Orbiting

some notes

Daniel K. Rui - October 5, 2026

Contents

0.1 0. Prologue . . . 2

1 Preliminaries **17**

2 Monday **18**
2.1 . . . 19
2.2 . . . 19
2.3 . . . 19
2.4 Indegree and Outdegree Lemma . . . 19

3 Tuesday **20**
3.1 Pruning? . . . 20

4 Wednesday **21**
4.1 Tail Eqrel is Hsmooth . . . 21
Simple Case . . . 21
General Case . . . 21
4.2 Limsup proof . . . 21
4.3 22 . . . 22
Bamboo-ing implies Hfin . . . 22
Pruning . . . 22
4.4 23 . . . 22
Extension Lemma . . . 22
Subadditivity of Normalized Cost: Aper . . . 22
Subadditivity of Normalized Cost: Commutative . . . 22

5 Thursday **23**
5.1 25 . . . 24

6 Sunday **24**
6.1 Key Idea 1: "folding" (homotopy) . . . 24
6.2 Producing the treeing $\Psi_1 \sqcup \Psi_2$ of $E_1 \cap E_2$ . . . 25
6.3 Key Idea 2: disjointify . . . 25

Temporary parasitism from Countable Combinatorics doc

You have been ordained as a powerful diety, tasked with the stewardship of a planet of humanoids called Cantorites. There are countably infinite Cantorites on their (infinite) planet, and you have lovingly given them all unique True Names $1, 2, \ldots$, but crucially, you do not tell them their True Names.

Each Cantorite (and you as their diety) has the amazing ability to essentially instantaneously read and write (and type, etc.) countably infinite bitstrings, e.g.

$$0101010100111010111011011111110\ldots$$

The Cantorites also have computers that can also read/write/input/output/display **countably infinite bitstrings** — this phrase we henceforth abbreviate to the term "bitstreams".

You give each Cantorite x a unique bitstream Name$(x) \in 2^{\mathbb{N}}$ as a Name, which they do know (unlike their True Name $\in \mathbb{N}$).

Each Cantorite in fact knows the Names of all other (countably infinite many) Cantorites.

No

Before you are countably many computers (which in your own head you can enumerate/christen as $1, 2, \ldots$), wired together in a network $\mathcal{C}$ ("C" for "class"). These are very special computers: they can, essentially instantaneously, read/write/input/output/display/manipulate **countably infinite bitstrings** — this phrase we henceforth abbreviate to the term "bitstreams".

Each computer x knows its own unique bitstream ID/Name Name$(x) \in 2^{\mathbb{N}}$. Each computer also knows the Names of all the other computers in the network, and knows the Names of the computers that any computer is directly connected/wired to. You can think of it as: each x has stored in its countably infinite memory a 2-column table, the 1st column listing out all computers $x' \in \mathcal{C}$, and 2nd column in row x' listing all Names of computers direct connected to x'.

Crucially, we do NOT assume that each "local list" (stored in each computer) is in the same order. Each individual for themself.

With just this information, it turns out that we can design some algorithms that each computer can run Privately and Independently (so no communication, no back-and-forth, no conversations, etc.) that can solve some interesting problems. Think of this as a toy model for distributed computing; it is an interesting and unexpected phenomenon that sometimes the infinite is easier to deal with than the finite. For example, infinity is much more symmetric/"fractal" than any finite number: e.g. $\mathbb{N} = \{\text{evens}\} \cup \{\text{odds}\}$ where each piece of the partition is essentially a rescaled copy of $\mathbb{N}$. No such finite number has this property. Said more crudely, $\frac{\infty}{2} = \infty$.

0.1 0. Prologue

GOALS: familiarity with Cantor space.

https://chat.deepseek.com/a/chat/s/86a073b3-297f-4cea-b267-2469eeab4b99

Question	Answer

Describe 5 infinite strings $x_1, \ldots, x_5$ made only out of 0's and 1's (these are called "bitstreams"). Obviously we can't write down all infinitely many digits, so in what way *can* one describe such things? Pair up and play a game: one person thinks of a bitstream, and tries to describe it in words, and the other person writes down as much of it as is appropriate. The pair wins if you agree that you both have the same bitstream in mind.	
A student says "The nth bit written is 0 if n is a prime integer, 1 otherwise." Write the first 15 terms. Is this a complete description? Why or why not?	
A student says "Start with 1 and 0. Then take what you have so far, and copy paste it right after what you have so far. Keep doing this". Write the first couple terms. Is this a complete description? Why or why not? Is there another description? Do the same for a student saying "The nth bit is $3^n \mod 2$".	
Consider the rule: start with 0, then repeatedly append the complement of what you have so far. So $0 \rightsquigarrow 01 \rightsquigarrow 0110 \rightsquigarrow 01101001 \rightsquigarrow \ldots$. Is this a "complete description" of an infinite sequence? Why or why not? Write the next two blocks (yes, seriously, because we will keep studying this sequence). What is the 253th digit? (Our convention is 0110... $\rightsquigarrow$ 1st digit is 0, 2nd digit is 1, etc.)	
Can you give a bitstream that not periodic? (You should be able to prove that the sequence you give is not periodic.)	
Is there a bitstream that is "square free" (a square is a string of the form $s * s$, i.e. s concat s)? Why or why not? (Bonus; doable upon deeply understanding the Thue-Morse sequence facts below + playing around) What if I allow you to use instead the "letters" 0,1,2 instead of just 0,1?	9/17/26 Ok the 2nd question turned out hard. Even with my hint of use Thue-Morse sequence, took an hour. Worked together with Math Circle assistants BC and MB, BC key breakthrough was: $0, 1, 2$ is mod 3, but $-1 = 2$, so want sequence in $\{-1, 0, 1\}$. We start with T-M (cuz I said so), how to get to -1, subtract. What to subtract? Well, only have T-M, so try T-M, but shift. Compute the first 30 terms. Holy fuck, it worked. Turns out BC's sequence is mod 3 negation of the 1st example in https://en.wikipedia.org/wiki/Square-free_word. Aha, need stronger than cube-free, need $(2+\epsilon)$-free (which is indeed proven below). In retrospect, $(2+\epsilon)$-not free, i.e. $(2+\epsilon)$-repeat $\iff$ the arrows $(f(k+1) - f(k))$ in between (going up, staying same, going down) form square.
If we think of 1 as white, 0 as black, is there a bitstream with no monochromatic 3-AP (3 equally spaced entries x_a, x_{a+d}, x_{a+2d} (separated by distance d) of the same color)?	
Come up with a definition of "eventually periodic" sequence. What is the data needed to specify one? Give 3 examples.	
Define a sequence by $x_n = 0$ if n is a power of 2, 1 otherwise. Is it periodic? Eventually periodic? Give a full justification/argument.	
Inspired by the Fibonacci sequence ("next = current + previous") define a "Fibonacci bitstream" (start with 0, then extend it to 01, and so on). What interesting properties about this sequence do you notice?	Balanced
Look at this picture https://en.wikipedia.org/wiki/Sturmian_word#/media/File:Fibonacci_word_cutting_sequence.png. Just looking at the picture, define the notion of a "cutting/billiards sequence" for an irrational slope θ. On Desmos, try out different values of θ. Does the sequence in the picture look familiar?... We will justify this later.	Fibonacci

Define the Thue-Morse bitstream t as follows: the nth digit t_n is the binary digit sum of n mod 2. Write out the first 20 terms. It should look familiar... justify the coincidence with a full argument. Questions: (1) If I promise you that $t_{2n} = 1$, what can you say about t_{2n+1}? (goal is to notice: change lsb from 0 to 1 always flips the total sum mod 2, so we would have to have $t_{2n+1} = 0$.) (2) prove that 000 and 111 do not appear in t. (easy with previous "mandatory bit flip" obs) (3) $t_{2n} = t_n$. I.e. taking every other bit of t starting at $n = 0$ gives back t. (4) $t_{2n+1} = \neg t_n$. I.e. taking every other bit of t starting at $n = 1$ gives "color inversion" of t. (5) in any 5 consecutive elements of the sequence, there must be appear 00 or 11. (proof: enemy case is 01010. But then taking every other term gives 000, which is impossible by (3)&(4) + (2).) (6) Prove that it is impossible that there is $a \geqslant 0, d > 0$ s.t. $s_a = s_{a+d} = s_{a+2d}$ and the substrings/"blocks" $s_{[a..a+d)}, s_{[a+d..a+2d)}$ are identical. (proof: induct on d. If d even, like in 01010 case in (5), done by (5) technique + induction. If $d \geqslant 3$ odd, then either $\exists$ double inside block like [0110], or at the end of 1 block like [01[00][00]10]. But then when repeat in 2nd copy, the double's 1st index changes parity. So we have one double (say 00), appearing at odd and even indices, contradicting (1).) (7) go through the whole argument and conclude t is cube free. (in fact (6) shows "$2 + \epsilon$"-free)	Our goal is to build (with proof) a "cube free" bitstream x, in the sense that there is no finite bitstring s so that sss appears in x. ~~What about one that is "cube free", in the sense that there is no finite bitstring s so that sss appears in x? Probably can't do this, too hard.~~ Need to intro Thue-Morse first, then scaffold proof as in https://www.cs.umb.edu/~offner/files/thue.pdf. Cube free is famous construction, I forgot how it is proved tho. Start with 0, then invert and append. So $0 \rightsquigarrow 01 \rightsquigarrow 0110 \rightsquigarrow 01101001 \rightsquigarrow 0110100110010110$. I guess prove by induction each iteration is cube-free? OLD SCAFFOLD (I added new one 9/25/26): Define the Thue-Morse bitstream t as follows: the nth digit t_n is the binary digit sum of n mod 2. Write out the first 20 terms. It should look familiar... justify the coincidence with a full argument. Prove: (1) $t_{2n} = t_n$. I.e. taking every other bit of t starting at $n = 0$ gives back t. (2) $t_{2n+1} = -t_n$. I.e. taking every other bit of t starting at $n = 1$ gives "inversion" of t. (4) prove that if $t_j = t_{j+1}$, then j must be odd. (5) 000 and 111 do not appear in t. (6) in any 5 consecutive elements of the sequence, there must be appear 00 or 11. (7) Prove that it is impossible that there is $a \geqslant 0, d > 0$ s.t. $s_a = s_{a+d} = s_{a+2d}$ and the substrings $s_{[a..a+d)}, s_{[a+d..a+2d)}$ are identical. (8) go through the whole argument and conclude t is cube free.

Start with 0, and keep applying the following "local manipulation" rule $[0^{\text{old}} \rightsquigarrow 01, 1^{\text{old}} \rightsquigarrow 0]$. Look familiar? Justify this coincidence (induct...). Kinda crazy that this "local evolution" rule, and the previous "global manipulation" rule produce the same sequence of strings!	Fibonacci
Start with the line segment between $(0, 0)$ and $(1, 1)$. Now apply two linear transformations: reflect across line $y = x$, then shear (linear transformation). What does this have to do with the "local manipulation" rule from before? Using what you figure out, prove that the "global manipulation" rule produces the same bitstream as the cutting/billiard sequence for $\theta = \lim_{n\to\infty} \frac{F_n}{F_{n+1}}$ pictured above.	$(1, 1) \rightsquigarrow (1, 1) \rightsquigarrow (2, 1)$. And $(2, 1)$ transforms to $(1, 2)$ and then $(3, 2)$. Hit corner point e.g. $(1, 1)$ interpret as hitting vertical=0, so $(0, 0)$ •–• $(1, 1)$ interpreted as cutting/billiard sequence "0". And $(0, 0)$ •–• $(2, 1)$ interpreted as "01", and $(0, 0)$ •–• $(3, 2)$ interpreted as "010". I figured out this proof by reading https://researchers.ms.unimelb.edu.au/~pjforr\spacefactor\@m{}unimelb/publications/HonoursTheses-DavidGlover.pdf especially Lemma 1.4.2 pg. 18
Given any string s, let the "weight" $w(s)$ be the number of 1's in s. Show that for any cutting/billiards bitstream z, the weights of length-n subwords of z can be 1 of at most 2 possible values. (BTW, look up "Sturmian word". This is the start of a big long story in "symbolic dynamics"!)	I also figured out by reaading https://researchers.ms.unimelb.edu.au/~pjforr\spacefactor\@m{}unimelb/publications/HonoursTheses-DavidGlover.pdf, "homotopical argument": think about moving line segment; as it crosses an interior lattice point, 01 flips to 10.

Konig, pigeonhole, Ramsey added 9/28/26

Question	Answer

[Picture of Cantor tree, growing down]. Each node is associated with a finite binary string. For example, the string "01011" corresponds to the node we reach by starting at the root, and then walking down 0, then 1, then 0, then 1, then 1. What do the infinite branches (starting at root) in this Cantor tree correspond to?	
Draw 5 examples of finite trees. (Each tree will grow downward from its root.) Try to draw ones that "look different" from each other! Write out the sizes/widths of each level (level-0 = root), and write the height of the tree (largest N so that level N is not empty).	
Describe (i.e. draw a "representative" finite portion of) an example of a tree with infinitely many nodes, but has no path of infinite length.	Dandelion/starfish.
Suppose we have an infinite tree $\mathfrak{T}$ growing down from a root (infinite means it has infinitely many vertices). And suppose each parent node only has finitely many children — this is called having "finite-width levels". Draw an example of such a tree. Prove that such a tree MUST have an infinite branch. Contrast this with the previous exercise. (This is a very important result in mathematics, called König's lemma.) If you get stuck on this problem, move on to the next exercises and come back later (one of the later problems is a hint for this problem!).	
Suppose we have a tree $\mathfrak{T}$ growing down from a root. Do NOT assume it has finite-width levels. Describe (i.e. draw a "representative" finite portion of) an example of such a tree $\mathfrak{T}$, that has no infinite branch, but that has infinite height (meaning every level from level-0=root, to level-n for any natural number n, is not empty).	
Let's think about the tree $\mathfrak{T}$ in the previous problem in a different way. Suppose you are standing at the root. At the end of each branch in your tree, is a bomb. Suppose you are forced to make a move. Then the previous exercise is saying that you will always eventually die, but you can choose to live for however long you want. Does that make any sense? How is this somewhat mind-boggling idea related to the following infinite chess position? https://mathoverflow.net/questions/63423/checkmate-in-omega-moves	
Suppose we have a tree $\mathfrak{T}$ growing down from a root, with finite-width levels. Suppose $\mathfrak{T}$ has NO infinite branches. Prove that $\mathfrak{T}$ is finite height (there is a level N so that the tree lies entirely above level N). Hint: use König's lemma ("contrapositive", if you know what that means).	
Consider the Cantor tree (infinite binary tree). Suppose there is a runner, Rolfson, standing at the root. Unfortunately, someone evil has placed pressure-mines at some configuration of the nodes in the Cantor tree. Suppose that the configuration is such that Rolfson can always find a path p_{100} to reach level 100. And a path p_{1000} to reach level 1000 (though maybe not related to p_{100}). And so on, for every natural number n. (In other words, Rolfson can manage to avoid all the (finitely many) mines at levels $\leqslant 100$. And $\leqslant 1000$. And etc.) Prove that Rolfson can in fact avoid all the mines, and run on forever into the sunset. (This is a rephrasing of the extremely important fact that Cantor space is a compact topological space. "Given a collection of basic opens, if every finite subcollection does not cover the whole space, then the collection itself doesn't cover the whole space.") Contrast this with the previous exercises. Try to make sense of it all in your head. (This is a famously mind-boggling topic!)	OLD version (unfortunately police chief idea doesn't make too much sense): Consider the Cantor tree (infinite binary tree). Suppose there is a thief, Thomas, standing at the root. The police chief has chosen infinitely many nodes to station a cop, in such a way that no matter how Thomas travel down this tree (i.e. no matter what sequence of left/right choices Thomas makes), Thomas will encounter a cop. (Too bad for Thomas!) Prove that it is possible to retain only finitely many cops in their original positions (and fire the rest), and still be able to catch Thomas (again no matter what choices Thomas makes).
Same situation as previous problem. Suppose that there is at most 1 mine per level $n \geqslant 1$. Prove that Rolfson (assuming he chooses left/right correctly always) can run forever.	Start at root. At most 1 mine on level 1, so can make first move safely. Same argument for level 2. Same argument for every n. Apply previous problem.

Pigeonhole, Ramsey, revisit the "3 AP = 3 equally spaced" problem from Bitstream Intro packet. I remember Gowers on

Numberphile had a quick proof https://www.youtube.com/watch?v=kE3OuzlkUnU, really like a comment " [at]prdoyle 6 years ago 7:52 - woah, mind blown. That was like a magic trick."

Suppose I have infinitely many balls (labelled say 1,2,3...), and 2 bins. Prove that no matter how I place the balls, there will be a bin with infinitely many balls. (This is a very important fact called the infinitary pigeonhole principle.)	
If you didn't like the non-physicality of infinitely many balls, what is the minimum number of balls so that no matter how I place the balls in 2 bins, there will be a bin with 3 balls? (This is an instance of the finitary pigeonhole principle.)	
Suppose I have 6 dots (say arranged as vertices of a regular hexagon), and I draw all pairs of lines between 2 vertices. How many lines do I draw? Suppose I color each line blue or red. Show that no matter how I color, there will always be a blue triangle or red triangle.	
Suppose I have infinitely many balls, and between each ball is a spring colored blue or red. Prove that I can choose infinitely many of these balls, place them into a bin, so that all the springs inside that bin (i.e. both endpoints are in the bin) are the same color.	
What if in the previous problem I had 3 colors: blue, red, yellow? What about 4?	
† Suppose I have infinitely many balls in a line labelled ①, ②, ③..., each colored either blue, red, yellow. (Note here, the balls are colored, whereas above, the springs between 2 balls are colored.) Prove that there are 3 balls of the same color, labelled $x < y < z$ respectively, so that $x + y = z$. Such triples are called "Schur triples". (Hint: use the coloring of the balls to get a coloring of the springs.)	Color the spring (i, j) (between ball i, and ball j) the color of the ball $(j - i)$. Profit. https://proofwiki.org/wiki/Schur%27s_Theorem_(Ramsey_Theory) inspires the next problem.
† Prove that no matter how the nats $1, 2, \ldots$ are colored by blue, red yellow, there are integers $x, y, z \in \mathbb{N}^+$ so that $\{x, y, z, x + y, y + z, x + y + z\}$ are all the same color. (Hint: use the coloring of the balls to get a coloring of the springs.)	Color the spring (i, j) (between ball i, and ball j) the color of the ball $(j - i)$. Having 4 balls with springs-between of the same color yield $\{l - k, k - j, j - i, l - j, k - i, l - i\}$ all the same color. Call the first three z, y, x. Then all 6 read $z, y, x, z + y, y + x, z + y + x$. https://proofwiki.org/wiki/Schur%27s_Theorem_(Ramsey_Theory)
You may use the previous exercise as a black box. Prove that there is an integer N, so that no matter how you color the nats from $[1..N]$ by blue, red, yellow, then there is guaranteed to be a Schur triple, i.e. nats $x < y < z \leqslant N$ of the same color, so that $x + y = z$. (Hint: use König's lemma.)	
Suppose I have a "ratstream", i.e. an infinite sequence of rational numbers. For example, $(a_1, a_2, \ldots) \coloneqq (-1, \frac{1}{2}, -\frac{1}{3}, \frac{1}{4}, \ldots)$. Here's one way to visualize ratstreams ("as buildings"): draw a horizontal numberline, and draw a rectangle with base $[-1, -\frac{1}{2}]$ and height a_1. Then draw another rectangle with base $[-\frac{1}{2}, -\frac{1}{3}]$ and height a_2, and so forth. Do this for the ratstream I gave you, and also another new ratstream that you come up with.	
A ratstream $(a_1, \ldots)$ is **weakly increasing** if $a_1 \leqslant a_2 \leqslant \ldots$. Define "weakly decreasing". Try to define what you think a **weakly increasing subsequence** of a ratstream is.	
Prove that any ratstream has either a weakly increasing subsequence or a weakly decreasing subsequence (it can have both, but we just want to guarantee it has at least one). (This is one of the most important theorems in undergraduate real analysis.)	Hint: visualize the ratstream as buildings, as above. Draw the line $x = 1$, color it blue. Say a building (in the ratstream visualization) has an "ocean view" if some part of it can see the blue line (unobstructed by other later buildings). Question: do infinitely many buildings have an ocean view? If yes, ...; if no, ...! Alternative proof: make a graph with vertex set $\mathbb{N}$, and color edge (i, j) red if $a_i \leqslant a_j$ and blue if $a_i > a_j$. Then use previous exercise.
Consider the sequence of 10 natural numbers $4, 6, 1, 9, 2, 3, 8, 5, 10, 7$. What is the longest increasing subsequence (LIS) that ends at 8? What is the longest decreasing subsequence (LDS) that ends at 5? Go left to right, and calculate the lengths u_k, d_k of the LIS and LDS ending at each position k. Why is it important/helpful/computationally efficient that you go left to right? (This is a prototypical example of something called a dynamic programming algorithm.)	

Do you notice anything about the pairs (u_k, d_k) for $k = 1..10$? Use the insights you gained from your previous computations/algorithms, to prove that if I have a list of 10 distinct integers, there must exist at least one increasing subsequence of length 3, or decreasing subsequence of length 3.	
How long must my list be, to guarantee there must exist at least one increasing subsequence of length n, or decreasing subsequence of length n?	
Above you saw infinitary and finitary versions of the same theorems. Which did you like more?	

van der Waerden (see my above comments on Gowers Numberphile). Also wow https://www.youtube.com/watch?v=XHMQLcjKicI this IAS talk on 3 color vdW numbers just dropped 13 hours ago!

Find an example of a bitstring of length 8 with no monochrome 3AP	11001100. There are 6 total, but annoying casework to show this.
Suppose I have a sequence of 9 bits, but I only tell you the middle third [...001...] (each dot = 1 empty spot). Suppose that I tell you that my sequence has no monochrome 3AP. Prove that I'm lying. (Hint: assume I'm telling the truth. That will force more and more structure, which you chase until you reach a contradiction.)	
Prove that every bitstring of length 9 has a monochrome 3AP. Hint: start with 9 spaces = [...][...][...], and consider the cases of what can happen in the middle third; then work your way out.	9 spaces = [...][...][...]. Look at middle block. Either [101] or [001] up to bitflip and string reversal symmetry. First case is easy. 2nd case, is a sequence of forces [..1001..] ⇝ [.1001..0] ⇝ [1.1001..0] ⇝ [101001.10] ⇝ [101001.10], ※. (Proof modified from one temporary ChatGPT gave me. Really wonderful proof, like pong bouncing back and forth! So glad I saw this, even if so late in my learning.) I actually isolated the 2nd case into immediately preceding exc, f.s.o. scaffolding
Suppose you are working in a group project, where the task is to color the number line $1, 2, 3 \ldots$ (in order) in blue/red sharpies, in such a way that there is no monochrome 6AP. Your grade is $N/1000$, where N is the largest number so that your blue/red coloring of $[1..N]$ has no monochrome 6AP. Unfortunately, some of your group members has already colored in positions $101, 201, 301, 401, 501$ with copies of the SAME length-36 block of the form $[R.....R.....R.....R.....R.....]$ (e.g. positions 101-136 are colored $[R.....R.....R.....R.....R.....]$). Prove that your group will definitely get a bad grade.	This isolates the "magic trick" in Gowers Numberphile video. Using that 2-color, 5AP vdW number is >100. https://en.wikipedia.org/wiki/Van_der_Waerden_number. Previously I wrote "with length-29 blocks of the form $[R.....R.....R.....R.....R]$", which is wrong!
We have 3 colors now: B,R,Y. You are in another group project, where the task is to color the number line $1, 2, 3 \ldots$ (in order) in B/R/Y sharpies, in such a way that there is no monochrome 3AP. Your grade is $N/100$, where N is the largest number so that your B/R/Y coloring of $[1..N]$ has no monochrome 3AP. Unfortunately, some of your group members has already colored in positions $11, 31$ with copies of the SAME length-9 block of the form $[R.Y.RY...]$ (e.g. positions 11-19 have colors $[R.Y.RY...]$). Prove that your group will definitely get a bad grade.	
We have 3 colors now: B,R,Y. A 4-block is a block of colors of size 4: [....], e.g. [BRRY]. A 2^+AP (in a colored block) is defined as 3 equally-spaced positions, where the first 2 positions have the same color. Each 4-block is guaranteed to have a 2AP (why?), so each 7-block is guaranteed to have a 2^+AP (why?). Now on an empty numberline, we arbitrarily place 2188 7-blocks, one after the other, starting at position 1. Call $M \in \mathbb{N}^+$ the last position colored; what is it numerically? Prove that if you color $[1..2M]$ with 3 colors B/R/Y arbitrarily, you will always see a pattern similar to the 9-block $[R.Y.RY...]$ in the previous exercise ("two 2APs focusing on the same endpoint").	
Combining the insights of the previous exercises, find an explicit value $N \in \mathbb{N}^+$ so that if you color $[1..N]$ arbitrarily by 3 colors, you will always find a monochrome 3AP.	

Familiarity with the background behind Ellentuck theorem.

Question	Answer
Think of a natural total order $\leqslant_{\text{lex}}$ on bitstreams.	Lexicographical
Come up with an infinite decreasing sequence $z_1 \geqslant_{\text{lex}} z_2 \geqslant_{\text{lex}} \ldots$. Does it have a lower bound?	
Prove that bitstreams $\in 2^\omega$ correspond to subsets of $\mathbb{N}$ (so given a bitstream, produce a subset; and given a subset, produce a bitstream — in such a way that doing one operation after the other is the same as doing nothing). I.e. go between language of bitstreams and subsets.	
Consider the subclass of bitstreams that have infinitely many ones — let's call it $\text{Inf1}(2^\omega) \subset 2^\omega$. What does this subclass correspond to in the language of subsets?	Infinite subsets of $\mathbb{N}$.
Come up with an infinite decreasing sequence $z_1 \geqslant_{\text{lex}} z_2 \geqslant_{\text{lex}} \ldots$ where each $z_i \in \text{Inf1}(2^\omega)$, which has no $\leqslant_{\text{lex}}$-lower bound $z \in \text{Inf1}(2^\omega)$.	$11\ldots \geqslant 01\ldots \geqslant 001\ldots \geqslant \ldots$
I'll define a new partial order, called "tail domination" $\leqslant_{\text{taildom}}$. Come up with 2 bitstreams that are incomparable w.r.t. this order	
Come up with 2 bitstreams x, y so that $x \leqslant_{\text{lex}} y$ but $x \not\leqslant_{\text{taildom}} y$	$0111\ldots \leqslant_{\text{lex}} 1000\ldots$
Come up with 2 bitstreams x, y so that $x \leqslant_{\text{taildom}} y$ but $x \not\leqslant_{\text{lex}} y$	$1000\ldots \leqslant_{\text{taildom}} 0111\ldots$
Come up with a sequence of infinitely many bitstreams $z_1, z_2, \ldots \in \text{Inf1}(2^\omega)$ that is both $\leqslant_{\text{lex}}$- and $\leqslant_{\text{taildom}}$-decreasing, and has no $\leqslant_{\text{lex}}$-lower bound in $\text{Inf1}(2^\omega)$.	Chances are the one you came up with above already works.
Consider your sequence of infinitely many bitstreams $z_1, z_2, \ldots \in \text{Inf1}(2^\omega)$ above (that is both $\leqslant_{\text{lex}}$- and $\leqslant_{\text{taildom}}$-decreasing). Does your sequence of bitstreams have a $\leqslant_{\text{taildom}}$-lower bound?	Yes, it must.
Useful Theorem: Prove that every infinite sequence of $\leqslant_{\text{taildom}}$-decreasing bitstreams has a $\leqslant_{\text{taildom}}$-lower bound.	See my Illustrations

G0-dichotomy, bitstream graphs. I've decided that any part of the proof of G0 is too hard to shoehorn into Math Circle. However, due to my thinking about it, I think that some Banach-Mazur game theory (https://math.stackexchange.com/questions/800744/banach-mazur-game-proof-about-winning-strategies full proof which I forced Deepseek to explain to me better https://chat.deepseek.com/a/chat/s/f5ead31d-2ea3-4e62-bdc0-50015eff0153), and maybe Evan's countable circuits could make it in (though his analytic separation is not trivial; very nice, but have to think about it). Then after seeing how that goes, then maybe G0's Baire measurable stuff and analytic separation stuff could be next step.

Question	Answer

8/15/26 in HK

Question	Answer
A set is "countable" means I can have a list (indexed by $\mathbb{N}$) so that every element $s \in S$ appears at least once. (1) Prove $\mathbb{Z}$ is countable. (2) $\mathbb{Z} \times \{0,1\}$ (3) $\mathbb{N}^2$ (4) $\mathbb{Q}$. And $\mathbb{Q}^2$ (5) set of finite binary strings (6) set of finite decimal strings (7) set of all finite lists of finite binary strings (8) $\mathbb{N}^{<\infty}$ (9) algebraic numbers	(7) treat comma as a character
Describe an (interesting, i.e. each one "novel") infinite list of bitstreams. Then give an example of a bitstream not on that list.	
Describe an algorithm that a countably infinite being can "run", that takes in an arbitrary list of bitstreams, and returns a bitstream not on the list. What does this mean in terms of $2^{\mathbb{N}}$ and countability?	
Geometry of rational numbers in line: draw the number line. Draw the "lattice" $L := \frac{1}{5}\mathbb{Z}$. Is every other rational $q \in \mathbb{Q}$ within distance $< \frac{1}{5}$ of L? Is there a minimal distance $d \in \mathbb{Q}$ so that every q is within distance $< d$ of L? What about is there a minimal distance $d \in \mathbb{Q}$ so that every q is within $\leqslant d$ of L?	
Geometry of rational numbers in plane: draw the plane. Draw the "lattice" $L := \frac{1}{5}\mathbb{Z}^2$. Is there a minimal distance $d \in \mathbb{Q}$ so that every q is within distance $< d$ of L? What about is there a minimal distance $d \in \mathbb{Q}$ so that every q is within $\leqslant d$ of L?	
	"Ball" always means "open ball". Prove every ball in the plane $\mathbb{R}^2$ contains a rational ball. Originally I wanted this exercise, but I'll just keep to rational balls. Only place "reals" will show up is nested intersection property.
Give an example of a countable decreasing sequence of open balls, so that the intersection of all balls is empty (i.e. there is no point $(x,y) \in \mathbb{R}^2$ that is in every single ball of the countable sequence).	maybe this is here, and have stuff on developing real analysis later, see 8/28/26
Consider the comfortably nested open chain $(B_1, B_2, \ldots) := B(\binom{1.1}{0}, 0.5) \supsetneq B(\binom{1.11}{0}, 0.05) \supsetneq B(\binom{1.111}{0}, 0.005) \supsetneq \ldots$. Draw this, and determine $\bigcap_n B_n$ in the plane.	
This is a (not comfortably) nested closed chain $[0,1] \supsetneq [0, \frac{1}{2}] \supsetneq [\frac{1}{4}, \frac{1}{2}] \supsetneq [\frac{1}{4}, \frac{5}{8}] \supsetneq [\frac{7}{16}, \frac{5}{8}] \supsetneq \ldots$. Draw this, determine the pattern, and determine $\bigcap_n B_n$ in the numberline (it's ok to not be rigorous).	
This is a (not comfortably) nested closed chain $[0,1] \supsetneq [\frac{1}{5}, \frac{4}{5}] \supsetneq [\frac{6}{25}, \frac{19}{25}] \supsetneq [\frac{31}{125}, \frac{94}{125}] \supsetneq$. Draw this, determine the pattern, and determine $\bigcap_n B_n$ in the numberline (it's ok to not be rigorous).	
Define the "Banach-Mazur game" $\texttt{BM}(A)$ for set $A \subseteq \mathbb{R}^2$: Alice and Bob play open rational balls; on the board/table one "old" ball and the rule is (whoever's turn it is) to play another "new" ball, comfortably contained inside the "old". Then if $\bigcap B_n \subseteq \mathbb{R}^2$ intersection of all balls played (which is non-empty, a fact you may use without proof), contains a point of A, then Alice wins. (1) Tell Bob how to win by the end of Round1 if $A = \mathbb{Z}^2$. (2) State and prove a conjecture about NWD sets and BM game.	
Suppose A is an open ball. Who would you rather play as?	
Suppose A is a single point. Play as Bob. How can you win?	
Suppose $A = \mathbb{Z}^2 \subseteq \mathbb{R}^2$. Play as Bob. How can you win?	
Suppose A is a circle (no interior). Play as Bob. How can you win?	
Suppose $A = \mathbb{Q}^2 \subseteq \mathbb{R}^2$. Play as Bob. How can you win?	
"Nowhere dense" set $S \subseteq \mathbb{R}^2$ means that you have a winning strategy in this game: Opponent chooses ball (any ball they want); You choose a smaller ball, that completely avoids S. (1) Prove that $(0,0)$ is NWD. (2) Prove that $\mathbb{Z}^2 \subseteq \mathbb{R}^2$ is NWD. (3) prove that boundary of circle, and square is NWD.	
Must a countable union of NWD sets be NWD?	
Prove that if $A \subseteq$ countable union of NWD, then Bob has winning strategy in $\texttt{BM}(A)$ game.	
Consider now "dyadic rational ball BM", where players are forced to play dyadic rational balls. (1) Prove that if Alice (or Bob) has winning strategy in "rational ball BM game", then Alice (or Bob) has winstrat in "dyadic rational ball BMG". (2) converse.	

Prove that there are only countably many boardstate positions p total for the "rational ball BMG".	
Suppose Bob has a winstrat β for $\texttt{BM}(A)$. For a rational ball B, define $\text{OutNext}(B;\beta)$ to be all the $x \in B$ that are "guaranteed out" by the instant after Bob plays his next move, i.e. the set of $x \in B$ s.t. for ANY rational ball $B' \bar{\subseteq} B$ (that Alice might play in response to B), Bob's winstrat β will choose $B'' \bar{\subseteq} B' \bar{\subseteq} B$ avoiding x ("$x \in B$ s.t. no matter what Alice does next, Bob's next response avoids x"). Using these sets, prove that [Bob has winstrat β] implies A covered by countably many NWD.	https://math.stackexchange.com/questions/800744/banach-mazur-game-proof-about-winning-strategies https://chat.deepseek.com/a/chat/s/f5ead31d-2ea3-4e62-bdc0-50015eff0153 Very important to think about "Out", that is the key insight that naive/first attempts at proof miss.
Suppose Alice has a winstrat α for $\texttt{BM}(A)$. Let U_1 denote the first move Alice plays (according to α). Using the "out sets" above, prove that [Alice has winstrat α] $\implies U_1 \setminus A$ is covered by countably many NWD. ("A is comeager in U_1". (2) Prove the converse.	
Call a set that can be covered by ctbly many NWD as "meager". Prove that ctbl union of meager is meager.	
Call a set A s.t. one of the 2 players has a winning strategy "BM". Call a set A that is an open symmetric-difference a meager set "Baire measurable. Prove that BM $\iff$ Baire measurable. (2) then prove the "Baire alternative": any BM set is either meager, or comeager in some open ball.	
Prove that BM sets are closed under complement and countable union. (And include all open balls)	
Consider any 2-coloring of irrational rotation circle graph. Prove that set of red pts can not be BM.	
Bonus: Consider G0 graph. Prove that any BM independent set must be meager. Conclude that χ_{BM} of G0 graph is uncountable.	

EDIT 9/14/26: Banach Mazur but in Cantor space. Get used to my star notation. Learn about clopen sets. Maybe this should come after the "Sturmian" worksheet? Possible order (~~"Exploration of DST"~~ "Explorations of the Infinite" unit): BM game in $\mathbb{R}^2$; formalizing $\mathbb{R}$; Sturmian; BM in Cantor space; König and Compactness; Ellentuck.

8/28/26

Question	Answer
Define a "real number" $\in [0,1]$ to be a decimal point followed by countable sequence of digits 0..9. Think about an entity/oracle/machine that spits out one new digit every time you nudge it (so it "contains"/"knows" the whole real number, but you only get access to any initial finite segment at any given time). Define a notion of $\leqslant$ on real numbers. (It is very important that this question is about $\leqslant$, not $=$ or $<$, since for the latter ones, the most naive definition is actually really bad!)	
Give some examples of real numbers r and algorithms that output r as a stream (in the above sense).	
Give an algorithm that takes in a real r, a finite decimal d, and returns $r+d$. I.e. given an oracle that outputs digits of r, and the finite string d, what should you do to act as the oracle for the real $r+d$?	
Assume that our notion of real numbers has a "notion of distance": every 2 real numbers x, y have a distance $\mathsf{d}(x,y)$ (symmetric in its inputs) that is also a real; and if $x, y \in \mathbb{Q}$ then $d(x,y) = \lvert y-x \rvert \in \mathbb{Q}$. Assume also $a \leqslant b \leqslant c \leqslant d \implies \mathsf{d}(b,c) \leqslant \mathsf{d}(a,d)$. Prove that $0.999\cdots$ (repeating) and $1.00\cdots$ (repeating) have distance 0.	
Define two reals to be equal iff their distance (as imagined in the previous question) is 0. Define $x < y$ to mean $x \leqslant y$ and $x \neq y$. Let $\ell \in \mathbb{Z}[\frac{1}{10}]$ be a finite decimal s.t. $2\ell \leqslant \mathsf{d}(x,y)$. Prove that if an interval $[r, r+\ell]$ contains x, then it doesn't contain y.	Interval can have rational or real endpoints, since I have problem above where we define addition of real+finitedecimal.
Prove that any countable sequence of nested intervals, whose lengths go to 0 (you know intuitively what this means), has intersection that contains at most 1 real. (Hint: suppose intersection contained x, and let $y \neq x$, and show y is NOT in the intersection.)	Use previous problem
Prove that every set of real numbers A has a least upper bound LUB (and greatest lower bound GLB)	Take max on every coordinate
Define a set U of real numbers to be open if given a real r, I can tell with just finite amount of data (i.e. receive finitely many bits from the oracle/machine outputting r) that $r \in U$. Prove that $(0.5, 0.6)$ is open.	
Define a set C of real numbers if its complement is open, i.e. given a real r, I can be sure by a finite time that $r \notin C$. Prove that $[0.5, 0.6]$ is closed.	
Is there a set of reals which is both open and closed? What about neither?	
Recall previous Exc "Give an example of a countable decreasing sequence of open balls, so that the intersection of all balls is empty (i.e. there is no point $(x,y) \in \mathbb{R}^2$ that is in every single ball of the countable sequence).". Can also exhibit this type of phenomenon using a nested sequence of sets of naturals.	Added 9/14/26. $[1..\infty) \supseteq [2..\infty) \supseteq \ldots$.
Prove (by contradiction) that a nested sequence of closed intervals I_n has non-empty intersection. (Hint: take sup of left endpoints.)	Proof: consider sup l_∞ of set of left endpoints $l_1 \leqslant l_2 \leqslant \ldots$ of the given intervals. WTS l_∞ in each I_n. Suppose it's not in some I_n (fixed). Then I must have been able to determine that by finite time. But any finite portion of l_∞ is completely decided and then left unchanged after some finite time (i.e. all l_m for m suff large have same "criminal" initial segment as l_∞). So then some l_m is definitely not in I_n, contradiction.
Prove that any sequence of reals has a monotone subsequence.	Hint: picture real r_i as height (y-axis) of building at position i (x-axis) and imagine there is a "beautiful view" at $x = \infty$. Consider the question: do infinitely many buildings have a "beautiful view"? I learned this "Spanish Hotel" proof from Clark Lyons (I guess Spanish Hotels are known for having beautiful views?)
Prove that a nested sequence of closed balls in $\mathbb{R}^2$ has non-empty intersection.	My solution: consider centers, project onto x and y axes, use monotone subsequence argument and redo the nested closed interval argument.
Using these nested closed interval/ball theorems — which tell us that we can "algorithmically" specify a real number in a different way as before (digit by digit) namely as a sequence of nested closed intervals with diam going to 0 — define a notion of addition, subtraction, multiplication, division for real numbers, that extend the same operations on $\mathbb{Q}$, and make $\mathbb{R}$ into a field.	

ToyExc (from Bar-Nata Khovanov homology): place $\pm$ on all edges of oriented hypercube graph (orient downward, more number of 0's layer points to fewer number of 0's layer) so that each (oriented) square has odd number of $-$ signs. Equivalently, 2 paths around each square, multiplying their edge signs, have unequal total signs.

(Solution: label vertices by wedges of dx's (written in unique correct order), e.g. highest vtx in square label say is dy, then points to $dx \wedge dy$ and $dy \wedge dz$. The signs are what is necessary to go from $dy \wedge dx$, $dy \wedge dz$ to the labeled diff form (so in this running ex, $-$ and $+$ resp.). Lowest vtx of square is then $dx \wedge dy \wedge dz$, and edge is whatever sign is needed to get e.g. $(dy \wedge dz) \wedge dx$ (always place the newest wedge on the right) to the label.

Verify the 2 paths unequal sign condition: the total signs of the 2 paths are the signs needed to correct the order in the 2 forms $dy \wedge dx \wedge dz$ and $dy \wedge dz \wedge dx$; i.e. the newest 2 wedges — on the rightmost as I said — appear in both orders, which differ by a $-$ by typical exterior der anticomm)

At last, the orbiting.

Start with the "conceptually simplest" model. Have connected countable graph (so countably many vertices, with edges between some pairs of vertices), with each edge really being thought of as 2 arrows (forward one and backward one), each arrow labeled by a natural number, so that it's always unambiguous for a vertex v to say "my ith neighbor" (i.e. there's $\leqslant 1$ vertex w so that the arrow $v \to w$ is labelled i — for any i and v).

On each node, we have any entity (Cantorite) that can do any countable process, like remember/hold in mind a whole bitstream (because that's just remembering countably many bits), or check equality between 2 bitstreams (that's just checking equality between 2 bits, countably many times).

In particular, each Cantorite can hold the entire countable graph in mind, so each Cantorite knows what numbered arrows to follow to get to every other vertex, knows e.g. "the paths $3, 2, 49$ and $2, 57, 1$ lead to the same vertex" or "the path $1, 3, 7, 2, 3, 6$ doesn't exist/leads no where".

Each node will be arbitrarily assigned a unique ID bitstream.

Our goal is to design algorithms, that each node can run individually, that works no matter what unique ID bitstreams were assigned, and s.t. every node in the graph computes the same output.

So for example, if a proposed algorithm involves the step "everyone go talk to the guy with ID 0000..", it is not a valid algorithm.

This sounds weird, so I'll do the first couple examples.

Question	Answer
proper $\mathbb{N}$-coloring of locally finite graph $\mathfrak{G}$ (compute a function)	locfinCOL-Algo: start with standard enumeration of finite bitstrings $(s_1, s_2, \ldots) := (0, 1, 00, 01, 10, 11, 000, \ldots)$. Define the color of node x to be the least n s.t. $\texttt{ID}(x)$ starts with bitstring s_n, but all neighbors y of x do NOT start with s_n. . Convince yourself that this is a countable computation, that works no matter what unique IDs were assigned, and that all nodes compute the same proper $\mathbb{N}$-coloring of $\mathfrak{G}$ (i.e. all nodes agree that the color of the node with $\texttt{ID} = \ldots$ is 31 or whatever.)
IIS: infinite independent set in locally finite $\mathfrak{G}$ (compute a set). (In fact the proof should work for any $\mathfrak{G}$ admitting a proper $\mathbb{N}$-coloring.)	IIS-Algo: run locfinCOL-Algo. Then choose the least n so that there are infinitely many nodes colored n, and output the (countable) set of all color-n nodes. . Convince yourself that this is a countable computation, that works no matter what unique IDs were assigned, and that all nodes compute the same IIS (i.e. all nodes have the same output (=countable set of nodes) in mind.)
MIS: maximal independent set (compute a set)	MIS-Algo: your turn!
proper $\Delta + 1$ coloring of deg $\leqslant \Delta$ graph (compute a function)	degbddCOL-Algo: your turn!
edge graph $\mathcal{G}$ (compute a graph). Think of $\mathcal{G}$ nodes as having their own Cantorites, and the $\mathfrak{G}$ Cantorites are all mass hallucinating the same "imaginary" Cantorite society $\mathcal{G}$.) An existing Cantorite society can mass hallucinate another Cantorite society into existence.	edgeGraph-Algo: for each pair of nodes that are endpoints of an edge in $\mathfrak{G}$, compare IDs to determine who is first; I'll notate such a pair as $\{x < y\}$. Then define a new node $\xi \in V(\mathcal{G}) := E(\mathfrak{G})$ with ID = interlace of $\texttt{ID}(x), \texttt{ID}(y)$. It is an easy countable computation to recover the original pair $\{x < y\}$ from this ID — simply de-interlace. Define an edge between $\xi, \eta \in V(\mathcal{G})$ iff the original pairs corresponding to ξ, η intersect. . Convince yourself that this is a countable computation, that works no matter what unique IDs were assigned, and that all nodes compute the same graph (i.e. all nodes have the same vertex set and IDs (=countable set of nodes with IDs) in mind, and same edge set in mind.)
Give an example of a countable graph, whose edge graph is not locally finite.	
canonical path between 2 vertices (compute a function $p : V(\mathcal{G}) \times V(\mathcal{G}) \smallsetminus \text{diag} \to \mathbb{N}^*$ taking in 2 distinct vertices x, y, returning a list of nats s.t. if you follow the arrows with those numbers, you'll go from x to y). For this problem, let's say that the arrow labels are positive nats $\mathbb{N}^+$.	Maybe I'll leave this as exercise to students, maybe not. pathXY-Algo: start with standard enumeration of natlists $(l_1, l_2, \ldots) := (1, 11, 2, 111, 12, 21, 3, 1111, 112, 121, 211, 13, 31, 4, \ldots)$, and output l_n where n is minimal s.t. l_n is a valid path from x to y. . Convince yourself that this is a countable computation, and that all nodes compute the same paths (i.e. all nodes agree on one path from x to y.)

proper $\mathbb{N}$-coloring of edge graph (compute a function). First, I'll give a failed algorithm that computes the function $c_0 : V(\mathcal{G}) \to \mathbb{N}^*$, that is pretty good but not perfect. Failed pairGraphCOL-Algo: for vertex $V(\mathcal{G}) \ni \xi \coloneqq \{x < y\}$, define $c_0(\xi) \coloneqq$ path between x, y computed by pathXY-Algo. . Why does c_0 fail? Fix this by using the failure spots of c_0 to define a subgraph $\mathcal{G}' \subseteq \mathcal{G}$, with same vertex set $V(\mathcal{G}') = V(\mathcal{G})$, but with max deg $\leqslant 2$; and then use locfinCOL-Algo to compute a function $c_1 : V(\mathcal{G}') \to \mathbb{N}$. Then $(c_0, c_1) : V(\mathcal{G}) \to \mathbb{N}^*$ should work.	Fail case of c_0 is $x \xrightarrow{k} y \xrightarrow{k} x$. Define subgraph $\{x < y\}, \{u < z\}$ to be adj in $\mathcal{G}'$ if $y = u$ and have same c_0. Correct pairGraphCOL-Algo:
Maximum matching in $\mathfrak{G}$	MIS in edge graph

We now focus on graphs where every vertex is adjacent to every other vertex (adjacency is an equivalence relation), notated as $\mathfrak{E}$ instead of $\mathfrak{G}$ to emphasize this difference.

Question	Answer
VSM vanishing sequence of markers (compute sequence of sets) Very important to draw VSM picture: falling sequence of horizontal lines in a rectangle, like receding tide.	VSM-Algo: Start with initseg 0. If i.m. ppl have that initseg, those ppl form $\breve{A}_1$. Else, increment lsb to 1, and those i.m. ppl will form $\breve{A}_1$. Then append lsb 0 (to running initseg). If i.m. ppl have that initseg, those ppl form $\breve{A}_2$. Else, increment lsb to 1, and those i.m. ppl will form $\breve{A}_2$. Etc. Finally, if $\bigcap_i \breve{A}_i$ is non-empty, it must be singleton. Dictator case. Can just delete him from all $\breve{A}_i$ to form A_i. This is an algorithm each person in the class can run silently in their head.
pairpair construction, or dictator	Take $\mathfrak{E}$, then get maximum matching $M_1 \subseteq E(\mathfrak{E}) = \binom{V(\mathfrak{E})}{2}$. If M_1 leads to incel, that's dictator. Else, take complete graph with vertices$=M_1$, find a maximum matching $M_2 \subseteq \binom{M_1}{2}$ on that. Etc.

Hyperfinite structure is if each node x can compute its "level n cohort" $[x]_n$, where

- (symmetry) if $y \in [x]_n$, then y will compute its level n cohort $[y]_n$ to be $= [x]_n \subseteq X$
- (finiteness) for all x and n, the cohort $[x]_n \subseteq X$ is finite,
- (nested) each level $n+1$ cohort is a union of level n cohort: $[x]_{n+1} = [x]_n \cup \bigcup_{i \in I} [y_i]_n \supseteq [x]_n$ (nested)
- (filling) and for every pair x, y, eventually they are in the same cohort: there is $N = N_{x,y}$ s.t. $n \geqslant N \implies [x]_n = [y]_n$.

Question	Answer
hfin implies Line-ing	22.1 TiOE, see MyIllustraions In level-1-cohorts, order nodes by ID lex order, creating tiny snakes. Then in level 2 cohort, order tiny snakes by lexID, then connect them head to tail, creating longer snakes. Continue.
Line-ing (each node has 2 neighbors) implies hfin	22.1 TiOE pg. 77, see MyIllustraions. 2 tools: FM (NedgeCOL), and VSM. Compute VSM $S_n \searrow \varnothing$. Claim: $\forall n$, the Line must dip in and out of S_n ("water") infinitely in both directions; or there is dictator. The level-n cohort $[x]_n$ of node x is: just itself if it is under water; or if it's dry, then $[x]_n \coloneqq$ all dry nodes connected to x by dry edges.

Suppose I have a class $X := \mathfrak{E}^+$, that I can split into 5 subclasses (so each x can compute its subclass $[x]_{\mathfrak{E}}$; and there are 5 subclasses total, i.e. $\{[x]_{\mathfrak{E}} : x \in X\}$ has cardinality 5) that are each hyperfinite. Prove that $\mathfrak{E}^+$ is also hyperfinite. Be careful, although WE (outside observers) may enumerate the subclasses 1..5, it may be that there is no way for every Cantorite to agree on the enumeration, so we can NOT say things like "the first subclass" in the algorithm.	KJL Prop. 1.3(vii) By searching through all outgoing arrows (in $\mathbb{N}$ order) greedily, each node x can compute $f_i(x)$ ($i = 1..5$) in all 5 different classes (so $f_1(x) = x$, then search until find the first $y =: f_2(x)$ outside of x's subclass $[x]_{\mathfrak{E}}$, etc. I write $[x]_n$ to mean x's level-n $\mathfrak{E}$-cohort, in its subclass $\subseteq [x]_{\mathfrak{E}}$. We now want to compute $[x]_n^+$ (to form a hyperfinite structure on $\mathfrak{E}^+$). For every other node y, we can compute its 5 representatives $f_1(y), ..., f_5(y)$. Then I compute my 5 representatives, and test if $f_1(x)$ shares a level-n $\mathfrak{E}$-cohort with (exactly) one rep of y; and so on $f_2(x), ..., f_5(x)$. I.e. y is level-n friends with me x, if each of their 5 reps is level-n friends to at least 1 of my 5 reps.

Multi-media. **G**ithub **I**ssues will have companion post with "countable combinatorics of Borel flavor" — see flags "See **GI**.x" — and iPad **N**otability will have pictures — see flags "See **N**Fig.x".

1 Preliminaries

What symbols will be used for in the sequel:

- X std. Borel space. Can always be equipped with a compatible Polish topology (just pretend it's Cantor or Baire space — can talk about countable basis, balls, total order, etc.).
- E CBER on X (type check: $E \subseteq X \times X$ Borel)
- $\mathcal{G}$ graphing, always Borel, usually of some CBER E (type check: $\mathcal{G} \subseteq E \subseteq X \times X$, always locc = locally countable)
-
-

Text abbreviations

- locc = locally countable, locfin = locally finite
- nbhd, nbd = neighborhood
- bdd = bounded
- c.m. = countably many, f.m. = finitely many
- ctbl, cbl = countable

The great blackbox theorems:

Fact 1.1: Luzin-Novikov (L-N)

Theorem 1.2: L-N Cor.: (locc Borel) Graph neighborhoods are Borel

Implies that locc G "connectedness relation" is Borel, degree $\deg_G$ is Borel function, etc.

Corollary 1.3: Borel chromatic number χ_B for locfin and bdd-deg Borel G /easy/

locfin proof is trivial, using "Borel nbhds" theorem and countable basis for topology.
[Borel MIS exists assuming [locc + ctbl χ_B] G] is trivial, using no tools.
Using MIS (of whatever regularity, Borel whatever), bdd-deg proof is trivial.

Theorem 1.4: L-N Cor.: CBER E generated by (c.m.) Borel functions

Borel $f_1, \dots, : X \to X$ s.t. $xEy \iff \exists n : x \overset{f_n}{\longmapsto} y$.

Theorem 1.5: gen by Borel fns Cor.: (generalized) Feldman-Moore /slightly tricky/

generalized Feldman-Moore is:

Corollary 1.6: gen F-M Cor.: Pairpair construction

Corollary 1.7: every ACBER has VSM

Build $V_1, \dots$ s.t. $\bigcap_i V_i = \varnothing$.
The "intended" solution here is via pairpair construction.

The easy "hack" way of proving VSM is to use dyadic decomposition (i.e. Borel iso to Cantor space), 3 sentence proof Lemma 6.7 in TiOE. This actually fits super well in my math circle "countable combinatorics" framework. *Proof:* Start with initseg 0. If i.m. ppl have that initseg, those ppl form $\breve{A}_1$. Else, increment lsb to 1, and those i.m. ppl will form $\breve{A}_1$.
Then append lsb 0 (to running initseg). If i.m. ppl have that initseg, those ppl form $\breve{A}_2$. Else, increment lsb to 1, and those i.m. ppl will form $\breve{A}_2$.
Etc.
Finally, if $\bigcap_i \breve{A}_i$ is non-empty, it must be singleton. Dictator case. Can just delete him from all $\breve{A}_i$ to form A_i.
This is an algorithm each person in the class can run silently in their head.

2 Monday

Always work with graphings $\mathcal{G}$ of E. By Feldman-Moore, may assume that it has a proper ω-coloring. We will always think about Feldman-Moore colored graphings instead of "L-graphings" that are discussed in the book. Boycott L-grapings! Colors for the win!

Proposition 2.1: 19.1 Trees are Best

Proof Sketch: assume f.s.o.c. exist word $w = c_1 c_2 \cdots c_n$ s.t.

$$A \coloneqq \{x \in X : x \overset{c_1}{\bullet\!\!-\!\!\bullet} \overset{c_2}{\bullet\!\!-\!\!\bullet} \cdots \overset{c_n}{\bullet\!\!-\!\!\bullet} x \text{ is proper cycle}\} \text{ has positive measure.}$$

Issue1: could be many c_1 edges in each cycle. (E.g. if only 3 colors total). Fix1: look at minimal c_1 edge.

Want to delete (minimal) edge c_1 in each cycle.

Issue2: in 1 cycle, there could be many c_1 edges, that are themselves part of other cycles. Then those cycles want to delete their minimal c_1 edge, Oh No! the original cycle has now been completely ripped apart.

Fix2: disjoint families!

(Recall we fixed w above)

Define $\mathscr{C} \coloneqq \{\text{all such cycles corresponding to word } w\} \subseteq \binom{X}{n}$. Put intersection relation $\sqcap$ on this vertex set to create a bounded degree Borel graph.

Fix finite Borel coloring of $(\mathscr{C}, \sqcap)$.

Choose the color class $\mathscr{C}'$ of maximal measure (definitely positive) — this is a disjoint family of cycles!

Then delete the minimal c_1 edge from each cycle in $\mathscr{C}'$. Win! ■

Detailed Proof:

2.1

Proposition 2.2: Line-ing implies Hyperfinite

Any treeing $\mathcal{T}$ of CBER E s.t. $\deg_{\mathcal{T}} = 2$ a.e. implies E hfin.

Proposition 2.3: Hyperfinite implies (directed!) Line-ing

CBER E hfin $\implies$ E directed line-able

Proof Sketch: line each (finite!) E_0 class. Then connect heads to tails appropriately, in next layer of the onion, using arbitrary Borel total order $\prec$ on X.

2.2

Pairpair construction (recall $\chi_B \leqslant \aleph_0 \implies$ can get maximal independent set) implies

- aperiodic hyperfinite subequivalence relation
- VSM

2.3

20.1 Orient edges toward transversal. Biject directed edges with its tail.

2.4 Indegree and Outdegree Lemma

Lemma 2.4

Let $R \subseteq X \times X \times \omega$ be a Borel colored directed multigraph.
Then

$$\int \text{indeg}_R(x) d\mu(x) = \int \text{outdeg}_R(x) d\mu(x).$$

Proof Sketch: can make R actually properly ω-colored (I think by Feldman-Moore).

Then for each color class, $R|_{c_1}$ ("R restricted to that color class") is actually a function, in fact a Borel E-invariant partial automorphism $X \dashrightarrow X$.

Then

$$\text{outdeg}_{R|_{c_1}}(x) = 1_{\text{dom}(R|_{c_1})}.$$

Equality of integrals follows from definition of E-invariant partial automorphism. ■

Using this Lemma, we can show that any Borel partial automorphism of edges (e.g. in sliding) preserves cost.

And also pruning in Tuesday: $C_\mu(G) = C_\mu(G') + \mu(\text{leaf edges of } G)$, where $G' = G \setminus \{\text{leaf edges}\}$.

Question: hyperfinite implies every graphing has spanning tree?

3 Tuesday

Sliding. Start with $\mathcal{G}$ graphing of E. Complete section S. Each vertex identifies **geodesic** to S. Edges in these geodesics get colored brown, oriented toward S. Brown edges form a treeing of a subeqrel s.t. S is a Borel transversal. Cost is $\mu(X \setminus S)$ by Indeg/Outdeg lemma above.

Slide $\mathcal{G}$ down to a graping $\mathcal{G}|_S$ of $E|_S$. This shows

$$C_\mu(\mathcal{G}) \leqslant C_{\mu|_S}(\mathcal{G}|_S) + \mu(X \setminus S)$$

OTOH, start with $\mathcal{G}|_S$. Make complete bipartite graphing $K[S, X \setminus S]$. Same **geodesic** idea above on this produces (height 1) tree above each $s \in S$... like grass. (Hahaha!) Then $\mathcal{G}|_S \sqcup \{\text{the grass}\}$ is graphing $\mathcal{G}$ of E where

$$C_\mu(\mathcal{G}) = C_{\mu|_S}(\mathcal{G}|_S) + \mu(X \setminus S)$$

3.1 Pruning?

Proposition 3.1: At least 1

E an aperiodic CBER and μ-E invariant measure, then $C_\mu(E) \geqslant \mu(X)$

Proof. If $\mu(X) = \infty$, this is clear, so we may assume that μ is a probability measure.

Let G be a graphing of E and note that we may assume that by throwing away a null set, G is a locally finite graph

Define $G' = G \setminus \{\text{leaf-edges}\}$. Recursively define $G^{\alpha+1} = G^{\alpha\prime}$. For a locally finite graph, G^ω is pruned.

Lemma 3.2: prune

$C_\mu(G') + \mu(\{\text{leaves}\}) = C_\mu(G)$

Proof of Lemma. Orient leaf edges towards leaf. Then apply Indegree and Outdegree Lemma □

Then use the fact that $C_\mu(G^\omega) \geqslant \mu(V(G^\omega)$ and add μ of leaves to get $\mu(X)$. □

Proposition 3.3: hyperfinite realize cost

Let E be aperiodic hyperfinite, then $E = \bigcup_n E_n$. Let $\mathcal{T}$ be a treeing of E, then $C_\mu(\mathcal{T}) = C_\mu(E) = \mu(X)$

Proof. We may assume that E_n is $\mathcal{T} \cap E_n$. Since each E_n is finite we may pick a transversal T_n. Then we have

$$C_\mu(\mathcal{T}) = \lim_{n\to\infty} C_\mu(\mathcal{T} \cap E_n) = \lim_{n\to\infty} \mu(X) - \mu(T_n) \leqslant \mu(X)$$

For the other direction use the previous theorem. □

4 Wednesday

4.1 Tail Eqrel is Hsmooth

Simple Case

$Z = \binom{\omega}{\omega}$ with delete min operation $T : Z \to Z$.

WTS (left)shift graph on $\binom{\omega}{\omega}$ (which is locfin) is hyperfinite. Conp is directed tree (1-ended!).

Define $f_n(x) :=$ keep walking along directed edges until hit a vertex of rank $\geqslant n$. (Rank measured from leaves)

Then define $xE_ny :\iff f_n(x) = f_n(y)$. Tail eqrel $E_t = \bigcup E_n$.

In general 1-ended tree, choose VSM $S_n \searrow \varnothing$. Define $f_n(x) :=$ the first $y \in S_n$ we hit after walking from x (along directed edges).

General Case

Wednesday: showing that $Z := X^\omega$ with shift graph $T : Z \to Z$ given by shift the sequence left by one is hypersmooth.

Each conp is 1-ended tree.

If we have VSM S_n with cofinality, we can define $xE_ny :\iff f_n(x) = f_n(y)$ where $f_n(x) :=$ first $y \in S_n$ that walking from x along shift graph will hit.

By 223D material, get countable coloring of (Z, T). If z is s.t. in its conp, $\forall n$, $C_{\geqslant n}$ is cofinal, then that's our VSM with cofinality.

Else, exists N s.t. $C_{\geqslant N}$ is not cofinal. Then can split conp into Before, which can use any color, and After (which everything in Before will eventually hit) that only uses colors $< N$. There must be color that is cofinal, say C_1.

Then restrict to C_1, merge edges until get another shift-type graph on C_1.

Iterate procedure: countable color, if $C_{1,\geqslant n}$ cofinal, VSM with cofinality, yay. Else, get $C_{11} \subsetneq C_1$, and $C_{11} \subseteq Z$ is now $\geqslant 2$-separated (in the graph (Z, T)), because of proper countable coloring.

So, either at some stage exit out of the iteration because get into the "VSM with cofinality" case, or finish ω stages of the iteration, at which point we have $C_1 \supsetneq C_{11} \supsetneq \dots$ which is itself a VSM with cofinality.

4.2 Limsup proof

[Daniel edit 6/17/26 after talking to Anton]

Think of X as embedded into $[0, 1]$ say. Define $L(x) := \limsup_{n\to\infty} T^n(x)$.

Observe that L is T-invariant. We can thus think of $L(x)$ as actually $L([x]_E)$.

We can define cofinal VSM by

$$S_n := \{x \in X : x > L(x) - \frac{1}{n}\}$$

It is obviously cofinal. What is $\bigcap_n S_n$? May not be zero, but $x \in \bigcap_n S_n \iff x = L(x) = L([x]_E)$. But because $X \hookrightarrow [0, 1]$, there can be $\leqslant 1$ such x per component. So either $\bigcap_n S_n$ picks out dictator, or is vanishing. Either way, we are happy.

4.3 22

Bamboo-ing implies Hfin

VSM. Take E_n to be things in between markers in S_n, and the guys in S_n itself as singletons. (If in between markers, there's infinite points, then that's a smooth situation.) Wow, we have E_n increasing to E.

Pruning

Prune. Easy case is if there is bamboo-ing left.

Otherwise, if pruned away the whole tree, then we're in "Simple Case" of above "Tail Eqrel is Hsmooth" section.

4.4 23

Extension Lemma

Finite μ!

Draw big landscape-oriented rectangle — that's X. Cut into thin-ish vertical rectangles — those are E-classes. Cut those into very thin rectangles — those are E_0-classes.

Get near-opt $\mathcal{G}_0$ of E_0 ("vertical"). Have complete section S of E_0, very thin "horizontal" line at bottom of Big Rectangle X.

In §21, we had $C(E|_S) \approx C(E) - 1$ (where $\approx$ means up to $\epsilon = \mu(S)$). Choose near-opt $\mathcal{G}_S$ of $E|_S$.

Then $\mathcal{G}_0 \sqcup \mathcal{G}_S$ is graphing of E of cost

$$\approx \underbrace{C(E_0)}_{\text{"vertical"}} + \underbrace{C(E) - 1}_{\text{"horizontal"}}$$

Subadditivity of Normalized Cost: Aper

Finite μ! $E_1 \cap E_2$ aperiodic! $\boxed{\infty}$ (perhaps not commutative)

aper hfin $E_0 \subseteq E_1 \cap E_2$. Has bamboo-ing $\mathcal{G}_0$, which we know (previous) has cost 1.

Use Extension Lemma to get to near-opt $\mathcal{G}_1, \mathcal{G}_2 \supseteq \mathcal{G}_0$ of E_1, E_2 resp.

Then we get graphing of $E := E_1 \vee E_2$:

$$\underbrace{\mathcal{G}_0}_{1} \sqcup \underbrace{(\mathcal{G}_1 \setminus \mathcal{G}_0)}_{\approx \text{opt}-1} \sqcup \underbrace{(\mathcal{G}_2 \setminus \mathcal{G}_0)}_{\approx \text{opt}-1},$$

saved a "-1"!!!

Can extend this to infinite and extending $\mathcal{G}_0$ to $\mathcal{G}_k$ for each E_k using $\epsilon/2^k$.

Subadditivity of Normalized Cost: Commutative

Finite μ! $E_1 \square E_2$ commutative. (perhaps not aper. $\boxed{\text{fin}}$)

Draw big landscape rectangle for $E_1 \vee E_2$-class. Cols are E_1 classes, rows are E_2 classes.

(Fix aper hfin $F \subseteq E_2$, so that at end of proof, we can say $C(F|_S) = \mu(S)$ — which will be $< \epsilon$ by VSM.)

Get grid of boxes. Bold/double-border those boxes that contain finitely many points, those are "A-boxes"

Because smooth part is msr 0, all cols either have ∞ many A-boxes, or none.

Let (orange) $S_n \subseteq A$ be VSM (in ptclr complete section by def) of $E_1|_A$, i.e. ($\star$) if col contains an A-box, then it contains an A-box with orange dot.

Again because smooth is msr 0, may assume ∞ many (orange) S_n points in each F-row. Draw orange oval in each F-class (very thin row), has ∞ many orange dots.

Recall if box contains orange dot, it must be an A-box (because $S_n \subseteq A$).

Thus every orange oval must hit infinitely many E_1-cols.

Case1: person in (A-)box in E_2-row, and orange oval hitting that same box. Then person can vertical-jump into the orange oval, then can travel to any point in orange oval.

Case2 (generalizing Case1): person in A-box. By ($\star$) above, there is A-box in same col with an orange oval. Orange oval hits infinitely many cols, and vertical-jumping back up to person's E_2-row implies dots in ∞ many cols (in the same E_2-row).

Case3: person not in A-box. Then that box is infinite itself.

$H :=$ HopScotch graph generated by orange oval and vertical jumping. Above Cases are to show that $H \cap E_2$ is aCBER.

Can now apply previous section.

5 Thursday

24.7

Again draw E_2-classes as cols, E_1-classes as rows. Not necc commutative.

Graphings $\mathcal{G}_1, \mathcal{G}_2$ of E_1, E_2. Call them "horizontal" and "vertical" edges resp.

Color edges in $\mathcal{G}_1 \smallsetminus E_2$ (horizontal edges going between 2 different columns) orange.

Definition 5.1: $\mathcal{G}_1$ *has ladder property for* E_2

he assumption is that if 2 cols have 1 orange edge/"stitch" between them, then there are ∞ many orange edges between them.

VSM on such orange edges to get $\mathcal{G}_1'$ (in ptclr, keep at least 1 orange edge between any 2 cols that were previously stitched together).

Then for any $\mathcal{G}_2' \supseteq \mathcal{G}_2$, we get $\mathcal{G}_1' \sqcup \mathcal{G}_2'$ graph $E_1 \vee E_2'$.

Main thing to prove is: if 2 dots are E_1-classmates (i.e. in same row), they are conn by $\mathcal{G}_1$(horizontal)-path, which can always be thought of as path of orange ($\mathcal{G}_1 \smallsetminus E_2$) or short (meaning contained in 1 column) horizontal edges.

Any given orange edge may have been deleted, but will always have a "parallel" orange edge that still lives in the VSM (that one can get to by sliding up and down the vertical($\mathcal{G}_2$) edges in each of the 2 columns).

Lemma 5.2: Restate 24.7: $[\mathcal{G}_1$ *ladder propy for* $E_2^-] \implies [\vee \text{fincost} \equiv \vee\varnothing]$

Let E_1 finite cost and $\mathcal{G}_1$ graphing.

Let E_2.

Suppose $\exists E_2^- \subseteq E_2$ s.t. $\mathcal{G}_1$ ladder propy for E_2^- (implied by commutative and E_2 aper)
Then $C(E_2 \vee E_1) \leqslant C(E_2)$, i.e. "$\vee E_1$" negligible.
More quantitatively, given any graphing $\mathcal{G}_2$, can adjoin $\mathcal{G}_2 \sqcup \epsilon\mathcal{G}_1$ (ϵ msr subset) to get graphing of the join.

Corollary 5.3: Minimum

Let E_1, E_2 finite cost, with graphings that are commutative, aper.
Then mutual ladder propy holds, so previous Lemma applies to give:
Then $C(E_2 \vee E_1) \leqslant \min\{C(E_1), C(E_2)\}$.

Conj (probably false): commuting fincost aCBER have commuting fincost graphings. (If this Conj were true, then Problem 24.4 would be solved)

5.1 25

$[E^+ : E] < \infty$

25.3. Main idea: each edge e in $\mathcal{G}^+$ can compute $\leqslant n$ destination edges (n possible ways to slide, but some slidings will collapse the edge to a single point).

25.6

Need to deal with squishing (slide an edge into a single point)

$\mathcal{T}$ treeing of E^+

$\mathcal{T}'$ = sliding of $\mathcal{T}$, to be treeing of E

$\mathcal{T}_0 \coloneqq$ {never squished} all edges of $\mathcal{T}$ that never get squished

$\mathcal{T}_1 \coloneqq \mathcal{T} \setminus \mathcal{T}_0 = \{\geqslant 1 \text{ squished}\}$.

Claim1: $\mathcal{T}_1$ covered by $(n-1)$ partial Borel autos

(2 paragraphs in book)

1 paragraph (easy) to see that (cost of sliding of $\mathcal{T}_0$) $C(\mathcal{T}_0') = nC(\mathcal{T}_0)$.

Then easy algebra.

Observe that if we prove $C(E|_S) \geqslant nC(E^+|_S) - \text{anyFunc}(n)\mu(S)$, then taking $\mu(S) \to 0$ and using §21 material we get to

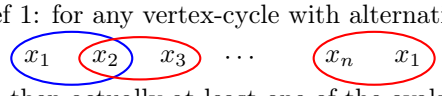
$$C(E) - 1 \geqslant n(C(E^+) - 1)$$

as desired.

6 Sunday

6.1 Key Idea 1: "folding" (homotopy)

Def 1: for any vertex-cycle with alternating-colored ovals



then actually at least one of the ovals must be purple.

DefNew: $E_1 \perp_\cap E_2$, means: any alternating **class-cycle** $\breve{C}_1 \bullet\!\!-\!\!\bullet C_2 \bullet\!\!-\!\!\bullet C_1' \bullet\!\!-\!\!\bullet \dots \bullet\!\!-\!\!\bullet \breve{C}_1$ — where $C_1 \bullet\!\!-\!\!\bullet C_2$ means that $C_1 \cap C_2$ — also has "backtrack" like $C_1 \bullet\!\!-\!\!\bullet C_2 \bullet\!\!-\!\!\bullet C_1$.

Call C_2 the "corner", and C_1 the "pre-corner".

Not hard to show Def1 $\iff$ DefNew.

($\implies$): start with class-cycle. Find $x_2 \in C_1 \cap C_2$, $x_3, \dots$ and $x_1 \in C_n \cap C_1$, which forms a vertex cycle with alternating ovals.

Def1 says (w.l.o.g. by reindexing) $x_2\ x_3$. This forces $C_1 = C_3$. Backtrack identified!

($\impliedby$): start with vertex cycle with alternating ovals. Those form class-cycle. Backtrack

$$C \ni x \in C' \ni x' \in C$$

implies $x_2\ x_3$.

In 27.5 (blue edges = Φ_1, red edges = Φ_2, green edges = Ψ_1, yellow edges = Ψ_2), we want to show that if we have the BlueGreen/RedYellow swap (say cool/warm swap for cool colors and warm colors) property, then get BlueGreen graphs E_1.

Given any path γ from $x_1 \rightsquigarrow x_n$ where $x_1, x_n \in C_1$, we claim that we can "fold"/"homotope" the path eventually to a (bryg)-path completely contained in C_1. Once we get to a bryg-path in C_1, appy cool/warm swap on all ry-edges to get purely bg-path in C_1.

Ok, how do we homotope? Start with path γ. Gives rise (choose arbitrary) to a class-cycle $C_1 \bullet\!\!-\!\!\bullet C_2 \bullet\!\!-\!\!\bullet \dots \bullet\!\!-\!\!\bullet C_1$.

By DefNew, this class-cycle must have backtrack.

Go to any backtrack. 2 cases:

1. if corner of backtrack is C_i for $i \neq 1$. Say corner C' is blue for sake of illustration. Suppose γ in corner class C' is bryg-path. Using cool/warm swap, convert it into a bg-path in C'.

   Apply cool/warm swap it into a ry-path in the pre-corner of corner C'.

   By induction (on length of class-cycle) we're done.

2. if corner of backtrack is C_1. Then pre-corner is C_2. There is subpath $\tilde{\gamma} \subseteq \gamma$ (endpts are in $C_1 \cap C_2$) starting and ending with class C_2. By induction (on length of class-cycle), there is ry-path homotopic to $\tilde{\gamma}$, contained $\subseteq C_2$.

   Apply cool/warm swap to turn that ry-path $\subseteq C_2$ (again whose endpts are in $C_1 \cap C_2$) into a bg-path in C_1. Done!

6.2 Producing the treeing $\Psi_1 \sqcup \Psi_2$ of $E_1 \cap E_2$

hfin case do it by induction. Finite case do it by counting.

6.3 Key Idea 2: disjointify

Notation:

Ω split graphing.

Λ near-opt graphing. Want best of both worlds: the splitness of Ω, but the optimality of Λ.

N_Ω s.t. **all-but-ϵ-cost** edges of Ω are colored by omega-colors $i \leqslant N^\Omega$.

$N_{\vec{\Lambda}}$ (called N_0 in book and Ely below) defined to be s.t. **all-but-ϵ-msr** x are s.t. for all omega-colors $i \leqslant N_\Omega$, exist $j \leqslant N_{\vec{\Lambda}}$ s.t. $\Omega^i(x) = \ell_j(x)$

(ℓ_j is enumeration of finite list of lambda-colors, i.e. finite list of finite lists of omega-colors. Pictorally, a lambda-edge is a mini-trapezoid (of omega-edges), and ℓ_j is a MegaTrapezoid (of omega-colors). Which we can interpret as "GPS directions (in terms of omega-directions)").

I.e. we have near-full-msr set $X' \subseteq X$ s.t. $\Omega^{\leqslant N_\Omega}$-subgraph completely remembered by $\ell_{\leqslant N_{\vec{\Lambda}}}$-directions.

At this point forget about Ω

Disjointify is disjointifying mini-trapezoids.

2 epsilon-chunks: "High" omega-edges of rare (too high) omega-colors, and "Bad Low" (low omega-colored) omega-edges that are bases of failed $\ell_{\leqslant N_{\vec{\Lambda}}}$-MegaTrapezoids.

Λ_0 connects vertices who were Good low-omega-color connected. So $\Theta \coloneqq [\Lambda_0 \sqcup$ 2 epsilon-chunks] is graphing of E (because it's: Good Low, High, Bad Low), near-optimal in fact.

Below is older stuff, joint with Ely.

[EDIT 5/2/26: need to define "tree structure". By thinking about $E_1 \perp_\cap E_2$, realize tree in graph theory means "homotopically trivial", $\pi_1 = 0$. If T is a tree in graph theory (no self loops), then any path $\Upsilon := (x \bullet\!\!-\!\!\bullet \ldots \bullet\!\!-\!\!\bullet x)$ must have a "backtrack", some $\ldots \bullet\!\!-\!\!\bullet y \bullet\!\!-\!\!\bullet z \bullet\!\!-\!\!\bullet y \bullet\!\!-\!\!\bullet \ldots$ appearing in path Υ. We can always truncate to just $\ldots \bullet\!\!-\!\!\bullet y \bullet\!\!-\!\!\bullet \ldots$, that's the homotopy.

In the case of $E_1 \perp_\cap E_2$, the situation is: any **class-path** $\check{C}_1 \bullet\!\!-\!\!\bullet C_2 \bullet\!\!-\!\!\bullet C_1' \bullet\!\!-\!\!\bullet \ldots \bullet\!\!-\!\!\bullet \check{C}_1$ — where $C_1 \bullet\!\!-\!\!\bullet C_2$ means that $C_1 \cap C_2$ — also has "backtrack" like $C_1 \bullet\!\!-\!\!\bullet C_2 \bullet\!\!-\!\!\bullet C_1$.]

Lemma 6.1: 27.5

X std Borel.
E_1, E_2 CBERs on X.
$E := E_1 \vee E_2$. μ finite E-invt measure
$F = E_1 \cap E_2$, independent (so red and blue hyperedges, form tree)
F hfin.

Assume $\Phi = \Phi_1 \sqcup \Phi_2$ graphing E, where Φ_1 edges only connect E_1-equivalent dots, etc.

Then, can find treeings Ψ_1, Ψ_2 that disjoint union to Ψ treeing subequivalence rel of F, and

$$\Phi_1 \sqcup \Psi_1 \text{ graph } E_1 \qquad \wedge \qquad \Phi_2 \sqcup \Psi_2 \text{ graph } E_2.$$

Proof: purple highlight = red$\wedge$blue highlight, means same class in F.
Φ_1 blue edges, Φ_2 red edges, Ψ_1 violet edges, Ψ_2 pink edges.
We will talk about "vioblue" paths (each edge is blue$\vee$violet), and "redpink" paths (each edge is red$\vee$pink), and "viopink" paths (each edge is violet$\vee$pink)

We first prove a sufficient condition, guaranteeing the conclusion of the Lemma.

Claim 1: if [in each purple class, vioblue-connected $\iff$ redpink-connected], then in fact [vioblue-paths graph E_1, and redpink-paths graph E_2], i.e.

$$F \cap \langle \Phi_1, \Psi_1 \rangle \overset{(\star)}{=} F \cap \langle \Phi_2, \Psi_2 \rangle \quad \Longrightarrow \quad \begin{cases} \langle \Phi_1, \Psi_1 \rangle = E_1 \\ \langle \Phi_2, \Psi_2 \rangle = E_2 \end{cases}$$

Proof of Claim: s.f.s.o.c. $\langle \Phi_1, \Psi_1 \rangle \subsetneq E_1$.

Draw one E_1-class C_1^0, with 2 dots x, y **not** connectable by vioblue-path. Because $\Phi := \Phi_1 \sqcup \Phi_2$ graphs E, must be connected by blue$\vee$red path. In particular, connected by a vioblue$\vee$redpink-path. Every such path can associate (possibly many) **class-path** (see above EDIT).

Search over all such $\{(x, y) \in E_1 : x, y \text{ not-vioblue-connectable}\}$ to find one such (x_0, y_0) s.t. it has a vioblue$\vee$redpink-path $\Upsilon := (x_0 \sim y_0)$ which has an associated **class-path** Υ_C that is as short as possible (beginning and ending with C_1^0).

We rule out $\boxed{\text{length } 1}$: if $\Upsilon := (x_0 \sim y_0)$ all in same E_1-class, then any redpink-substring $x' \sim y'$ inside of Υ means in particular that $(x', y') \in F$, so by $(\star)$, can vioblue-connect x', y'. Thus we can exchange all redpink-substrings of Υ with vioblue-substrings, to get x_0, y_0 are vioblue-connected, ※.

We rule out $\boxed{\text{length} > 1}$: find a "backtrack" in Υ_C, say $E_1 \bullet\!\!-\!\!\bullet \ldots C_1 \bullet\!\!-\!\!\bullet C_2 \bullet\!\!-\!\!\bullet C_1 \ldots \bullet\!\!-\!\!\bullet E_1$. Must be some $z \in C_2$ not in the class C_1. Must also be some $x', y' \in C_1$ so that in Υ, we have $x_0 \ldots x' \sim z \sim y' \ldots y_0$. Must be that in (the vioblue$\vee$redpink-path) $(x' \sim z)$, pass through a point in $x'' \in C_1 \cap C_2$. (Because in path Υ, every dot is either E_1-classmates or E_2-classmates with

its neighbors. Walking along ($x' \rightsquigarrow z$) must transition between C_1-nbr with x', and C_2-nbr with z. At transition point x'', it is C_1-nbr with its predecessor and C_2-nbr with its successor.)

Similarly, get y''. So both $x'', y'' \in C_1 \cap C_2$ meaning $(x'', y'') \in F$. In previous paragraph, could have chosen x'', y'' so that $(x'' \rightsquigarrow z \rightsquigarrow y'') \subseteq \Upsilon$ is all in class C_2. By same argument in len=1 case above, can transform to redpink-path $(x'' \rightsquigarrow y'')$ all in C_2. By ($\star$), can transform to purely vioblue-path $(x'' \overset{\text{new}}{\rightsquigarrow} y'')$ all in C_1. But then for path $x_0 \rightsquigarrow x'' \overset{\text{new}}{\rightsquigarrow} y'' \rightsquigarrow y_0$, can associate same class-path as before but with the "backtrack" (the one analyzed above) deleted! ※minimality.

We now construct Ψ_1, Ψ_2. Base case: ${\Psi_1}^0, {\Psi_2}^0 := \varnothing$.
By F hfin, $= \bigcup^{\uparrow} F^n$ FBERs, where $F^0 :=$(singleton equiv rel).

Inductive assertion: ${\Psi_1}^n, {\Psi_2}^n$ connect dots in same F^n-class (so $\langle {\Psi_1}^n, {\Psi_2}^n \rangle \subseteq F^n$).

Furthermore, $(x, y) \in F^n$ implies

1. x, y viopink$_n$-conn $({\Psi_1}^n \sqcup {\Psi_2}^n) \implies$ both $\begin{cases} \text{vioblue}_n\text{-conn } ({\Psi_1}^n \sqcup \Phi_1) \\ \text{redpink}_n\text{-conn } ({\Psi_2}^n \sqcup \Phi_2) \end{cases}$ (so same green^n-class)
2. $\begin{cases} \text{vioblue}_n\text{-connected } ({\Psi_1}^n \sqcup \Phi_1) \overset{n \text{ even}}{\implies} \text{redpink}_n\text{-connected } ({\Psi_2}^n \sqcup \Phi_2) \\ \text{redpink}_n\text{-connected } ({\Psi_2}^n \sqcup \Phi_2) \overset{n \text{ odd}}{\implies} \text{vioblue}_n\text{-connected } ({\Psi_1}^n \sqcup \Phi_1) \end{cases}$

And ${\Psi_1}^n \sqcup {\Psi_2}^n$ is treeing (will use 1. above)

Case $n+1$ is even.
F^{n+1}-class is finite, so draw purple class, finite. Inside it, draw green circle for classes s.t. any 2 classmates are vioblue$_n \wedge$redpink$_n$-connected; green classes must also be finite. So can (in Borel way) pick green dot/representative in each green class.

So, picture is (finite) purple F^{n+1}-class, inside are many green classes.

Some green classes (dots vioblue$\wedge$redpink connected) collect into black class, of just dots that are vioblue-connected.

DEFINE

- new edges in ${\Psi_2}^{n+1}$ to be pink edges going in cycle, hitting every green dot (do this in every black class). Delete 1 edge in this cycle to make it a tree.
- keep ${\Psi_1}^{n+1} := {\Psi_1}^n$.

Claim: Inductive Assertion continues to hold.

Check 2.$(n+1)$ holds. Well, vioblue$_{n+1} :=$ vioblue$_n$, and we made sure to redpink$_{n+1}$-connect all purple$\wedge$vioblue$_n$-classes.

Check 1.$(n+1)$ holds. Well, new pink$_{n+1}$ edges only connect dots that were already vioblue$_{n+1} :=$ vioblue$_n$-connected.

(At level $n+1$) We've check 1. and 2. so now we check treeing. Only way we could have added cycle was by adding pink$_{n+1}$ edge between x, y that were already viopink$_n$-connected. But by 1.(n) that implies x, y would have been in same green^n class; but new pink$_{n+1}$ edges only go **between** different green^n classes.

Theorem 6.2: 27.2

X std Borel.
E_1, E_2 CBERs on X.
$E \coloneqq E_1 \vee E_2$. μ finite E-invt measure, and $C_\mu(E_1), C_\mu(E_2) < \infty$
$F = E_1 \cap E_2$, independent (so red and blue hyperedges, form tree)
F hfin.

Then

$$C_\mu^F(E) = C_\mu^F(E_1) + C_\mu^F(E_2)$$

If we could find decomposable graphings of E which approach the cost of E, we would be done. This may not be possible directly, but an auxiliary method is auspicious.

1. Fix graphings Ω_1 and Ω_2 of E_1 and E_2 respectively, of finite cost.
2. Define $\Omega \coloneqq \Omega_1 \sqcup \Omega_2$
3. Fix an ω-edge coloring $c_\Omega : \Omega \to \omega$.
4. Additionally, fix $\epsilon > 0$
5. Let Λ be a near-optimal graphing of E, i.e. $C_\mu(\Lambda) \leqslant C_\mu(E) + \epsilon/3$
6. Define
$$c_\Lambda(\lambda) \coloneqq (\text{lex min list of } \Omega\text{-colors needed to walk edge } \lambda)$$
(w.r.t. pre-chosen order $\prec$ on X) which is a countable edge coloring of Λ
7. Choose N large enough so that $\Sigma_{i>N}\, C\mu(c_\Omega^{-1}(i)) \leqslant \frac{\epsilon}{3}$.
8. Enumerate lists of Λ-colors as $\ell_1, \ell_2, \dots$
9. Define $\ell_i(x) \in X$ to be the vertex we walk to by taking edges in Λ colored with the list of colors specified in ℓ_i.
10. For each $i \leqslant N$, define $\Omega_i(x) \in X$ to be the vertex $v \overset{\Omega_i}{\bullet\!\!-\!\!\bullet} x$ where Ω_i is referring to edge being colored $i \in \omega$.
$$W_n^i \coloneqq \{x \in X : \Omega_i(x) \neq \ell_j(x)\ j \in n\}$$
Notde that decreasing $W_n^i \supseteq W_{n+1}^i$ and $\bigcap_n W_n^i = \varnothing$. Thus $\mu(W_n^i) \to_n 0$.
11. Pick N_0 such that $\mu(W_{N_0}^i) \leqslant \frac{\epsilon}{3(N+1)}$.
12. Note that there are finitely many colors $i_1, \dots, i_n$ used to write the lists $\ell_0, \dots, \ell_{N_0}$. Let $\Lambda_0 \coloneqq \bigsqcup_{j \leqslant n} c_\Lambda^{-1}(i_j)$.
13. Let
$$\Theta \coloneqq \Lambda_0 \sqcup \{e \in \Omega \text{ such that } e = v \overset{\Omega_i}{\bullet\!\!-\!\!\bullet} x \text{ and } v \text{ in } W_{N_0}^i\ \}_{i \leqslant N} \sqcup \{c_\Omega^{-1}(i)\}_{i \geqslant N}$$

In less formal terms, we take red-paths if there exist such paths and blue paths if there do not exist red paths and finally all green paths. Θ is thus a graphing of E and it is also near-optimal, i.e.

$$C_\mu(\Theta) \leqslant C_\mu(E) + \frac{\epsilon}{3} + (N+1) \cdot \frac{\epsilon}{3(N+1)} + \frac{\epsilon}{3} = C_\mu(E) + \epsilon$$

Clearly Λ_0 is blocking us from having a decomposable graphing and is what is contributing to the vast majority of the cost. Nevertheless, we can "disjointify" to prove the inequality $C_\mu^F(\Theta) \geqslant C_\mu^F(E_1) + C_\mu^F(E_2)$

Returning to $\Lambda_0 = \bigsqcup_{j \leqslant n} c_\Lambda^{-1}(i_j)$. For each Λ color i of $i_1, \ldots, i_n$, recall that i corresponds with a finite list of Ω-colors needed to walk the λ-edge colored i. Thus we can consider the left and right endpoints of each Ω j-colored edge of this list $j_0, \ldots, j_{k(i)}$. Let V_L^j and V_R^j correspond to the left and right vertices of all j colored Ω edges. Then for each i, there is an associated list $\{V_L^{j_l}, V_R^{j_l}\}_{0 \leqslant l \leqslant k(i)}^i$ where $k(i)$ is the length of the list of Ω-colors. For simplicity, we can write this list as $\{V_r^i\}_{r \leqslant k(i)}$ where $V_0^i = V_L^{j_0}, V_1^i = V_R^{j_0}, \ldots$ and $k(i)$ is reset as *twice* as many as $k(i)$. Fix copies $\overline{V_r^{i_l}}$ for each i_l and $0 < r < k(i_l)$ and set

$$\overline{X} = X \sqcup \bigsqcup_{l \leqslant n} \bigsqcup_{0 < r < k(i_l)} \overline{V_r^{i_l}}$$

We then affix a copy $\mu \restriction V_r^{i_l}$ to each $\overline{V_r^{i_l}}$, to get a finite measure $\overline{\mu}$ on $\overline{X}$

We also make copies of edges. Again, for each Λ-color $i \in \{i_1, \ldots, i_n\}$ there is a list of Ω-colors $j_0, \ldots, j_{k(i)}$. We fix copies $\Theta_0 \coloneqq \bigcup_{r=1}^n \{\overline{c_\Omega^{-1}(j_l)}\}_{0 \leqslant l \leqslant k(i_r)}$, where each $e \in \overline{c_\Omega^{-1}(j_l)}$ is of the form $e = v \overset{\Omega_{j_l}}{\bullet\!\!-\!\!\bullet} x$ where $v \in \overline{V_L^{j_l}}$ and $x \in \overline{V_R^{j_l}}$ unless $l = 0$ in which $v \in V_L^{j_0} \subseteq X$ or unless $l = k(i)$ in which $x \in V_R^{j_{k(i)}} \subseteq X$. Let $\pi : \overline{X} \to X$ refer to the canonical projection such that $\pi \restriction X = id_X$ and if $\overline{x} \in \overline{X}$ is a copy of $x \in X$, $\pi(\overline{x}) = x$. Clearly, π is a finite-to-one map. Let $R_1 = \pi^{-1}(E_1), R_2 = \pi^{-1}(E_2)$ and let $R = \pi^{-1}(E)$. Verifiably, $R_1 \perp_\cap R_2$ and so $R = R_1 *_\cap R_2$ and their intersection is hyperfinite. Additionally, X is visibly a complete section for $R, R_1, R_2, R_1 \cap R_2$, so $C_{\overline{\mu}}(R_i) = C_\mu(E_i) + \overline{\mu}(\overline{X}) - \mu(X) \implies C_{\overline{\mu}}(R_i) - \overline{\mu}(\overline{X}) = C_\mu(E_i) - \mu(X)$. The same holds with R, E respectively substituting for R_i, E_i. Hence, write:

$$\overline{\Theta} \coloneqq \Theta_0 \sqcup (\Theta \setminus \Lambda_0)$$

$\overline{\Theta}$ is a decomposable graphing of R as Θ_0 handles the graphing through the copies (which covers each λ-edge we delete in Λ_0) and $\Theta \setminus \Lambda_0$ covers the rest of the original graphing of E on X. Thus by Lemma [INSERT LEMMA 27.4 HERE]

$$C_{\overline{\mu}}(\overline{\Theta}) - \overline{\mu}(\overline{X}) \geqslant \big(C_{\overline{\mu}}(R_1) - \overline{\mu}(\overline{X})\big) + \big(C_{\overline{\mu}}(R_2) - \overline{\mu}(\overline{X})\big) - \big(C_{\overline{\mu}}(R_1 \cap R_2) - \overline{\mu}(\overline{X})\big) \tag{1}$$

Since $C_{\overline{\mu}}(\overline{\Theta}) - \overline{\mu}(\overline{X}) = C_\mu(\Theta) - \mu(X)$, this gives

$$C_\mu(\Theta) - \mu(X) \geqslant \big(C_\mu(E_1) + C_\mu(E_2) - C_\mu(F)\big) - \mu(X) \tag{2}$$

$$C_\mu(\Theta) \geqslant C_\mu(E_1) + C_\mu(E_2) - C_\mu(F) \tag{3}$$

Since $C_\mu(\Theta) \leqslant C_\mu(E) + \epsilon$, we obtain

$$C_\mu(E) + \epsilon \geqslant C_\mu(E_1) + C_\mu(E_2) - C_\mu(F) \tag{4}$$

Since $\epsilon > 0$ is arbitrary, this completes the proof. $\square$