

Walkin' Robin: Walk on Stars with Robin Boundary Conditions

BAILEY MILLER*, Carnegie Mellon University, USA

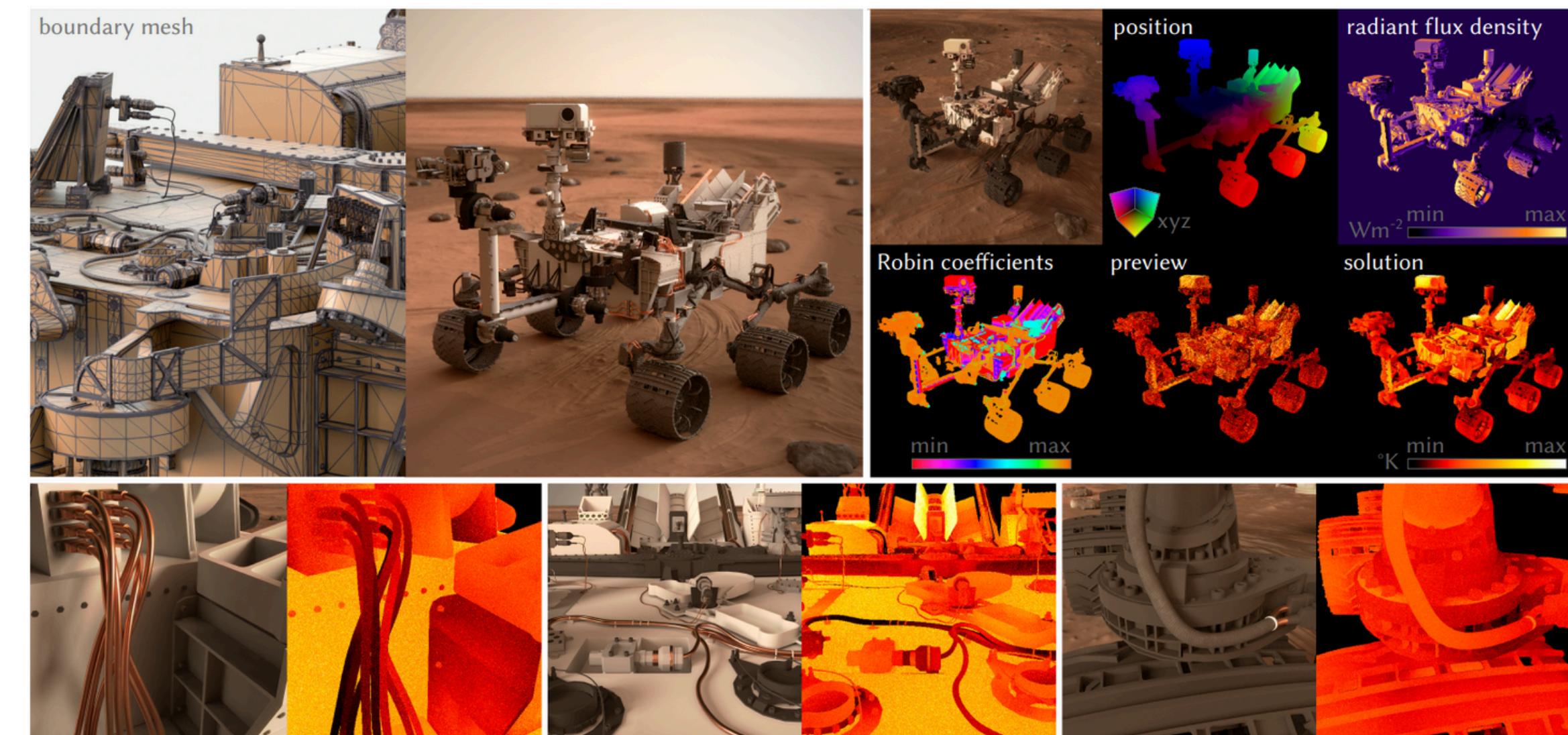
ROHAN SAWHNEY*, NVIDIA, USA

KEENAN CRANE[†], Carnegie Mellon University, USA

IOANNIS GKIOULEKAS[†], Carnegie Mellon University, USA

ACM Transactions on Graphics (SIGGRAPH) 2024

Best paper award



賴蒔雨 Shih-Yu Lai

重要性

- 處理高維問題，避免維度詛咒(Curse of Dimensionality)
- 無需結構化網格，適用於複雜幾何與不確定性量化 (Uncertainty Quantification)
- 透過隨機取樣與統計平均，適用於隨機 PDE

傳統數值方法的瓶頸

- 維度詛咒：高維 PDE (如 Boltzmann、Black-Scholes 方程) 計算成本極高
- 複雜幾何邊界：有限差分 (FDM) 和有限元素 (FEM) 需良好網格，複雜幾何時困難
- 全局求解高成本：矩陣求解資源消耗巨大
- 非線性問題困難：如 Navier-Stokes 方程，傳統方法求解受限

Monte Carlo 方法的內容與應用

- Feynman-Kac 公式：將 PDE 轉為隨機微分方程 (SDE)，用隨機模擬求解
- 隨機網格與粒子方法：適用於流體力學、氣體動力學 (DSMC)
- 多重 Monte Carlo (MLMC)：減少方差，提高收斂速度
- 機器學習結合：Physics-Informed Neural Networks (PINNs)、Deep BSDE

目前挑戰

- 收斂速度與方差減少：Quasi-Monte Carlo, Importance Sampling, Control Variates
- 高效數值實現與並行計算：GPU/TPU 加速與大規模計算優化
- 非馬可夫過程的 Monte Carlo 求解：解決延遲 PDE、非局部 PDE 的應用限制

SESSION: Monte Carlo for PDEs

RESEARCH-ARTICLE Velocity-Based Monte Carlo Fluids

Ryuksuke Sugimoto, Christopher Batty, Toshiya Hachisuka

Article No.: 8, Pages 1–11 • <https://doi.org/10.1145/3641519.3657405>

We present a velocity-based Monte Carlo fluid solver that overcomes the limitations of its existing vorticity-based counterpart. Because the velocity-based formulation is more commonly used in graphics, our Monte Carlo solver can be readily extended ...

1 226 3



RESEARCH-ARTICLE OPEN ACCESS Neural Monte Carlo Fluid Simulation

Pranav Jain, Ziyin Qu, Peter Yichen Chen, Oded Stein

Article No.: 9, Pages 1–11 • <https://doi.org/10.1145/3641519.3657438>

The idea of using a neural network to represent continuous vector fields (i.e., neural fields) has become popular for solving PDEs arising from physics simulations. Here, the classical spatial discretization (e.g., finite difference) of PDE solvers is ...

1 1,040 6



RESEARCH-ARTICLE Neural Control Variates with Automatic Integration

Zilu Li, Guandao Yang, Qingqing Zhao, Xi Deng, Leonidas Guibas, + 2

Article No.: 10, Pages 1–9 • <https://doi.org/10.1145/3641519.3657395>

This paper presents a method to leverage arbitrary neural network architecture for control variates. Control variates are crucial in reducing the variance of Monte Carlo integration, but they hinge on finding a function that both correlates with the ...

0 660 2



RESEARCH-ARTICLE OPEN ACCESS A Differential Monte Carlo Solver For the Poisson Equation

Zihan Yu, Lifan Wu, Zhiqian Zhou, Shuang Zhao

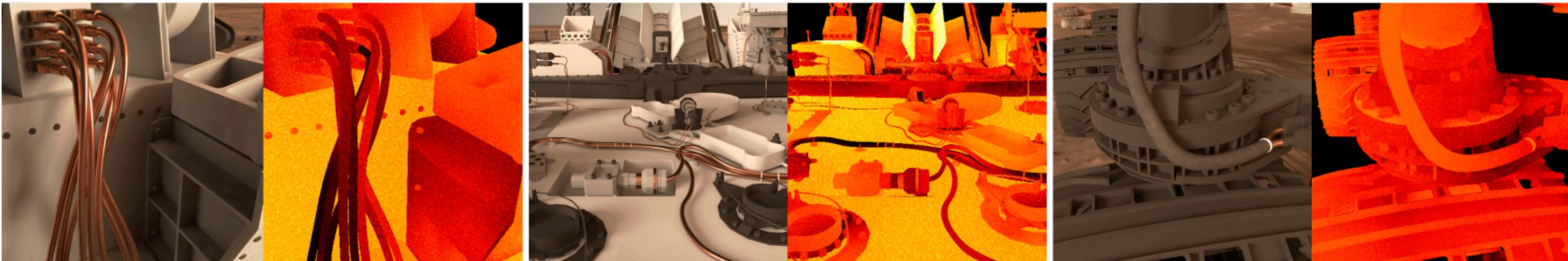
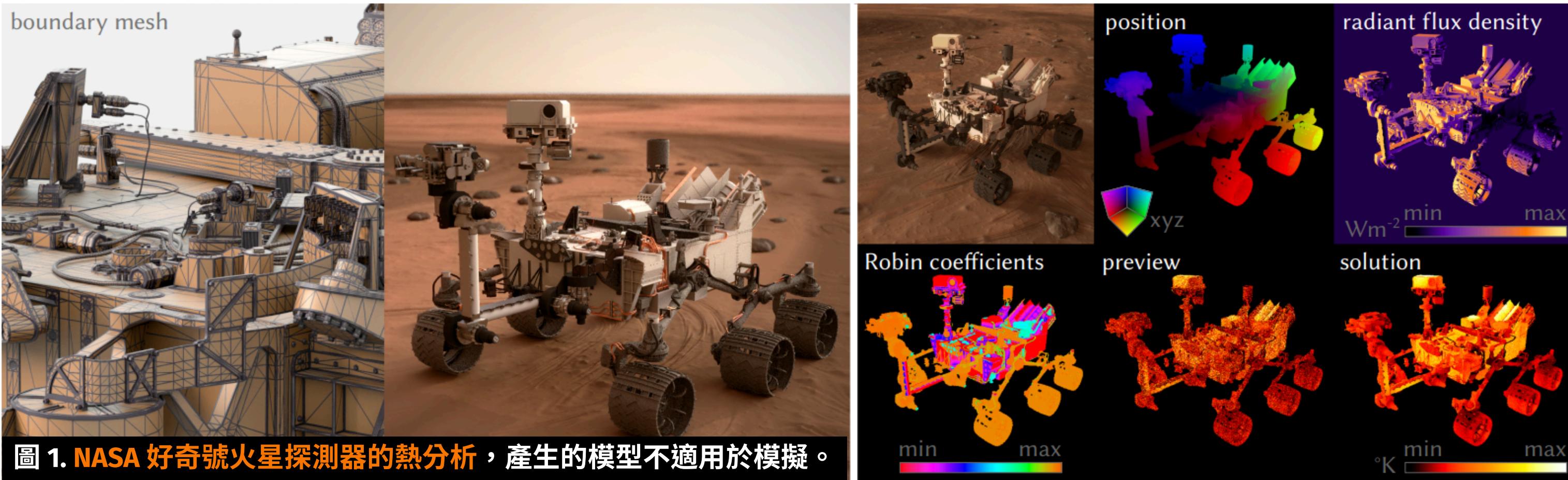
Article No.: 11, Pages 1–10 • <https://doi.org/10.1145/3641519.3657460>

The Poisson equation is an important partial differential equation (PDE) with numerous applications in physics, engineering, and computer graphics. Conventional solutions to the Poisson equation require discretizing the domain or its boundary, which can ...

3 555 1



Motivation



將溫度保持在指定的熱限制內對任務成功至關重要，但複雜的幾何形狀不易透過有限元素模型捕獲，因此熱建模很難整合到設計階段。A grid-free Monte Carlo solver that supports Robin boundary conditions:

- 快速、漸進地計算真實的溫度估計，即使對於極其複雜的幾何形狀也是如此，而無需對域進行體積網格劃分。
- 「延遲著色」方法提供了輸出敏感的評估，僅計算螢幕空間中可見點（右上）的溫度值。
- 可以分析局部感興趣區域的溫度，而無需計算全局解（底）。

研究背景

- 幾何模型日益複雜，**grid-free Monte Carlo methods**在邊界值問題**BVPs** (boundary value problems) 求解的一種有效方式，例如 Walk on Spheres (WoS) [Muller 1956]。近年來在科學計算與圖學領域重新受到關注。
- 不需要對問題域進行體積網格劃分 (volumetric meshing)，甚至不需要對邊界進行高品質網格劃分 (high-quality boundary meshing)。
- 使得 WoS 在近年來被廣泛應用於求解穩態 (steady-state)、時間相關 (time-dependent) 以及弱非線性偏微分方程 (weakly non-linear PDEs)
- 傳統方法 (FEM, FDM) 面臨挑戰：
 - 網格生成困難：高細節、薄特徵幾何易導致計算失敗。
 - 難以並行化：全域求解難以高效並行。
 - 計算資源需求大：高解析度網格使計算成本急劇上升。
- **Walk on Spheres (WoS)** 方法優勢：
 - 無需網格，易並行，適用於複雜幾何。
 - 但僅支持 Dirichlet 邊界條件。
- **Walk on Stars (WoSt)** 方法解決 WoS 限制，可處理 Dirichlet 和 Neumann 邊界，但 **Robin** 邊界仍未被支持。

Introduction

邊界條件與 BVP 求解

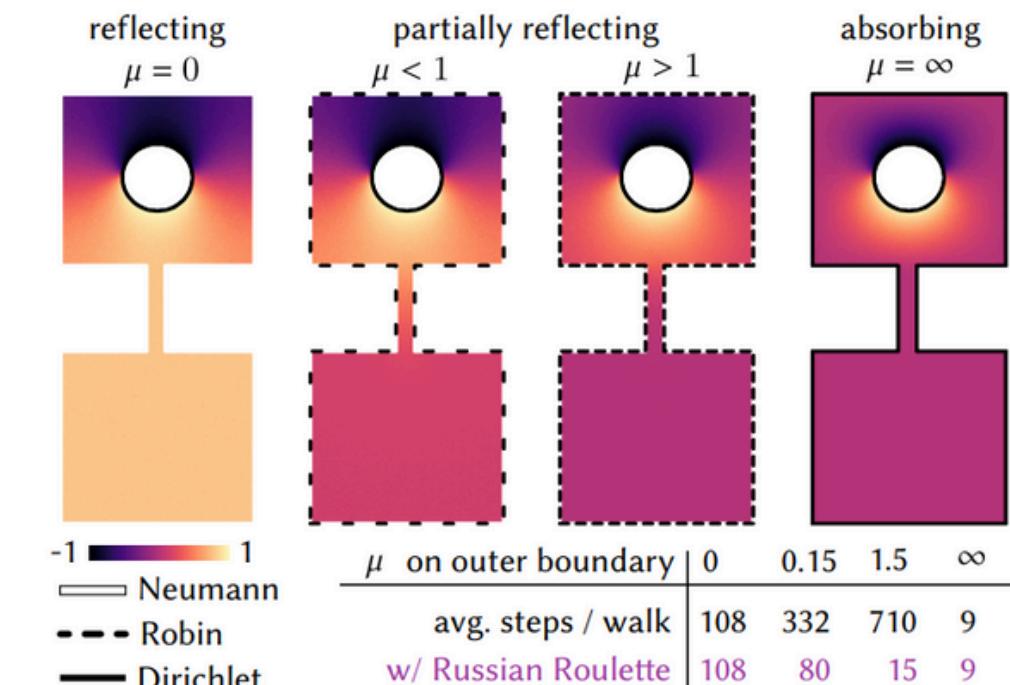
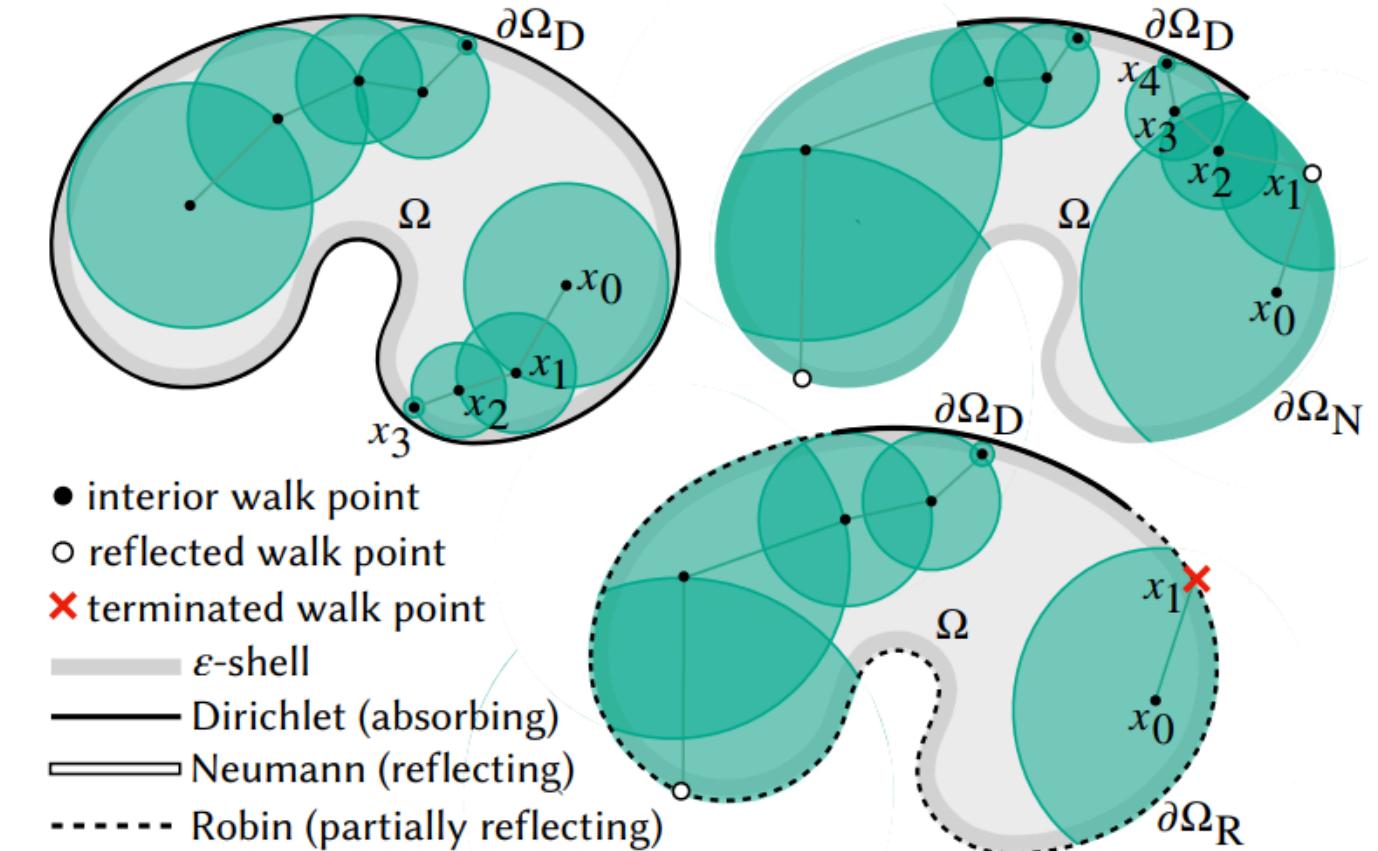
- 邊界條件控制 BVP 解的物理約束 (如溫度、電壓、力、速度)。
- 混合 Dirichlet 與 Neumann 邊界條件將解及其導數約束於邊界。

WoSt 方法概述

- 擴展 WoS，支援混合 Dirichlet 與 Neumann 邊界條件[Sawhney et al. 2023]。
- WoSt 模擬星形區域內隨機遊走 (random walks) 來解 BVP，模擬布朗運動：
 - Dirichlet 邊界：吸收條件。
 - Neumann 邊界：反射條件。
 - Robin 邊界: 部分反射條件。
- 該條件將解及其導數 (分別) 約束在邊界上。
- 將隨機射線方向與當前星形區域 (star-shaped regions) 相交來選擇下一個行走位置。

Robin 邊界條件(部分反射)的重要性

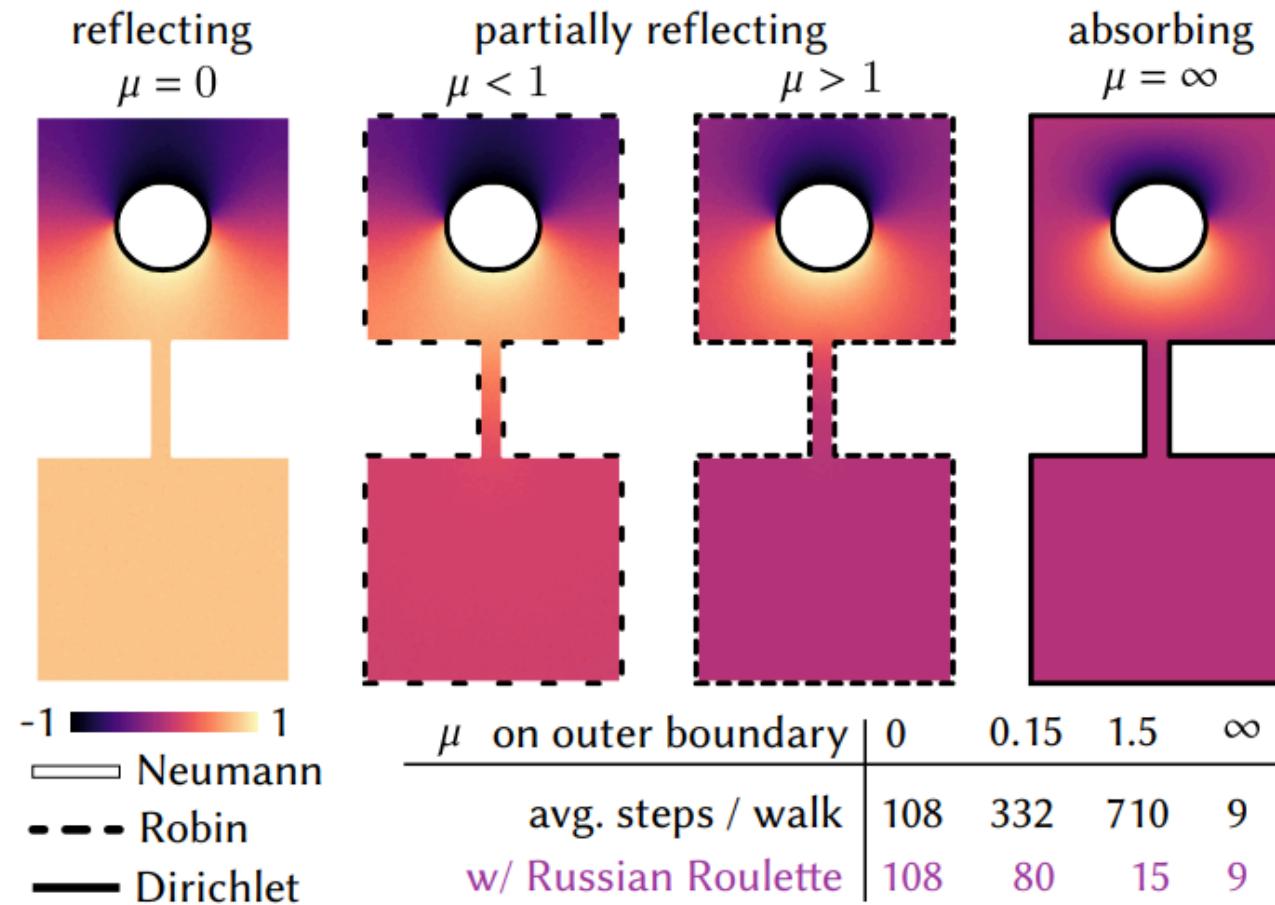
- 從布朗運動的角度來看，Robin 邊界條件同時具有reflecting與absorbing特性（圖 3）。
- Robin 邊界條件提供了更高的物理真實性，大多數真實材料的邊界都是部分吸收(Dirichlet)、部分反射(Neumann) 行為，不僅僅是純吸收 (如黑體) 或純反射 (如理想絕緣體)，廣泛應用於：
 - 熱傳導 (輻射+對流)
 - 電磁學 (電場與磁場混合條件)
 - 流體力學 (部分反射+部分滲透)
- 蒙特卡羅估計量在 Robin 條件下通常比 Neumann 條件下更有效率 (第 4 節)：
 - 模擬 partially reflecting Brownian motion可以直接在 Robin 邊界被吸收 [Grebennikov 2006]。
 - 相比之下，Neumann 反射必須進行多次遊走才能到達 Dirichlet 邊界，導致計算時間變長。



Introduction

主要貢獻

- 引入反射函數 (reflectance function, Equation 7) 依賴於 Robin 係數 μ , 作為遊走過程中的乘法權重，確保估計方差受控。



$$\rho_\mu(x, z) \doteq \begin{cases} 1 - \mu(z) \frac{G^B(x, z)}{P^B(x, z)}, & \text{on } \partial S_{tR}, \\ 1, & \text{on } \partial S_{tB}. \end{cases} \quad (7)$$

我們將 WoSt 方法擴展為可求解以下形式的 BVP :

$$\begin{aligned} \Delta u(x) &= f(x) && \text{on } \Omega, \\ u(x) &= g(x) && \text{on } \partial\Omega_D, \\ \frac{\partial u(x)}{\partial n_x} - \mu(x)u(x) &= h(x) && \text{on } \partial\Omega_R, \end{aligned} \quad (1)$$

其中：

- Ω 是定義域，其邊界 $\partial\Omega$ 被劃分為 Dirichlet 邊界 $\partial\Omega_D$ 和 Robin 邊界 $\partial\Omega_R$ 。
- $\mu(x) \geq 0$ 是 Robin 係數，可變化於 $\partial\Omega_R$ 。
- 當 $\mu = 0$ 時，恢復 Neumann 邊界條件。
- 當 $\mu \rightarrow \infty$ 時，恢復 Dirichlet 邊界條件。

- 隨機遊走技術改進：
 - 限制反射函數在 0 到 1 間，使遊走步長不會無限增長，提高估計精度。
 - 動態調整星形區域步長，保持估計方差有界，提升解的穩定性。
 - 使用 **Russian roulette** 技術 [Pharr et al. 2016]，對 Robin 邊界上的遊走進行 **probabilistic termination** (圖 2、3)，提高計算效率。
- 比傳統無網格方法更精確：
 - WoSt 在非凸域中比 Walk on Boundary (WoB) 方法誤差小數個數量級。
 - 估計量保證蒙特卡羅收斂性 (Monte Carlo convergence)，適用於任何 Dirichlet、Neumann 和 Robin 邊界條件的組合 (圖 11、12)。

實作與優化

- **Monte Carlo積分估計**：利用隨機遊走模擬布朗運動來解 PDE。
- **空間加速結構**：
 - 使用空間化法向圓錐層次結構 (Spatialized Normal Cone Hierarchy, **SNCH**) 高效確定區域大小，並僅訪問 BVP 邊界上一小部分的幾何元素 [Johnson and Cohen 2001]。
 - 結合方差降低技術雙向求解 (**Bidirectional Formulation**) [Qi et al. 2022]。
 - 邊界值快取 (Boundary Value Caching, **BVC**) [Miller et al. 2023]，顯著降低解的方差。
- **無網格求解**：
 - 避免 FEM/FDM 網格劃分，無需建網格，即可高效求解複雜幾何問題(如 NASA 好奇號熱分析)，保留了無網格方法的優勢：
 - **geometric robustness** :可針對局部區域計算，避免全域求解
 - **scalability**: 無需額外的幾何預處理
 - **output sensitivity**
 - **parallelism**: 適合 GPU/TPU，對高細節幾何具有次線性擴展性。
- 成為解決高細節幾何結構的 **PDE** 問題的一種高效工具，在科學計算與工程應用中具備廣泛潛力。

Related Work

1. Grid-based PDE Solvers

- **有限元素法 (FEM)** :

- 需要高品質網格，處理高細節、複雜幾何時極具挑戰性。
- 網格生成是主要瓶頸，可能因**自交錯 (self-intersections)** 失敗或耗盡內存。
- Meshless FEM需要對整個域進行採樣，若採樣不足會導致收斂停滯或完全失敗。

- **邊界元素法 (BEM)** :

- 只需對邊界進行離散化，但計算中需要求解**dense matrix**，存儲和計算成本隨幾何細節的增長呈二次增長 $O(n^2)$ 。
- 需要特殊的矩陣近似技術來處理大規模幾何結構。

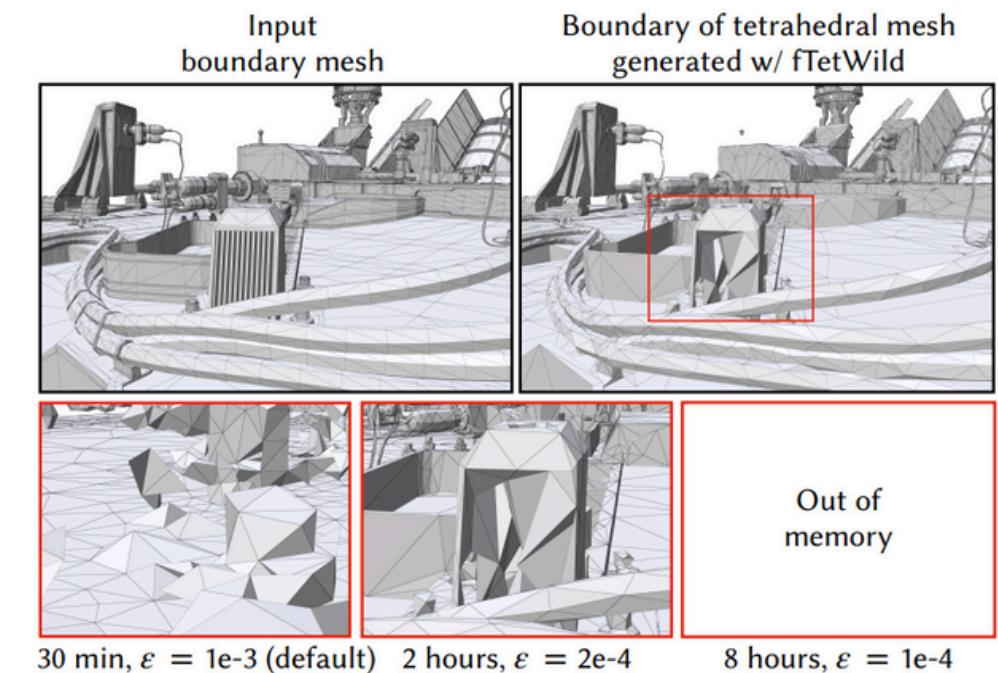
- **有限差分法 (FDM)** :

- 需要規則網格或八叉樹 (**octree**)，但當解析度提高時，計算和內存需求呈立方增長 $O(n^3)$ 。
- 邊界條件的處理困難，特別是在非對齊 (**non-aligned**) 幾何上。

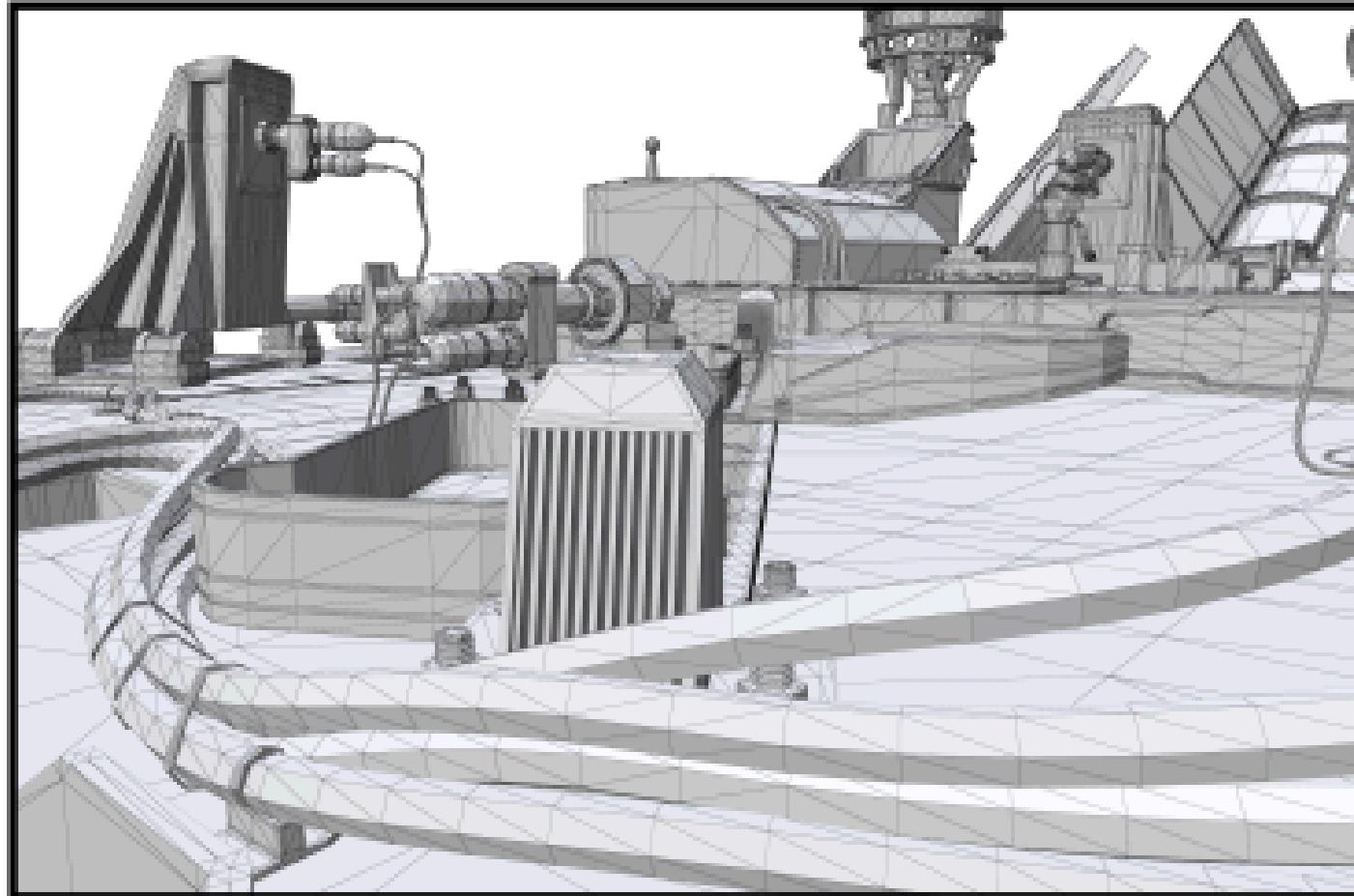
- **基於物理的深度學習 (physics-informed neural networks, PINNs)** :

- 無需網格，但訓練成本高，邊界條件難處理。
- 需要高成本的**hyperparameter tuning**，並且難以保證數值穩定性。

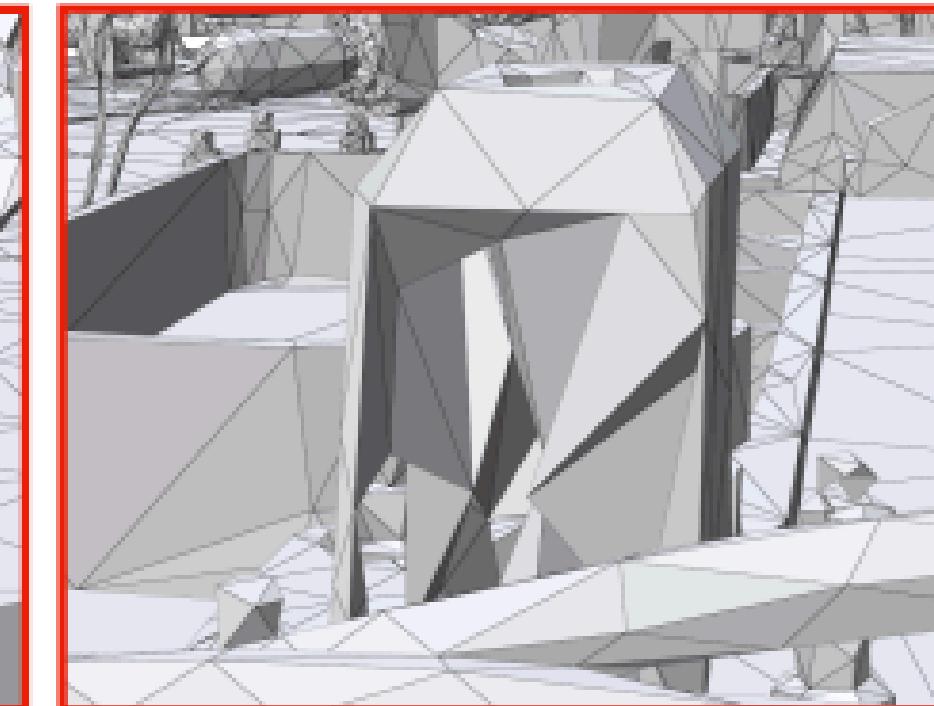
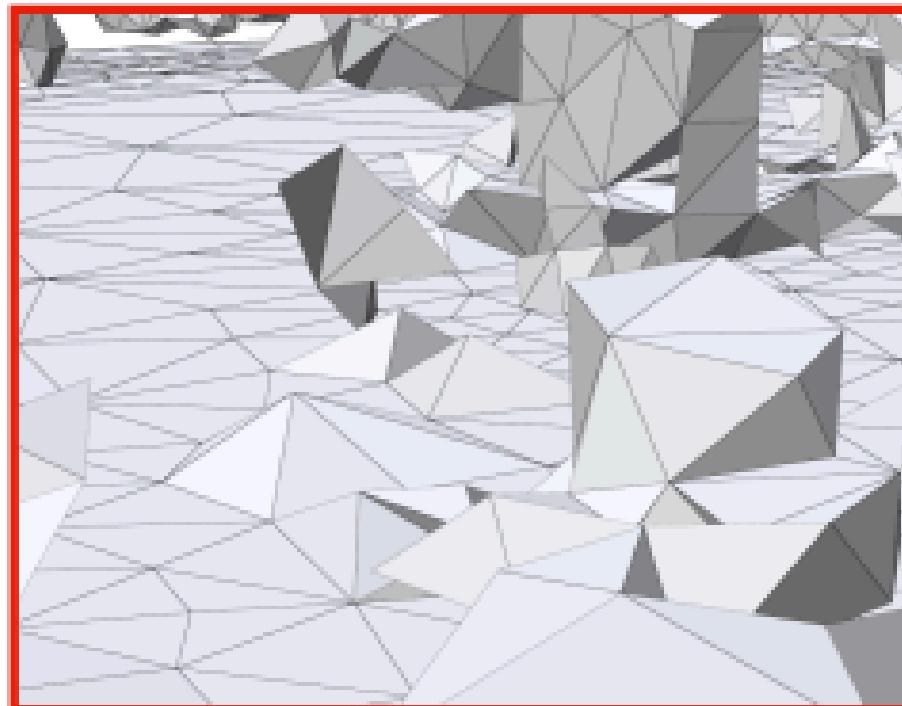
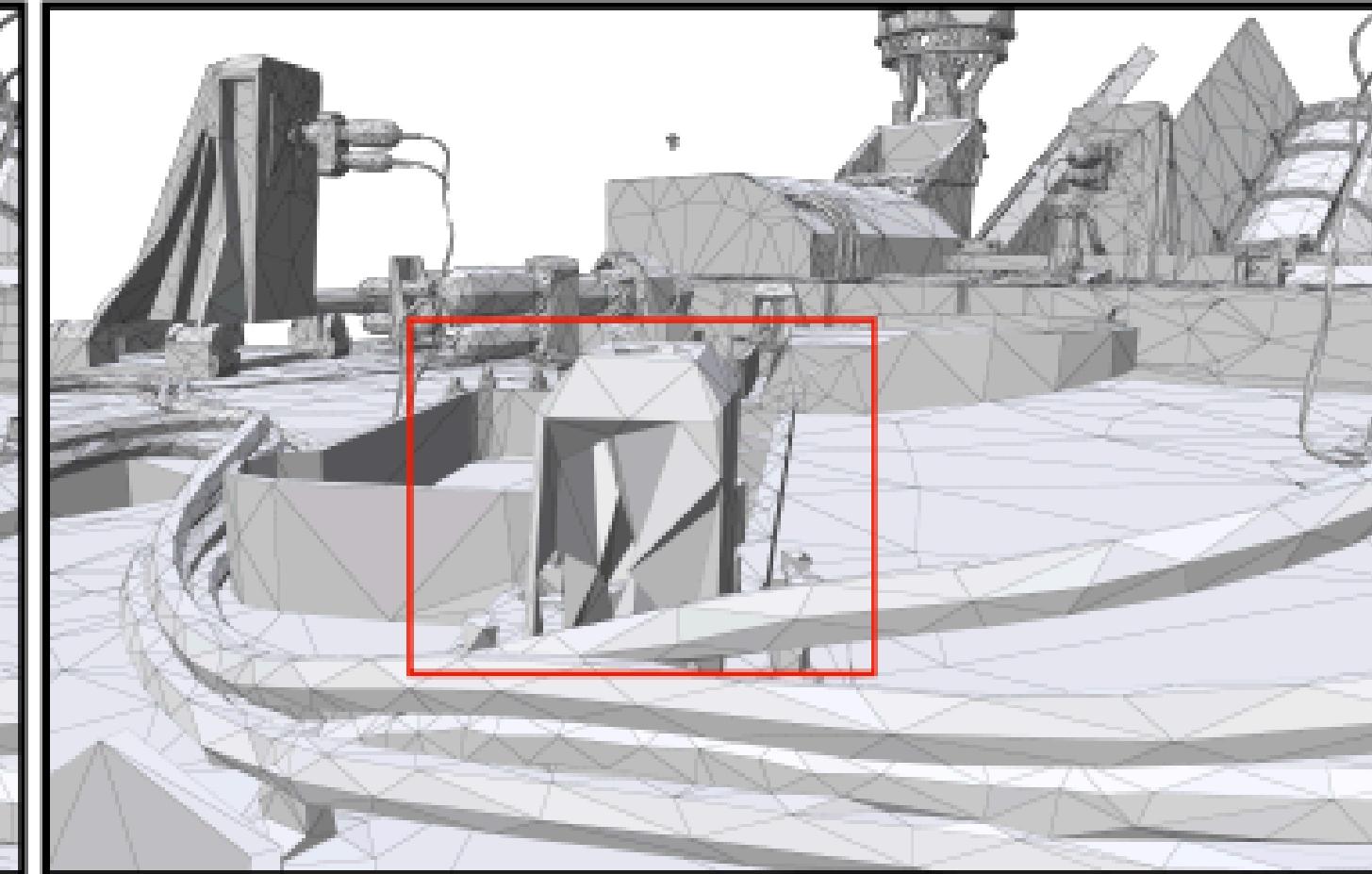
對比而言，Grid-Free Monte Carlo methods 無需全域求解，能在關鍵區域獨立計算解，避免FEM/FDM 的全域離散化開銷。



Input
boundary mesh



Boundary of tetrahedral mesh
generated w/ fTetWild



Out of
memory

30 min, $\epsilon = 1e-3$ (default)

2 hours, $\epsilon = 2e-4$

8 hours, $\epsilon = 1e-4$

Related Work

2. Grid-free Monte Carlo Methods

這類方法基於隨機遊走來模擬 布朗運動 (Brownian motion)，適用於求解橢圓型 PDEs (如 Poisson 方程)

優勢：

- 無需全域離散化，可局部求解。
- 可獨立並行計算，適合 GPU 加速。
- 避免幾何混疊 (aliasing)，適用於高細節幾何。

2.1. Discretized Random Walks

- 模擬隨機微分方程 (SDEs)，需平衡步長大小以控制速度與誤差。
- 需要在時間步長上做權衡：
 - 步長太小 → 運行速度慢。
 - 步長太大 → 產生高偏差 (high bias)。

2.2. Continuous Random Walks

- **WoS (Walk on Spheres) :**
 - 使用閉式分佈 (closed-form distributions) 來確保大步長的準確性，避免時間離散化誤差。比離散化方法更高效，因為不需要小步長來提高精度。
 - 改進：Optimized Implementations、Variance Reduction Techniques、擴展至更一般 BVPs。
- **WoSt (Walk on Stars) :**
 - 由 Sawhney et al. (2023) 提出，star-shaped regions 擴展 WoS，支援 Dirichlet 與 Neumann 邊界條件。
 - 引入 Reflectance Function 處理部分吸收/部分反射邊界條件。
 - SNCH (Spatialized Normal Cone Hierarchy) 加速計算。

2.3. WoB (Walk on Boundary)

一種替代的無網格方法，使用隨機遊走來在邊界上進行評估：

- 遞歸估算單層與雙層勢 (single & double layer potentials)。
- 非凸域中誤差極大，因為在非凸域中，遊走貢獻會隨步長增長導致無界方差 (unbounded variance)。
- Robin 邊界條件進一步加劇這些問題，使WoB 方法在這類問題上效果不佳。

3. 總結

傳統方法 FEM, BEM, FDM 依賴網格計算成本高，難處理複雜幾何。
無網格方法 (Monte Carlo) 逐步發展：

- WoS 適用 Dirichlet，但無法處理 Neumann/Robin。
- **WoSt (本論文方法)**：擴展至 Robin 邊界條件，具有更好的數值穩定性與收斂性 (Monte Carlo rate of convergence)。應用 Russian Roulette，實現無偏低方差的早期終止，比 WoB 方法更高效。
- WoB：非凸域誤差極高，不適合 Robin 邊界問題。

ALGORITHM 1: WALKONSTARS(x , n_x , ε)

Note: Code annotated with comments in green indicates our modifications to the WALKONSTARS algorithm by Sawhney et al. [2023].

Input: Starting position $x \in \Omega$ of random walk, normal n_x at x (undefined if $x \notin \partial\Omega_R$), ε -shell parameter.

Output: Single-sample Monte Carlo estimate $\hat{u}(x)$ for Equation 1.

```
1:  $d, \bar{x} \leftarrow \text{DISTANCEABSORBINGBOUNDARY}(x)$                                 ► Compute distance to absorbing boundary  $\partial\Omega_D$  with Dirichlet conditions ( $\infty$  if  $\partial\Omega_D = \emptyset$ )
2: if  $d < \varepsilon$  then return  $g(\bar{x})$                                          ► Return boundary value  $g$  at closest pt  $\bar{x}$  if  $x \in \partial\Omega_D^\varepsilon$ 
3:  $R \leftarrow \text{STARRADIUSREFLECTINGBOUNDARY}(x, d)$                                 ► Compute radius of star region  $St(x, R)$  containing reflecting boundary  $\partial\Omega_R$ , such that  $R \leq d$  (Alg. 2)
4:  $R \leftarrow \max(\varepsilon, R)$                                                        ► Also ensure  $R \geq \varepsilon$ 
5:  $v \leftarrow \text{SAMPLEUNITSPHERE}()$                                               ► Sample a direction  $v$  uniformly on the unit sphere
6: if  $x \in \partial\Omega_R$  and  $n_x \cdot v > 0$  then  $v \leftarrow -v$                                 ► If  $x$  lies on  $\partial\Omega_R$ , ensure  $v$  is sampled on hemisphere with axis  $-n_x$ 
7:  $\text{hit}, p, n_p \leftarrow \text{INTERSECTREFLECTINGBOUNDARY}(x, v, R)$                 ► Intersect  $\partial St_R$  (boundary inside star region) with ray  $x + Rv$ , and get first hit
8: if not hit then  $p \leftarrow x + R v$                                          ► If there is no hit with  $\partial St_R$ , intersect  $\partial St_B$  (spherical portion of star region) instead
9:  $\hat{I}_h \leftarrow \text{REFLECTINGBOUNDARYESTIMATE}(x, R)$                                 ► Estimate contribution from boundary term  $h$  on  $\partial St_R$  [Sawhney et al. 2023, Alg. 1, lines 18-22]
10:  $\hat{I}_f \leftarrow \text{SOURCEESTIMATE}(x, p, v, R)$                                          ► Estimate contribution from source term  $f$  in  $St$  [Sawhney et al. 2023, Alg. 1, lines 24-26]
11:  $\rho_\mu \leftarrow \text{CLAMP}(1 - \mu(p)G^{B(x,R)}(x, p)/P^{B(x,R)}(x, p), 0, 1)$       ► Compute reflectance (Eq. 7) and clamp it to [0, 1]
12: if  $\rho_\mu < \text{SAMPLEUNIFORM}(0, 1)$  then return  $-\hat{I}_h + \hat{I}_f$           ► Attempt to terminate walk using Russian roulette
13: return WALKONSTARS( $p, n_p, \varepsilon$ )  $- \hat{I}_h + \hat{I}_f$                                 ► Repeat procedure from updated walk position ( $n_p$  is undefined if  $x \notin \partial\Omega_R$ )
```

Background

BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過**邊界積分方程 (BIE)** 來求解帶有 Robin 邊界條件的橢圓型PDEs。

1. 邊界積分方程 (Boundary Integral Equation, BIE)

對於一個定義在區域 Ω 內的 Poisson 方程：

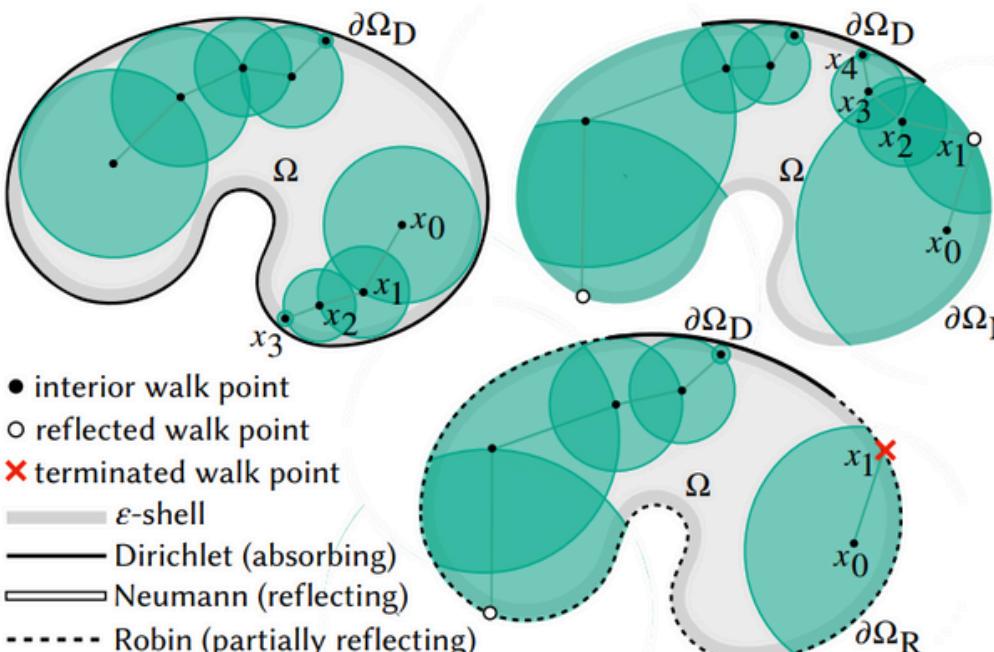
$$\Delta u(x) = f(x), \quad x \in \Omega$$

其邊界 $\partial\Omega$ 可以分為：

- Dirichlet 邊界 : $u(x) = g(x), \quad x \in \partial\Omega_D$
- Neumann 邊界 : $\frac{\partial u(x)}{\partial n_x} = h(x), \quad x \in \partial\Omega_N$
- Robin 邊界 (結合 Dirichlet 和 Neumann) :

$$\frac{\partial u(x)}{\partial n_x} - \mu(x)u(x) = h(x), \quad x \in \partial\Omega_R$$

其中 $\mu(x)$ 為 Robin 係數，決定邊界對隨機遊走的吸收或反射程度。



根據**邊界積分方程 (BIE)** 理論，Poisson 方程的解可以表示為：

我們假設 Ω 是一個封閉 (watertight) 的區域，並具有光滑的邊界 $\partial\Omega$ ，其中 A 和 C 是 Ω 與 \mathbb{R}^N 的任意子集。

對於任意點 $x \in \mathbb{R}^N$ ，Poisson 方程的解滿足以下條件 [Costabel 1987; Hunter and Pullan 2001]：

$$\begin{aligned} \alpha(x) u(x) &= \int_{\partial A} P^C(x, z) u(z) - G^C(x, z) \frac{\partial u(z)}{\partial n_z} dz \\ &+ \int_A G^C(x, y) f(y) dy, \end{aligned} \tag{2}$$

其中：

- $\alpha(x) = 1$ 若 $x \in A$ · $\alpha(x) = 1/2$ 若 $x \in \partial A$ · $\alpha(x) = 0$ 否則。
- $G^C(x, z)$ 是 Poisson 方程的 Green's function (格林函數)。
- $P^C(x, z) = \frac{\partial G^C}{\partial n}$ 是 Poisson kernel (Poisson 核)。
- Poisson 方程的 Green's function 和 Poisson kernel 在一般情況下沒有閉式解 (closed-form solution)，但對於某些特例，如 自由空間 ($C = \mathbb{R}^N$) 或 球形域 ($C = B$)，可以找到顯式表達式 [Sawhney et al. 2023, 附錄 A.1]。

Background

BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過**邊界積分方程 (BIE)** 來求解帶有 Robin 邊界條件的橢圓型PDEs。

1. 邊界積分方程 (Boundary Integral Equation, BIE)

對於一個定義在區域 Ω 內的 Poisson 方程：

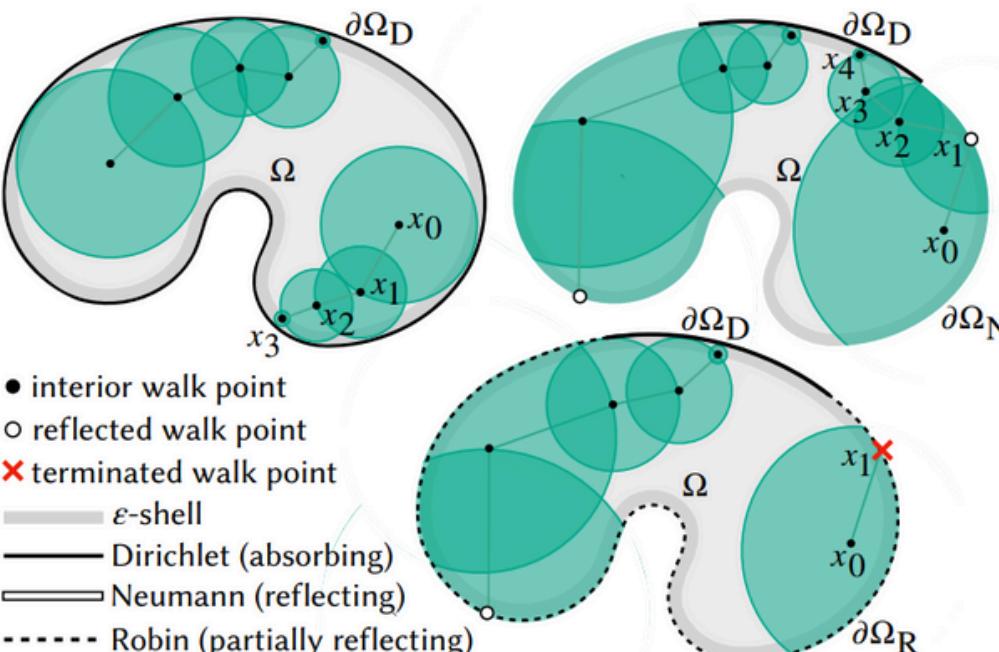
$$\Delta u(x) = f(x), \quad x \in \Omega$$

其邊界 $\partial\Omega$ 可以分為：

- Dirichlet 邊界 : $u(x) = g(x), \quad x \in \partial\Omega_D$
- Neumann 邊界 : $\frac{\partial u(x)}{\partial n_x} = h(x), \quad x \in \partial\Omega_N$
- Robin 邊界 (結合 Dirichlet 和 Neumann) :

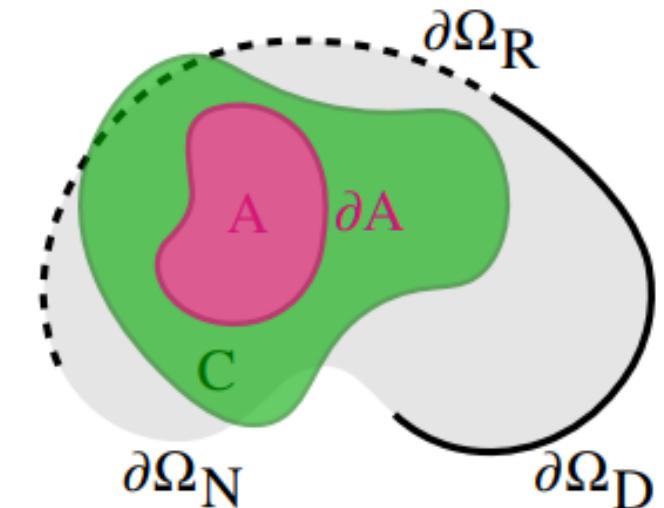
$$\frac{\partial u(x)}{\partial n_x} - \mu(x)u(x) = h(x), \quad x \in \partial\Omega_R$$

其中 $\mu(x)$ 為 Robin 條數，決定邊界對隨機遊走的吸收或反射程度。



根據**邊界積分方程 (BIE)** 理論，Poisson 方程的解可以表示為：

$$\begin{aligned} \alpha(x) u(x) = & \int_{\partial A} P^C(x, z) u(z) - G^C(x, z) \frac{\partial u(z)}{\partial n_z} dz \\ & + \int_A G^C(x, y) f(y) dy, \end{aligned} \quad (2)$$



- 我們需要確定邊界 ∂A 上的解 $u(z)$ 以及其法向導數 $\frac{\partial u}{\partial n_z}(z)$ (如圖中的內嵌圖示)。

- 然而，我們只能透過邊界條件來獲取這些信息：

- Dirichlet 邊界條件：在 $\partial\Omega_D$ 上，解 u 由 g 覆蓋，即 $u = g$ 。
- Neumann 邊界條件：在 $\partial\Omega_N$ 上，解的法向導數 $\frac{\partial u}{\partial n}$ 由 h 覆蓋，即 $\frac{\partial u}{\partial n} = h$ 。
- Robin 邊界條件：在 $\partial\Omega_R$ 上，解滿足：

$$\frac{\partial u}{\partial n} - \mu u = h$$

這是一個 Dirichlet 和 Neumann 邊界條件的線性組合。

- 源項 f 在區域 Ω 內 (即 A) 已經給定，但一般來說， u 和 $\frac{\partial u}{\partial n}$ 在 ∂A 上是未知的，需要使用數值方法 (如 WoSt 方法) 來估計 (見第 3.2 節)。

Background

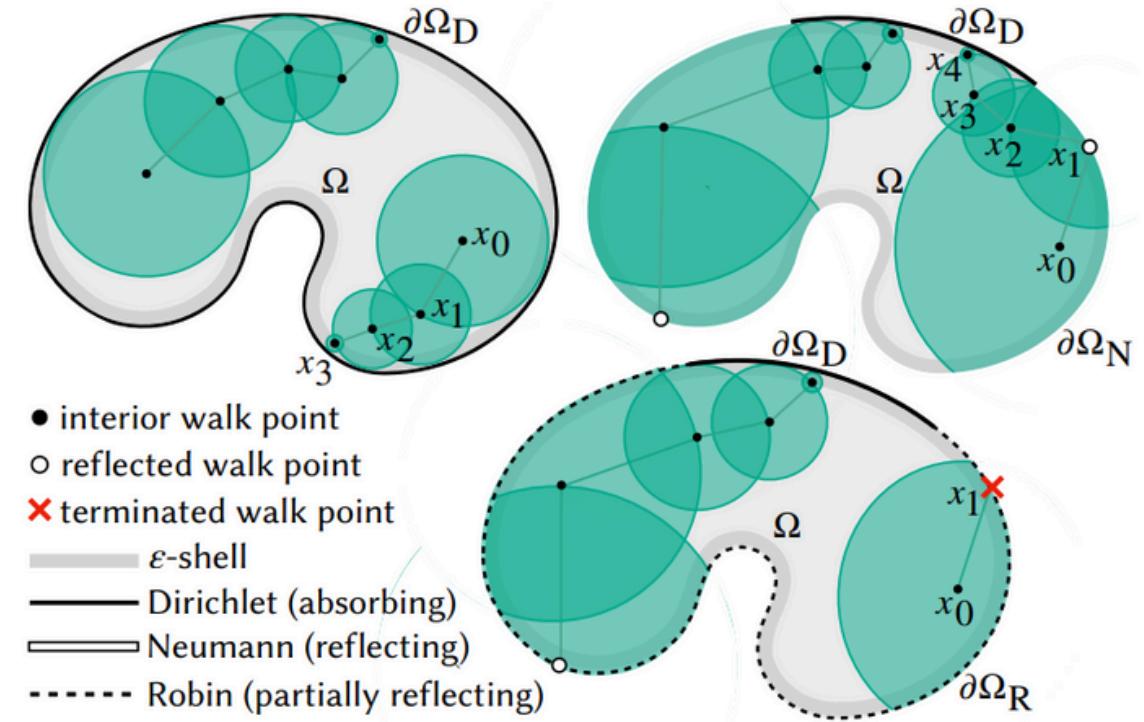
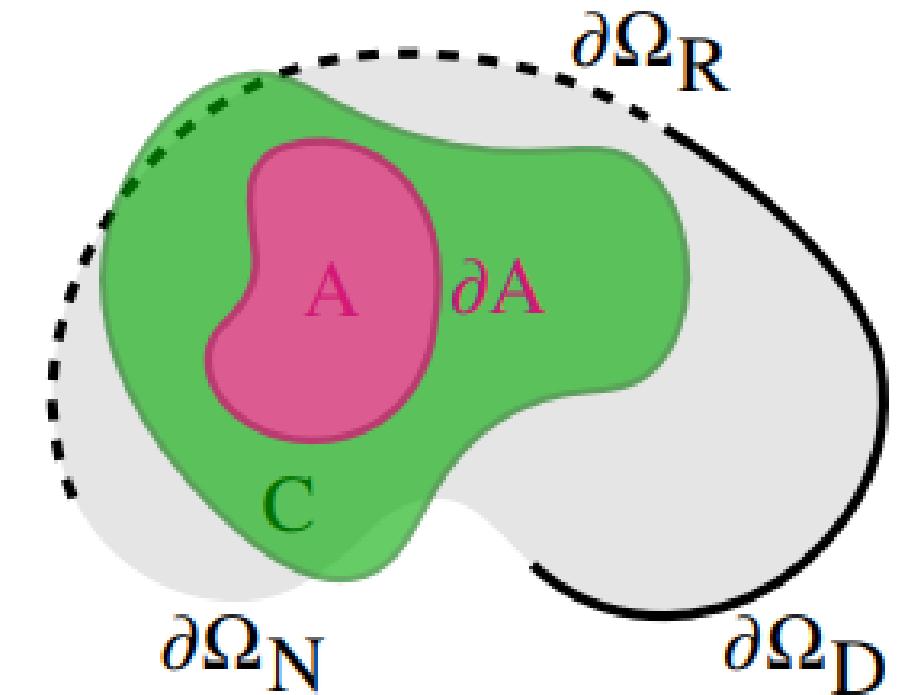
BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過邊界積分方程 (BIE) 來求解帶有 Robin 邊界條件的橢圓型PDEs。

2. Walk on Stars (WoSt) 方法

WoSt 方法是基於隨機遊走的數值方法，透過模擬 布朗運動 (Brownian motion) 來求解 PDEs。它的基本步驟如下：

1. 在內部隨機選擇一個點 x_k ，並確定其所在的星形區域 (star-shaped region) $St(x_k, R)$ 。
2. 隨機選擇下一個步驟 x_{k+1}
 - 如果碰到 Dirichlet 邊界，則直接終止並使用邊界值 $g(x)$ 。
 - 如果碰到 Neumann 邊界，則反射並繼續遊走。
 - 如果碰到 Robin 邊界，則根據 Robin 係數 $\mu(x)$ 決定是反射還是吸收。
3. 重複步驟 2，直到遊走終止。

相比於 Walk on Spheres (WoS) 方法，WoSt 允許在 非球形區域 內進行隨機遊走，從而支持 Neumann 和 Robin 邊界條件。



Background

BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過**邊界積分方程 (BIE)** 來求解帶有 Robin 邊界條件的橢圓型PDEs。

3. WoSt 方法如何處理 Robin 邊界條件

對於 Robin 邊界條件：

$$\frac{\partial u}{\partial n} - \mu u = h$$

WoSt 方法需要對 星形區域的大小 R 進行調整，以確保估計值的方差受控。具體來說：

- 定義 反射函數 (Reflectance Function) :

$$\rho_\mu(x, z) = 1 - \mu(z) \frac{G_B(x, z)}{P_B(x, z)}$$
$$\rho_\mu(x, z) \doteq \begin{cases} 1 - \mu(z) \frac{G^B(x, z)}{P^B(x, z)}, & \text{on } \partial S_R, \\ 1, & \text{on } \partial S_B. \end{cases} \quad (7)$$

如果 ρ_μ 落在區間 $[0,1]$ ，則 WoSt 遊走過程是穩定的，否則解可能發散。

- 根據 ρ_μ 調整星形區域的大小 R ，確保 隨機遊走不會無界增長。
- 使用 Russian Roulette 技術，讓部分遊走過程提前終止，提高計算效率。

Background

BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過**邊界積分方程 (BIE)** 來求解帶有 Robin 邊界條件的橢圓型PDEs。

3.2.1 星形區域 (Star-Shaped Regions)

- 原始的 **Walk on Spheres (WoS)** 方法 [Muller, 1956] 針對純 Dirichlet 問題，通過隨機遊走於完全包含於 Ω 的球體來求解 BVP (左上，圖 2)。
- **WoSt 方法** 則改進了這一點，使用 **星形區域 (star-shaped regions)** 來進行隨機遊走，確保從點 $x \in \Omega$ 能夠看到其邊界。
- WoSt 通過將球體 $B(x, R)$ (半徑 R 為 x 到 Dirichlet 邊界最近點的距離) 與 **最近的 Neumann 邊界剪影點** 相交來形成這些區域 (右圖，圖 5)。
- 因此， B 可以包含 Ω_N 的部分，但不包含 Ω_D 。

定義：

- $\partial St_N : \Omega_N \cap \partial St$ (星形區域 St 的 Neumann 邊界部分)，具有 **已知導數條件** $\partial u / \partial n = h$ (由方程 1 紿定)。
- $\partial St_B : \partial B \cap \partial St$ (星形區域的球形部分)。

給定 $A = St$ 和 $C = B$ ，則方程 2 變為：

$$\alpha(x) u(x) = \int_{\partial St(x,R)} P^B(x,z) u(z) dz - \int_{\partial St_N(x,R)} G^B(x,z) h(z) dz + \int_{St(x,R)} G^B(x,y) f(y) dy. \quad (3)$$

關鍵簡化：與方程 2 相比，這裡的 $u(z)$ 是方程中唯一的未知量，且 **Neumann 邊界 ∂St_N 上的導數 $\partial u / \partial n(z)$ 並不需要在 ∂St_B 上使用，因為 $G^B(x,z) = 0$ 。

Background

BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過**邊界積分方程 (BIE)** 來求解帶有 Robin 邊界條件的橢圓型PDEs。

3.2.2 蒙特卡羅估計器 (Monte Carlo Estimator)

- 蒙特卡羅方法 透過數值積分近似 $I := \int_A \phi(x) dx :$

$$\hat{I} \doteq \frac{1}{N} \sum_{n=1}^N \frac{\phi(x^n)}{p^A(x^n)}, \quad x^n \sim p^A, \quad (4)$$

- 其中 x^n 是獨立隨機樣本，來自非零機率密度 p^A 。
- 當 $N = 1$ 時，得到 WoSt 估計器：

WALK ON STARS WITH DIRICHLET-NEUMANN CONDITIONS

$$\begin{aligned} \widehat{u}(x_k) &= \frac{P^B(x_k, x_{k+1}) \widehat{u}(x_{k+1})}{\alpha(x_k) p^{\partial St(x_k, R)}(x_{k+1})} \\ &\quad - \frac{G^B(x_k, z_{k+1}) h(z_{k+1})}{\alpha(x_k) p^{\partial St_N(x_k, R)}(z_{k+1})} + \frac{G^B(x_k, y_{k+1}) f(y_{k+1})}{\alpha(x_k) p^{St(x_k, R)}(y_{k+1})}. \end{aligned} \quad (5)$$

該估計器為遞歸形式，因為 \widehat{u} 出現在方程 5 的兩側，從而產生了一個 隨機遊走 (random walk)，即「星上行走」(Walk on Stars) 的名稱來源。

- 在每一步 k ，WoSt 從概率密度 p^{St}, p^{St_N}, p^{St} 中進行採樣：
 - x_{k+1} 位於 ∂St 。
 - z_{k+1} 位於 ∂St_N 。
 - y_{k+1} 位於 St 。
- 若問題的 Dirichlet 邊界 Ω_D 被達到，則 WoSt 在 ϵ -shell Ω_D^ϵ 內終止。

Background

BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過**邊界積分方程 (BIE)** 來求解帶有 Robin 邊界條件的橢圓型PDEs。

3.2.3 採樣星形區域 (Sampling Star-Shaped Regions)

- WoSt 選擇下一步的遊走位置 x_{k+1} 透過 **重要性抽樣 (importance sampling)** Poisson 核 $P^B(x_k, x_{k+1})$ 。
- 這與 **蒙特卡羅光線追蹤 (Monte Carlo ray tracing)** 相似：
 - **方向採樣 (direction sampling)**：從 x_k 以均勻分佈的方向投射射線，選擇其與 ∂St 的交點作為 x_{k+1} 。
 - **半球方向採樣 (hemispherical direction sampling)**：若 x_k 在 Ω_N ，則從半球方向抽樣，以避免乘積因子增加。
 - 此機制確保 **遊走通量 (walk throughput)** 恒為 1，即：

$$p^{St}(x_k, x_{k+1}) = P^B(x_k, x_{k+1})/\alpha(x_k).$$

3.2.4 Epsilon Clamp

- WoSt 限制 $St(x_k, R)$ 的半徑 R 不得小於某個用戶定義的 ϵ ，以確保：
 - 遊走不會卡在 **Neumann 邊界的凹陷處**（當最近剪影點距離趨近於 0 時）。
 - 這與 **ϵ -shell Ω_D^ϵ** 類似，會引入一定的偏差，但能提高穩定性。
 - 對於 **Robin 邊界條件**，我們也將應用 **ϵ -clamp**，並使用方向抽樣來確定下一步 x_{k+1} 。

Background

BVPs的數學基礎，以及 Walk on Stars (WoSt) 如何透過**邊界積分方程 (BIE)** 來求解帶有 Robin 邊界條件的橢圓型PDEs。

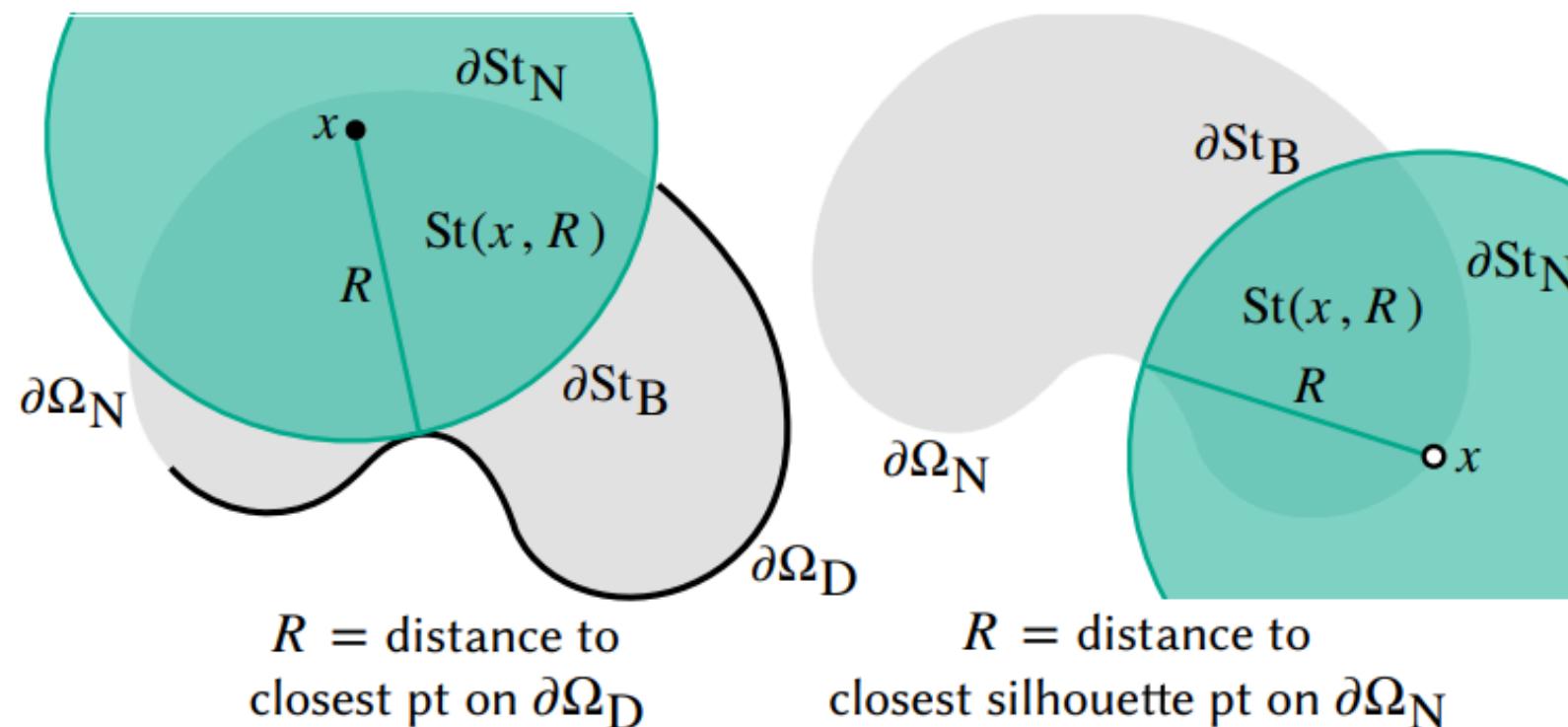


圖 5：WoSt 如何在 Dirichlet-Neumann 邊界條件下定義隨機遊走

- 左圖：在 Dirichlet 邊界條件下， R 為最近 Dirichlet 邊界點的距離。
- 右圖：在 Neumann 邊界條件下， R 為最近 Neumann 的距離。

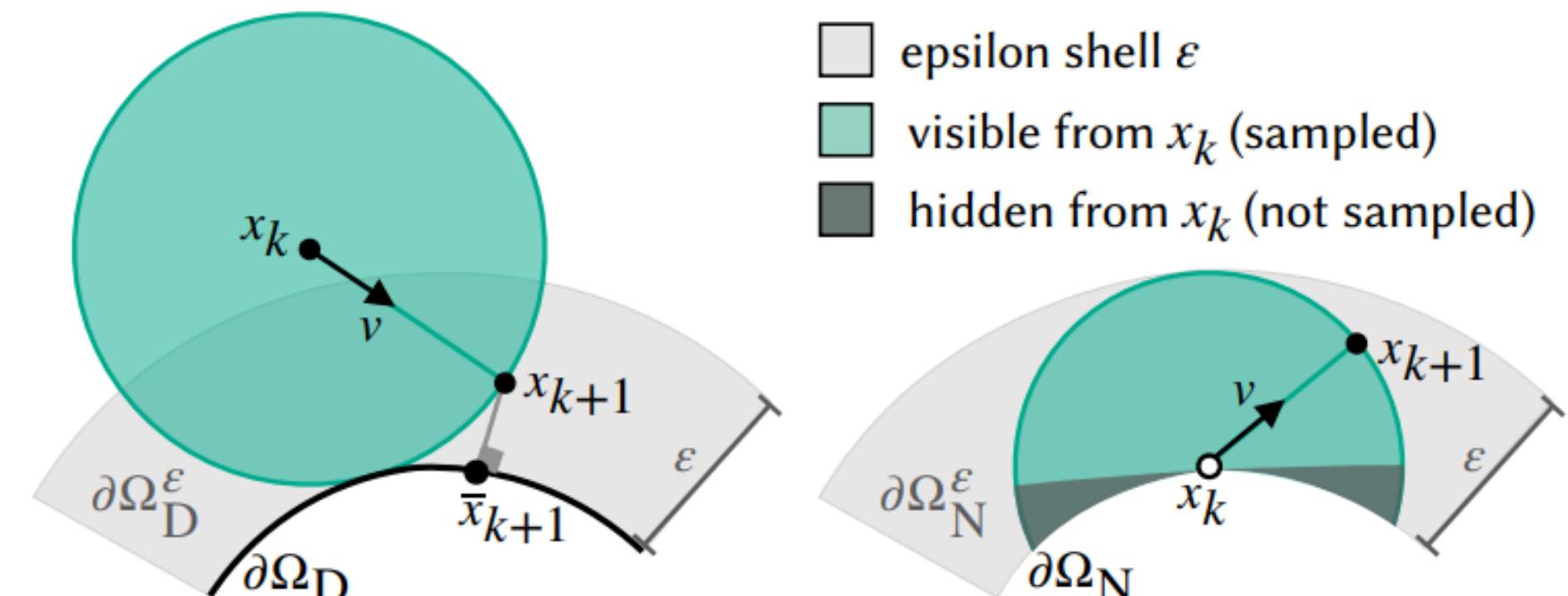


圖 6：WoSt 如何在 ϵ -shell 中終止

- 左圖：在 Dirichlet 邊界條件下，遊走會在 ϵ -shell 內終止，使用最近投影點的 Dirichlet 數據。
- 右圖：在 Neumann 邊界條件下，若最近剪影點距離趨近 0，則遊走僅在可見部分採樣，隱含假設 BVP 解為 0。

為了推導支持 Robin 邊界條件的邊界積分方程 (BIE)，我們主要遵循方程 3 的推導過程 [Sawhney et al. 2023, 第 4.2 節]。

關鍵改進

- 我們引入 Brakhage-Werner 技巧（來自勢能理論 [Nédélec 2001]）。
- Robin 邊界條件透過 $\partial u / \partial n = \mu \cdot u + h$ 替代 Neumann 條件（見方程 1）。
- 定義新的邊界區域：
 - $\partial St_R : \partial \Omega_R \cap \partial St$ (Robin 邊界部分)。
 - $\partial St_B : \partial B \cap \partial St$ (球形邊界部分)。

修正後的 BIE

$$\alpha(x)u(x) = \int_{\partial St(x,R)} \rho_\mu(x,z) P^B(x,z) u(z) dz - \int_{\partial St_R(x,R)} G^B(x,z) h(z) dz + \int_{St(x,R)} G^B(x,y) f(y) dy. \quad (6)$$

其中 $\rho_\mu(x,z)$ 定義如下：

$$\rho_\mu(x,z) = \begin{cases} 1 - \mu(z) \frac{G^B(x,z)}{P^B(x,z)}, & \text{on } \partial St_R, \\ 1, & \text{on } \partial St_B. \end{cases} \quad (7)$$

這與 Neumann 邊界條件（方程 3）幾乎相同，且當 $\mu = 0$ 時會退化為 Neumann 方程。

WoSt 估計器（適用於 Robin 邊界條件）

這與 Dirichlet-Neumann 邊界條件的 WoSt 估計器唯一的不同之處在於引入了 ρ_μ 。

WALK ON STARS WITH ROBIN CONDITIONS

$$\widehat{u}(x_k) = \frac{\rho_\mu(x_k, x_{k+1}) P^B(x_k, x_{k+1}) \widehat{u}(x_{k+1})}{\alpha(x_k) p^{\partial St(x_k,R)}(x_{k+1})} - \frac{G^B(x_k, z_{k+1}) h(z_{k+1})}{\alpha(x_k) p^{\partial St_R(x_k,R)}(z_{k+1})} + \frac{G^B(x_k, y_{k+1}) f(y_{k+1})}{\alpha(x_k) p^{St(x_k,R)}(y_{k+1})}. \quad (8)$$

4.2 使用反射函數選擇球體半徑

選擇合適的球體半徑 R 是 WoSt 在 Robin 邊界條件下數值穩定性的關鍵。

錯誤的選擇方式

- 直觀選擇： $R_0 = d_{\text{silhouette}}(x)$ ，即最近的剪影點距離（與 Neumann 問題類似）。
- 問題：

- 解的估計 \hat{u} 累積了一個不可控的 通量 (throughput) $\prod_k \rho_\mu(x_{k-1}, x_k)$ 。
- ρ_μ 可能在隨機遊走過程中 改變符號，導致數值不穩定。

改進：收縮半徑

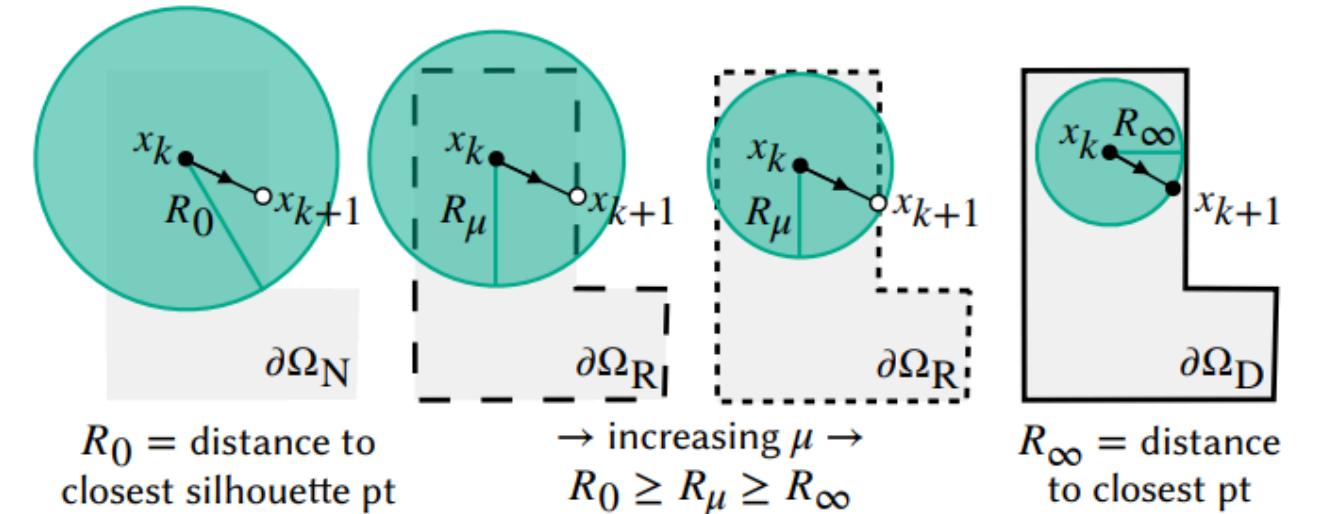
- 為確保通量 ρ_μ 在 $[0,1]$ 範圍內，我們選擇 $R \leq R_0$ ，使得：

$$\rho_\mu(x, z) = 1 - \mu(z) \frac{r}{\cos \theta} \left(1 - \frac{r}{R}\right), \quad \forall z \in \partial S t_R.$$

- 透過變形，我們獲得：

$$R \leq \frac{r}{1 - \frac{\cos \theta}{\mu(z)} r}, \quad \text{當 } r > \frac{\cos \theta}{\mu(z)} r.$$

當 μ 從 0 增加到無窮大時， R 會從 Neumann 半徑 $R_0 = d_{\text{silhouette}}(x)$ 平滑過渡到 Dirichlet 半徑 $R_\infty = d_{\text{Dirichlet}}(x)$ （圖 7）。



當 μ 從 0 增加到無窮大時， R 會從 Neumann 半徑 $R_0 = d_{\text{silhouette}}(x)$ 平滑過渡到 Dirichlet 半徑 $R_\infty = d_{\text{Dirichlet}}(x)$ （圖 7）。

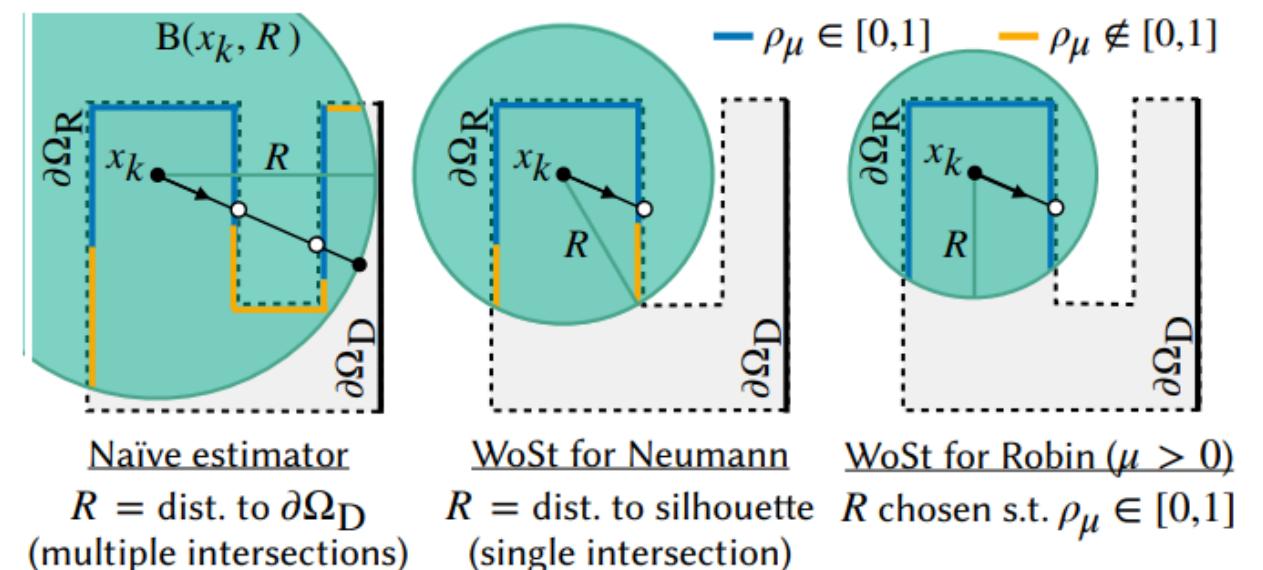


圖 8：三種不同的球體半徑選擇方式

- 左圖：傳統選擇 R 為 Dirichlet 邊界的距離，但這會導致多重交點。
- 中圖：WoSt for Neumann 限制 R 使其僅與 最近的剪影點 相交。
- 右圖：WoSt for Robin 進一步收縮 R ，以確保 ρ_μ 始終在 $[0,1]$ 範圍內。

4.3 使用俄羅斯輪盤技術終止遊走

在隨機遊走過程中，當 $\rho_\mu(x_k, x_{k+1})$ 太小時，直接終止該路徑可以提高計算效率。

方法

- 在第 k 步，我們以機率 $1 - \rho_\mu(x_k, x_{k+1})$ 終止遊走。
- 若未終止，則對應的 通量貢獻被放大，確保整體估計保持無偏（即保持數學期望不變）。
- 這樣做的優勢：
 - 提高計算效率，不必等遊走抵達 Ω_D^ϵ 再終止。
 - 在純 Neumann 問題中不可行（因為所有反射機率均為 1）。

5.1 計算三角形的半徑界限 (Computing Radius Bounds for Triangles)

在 Robin 邊界條件下，WoSt 方法需要確定 星形區域 (star-shaped region) 的半徑。

計算星形區域半徑的方法如下：

1. 使用最接近輪廓點查詢 (Closest Silhouette Point Query, CSPQ) :

- 相對於查詢點 x ，對三角形網格執行 CSPQ 以獲取初始半徑。
- 然後遍歷該半徑內的所有三角形，估計每個三角形 t 的半徑上界，並取最小值作為最終半徑。

2. 解析計算上界半徑 (Analytical Upper Bound Computation) :

- 直接計算半徑上界，而非依賴數值近似。
- 假設所有點 z 都具有最大 Robin 係數 μ_{\max} ，並考慮查詢點 x 到三角形所在平面的距離 h 。
- 公式：

$$R \leq \frac{\mu_{\max} h^2}{\mu_{\max} h \cos \theta - \cos^3 \theta}$$

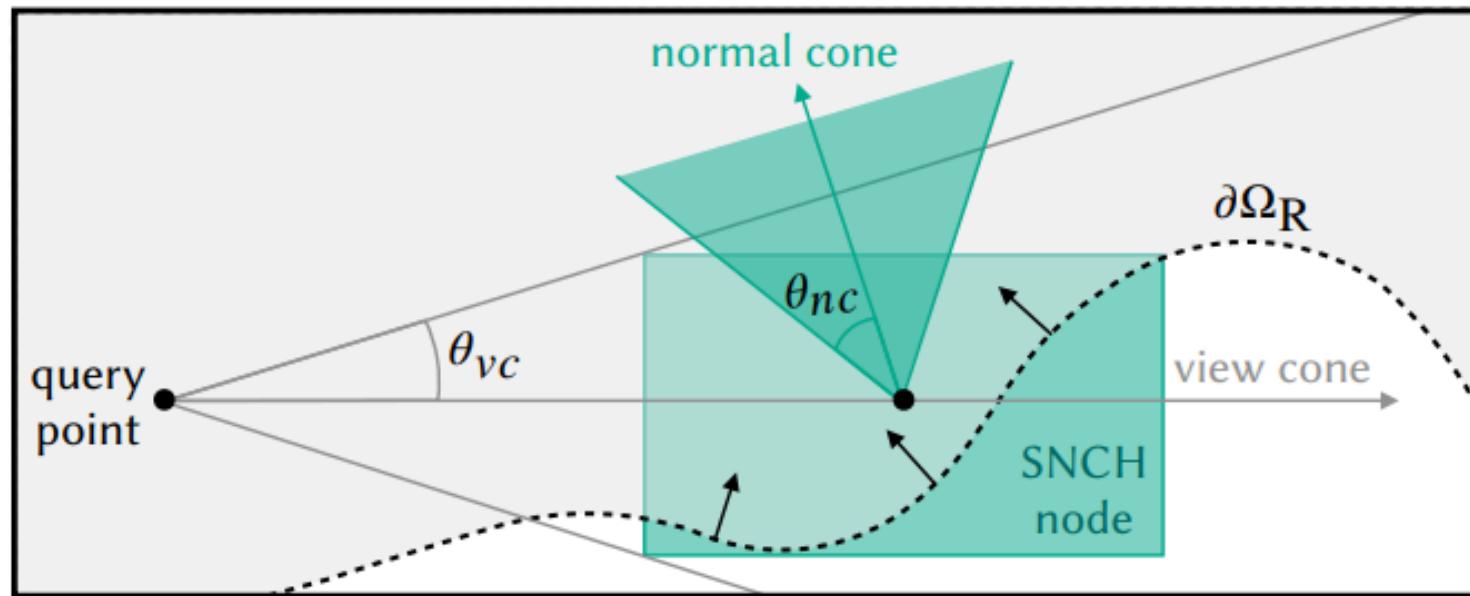
當：

$$\cos \theta \leq \sqrt{\mu_{\max} h}$$

- 透過對 $\cos \theta$ 取導數並設為零，可以獲得解析解：

$$\cos \theta = \sqrt{\frac{\mu_{\max} h}{3}}$$

- 此解會被夾在三角形內的最小和最大角度之間，並代入原公式以計算最終半徑界限。



3. SNCH (Spatialized Normal Cone Hierarchy) 加速技術

- 為了高效執行該計算，使用 SNCH 層級結構 來存儲：
 - 空間範圍 (Bounding Box)
 - 法向錐 (Normal Cone)
 - 最小與最大 Robin 係數
- 這樣的層級結構允許 快速排除不相關的三角形區域，從而加速計算。

5.2 加速星形區域查詢 (Accelerating Star-Shaped Region Queries)

不需要遍歷網格中的所有三角形來確定星形區域的半徑。

優化方法如下：

1. 建立層級加速結構

- 為三角形的 **每個節點** 儲存：
 - 空間範圍 (Bounding Box)
 - 法向錐 (Normal Cone)
 - Robin 條數的最小與最大值
- 法向錐的計算方式：
 - 取節點內所有三角形法向量的平均值作為 **錐軸** (Cone Axis) 。
 - **錐半角** (Cone Half-Angle) 設為 **所有法向量與軸之間的最大偏差** 。

2. 搜尋時的查詢流程

- 類似於 CSPQ 查詢，但會對 SNCH 節點進行 **保守半徑估計**，藉此過濾不必要的遍歷區域。
- 在 **查詢點 x** 處建立一個**視錐** (View Cone)，以確定哪些三角形應該被考慮。

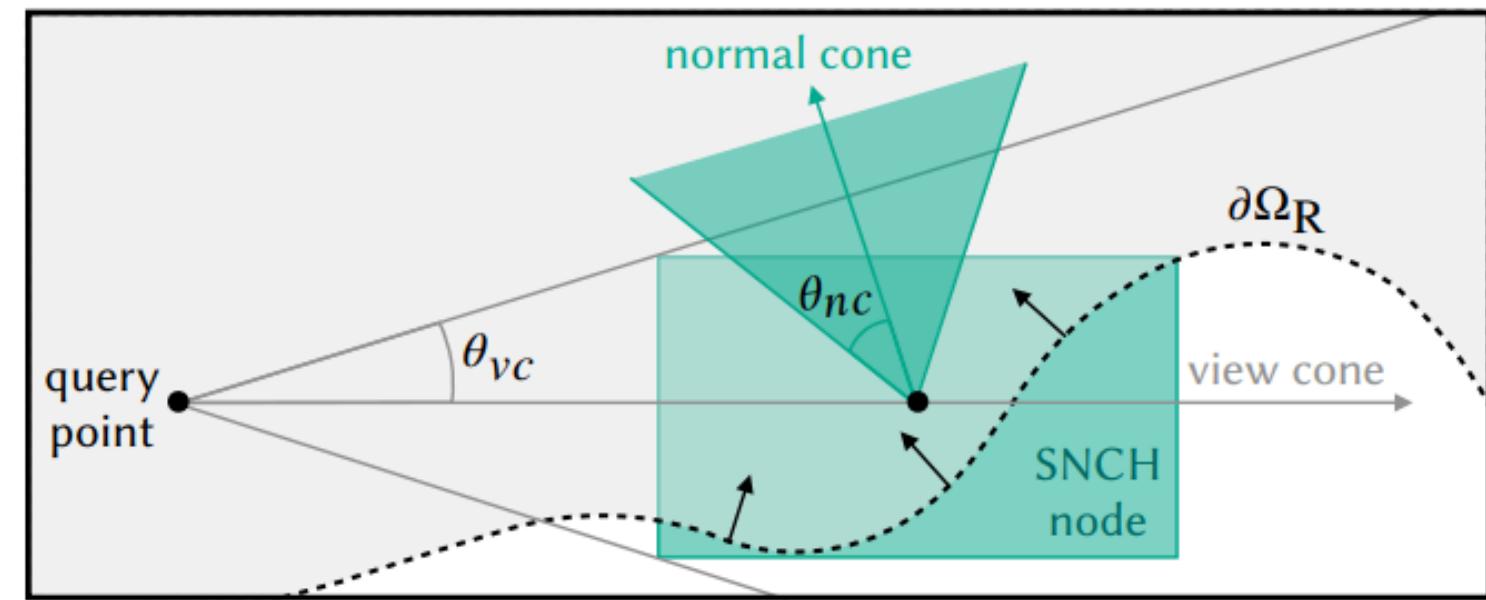


Figure 9. Like Sawhney et al. [2023], we employ a spatialized normal cone hierarchy to accelerate our star-shaped region query for a Robin boundary $\partial\Omega_R$. We use the spatial and angular bounds for the geometry inside an SNCH node (via each node's bounding box and normal cone) to compute a conservative radius for the star-shaped region, which determines whether a node can be skipped as we traverse the hierarchy.

5.3 最終實作

- 最終演算法實作於開源庫 Zombie [Sawhney and Miller 2023]，但也可適用於任何支援以下查詢的邊界表示法：
 1. 最近點查詢 (Closest Point Query, CPQ)
 2. 星形區域半徑查詢
 3. 對 Robin 邊界進行光線交點計算
 4. 對 Robin 邊界進行已知數據點取樣
- 邊界可以是開放的或封閉的，並且可以包含自交錯 (self-intersections) 、裂縫 (cracks) 或孔洞 (holes) 。

6. VARIANCE REDUCTION (方差降低)

本節介紹了兩種針對 Walk on Stars (WoSt) 方法 已開發的 方差降低技術 (variance reduction techniques)，並將其擴展至 Robin 邊界條件，以提高估計的數值品質。

6.1 Boundary Value Caching (邊界值快取 · BVC)

WoSt 方法在估計橢圓型偏微分方程 (PDEs) 解時，對每個點的計算是獨立的，這導致了計算的冗餘與噪聲的增加。

為了降低這種冗餘，邊界值快取 (Boundary Value Caching, BVC) 提出了一種改進方法：

1. 預先使用 WoSt 方法來估計隨機點上的解 u 以及其導數 $\frac{\partial u}{\partial n}$ 。
2. 將這些估計值快取 (cache) 起來，並用於隨後的蒙特卡羅計算。
3. 利用已快取的邊界值來直接評估邊界積分方程 (BIE)，避免重複計算。

BVC 的優勢

- 降低估計的方差與噪聲：與原始 WoSt 方法相比，BVC 產生的解更平滑（如 Dirichlet 與 Neumann 問題）。
- 適用於 Robin 邊界條件：

- 由於 Robin 邊界條件滿足：

$$\frac{\partial u}{\partial n} = \mu u + h$$

- 因此 不需要分別估計 u 和 $\frac{\partial u}{\partial n}$ ，只需快取 u 並將其帶入 BIE 即可。

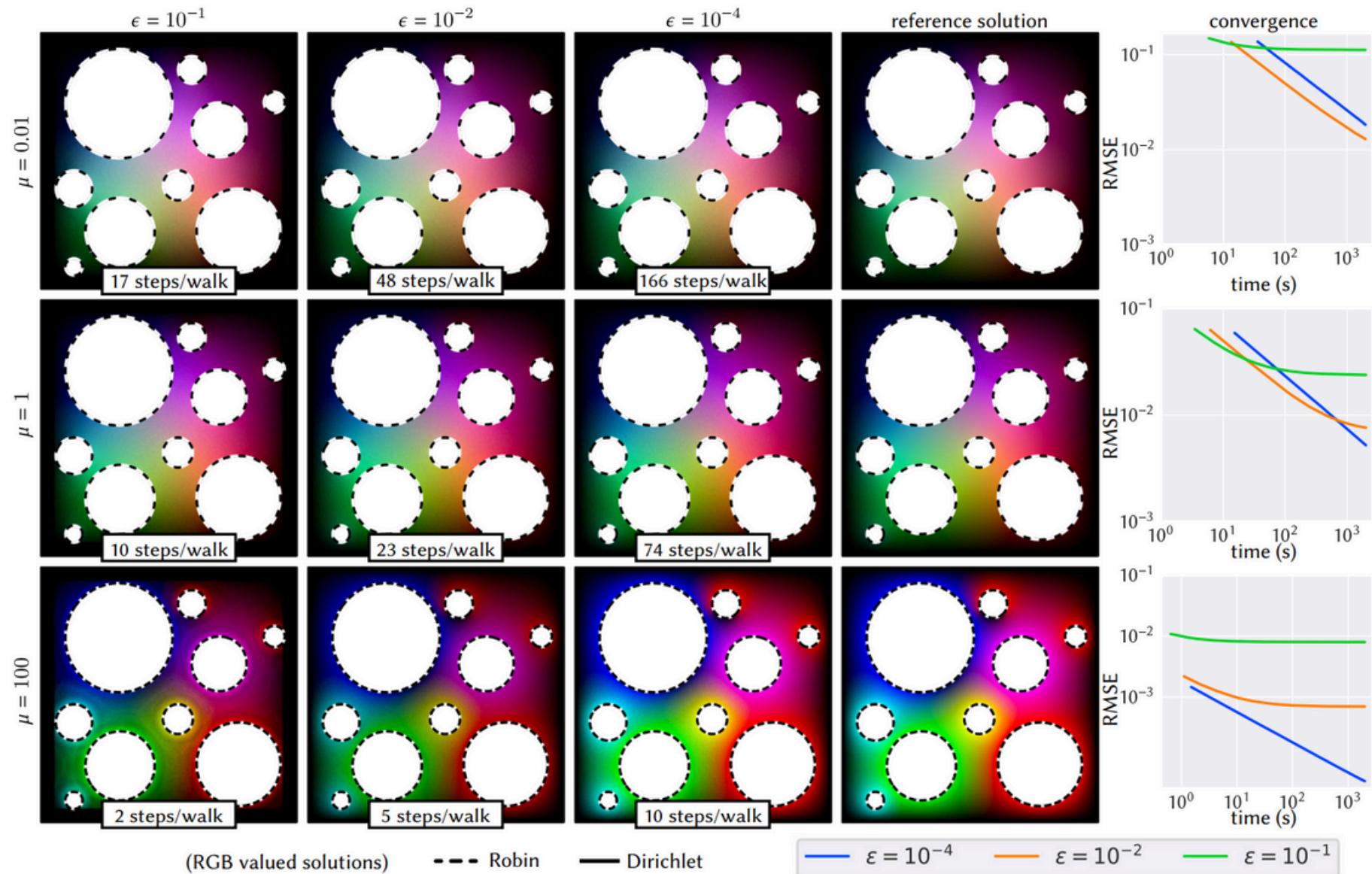


Figure 11. WoSt uses a single ϵ -shell parameter to control the tradeoff between bias in a solution estimate and the number of steps in a walk—in general, this parameter requires little-to-no hand-tuning as bias drops predictably with decreasing ϵ values. Top two rows: For more reflecting Robin boundaries with smaller coefficients μ , bias manifests as a global darkening in the solution estimate for large ϵ , with runtime improvements typically outweighing the relative increase in bias. Bottom row: For more absorbing Robin boundaries with larger coefficients μ , a large ϵ -shell produces a Voronoi-like solution that extends prescribed boundary values further into the domain interior—a similar bias is observed with WoS for pure Dirichlet problems [Sawhney and Crane 2020, Fig. 14]. Bias quickly goes away as ϵ decreases, with only a small increase in the average number of steps per walk.

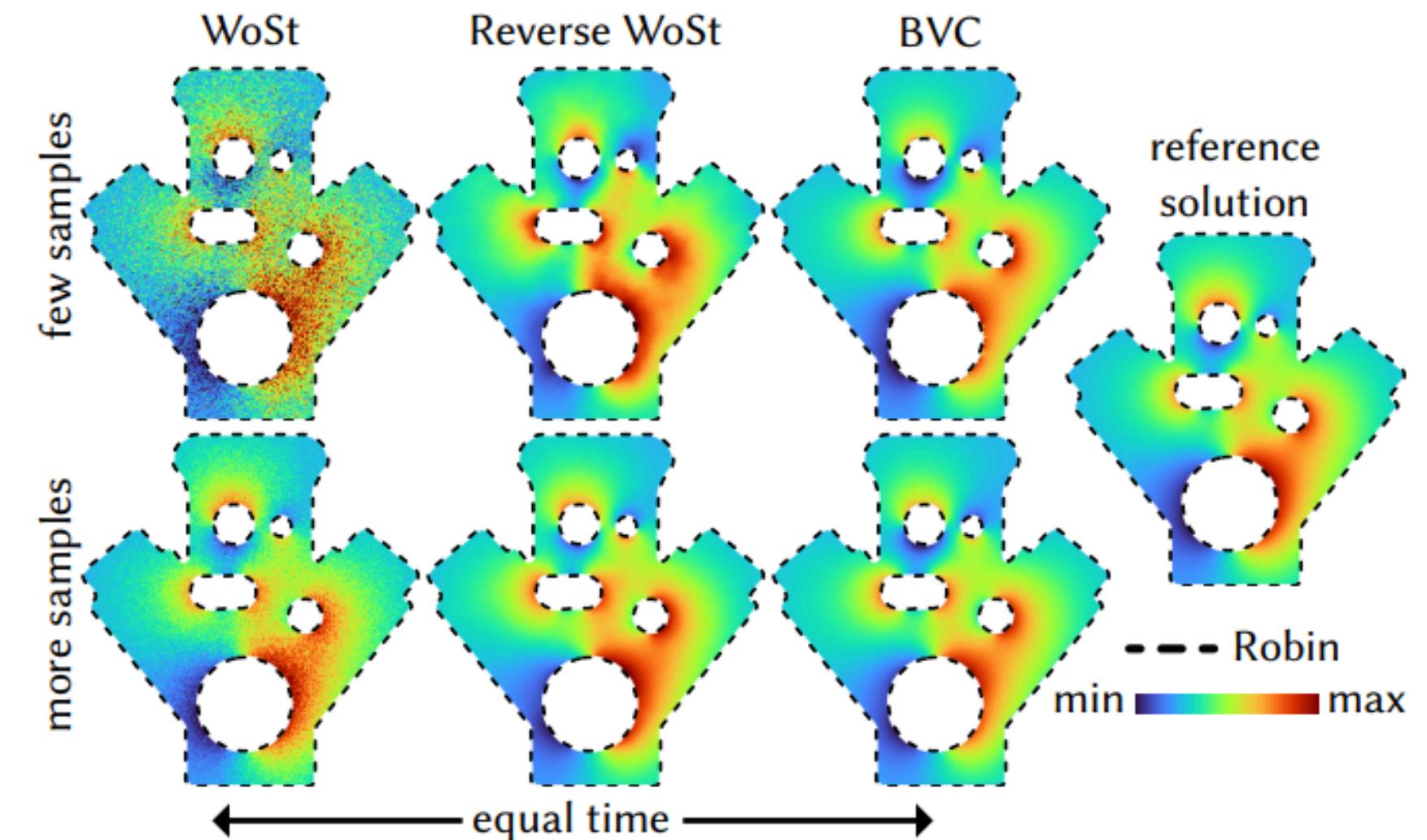
6.2 Bidirectional Formulation (雙向求解)

除了 BVC 之外，另一種方差降低技術是 **雙向求解 (Bidirectional Formulation)**：

- 標準 WoSt 方法 是單向的，即從內部點開始隨機遊走，最終到達邊界並獲得估計值。
- 雙向求解則從邊界與內部點同時進行運算，以減少估計過程中的不確定性。

具體方法

- 對於 Walk on Spheres (WoS) 方法，已有研究 [Qi et al. 2022] 提出雙向方法：
 1. 從內部點 x 向邊界進行隨機遊走。
 2. 從邊界上的隨機點 z 反向模擬其對內部點 x 的影響。
 3. 結合這兩個方向的資訊來降低方差。
- 在 WoSt 上擴展這一技術，使其適用於 Robin 邊界條件：
 - 結合雙向估計與 BVC，以進一步降低估計誤差。
 - 比傳統 WoSt 方法更穩定，且計算結果更平滑（如圖 10 所示）。



6.3 數值結果與比較

本節的數值結果顯示：

1. 相比於標準 WoSt 方法，應用 BVC 和雙向求解後，估計結果更加平滑且噪聲更低。
2. 在相同計算時間內，使用 BVC 的 WoSt 方法 RMSE 明顯低於傳統 WoSt 方法。
3. 在 Robin 邊界條件問題中，結合 BVC 和雙向求解的 WoSt 方法比 Walk on Boundary (WoB) 方法更加準確且方差更小。

7. EVALUATION (評估)

本節評估了 WoSt (Walk on Stars) 方法在不同情境下的數值表現，並與現有方法進行比較。主要關注：

1. 停止容差 (Stopping Tolerances) 與收斂性
2. 與 Walk on Boundary (WoB) 方法的比較
3. 在熱傳導問題中的應用

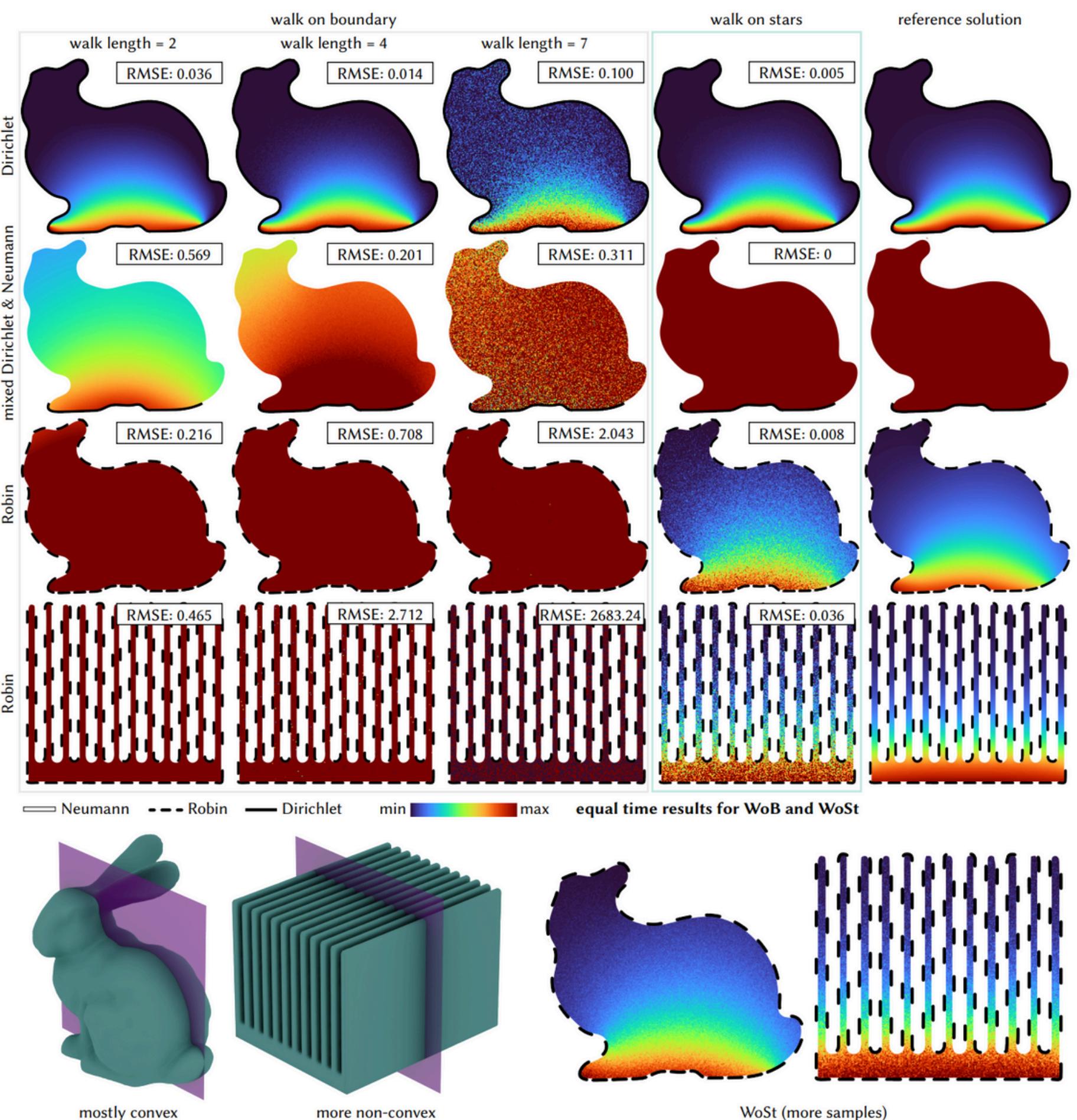
7.1 Stopping Tolerances and Convergence (停止容差與收斂性)

本節探討 WoSt 方法的停止準則 (Stopping Criterion) 如何影響 計算誤差與運行時間。

- WoSt 方法引入了一個參數 ϵ 來控制 蒙特卡羅遊走過程的終止條件：
 - ϵ 定義了一個邊界上的 ϵ -shell，控制步長大小。
 - 較大的 ϵ 允許更大的步長，使得遊走終止得更快，從而降低計算時間，但可能會引入偏差 (bias)。
 - 較小的 ϵ 使得遊走過程更細緻，減少偏差，但會增加計算成本。

數值結果

- 在 Dirichlet 與 Neumann 邊界條件下：
 - ϵ 較大 時，計算結果仍能保持較低偏差，但計算速度顯著提高。
 - 這與 Walk on Spheres (WoS) 方法在純 Dirichlet 問題中的行為相似。
- 在 Robin 邊界條件下：
 - 較大 ϵ 可能會導致偏差，使得解在整體上變得較暗 (reflecting Robin conditions)。
 - 但計算時間的減少通常能夠抵消這些影響，特別是當 Robin 係數 μ 較小時。

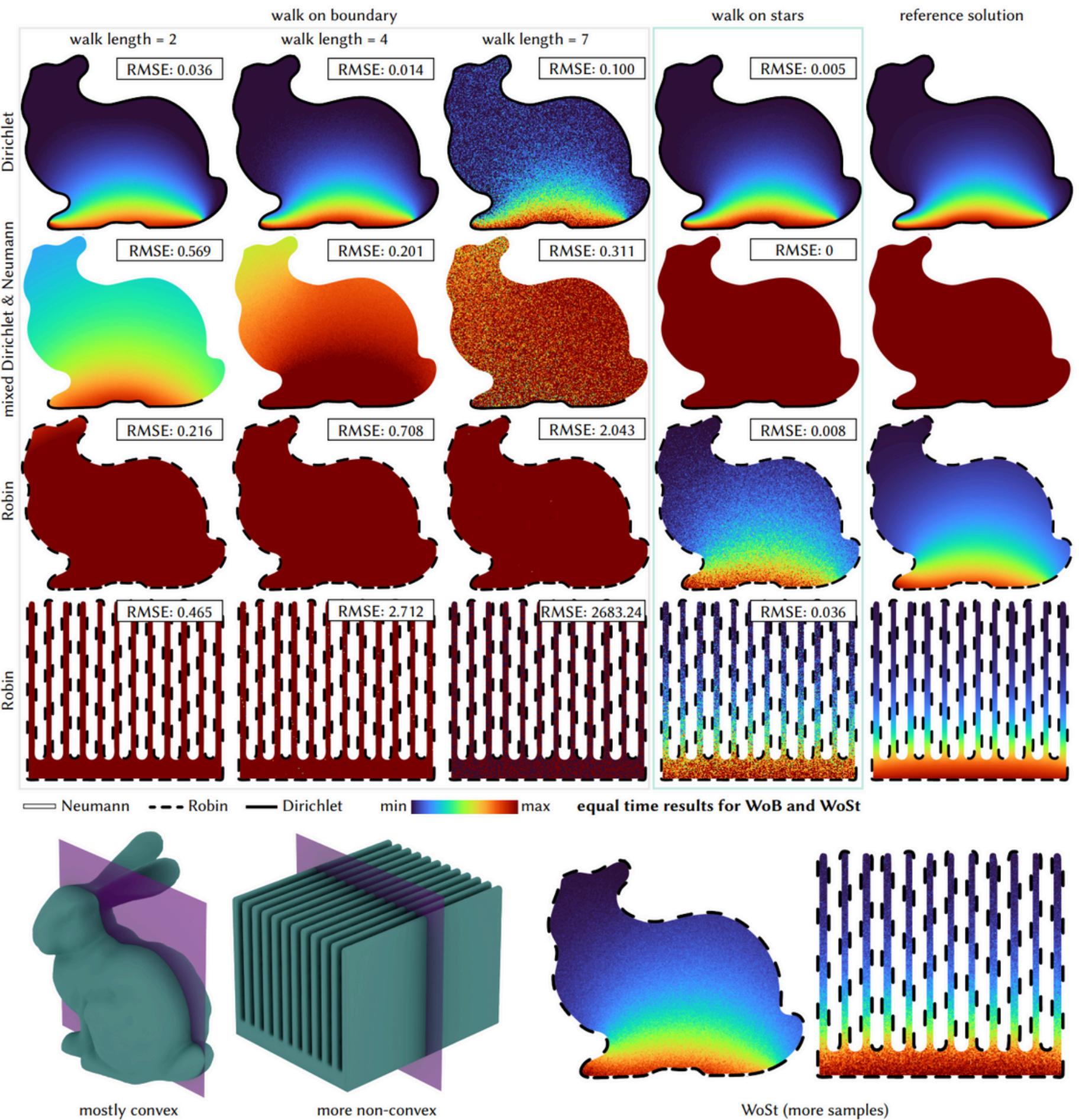


結論

- 較大的 ε 通常能夠顯著減少計算時間，而不會過度影響解的準確性。
 - 在 WoSt 方法中， ε 參數不需要手動微調，因為收斂行為可預測，適用於自適應調整7.2
- Comparison with the Walk on Boundary Method (與 WoB 方法的比較) 本節比較了 WoSt 方法與 Walk on Boundary (WoB) 方法 在數值誤差與計算效率上的表現。

關鍵比較

- WoB 方法的變異數較高，尤其在非凸區域：
 - WoB 依賴於邊界上的隨機遊走，因此在 非凸領域 可能會產生極端偏差。
 - WoSt 方法透過星形區域 (star-shaped regions) 進行隨機遊走，能夠減少這種變異數的影響。
- WoSt 方法在 Robin 邊界條件下更加穩健：
 - WoB 方法的誤差隨著幾何複雜度的增加而劇烈上升，特別是在混合邊界條件 (Dirichlet + Robin) 問題中。
 - WoSt 方法在 Robin 邊界條件下仍能保持低誤差，並且比 WoB 方法更具計算效率。



結果分析

- WoSt 方法比 FEM 更適合局部區域的分析，可避免對整個幾何域進行計算。
- 與 FEM 方法的誤差保持在 1-2% 內，但計算時間大幅減少。
- WoSt 方法允許逐步增量計算（progressive evaluation），可根據需求調整計算範圍。

Table 1. Minimum and maximum estimated solution values, total number of walks, and average walk length for the WoB and WoSt results in Figure 12. Though WoSt generally requires longer walks than WoB, its solution estimates have significantly less error compared to WoB *at equal time* with fewer walks per point.

	estimator	WoB(2)	WoB(4)	WoB(7)	WoSt($\varepsilon=0.001$)
Dirichlet	min value	-0.03	-0.03	-0.48	0
	max value	1.08	1.02	1.39	1.0
	walks per point	1.2×10^7	7×10^6	1.6×10^6	5.5×10^3
	avg. walk length	2	4	7	14
mixed	min value	0.22	0.51	-14.5	1.0
	max value	0.916	1.36	10.06	1.0
	walks per point	1.2×10^7	6.1×10^6	3.5×10^6	6
	avg. walk length	2	4	7	4.6×10^2
Robin (bunny)	min value	0.12	0.43	-17.6	0
	max value	0.47	2.75	53.82	0.18
	walks per point	1.7×10^7	8.3×10^6	4.8×10^6	96
	avg. walk length	2	4	7	62
Robin (heatsink)	min value	0	-9.12	-2.3×10^4	0
	max value	0.82	30.30	4.1×10^4	0.58
	walks per point	1.5×10^7	7.6×10^6	4.3×10^6	36
	avg. walk length	2	4	7	1.6×10^2

ALGORITHM 2: STARRADIUSREFLECTINGBOUNDARY($T = \text{SNCH}(\text{triangles}_{\partial\Omega_R})$, x , R , $d_T^{\min} = 0$)

Note: Code annotated with **comments in green** indicates our modifications to the corresponding procedure in Sawhney et al. [2023].

Input: Spatialized normal cone hierarchy T , query point $x \in \mathbb{R}^3$, current radius estimate R , and minimum distance d_T^{\min} to T 's aabb from x (0 if $x \in$ aabb).

Output: Radius of star-shaped region $\text{St}(x)$ containing a portion of the reflecting boundary $\partial\Omega_R$.

```
1: if  $d_T^{\min} > R$  then return  $R$                                 ▷Ignore nodes outside current radius estimate
2: if  $T$ .isLeaf then
3:   for  $t$  in  $T$ .triangles do
4:      $R \leftarrow \text{STARRADIUSTRIANGLE}(t, x, R)$                       ▷Compute radius bound for triangle  $t$  (Alg. 3)
5: else
6:   visitLeft,  $R_{\text{left}}^{\min}, R_{\text{left}}^{\max} \leftarrow \text{VISITNODE}(T.\text{left}, x, R)$           ▷Determine whether to visit left node (Alg. 4)
7:   if visitLeft then  $R \leftarrow \min(R, R_{\text{left}}^{\max})$ 
8:   visitRight,  $R_{\text{right}}^{\min}, R_{\text{right}}^{\max} \leftarrow \text{VISITNODE}(T.\text{right}, x, R)$            ▷Determine whether to visit right node (Alg. 4)
9:   if visitRight then  $R \leftarrow \min(R, R_{\text{right}}^{\max})$ 
10:  if visitLeft and visitRight then
11:    if  $R_{\text{left}}^{\min} < R_{\text{right}}^{\min}$  then                                         ▷Visit closer node first
12:       $R \leftarrow \text{STARRADIUSREFLECTINGBOUNDARY}(T.\text{left}, x, R, R_{\text{left}}^{\min})$ 
13:       $R \leftarrow \text{STARRADIUSREFLECTINGBOUNDARY}(T.\text{right}, x, R, R_{\text{right}}^{\min})$ 
14:    else
15:       $R \leftarrow \text{STARRADIUSREFLECTINGBOUNDARY}(T.\text{right}, x, R, R_{\text{right}}^{\min})$ 
16:       $R \leftarrow \text{STARRADIUSREFLECTINGBOUNDARY}(T.\text{left}, x, R, R_{\text{left}}^{\min})$ 
17:    else if visitLeft then  $R \leftarrow \text{STARRADIUSREFLECTINGBOUNDARY}(T.\text{left}, x, R, R_{\text{left}}^{\min})$           ▷Visit only left node
18:    else if visitRight then  $R \leftarrow \text{STARRADIUSREFLECTINGBOUNDARY}(T.\text{right}, x, R, R_{\text{right}}^{\min})$           ▷Visit only right node
19:  return  $R$ 
```

ALGORITHM 3: STARRADIUSTRIANGLE($t, x, R^{\max} = \infty$)

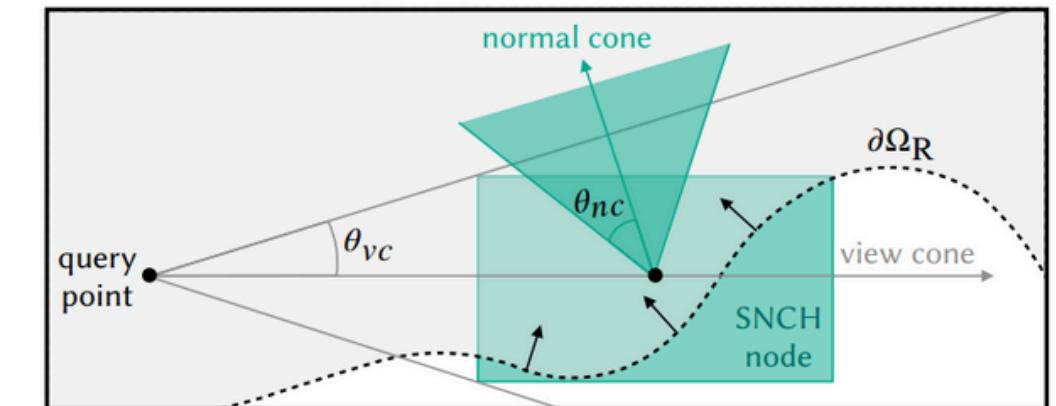
Note: Code annotated with **comments in green** indicates our modifications to the corresponding procedure in Sawhney et al. [2023].

Input: Triangle t with min and max robin coefficients μ^{\min} and μ^{\max} , query point $x \in \mathbb{R}^3$, and radius bound R^{\max} .

Output: Radius of star-shaped region for triangle t .

```

1:  $R_t \leftarrow R^{\max}$                                      ▷ Initialize radius value
2:  $d^{\text{closest}}, x^{\text{closest}} \leftarrow \text{CLOSESTPOINTTRIANGLE}(t, x)$ 
3: if  $d^{\text{closest}} > R_t$  then return  $R_t$            ▷  $t$  is outside radius bound, return the radius bound
4: if  $t.\mu^{\max} \equiv \infty$  then return  $d^{\text{closest}}$       ▷  $t$  has Dirichlet conditions, return distance to closest point on  $t$ 
5:  $n_t \leftarrow \text{TRIANGLENORMAL}(t)$ 
6: for  $e$  in  $t.\text{adjacentEdges}$  do                  ▷ Visit edges adjacent to  $t$  and compute distance to closest silhouette edge
7:    $p_e \leftarrow \text{CLOSESTPOINTEdge}(e, x)$ 
8:    $v \leftarrow p_e - x$ 
9:    $d_e \leftarrow |v|$ 
10:  if  $d_e < R_t$  then
11:    hasAdjacentTriangle,  $n_{\text{adj}} \leftarrow \text{ADJACENTTRIANGLENORMAL}(t, e)$ 
12:    isSilhouetteEdge  $\leftarrow$  not hasAdjacentTriangle or  $(v \cdot n_t) \cdot (v \cdot n_{\text{adj}}) \leq 0$ 
13:    if isSilhouetteEdge then  $R_t \leftarrow \min(R_t, d_e)$ 
14:  if  $t.\mu^{\min} \equiv 0$  then return  $R_t$           ▷  $t$  has Neumann conditions, return distance to closest silhouette edge
15: else                                         ▷  $t$  has Robin conditions, compute radius bound for  $t$  (Eq. 12)
16:    $d^{\text{farthest}}, x^{\text{farthest}} \leftarrow \text{FARTHESTPOINTTRIANGLE}(t, x)$ 
17:    $\cos^{\max} \theta \leftarrow |n_t \cdot (x - x^{\text{farthest}})| / d^{\text{farthest}}$ 
18:    $\cos^{\min} \theta \leftarrow |n_t \cdot (x - x^{\text{closest}})| / d^{\text{closest}}$ 
19:    $h \leftarrow \text{DISTANCEPLANE}(x^{\text{closest}}, n_t)$ 
20:    $\mu h \leftarrow t.\mu^{\max} \cdot h$ 
21:   if  $\sqrt{\mu h} < \cos^{\min} \theta$  then return  $R_t$ 
22:    $\cos \theta \leftarrow \text{CLAMP}(\sqrt{\mu h / 3}, \cos^{\min} \theta, \cos^{\max} \theta)$ 
23:   return  $\min\left(R_t, \frac{\mu h^2}{\mu h \cos \theta - \cos^3 \theta}\right)$ 
```



ALGORITHM 4: VISITNODE($T, x, R = \infty$)

Note: Code annotated with **comments in green** indicates our modifications to the corresponding procedure in Sawhney et al. [2023].

Input: Spatialized normal cone hierarchy node T with min and max robin coefficients μ^{\min} and μ^{\max} , query point $x \in \mathbb{R}^3$, and current radius estimate R .

Output: Whether to traverse the node, as well as min and max bounds on the star radius computed using T 's aabb and cone.

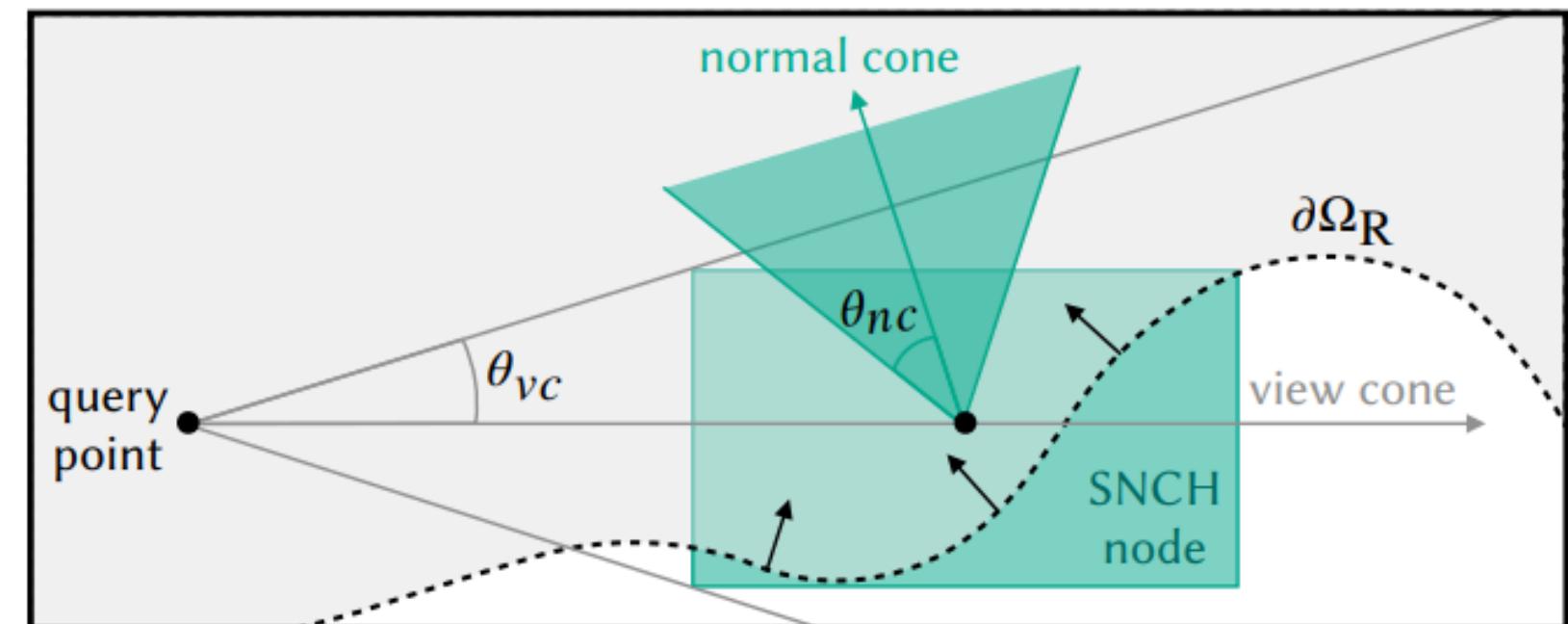
```

1: visit,  $d^{\min}, d^{\max} \leftarrow \text{INTERSECTAABBS_SPHERE}(T.\text{aabb}, x, R)$        $\triangleright$  Intersect aabb with sphere  $\partial B(x, R)$ , and compute min and max distance to aabb from  $x$ 
2: if not visit then return false,  $0, \infty$                                           $\triangleright$  Do not visit node as aabb does not intersect  $B(x, R)$ 
3: if  $T.\mu^{\min} \equiv \infty$  then return true,  $d^{\min}, d^{\max}$                                  $\triangleright$   $T$  only contains a Dirichlet boundary, visit node
4: hasSilhouette,  $|\cos^{\min} \theta|, |\cos^{\max} \theta| \leftarrow \text{HAS SILHOUETTE}(T.\text{aabb}, T.\text{cone}, x)$            $\triangleright$  Sawhney et al. [2023, Alg. 4]
5: if hasSilhouette then return true,  $d^{\min}, \infty$      $\triangleright$  Visit node since the normal & view cones formed by aabb &  $x$  likely contain a pair of orthogonal directions
6: else if  $T.\mu^{\max} \equiv 0$  then return false,  $0, \infty$      $\triangleright$  Do not visit node as  $T$  contains an entirely front- or back-facing Neumann boundary with no silhouette edge
7:  $R^{\min} \leftarrow d^{\max} \leq \frac{|\cos^{\min} \theta|}{T.\mu^{\max}} ? \infty : d^{\min} / \left(1 - \frac{|\cos^{\min} \theta|}{T.\mu^{\max} d^{\max}}\right)$             $\triangleright$  Compute minimum radius bound for Robin boundary inside  $T$  (Eq. 11)
8:  $R^{\max} \leftarrow d^{\min} \leq \frac{|\cos^{\max} \theta|}{T.\mu^{\min}} ? \infty : d^{\max} / \left(1 - \frac{|\cos^{\max} \theta|}{T.\mu^{\min} d^{\min}}\right)$             $\triangleright$  Compute maximum radius bound for Robin boundary inside  $T$  (Eq. 11)
9: return true,  $R^{\min}, R^{\max}$                                                $\triangleright$  Visit node

```

$$R \leq \frac{r}{1 - \frac{\cos \theta}{\mu(z) r}} \quad \text{when } r > \frac{\cos \theta}{\mu(z)}, \quad (11)$$

$$R \leq \frac{\mu^{\max} h^2}{\mu^{\max} h \cos \theta - \cos^3 \theta} \quad \text{when } \cos \theta \leq \sqrt{\mu^{\max} h}. \quad (12)$$



Conclusion

- WoSt 方法統一了處理一階線性邊界條件的框架，使其能夠高效地模擬物理邊界條件，並且計算結果更加真實（ greater physical realism ）。
- WoSt 在複雜幾何結構中仍能穩健運行：
 - 不需要幾何預處理（ geometric preprocessing ）或體積網格劃分（ volumetric meshing ）。
 - 誤差會隨樣本數量增加而可預測地下降（ predictable error decay ）。
- 相較於傳統方法（如有限元法 FEM ），WoSt 具備更高效的計算能力，特別是在涉及高細節幾何時。