Rigidity Percolation on Dual Networks

Abby Davenport
Alma College
University of Michigan REU
PI: Xiaoming Mao
August 7, 2020

During my time in the REU program at the University of Michigan, my goal was to look at how a dual network changes when the original network undergoes rigidity percolation. Using Mathematica, I created an algorithm that generated the direct lattice and the equilibrium matrix (Q). Currently, I am developing a code to create the dual lattice. Once this step is successful, I will be able to look at the effects of rigidity percolation on the dual lattice and the states of self-stress (SSS) of the direct lattice.

Background

To be able to successfully complete this project, it was necessary that I developed a firm foundation about the fundamental components of lattice structures. In a network, the SSS are the tension distribution on bonds that results in no net force on any sites [1]. These can be found by looking at the Q matrix, which relates the tension of each of the bonds with the force on each of the sites: $Q \cdot t = f$. The dimension of the null space of Q is equal to the number of SSS.

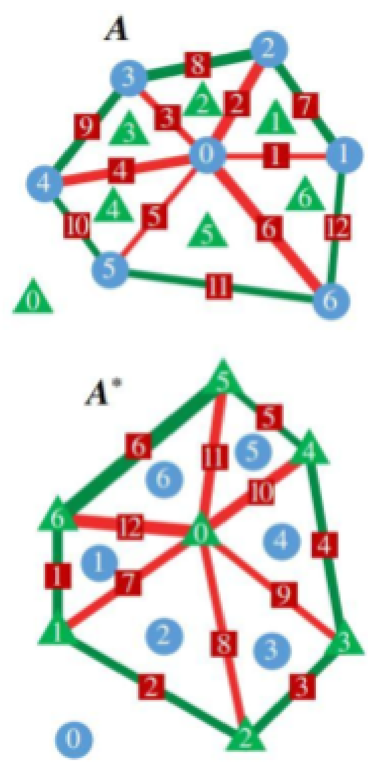

Figure 1. Reciprocal Diagrams [2]

Maxwell reciprocal diagrams can be constructed from the SSS of the original diagram. Two diagrams are said to be reciprocals if (i) there are an equal amount of edges, (ii) the corresponding edges are perpendicular to each other, and (iii) the corresponding edges that form a face in one diagram converge to a site in the other [2]. Figure 1 demonstrates this idea, where A is the original diagram and A* is the reciprocal. Also, to determine the length of the edges in the reciprocal diagram, the SSS can be used. Since the total force on each site is zero, this condition can be written as

$$\sum_{i}^{site\ \alpha} t_i b_i = 0,$$

where t is the tension of the bonds and b is the bond length [2]. As a result, they lead to a closed polygon face in the reciprocal diagram A*. Any linear combination of the SSS gives a t vector, and the magnitude of this vector will give you the length on the reciprocal bond.

Rigidity percolation is defined as the emergence/loss of a spanning rigid cluster as bonds are randomly added/removed. In order for a lattice to be considered rigid, the average amount of bonds per site (⟨z⟩) needs to be greater than or equal to two times the degrees of freedom of each site. For example, a triangular lattice is considered to be rigid because it contains six bonds per site. I mainly focused on the triangular lattice for the majority of my project because it is over constrained and can undergo rigidity percolation.

Coding Process

To begin, I first needed to create a code for the original lattice, which could then be used to create its reciprocal. I used the kagome lattice, shown in Figure 2a., as my direct lattice. I defined the unit cell for it, which consists of three sites and six bonds, and drew the lattice by repeating the unit cell in all directions. Then, I generated the Q matrix for one unit cell, and was then able to find the null space of that matrix, representing the SSS. There were three basis vectors in the null space, implying there are three SSS. To create the reciprocal lattice, which is the dice lattice shown in Figure 2b, I added those basis vectors to define the tension vector, multiplied each bond length by its corresponding tension, and rotated the result by ninety degrees, since the edges are perpendicular. Putting the edges together, I created the unit cell for the reciprocal and graphed the whole lattice. Also, because there are three SSS, there are at least three reciprocal diagrams that can be created. To demonstrate this, I generated a code that used any linear combination of the SSS to create the tension vector, and then went through the same process to determine the reciprocal diagram. Some examples of this are shown in Figure 2c, 2d.

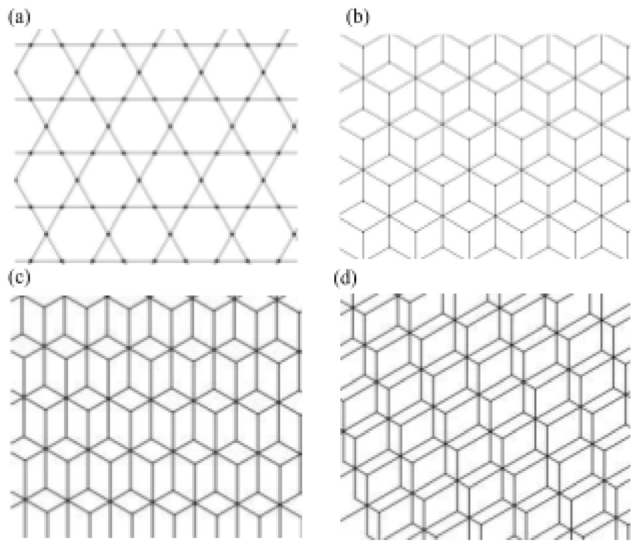

Figure 2. (a) the kagome lattice and its reciprocal diagrams are shown in (b), (c), and (d)

Now that I had a code that created a lattice, I could move on to the next step; creating a lattice that could undergo rigidity percolation. Instead of looking at the individual unit cells, I had to create an algorithm that treated the whole lattice as a single unit cell. Using a similar method as before, I created a small triangular lattice, shown in Figure 3, and

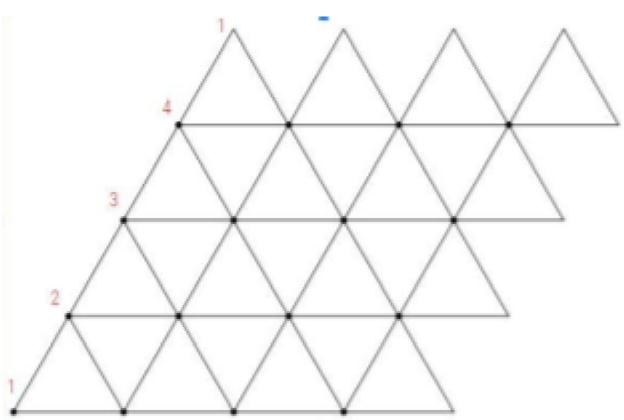

Figure 3. Triangular lattice with periodic boundary conditions

defined periodic boundary conditions. This means that the sites repeat once they reach outside of the lattice, implying that the whole lattice is repeated in all directions. Once this condition was set, I developed a code that created the Q matrix of the whole lattice and found the SSS from that. I was then able to determine the lengths of the corresponding edges in the reciprocal diagram from those SSS. This provided me with a firm foundation to be able to create a code for the reciprocal lattice.

Next Steps

I have been given the amazing opportunity to continue to work on this project with Prof. Mao after this REU program. My next step for this project is to create the reciprocal lattice. To do this, I need to design a code that will recognize the faces in the triangular lattice, and then use that to create the corresponding site in the reciprocal diagram. Once this is developed, I will be able to delete bonds or sites in the triangular lattice and look at how the reciprocal lattice changes. Also, while deleting bonds, I will look at how the SSS change for the triangular lattice.

After we have these tools developed, we will collect data on the evolution of the reciprocal lattice as the real space lattice is undergoing rigidity percolation, and we will extract statistical relations from the dual pair, aiming at obtaining new understandings of the rigidity percolation problem.

References:

[1]Mao, Xiaoming, and Tom C. Lubensky. “Maxwell Lattices and Topological Mechanics.” *Annual Review of Condensed Matter Physics*, vol. 9, no. 1, 2018, pp. 413–433., doi:10.1146/annurev-conmatphys-033117-054235.

[2]Zhou, Di, et al. “Topological Boundary Floppy Modes in Quasicrystals.” *Physical Review X*, vol. 9, no. 2, 2019, doi:10.1103/physrevx.9.021054.

Acknowledgements
I would like to thank Prof. Myron Campbell and Prof. James Liu for running a successful REU program and adapting to the difficult circumstances caused by COVID-19. I would also like to thank Prof. Xiaoming Mao for taking me into her group and teaching me a new concept of physics. Lastly, I would like to thank Harry Liu and others in the Topological Mechanic study group, who helped me throughout the process of designing new code.