$\mathcal{V}$-antichains and The Realisability Problem

Ben Jarvis

Based on joint work with Robert Brignall

7th July 2025

Outline of talk

- At PP 2024 I used $\mathcal{V}$-antichains to conjecture a classification of 'small' antichains (those with growth rate smaller than $\nu_{\mathcal{L}} \approx 3.28277$.)

Outline of talk

- At PP 2024 I used $\mathcal{V}$-antichains to conjecture a classification of 'small' antichains (those with growth rate smaller than $\nu_{\mathcal{L}} \approx 3.28277$.)
- In this talk I will bring the story up-to-date, summarising my current knowledge about antichains up to growth rate $\nu_c \approx 3.51205$, mentioning in particular the existence of intervals of antichain growth rates which appear at around 3.5.

Outline of talk

- At PP 2024 I used $\mathcal{V}$-antichains to conjecture a classification of 'small' antichains (those with growth rate smaller than $\nu_{\mathcal{L}} \approx 3.28277$.)
- In this talk I will bring the story up-to-date, summarising my current knowledge about antichains up to growth rate $\nu_c \approx 3.51205$, mentioning in particular the existence of intervals of antichain growth rates which appear at around 3.5.
- Finally, I will introduce a connection with an open problem regarding binary factorial languages, which I dub The Realisability Problem.

Antichains

What is an antichain?

- An antichain is a collection of permutations, none of which is contained in any other

Antichains

What is an antichain?

- An antichain is a collection of permutations, none of which is contained in any other
- *Infinite* antichains in particular have strong structural consequences for the theory of permutation classes. Classes which contain infinite antichains have various 'bad' properties, to such an extent that we have a name for classes which do *not* contain infinite antichains: *well-quasi-ordered*.

Antichains

What is an antichain?

- An antichain is a collection of permutations, none of which is contained in any other
- *Infinite* antichains in particular have strong structural consequences for the theory of permutation classes. Classes which contain infinite antichains have various 'bad' properties, to such an extent that we have a name for classes which do *not* contain infinite antichains: *well-quasi-ordered*.
- Despite this, relatively few (genuinely distinct) infinite antichains of permutations are known, especially at small growth rates.

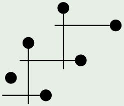
The smallest infinite antichain

The smallest (in the sense of growth rate of its downward-closure) infinite antichain is the famous *antichain of oscillations*, which occurs at growth rate $\kappa \approx 2.20557$:

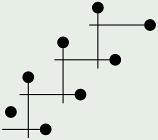
The smallest infinite antichain

The smallest (in the sense of growth rate of its downward-closure) infinite antichain is the famous *antichain of oscillations*, which occurs at growth rate $\kappa \approx 2.20557$:

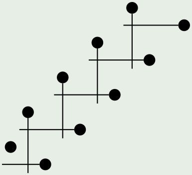
Chain of Oscillations



$\leq$



$\leq$

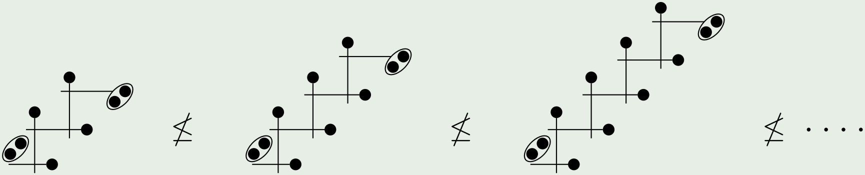


$\leq \cdots$

The smallest infinite antichain

The smallest (in the sense of growth rate of its downward-closure) infinite antichain is the famous *antichain of oscillations*, which occurs at growth rate $\kappa \approx 2.20557$:

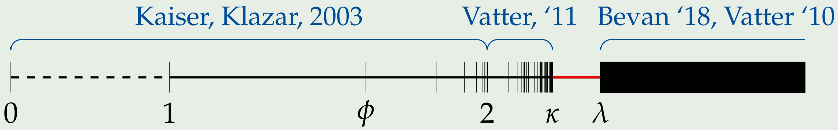
The Antichain $\mathcal{O}$



The smallest infinite antichain

The smallest (in the sense of growth rate of its downward-closure) infinite antichain is the famous *antichain of oscillations*, which occurs at growth rate $\kappa \approx 2.20557$:

Explains phase-transitions at $\kappa \approx 2.206$ and $\lambda \approx 2.357$ in growth-rate diagram:



Motivation

- Vatter/Bevan construction of permutation classes of every growth rate after $\lambda \approx 2.357$ feels a bit like a cheat: is there a deeper structure to the number line after λ that is obscured by the oscillations?

Motivation

- Vatter/Bevan construction of permutation classes of every growth rate after $\lambda \approx 2.357$ feels a bit like a cheat: is there a deeper structure to the number line after λ that is obscured by the oscillations?
- Motivates interest in classifying the growth rates of permutation classes with bounded oscillations. The solid block beginning at λ will be 'peeled back' to reveal further structure underneath, at least until the appearance of the *second* infinite antichain.

Motivation

- Vatter/Bevan construction of permutation classes of every growth rate after $\lambda \approx 2.357$ feels a bit like a cheat: is there a deeper structure to the number line after λ that is obscured by the oscillations?
- Motivates interest in classifying the growth rates of permutation classes with bounded oscillations. The solid block beginning at λ will be 'peeled back' to reveal further structure underneath, at least until the appearance of the *second* infinite antichain.
- This suggests that the key to understanding the deeper structure of the set of growth rates of permutation classes may be to understand where infinite antichains appear.

Motivation

- Vatter/Bevan construction of permutation classes of every growth rate after $\lambda \approx 2.357$ feels a bit like a cheat: is there a deeper structure to the number line after λ that is obscured by the oscillations?
- Motivates interest in classifying the growth rates of permutation classes with bounded oscillations. The solid block beginning at λ will be 'peeled back' to reveal further structure underneath, at least until the appearance of the *second* infinite antichain.
- This suggests that the key to understanding the deeper structure of the set of growth rates of permutation classes may be to understand where infinite antichains appear.
- Infinite antichains are very difficult to find, with relatively few being known. They seem especially rare for small growth rates (smaller than about 3.5). Can we classify 'small' antichains?

$\mathcal{V}$-antichains.

- We have a process for turning a binary sequence into an antichain with relatively small growth rate.

$\mathcal{V}$-antichains.

- We have a process for turning a binary sequence into an antichain with relatively small growth rate.
- These $\mathcal{V}$-classes come with an in-built structure theorem which will make them easy to enumerate, in the sense of determining the growth rate, as long as we have the recurrent complexity sequence of the defining binary sequence.

$\mathcal{V}$-antichains.

- We have a process for turning a binary sequence into an antichain with relatively small growth rate.
- These $\mathcal{V}$-classes come with an in-built structure theorem which will make them easy to enumerate, in the sense of determining the growth rate, as long as we have the recurrent complexity sequence of the defining binary sequence.
- For every $\mathcal{V}$-class we can construct a corresponding $\mathcal{V}$-antichain, an infinite antichain appearing at the same growth rate.

$\mathcal{V}$-antichains.

- We have a process for turning a binary sequence into an antichain with relatively small growth rate.
- These $\mathcal{V}$-classes come with an in-built structure theorem which will make them easy to enumerate, in the sense of determining the growth rate, as long as we have the recurrent complexity sequence of the defining binary sequence.
- For every $\mathcal{V}$-class we can construct a corresponding $\mathcal{V}$-antichain, an infinite antichain appearing at the same growth rate.
- Thus we will end up with a very large example class of antichains, whose growth rates we can find with relative ease.

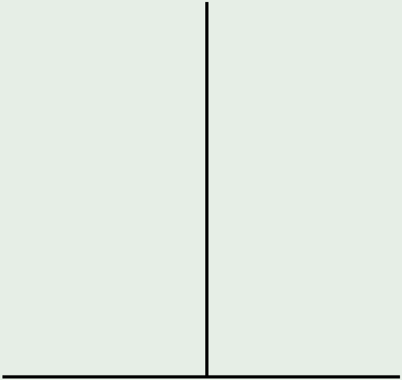
$\mathcal{V}$-antichains.

- We have a process for turning a binary sequence into an antichain with relatively small growth rate.
- These $\mathcal{V}$-classes come with an in-built structure theorem which will make them easy to enumerate, in the sense of determining the growth rate, as long as we have the recurrent complexity sequence of the defining binary sequence.
- For every $\mathcal{V}$-class we can construct a corresponding $\mathcal{V}$-antichain, an infinite antichain appearing at the same growth rate.
- Thus we will end up with a very large example class of antichains, whose growth rates we can find with relative ease.
- In particular, using binary sequences with low complexity will allow us to construct infinite antichains with very low growth rates. We shall use these to conjecture a complete classification of 'small' infinite antichains.

The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

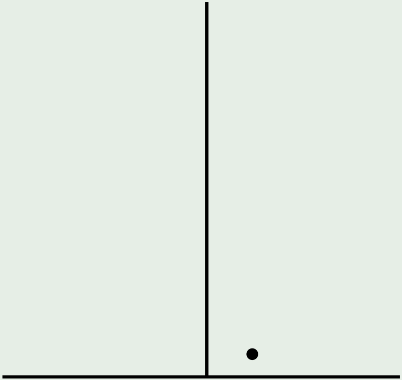
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

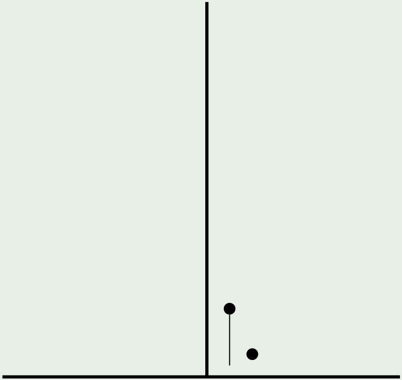
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

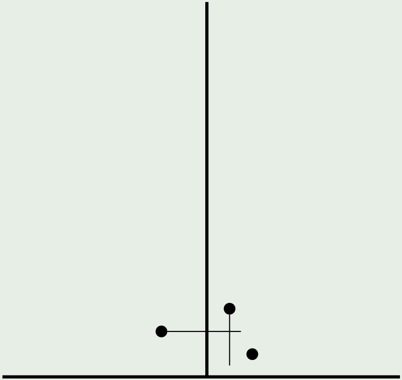
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

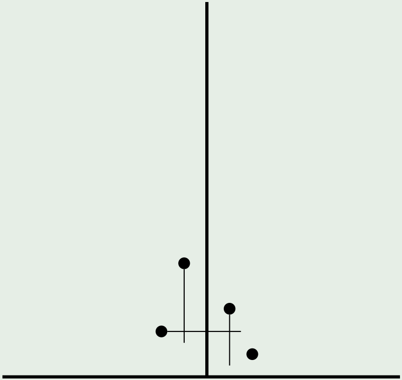
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

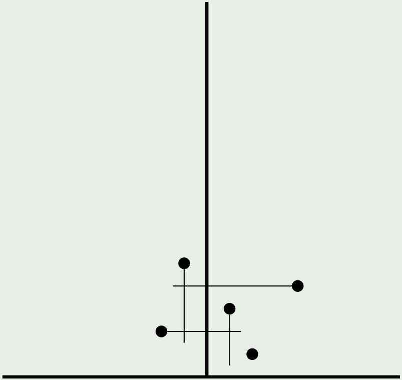
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

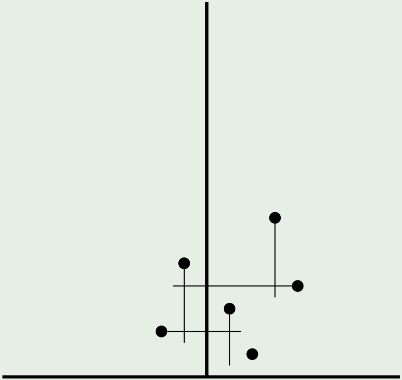
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

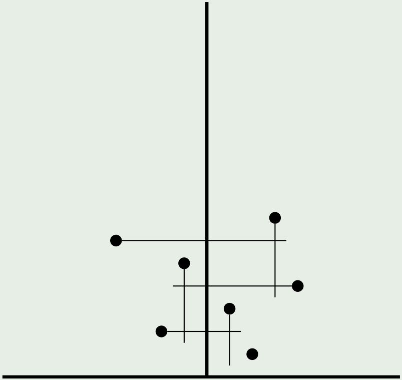
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

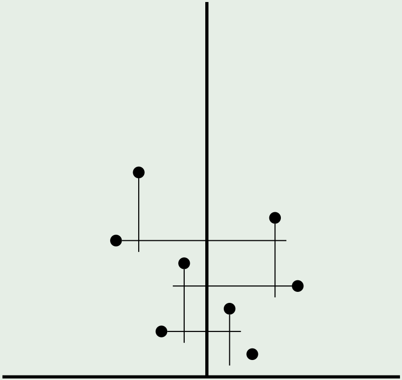
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

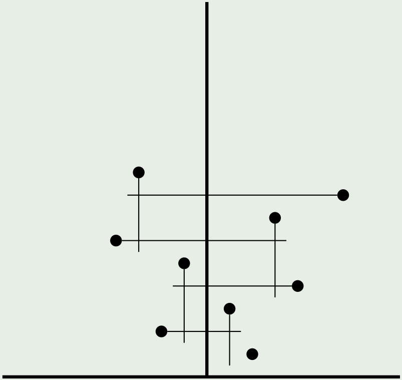
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

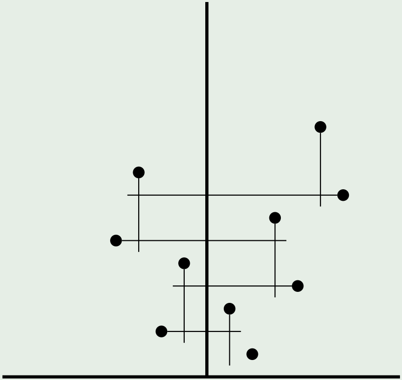
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

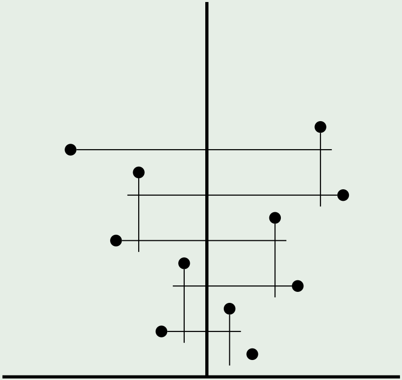
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

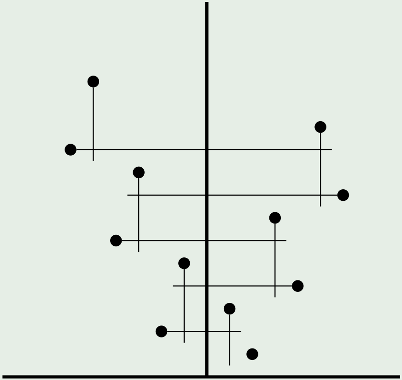
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

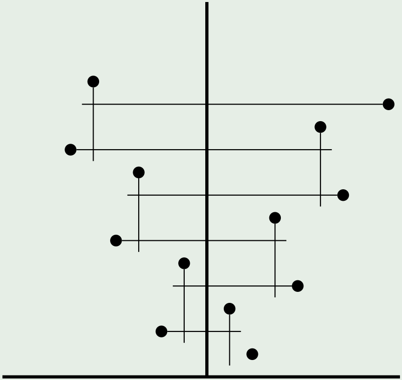
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

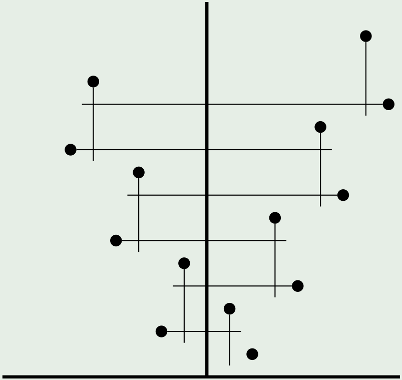
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

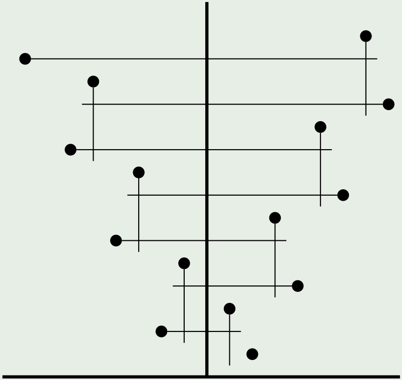
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

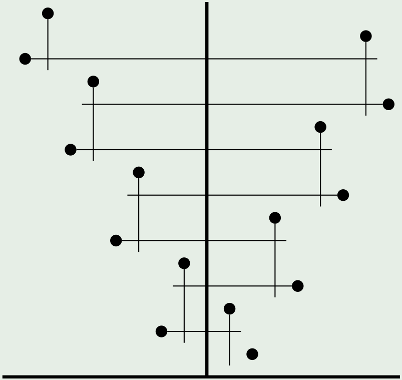
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

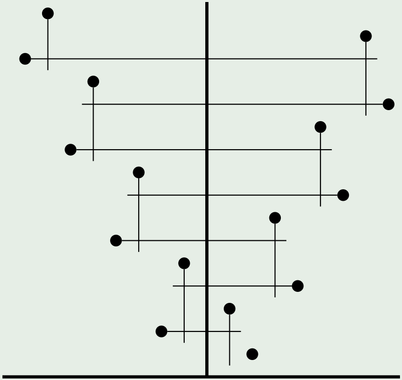
Constructing the permutation class $\mathcal{V}$



The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

Constructing the permutation class $\mathcal{V}$

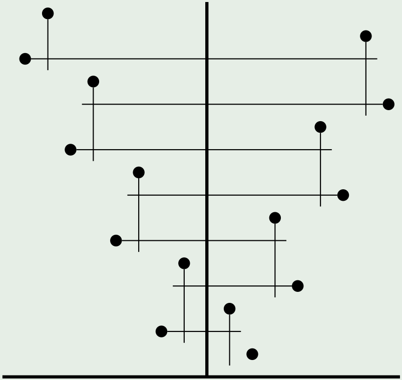


The class $\mathcal{V}$ consists of all of the permutations that can be found anywhere inside this (infinite) diagram.

The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

Constructing the permutation class $\mathcal{V}$



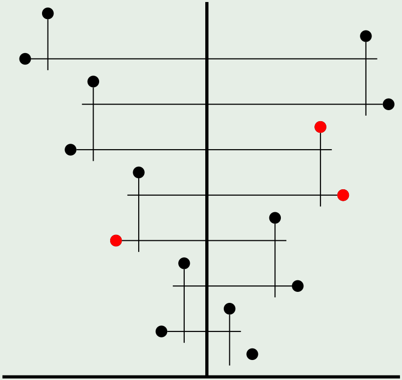
Examples:

The class $\mathcal{V}$ consists of all of the permutations that can be found anywhere inside this (infinite) diagram.

The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

Constructing the permutation class $\mathcal{V}$



Examples:

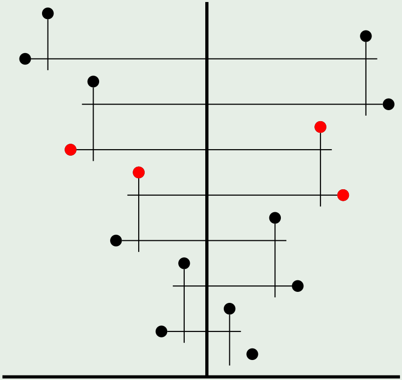
- $132 \in \mathcal{V}$

The class $\mathcal{V}$ consists of all of the permutations that can be found anywhere inside this (infinite) diagram.

The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

Constructing the permutation class $\mathcal{V}$



Examples:

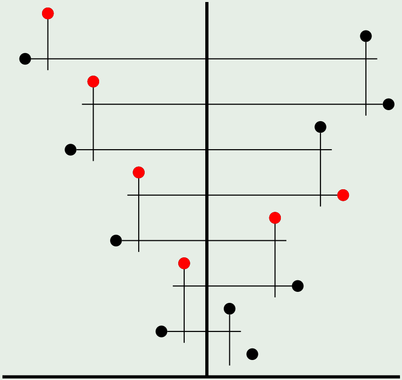
- $132 \in \mathcal{V}$
- $3241 \in \mathcal{V}$

The class $\mathcal{V}$ consists of all of the permutations that can be found anywhere inside this (infinite) diagram.

The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

Constructing the permutation class $\mathcal{V}$



Examples:

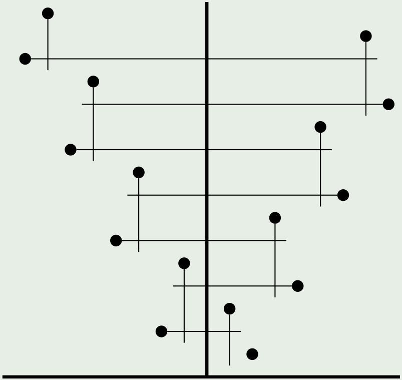
- $132 \in \mathcal{V}$
- $3241 \in \mathcal{V}$
- $654123 \in \mathcal{V}$

The class $\mathcal{V}$ consists of all of the permutations that can be found anywhere inside this (infinite) diagram.

The *second*-smallest infinite antichain?

We construct a permutation class $\mathcal{V}$ and show that an infinite antichain appears at the same growth rate:

Constructing the permutation class $\mathcal{V}$



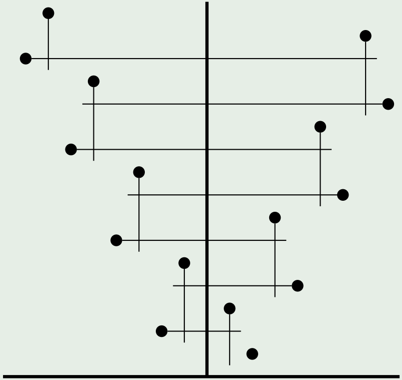
Examples:

- $132 \in \mathcal{V}$
- $3241 \in \mathcal{V}$
- $654123 \in \mathcal{V}$
- $12543 \notin \mathcal{V}$

The class $\mathcal{V}$ consists of all of the permutations that can be found anywhere inside this (infinite) diagram.

Counting the class $\mathcal{V}$

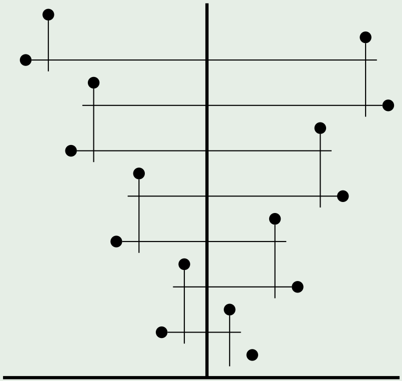
Enumerating $\mathcal{V}$



- Every $\pi \in \mathcal{V}$ is contained in this (infinite) diagram

Counting the class $\mathcal{V}$

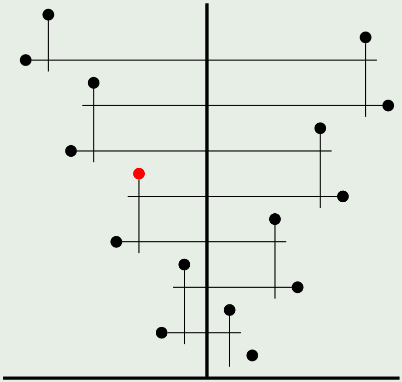
Enumerating $\mathcal{V}$



- Every $\pi \in \mathcal{V}$ is contained in this (infinite) diagram
- As soon as we remove an interior point of a $\mathcal{V}$-permutation it decomposes into the 'sum' of two contiguous chunks ('$\boxplus$-indecomposables')

Counting the class $\mathcal{V}$

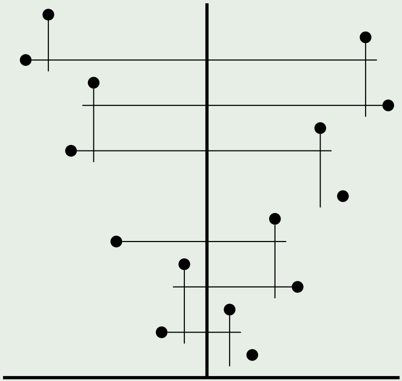
Enumerating $\mathcal{V}$



- Every $\pi \in \mathcal{V}$ is contained in this (infinite) diagram
- As soon as we remove an interior point of a $\mathcal{V}$-permutation it decomposes into the 'sum' of two contiguous chunks ('$\boxplus$-indecomposables')

Counting the class $\mathcal{V}$

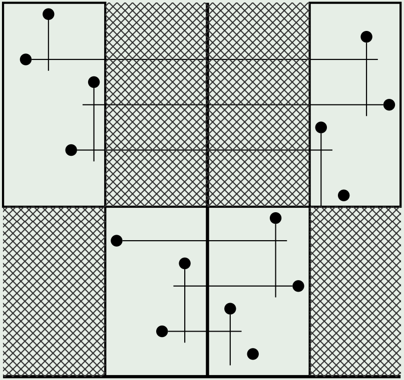
Enumerating $\mathcal{V}$



- Every $\pi \in \mathcal{V}$ is contained in this (infinite) diagram
- As soon as we remove an interior point of a $\mathcal{V}$-permutation it decomposes into the 'sum' of two contiguous chunks ('$\boxplus$-indecomposables')

Counting the class $\mathcal{V}$

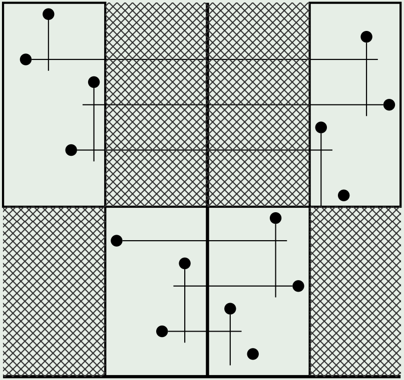
Enumerating $\mathcal{V}$



- Every $\pi \in \mathcal{V}$ is contained in this (infinite) diagram
- As soon as we remove an interior point of a $\mathcal{V}$-permutation it decomposes into the 'sum' of two contiguous chunks ('$\boxplus$-indecomposables')

Counting the class $\mathcal{V}$

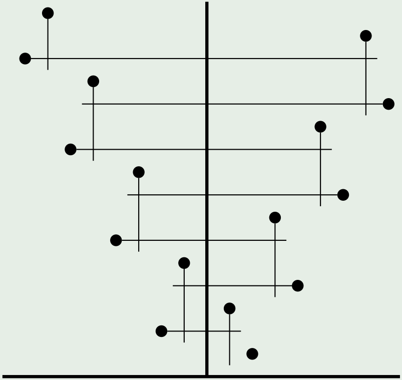
Enumerating $\mathcal{V}$



- Every $\pi \in \mathcal{V}$ is contained in this (infinite) diagram
- As soon as we remove an interior point of a $\mathcal{V}$-permutation it decomposes into the 'sum' of two contiguous chunks ('$\boxplus$-indecomposables')
- **Idea for enumerating $\mathcal{V}^\circ$:** find g.f. $g(z)$ of contiguous chunks first (easy), then 'glue' these together with the sequent operator: $f(z) = \frac{1}{1-g(z)}$

Enumerating $\mathcal{V}$

The Class $\mathcal{V}$



$\mathcal{V}^\circ$ has generating function

$$f(z) = \frac{1-z}{1-3z-2z^4}$$

Hence $\mathcal{V}$ has growth rate $\nu \approx 3.069$

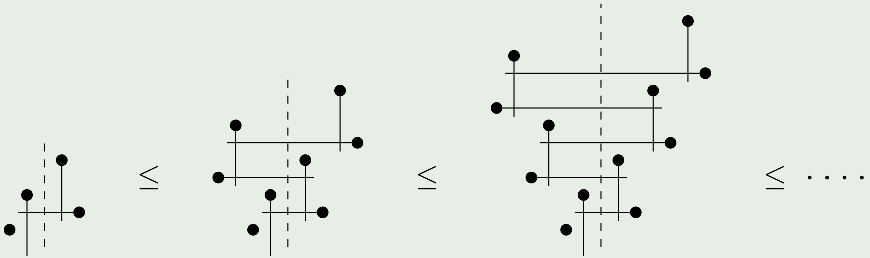
The Antichain $\mathcal{V}$

- We can construct an infinite antichain occuring at the same growth rate, $\nu \approx 3.069$

The Antichain $\mathcal{V}$

- We can construct an infinite antichain occuring at the same growth rate, $\nu \approx 3.069$

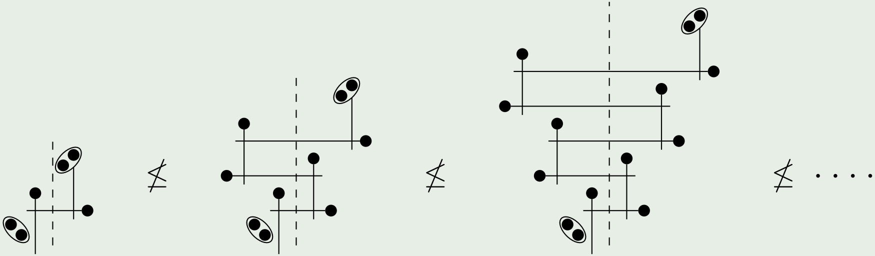
The Chain $\mathcal{V}$



The Antichain $\mathcal{V}$

- We can construct an infinite antichain occuring at the same growth rate, $\nu \approx 3.069$

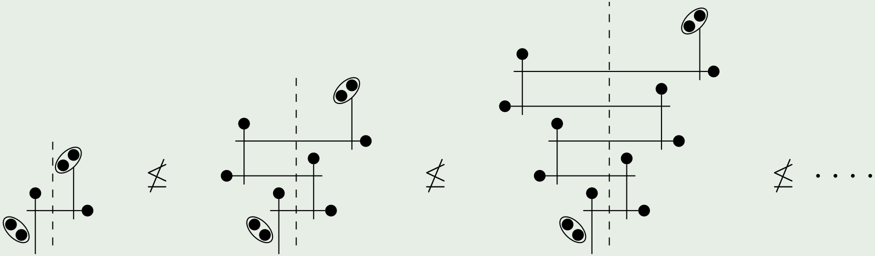
The Antichain $\mathcal{V}$



The Antichain $\mathcal{V}$

- We can construct an infinite antichain occuring at the same growth rate, $\nu \approx 3.069$

The Antichain $\mathcal{V}$



The downward closure of this antichain is sandwiched between the classes $\mathcal{V}$ and $\mathcal{V}^{+2}$, both of which have growth rate $\nu \approx 3.069$.

The Antichain $\mathcal{V}$

- We can construct an infinite antichain occuring at the same growth rate, $\nu \approx 3.069$
- Brignall conjectured that this is the second-smallest infinite antichain at PP 2018

The Antichain $\mathcal{V}$

- We can construct an infinite antichain occuring at the same growth rate, $\nu \approx 3.069$
- Brignall conjectured that this is the second-smallest infinite antichain at PP 2018
- This is *probably* true, though proof is still ongoing

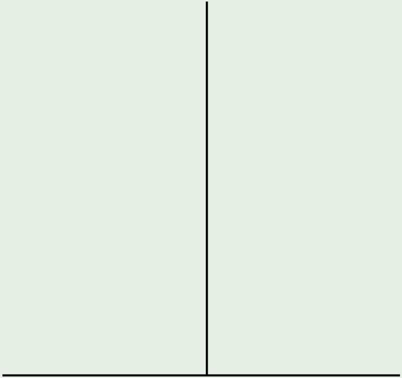
The Antichain $\mathcal{V}$

- We can construct an infinite antichain occuring at the same growth rate, $\nu \approx 3.069$
- Brignall conjectured that this is the second-smallest infinite antichain at PP 2018
- This is *probably* true, though proof is still ongoing
- Most importantly for our purposes, we can generalise the construction of $\mathcal{V}$ to generate an antichain for any given binary sequence...

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

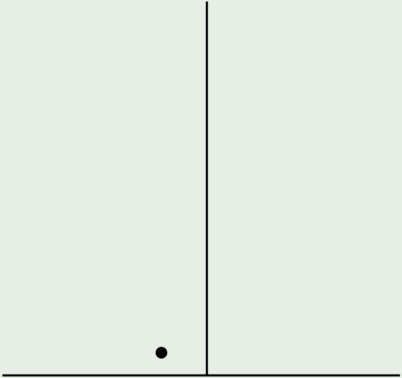


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

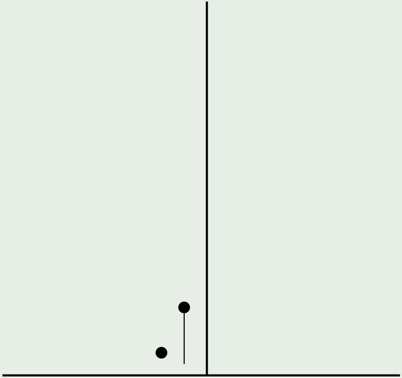


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

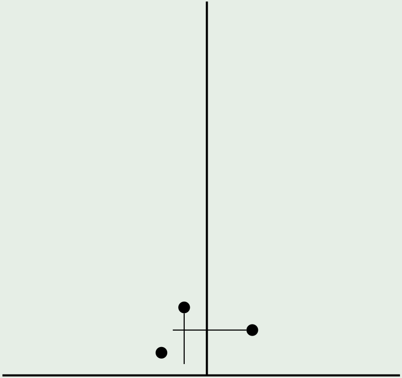


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

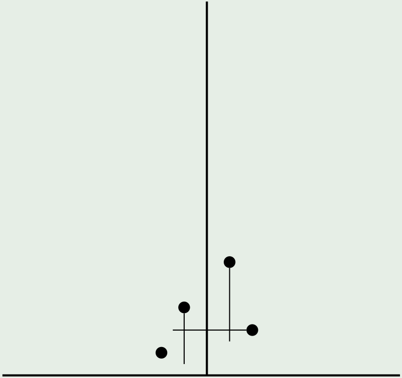


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

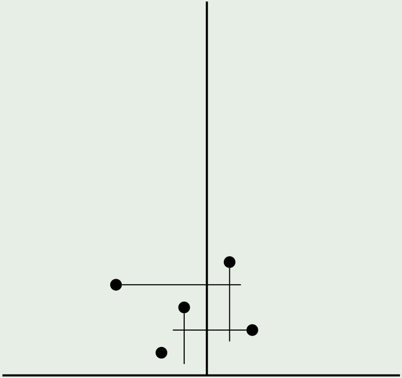


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

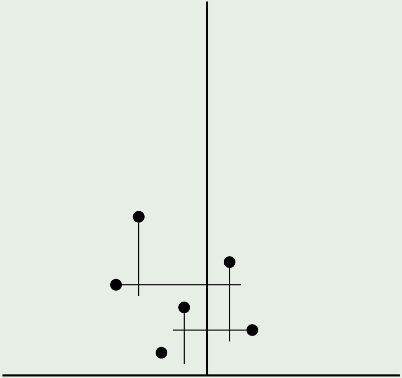


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

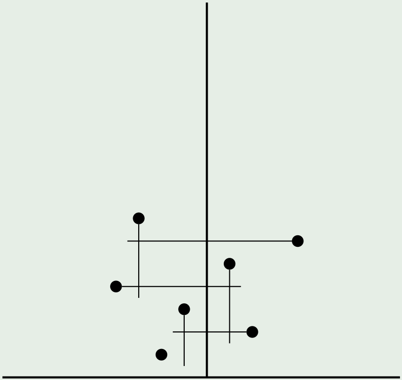


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

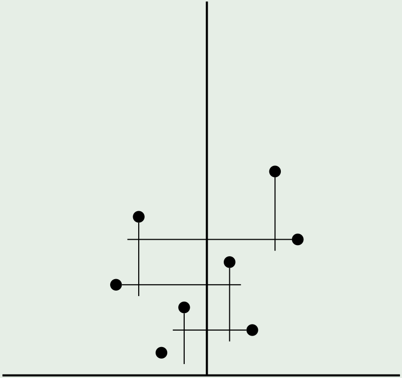


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

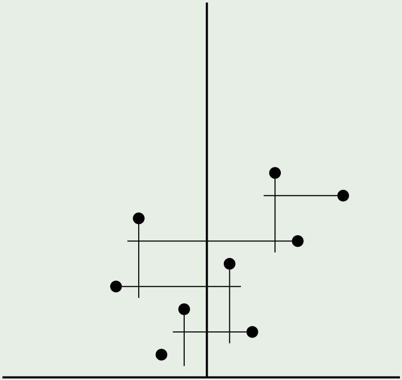


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

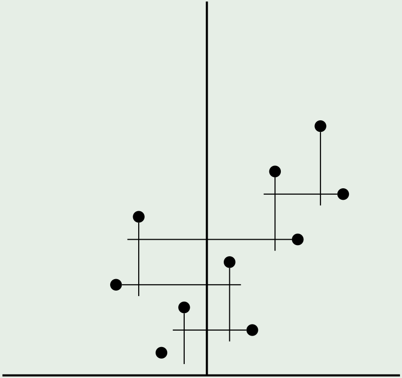


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

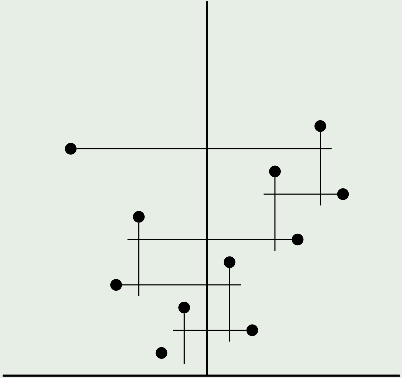


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

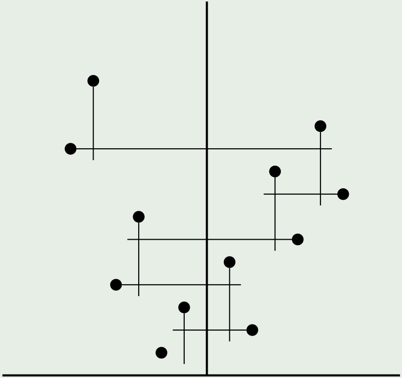


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

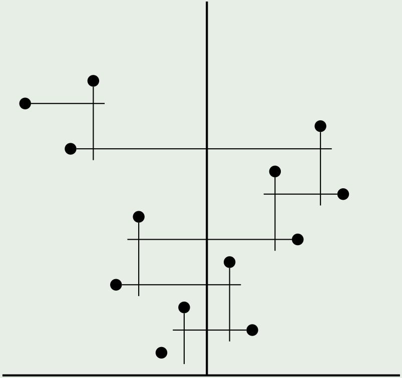


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

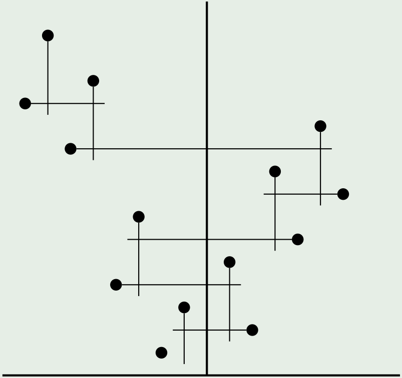


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

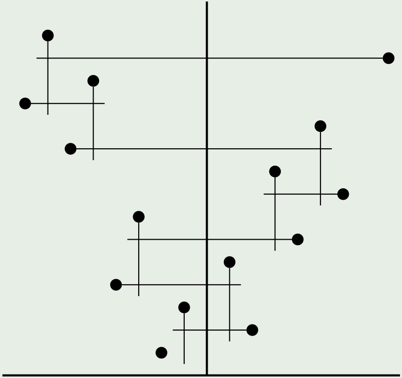


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$

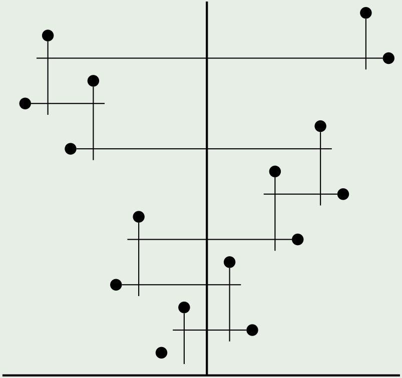


- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Turning a binary sequence into a $\mathcal{V}$-class

We use a binary sequence $\underline{b} \in \{0,1\}^{\mathbb{N}}$ to generate a $\mathcal{V}$-class:

Example: $\underline{b} = \overline{10100110} = 101001101010011010100110\ldots$



- Left step for each 1; right step for each 0
- Intersperse with up steps
- The class $\mathcal{V}_{\underline{b}}$ consists of all permutations that can be found somewhere in this (infinite) diagram

Enumerating $\mathcal{V}$-classes

- Same idea as for $\mathcal{V}$: count contiguous chunks ('$\boxplus$-indecomposables') and 'glue together' with sequent operator.
- A **recurrent factor** of $\underline{b}$ is a finite binary word which appears as a substring in $\underline{b}$ **infinitely-often**.
- For a given binary sequence $\underline{b}$, let $(q(n))$ denote the **recurrent complexity sequence**: $q(n)$ is the number of recurrent factors of $\underline{b}$ of length n.
- The recurrent complexity sequence $(q(n))$ of $\underline{b}$ allows us to find the growth rate of the corresponding $\mathcal{V}$-class.

Precisely: if $t(z) = \sum_{n=1}^{\infty} q(n) z^n$ and $q(1) = 2$ then the growth rate of $\mathcal{V}_{\underline{b}}$ is the reciprocal of the smallest positive real solution of the equation

$$\frac{(1+z)^2 t(z^2)}{z^2} - 2 - 2z - q(2)z^2 - 2z^3 = 1.$$

Enumerating $\mathcal{V}$-classes

To summarise:

- From a binary sequence $\underline{b}$ we define the $\mathcal{V}$-class $V_{\underline{b}}$.

Enumerating $\mathcal{V}$-classes

To summarise:

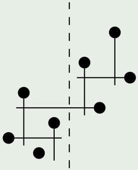
- From a binary sequence $\underline{b}$ we define the $\mathcal{V}$-class $V_{\underline{b}}$.
- Finding the growth rate of $V_{\underline{b}}$ is reduced to finding the (recurrent) complexity sequence of $\underline{b}$.

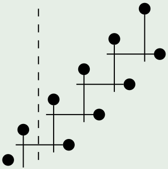
Enumerating $\mathcal{V}$-classes

To summarise:

- From a binary sequence $\underline{b}$ we define the $\mathcal{V}$-class $V_{\underline{b}}$.
- Finding the growth rate of $V_{\underline{b}}$ is reduced to finding the (recurrent) complexity sequence of $\underline{b}$.

$\underline{b}_1 = \overline{1100}$ vs $\underline{b}_2 = \overline{10000}$



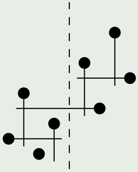


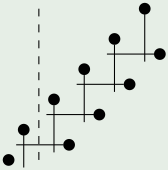
Enumerating $\mathcal{V}$-classes

To summarise:

- From a binary sequence $\underline{b}$ we define the $\mathcal{V}$-class $V_{\underline{b}}$.
- Finding the growth rate of $V_{\underline{b}}$ is reduced to finding the (recurrent) complexity sequence of $\underline{b}$.

$\underline{b}_1 = \overline{1100}$ vs $\underline{b}_2 = \overline{10000}$



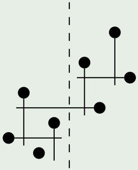


Enumerating $\mathcal{V}$-classes

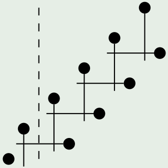
To summarise:

- From a binary sequence $\underline{b}$ we define the $\mathcal{V}$-class $V_{\underline{b}}$.
- Finding the growth rate of $V_{\underline{b}}$ is reduced to finding the (recurrent) complexity sequence of $\underline{b}$.

$\underline{b}_1 = \overline{1100}$ vs $\underline{b}_2 = \overline{10000}$



$q(n) : 2, 4, 4, 4, 4, 4, \ldots$



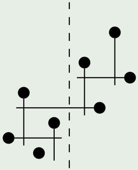
$q(n) : 2, 3, 4, 5, 5, 5, \ldots$

Enumerating $\mathcal{V}$-classes

To summarise:

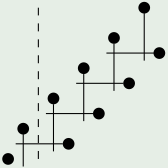
- From a binary sequence $\underline{b}$ we define the $\mathcal{V}$-class $V_{\underline{b}}$.
- Finding the growth rate of $V_{\underline{b}}$ is reduced to finding the (recurrent) complexity sequence of $\underline{b}$.

$\underline{b}_1 = \overline{1100}$ vs $\underline{b}_2 = \overline{10000}$



$q(n) : 2, 4, 4, 4, 4, 4, \ldots$

$gr(\mathcal{V}(2,2)) \approx 3.39752$



$q(n) : 2, 3, 4, 5, 5, 5, \ldots$

$gr(\mathcal{V}(1,4)) \approx 3.28248$

$\mathcal{V}$-antichains

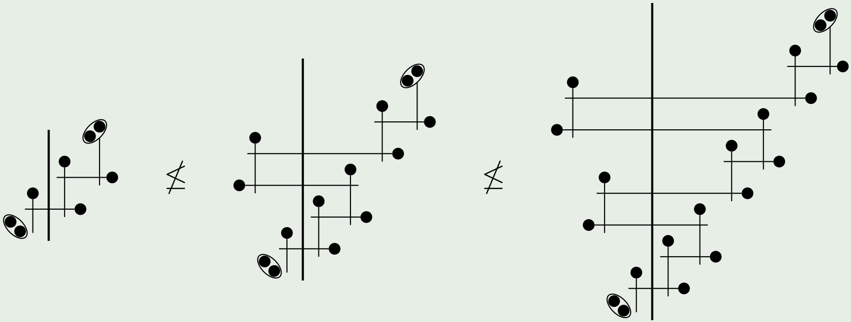
We can now generate an (infinite) antichain which occurs at the same growth rate as $\mathcal{V}_{\underline{b}}$:

$\mathcal{V}$-antichains

We can now generate an (infinite) antichain which occurs at the same growth rate as $\mathcal{V}_{\underline{b}}$:

The Antichain $\mathcal{V}(1,2)$

Taking $\underline{b} = \overline{100}$:

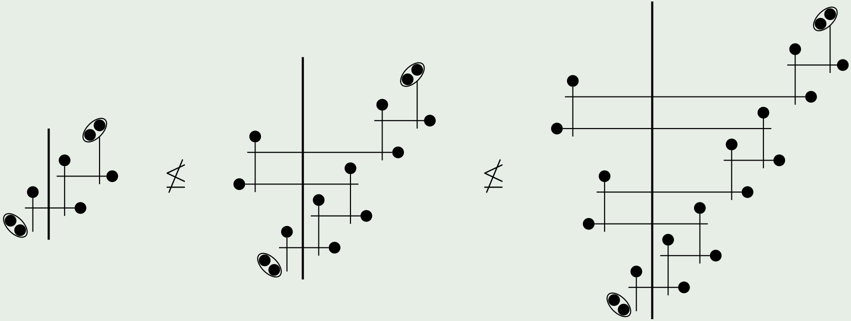


$\mathcal{V}$-antichains

We can now generate an (infinite) antichain which occurs at the same growth rate as $\mathcal{V}_{\underline{b}}$:

The Antichain $\mathcal{V}(1,2)$

Taking $\underline{b} = \overline{100}$:



$\mathcal{V}$-antichains

We can now generate an (infinite) antichain which occurs at the same growth rate as $\mathcal{V}_{\underline{b}}$:

This gives us an extraordinarily large class of *genuinely distinct* infinite antichains whose growth rates we can find (as long as we can enumerate the recurrent factors of the defining binary sequence).

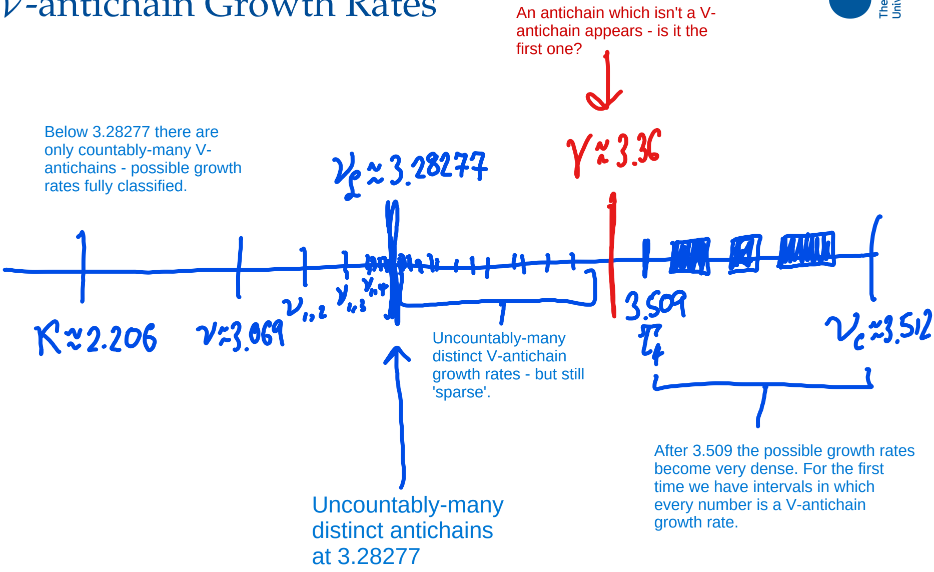
$\mathcal{V}$-antichains

We can now generate an (infinite) antichain which occurs at the same growth rate as $\mathcal{V}_{\underline{b}}$:

This gives us an extraordinarily large class of *genuinely distinct* infinite antichains whose growth rates we can find (as long as we can enumerate the recurrent factors of the defining binary sequence).

In particular, choosing defining binary sequences with low complexity will generate antichains with low growth rates...

$\mathcal{V}$-antichain Growth Rates



Uncountably-many antichain g.r.s in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$

Recall that $\nu_{\mathcal{L}} \approx 3.28277$. This is the first growth rate at which there are uncountably many genuinely distinct $\mathcal{V}$-antichains. Let $\varepsilon > 0$:

Uncountably-many growth rates in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$.

Uncountably-many antichain g.r.s in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$

Recall that $\nu_{\mathcal{L}} \approx 3.28277$. This is the first growth rate at which there are uncountably many genuinely distinct $\mathcal{V}$-antichains. Let $\varepsilon > 0$:

Uncountably-many growth rates in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$.

- Consider the following 'antichain' in the language of binary words:

$$\mathcal{A}_{\mathcal{L}} = \{11, 101, 1001, 10001, 100001, 1000001, \dots\}$$

Uncountably-many antichain g.r.s in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$

Recall that $\nu_{\mathcal{L}} \approx 3.28277$. This is the first growth rate at which there are uncountably many genuinely distinct $\mathcal{V}$-antichains. Let $\varepsilon > 0$:

Uncountably-many growth rates in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$.

- Consider the following 'antichain' in the language of binary words:

$$\mathcal{A}_{\mathcal{L}} = \{11, 101, 1001, 10001, 100001, 1000001, \dots\}$$

- Take a subset $B \subseteq \mathcal{A}$ and consider the avoidance set $Av(B)$, consisting of all binary words which do not contain any element of B as a factor (ie., contiguous subsequence).

Uncountably-many antichain g.r.s in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$

Recall that $\nu_{\mathcal{L}} \approx 3.28277$. This is the first growth rate at which there are uncountably many genuinely distinct $\mathcal{V}$-antichains. Let $\varepsilon > 0$:

Uncountably-many growth rates in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$.

- Consider the following 'antichain' in the language of binary words:

$$\mathcal{A}_{\mathcal{L}} = \{11, 101, 1001, 10001, 100001, 1000001, \dots\}$$

- Take a subset $B \subseteq \mathcal{A}$ and consider the avoidance set $Av(B)$, consisting of all binary words which do not contain any element of B as a factor (ie., contiguous subsequence).
- **Show that there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is precisely $Av(B)$.**

Uncountably-many antichain g.r.s in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$

Recall that $\nu_{\mathcal{L}} \approx 3.28277$. This is the first growth rate at which there are uncountably many genuinely distinct $\mathcal{V}$-antichains. Let $\varepsilon > 0$:

Uncountably-many growth rates in $(\nu_{\mathcal{L}}, \nu_{\mathcal{L}} + \varepsilon)$.

- Consider the following 'antichain' in the language of binary words:

$$\mathcal{A}_{\mathcal{L}} = \{11, 101, 1001, 10001, 100001, 1000001, \dots\}$$

- Take a subset $B \subseteq \mathcal{A}$ and consider the avoidance set $Av(B)$, consisting of all binary words which do not contain any element of B as a factor (ie., contiguous subsequence).
- **Show that there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is precisely $Av(B)$.**
- Enumerate the language $Av(B)$ and substitute generating function into growth rate operator.

The Realisability Problem.

We would like to be able to generalise the previous construction to general antichains. Doing so with 'denser' antichains would allow us to construct longer intervals of (**trim**) antichain growth rates. But we run up against the **realisability problem:**

The Realisability Problem.

We would like to be able to generalise the previous construction to general antichains. Doing so with 'denser' antichains would allow us to construct longer intervals of (**trim**) antichain growth rates. But we run up against the **realisability problem:**

The Realisability Problem.

- Let $\{0,1\}^*$ denote the set of all finite binary words and let B be a subset of $\{0,1\}^*$, none of whose elements is a factor (ie., contiguous subsequence) of any other.
- We take B to be the antidictionary of a factorial language: $Av(B)$ is the set of binary words which do not contain any element of B as a factor.
- **Is there an infinite binary sequence $\underline{b}$ whose set of recurrent factors is precisely $Av(B)$?**

The Realisability Problem.

We would like to be able to generalise the previous construction to general antichains. Doing so with 'denser' antichains would allow us to construct longer intervals of (**trim**) antichain growth rates. But we run up against the **realisability problem:**

The Realisability Problem.

- Let $\{0,1\}^*$ denote the set of all finite binary words and let B be a subset of $\{0,1\}^*$, none of whose elements is a factor (ie., contiguous subsequence) of any other.
- We take B to be the antidictionary of a factorial language: $Av(B)$ is the set of binary words which do not contain any element of B as a factor.
- **Is there an infinite binary sequence $\underline{b}$ whose set of recurrent factors is precisely $Av(B)$?**

Ideally we would like an effective decision procedure that answers this for any given B.

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Examples:

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Examples:

- $Av(11, 101, 1001, 10001, 100001, \ldots)$ is realisable. Then

$$\underline{b}_{\mathcal{L}} = 1010010001000010000010000000\ldots \quad \text{works.}$$

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Examples:

- $Av(11, 101, 1001, 10001, 100001, \dots)$ is realisable. Then

$$\underline{b}_{\mathcal{L}} = 101001000100001000001000000\dots \quad \text{works.}$$

- $Av(11)$ is realisable: Write $Av(11) = \{b_1, b_2, b_3, \dots\}$. Then

$$\underline{b} = b_1 0 b_1 0 b_2 0 b_1 0 b_2 0 b_3 0 b_1 \dots \quad \text{works.}$$

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Examples:

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Examples:

- $Av(0001)$ is not realisable: $\underline{b}$ must contain 000 as a recurrent factor, but then there is no way to back to a 1 without introducing a copy of 0001.

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Examples:

- $Av(0001)$ is not realisable: $\underline{b}$ must contain 000 as a recurrent factor, but then there is no way to back to a 1 without introducing a copy of 0001.
- $Av(1010, 1011)$ is not realisable: $\underline{b}$ must contain 101 as a recurrent factor but then next letter introduces a copy of either 1010 or 1011.

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Examples:

- $Av(0001)$ is not realisable: $\underline{b}$ must contain 000 as a recurrent factor, but then there is no way to back to a 1 without introducing a copy of 0001.
- $Av(1010, 1011)$ is not realisable: $\underline{b}$ must contain 101 as a recurrent factor but then next letter introduces a copy of either 1010 or 1011.
- $Av(0101011, 00)$ is not realisable: consider right-extensions of 010101...

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Partial results:

- **Singleton antidictionaries:** If β is a binary word which is **not** of the form $1000\ldots0$, $000\ldots01$, $0111\ldots1$ or $111\ldots10$ then the language $Av(\{\beta\})$ is realisable.

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Partial results:

- **Singleton antidictionaries:** If β is a binary word which is **not** of the form $1000\ldots0$, $000\ldots01$, $0111\ldots1$ or $111\ldots10$ then the language $Av(\{\beta\})$ is realisable.
- **Enveloped antidictionaries:** If every word in B both begins and ends with a 1 then $Av(B)$ is realisable.

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Partial results:

- **Singleton antidictionaries:** If β is a binary word which is **not** of the form $1000\ldots0$, $000\ldots01$, $0111\ldots1$ or $111\ldots10$ then the language $Av(\{\beta\})$ is realisable.
- **Enveloped antidictionaries:** If every word in B both begins and ends with a 1 then $Av(B)$ is realisable.
- If B contains a word b and its initial-dual or terminal-dual then $Av(B)$ is not realisable.

Partial Results on Realisability.

Suppose B is a set of binary words, none of which is a factor of another. We say that $Av(B)$ is **realisable** if there is an infinite binary sequence $\underline{b}$ whose set of recurrent factors is $Av(B)$.

Partial results:

- **Singleton antidictionaries:** If β is a binary word which is **not** of the form $1000\ldots0$, $000\ldots01$, $0111\ldots1$ or $111\ldots10$ then the language $Av(\{\beta\})$ is realisable.
- **Enveloped antidictionaries:** If every word in B both begins and ends with a 1 then $Av(B)$ is realisable.
- If B contains a word b and its initial-dual or terminal-dual then $Av(B)$ is not realisable.

A complete solution to this problem would **likely** enable us to construct longer intervals of growth rates of (trim) antichains.

A Mysterious Fact:

It is certainly not the case that every avoidance set $Av(B)$ of binary words is realisable. But every **unbounded** avoidance set $Av(B)$ of which I am aware has **precisely the same enumeration sequence as a realisable language.** Is this a coincidence or a general result? If true it would allow the construction of wider intervals of trim antichains.

Examples:

$$Av(01) \cong Av(11, 101, 1001, 10001, \ldots)$$

$$Av(001) \cong Av(101, 10001, 1000001, 100000001, \ldots)$$

$$Av(010, 11) \cong Av(11, 101, 1001, 0000)$$

non-realisable | Realisable

The Realisability Problem.

We would like to be able to generalise the previous construction to general antichains. Doing so with 'denser' antichains would allow us to construct longer intervals of (**trim**) antichain growth rates. But we run up against the **realisability problem:**

The Realisability Problem.

- Let $\{0,1\}^*$ denote the set of all finite binary words and let B be a subset of $\{0,1\}^*$, none of whose elements is a factor (ie., contiguous subsequence) of any other.
- We take B to be the antidictionary of a factorial language: $Av(B)$ is the set of binary words which do not contain any element of B as a factor.
- **Is there an infinite binary sequence $\underline{b}$ whose set of recurrent factors is precisely $Av(B)$?**

Ideally we would like an effective decision procedure that answers this for any given B.