

AXIOM SLCA Symbolic Grid – Local Neighbor Inhibition Guide

■ Overview

This guide explains how to integrate local neighbor inhibition into AXIOM's symbolic node grid using the Spatial Leaky Competing Accumulator (SLCA) model. This enhancement reflects neurobiological realism by allowing each symbolic node to compete with its spatial neighbors via lateral inhibition, better modeling interpretive conflict and coherence dynamics.

■ Local Inhibition Equations

Each node (i,j) updates its accumulators with inhibition contributions from the same accumulators in neighboring nodes. This is applied to both coherent and conflicted accumulators using the following logic:

```
let inhibition_coherent = 0; let inhibition_conflicted = 0; for (const [m, n]
of getNeighbors(i, j)) { inhibition_coherent += acc_conflicted[m][n];
inhibition_conflicted += acc_coherent[m][n]; } inhibition_coherent *=
inhibition; inhibition_conflicted *= inhibition;
```

■■ Accumulator Update Rule

```
const prev_coh = acc_coherent[i][j]; const prev_con = acc_conflicted[i][j];
acc_coherent[i][j] = prev_coh + I_coh[i][j] - inhibition_coherent - leak *
prev_coh + (Math.random() * 2 - 1) * noiseSD; acc_conflicted[i][j] = prev_con
+ I_con[i][j] - inhibition_conflicted - leak * prev_con + (Math.random() * 2 -
1) * noiseSD; acc_coherent[i][j] = Math.max(Math.min(acc_coherent[i][j],
maxVal), minVal); acc_conflicted[i][j] =
Math.max(Math.min(acc_conflicted[i][j], maxVal), minVal);
```

■ Notes

- `getNeighbors(i,j)` returns a list of 4 or 8 neighboring grid coordinates.
- `acc_coherent` and `acc_conflicted` are 2D arrays storing the node values.
- Inhibition is lateral and local, based on spatial proximity in the lattice.
- This model enables spatially emergent conflict resolution patterns in AXIOM's symbolic field.

Prepared for AXIOM Lead Integration | August 2025