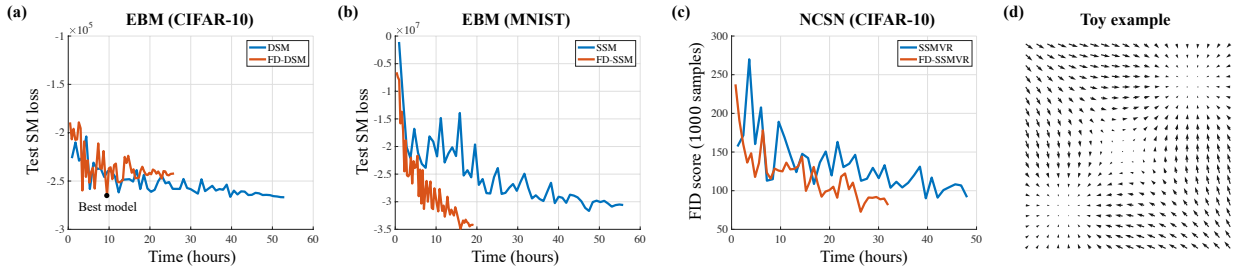

Figure A: **(a)** Loss for DSM and FD-DSM; **(b)** Loss for SSM and FD-SSM; **(c)** FID scores on 1,000 samples (higher than those reported on 50,000 samples in the paper) for SSMVR and FD-SSMVR; **(d)** Score distribution on toy example trained by FD-SSM. The ranges of x and y axis are both $[-8, 8]$.

We thank all the reviewers for their valuable comments. Below, we address the detailed comments of each reviewer.

To Reviewer #1. **Empirical evaluation:** Thank you for the suggestion. In the revision, we will move more details into the full paper to improve the reading experience. **Flow-based models:** For fair comparisons with the SM baselines (e.g., SSM), our evaluation tasks mainly follow the settings in Song et al. [57], including the flow-based experiments. Technically, Table 3 shows that our method can also outperform on invertible architectures, where we can regard flow-based models as special instantiations of EBMs. **About additional feedback:** We really appreciate for the detailed caveats list, and we will carefully polish them in the final version. As to the reasonable suggestion on the title, we will also discuss on it, thank you! Below we answer the remaining questions in the list. 20-21: Here the better stability and mode coverage are compared to GANs. We will make the claim more precise. 193: Yes, the subscript of norm should be 2, which is a uniform distribution on the hypersphere. 251: For deep EBMs, we experiment on the range of $\epsilon \in [0.01, 1]$ and obtain comparable model performance. We will include complete results in the revision.

To Reviewer #2. **Practical improvements:** Indeed, SM methods only involve second- or third-order derivatives, which may not result in dramatic speedup using our FD reformulation. Thus our main future work is to better exploit our FD formulas in other higher-order cases. Some potential candidates include applications in meta-learning and those described in L150-153. **Variance of FD:** As discussed in L314-317, the random projection trick required by the FD formula is its main downside, which may outweigh the gain on efficiency for low-order computations. We provide the loss curve of DSM and FD-DSM w.r.t. time in Fig. A(a). As seen, FD-DSM can achieve the best model (lowest SM loss) faster, but eventually converges to higher loss compared to DSM. In contrast, as shown in Fig. A(b)(c), when applying FD on SSM-based methods, the improvements are much more significant. **Repeating experiments:** Thank you for the suggestion. We will repeat the experimental trials more times to alleviate the effect of randomness.

To Reviewer #3. **Memory usage:** Our method also consistently improve the memory usage across different models and datasets. For example, on the NCSN model, SSM and FD-SSM use 6.3G and 5.4G memory per GPU (both run on four GPUs), respectively. We will include more complete results on memory usage in the revision. **Previous work on Lemma and Theorem 1:** As far as we know, we can only find out the reference like [21] on the univariate case (still slightly different from the one-dimensional instantiation of our formulas). We will keep searching for the related literature on the multivariate versions and add proper references then. **Performance gap of NCSN:** The published FID results in NCSN are based on DSM. Although DSM is a biased estimator, it naturally adapts to the annealed Langevin dynamics used in NCSN, compared to SSM. As in L284-285, our NCSN experiment is mainly to demonstrate that our method makes the unbiased SSM (VR) estimator more efficient on score-based networks. We will explain this clearly. **Clarity:** We really appreciate the suggestions and the typos list. We will carefully polish them in the revision.

To Reviewer #4. **Applicable cases:** Our FD formula is a general technique to estimate the directional derivatives and their related objectives. Some typical applicable cases include gradient norms, gradient projection, Hessian trace, and Hessian Frobenius norm, etc. **Toy example:** We adopt the data distribution $0.8\mathcal{N}([5, 5], I)+0.2\mathcal{N}([-5, -5], I)$ as a 2D toy example. We use a three-layer MLP as our EBM model and set $\epsilon = 0.1$. The result of the score distribution trained by FD-SSM is given in Fig. A(d), which is almost the same as SSM (full details will be added). **Sensitivity analysis of ϵ:** Here we report the test NLLs for DKEF model on the Parkinson dataset: $14.17(\epsilon = 0.1)$, $13.51(\epsilon = 0.05)$, $14.03(\epsilon = 0.02)$, $14.00(\epsilon = 0.01)$, which indicates the insensitivity for $\epsilon \in [0.01, 0.1]$. Complete results on EBMs w.r.t. ϵ will be involved in the revision. **Trace plot of SM loss:** The plot is given in Fig. A(b) for SSM and FD-SSM. We can see that FD-SSM has significantly better efficiency and is consistent with SSM. **Large-scale experiments:** FD-SSMVR is the FD version of SSMVR, as formulated in L204-212. As described in [57], SSMVR is a variance reduction form of SSM when the projection distribution $p(v)$ satisfies certain conditions. Since we experiment on various combinations of datasets, models, tasks, and SM methods, we attempt to make the results as diverse as possible in the limited space. We will involve more complete results in the revision. **Proof of Theorem 2:** Our FD reformulations are asymptotically unbiased when $\epsilon \to 0$. Thus, under the condition in Lemma 2, $\forall \delta > 0$, there $\exists \epsilon$, such that the remainder term (estimation bias) can be uniformly bounded by δ in B, i.e., the gradients of FD-SSM and SSM can be sufficiently aligned in B. As $\epsilon \to 0$, the set B could tend to $\mathbb{R}^d \times \mathcal{S} \setminus \{\text{stationary points}\}$, and the optimization track can converge in B to the stationary points (may not exactly locate on). **Comparisons to other AD methods:** Thank you for the suggestion. We will try to add extensive experiments on other types of automatic differentiation methods in the revision.