We thank the reviewers for their constructive feedback and address the common concerns across the four reviews.

Difference with REINFORCE We agree we didn't clearly explain the difference between our estimator and REINFORCE. We have made the following changes to the manuscript.

1) Rather than stating "the gradient estimator we derive ... does not require the high-variance REINFORCE..." in the introduction and throughout, we now say we derive a "score function estimator" to emphasize the fact that our estimator belongs in the family of estimators that use the log-derivative trick.

2) We have added a new appendix section clarifying the relationship as follows. Assuming $\pi_\beta(\mathbf{z}) = \hat{\pi}_\beta(\mathbf{z})/Z_\beta$ depends on parameters λ, to compute $\nabla_\lambda \mathbb{E}_{\pi_\beta(\mathbf{z})}[f(\mathbf{z})]$ one can use the:

$$\begin{aligned}
\text{REINFORCE ESTIMATOR:} \quad & \mathbb{E}_{\pi_\beta(\mathbf{z})}\left[f(\mathbf{z})\nabla_\lambda \log \pi_\beta(\mathbf{z})\right] \\
\text{REINFORCE ESTIMATOR + BASELINE:} \quad & \mathbb{E}_{\pi_\beta(\mathbf{z})}\left[\left(f(\mathbf{z}) - \mathbb{E}_{\pi_\beta(\mathbf{z})}\left[f(\mathbf{z})\right]\right)\nabla_\lambda \log \pi_\beta(\mathbf{z})\right] \\
\text{COVARIANCE ESTIMATOR (ours):} \quad & \mathbb{E}_{\pi_\beta(\mathbf{z})}\left[\left(f(\mathbf{z}) - \mathbb{E}_{\pi_\beta(\mathbf{z})}\left[f(\mathbf{z})\right]\right)\left(\nabla_\lambda \log \hat{\pi}_\beta(\mathbf{z}) - \mathbb{E}_{\pi_\beta(\mathbf{z})}\left[\nabla_\lambda \log \hat{\pi}_\beta(\mathbf{z})\right]\right)\right]
\end{aligned}$$

We emphasize that our estimator applies in the general case of expectations over $\pi_\beta(\mathbf{z})$, where the REINFORCE estimator $\mathbb{E}_{\pi_\beta(\mathbf{z})}[f(\mathbf{z})\nabla_\phi \log \pi_\beta(\mathbf{z})]$ would require differentiating through $\log \pi_\beta(\mathbf{z})$ which contains the intractable normalizing constant. Additionally unlike REINFORCE, where a baseline is typically added ad-hoc to reduce variance, the baseline $b = \mathbb{E}_{\pi_\beta(\mathbf{z})}[f(\mathbf{z})]$ naturally appears as a result of differentiating through $\pi_\beta(\mathbf{z})$ using the identity $\nabla_\lambda \log Z_{\lambda,\beta}(\mathbf{x}) = \mathbb{E}_{\pi_\beta(\mathbf{z})}[\nabla_\lambda \log \tilde{\pi}_{\lambda,\beta}(\mathbf{z})]$ derived in appendix E.

Low variance To address concern 2.6 of reviewer 1 directly, in section 6.1 (Figures 3 and 4) we compare the TVO against VIMCO which uses REINFORCE updates, and in 6.2 we compare the TVO against VAEs and IWAE which use the reparameterization trick. In figure 4 we plot the ϕ gradient std of a discrete VAE and compare against VIMCO and reweighted-wake sleep. Our method has lower variance than VIMCO which uses REINFORCE updates. However we agree we need to discuss the low variance properties of our estimator further. We have made the following changes to the manuscript:

1) After clarifying the relationship between the covariance estimator and REINFORCE in the aforementioned addition to the appendix, we observe the $\mathbb{E}_{q_\phi(\mathbf{z})}[f(\mathbf{z})]$ baseline that is implicit in our covariance estimator is equivalent to the "average baseline" commonly used in reinforcement learning to reduce variance. This provides a theoretical justification for using the average baseline which is typically chosen because of empirical success and intuitive appeal.

2) We include additional experimental results to report the effect of using the covariance estimator to train a model on the ELBO, and compare against the same model trained using REINFORCE and the reparameterization trick. We plot the std. of the θ, ϕ gradients across 10 trials. Preliminary results indicate the variance of our estimator is empirically equivalent to the variance of the reparameterization trick, while the variance of the reinforce estimator is unstable despite having 10x samples.

3) We include a table explicitly reporting the mean gradient std of the TVO compared to the REINFORCE-based VIMCO (reproduced below) and update the writing in the experimental section to make it clear that section 6.1 and 6.2 are designed to compare our method against REINFORCE and the reparameterization trick respectively.

4) The second source of variance reduction comes from using the 'Common Random Numbers' (CRN) technique from Owen (chapter 8.6), which we now refer to by name in the manuscript. The terms in the TVO are highly correlated, thus we expect reusing a single batch of samples for each additional term will act to reduce variance according to equation 8.21 in Owen. However, because the covariance term breaks into both positive and negative terms, CRN could potentially increase variance. We therefore have included an additional experiment running the TVO with and w/o CRN and include the results in tabular form below.

Connections to Wake-Sleep The endpoints of the TVI, which the TVO approximates, corresponds to the two objectives used in Wake-sleep to jointly learn a generative model and inference network. Therefore we can view the objectives as two approximations (a left and right Riemannian sum) of a single objective, the TVI. In section 4 we discuss how the left endpoint (i.e. $\beta = 0$) corresponds to the first objective (the wake-phase θ update) which in turn corresponds to the ELBO discussed in detail in section 1. The right endpoint (i.e. $\beta = 1$) corresponds to the second objective, of which there are two variants (the wake-phase ϕ update and sleep-phase ϕ update) determined by whether one uses real or simulated data. Revisiting the text we feel we relied too heavily on the description of WS given by (Le et al, 2018b) and will revise the manuscript to make this connection more clear.

Particles	2	5	10	50
Reinforce (VIMCO)	4.48	8.10	5.72	5.57
TVO	5.48	4.31	2.39	1.34

Iterations	10	10m	20m	30m	40m
TVO w/o CRN	52.33	2.88	2.57	2.39	2.47
TVO w/ CRN	8.19	1.38	1.17	1.05	1.03

Table 1: Mean gradient std across 10 seeds for TVO vs REINFORCE (Left) and TVO with and without common random numbers (CRN) (Right)