Supplementary Material: Mathematical Derivations and Algorithms

Xiluo Deng, supervised by James Osborne

1 Transition Probabilities Derivation

1.1 1D-Moran Process

Consider a single population of size N with i mutants and $N - i$ wild-type cells. Mutants have proliferative advantage λ and wild-type cells have fitness 1. In a frequency-dependent Moran process, the transition probabilities for the number of mutants i to change are given by the probabilities of birth and death events.

Under the setting of Fig. 1, at each time step, one cell is chosen to proliferate proportional to λ, and one of the two neighbors of this chosen cell is chosen uniformly to be replaced (die).
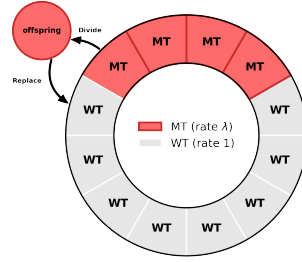

Figure 1: Schematic of the cellular grid layout in the Moran process model.

Let $(X_t)_{t \geq 0}$ be a discrete-time Markov chain on th isstate space $\mathcal{S} = \{0, 1, \ldots, N\}$, where X_t denotes the number of mutants at time step t. The states 0 (extinction of mutants) and N (fixation of mutants) are absorbing states ($P_{0,0} = 1$ and $P_{N,N} = 1$).

Given the total weight of the population $W(i) = \lambda i + 1(N - i)$, the transition probabilities are:

- **Probability of $i \to i + 1$ (Birth of Mutant, Death of WT):**

$$P_{i,i+1} = \underbrace{\frac{2\lambda}{\lambda i + N - i}}_{\text{Select 1 of 2 MTs on boundary}} \times \underbrace{\frac{1}{2}}_{\text{replaces neighbor WT}} = \frac{\lambda}{N + (\lambda - 1)i}$$

- **Probability of $i \to i - 1$ (Birth of WT, Death of Mutant):**

$$P_{i,i-1} = \underbrace{\frac{2}{\lambda i + N - i}}_{\text{Choose 1 of 2 WTs on boundary}} \times \underbrace{\frac{1}{2}}_{\text{replaces neighbor MT}} = \frac{1}{N + (\lambda - 1)i}$$

- **Probability of $i \to i$ (No change):**

$$P_{i,i} = 1 - P_{i,i+1} - P_{i,i-1}$$

1

1.2 Microscopic Spatial Moran Process

We model the crypt as a graph $G = (V, E)$ where nodes represent cells and edges represent spatial proximity, following the Moran-like processes on graphs framework described in [1]. For a cylindrical crypt of size $M \times N$, the vertex set V is indexed by coordinates $u = (r, c)$ with $1 \leq r \leq M$ rows and $1 \leq c \leq N$ columns.

The state of the system at time step k is a binary vector $\mathbf{X}_k \in \mathcal{X} = \{0, 1\}^{|V|}$, where $X_{k,u} = 1$ denotes a mutant cell at position u, and 0 denotes a wild-type cell. The process is a discrete-time Markov chain $(\mathbf{X}_k)_{k \geq 0}$ with two absorbing states: $\mathbf{0}$ (extinction) and $\mathbf{1}$ (fixation).

Parameters and Selection Policy

The evolutionary dynamics are governed by two key components:

1. **Selection Policy $\hat{\boldsymbol{\pi}}(\alpha)$:** This distribution dictates the probability that a wild-type cell at position u is chosen to divide, based on a fitness gradient α. Cells at the bottom of the crypt $(r = 1)$ have the highest proliferative potential:

$$\hat{\pi}(u) = \frac{\alpha^{M-r+1}}{Z}, \quad \text{where } Z = N \sum_{k=1}^{M} \alpha^k.$$

   Mutants possess an additional proliferative advantage λ.

2. **Replacement Weights W:** The adjacency matrix $W = [w_{uv}]$ defines the probability that a dividing cell at u replaces a neighbor at v. Assuming uniform replacement among neighbors on a hexagonal mesh (degree $d_u = 6$ for interior, $d_u = 4$ for boundary):

$$w_{uv} = \begin{cases} 1/d_u & \text{if } v \in \mathcal{N}(u), \quad \mathcal{N}(u) \text{ denotes the set of neighbors of vertex } u \\ 0 & \text{otherwise.} \end{cases}$$
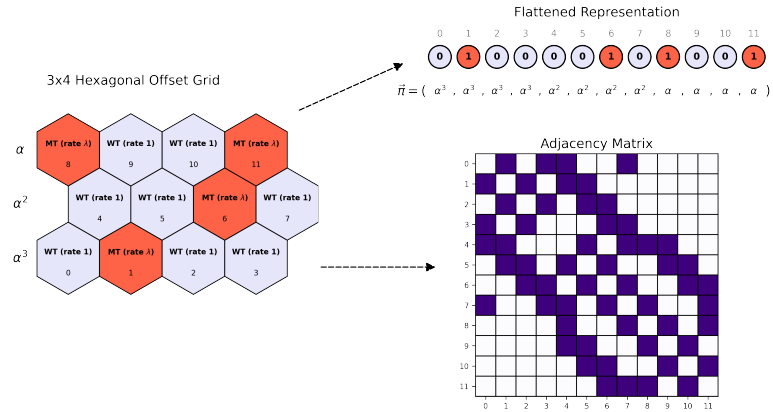
See Fig. 2 for a schematic illustration.

Figure 2: Illustration of the Microscopic model on a 3x4 crypt grid.

Transition Probabilities

Let $\zeta_k = \mathbf{X}_k \hat{\boldsymbol{\pi}}^T$ be the weighted frequency of mutants. The probability of selecting a specific cell v to divide depends on its type and position:

$$P(\text{select } v) = \frac{\hat{\pi}(v) \cdot (\lambda \cdot \mathbb{I}_{\{X_{k,v}=1\}} + \mathbb{I}_{\{X_{k,v}=0\}})}{1 + (\lambda - 1)\zeta_k}$$

A transition $\mathbf{x} \to \mathbf{y}$ occurs when a node u changes its type. This happens if a neighbor v is selected to divide and its offspring replaces u.

- **Birth of a Mutant at** u **($0 \to 1$):** Node u is currently WT and is replaced by a chosen Mutant neighbor v.

$$P(x_u : 0 \to 1) = \sum_{v \in \mathcal{N}(u), x_v = 1} \underbrace{\frac{\lambda \hat{\pi}(v)}{1 + (\lambda - 1)\zeta_k}}_{\text{Select Mutant } v} \times \underbrace{w_{vu}}_{\text{Replaces } u}$$

- **Death of a Mutant at** u **($1 \to 0$):** Node u is currently Mutant and is replaced by a chosen WT neighbor v.

$$P(x_u : 1 \to 0) = \sum_{v \in \mathcal{N}(u), x_v = 0} \underbrace{\frac{\hat{\pi}(v)}{1 + (\lambda - 1)\zeta_k}}_{\text{Select WT } v} \times \underbrace{w_{vu}}_{\text{Replaces } u}$$

For any configuration $\mathbf{x} \in \mathcal{X}$, let $S_{\mathbf{x}}$ denote the set of adjacent states (i.e., those differing from $\mathbf{x}$ by exactly one vertex label):

$$S_{\mathbf{x}} = \{\mathbf{y} \in \mathcal{X} \mid \|\mathbf{x} - \mathbf{y}\|_1 = 1\}$$

The probability of remaining in the current configuration (no state change) is then:

$$P(\mathbf{x} \to \mathbf{x}) = 1 - \sum_{\mathbf{y} \in S_{\mathbf{x}}} P(\mathbf{x} \to \mathbf{y})$$

1.3 Macroscopic Spatial Moran Model

For a cylindrical crypt with M rows and N columns, we reduce complexity by tracking only the number of mutants per row. Let $n_j \in \{0, \ldots, N\}$ be the mutant count in row j. The state is $\mathbf{n} = (n_1, \ldots, n_M)^T$. The process is modeled as a discrete-time Markov chain $(\mathbf{n}_t)_{t \geq 0}$ on state space $\mathcal{S} = \{0, \ldots, N\}^M$, assuming each layer is well-mixed.

The one-step transition sample space is:

$$\Omega = \{\text{Stay}\} \cup \bigcup_{j=1}^{M} \{(j, \uparrow), (j, \downarrow)\}$$

where $(j, \uparrow)$ and $(j, \downarrow)$ denote a mutant increase or decrease in row j. The recurrence relation of $\mathbf{n}$ in terms of the random outcome $\xi_t \in \Omega$ is:

$$n_j(t+1) = n_j(t) + \mathbb{I}_{\{\xi_t = (j, \uparrow)\}} - \mathbb{I}_{\{\xi_t = (j, \downarrow)\}}, \qquad j = 1, \ldots, M.$$

With selection policy $\hat{\pi}_j = \frac{\alpha^{M-j+1}}{Z}$, $Z = N \sum_{j=1}^{M} \alpha^j$ and effective degree d_j (4 for boundary, 6 otherwise), the transition probabilities are:

$$P(\xi_t = (j, \uparrow)) = \frac{2}{d_j} \left[\underbrace{\frac{\lambda n_j \hat{\pi}_j}{D} \cdot \frac{N - n_j}{N-1}}_{\text{intra-layer birth}} + \sum_{i \in \mathcal{N}(j)} \underbrace{\frac{\lambda n_i \hat{\pi}_i}{D} \cdot \frac{N - n_j}{N}}_{\text{inter-layer birth}} \right]$$

$$P(\xi_t = (j, \downarrow)) = \frac{2}{d_j} \left[\underbrace{\frac{(N - n_j) \hat{\pi}_j}{D} \cdot \frac{n_j}{N-1}}_{\text{intra-layer death}} + \sum_{i \in \mathcal{N}(j)} \underbrace{\frac{(N - n_i) \hat{\pi}_i}{D} \cdot \frac{n_j}{N}}_{\text{inter-layer death}} \right]$$

where $D = N + (\lambda - 1)\mathbf{n}\hat{\boldsymbol{\pi}}^T$. The terms represent:

- $\frac{2}{d_j}$: Probability of selecting a specific connected layer neighbor for replacement.

- $\frac{\lambda n_j \hat{\pi}_j}{D}$: Probability of selecting a mutant in layer j to reproduce. (Replacing λn_j with $N - n_j$ gives probability for WT).

- $\frac{N - n_j}{N-1}$: Probability that a chosen target in the *same* layer is WT. Similarly, $\frac{n_j}{N-1}$ is the probability of targeting a MT.

- $\frac{N - n_i}{N}$: Probability that a chosen target in a *neighbor* layer i is WT. Similarly, $\frac{n_j}{N}$ is the probability of targeting a MT.

The probability of remaining in the same state is:

$$P(\xi_t = \text{Stay}) = 1 - \sum_{j=1}^{M} \left[P(\xi = (j, \uparrow)) + P(\xi = (j, \downarrow)) \right]$$

2 Solving the Markov Chains in Each Model

To determine the fixation probability of the system, we solve for the absorbing probabilities of the associated Markov chains.

2.1 1D Moran Process

The 1D Moran process is a birth-death process on the state space $\mathcal{S} = \{0, 1, \ldots, N\}$. Let ϕ_i be the probability of fixation (reaching state N) starting from i mutants. The boundary conditions are $\phi_0 = 0$ (extinction) and $\phi_N = 1$ (fixation).
For any interior state $1 \leq i \leq N - 1$, the probability satisfies the recursion:

$$\phi_i = P_{i,i+1}\phi_{i+1} + P_{i,i-1}\phi_{i-1} + (1 - P_{i,i+1} - P_{i,i-1})\phi_i$$

Rearranging terms leads to the difference equation:

$$P_{i,i+1}(\phi_{i+1} - \phi_i) = P_{i,i-1}(\phi_i - \phi_{i-1}) \quad \implies \quad \phi_{i+1} - \phi_i = \gamma_i(\phi_i - \phi_{i-1})$$

where $\gamma_i = \frac{P_{i,i-1}}{P_{i,i+1}}$. By applying this recursively, we can express the difference in terms of the first step ($\phi_1 - \phi_0 = \phi_1$):

$$\phi_{i+1} - \phi_i = \left(\prod_{k=1}^{i} \gamma_k\right) \phi_1$$

Since ϕ_i is the sum of these differences ($\phi_i = \sum_{j=0}^{i-1}(\phi_{j+1} - \phi_j)$), we obtain:

$$\phi_i = \phi_1 \sum_{j=0}^{i-1} \prod_{k=1}^{j} \gamma_k$$

Using the normalization condition $\phi_N = 1$ to solve for ϕ_1, the general solution is:

$$\phi_i = \frac{\sum_{j=0}^{i-1} \prod_{k=1}^{j} \gamma_k}{\sum_{j=0}^{N-1} \prod_{k=1}^{j} \gamma_k}$$

For the Moran process with constant fitness λ, the ratio $\gamma_k = 1/\lambda$ is constant. The solution depends on whether selection is neutral or not:

- **Neutral Drift ($\lambda = 1$):** Here $\gamma_k = 1$, so the sums become simple arithmetic progressions ($\sum_{j=0}^{i-1} 1 = i$). The fixation probability is equal to the initial proportion of mutants:

$$\phi_i = \frac{i}{N}$$

- **Selection ($\lambda \neq 1$):** Here the sums are geometric series. Substituting into the equation yields the closed-form solution:

$$\phi_i = \frac{1 - \lambda^{-i}}{1 - \lambda^{-N}}$$

This provides an exact analytical solution for the ring case. As shown in Fig. 3, the analytical results align with the simulation. The graph is obtained from here and the simulation code here.
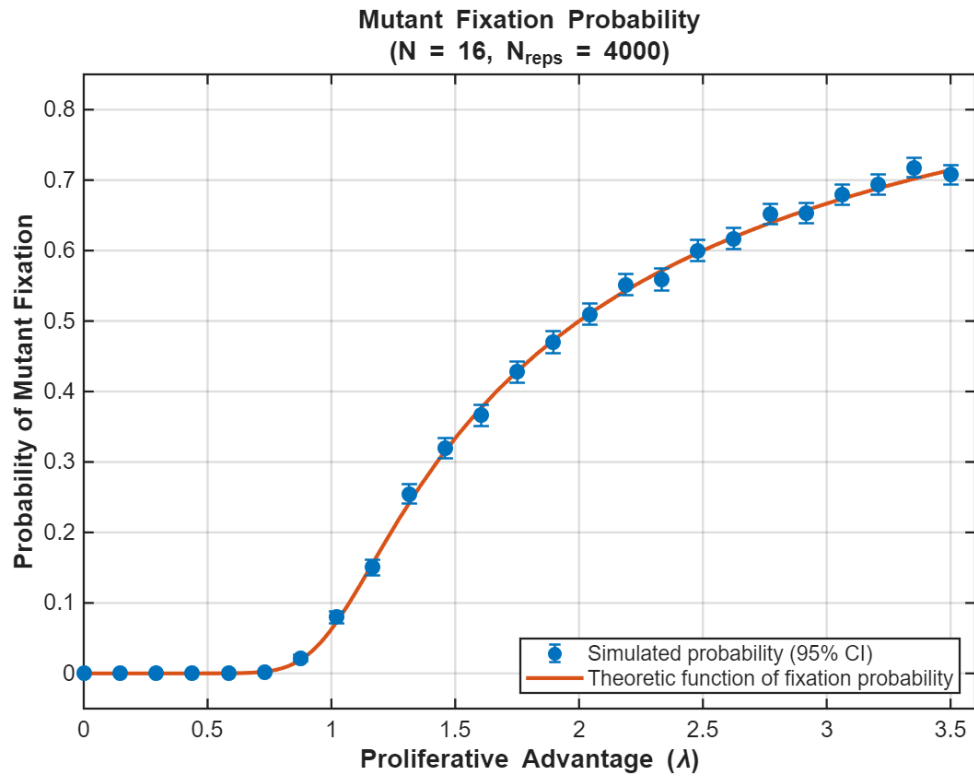
Figure 3: The fixation probability when $i = 1$ (simulation v.s analytical solution)

2.2 Microscopic Spatial Moran Process (MicSMP)

For the Microscopic model, the state space size is $2^{|V|}$, which grows exponentially with the number of cells. However, for small systems (e.g., typical crypt sizes of $N \times M \leq 20$ cells), we can compute the exact fixation probabilities by numerically solving the linear system. The procedure mirrors the sparse matrix construction used in the macroscopic solver:

1. **State Generation:** Enumerate all $2^{|V|}$ possible configurations $\mathbf{x} \in \mathcal{X}$. Map each binary configuration vector $\mathbf{x} \in \{0,1\}^{|V|}$ to a unique integer index $k(\mathbf{x}) \in \{1, \ldots, 2^{|V|}\}$ (e.g., via bitwise interpretation).

2. **Sparse Matrix Construction:** Initialize a sparse matrix $\mathbf{A}$ of size $2^{|V|} \times 2^{|V|}$ and a Right-Hand Side (RHS) vector $\mathbf{b}$. Iterate through every state $\mathbf{x}$ (with index k):

   - **Absorbing States:** If $\mathbf{x} = \mathbf{0}$ (Extinction) or $\mathbf{x} = \mathbf{1}$ (Fixation): Set $A_{kk} = 1$, and set $b_k = 1$ if $\mathbf{x} = \mathbf{1}$ (Fixation), else $b_k = 0$.

   - **Transient States:** For any transient configuration $\mathbf{x}$, the fixation probability $\phi_{\mathbf{x}}$ satisfies the equilibrium condition.

     - Identify the set of neighbor states $S_{\mathbf{x}}$ as defined in Section 1.2. These states $\mathbf{y}$ differ from $\mathbf{x}$ by exactly one vertex (one bit flip).
     - For each $\mathbf{y} \in S_{\mathbf{x}}$, calculate the transition probability $P(\mathbf{x} \to \mathbf{y})$ using the weights and selection policy derived in Section 1.2.
     - Update the matrix entries for row k:

     $$A_{kk} = \sum_{\mathbf{y} \in S_{\mathbf{x}}} P(\mathbf{x} \to \mathbf{y}) \quad \text{(Sum of outgoing probabilities)}, \qquad A_{kj} = -P(\mathbf{x} \to \mathbf{y})$$

     where $j = k(\mathbf{y})$ is the index of state $\mathbf{y}$.
     - This formulation implements $\phi_{\mathbf{k}} \sum_{j \neq k} P_{kj} - \sum_{j \neq k} P_{kj} \phi_j = 0$ by using $1 - P_{kk} = \sum_{j \neq k} P_{kj}$, and the RHS vector $b_k = 0$.

3. **Linear Solver:** Solve the sparse linear system $\mathbf{A}\phi = \mathbf{b}$ to obtain the vector ϕ. The entry $\phi_{k(\mathbf{x}_0)}$ corresponding to the initial configuration $\mathbf{x}_0$ gives the invasion probability.

The code for implementing MicSMP is here.

2.3 Macroscopic Spatial Moran Model (MacSMM)

We apply the same numerical approach to the Macroscopic model, but on the reduced state space $\mathcal{S} = \{0, \ldots, N\}^M$ which tracks only the number of mutants per layer. The size of this state space is $(N+1)^M$, which is significantly smaller than 2^{MN} for large N.

1. **State Generation:** Enumerate all $(N+1)^M$ possible states $\mathbf{n} = (n_1, \ldots, n_M)^T$ where $n_j \in \{0, \ldots, N\}$. Map each vector $\mathbf{n}$ to a unique integer index $k(\mathbf{n})$ (e.g., using a base-$(N+1)$ expansion).

2. **Sparse Matrix Construction:** Initialize matrix $\mathbf{A}$ and vector $\mathbf{b}$ of size $(N+1)^M$. Iterate through every state $\mathbf{n}$ (with index k):

   - **Absorbing States:** If $\mathbf{n} = \mathbf{0}$ (Extinction) or $\mathbf{n} = \mathbf{N}$ (Fixation, all $n_j = N$): Set $A_{kk} = 1$, and set $b_k = 1$ if Fixation, else $b_k = 0$.
   - **Transient States:** For any other state $\mathbf{n}$:
     - Identify reachable neighbor states. The only possible transitions are single birth/death events in a specific layer j:

     $$\mathbf{n}_{\text{up},j} = (n_1, \ldots, n_j + 1, \ldots, n_M) \quad \text{if } n_j < N$$
     $$\mathbf{n}_{\text{down},j} = (n_1, \ldots, n_j - 1, \ldots, n_M) \quad \text{if } n_j > 0$$

     - Calculate probabilities $P(\xi = (j, \uparrow))$ and $P(\xi = (j, \downarrow))$ using the formulas in Section 1.3 for each layer j.
     - Update matrix entries for row k:

     $$A_{kk} = \sum_{j=1}^{M} [P(\xi = (j, \uparrow)) + P(\xi = (j, \downarrow))] \quad \text{(Total rate out)}$$
     $$A_{k,k(\mathbf{n}_{\text{up},j})} = -P(\xi = (j, \uparrow))$$
     $$A_{k,k(\mathbf{n}_{\text{down},j})} = -P(\xi = (j, \downarrow))$$

3. **Linear Solver:** Solve $\mathbf{A}\boldsymbol{\phi} = \mathbf{b}$. The solution $\phi_{k(\mathbf{n}_0)}$ for an initial state (e.g., $\mathbf{n}_0 = (1, 0, \ldots, 0)^T$) provides the macroscopic fixation probability.

The code for implementing MacSMM is here.

3 Special Example in MicSMP

In the context of the spatial Moran process, the graph structure can significantly influence evolutionary dynamics. However, the *Isothermal Theorem*, established by Lieberman, Nowak, and Hauert [2], provides a specific condition under which the spatial structure has no effect on fixation probabilities.

3.1 Definition and Theorem

Following the definition in [1], a graph with weights $W = [w_{uv}]$ is called **isothermal** if the matrix W is bistochastic, meaning the sum of incoming weights equals the sum of outgoing weights for every vertex:

$$\sum_{u \in V} w_{vu} = \sum_{v \in V} w_{uv} = 1, \quad \forall v \in V.$$

Theorem (Isothermal Property): Let the system be governed by a bistochastic weight matrix $\mathbf{W}$ and uniform selection (as reformulated in [1]). For any initial configuration with i mutants and advantage λ, the probability of fixation is identical to the expression derived in Section 2.1:

$$\rho_i = \begin{cases} \frac{i}{N_{total}} & \text{if } \lambda = 1, \\ \frac{1-\lambda^{-i}}{1-\lambda^{-N_{total}}} & \text{if } \lambda \neq 1, \end{cases}$$

where $N_{total} = |V|$ is the total population size.

3.2 Application to Crypt Model (Row=2, Alpha=1)

For the specific implementation of the cylindrical crypt model:

- **Uniformity ($\alpha = 1$):** When the advantage gradient parameter $\alpha = 1$, the selection policy $\hat{\pi}$ becomes uniform.

- **Regularity (Row=2):** If the mesh is constructed such that every node has the same degree and symmetric connection weights, the graph is isothermal. In our setting, it is a crypt with two layers.

- **Result:** In this regime (Row=2, $\alpha = 1$), the spatial simulation results for fixation probability ρ must collapse exactly onto the theoretical curve of the 1D Moran model (Fig.3), regardless of the spatial arrangement of the mutants. Conversely, if $\alpha \neq 1$, the selection asymmetry violates the condition, and ρ diverges from the standard Moran result.
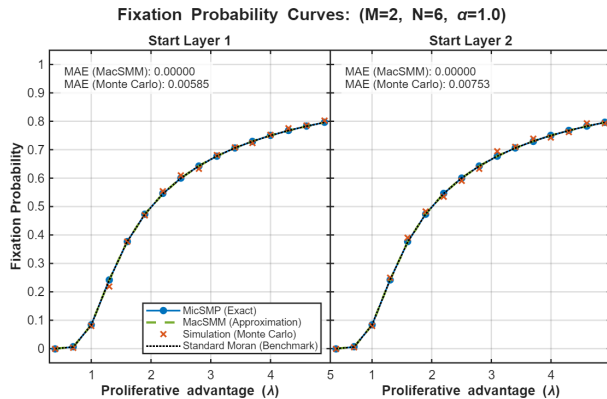

Figure 4: The results comparison for all solvers and simulation to Standard Moran curve, in case of 2x6 crypt with $\alpha = 1$

4 Simulation Algorithms

We perform stochastic simulations using a discrete-time Monte Carlo approach as implemented in here. The code supports two modes: a fully **Spatial** simulation on a hexagonal grid (corresponding to the Microscopic model) and a **Well-Mixed** layer approximation (corresponding to the Macroscopic model).

The simulation proceeds in discrete time steps. In each step, a birth-death event occurs: one cell is selected to proliferate, and one of its neighbor cell will be replaced.

Algorithm 1 Discrete-Time Simulation of Crypt Dynamics

1: **Input:** Grid size $M \times N$, Avantage λ, Gradient α, Mode ('Spatial' or 'Well-Mixed').
2: **Initialize:** Grid $\mathbf{X}$ with initial mutant configuration. Time $t = 0$.
3: **while** not Fixation (all 1) AND not Extinction (all 0) **do**
4:     $t \leftarrow t + 1$.
5:     **// Step 1: Selection for Reproduction (Birth)**
6:     **for** each cell $u = (r, c)$ **do**
7:         Calculate base weight $w_u = \alpha^{M-r+1}$.
8:         **if** X_u is Mutant **then**
9:             $w_u \leftarrow w_u \cdot \lambda$.
10:        **end if**
11:    **end for**
12:    Calculate total weight $W_{tot} = \sum_u w_u$.
13:    Select proliferating cell u_{birth} with probability $P(u) = w_u / W_{tot}$.
14:    **// Step 2: Selection for Replacement (Death)**
15:    **if** Mode is 'Spatial' (Microscopic) **then**
16:        Identify spatial neighbors $\mathcal{N}(u_{birth})$ on the hexagonal grid.
17:   ▷ Grid accounts cylindrical boundary conditions.
18:        Select target $v_{death} \in \mathcal{N}(u_{birth})$ uniformly at random.
19:    **else if** Mode is 'Well-Mixed' (Macroscopic) **then**
20:        Identify connected layers for row r_{birth}:
21:            Same (r), Up $(r-1)$, Down $(r+1)$.
22:        Select a target row r_{target} uniformly ($\frac{1}{3}$ for Same, Up, Down).
23:        Select target cell v_{death} uniformly at random from row r_{target}.
24:    **end if**
25:    **// Step 3: Update State**
26:    $X_{v_{death}} \leftarrow X_{u_{birth}}$.
27: **end while**
28: **return** Fixation Result.

References

[1] P. Keller and M. Ugurlu, "Fixation probability in Moran-like Processes on graphs," *arXiv preprint arXiv:2403.12598*, 2024.

[2] E. Lieberman, C. Hauert, and M. A. Nowak, "Evolutionary dynamics on graphs," *Nature*, vol. 433, no. 7023, pp. 312–316, Jan. 2005, doi: https://doi.org/10.1038/nature03204.