Project 2 Team Mission Statement

Team Name

Dots^3

Team Information

Tan Yin Xi	98230564
Woonha Kim	94217566
Zhang Liu	85156883
Cai Lize	90374869

Meeting Schedule

Regular team meeting: Mondays, 4-6pm, at the Library's Group Study Room.
During the Recess Week, we will meet on Friday 28. Time is to be confirmed.
If we are behind schedule, we will add more meeting hours, depending on the availability of the team.

Important Deadlines

Meeting with Prof. Stamps: 28 Feb 15:10pm
Activity Log: 8 March 2020 18:00
Report: 8 March 2020 18:00
Reflection & Peer Assessment: 10 March 2020 at 18:00

Project Approach

Overall Approach

Since there are two parts to this project, we will solve the first part before choosing a specialisation/generalization/variation. For the first part of the project, we will work on our own and discuss our solutions together. After we have developed some results for the first part, we will then carry out individual research on specialisations, generalisations, and/or variations. Then we will come together again to present our research and choose a topic together.

This could apply to almost any two part project.

Timeline

Our aim is to tackle the first part of the project in Week 6. Then we can discuss how to continue with the second part of the project during the Recess Week, before we meet Prof. Stamps on 28 Feb. That will leave us about a week to continue exploration and finish the project.

Before each meeting,

- Each member will attempt to solve the questions, research on their own and prepare to present their findings to other team members.
- If they encounter any difficulties, they can come with questions to clarify during the meeting.
- Must notify the team members beforehand in case of missing a meeting.

During each meeting,

- Present their findings, approaches, research, resources, and any other ideas relevant to the project;
- Clarify any questions they may have;
- Take photos of important results;
- Document the progress on the Activity Log.

After each meeting,

- Members will individually work on any areas that we were stuck at during the meeting.
- We will review the work we did.
- Continue researching and gathering ideas for the second part of the assignment.
- We will delegate what work we discussed during the meeting that has to be written in Overleaf equally

Disagreement

If any disagreements arise, we will ensure that we hear from all the teammates and discuss with the entire team. When we cannot agree on a particular answer, we will postpone the discussion and bring the questions to Prof. Stamps during office hours.
We will ensure that everyone agrees on a certain solution before moving on and that we are willing to help each other through any misunderstandings.

Writing

We will use Google Drive for the Activity Log and Overleaf to write our Report.

Team Guidelines

- We will listen to each other patiently and respectfully.
- We will not shoot down others' ideas in a disrespectful manner.
- We agree to make an effort in ensuring that everyone is able to speak openly and share their ideas for approaching this project and solving problems.
- We will ensure that every member is at a similar level of understanding before proceeding with the project.
- We will think about the problems independently before our group meetings.
- We will attend each meeting with an open mind.
- We will attend each meeting punctually and respect each other's time. We will let the team know in advance if we have valid reasons for absence.
- If any disagreements arise, we will listen to opposing sides without prejudice.
- We will value every idea because mistakes can be productive too.
- We will all make an effort to conduct our individual research and enquiry into our chosen topic for specialisation, generalisation or variation.

Evaluation Criteria

- The team member contributed (less/more than) a fair share of ideas to the project.
- The team member contributed (less/more than) a fair share of organization to the project.
- The team member contributed (less/more than) a fair share of writing to the project.
- The team member participated (less/more effectively) during the problem-solving sessions.
- The team member attended (none/some/all) of the meetings planned for the group.
- The team member engaged (less/more actively) in the specific area of research they are responsible for.

Signatures

Cai Lize ____________________

Tan Yin Xi ____________________

Woonha Kim ____________________

Zhang Liu ____________________

Meeting 1: 17 Feb 2020

- **Meeting duration & location:** 2 hours, 4-6pm, GSR6

- **Members present:** Zhang Liu, Lize, Kim, Yin Xi

- **Summary:**
  - The meeting began with a discussion on part 1(a) of the project. Prior to the meeting, we had begun individually thinking about the question. Thus, during the meeting, we simply compared our answers for 1(a) and agreed that we arrived at the answer.
  - Answer to 1(a): 2^n
    - Since the 2 directions involved are East (E) and West (W) and they are not self-cancelling (that is, they are not in opposite directions), we do not have to worry about the possibility of the travel paths being non-self-avoiding.
    - Since at each end point (including the origin), the subsequent step can have 2 options, either E or W, we conclude that the explicit formula for 1(a) would be 2^n where n is the length of the path. ✓
  - Then we progressed to discussing part 1(b) and began by listing out the non-self-avoiding paths for n<=3.

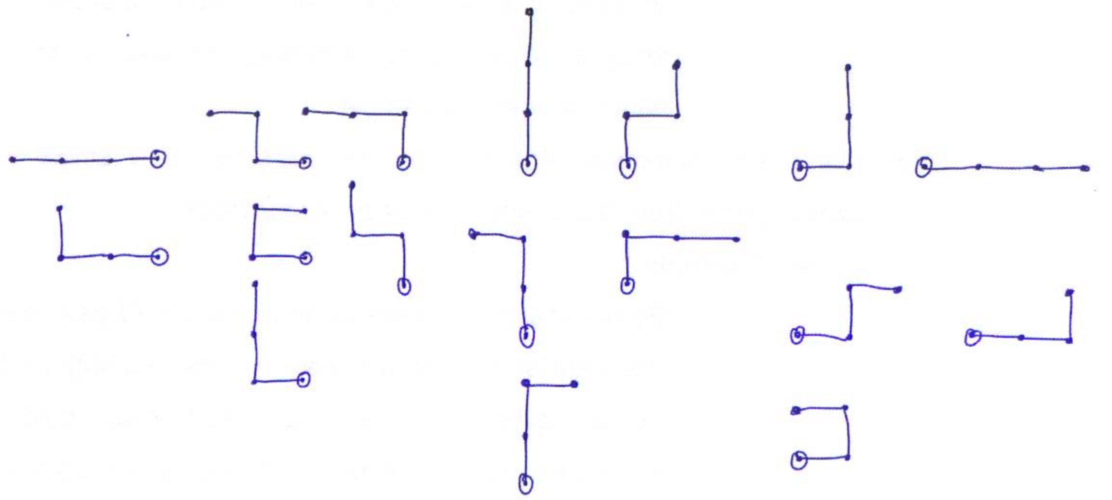


Figure 1: Examples of Self-avoiding Paths

  - We then noticed that unlike part 1(a), each subsequent step is now dependent on the previous one. For example, if we make an E step, we can only take 2 options (either E or N) instead of all 3 options (E, N, W) to make it self-avoiding. But when we take an N step, we can proceed with all three options with no risk of going back to itself.

- This is because now we have both East and West as our options and they travel in the opposite directions.
  - We first considered labelling the directions numerically, labelling opposite directions with opposite signs (e.g. E as +1, and W as -1).
    - **Example:**
      - We thought of this because we first considered a simple case of 2 steps: travelling E then W. In this case, we could identify a non self-avoiding case when one of the coordinates gives 1 - 1=0.
    - **Caveats:**
      - However, we realised that this was problematic as we thought of the following example: travelling E, N, W. In this case, using E as +1 and W as -1 would not be accurate as we would declare the path as non self-avoiding when it still is.
      - More generally, we were also thinking about representing the directions as integers, and making sure that the sum would not add up to 0. However, we realised that this was not feasible because mirror images could be self-avoiding yet add up to 0 OR lattice w/ a stick example
  - We then considered making the non-pairs to be a single block, drawing links from the chapter ***Fibonacci Identities***.
    - **Example:**
      - By non-pairs, we were referring to a E-W pair - we had the idea of making this pair into the rectangular block like in Figure 1.2 of the chapter, so that we could "shift" it around to mimic it being placed at any point in the lattice path.
    - **Caveats:**
      - However, we realised a few flaws:
      1. Possibility of overcounting

2. We were only considering the E-W block in its isolated self, but there would still be a violation if the block before it is a N-W, for example.

- We then considered making N-E strings, and inserting W only when there are 2 consecutive Ns present

	N	E	W
$n=2$	NN	EN	WN
	NE	EE	WW
	NW		
$n=3$	NNN	ENE	WNE
	NNE	ENN	WNW
	NNW	ENW	WNN
	NEN	EEN	WWN
	NEE	EEE	WWW
	NWW		
	NWN		

Figure 2: Strings of Self-avoiding Path for N, E, W

- We thought it would be similar to a question in Project 1, where we wanted to find the number of lattice paths that do not cross the diagonal line. So we tried writing down the paths using alphabets and rearranged the strings (E for East, W for West, N for North)
- This method did not seem to work because it is hard to tell which strings are the bad paths. Why is that?
- After the failed attempts, we began listing out the examples and found a pattern:

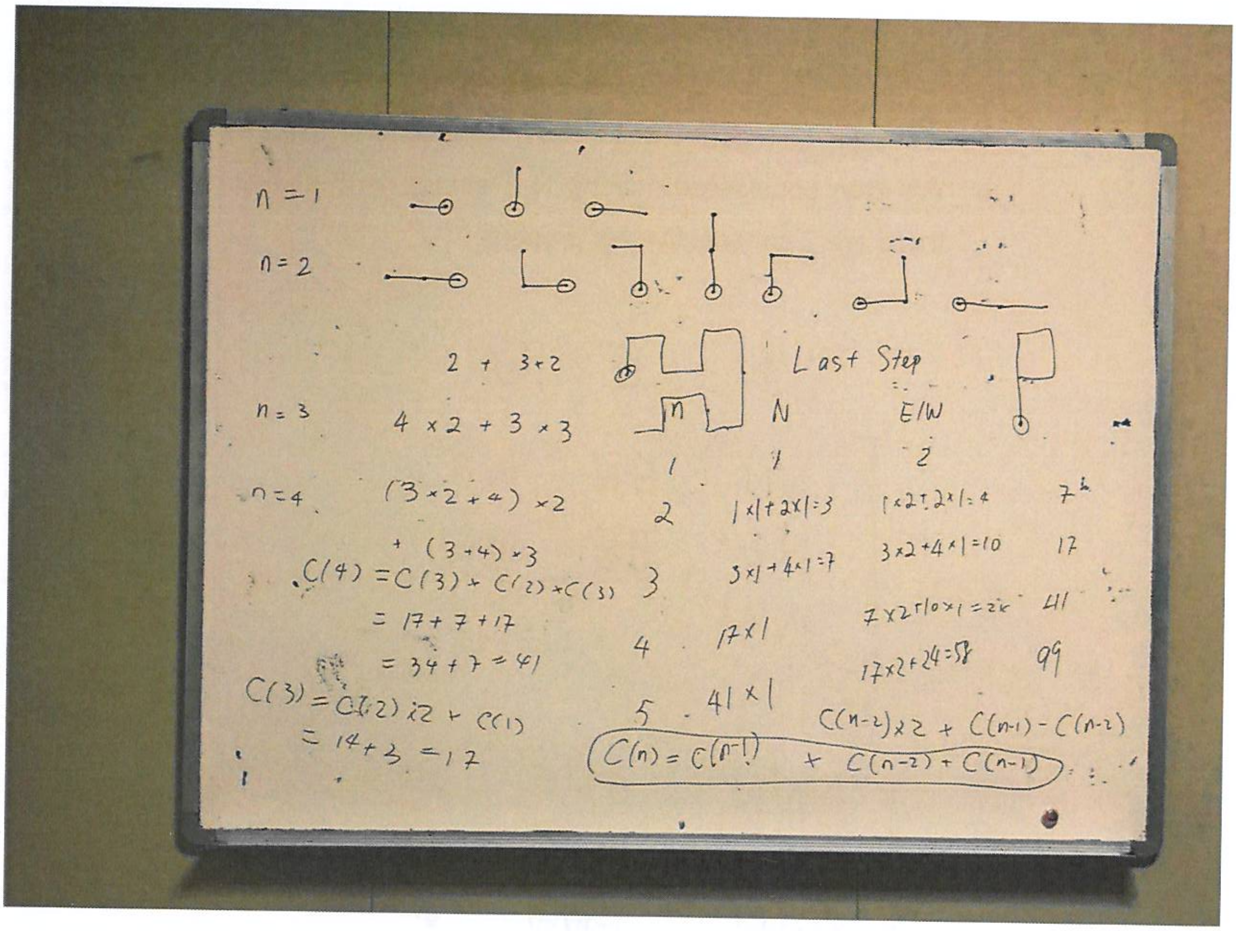


Figure 3: Recurrence Relation for Part 1(b)

Last Step

n	N	E/W	total
1	1	2	3
2	$1\times1+2\times1=3$	$1\times2+2\times1=4$	7
3	$3\times1+4\times1=7$	$3\times2+4\times1=10$	17
4	$7\times1+10\times1=17$	$7\times2+10\times1=24$	41
5	$41\times1=41$	$17\times2+24=58$	99

$$C(n) = C(n-1) + C(n-2)\times 2 + C(n-1)$$
$$= C(n-1)\times 2 + C(n-2)$$

Figure 4: Steps Taken to Derive Recurrence Relation

- **Explanation of Figure 3:**
  - This was the rough working that we did. We began by drawing out the possible pathways, and considering how we can count the total number in a more systematic manner. This is illustrated by the top half of the work done on the whiteboard. The bottom half illustrates our initial process of calculating the total pathways, which is made clearer in figure 4.
  - When the previous step ends with an E or W step, the next step can only be E/N or W/N. However, when the previous step ends with an N step, the next step can take N/E/W.

                - This inspires our counting method.
            - **Explanation of Figure 4:**
                - Figure 4 illustrates the table that was conjured when we thought about the problem in a more systematic manner. As mentioned previously, the thing that made this problem difficult was the fact that each subsequent step was dependent on the previous one. Therefore, we decided to split it up into 2 cases: N and E/W.
                - We split the tables up in this manner because N can be matched with any subsequent step, whilst both E and W are restricted (they cannot be paired with W and E respectively).
                - The way we derived the equations were as follow:
                    - Let C(n) denote the number of self-avoiding paths for n steps.
                    - $C(n) = 2C(n-1) + C(n-2)$
        - Discussion on 1(c)
            - We briefly discussed 1(c). We thought it might be interesting to look at what happens at each end point, and we may get similar results as 1(a).
            - When n<=3, we think the number of paths = 4 * 3^(n-2).
                - At the origin, the path can proceed in all four directions. At subsequent end points, the path can only continue in 3 directions in order to be self-avoiding.
                - However, this formula does not work for n>=4. At the third end point, the situation is different for different paths.
                    - If the n=3 path is N-E-S, the next step can only be E or S to be self-avoiding. So there are only 2 options available.
                    - If the n=3 path is N-E-E, the next step can be N, E or S and the path will still be self-avoiding. So there are 3 options available.
- **Connections to class made:**
    - Fibonacci identities

- Principle of inclusion/exclusion
- Recursive relations
- Combinatorics (n choose k)

- **Questions raised/actionables:**
  - How do we come up with an explicit formula for our recurrence relation?

Meeting 2: 24 Feb 2020

- **Meeting duration & location:** 2 hours, 4-6pm, GSR6
- **Members present:** Zhang Liu, Lize, Yin Xi
- **Summary:**

  This meeting comprised of 3 parts:
  - **Part 1: Solving for part 1(b) after the generating functions, lecture 1:**
    - After the lecture Prof. Stamps gave on 20th Feb, we realised that we could apply the techniques regarding the derivation of the explicit formula for Fibonacci numbers to our current recurrence formula (ref. Figure 1)
    - We noticed that our current recurrence relation had similarities to the Fibonacci sequence:
      - From Figure 2, it is seen that our equation was:

        $$C(n) = 2C(n-1) + C(n-2)$$
      - On the other hand, the Fibonacci sequence is given by:

        $$f\_(n+2) = f\_(n) + f\_(n+1)$$
      - Therefore, we decided to transform our equation to suit the pattern of the Fibonacci's, which was done by replacing n with n+2:

        $$C(n+2) = 2C(n+1) + C(n)$$

        $$\text{Where } C(0) = 1,\ C(1) = 3$$
      - C(0) = 1 makes sense because when there are no steps, the path is just the origin point. The path is self-avoiding.
      - With this new equation, we began applying the same techniques of deriving the explicit formula, arriving at the following conclusion:

$$a_0 = 1, \quad a_1 = 3, \quad a_{n+2} = a_n + 2a_{n+1}$$

$$\text{Let } G(x) = \sum_{n=0}^{\infty} a_n x^n$$

$$G(x) - a_0 - a_1 x = \sum_{n=0}^{\infty} a_{n+2} x^{n+2} = \sum_{n=0}^{\infty} (a_n + 2a_{n+1}) x^{n+2}$$

$$= \sum_{n=0}^{\infty} a_n x^{n+2} + \sum_{n=0}^{\infty} 2a_{n+1} x^{n+2}$$

$$= x^2 G(x) + 2x\left(G(x) - a_0\right)$$

$$G(x) - 1 - 3x = x^2 G(x) + 2x\left(G(x) - 1\right)$$

$$G(x) - x^2 G(x) - 2x\, G(x) = 1 + 3x - 2x = 1 + x$$

$$G(x) = \frac{1 + x}{(1 - 2x - x^2)}$$

$$= \frac{-1 - x}{(1 + \sqrt{2} + x)(1 - \sqrt{2} + x)}$$

$$= \frac{\alpha}{1 + \sqrt{2} + x} + \frac{\beta}{1 - \sqrt{2} + x}$$

Figure 5: Derivation of Explicit Formula for Part 1(b)

$$\alpha(1-\sqrt{2}) + \beta(1+\sqrt{2}) = -1$$

$$(-1-\beta)(1-\sqrt{2}) + \beta(1+\sqrt{2}) = -1$$

$$\cancel{-1} + \sqrt{2} \cancel{-\beta} + \beta\sqrt{2} \cancel{+\beta} + \beta\sqrt{2} = \cancel{-1}$$

$$2\sqrt{2}\beta = -\sqrt{2}$$

$$\beta = \frac{-\sqrt{2}}{2\sqrt{2}} = \frac{\sqrt{2}}{\sqrt{8}} = \sqrt{\frac{1}{4}} = -\frac{1}{2}$$

$$\alpha = -1 - \beta = -1 + \frac{1}{2} = -\frac{1}{2}$$

$$\frac{-\frac{1}{2}}{1+\sqrt{2}+x} + \frac{-\frac{1}{2}}{1-\sqrt{2}+x}$$

$$= \frac{\frac{1-\sqrt{2}}{2}}{1-(1-\sqrt{2})x} + \frac{\frac{1+\sqrt{2}}{2}}{1-(1+\sqrt{2})x}$$

$$\boxed{a_n = \left(\frac{1-\sqrt{2}}{2}\right)(1-\sqrt{2})^n + \frac{1+\sqrt{2}}{2}(1+\sqrt{2})^n}$$

$$n = 0, \quad a_0 = 1$$

$$n = 1, \quad a_1 = \frac{1-2\sqrt{2}+2}{2} + \frac{1+2\sqrt{2}+2}{2}$$

$$= 3$$

$$n = 2, \quad a_2 = 7$$

Figure 6: Derivation of Explicit Formula for Part 1(b) (Continued)

- **Part 2: Deciding on topic for proposal of part 2**

  After solving the explicit formula, we began thinking about the direction we would want to take regarding the second part of the project. We arrived at the following 2 options that we wanted to consult Prof. Stamps with during our upcoming office hours on 28th Feb.

  - Specialisation:
    - We were thinking of restricting the paths to a *N x N* lattice. In this lattice, we would locate a point *(n/2, n/2)* which is the middle of the lattice. By starting at the origin of *(0, 0)*, we wanted to consider how many self-avoiding paths could be constructed to get to the point *(n/2, n/2)* without exceeding the *N x N* space.
  - Variation:
    - We were thinking of adding more directions to the current ones in part 1(c). More specifically, we wanted to add 4 more diagonal directions - North-East, North-West, South-East, South-West.
    - When we were considering this, we thought about the possible polygons that could be constructed since we would be travelling in diagonals.
- **Part 3:** Brainstorming of part 1(c)
  - We decided to try approaching this question by considering the types of bad paths that can be constructed.
  - It seems there are two types of bad paths: closed polygon and a "flag" shape.
    - for n >= 4 and n is even, it is possible to form a bad path by creating a closed polygon
    - When n>=4 and n is odd or even, the bad paths consists of a "flag" shape (a path adding to a polygon).

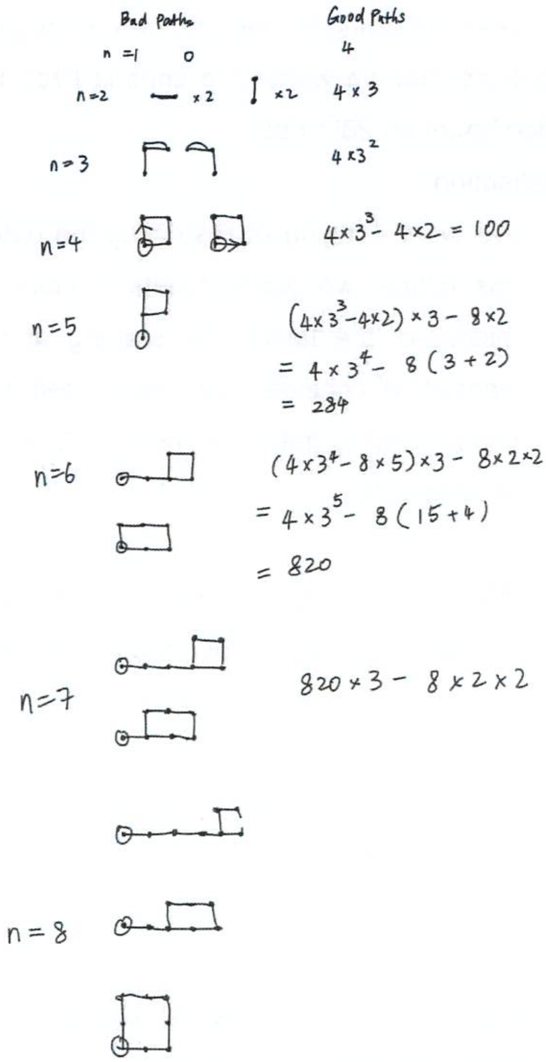


Figure 7: Diagram of Possible Bad Paths

- We started looking at the closed polygons that can be formed for even n>=4.

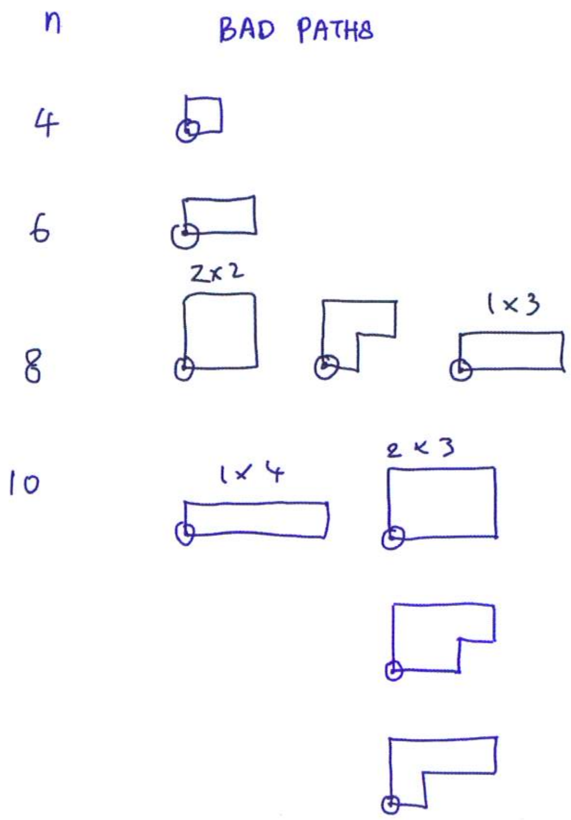


Figure 8: Diagram of Possible Bad Paths

- We notice that each of these shapes can
    - Flip in the x-axis
    - Flip in the y-axis
    - Flip in y=x or y=-x
    - Rotate 90/180/270 degrees around the origin
- Since the path is a closed shape, the starting point can be shifted.
- Basically, there are many variations for each closed path. So a more methodological approach is required to count the number of paths.

- **Connections to class made:**
    - Generating functions, part 1

- **Questions raised/actionables:**

- Which proposal should we conduct our investigation upon?
    - More specifically, which query is more interesting and feasible in the remaining time we have left.?
- How do we exactly count/think about all the bad paths that could be made for 1(c)?
- To begin formally typing the derivation of the explicit formula from the recurrence relation seen in Figure 3 into overleaf.

Meeting 3.1: 28 Feb 2020

- **Meeting duration & location:** 2 hours, 1-3pm, GSR6

- **Members present:** Zhang Liu, Lize, Kim, Yin Xi

- **Summary:**

  This meeting consisted of 2 parts.

  Part 1: Thinking of obstructions for part 1(c)

  - Counting obstructions:
    - Following our previous meeting, we began this meeting by considering the number of obstructions (bad paths) that could be formed for an integer n.

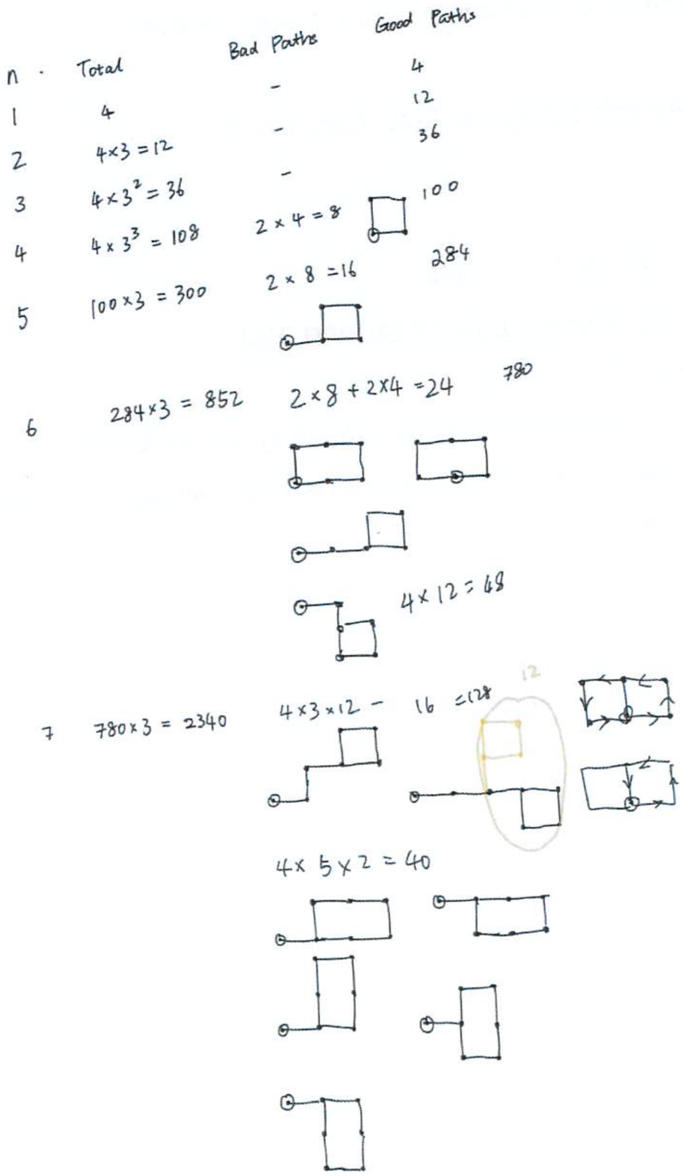


Figure 9: Counting the Paths

- For n=1, the number of self-avoiding paths is 4, that is taking one step in each of the directions.
- For n >= 1, we do not consider the path that goes back to the previous point, that is a sequence of 2 steps in the opposite directions. Thus, for n>=1, the total number of possible paths is the number of (n-1)-step self-avoiding paths multiplied by 3, excluding the direction opposite to the previous step.

- Thus, without considering the path that goes back to itself immediately, for 1<= n <= 3, there are no bad paths.
- The first example of a bad path appears when n=4. For n = 4, the only bad path that can be constructed is a unit square. Starting from the origin, 8 different square paths can be formed.

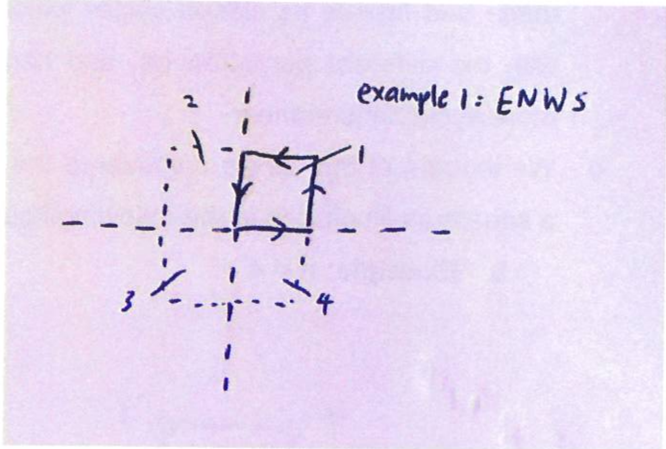


Figure 10: Reflections of a Square

  - In figure 10, we first consider square 1 and notice that there are 2 possible paths that can form square 1. Starting from the origin, one path can go clockwise (NESW) and another can go anticlockwise (ENWS). Thus, there are 2 paths that can form the unit square in the first quadrant.
  - Additionally, we notice that for the square in the first quadrant can be reflected in x-axis, y-axis, or about the origin, forming squares 4, 2, 3 respectively. That means that for each reflected square, there are 2 paths.
  - Since there are 4 reflections, we have a total of 4 x 2 = 8 bad paths.
- When n=5, instead of a closed polygon, the bad path is now a "flag" (a "stick" plus a square).
  - Just like there are two ways to form a particular square, there are two ways to form a particular "flag". But instead of having 4 reflections, the flag has 8 different reflections.
  - Thus, we have in total 2 x 8 = 16 bad paths.
- However, we began to face problems when n = 6.
  - When n=6, the bad path can be either a closed rectangle or a flag. There are a lot more variations of the bad paths.

- As n increases to 7, there are even more possible bad paths, which seem difficult to be counted methodologically.

- Possible counting methods:
    - Upon realising that it might be quite difficult to conjure up the different "bad" figures as n gets larger, we considered the possibility of treating these bad figures as closed cycles which would allow us to come up with the different permutations, and hence different bad paths, in a more systematic manner.
    - We thought of this as we considered the bad paths for n = 4, which is a square as illustrated in the following figure.
        - **Example: n = 4**

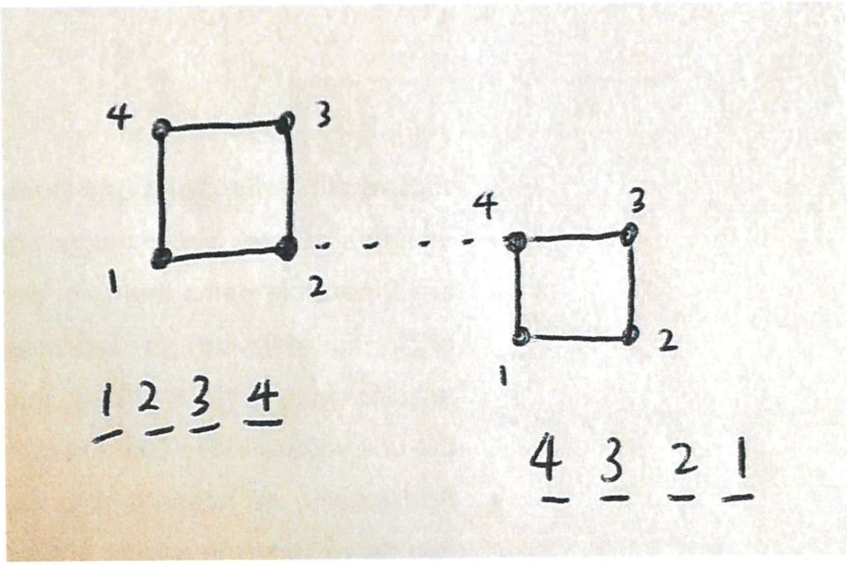


Figure 11: Permutations for n = 4

        - In order to mimic the reflection that would happen in our 2 dimensional lattice space, we can permute the arrangement of the numbers differently to still obtain the same "bad" figure.
        - From figure 11, we notice that one way of reading this permutation can be by reading the dots as 1234, which corresponds to ENWS. Similarly, we can "reflect" this, and have the permutation be 4321 which corresponds to ESWN instead.
        - This led us to question if it was possible to use permutations to count the number of bad paths instead.

- However, we were quite unsure about how to extrapolate this idea of a cycle beyond n = 4, as it would only work best if the bad path was a closed cycle like the square in our example.

Part 2: Developing proposal for part 2

- Building off the previous proposals we had regarding part 2 of the project, we decided to focus on the specialisation - limiting the lattice area to an *N x N* grid with the starting point at (0,0) and the end point at (n/2,n/2).
  - Initially we wanted to know if we can count the number of self-avoiding paths to a particular point in R^2 without other restrictions. We soon realized that there could be infinitely many self-avoiding paths to a particular point (except for the origin) in R^2 starting from the origin because the path can take infinitely many detours before reaching the desired point.
  - As such, we constrained the grid to *N x N.*
  - **Example: n = 4**

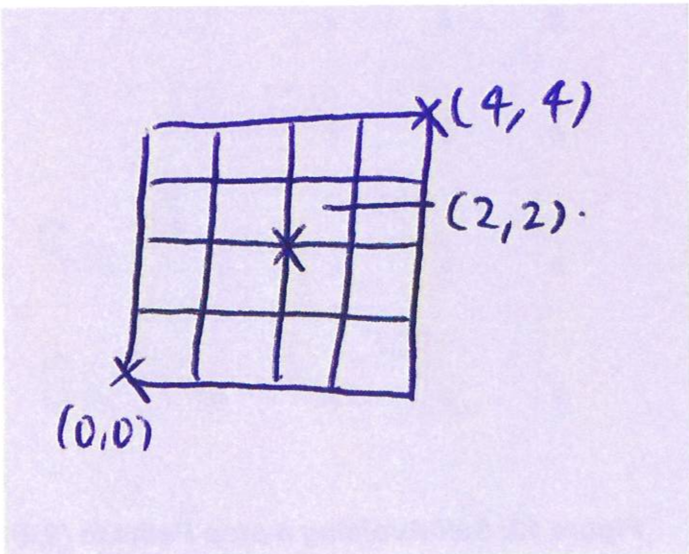


Figure 12: 4 x 4 grid

In this example, we want to limit the lattice space to be from (0,0) to (4,4), with the intention of the lattice path starting from (0,0) and ending up at (2,2) whilst being self-avoiding.

  - We decided to use the example of n = 4, as it was more interesting than using n = 2 (as counting the number of bad paths was rather straightforward for this) yet still relatively manageable for us to draw the possible paths out.

- We noticed that the minimum number of steps that have to be taken to reach (2,2) is 4, as we have to take at least 2 up and 2 right steps from the point (0,0) to reach (2,2). We also noticed that the path taken to reach (n/2,n/2) always has to be an even number of steps.
- Thus, we tried to list out the number of bad paths for the different even numbers of n, starting from n = 4 for this case.
- The process of listing out the possible bad paths was similar to 1(c) - we were thus stuck as we didn't know how to proceed in counting the number of bad paths in a systematic and effective manner.
- As one can see, when n=6, there are many possible self-avoiding paths to (2,2) in a 4 by 4 grid.

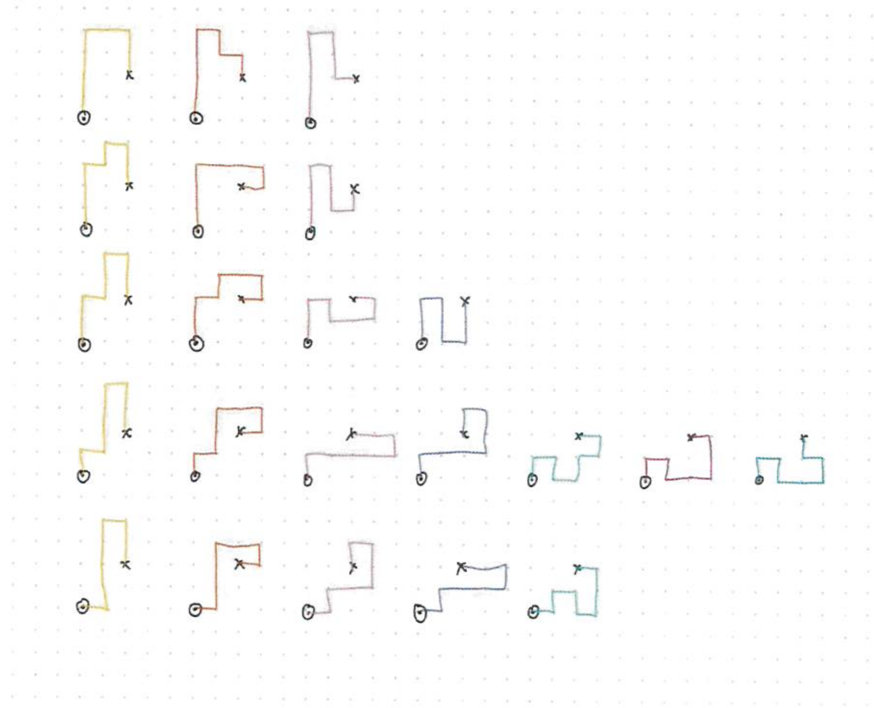


Figure 13: Self-Avoiding 6-step Paths to (2,2) from (0,0) in a 4 by 4 Grid

- We thus realised that this proposal proved to be much more complicated than we anticipated - we had initially thought that it would be more manageable since the first step of this path is always restricted to 2 possible options (either East or North), but realised that it was quite complicated as the paths could loop around the entire *N x N* grid before reaching the middle point, as illustrated in figure 14.
  - **Example: Long pathway threading through most of *N x N* grid**

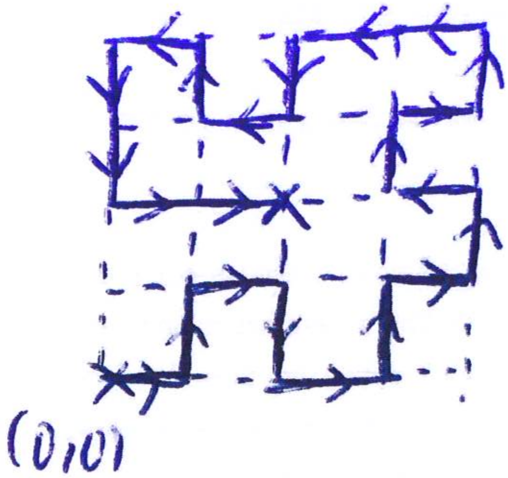


Figure 14: Long Pathway for 4 x 4 grid

- **Connections to class made:**
  - Cycles & permutations

- **Questions raised/actionables:**
  - Meeting with Prof. Stamps to finalise proposal topic

Meeting 3.2: 28 Feb 2020

- **Meeting duration & location:** ½ hour, 3-3:30pm, Prof. Stamps' Office

- **Members present:** Zhang Liu, Lize, Kim, Yin Xi

- **Summary:**

  We met Prof Stamps to discuss our progress and received feedback on both Part 1(c) and Part 2. There were two main takeaways from the meeting.

1. Even if we don't obtain an explicit formula for part 1(c), we can still discuss our observations of the "bad paths", i.e. the obstructions. We can
   a. Identify the possible minimal (no redundancy/over-counting) obstructions, e.g., a path that contains a segment where there are as many E's as W's and as many N's and S's.
   b. Characterize each type of obstructions: express each obstruction in the simplest way possible. Here we can make use of the concept of cycles to characterize the obstruction to make it more systematic, such as "a bad path containing a n-cycle". The n in n-cycle represents the number of steps the path takes when it returns to a previous point. For example, a EW or NS path is a 2-cycle. A NESW is a 4-cycle.
   c. As for the minimum, we can also consider the 'substring' of the string representing the path, e.g., "contain no shorter string that has this particular property."
   d. Count the number of cycles contained in a n-step path. Here we can potentially make use of the Principle of Inclusion-Exclusion, e.g., we can count the number of 2-cycles in a n-step, and eventually reach a formula similar to the alternating sum.
2. We also discussed possible extensions of Part 1 with Prof Stamps.
   a. Upon hearing our idea of finding the number of self-avoiding paths to the midpoint in an n by n grid, he suggested the midpoint was rather arbitrary and it restricted our n to be only the even numbers. Overall the extension sounded too specific.
   b. Instead, Prof. Stamps suggested we can consider the number of self-avoiding paths to go from (0,0) to (n,n) in an n by n grid.
   c. Later we can extend it to an a by b rectangular grid and count the number of self-avoiding paths from (0,0) to (a,b).

3. We briefly mentioned another idea for the extension, which is to add the diagonal directions, but all agreed that it would be too challenging since we had not solved Part 1 (c).

- **Connections to class made:**

  With suggestions from Prof Stamps, we were able to have a better understanding of how to make connections to what we've learnt in class. To be specific, we now know that we can characterize the different types of obstructions in terms of i-cycles and count the number of obstructions of each type using the Principle of Inclusion-Exclusion.

- **Questions raised/actionables:**

  After this consultation, we were clearer on the direction we should be heading towards. In the next meeting, we can focus our attention on the following:

1. Come up with a list of all possible types of minimal obstructions (as explained above);
2. Characterize each type (potentially useful tool: i-cycle);
3. Count each type (potentially useful tool: Principle of Inclusion-Exclusion).

Meeting 4: 2 Mar 2020

- **Meeting duration & location:** 2 hours, 4-6pm, Group Study Room 6

- **Members present:** Zhang Liu, Lize, Kim, Yin Xi

- **Summary:**
    - Firstly, we continued our investigation of Part 2, taking Prof. Stamps' suggestions from the previous meeting
        - We decided to study the number of self-avoiding paths from (0,0) to (N,N) in an $N \times N$ grid and extend our observation to an $a \times b$ grid.
        - Adopting the same approach as Part 1, we first looked at the case where we are only allowed two positive directions, namely East and North.
            - This is similar to Portfolio Problem 3. √
            - For example, when $N = 3$, it takes exactly 3 E steps and 3 North steps to reach (3,3). The number of self-avoiding paths is then $6!/(3! \times 3!) = 20$.
            - In general, the number of self-avoiding paths from (0,0) to (N,N) with only $\{N, E\}$ directions is $(2N)!/N! \times N!$
            - Now we can extend the result to an $a \times b$ grid. The number of self-avoiding paths from (0,0) to (a,b) with only {N,E} directions is $(a+b)!/(a! \times b!)$. It takes exactly a E's and b N's to reach (a, b).

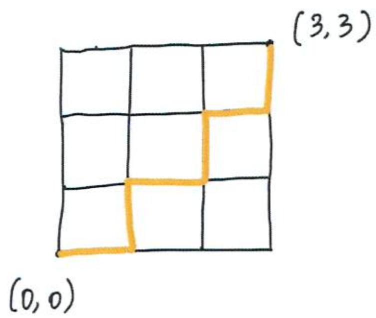


Figure 15: Illustration of Possible Pathway from (0,0) to (3,3)

- Then we moved on to the case where we are allowed three directions.

- We realized that it matters whether we choose {N,E,W} or {N,S,E} as our available steps. In other words, whether we add the negative step in the horizontal or vertical direction changes the way we count the number of self-avoiding paths in an $a \times b$ grid.
- We still started by looking at a 3×3 grid.

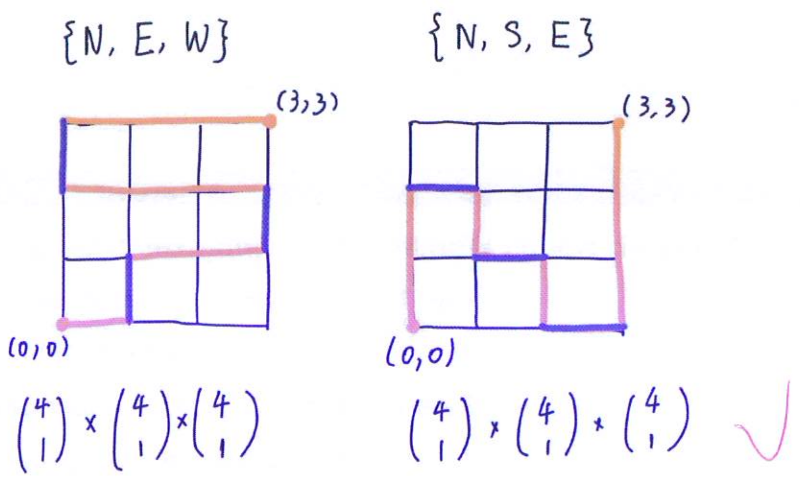


Figure 16: Counting of Total Pathways

  - After some explorations we realized that if we have {N,E,W}, we do not know how many E's and W's are involved, but we know for sure that we will require exactly 3 N's as we do not have S's. Moreover, the 3 N's have to be in 3 different rows. Most importantly, a particular set of the North steps on the grid gives us a unique self-avoiding path from (0,0) to (3,3) with {N,E,W} steps.
  - In each row, we have 4 choices of the North Step (highlighted in purple). So in total we have $\binom{4}{1}^3 = 4^3$ self-avoiding paths.
  - Similarly, if we have {N,S,E} steps available, we are fixing the E's instead of the N's (highlighted in purple). Therefore, we also have 4^3 self-avoiding paths.

- In general, in an $N \times N$ grid, we will have $(N+1)^N$ self-avoiding paths from (0,0) to (N,N) with three directions (two of which are in the opposite directions).

- The same approach can be applied to an $a \times b$ grid, but the directions will change the final result.

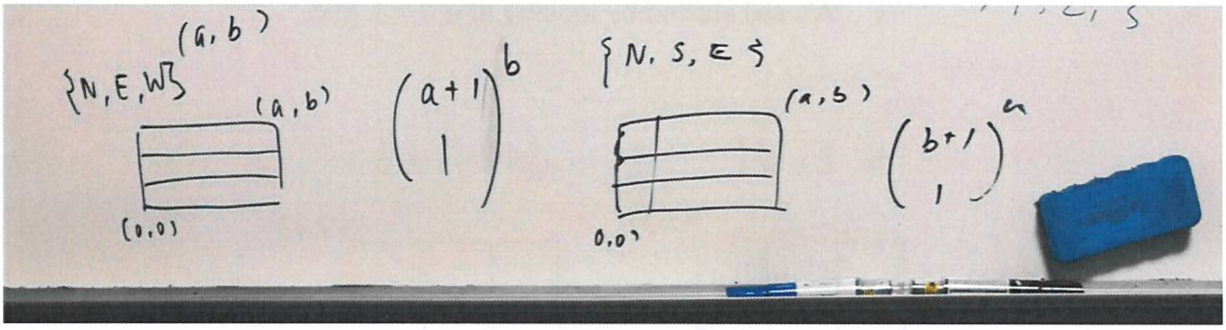


Figure 17: Rough Working of Counting Total Pathways

- If we have {N,S,E}, the number of self-avoiding paths is $(b+1)^a$ because we are fixing the horizontal steps.
- If we have {N,E,W}, the number of self-avoiding paths is $(a+1)^b$ because we are fixing the vertical steps.

- Then, we moved on to consider the number of self-avoiding paths with all four possible steps.
  - However, similar to 1(c), the self-avoiding paths became much more complicated when we are allowed all 4 directions.
  - One observation we made was if we wanted to utilize all four directions, we could start from the base case, which is the path along the edges of the grid (shown in black), and nudge it in both on the horizontal side and the vertical side (shown in orange).

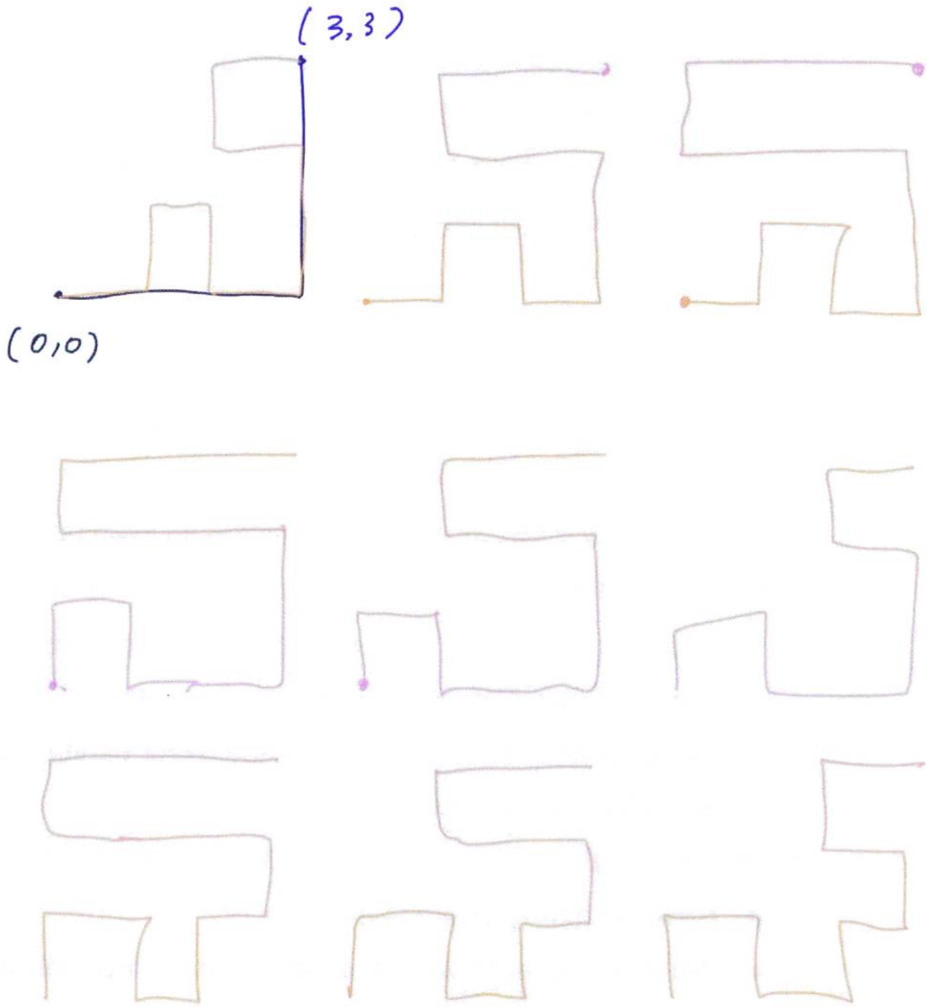


Figure 18: Listing Possible Pathways

- However, one problem we encountered with these "nudges" is we could not develop a way of counting them systematically.
- Another observation that extended from this was we could not have a self-avoiding path utilizing all 4 directions in a 2×2 grid because we cannot accomodate for 2 nudges in such a grid.

- Secondly, we continued with our investigation into Part 1(c).
  - Prof. Stamps suggested we could categorize the obstructions and study a specific kind of obstructions. We decided to make some observations about the "n-cycles". We define "n-cycles" to be a segment of the path where in total there are n steps, and the number

of N's equals the number of S's, the number of E's equals the number of W's.

- Visually, an n-cycle will be a closed polygon on the grid.
- Some intuitive observations are:
    - The smallest cycle has to be a 2-cycle, where the path goes back to its beginning point. There are only 4 2-cycles: NS, SN, EW, and WE.
    - The first nontrivial case will be a 4-cycle, which looks like a square on the grid.
    - It makes sense that n has to be even for a cycle to be formed. If we let the number of E's be h and the number of N's be v, we have $n = 2h + 2v = 2(h + v)$, which is an even number.
    - This is akin to partitioning an even number n into two even numbers $2h$ and $2v$. We thus thought of counting the strong partitions, but that proved to be less insightful. The number of strong partitions does not tell us the exact partitioning, which would be important for permutation later.
- At first, we thought the number of n-cycles would be $n!/((h!)^2 \times (v!)^2)$
    - However, this is an overestimation.
    - We need to exclude the cases where it includes a smaller cycle.
    - For example, for $n = 4$, we cannot have NSEW. That does not form a square. It's merely a pair of 2-cycles.
    - For n=6, we cannot have 2-cycles or 4-cycles within the strings.

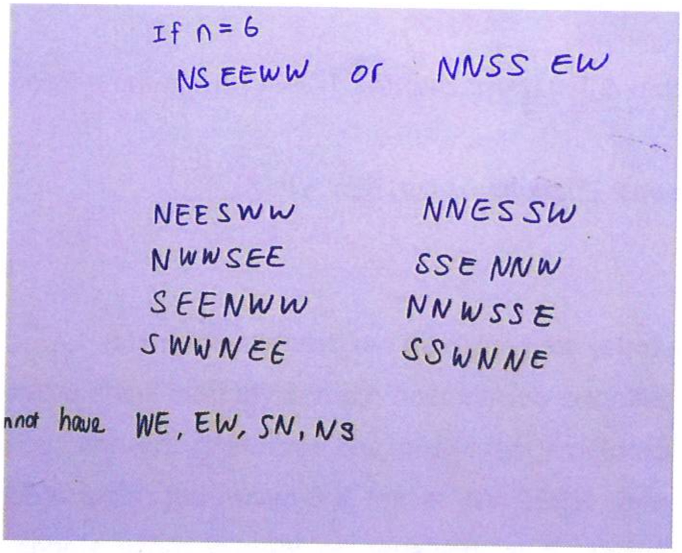


Figure 19: Strings of Possible Pathways to Form Closed Cycle/Closed Cycle Within

  - It seems that the first step and the last step also have to be in different directions (horizontal vs. vertical). The cycle contains a smaller cycle.
- Lastly, we started making what we had solved so far into a report on Overleaf.

- **Connections to class made:**
  - Portfolio Problem 3

- **Questions raised/actionables:**
  - Begin writing into Overleaf

Meeting 5: 5 Mar 2020

- **Meeting duration & location:** 2 hours, 1-3pm, Programme Room

- **Members present:** Zhang Liu, Lize, Kim, Yin Xi

- **Summary:**
  - In class today we continued our discussion on 1(c).
    - With the permutation method we tried in the previous meeting, we had problems identifying and excluding the non-cycles. For instance, for n=6, NSEEWW is not a 6-cycle, but rather it contains two separate cycles. It is, however, to identify those strings using the previous formula.
  - We asked Prof. Stamps how we could use the Inclusion-Exclusion principle to make sure we are not overcounting or undercounting.
    - Prof. Stamps suggested that we could fix the length of cycles, and count the number of ways it can appear in a n-length path.
    - We first started studying the possible numbers and positions of 2-cycles in a 4-length path.
      - If we have at least one 2-cycle, there are three positions for the 2-cycle to appear.
      - There are 4 different 2-cycles, namely EW, WE, NS, and SN.

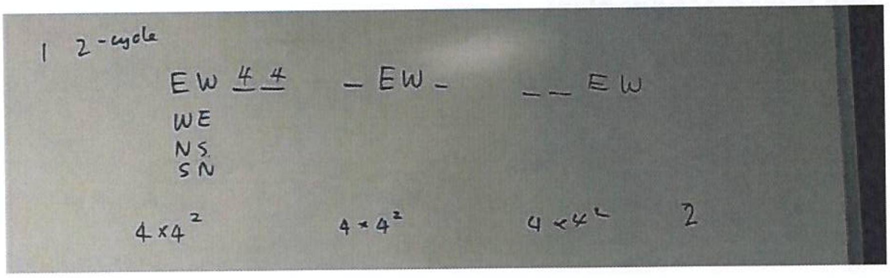


Figure 20: Counting 2-cycles

      - In each case, we are left with two steps to fill in to form a length 4 path.
      - Therefore, in each case there are 4×4^2 paths. In total we have $3\times4\times4^2$ cases of a length 4 path that contains a two-cycle.

- Then we moved on to investigate the number of paths that contain 2 2-cycles.

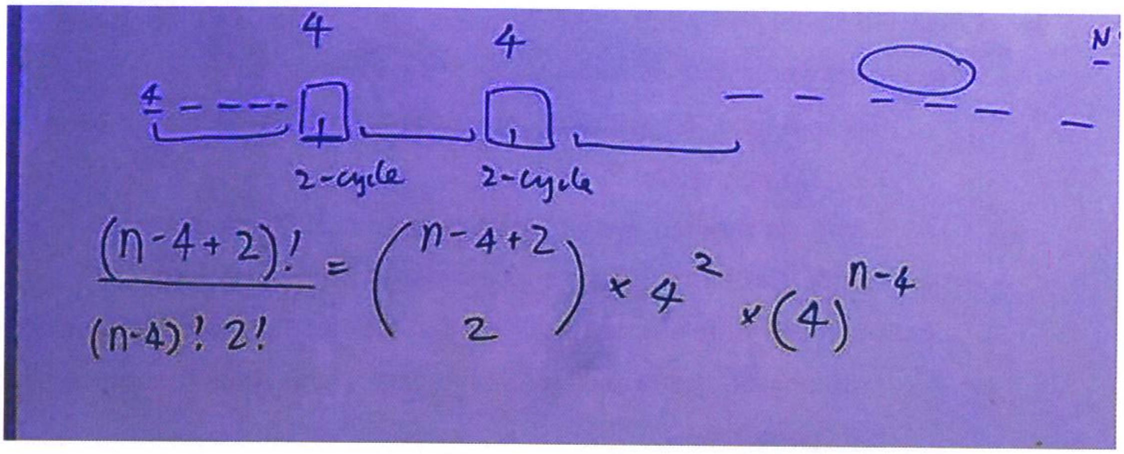


Figure 21: Counting 2-cycles that Lie Within a String

- We realize the number of paths that contain at least 2 2-cycles can be counted by first fixing the 2 2-cycles and then filling the paths before and after those 2 cycles.
- If the total length of paths is n, we have $(n-4+2)!/((n-4)!2!)$ different arrangements that contain at least 2 2-cycles.
- Each of the 2-cycles, there are 4 different choices. For each of the remaining (n-4) steps, there are 4 choices. Thus, the total number is $\binom{n-4+2}{2} \times 4^2 \times 4^{(n-4)}$

- We then made the conjecture that the number of length n paths that contain k 2-cycles is

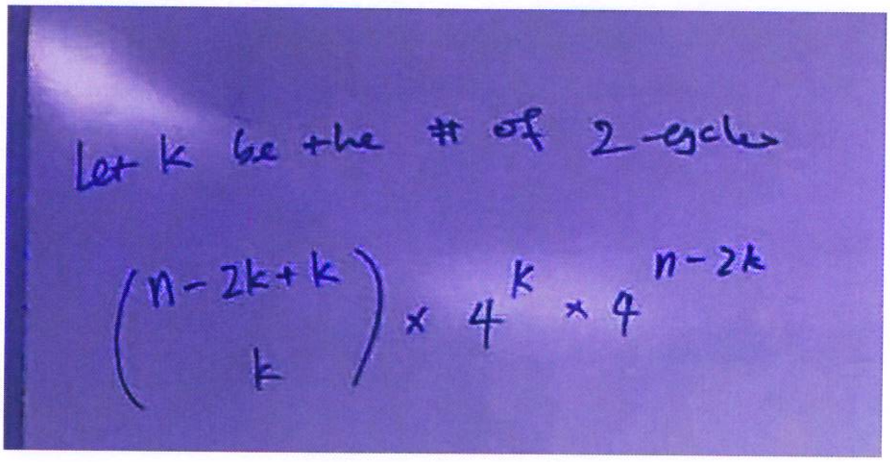


Figure 22: Conjecture

- However, we soon realize this counting method is problematic.

- When we count the number of 2-cycles, we overcounted the cases where there are consecutive 2-cycles.
    - For example, for n=5, we could have NSNSN, this is a series of 4 2-cycles connected to each other.
    - Using our method, we would've overcounted cases like those.
    - Hence, the formula is an overestimation of the bad paths.
    - From the problem, we realized that we need to not only consider the positions of cycles, but also take care of the steps that come next to the cycle.

- **Connections to class made:**
    - Portfolio Problem 3

- **Questions raised/actionables:**
    - Limiting our attention to the kinds of obstructions we wanted to consider: in this meeting, we had tried to consider counting obstructions that have cycles within them. However, we soon realised that it is rather complicated, and hence sought to look at strings that are a whole cycle itself. An example of a "full-cycle" string is ENWS which forms a closed square.

Meeting 6: 6 Mar 2020

- **Meeting duration & location:** 1.5 hours, 10:00-11:30am, Prof. Stamps' Office

- **Members present:** Zhang Liu, Kim, Yin Xi

- **Summary:**
    - We met up with Prof. with the intention of verifying the following few ideas we had in mind:
        - ***Method for E, N, W of Part 2***
            - We went through our solution regarding this part of the prompt with Prof. Essentially, we wanted to check whether our idea of fixing the horizontal and vertical lines of choice can help us determine the total number of possible paths available to get from (0,0) to (a,b).
            - It seems like our idea is generally sound, with the reminder that we are able to do this because we have restricted the space of the lattice from (0,0) to (a,b). If we did not restrict this space, we would have an infinite number of solutions.
            - It was also here that we noticed it would be very difficult for us to extrapolate this same logic when considering E, N, W, S for this investigation later on, as our idea behind choosing to focus/"choose" the horizontal or vertical lines were contingent on the possible options we were working with: for example, because we were working with E, N, W, it was impossible for the vertical direction to "go back and forth". Therefore, we could focus on counting the total vertical steps to get to (a,b) from (0,0). If, on the other hand, we were working with E, N, S, then we would by the same logic have to focus on counting the horizontal lines instead.
        - ***Setting the "k", for 4^(n-k)***
            - Part 1(c): Inclusion/exclusion of cycles
    - We begin by reviewing the possible options that we could take, starting off with eliminating "full" cycles, which are paths that do not contain a cycle within a cycle.

- We went through the following criterias for this to happen, as well as the possible violations that might still take place:
  - **<u>Criteria</u>** - N = S & E = W: in order to form this "full" path, the total number of Norths have to equal the total number of Souths, and the total number of Easts have to be equal to the total number of Wests
    - **<u>Possible violations:</u>** One violation to this idea that we talked about was the path East-East-West-West
  - **<u>Principle of inclusion/exclusion:</u>** Prof. suggested that we could perform an inclusion/exclusion counting using the following ideas:
    - Step 1: Count all the possible cycles, 4^n
    - Step 2: Throw away the things that aren't cycles (they contain smaller cycles)
    - Step 3: Plus back the full cycles? (so that we exclude cases that have cycles within cycle)
  - **<u>Proposed alternative:</u>**
    - Prof. also suggested that we could generate all cycles recursively. He questioned that if all the "full-cycle" paths have to be of an even length, then we can think of how the subsequent cycles (e.g. n = 6 to n = 8) are formed from either inserting a pair of NS edges or a pair of EW edges. If so, we question how we could possibly count this
      - **<u>Possible violation:</u>**
      - We were thinking of the example where we are giving a simple n = 4 string of ENWS. We thought of how we could add in the NS pair into this string which we would re-express as the following for better visualisation: 1E2N3W4S. We realised that if we added N at 2, or N at 3, we would end up with the exact same solution. There thus runs the risk of overcounting that we have to know how to mathematically account for.

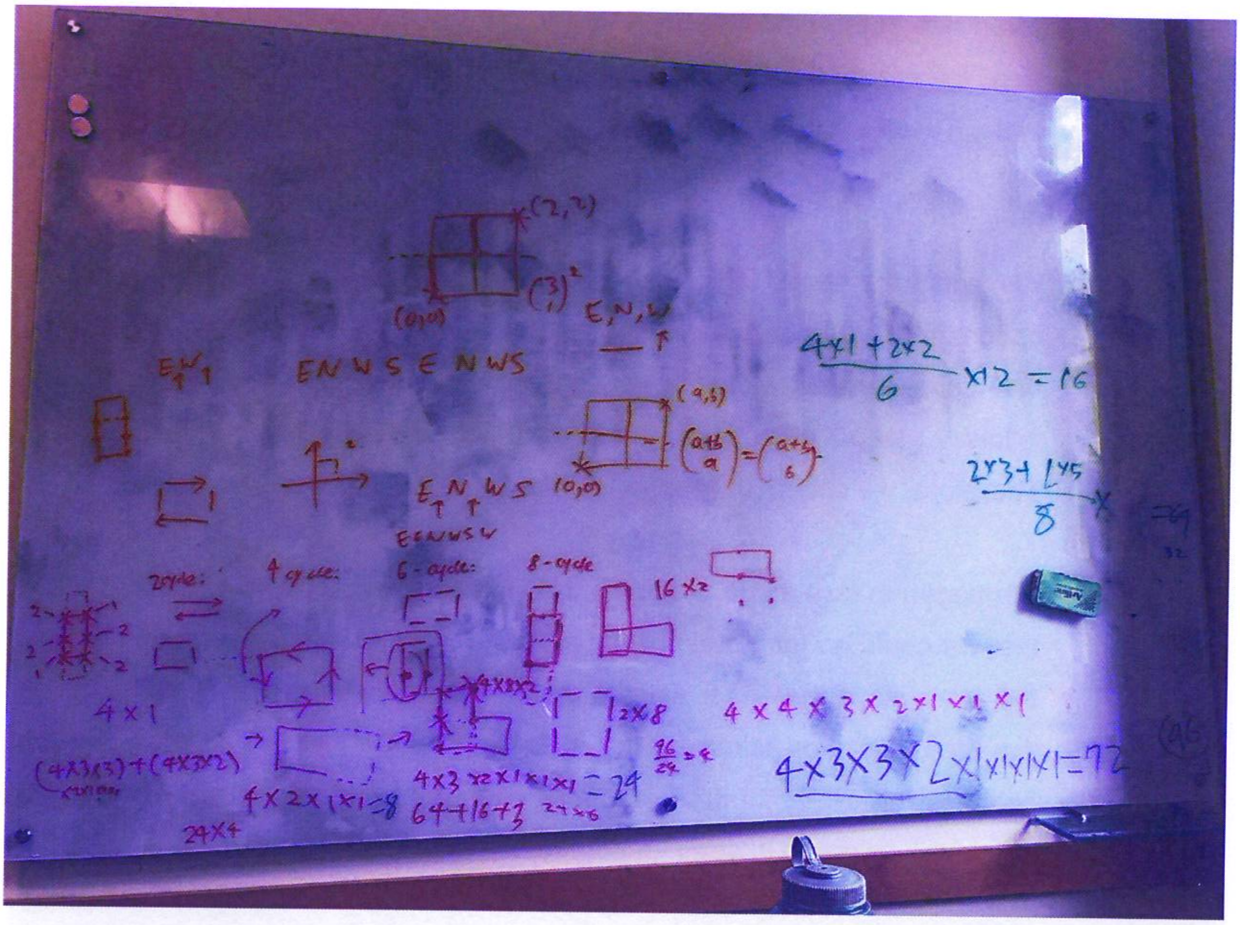


Figure 23: Rough Working in Counting Bad Paths for 1(c)

At the bottom of this image, we were trying to figure out the total number of “full-cycle” obstructions that could be constructed by each even n.

1. We started out with n = 2:
    a. When n = 2, there are 2 steps that have to be taken to form that length of path. The first step has 4 options. To form a closed cycle, the second step must take a step in the direction opposite to the first step, which essentially leaves it limited to only 1 direction. Therefore, the total number of paths that we obtained is 4 times 1 = 4.
2. n = 4:
    a. When n = 4, there are 4 steps that have to be taken to form the closed square shape. Similarly, the first step has 4 possible directions. Since we are looking to form a closed cycle/square, the second step is left with 2 possible options (if the first step taken was vertical, the second step has both horizontal options; similarly, if the first step taken was horizontal, then the second step

taken has both vertical options to choose from). After selecting this second step, the rest of the path has no options in order to form the required closed cycle. Therefore, the total number of paths that we obtained is 8 (4 x 2 x 1 x 1).

3. n = 6:
   a. We apply the same logic to n = 6. For the first step, there are 4 possible options that could be taken. After this point, there are now 3 possible options that can be taken. After which, only 2 options remain and thereafter, the following steps only have 1 option if we want to obtain this full cycle. Therefore, we obtain 24 (4 x 3 x 2 x 1 x 1 x 1)
4. n = 8:
   a. We began to face some problems the moment we reached n = 8. We realised that we could no longer simply multiply by looking at the consequent steps as there are steps thereafter whose total number of options available depend on the step that was taken before it. Therefore, we decided to draw out the possible full cycles that could occur and count it individually. We therefore obtain the following images for counting:

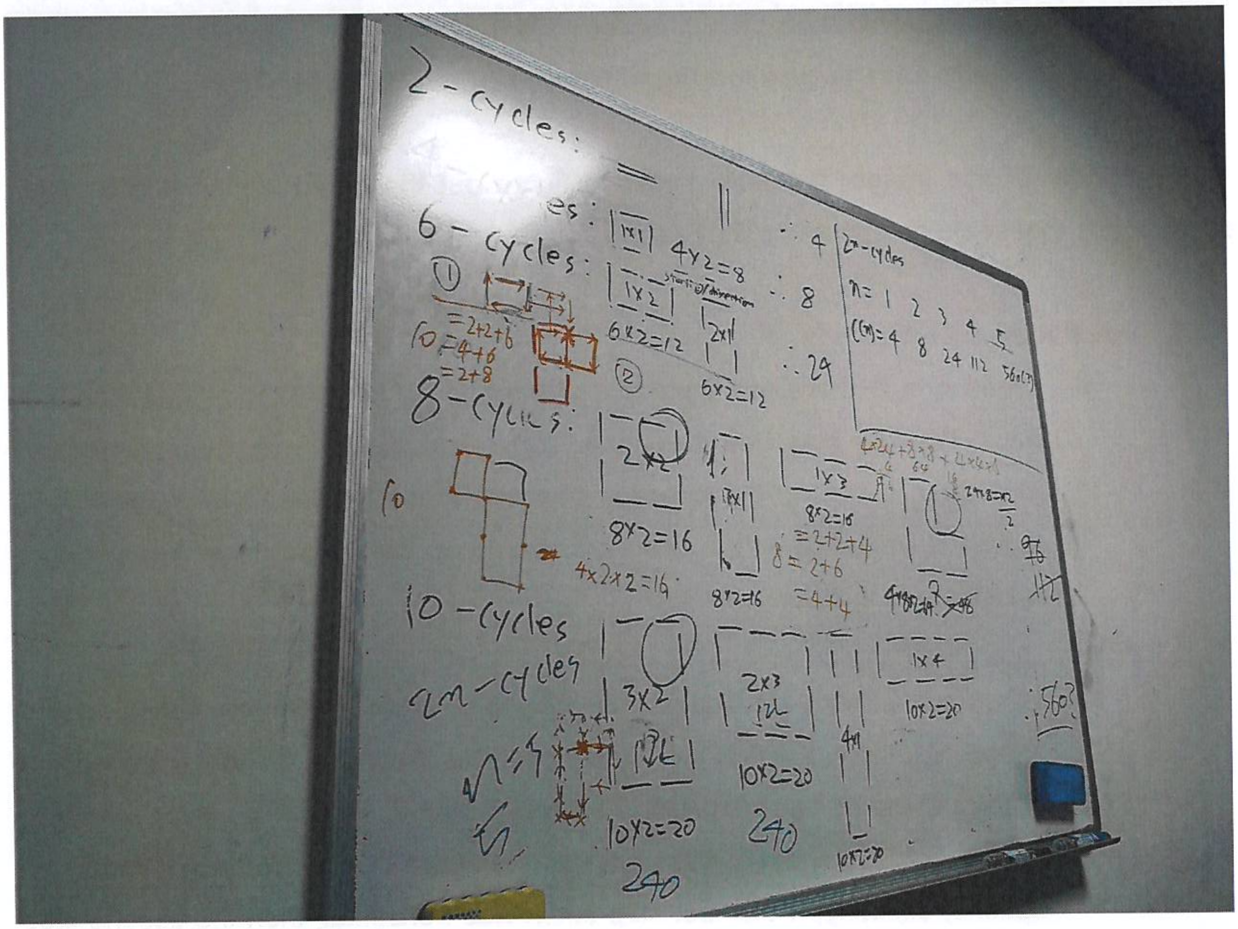


Figure 24: Rough Working to Counting Bad Full Cycles

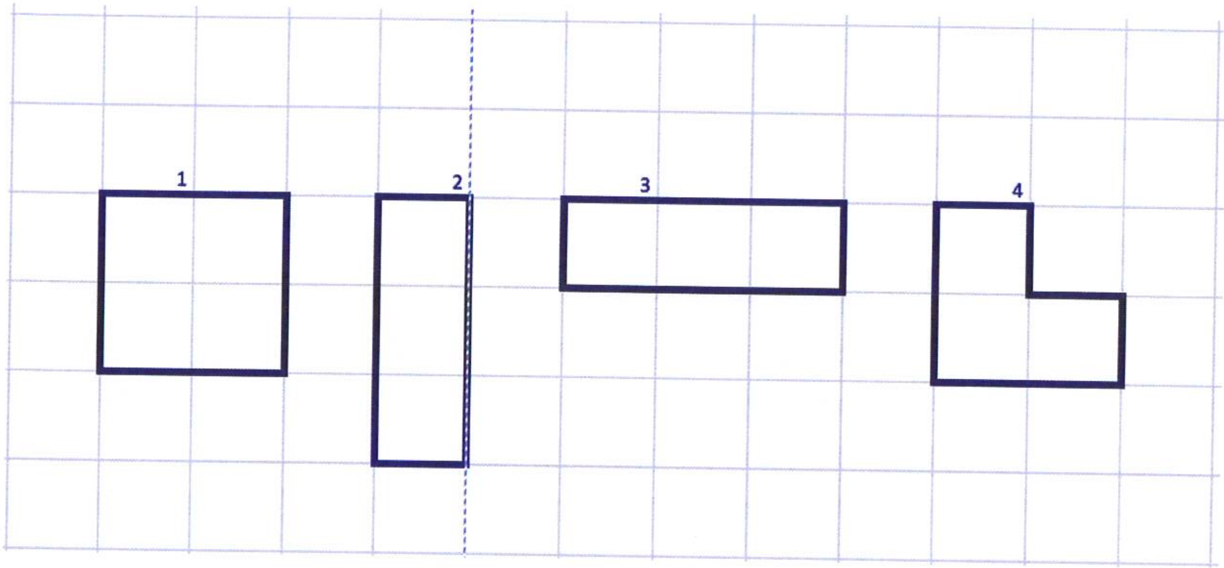


Figure 25: Full Cycle Bad Paths for n = 8

As shown in figure 25, the 4 bad cycles that we considered for n = 8 are illustrated accordingly. For the cycle 1, we counted 8 x 2 = 16 paths (because there are 8 different starting points, that have 2 possible directions they could start in). Similarly, for cycle 2 and 3, we equally obtain 8 x 2 = 16 paths by the same logic. For cycle 4, we obtain a total of 4 x 8 x 2 = 48 bad paths for that particular shape. Therefore, the total sum for the possible bad paths for n = 8 is equal to 16(3) + 48 = 112. After arriving at this number, we realised that there seemed to be no underlying pattern to counting the bad paths in this manner. Furthermore, as n got larger, the possible figure configurations would become too many to consider and count.

- **Questions raised/actionables:**
  - How should we strategically limit our attention into focusing on the bad paths that we could count?

Meeting 7: 7 Mar 2020

- **Meeting duration & location:** 2 hours, 8-10pm, GSR 7

- **Members present:** Zhang Liu, Kim, Lize, Yin Xi

- **Summary:**
  - As we realised that there are too many obstructions to count, we decided to narrow our focus on counting obstructions that we could possibly derive a closed formula for. Our rationale behind this is that we could subtract these obstructions from the upper bound that we currently have: 4^n
  - Removing rectangle obstructions (we will be using n = 8 as an example for this):
    - We begin by considering the following "full" rectangles that can be counted. These rectangles are considered "full" when their width and height are both > 1.

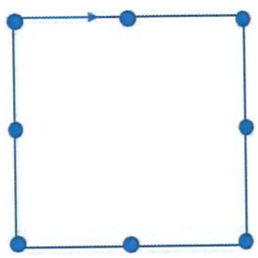


Figure 26: Example of a "Full" Rectangle, 2 by 2

    - We then considered the "skinny" and "flat" rectangles, which is when their width and height are both respectively equal to 1.

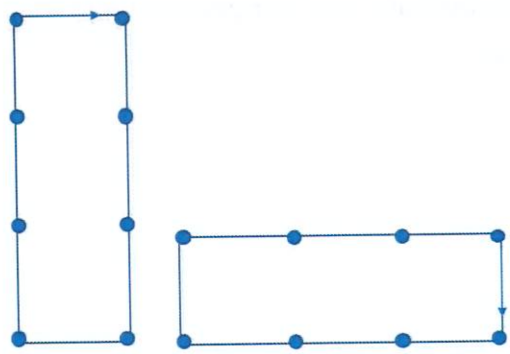


Figure 27: Examples of "Skinny" and "Flat" Rectangles, 3 by 1 and 1 by 3 Respectively

- As we were counting the total number of ways we could form these obstructions, we realised that they could all be counted, and add up to, in the same way.
- This is because they all had the same number of possible starting points, that can be multiplied by 2 to account for both possible ways of formation. Therefore, for this instance, we would obtain 8 times 2 = 16 for each rectangle type.
- We were thus confident that we could start by subtracting these rectangles that form for every even n from n = 8 onwards.

- Removing "L" obstructions
  - Another possible obstruction that we were confident in counting the number of L obstructions that could be formed.
  - We were inspired to count the total number of this type of obstructions because we realised that they could be easily formed from the initial rectangle by "nudging" a corner in as shown below:

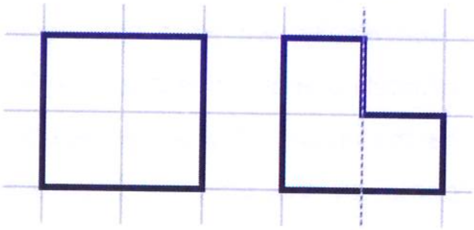


Figure 28: Obtaining L-shape by Nudging Upper-Right Corner in

- **Questions raised/actionables:**
  - Start thinking about part 2(c) in preparation for meeting the next day

Meeting 8: 8 Mar 2020

- **Meeting duration & location:** 2 hours, 12-6pm, GSR 5

- **Members present:** Zhang Liu, Kim, Lize, Yin Xi

- **Summary:**
  - This meeting was divided into 2 parts:
    - Working on part 2(c) of our group's investigation:
      - After discussing, we were thinking of applying the same lower and upper bound idea that we did for part 1(c)
      - However, we quickly realised that it was very difficult for us to come up with an upper bound - unlike the previous problem, not every step is given the option of 4 directions (for example, the first step itself can only be East or North)
      - We therefore decided to maximise our time by focusing on increasing the lower bound instead, as it seemed more feasible - specifically, we were thinking of creating paths that would definitely use all 4 directions (N, S, E, W)
      - We thus came up with the idea of counting the paths that have "dents" as this ensures that we would be utilising the South direction as well.

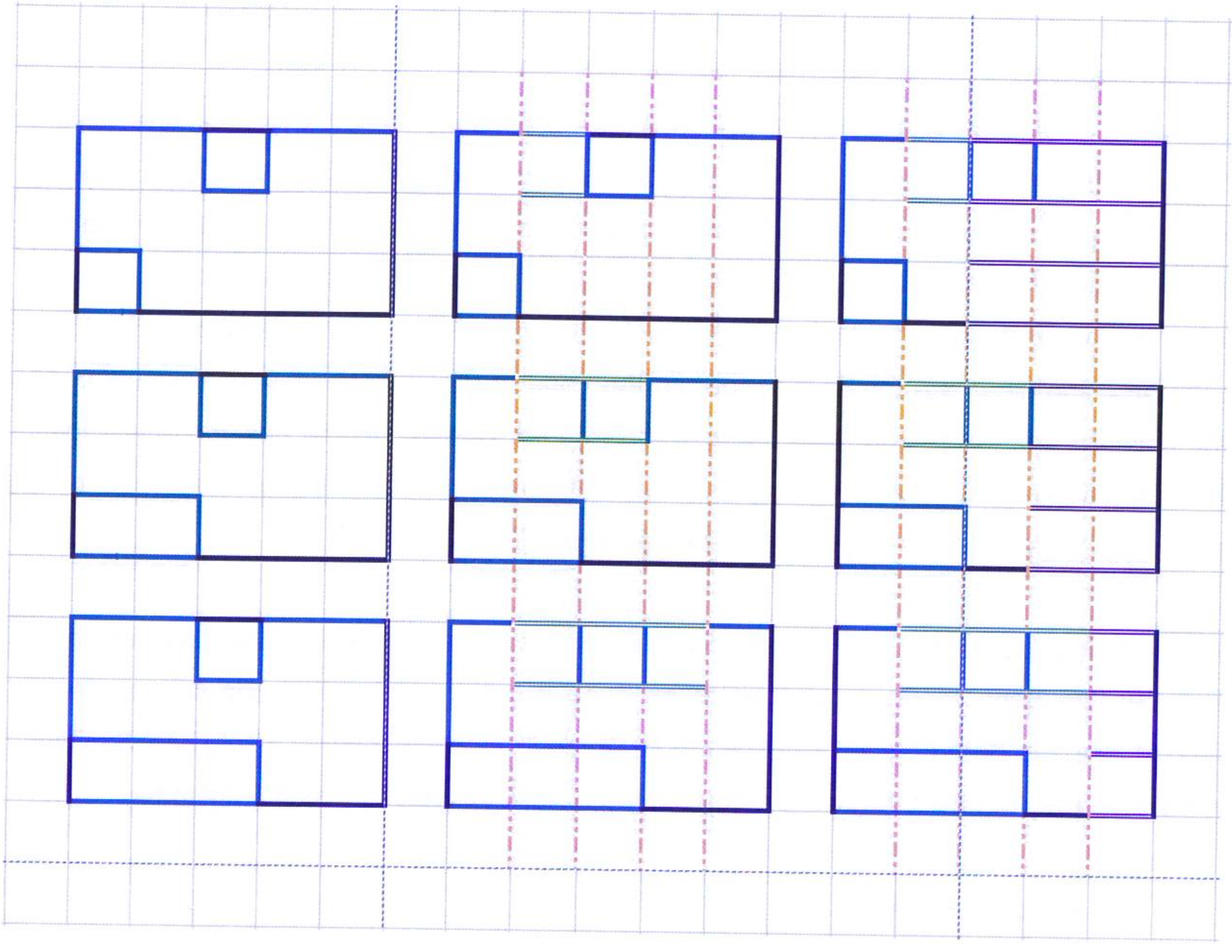


Figure 29: Examples of "dented" paths

From figure 29, we listed out 3 possible "dented" paths that we wanted to make sure that our lower bound accounts for. In this example, we created a lattice path of (5,3).

In the first row, we wanted to count the total number of paths that had the first dent in the lower left corner. The first image shown in the first row accounts for one such path that contains a dent in the lower left corner. To count the total number of paths that have a dent in this lower left corner, we devised the following counting strategy that is illustrated by the following 2 images in the first row.

We first begin by identifying how deep the dent is. In the first example illustrated in the first row, we let the dent be 1 border deep, essentially "occupying" a square 1 by 1. From this, we note that even though the next column has a total of b + 1 (in this case, 4) lines, we can only choose one of the first 2 (illustrated in green) for the path to follow to ensure that the lines do not intersect. We obtain the number 2 by discounting 2 (which are essentially the lines that the dent would occupy) from the total number of horizontal lines available (b + 1), and therefore have (b + 1) - 2 = (b - 1) = 3 - 1 = 2.

Lastly, we determine how to count the subsequent columns after the "green" one. This is illustrated in purple in figure _. We note that after accounting for the "dent", the path is free to occupy any of the horizontal lines in the subsequent columns. In the first row, we see that the last 3 columns have (b + 1) = 4 options each that the path could traverse as the dent is no longer an obstruction to them. We notice that we can obtain this number 3 by subtracting both the columns where the dent occurs (in the case of the first row, the dent occurs in the first column) and the column after the dent ends (in the case of the first row, the dent ends in the first column but the 2nd column also has restrictions placed onto it as discussed in the paragraph before this. Therefore, we obtain 3 by (a - 2) = 5 - 2 = 3.

After noticing this pattern, we sought to extrapolate this idea to varying dent degrees. In the first row, the dent degree is defined to be 1 as it only occupies that one column. In the second row, the dent degree is defined to be 2, as it occupies the first 2 columns. Similarly, in the third row, the dent degree is defined to be 3, as it occupies the first 3 columns.

- **Limitations:**
  - The first limitation that we impose on the dents is that it is a fixed height of 1, with varying widths. As shown in figure 29, the dents always have a height of 1, but we extend the widths accordingly to identify a pattern we could work with.
  - We are always fixing the first dent that occurs: in the first row, the first dent occupies the lower left square. In the second row, the first dent occupies the lower 1 by 2 rectangle. In the third row, the first dent occupies the lower 1 by 3 rectangle.
  - Through this method, we realised that we might be slightly overcounting, as we make it a requirement that we are only counting paths that have at least one dent subsequently after fixing the first one that occurs (previous point). As we are using the *n choose k* method for counting the lines, we will run into the possibility of counting the following case:

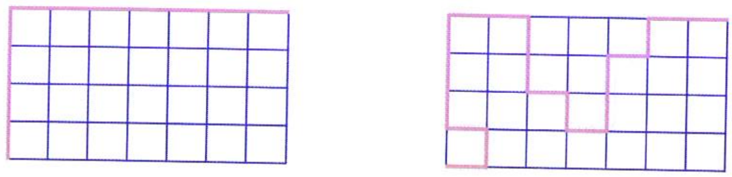


Figure 30: "Undented path"

- We are not interested in counting this path as it does not utilise all 4 directions. In fact, the path is the following string (as it starts from (0,0)): ENWNNEEEEE. We notice, therefore, that it does not consider any Souths. We thus decided to subtract 1 from our total summation to account for this fact.

- **Possible extensions:**
  - As the path we are restricted to counting now specifically considers the dents occurring in the lower left corner of the lattice grid, we were considering the possibility of extending such a counting technique by fixing the dents to other areas of the lattice grid. For example, we can count the total number of paths that have a dent in the upper right corner, just before reaching the point (a, b).

- Proof-reading:
  - We decided to leave the last 2 hours before the project deadline to proofread both the activity log, and the report together.

PROJECT 2: LATTICE DO IT AGAIN!

Team: Lattice Paths![1]
Cai Lize, Tan Yin Xi, Woonha Kim, Zhang Liu

March 8, 2020

[1]Let us pass!

Contents

1. Introduction

The video game Snake allows players to maneuver a moving sequence of dots, joined to resemble a real snake, in a pixelated space. The goal of this game is to direct the snake to "eat" the fruits represented by single pixels so that the snake grows in length. When the snake accidentally "eats" itself, which is when the sequence of dots intersects itself, the game is over.

There is beauty in a growing line that does not intersect itself. Otherwise the game Snake would not have been so addictive. We can study lines that go on forever in a grid more formally using tools in combinatorics. Such lines are called self-avoiding lattice paths (SAP).

What is this?

In this project, we study several types of SAPs in $\mathbb{Z}^2$. In Part 1, we begin our investigation by counting the numbers of SAPs in $\mathbb{Z}^2$ starting from $(0, 0)$ with only the positive directions $\{E, N\}$ and with three directions $\{E, N, W\}$. We also study the self-avoiding $\{E, N, W, S\}$-lattice paths but realize that it may not be possible to derive an explicit formula for the number of such paths. Instead, we present a lower and upper bound for the number of self-avoiding $\{E, N, W, S\}$-lattice paths.

In Part 2, we restrict the grid to be finite, as opposed to the entire $\mathbb{Z}^2$. We also restrict the end point to be the diagonal opposite of the origin on the finite grid. Again, we give an explicit formula for the numbers of self-avoiding paths in such a grid with only the positive directions $\{E, N\}$ and with three directions. We present a lower bound for the number of $\{E, N, W, S\}$-lattice paths in a grid.

2. Self-avoiding X-lattice paths of length N

Before we begin our investigation, we should first define lattice paths and self-avoiding lattice paths in $\mathbb{Z}^2$.

Definition 2.1. Let $E = (1, 0), N = (0, 1), W = (-1, 0)$, and $S = (0, -1)$ and consider a fixed subset $X \subseteq \{E, N, W, S\}$. An **X-lattice path of length n** is a sequence of ordered pairs

$$(a_0, b_0), (a_1, b_1), \ldots, (a_n, b_n)$$

such that $(a_0, b_0) = (0, 0)$ and $(a_k - a_{k-1}, b_k - b_{k-1}) \in X$ for every $k \in [n]$.

Definition 2.2. A lattice path is called **self-avoiding** if $(a_i, b_i) \neq (a_j, b_j)$ for every $0 \leq i < j \leq n$.

Now with these two definitions, we can start our investigation.

2.1. $X = \{E, N\}$. We begin our investigation by first considering only the positive directions, that is when $X = \{E, N\}$.

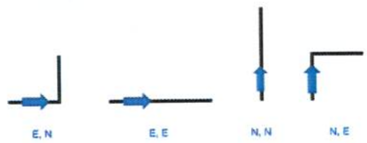


Figure 1. Self-avoiding paths of N and E for n = 2

In Figure 1, we enumerated the possible paths that can be made with $\{E, N\}$ for $n = 2$.

Notice that the 2 directions in this X are not self-cancelling as N is positive in the vertical direction, while E is positive in the horizontal direction. That means any sequences of E and N are always self-avoiding.

Thus, for a $\{E, N\}$-lattice path of length n, at each lattice point, there are 2 choices for the subsequent step: either E or W. Therefore, the total number of possible paths that is of length n is given by 2^n.

2.2. $X = \{E, N, W\}$. Now we have counted the number of $\{E, N\}$-lattice path of length n in $\mathbb{Z}^2$, we add one negative direction W to the set X and count the number of $\{E, N, W\}$-lattice path of length n.

Figure 2. Self-avoiding paths of E, N and W for n = 3

In Figure 2, we enumerated the possible paths that can be made with $\{E, N, W\}$ for $n = 3$.

From the previous case, we develop an approach that is to consider the choices of steps at each lattice point on the path. The case for $X = \{E, N\}$ is relatively easy. At every lattice point, there is no possibility of returning to the previous point because both N and E are positive steps. However, as we add a negative step to our set of steps, counting becomes more complicated. As sequences EW and WE are self-cancelling, now it is possible for the lattice path to go back to the previous point. In other words, if a lattice path ends with E, the next step can only be either N or E for the path to be self-avoiding. Similarly, if a lattice path ends with W, the next step can only be either N or W.

Notice that in the vertical direction, we have only the positive step N. Thus, it is not possible for the path to cancel itself vertically. That means, when a lattice path ends with N, the subsequent step can be any step in the set $\{E, N, W\}$.

In summary, when a $\{E, N, W\}$-lattice path of length n ends with E or W, we have only two choices for the $(n+1)$-th step. On the other hand, when a $\{E, N, W\}$–lattice path n ends with N, we have three choices for the $(n+1)$-th step.

Let $C(n)$ denote the total number of $\{E, N, W\}$-lattice path of length n. Let $C_N(n)$ denote the number of paths ends with an N step and $C_{E,W}(n)$ denote the number of paths ends with either an E or a W step. Given that $X = \{E, N, W\}$, we know the lattice paths have to end with $E, N,$ or W. Thus,

$$C(n) = C_N(n) + C_{E,W}(n).$$

From our analysis above, we also know that for $n \geq 1$, $C_N(n)$ and $C_{E,W}(n)$ both depend on $C_N(n-1)$ and $C_{E,W}(n-1)$, in the following ways:

$$C_N(n) = C_N(n-1) \times 1 + C_{E,W}(n-1) \times 1$$
$$C_{E,W}(n) = C_N(n-1) \times 2 + C_{E,W}(n-1) \times 1.$$

Using those equations, we can tabulate $C_N(n), C_{E,W}(n), C_(n)$ up to $n = 5$.

n	$C_N(n)$	$C_{E,W}(n)$	C(n)
1	1	2	1+2=3
2	$1 \times 1 + 2 \times 1 = 3$	$1 \times 2 + 2 \times 1 = 4$	$3 + 4 = 7$
3	$3 \times 1 + 4 \times 1 = 7$	$3 \times 2 + 4 \times 1 = 10$	$7 + 10 = 17$
4	$7 \times 1 + 10 \times 1 = 17$	$7 \times 2 + 10 \times 1 = 24$	$17 + 24 = 41$
5	$17 \times 1 + 24 \times 1 = 41$	$17 \times 2 + 24 \times 1 = 56$	$41 + 58 = 99$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

We notice that for $n \geq 3$,

$$\begin{aligned} C(n) &= C_N(n) + C_{E,W}(n) \\ &= C_N(n-1) \times 1 + C_{E,W}(n-1) \times 1 + C_N(n-1) \times 2 + C_{E,W}(n-1) \times 1 \\ &= (C_N(n-1) + C_{E,W}(n-1)) \times 2 + C_N(n-1) \\ &= C(n-1) \times 2 + (C_N(n-2) \times 1 + C_{E,W}(n-2) \times 1) \\ &= 2C(n-1) + C(n-2). \end{aligned}$$

Could you have arrived at this in fewer steps?

We can test this formula by looking at $n = 5$. When $n = 5$, $C(5) = 99 = 41 \times 2 + 17 = 2C(4) + C(3)$.

To make this formula work for $n = 2$, we let $C_0 = 1$. It makes sense because there is only one path of 0 steps that is the origin point itself.

In this way, we find a recursive formula for $C(n)$ in terms of $C(n-1)$ and $C(n-2)$ for $n \geq 2$. With this recursive relation, we can find an explicit formula by finding a generating function for a sequence a_n, where $a_n = C(n)$.

Let $G(x) = \sum_{n=0}^{\infty} a_n x^n$ and $a_0 = 1, a_1 = 3$.

We know that $a_{n+2} = a_n + 2a_{n+1}$. Therefore,

$$\sum_{n=0}^{\infty} a_{n+2}x^{n+2} = \sum_{n=0}^{\infty}(a_n + 2a_{n+1})x^{n+2}.$$

But the left-hand side of the equation is also $G(x) - a_0x^0 - a_1x^1$, so we have the following:

$$\begin{aligned}
G(x) - a_0 - a_1x &= \sum_{n=0}^{\infty}(a_n + 2a_{n+1})x^{n+2} \\
&= \sum_{n=0}^{\infty} a_nx^{n+2} + \sum_{n=0}^{\infty} 2a_{n+1}x^{n+2} \\
&= x^2G(x) + 2x(G(x) - a_0) \\
\cancel{\Rightarrow} G(x) - 1 - 3x &= x^2G(x) + 2x(G(x) - a_0) \\
G(x) - x^2G(x) - 2xG(x) &= 1 + 3x - 2x \\
(1 - 2x - x^2)G(x) &= 1 + x \\
G(x) &= \frac{1+x}{1-2x-x^2} \\
&= \frac{-1-x}{(1+\sqrt{2}+x)(1-\sqrt{2}+x)} \\
&= \frac{\alpha}{(1+\sqrt{2}+x)} + \frac{\beta}{(1-\sqrt{2}+x)}.
\end{aligned}$$

This implies

Why not factor it into the form $(1-ax)(1-bx)$?

Now, we can solve for α and β.

$$\begin{aligned}
\alpha(1-\sqrt{2}) + \beta(1+\sqrt{2}) &= -1 \\
(-1-\beta)(1-\sqrt{2}) + \beta(1+\sqrt{2}) &= -1 \\
-1 + \sqrt{2} - \beta + \beta\sqrt{2} + \beta + \beta\sqrt{2} &= -1 \\
2\sqrt{2}\beta &= -\sqrt{2} \\
\beta &= \frac{-\sqrt{2}}{2\sqrt{2}} = \frac{1}{2}.
\end{aligned}$$

Therefore,

Thus, $\alpha = -1 - \beta = -1 + \frac{1}{2} = -\frac{1}{2}$. Substitute α and β back to the expression for $G(x)$, we obtain

$$\begin{aligned}
G(x) &= \frac{\alpha}{(1+\sqrt{2}+x)} + \frac{\beta}{(1-\sqrt{2}+x)} \\
&= \frac{-\frac{1}{2}}{(1+\sqrt{2}+x)} + \frac{\frac{1}{2}}{(1-\sqrt{2}+x)} \\
&= \frac{\frac{1-\sqrt{2}}{2}}{1-(1-\sqrt{2})x} + \frac{\frac{1+\sqrt{2}}{2}}{1-(1+\sqrt{2})x} \\
&= (\frac{1-\sqrt{2}}{2})\frac{1}{1-(1-\sqrt{2})x} + (\frac{1+\sqrt{2}}{2})\frac{1}{1-(1+\sqrt{2})x}.
\end{aligned}$$

Thus,

$$C(n) = (\frac{1-\sqrt{2}}{2})(1-\sqrt{2})^n + (\frac{1+\sqrt{2}}{2})(1+\sqrt{2})^n. = \frac{(1-\sqrt{2})^{n+1} + (1+\sqrt{2})^{n+1}}{2}$$

We check if our closed formula is true by checking against when $n = 0$ and $n = 1$. Indeed,

$$\begin{aligned}
C(0) &= (\frac{1-\sqrt{2}}{2})(1-\sqrt{2})^0 + (\frac{1+\sqrt{2}}{2})(1+\sqrt{2})^0 \\
&= (\frac{1-\sqrt{2}}{2}) + (\frac{1+\sqrt{2}}{2}) \\
&= 1, \\
C(1) &= (\frac{1-\sqrt{2}}{2})(1-\sqrt{2})^1 + (\frac{1+\sqrt{2}}{2})(1+\sqrt{2})^1 \\
&= \frac{1}{2}(1-\sqrt{2})^2 + \frac{1}{2}(1+\sqrt{2})^2 \\
&= \frac{1}{2}(1+2) \\
&= 3.
\end{aligned}$$

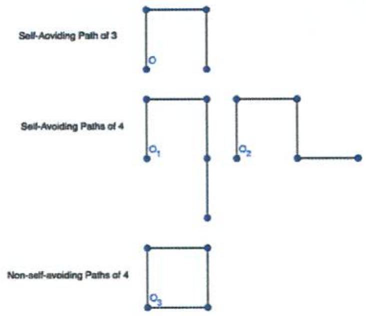
Thus, we find a formula for counting the number of self-avoiding E, N, W-lattice path of length n, that is $C(n) = (\frac{1-\sqrt{2}}{2})(1-\sqrt{2})^n + (\frac{1+\sqrt{2}}{2})(1+\sqrt{2})^n$.

2.3. $X = \{E, N, W, S\}$. Now we have counted the number of self-avoiding $\{E, N, W\}$-lattice paths of length n, we finally add the last negative step to the set of steps X. For smaller n, it is relatively easy to determine the number of self-avoiding $\{E, N, W, S\}$-lattice paths of length n.

n	Self-Avoiding Paths
1	4
2	$4 \times 3 = 12$
3	$4 \times 3^2 = 36$

For $1 \leq n \leq 3$, we can still count the number of self-avoiding paths by considering what happens at each lattice point. At the origin, there are 4 possible choices of steps, namely E, N, W, or S. At the next lattice point, however, there are only 3 choices of steps for the path to be self-avoiding. That is, the path cannot go back to the previous point immediately. For example, if we have E as the first step, the next step cannot be W because otherwise the path goes back to $(0,0)$. Thus, the numbers of $\{E, N, W, S\}$-lattice paths of 2 and 3 are 12 and 36 respectively.

Unlike the previous cases, however, the self-avoiding $\{E, N, W, S\}-$lattice paths of 4 does not only depend on the previous step. We could have a self-avoiding path of 3 that is NES, but at the last lattice point, we are left with only 2 options instead of 3 for the path to remain self-avoiding as shown in Figure 3.



Why not NESW?

FIGURE 3. Self-avoiding paths of 4 from a self-avoiding path of 3

We notice that the only ~~non-~~self-~~avoiding~~ intersecting paths of 4 are in the shape of a square. Let's call these non-self-avoiding paths in general obstructions.

Definition 2.3. A lattice path is an **obstruction of n** if $(a_n, b_n) = (a_i, b_i)$ for some i, $0 \leq i \leq n$.

We can thus count the number of the squares and subtract that from the number of paths not immediately returning to the previous point. There are two ways of forming a particular square path that has the origin as one of its vertices as shown in Figure 4. One path is $NESW$ and the other $ENWS$.

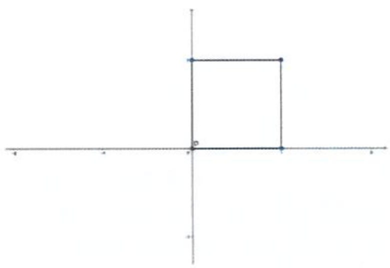


FIGURE 4. A non-self-avoiding path of 4

Moreover, we can reflect the unit square in the x-axis, y-axis, and about the origin. Thus, in total, we have 4 unit squares that has the origin as one of it's vertices. For each of the unit squares, there are 2 possible paths. Hence, we have $4 \times 2 = 8$ $\{E, N, W, S\}$-lattice paths of 4 that form a square.

If we still use $C(n)$ to denote the number of self-avoiding paths, we can compute the following table.

n	Paths Not Immediately Returning	Other Obstructions	$C(n)$
1	4	-	4
2	12	-	12
3	36	-	36
4	$4 \times 3^3 = 108$	$4 \times 2 = 8$	$108 - 8 = 100$

The cases for $n = 5$ is slightly more complicated. Though 5 unit steps cannot form a closed shape, 5 steps can still form an obstruction that is extended from a unit square.

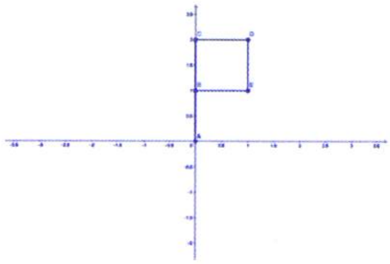


FIGURE 5. An obstruction of length 5

An obstruction of length 5 as shown in Figure 5 can be formed in two ways: $NNESW$ or $NENWS$. This particular shape can rotate and reflect in the grid. It is not hard to visualize there are in total 8 shapes like this around the origin in $\mathbb{Z}^2$. Hence, the number of obstructions of length 5 is $8 \times 2 = 16$.

n	Paths Not Immediately Returning	Other Obstructions	$C(n)$
1	4	-	4
2	12	-	12
3	36	-	36
4	108	8	100
5	$100 \times 3 = 300$	16	$300 - 16 = 284$

We notice that the obstructions of length 4 are different from those of length 5. In general, the obstructions of an even length can form a closed shape whereas an obstruction of an odd length cannot be closed. It might be interesting later, it will be as it turns out, to give a different name to the obstructions of even lengths.

However, as n grows larger, it is increasingly difficult to count the number of obstructions. For example, when $n = 6$, the obstructions are shown in Figure 6.

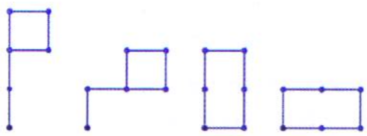


FIGURE 6. Obstructions of length 6

Since the obstructions become increasingly complicated as n becomes larger, we will give the lower and upper bounds of $C(n)$, instead of giving a formula of $C(n)$.

2.3.1. *Lower bound.* It is relatively easy to give the lower bound of $C(n)$.

Note that for any n, the self-avoiding E, N, W-lattice paths of length n are always a subset of E, N, W, S-lattice paths of length n. Therefore,

$$C(n) \geq (\frac{1-\sqrt{2}}{2})(1-\sqrt{2})^n + (\frac{1+\sqrt{2}}{2})(1+\sqrt{2})^n.$$

How go is this lower bound?

2.3.2. *Upper bound.* It is trickier to calculate the upper bound for $C(n)$ because it is difficult to count the obstructions of n.

To begin, we know from the discussion above that $C(n) \leq 4 \times 3^{n-1}$ because $4 \times 3^{n-1}$ counts the number of lattice paths of n that do not return immediately to its previous step.

However, we can improve this upper bound by considering some subsets of the obstructions, specifically closed lattice path of length n. We will first give a definition to characterize such obstructions and then propose methods for counting them in the following subsections.

Definition 2.4. An obstruction of n is an **n-cycle** if $(a_n, b_n) = (a_0, b_0)$ and $(a_i, b_i) \neq (a_j, b_j)$ for every $0 < i < j < n$.

2.3.3. *Counting the cycles: rectangles and L shapes.* As we have noted previously, when $n = 2k$, that is when n is even, there are some $2k$-cycles amongst the obstructions of n. Here we propose a way of counting the number of $2k$-cycles. We can improve the upper bound for $C(n)$ by subtracting the number of $2k$-cycles.

Note that the cycles can be in the shape of rectangles or concave polygons, as shown in Figure 7.

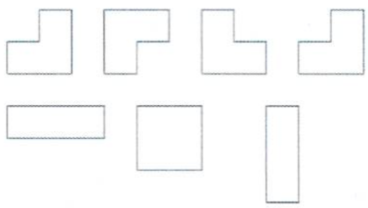


FIGURE 7. All 8-step polygons on a grid (Adapted from Asia Pacific Mathematics Newsletter)

We know the number of rectangles of a circumference n is the number of strong compositions of $\frac{n}{2}$ as 2 integers. Each pair of integers gives the width and length of a rectangle whose circumference is n. That is,

$$\binom{n/2-1}{1} = \frac{n}{2} - 1.$$

For example, when $n = 8$, there are in total $4 - 1 = 3$ strong compositions of 4 as 2 integers: $1 + 3$, $2 + 2$, $3 + 1$. Each corresponds to a rectangle of circumference 8.

For each of the rectangles, there are n vertices and each of the n vertices can be a choice for the starting point of the lattice path. For each starting point, there are 2 ways of completing the path: in clockwise and anti-clockwise directions.

Therefore, when n is even, the number of n-cycles in the form of a rectangle is

$$(\frac{n}{2} - 1) \times n \times 2.$$

Now, we consider a related subset of n-cycles. In Figure 7, we see 4 L-shaped 8-cycles. These can be formed by nudging a corner of the 2×2 rectangle.

More generally, a corner of an a by b rectangle, where $a > 1$ and $b > 1$, can be nudged to form an L-shape in $(a-1) \times (b-1)$ ways, as shown in Figure 8. The same can be repeated for each of the 4 corners of the rectangle.

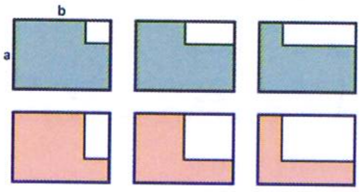


FIGURE 8. Each corner can be nudged in $(a-1) \times (b-1)$ ways

For each of the L shapes, there are again n different choices for the starting point and for each choice of starting point, there are 2 directions in which the cycle can be completed.

Therefore, when n is even, the number of L-shapes of length n is

$$\sum_{a+b=\frac{n}{2}} (\frac{n}{2} - 1 - 2) \times n \times 2 \times (a-1) \times (b-1) \times 4$$
$$= 8 \sum_{a+b=\frac{n}{2}} n(\frac{n}{2} - 3)(a-1)(b-1)$$

Notice that we need to subtract 2 from the number of rectangles because we cannot nudge the rectangles whose width or length is 1.

Now we have considered the cases where n is even, we can use our results for cases when n is odd.

When n is odd, one type of obstructions can be formed by adding a step at the starting point of each $(n-1)$-cycles, as shown in Figure 9. In other words, the number of n obstructions is greater or equal to the number of $(n-1)$-cycles.

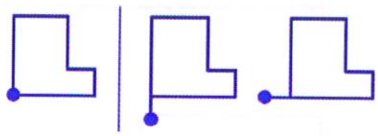


FIGURE 9. A 12-cycle (left) can be extended to an obstruction of 13 in two ways (middle and right)

Therefore, when n is odd and $k = n - 1$, the number of obstructions of n is at least

$$8 \sum_{a+b=\frac{k}{2}} n(\frac{k}{2} - 3)(a-1)(b-1).$$

This is an underestimation of obstructions of an odd n because at certain starting points, such as the one shown in Figure 9, one $(n-1)$-cycle can be extended to more than 1 obstructions of n.

Nevertheless, we can sum the numbers of rectangular and L-shaped cycles from 4 to n. When n is odd, we simply have the number of k-cycles up to $k = n - 1$ multiplied by 2, that is,

$$2 \times \left(2 \times \sum_{k=\{4,6,8,\ldots\}}^{n-1} k(\frac{k}{2} - 1) \right) + \\ 2 \times \left(8 \times \sum_{k=\{8,10,12,\ldots\}}^{n-1} \sum_{a+b=k/2} k(\frac{k}{2} - 3)(a-1)(b-1) \right).$$

When n is even, it is more complicated.

$$2 \times \left(2 \times \sum_{k=\{4,6,8,\ldots\}}^{n-2} k(\frac{k}{2} - 1) \right) + \\ 2 \times \left(8 \times \sum_{k=\{8,10,12,\ldots\}}^{n-2} \sum_{a+b=k/2} k(\frac{k}{2} - 3)(a-1)(b-1) \right) + \\ 8 \sum_{a+b=\frac{n}{2}} n(\frac{n}{2} - 3)(a-1)(b-1).$$

By subtracting the above sum from $4 \times 3^{n-1}$, we improve the upper bound for $C(n)$.

This method does not consider all of the obstructions, however. As n becomes larger, there could be obstructions in other irregular concave polygons, such as the ones shown in Figure 10.

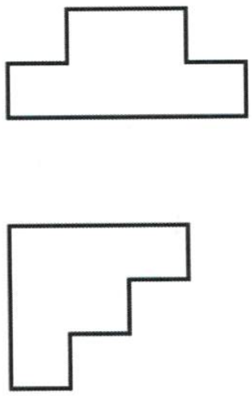
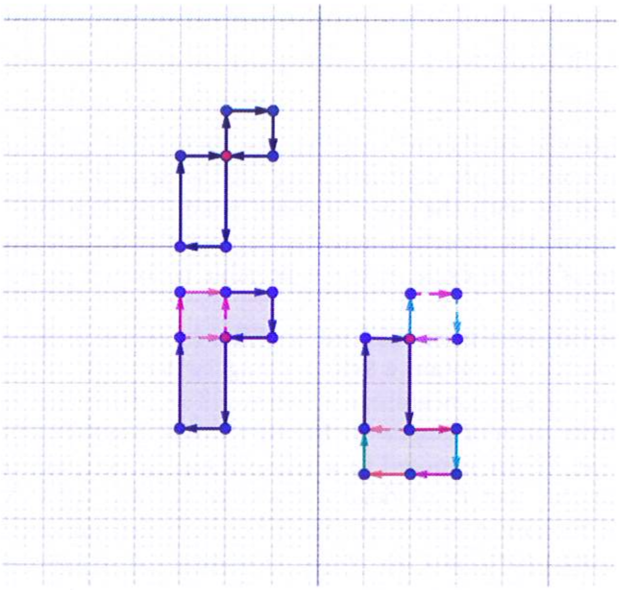


FIGURE 10. Obstructions of 12 that are not L-shaped

Therefore, we introduce a new method of counting that does not consider the exact shape of obstructions.

2.3.4. *Counting the cycles: the concatenation method.* The counting method in the previous section only accounts for the number of cycles with rectangle shape and L-shape. On the other hand, we propose another method that focuses on a more generic subset of n-cycles. Unlike the previous counting method, this new method disregards the shape of the cycles. The key idea behind the method in this section is to form the $(n+m)$-cycle by concatenating a n-cycle and a m-cycle.

We will illustrate the construction process with obtaining a 10-cycle. We obtain the 10-cycle from concatenating a 4-cycle to a 6-cycle. First, choose the starting point from the four corners of the 6-cycle. After fixing the starting point of the 6-cycle, the 4-cycle has to be attached this point, which we call "concatenation point" for convenient reference. After concatenating the two cycles, we can rearrange the edges to form a 10-cycles. The shape of the final cycle will depend on the concatenation point and the directions of both the 6-cycle and 4-cycle. To be specific, if they are of the same direction, the final 10-cycle will give an "L-shape";if they are of the opposite directions, the final 10-cycle will give a "staggered shape." The rearrangement process is shown below.



Nice figure!

FIGURE 11. Rearranging the edges of concatenated cycles with the same direction.

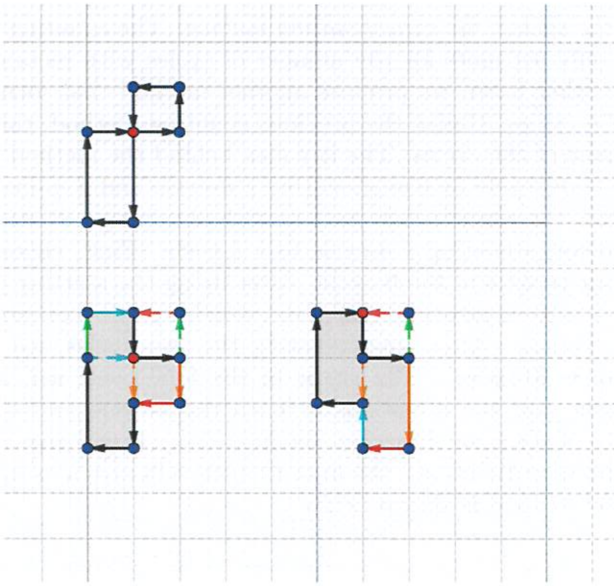


FIGURE 12. Rearranging the edges of concatenated cycles with opposite directions.

Let $D(n)$ denote the number of n-cycles. For $D(n)D(m)$ gives an upper bound for $D(n+m)$ because simply concatenating all the possible cycles will give rise to repeated final shape with the same starting point and direction; there are also possibilities that given the direction and starting point of n-cycle and the direction and starting point of the m-cycle, it is not possible to form a $(n+m)$-cycle which fits our definition.

We can extend this idea from counting n-cycles to counting the number of self-avoiding lattice paths. To obtain a self-avoiding lattice path of length k, we can simply find $n+m=k$ and concatenate a self-avoiding lattice path of length n and a self-avoiding lattice path of length m. In other words, a self-avoiding lattice path of length $n+m$ can be decomposed to a self-avoiding lattice path of length n and a self-avoiding lattice. Let $C(n)$ denote the total number of$\{E, N, W, S\}$-lattice path of length n, for any $n \geq 0 \in \mathbb{N}$. Since we have decomposed the process of forming a (n+m)-step path into two steps, 1) finding the number of n-step paths, 2) finding the number of m-step paths. Thus, the number of ways for this process is $C(n)C(m)$. However, this "naive" process involves the possibilities of two paths intersecting each other, as shown below.

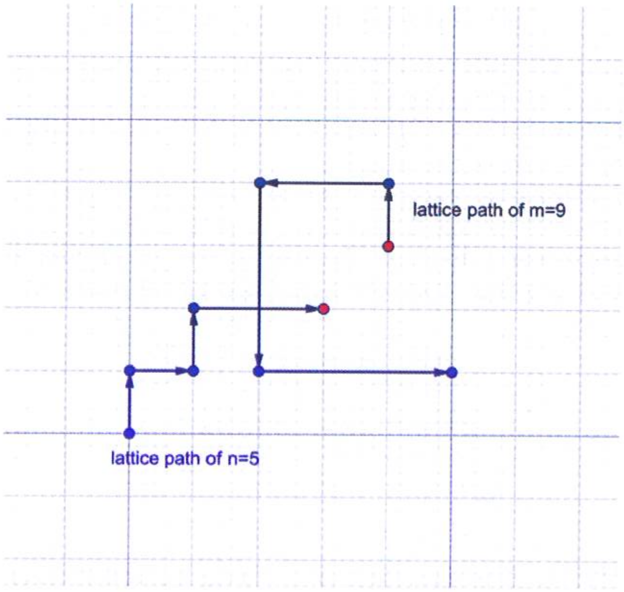


FIGURE 13. Scenario where a concatenation causes intersecting lattice paths.

Therefore, the actual number of paths for $(n+m)$-step lattice paths $C(n+m)$ will be less than or equal to $C(n)C(m)$, i.e.,

$$C(n+m) \leq C(n)C(m),$$

for any $n+m=k, n \geq 0, m \geq 0$.

When $n=0$ or $m=0$, $C(n+m)=C(n)C(m)$.

Unlike the first method that gives a closed formula for the number of obstructions, this concatenation method gives us a recursive relation to estimate the upper bound of the number of n-cycles and the number of self-avoiding lattice paths of length n.

2.3.5. *Summary.* Up to this point, we have studied the number of self-avoiding lattice paths with 2, 3, and 4 options for steps. We notice that the restrictions on the steps restricts the total number of self-avoiding lattice paths. For $\{E,N\}$- and $\{E,N,W\}$-lattice paths, we can find an explicit formula for the number of self-avoiding paths. In each of those cases, we can find the number by considering the options of steps at each lattice point.

But for $\{E,N,W,S\}$-lattice paths, the lift restriction on the options for steps makes it difficult to count the total number of self-avoiding paths. Instead, we present a lower and upper bound for the number as a way to approximate the number $C(n)$. We develop two ways of counting subsets of obstructions. The first one counts the obstructions of n in the shapes of rectangles and L-shapes and gives a closed formula for the number of such obstructions. The second one finds a recursive relationship between different cycles.

Although neither of the two methods covers all obstructions, we believe they still provide insights into the nature of obstructions and are still huge improvements for the upper bound.

Excellent!

3. Specialisation X a by b lattice

3.1. Introduction. We have seen from the discussion above that it is difficult to count the number of self-avoiding $\{E, N, W, S\}$-lattice paths in the entire $\mathbb{Z}^2$. Therefore, we will impose some restrictions on the lattice paths and study a special subset of self-avoiding lattice paths in $\mathbb{Z}^2$.

Instead of considering the entire $\mathbb{Z}^2$, we will restrict the grid to a b by a grid. Furthermore, instead of having different end points of the lattice paths, we will restrict the ending points to be (a, b). In other words, every path must start from point $(0, 0)$ and end at point (a, b), while the path cannot leave the a by b grid.

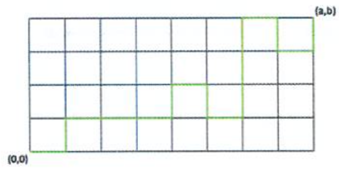


Figure 14. A self-avoiding lattice path from $(0, 0)$ to (a, b)

We will follow the same approach to Part 1 for Part 2 of our investigation. Specifically, we will consider the number of self-avoiding lattice paths in an b by a grid with only the positive steps, with three steps, and with all four steps.

3.2. $X = \{E, N\}$. Similar to Section 2.1, any lattice paths consisting of E's and N's will always be self-avoiding.

For a path from $(0, 0)$ to reach (a, b) ~~is $a + b$,~~ we need in total a E's and b N's. The strings of E's and N's can be rearranged and any permutation of a E's and b N's represents a path from $(0, 0)$ to (a, b).

Therefore, the explicit formula for the number of self-avoiding $\{E, N\}$-lattice paths from $(0, 0)$ to (a, b) is the number of permutations of a E's and b N's, which is

$$\binom{a+b}{a} = \binom{a+b}{b} = \frac{(a+b)!}{a! \times b!}.$$

3.3. $X = \{E, N, W\}$ or $X = \{E, N, S\}$.

3.3.1. $X = \{E, N, W\}$. Unlike the previous case where only two steps are involved, When we have three steps to choose from, we do not know how many horizontal steps there are in a self-avoiding $\{E, N, W\}$-lattice path from $(0, 0)$ to (a, b). However, we *do* know that any self-avoiding $\{E, N, W\}$-lattice path from $(0, 0)$ to (a, b) has exactly b N's.

Furthermore, we know that each N step has to be in a different row of the grid. In other words, every row in the grid has exactly one vertical line that is part of the self-avoiding path, as shown in Figure 15.

Now, in each row of the b by a grid, there are $(a+1)$ vertical lines to choose from. In total, there are b rows. Thus, the number of self-avoiding $\{E, N, W\}$-lattice path from $(0, 0)$ to (a, b) is

$$\binom{a+1}{1}^b. \quad = \quad (a+1)^b$$

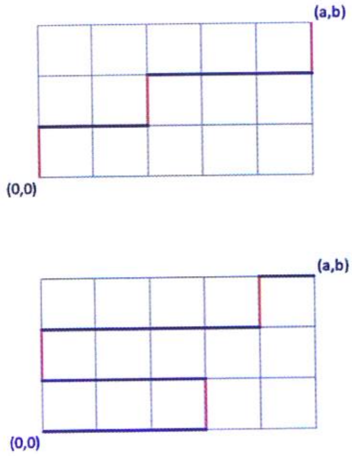


FIGURE 15. A different set of N's determines a different self-avoiding lattice path

3.3.2. $X = \{E, N, S\}$. Similarly, when we have the set of steps as $\{E, N, S\}$, we do not know how many vertical steps there are in a self-avoiding lattice path, but we do not know each self-avoiding lattice path takes exactly a E's.

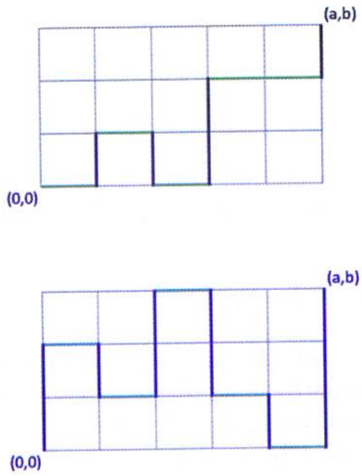


FIGURE 16. A different set of E's determines a different self-avoiding lattice path

Using the same grid, we now consider the vertical lines instead in Figure 16. Note that we now choose the E's from $(b+1)$ choices in each column and there are a columns in total. Therefore, the number of self-avoiding $\{E, N, S\}$-lattice path from $(0,0)$ to (a,b) is

$$\binom{b+1}{1}^a. = (b+1)^a$$

3.4. $X = \{E, N, W, S\}$. From Section 2.3, we learn that it is difficult to find an explicit formula for the number of self-avoiding $\{E, N, W, S\}$-lattice paths in $\mathbb{Z}^2$. In an a by b grid, it is also not easy to count the number of self-avoiding $\{E, N, W, S\}$-lattice paths. Instead, we will try to find a lower bound for the number and improve our lower bound by counting a subset of $\{E, N, W, S\}$-lattice paths.

3.4.1. *Initial Lower Bound.* Note that the set of all self-avoiding $\{E, N\}$-lattice paths are subsets of self-avoiding $\{E, N, W\}$-lattice paths and self-avoiding $\{E, N, S\}$-lattice paths. Furthermore, the sets of all self-avoiding $\{E, N, W\}$- and $\{E, N, S\}$-lattice paths are also subsets of the self-avoiding $\{E, N, W, S\}$-lattice paths. The relationship between the different sets of lattice paths are explained in Figure 17

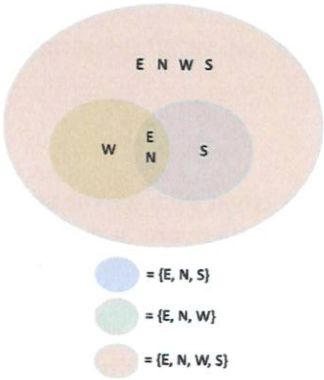


FIGURE 17. Relationship between different kinds of lattice paths

Therefore we find a lower bound for the number of self-avoiding $\{E, N, W, S\}$-lattice paths from $(0,0)$ to (a,b) by applying the Inclusion-Exclusion principle.

$$LowerBound = \binom{a+1}{1}^b + \binom{b+1}{1}^a - \binom{a+b}{b}.$$

3.4.2. *Improving the lower bound.* We can improve our lower bound by counting a subset of $\{E, N, W, S\}$-lattice paths that indeed involves all four steps. We note that such a path can be constructed by making 2 "dents" from a simple $\{E, N\}$-lattice path along the boundary of the b by a grid.

FIGURE 18. A $\{E, N, W, S\}$-lattice path (right) constructed by making two dents on a $\{E, N\}$-lattice path (left)

Thus, from here onward, we will study the number of $\{E, N, W, S\}$-lattice that can be constructed by adding 2 dents on a lattice path along the boundary of the a by b grid.

Let us start with an example of 3 by 5 grid, as shown in Figure 19. If we fix the first dent to be in the bottom row and has a dented length 1, we can make the second dent in the purple area and still ensure the lattice path is self-avoiding.

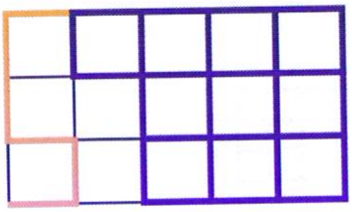


FIGURE 19. The purple area is where we can make the second dent

Thus, for this particular orange dent, it can form

$$\binom{2}{1} \times \binom{4}{1}^3 - 1 = 2 \times 4^3 - 1$$

self-avoiding $\{E, N, W, S\}$-lattice paths that reach $(5, 3)$. The -1 comes from the path where there is no second dent.

Now, in a 3 by 5 grid, there are 5 ways of making the first dent, as shown in Figure 20. When the dent of the first length is different, the area for which the second dent can occur is also different.

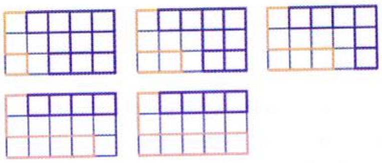


FIGURE 20. There are 5 positions for the first dent

We can tabulate the number of self-avoiding lattice paths in each cases.

Dent Length	Number of 2-Dent Paths
1	$2 \times 4^3 - 1$
2	$2^2 \times 4^2 - 1$
3	$2^3 \times 4^1 - 1$
4	$2^4 \times 4^0 - 1$
5	$2^4 \times 4^0 - 1$

In general, there are a different choices for the first dent in the bottom row. For the first dent of length i, $1 \leq i \leq a-1$, the number of second dents will be $\binom{b-1}{1}^i \times \binom{b+1}{1}^{(a-i-1)} - 1$. And for when the first dent has length a, the number of second dents will still be $\binom{b-1}{1}^{(a-1)} - 1$.

Therefore, in a b by a grid, the number of 2-dent self-avoiding $\{E, N, W, S\}$-lattice paths with the first dent in the first row is

$$\left(\sum_{i=1}^{a-1}(b-1)^i \times (b+1)^{(b-i-1)} - 1\right) + (b-1)^{(a-1)} - 1.$$

For each path that we have counted, we can find a similar path by finding a "flipped" path. For instance, the dent in Figure 19 can be reflected in the two gray lines as shown in Figure 21 to produce the dented path on the right of Figure 21.

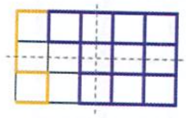


FIGURE 21. The dented path on the left can be reflected in the gray lines to produce the path on the right

Hence, we can find the number of 2-dented paths with the horizontal dent in either the first or last row by multiplying the original equation by two:

$$2\left(\left(\sum_{i=1}^{a-1}(b-1)^i \times (b+1)^{(b-i-1)} - 1\right) + (b-1)^{(a-1)} - 1\right).$$

Therefore, our improved lower bound becomes

$$LowerBound = \binom{a+1}{1}^b + \binom{b+1}{1}^a - \binom{a+b}{b} + 2\left(\left(\sum_{i=1}^{a-1}(b-1)^i \times (b+1)^{(b-i-1)} - 1\right) + (b-1)^{(a-1)} - 1\right).$$

We should also consider the cases where we fix the first dent in the vertical position, that is having the dent in the first or last column as shown on the left of Figure 22. But we will run into the problem of overcounting the paths like that on the right of Figure 22.

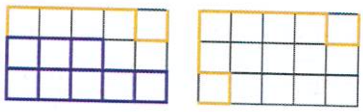


FIGURE 22. Considering the cases of fixing vertical dents will overcount the paths on the right

The solution to over counting is beyond the scope of this project report but we believe the method we have developed is still illuminating.

There are other self-avoiding $\{E, N, W, S\}$-lattice paths not counted using our method. For example, the paths such as the one in Figure 23 where more complicated detours are involved are not considered in our formula.

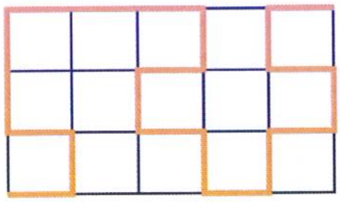


FIGURE 23. More complicated self-avoiding paths not considered in our formula

3.5. **Summary.** Like in Part 1, we are able to find relatively easily the explicit formulas for the numbers of self-avoiding lattice paths with 2 or 3 options of steps, but we are not able to find exactly the number of $\{E, N, S, W\}$-lattice paths from $(0,0)$ to (a,b). This tells us the set of steps indeed imposes great restrictions so much so that the number of self-avoiding lattice paths is reduced significantly.

What is different from Part 1, however, is in a finite b by a grid, we are not able to determine a reasonable upper bound for the number of self-avoiding $\{E, N, W, S\}$-lattice paths because of the bounded conditions. Therefore, the added condition does not necessarily help us find the number of self-avoiding path in this case, but rather increases the difficulties.

On the other hand, the new condition is not all that unhelpful. Instead of improving the upper bound by counting the number of obstructions, it becomes easier in the b by a grid to count the lattice paths that involve steps in all four directions, because we can now bound the paths. Thus, a question worth investigating in future is if we can improve the lower bound of the $C(n)$ in Part 1 by studying some finite spaces in $\mathbb{Z}^2$.

4. Conclusion

In this project, we have explored the different types of self-avoiding lattice paths — with two, three, and four options for steps — both in the infinite space of $\mathbb{Z}^2$ and the finite space of a b by a grid. We notice that in each of the spaces, it is relatively easy to solve for the self-avoiding lattice paths that involve only 2 or 3 steps. In both spaces, we are unable to find an explicit formula for the number of self-avoiding $\{N, E, S, W\}$-lattice paths.

Though we are not able to find the exact number of self-avoiding $\{N, E, S, W\}$-lattice paths, we have developed ways of counting subsets of obstructions and self-avoiding paths. These methods may be insightful for further explorations.

There are also other questions that arise from our investigation, such as how to count the obstructions in the shapes of other concave polygons and how to find the maximum number of steps in a self-avoiding lattice path in a b by a grid. These questions are worthy of further investigations.