Wave Function Collapse Algorithm, and its use in procedural generation.

Byron Fanti (n12360562)

May 30, 2025

1 Motivation

In games, simulations, and digital art, making large, detailed environments by hand takes a lot of time and effort. Developers often need a way to generate levels, textures, or tilemaps that feel natural—without manually placing every single tile or asset. Wave Functio Collapse (WFC) is a procedural generation algorithm that creates new layouts based on an example input, while still following rules like pattern conformity and size limits. It was first released by Maxim Gumin in 2016 and has been used widely in game dev ever since—for things like terrain, buildings, and pixel art (Gumin, 2016; Summerville et al., 2019). The algorithm is inspired by a concept in quantum mechanics with the same name. In quantum theory, a particle exists in a superposition of possible states until it's measured, which causes it to collapse into a single outcome. WFC applies this idea to tiles: each grid cell starts with a "superposition" of all possible tile options. The algorithm picks one (collapses it), then adjusts the surrounding tiles to make sure everything still fits together according to the contraints (Gumin, 2016; Chentanez et al., 2020; Karth & Smith, 2017).

2 Materials

The main input to the WFC algorithm is a sample image or tilemap that shows what kind of pattern or layout you want to replicate. The algorithm looks at this sample and determines which tiles can go next to each other (Gumin, 2016). From there it builds a set of constraints that say, for example, "this tile can only go above that tile," or "this one can't be next to that one." These constraints are what guide the generation process. (Karth & Smith, 2017).

3 Method

The Wave Function Collapse algorithm works by filling in a grid, one tile at a time, based on a set of rules it learns from an input example. At the start every cell in the grid could be any tile, it's in what's called a superposition. The algorithm picks the most "constrained" cell (the one with the fewest valid options left) and collapses it into a single tile.

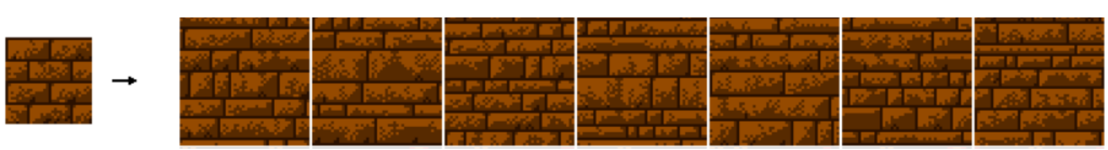

Figure 1 - Image source: Gumin, M. (2016). Wave Function Collapse. Retrieved from https://github.com/mxgmn/WaveFunctionCollapse

To figure out which cell to collapse next, it uses something called Shannon entropy, which is basically a measure of uncertainty. The more tile options a cell has, the higher its entropy:

$$H = -\sum_{i=1}^{n} p_i \log p_i \qquad (1)$$

Where: H is the entropy, p_i is the probability of each tile being chosen, and n is the number of possible tiles.

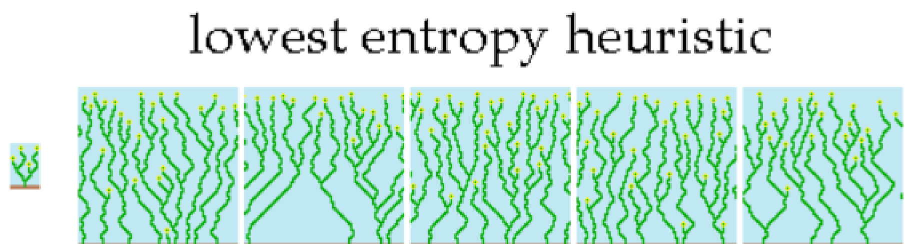

Figure 2 - Image source: Gumin, M. (2016). Wave Function Collapse. Retrieved from https://github.com/mxgmn/WaveFunctionCollapse

So the cell with the lowest entropy (most restricted) gets collapsed first. Once it picks a tile, it propagates that choice to the tiles around it. If a tile can only sit next to certain others, the neighbouring cells lose any options that would break those rules. This continues until the whole grid is filled in, or it hits a contradiction and has to start over or backtrack.

The actual constraints are based on what's allowed to sit next to what. That's done with adjacency rules, which can be described like this (as seen in the above png):

$$C(t_i, t_j, d) = \begin{cases} 1 & \text{if tile } t_i \text{ can have } t_j \text{ in direction } d \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

That just means if tile A can be to the left of tile B, the rule will say "1" for that case. For more information and an example see: https://github.com/byrfan/WaveFunctionCollapse

4 References

Gumin, M. (2016). Wave Function Collapse algorithm. Retrieved from https://github.com/mxgmn/WaveFunctionCollapse

Summerville, A., Snodgrass, S., Mateas, M., & Togelius, J. (2018). Procedural Content Generation via Machine Learning (PCGML). IEEE Transactions on Games.

Karth, I., & Smith, A. M. (2017). WaveFunctionCollapse is constraint solving in the wild. Proceedings of the 12th International Conference on the Foundations of Digital Games.

Chentanez, N., Prachyabrued, P., & Müller, M. (2020). Neural Wave Function Collapse for Procedural Content Generation. ACM SIGGRAPH Asia 2020 Technical Communications.