Grid Field Formulation

Vinamr Jain

September 15, 2025

Contents

1 Field Implementation Strategy

1.1 Design Philosophy

The implementation follows a two-stage paradigm: first generate a spatially smooth continuous field that satisfies global constraints, then discretize it into probability mass functions while enforcing local constraints.

1.2 Stage 1: Continuous Field Generation

Continuous Field Formulation: The continuous displacement field $\mathbf{c}(\mathbf{r}) = (c_u(\mathbf{r}), c_v(\mathbf{r}))$ is defined over the spatial domain, where $\mathbf{r} = (x, y, z)$ represents continuous spatial coordinates. At discrete grid points, we evaluate:

$$\mathbf{c}_{i,j,k} = \mathbf{c}(\mathbf{r}_{i,j,k}) = (c_u(\mathbf{r}_{i,j,k}), c_v(\mathbf{r}_{i,j,k}))$$

where $\mathbf{r}_{i,j,k}$ are the physical coordinates corresponding to grid indices (i, j, k). Spatial smoothness is ensured by construction through the continuous field's inherent smoothness properties and spatial correlation structure. The constructed field must satisfy approximate flow conservation.

1.3 Stage 2: Construct Local PMFs from the Continuous Field

Input: A continuous vector field $\mathbf{c}_{i,j,k} = (c_u, c_v)_{i,j,k}$

Output: A valid PMF $p_{i,j,k}(u, v)$ at each grid point.

Define Local Weights: For each grid point (i, j, k), use the vector $\mathbf{c}_{i,j,k}$ as the mean of a local bivariate normal PDF $\mathcal{N}(\cdot \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}_{\text{local}})$. Calculate weights for all $(u, v) \in \mathcal{D}$:

$$w_{ijk}(u, v) = \mathcal{N}((u, v) \mid \boldsymbol{\mu} = \mathbf{c}_{ijk}, \boldsymbol{\Sigma}_{\text{local}})$$

Enforce Boundary Conditions: Define an indicator function $I_{ijk}(u, v)$ that is 1 if the displacement (u, v) from (i, j) is within the grid boundaries, and 0 otherwise. Apply it:

$$w'_{ijk}(u, v) = w_{ijk}(u, v) \cdot I_{ijk}(u, v)$$

where, $\mathbb{I}_{i,j,k}(u, v) = 1$ if $(i + u, j + v) \in [1, N_x] \times [1, N_y]$, and 0 otherwise.

Normalize: Calculate the normalization constant $Z_{ijk} = \sum_{(u,v)\in\mathcal{D}} w'_{ijk}(u, v)$. The final PMF is:

$$p_{i,j,k}(u, v) = \frac{w'_{i,j,k}(u, v)}{Z_{i,j,k}}$$

Local Constraints:

- **PMF Properties**: The weights $w'_{i,j,k}(u, v) \geq 0$ are non-negative by construction and normalization ensures $\sum_{(u,v)\in\mathcal{D}} p_{i,j,k}(u, v) = 1$.
- **Boundary Conditions**: Enforced exactly by the indicator function $\mathbb{I}_{i,j,k}(u, v)$. When $(i + u, j + v) \notin [1, N_x] \times [1, N_y]$, we have $\mathbb{I}_{i,j,k}(u, v) = 0$, which zeroes out $w'_{i,j,k}(u, v)$ and consequently $p_{i,j,k}(u, v) = 0$.

- **Variance Bounds**: Controlled by choice of $\mathbf{\Sigma}_{\text{local}}$. Define the local covariance matrix as:

$$\mathbf{\Sigma}_{\text{local}} = \begin{pmatrix} \sigma_u^2 & \rho_{uv}\sigma_u\sigma_v \\ \rho_{uv}\sigma_u\sigma_v & \sigma_v^2 \end{pmatrix}$$

  The resulting PMF $p_{i,j,k}(u,v)$ is a discretized and renormalized version of $\mathcal{N}(\mathbf{c}_{i,j,k}, \mathbf{\Sigma}_{\text{local}})$. While boundary truncation and renormalization modify the exact variance, the choice of $\mathbf{\Sigma}_{\text{local}}$ directly controls the concentration of probability mass around the mean $\mathbf{c}_{i,j,k}$, allowing approximate satisfaction of the variance bounds through appropriate parameter selection.

- **Covariance Structure**: The cross-correlation ρ_{uv} in $\mathbf{\Sigma}_{\text{local}}$ directly controls the correlation between $U_{i,j,k}$ and $V_{i,j,k}$ components in the resulting PMF, subject to boundary effects and renormalization.

2 Continuous Field Generation

2.1 Method I: Streamfunction-Based Incompressible Fields

2.1.1 General Framework for Incompressible Field Generation

Theoretical Foundation: Any divergence-free vector field in two dimensions can be expressed in terms of a scalar streamfunction. This provides a systematic approach for generating incompressible displacement fields that maintains exact flow conservation while preserving spatial correlation properties.

Given any continuous scalar field $\Psi(\mathbf{r})$ defined over the spatial domain $\mathbf{r} = (x, y, z)$, we can construct an exactly incompressible displacement field via:

$$\mathbf{c}(\mathbf{r}) = \nabla \times (\Psi(\mathbf{r})\hat{\mathbf{z}}) = \left(-\frac{\partial \Psi}{\partial y}, \frac{\partial \Psi}{\partial x}\right)$$

This construction automatically satisfies $\nabla \cdot \mathbf{c} = 0$ since:

$$\nabla \cdot \mathbf{c} = -\frac{\partial^2 \Psi}{\partial x \partial y} + \frac{\partial^2 \Psi}{\partial y \partial x} = 0$$

Implementation Pipeline:

1. **Continuous Field Specification**: Define a continuous scalar field $\Psi(\mathbf{r})$ with desired spatial correlation and smoothness properties
2. **Extended Grid Sampling**: Evaluate Ψ on an extended $N = (N_x + 1) \times (N_y + 1) \times N_z$ grid to support boundary derivative calculations
3. **Discrete Differentiation**: Apply central difference operators to compute displacement components on the interior $N_x \times N_y \times N_z$ grid
4. **Linear Transformation**: Express the velocity derivation as matrix operations

Discrete Grid Evaluation: For grid coordinates $\mathbf{r}_{i,j,k} = (i\Delta x, j\Delta y, k\Delta z)$, sample the streamfunction on the extended grid:

$$\mathbf{\Psi} = [\Psi_{0,0,1}, \Psi_{0,1,1}, \ldots, \Psi_{N_x,N_y,N_z}]^T$$

Central Difference Implementation: Compute displacement field components for interior points $(i, j, k) \in \{1, \ldots, N_x\} \times \{1, \ldots, N_y\} \times \{1, \ldots, N_z\}$:

Horizontal Component (u-direction):

$$c_{u,i,j,k} = -\frac{\partial \Psi}{\partial y}\bigg|_{i,j,k} \approx -\frac{\Psi_{i,j+1,k} - \Psi_{i,j-1,k}}{2\Delta y}$$

Vertical Component (v-direction):

$$c_{v,i,j,k} = \frac{\partial \Psi}{\partial x}\bigg|_{i,j,k} \approx \frac{\Psi_{i+1,j,k} - \Psi_{i-1,j,k}}{2\Delta x}$$

Matrix Formulation: Express as linear transformations:

$$\mathbf{c}_u = -\frac{1}{2\Delta y}\mathbf{D}_y\boldsymbol{\Psi}, \quad \mathbf{c}_v = \frac{1}{2\Delta x}\mathbf{D}_x\boldsymbol{\Psi}$$

where $\mathbf{D}_x$ and $\mathbf{D}_y$ are $N \times (N+1)$ sparse matrices encoding central difference operators, mapping from the extended streamfunction grid to the interior displacement field.

2.2 Method Ia: Gaussian Process Streamfunction

Continuous Streamfunction Model: Define the streamfunction as a Gaussian random field:

$$\Psi(\mathbf{r}) \sim \mathcal{GP}(m(\mathbf{r}), K(\mathbf{r}, \mathbf{r}'))$$

Statistical Specification:

- **Mean Function**: $m(\mathbf{r})$
- **Covariance Kernel**: $K(\mathbf{r}, \mathbf{r}') = \sigma_\Psi^2 \exp\left(-\frac{\|\mathbf{r}-\mathbf{r}'\|^2}{2\ell^2}\right)$
- **Hyperparameters**: σ_Ψ^2 controls field variance, ℓ controls spatial correlation length

Discrete Sampling: Generate streamfunction values on the extended grid:

$$\boldsymbol{\Psi} \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$$

where $[\mathbf{K}]_{(i,j,k),(i',j',k')} = K(\mathbf{r}_{i,j,k}, \mathbf{r}_{i',j',k'})$ is the covariance matrix for $(N_x+1) \times (N_y+1) \times N_z$ grid points.

Resulting Field Properties:

- **Exact Incompressibility**: $\nabla \cdot \mathbf{c} = 0$ by construction
- **Spatial Smoothness**: Inherited from Gaussian process smoothness (infinitely differentiable for squared exponential kernel)
- **Statistical Structure**: Velocity components $\mathbf{c}_u$ and $\mathbf{c}_v$ are jointly Gaussian with covariance determined by streamfunction kernel and difference operators

2.3 Method Ib: Simplex Noise Streamfunction

Multi-Octave Streamfunction Construction: Define the streamfunction using simplex noise with fractal structure:

$$\Psi(\mathbf{r}) = \sum_{m=0}^{M-1} A_m \eta_\Psi(2^m \mathbf{r}/L)$$

where $\eta_\Psi(\mathbf{r})$ is a base simplex noise function with the multi-octave parameters:

Octave Parameters:

- M: Number of octaves controlling field complexity
- $A_m = A_0\beta^m$: Amplitude decay with persistence parameter $\beta \in (0, 1)$
- L: Base length scale controlling spatial correlation
- 2^m: Frequency doubling between octaves

Amplitude Normalization: For consistent field statistics:

$$A_0 = \sigma_\Psi \left(\sum_{m=0}^{M-1} \beta^{2m} \right)^{-1/2}$$

where σ_Ψ is the desired streamfunction standard deviation.

Grid Evaluation: Sample the streamfunction at extended grid points:

$$\Psi_{i,j,k} = \sum_{m=0}^{M-1} A_m \eta_\Psi(2^m(i, j, k)/L)$$

Resulting Field Properties:

- **Exact Incompressibility**: $\nabla \cdot \mathbf{c} = 0$ by streamfunction construction
- **Fractal Structure**: Multi-octave construction provides natural turbulence-like patterns
- **Computational Efficiency**: Direct evaluation without matrix operations

2.4 Method II: Multivariate Gaussian

Theoretical Foundation: This approach models displacement means as an multivariate Gaussian with a constraint-satisfying mean vector, ensuring incompressibility in expectation

State Vector Definition: Define the global mean displacement vector:

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_u \\ \boldsymbol{\mu}_v \end{pmatrix} \in \mathbb{R}^{2N}$$

where $\boldsymbol{\mu}_u = [\mu_{u,1,1,1}, \ldots, \mu_{u,N_x,N_y,N_z}]^T$ and $\boldsymbol{\mu}_v = [\mu_{v,1,1,1}, \ldots, \mu_{v,N_x,N_y,N_z}]^T$ with $N = N_x N_y N_z$ total grid points.

Spatial Covariance Structure: The covariance matrix has block structure:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_u^2 \mathbf{K} & \rho_{uv}\sigma_u\sigma_v\mathbf{K} \\ \rho_{uv}\sigma_u\sigma_v\mathbf{K} & \sigma_v^2\mathbf{K} \end{pmatrix}$$

The spatial correlation matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ is defined by:

$$[\mathbf{K}]_{(i,j,k),(i',j',k')} = \exp\left(-\frac{\|(i, j, k) - (i', j', k')\|^2}{2\ell^2} \right)$$

where $\ell > 0$ controls spatial correlation length.

Constraint-Satisfying Mean Construction: For interior grid points, construct mean vector $\boldsymbol{\mu}_0$ satisfying:

$$\frac{\mu_{u,i+1,j,k} - \mu_{u,i-1,j,k}}{2} + \frac{\mu_{v,i,j+1,k} - \mu_{v,i,j-1,k}}{2} = 0$$

for all $(i, j, k) \in \{2, \ldots, N_x - 1\} \times \{2, \ldots, N_y - 1\} \times \{1, \ldots, N_z\}$.

The constraint matrix $\mathbf{A} \in \mathbb{R}^{N_{\text{int}} \times 2N}$ encodes the discrete divergence operator, where $N_{\text{int}} = (N_x - 2)(N_y - 2)N_z$ is the number of interior points. For each interior point (i, j, k), let $\mathbf{e}_\ell$ denote the standard basis column vector with 1 at position ℓ and 0 elsewhere. The constraint matrix row for interior point (i, j, k) is:

$$[\mathbf{A}]_{(i,j,k)} = \frac{1}{2}\left(-\mathbf{e}^T_{\text{idx}(i-1,j,k)} + \mathbf{e}^T_{\text{idx}(i+1,j,k)} - \mathbf{e}^T_{N+\text{idx}(i,j-1,k)} + \mathbf{e}^T_{N+\text{idx}(i,j+1,k)}\right)$$

where $\text{idx}(\cdot)$ maps 3D coordinates to linear indices in $[1, N]$

Let μ_0 denote the solution of the constraint system $\mathbf{A}\boldsymbol{\mu}_0 = \mathbf{0}$

Sampling: Generate field realizations via:

$$\boldsymbol{\mu} \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma})$$

Expected Incompressibility: Since $\mathbb{E}[\boldsymbol{\mu}] = \boldsymbol{\mu}_0$ and $\mathbf{A}\boldsymbol{\mu}_0 = \mathbf{0}$:

$$\mathbb{E}[\nabla \cdot \mathbf{c}] = \mathbb{E}[\mathbf{A}\boldsymbol{\mu}] = \mathbf{A}\mathbb{E}[\boldsymbol{\mu}] = \mathbf{A}\boldsymbol{\mu}_0 = \mathbf{0}$$