Solving Inverse PDE Problems using Grid-Free Monte Carlo Estimators: Supplemental Material

EKREM FATIH YILMAZER, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
DELIO VICINI, Google Inc., Switzerland
WENZEL JAKOB, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

A GREEN'S FUNCTION AND POISSON KERNEL

In this section, we provide the Green's function and Poisson kernel. The Green's function for a ball $B(x)$ is:

$$G^\sigma_{2D}(x,y) = G^\sigma_{2D}(r) = \frac{1}{2\pi}\left[K_0(r\sqrt{\sigma}) - \frac{K_0(R\sqrt{\sigma})}{I_0(R\sqrt{\sigma})} I_0(r\sqrt{\sigma})\right]$$
$$G^\sigma_{3D}(x,y) = G^\sigma_{3D}(r) = \frac{1}{4\pi}\left[\frac{sinh((R-r)*\sqrt{\sigma})}{r sinh(R\sqrt{\sigma})}\right] \quad (1)$$

where $r \coloneqq \|y-x\|$, R is the radius of the ball $B(x)$, I_0 and K_0 are the zeroth order modified Bessel functions of the first and the second kind. When sampling proportional to the Green's function, we normalize the PDF by dividing by:

$$|G^\sigma_{2D}| = \int_{B(x)} G^\sigma(x,y)\,\mathrm{d}y = \frac{1}{\sigma}\left[1 - \frac{1}{I_0(R\sqrt{\sigma})}\right]$$
$$|G^\sigma_{3D}| = \int_{B(x)} G^\sigma(x,y)\,\mathrm{d}y = \frac{1}{\sigma}\left[1 - \frac{R\sqrt{\sigma}}{sinh(R\sqrt{\sigma})}\right] \quad (2)$$

The Poisson kernel is the derivative of the Green's function along boundary normal:

$$P^\sigma_{2D}(x,y) = \frac{n_y \cdot (y-x)}{2\pi r^2} Q^\sigma_{2D}(x,y)$$
$$Q^\sigma_{2D}(x,y) = Q^\sigma(r) = \left[K_1(r\sqrt{\sigma}) + I_1(r\sqrt{\sigma})\frac{K_0(R\sqrt{\sigma})}{I_0(R\sqrt{\sigma})}\right] r\sqrt{\sigma}$$
$$P^\sigma_{3D}(x,y) = \frac{n_y \cdot (y-x)}{4\pi r^3} Q^\sigma_{3D}(x,y) \quad (3)$$
$$Q^\sigma_{3D}(x,y) = e^{-r\sqrt{\sigma}}(1 + r\sqrt{\sigma}) + \left[cosh(r\sqrt{\sigma}) r\sqrt{\sigma} - sinh(r\sqrt{\sigma})\right]\frac{e^{-R\sqrt{\sigma}}}{sinh(R\sqrt{\sigma})},$$

where n_y is the unit normal at y. Q^σ converges to 1 as $\sigma \to 0$. We use the notation $G^\sigma_{r,2D}(r) = 2\pi r G^\sigma_{2D}(r)$ and $G^\sigma_{r,3D}(r) = 4\pi r^2 G^\sigma_{3D}(r)$ for the radial distribution of the Green's function that we use for sampling.

B DELTA TRACKING WITH WALK ON STARS

In this section, we extend the delta tracking solver for spatially-varying PDEs to handle Neumann boundaries. Our goal is to solve the following mixed-boundary condition problem:

$$\begin{aligned} \nabla\left[\alpha(x)\nabla(u(x))\right] - \sigma(x)u(x) &= 0 && x \in \Omega \\ u(x) &= g(x) && x \in \partial\Omega_d \\ \partial_n u(x) &= h(x) && x \in \partial\Omega_n\,. \end{aligned} \quad (4)$$

Authors' Contact Information: Ekrem Fatih Yilmazer, ekrem.yilmazer@epfl.ch, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland; Delio Vicini, vicini@google.com, Google Inc., Zurich, Switzerland; Wenzel Jakob, wenzel.jakob@epfl.ch, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland.

We assume the source term to be zero to slightly reduce notational clutter. We then apply Girsanov and delta tracking transformations and arrive at a screened Poisson equation:

$$\begin{aligned} \Delta U(x) - \bar{\sigma}U(x) &= -(\bar{\sigma} - \sigma'(x))U(x) && x \in \Omega \\ U(x) &= \sqrt{\alpha(x)}g(x) && x \in \partial\Omega_d \\ \partial_n U(x) &= \frac{1}{2}U(x)\partial_n\alpha(x) + \sqrt{\alpha(x)}h(x) && x \in \partial\Omega_n, \end{aligned} \quad (5)$$

where $U(x)$ and $\sigma'(x)$ are identical to the Dirichlet-only version that is presented in the main text. For completeness, we repeat their definitions here:

$$u(x) = \alpha(x)^{-1/2}U(x) \quad (6)$$
$$\sigma'(x) = \frac{\sigma(x)}{\alpha(x)} + \frac{1}{2}\left(\frac{\Delta\alpha(x)}{\alpha(x)} - \frac{|\nabla(\ln\alpha(x))|^2}{2}\right). \quad (7)$$

Unlike the Dirichlet-only version, Equation 5 now relates the solution to its normal derivative on the boundary $\partial\Omega_n$. This is a Robin boundary condition, which is not supported by the methods we build on. We avoid handling Robin boundary conditions by requiring the diffusion coefficient's normal derivative ($\partial_n\alpha(x)$) to be zero on $\partial\Omega_n$.

Assuming $h(z) = 0$ to further reduce notational complexity, the solution of Equation 5 satisfies:

$$\kappa(x)U(x) = \int_{\partial St} P^{\bar{\sigma}}(x,x')U(x')\,\mathrm{d}x' + \int_{St} G^{\bar{\sigma}}(x,x')U(x')\,\mathrm{d}x$$
$$\kappa(x) = 1/2 \quad x \in \partial\Omega_n, \quad \kappa(x) = 1 \quad x \notin \partial\Omega_n, \quad (8)$$

Similar to the delta tracking version of walk on spheres, we only recursively evaluate one of the two integrals. We follow Sawhney et al. [2022] and write the integral equation in the directional domain:

$$\kappa(x)U(x) = \frac{1}{|S|}\int_{\omega\in S}\left(Q^{\bar{\sigma}}(r_i)U(\omega,r_i) + \int_0^{r_i} G^{\bar{\sigma}}_r(r)U(\omega,r)\,\mathrm{d}r\right)\mathrm{d}\omega. \quad (9)$$

Here, S is the unit sphere of directions, $|S|$ its surface area, r_i is the distance from the star's origin to ∂St along direction ω. Similar to walk on spheres, we use the following relation to determine sampling weights:

$$Q^{\bar{\sigma}}(r_i) = 1 - \bar{\sigma}\int_0^{r_i} G^{\bar{\sigma}}_r(r)\,\mathrm{d}r = 1 - \bar{\sigma}|G|_0^{r_i}, \quad (10)$$

where we use $|G|_0^{r_i}$ as a shorthand for the integral of $G^{\bar{\sigma}}_r(r)$ from 0 to r_i. We would like to sample the first term in Equation 9 with a probability of $1-\bar{\sigma}|G|_0^{r_i}$. We first decide to either sample the volume or the boundary term. This is done using a boundary probability of

$\mathbb{P}_{\partial St} = 1 - \bar{\sigma}|G^{\bar{\sigma}}|$. In either case, we then sample a uniform direction ω.

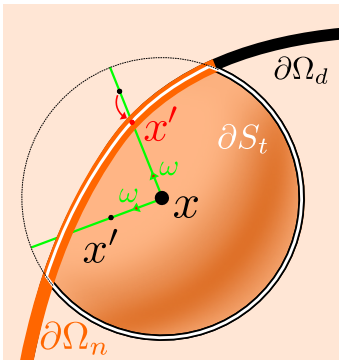


Fig. 1. St sampling in delta tracking.

If the boundary integral is selected, the final sample is the intersection the ray (x, ω) and ∂St. Otherwise, we sample a distance $r \in [0, R]$ proportional to the radial Green's function ($G^{\bar{\sigma}}_r(r)$) and project the sampled point back to ∂St if $r > r_i$. This way, we increase the boundary selection probability to the desired $\mathbb{P}_{\partial St} = 1 - \bar{\sigma}|G|^{r_i}_0$.

The sampling of the terms that occur when $f(x) \neq 0$ and/or $h(z) \neq 0$ is straightforward and follows the estimators discussed in the main text.

C NEUMANN BOUNDARY SAMPLING

In this section, we present a sampling strategy for Neumann boundary contributions that does not require a BVH. Recall the Neumann boundary term in the boundary integral equation:

$$\int_{\partial St_n} G^\sigma(x, z) h(z) \, \mathrm{d}z \tag{11}$$

To estimate this integral using Monte Carlo, we would like to uniformly sample points in ∂St_n. A naive approach would be to sample points on $\partial\Omega_n$ and check if they are in ∂St_n. This is inefficient, as ∂St_n might only be a relatively small subset of $\partial\Omega_n$. Sawhney et al. [2023] use a BVH to sample points that are likely in ∂St_n. Since we model the domain boundary using Bézier curves, building an efficient BVH is challenging and we instead use a directional sampling strategy. Parameterizing the above integral over directions yields:

$$\int_S G^\sigma(x, z(x,\omega)) h(z(x,\omega)) |J(x,\omega)| \mathbb{1}_{St_n}(z(x,\omega)) \, \mathrm{d}\omega. \tag{12}$$

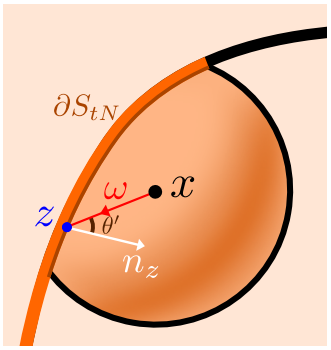


Fig. 2. Quantities used in Equation 13.

Here, ω is a direction vector, $z(x, \omega)$ the intersection of the ray (x, ω) with ∂St_n, $h(z(x, \omega))$ the Neumann boundary value, $|J(x, \omega)|$ the Jacobian of the parameterization and $\mathbb{1}_{St_n}(z(x, \omega))$ indicates if the ray intersection is in ∂St_n. Expanding the Jacobian term, the equation becomes:

$$\int_S G^\sigma(r) h(z(x,\omega)) \frac{r}{\cos(\theta')} \mathbb{1}_{St_n}(z(x,\omega)) \, \mathrm{d}\omega, \tag{13}$$

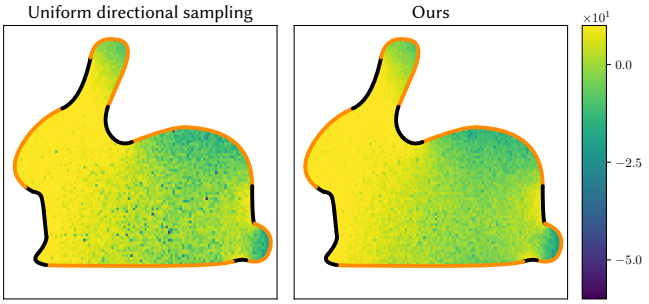
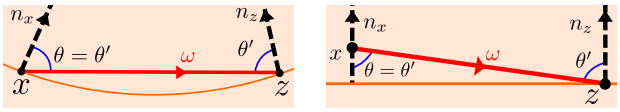


Fig. 4. Comparison between directionally uniform and our sampling. Black lines are Dirichlet and orange lines Neumann boundaries.

where $r := \|z(x, \omega) - x\|$ is the ray intersection distance and θ' is the angle between the normal at the ray intersection (n_z) and ω. The Green's function only depends on the distance and we therefore write it as $G^\sigma(r)$. Figure 2 illustrates some of the terms.

We found the division by the cosine term to cause high variance, as it approaches zero at grazing angles. We therefore use an importance sampling method that approximately samples proportional to this term. Assuming that ∂St_n is convex, we only encounter grazing angles when the star's origin x is very close to the boundary. If x is on the boundary, we approximate the boundary as a circle (Figure 3a). In that case, the angle θ' at location z is identical to the angle θ at x. We therefore generate samples in the upper semicircle proportional to $(\cos\theta)^{-1}$, with $\theta \in [-\pi/2, \pi/2]$. If the point is not on the boundary but close to it (Figure 3b), we apply the same sampling, but use the reverse normal direction. We clamp the PDF to constant value for angles $|\theta| > \pi/2 - \epsilon$, as $(\cos\theta)^{-1}$ itself is not integrable. We found this clamping to work well, since for these extreme angles either r (if x is on the boundary) or $G^\sigma(r)$ (if x is in the interior) approaches zero. Finally, we use multiple importance sampling to combine our strategy with uniform sampling. After generating the sample, we compute the ray intersection to check if the sampled point is in ∂St_n.



(a) x is on the boundary (b) x is near the boundary

Fig. 3. Approximation of the boundary for different cases.

Our sampling strategy is not as efficient as using a BVH, but it is easy to implement and works for any surface representation. Figure 4 compares the results of directionally uniform and our sampling. Uniform directional sampling occasionally produces outlier samples, causing noise in the final estimator.

D FD RESULTS IN 2D

Figure 5 shows gradients with respect to coefficients of the elliptic PDE. The objective function is the L_2 norm of a 32×32 solution image (i.e., the optimization tries to reduce the value of the solution)

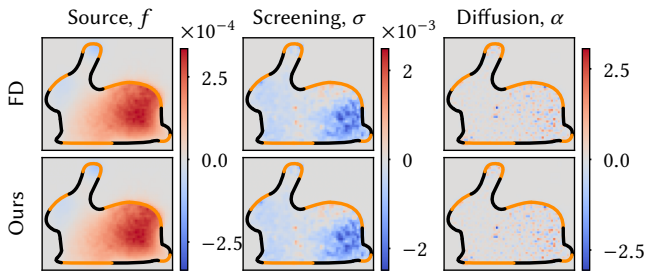


Fig. 5. Comparison between our gradient estimate and the finite differences reference for the various parameters of the general elliptic PDE in 2D.

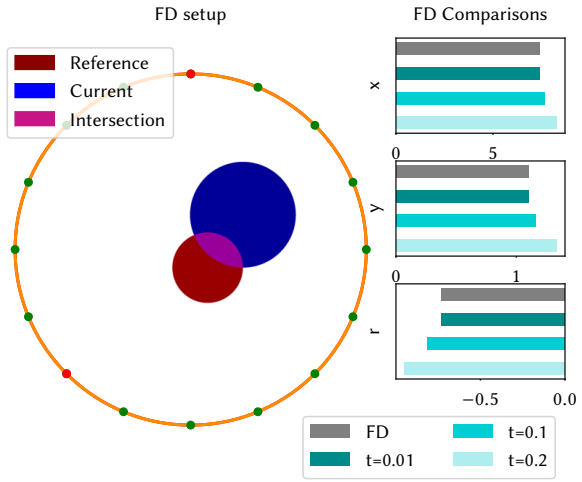


Fig. 6. Finite difference computation of the circle representation. As for SDFs (see main text), we evaluate the gradients for different normal derivative evaluation offsets t. A smaller offset significantly reduces bias.

and the coefficients are 64×64 textures with cubic interpolation. As in the 3D example, gradients constructed using path replay and weight windowing reveal a perfect match against the reference.

E FD RESULTS FOR CIRCLES

In this section, we present additional validations for the circle representation for discrete EIT. Figure 6 provides gradient validations and Figure 7 shows optimization results for a few synthetic examples

REFERENCES

Rohan Sawhney, Bailey Miller, Ioannis Gkioulekas, and Keenan Crane. 2023. Walk on Stars: A Grid-Free Monte Carlo Method for PDEs with Neumann Boundary Conditions. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2023).

Rohan Sawhney, Dario Seyb, Wojciech Jarosz, and Keenan Crane. 2022. Grid-free Monte Carlo for PDEs with spatially varying coefficients. *ACM Trans. Graph. (Proc. SIGGRAPH)* 41, 4 (2022).

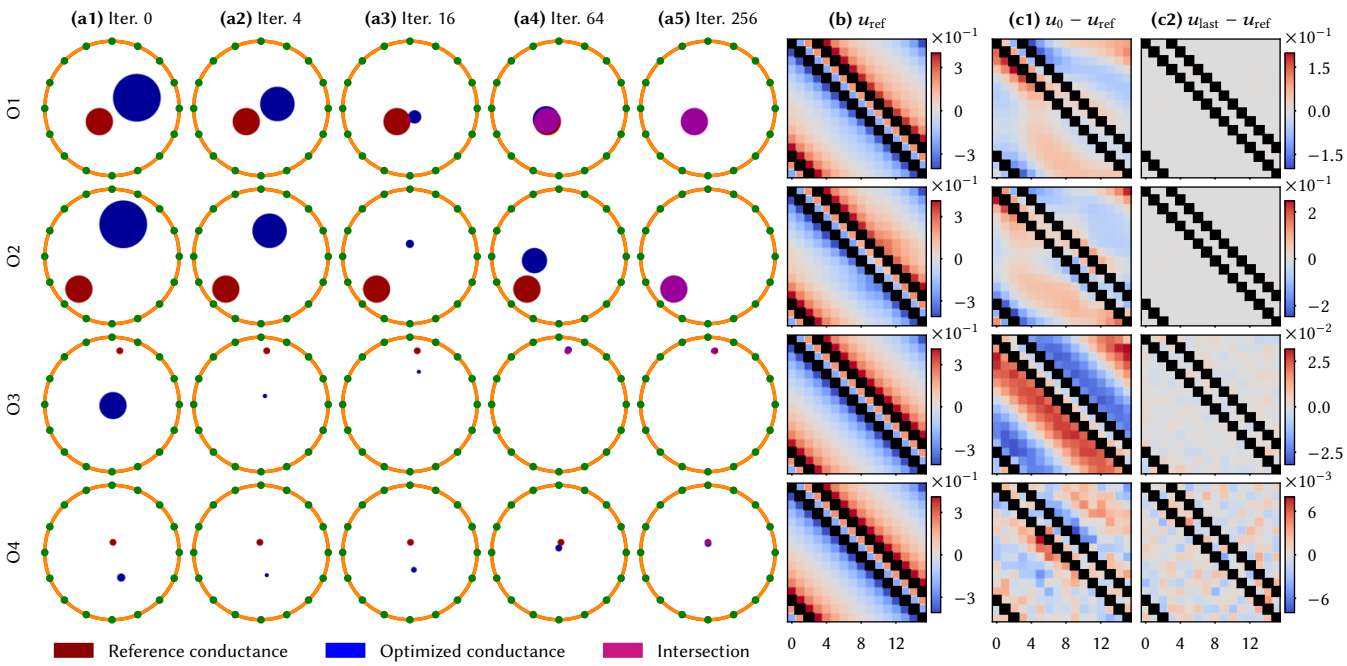


Fig. 7. Optimization results of discrete EIT using the circle representation. Columns **(a)** show the optimization progress, column **(b)** the reference voltage values and columns **(c)** show the error to the reference both at the start and end of optimization.