Wang Tiles and Turing Machines

CSE 30151: Course Project 3

Lauren Ferrara

April 19, 2016

Relevance: In this paper, I discuss Wang Tiles and their relation to Turing Machines. Additionally for this project, I wrote programs to demonstrate how Wang Tiles can be used to make computer graphics and how Turing Machines can be converted to Wang Tiles.

Effort: I spent 10 hours working on this project in order to develop a relevant topic, conduct background research, and write the programs and paper.

Wang Tiles for Computer Graphics

Wang Tiles, first proposed by Hao Wang in 1961, are unit squares with different colors or patterns on each side. A set of these tiles is created so that some tiles have sides that match. Then tiles are placed, without rotating or reflecting the tiles, so that adjacent sides have matching colors/patterns. Wang Tiles have become widely used in computer graphics to make textures for backgrounds, because they can be rendered at run time while maintaining continuity with direct neighbors. This keeps backgrounds from looking choppy and unrealistic.

To demonstrate this, I created a python script with a complete set of 16 tiles that would generate a random Wang Tile texture. A complete set is made when every possible combination of colored edges is made. A complete set is N^2M^2 where N is the number of different colors for the horizontal edges and M is the number of different colors for the vertical edges. In my tile set, I used 2 horizontal edge colors (red and green) and 2 vertical edge colors (yellow and blue), so I needed 16 tiles to create a complete set. A complete set is one way to guarantee a full tiling of a given plane. The 16 tiles I used are as shown below:[1]

[1]Photo Credit: http://procworld.blogspot.com/2013/01/introduction-to-wang-tiles.html

Figure 1: Wang Tile Set

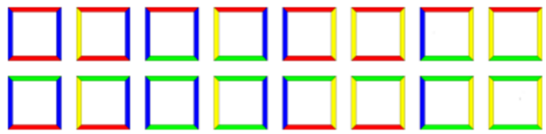


Figure 2: Wang Tile Background with 16 Tile Set

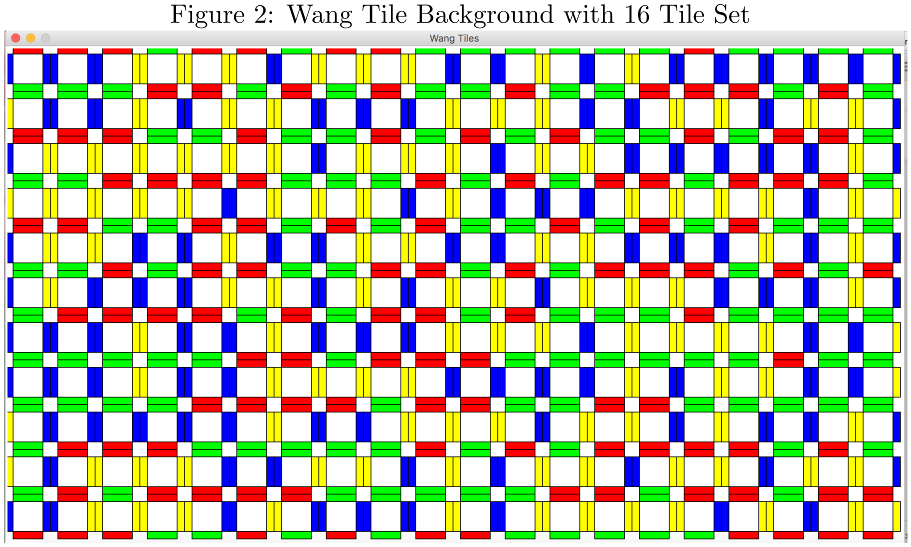
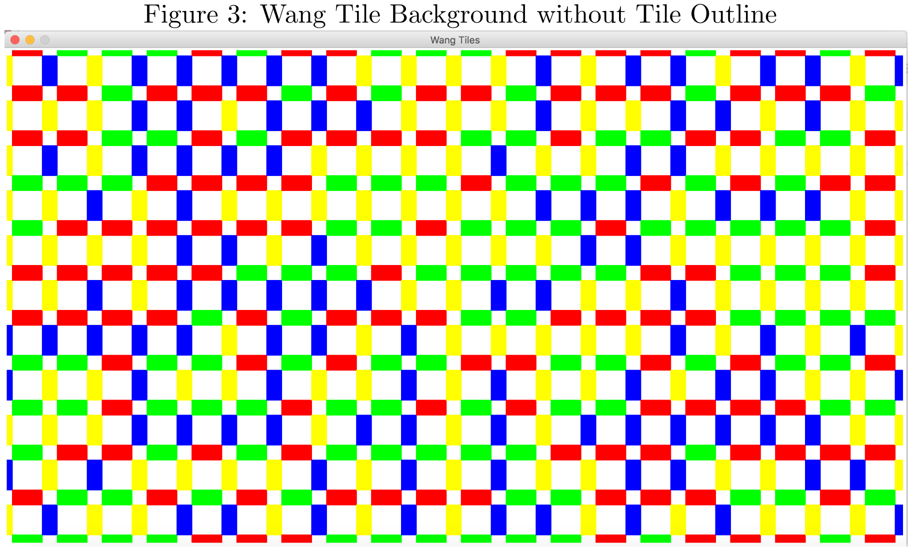


Figure 3: Wang Tile Background without Tile Outline



Relation to Turing Machines

There are two basic questions when given a set of Wang Tiles:

1. Can the Wang Tiles fill a given plane?
2. Can they do this periodically (with a repeating pattern)?

Hao Wang conjectured that these two questions were related: if a finite set of Wang tiles can tile an entire plane, then there must be a periodic tiling to do so. This hypothesis implied an algorithm to decide if a certain Wang tile set could fit a plane.

However, Robert Berger, one of Wang's students, disproved this theory. He successfully showed that any Turing Machine could be converted into an equivalent set of Wang Tiles. Therefore, if an algorithm could decide if a Wang tile set could fill a plane, it would in essence decide the Halting Problem, which had already been proven to be impossible. This therefore led to a contradiction. Thus, whether a set of Wang tiles can tile a plane is undecidable.

To verify this proof, we must explore how a Turing Machine can be converted into a set of Wang tiles. First there must be a set of alphabet tiles, equivalent to the alphabet, $sigma$, of the Turing Machine. These will have particular colors to represent certain symbols on opposite sides and a meaningless color on the other two sides. Then there must be transition tiles, equivalent to each read, write, move transition for the Turing Machine. On the input side of the tile will be a color representing the read symbol in a certain state. The opposite side will have the color of the write symbol. The bottom will have a color to represent the next state if the head is to move right. This color will be at the top if the head is to move left. The other side will be the meaningless color. Then there must be merge tiles to take the input symbol and a state and move the head into the state (q, input symbol), which is given as input to the transition tile. Then we must have special symbols to represent the start state, the accept state, and the initial string.

To make this more concrete, I wrote a python program that would have Wang Tiles act like a Turing Machine that takes an initial input string of 0s and 1s and converts it to its binary complement. Each column of Wang Tiles represents a move of the Turing Machine with the first column representing the input string. The right edges of the last column of tiles contains the binary complement of the input string up until the black edge.

Figure 4: Wang Tiles Used for Turing Machine Simulation

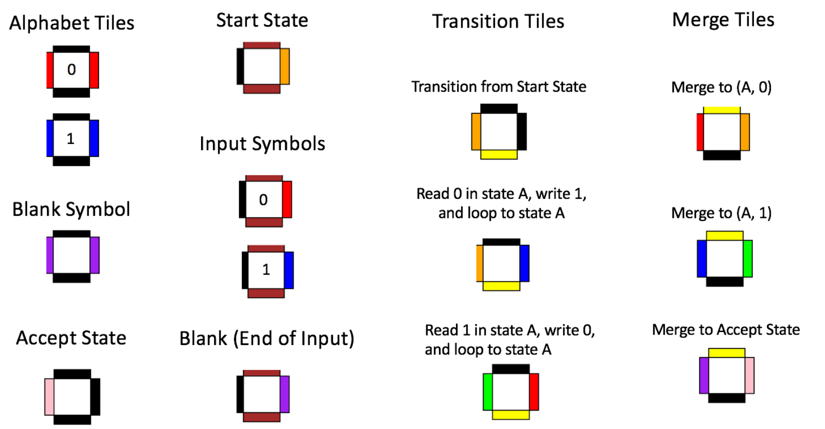


Figure 5: Wang Tile Turing Machine for Binary Complement