A unified view of likelihood ratio and reparameterization gradients: Supplementary Materials

A Additional background literature on LR and RP

See the work by Mohamed et al. (2019) for a recent extensive general review of Monte Carlo gradient estimators, such as LR and RP. Here we discuss the main points about LR and RP in the literature, and how these are related to our work.

Advantages and disadvantages of LR and RP: The variance of LR and RP gradients has been of central importance in their research. Typically, RP is said to be more accurate and scale better with the sampling dimension (Rezende et al., 2014)—this claim is also backed by theory (Xu et al., 2019; Nesterov and Spokoiny, 2017); however, **there is no guarantee that RP outperforms LR**. In particular, for multimodal $\phi(x)$ (Gal, 2016) or chaotic systems (Parmas et al., 2018), LR can be arbitrarily better than RP (e.g., the latter showed that LR can be 10^6 more accurate in practice). Moreover, RP is not directly applicable to discrete sampling spaces, but requires continuous relaxations (Maddison et al., 2016; Jang et al., 2016; Tucker et al., 2017). Differentiable RP is also not always possible, but implicit RP gradients have increased the number of usable distributions (Figurnov et al., 2018). Because LR and RP both have advantages and disadvantages, optimal estimation techniques will require a combination of LR and RP as considered in the flow gradients.

Variance reduction techniques: Techniques for variance reduction have been extensively studied, including control variates/baselines (Grathwohl et al., 2017; Greensmith et al., 2004b; Tucker et al., 2018; Gu et al., 2015; Geffner and Domke, 2018; Gu et al., 2016) as well as Rao-Blackwellization (Titsias and Lázaro-Gredilla, 2015; Ciosek and Whiteson, 2018; Asadi et al., 2017). One can also combine the best of both LR and RP gradients by dynamically reweighting them (Parmas et al., 2018; Metz et al., 2019).

LR and RP on computational graphs: Several methods for computing LR and RP gradients on graphs of computations exist (Schulman et al., 2015a; Weber et al., 2019; Parmas, 2018; Foerster et al., 2018; Mao et al., 2019; Farquhar et al., 2019). Among these works, Schulman et al. (2015a) provided a simple way to obtain the gradient estimators using automatic differentiation of a surrogate objective on stochastic computation graphs; Weber et al. (2019) extended this work to be also applicable for gradient estimators using critics; Parmas (2018) provided an intuitive abstract framework for reasoning about gradient estimators by turning the stochastic graph deterministic through considering gradients of the marginal distributions w.r.t. the distributions at the other nodes, thus allowing to apply the total derivative rule; Foerster et al. (2018) extended the surrogate loss concept for higher order derivatives; Mao et al. (2019) provided a baseline for higher order LR estimators, and Farquhar et al. (2019) derived a way to trade off bias and variance.

Importance sampling: Importance sampling for reducing LR gradient variance was previously considered in variational inference (Ruiz et al., 2016a), who proposed to sample from the same distribution while tuning the variance. In reinforcement learning, importance sampling has been studied for sample reuse via off-policy policy evaluation (Thomas and Brunskill, 2016; Jiang and Li, 2016; Gu et al., 2017; Munos et al., 2016; Jie and Abbeel, 2010), but modifying the policy to improve gradient accuracy has not been considered.

Other: The flow theory in Sec. 4 was concurrently derived by (Jankowiak and Obermeyer, 2018), but their work focused on deriving new RP gradient estimators, and they do not discuss the duality. Our derivation is also more visual. We discuss the relationship between their work and ours in more detail in App. E.2. The flow gradient estimator in Eq. (19) was also presented in a work in progress by Wu (2019) slightly earlier, but in their derivation they do not use the divergence theorem, and assume that $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ belongs to a reparameterizable variable, whereas we do not make such an assumption, and argue that the flow is a more fundamental concept than the reparameterization of the variable. An advantage of their work is that they show a new polynomial-based gradient estimator, whereas we focus on characterizing the space of estimators. One point is that there always exists a flow corresponding to a reparameterization; however, it is an open question whether there exists a reparameterization corresponding to each flow $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$. Our main contribution regarding this estimator is the discussion around it, which claims that the flow $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ is a fundamental concept that characterizes the space of all possible single sample estimators, as well as the visualizations and physical intuitions that we provide.

B Likelihood ratio gradient basics

The likelihood ratio (LR) gradient estimator is given by

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{E}_{x\sim p(x;\theta)}\left[\phi(x)\right] = \mathbb{E}_{x\sim p(x;\theta)}\left[\frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\theta}\phi(x)\right]. \tag{20}$$

For a Gaussian $p\left(x;\theta\right)$:

$$\begin{aligned} \log p\left(x;\theta\right) &= -\frac{1}{2}\log(2\pi) - \log(\sigma) - \frac{(x-\mu)^2}{2\sigma^2}, \\ \frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\mu} &= \frac{x-\mu}{\sigma^2} = \frac{\epsilon}{\sigma}, \\ \frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\sigma} &= \frac{(x-\mu)^2}{\sigma^3} - \frac{1}{\sigma} = \frac{\epsilon^2}{\sigma} - \frac{1}{\sigma}, \\ \text{where} \quad x &= \mu + \epsilon\sigma \text{ and } \epsilon \sim \mathcal{N}(0,1). \end{aligned} \tag{21}$$

Baselines to reduce gradient variance: The LR gradient estimator on its own has a large variance, and techniques have to be used to stabilize it. A common technique is to subtract a constant baseline b from the $\phi(x)$ values, so that the gradient estimator becomes

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{E}_{x\sim p(x;\theta)}\left[\frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\theta}\left(\phi(x) - b\right)\right]. \tag{22}$$

In practice, using $b = \mathbb{E}_{x\sim p(x;\theta)}\left[\phi(x)\right]$ works well, but one can also derive an optimal baseline (Weaver and Tao, 2001). We outline the derivation below. The gradient variance when a baseline is used can be expressed as

$$\begin{aligned} \mathbb{V}_{x\sim p(x;\theta)}\left[\frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\theta}\left(\phi(x) - b\right)\right] = \; & \mathbb{E}_{x\sim p(x;\theta)}\left[\left(\frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\theta}\phi(x)\right)^2\right] \\ & - 2\mathbb{E}_{x\sim p(x;\theta)}\left[\left(\frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\theta}\right)^2\phi(x)b\right] \\ & + \mathbb{E}_{x\sim p(x;\theta)}\left[\left(\frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\theta}b\right)^2\right]. \end{aligned} \tag{23}$$

Taking the derivative of Eq. (23) w.r.t. b and setting to zero gives the optimal baseline as

$$b_{opt} = \frac{\mathbb{E}_{x\sim p(x;\theta)}\left[\left(\frac{\mathrm{d}\log p(x;\theta)}{\mathrm{d}\theta}\right)^2\phi(x)\right]}{\mathbb{E}_{x\sim p(x;\theta)}\left[\left(\frac{\mathrm{d}\log p(x;\theta)}{\mathrm{d}\theta}\right)^2\right]}. \tag{24}$$

In practice, for example if $\phi(x)$ is linear and $p\left(x;\theta\right)$ is Gaussian then $b_{opt} = \mathbb{E}_{x\sim p(x;\theta)}\left[\phi(x)\right]$, so the gain from trying to use an optimal baseline is often small.

Antithetic sampling: An often used technique is to sample points x in pairs opposite to each other, s.t. $x_+ = \mu + \sigma\epsilon$ and $x_- = \mu - \sigma\epsilon$. This technique is particularly often used in evolution strategies' research (Salimans et al., 2017; Mania et al., 2018). We will explain that when this technique is used, then a baseline has no effect because it cancels. The derivation is easy to see by considering that for a Gaussian: $\frac{\mathrm{d}\log p(x;\theta)}{\mathrm{d}\mu} = \frac{\epsilon}{\sigma}$, so

$$\begin{aligned} & \frac{\mathrm{d}\log p\left(x_+;\theta\right)}{\mathrm{d}\mu}\left(\phi(x_+) - b\right) + \frac{\mathrm{d}\log p\left(x_-;\theta\right)}{\mathrm{d}\mu}\left(\phi(x_-) - b\right) \\ & = \frac{\epsilon}{\sigma}\left(\phi(x_+) - b - \left(\phi(x_-) - b\right)\right) = \frac{\epsilon}{\sigma}\left(\phi(x_+) - \phi(x_-)\right). \end{aligned}$$

Relationship to finite difference methods: Finite difference methods also use the function values $\phi(x)$ to estimate a derivative, so it may appear that the LR gradient estimator is a finite difference estimator. Finite difference estimators work by estimating the slope of the function, by evaluating the change between two points, i.e.

$$\frac{\mathrm{d}\phi(x)}{\mathrm{d}x} \approx \frac{\phi(x_+) - \phi(x_-)}{\Delta x}. \tag{25}$$

In the antithetic sampling case, $\Delta x = 2\sigma\epsilon$, so the estimator is

$$\frac{\mathrm{d}\phi(x)}{\mathrm{d}x} \approx \frac{\phi(x_+) - \phi(x_-)}{2\sigma\epsilon}. \tag{26}$$

Clearly, this is different to the LR gradient estimator averaged over one pair of samples $\frac{1}{2}\left(\frac{\epsilon}{\sigma}\phi(x_+) + \frac{-\epsilon}{\sigma}\phi(x_-)\right)$:

$$\frac{\epsilon\left(\phi(x_+) - \phi(x_-)\right)}{2\sigma}, \tag{27}$$

because the ϵ is in the wrong place. In Sec. 3 we explain that the LR gradient estimator is a different concept to finite differences, which is not trying to fit a linear function onto $\phi(x)$.

C Probability boxes formal derivation

Here we give a more formal derivation of the "probability boxes" explanation in Sec. 3. In the theory, we explained that the expectation can be written as a weighted average $\sum_{i=1}^{N} P(x_i)\phi(x_i)$ of function values, where the weight is given by the probabilities. Then, the LR gradient estimated the derivative of this expectation w.r.t. θ by differentiating the $P(x_i)$ term, whereas the RP gradient worked by differentiating the $\phi(x_i)$ term. In this section, we will formally define the locations and edges of the "boxes" when a continuous integral is discretized, and show that in the LR case, indeed $\phi(x_i)$ in the "box" stays fixed, whereas in the RP case, indeed the probability mass $P(x_i)$ in the "box" stays fixed as the parameters θ are perturbed.

The explanation relies on a first principles thinking about the effect that changing the parameters of a probability distribution θ has on infinitesimal "boxes" of probability mass (Fig. 2). Both LR and RP are trying to estimate

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\int p\left(x;\theta\right)\phi(x)\mathrm{d}x. \tag{28}$$

A typical finite explanation of Riemann integrals is performed by discretizing the integrand into "boxes" of size Δx, and summing:

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\sum_{i=1}^{N} p\left(x_i;\theta\right)\Delta x_i \phi(x_i). \tag{29}$$

Taking the limit as $N \to \infty$ recovers the true integral. In this equation, $P(x_i) = p\left(x_i;\theta\right)\Delta x_i$ is the amount of probability mass inside the "box", and $\phi(x_i)$ is the function value inside the "box".

RP estimator: Such a view can be used to explain RP gradients. In this case, the boundaries of the "box" are fixed with reference to the shape of the probability distribution, i.e. for each i we define the center of the box as

$$x_i = g(\epsilon_i;\theta), \tag{30}$$

and the boundaries as

$$g(\epsilon_i \pm \Delta\epsilon/2;\theta), \tag{31}$$

where ϵ_i is the reference position on a fixed simple distribution, $p\left(\epsilon\right)$. The amount of probability mass assigned to each "box" stays fixed at

$$P_i = p\left(\epsilon\right)\Delta\epsilon\ ; \tag{32}$$

however, the center of the "box" moves, so the function value $\phi(x_i)$ inside each "box" changes by

$$\delta\phi_i = \phi\left(g(\epsilon_i;\theta+\delta\theta)\right) \ - \ \phi\left(g(\epsilon_i;\theta)\right) = \phi(x_i \ + \ \delta x_i) \ - \ \phi(x_i). \tag{33}$$

The full derivative can then be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{E}_{x\sim p(x;\theta)}\left[\phi(x)\right] \approx \frac{1}{\delta\theta}\sum_{i=1}^{N} P_i \delta\phi_i = \sum_{i=1}^{N} P_i \frac{\delta\phi_i}{\delta x_i}\frac{\delta x_i}{\delta\theta}. \tag{34}$$

Taking the infinitesimal limit $N \to \infty$, and noting $P_i = p\left(x_i;\theta\right)\Delta x_i$, we obtain the RP estimator

$$\int p\left(x;\theta\right)\frac{\mathrm{d}\phi(x)}{\mathrm{d}x}\frac{\mathrm{d}x}{\mathrm{d}\theta}\ \mathrm{d}x. \tag{35}$$

We see that RP essentially estimates the gradient by keeping the probability mass inside each "box" fixed, but estimating how the function value ϕ inside the "box" changes as the parameters θ are perturbed.

LR estimator: The LR gradient, on the other hand, keeps the boundaries of the "boxes" fixed, i.e. the center of the box is at x_i, and the boundaries at

$$x_i \pm \Delta x_i/2. \tag{36}$$

Now, as the boundaries are independent of θ, the function value $\phi(x_i)$ inside the box stays fixed, even as θ is perturbed by $\delta\theta$; however, the probability mass inside the box changes, because the density changes by

$$\delta p_i = p\left(x_i;\theta+\delta\theta\right) - p\left(x_i;\theta\right). \tag{37}$$

The full derivative can be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{E}_{x\sim p(x;\theta)}\left[\phi(x)\right] \approx \frac{1}{\delta\theta}\sum_{i=1}^{N}\Delta x_i \delta p_i \phi(x_i) = \sum_{i=1}^{N} p\left(x_i;\theta\right)\Delta x_i \frac{\delta p_i/\delta\theta}{p\left(x_i;\theta\right)}\phi(x_i). \tag{38}$$

Where we have multiplied and divided by $p\left(x_i;\theta\right)$. Taking the infinitesimal limit recovers the LR gradient

$$\int p\left(x;\theta\right)\frac{\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}}{p\left(x;\theta\right)}\phi(x)\ \mathrm{d}x = \mathbb{E}_{x\sim p(x;\theta)}\left[\frac{\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}}{p\left(x;\theta\right)}\phi(x)\right]. \tag{39}$$

The transformation

$$p\left(x;\theta\right)\frac{\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}}{p\left(x;\theta\right)} = p\left(x;\theta\right)\frac{\mathrm{d}\log p\left(x;\theta\right)}{\mathrm{d}\theta} \tag{40}$$

is known as the log-derivative trick, and it may appear to be the essence behind the LR gradient, but actually the multiplication and division by $p(x;\theta)$ is just a special case of the more general Monte Carlo integration principle. Any integral $\int f(x)\ \mathrm{d}x$ can be approximated by sampling from a distribution $q(x)$ as

$$\int f(x)\ \mathrm{d}x = \int q(x)\frac{f(x)}{q(x)}\ \mathrm{d}x = \mathbb{E}_{x\sim q(x)}\left[\frac{f(x)}{q(x)}\right]. \tag{41}$$

Rather than thinking of the LR gradient in terms of the log-derivative term, it may be better to think of it as simply estimating the integral $\int \frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}\phi(x)\ \mathrm{d}x$ by applying the appropriate importance weights to samples from $p(x;\theta)$. Thus, we see that in the discretized case, the LR gradient picks $q(x) = p\left(x;\theta\right)$ (Jie and Abbeel, 2010) and performs Monte Carlo integration to approximate $\frac{1}{\delta\theta}\sum_{i=1}^{N}\Delta x_i \delta p_i \phi(x_i)$ by sampling from $P(x_i) = \Delta x_i p\left(x_i;\theta\right)$. To summarize: LR estimates the gradient by keeping the boundaries of the boxes fixed, measuring the change in probability mass in each box, and weighting by the function value: $\phi(x_i)\delta p$.

Sometimes, the LR gradient is described as being "kind of like a finite difference gradient" (Salimans et al., 2017; Mania et al., 2018), but here we see that it is a different concept, which does not rely on fitting a straight line between differences of ϕ (App. B), but estimates how probability mass is reallocated among different ϕ values via Monte Carlo integration by sampling from $p\left(x;\theta\right)$.

D Derivations for the probability flow theory

(The notation and divergence theorem proof are the same as in the main paper, and provided here for reference.) We illustrate the background information in 3 dimensions, but it generalizes straightforwardly to higher dimensions.

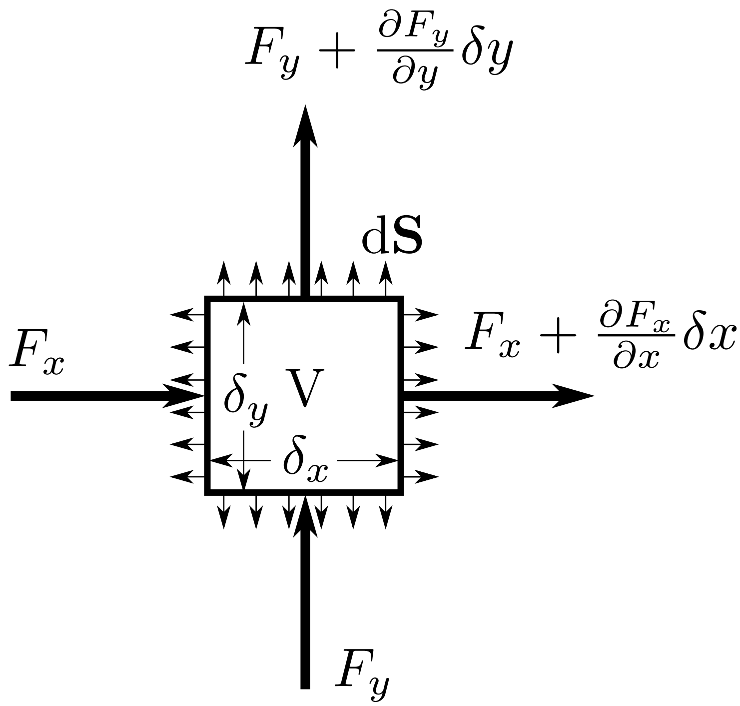


Figure 5: Illustration of the divergence theorem.

Notation:
$\boldsymbol{F} = [F_x(x,y,z), F_y(x,y,z), F_z(x,y,z)]$ is a vector field.
$\phi(x,y,z)$ is a scalar field (a scalar function).
Div operator: $\nabla \cdot \boldsymbol{F} = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z}$.
Grad operator: $\nabla \phi = [\frac{\partial \phi}{\partial x}, \frac{\partial \phi}{\partial y}, \frac{\partial \phi}{\partial z}]$.

D.1 Basic vector calculus and fluid mechanics

The vector field $\boldsymbol{F}$ could be for example thought of as a local flow velocity for some fluid. If $\boldsymbol{F}$ is the density flow rate, then the div operator essentially measures how much the density is decreasing at a point. If the outflow is larger than the inflow, the density would decrease and vice versa. The divergence theorem, illustrated in Fig. 5 illustrates how this change in density can be measured in two separate ways: one could integrate the divergence across the volume, or one could integrate the inflow and outflow across the surface. The divergence theorem states:

$$\int_V \nabla \cdot \boldsymbol{F} \mathrm{d}V = \int_S \boldsymbol{F} \cdot \mathrm{d}\boldsymbol{S}. \tag{42}$$

To prove the claim, consider the infinitesimal box in Fig. 5. The divergence can be calculated as $\delta x \delta y (\frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y})$. On the other hand, to take the integral across the surface, note that the surface normals point outwards, and the integral becomes $\delta_y(-F_x + F_x - \frac{\partial F_x}{\partial x}\delta x) + \delta x(-F_y + F_y + \frac{\partial F_y}{\partial y}\delta y) = \delta x \delta y (\frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y})$, which is the same as the divergence. To generalize this to arbitrarily large volumes, notice that if one stacks the boxes next to each other, then the surface integral across the area where the boxes meet cancels out, and only the integral across the outer surface remains. For an incompressible flow, the density does not change, and the divergence must be zero.

D.2 Derivation of probability surface integral

We will show that the LR estimator tries to integrate $\int_S \phi(\tilde{\boldsymbol{x}}) \nabla_\theta \tilde{g}(\epsilon_x, \epsilon_h) \mathrm{d}\boldsymbol{S}$. First, note that $\mathrm{d}\boldsymbol{S} = \hat{\boldsymbol{n}} \mathrm{d}S$, and it is necessary to express the normalized surface vector $\hat{\boldsymbol{n}}$. To do so, we first express the tangent vectors $\boldsymbol{t}$, then find the vector perpendicular to all of them (this is exactly the normal vector).

All tangent vectors are characterized by the equation $\boldsymbol{t} = \left[\boldsymbol{r}, \frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}} \cdot \boldsymbol{r}\right]$, where $\boldsymbol{r}$ is an arbitrary vector. The normal vector $\boldsymbol{n}$ is such that $\boldsymbol{t} \cdot \boldsymbol{n} = 0$ for all tangent vectors. Therefore, the normal vector $\boldsymbol{n} = \left[-\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}, 1\right]$, satisfies the equation, because $\boldsymbol{n} \cdot \boldsymbol{t} = -\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}} \cdot \boldsymbol{r} + 1 \cdot \frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}} \cdot \boldsymbol{r} = 0$. Finally, we normalize the vector:

$$\hat{\boldsymbol{n}} = {\left[-\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}},1\right]}\Big/{\sqrt{\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)^T+1}}\,. \tag{43}$$

Next, we perform a change of coordinates from the surface elements $\mathrm{d}S$ to cartesian coordinates $\mathrm{d}\boldsymbol{x}$. When projecting a surface element $\mathrm{d}S$ with unit normal $\hat{\boldsymbol{n}}$ to a plane with unit normal $\hat{\boldsymbol{m}}$, the projected area is given by $\mathrm{d}\boldsymbol{x} = |\hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{m}}|\, \mathrm{d}S$, therefore, as $\hat{\boldsymbol{m}} = [\boldsymbol{0}, 1]$ for the $\boldsymbol{x}$-plane, we have $\mathrm{d}\boldsymbol{x} = \mathrm{d}S \left| \frac{1}{\sqrt{\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)^T+1}} \left[-\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}, 1\right] \cdot [\boldsymbol{0}, 1] \right| = {\mathrm{d}S}\Big/{\sqrt{\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)^T+1}}$, which leads to

$$\mathrm{d}S = \sqrt{\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)^T + 1}\ \mathrm{d}\boldsymbol{x}. \tag{44}$$

Plugging Eqs. (43) and (44) into the right-hand side of Eq. (17) we get

$$\int_X \phi(\tilde{\boldsymbol{x}}) \nabla_\theta \tilde{g}(\epsilon_x, \epsilon_h) \cdot \frac{\left[-\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}, 1\right]}{\sqrt{\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)^T + 1}} \sqrt{\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)\left(\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}\right)^T + 1}\ \mathrm{d}\boldsymbol{x} = \int_X \phi(\tilde{\boldsymbol{x}}) \nabla_\theta \tilde{g}(\epsilon_x, \epsilon_h) \cdot \left[-\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}, 1\right]\ \mathrm{d}\boldsymbol{x}. \tag{45}$$

Recall that the last element of $\tilde{g}(\epsilon_x, \epsilon_h)$ is $\epsilon_h p\,(g(\epsilon_x); \theta)$, and that at the boundary surface $\epsilon_h = 1$, then the scalar product term $\nabla_\theta \tilde{g}(\epsilon_x, \epsilon_h) \cdot \left[-\frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}, 1\right]$ turns into $-\nabla_\theta g(\epsilon_x) \cdot \frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}} + \frac{\partial \epsilon_h p(g(\epsilon_x);\theta)}{\partial \theta}\Big|_{\epsilon_x = const, \epsilon_h = 1}$. The last term $\frac{\partial p(g(\epsilon_x);\theta)}{\partial \theta}\Big|_{\epsilon_x = const}$ can be thought of as the rate of change of the probability density while following a point moving in the flow induced by perturbing θ. This quantity can be expressed with the material derivative $\frac{\partial p(g(\epsilon_x);\theta)}{\partial \theta}\Big|_{\epsilon_x = const} = \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta} + \nabla_\theta g(\epsilon_x) \cdot \frac{\mathrm{d}p}{\mathrm{d}\boldsymbol{x}}$. Finally, substituting into Eq. (45):

$$\int_S \phi(\tilde{\boldsymbol{x}}) \nabla_\theta \tilde{g}(\epsilon_x, \epsilon_h)\ \mathrm{d}\boldsymbol{S} = \int_X \phi(\boldsymbol{x}) \frac{\mathrm{d}p\,(\boldsymbol{x};\theta)}{\mathrm{d}\theta}\ \mathrm{d}\boldsymbol{x}. \tag{46}$$

E Characterizing the space of all likelihood ratio and reparameterization gradients

E.1 Reparameterization gradients are not unique

What happens if we perform the same kind of analysis as in Theorem 2 for the RP gradient? To examine this idea, first we extend the argumentation in the main paper to include non-invertible $g(\epsilon; \theta)$, i.e., consider the case when multiple different ϵ lead to the same x via $x = g(\epsilon; \theta)$. In Sec. 2.2 we argued that if g is invertible, we can write the RP gradient as $\frac{\mathrm{d}\phi(x)}{\mathrm{d}x} \frac{\partial g(\epsilon;\theta)}{\partial \theta}\Big|_{\epsilon = S(x;\theta)}$, where $\frac{\partial g(\epsilon;\theta)}{\partial \theta_i}\Big|_{\epsilon = S(x;\theta)}$ corresponds to $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$, which is a vector field. In the non-invertible case, similarly denote $S(x; \theta)$ is the set of ϵ, s.t. $x = g(\epsilon; \theta)$. For each x, we can employ Bayes' rule to derive the posterior distribution of the ϵ that generated x, and integrate across this distribution to obtain the Rao-Blackwellized estimator $\frac{\mathrm{d}\phi(x)}{\mathrm{d}x} \int_{S(x;\theta)} p(\epsilon) \frac{\partial g(\epsilon;\theta)}{\partial \theta}\Big|_{\epsilon \in S(x;\theta)} \mathrm{d}\epsilon \Big/ \int_{S(x;\theta)} p(\epsilon) \mathrm{d}\epsilon$, where $\int_{S(x;\theta)} p(\epsilon) \frac{\partial g(\epsilon;\theta)}{\partial \theta_i}\Big|_{\epsilon \in S(x;\theta)} \mathrm{d}\epsilon \Big/ \int_{S(x;\theta)} p(\epsilon) \mathrm{d}\epsilon$ corresponds to $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$, which is a vector field. Thus, even in the non-invertible case, the reparameterization gradient can be expressed as a dot product between $\nabla_{\boldsymbol{x}} \phi(\boldsymbol{x})$ and a vector field $\boldsymbol{u}(\boldsymbol{x})$.

Now we show that the type of analysis in Thm. 2 does not lead to a uniqueness claim for RP gradients. Similarly, suppose that there exist $\boldsymbol{u}(\boldsymbol{x})$ and $\boldsymbol{v}(\boldsymbol{x})$, s.t. $\int \nabla \phi(\boldsymbol{x}) \cdot \boldsymbol{u}(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} = \int \nabla \phi(\boldsymbol{x}) \cdot \boldsymbol{v}(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x}$ for any $\phi(\boldsymbol{x})$. Rearrange the equation into $\int \nabla\phi(\boldsymbol{x}) \cdot (\boldsymbol{u}(\boldsymbol{x}) - \boldsymbol{v}(\boldsymbol{x}))\ \mathrm{d}\boldsymbol{x} = 0$. Then, if we can pick $\nabla\phi(\boldsymbol{x}) = \boldsymbol{u}(\boldsymbol{x}) - \boldsymbol{v}(\boldsymbol{x})$ it would lead to $\boldsymbol{u} = \boldsymbol{v}$, which would show the uniqueness. In the one-dimensional case, this is possible, and in this case all pure reparameterization or pathwise gradients are equivalent. However, in higher dimensions, it is not necessarily possible to pick such $\phi(\boldsymbol{x})$. In particular, the integral of $\nabla\phi(\boldsymbol{x})$ over any closed path is 0, but this is not necessarily the case for $\boldsymbol{u} - \boldsymbol{v}$. Therefore, the same kind of analysis does not lead to a claim of uniqueness. Indeed, concurrent work (Jankowiak and Obermeyer, 2018) showed that there are an infinite amount of possible reparameterization gradients, and the minimum variance[8] is achieved by the optimal transport flow.

E.2 Flow gradient estimator

In this section, we provide the derivations and proofs for the flow gradient estimator in Eq. (19). While our derivation with a height reparameterization in Sec. 4 shows the duality between LR and RP, and allows for intuitive visualizations, the method of derivations by Jankowiak and Obermeyer (2018) is algebraically easier to work with, so we adopt their style in the following sections. Recall the divergence theorem $\int_V \nabla_{\boldsymbol{x}} \cdot \boldsymbol{F} \mathrm{d}V = \int_S \boldsymbol{F} \cdot \mathrm{d}\boldsymbol{S}$, where $\boldsymbol{F}$ is an arbitrary continuous piecewise differentiable vector field. We can pick $\boldsymbol{F} = p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x})\phi(\boldsymbol{x})$, and a surface S at infinity, enclosing the whole volume, which leads to the equation

$$\int_V \nabla_{\boldsymbol{x}} \cdot \left(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x})\phi(\boldsymbol{x})\right) \mathrm{d}V = \int_S p(\boldsymbol{x};\theta)\, \phi(\boldsymbol{x})\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) \cdot \mathrm{d}\boldsymbol{S}. \tag{47}$$

Note that the boundary S is at $\|\boldsymbol{x}\| \to \infty$, and in this case $p(\boldsymbol{x};\theta) \to 0$, meaning that $\int_S p(\boldsymbol{x};\theta)\, \phi(\boldsymbol{x})\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) \cdot \mathrm{d}\boldsymbol{S} = 0$, as long as $\phi(\boldsymbol{x})\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ does not go to infinity faster than $p(\boldsymbol{x};\theta)$ goes to 0.[9] Hence,

$$\begin{aligned} &\int_V \nabla_{\boldsymbol{x}} \cdot \left(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x})\phi(\boldsymbol{x})\right) \mathrm{d}V = 0 \\ \Rightarrow &\int_V \Big(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x}) \cdot \nabla_{\boldsymbol{x}}\phi(\boldsymbol{x}) + \phi(\boldsymbol{x})\nabla_{\boldsymbol{x}} \cdot \big(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big)\Big) \mathrm{d}V = 0. \end{aligned} \tag{48}$$

As the integral in Eq. (48) is 0, we can add it to the integral of $\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}\phi(\boldsymbol{x})$ without changing the expectation:

$$\begin{aligned} &\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\left[\phi(\boldsymbol{x})\right] \\ &= \int_X \Big(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x}) \cdot \nabla_{\boldsymbol{x}}\phi(\boldsymbol{x}) + \phi(\boldsymbol{x})\nabla_{\boldsymbol{x}} \cdot \big(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big)\Big) + \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}\phi(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} \\ &= \int_X p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x}) \cdot \nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \int_X \left(\nabla_{\boldsymbol{x}} \cdot \big(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big) + \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}\right)\phi(\boldsymbol{x})\mathrm{d}\boldsymbol{x}. \end{aligned} \tag{49}$$

By importance sampling from $q(\boldsymbol{x})$, the estimator is characterized by the equation:

$$\begin{aligned} &\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\left[\phi(\boldsymbol{x})\right] \\ &= \mathbb{E}_{\boldsymbol{x}\sim q(\boldsymbol{x})}\left[\frac{p(\boldsymbol{x};\theta)}{q(\boldsymbol{x})}\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) \cdot \nabla_{\boldsymbol{x}}\phi(\boldsymbol{x}) + \frac{1}{q(\boldsymbol{x})}\left(\nabla_{\boldsymbol{x}} \cdot \big(p(\boldsymbol{x};\theta)\, \boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big) + \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}\right)\phi(\boldsymbol{x})\right]. \end{aligned} \tag{50}$$

Next, we show that the estimator in Eq. (50) characterizes the space of all possible single sample unbiased gradient estimators that combine the function value $\phi(\boldsymbol{x})$ and function derivative $\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})$ information, and have the product form given in Eq. (4). The proof is analogous to the proof of uniqueness of the LR gradient estimator in Theorem 2. Remember that $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ is a completely arbitrary vector field (and hence covers all possible ways to

[8]By minimum variance, we mean the minimum variance achievable without assuming knowledge of $\phi(\boldsymbol{x})$, or alternatively that it is approximately linear in the sampling range, $\nabla\phi(\boldsymbol{x}) \approx \boldsymbol{A}$. Their result holds for arbitrary dimensionality.

[9]Note that the case when $p(\boldsymbol{x};\theta)\, \phi(\boldsymbol{x})\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) \not\to 0$, does not correspond to any sensible estimator, because the value of $\phi(\boldsymbol{x})$ at $\|\boldsymbol{x}\| \to \infty$ will have an influence on the value of the gradient estimation. In that case, because $p(\boldsymbol{x};\theta) \to 0$ the probability of sampling at infinity will tend to 0, and the gradient variance will explode. This condition does however mean that if one wants to construct a sensible estimator, care must be taken to ensure that $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ does not go to infinity too fast, e.g. as explained by Jankowiak and Karaletsos (2019).

multiply with $\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})$); we show that for each $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ the corresponding weighting function $\psi(\boldsymbol{x})$ for $\phi(\boldsymbol{x})$ that gives an unbiased gradient estimator is unique, i.e.

$$\psi(\boldsymbol{x}) = \frac{1}{q(\boldsymbol{x})}\left(\nabla_{\boldsymbol{x}} \cdot \big(p(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big) + \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}\right).$$

Theorem 4 (The flow gradient estimator characterizes the space of all single sample unbiased LR–RP estimators)**.** *Every unbiased gradient estimator of the form*

$$\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\left[\phi(\boldsymbol{x})\right] = \mathbb{E}_{\boldsymbol{x}\sim q(\boldsymbol{x})}\left[\boldsymbol{v}(\boldsymbol{x})\cdot\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x}) + \psi(\boldsymbol{x})\phi(\boldsymbol{x})\right],$$

where $\phi(\boldsymbol{x})$ is an arbitrary function, is a special case of the estimator characterized by

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\left[\phi(\boldsymbol{x})\right] = \mathbb{E}_{\boldsymbol{x}\sim q(\boldsymbol{x})}\Bigg[&\frac{p(\boldsymbol{x};\theta)}{q(\boldsymbol{x})}\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\cdot\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})\\ &+\frac{1}{q(\boldsymbol{x})}\left(\nabla_{\boldsymbol{x}}\cdot\big(p(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big) + \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}\right)\phi(\boldsymbol{x})\Bigg].\end{aligned}$$

Proof. Note that there is a corresponding $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ for an arbitrary $\boldsymbol{v}(\boldsymbol{x})$ given by $\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) = \boldsymbol{v}(\boldsymbol{x})\frac{q(\boldsymbol{x})}{p(\boldsymbol{x};\theta)}$. We will show that $\psi(\boldsymbol{x}) = \frac{1}{q(\boldsymbol{x})}\left(\nabla_{\boldsymbol{x}}\cdot\big(p(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big) + \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}\right)$ is the unique function, s.t. the estimator with $\boldsymbol{v}(\boldsymbol{x})$ is unbiased. Suppose that there exist $\psi(\boldsymbol{x})$ and $f(\boldsymbol{x})$, s.t.

$$\int q(\boldsymbol{x})\left(\boldsymbol{v}(\boldsymbol{x})\cdot\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x}) + \psi(\boldsymbol{x})\phi(\boldsymbol{x})\right)\,\mathrm{d}\boldsymbol{x} = \int q(\boldsymbol{x})\left(\boldsymbol{v}(\boldsymbol{x})\cdot\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x}) + f(\boldsymbol{x})\phi(\boldsymbol{x})\right)\,\mathrm{d}\boldsymbol{x}$$

for any $\phi(\boldsymbol{x})$. Rearrange the equation into

$$\int q(\boldsymbol{x})\phi(\boldsymbol{x})\left(\psi(\boldsymbol{x}) - f(\boldsymbol{x})\right)\,\mathrm{d}\boldsymbol{x} = 0,$$

then pick $\phi(\boldsymbol{x}) = \psi(\boldsymbol{x}) - f(\boldsymbol{x})$ from which we get

$$\int q(\boldsymbol{x})\left(\psi(\boldsymbol{x}) - f(\boldsymbol{x})\right)^2\,\mathrm{d}\boldsymbol{x} = 0.$$

Therefore, $\psi = f$. As $\boldsymbol{v}(\boldsymbol{x})$ was arbitrary, and there is exactly one corresponding $\psi(\boldsymbol{x})$ for each $\boldsymbol{v}(\boldsymbol{x})$, then the flow gradient estimator characterizes all possible single sample unbiased gradient estimators. □

A corollary of Theorem 4 is that the estimator is independent of $\phi(\boldsymbol{x})$ if and only if

$$\nabla_{\boldsymbol{x}}\cdot\big(p(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big) + \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i} = 0,$$

which is the transport equation required by the work of Jankowiak and Obermeyer (2018). This means that the work by Jankowiak and Obermeyer (2018) characterized the space of all possible RP gradients. On the other hand, if $\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) = \boldsymbol{0}$, then we recover the unique LR gradient estimator.

E.3 A few examples of other estimators as special cases of the flow gradient

Notation:
$g(\epsilon;\theta) = \boldsymbol{x}$ is the reparameterization transformation.
$S(\boldsymbol{x};\theta) = \epsilon$ is the standardization function (the inverse of $g(\epsilon;\theta)$).
$|X|$ is the determinant of a matrix X.

Weighted average of LR and RP: Consider a gradient estimator given by

$$\mathbb{E}_{\boldsymbol{x}\sim p(x;\theta)}\left[k\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})\nabla_{\theta}g(\boldsymbol{x})+(1-k)\frac{\mathrm{d}\log p\,(x;\theta)}{\mathrm{d}\theta}\phi(\boldsymbol{x})\right], \tag{51}$$

where $k\in[0,1]$ is a weighting factor for the two gradient estimators, and the gradient w.r.t. one parameter, $\nabla_{\theta_i}g(\boldsymbol{x})$, is defined as $\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\Big|_{\epsilon=S(\boldsymbol{x};\theta)}$, where $g(\epsilon;\theta)$ is a typical reparameterization. We will show that this estimator belongs to the flow estimator class in Eq. (50). The corresponding flow field for any θ_i is $\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) = k\nabla_{\theta_i}g(\boldsymbol{x}) = k\,\frac{\partial g(\epsilon;\theta)}{\partial\theta}\Big|_{\epsilon=S(\boldsymbol{x};\theta)}$. Because g is a reparameterization, the flow without the k factor, $\tilde{\boldsymbol{u}}_{\theta_i} = \nabla_{\theta_i}g(\boldsymbol{x})$, satisfies the transport equation (i.e. the $\phi(\boldsymbol{x})$ term would disappear), $\nabla_{\boldsymbol{x}}\cdot\left(p\,(\boldsymbol{x};\theta)\,\tilde{\boldsymbol{u}}_{\theta_i}(\boldsymbol{x})\right)+\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}=0$, and hence, $\nabla_{\boldsymbol{x}}\cdot\left(p\,(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\right)=-k\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}$, which gives the desired result when plugging into Eq. (50).

For clarity, we will show a specific example with a Gaussian distribution, and considering the gradient w.r.t. one of the mean parameters μ_i, in which case $g=\mu+\sigma\epsilon$ and $\nabla_{\mu_i}g(\boldsymbol{x})=[0,...,1,0,...,0]$, where the 1 is at the i^{th} dimension. In this case, $\nabla_{\boldsymbol{x}}\cdot\left(p\,(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\right)=\nabla_{\mu_i}g(\boldsymbol{x})\cdot\nabla_{\boldsymbol{x}}p\,(\boldsymbol{x};\theta)+p\,(\boldsymbol{x};\theta)\,\nabla_{\boldsymbol{x}}\cdot\nabla_{\mu_i}g(\boldsymbol{x})$. Note that $\nabla_{\boldsymbol{x}}\cdot\nabla_{\mu_i}g(\boldsymbol{x})=0$ because $\nabla_{\mu_i}g(\boldsymbol{x})$ is constant, and $\nabla_{\mu_i}g(\boldsymbol{x})\cdot\nabla_{\boldsymbol{x}}p\,(\boldsymbol{x};\theta)=[0,...,1,...,0]\cdot\left[\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}x_1},...,\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}x_i},...,\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}x_D}\right]=\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}x_i}$. Finally, note that $\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}x_i}=-\frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\mu_i}$, which gives the desired result.

Reparameterization gradient: Here, we show explicitly that the reparameterization flow, $\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) = \frac{\partial g(\epsilon;\theta)}{\partial\theta}\Big|_{\epsilon=S(\boldsymbol{x};\theta)}$, satisfies the transport equation, i.e.

$$\nabla_{\boldsymbol{x}}\cdot\left(p\,(\boldsymbol{x};\theta)\,\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}\right)+\frac{\mathrm{d}p\,(\boldsymbol{x};\theta)}{\mathrm{d}\theta_i}=0. \tag{52}$$

First, we expand the divergence:

$$\begin{aligned}\nabla_{\boldsymbol{x}}\cdot\left(p\,(\boldsymbol{x};\theta)\,\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}\right)=&\nabla_{\boldsymbol{x}}p\,(\boldsymbol{x};\theta)\cdot\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}\\&+p\,(\boldsymbol{x};\theta)\,\mathrm{Tr}\left[\frac{\partial^2 g(\epsilon;\theta)}{\partial\theta_i\partial\boldsymbol{\epsilon}}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}\frac{\mathrm{d}S(\boldsymbol{x};\theta)}{\mathrm{d}\boldsymbol{x}}\right].\end{aligned} \tag{53}$$

Consider the equation

$$p\,(\epsilon)=p\,(\boldsymbol{x};\theta)\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|=p\,(g(\epsilon;\theta);\theta)\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|, \tag{54}$$

where the determinant factor comes from the change of coordinates. When differentiating Eq. (54) w.r.t. θ, it becomes 0, because $p\,(\epsilon)$ does not depend on θ, and we will show that this gives rise to the transport condition in Eq. (52).

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}\theta_i}\left(p\,(g(\epsilon;\theta);\theta)\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|\right)=0\\&=\left(\frac{\partial p\,(g(\epsilon;\theta);\theta)}{\partial\theta_i}\bigg|_{g=\boldsymbol{x}}+\frac{\partial p\,(\boldsymbol{x};\theta)}{\partial\boldsymbol{x}}\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right)\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|+p\,(g(\epsilon;\theta);\theta)\,\frac{\partial}{\partial\theta_i}\left(\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|\right)\\&=\left(\frac{\partial p\,(\boldsymbol{x};\theta)}{\partial\theta_i}+\frac{\partial p\,(\boldsymbol{x};\theta)}{\partial\boldsymbol{x}}\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right)\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|+p\,(g(\epsilon;\theta);\theta)\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|\mathrm{Tr}\left[\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}^{-1}\frac{\partial^2 g(\epsilon;\theta)}{\partial\theta_i\partial\epsilon}\right]\\&=\frac{\partial p\,(\boldsymbol{x};\theta)}{\partial\theta_i}+\frac{\partial p\,(\boldsymbol{x};\theta)}{\partial\boldsymbol{x}}\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}+p\,(\boldsymbol{x};\theta)\,\mathrm{Tr}\left[\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}^{-1}\frac{\partial^2 g(\epsilon;\theta)}{\partial\theta_i\partial\epsilon}\right],\end{aligned} \tag{55}$$

where we used the matrix identity $\frac{\partial|X|}{\partial y}=|X|\mathrm{Tr}\left[X^{-1}\frac{\partial X}{\partial y}\right]$, e.g. see the matrix cookbook (Petersen and Pedersen, 2012). Also, note that we canceled the $\left|\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}\right|$ term, because the total sum is 0, so division by a constant does not affect the equation. Further, note that $\frac{\mathrm{d}g(\epsilon;\theta)}{\mathrm{d}\epsilon}^{-1} = \frac{\mathrm{d}S(\boldsymbol{x};\theta)}{\mathrm{d}\boldsymbol{x}}$ from the inverse function theorem, because g and S are inverses. Finally, note that $\mathrm{Tr}\left[\frac{\mathrm{d}S(\boldsymbol{x};\theta)}{\mathrm{d}\boldsymbol{x}}\frac{\partial^2 g(\epsilon;\theta)}{\partial\theta_i\partial\epsilon}\right] = \mathrm{Tr}\left[\frac{\partial^2 g(\epsilon;\theta)}{\partial\theta_i\partial\epsilon}\frac{\mathrm{d}S(\boldsymbol{x};\theta)}{\mathrm{d}\boldsymbol{x}}\right]$ because $\mathrm{Tr}\,[AB] = \mathrm{Tr}\,[BA]$ for arbitrary matrices A and B. Combining the results in Eqs. (52) and (53) we see that they are the same as Eq. (55); hence, reparameterization gradients satisfy the transport equation for arbitrary invertible transformations g and S.

Implicit reparameterization gradient: The flow field for implicit reparameterization gradients (Figurnov et al., 2018) is given by

$$\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) = -\left(\nabla_{\boldsymbol{x}} S(\boldsymbol{x};\theta)\right)^{-1}\nabla_{\theta_i} S(\boldsymbol{x};\theta) = -\left(\frac{\partial S(\boldsymbol{x};\theta)}{\partial \boldsymbol{x}}\right)^{-1}\frac{\partial S(\boldsymbol{x};\theta)}{\partial\theta_i}. \tag{56}$$

We will show that the transport equation is satisfied by showing the equivalence to the flow field for the explicit reparameterization gradient case. We perform the reverse of the implicit reparameterization gradient derivation by Figurnov et al. (2018). We can write $\boldsymbol{x} = g(S(\boldsymbol{x};\theta);\theta)$. Then, by taking the derivative w.r.t. θ_i of both sides, the left-hand side will be 0, because it is independent of θ_i, and the right-hand side will give us our desired equation:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\theta_i}\Big(g(S(\boldsymbol{x};\theta);\theta)\Big) &= \frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)} + \frac{\partial g(\epsilon;\theta)}{\partial\epsilon}\frac{\partial S(\boldsymbol{x};\theta)}{\partial\theta_i} = 0\\ \Rightarrow \frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)} &= -\frac{\partial g(\epsilon;\theta)}{\partial\epsilon}\frac{\partial S(\boldsymbol{x};\theta)}{\partial\theta_i}.\end{aligned} \tag{57}$$

Now, note that based on the inverse function theorem $\left(\frac{\partial S(\boldsymbol{x};\theta)}{\partial\boldsymbol{x}}\right)^{-1} = \frac{\partial g(\epsilon;\theta)}{\partial\epsilon}$, so Eqs. (56) and (57) are the same. Note that $\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\Big|_{\epsilon=S(\boldsymbol{x};\theta)}$ was the flow $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ for the explicit reparameterization gradient in the previous section. Therefore, the implicit reparameterization gradient also explicitly satisfies the transport equation.

Generalized reparameterization gradient: Our work also generalizes the generalized reparameterization gradient (GRP) (Ruiz et al., 2016b). Unlike standard RP, in GRP, the distribution for ϵ may also depend on θ, i.e. $p(\epsilon;\theta)$. Ruiz et al. (2016b) showed that GRP can be written with the equation:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\left[\phi(\boldsymbol{x})\right] &= \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\bigg[\boldsymbol{h}_{\theta_i}\big(S(x;\theta);\theta\big)\cdot\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})\\ &+\bigg(\nabla_{\boldsymbol{x}}\log p(\boldsymbol{x};\theta)\cdot\boldsymbol{h}_{\theta_i}(S(\boldsymbol{x};\theta);\theta) + \frac{\partial\log p(\boldsymbol{x};\theta)}{\partial\theta_i} + u_{\theta_i}(S(\boldsymbol{x};\theta);\theta)\bigg)\phi(\boldsymbol{x})\bigg]\\ &= \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\bigg[\boldsymbol{h}_{\theta_i}\big(S(x;\theta);\theta\big)\cdot\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})\\ &+\frac{1}{p(\boldsymbol{x};\theta)}\bigg(\nabla_{\boldsymbol{x}} p(\boldsymbol{x};\theta)\cdot\boldsymbol{h}_{\theta_i}(S(\boldsymbol{x};\theta);\theta) + p(\boldsymbol{x};\theta)\,u_{\theta_i}(S(\boldsymbol{x};\theta);\theta) + \frac{\partial p(\boldsymbol{x};\theta)}{\partial\theta_i}\bigg)\phi(\boldsymbol{x})\bigg],\end{aligned} \tag{58}$$

where $\boldsymbol{h}_{\theta_i}(\epsilon;\theta) = \frac{\partial g(\epsilon;\theta)}{\partial\theta_i}$, $u_{\theta_i}(\epsilon;\theta) = \frac{\partial\log J(\epsilon;\theta)}{\partial\theta_i}$, and $J(\epsilon;\theta) = \left|\frac{\partial g(\epsilon;\theta)}{\partial\epsilon}\right|$. Note that $u_{\theta_i}(\epsilon;\theta) \neq \boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ as in our previous notation, and the cause of the confusing notation is that we chose to use the same notation in Eq. (58) as the work by Ruiz et al. (2016b). Comparing the $\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})$ terms in Eqs. (58) and (50) it is clear that we must have

$$\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) = \boldsymbol{h}_{\theta_i}(S(\boldsymbol{x};\theta);\theta) = \frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}. \tag{59}$$

By expanding the divergence term in Eq. (50), $\nabla_{\boldsymbol{x}}\cdot\big(p(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\big) = \nabla_{\boldsymbol{x}} p(\boldsymbol{x};\theta)\cdot\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) + p(\boldsymbol{x};\theta)\,\nabla_{\boldsymbol{x}}\cdot\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$, and comparing the $\phi(\boldsymbol{x})$ terms, one sees that to achieve equivalence between the two equations, we must have

$$u_{\theta_i}(S(\boldsymbol{x};\theta);\theta) = \nabla_{\boldsymbol{x}}\cdot\boldsymbol{u}_{\theta_i}(\boldsymbol{x}) = \nabla_{\boldsymbol{x}}\cdot\boldsymbol{h}_{\theta_i}(S(\boldsymbol{x};\theta);\theta). \tag{60}$$

The left-hand side term is

$$u_{\theta_i}(S(\boldsymbol{x};\theta);\theta) = \frac{\partial \log J(\epsilon;\theta)}{\partial \theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)} = \frac{\partial}{\partial \theta_i}\left(\log\left|\frac{\partial g(\epsilon;\theta)}{\partial \epsilon}\right|\right)\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}$$
$$= \mathrm{Tr}\left[\frac{\partial g(\epsilon;\theta)}{\partial \epsilon}^{-1}\frac{\partial^2 g(\epsilon;\theta)}{\partial \theta_i \partial \epsilon}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}\right] \tag{61}$$
$$= \mathrm{Tr}\left[\frac{\partial^2 g(\epsilon;\theta)}{\partial \theta_i \partial \epsilon}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}\frac{\partial S(\boldsymbol{x};\theta)}{\partial \boldsymbol{x}}\right],$$

where we used the fact that $\frac{\partial g(\epsilon;\theta)}{\partial \epsilon}^{-1} = \frac{\partial S(\boldsymbol{x};\theta)}{\partial \boldsymbol{x}}$ from the inverse function theorem, as well as the matrix identities $\frac{\partial}{\partial y}\log|X| = \mathrm{Tr}[X^{-1}\frac{\partial X}{\partial y}]$ and $\mathrm{Tr}[AB] = \mathrm{Tr}[BA]$, e.g. see the matrix cookbook (Petersen and Pedersen, 2012). We will show that the right-hand side term is the same:

$$\nabla_{\boldsymbol{x}} \cdot \boldsymbol{h}_{\theta_i}(S(\boldsymbol{x};\theta);\theta) = \nabla_{\boldsymbol{x}} \cdot \frac{\partial g(\epsilon;\theta)}{\partial \theta_i}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)} = \mathrm{Tr}\left[\frac{\partial^2 g(\epsilon;\theta)}{\partial \theta_i \partial \epsilon}\bigg|_{\epsilon=S(\boldsymbol{x};\theta)}\frac{\partial S(\boldsymbol{x};\theta)}{\partial \boldsymbol{x}}\right], \tag{62}$$

which is the desired result. Therefore, the GRP gradient is a special case of the flow gradient estimator in Eq. (50). Note that in the standard reparameterization gradient derivation, we used the fact that $\frac{\mathrm{d}p(\epsilon)}{\mathrm{d}\theta_i} = 0$ to show that the transport equation holds, and hence that the multiplier for $\phi(\boldsymbol{x})$ disappears, but in the GRP case, the distribution for ϵ depends on θ, so $\frac{\mathrm{d}p(\epsilon;\theta)}{\mathrm{d}\theta_i} \neq 0$, and the $\phi(\boldsymbol{x})$ term does not disappear. Finally, note that it is an open question whether the reverse may also be true—could it be that the GRP and flow gradient estimator spaces are equal? To show that they are equal, one would have to find a generalized reparameterization corresponding to each arbitrary $\boldsymbol{u}_{\theta_i}(\boldsymbol{x})$. However, we believe that if at all possible, the reparameterization corresponding to some complicated flow field may be quite bizarre, while in the flow framework, one just has to do a dot product between the flow and the gradient to compute the estimator.

E.4 Flow gradients with discontinuities

In Eq. (47) when applying the divergence theorem, we assumed that $\boldsymbol{F} = p(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\phi(\boldsymbol{x})$ is a continuous piecewise differentiable vector field, so that the divergence theorem would hold. Moreover, in Eq. (49), we assumed that $\frac{\mathrm{d}}{\mathrm{d}\theta}\int p(\boldsymbol{x};\theta)\,\phi(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = \int \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta}\phi(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$, which will not be true when $p(\boldsymbol{x};\theta)$ has discontinuities. Here, we extend the theory to all possible discontinuities. Our work casts previous work on RP gradients for discontinuous functions $\phi(\boldsymbol{x})$ into the flow gradient framework (Lee et al., 2018; Cong et al., 2019), but also characterizes all other possible discontinuities, which have not been considered in the literature yet. We consider first what to do about discontinuities in $\boldsymbol{F}$, then discuss discontinuities in $p(\boldsymbol{x};\theta)$ (which would automatically mean that $\boldsymbol{F}$ is also discontinuous).

Discontinuities in $\boldsymbol{F} = p(\boldsymbol{x};\theta)\,\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\phi(\boldsymbol{x})$: A well-known result states that if there are discontinuities in $\boldsymbol{F}$ appearing on the surface M, then the divergence theorem has to be modified by adding the jump across the surface, giving the equation

$$\int_V \nabla_{\boldsymbol{x}} \cdot \boldsymbol{F}\mathrm{d}V = \int_S \boldsymbol{F}\cdot\mathrm{d}\boldsymbol{S} + \int_M \Delta\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{M}, \tag{63}$$

where S is the surface enclosing the volume V, M is the surface inside V where the discontinuities occur, and $\Delta\boldsymbol{F}$ is the difference in the vector field $\boldsymbol{F}$ between the two sides of the discontinuity. We give a short proof of this claim:

Partition the volume V into K disjoint regions, s.t. $\boldsymbol{F}$ is continuous in each region V_i, and the discontinuity surface M is contained at the surface boundaries of the disjoint regions. In this case, because $\boldsymbol{F}$ is continuous in each region V_i, we can apply the divergence theorem for each region separately, and sum the results.

$$\sum_{i=1}^{K}\int_{V_i}\nabla_{\boldsymbol{x}}\cdot\boldsymbol{F}\mathrm{d}V = \sum_{i=1}^{K}\int_{S_i}\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{S}. \tag{64}$$

It will turn out, that the flow across the inner surfaces will cancel, unless there is a discontinuity, which gives the additional term in Eq. (63). We write the surface integral, as the sum over the outer surfaces $S_{out,i}$ (i.e. exterior to V), which add up to S, and the inner surfaces $S_{in,i}$ (i.e. interior to V), which enclose M as well as the other region boundaries. Then, the right-hand side term becomes:

$$\begin{aligned}\sum_{i=1}^{K}\int_{S_i}\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{S} &= \sum_{i=1}^{K}\int_{S_{out,i}}\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{S}+\sum_{i=1}^{K}\int_{S_{in,i}}\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{S}\\ &= \int_{S}\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{S}+\int_{M}\Delta\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{M},\end{aligned} \tag{65}$$

where each outer surface only appears for one of the volumes V_i, and gives the first term, while each inner surface belongs to two regions V_j and V_k, which gives rise to the difference in the vector fields because the surface normal pointing outward from the volume is in opposite directions for the two separate volumes. Note that if there is no discontinuity at an inner surface, then the difference in vector fields is 0, and it can be ignored, so only the integral across M matters. Finally, note that $\sum_{i=1}^{K}\int_{V_i}\nabla_{\boldsymbol{x}}\cdot\boldsymbol{F}\mathrm{d}V=\int_{V}\nabla_{\boldsymbol{x}}\cdot\boldsymbol{F}\mathrm{d}V$, which concludes the proof. To estimate the surface integral, one could sample additional points on the surface M, and apply importance sampling to estimate the integral, e.g. as done by Lee et al. (2018). Once, we have estimated the surface integral across M, the flow gradient estimator in Eq. (50), which we denote here by E, has to be modified to make it unbiased. In particular, in Eq. (48) we assumed that the divergence will be 0, but in the discontinuous case, it will instead equal $G=\int_M\Delta\boldsymbol{F}\cdot\mathrm{d}\boldsymbol{M}$, so we must subtract it from the estimator, to make it unbiased, i.e. we use the estimator $E-G$, or in practice, not exactly G, but an estimator for G:

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}\left[\phi(\boldsymbol{x})\right]\\ &=\mathbb{E}_{\boldsymbol{x}\sim q(\boldsymbol{x})}\left[\frac{p\left(\boldsymbol{x};\theta\right)}{q\left(\boldsymbol{x}\right)}\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\cdot\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x})+\frac{1}{q\left(\boldsymbol{x}\right)}\left(\nabla_{\boldsymbol{x}}\cdot\left(p\left(\boldsymbol{x};\theta\right)\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\right)+\frac{\mathrm{d}p\left(\boldsymbol{x};\theta\right)}{\mathrm{d}\theta_i}\right)\phi(\boldsymbol{x})\right]\\ &-\mathbb{E}_{\boldsymbol{x}\sim q_M(\boldsymbol{x})}\left[\frac{1}{q_M(\boldsymbol{x})}\Delta\left(p\left(\boldsymbol{x};\theta\right)\phi(\boldsymbol{x})\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\right)\cdot\hat{\boldsymbol{m}}(\boldsymbol{x})\right],\end{aligned} \tag{66}$$

where q_M is a probability distribution on the surface M, where the discontinuity occurs, $\Delta(\cdot)$ computes the change across the discontinuity, and $\hat{\boldsymbol{m}}(\boldsymbol{x})$ is the surface normal vector on M pointing in the opposite direction in which Δ is computed.

Another intuitive way to prove the claim in Eq. (63) would be to define a parameterized smooth relaxation of the vector field $\boldsymbol{F}(\boldsymbol{x})$, where the discontinuous steps are swapped with smooth transitions, given by $\tilde{\boldsymbol{F}}(\boldsymbol{x};\gamma)$, where $\lim_{\gamma\to\infty}\tilde{\boldsymbol{F}}(\boldsymbol{x};\gamma)=\boldsymbol{F}(\boldsymbol{x})$. In this case, the divergence theorem can be applied on $\tilde{\boldsymbol{F}}$, and taking the limit will give the theorem for $\boldsymbol{F}$. It would turn out that the integral of $\nabla_{\boldsymbol{x}}\cdot\tilde{\boldsymbol{F}}$ over the discontinuous regions M would not disappear as the tightness of the smooth transitions is increased, because while the region of integration shrinks, the magnitude of the gradient would also increase, and the integral across the discontinuity will tend to the change in $\boldsymbol{F}$.

We can now cast previous works (Lee et al., 2018; Cong et al., 2019) about discontinuities into the flow gradient framework, and show that they arise by considering a discontinuity in $\boldsymbol{F}$. In particular, they consider the special case of a discontinuity in $\phi(\boldsymbol{x})$, which we will explain next.

Discontinuities in $\phi(\boldsymbol{x})$: When $\phi(\boldsymbol{x})$, has discontinuities with jumps $\Delta\phi(\boldsymbol{x})$ at the surface M, then $\Delta\boldsymbol{F}$ in Eq. (63) is $\Delta(p\left(\boldsymbol{x};\theta\right)\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\phi(\boldsymbol{x}))=p\left(\boldsymbol{x};\theta\right)\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\Delta\phi(\boldsymbol{x})$. We will explain that this gives rise to the reparameterization gradients with discontinuous models by Lee et al. (2018). In their Theorem 1, they gave the equation

$$\nabla_{\theta_i}\mathrm{ELBO}=\mathbb{E}_{\epsilon\sim p(\epsilon)}\left[\nabla_\theta h(\epsilon;\theta)\right]+\sum_{i=1}^{K}\int_{S_i}p\left(\epsilon\right)h(\epsilon;\theta)\boldsymbol{V}(\epsilon;\theta)\cdot\mathrm{d}\boldsymbol{S}_\epsilon, \tag{67}$$

where $\mathrm{ELBO}=\mathbb{E}_{\epsilon\sim p(\epsilon)}\left[h(\epsilon;\theta)\right]$, the S_i are the surfaces of K disjoint continuous volumes (similar to our proof of the discontinuous divergence theorem in Eq. (64)), $h(\epsilon;\theta)=\phi(g(\epsilon;\theta))$, and $V(\epsilon;\theta)=\left.\frac{\partial S(\boldsymbol{x};\theta)}{\partial\theta_i}\right|_{\boldsymbol{x}=g(\epsilon;\theta)}$. To compare their equation to our derivation, we must change the coordinates from the ϵ space to the $\boldsymbol{x}$ space. The main difference is the change in the direction of the surface normal $\hat{\boldsymbol{n}}_\epsilon$ on each S_i to $\hat{\boldsymbol{n}}_{\boldsymbol{x}}$, as well as the change in the vector field $\boldsymbol{V}(\epsilon;\theta)$. In our derivation, we will first transform the surface integral into a volume integral by using the divergence theorem, and then perform the change in coordinates, as this allows to ignore the change in orientation of the surface normal. The surface integral is equal to the volume integral of the divergence:

$$\int_{S_i} p(\epsilon)\, h(\epsilon;\theta)\boldsymbol{V}(\epsilon;\theta)\cdot \mathrm{d}\boldsymbol{S}_\epsilon = \int_{V_i} \nabla\cdot\Big(p(\epsilon)\, h(\epsilon;\theta)\boldsymbol{V}(\epsilon;\theta)\Big)\mathrm{d}V_\epsilon. \tag{68}$$

The vector field $\boldsymbol{V}(\epsilon;\theta)$ denotes a rate of change in the ϵ space; to change the coordinates to the $\boldsymbol{x}$ space, we apply the chain rule:

$$\boldsymbol{u}(\boldsymbol{x};\theta) = \frac{\partial g(\epsilon;\theta)}{\partial\epsilon}\boldsymbol{V}(\epsilon;\theta) = \frac{\partial g(\epsilon;\theta)}{\partial\epsilon}\left.\frac{\partial S(\boldsymbol{x};\theta)}{\partial\theta_i}\right|_{\boldsymbol{x}=g(\epsilon;\theta)} = -\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)},$$

from Eq. (57).

When performing a change of coordinates for the divergence, we can apply the Voss-Weyl formula (Grinfeld, 2013)

$$\nabla_\epsilon\cdot\boldsymbol{V}(\epsilon;\theta) = \frac{1}{|J|}\nabla_{\boldsymbol{x}}\cdot\left(|J|\boldsymbol{u}(\boldsymbol{x};\theta)\right), \tag{69}$$

where $|J|$ is the determinant of the Jacobian $J=\frac{\partial S(\boldsymbol{x})}{\partial\boldsymbol{x}}$. Similarly, the change in coordinates for the volume element is given by $\mathrm{d}V_\epsilon = |J|\mathrm{d}V_{\boldsymbol{x}}$. Combining the results, we have

$$\begin{aligned}\int_{V_i}\nabla_\epsilon\cdot\Big(p(\epsilon)\, h(\epsilon;\theta)\boldsymbol{V}(\epsilon;\theta)\Big)\mathrm{d}V_\epsilon &= -\int_{V_i}\nabla_{\boldsymbol{x}}\cdot\left(\left|\frac{\partial S(\boldsymbol{x})}{\partial\boldsymbol{x}}\right| p(\epsilon)\,\phi(\boldsymbol{x})\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)}\right)\mathrm{d}V_{\boldsymbol{x}}\\ &= -\int_{V_i}\nabla_{\boldsymbol{x}}\cdot\left(p(\boldsymbol{x};\theta)\,\phi(\boldsymbol{x})\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)}\right)\mathrm{d}V_{\boldsymbol{x}},\end{aligned} \tag{70}$$

where $\phi(\boldsymbol{x}) = h(S(\boldsymbol{x}))$, as $h(\epsilon;\theta) := \phi(g(\epsilon;\theta))$, and $\left|\frac{\partial S(\boldsymbol{x})}{\partial\boldsymbol{x}}\right| p(\epsilon) = p(\boldsymbol{x};\theta)$, and the minus sign comes because $\boldsymbol{u}(\boldsymbol{x};\theta) = -\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)}$. Finally, we apply the divergence theorem again:

$$-\int_{V_i}\nabla_{\boldsymbol{x}}\cdot\left(p(\boldsymbol{x};\theta)\,\phi(\boldsymbol{x})\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)}\right)\mathrm{d}V_{\boldsymbol{x}} = -\int_{S_i} p(\boldsymbol{x};\theta)\,\phi(\boldsymbol{x})\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)}\cdot\mathrm{d}\boldsymbol{S}_{\boldsymbol{x}}. \tag{71}$$

Note that $\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)} = \boldsymbol{u}_{\theta_i}(\boldsymbol{x})$ is the flow for the reparameterization gradient in Eq. (52). Therefore, we have

$$-\sum_{i=1}^{K}\int_{S_i} p(\boldsymbol{x};\theta)\,\phi(\boldsymbol{x})\left.\frac{\partial g(\epsilon;\theta)}{\partial\theta_i}\right|_{\epsilon=S(\boldsymbol{x};\theta)}\cdot\mathrm{d}\boldsymbol{S}_{\boldsymbol{x}} = -\int_M p(\boldsymbol{x};\theta)\,\Delta\phi(\boldsymbol{x})\boldsymbol{u}_{\theta_i}(\boldsymbol{x})\cdot\mathrm{d}\boldsymbol{M}, \tag{72}$$

where the change to $\Delta\phi(\boldsymbol{x})$ instead of summing the flow through each surface is analogous to what we explained in Eq. (65): the integral across the inner surfaces occurs in two of the disjoint volumes, which gives rise to integrating the change in the vector field. We can now see that their estimator, given in Eq. (67), is analogous to our estimator for the discontinuous case in Eq. (66), but performed in the ϵ space, and while assuming a perfect reparameterization.

GO gradient estimator: Next, we show that the GO gradient estimator for discrete variables (Cong et al., 2019) can also be derived from our framework. We illustrate the equivalence on a simplified 1-dimensional case, which is straightforward to generalize to arbitrary dimensions. Cong et al. (2019) consider a discrete variable $y\in\{0,...,\infty\}$, a function $\phi(y)$, and a probability distribution $p(y;\theta)$. Moreover, they define $Q(y;\theta)=\sum_{i=0}^{y}p(i;\theta)$. Then, they provide the gradient estimator:

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{E}_{y\sim p(y;\theta)}\left[\phi(y)\right] = \mathbb{E}_{y\sim p(y;\theta)}\left[\frac{-1}{p(y;\theta)}\frac{\mathrm{d}Q(y;\theta)}{\mathrm{d}\theta}\Big(\phi(y+1)-\phi(y)\Big)\right], \tag{73}$$

We will show that this estimator can be derived from the flow theory. Consider a continuous space of x, with $\phi(x)=\phi(y)$, when $y\le x<y+1$, and $\int_y^{y+1}p(x;\theta)\,\mathrm{d}x = p(y;\theta)$, which occurs when $p(x;\theta)=p(y;\theta)$ if

$y \leq x < y+1$, then the expectation over x and y give the same results. Next, we define a flow field $\boldsymbol{u}_{\theta_i}(x)$ that satisfies the transport equation, i.e. the $\phi(x)$ term in the flow gradient estimator will disappear, and only the $\nabla_{\boldsymbol{x}}\phi(x)$ terms will remain. But $\nabla_{\boldsymbol{x}}\phi(x) = 0$ everywhere, so that term also disappears. It will only be necessary to estimate the flow $\boldsymbol{u}_{\theta_i}(x)$ through the discontinuous boundaries at each $x = y+1$, i.e. the q_M term in Eq. (66). We apply the divergence theorem on $\nabla_{\boldsymbol{x}} \cdot (p(x;\theta)\,\boldsymbol{u}_{\theta_i}(x))$:

$$\begin{aligned}\int_0^{y+1} \nabla_{\boldsymbol{x}} \cdot (p(x;\theta)\,\boldsymbol{u}_{\theta_i}(x))\,\mathrm{d}x &= \int_S p(x;\theta)\,\boldsymbol{u}_{\theta_i}(x) \cdot \mathrm{d}\boldsymbol{S} \\ &= p(x=y+1;\theta)\,\boldsymbol{u}_{\theta_i}(x=y+1) = p(y;\theta)\,\boldsymbol{u}_{\theta_i}(y+1),\end{aligned} \tag{74}$$

where the equation comes from the fact that the only exterior surface with non-zero $\boldsymbol{u}_{\theta_i}(x)$ is at $x = y+1$, and by definition $p(x=y+1;\theta) = p(y;\theta)$. Moreover, note that because of the transport equation $\nabla_{\boldsymbol{x}} \cdot (p(x;\theta)\,\boldsymbol{u}_{\theta_i}(x)) = -\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}$, and hence, we can obtain another expression for $p(y;\theta)\,\boldsymbol{u}_{\theta_i}(y+1)$ by computing the integral

$$\int_0^{y+1} \nabla_{\boldsymbol{x}} \cdot (p(x;\theta)\,\boldsymbol{u}_{\theta_i}(x))\,\mathrm{d}x = \sum_{i=0}^{y} -\frac{\mathrm{d}p(i;\theta)}{\mathrm{d}\theta} = -\frac{\mathrm{d}Q(y;\theta)}{\mathrm{d}\theta}. \tag{75}$$

Therefore, we have $p(y;\theta)\,\boldsymbol{u}_{\theta_i}(y+1) = -\frac{\mathrm{d}Q(y;\theta)}{\mathrm{d}\theta}$. Now, noting that the jump across the surface at $x = y+1$ is given by $\phi(y+1) - \phi(y)$, and taking into account of the opposite direction of $\hat{\boldsymbol{m}}$ we have that

$$\Delta\boldsymbol{F}(x=y+1) \cdot \hat{\boldsymbol{m}}(x=y+1) = \frac{\mathrm{d}Q(y;\theta)}{\mathrm{d}\theta}\Big(\phi(y+1) - \phi(y)\Big).$$

Plugging into Eq. (66), while noting that $q_M(x=y+1) := p(y;\theta)$, we obtain the GO gradient estimator. In the original derivation of Cong et al. (2019), they used an algebraic derivation based on integration by parts. The difficulty was how to extend the derivation for discrete variables. They extended it by using an Abel transformation (Abel, 1895) and summation by parts. Our derivation, on the other hand, casts this gradient estimator into the flow gradient framework, and provides a physical insight into the principle behind the estimator.

Discontinuities in $p(\boldsymbol{x};\theta)$: When there are discontinuities in $p(\boldsymbol{x};\theta)$, then $\boldsymbol{F}$ will be discontinuous, which is straightforward to deal with based on our previous example; however, that is not enough, because $\int \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta}\phi(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \neq \frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x};\theta)}[\phi(\boldsymbol{x})]$ in general. To deal with the mismatch, it becomes necessary to split the domain of integration into piecewise continuous domains, then perform the integration by taking into account the movement of the discontinuous boundary. In general, when differentiating an integral of a function $f(\boldsymbol{x};\theta)$ in a moving domain $D(\theta)$, the integral is given by

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\int_{D(\theta)} f(\boldsymbol{x};\theta)\mathrm{d}\boldsymbol{x} = \int_{D(\theta)} \frac{\mathrm{d}f(\boldsymbol{x};\theta)}{\mathrm{d}\theta}\mathrm{d}\boldsymbol{x} + \int_S f(\boldsymbol{x};\theta)\boldsymbol{v}(\boldsymbol{x}) \cdot \mathrm{d}\boldsymbol{S}, \tag{76}$$

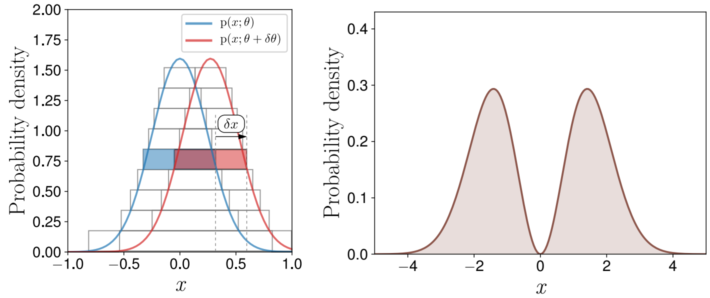
where S is the surface around the domain and $\boldsymbol{v}(\boldsymbol{x})$ is the velocity of the location of the point on the domain, i.e. $\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}\theta}$, where $\boldsymbol{x} \in S$. Performing the usual split of the domain of integration into piecewise continuous domains, and applying the rule for integration under moving boundaries gives the equation

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\int p(\boldsymbol{x};\theta)\,\phi(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = \int \frac{\mathrm{d}p(\boldsymbol{x};\theta)}{\mathrm{d}\theta}\phi(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \int_C \phi(\boldsymbol{x})\Delta p(\boldsymbol{x};\theta)\,\boldsymbol{v}(\boldsymbol{x}) \cdot \mathrm{d}\boldsymbol{S}, \tag{77}$$

where C is the surface where the discontinuities occur, and $\Delta p(\boldsymbol{x};\theta)$ is the change in the probability density across the surfaces, computed in the opposite direction to the surface normal $\mathrm{d}\boldsymbol{S}$. Adding the correction factor based on an estimate of the integral across C will unbias the gradient estimator. The same trick was applied for deriving the estimator for RP gradients under discontinuities (Lee et al., 2018).

F Slice integral importance sampling

From Theorem 2 we saw that unlike the RP gradient case, the weighting ψ for function values $\phi(\boldsymbol{x})$ with $\boldsymbol{x} \sim p(\boldsymbol{x};\theta)$ to obtain an unbiased estimator for the gradient $\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{E}[\phi(\boldsymbol{x})]$ is unique. The only option to reduce the variance by changing the weighting would then be to sample from a different distribution $q(\boldsymbol{x})$ via importance



(a) Probability slices (b) Probability density

Figure 6: Slice integral sampling motivation and distribution for a Gaussian base distribution.

sampling. Motivated by the resemblance of the "boxes" theory in Sec. 3 to the Riemann integral, we propose to sample horizontal slices of probability mass resembling the Lebesgue integral. Such an approach appears attractive, because if the location of the slice is moved by modifying the parameters of the distribution (e.g., by changing the mean), then the derivative of the expected value of the integral over the slice will depend only on the value at the edges of the slice (because the probability density in the middle would not change). To obtain the gradient estimator, it will only be necessary to compute the probability density $q_L(\boldsymbol{x}; \theta)$. We derive such a "slice integral" distribution corresponding to the Gaussian distribution. The method resembles the seminal work by Neal (2003) on slice sampling in Markov chain Monte Carlo methods. We call our new distribution the L-distribution, and it is plotted in Fig. 6b. In retrospect, it turned out that this distribution is optimal under the assumption that $\phi(x)$ is linear.

Derivation of the pdf of the L-distribution: One way to sample whole slices of a probability distribution would be to sample a height h between 0 and $p_{\max}$ proportionally to the probability mass at that height. The probability mass at a height h is just given by $2|x-\mu|$ where x is such that $p\left(x;\mu,\sigma\right) = h$, i.e., $2|x-\mu|$ is the distance between the edges of $p\left(x;\mu,\sigma\right)$. The probability mass corresponding to x is then given by $2|x-\mu|\mathrm{d}h$. Performing a change of coordinates to the x-domain, and splitting the mass between the two edges of the slice, we get $|x-\mu|\mathrm{d}h = |x-\mu|\left|\frac{\mathrm{d}p(x;\mu,\sigma)}{\mathrm{d}x}\right|\mathrm{d}x$. This gives a closed-form normalized pdf for the L-distribution:

$$
\begin{aligned}
q_L(x;\mu,\sigma) = |x-\mu|\left|\frac{\mathrm{d}p\left(x;\mu,\sigma\right)}{\mathrm{d}x}\right| &= |x-\mu|p\left(x;\mu,\sigma\right)\frac{|x-\mu|}{\sigma^2} \\
&= \frac{|x-\mu|^2}{\sqrt{2\pi}\,\sigma^3}\exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right).
\end{aligned} \tag{78}
$$

One can recognize that Eq. (78) is actually just a Maxwell-Boltzmann distribution reflected about the origin with the probability mass split between the two sides.

Sampling from the L-distribution: To sample from this distribution, it is necessary to sample points proportionally to the length of the slices. It suffices to sample uniformly in the area under the curve in the space augmented with the height dimension h, then selecting the slice on which the sampled point lies. This can be achieved with the three steps: 1) sample a point from the base distribution: $x_s \sim p\left(x;\mu,\sigma\right)$, 2) sample a height: $h \sim \mathrm{unif}\left(0, p\left(x_s;\mu,\sigma\right)\right)$, 3) sample x from one of the two edges of the slice $p^{-1}(h;\mu,\sigma) = \{x : p\left(x;\theta\right) = h\}$, where $p^{-1}(h)$ inverts the pdf, and computes the locations of x that give a probability density h. For the L-distribution, this can be achieved by sampling $\epsilon_x \sim \mathcal{N}(0,1)$ and $\epsilon_h \sim \mathrm{unif}(0,1)$ and transforming these by the equation:

$$
x = \mu \pm \sigma\sqrt{-2\log(\epsilon_h) + \epsilon_x^2}\ . \tag{79}
$$

Now it is straightforward to obtain the LR gradient estimator:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\mu}\mathbb{E}_{x\sim p(x;\theta)}\left[\phi(x)\right] &= \mathbb{E}_{x\sim q(x;\theta)}\left[\frac{\frac{\mathrm{d}p}{\mathrm{d}\mu}}{q(x;\theta)}\phi(x)\right] = \mathbb{E}_{x\sim q(x;\theta)}\left[\frac{1}{x-\mu}\phi(x)\right] \\ &= \mathbb{E}\left[\frac{\mathrm{sgn}(x-\mu)}{\sigma\sqrt{-2\log(\epsilon_h)+\epsilon_x^2}}\phi(x)\right].\end{aligned} \tag{80}$$

References

Abel, N. H. (1895). *Untersuchungen über die Reihe: 1+(m/1) x+ m·(m-1)/(1· 2)· x2+ m·(m-1)·(m-2)/(1· 2· 3)· x3+...* Number 71. W. Engelmann. E.4

Asadi, K., Allen, C., Roderick, M., Mohamed, A.-r., Konidaris, G., and Littman, M. (2017). Mean actor critic. *stat*, 1050:1. A

Ciosek, K. and Whiteson, S. (2018). Expected policy gradients. In *Thirty-Second AAAI Conference on Artificial Intelligence*. A

Cong, Y., Zhao, M., Bai, K., and Carin, L. (2019). Go gradient for expectation-based objectives. *arXiv preprint arXiv:1901.06020*. 2.4, 4, E.4, E.4, E.4, E.4

Conti, E., Madhavan, V., Such, F. P., Lehman, J., Stanley, K., and Clune, J. (2018). Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. In *Advances in Neural Information Processing Systems*, pages 5027–5038. 1

Farquhar, G., Whiteson, S., and Foerster, J. (2019). Loaded dice: Trading off bias and variance in any-order score function estimators for reinforcement learning. *arXiv preprint arXiv:1909.10549*. A

Figurnov, M., Mohamed, S., and Mnih, A. (2018). Implicit reparameterization gradients. In *Advances in Neural Information Processing Systems*, pages 441–452. 2.2, A, E.3, E.3

Foerster, J., Farquhar, G., Al-Shedivat, M., Rocktäschel, T., Xing, E., and Whiteson, S. (2018). Dice: The infinitely differentiable Monte Carlo estimator. In *International Conference on Machine Learning*, pages 1529–1538. A

Gal, Y. (2016). *Uncertainty in deep learning*. PhD thesis, PhD thesis, University of Cambridge. 2.4, A

Geffner, T. and Domke, J. (2018). Using large ensembles of control variates for variational inference. In *Advances in Neural Information Processing Systems*, pages 9960–9970. A

Glynn, P. W. (1990). Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84. 1

Grathwohl, W., Choi, D., Wu, Y., Roeder, G., and Duvenaud, D. (2017). Backpropagation through the void: Optimizing control variates for black-box gradient estimation. *arXiv preprint arXiv:1711.00123*. A

Greensmith, E., Bartlett, P. L., and Baxter, J. (2004a). Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5(Nov):1471–1530. 2.4

Greensmith, E., Bartlett, P. L., and Baxter, J. (2004b). Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5(Nov):1471–1530. A

Grinfeld, P. (2013). *Introduction to tensor analysis and the calculus of moving surfaces*. Springer. E.4

Gu, S., Levine, S., Sutskever, I., and Mnih, A. (2015). MuProp: Unbiased backpropagation for stochastic neural networks. *arXiv preprint arXiv:1511.05176*. A

Gu, S., Lillicrap, T., Ghahramani, Z., Turner, R. E., and Levine, S. (2016). Q-prop: Sample-efficient policy gradient with an off-policy critic. *arXiv preprint arXiv:1611.02247*. A

Gu, S. S., Lillicrap, T., Turner, R. E., Ghahramani, Z., Schölkopf, B., and Levine, S. (2017). Interpolated policy gradient: Merging on-policy and off-policy gradient estimation for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 3846–3855. A

Ha, D. and Schmidhuber, J. (2018). Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, pages 2450–2462. 1

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347. 1

Jang, E., Gu, S., and Poole, B. (2016). Categorical reparameterization with Gumbel-Softmax. *arXiv preprint arXiv:1611.01144*. A

Jankowiak, M. and Karaletsos, T. (2019). Pathwise derivatives for multivariate distributions. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 333–342. 4

Jankowiak, M. and Obermeyer, F. (2018). Pathwise derivatives beyond the reparameterization trick. In *International Conference on Machine Learning*, pages 2240–2249. 2.4, 4, 4, A, E.1, E.2, E.2

Jiang, N. and Li, L. (2016). Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning*, pages 652–661. A

Jie, T. and Abbeel, P. (2010). On a connection between importance sampling and the likelihood ratio policy gradient. In *Advances in Neural Information Processing Systems*, pages 1000–1008. 2.3, A, C

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*. 1

L'Ecuyer, P. (1990). A unified view of the IPA, SF, and LR gradient estimation techniques. *Management Science*, 36(11):1364–1383. 2.4

L'Ecuyer, P. (1991). An overview of derivative estimation. In *1991 Winter Simulation Conference Proceedings.*, pages 207–217. IEEE. 2.4

Lee, W., Yu, H., and Yang, H. (2018). Reparameterization gradient for non-differentiable models. In *Advances in Neural Information Processing Systems*, pages 5553–5563. 4, E.4, E.4, E.4, E.4, E.4

Liu, H., Feng, Y., Mao, Y., Zhou, D., Peng, J., and Liu, Q. (2017). Action-depedent control variates for policy optimization via stein's identity. *arXiv preprint arXiv:1710.11198*. 2.4

Maddison, C. J., Mnih, A., and Teh, Y. W. (2016). The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*. A

Mania, H., Guy, A., and Recht, B. (2018). Simple random search of static linear policies is competitive for reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 1800–1809. 3, B, C

Mao, J., Foerster, J., Rocktäschel, T., Al-Shedivat, M., Farquhar, G., and Whiteson, S. (2019). A baseline for any order gradient estimation in stochastic computation graphs. In *International Conference on Machine Learning*, pages 4343–4351. A

Metz, L., Maheswaranathan, N., Nixon, J., Freeman, C. D., and Sohl-Dickstein, J. (2019). Understanding and correcting pathologies in the training of learned optimizers. In *International Conference on Machine Learning*. A

Mohamed, S., Rosca, M., Figurnov, M., and Mnih, A. (2019). Monte Carlo gradient estimation in machine learning. *arXiv preprint arXiv:1906.10652*. 1, 2.4, 2, A

Munos, R., Stepleton, T., Harutyunyan, A., and Bellemare, M. (2016). Safe and efficient off-policy reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 1054–1062. A

Neal, R. M. (2003). Slice sampling. *The annals of statistics*, 31(3):705–767. F

Nesterov, Y. and Spokoiny, V. (2017). Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566. A

Owen, A. B. (2013). *Monte Carlo theory, methods and examples*. 2.4

Parmas, P. (2018). Total stochastic gradient algorithms and applications in reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 10204–10214. A

Parmas, P., Rasmussen, C. E., Peters, J., and Doya, K. (2018). PIPPS: Flexible model-based policy search robust to the curse of chaos. In *International Conference on Machine Learning*, pages 4062–4071. 2.4, A, A

Peters, J. and Schaal, S. (2008). Reinforcement learning of motor skills with policy gradients. *Neural networks*, 21(4):682–697. 1

Petersen, K. B. and Pedersen, M. S. (2012). The matrix cookbook (version: November 15, 2012). E.3, E.3

Ranganath, R., Tran, D., and Blei, D. (2016). Hierarchical variational models. In *International Conference on Machine Learning*, pages 324–333. 2.4, 2.4

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, pages 1278–1286. 1, A

Riesz, F. (1907). *Sur une espèce de géométrie analytique des systèmes de fonctions sommables*. Gauthier-Villars. 4

Ruiz, F., Titsias, M., and Blei, D. (2016a). Overdispersed black-box variational inference. In *32nd Conference on Uncertainty in Artificial Intelligence 2016, UAI 2016*, pages 647–656. A

Ruiz, F. J., Titsias, M. K., and Blei, D. (2016b). The generalized reparameterization gradient. In *Advances in Neural Information Processing Systems*, pages 460–468. 4, E.3, E.3

Salimans, T., Ho, J., Chen, X., Sidor, S., and Sutskever, I. (2017). Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*. 1, 3, B, C

Schulman, J., Heess, N., Weber, T., and Abbeel, P. (2015a). Gradient estimation using stochastic computation graphs. In *Advances in Neural Information Processing Systems*, pages 3528–3536. A

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. (2015b). Trust region policy optimization. In *International Conference on Machine Learning*, pages 1889–1897. 1

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*. 1

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge. 1

Sutton, R. S., McAllester, D. A., Singh, S. P., and Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063. 1

Thomas, P. and Brunskill, E. (2016). Data-efficient off-policy policy evaluation for reinforcement learning. In *International Conference on Machine Learning*, pages 2139–2148. A

Titsias, M. K. and Lázaro-Gredilla, M. (2015). Local expectation gradients for black box variational inference. In *Advances in neural information processing systems*, pages 2638–2646. A

Tucker, G., Bhupatiraju, S., Gu, S., Turner, R., Ghahramani, Z., and Levine, S. (2018). The mirage of action-dependent baselines in reinforcement learning. In *International Conference on Machine Learning*, pages 5022–5031. A

Tucker, G., Mnih, A., Maddison, C. J., Lawson, J., and Sohl-Dickstein, J. (2017). REBAR: Low-variance, unbiased gradient estimates for discrete latent variable models. In *Advances in Neural Information Processing Systems*, pages 2627–2636. A

Walder, C. J., Nock, R., Ong, C. S., and Sugiyama, M. (2019). New tricks for estimating gradients of expectations. *arXiv preprint arXiv:1901.11311*. 2

Weaver, L. and Tao, N. (2001). The optimal reward baseline for gradient-based reinforcement learning. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 538–545. Morgan Kaufmann Publishers Inc. B

Weber, T., Heess, N., Buesing, L., and Silver, D. (2019). Credit assignment techniques in stochastic computation graphs. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2650–2660. A

Wierstra, D., Schaul, T., Peters, J., and Schmidhuber, J. (2008). Natural evolution strategies. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pages 3381–3387. IEEE. 1

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256. 1

Wu, A. (2019). Generalized transformation-based gradient. *arXiv preprint arXiv:1911.02681*. A

Xu, M., Quiroz, M., Kohn, R., and Sisson, S. A. (2019). Variance reduction properties of the reparameterization trick. In *International Conference on Artificial Intelligence and Statistics*. A