Compacted binary trees and minimal automata admit stretched exponentials

CanaDAM 2021 Online 05/2021

Michael Wallner

joint work with Andrew Elvey Price and Wenjie Fang

https://dmg.tuwien.ac.at/mwallner

Institute of Discrete Mathematics and Geometry, TU Wien, Austria

(Austrian Science Fund (FWF): J 4162 and P 34142)

Compacted binary trees admit a stretched exponential, JCTA, Vol. 177(105306), Jan. 2021; ArXiv:1908.11181

Asymptotics of minimal deterministic finite automata recognizing a finite binary language, LIPIcs, AofA 2020

I have an open PhD position to offer in my project "Stretched exponentials and beyond". Feel free to contact me if you are interested!

Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.

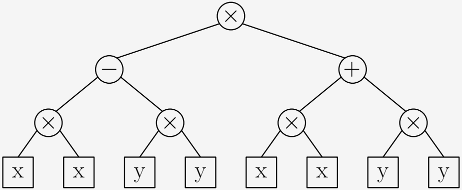
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



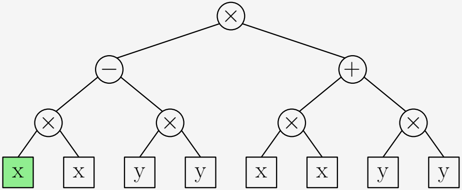
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



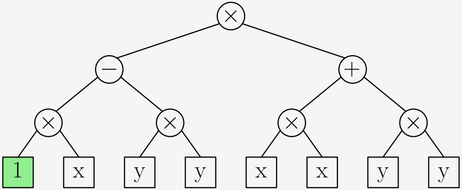
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$

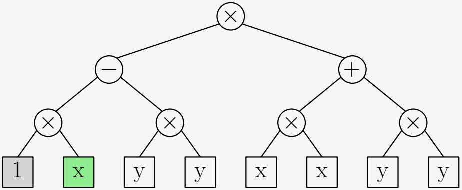
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$

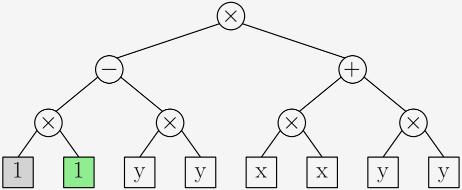
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$

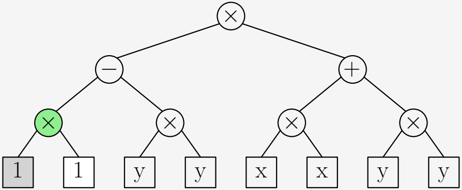
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$

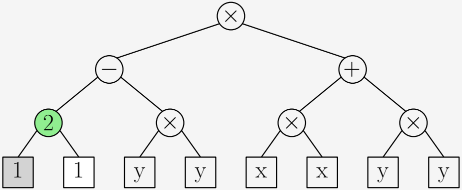
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$

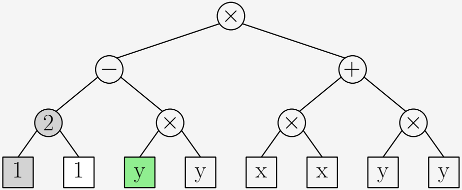
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

`(* (- (* x x) (* y y)) (+ (* x x) (* y y)))`

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1))$

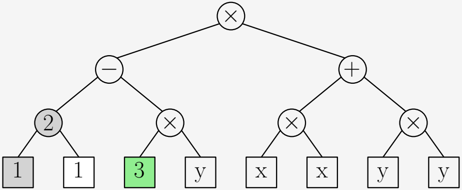
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0))$

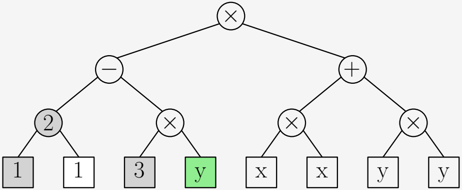
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$

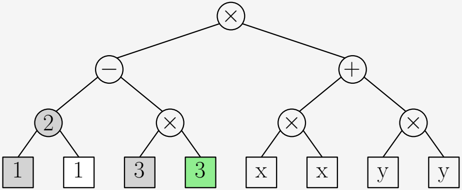
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1,(x,0,0)),\quad (2,(\times,1,1)),\quad (3,(y,0,0))$

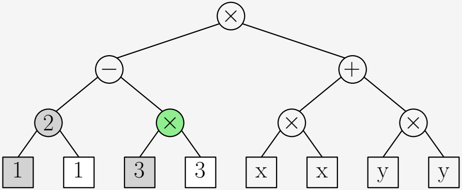
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$

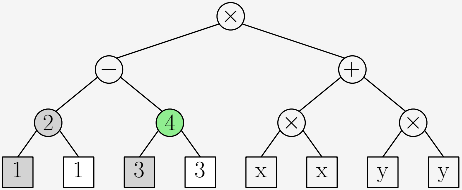
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3))$

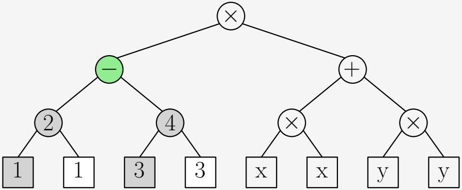
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3))$

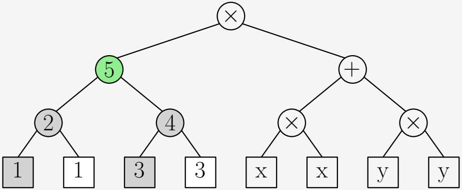
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$

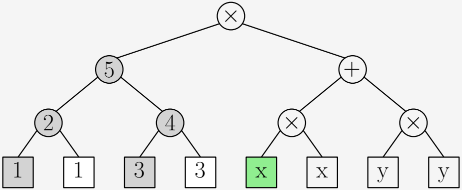
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1,(x,0,0)),\quad (2,(\times,1,1)),\quad (3,(y,0,0)),\quad (4,(\times,3,3)),\quad (5,(-,2,4))$

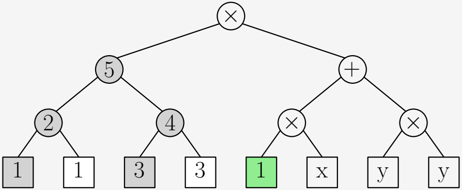
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1,(x,0,0)),\quad (2,(\times,1,1)),\quad (3,(y,0,0)),\quad (4,(\times,3,3)),\quad (5,(-,2,4))$

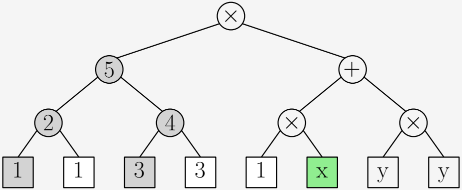
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3)), \quad (5, (-, 2, 4))$

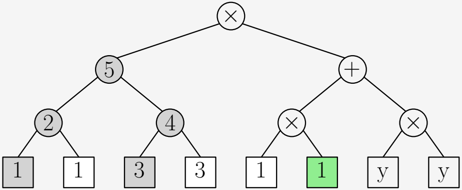
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$

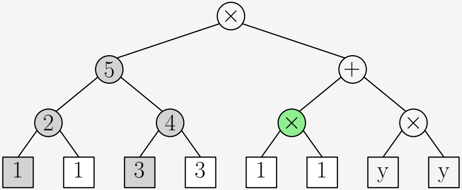
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3)), \quad (5, (-, 2, 4))$

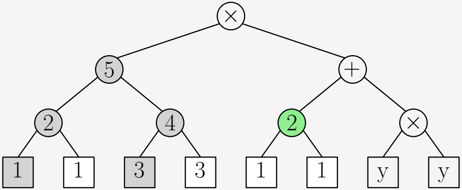
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$

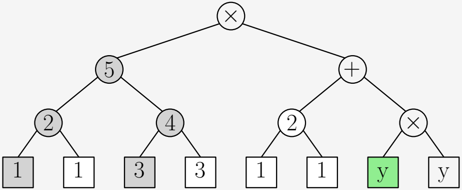
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3)), \quad (5, (-, 2, 4))$

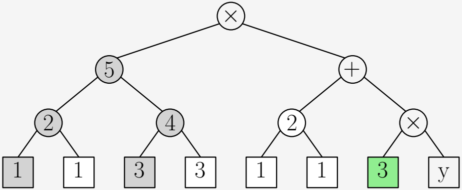
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$

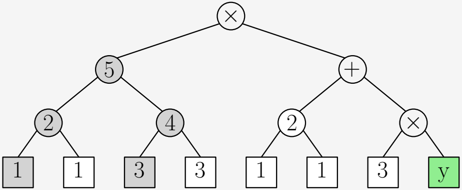
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3)), \quad (5, (-, 2, 4))$

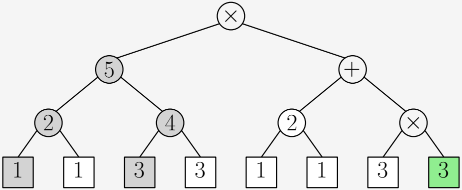
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3)), \quad (5, (-, 2, 4))$

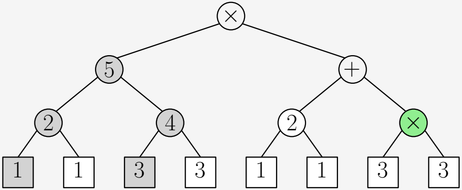
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0)), \quad (2, (\times, 1, 1)), \quad (3, (y, 0, 0)), \quad (4, (\times, 3, 3)), \quad (5, (-, 2, 4))$

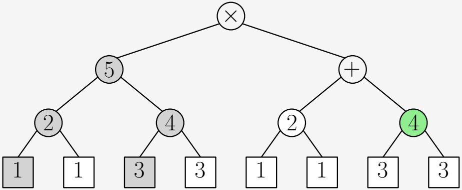
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$

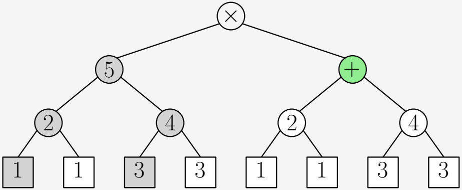
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$

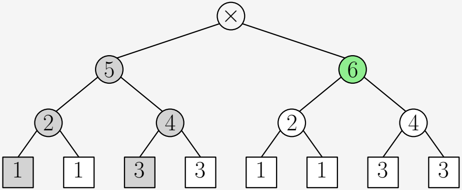
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$, $(6, (+, 2, 4))$

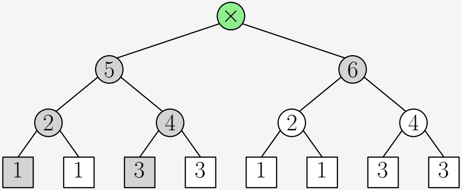
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1, (x, 0, 0))$, $(2, (\times, 1, 1))$, $(3, (y, 0, 0))$, $(4, (\times, 3, 3))$, $(5, (-, 2, 4))$, $(6, (+, 2, 4))$

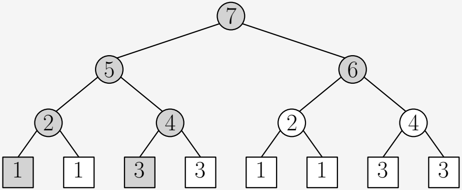
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1,(x,0,0))$, $(2,(\times,1,1))$, $(3,(y,0,0))$, $(4,(\times,3,3))$, $(5,(-,2,4))$,
$(6,(+,2,4))$, $(7,(\times,5,6))$

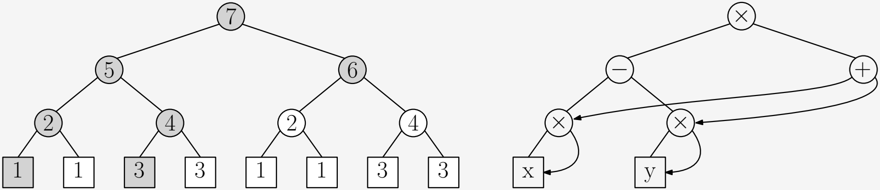
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1,(x,0,0))$, $(2,(\times,1,1))$, $(3,(y,0,0))$, $(4,(\times,3,3))$, $(5,(-,2,4))$, $(6,(+,2,4))$, $(7,(\times,5,6))$

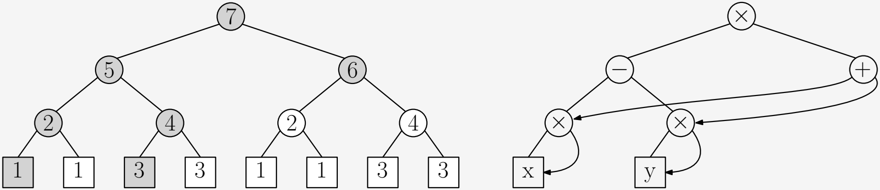
Motivation: Efficiently store redundant information

Example

Consider the labeled tree necessary to store the arithmetic expression

```
(* (- (* x x) (* y y)) (+ (* x x) (* y y)))
```

which represents $(x^2 - y^2)(x^2 + y^2)$.



$(1,(x,0,0))$, $(2,(\times,1,1))$, $(3,(y,0,0))$, $(4,(\times,3,3))$, $(5,(-,2,4))$, $(6,(+,2,4))$, $(7,(\times,5,6))$

Definition

Compacted tree is the directed acyclic graph computed by this procedure.

Compacted trees

- Efficient algorithm to compute compacted tree: expected time $\mathcal{O}(n)$
- Analyzed by [Flajolet, Sipala, Steyaert 1990]: A tree of size n has a compacted form of expected size

$$C\frac{n}{\sqrt{\log n}},$$

where C is explicit related to the type of trees and the statistical model.

- Applications:
  - **XML-Compression** [Bousquet-Mélou, Lohrey, Maneth, Noeth 2015]
  - **Compilers** [Aho, Sethi, Ullman 1986]
  - **LISP** [Goto 1974]
  - **Data storage** [Meinel, Theobald 1998], [Knuth 1968], etc.

Reverse question

How many compacted trees of (compacted) size n exist?

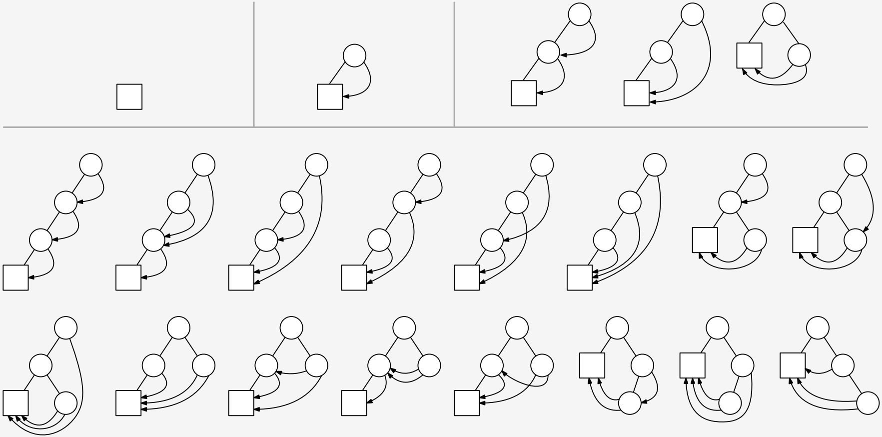
Compacted (unlabeled binary) trees

- Size: number of internal nodes
- c_n: number of compacted trees of size n

  $(c_n)_{n \geq 0} = (1, 1, 3, 15, 111, 1119, 14487, \dots)$
- Important: Subtrees are unique!

Simple bounds

$$n! \leq c_n \leq \frac{1}{n+1}\binom{2n}{n} n!$$



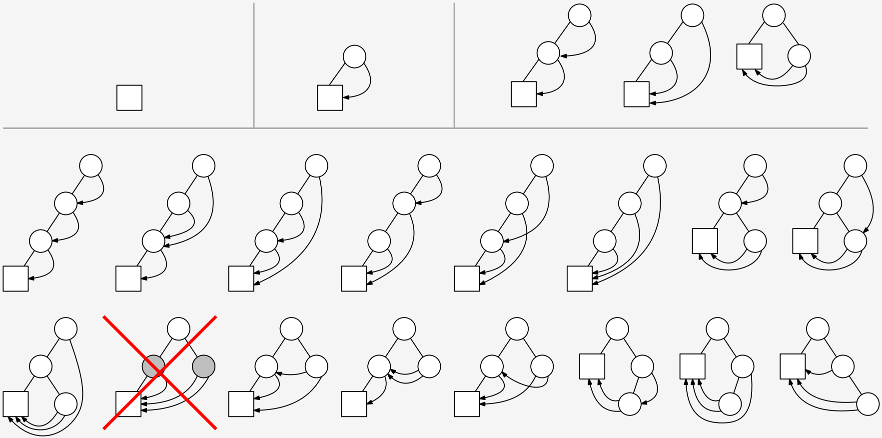
Compacted (unlabeled binary) trees

- Size: number of internal nodes
- c_n: number of compacted trees of size n

  $(c_n)_{n \geq 0} = (1, 1, 3, 15, 111, 1119, 14487, \dots)$
- Important: Subtrees are unique!

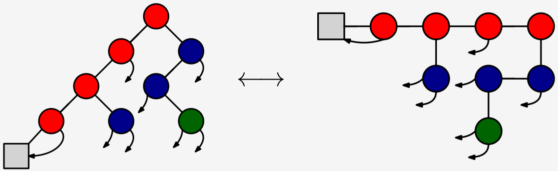
Simple bounds

$$n! \leq c_n \leq \frac{1}{n+1}\binom{2n}{n} n!$$



Bounded right height (Previous work)

The **right height** of a binary tree is the maximal number of right children on any path from the root to a leaf (not going through pointers).



Theorem [Genitrini, Gittenberger, Kauers, W 2020]

The number $c_{k,n}$ of compacted trees with right height at most k is for $n \to \infty$ asymptotically equivalent to

$$c_{k,n} \sim \kappa_k n! \left(4 \cos\left(\frac{\pi}{k+3}\right)^2\right)^n n^{-\frac{k}{2} - \frac{1}{k+3} - \left(\frac{1}{4} - \frac{1}{k+3}\right)\cos\left(\frac{\pi}{k+3}\right)^{-2}},$$

where $\kappa_k \in \mathbb{R} \setminus \{0\}$ is independent of n.

Main result compacted trees

A stretched exponential μ^{n^σ} appears!

Theorem [Elvey Price, Fang, W 2021]

The number of compacted binary trees satisfies for $n \to \infty$

$$c_n = \Theta\left(n!\, 4^n e^{3a_1 n^{1/3}} n^{3/4}\right),$$

with $a_1 \approx -2.338$: largest root of the Airy function $\mathrm{Ai}(x) = \frac{1}{\pi}\int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$.

Main result compacted trees

A stretched exponential μ^{n^σ} appears!

Theorem [Elvey Price, Fang, W 2021]

The number of compacted binary trees satisfies for $n \to \infty$

$$c_n = \Theta\left(n!\, 4^n e^{3a_1 n^{1/3}} n^{3/4}\right),$$

with $a_1 \approx -2.338$: largest root of the Airy function $\mathrm{Ai}(x) = \frac{1}{\pi}\int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$.

Conjecture

Experimentally we find

$$c_n \sim \gamma_c n! 4^n e^{3a_1 n^{1/3}} n^{3/4},$$

where

$$\gamma_c \approx 173.12670485.$$

Other appearances of stretched exponentials

Known exactly:

- Integer partitions:

$$\sim (4\sqrt{3})^{-1} e^{\pi(2n/3)^{1/2}} n^{-1}$$

- Pushed Dyck paths [Beaton, McKay 2014], [Guttmann 2015]:

$$\sim C_1 4^n e^{-3\left(\frac{\pi \log 2}{2}\right)^{2/3} n^{1/3}} n^{-5/6}$$

- Cogrowth sequence of a lamplighter group variant of $\mathbb{Z}_2 \wr \mathbb{Z}$ [Revelle 2003]:

$$\sim C_2 \mu^n e^{-3(\pi \log(2)/2)^{2/3} n^{1/3}} n^{1/6}$$

- Phylogenetic tree-child networks [Fuchs, Yu, Zhang 2020]:

$$\Theta\left(n^{2n} (12e^{-2})^n e^{a_1 (3n)^{1/3}} n^{-2/3} \right)$$

Conjectured:

- Permutations avoiding 1324 [Conway, Guttmann, Zinn-Justin 2018]:

$$\approx \mu^n e^{-cn^{1/2}}$$

- Pushed self avoiding walks [Beaton, Guttmann, Jensen, Lawler 2015]:

$$\approx \mu^n e^{-cn^{3/7}}$$

- and recently more and more appear in group theory, queuing theory, ...

Other appearances of stretched exponentials

Known exactly:

- Integer partitions:

$$\sim (4\sqrt{3})^{-1} e^{\pi(2n/3)^{1/2}} n^{-1}$$

- Pushed Dyck paths [Beaton, McKay 2014], [Guttmann 2015]:

$$\sim C_1 4^n e^{-3\left(\frac{\pi \log 2}{2}\right)^{2/3} n^{1/3}} n^{-5/6}$$

- Cogrowth sequence of a lamplighter group variant of $\mathbb{Z}_2 \wr \mathbb{Z}$ [Revelle 2003]:

$$\sim C_2 \mu^n e^{-3(\pi \log(2)/2)^{2/3} n^{1/3}} n^{1/6}$$

- Phylogenetic tree-child networks [Fuchs, Yu, Zhang 2020]:

$$\Theta\left(n^{2n}(12e^{-2})^n e^{a_1(3n)^{1/3}} n^{-2/3}\right)$$

Conjectured:

- Permutations avoiding 1324 [Conway, Guttmann, Zinn-Justin 2018]:

$$\approx \mu^n e^{-cn^{1/2}}$$

- Pushed self avoiding walks [Beaton, Guttmann, Jensen, Lawler 2015]:

$$\approx \mu^n e^{-cn^{3/7}}$$

- and recently more and more appear in group theory, queuing theory, . . .

Deterministic finite automata (DFA)

DFA on alphabet $\{a, b\}$

Graph with

- two outgoing edges from each node (state), labelled a and b
- An initial state q_0
- A set F of *final states* (coloured green).

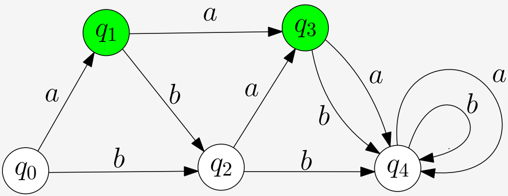


Figure: DFA

Deterministic finite automata (DFA)

DFA on alphabet $\{a, b\}$

Graph with

- two outgoing edges from each node (state), labelled a and b
- An initial state q_0
- A set F of *final states* (coloured green).

Properties

- **Language:** the set of accepted words
- **Minimal:** no DFA with fewer states accepts the same language
- **Acyclic:** no cycles (except loops at unique sink)

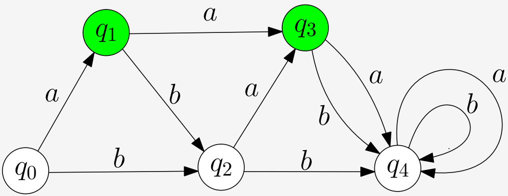


Figure: DFA, which is the minimal DFA recognizing the language $\{a, aa, ba, aba\}$.

Counting minimal acyclic DFAs

- Studied by Domaratzki, Kisman, Shallit, and Liskovets 2002–2006
- **Open problem:** Asymptotics
- Best bounds were out by an exponential factor

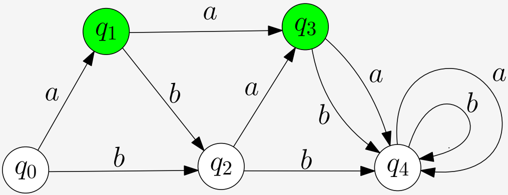


Figure: DFA, which is the minimal DFA recognizing the language $\{a, aa, ba, aba\}$.

Main result minimal DFAs

A stretched exponential μ^{n^σ} appears again!

Theorem [Elvey Price, Fang, W 2020]

The number m_n of minimal DFAs with $n+1$ states recognizing a finite binary language satisfies for $n \to \infty$

$$m_n = \Theta\left(n!\, 8^n e^{3a_1 n^{1/3}} n^{7/8}\right),$$

with $a_1 \approx -2.338$: largest root of the Airy function $\mathrm{Ai}(x) = \frac{1}{\pi}\int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$.

Main result minimal DFAs

A stretched exponential μ^{n^σ} appears again!

Theorem [Elvey Price, Fang, W 2020]

The number m_n of minimal DFAs with $n+1$ states recognizing a finite binary language satisfies for $n \to \infty$

$$m_n = \Theta\left(n!\, 8^n e^{3a_1 n^{1/3}} n^{7/8}\right),$$

with $a_1 \approx -2.338$: largest root of the Airy function $\mathrm{Ai}(x) = \frac{1}{\pi}\int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$.

Conjecture

Experimentally we find

$$m_n \sim \gamma n! 8^n e^{3a_1 n^{1/3}} n^{7/8},$$

where

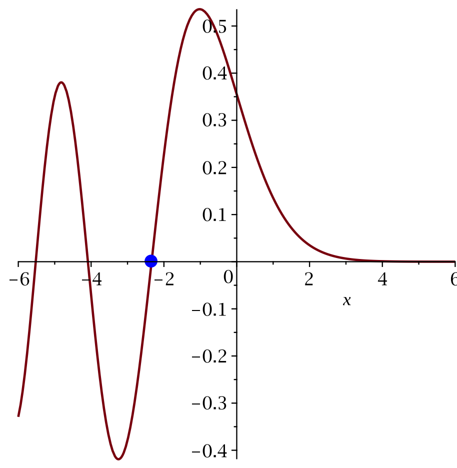
$$\gamma \approx 76.438160702.$$

What is the Airy function?

Properties

- $\mathrm{Ai}(x) = \frac{1}{\pi} \int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$
- Largest root $a_1 \approx -2.338$
- $\lim_{x\to\infty} \mathrm{Ai}(x) = 0$
- Also defined by $\mathrm{Ai}''(x) = x\mathrm{Ai}(x)$

- [Banderier, Flajolet, Schaeffer, Soria 2001]: Random Maps
- [Flajolet, Louchard 2001]: Brownian excursion area



How to prove this?

Theorem [Elvey Price, Fang, W 2020]

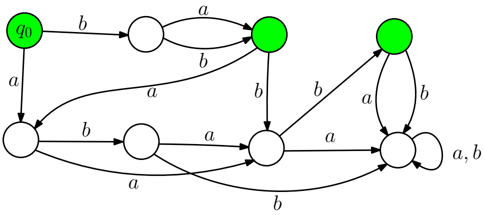
The number m_n of minimal DFAs with $n+1$ states recognizing a finite binary language satisfies for $n \to \infty$

$$m_n = \Theta\left(n!\, 8^n e^{3a_1 n^{1/3}} n^{7/8}\right),$$

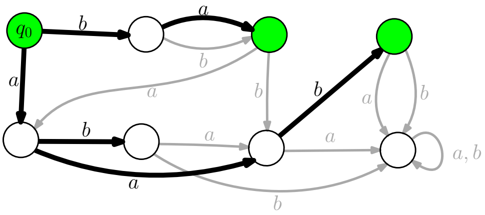
with $a_1 \approx -2.338$: largest root of the Airy function $\mathrm{Ai}(x) = \frac{1}{\pi}\int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$.

1. Bijection to decorated Dyck paths
2. Two-parameter recurrence relation for decorated Dyck paths
3. Heuristic analysis of recurrence
4. Inductive proof of asymptotically tight bounds using heuristics

Bijection to decorated paths

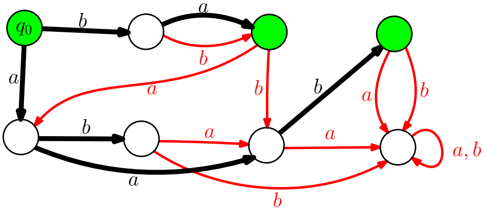


Bijection to decorated paths



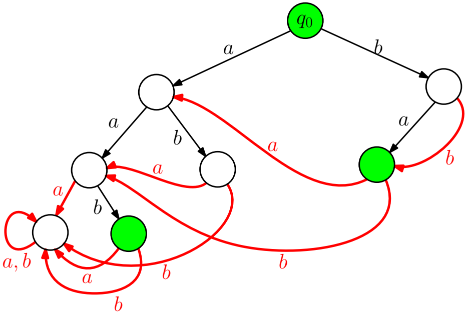
- Highlight spanning tree given by depth first search (ignoring the sink)
- I.e., black path to each vertex is first in lexicographic order
- Colour other edges red
- Draw as a binary tree with *a* edges pointing left and *b* edges pointing right

Bijection to decorated paths



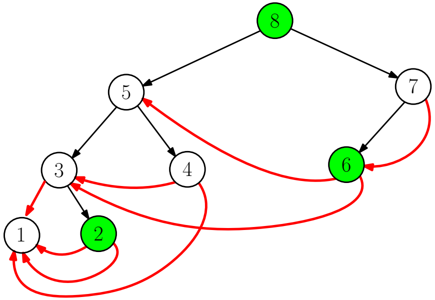
- Highlight spanning tree given by depth first search (ignoring the sink)
- I.e., black path to each vertex is first in lexicographic order
- Colour other edges red
- Draw as a binary tree with *a* edges pointing left and *b* edges pointing right

Bijection to decorated paths



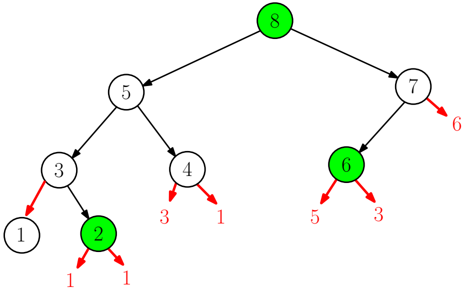
- Highlight spanning tree given by depth first search (ignoring the sink)
- I.e., black path to each vertex is first in lexicographic order
- Colour other edges red
- Draw as a binary tree with *a* edges pointing left and *b* edges pointing right

Bijection to decorated paths



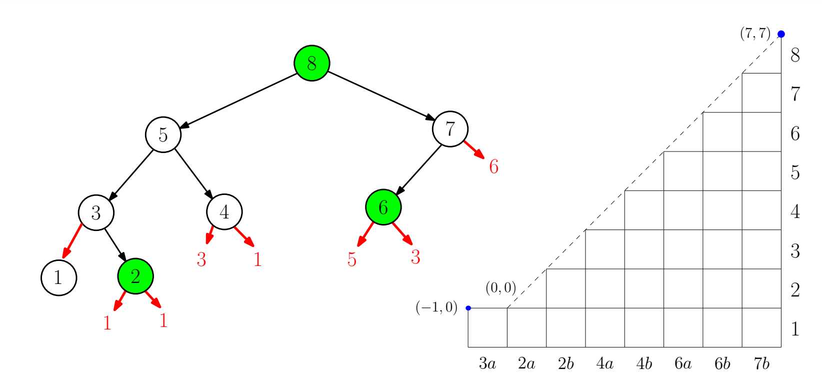
- Label nodes in post-order. By construction red edges point from a larger number to a smaller number
- → Label pointers

Bijection to decorated paths

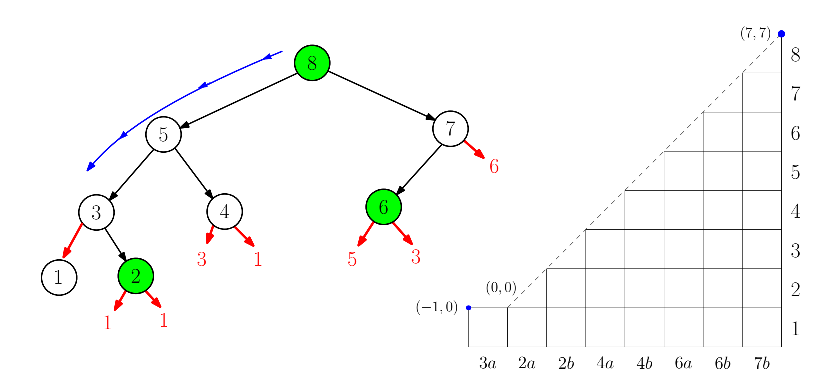


- Label nodes in post-order. By construction red edges point from a larger number to a smaller number
- $\rightarrow$ Label pointers

Bijection to decorated paths



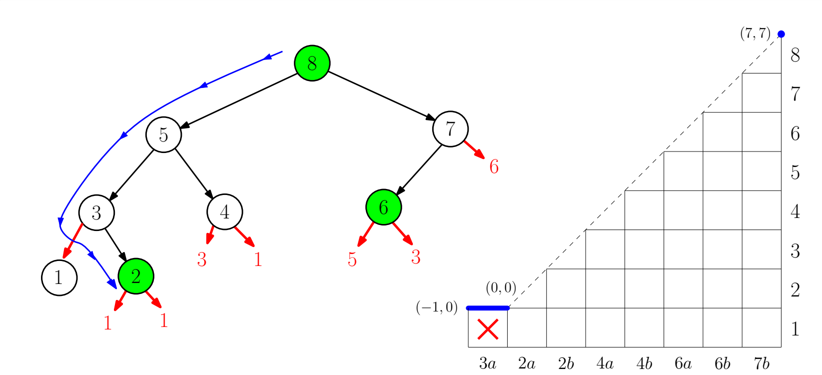
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

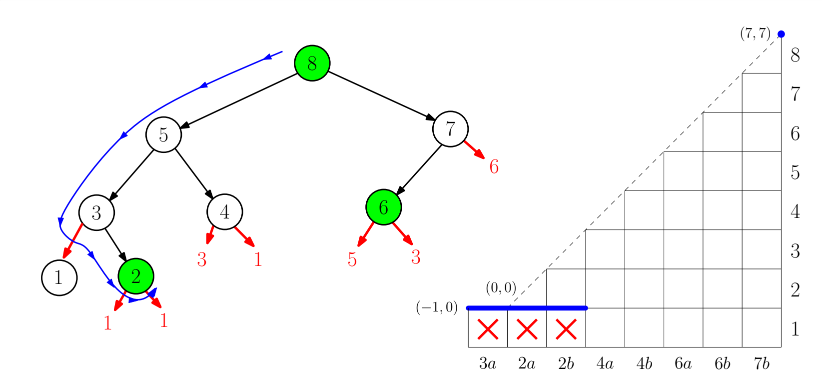
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

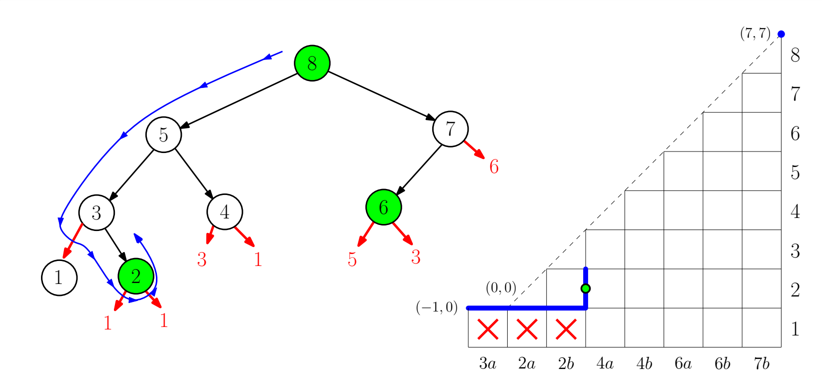
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

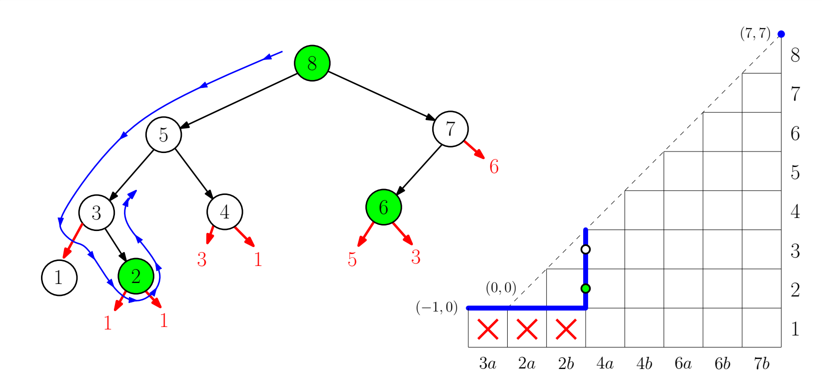
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

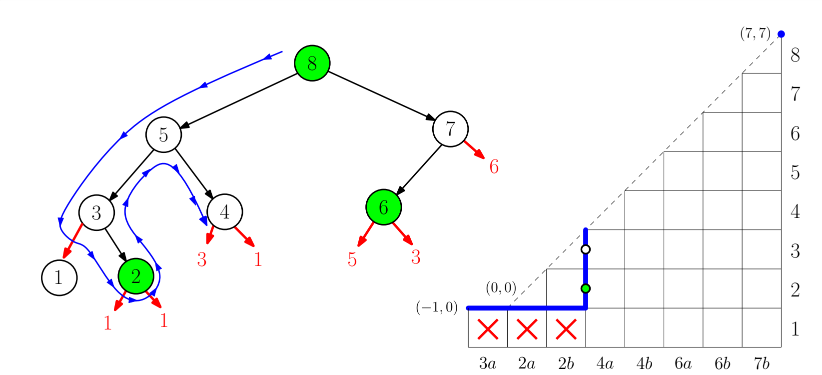
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

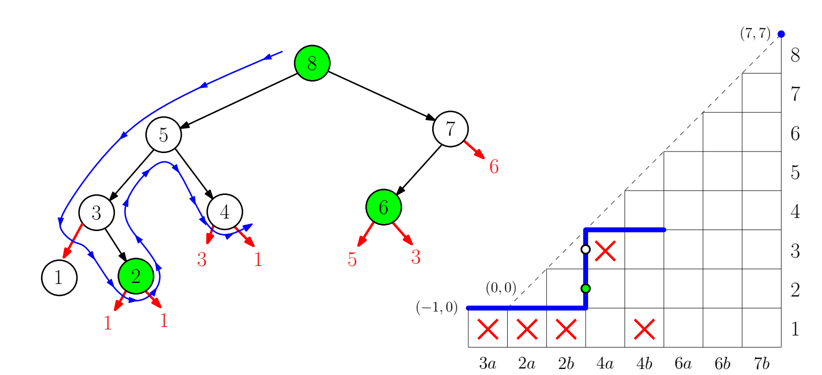
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

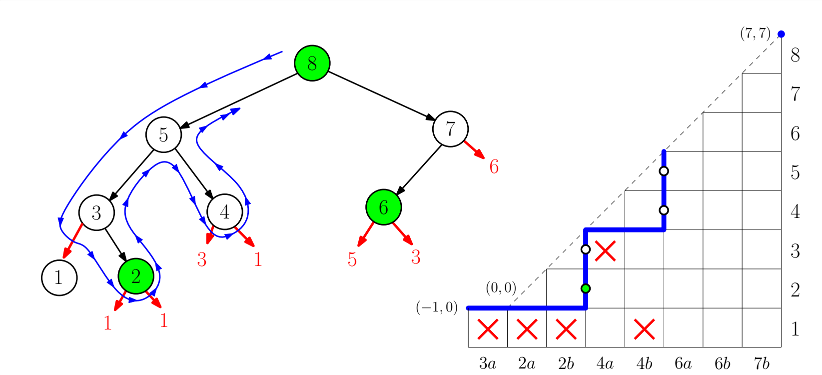
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

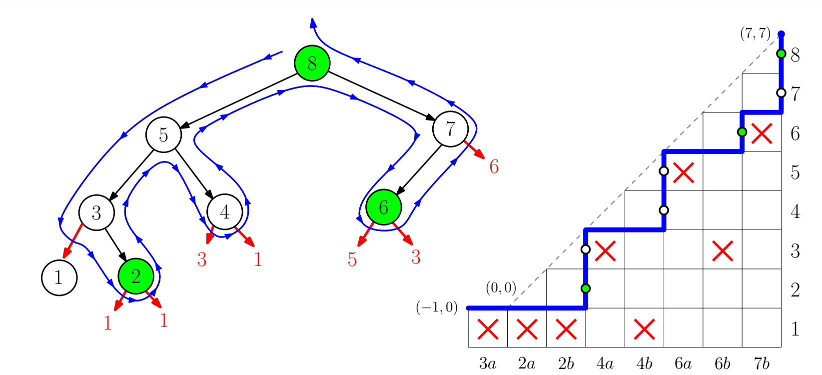
Bijection to decorated paths



When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

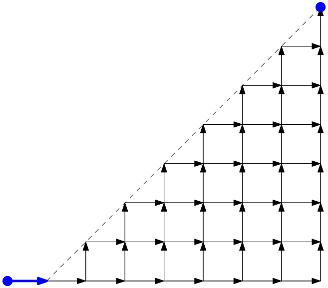
Bijection to decorated paths

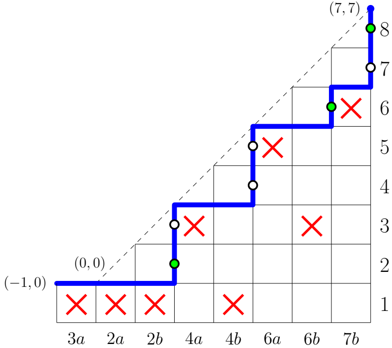


When the tree traversal...

- goes up: add up step with color matching the corresponding node.
- passes a pointer:
  - add horizontal step
  - mark box corresponding to pointer label

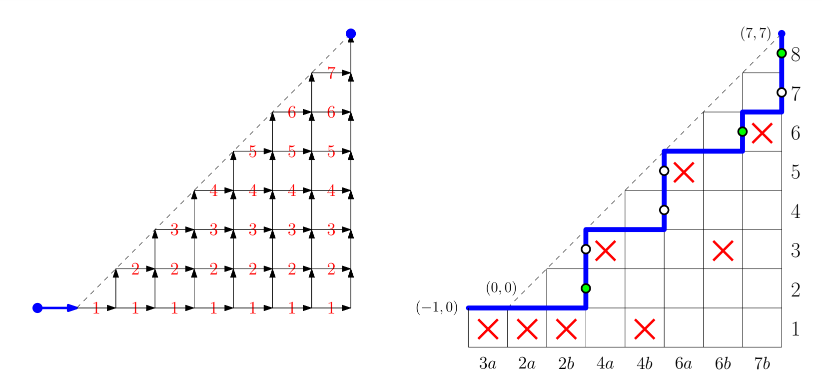
Decorated paths





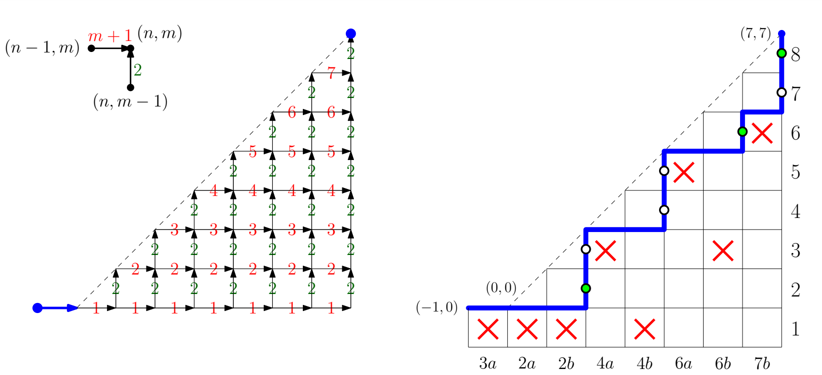
- Path starts at $(-1, 0)$ and ends at (n, n)
- Path stays below diagonal (after first step)
- One box is marked below each horizontal step
- Each vertical step is colored white or green

Decorated paths



- Path starts at $(-1, 0)$ and ends at (n, n)
- Path stays below diagonal (after first step)
- One box is marked below each horizontal step
- Each vertical step is colored white or green

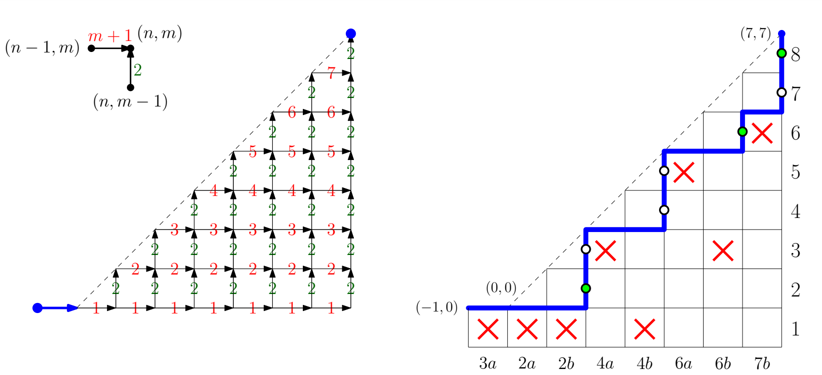
Decorated paths



- Path starts at $(-1,0)$ and ends at (n,n)
- Path stays below diagonal (after first step)
- One box is marked below each horizontal step
- Each vertical step is colored white or green

By the bijection: The number of these paths is the number d_n of acyclic DFAs with $n+1$ nodes.

Decorated paths



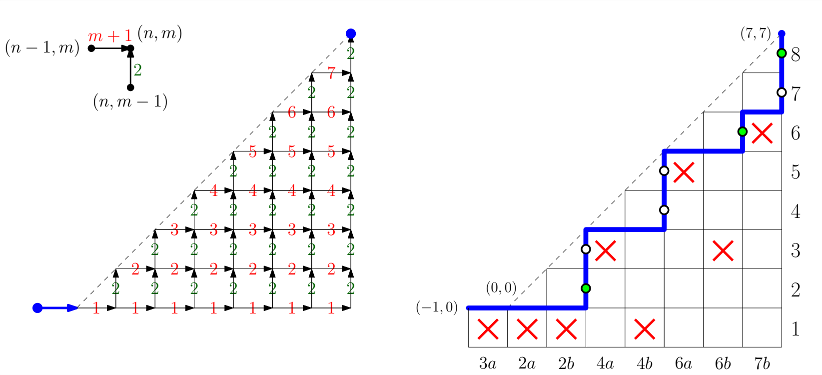
Recurrence: Denote by $a_{n,m}$ the number of paths ending at (n, m).

$$a_{n,m} = 2a_{n,m-1} + (m+1)a_{n-1,m}, \qquad \text{for } n \geq m$$

$$a_{-1,0} = 1.$$

By the bijection: $d_n = a_{n,n}$ is the number of acyclic DFAs with $n+1$ nodes.

Decorated paths



Recurrence: Denote by $a_{n,m}$ the number of paths ending at (n, m).

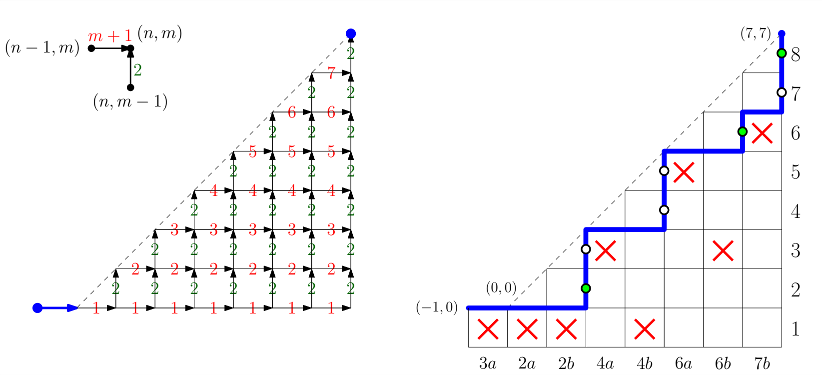
$$a_{n,m} = 2a_{n,m-1} + (m+1)a_{n-1,m}, \qquad \text{for } n \geq m$$

$$a_{-1,0} = 1.$$

By the bijection: $d_n = a_{n,n}$ is the number of acyclic DFAs with $n+1$ nodes.

What about minimality?

Recurrence for minimal DFAs



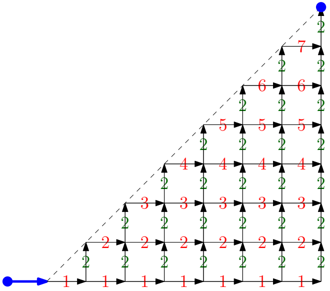
Recurrence: Denote by $b_{n,m}$ the number of paths ending at (n, m).

$$b_{n,m} = 2b_{n,m-1} + (m+1)b_{n-1,m} - mb_{n-2,m-1}, \qquad \text{for } n \geq m,$$

$$b_{-1,0} = 1.$$

Now: $m_n = b_{n,n}$ is the number of minimal acyclic DFAs with $n+1$ nodes.

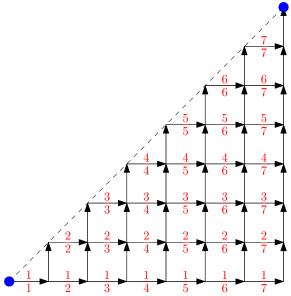
Transforming the recurrence for minimal DFAs



Transformation:

$$e_{n+m,n-m} = \frac{1}{n!2^m} b_{n,m}.$$

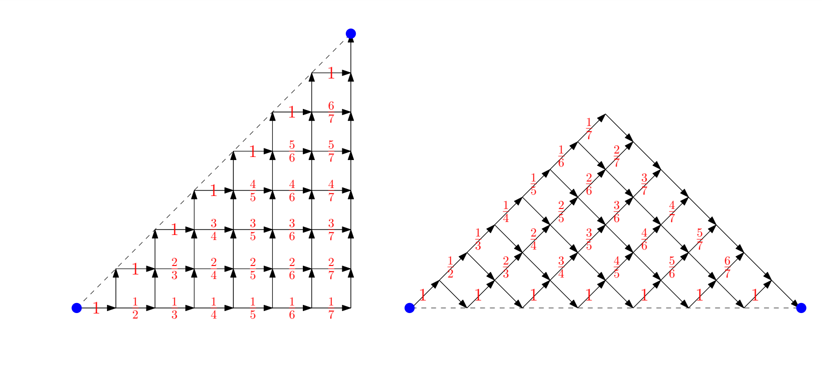
Transforming the recurrence for minimal DFAs



Transformation:

$$e_{n+m,n-m} = \frac{1}{n!2^m} b_{n,m}.$$

Transforming the recurrence for minimal DFAs



Transformation: $$e_{n+m,n-m} = \frac{1}{n!2^m} b_{n,m}.$$

New recurrence:

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}.$$

Now: $m_n = n! 2^n e_{2n,0}$.

Heuristics

We want to understand $e_{n,m}$ for large (fixed) n.

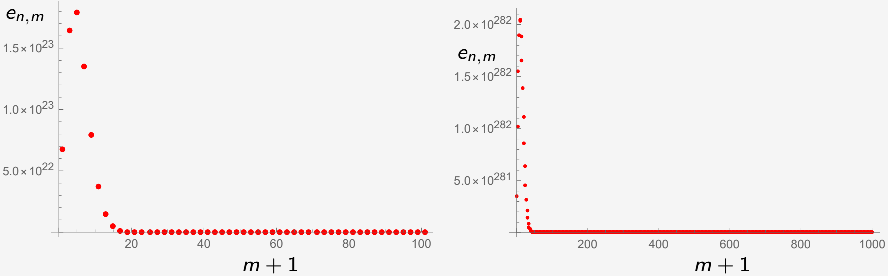


Figure: Plots of $e_{n,m}$ against $m+1$. **Left:** $n = 100$, **Right:** $n = 1000$.

- Let's zoom in to the left (small m) where interesting things are happening.
- It seems to be converging to something...

Guess: $e_{n,m} \approx h(n) f\left(\frac{m+1}{g(n)}\right)$. Moreover, we guess $g(n) = \sqrt[3]{n}$.

Heuristics

We want to understand $e_{n,m}$ for large (fixed) n.

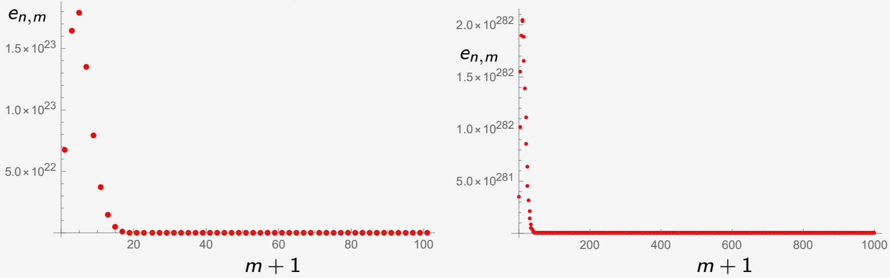


Figure: Plots of $e_{n,m}$ against $m+1$. **Left:** $n = 100$, **Right:** $n = 1000$.

- Let's zoom in to the left (small m) where interesting things are happening.
- It seems to be converging to something...

Guess: $e_{n,m} \approx h(n) f\left(\frac{m+1}{g(n)}\right)$. Moreover, we guess $g(n) = \sqrt[3]{n}$.

Heuristics

We want to understand $e_{n,m}$ for large (fixed) n.

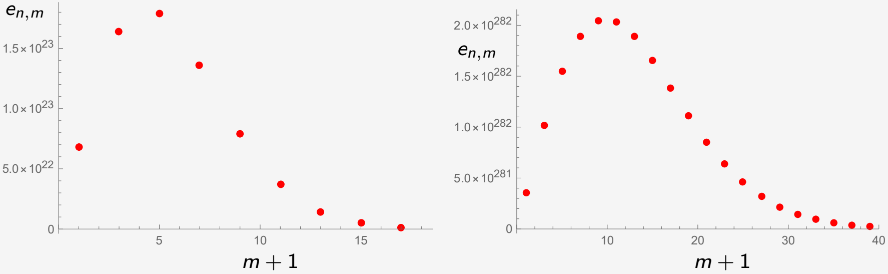


Figure: Plots of $e_{n,m}$ against $m+1$. **Left:** $n = 100$, **Right:** $n = 1000$.

- Let's zoom in to the left (small m) where interesting things are happening.
- It seems to be converging to something...

Guess: $e_{n,m} \approx h(n) f\left(\frac{m+1}{g(n)}\right)$. Moreover, we guess $g(n) = \sqrt[3]{n}$.

Heuristics

We want to understand $e_{n,m}$ for large (fixed) n.

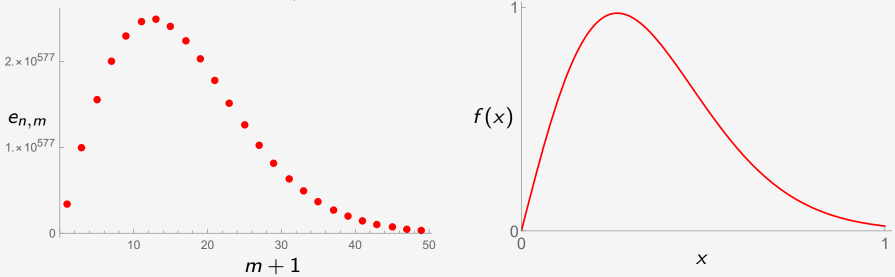


Figure: **Left:** Plot of $e_{n,m}$ against $m+1$ for $n = 2000$. **Right:** Limiting function $f(x)$.

- Let's zoom in to the left (small m) where interesting things are happening.
- It seems to be converging to something...

Guess: $e_{n,m} \approx h(n) f\left(\frac{m+1}{g(n)}\right)$. Moreover, we guess $g(n) = \sqrt[3]{n}$.

Heuristics

We want to understand $e_{n,m}$ for large (fixed) n.

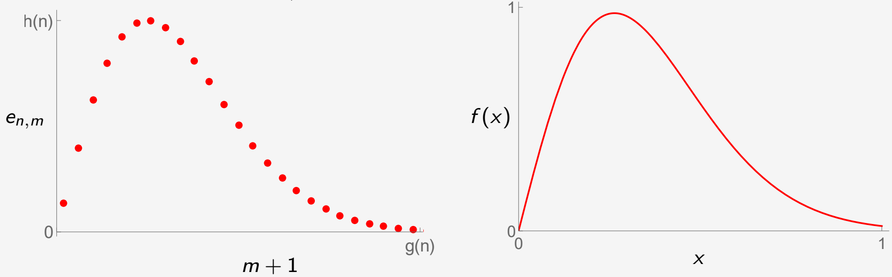


Figure: **Left:** Plot of $e_{n,m}$ against $m+1$ for $n = 2000$. **Right:** Limiting function $f(x)$.

- Let's zoom in to the left (small m) where interesting things are happening.
- It seems to be converging to something...

Guess: $e_{n,m} \approx h(n) f\left(\frac{m+1}{g(n)}\right)$. Moreover, we guess $g(n) = \sqrt[3]{n}$.

Heuristics

We want to understand $e_{n,m}$ for large (fixed) n.

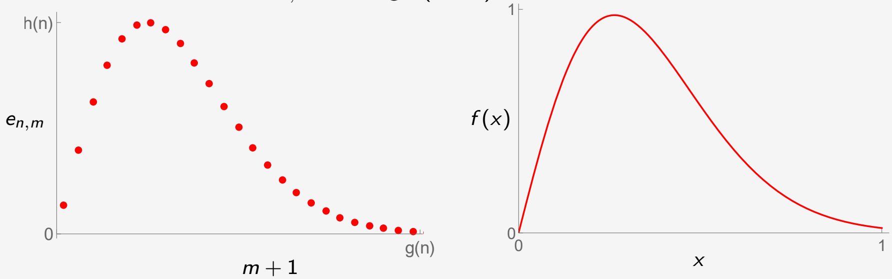


Figure: **Left:** Plot of $e_{n,m}$ against $m+1$ for $n = 2000$. **Right:** Limiting function $f(x)$.

- Let's zoom in to the left (small m) where interesting things are happening.
- It seems to be converging to something...

Guess: $e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right)$. Moreover, we guess $g(n) = \sqrt[3]{n}$.

Heuristic analysis of weighted paths

Recurrence

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}.$$

- **Ansatz (a):** $e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right).$

Heuristic analysis of weighted paths

Recurrence

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}.$$

- **Ansatz (a):** $e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right)$.
- Substitute into recurrence and set $m = \kappa\sqrt[3]{n} - 1$:

$$s_n := \frac{h(n)}{h(n-1)} \approx 2 + \frac{f''(\kappa) - 2\kappa f(\kappa)}{f(\kappa)} n^{-2/3} + O(n^{-1})$$

Heuristic analysis of weighted paths

Recurrence

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}.$$

- **Ansatz (a):** $e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right)$.
- Substitute into recurrence and set $m = \kappa\sqrt[3]{n} - 1$:

$$s_n := \frac{h(n)}{h(n-1)} \approx 2 + \frac{f''(\kappa) - 2\kappa f(\kappa)}{f(\kappa)} n^{-2/3} + O(n^{-1})$$

- **Ansatz (b):**

$$s_n = 2 + c n^{-2/3} + O(n^{-1}) \qquad \Rightarrow \qquad h(n) \approx 2^n e^{\frac{3c}{2} n^{1/3}}$$

Heuristic analysis of weighted paths

Recurrence

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}.$$

- **Ansatz (a):** $e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right)$.
- Substitute into recurrence and set $m = \kappa\sqrt[3]{n} - 1$:

$$s_n := \frac{h(n)}{h(n-1)} \approx 2 + \frac{f''(\kappa) - 2\kappa f(\kappa)}{f(\kappa)} n^{-2/3} + O(n^{-1})$$

- **Ansatz (b):**

$$s_n = 2 + c n^{-2/3} + O(n^{-1}) \qquad \Rightarrow \qquad h(n) \approx 2^n e^{\frac{3c}{2} n^{1/3}}$$

- Solution

$$f''(\kappa) = (2\kappa + c) f(\kappa) \qquad \Rightarrow \qquad f(\kappa) = \mathrm{Ai}(2^{-2/3}(2\kappa + c))$$

Where c is constant and Ai is the Airy function.

Heuristic analysis of weighted paths

Recurrence

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}.$$

- **Ansatz (a):** $e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right)$.
- Substitute into recurrence and set $m = \kappa\sqrt[3]{n} - 1$:

$$s_n := \frac{h(n)}{h(n-1)} \approx 2 + \frac{f''(\kappa) - 2\kappa f(\kappa)}{f(\kappa)} n^{-2/3} + O(n^{-1})$$

- **Ansatz (b):**

$$s_n = 2 + c n^{-2/3} + O(n^{-1}) \qquad \Rightarrow \qquad h(n) \approx 2^n e^{\frac{3c}{2} n^{1/3}}$$

- Solution

$$f''(\kappa) = (2\kappa + c) f(\kappa) \qquad \Rightarrow \qquad f(\kappa) = \mathrm{Ai}(2^{-2/3}(2\kappa + c))$$

  Where c is constant and Ai is the Airy function.
- Boundary condition $e_{n,-1} = 0$. Then $f(0) = 0$ implies $c = 2^{2/3} a_1$, where $a_1 \approx -2.338$ satisfies $\mathrm{Ai}(a_1) = 0$.

Refined heuristic analysis

1. **Ansatz of order 1:**

$$e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right),$$
$$s_n = 2 + cn^{-2/3} + O(n^{-1}).$$

yields estimates $c = 2^{2/3} a_1$ such that

$$h(n) \approx 2^n e^{3a_1(n/2)^{1/3}} \quad \text{and} \quad f(\kappa) = \mathrm{Ai}(2^{1/3}\kappa + a_1).$$

2. **Ansatz of order 2:**

$$e_{n,m} \approx h(n)\left(f_0\left(\frac{m+1}{\sqrt[3]{n}}\right) + n^{-1/3} f_1\left(\frac{m+1}{\sqrt[3]{n}}\right)\right),$$
$$s_n = 2 + cn^{-2/3} + dn^{-1} + O(n^{-4/3}).$$

yields estimates $d = 29/12$ such that

$$h(n) \sim const \cdot 2^n e^{3a_1(n/2)^{1/3}} n^{29/24} \quad \text{and} \quad f_0(\kappa) = \mathrm{Ai}(2^{1/3}\kappa + a_1).$$

This way we conjecture the asymptotic form for acyclic minimal DFAs:

$$m_n = 2^n n! e_{2n,0} = \Theta\left(n! 8^n e^{3a_1 n^{1/3}} n^{7/8}\right).$$

Refined heuristic analysis

1. **Ansatz of order 1:**

$$e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right),$$
$$s_n = 2 + cn^{-2/3} + O(n^{-1}).$$

yields estimates $c = 2^{2/3} a_1$ such that

$$h(n) \approx 2^n e^{3a_1 (n/2)^{1/3}} \quad \text{and} \quad f(\kappa) = \mathrm{Ai}(2^{1/3}\kappa + a_1).$$

2. **Ansatz of order 2:**

$$e_{n,m} \approx h(n)\left(f_0\left(\frac{m+1}{\sqrt[3]{n}}\right) + n^{-1/3} f_1\left(\frac{m+1}{\sqrt[3]{n}}\right)\right),$$
$$s_n = 2 + cn^{-2/3} + dn^{-1} + O(n^{-4/3}).$$

yields estimates $d = 29/12$ such that

$$h(n) \sim const \cdot 2^n e^{3a_1 (n/2)^{1/3}} n^{29/24} \quad \text{and} \quad f_0(\kappa) = \mathrm{Ai}(2^{1/3}\kappa + a_1).$$

This way we conjecture the asymptotic form for acyclic minimal DFAs:

$$m_n = 2^n n! e_{2n,0} = \Theta\left(n! 8^n e^{3a_1 n^{1/3}} n^{7/8}\right).$$

Refined heuristic analysis

1. **Ansatz of order 1:**

$$e_{n,m} \approx h(n) f\left(\frac{m+1}{\sqrt[3]{n}}\right),$$
$$s_n = 2 + cn^{-2/3} + O(n^{-1}).$$

yields estimates $c = 2^{2/3} a_1$ such that

$$h(n) \approx 2^n e^{3a_1 (n/2)^{1/3}} \qquad \text{and} \qquad f(\kappa) = \mathrm{Ai}(2^{1/3}\kappa + a_1).$$

2. **Ansatz of order 2:**

$$e_{n,m} \approx h(n) \left(f_0\left(\frac{m+1}{\sqrt[3]{n}}\right) + n^{-1/3} f_1\left(\frac{m+1}{\sqrt[3]{n}}\right)\right),$$
$$s_n = 2 + cn^{-2/3} + dn^{-1} + O(n^{-4/3}).$$

yields estimates $d = 29/12$ such that

$$h(n) \sim const \cdot 2^n e^{3a_1 (n/2)^{1/3}} n^{29/24} \qquad \text{and} \qquad f_0(\kappa) = \mathrm{Ai}(2^{1/3}\kappa + a_1).$$

This way we conjecture the asymptotic form for acyclic minimal DFAs:

$$m_n = 2^n n! e_{2n,0} = \Theta\left(n! 8^n e^{3a_1 n^{1/3}} n^{7/8} \right).$$

The end

Theorem

The number m_n of minimal DFAs recognizing a finite binary language and the number c_n (r_n) of compacted (relaxed) binary trees satisfy for $n \to \infty$

$$m_n = \Theta\left(n!\, 8^n e^{3a_1 n^{1/3}} n^{7/8}\right),$$

$$c_n = \Theta\left(n!\, 4^n e^{3a_1 n^{1/3}} n^{3/4}\right),$$

$$r_n = \Theta\left(n!\, 4^n e^{3a_1 n^{1/3}} n\right),$$

with $a_1 \approx -2.338$: largest root of the Airy function $\mathrm{Ai}(x) = \frac{1}{\pi}\int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$.

Many future research directions:

- Multiplicative constant? Does it exist?
- Characterizing 2-parameter recurrences admitting stretched exponentials.
- Limit shapes: expected height? longest word? etc.
- Further applications to biology and queuing theory.

Open PhD position in my project "Stretched exponentials and beyond"!

The end

Theorem

The number m_n of minimal DFAs recognizing a finite binary language and the number c_n (r_n) of compacted (relaxed) binary trees satisfy for $n \to \infty$

$$m_n = \Theta\left(n!\, 8^n e^{3a_1 n^{1/3}} n^{7/8}\right),$$

$$c_n = \Theta\left(n!\, 4^n e^{3a_1 n^{1/3}} n^{3/4}\right),$$

$$r_n = \Theta\left(n!\, 4^n e^{3a_1 n^{1/3}} n\right),$$

with $a_1 \approx -2.338$: largest root of the Airy function $\mathrm{Ai}(x) = \frac{1}{\pi}\int_0^\infty \cos\left(\frac{t^3}{3} + xt\right) dt$.

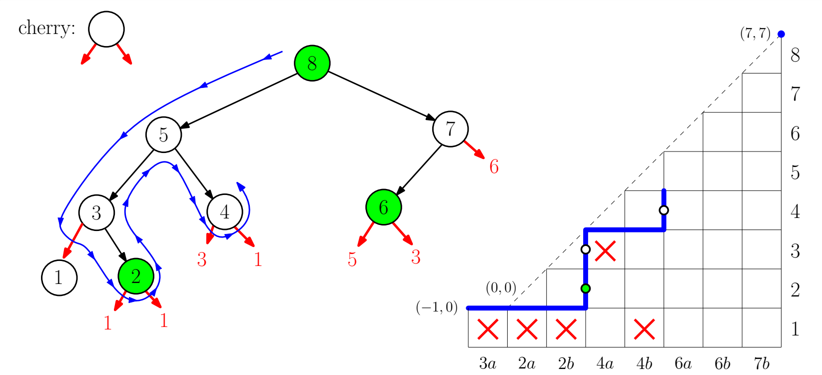
Many future research directions:

- Multiplicative constant? Does it exist?
- Characterizing 2-parameter recurrences admitting stretched exponentials.
- Limit shapes: expected height? longest word? etc.
- Further applications to biology and queuing theory.

Open PhD position in my project "Stretched exponentials and beyond"!

Backup

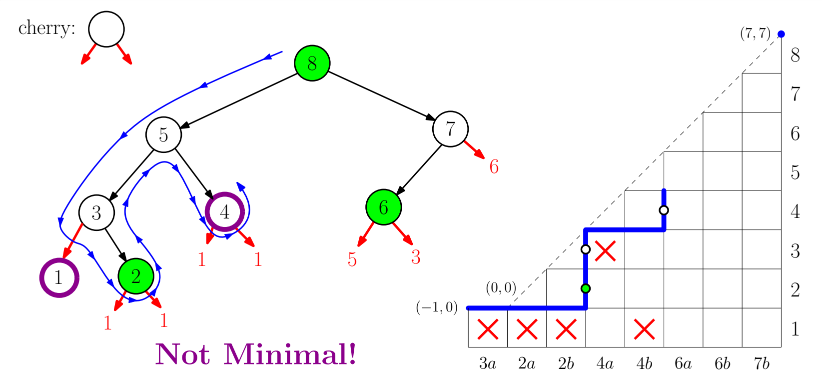
Minimal acyclic DFAs



For the DFA to be minimal, no state can be equivalent to a previous state:

- only possible if the new node is a cherry.
- If cherry is labeled m, then $m-1$ choices (of pointer labels and state color) must be avoided.
- Cherry corresponds to $\rightarrow\rightarrow\uparrow$ in path.

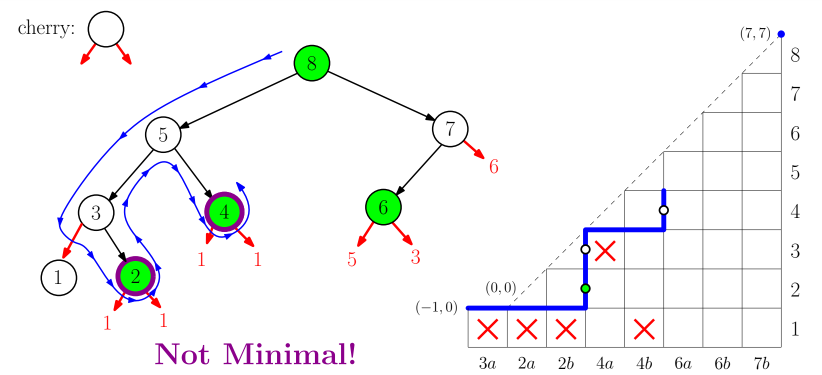
Minimal acyclic DFAs



For the DFA to be minimal, no state can be equivalent to a previous state:

- only possible if the new node is a cherry.
- If cherry is labeled m, then $m - 1$ choices (of pointer labels and state color) must be avoided.
- Cherry corresponds to → → ↑ in path.

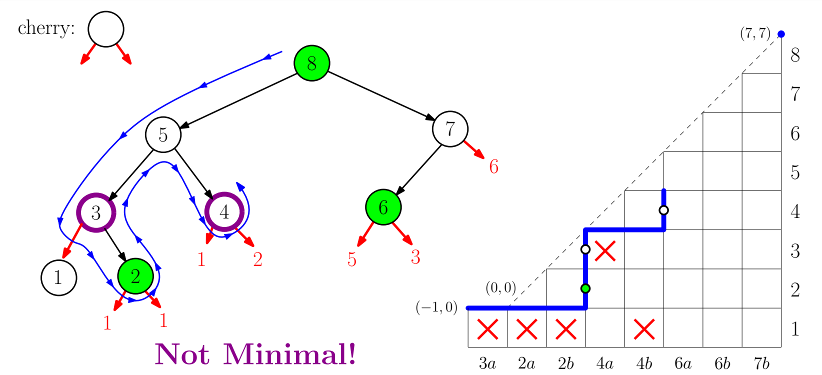
Minimal acyclic DFAs



For the DFA to be minimal, no state can be equivalent to a previous state:

- only possible if the new node is a cherry.
- If cherry is labeled m, then $m-1$ choices (of pointer labels and state color) must be avoided.
- Cherry corresponds to $\rightarrow \rightarrow \uparrow$ in path.

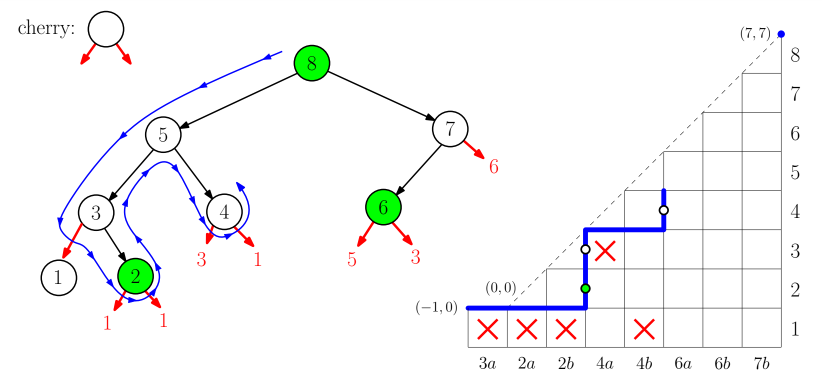
Minimal acyclic DFAs



For the DFA to be minimal, no state can be equivalent to a previous state:

- only possible if the new node is a cherry.
- If cherry is labeled m, then $m - 1$ choices (of pointer labels and state color) must be avoided.
- Cherry corresponds to $\rightarrow\rightarrow\uparrow$ in path.

Minimal acyclic DFAs

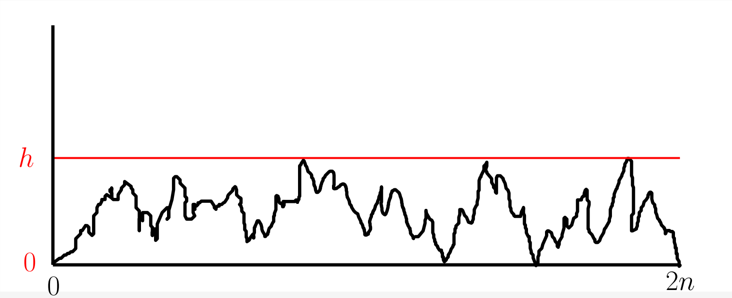


For the DFA to be minimal, no state can be equivalent to a previous state:

- only possible if the new node is a cherry.
- If cherry is labeled m, then $m-1$ choices (of pointer labels and state color) must be avoided.
- Cherry corresponds to $\rightarrow \rightarrow \uparrow$ in path.

Side note: Pushed Dyck paths

Dyck paths of length $2n$ where paths of height h get weight 2^{-h}



Consider paths with max height $h = n^{\alpha}$ (for $0 < \alpha \leq 1/2$):

Number of paths $\approx 4^n e^{-c_1 n^{1-2\alpha}}$, **Weight** $= 2^{-n^{\alpha}} = e^{-\log(2) n^{\alpha}}$.

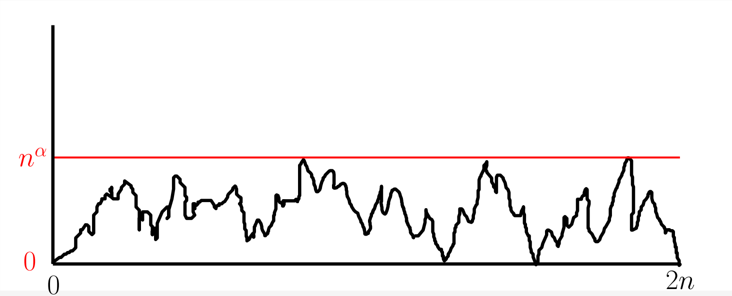
Weighted number of paths is $\approx 4^n e^{-c_1 n^{1-2\alpha} - \log(2) n^{\alpha}}$.

Maximum occurs when $\alpha = 1/3$ and is equal to $4^n e^{-cn^{1/3}}$.

Our case: weights decrease similarly with height so we expect similar behavior

Side note: Pushed Dyck paths

Dyck paths of length $2n$ where paths of height h get weight 2^{-h}



Consider paths with max height $h = n^\alpha$ (for $0 < \alpha \leq 1/2$):

Number of paths $\approx 4^n e^{-c_1 n^{1-2\alpha}}$, **Weight** $= 2^{-n^\alpha} = e^{-\log(2) n^\alpha}$.

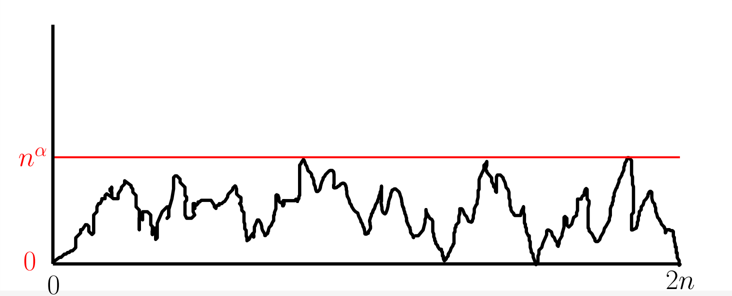
Weighted number of paths is $\approx 4^n e^{-c_1 n^{1-2\alpha} - \log(2) n^\alpha}$.

Maximum occurs when $\alpha = 1/3$ and is equal to $4^n e^{-cn^{1/3}}$.

Our case: weights decrease similarly with height so we expect similar behavior

Side note: Pushed Dyck paths

Dyck paths of length $2n$ where paths of height h get weight 2^{-h}



Consider paths with max height $h = n^\alpha$ (for $0 < \alpha \leq 1/2$):

Number of paths $\approx 4^n e^{-c_1 n^{1-2\alpha}}$, **Weight** $= 2^{-n^\alpha} = e^{-\log(2) n^\alpha}$.

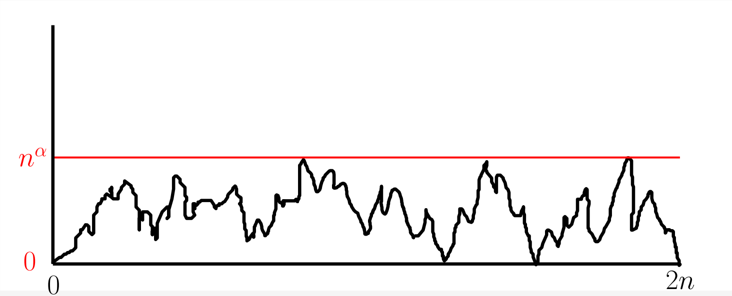
Weighted number of paths is $\approx 4^n e^{-c_1 n^{1-2\alpha} - \log(2) n^\alpha}$.

Maximum occurs when $\alpha = 1/3$ and is equal to $4^n e^{-cn^{1/3}}$.

Our case: weights decrease similarly with height so we expect similar behavior

Side note: Pushed Dyck paths

Dyck paths of length $2n$ where paths of height h get weight 2^{-h}



Consider paths with max height $h = n^\alpha$ (for $0 < \alpha \leq 1/2$):

Number of paths $\approx 4^n e^{-c_1 n^{1-2\alpha}}$, **Weight** $= 2^{-n^\alpha} = e^{-\log(2) n^\alpha}$.

Weighted number of paths is $\approx 4^n e^{-c_1 n^{1-2\alpha} - \log(2) n^\alpha}$.

Maximum occurs when $\alpha = 1/3$ and is equal to $4^n e^{-cn^{1/3}}$.

Our case: weights decrease similarly with height so we expect similar behavior

Proof method

Recall:

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}$$

Number of minimal acyclic DFAs is $m_n = 2^n n! e_{2n,0}$.

Method:

Find sequences $X_{n,m}$ and $Y_{n,m}$ with the same asymptotic form, such that

$$X_{n,m} \leq e_{n,m} \leq Y_{n,m},$$

for all m and all n large enough.

Proof method

Recall:

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}$$

Number of minimal acyclic DFAs is $m_n = 2^n n! e_{2n,0}$.

Method:

Find sequences $X_{n,m}$ and $Y_{n,m}$ with the same asymptotic form, such that

$$X_{n,m} \leq e_{n,m} \leq Y_{n,m},$$

for all m and all n large enough.

How to find them?

1. Use heuristics
2. Fiddle until $X_{n,m}$ and $Y_{n,m}$ satisfy the recurrence of $e_{n,m}$ with the equalities replaced by inequalities:

   $$= \quad \longrightarrow \quad \leq \text{ and } \geq$$

3. Prove $X_{n,m} \leq e_{n,m} \leq Y_{n,m}$ by induction.

Proof method

Recall:

$$e_{n,m} = \left(1 - \frac{2(m+1)}{n+m}\right) e_{n-1,m-1} + e_{n-1,m+1} - \frac{n-m}{(n+m)(n+m-2)} e_{n-3,m-1}$$

Number of minimal acyclic DFAs is $m_n = 2^n n! e_{2n,0}$.

Method:

Find sequences $X_{n,m}$ and $Y_{n,m}$ with the same asymptotic form, such that

$$X_{n,m} \leq e_{n,m} \leq Y_{n,m},$$

for all m and all n large enough.

How to find them?

1. Use heuristics
2. Fiddle until $X_{n,m}$ and $Y_{n,m}$ satisfy the recurrence of $e_{n,m}$ with the equalities replaced by inequalities:

$$= \quad \longrightarrow \quad \leq \text{ and } \geq$$

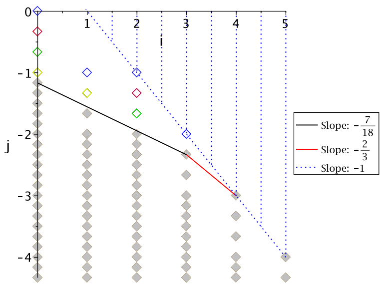
3. Prove $X_{n,m} \leq e_{n,m} \leq Y_{n,m}$ by induction.

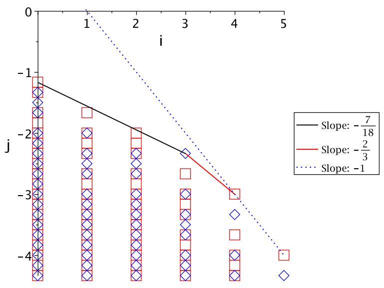
Unfortunately very technical (and not suited for the end of a talk ;))

Technicalities for compacted trees and minimal DFAs

Lots of technicalities:

- Before induction, we have to remove the negative term from the recurrence, but we have to do so precisely for asymptotics to stay the same.
- We only prove bounds for small *m*; we prove that large *m* terms don't matter
- The lower bound is negative for very large *m*, so we have to be careful with induction
- We only prove the bounds for sufficiently large *n*, but this only makes a difference to the constant term. Proof involves colorful Newton polygons:





Relaxed problem (relaxed compacted trees)

Recurrence for relaxed compacted trees

$$d_{n,m} = \frac{n-m+2}{n+m} d_{n-1,m-1} + d_{n-1,m+1}.$$

Lemma (lower bound)

For all $n, m \geq 0$ let

$$\tilde{X}_{n,m} := \left(1 - \frac{2m^2}{3n} + \frac{m}{2n}\right) \mathrm{Ai}\left(a_1 + \frac{2^{1/3}(m+1)}{n^{1/3}}\right) \qquad \text{and}$$

$$\tilde{s}_n := 2 + \frac{2^{2/3} a_1}{n^{2/3}} + \frac{8}{3n} - \frac{1}{n^{7/6}}.$$

Then, for any $\varepsilon > 0$, there exists an $\tilde{n}_0$ such that

$$\tilde{X}_{n,m} \tilde{s}_n \leq \frac{n-m+2}{n+m} \tilde{X}_{n-1,m-1} + \tilde{X}_{n-1,m+1},$$

for all $n \geq \tilde{n}_0$ and for all $0 \leq m < n^{1-\varepsilon}$.

Lower bound – Expansion

1. Transform to $P_{n,m} \geq 0$ for

$$P_{n,m} := -\tilde{X}_{n,m}\tilde{s}_n + \frac{n-m+2}{n+m}\tilde{X}_{n-1,m-1} + \tilde{X}_{n-1,m+1}.$$

where $(\sigma_i, \tau_j \in \mathbb{R})$

$$\tilde{s}_n := \sigma_0 + \frac{\sigma_1}{n^{1/3}} + \frac{\sigma_2}{n^{2/3}} + \frac{\sigma_3}{n} + \frac{\sigma_4}{n^{7/6}},$$

$$\tilde{X}_{n,m} := \left(1 + \frac{\tau_2 m^2 + \tau_1 m}{n}\right) \mathrm{Ai}\left(a_1 + \frac{2^{1/3}(m+1)}{n^{1/3}}\right).$$

2. Expand $\mathrm{Ai}(z)$ in a neighborhood of

$$\alpha = a_1 + \frac{2^{1/3}m}{n^{1/3}},$$

using $\mathrm{Ai}''(z) = z\mathrm{Ai}(z)$. Then

$$P_{n,m} = p_{n,m}\mathrm{Ai}(\alpha) + p'_{n,m}\mathrm{Ai}'(\alpha),$$

where $p_{n,m}$ and $p'_{n,m}$ are power series in $n^{-1/6}$ whose coefficients are polynomials in m.

Lower bound – Expansion

1. Transform to $P_{n,m} \geq 0$ for

$$P_{n,m} := -\tilde{X}_{n,m}\tilde{s}_n + \frac{n-m+2}{n+m}\tilde{X}_{n-1,m-1} + \tilde{X}_{n-1,m+1}.$$

where $(\sigma_i, \tau_j \in \mathbb{R})$

$$\tilde{s}_n := \sigma_0 + \frac{\sigma_1}{n^{1/3}} + \frac{\sigma_2}{n^{2/3}} + \frac{\sigma_3}{n} + \frac{\sigma_4}{n^{7/6}},$$

$$\tilde{X}_{n,m} := \left(1 + \frac{\tau_2 m^2 + \tau_1 m}{n}\right) \mathrm{Ai}\left(a_1 + \frac{2^{1/3}(m+1)}{n^{1/3}}\right).$$

2. Expand $\mathrm{Ai}(z)$ in a neighborhood of

$$\alpha = a_1 + \frac{2^{1/3}m}{n^{1/3}},$$

using $\mathrm{Ai}''(z) = z\mathrm{Ai}(z)$. Then

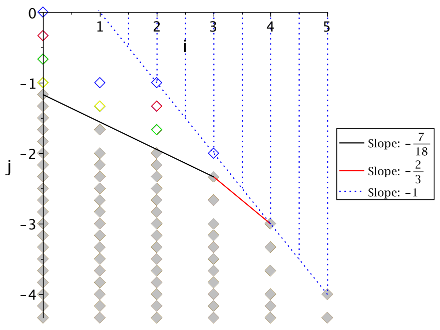
$$P_{n,m} = p_{n,m}\mathrm{Ai}(\alpha) + p'_{n,m}\mathrm{Ai}'(\alpha),$$

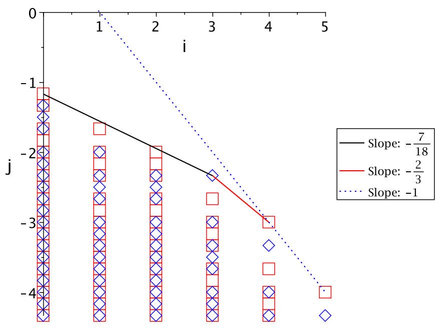
where $p_{n,m}$ and $p'_{n,m}$ are power series in $n^{-1/6}$ whose coefficients are polynomials in m.

Lower bound – Polygon

We get

$$P_{n,m} = \mathrm{Ai}(\alpha)\left(-\frac{\sigma_4}{n^{7/6}} - \frac{2^{5/3}a_1 m}{3n^{5/3}} - \frac{41m^2}{9n^2} - \frac{2^{8/3}a_1 m^3}{3n^{8/3}} - \frac{34m^4}{9n^3} - \frac{62m^5}{135n^4} + \dots\right) +$$
$$\mathrm{Ai}'(\alpha)\left(\frac{2^{1/3}(2\tau_1 - 1)}{n^{4/3}} + \frac{2^{1/3}}{n^{3/2}} - \frac{8a_1 m}{9n^2} + \frac{2^{1/3}(24\tau_1 - 31)m^2}{9n^{7/3}} - \frac{2^{13/3}m^3}{9n^{7/3}} + \dots\right.$$



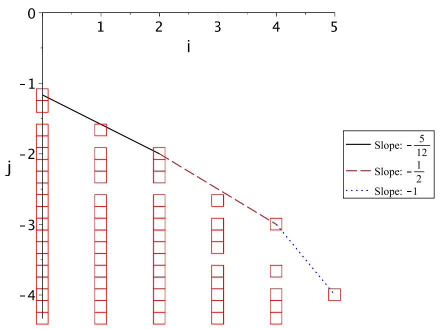


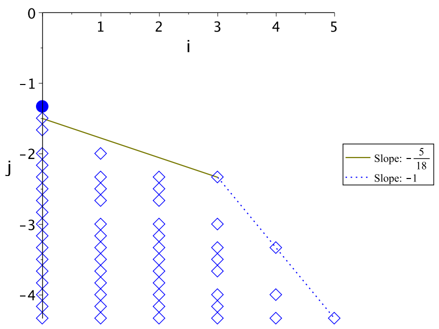
Lower bound – Case analysis

3. Treat $p_{n,m}$ and $p'_{n,m}$ separately and prove that all dominating terms (corners of convex hull) are positive.

$$p_{n,m} = \sum \tilde{a}_{i,j} m^i n^j$$

$$p'_{n,m} = \sum \tilde{a}'_{i,j} m^i n^j$$





non-zero coefficients

Relaxed trees: Proof idea – lower bound

Main idea

Suppose $(X_{n,m})_{n \geq m \geq 0}$ and $(s_n)_{n \geq 1}$ satisfy

$$X_{n,m} s_n \leq \frac{n-m+2}{n+m} X_{n-1,m-1} + X_{n-1,m+1}, \qquad (1)$$

for all sufficiently large n and all integers $m \in [0, n]$.

Define $(h_n)_{n \geq 0}$ by $h_0 = 1$ and $h_n = s_n h_{n-1}$; then prove that

$$X_{n,m} h_n \leq b_0 d_{n,m}$$

for some constant b_0 by induction:

Relaxed trees: Proof idea – lower bound

Main idea

Suppose $(X_{n,m})_{n\geq m\geq 0}$ and $(s_n)_{n\geq 1}$ satisfy

$$X_{n,m}s_n \leq \frac{n-m+2}{n+m}X_{n-1,m-1} + X_{n-1,m+1}, \tag{1}$$

for all sufficiently large n and all integers $m \in [0,n]$.

Define $(h_n)_{n\geq 0}$ by $h_0 = 1$ and $h_n = s_n h_{n-1}$; then prove that

$$X_{n,m}h_n \leq b_0 d_{n,m}$$

for some constant b_0 by induction:

Relaxed trees: Proof idea – lower bound

Main idea

Suppose $(X_{n,m})_{n \geq m \geq 0}$ and $(s_n)_{n \geq 1}$ satisfy

$$X_{n,m} s_n \leq \frac{n-m+2}{n+m} X_{n-1,m-1} + X_{n-1,m+1}, \qquad (1)$$

for all sufficiently large n and all integers $m \in [0, n]$.

Define $(h_n)_{n \geq 0}$ by $h_0 = 1$ and $h_n = s_n h_{n-1}$; then prove that

$$X_{n,m} h_n \leq b_0 d_{n,m}$$

for some constant b_0 by induction:

$$X_{n,m} h_n \overset{(1)}{\leq} \frac{n-m+2}{n+m} X_{n-1,m-1} h_{n-1} + X_{n-1,m+1} h_{n-1}$$

$$\overset{\text{(Induction)}}{\leq} \frac{n-m+2}{n+m} b_0 d_{n-1,m-1} + b_0 d_{n-1,m+1}$$

$$\overset{\text{Rec. } d_{n,m}}{=} b_0 d_{n,m}.$$

Relaxed trees: Proof idea – lower bound

Main idea

Suppose $(X_{n,m})_{n\geq m\geq 0}$ and $(s_n)_{n\geq 1}$ satisfy

$$X_{n,m}s_n \leq \frac{n-m+2}{n+m}X_{n-1,m-1} + X_{n-1,m+1}, \tag{1}$$

for all sufficiently large n and all integers $m \in [0, n]$.

Define $(h_n)_{n\geq 0}$ by $h_0 = 1$ and $h_n = s_n h_{n-1}$; then prove that

$$X_{n,m}h_n \leq b_0 d_{n,m}$$

for some constant b_0 by induction:

$$\begin{aligned} X_{n,m}h_n &\overset{(1)}{\leq} \frac{n-m+2}{n+m}X_{n-1,m-1}h_{n-1} + X_{n-1,m+1}h_{n-1} \\ &\overset{\text{(Induction)}}{\leq} \frac{n-m+2}{n+m}b_0 d_{n-1,m-1} + b_0 d_{n-1,m+1} \\ &\overset{\text{Rec. } d_{n,m}}{=} b_0 d_{n,m}. \end{aligned}$$

Relaxed trees: Proof idea – lower bound

Main idea

Suppose $(X_{n,m})_{n\geq m\geq 0}$ and $(s_n)_{n\geq 1}$ satisfy

$$X_{n,m}s_n \leq \frac{n-m+2}{n+m}X_{n-1,m-1} + X_{n-1,m+1}, \qquad (1)$$

for all sufficiently large n and all integers $m \in [0,n]$.

Define $(h_n)_{n\geq 0}$ by $h_0 = 1$ and $h_n = s_n h_{n-1}$; then prove that

$$X_{n,m}h_n \leq b_0 d_{n,m}$$

for some constant b_0 by induction:

$$\begin{aligned} X_{n,m}h_n &\overset{(1)}{\leq} \frac{n-m+2}{n+m}X_{n-1,m-1}h_{n-1} + X_{n-1,m+1}h_{n-1} \\ &\overset{\text{(Induction)}}{\leq} \frac{n-m+2}{n+m}b_0 d_{n-1,m-1} + b_0 d_{n-1,m+1} \\ &\overset{\text{Rec. } d_{n,m}}{=} b_0 d_{n,m}. \end{aligned}$$