Growing binary trees

Olivier Bodini[1], Antoine Genitrini[2], **Khaydar Nurligareev**[2]
(1) LAGA, USPN; (2) LIP6, Sorbonne University

Workshop COMETA-GAE

IRIF, Paris Cité

June 15, 2026

Settings

Vertex types

- $\bullet$ internal node
- $\circ$ anchor (active leaf)
- $\square$ leaf (dead leaf)

Settings

Vertex types

- $\bullet$ internal node
- $\circ$ anchor (active leaf)
- $\square$ leaf (dead leaf)

$t = 0$

$\circ$

Growing process

- At the beginning ($t = 0$), our tree is an anchor $\circ$

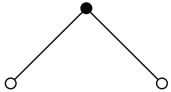
Olivier Bodini[1], Antoine Genitrini[2], Khaydar Nurligareev[2] [-2em] (1) LAGA, USPN; (2) LIP6, Sorbonne University

Settings

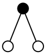
Vertex types

- ● internal node
- ○ anchor (active leaf)
- □ leaf (dead leaf)

$t = 1$



Growing process

- At the beginning ($t = 0$), our tree is an anchor ○
- At any moment $t \geqslant 1$, replace each anchor ○ by
  - a leaf □
  - or a subtree 

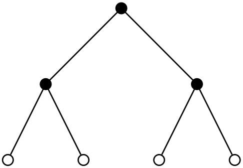
Studied objects: active trees (i.e. trees that have anchors)

Settings

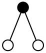
Vertex types

- ● internal node
- ○ anchor (active leaf)
- □ leaf (dead leaf)

$t = 2$



Growing process

- At the beginning ($t = 0$), our tree is an anchor ○
- At any moment $t \geqslant 1$, replace each anchor ○ by
  - a leaf □
  - or a subtree 

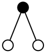
Studied objects: active trees (i.e. trees that have anchors)

Settings

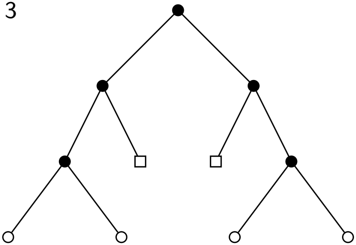
Vertex types

- $\bullet$ internal node
- $\circ$ anchor (active leaf)
- $\square$ leaf (dead leaf)

Growing process

- At the beginning ($t = 0$), our tree is an anchor $\circ$
- At any moment $t \geqslant 1$, replace each anchor $\circ$ by
  - a leaf $\square$
  - or a subtree 

$t = 3$



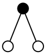
Studied objects: active trees (i.e. trees that have anchors)

Settings

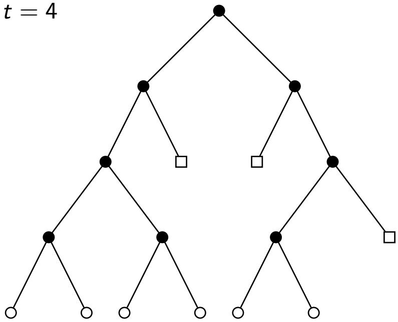
Vertex types

- ● internal node
- ○ anchor (active leaf)
- □ leaf (dead leaf)

Growing process

- At the beginning ($t = 0$), our tree is an anchor ○
- At any moment $t \geqslant 1$, replace each anchor ○ by
  - a leaf □
  - or a subtree 

$t = 4$



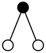
Studied objects: active trees (i.e. trees that have anchors)

Settings

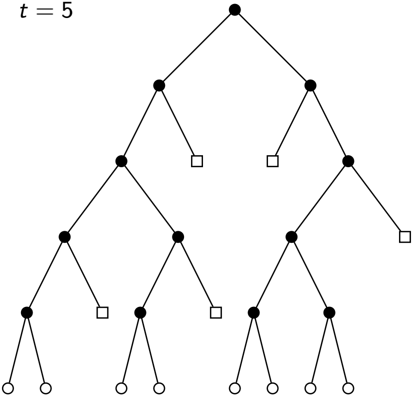
Vertex types

- ● internal node
- ○ anchor (active leaf)
- □ leaf (dead leaf)

Growing process

- At the beginning ($t = 0$), our tree is an anchor ○
- At any moment $t \geqslant 1$, replace each anchor ○ by
  - a leaf □
  - or a subtree 

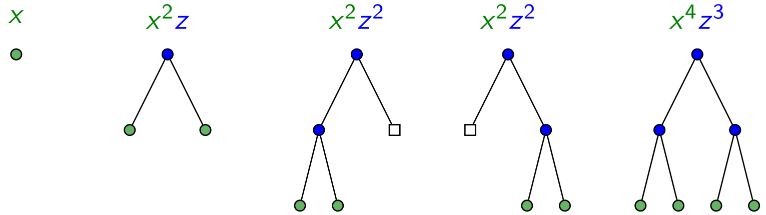
$t = 5$



Studied objects: active trees (i.e. trees that have anchors)

Counting

$$t_{n,m} = \{\text{active trees with } n \text{ internal nodes } m \text{ anchors}\}$$



$t_{0,1} = 1$ $t_{1,2} = 1$ $t_{2,2} = 2$ $t_{3,4} = 1$

Marking variables:

- x marks anchors
- z marks internal nodes

$$T(x,z) = \sum_{n=0}^{\infty} \sum_{m=1}^{\infty} t_{n,m} x^m z^n$$

$$T(x,z) = x + x^2 z + 2x^2 z^2 + \dots$$

Functional equation

- Replacement rules and specification through substitution:

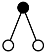
$$x \mapsto 1 \qquad\qquad x \mapsto zx^2$$

- Equation:

$$T(x,z) = x + T(1+zx^2, z) - T(1,z)$$

Functional equation

- Replacement rules and specification through substitution:

$$\circ \mapsto \square \qquad\qquad \circ \mapsto$$ 

$$x \mapsto 1 \qquad\qquad x \mapsto zx^2$$

- Equation:

$$T(x,z) = x + T(1+zx^2, z) - C(z)$$

  - $C(z) = 1 + zC^2(z)$
  - C_n are Catalan numbers

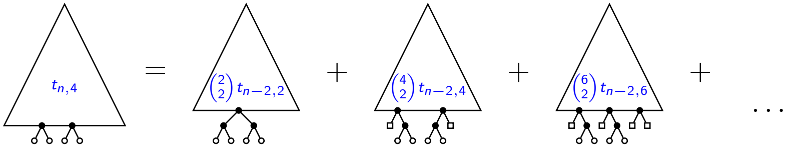
$$\sum_{m=1}^{\infty} t_{n,m} = C_n$$

- Recurrent relation $(n, k > 0)$:

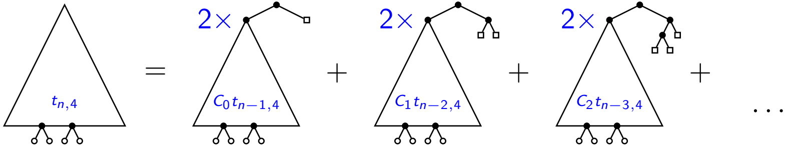
$$t_{n,2k-1} = 0 \qquad \text{and} \qquad t_{n,2k} = \sum_{m=k}^{\infty} \binom{m}{k} t_{n-k,m}$$

Recurrent relations

- Exact: $$t_{n,2k} = \sum_{m=\lceil k/2 \rceil}^{\infty} \binom{2m}{k} t_{n-k,2m} \qquad (n > 0)$$



- Asymptotic: $$t_{n,2k} > 2 \sum_{m=0}^{r} C_m t_{n-m-1,2k} \qquad (n \gg r)$$



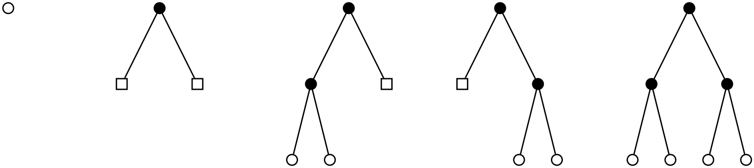
Mandelbrot Polynomials

- Define

$$p_0(x,z) = x, \qquad p_{h+1}(x,z) = 1 + z p_h^2(x,z)$$

- $p_h(x,z)$ counts binary trees:
  - of height at most h,
  - anchors are at level h,
  - leaves are at levels $\ell < h$

for $\ell < h$:
$[z^\ell] p_h(x,z) = C_\ell$



$$p_2(x,z) = 1 + z + 2x^2z^2 + x^4z^3$$

Mandelbrot Polynomials

- Define

$$p_0(x, z) = x, \qquad p_{h+1}(x, z) = 1 + z p_h^2(x, z)$$

- $p_h(x, z)$ counts binary trees:
  - of height at most h,
  - anchors are at level h,
  - leaves are at levels $\ell < h$

for $\ell < h$:
$[z^\ell] p_h(x, z) = C_\ell$

- Define

$$q_h(x, z) = z p_h(x, z)$$

- $q_h(1, z)$ are known as Mandelbrot Polynomials (see [CCC21])

- properties of $q_h(1, z) \quad \leftrightarrow \quad$ properties of $p_h(x, z)$

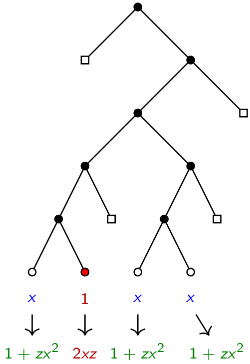
Cumulative value of anchors

- Denote

$$\widetilde{T}(x,z) := \frac{\partial T}{\partial x}(x,z) = \sum_{n=0}^{\infty}\sum_{m=1}^{\infty} m t_{n,m} x^{m-1} z^n$$

- Equation:

$$\widetilde{T}(x,z) = 1 + 2xz\,\widetilde{T}(1+zx^2, z)$$



Cumulative value of anchors

- Denote

$$\widetilde{T}(x,z) := \frac{\partial T}{\partial x}(x,z) = \sum_{n=0}^{\infty}\sum_{m=1}^{\infty} m t_{n,m} x^{m-1} z^n$$

- Equation:

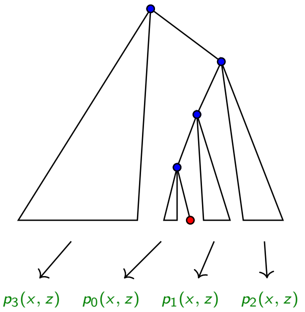
$$\widetilde{T}(x,z) = 1 + 2xz\,\widetilde{T}(1+zx^2, z)$$

- Relation:

$$\widetilde{T}(x,z) = 1 + \sum_{h=1}^{\infty} (2z)^h \prod_{\ell=0}^{h-1} p_\ell(x,z)$$

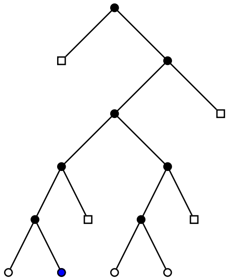
- Via Mandelbrot polynomials:

$$\widetilde{T}(1,z) = 1 + \sum_{h=1}^{\infty} 2^h \prod_{\ell=0}^{h-1} q_\ell(1,z)$$

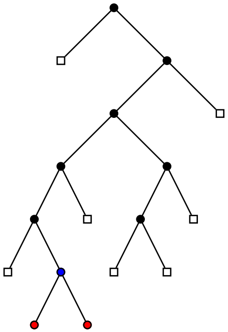


Counting equivalence

$$z\,\widetilde{T}(1,z) = [x^2]\,T(x,z)$$



$\Leftrightarrow$



It is equivalent to count:

- trees with a distinguished anchor (left),
- trees with exactly two anchors (right).

Anchor distributions

n	1	2	3	4	5	6	7	8	9	10	11
$t_{n,2}$	1	2	4	12	32	104	328	1 080	3 648	12 544	43 600
$t_{n,4}$	0	0	1	2	10	24	92	308	1 028	3 584	12 736
$t_{n,6}$	0	0	0	0	0	4	8	40	176	584	2 144
$t_{n,8}$	0	0	0	0	0	0	1	2	10	84	282
$t_{n,10}$	0	0	0	0	0	0	0	0	0	0	24
$t_{n,12}$	0	0	0	0	0	0	0	0	0	0	0
C_n	1	2	5	14	42	132	429	1 430	4 862	16 796	58 786

No horizontal line ($t_{n,2k}$) appears in the OEIS

Anchor distributions

n	1	2	3	4	5	6	7	8	9	10	11
$t_{n,2}$	1	2	4	12	32	104	328	1 080	3 648	12 544	43 600
$t_{n,4}$	0	0	1	2	10	24	92	308	1 028	3 584	12 736
$t_{n,6}$	0	0	0	0	0	4	8	40	176	584	2 144
$t_{n,8}$	0	0	0	0	0	0	1	2	10	84	282
$t_{n,10}$	0	0	0	0	0	0	0	0	0	0	24
$t_{n,12}$	0	0	0	0	0	0	0	0	0	0	0
C_n	1	2	5	14	42	132	429	1 430	4 862	16 796	58 786

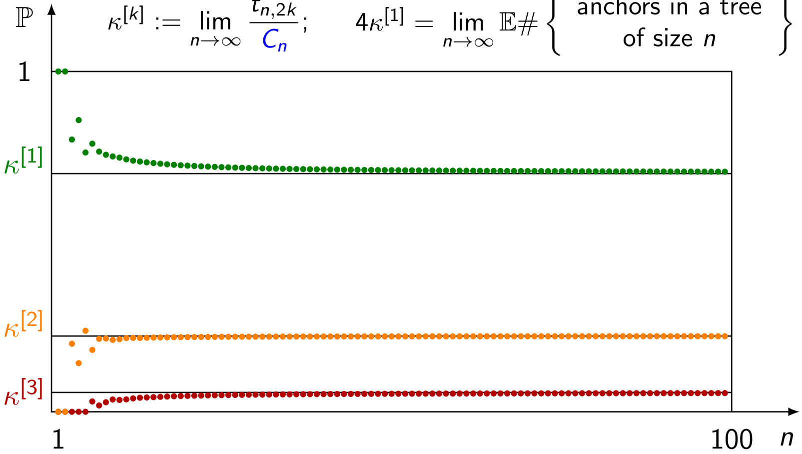
It looks like eventually

- $\frac{t_{n,2}}{C_n}$ is decreasing,
- $\frac{t_{n,2k}}{C_n}$ is increasing for $k > 1$,

and there are some limits.

Proportions of the first three lines

$$\kappa^{[k]} := \lim_{n\to\infty} \frac{t_{n,2k}}{C_n}; \qquad 4\kappa^{[1]} = \lim_{n\to\infty} \mathbb{E}\#\left\{ \begin{array}{c} \text{anchors in a tree} \\ \text{of size } n \end{array} \right\}$$



$$\kappa^{[1]} = 0.7009\ldots; \quad \kappa^{[2]} = 0.2226\ldots; \quad \kappa^{[3]} = 0.0568\ldots; \quad \kappa^{[4]} = 0.0147\ldots$$

Fixed number of anchors

- Goal: study limit proportions $\kappa^{[k]}$
- Method: asymptotic approach [FO82]
  - $p_h(1, z)$ GF of trees of height at most h,

$$e_h(z) := C(z) - p_h(1, z)$$

Fixed number of anchors

- Goal: study limit proportions $\kappa^{[k]}$
- Method: asymptotic approach [FO82]
  - $p_h(1,z)$ GF of trees of height at most h,
$$e_h(z) := C(z) - p_h(1,z)$$
  - $Q^{[k]}(z)$ GF of trees with $2k$ anchors,
$$T(x,z) = x + \sum_{k=1}^{\infty} Q^{[k]}(z) x^{2k}$$
  - $Q_h^{[k]}(z)$ GF of trees of height h with $2k$ anchors,
$$Q^{[k]}(z) = \sum_{h=1}^{\infty} Q_h^{[k]}(z)$$

$$\boxed{Q_{h+1}^{[1]}(z) = 2zC(z)\left(1 - \frac{e_h(z)}{C(z)}\right) Q_h^{[1]}(z)}$$

Scaling limits

Let

- $0 < U_1 < U_2$,
- $\tau = \sqrt{1-4z}$,
- $\tau \to 0^+ \quad (z \to 1/4), \quad h\tau \in [U_1, U_2]$.

In this case:

- Error:

$$\frac{e_h(z)}{\tau} = \frac{4}{e^{h\tau}-1} + O\left(\tau \log \frac{1}{\tau}\right)$$

- Trees with two anchors:

$$\frac{Q_h^{[1]}(z)}{\tau^2} = \frac{\kappa^{[1]}}{\sinh^2(h\tau/2)} + O\left(\tau \log \frac{1}{\tau}\right)$$

with $\kappa^{[1]} = 4 \lim_{h\to\infty} h^2 Q_h^{[1]}(1/4)$

Asymptotics for trees with two anchors

- Singular expansion:

$$Q^{[1]}(z) = Q^{[1]}(1/4) + 2\kappa^{[1]}\sqrt{1-4z} + O\big(\sqrt{1-4z}\big),$$

as $z \to 1/4$

- Asymptotics:

$$t_{n,2} = [z^n]Q^{[1]}(z) \sim \frac{\kappa^{[1]}}{\sqrt{\pi}} 4^n n^{-3/2}$$

- Limit proportions:

$$\frac{t_{n,2}}{C_n} \sim \kappa^{[1]} \qquad \frac{t_{n,2k}}{C_n} \sim \kappa^{[k]} \sim 4^{1-k}$$

(since $C_n \sim 4^n n^{-3/2}/\sqrt{\pi}$)

Column non-zero values

n	1	2	3	4	5	6	7	8	9	10	11
$t_{n,2}$	1	2	4	12	32	104	328	1 080	3 648	12 544	43 600
$t_{n,4}$	0	0	1	2	10	24	92	308	1 028	3 584	12 736
$t_{n,6}$	0	0	0	0	0	4	8	40	176	584	2 144
$t_{n,8}$	0	0	0	0	0	0	1	2	10	84	282
$t_{n,10}$	0	0	0	0	0	0	0	0	0	0	24

Column non-zero values

n	1	2	3	4	5	6	7	8	9	10	11
$t_{n,2}$	1	2	4	12	32	104	328	1 080	3 648	12 544	43 600
$t_{n,4}$	0	0	1	2	10	24	92	308	1 028	3 584	12 736
$t_{n,6}$	0	0	0	0	0	4	8	40	176	584	2 144
$t_{n,8}$	0	0	0	0	0	0	1	2	10	84	282
$t_{n,10}$	0	0	0	0	0	0	0	0	0	0	24

- Define $a_n = \max\{k \colon t_{n,2k} > 0\}$ (OEIS: A006949)

n	1	2	3	4	5	6	7	8	9	10
a_n	1	1	2	2	2	3	4	4	4	4
a_{n+10}	5	6	6	7	8	8	8	8	8	9
a_{n+20}	10	10	11	12	12	12	13	14	14	15
a_{n+30}	16	16	16	16	16	16	17	18	18	19

Column non-zero values

n	1	2	3	4	5	6	7	8	9	10	11
$t_{n,2}$	1	2	4	12	32	104	328	1 080	3 648	12 544	43 600
$t_{n,4}$	0	0	1	2	10	24	92	308	1 028	3 584	12 736
$t_{n,6}$	0	0	0	0	0	4	8	40	176	584	2 144
$t_{n,8}$	0	0	0	0	0	0	1	2	10	84	282
$t_{n,10}$	0	0	0	0	0	0	0	0	0	0	24

- Define $a_n = \max\{k : t_{n,2k} > 0\}$ (OEIS: A006949)

n	1	2	3	4	5	6	7	8	9	10
a_n	1	1	2	2	2	3	4	4	4	4
a_{n+10}	5	6	6	7	8	8	8	8	8	9
a_{n+20}	10	10	11	12	12	12	13	14	14	15
a_{n+30}	16	16	16	16	16	16	17	18	18	19

- Define $b_n = \#\{k : a_k = n\}$ (OEIS: A135560)

$$(b_n) = 2, 3, 1, 4, 1, 2, 1, 5, 1, 2, 1, 3, 1, 2, 1, 6, 1, 2, 1, \ldots$$

One property of (a_n)

Lemma

The sequence (a_n) satisfies

$$a_1 = 1, \qquad a_n = \max\{k\colon k \leqslant 2a_{n-k}\}$$

We have

$$t_{n,2k} = \sum_{m=\lceil k/2 \rceil}^{a_{n-k}} \binom{2m}{k} t_{n-k,2m}$$

and

$$t_{n,2k} \neq 0 \qquad \Leftrightarrow \qquad \exists\, m\colon\ k \leqslant 2m \leqslant 2a_{n-k}$$

Hence,

$$k \leqslant 2a_{n-k}$$

Description of (b_n)

Proposition

The sequence (b_n) satisfies

$$b_n = \begin{cases} p+2 & \text{if } n = 2^p, \\ p+1 & \text{if } n = 2^p a,\ a \text{ is odd},\ a > 1\,. \end{cases}$$

In particular,

- $b_{2n} = b_n + 1$ for even indices,
- $b_{2n+1} = 1$ for odd indices greater than 1,
- $b_1 = 2$.

Induction based on $a_n = \max\{k\colon k \leqslant 2a_{n-k}\}$

Description of (a_n)

Proposition

The sequence (a_n) satisfies

$$a_n = a_{n-1-a_{n-1}} + a_{n-2-a_{n-2}}, \qquad a_0 = a_1 = a_2 = 1$$

- Question. How to explain this recurrence combinatorially?
- (a_n) is known as a meta-Fibonacci sequence

Corollary (see [T92])

- $\lim_{n\to\infty} \frac{a_n}{n} = \frac{1}{2}$
- $\sum_{n=0}^{\infty} a_n z^n = z \sum_{n=0}^{\infty} \prod_{i=1}^{n} (z + z^{2^i})$

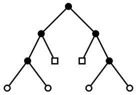
Trees of fixed height

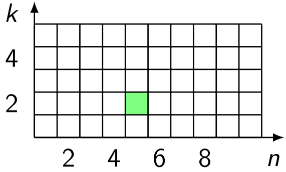
$$t_{n,2k,h} = \#\{\text{active trees with } n \text{ internal nodes } 2k \text{ anchors of height } h\}$$

k																
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0
6	0	0	0	0	0	0	0	0	0	0	0	4	28	0	0	0
	0	0	0	0	0	0	0	0	0	0	24	56	0	0	0	0
4	0	0	0	0	0	0	0	2	6	60	70	0	0	0	0	0
	0	0	0	0	0	0	8	24	80	56	0	0	0	0	0	0
2	0	0	0	0	4	16	36	60	28	0	0	0	0	0	0	0
	0	0	0	8	16	24	24	8	0	0	0	0	0	0	0	0
$t_{n,2k,4}$		2		4		6		8		10		12		14		16

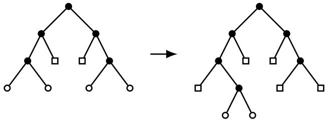
n

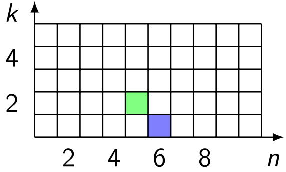
From h to $h+1$



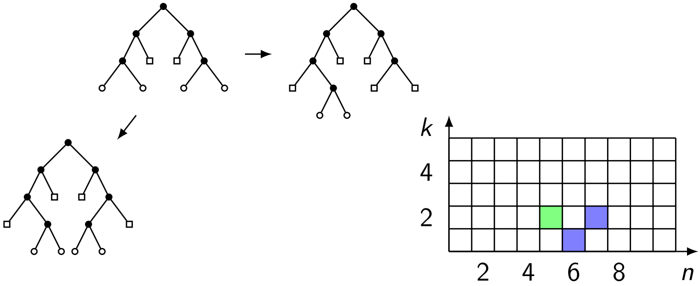


From h to $h+1$

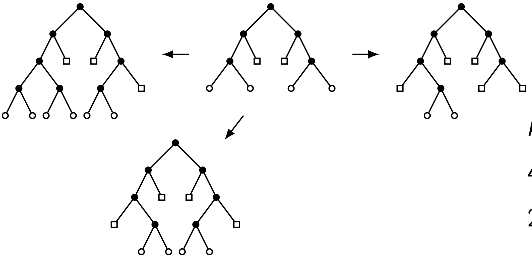


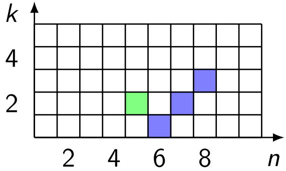


From h to $h+1$

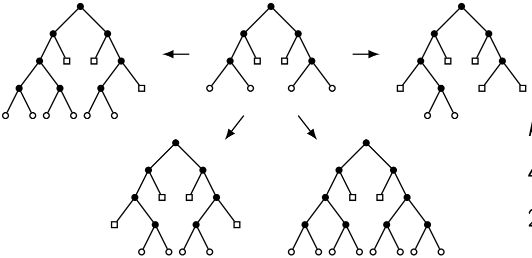


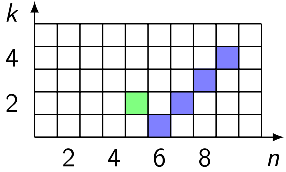
From h to $h+1$



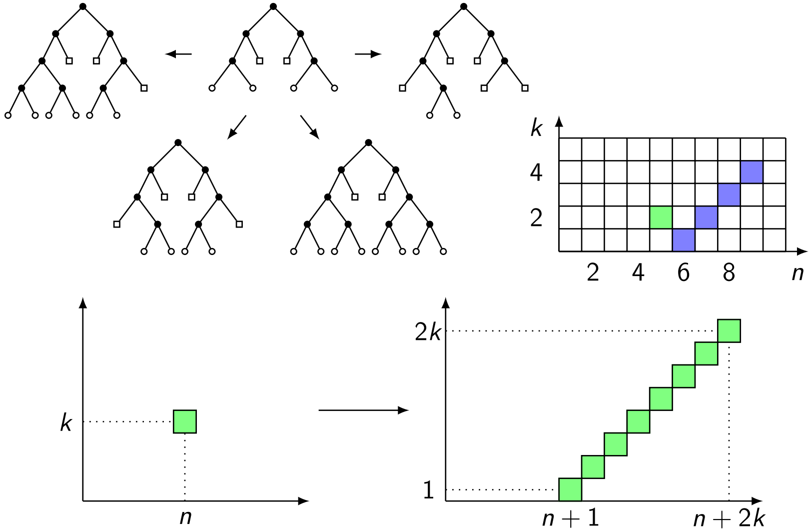


From h to $h+1$

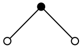


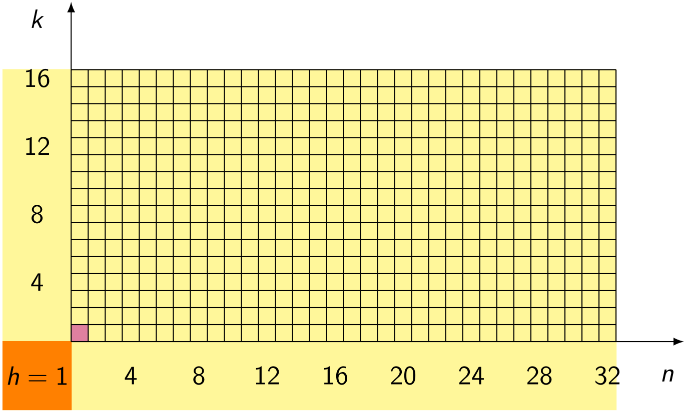


From h to $h+1$

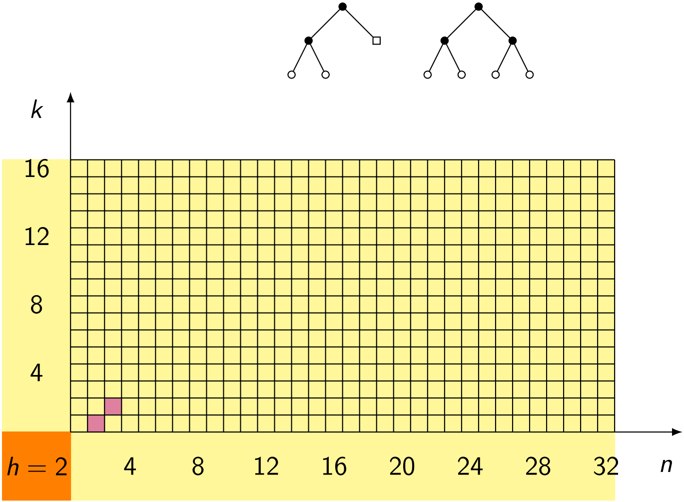


Non-zero domain properties

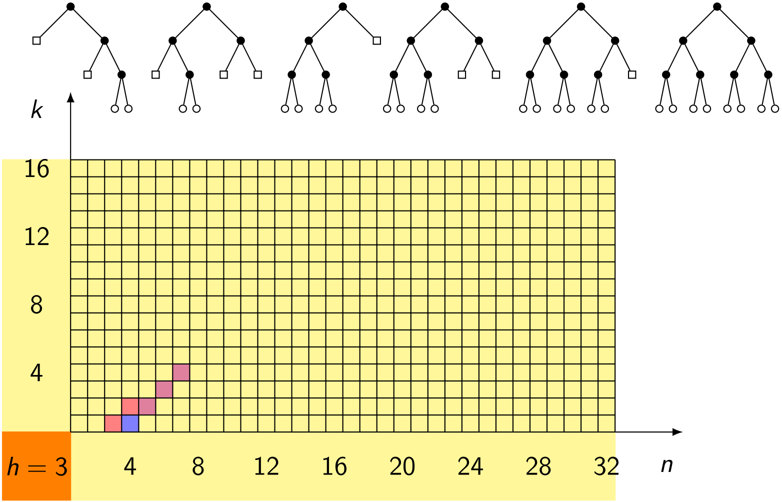




Non-zero domain properties



Non-zero domain properties



Non-zero domain properties

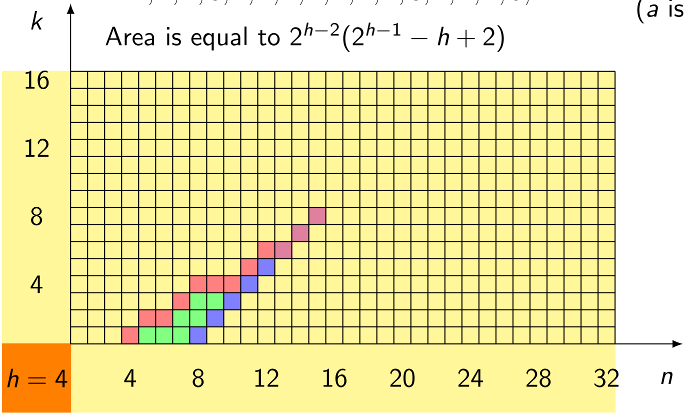
- (blue square) is on the line $n - k = 2^{h-1} - 1$
- (red square) is encoded by the ruler function $r(x)$: $1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1, 5, \ldots$

Area is equal to $2^{h-2}(2^{h-1} - h + 2)$

ruler function:

$r(2^p a) = p + 1$

(a is odd)



Non-zero domain properties

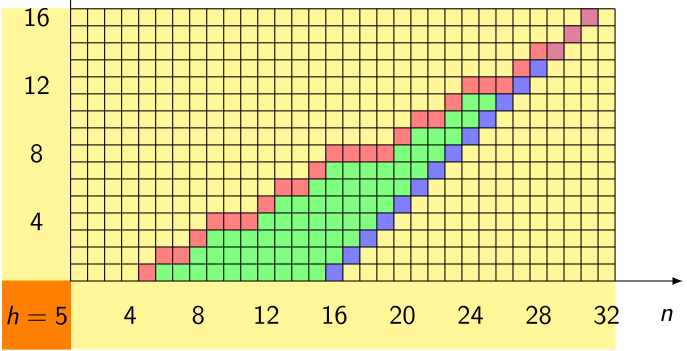
- ■ (blue) is on the line $n - k = 2^{h-1} - 1$
- ■ (red) is encoded by the ruler function $r(x)$: $1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1, 5, \ldots$

Area is equal to $2^{h-2}(2^{h-1} - h + 2)$

ruler function:

$r(2^p a) = p + 1$

(a is odd)



Non-zero domain properties

- ■ is on the line $n - k = 2^{h-1} - 1$
- ■ is encoded by the ruler function $r(x)$: $1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1, 5, \ldots$

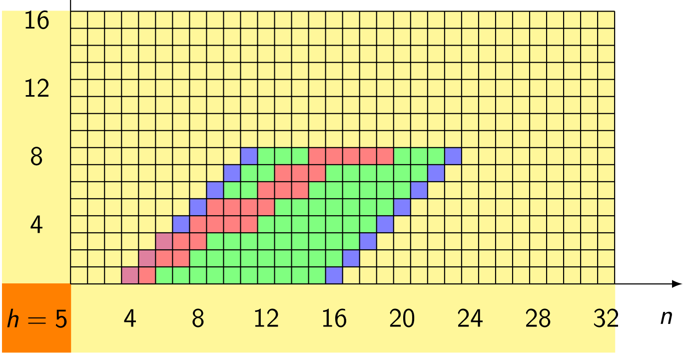
Area is equal to $2^{h-2}(2^{h-1} - h + 2)$

<u>ruler function:</u>

$r(2^p a) = p + 1$

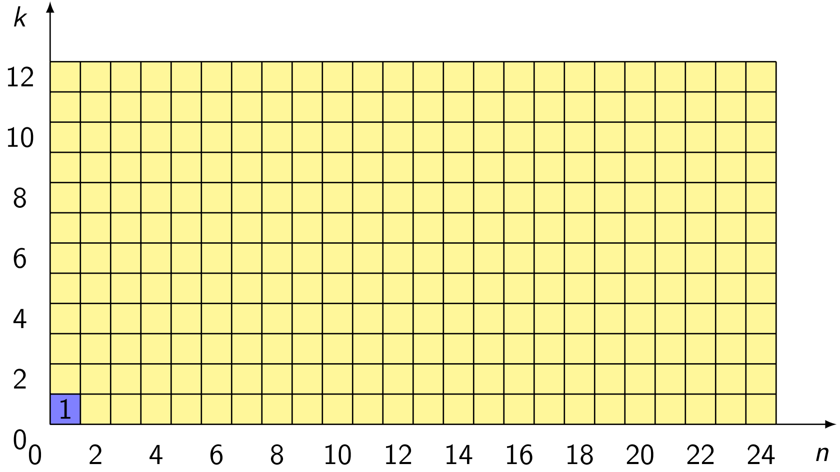
(a is odd)

k



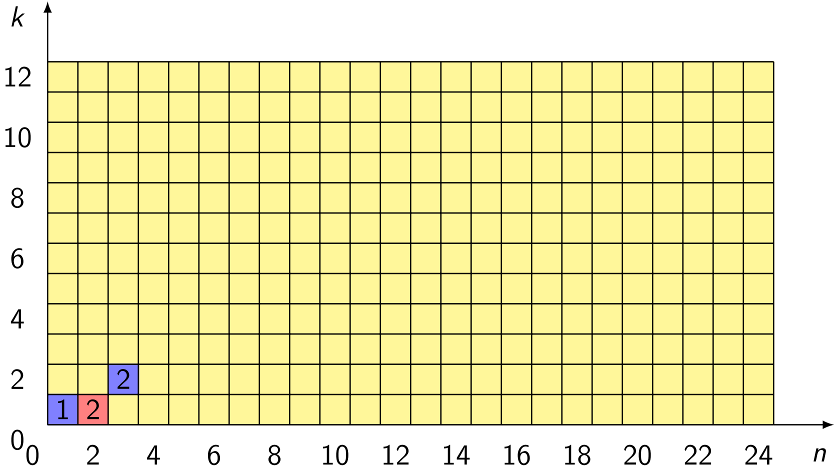
Behavior of the sequence $a_n = \max\{k : t_{n,2k} > 0\}$

Take the union of all non-zero domains



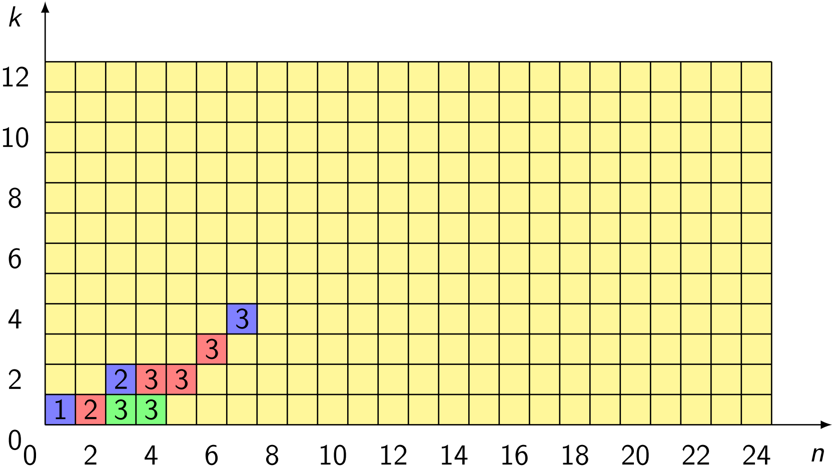
Behavior of the sequence $a_n = \max\{k \colon t_{n,2k} > 0\}$

Take the union of all non-zero domains



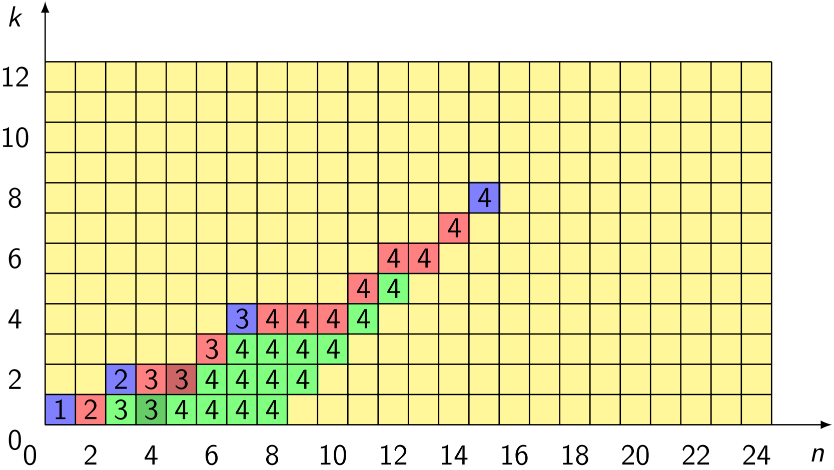
Behavior of the sequence $a_n = \max\{k\colon t_{n,2k} > 0\}$

Take the union of all non-zero domains



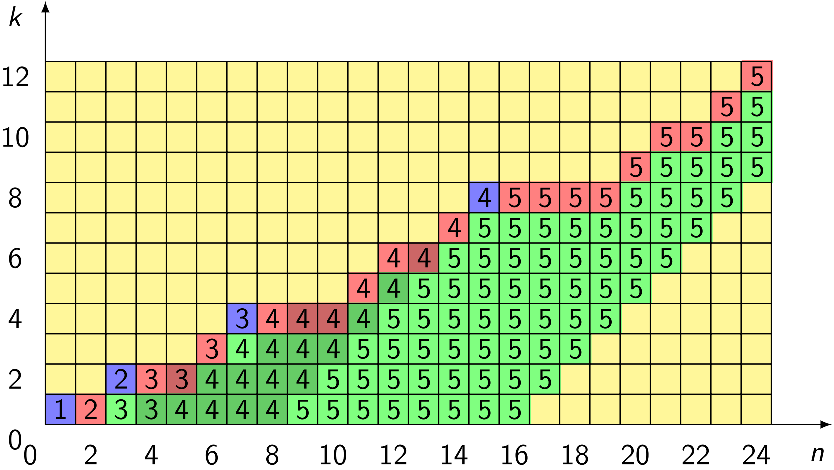
Behavior of the sequence $a_n = \max\{k \colon t_{n,2k} > 0\}$

Take the union of all non-zero domains



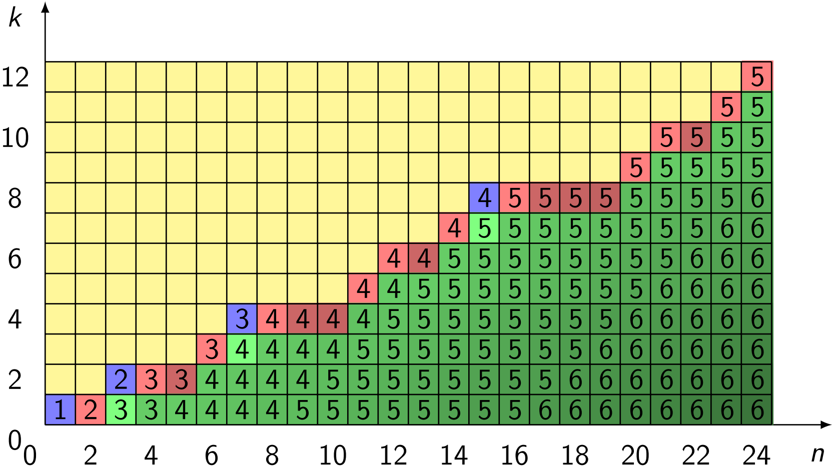
Behavior of the sequence $a_n = \max\{k\colon t_{n,2k} > 0\}$

Take the union of all non-zero domains



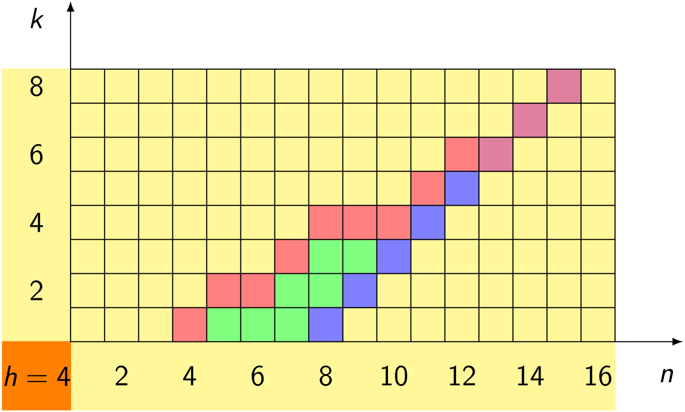
Behavior of the sequence $a_n = \max\{k : t_{n,2k} > 0\}$

Take the union of all non-zero domains



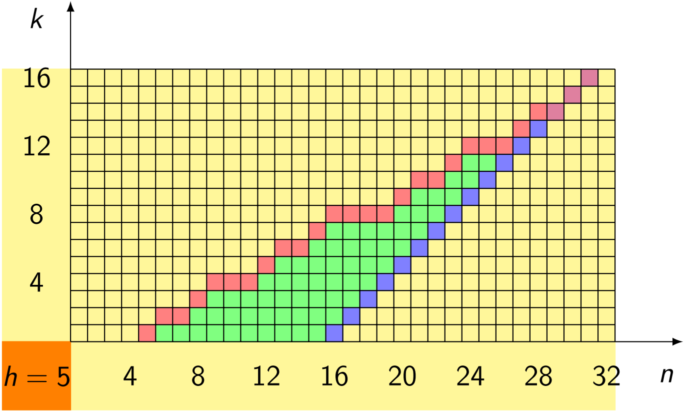
Scaling limit

The scaling limit of the non-zero domain (normalized by 2^{h-1}), as $h \to \infty$, is ...



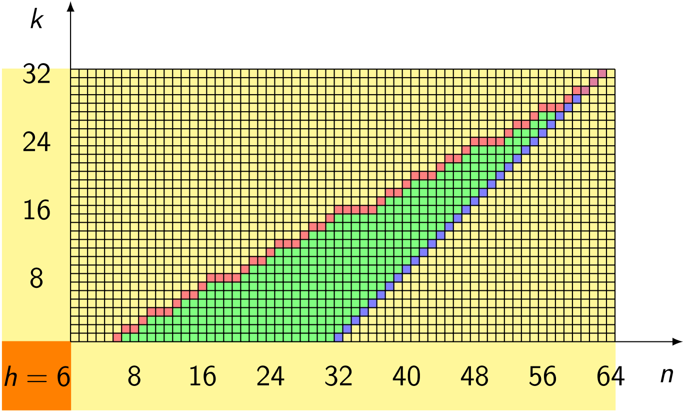
Scaling limit

The scaling limit of the non-zero domain (normalized by 2^{h-1}), as $h \to \infty$, is ...



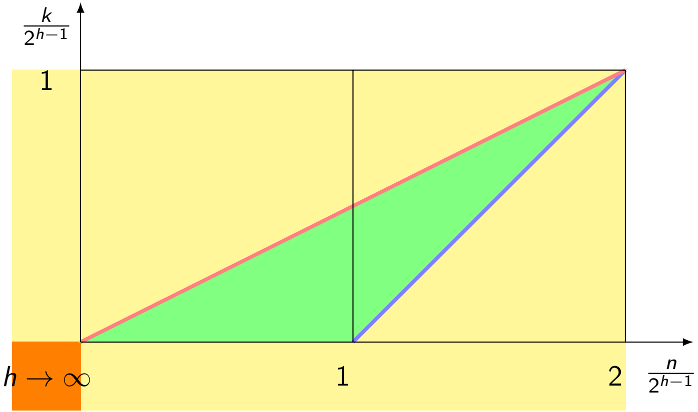
Scaling limit

The scaling limit of the non-zero domain (normalized by 2^{h-1}), as $h \to \infty$, is . . .

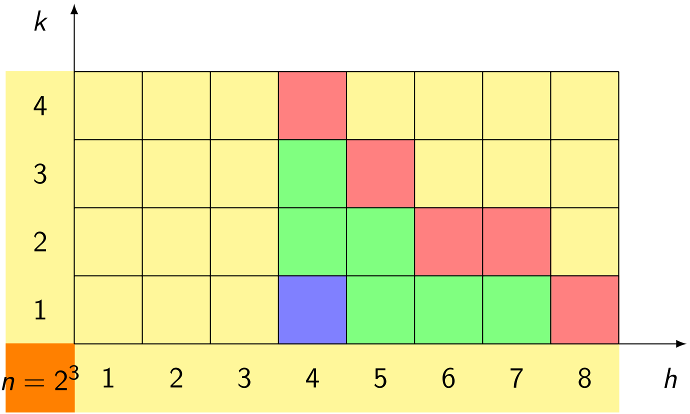


Scaling limit

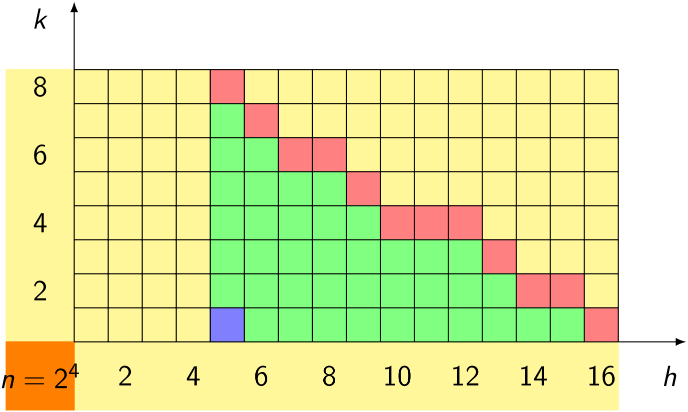
The scaling limit of the non-zero domain (normalized by 2^{h-1}), as $h \to \infty$, is the triangle with vertices $(0, 0)$, $(1, 0)$, and $(2, 1)$.



Trees of fixed size



Trees of fixed size



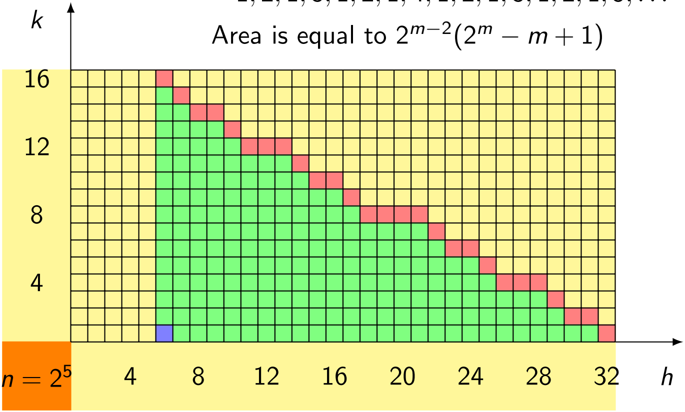
Trees of fixed size

For $n = 2^m$:

- is at $(m+1, 1)$
- is encoded by the ruler function:

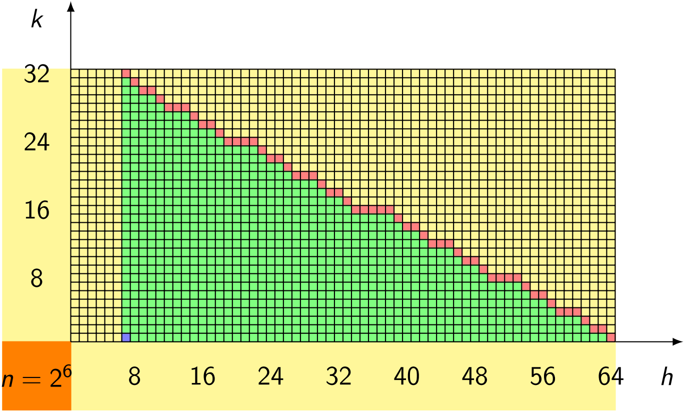
  $1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1, 5, \ldots$

Area is equal to $2^{m-2}(2^m - m + 1)$



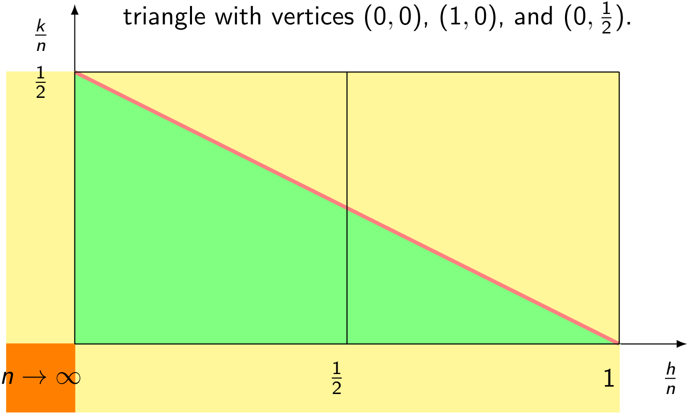
Trees of fixed size

The scaling limit of the non-zero domain
(normalized by n), as $n \to \infty$, is ...



Trees of fixed size

The scaling limit of the non-zero domain (normalized by n), as $n \to \infty$, is the triangle with vertices $(0,0)$, $(1,0)$, and $(0, \frac{1}{2})$.

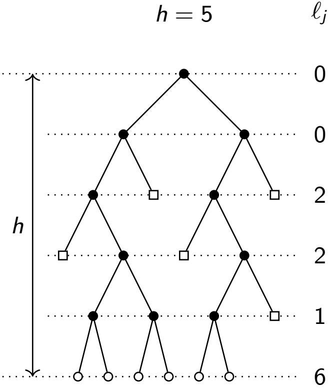


Profile of a tree

- h = height
- $\ell_j = \#\{\text{leaves at } j\text{th level}\}$
- $(\ell_0, \ldots, \ell_h)$ = tree profile

$$\sum_{j=0}^{h} \frac{\ell_j}{2^j} = 1$$

(Kraft-McMillan equality, [CT06])



$$\frac{2}{2^2} + \frac{2}{2^3} + \frac{1}{2^4} + \frac{6}{2^5} = \frac{1}{2} + \frac{1}{4} + \frac{1}{16} + \frac{3}{16} = 1$$

Profile of a tree

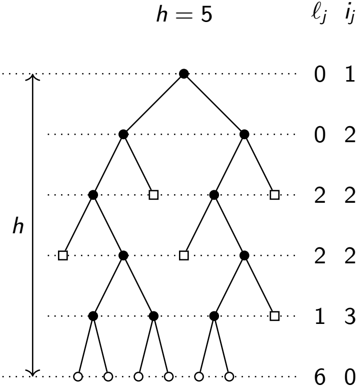
- h = height
- $\ell_j = \#\{\text{leaves at } j\text{th level}\}$
- $(\ell_0, \dots, \ell_h)$ = tree profile

$$\sum_{j=0}^{h} \frac{\ell_j}{2^j} = 1$$

(Kraft-McMillan equality, [CT06])

- $i_j = \#\{\text{nodes at } j\text{th level}\}$

$$i_0 = 1, \quad i_j + \ell_j = 2 i_{j-1}$$



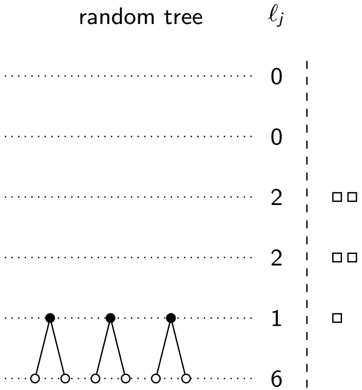
$$\#\left\{ \begin{array}{c} \text{trees with a valid} \\ \text{profile } (\ell_0, \dots, \ell_h) \end{array} \right\} = \prod_{j=0}^{h-1} \binom{2 i_j}{\ell_{j+1}}$$

Sampling trees with a given profile

Input: $L = (\ell_0, \ldots, \ell_h)$

Output: uniform tree with profile L

```
function UNIFORM_TREE(L)
    T = sequence of ℓ_h/2 nodes,
    each one pointing to 2 leaves
    for i from h − 1 to 0 do
        T′ = shuffling T with ℓ_i leaves
        T = sequence of |T′|/2 nodes,
        with the ith node pointing
        to nodes 2i and 2i + 1 of T′
    return the first node of T
```



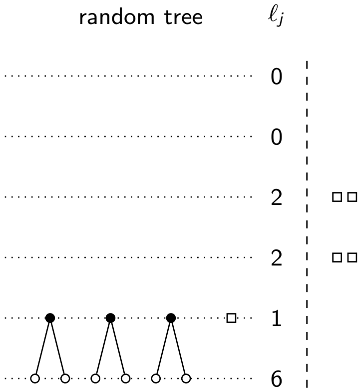
The algorithm is optimal in time, space, and random bit consumption.

Sampling trees with a given profile

Input: $L = (\ell_0, \ldots, \ell_h)$

Output: uniform tree with profile L

```
function UNIFORM_TREE(L)
    T = sequence of ℓ_h/2 nodes,
    each one pointing to 2 leaves
    for i from h − 1 to 0 do
        T′ = shuffling T with ℓ_i leaves
        T = sequence of |T′|/2 nodes,
        with the ith node pointing
        to nodes 2i and 2i + 1 of T′
    return the first node of T
```



The algorithm is optimal in time, space, and random bit consumption.

Sampling trees with a given profile

Input: $L = (\ell_0, \ldots, \ell_h)$

Output: uniform tree with profile L

1: **function** UNIFORM_TREE(L)

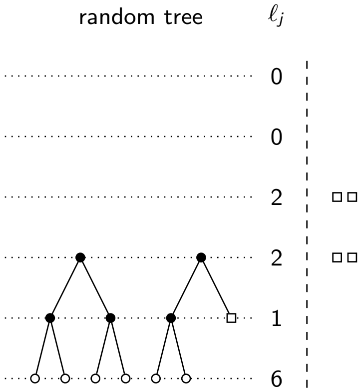
2: T = sequence of $\ell_h/2$ nodes, each one pointing to 2 leaves

3: **for** i from $h-1$ to 0 **do**

4: $T' =$ shuffling T with ℓ_i leaves

5: T = sequence of $|T'|/2$ nodes, with the ith node pointing to nodes $2i$ and $2i+1$ of T'

6: **return** the first node of T



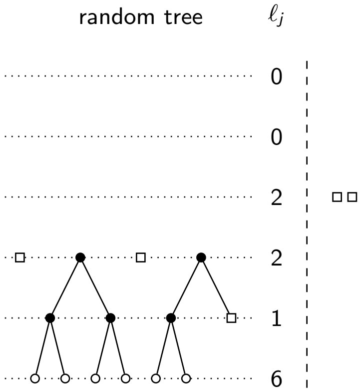
The algorithm is optimal in time, space, and random bit consumption.

Sampling trees with a given profile

Input: $L = (\ell_0, \ldots, \ell_h)$

Output: uniform tree with profile L

```
function UNIFORM_TREE(L)
    T = sequence of ℓ_h/2 nodes,
    each one pointing to 2 leaves
    for i from h − 1 to 0 do
        T′ = shuffling T with ℓ_i leaves
        T = sequence of |T′|/2 nodes,
        with the ith node pointing
        to nodes 2i and 2i + 1 of T′
    return the first node of T
```



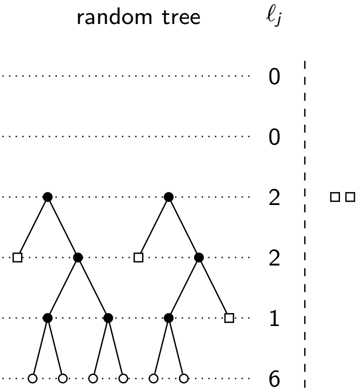
The algorithm is optimal in time, space, and random bit consumption.

Sampling trees with a given profile

Input: $L = (\ell_0, \dots, \ell_h)$

Output: uniform tree with profile L

function UNIFORM_TREE(L)
  $T =$ sequence of $\ell_h/2$ nodes, each one pointing to 2 leaves
  for i from $h-1$ to 0 **do**
    $T' =$ shuffling T with ℓ_i leaves
    $T =$ sequence of $|T'|/2$ nodes, with the ith node pointing to nodes $2i$ and $2i+1$ of T'
  return the first node of T



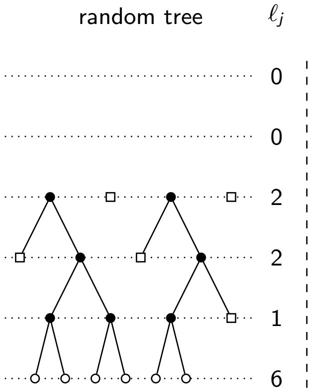
The algorithm is optimal in time, space, and random bit consumption.

Sampling trees with a given profile

Input: $L = (\ell_0, \ldots, \ell_h)$

Output: uniform tree with profile L

function $\mathrm{UNIFORM\_TREE}(L)$
  T = sequence of $\ell_h/2$ nodes, each one pointing to 2 leaves
  for i from $h-1$ to 0 **do**
    T' = shuffling T with ℓ_i leaves
    T = sequence of $|T'|/2$ nodes, with the ith node pointing to nodes $2i$ and $2i+1$ of T'
  return the first node of T



The algorithm is optimal in time, space, and random bit consumption.

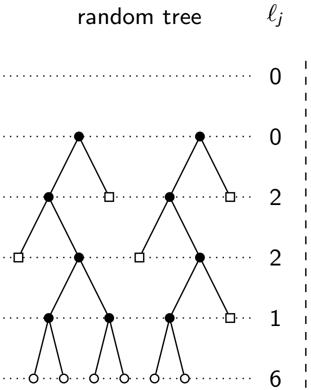
Olivier Bodini[1], Antoine Genitrini[2], Khaydar Nurligareev[2] [-2em] (1) LAGA, USPN; (2) LIP6, Sorbonne University

Sampling trees with a given profile

Input: $L = (\ell_0, \dots, \ell_h)$

Output: uniform tree with profile L

function $\mathrm{UNIFORM\_TREE}(L)$
    $T =$ sequence of $\ell_h/2$ nodes, each one pointing to 2 leaves
    for i from $h-1$ to 0 **do**
        $T' =$ shuffling T with ℓ_i leaves
        $T =$ sequence of $|T'|/2$ nodes, with the ith node pointing to nodes $2i$ and $2i+1$ of T'
    return the first node of T



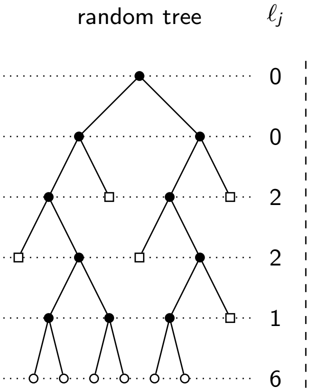
The algorithm is optimal in time, space, and random bit consumption.

Sampling trees with a given profile

Input: $L = (\ell_0, \ldots, \ell_h)$

Output: uniform tree with profile L

function UNIFORM_TREE(L)
    T = sequence of $\ell_h/2$ nodes,
    each one pointing to 2 leaves
    for i from $h-1$ to 0 **do**
        T' = shuffling T with ℓ_i leaves
        T = sequence of $|T'|/2$ nodes,
        with the ith node pointing
        to nodes $2i$ and $2i+1$ of T'
    return the first node of T



The algorithm is optimal in time, space, and random bit consumption.

Conclusion

1. Studied objects:
   - growing binary trees.

2. Related objects:
   - Mandelbrot polynomials,
   - meta-Fibonacci sequences.

3. Results:
   - relations for generating functions,
   - limits for anchor distributions,
   - behavior of the maximal number of anchors,
   - limit forms of non-zero domains,
   - uniform sampling of trees with a given profile.

Conclusion

1. Studied objects:
   - growing binary trees.

2. Related objects:
   - Mandelbrot polynomials,
   - meta-Fibonacci sequences.

3. Results:
   - relations for generating functions,
   - limits for anchor distributions,
   - behavior of the maximal number of anchors,
   - limit forms of non-zero domains,
   - uniform sampling of trees with a given profile.

Thank you for your attention!

Literature I

[26a] Olivier Bodini, Antoine Genitrini, Khaydar Nurligareev
Growing binary trees
arXiv: 2603.25972, 2026.

[26b] Olivier Bodini, Antoine Genitrini, Khaydar Nurligareev
The distribution of the deepest leaves in binary trees
arXiv: 2605.12821, 2026.

[CCC21] Neil J. Calkin, Eunice Y. S. Chan, Robert M. Corless
Some Facts and Conjectures about Mandelbrot Polynomials
Maple Transactions, 2021.

Literature II

[CT11] T. Cover, J. Thomas
Elements of information theory
Hoboken, NJ: John Wiley & Sons, 2006.

[FO82] P. Flajolet, A. Odlyzko
The average height of binary trees and other simple trees.
Journal of Computer and System Sciences, 25 (2), 1982, pp. 171–213.

[T92] Stephen M. Tanny
A well-behaved cousin of the Hofstadter sequence
Discrete Mathematics, 105 (1-3), 1992, pp. 227–239.