Is Pascal's Triangle a Fractal ?

Ankith A Das

The University of Sydney

May 12, 2019

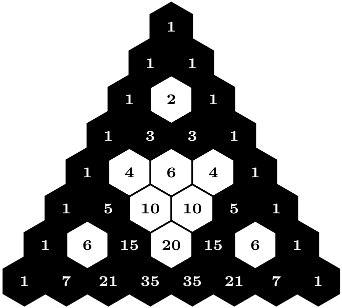
Pascal's Triangle

$$
\begin{array}{ccccccccccccccc}
 & & & & & & & 1 & & & & & & & \\
 & & & & & & 1 & & 1 & & & & & & \\
 & & & & & 1 & & 2 & & 1 & & & & & \\
 & & & & 1 & & 3 & & 3 & & 1 & & & & \\
 & & & 1 & & 4 & & 6 & & 4 & & 1 & & & \\
 & & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & & \\
 & 1 & & 6 & & 15 & & 20 & & 15 & & 6 & & 1 & \\
1 & & 7 & & 21 & & 35 & & 35 & & 21 & & 7 & & 1
\end{array}
$$

- One of the earliest mentions was in a Chinese document at around 1303 AD
- Looks pretty innocent right?

Pascal's Triangle

- It is observed by coloring
  1. all odd numbers black
  2. all even numbers white



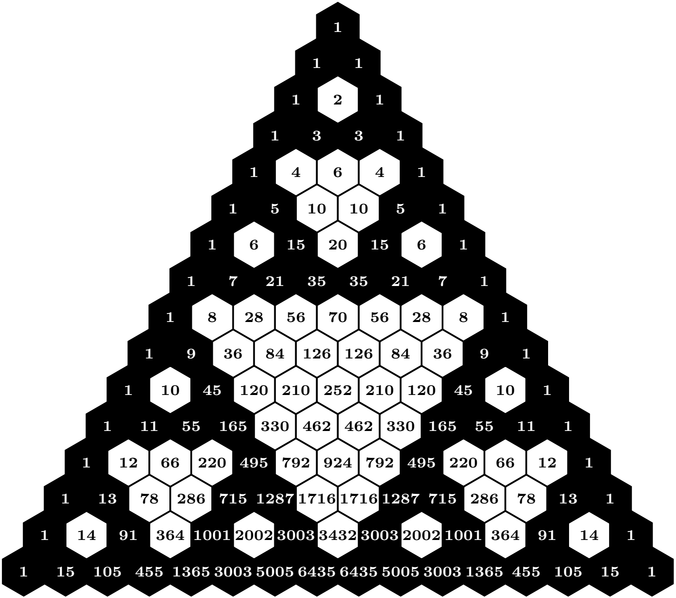


Figure: A bigger picture

Pascal's Triangle

- So how can we formulate this pattern ? We can begin by looking at binomial coefficients

$$
\begin{aligned}
(1+x)^0 &= \qquad 1 \\
(1+x)^1 &= \quad 1+1x \\
(1+x)^2 &= 1+2x+1x^2 \\
&\vdots \\
(1+x)^n &= a_0 + a_1 x + \cdots + a_n x^n
\end{aligned}
$$

where coefficients are given by

$$a_k = \binom{n}{k} = \frac{n!}{(n-k)!k!}, \quad 0 \leq k \leq n$$

Pascal's Triangle

- Checking if a binomial coefficient is odd or even by computing $\frac{n!}{(n-k)!k!}$ is a bad idea

$$
\begin{aligned}
50! =& 30414093201713378043612608166064768844377 \\
& 641568960512000000000000
\end{aligned}
$$

- Even if we use the recursive formula

$$\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$$

$$\binom{40}{20} = 137846528820 > 2^{32}$$

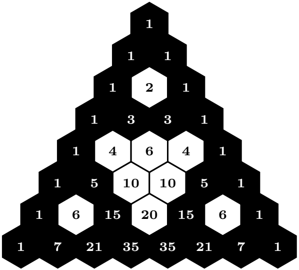
which is a big number (computer has fixed precision)

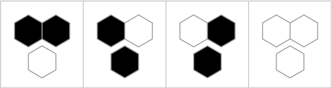
- Fortunately, we don't need to compute these large numbers

Pascal's Triangle

- In order to color a row in the triangle, we only need to know the color of the previous row

$\binom{n+1}{k} =$	$\binom{n}{k-1} +$	$\binom{n}{k}$
even	even	even
odd	odd	even
odd	even	odd
even	odd	odd





- The idea of looking at neighbor cells to determine the state of a new cell is the core idea of cellular automata

Cellular Automata

- Consists of a grid of cells, each one with a finite number of states, like 1 or 0.
- For each cell, a neighborhood is defined.
- To run cellular automata, we need 2 pieces of information
  1. Initial state of cells
  2. Fixed set of rules (rule table) that determine the state of new cell depending upon the states of current and neighborhood cells.
- The rules should not depend on the position of the group of cells within the layer.

Cellular Automata

So lets run some

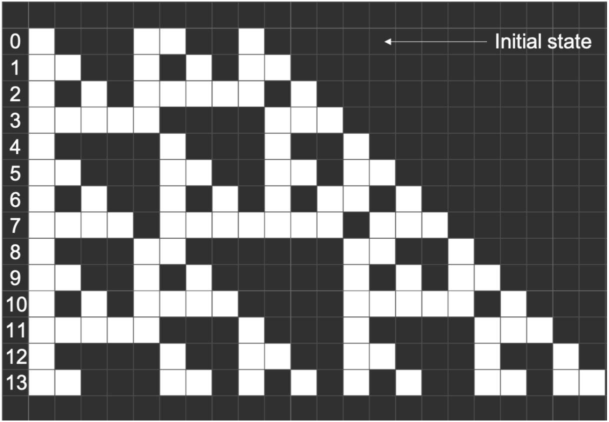


Figure: 13 Generations of a 1-D Cellular Automata

Elementary Cellular Automata

- Simplest form of Cellular Automata (CA)
- 2 State CA
- Next generation depends on itself,cell to the left and, cell to the right (Neighbors).
- All the rules can be numbered in a nice way using binary

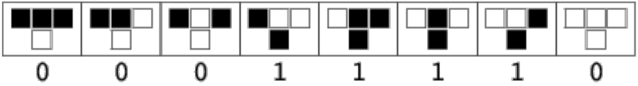


Figure: Elementary rule 30 $= (00011110)_2$

- Total number of rules are $2^{2^3} = 256$
- Mathematica can do this for you very easily using CellularAutomaton[ ]

Elementary Cellular Automata

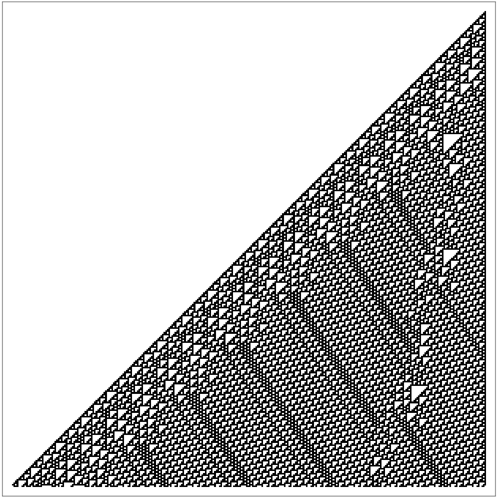


Figure: Rule 110, this might be special

Elementary Cellular Automata

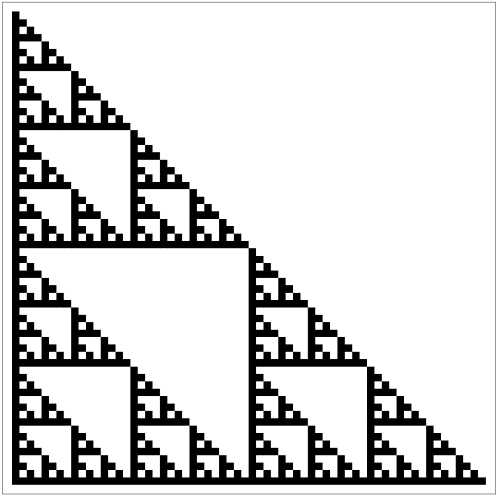


Figure: Rule 60. Sierpinski Triangle*

Elementary Cellular Automata

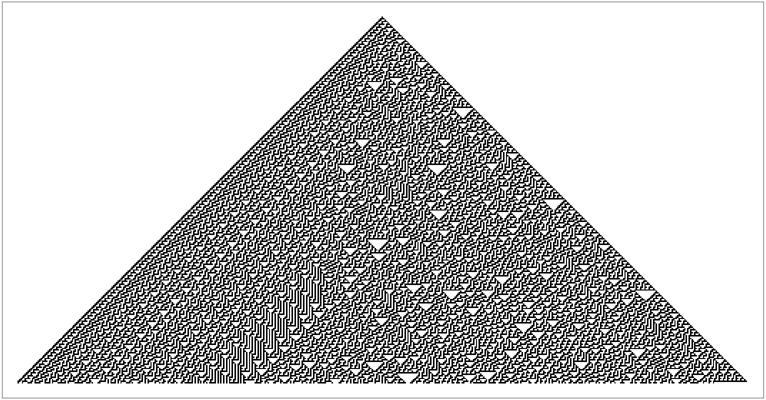


Figure: Rule 30

Kinda chaotic ?

CA and polynomials

- Let's look at the powers of $r(x) = 1 + x$:

$$
\begin{aligned}
(r(x))^0 &= 1 \\
(r(x))^1 &= 1 + x \\
(r(x))^2 &= 1 + 2x + x^2 \\
(r(x))^3 &= 1 + 3x + 3x^2 + x^3 \\
&\vdots \\
(r(x))^n &= a_0(n) + a_1(n)x + a_2(n)x^2 + \cdots + a_n(n)x^n
\end{aligned}
$$

- By the addition rule of binomial coefficients

$$
a_k(n) = a_{k-1}(n-1) + a_k(n-1)
$$

- $a_k(n)$ gives the state of n$^{\text{th}}$ layer CA*

CA and polynomials

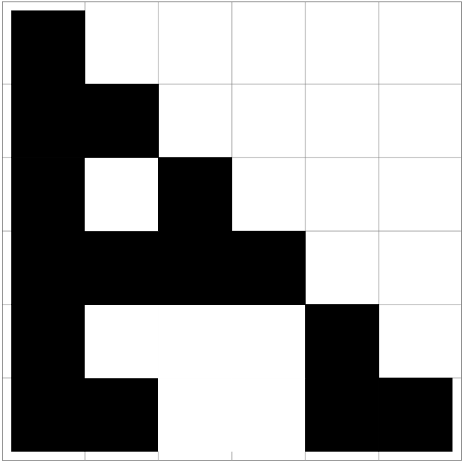
- Looking at the divisibility properties of $a_k(n)$ with 2

$$a_k(n) \equiv 0 \pmod 2 \qquad or \qquad a_k(n) \equiv 1 \pmod 2$$

- With this, the addition rules in mod 2 arithmetic simplifies to

$\binom{n+1}{k} =$	$\binom{n}{k-1}$	$+ \binom{n}{k}$
0	0	0
1	1	0
1	0	1
0	1	1

- Exactly same rule set used to color the Pascal Triangle



- Thus, this figure shows the coefficients of the powers of $r(x) = 1 + x \mod 2$

Generalizations

- There are 2 ways to generalize,
  1. A different coefficient modulo integer
  2. A different polynomial
- Ex: Let $r(x) = 1 + 2x$

$$
\begin{aligned}
(r(x))^0 &= 1 \\
(r(x))^1 &= 1 + 2x \\
(r(x))^2 &= 1 + 4x + 4x^2 \\
(r(x))^3 &= 1 + 6x + 12x^2 + 8x^3 \\
&\vdots \\
(r(x))^n &= a_0(n) + a_1(n)x + \cdots + a_n(n)x^n
\end{aligned}
$$

- By pattern,

$$a_k(n) = a_k(n-1) + 2a_{k-1}(n-1)$$

- By looking at the divisibility property with p = 3, we get

$$
\begin{aligned}
(r(x))^0 &= 1 \\
(r(x))^1 &= 1\ 2 \\
(r(x))^2 &= 1\ 1\ 1 \\
(r(x))^3 &= 1\ 0\ 0\ 2
\end{aligned}
$$

- The cellular automaton would be $a_{n,k} = a_{n-1,k} + 2a_{n-1,k-1} \mod 3$

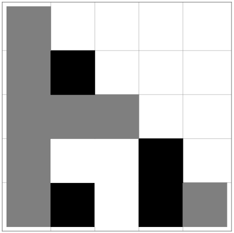


Figure: CA for $r(x) = 1 + 2x \mod 2$

Binomial Coefficients and Divisibility

- Discuss the question of whether a binomial coefficient is divisible by p or not.
- Black and white coloring of the Pascal triangle depending on divisibility with p (mod p)
- We will also see that in order to understand the patterns formed by mod p, we should look at the patters formed by the prime factors of p
- We will look at a direct, non-recursive computation of binomial divisibility by p

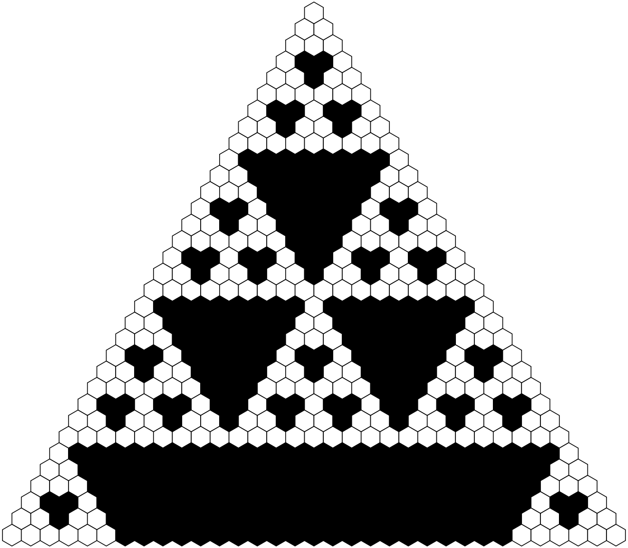


Figure: Pascal triangle Mod 3

- We define a new coordinate system such that, at position (n, k) the binomial coefficient is

$$\binom{n+k}{k} = \frac{(n+k)!}{n!k!}$$

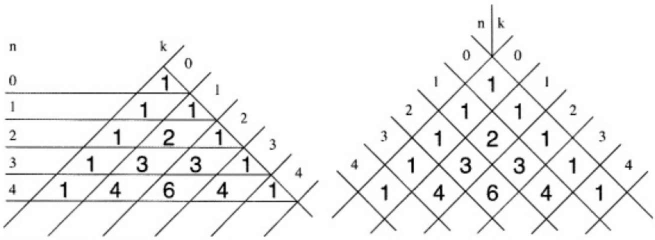


Figure: A new coordinate system

Divisibility Sets

- We formally define our problem

$$P(r) = \left\{ (n, k) \middle| \binom{n+k}{n} \text{ is not divisible by } r \right\}$$

- Observe that if p and q are two different prime numbers and a given integer r is **not** divisible by $p \cdot q$, then it is **not** divisible by either p or q

$$P(pq) = P(p) \cup P(q), \text{ if } p \neq q,\ p, q \text{ prime}$$

- It is the negation of the statement, if p and q divides r, then r is divisible by both p and q.
- Ex: $P(6) = P(2) \cup P(3)$

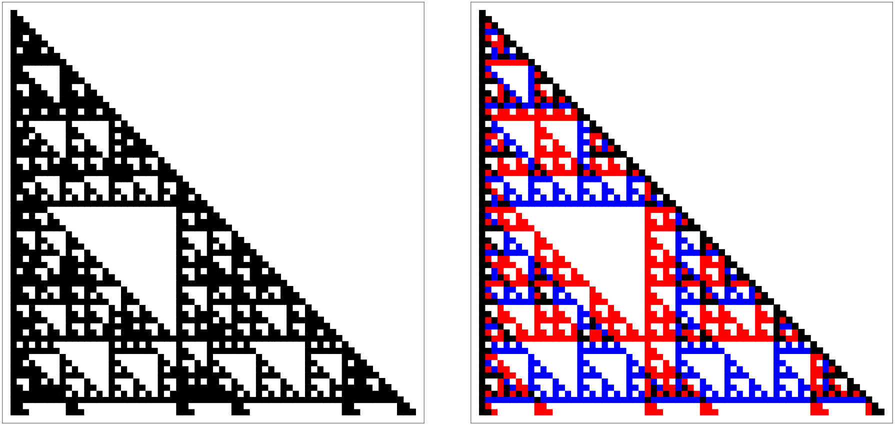


Figure: Right shows mod 6 pattern and left shows the union of mod 2 (Red) and mod 3 (Blue)

- For an integer $r = p_1^{n_1} \dots p_s^{n_s}$ where $p_1, \dots, p_s$ are primes

$$P(r) = P(p_1^{n_1}) \cup ... \cup P(p_s^{n_s})$$

- So to understand the pattern formed by $P(r)$, we just need to understand the pattern of $P(p^n)$

Kummer's Result and p-adic numbers

- To understand Kummer's result, we need to consider numbers with base p, where p is prime
- Like the decimal system, p-adic expansion of an integer is given by

$$n = a_0 + a_1 p + \cdots + a_m p^m$$

  where $a_i \in \{0, 1, \ldots, p-1\}$
- The p-adic representation would be

$$n = (a_m a_{m-1} \ldots a_0)_p$$

- Ex: $15_{10} = (1111)_2 = (120)_3 = (30)_5 = (21)_7 = (14)_{11}$

- We define carry function c_p

$$c_p(n,k) = \text{number of carries in the p-adic addition of n and k}$$

- Ex: For $n = 15$ and $k = 8$

$$k = (08)_{10} = (1000)_2 = (022)_3 = (13)_5 = (11)_7 = (08)_{11}$$

- If we take the binary addition $\begin{array}{cccccc} & 0 & 1 & 1 & 1 & 1 \\ + & 0_1 & 1 & 0 & 0 & 0 \\ \hline & 1 & 0 & 1 & 1 & 1 \end{array}$

$$c_2(15,8) = 1$$

- Similarly, we can do the same for $p = 3, 5, 7 \ldots$

Kummer's result

- Let $\tau = c_p(n, k)$, then $\binom{n+k}{k}$ is divisible by p^τ but not $p^{\tau+1}$
- So, prime factorization of $\binom{n+k}{k}$ contains exactly $c_p(n, k)$ factors of p
- This result is pretty amazing, since it gives a direct method to check if a binomial coefficient is divisible by p or not.
- For $n = 15$ and $k = 8$

$$\binom{15+8}{8} = \binom{23}{8} = 2 \cdot 3 \cdot 11 \cdot 17 \cdot 19 \cdot 23$$

- So the Kummer's result implies

$$c_p(17, 8) = \begin{cases} 1, & \text{for } p = 2, 3, 11, 17, 19 \\ 0, & \text{otherwise} \end{cases}$$

- Applying Kummer's result to Divisibility Set gives

$$P(p) = \{(n,k) |\ c_p(n,k) = 0\}$$

- That is, the number of carries $c_p(n,k)$ is only 0 if and only if

$$a_i + b_i < p, \quad i = 0, \ldots, m$$

  where a_i and b_i are the p-adic digits of n and k.
- This is called the *mod-p* condition.
- For prime powers,

$$P(p^\tau) = \{(n,k) |\ c_p(n,k) < \tau\}$$

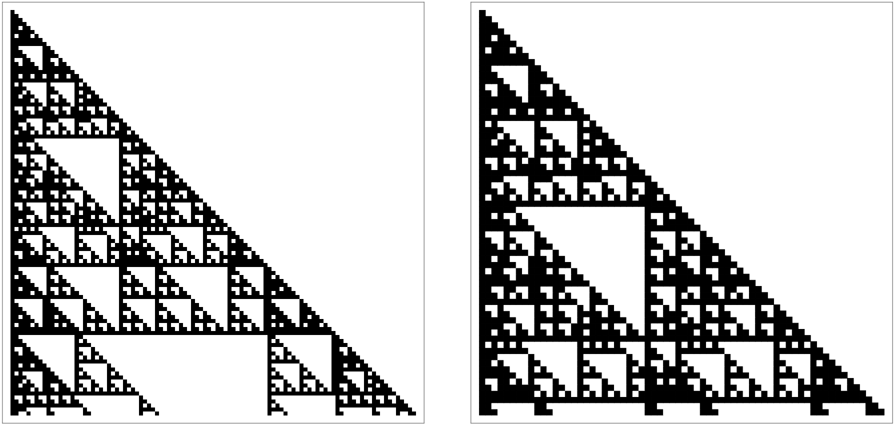


Figure: $P(6)$ generated using carry function (left) vs binomial divisibility (right)

Iterated Function System

- It is a method of constructing fractals using a set of contraction mappings.
- A contraction mapping is an affine linear transformation

$$f(x, y) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

- Union of these contraction mapping gives the Hutchinson equation of the fractal
- If you put a probability factor to these contraction mappings, you get some pretty images.

- The Hutchinson operator for Sierpinski Triangle is:

$$S = w_{00}(S) \cup w_{01}(S) \cup w_{10}(S)$$

  where w_{ij} are contraction mappings of a unit square

$$\begin{aligned} w_{00} &= (x/2, y/2) \\ w_{01} &= (x/2, y/2 + 1/2) \\ w_{10} &= (x/2 + 1/2, y/2) \\ w_{11} &= (x/2 + 1/2 + y/2 + 1/2) \end{aligned}$$

- So how do we know for sure that our Pascal Sierpinski triangle is the same as the Hutchinson operator Sierpinski Triangle?
- Let's construct Mod 2 Pascal triangle in a unit square

- Let Q be the unit square

$$Q = \{(x, y) |\ (x, y) \in [0, 1] \times [0, 1]\}$$

- Expand x and y in base 2

$$x = \sum_{i=1}^{\infty} a_i 2^{-i},\ \ a_i \in \{0, 1\}$$
$$y = \sum_{i=1}^{\infty} b_i 2^{-i},\ \ b_i \in \{0, 1\}$$

- From Kummer's result, we know that the coordinates of the points divisible by 2 are those which have no carries in the binary addition of the coordinates

$$S = \{(x, y) \in Q |\ a_i + b_i \leq 1\ \forall i\}$$

Proof

- To show both are the same, we need to show

$$w_{00}(S) \cup w_{01}(S) \cup w_{10}(S) \subset S$$

  and,

$$w_{00}(S) \cup w_{01}(S) \cup w_{10}(S) \supset S$$

- Take any point $(x, y) \in S$

$$(x, y) = (0, a_1 a_2 a_3 \ldots, 0.b_1 b_2 \ldots)$$

  where $a_i + b_i \leq 1$
- Applying the 3 transformations gives

$$w_{00}(0, a_1 a_2 \ldots, 0.b_1 b_2 \ldots) = (0.0 a_1 a_2 \ldots, 0.0 b_1 b_2 \ldots)$$
$$w_{01}(0, a_1 a_2 \ldots, 0.b_1 b_2 \ldots) = (0.0 a_1 a_2 \ldots, 0.1 b_1 b_2 \ldots)$$
$$w_{10}(0, a_1 a_2 \ldots, 0.b_1 b_2 \ldots) = (0.1 a_1 a_2 \ldots, 0.0 b_1 b_2 \ldots)$$

- Clearly, all these points are also in S.
- To show the second relation, we take any point $(x, y) \in S$, and we have to provide another point $(x', y') \in S$ such that (x, y) is one of the images of $w_{00}(x', y'), w_{10}(x', y')$, or $w_{01}(x', y')$. We choose

$$(x', y') = (0.a_2 \ldots, b_2 \ldots)$$

- We obtain

$$(x, y) = \begin{cases} w_{00}(x', y') & \text{if } a_1 = 0 \text{ and } b_1 = 0 \text{ or} \\ w_{01}(x', y') & \text{if } a_1 = 0 \text{ and } b_1 = 1 \text{ or} \\ w_{10}(x', y') & \text{if } a_1 = 1 \text{ and } b_1 = 0 \end{cases}$$

- This completes our proof.

Concequences

- The binary representation allows us to see Hutchinson operator in action applied to a point inside square
- If (x, y) is an arbitrary point in Q, the applying the map w_{00}, w_{01}, w_{10} again and again yields points with leading binary decimal points satisfying $a_i + b_i \leq 1$
- In symbols,

$$A_0 = Q$$

Then running the IFS gives

$$A_n = w_{00}(A_{n-1}) \cup w_{01}(A_{n-1}) \cup w_{10}(A_{n-1})$$

where the leading n binary digits satisfy $a_i + b_i \leq 1$

- Finally, the sequence will lead to Sierpinski triangle

$$A_\infty = S$$

IFS for other primes

- Now, we can finally look into the global pattern formation of divisibility of binomial coefficients with primes, i.e the global patterns formed by

$$P(r) = \left\{ (n,k) \,\middle|\, \binom{n+k}{n} \text{is not divisible by } r \right\}$$

- We construct an IFS: Divide the unit square Q in p^2 congruent square $Q_{a,b}$ with $a, b \in \{0, \ldots, p-1\}$. We introduce the contraction mappings

$$w_{a,b}(x,y) = \left(\frac{x+a}{p}, \frac{y+b}{p} \right)$$

  where

$$w_{a,b}(Q) = Q_{a,b}$$

- Then we set the restriction to define the set of transformation

$$a + b \leq p - 1$$

- This restriction follows from Kummer's result,i.e $\binom{n+k}{n}$ is indivisible by p when there is no carries in the p-adic addition of n, k
- The Hutchinson operator for these contractions,

$$W_p(A) = \bigcup_{a+b<p} w_{a,b}(A)$$

Conlcusion

- We approached the problem of coloring (or divisibility) of pascal's triangle in 3 different ways
  1. First, we looked at the macroscopic nature of divisibility, i.e how the neighbors affected the divisibility of a cell. We then explored the realm of cellular automata and how it extends this idea to general polynomials. We also peaked into some open ended problems
  2. Second, we looked at the microscopic nature of divisibility,i.e how the divisibility of a cell depends on its coordinate (Kummer's Result). We looked how p-adic numbers gave insight about binomial divisibility with primes. We also extended this idea of divisibility to all integers.
  3. Finally, we looked at the global pattern using Iterated function system. We showed that the limit of Pascal Mod 2 pattern was indeed the Sierpinski triangle. We also explored the global patterns of other primes and their IFS.

- Each method gave a different perspective of the same problem.
- This entire chapter is trying to solve the jig-saw puzzle that relates fractals, Pascal's triangle, and Cellular Automata.
- Understanding that these 3 very different things are all tied up by one common thread is, honestly, amazing.

Thank You!