Differentiation Strategies for Acoustic Inverse Problems: Admittance Estimation and Shape Optimization

Nikolas Borrel-Jensen
Pasteur Labs
Brooklyn, NY USA
nikolas.borrel@simulation.science

Josiah Bjorgaard
Pasteur Labs
Brooklyn, NY USA
josiah.bjorgaard@simulation.science

Abstract

We demonstrate a practical differentiable programming approach for acoustic inverse problems through two applications: admittance estimation and shape optimization for resonance damping. First, we show that JAX-FEM's automatic differentiation (AD) enables direct gradient-based estimation of complex boundary admittance from sparse pressure measurements, achieving 3-digit precision without requiring manual derivation of adjoint equations. Second, we apply randomized finite differences to acoustic shape optimization, combining JAX-FEM for forward simulation with PyTorch3D for mesh manipulation through AD. By separating physics-driven boundary optimization from geometry-driven interior mesh adaptation, we achieve 48.1% energy reduction at target frequencies with 30-fold fewer FEM solutions compared to standard finite difference on the full mesh. This work showcases how modern differentiable software stacks enable rapid prototyping of optimization workflows for physics-based inverse problems, with automatic differentiation for parameter estimation and a combination of finite differences and AD for geometric design.

1 Introduction

Differentiable programming has revolutionized machine learning, but its potential for scientific computing remains underexplored [1, 2]. While automatic differentiation (AD) through neural networks is standard practice, applying AD to traditional physics simulators (finite elements, finite differences, etc.) introduces unique challenges: sparse linear solvers, complex geometries, and expensive forward passes [3]. This paper demonstrates how modern differentiable software stacks enable efficient solutions to acoustic inverse problems through appropriate choice of differentiation strategies.

Problem 1: Admittance Estimation. Acoustic boundary conditions are characterized by complex admittance values $\beta = \beta_r + i\beta_i$ representing material absorption and reactivity [4]. Estimating these parameters from sparse pressure measurements requires solving an inverse problem: given measurements $\{p_i^{\text{meas}}\}$, find β such that the forward model predictions match observations. Traditional approaches require deriving and implementing adjoint equations for gradient computation. Leveraging JAX-FEM's ([5]) automatic differentiation to compute exact gradients $\nabla_\beta J$ we eliminate manual adjoint derivation and efficiently compute gradients for parameter optimization.

Problem 2: Shape Optimization. Room acoustics are strongly affected by geometry [4]. Square rooms exhibit resonances at specific frequencies, creating hot spots that degrade sound quality. Optimizing room shape to minimize energy at undesired frequencies requires repeatedly solving the

1st Workshop on Differentiable Systems and Scientific Machine Learning @ EurIPS 2025.

Helmholtz equation. We efficiently enable a hybrid strategy where randomized finite differences for boundary optimization are combined with automatic differentiation for interior mesh morphing.

This paper demonstrates how to use differentiable physics (JAX-FEM [5]) and differentiable geometry (PyTorch3D [6]) to enable practical optimization workflows in acoustics. Our results show: (1) 3-digit precision admittance estimation from 25 sparse measurements using AD, and (2) 48.1% resonance energy reduction with 30-fold speedup over standard finite difference (FD) optimization.

2 Theory

2.1 Acoustic Forward Model

The acoustic pressure field $p(\mathbf{x})$ in a 2D domain Ω with frequency f is governed by the Helmholtz equation [4]

$$\nabla^2 p + k^2 p = -f(\mathbf{x}) \quad \text{in } \Omega, \tag{1}$$

where $k = 2\pi f/c$ is the wavenumber, c the speed of sound, and $f(\mathbf{x})$ a source term. We model a Gaussian point source at location $\mathbf{s}$ as

$$f(\mathbf{x}) = A \exp\left(-\frac{|\mathbf{x}-\mathbf{s}|^2}{2\sigma^2}\right). \tag{2}$$

At the boundaries $\partial\Omega$, we impose an admittance boundary condition $\frac{\partial p}{\partial n} + ik\beta p = 0$ on $\partial\Omega$, where $\beta = \beta_r + i\beta_i$ is the complex normalized admittance (dimensionless) representing wall absorption. The real part β_r characterizes energy dissipation, while the imaginary part β_i accounts for reactive effects.

2.2 Inverse Problem Formulation

Given sparse pressure measurements $\{p_i^{\text{meas}}\}$ at locations $\{\mathbf{x}_i\}$, we seek to estimate β by minimizing the weighted least-squares objective:

$$J(\beta) = w_{\text{mag}} \sum_i \left(|p_i^{\text{pred}}| - |p_i^{\text{meas}}|\right)^2 + w_{\text{phase}} \sum_i \left(\angle p_i^{\text{pred}} - \angle p_i^{\text{meas}}\right)^2 + w_{\text{rel}} \left(\frac{\|\mathbf{p}^{\text{pred}} - \mathbf{p}^{\text{meas}}\|_2}{\|\mathbf{p}^{\text{meas}}\|_2}\right)^2, \tag{3}$$

where p_i^{pred} denotes the predicted pressure from the forward model with current β estimate.

2.3 Gradient-Based Optimization with Autodifferentiation

We employ gradient-based optimization to minimize $J(\beta)$, requiring computation of $\nabla_\beta J$. Traditional approaches necessitate deriving and implementing adjoint equations [1]. Instead, we leverage the autodifferentiation capabilities of JAX-FEM, which automatically computes exact gradients through the entire FEM solve via reverse-mode differentiation. This transforms the complete forward solve $\beta \to \mathbf{K}(\beta) \to \mathbf{p}^{\text{pred}} \to J$ into a differentiable computational graph, enabling efficient gradient computation without manual derivation of sensitivity equations. The matrix $\mathbf{K}$ is a complex matrix explained in Appendix A.1.

2.4 Geometry optimization

We define the PDE-constrained shape optimization problem as

$$\min_{\partial\Omega} J(\partial\Omega); \mathcal{F}(u;\Omega) = 0, \tag{4}$$

where $\mathcal{F}$ is the governing PDE constraint and u is the state variable . JAX-FEM does not implement AD w.r.t. the mesh. To perform optimization with Equation 4 we instead exploit a key structural

property, the objective function is defined primarily through the boundary vertices $\mathbf{V}_b$ and the internal vertices $\mathbf{V}_i$ are used only for discretization. Discretizing Ω with finite elements requires N_b boundary vertices and $N_i \gg N_b$ interior vertices. This structure appears across acoustics, electromagnetics, heat transfer, structural mechanics, and fluid dynamics where boundary conditions dominate performance. Instead of computing gradients w.r.t. all $N = N_b + N_i$ vertices, we can perform optimization with a two-stage hybrid framework, detailed in Appendix A.2.

In stage 1 we perform boundary optimization with randomized gradients. To compute $\nabla_{\mathbf{V}_b} J$ efficiently without deriving adjoints, we use randomized finite differences [7] based on the Johnson-Lindenstrauss lemma [8]. Sample S random directions $\mathbf{d}_s \sim \mathcal{N}(0, I_{N_b \times d})$, compute directional derivatives $g_s = [J(\mathbf{V}_b + \epsilon \mathbf{d}_s) - J(\mathbf{V}_b)]/\epsilon$, and accumulate the gradient estimate $\hat{\nabla}_{\mathbf{V}_b} J = \frac{1}{S} \sum_{s=1}^{S} g_s \mathbf{d}_s$. This estimator is unbiased with variance $\propto 1/S$ and requires only S PDE solves (typically $S =$ 20-50) versus $2N_b$ for coordinate-wise finite differences. The PDE solver is treated as a black box so there is no need to implement AD or derive adjoints.

In stage 2 we perform interior optimization via AD. Given fixed boundary $\mathbf{V}_b$, we optimize $\mathbf{V}_i$ for mesh quality using

$$\mathcal{L}_{\text{mesh}}(\mathbf{V}_i; \mathbf{V}_b) = w_e \mathcal{L}_{\text{edge}} + w_l \mathcal{L}_{\text{Laplacian}} + w_n \mathcal{L}_{\text{normal}} + w_A \mathcal{L}_{\text{area}}, \tag{5}$$

where the losses $\mathcal{L}_{\text{edge}}$ penalize edge length irregularity, $\mathcal{L}_{\text{Laplacian}}$ Laplacian non-uniformity, $\mathcal{L}_{\text{normal}}$ face normal inconsistency, and $\mathcal{L}_{\text{area}}$ is preserving the floor area. These purely geometric losses evaluate in milliseconds via autodiff and are implemented in PyTorch3D [6], requiring no PDE solves. Interior gradients are computed via backpropagation, boundary gradients are zeroed to keep $\mathbf{V}_b$ fixed, and standard optimizers (Adam) update interior vertices. Each iteration alternates: (1) update $\mathbf{V}_b$ via S randomized gradient samples through the PDE solver, and (2) update $\mathbf{V}_i$ via M_{inner} autodiff-based mesh quality steps. Total cost is $\sim S$ PDE solves per iteration versus $\sim 2N$ for full-mesh coordinate-wise FD.

3 Results

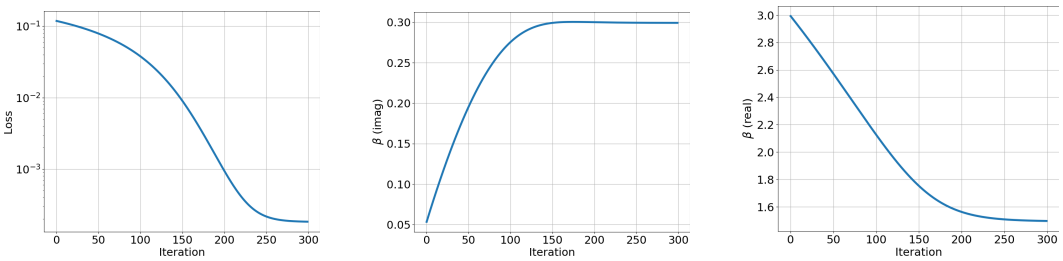
3.1 Admittance estimation

We estimate the complex boundary admittance from sparse pressure measurements in a 2 m $\times$ 2 m domain at frequency $f = 1000$ Hz. The reference pressure field is generated with true admittance $\beta_{\text{true}} = 1.5 + 0.3i$, and 2% normally distributed noise is added to simulate measurement uncertainty. A Gaussian source (defined in Equation 2) is placed at the domain center $\mathbf{s} = (1.0\text{ m}, 1.0\text{ m})$ with width $\sigma = 0.086$ m and amplitude $A = 1000$. Measurements are collected at 3 cluster locations centered at $\mathbf{x}_0 = (0.7\text{ m}, 1.7\text{ m})$, $\mathbf{x}_1 = (1.0\text{ m}, 1.7\text{ m})$, and $\mathbf{x}_2 = (1.3\text{ m}, 1.7\text{ m})$. Each cluster includes all finite element quadrature points within a 0.1 m radius, totaling 25 measurement locations. This sparse arrangement mimics a realistic microphone array where physical size constraints limit sensor density.

We used the SGD optimizer with learning rate 0.1 for 300 iterations, starting from an initial normalized admittance of $\beta_{\text{init}} = 3.0 + 0.05i$. The weights for the loss terms in Equation 3 are set to $w_{\text{mag}} = 0.5$, $w_{\text{phase}} = 0.1$, and $w_{\text{rel}} = 5.0$. We found it beneficial to set the weight for the relative error term large to learn the correct amplitude scaling. The optimizer converged after 300 iterations, estimating the admittance to three-digit precision at $\beta_{\text{est}} = 1.496 + 0.299i$, compared to the true value of $\beta_{\text{true}} = 1.5 + 0.3i$. This demonstrates that JAX-FEM's autodifferentiation successfully recovered the boundary admittance.

3.2 Room geometry optimization

We demonstrate geometry optimization on room acoustics by minimizing resonances in enclosed spaces. Square/rectangular rooms exhibit strong standing waves at specific frequencies creating hot spots which are problematic for recording studios and concert halls. The total acoustic energy $\int_\Omega |p|^2 d\mathbf{x}$ is minimized while preserving floor area $\mathcal{L}_{\text{area}} = (|\Omega| - |\Omega_0|)^2$ in Equation 5 with $w_A = 100$. We set up a test case for a 4m $\times$ 4m room, discretized into 258 vertices (68 boundary, 190 interior) and 458 triangular elements. The acoustic source is positioned at 0.5m $\times$ 0.5m from the center of the room with an amplitude factor of 1000 and frequency of 100 Hz close to the (2,1) mode. The room walls are given an absorptive boundary condition with admittance $\beta = 1.5 + 0.2j$.

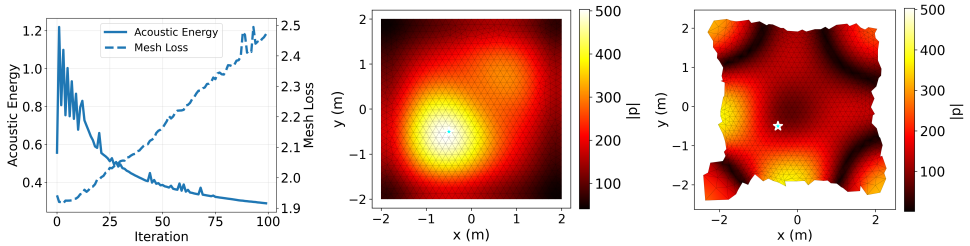


(a) Loss for 300 iterations.

(b) Imaginary part of the estimated normalized admittance β.

(c) Real part of the estimated normalized admittance β.

Figure 1: Convergence of the normalized admittance estimation. The initial guess was $\beta_{\text{est}} = 3.0 + 0.05$ and the method estimated the real admittance β_{true} within 3 digits.



(a) Optimization metrics.

(b) Mesh and $|p|$ at step 0.

(c) Mesh and $|p|$ after at step 100.

Figure 2: Results of room geometry optimization w.r.t. acoustic energy over 100 steps, showing total acoustic energy (scaled by 10^{-8}) and mesh loss (scaled by 10^2). Initial mesh and $|p|$ field and mesh geometry are shown along with mesh and $|p|$ at step 100.

We use separate Adam optimizers for each stage with learning rates of 0.1 and perform 20 mesh optimization steps for each boundary optimization step and 30 JAX-FEM solutions for the random gradient calculation. We note that this would require 916 solutions to determine an FD gradient, which is over a 30-fold reduction in number of JAX-FEM solutions required.

As shown in Figure 2a, the acoustic energy decreases while the original mesh shown in Figure 2b deforms in a manner that suppresses standing waves as shown in Figure 2c. The $\mathcal{L}_{\text{mesh}}$ increases by $\sim$25% over the total optimization, which is likely due to the more complex geometry. We also note that some exploration is observed which initially increases the acoustic energy. This could be contributed to by the stochasticity in calculation of $\nabla_{\mathbf{V}_b} J$.

Avenues of further work include multiband objectives, improved mesh conditioning, and experimentation with 3 dimensional systems. Although a multiband objective targeting the lower modes would have been of particular interest, the JAX-FEM solver failed to converge for certain configurations, which limited our ability to perform multi-band optimizations. Improvements to the solver should enable this. The optimized boundary shape shown in Figure 2c lacks typical regularity (smoothness) required for real-world applications, so additional cost functions and smoothing for boundary optimization will be explored. Finally, we intend to apply the described methods to 3D environments.

4 Conclusion

We demonstrated how automatic differentiation and randomized FD address inverse problems in room acoustics. First, we exploited AD through a numerical FEM solver to estimate complex boundary admittance from sparse measurements, achieving 3-digit accuracy. Second, we suppressed resonant frequencies through shape optimization by combining randomized FD for boundary design with automatic differentiation for interior mesh quality. Our results showcase how selecting differentiation strategies appropriate to problem structure enables efficient solutions to physics-based inverse problems.

References

[1] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18:1–43, 2018.

[2] Alexander Lavin, Hector Zenil, Brooks Paige, David C. Krakauer, Justin Emile Gottschlich, Timothy G. Mattson, Anima Anandkumar, Sanjay Choudry, Kamil Rocki, Atilim Gunecs Baydin, Carina Prunkl, Olexandr Isayev, Erik J Peterson, Peter Leonard McMahon, Jakob H. Macke, Kyle Cranmer, Jiaxin Zhang, Haruko Murakami Wainwright, Adi Hanuka, Manuela M. Veloso, Samuel A. Assefa, Stephan Zheng, and Avi Pfeffer. Simulation intelligence: Towards a new generation of scientific methods. *ArXiv*, abs/2112.03235, 2021.

[3] Philipp Holl, Yuanming Hu, Jie Xu, Wojciech Matusik, Vladlen Koltun, and Bernd Bickel. DiffPD: Differentiable projective dynamics. In *ACM SIGGRAPH Asia 2020 Conference Proceedings*, 2020.

[4] Heinrich Kuttruff. *Room Acoustics*. CRC Press, 6 edition, 2016.

[5] Tianju Xue, Chengyang Li, Yuting Hu, Weiqi Hu, Linjing Zhang, Yan Liu, Lexing Li, Xiaoyan Hu, and Yue Ma. JAX-FEM: A differentiable GPU-accelerated 3D finite element solver for automatic differentiation and inverse design. *Computer Physics Communications*, 291:108802, 2023.

[6] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Thomas Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3D deep learning with PyTorch3D. *arXiv preprint arXiv:2007.08501*, 2020.

[7] Katya Scheinberg. Finite difference gradient approximation: To randomize or not? *INFORMS Journal on Computing*, 34(5):2384–2388, 2022.

[8] W. B. Johnson and J. Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.

A Technical Appendices and Supplementary Material

A.1 Finite Element Discretization

Using the finite element method with basis functions ϕ_j, we approximate $p \approx \sum_j p_j \phi_j$. The weak formulation, obtained by multiplying by test functions v and integrating by parts, yields

$$\int_\Omega \nabla p \cdot \nabla v \, dA - k^2 \int_\Omega pv \, dA + ik\beta \int_{\partial\Omega} pv \, ds = \int_\Omega sv \, dA \tag{6}$$

This results in a complex linear system $\mathbf{K}(\beta)\mathbf{p} = \mathbf{f}$, where the system matrix $\mathbf{K}$ depends on the admittance parameter β.

A.2 Two-Stage Randomized Gradient Algorithm for Shape Optimization

Symbol	Description
$\mathbf{V}_b \in \mathbb{R}^{N_b \times d}$	Boundary vertices
$\mathbf{V}_i \in \mathbb{R}^{N_i \times d}$	Interior vertices
N_b	Number of boundary vertices
N_i	Number of interior vertices
d	Spatial dimension ($d = 2$ or 3)
$J(\mathbf{V}_b)$	Objective function (PDE-constrained)
S	Number of random samples
ϵ	Finite difference perturbation size
η_b, η_i	Learning rates for boundary/interior
w_e, w_l, w_n	Mesh quality loss weights
M_{inner}	Interior optimization iterations per outer step
K	Maximum outer iterations
τ	Convergence tolerance
$\mathcal{L}_{\text{edge}}$	Edge length regularity loss
$\mathcal{L}_{\text{Laplacian}}$	Laplacian smoothness loss
$\mathcal{L}_{\text{normal}}$	Face normal consistency loss

Algorithm 1 Randomized Boundary Gradient Estimation

Input: $\mathbf{V}_b \in \mathbb{R}^{N_b \times d}$, $\epsilon > 0$
Output: $\hat{\nabla}_{\mathbf{V}_b} J \in \mathbb{R}^{N_b \times d}$

```
1: ∇̂_{V_b} J ← 0
2: J_0 ← J(V_b)
3: for s = 1, ..., S do
4:     d_s ~ N(0, I_{N_b × d})
5:     g_s ← (J(V_b + ε d_s) − J_0) / ε
6:     ∇̂_{V_b} J ← ∇̂_{V_b} J + g_s d_s
7: end for
8: ∇̂_{V_b} J ← (1/S) ∇̂_{V_b} J
9: return ∇̂_{V_b} J
```

Algorithm 2 Interior Mesh Optimization

Input: $\mathbf{V}_i \in \mathbb{R}^{N_i \times d}$, $\mathbf{V}_b \in \mathbb{R}^{N_b \times d}$, $\eta_i > 0$
Output: $\mathbf{V}_i^* \in \mathbb{R}^{N_i \times d}$

```
for m = 1, ..., M_inner do
    L_mesh(V_i; V_b) ← w_e L_edge + w_l L_Laplacian + w_n L_normal
    g_i ← ∇_{V_i} L_mesh(V_i; V_b)
    V_i ← V_i − η_i g_i
end for
return V_i
```

Algorithm 3 Two-Stage Hybrid Geometry Optimization

Input: $\mathbf{V}_b^{(0)} \in \mathbb{R}^{N_b \times d}$, $\mathbf{V}_i^{(0)} \in \mathbb{R}^{N_i \times d}$, $\tau > 0$
Output: $(\mathbf{V}_b^*, \mathbf{V}_i^*)$

```
k ← 0, J_prev ← ∞
while k < K and |J(V_b^(k)) − J_prev| > τ do
    J_prev ← J(V_b^(k))
    ∇̂_{V_b} J ← BoundaryGradient(V_b^(k), S, ε)
    V_b^(k+1) ← V_b^(k) − η_b ∇̂_{V_b} J
    V_i^(k+1) ← InteriorOptimization(V_i^(k), V_b^(k+1), w_e, w_l, w_n, η_i, M_inner)
    k ← k + 1
end while
return (V_b^(k), V_i^(k))
```

NeurIPS Paper Checklist

1. **Claims**

Yes Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes] .

Justification: We make only quantitative claims in the abstract and introduction and make no bold claims.

2. **Limitations**

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: We discuss limitations of our method from both a theoretical and practical aspect.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

NA Answer: [Yes] .

Justification: We have cited all relevant theory and we are cross-referencing when required

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes] .

Justification: Code will be provided for reproducing all results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes] .

Justification: The data can be generated with the code and will be available.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We explain important settings, i.e. how the experiments were setup, the libraries used and important parameters. We limit the detail level for some internal settings, such as which internal jax-FEM solver was used and the specifics. If one wants to reproduce the exact results, we refer to the code.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The paper does not investigate variability.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the time spend for running the experiment and some details about the compute resource.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: No data or participants were involved, no negative social impact or harmful impact.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: The topic has not direct potential for societal impact in any significant form.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA] .

    Justification: None of the above mentioned or related aspects are involved

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes] .

Justification: All code was written by the authors and used libraries have licences aligned with our usage.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA] .

    Justification: We only contribute with code, no assets are produced.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No human subject were involved.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No participants involved.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: Not involved.