RESAMPLING GRADIENTS VANISH IN DIFFERENTIABLE SEQUENTIAL MONTE CARLO SAMPLERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Annealed Importance Sampling (AIS) moves particles along a Markov chain from a tractable initial distribution to an intractable target distribution. The recently proposed Differentiable AIS (DAIS) (Geffner & Domke, 2021; Zhang et al., 2021) enables efficient optimization of the transition kernels of AIS and of the distributions. However, we observe a low effective sample size in DAIS, indicating degenerate distributions. We thus propose to extend DAIS by a resampling step inspired by Sequential Monte Carlo. Surprisingly, we find empirically—and can explain theoretically—that it is not necessary to differentiate through the resampling step which avoids gradient variance issues observed in similar approaches for Particle Filters (Maddison et al., 2017a; Naesseth et al., 2018; Le et al., 2018).

1 SEQUENTIAL MONTE CARLO METHODS AND RESAMPLING

Sequential Monte Carlo (SMC) (Doucet et al., 2001; Liu & Liu, 2001) and Annealed Importance Sampling (AIS) (Neal, 2001)) are related methods to sample from unnormalized distributions and to estimate their normalization constants. SMC simulates the evolution of a set of particles through a Markov chain, where each transition takes three steps: (i) independent transitions of particles using either a given dynamics model (Particle Filters (PFs) (Gordon et al., 1993; Kong et al., 1994)) or Markov Chain Monte Carlo moves that interpolate between an initial and target distribution (SMC Samplers (SMCSs) (Del Moral et al., 2006)); (ii) re-weighting each particle's probability; and (iii) resampling, i.e., replacing particles with low weights by "clones" of particles with high weights.

AIS is similar to SMCSs without resampling (see Appendix A). The recently proposed Differentiable AIS (DAIS) (Geffner & Domke, 2021; Zhang et al., 2021) omits the non-differentiable Metropolis-Hastings (MH) accept/reject step (Metropolis et al., 1953; Hastings, 1970). However, in high dimensions, DAIS suffers—similar to a SMCS without resampling—from a small Effective Sample Size (ESS) (Liu & Chen, 1998), i.e., a highly skewed distribution of particle weights where only few particles contribute (Figure 1, 🔈).

We observe that turning DAIS into a Differentiable SMCS (DSMCS, proposed) by introducing a resampling step (iii) leads to tighter bounds on the normalization constant in a standard experiment (problem setup as in Doucet et al. (2022); details in Appendix B) across a wide range of settings even when gradients due to resampling are ignored ("∗-Cat") (see Figure 2 (a) & (b) where blue uses resampling and red does not; more detailed experimental data in Appendix C). Surprisingly, we observe that taking gradients due to resampling into account ("∗-GST") (red in Figure 2 (c) & (d))—which we do with the Gapped Straight Through estimator (GST) (Fan et al., 2022)—hardly affects the tightness of the bound. The next section explains this finding theoretically.

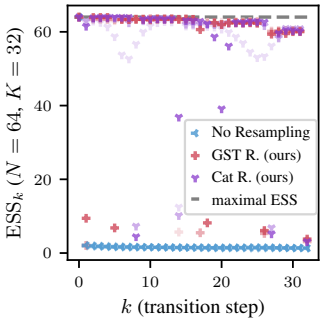

Figure 1: ESS for various DSMCSs at epochs 100, 300, and 500 (increasing opacity).

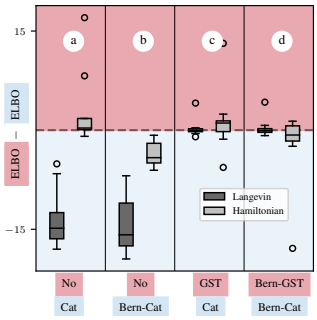

Figure 2: Difference of ELBOs. Red and blue backgrounds show method with larger ELBO (red: upper half & upper row of x-axis).

2 GRADIENTS DUE TO RESAMPLING IN A DIFFERENTIABLE SMCS

We formalize DSMCS and analyze its resampling gradient theoretically. We seek the normalization $Z := \int_{\mathbb{R}^n} \gamma(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ of an unnormalized density γ on $\mathbb{R}^n$. A SMCS samples N trajectories through a series $(\pi_k)_{k=0}^K$ of distributions $\pi_k(\boldsymbol{x}) \propto \gamma_k(\boldsymbol{x}) := \pi_0(\boldsymbol{x})^{1-\beta_k}\,\gamma(\boldsymbol{x})^{\beta_k}$ (with $0 = \beta_0 < \beta_1 < \ldots < \beta_K = 1$), which interpolate between a tractable initial distribution π_0 and the target $\pi_K = \gamma/Z$. First, one draws N independent samples $\{\tilde{\boldsymbol{z}}_0^i\}_{i=1}^N$ from π_0. For each subsequent step $k \in \{1, \ldots, K\}$, one then draws samples $\{\boldsymbol{z}_k^i\}_{i=1}^N$ from a forward kernel $F_k(\boldsymbol{z}_k^i \,|\, \tilde{\boldsymbol{z}}_{k-1}^i)$ and one calculates importance weights $\tilde{w}_k(\boldsymbol{z}_k^i, \tilde{\boldsymbol{z}}_{k-1}^i) := \gamma_k(\boldsymbol{z}_k^i) B_k(\tilde{\boldsymbol{z}}_{k-1}^i \,|\, \boldsymbol{z}_k^i) \,/\, (\gamma_{k-1}(\tilde{\boldsymbol{z}}_{k-1}^i) F_k(\boldsymbol{z}_k^i \,|\, \tilde{\boldsymbol{z}}_{k-1}^i))$, where both F_k and B_k leave π_k invariant. Similar to DAIS we leave out the MH steps in order to make the kernels differentiable. Finally, one (optionally) resamples from within the set of particles by setting $\tilde{\boldsymbol{z}}_k^i := \boldsymbol{z}_k^{\iota_k(i)}$ where the mapping $\iota_k : \{1, \ldots, N\} \to \{1, \ldots, N\}$ is drawn from a distribution $\mathcal{M}(\iota_k \,|\, \{w_k^i\}_{i=1}^N)$ that satisfies that each $\boldsymbol{z}_k^i$ is sampled $N w_k^i$ times in expectation (i.e., $\mathbb{E}_{\mathcal{M}}\left[|\iota_k^{-1}(j)|\right] = N w_k^i$), where $w_k^i = \tilde{w}_k(\boldsymbol{z}_k^i, \tilde{\boldsymbol{z}}_{k-1}^i) / \sum_{j=1}^N \tilde{w}_k(\boldsymbol{z}_k^j, \tilde{\boldsymbol{z}}_{k-1}^j)$. In practice, the simplest way to satisfy this requirement is to draw the function values $\iota_k(i)$ i.i.d. for all i from a categorical distribution with probabilities $(w_k^1, \ldots, w_k^N)$. We combine the derivation in Geffner & Domke (2021); Zhang et al. (2021) and Del Moral et al. (2006) arriving at the lower bound $\mathcal{L} \leq \log Z$ with

$$\mathcal{L} = \mathbb{E}_{[\prod_i \pi_0(\boldsymbol{z}_0^i)] \prod_{k=1}^K [\prod_i F(\boldsymbol{z}_k^i | \boldsymbol{z}_{k-1}^{\iota_{k-1}(i)})]\, \mathcal{M}(\iota_k | \{w_k^i\}_{i=1}^N)} \left[\sum_{k=1}^K \log \left(\sum_{i=1}^N \alpha_k^i \, \tilde{w}_k(\boldsymbol{z}_k^i, \boldsymbol{z}_{k-1}^{\iota_{k-1}(i)}) \right) \right] \quad (1)$$

where the expectation $\mathbb{E}[\,\cdot\,]$ is taken over the sampling process described above, $\alpha_k^i = \frac{1}{N}$ if one resamples in step k and $\alpha_k^i = w_k^i$ otherwise, and $\iota_0 =$ Identity as resampling for $k = 0$ is not useful.

Empirically, resampling tends to achieve high Effective Sample Size $\mathrm{ESS}_k := 1/\sum_{i=1}^N (w_k^i)^2 \in [1, N]$, regardless of whether we take gradients due to resampling into account (Figure 1, ✚) or not (Figure 1, ♥), and it makes the bound on Z tighter (Figure 2, (a) & (b)). We find this to be true both for models that resample at a given step k with probability $1 - (\mathrm{ESS}_k - 1)/(N - 1)$ (Figure 2, "Bern-$*$"), as well as for models that resample for every k (Figure 2, "GST" & "Cat").

It may seem at first sight that resampling could make it more complicated to differentiate through DSMCSs for gradient based maximization of $\mathcal{L}$: as $\mathcal{M}(\iota_k \,|\, \{w_k^i\}_{i=1}^N)$ is a discrete distribution, differentiating through samples from it typically leads to either high gradient variance (e.g., when using REINFORCE gradient estimates) or to biased gradient estimates (e.g., when using the GST estimator). However, the ablation studies in Figure 2 (c) & (d) (discussed in Section 1) indicate that differentiating through the resampling step is not necessary. It turns out that this can be explained theoretically as a consequence of the high ESS that resampling causes:

Theorem 1 *The gradient of the resampling step vanishes if the ESS is maximal, i.e., if $\mathrm{ESS}_k = N\ \forall k$.*

Proof. We rewrite the expectation over $\{\iota_k\}_{k=1}^K$ in Eq. 1 as $\sum_{\{\iota_k\}_{k=1}^K} (\prod_k \mathcal{M}(\iota_k \,|\, \theta_k))[\cdots]$ where θ_k are opaque parameters that allow us to track gradients through $\mathcal{M}$. For each term in this sum, we trace back the indices $\{j_k\}_{k=0}^K$ that lead to index $j_K := 1$ at the last step K by recursively defining $j_{k-1} := \iota_{k-1}(j_k)$. By Jensen's inequality for the strictly convex function $x \mapsto x^2$, an ESS of N implies $w_k^i = \frac{1}{N}\ \forall k, i$, and thus $\tilde{w}_k(\boldsymbol{z}_k^i, \tilde{\boldsymbol{z}}_{k-1}^i) = \tilde{w}_k(\boldsymbol{z}_k^i, \boldsymbol{z}_{k-1}^{\iota_{k-1}(i)})$ is independent of i for all k. We can thus replace the average over i inside the logarithm in Eq. 1 with the single term with $i = j_k$,

$$\mathcal{L}_{(\mathrm{ESS}=N)} = \sum_{\{\iota_k\}_{k=1}^K} \left(\prod_{k=1}^K \mathcal{M}(\iota_k \,|\, \theta_k) \right) \mathbb{E}_{\pi_0(\boldsymbol{z}_0^{j_0}) \prod_{k=1}^K F(\boldsymbol{z}_k^{j_k} \,|\, \boldsymbol{z}_{k-1}^{j_{k-1}})} \left[\sum_{k=1}^K \log \tilde{w}_k(\boldsymbol{z}_k^{j_k}, \boldsymbol{z}_{k-1}^{j_{k-1}}) \right]. \quad (2)$$

Here, we also marginalized trivially over all $\boldsymbol{z}_k^i$ with $i \neq j_k$. The remaining coordinates $\{\boldsymbol{z}_k^{j_k}\}_{k=0}^K$ describe a single trajectory that can no longer branch. Thus, they are *independent* integration variables, i.e., we may rename them by dropping the superscripts j_k to stress that the expectation $\mathbb{E}[\,\cdot\,]$ over particle coordinates in Eq. 2 no longer depends on $\{\iota_k\}_{k=1}^K$. Thus, $\nabla_{\theta_k} \mathcal{L} = 0$. □

Discussion In practice, we only observe $\mathrm{ESS}_k \approx N$ (see Figure 1). However, we can empirically verify that the gradients corresponding to the resampling operation indeed do not seem to have a significant impact on the results (see Figure 2). Notably, leaving out the gradients corresponding to resampling removes much of the gradient variance that has been reported to hurt the estimation of the log normalization constant in PFs (Maddison et al., 2017a; Naesseth et al., 2018; Le et al., 2018).

URM STATEMENT

The authors acknowledge that at least one key author of this work meets the URM criteria of ICLR 2023 Tiny Papers Track.

REFERENCES

Adrien Corenflos, James Thornton, George Deligiannidis, and Arnaud Doucet. Differentiable particle filtering via entropy-regularized optimal transport. In *International Conference on Machine Learning*, pp. 2100–2111. PMLR, 2021. 4

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006. 1, 2, 4

Arnaud Doucet, Simon J Godsill, and C Andrieu. *On sequential simulation-based methods for Bayesian filtering*. Department of Engineering, University of Cambridge Cambridge, UK, 1998. 4

Arnaud Doucet, Nando De Freitas, Neil James Gordon, et al. *Sequential Monte Carlo methods in practice*, volume 1. Springer, 2001. 1

Arnaud Doucet, Will Sussman Grathwohl, Alexander G. D. G. Matthews, and Heiko Strathmann. Score-based diffusion meets annealed importance sampling. 2022. 1, 4, 6

Ting-Han Fan, Ta-Chung Chi, Alexander I Rudnicky, and Peter J Ramadge. Training discrete deep generative models via gapped straight-through estimator. In *International Conference on Machine Learning*, pp. 6059–6073. PMLR, 2022. 1, 5

Tomas Geffner and Justin Domke. Mcmc variational inference via uncorrected hamiltonian annealing. *Advances in Neural Information Processing Systems*, 34:639–651, 2021. 1, 2, 4

Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE proceedings F (radar and signal processing)*, volume 140, pp. 107–113. IET, 1993. 1

W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970. 1

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017. 5

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

Augustine Kong, Jun S Liu, and Wing Hung Wong. Sequential imputations and bayesian missing data problems. *Journal of the American statistical association*, 89(425):278–288, 1994. 1

Tuan Anh Le, Maximilian Igl, Tom Rainforth, Tom Jin, and Frank Wood. Auto-encoding sequential monte carlo. In *International Conference on Learning Representations*, 2018. 1, 2, 4

Jun S Liu and Rong Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American statistical association*, 93(443):1032–1044, 1998. 1, 4

Jun S Liu and Jun S Liu. *Monte Carlo strategies in scientific computing*, volume 75. Springer, 2001. 1

Chris J Maddison, John Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Teh. Filtering variational objectives. *Advances in Neural Information Processing Systems*, 30, 2017a. 1, 2, 4

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017b. 5

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953. 1

Christian Naesseth, Scott Linderman, Rajesh Ranganath, and David Blei. Variational sequential monte carlo. In *International conference on artificial intelligence and statistics*, pp. 968–977. PMLR, 2018. 1, 2, 4

Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11:125–139, 2001. 1, 4

Guodong Zhang, Kyle Hsu, Jianing Li, Chelsea Finn, and Roger B Grosse. Differentiable annealed importance sampling and the perils of gradient noise. *Advances in Neural Information Processing Systems*, 34:19398–19410, 2021. 1, 2, 4

A Related Work

Figure 3 gives a brief overview of how the proposed DSMCS fits into the landscape of (differentiable) SMC and AIS methods.

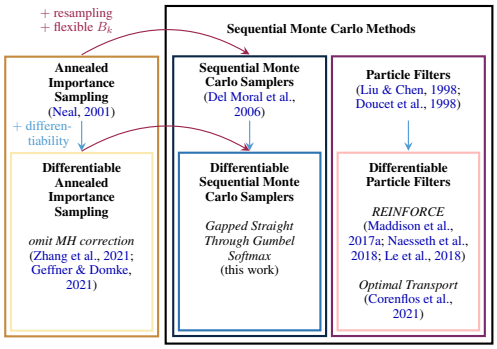

Figure 3: AIS and SMC methods.

B Details on Experiments

We replicate the static target experiment by Doucet et al. (2022). We estimate the log normalization constant of $p = \gamma/Z$ being a 50-dimensional Gaussian mixture with 8 components (note that $\log Z = 0$). The means of the Gaussian mixture p are sampled from the Gaussian distribution $\mathcal{N}(3, \boldsymbol{I})$. The variance of each component is set to 1. The initial distribution $q = \pi_0$ is chosen to be a Gaussian with large diagonal variance, namely $\mathcal{N}(0, 9\boldsymbol{I})$. We run the DSMCS with various resampling schemes for $K \in \{8, 16, 32\}$ and $N \in \{8, 32, 64\}$ utilizing a small neural network predicting the step sizes $\{\delta_k = \hat{\delta} \cdot \sigma(\mathrm{NN}(k))\}_{k=1}^{K}$ with the same architecture as proposed by Doucet et al. (2022). Additionally, we learn the annealing schedule β_k. For the Hamiltonian Markov kernel we also learn the (scaled) mass matrix $\boldsymbol{M} = c\boldsymbol{I}$ and a scalar for the momentum refreshment ρ. All models maximize the bound $\mathcal{L}$ for 500 epochs (each consisting of 10 iterations) with the Adam optimizer (Kingma & Ba, 2014) in batches of 64.

Appendix B.1 gives an overview of the resampling schemes that we evaluate the DSMCS models with on the static target experiment. Appendix B.2 describes the unadjusted overdamped Langevin kernel utilized in our experiments. Appendix B.3 describes the Hamiltonian Markov kernel (that incorporates Hamiltonian dynamics via the underdamped Langevin equation). For a thorough derivation of these kernels, please see the original work of Doucet et al. (2022).

B.1 MODELS

We employ the proposed DSMCS with multiple resampling schemes that either include the resampling gradient or ignore it. The distribution $\mathcal{M}(\iota_k \mid \{w_k^i\}_{i=1}^N)$ takes the following four different functional forms.

- **Cat** (Categorical Resampling (without gradients)): We choose a categorical distribution for $\mathcal{M}(\iota_k \mid \{w_k^i\}_{i=1}^N)$ and draw $\iota_k(i) \sim \text{Cat}(\{w_k^i\}_{i=1}^N)$.
- **Bern-Cat** (Categorical Resampling (without gradients) with Bernoulli Relaxation (without gradients)): We augment the categorical distribution from above with an additional Bernoulli random variable that "decides" whether to resample at iteration k for each batch, namely $\iota_k(i) = (1 - b_k)i + b_k c_k^i$, where $c_k^i \sim \text{Cat}(\{w_k^i\}_{i=1}^N)$ and $b_k \sim \text{Bern}\left(1 - (1/\sum_{j=1}^N (w_k^j)^2 - 1)/(N-1)\right)$. This idea is inspired by SMC where one typically employs a (hard) threshold of $N/2$ and resamples particles if the threshold is undercut.
- **GST** (Gapped Straight Through Gumbel-Softmax Resampling (with gradients)): This resampling scheme is similar to **Cat** but with resampling gradients. We utilize the Gapped Straight Through (GST) estimator (Fan et al., 2022) for the categorical distribution. This estimator is known to reduce gradient variance compared to the Gumbel-Softmax (Jang et al., 2017; Maddison et al., 2017b) estimator. We train models with two temperatures $\tau \in \{0.1, 1.0\}$.
- **Bern-GST** (Gapped Straight Through Gumbel-Softmax Resampling with Bernoulli Relaxation (with gradients)): This resampling scheme builds on the **GST** resampling scheme but also reparametrizes the Bernoulli variable utilizing the GST estimator. We train models with two temperatures $\tau \in \{0.1, 1.0\}$.

B.2 LANGEVIN MARKOV KERNEL

For a specific k we choose the Markov kernels

$$F_k(\boldsymbol{z}_k^i \mid \check{\boldsymbol{z}}_{k-1}^i) = \mathcal{N}(\boldsymbol{z}_k^i \mid \check{\boldsymbol{z}}_{k-1}^i + \delta_k \nabla \log \gamma_k(\check{\boldsymbol{z}}_{k-1}^i), 2\delta_k \boldsymbol{I}),$$
$$B_k(\check{\boldsymbol{z}}_{k-1}^i \mid \boldsymbol{z}_k^i) = F_k(\check{\boldsymbol{z}}_{k-1}^i \mid \boldsymbol{z}_k^i),$$

and compute the incremental weight as

$$\tilde{w}_k = \frac{\gamma_k(\boldsymbol{z}_k^i) B_k(\check{\boldsymbol{z}}_{k-1}^i \mid \boldsymbol{z}_k^i)}{\gamma_{k-1}(\check{\boldsymbol{z}}_{k-1}^i) F_k(\boldsymbol{z}_k^i \mid \check{\boldsymbol{z}}_{k-1}^i)},$$

where $\gamma_k(\boldsymbol{z}_k^i) = (1 - \beta_k) \log q(\boldsymbol{z}_k^i) + \beta_k \log p(\boldsymbol{z}_k^i)$.

B.3 HAMILTONIAN MARKOV KERNEL

For the Hamiltonian kernel, let $\boldsymbol{v}$ denote the momentum variable, let $\boldsymbol{M}$ denote the mass matrix, let ρ denote the damping coefficient, and let $L(\boldsymbol{z}_k^i, \boldsymbol{v}_k^i)$ denote the leap frog integrator for variables $\boldsymbol{z}_k^i, \boldsymbol{v}_k^i$ utilizing a step size of δ_k. We have

$$\tilde{\boldsymbol{v}}_k^i \sim \mathcal{N}(\rho \check{\boldsymbol{v}}_{k-1}^i, (1 - \rho^2)\boldsymbol{M}),$$
$$\text{and } \boldsymbol{z}_k^i, \boldsymbol{v}_k^i = L(\check{\boldsymbol{z}}_{k-1}^i, \tilde{\boldsymbol{v}}_k^i).$$

We compute the incremental weight as

$$\tilde{w}_k = \frac{\gamma_k(\boldsymbol{z}_k, \boldsymbol{v}_k) \mathcal{N}(\check{\boldsymbol{v}}_{k-1}^i \mid \rho \tilde{\boldsymbol{v}}_k^i, (1 - \rho^2)\boldsymbol{M})}{\gamma_{k-1}(\check{\boldsymbol{z}}_{k-1}^i, \check{\boldsymbol{v}}_{k-1}^i) \mathcal{N}(\tilde{\boldsymbol{v}}_k^i \mid \rho \check{\boldsymbol{v}}_k^i, (1 - \rho^2)\boldsymbol{M})},$$

where $\gamma_k(\boldsymbol{z}_k^i, \boldsymbol{v}_k^i) = (1 - \beta_k) \log q(\boldsymbol{z}_k^i) + \beta_k \log p(\boldsymbol{z}_k^i) + \log \mathcal{N}(\boldsymbol{v}_k^i, \boldsymbol{0}, \boldsymbol{M})$.

C QUANTITATIVE RESULTS

Appendix C.1 discusses Figure 1 of the main text and provides quantitative results. Appendix C.2 discusses Figure 2 of the main text and provides quantitative results for all experiments.

C.1 EFFECTIVE SAMPLE SIZE

Table 1 shows the quantitative results of Figure 1 over three resampling schemes. We visualize the ESS for epochs 100, 300, and 500 with different opacities. The ESS is an average of 10 iterations and the batch size of 64. We compare a DSMCS without resampling (No Resampling) to a DSMCS utilizing GST resampling gradients with a temperature of $\tau = 0.1$ (GST R.), and to a DSMCS that employs resampling but does not make use of any resampling gradients (Cat R.). Additionally, the plot shows the maximally achievable ESS of 64.

Table 1: Quantitative results of ESS for No Resampling, GST R., and Cat R. at the final epoch.

k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
No Resampling	64.0	2.09	1.95	1.86	1.77	1.7	1.68	1.65	1.62	1.61	1.59	1.57	1.53	1.49	1.48	1.48	1.51
GST R.	64.0	9.43	63.9	63.91	63.91	6.81	63.35	63.32	63.39	63.38	63.23	63.47	63.46	63.49	63.42	63.46	63.4
Cat R.	64.0	61.52	63.79	63.92	63.94	63.94	63.94	63.9	4.43	63.37	63.51	63.53	63.39	63.5	36.87	63.06	63.0

k	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
No Resampling	1.52	1.48	1.47	1.48	1.49	1.49	1.49	1.53	1.54	1.51	1.49	1.44	1.4	1.37	1.39	1.36
GST R.	60.62	8.18	62.44	61.98	62.44	62.46	62.42	62.47	62.33	6.1	59.37	59.94	59.55	60.04	60.08	3.79
Cat R.	62.86	62.18	58.62	39.12	56.15	62.57	62.52	62.77	62.83	62.65	5.22	59.61	60.39	60.3	60.15	3.08

We find that the ESS of DSMCS without resampling—being very similar to the DAIS model—drops after just one iteration to value around 1.5. For the GST R. model (utilizing straight-through gradients corresponding to the resampling operation) we find that the ESS stays mostly at around values of 63, dropping for larger $k \geq 27$ to values around 60. The Cat R. model (trained without any gradients corresponding to resampling) shows similar behavior. For the models utilizing resampling (with and without resampling gradients) we also notice a few outliers. For earlier epochs (e.g., 100) we see that the Cat model has a larger variance in its ESS (disregarding outliers).

C.2 STATIC TARGETS

Figure 2 shows a summary of the quantitative results presented in Table 2 and Table 3 below. Table 2 shows results on the static target experiment of the Langevin Markov kernel and Table 3 shows results of the Hamiltonian Markov kernel for various N, K, resampling schemes, and step sizes.

Figure 2 shows in (a) the differences between ELBO means "No" and "Cat" for the Langevin kernel and the Hamiltonian kernel at a step size $\hat{\delta} = 0.25$ as proposed by Doucet et al. (2022). We find large improvements for the Langevin kernel. The Hamiltonian kernel does not benefit from resampling in this case.

Figure 2 (b) shows the difference "No" and "Bern-Cat" where the later one employs the Bernoulli relaxation for resampling. Here, we see for both kernels an improvement of employing resampling. The discrepancy between (a) and (b) might be explained by the fact that the models in (b) only resample once the ESS drops below $N/2$: As a result the models might not suffer from information loss due to many resampling steps.

In Figure 2 (c) we evaluate the difference between models utilizing gradients corresponding to resampling (GST) and models without including gradients corresponding to resampling (Cat). We can verify the theoretical contribution of this paper: Given that both models admit an almost maximal ESS, there should be no difference between the models that include the resampling gradients and the ones that ignore the resampling gradients. However, we note that this trend is not fully visible for the Hamiltonian kernel.

Figure 2 (d) shows similar results but for models with the Bernoulli relaxation for resampling. We find that including the gradient corresponding to resampling (Bern-GST) does not have significant benefits compared to not including the gradient (Bern-Cat). This finding is again mostly visible for the Langevin kernel.

Table 2: Results of Langevin kernel (means and standard deviations over three runs). For details see main text. N/A denotes experiments that did not converge. Appendix B describes the experimental setup. For a description of the models please consult Appendix B.1.

$\hat{\delta}$		No				Cat				Bern-Cat		
	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
0.1	8	-184.14±4.32	-153.37±2.53	-143.37±2.84	8	-172.21±2.36	-141.7±3.15	-129.57±1.11	8	-173.67±5.13	-140.68±4.54	-127.16±2.52
	16	-136.93±2.65	-114.84±3.28	-105.82±1.02	16	-124.21±3.67	-96.42±2.48	-89.19±2.32	16	-122.16±3.53	-96.9±1.52	-85.31±0.41
	32	-90.48±0.85	-74.15±1.93	-68.51±1.42	32	-75.55±2.2	-63.31±1.94	-59.62±3.0	32	-76.06±2.35	-62.33±0.71	-59.9±1.65
0.25	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-118.29±2.95	-97.25±1.06	-90.67±1.63	8	-103.96±1.35	-80.89±2.19	-72.66±0.76	8	-103.77±3.36	-79.73±1.42	-71.18±1.72
	16	-74.2±1.32	-60.55±1.76	-55.48±0.29	16	-59.36±2.14	-44.11±1.82	-39.02±1.08	16	-58.37±1.53	-42.75±0.98	-38.45±1.54
	32	-42.25±0.39	-33.49±0.82	-30.28±0.54	32	-29.78±0.98	-26.9±0.53	-25.18±0.81	32	-31.22±0.79	-26.59±0.07	-23.16±0.95
1.0	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-33.55±0.91	-26.45±0.33	-23.68±0.22	8	-23.88±0.61	-15.53±0.49	-12.74±0.36	8	-24.32±0.7	-15.79±0.53	-13.31±0.35
	16	-19.06±0.39	-14.24±0.38	-12.59±0.26	16	-11.22±0.39	-7.4±0.88	-8.34±1.65	16	-11.83±0.4	-8.38±0.33	-9.18±0.35
	32	-9.79±0.29	-6.91±0.41	-5.85±0.35	32	-6.87±0.07	-16.59±0.7	-22.96±7.68	32	-7.62±0.48	-4.11±0.55	N/A

$\hat{\delta}$		GST ($\tau = 1.0$)			Bern-GST ($\tau = 1.0$)				GST ($\tau = 0.1$)				Bern-GST ($\tau = 0.1$)			
	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
0.1	8	-190.2±6.68	-151.38±1.99	-148.92±10.88	8	-183.73±5.01	-145.47±2.18	-136.43±13.44	8	-172.3±2.42	-141.67±3.04	-130.08±1.12	8	-173.84±4.92	-139.26±2.95	-126.92±2.85
	16	-147.98±10.44	-116.82±0.74	-122.78±10.71	16	-141.35±4.55	-108.57±5.38	-90.19±10.53	16	-124.31±3.65	-96.6±2.72	-88.81±3.07	16	-122.03±3.72	-96.98±1.53	-85.66±1.53
	32	-103.87±9.24	-69.95±8.46	-50.74±6.98	32	-76.51±2.85	-50.43±0.3	-38.86±0.3	32	-75.64±2.27	-62.72±1.97	-62.69±0.76	32	-74.28±2.78	-61.28±0.86	-60.49±0.9
0.25	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-105.61±4.14	-79.78±2.3	-72.53±0.79	8	-109.65±6.27	-79.06±1.84	-69.29±1.43	8	-104.0±1.65	-80.54±1.34	-73.17±0.58	8	-103.84±3.32	-79.59±1.25	-70.46±2.09
	16	-66.4±11.06	-43.13±2.6	-39.35±5.41	16	-63.07±6.52	-44.66±4.22	-33.82±0.22	16	-59.24±2.24	-44.34±1.18	-40.03±0.92	16	-58.45±2.07	-43.56±1.0	-38.21±0.37
	32	-32.37±1.44	-17.98±0.87	-14.18±0.46	32	-31.09±1.13	-15.53±0.6	-11.13±0.4	32	-29.84±0.89	-26.69±0.72	-21.09±4.08	32	-32.04±0.82	-26.92±0.27	-18.93±1.36
1.0	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-31.26±6.48	-19.78±0.42	-15.41±0.73	8	-25.75±3.15	-19.27±2.79	-12.17±0.39	8	-23.89±0.41	-15.69±0.73	-12.76±0.74	8	-24.04±0.84	-15.77±0.15	-13.49±0.48
	16	-24.82±3.89	-8.17±0.63	-6.04±0.14	16	-48.11±24.02	-7.46±0.25	-6.32±0.3	16	-11.61±0.97	-7.62±1.02	-6.18±0.28	16	-11.42±0.6	-8.2±1.34	-6.46±0.74
	32	-22.55±15.4	-6.88±2.11	-5.97±3.66	32	-16.05±5.7	-4.21±0.13	N/A	32	-6.66±0.17	-16.39±0.28	-18.75±0.34	32	-7.57±0.16	-4.05±0.22	-6.3±2.16

Table 3: Results of Hamiltonian kernel (means and standard deviations over three runs). For details see main text. N/A denotes experiments that did not converge. Appendix B describes the experimental setup. For a description of the models please consult Appendix B.1.

$\hat{\delta}$		No				Cat				Bern-Cat		
	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
0.1	8	-27.05±1.69	-18.2±2.22	-17.08±1.55	8	-24.15±1.3	-16.59±0.75	-14.88±0.7	8	-20.14±0.46	-14.56±0.37	-10.57±0.45
	16	-16.44±0.92	-8.63±0.37	-6.78±0.67	16	-14.67±0.58	-11.44±0.64	-11.35±2.93	16	-11.06±0.89	-6.21±0.28	-10.83±4.83
	32	-9.81±0.87	-4.55±0.75	-20.1±9.22	32	-7.55±1.07	-6.04±1.52	-3.96±0.36	32	-4.36±0.84	-4.63±0.1	-1.98±0.2
0.25	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-26.65±0.31	-18.62±0.65	-15.14±0.56	8	-26.58±1.72	-18.94±0.31	-15.33±0.44	8	-20.58±0.37	-13.16±0.71	-10.98±0.48
	16	-14.42±0.27	-7.58±0.31	-7.25±1.26	16	-13.49±0.24	-9.35±1.22	-8.95±0.49	16	-9.48±0.56	-6.8±0.29	-5.26±1.03
	32	-10.33±0.39	-6.51±1.44	-2.18±0.25	32	-10.42±1.39	-23.53±22.83	-10.4±3.26	32	-5.36±0.25	-4.69±0.44	1.0±0.14
1.0	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-25.38±1.39	-19.41±0.49	-16.68±0.3	8	-25.91±0.87	-28.69±1.41	-18.42±1.33	8	-23.83±0.76	-25.6±2.12	-18.79±2.09
	16	-16.79±2.6	-9.02±0.83	-7.17±0.75	16	-18.23±2.54	-13.76±1.2	-15.38±3.86	16	-14.04±1.17	-8.1±1.84	-11.64±5.23
	32	-91.03±114.19	-3.31±0.38	-4.95±3.16	32	-12.17±1.15	-12.65±0.79	-13.46±1.79	32	N/A	-5.82±0.48	-4.64±0.4

$\hat{\delta}$		GST ($\tau = 1.0$)			Bern-GST ($\tau = 1.0$)				GST ($\tau = 0.1$)				Bern-GST ($\tau = 0.1$)			
	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
0.1	8	-228.25±0.69	-138.01±17.07	-125.01±10.53	8	-116.69±74.94	-108.62±7.04	-85.47±7.21	8	-25.34±2.16	-16.58±1.51	-15.34±0.15	8	-22.94±1.77	-13.64±0.12	-12.02±0.38
	16	N/A	N/A	N/A	16	N/A	-199.13±10.01	-181.45±7.17	16	-15.88±0.29	-11.01±1.15	-11.82±5.66	16	-18.27±6.09	-7.77±1.93	-8.54±0.5
	32	N/A	N/A	-176.48±9.59	32	N/A	-26.49±19.09	-4.62±3.68	32	-9.3±1.76	-7.49±1.1	-5.69±0.27	32	-6.21±3.1	-2.74±0.14	-3.48±0.45
0.25	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-234.5±12.2	-122.28±36.56	-50.93±12.21	8	-58.96±20.3	-103.45±44.46	-38.46±7.0	8	-24.16±1.34	-17.58±0.03	-14.25±0.04	8	-23.02±2.82	-14.22±0.7	-10.03±0.5
	16	-156.89±34.08	-150.69±42.62	-104.45±16.28	16	-99.05±59.42	-101.3±6.31	-128.49±48.3	16	-13.35±0.19	-9.57±1.15	-14.59±5.35	16	-9.89±0.72	-24.69±4.25	-4.71±0.73
	32	-59.33±28.63	-36.89±17.48	-32.68±14.01	32	N/A	-4.93±4.57	0.02±0.96	32	-9.01±0.53	-10.39±1.85	-11.76±1.24	32	-4.0±0.36	N/A	-0.39±0.55
1.0	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64	K \ N	8	32	64
	8	-35.55±2.63	-38.06±8.11	-44.1±15.84	8	-30.82±3.26	-31.95±5.74	-49.46±9.09	8	-25.95±1.72	-22.38±1.98	-20.66±2.85	8	-24.79±2.27	-17.22±0.31	-15.86±0.55
	16	-25.94±6.86	-29.06±7.48	-45.72±7.83	16	-13.74±1.14	N/A	-47.35±20.14	16	-15.63±0.62	-14.62±1.86	-37.45±12.66	16	N/A	-8.29±2.11	N/A
	32	-11.34±2.02	-244.71±296.56	-17.58±1.35	32	N/A	N/A	-5.16±0.12	32	-11.81±2.21	-16.4±5.07	-19.54±4.19	32	N/A	-8.88±2.08	N/A