Why Simplex Noise for Neuron Density Fields

Technical Note for SKA-Riemannian Neural Fields

Bouarfa Mahi
Quantiota
info@quantiota.org

January 2026

Abstract

The SKA Riemannian Neural Fields framework requires spatially coherent density fields $\rho(\mathbf{r})$ in arbitrary dimensions. This note explains why Simplex noise is the only known practical procedural noise function that simultaneously enables biologically plausible, computationally tractable density substrates in 4D and 5D information manifolds.

1 The Problem

The metric tensor in the Riemannian Neural Fields framework depends on density gradients:

$$g_{ij}(\mathbf{r}) = \alpha \cdot (\nabla h)_i(\nabla h)_j + \beta \cdot (\nabla \rho)_i(\nabla \rho)_j + \gamma \cdot \delta_{ij} \tag{1}$$

This requires a density field $\rho(\mathbf{r})$ with the following properties:

- Smooth gradients $\nabla\rho$ for metric tensor construction
- No directional artifacts that would bias geodesic paths
- Computational scalability to higher dimensions (4D, 5D, ...)
- Spatial coherence that persists as dimension increases
- Biological plausibility (30,000–180,000 neurons/mm^3)

2 Why Not Perlin Noise?

Ken Perlin's original noise function (1985) uses a hypercubic grid. In dimension n, each grid cell has 2^n vertices, leading to:

Property	**Perlin Noise**	**Simplex Noise**
Grid structure	Hypercube	Simplex
Vertices per cell	2^n	$n+1$
Complexity	$O(2^n)$	$O(n^2)$
Directional artifacts	Yes (worsen with D)	No
Gradient continuity	Degrades	Preserved

Table 1: Comparison of noise functions for high-dimensional applications.

At $n = 5$: Perlin requires 32 vertices per cell; Simplex requires 6.

3 Simplex Geometry

The n-simplex is the simplest polytope in n dimensions:

Dimension	**Simplex**	**Vertices**
2D	Triangle	3
3D	Tetrahedron	4
4D	5-cell (pentachoron)	5
5D	5-simplex	6
nD	n-simplex	$n+1$

Table 2: Simplex structure scales linearly with dimension.

This linear scaling is why Simplex noise remains computationally tractable in high dimensions while Perlin noise becomes exponentially expensive.

4 Alternatives Considered

Alternative	**Failure Mode**
Perlin noise	Exponential cost, artifacts above 3D
Random field	No spatial coherence $\Rightarrow$ meaningless $\nabla\rho$
Gaussian random field	No bounded, artifact-free gradient control
Uniform density	No structure to drive geodesic differentiation
Analytic functions	No biological plausibility

Table 3: Why alternative approaches fail for RNF density substrates.

5 Empirical Validation

The supplementary material demonstrates coherent density fields across dimensions:

Dim	**Grid**	**Cells**	**Neurons**	**Std Dev**
3D	15^3	3,375	0.34M	± 20.1
4D	15^4	50,625	5.10M	± 14.2
5D	15^5	759,375	75.94M	± 10.3

Table 4: Density statistics across dimensions. The narrowing standard deviation reflects the central limit theorem in higher-dimensional sampling, not loss of structure.

Crucially, the *spatial coherence persists*—smooth gradients and organic density variations are preserved in 4D and 5D, as shown in the visualization figures.

6 Implications

Most high-dimensional constructs collapse:

- Euclidean distance loses discriminative power
- Volume concentrates on hypersphere shells

- Clustering algorithms fail
- Nearest-neighbor search becomes meaningless

The Riemannian Neural Fields framework avoids these problems because:

1. Simplex noise provides artifact-free, coherent gradients in any D
2. The metric tensor is built from gradients, not Euclidean distances
3. Geodesics follow intrinsic curvature, not ambient geometry
4. The entropy functional is dimension-independent

7 Conclusion

Simplex noise is not an arbitrary choice. It is the only known practical procedural noise function that simultaneously enables:

- $O(n^2)$ computational complexity
- Artifact-free gradient fields
- Dimensional scalability
- Biological plausibility

This choice *enables* the framework's extension to 4D and 5D information manifolds. With any other noise function, the Riemannian Neural Fields framework would collapse in high dimensions.

References

Perlin, K. (1985). An image synthesizer. *SIGGRAPH Computer Graphics*, 19(3), 287–296.

Perlin, K. (2002). Improving noise. *ACM SIGGRAPH 2002 Papers*, 681–682.

Gustavson, S. (2005). Simplex noise demystified. Technical Report, Linköping University.