Modeling Small Scale Weather with Cellular Automata

Kathryn Litzau and Matt Wisby
Faculty Mentor: Dr. Chris Ahrendt
University of Wisconsin-Eau Claire

1. Introduction

Goal of Our Project:
This project’s ultimate goal is to use cellular automata to model small-scale weather patterns. Each weather variable we account for in our model has its own automaton. We develop a set of rules for how each automaton behaves and how they influence each other.

Cellular Automata:
A standard two-dimensional cellular automaton is a discrete model that changes over time steps by sets of rules that govern the cells. Each cell has a finite number of states it can be in. The new state of a cell is determined after each time step by the states of the cell and its eight surrounding cells.

Advection and Diffusion:
Advection is the transfer of substance by a fluid, in this case wind. Diffusion is the transfer or spreading of heat in a given area. Warm pockets of air surrounded by cooler air will eventually diffuse out so all the air has the same temperature.

The standard PDE used to represent advection/diffusion was applied to three common weather variables:

- Volume fraction of rain: \( \partial R / \partial t + \langle \vec{q} \cdot \vec{v} \rangle = R_{\text{rain}} - R_{\text{cond}} \)
- Temperature: \( \partial T / \partial t + \langle \vec{q} \cdot \vec{v} \rangle = -QR_{\text{cond}} - \beta V \)

Time Step and Distance Choice:
In the Wisconsin area 60 km/hr is a reasonable maximum wind speed. The cells of the cellular automata are square and are of length \( \Delta x = \Delta y = 60 \text{ km/hr} \).

2. Dispersion

Dispersion is represented by the target cell expanding into the 8 adjacent cells. The target cell’s contents are then distributed based on the percentage that overlaps the surrounding cells.

\[
\begin{align*}
\% \text{ left in same cell} & = \frac{3}{8} \\
\% \text{ moves U,D,R or L} & = \frac{3}{4} \\
\% \text{ moves diagonal} & = \frac{1}{8}
\end{align*}
\]

3. Wind and Rules

Wind is modeled by taking the target cell contents and moving it to different cells. Let \( \vec{q} = (u, v) \) be the wind vector in a given cell.

- The horizontal length left in the current cell:
  \[ \Delta x - u = a \]
- The vertical length left in the current cell:
  \[ \Delta y - v = b \]
- \% left in same cell:
  \[ \frac{\Delta x - u}{\Delta x} \]
- \% moves left or right:
  \[ \frac{\Delta y - v}{\Delta x} \]
- \% moves up or down:
  \[ \frac{\Delta x - u}{\Delta y} \]
- \% moves diagonal:
  \[ \frac{\Delta y - v}{\Delta y} \]

Curl:
To analyze rotations in the wind, we approximate curl. The curl’s magnitude tells us the magnitude of rotation and the sign tells us the direction. We use Green’s Theorem to approximate the z-component of curl.

Green’s Theorem is given by:
\[
\oint_C \vec{f} \cdot d\vec{r} = \iint_D \left( \frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} \right) \, dx \, dy
\]

The line integrals we found were:
\[
\oint a \, dx = 0 \\
\oint b \, dy = 0 \\
\oint c \, dx = 0 \\
\oint d \, dy = 0
\]

Thus
\[
\frac{1}{4} \oint \vec{f} \cdot d\vec{r} = a + b + c + d
\]

Our Model’s Rules:
- The change in concentration of moisture follows rain outflow and evaporation:
  \[ \Delta m = -R_{\text{cond}} + V \]
- The change in temperature follows rain outflow and evaporation:
  \[ \Delta m = -R_{\text{cond}} + V \]
- Rain condensing depends on the dew point, \( T_{\text{dew}}(m) \), and by the temperature:
  \[ R_{\text{cond}} = \delta(T(m) - T_{\text{dew}}(m)) \]
- The dew point depends on the moisture:
  \[ T_{\text{dew}} = \frac{T_{0} \gamma}{\gamma + 1} \]
- Evaporation depends on the temperature and the topography:
  \[ V = B(T - T_{\text{dew}}) \]

\( \gamma = 0.5, \delta = 0.25, B = 0.5 \) and \( T_{\text{dew}} = 20^\circ C \).

4. Simulation

Below is an example of all of the variables interacting with each other. The wind grid is shown below and stays constant throughout the entire simulation. The initial temperature values were randomly generated within a specific range. The orange indicates warmer temperatures. This is the initial time step; the temperature will change as the simulation progresses.