Prime Factorization from a Two-Bit-per-Integer Encoding

Orges Leka with help from ChatGPT 5

November 4, 2025

Abstract

We show that the complete set of prime factorizations of $1, \ldots, n$ is faithfully encoded by a Dyck word w_n of length $2n$ that captures the shape of a prime-multiplication tree T_n. From w_n alone and the list of primes up to n, all factorizations can be enumerated in total time $\Theta(n \log \log n)$ and $O(n)$ space, which is optimal up to constants due to the output size. We formalize admissible insertions, prove local commutativity and global confluence (any linear extension of the ancestor poset yields T_N), and investigate the direct limit tree T_∞. Under an explicit uniform-insertion heuristic, the pooled insertion index obeys an exact mixture-of-uniforms law with density $f(x) = -\log x$ on $(0, 1)$, matching simulations.

Contents

1 Introduction **2**

2 The prime-multiplication tree T_n **3**

3 Encoding T_n as a Dyck word w_n **3**
3.1 Triangles in the Dyck path of w_n 4

4 Epistemic reconstruction: from w_n to all factorizations **5**
4.1 Data and primitives 5
4.2 Reconstruction algorithm 6

5 The insertion index distribution: a mixture-of-uniforms law **7**
5.1 Hypotheses 7
5.2 Discrete law (exact) 8
5.3 Scaling limit via pooling 9

6 Commutativity of admissible insertions and confluence to T_N **10**
6.1 Parent, child set, and prime rank 10
6.2 Admissible insertion at a fixed horizon 10
6.3 Local commutativity for admissible insertions 11
6.4 Global confluence: any linear extension yields T_N 13
6.5 Worked example: swapping 5 and 6 at horizon $N = 6$ 14

7 The limit tree T_∞ **15**
7.1 A category of rooted ordered trees and embeddings 15
7.2 The direct limit tree and its relation to T_n 15
7.3 An explicit set–theoretic description of T_∞ 16

8 Abstract properties of the limit tree T_∞ 16
8.1 Unique path property: arithmetic determinism . . . 16
8.2 Recursive self-similarity . . . 17
8.3 Order structure vs. divisibility . . . 17
8.4 Topological and degree properties . . . 18

9 An abstract divisor–ordered setting for the factorization tree 18
9.1 Ordered unique-factorization monoids . . . 18
9.2 The abstract prime-restricted growth tree . . . 19
9.3 Uniqueness of paths and basic properties . . . 20
9.4 Finite truncations and Dyck encoding . . . 21
9.5 Examples . . . 21

10 Rational functions attached to the trees 24
10.1 Example values of $p(T_n, x)$ . . . 26

11 Factorization trees, the $p(T_n, x)$ recursion, and the link to $\pi(n)$ 27
11.1 Definitions and basic setup . . . 27
11.2 Part (a): recursion in n and the branched continued fraction . . . 27
11.3 Part (b): the identity for $c(n)$ and bounds by $\pi(n)$ . . . 28

A SymPy/Numpy/Matplotlib script replicating the experiments 29

1 Introduction

Motivation. Prime factorization across $1, \ldots, n$ carries strong global structure: every integer appears exactly once as a product of nondecreasing primes. We make this structure explicit by organizing all factorizations into a single rooted ordered tree T_n: a node m has children mp with p prime and $P_1(m) \le p \le \lfloor n/m \rfloor$, ordered by increasing p. A depth-first traversal outputs a Dyck word w_n of length $2n$. Thus T_n (hence all factorizations up to n) is faithfully encoded by $2n$ bits.

Epistemic message. From w_n alone and a precomputed list of primes up to n (via a sieve in $O(n \log\log n)$ time), one can reconstruct the prime factorization of *every* $k \le n$ in total time $\Theta(n \log\log n)$ using only primality/progression of primes. This matches the information-theoretic output lower bound and yields an amortized $\Theta(\log\log n)$ per integer.

Contributions.

- A tree/Dyck encoding $T_n \leftrightarrow w_n$ that supports $\Theta(n \log\log n)$ total-time reconstruction of all factorizations $1 \le k \le n$.
- A self-similar functional system for Q_r leading to branched S-fractions, Mahler-type scaling, prime-series bounds, and local Taylor schemes.
- Simple two-sided bounds and convergents (first levels of the S-fraction) that already bracket p_∞ tightly and are numerically stable.

Organization. Section 2 defines the trees T_n and their Dyck encoding and proves uniqueness/coverage properties. Section 5 develops the insertion-index model. An executable script for testing these ideas is printed at the Appendix.

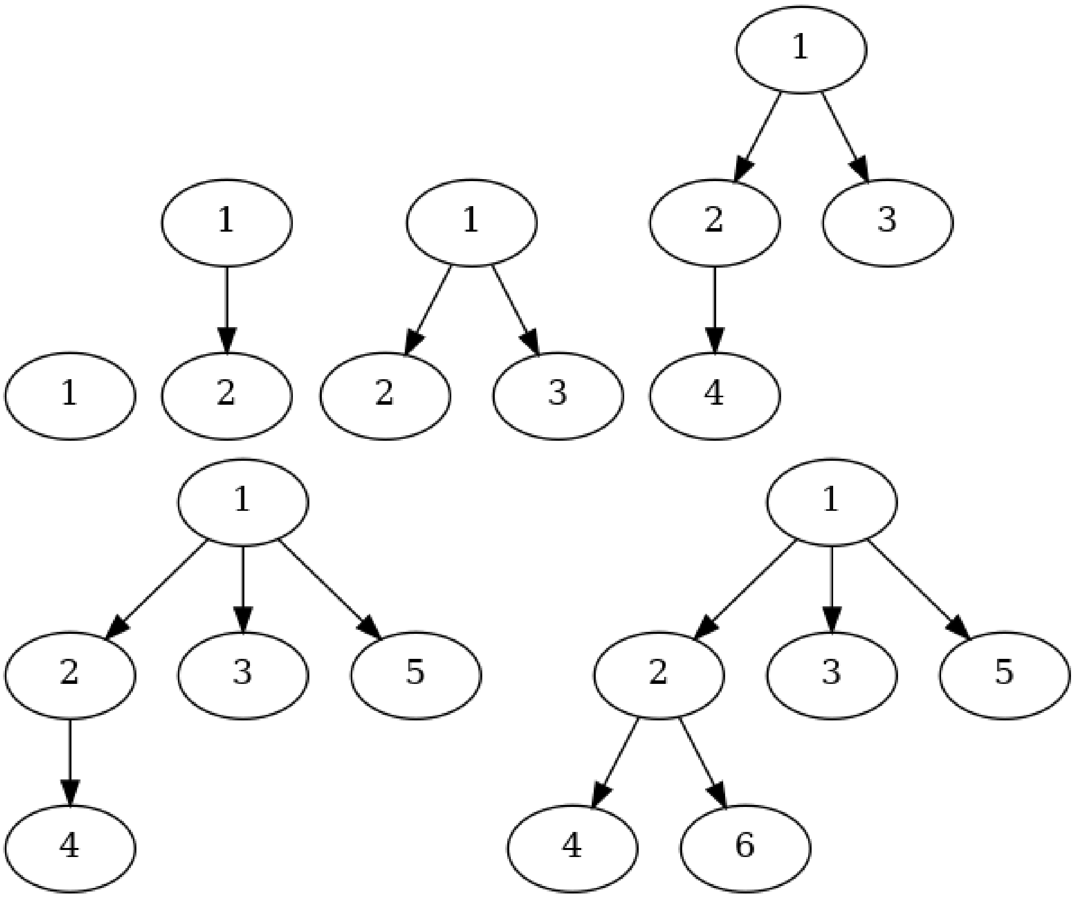

Figure 1: Some trees.

2 The prime-multiplication tree T_n

Let $P_1(1) := 1$ and, for $m \geq 2$, let $P_1(m)$ denote the largest prime divisor of m.

Definition 2.1 (Local trees and the global tree)**.** For $1 \leq m \leq n$, the rooted tree $T_{n,m}$ is defined recursively:

- the root is the integer m;
- if there exist primes p with $P_1(m) \leq p \leq \lfloor n/m \rfloor$, then the children of m are the roots of the trees $T_{n,mp}$ over all such primes p (in increasing order of p); otherwise m is a leaf.

We set $T_n := T_{n,1}$.

Remark 2.2 (Coverage, uniqueness, and order)**.** Each integer $1 \leq m \leq n$ appears *exactly once* as a node: represent $m = \prod_{i=1}^{r} p_i$ with nondecreasing primes $p_1 \leq \cdots \leq p_r$; then the path $1 \to p_1 \to p_1 p_2 \to \cdots \to m$ respects the rule $p_{i+1} \geq p_i = P_1(\prod_{j \leq i} p_j)$ and yields the unique node m. The children of m are exactly the mp with $p \in [P_1(m), \lfloor n/m \rfloor]$ prime.

3 Encoding T_n as a Dyck word w_n

Traverse T_n in depth-first (preorder) fashion. Output a "1" when a node is visited (opening) and a "0" when its subtree is fully processed (closing). This yields a Dyck word $w_n \in \{1,0\}^{2n}$ of length $2n$. The mapping $T_n \leftrightarrow w_n$ is bijective: from a Dyck word one can reconstruct the underlying ordered rooted tree by the usual stack algorithm in $O(n)$ time.

3.1 Triangles in the Dyck path of w_n

We view the Dyck word $w_n \in \{1,0\}^{2n}$ obtained from the preorder traversal of T_n as a lattice path that starts at height 0, interprets each symbol 1 as an up-step (+1) and each 0 as a down-step (−1), and never goes below 0.

What we count. For a Dyck word w we define the number of (isosceles right) *triangles* $t(w)$ purely combinatorially as follows, using the standard first-return decomposition. Let $|A|_1$ denote the number of symbols 1 contained in a word A.

- $t(\varepsilon) := 0$.
- Every nonempty Dyck word is uniquely of the form

  $$w \; = \; 1\,A\,0\,B,$$

  with A, B Dyck words (the first return to height 0). We then set

  $$\boxed{t(w) \; = \; (1 + 2\,|A|_1) \; + \; t(A) \; + \; t(B).} \tag{1}$$

Geometric intuition. The term 1 counts the outer triangle of the mountain $1A0$. Each of the $|A|_1$ inner peaks of A creates exactly two boundary triangles along the sides of that outer triangle, hence $2|A|_1$. The triangles strictly inside the mountain are exactly those of A, and the triangles to the right are those of B. These three classes are disjoint, which yields (1). For $n \geq 1$ we write $t_n := t(w_n)$.

Inserting a leaf 10 increases t by $1 + 2d$. When we pass from w to w^+ by inserting one new leaf 10 at some position of the word (this is precisely what happens when we move from T_{n-1} to T_n), the *insertion depth* d is the height of the path at the insertion point (equivalently, the current stack depth).

Lemma 3.1 (Linear increment)**.** *If w^+ is obtained from w by inserting 10 at depth d, then*

$$t(w^+) - t(w) \; = \; 1 + 2d.$$

Proof. We argue by induction on d using (1).

Base $d = 0$. Inserting at height 0 appends an additional mountain 10 to the right (or, equivalently, modifies only the B-part in a decomposition $w = 1A0\,B$). Then $t(B)$ increases by $t(10) - t(\varepsilon) = 1$, and all other terms in (1) are unchanged. Hence $\Delta t = 1 = 1 + 2 \cdot 0$.

Induction step. Suppose $d \geq 0$ and the insertion occurs inside the A-part of the decomposition $w = 1A0\,B$. Let A^+ be A after the same insertion. Then $|A^+|_1 = |A|_1 + 1$ and by (1)

$$t(w^+) - t(w) = \Big(1 + 2|A^+|_1 + t(A^+) + t(B)\Big) - \Big(1 + 2|A|_1 + t(A) + t(B)\Big) = 2 + \big(t(A^+) - t(A)\big).$$

The insertion depth inside A is $d - 1$. By the induction hypothesis applied to A we have $t(A^+) - t(A) = 1 + 2(d-1)$. Therefore $t(w^+) - t(w) = 2 + \big(1 + 2(d-1)\big) = 1 + 2d$, as claimed. □

Depth equals Ω. In the prime-multiplication tree, a node m has depth

$$\mathrm{depth}(m) = \Omega(m),$$

the number of prime factors of m counted with multiplicity, because the unique path $1 \to p_1 \to p_1 p_2 \to \cdots \to m = \prod_{i=1}^{r} p_i$ performs exactly $r = \Omega(m)$ prime multiplications.

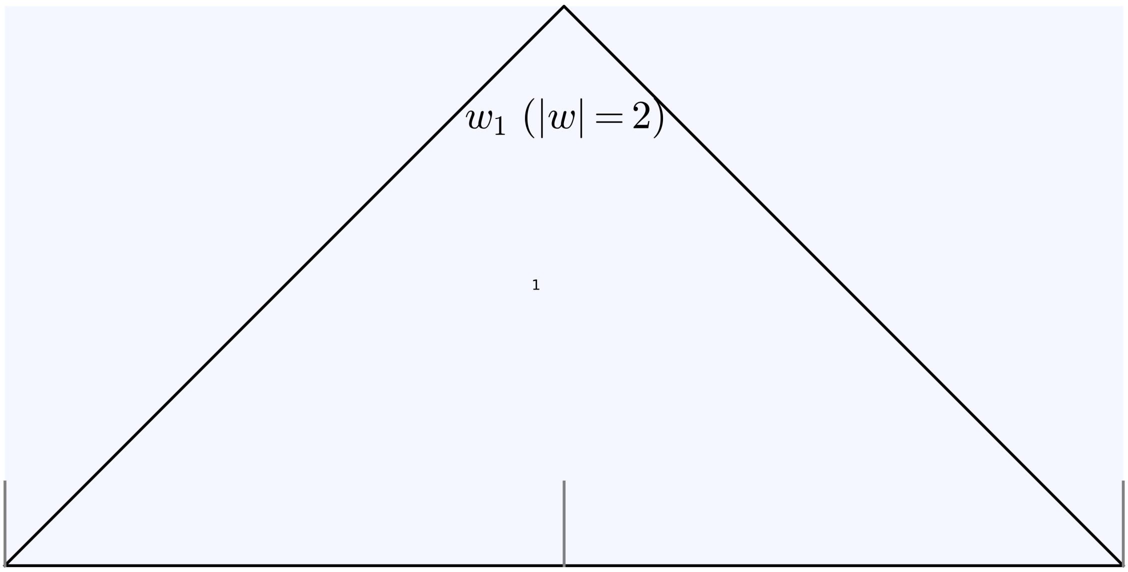

Figure 2: Dyck path for $n = 1$ with numbers at corresponding levels

Theorem 3.2. *Let $t_n := t(w_n)$. Then*

$$\boxed{t_n \;=\; t_{n-1} + 2\,\Omega(n) + 1}$$

and consequently

$$\boxed{t_n \;=\; \sum_{k=1}^{n} \bigl(1 + 2\,\Omega(k)\bigr) \;=\; n \;+\; 2\sum_{k=1}^{n} \Omega(k).}$$

Proof. Passing from T_{n-1} to T_n inserts the new leaf n at depth $d = \Omega(n)$. By Lemma 3.1 the increment in the triangle count is $1 + 2d = 1 + 2\Omega(n)$, which proves the recursion. Summing the recursion and using $t_1 = t(10) = 1$ yields the closed form. $\square$

Remark 3.3. By Hardy–Ramanujan, $\frac{1}{n}\sum_{k\le n} \Omega(k) \sim \log\log n$. Hence $t_n = n + 2\sum_{k\le n} \Omega(k)$ grows like $2n \log\log n + n$. Geometrically, $t(w)$ counts all (possibly nested) isosceles right triangles whose legs are subsegments of the Dyck path; the recursion (1) matches exactly the first-return decomposition of the path.

4 Epistemic reconstruction: from w_n to all factorizations

We now show that from w_n alone we can list prime factorizations of all $1 \le k \le n$ quickly.

4.1 Data and primitives

- Input: the word w_n of length $2n$ (equivalently the ordered rooted tree T_n).
- Primitive 1: primality testing (AKS gives polylog time; in practice Miller–Rabin).
- Primitive 2: the list of primes $\le n$, which we precompute by a sieve in time $O(n \log\log n)$ and space $O(n)$; we also keep the array $\mathsf{primes}[1..\pi(n)]$ and a "next prime" iterator.

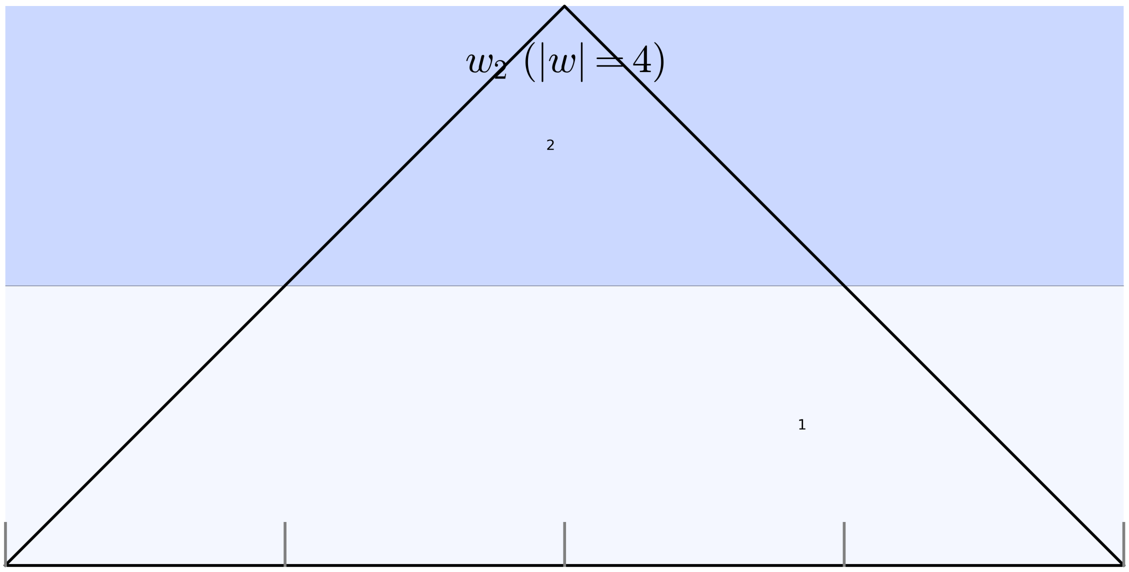

Figure 3: Dyck path for $n = 2$ with numbers at corresponding levels

4.2 Reconstruction algorithm

Step A: parse w_n to T_n (linear time). Scan w_n with a stack to build the ordered rooted tree (children order inherited from the parentheses nesting). This costs $O(n)$.

Step B: label the nodes and output factorizations. We do a single DFS over T_n. At each node we maintain:

- the multiset (ordered list) of prime factors along the path; its last entry is $P_1(m)$;
- the current integer value m is not needed to output the factorization, but it can be produced on the fly as a product of small primes (fits in $\tilde{O}(\log n)$ bits).

For a node labeled by the path primes $(p_1 \le \cdots \le p_r)$, with $P_1(m) = p_r$ (or 1 at the root), the children are exactly the primes p in the interval

$$\mathcal{I}(m) := \{\, p \text{ prime} : \ p_r \le p \le \lfloor n/m \rfloor \,\},$$

in increasing order. We enumerate $\mathcal{I}(m)$ using the precomputed prime list and create children with factor lists $(p_1, \dots, p_r, p)$. We output the factorization of each visited node (i.e., each $m \le n$) as we go.

Complexity analysis.

- *Parsing* $w_n \to T_n$: $O(n)$ time, $O(n)$ space.
- *Prime precomputation*: Sieve up to n in $O(n \log\log n)$ time, $O(n)$ space.
- *Traversal and enumeration*: Over the entire tree, the total number of children equals the number of edges, $n - 1$. With a pointer into the prime table for each interval, we spend $O(1)$ amortized per child; thus $O(n)$ total.

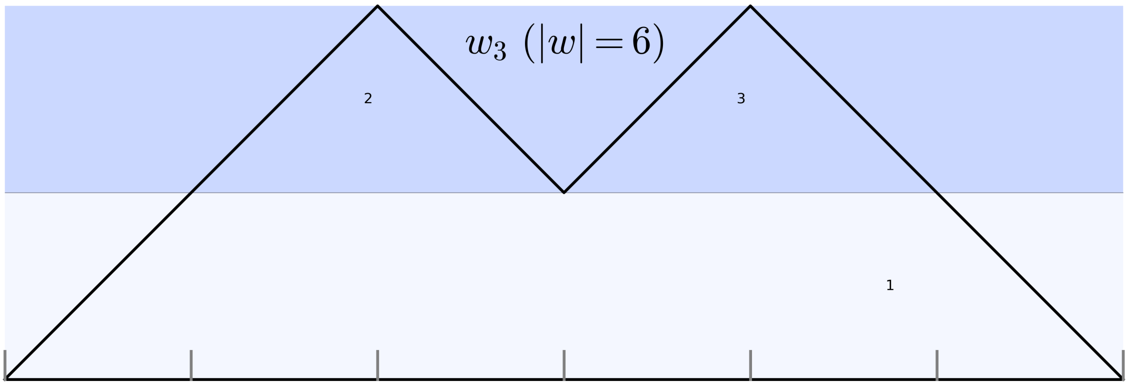

Figure 4: Dyck path for $n = 3$ with numbers at corresponding levels

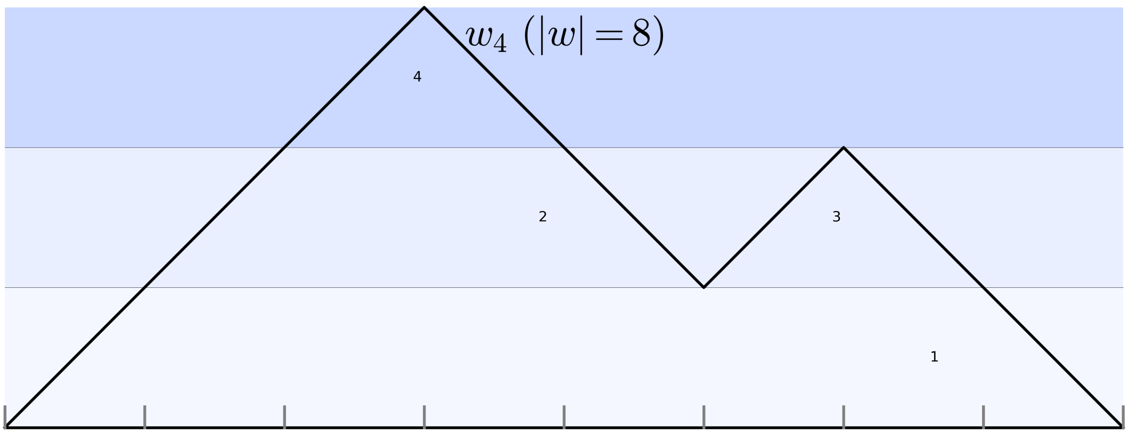

Figure 5: Dyck path for $n = 4$ with numbers at corresponding levels

- *Output size lower bound*: The total length of all factorizations is $\sum_{k \le n} \omega(k) = n \log\log n + Bn + o(n)$ (Hardy–Ramanujan), so the output alone costs $\Theta(n \log\log n)$.

Theorem 4.1 (Tight asymptotics)**.** *Given the word w_n (length $2n$), one can output the prime factorization of every integer $1 \le k \le n$ in total time*

$$\boxed{\Theta(n \log\log n) \quad \textit{time and} \quad O(n) \quad \textit{space},}$$

which is optimal up to constants due to the $\Theta(n \log\log n)$ output size. The amortized time per integer is $\Theta(\log\log n)$; the per-integer work equals $\Theta(\omega(k))$ with $\mathbb{E}[\omega(k)] \sim \log\log n$ and worst-case $O(\log n)$.

5 The insertion index distribution: a mixture-of-uniforms law

5.1 Hypotheses

(U@fix) (*Uniformity at a fixed step*) For each fixed $n \ge 2$ the insertion index

$$s_n \in \{0, 1, \ldots, 2(n-1) - 1\}$$

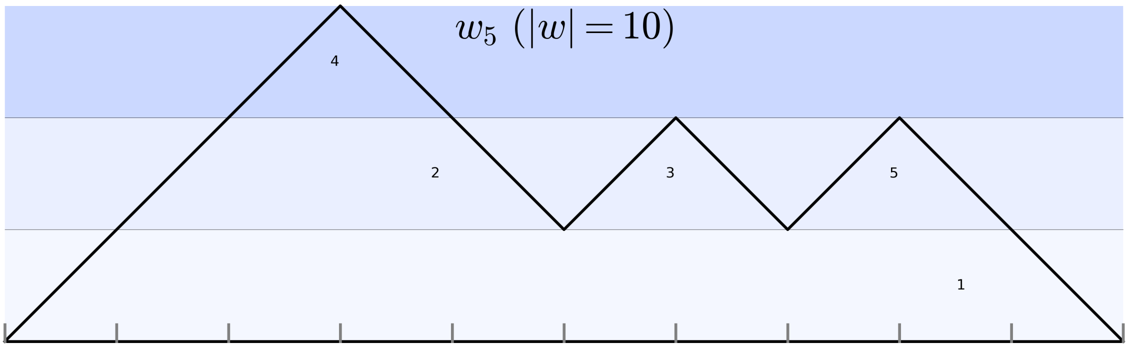

Figure 6: Dyck path for $n = 5$ with numbers at corresponding levels

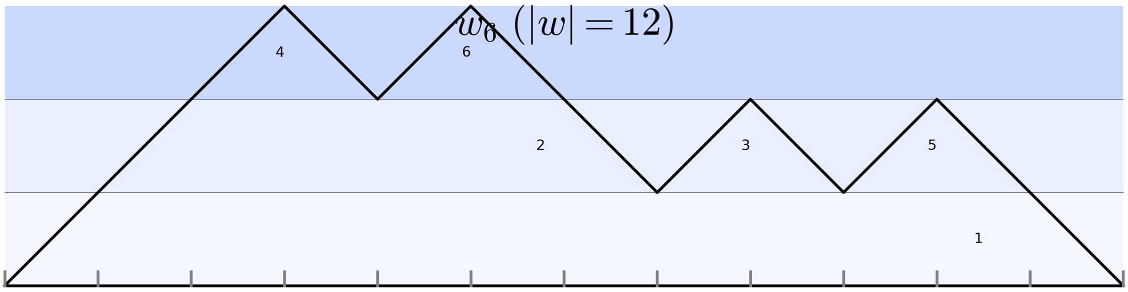

Figure 7: Dyck path for $n = 6$ with numbers at corresponding levels

is *uniformly distributed* over the $2(n-1)$ available slots, i.e. $\mathbb{P}(s_n = s) = \frac{1}{2(n-1)}$ for all admissible s.

(Pool) (*Pooling across steps*) For fixed $N \geq 2$ we first draw n *uniformly* from $\{2, \ldots, N\}$ and then, *conditional* on n, draw S *uniformly* from $\{0, \ldots, 2(n-1)-1\}$ (i.e., we forget n afterwards and look at the pooled indices).

5.2 Discrete law (exact)

Proposition 5.1 (Mixture-of-uniforms, discrete)**.** *Under **(U@fix)** and **(Pool)**, for* $s \in \{0, 1, \ldots, 2(N-1)-1\}$ *we have*

$$\mathbb{P}(S = s) = \frac{1}{N-1} \sum_{n=2}^{N} \frac{\mathbf{1}\{s < 2(n-1)\}}{2(n-1)} = \frac{H_{N-1} - H_{\lfloor s/2 \rfloor}}{2(N-1)},$$

where $H_m = \sum_{k=1}^{m} \frac{1}{k}$ *(and* $H_0 := 0$*) are the harmonic numbers.*

Proof. Fix s. This value can only occur for steps with $2(n-1) > s$, i.e. $n \geq \lfloor s/2 \rfloor + 2$. Conditional on such n, the probability of that specific s equals $1/(2(n-1))$ (by **(U@fix)**). Averaging over $n \in \{2, \ldots, N\}$ gives

$$\mathbb{P}(S = s) = \frac{1}{N-1} \sum_{n=\lfloor s/2 \rfloor + 2}^{N} \frac{1}{2(n-1)} = \frac{1}{2(N-1)} \sum_{m=\lfloor s/2 \rfloor + 1}^{N-1} \frac{1}{m} = \frac{H_{N-1} - H_{\lfloor s/2 \rfloor}}{2(N-1)}.$$

□

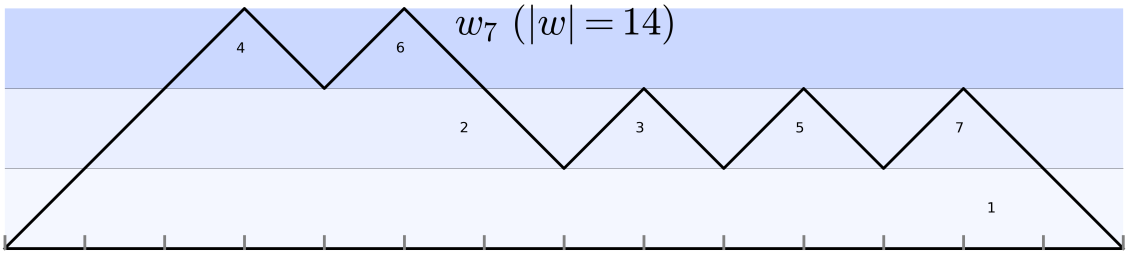

Figure 8: Dyck path for $n = 7$ with numbers at corresponding levels

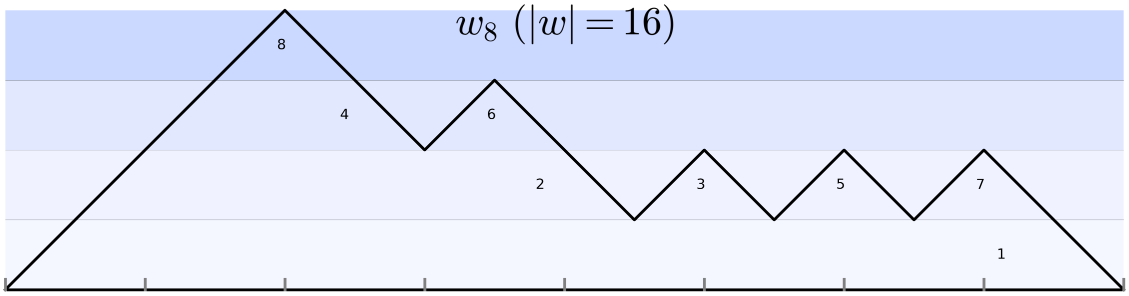

Figure 9: Dyck path for $n = 8$ with numbers at corresponding levels

Remark 5.2 (Shape)**.** The mixture is *heaviest near the middle* (many steps contribute) and *lightest near the extremes* (few steps contribute); the weights are governed by harmonic sums. There are *no* free parameters.

5.3 Scaling limit via pooling

Let $S_{\max} := 2(N-1)$ and $X := S/S_{\max} \in [0,1]$.

Proposition 5.3 (Continuum limit)**.** *As $N \to \infty$, X converges in distribution to a continuous law on $(0,1)$ with CDF and density*

$$F(x) = x(1 - \log x), \qquad f(x) = -\log x, \qquad x \in (0,1).$$

Moreover, the moments satisfy $\mathbb{E}[X^k] = \int_0^1 x^k(-\log x)\,dx = \frac{1}{(k+1)^2}$, in particular

$$\mathbb{E}[X] = \frac{1}{4}, \qquad \mathrm{Var}(X) = \frac{7}{144}.$$

Proof. For $s = \lfloor x S_{\max} \rfloor$ we have $H_{N-1} - H_{\lfloor s/2 \rfloor} \sim \log\big((N-1)/\lfloor s/2 \rfloor\big) \sim \log(1/x)$. From Proposition 5.1 this yields the Riemann-sum limit $F(x) = \int_0^x (-\log u)\,du = x(1 - \log x)$, and differentiation gives $f(x) = -\log x$. The moments follow from $\int_0^1 x^k(-\log x)\,dx = (k+1)^{-2}$. $\square$

Remark 5.4 (Stochastic interpretation)**.** Draw $U \sim \mathrm{Unif}(0,1)$ and, *conditional* on $U = u$, let $X \mid (U = u) \sim \mathrm{Unif}(0,u)$. Then the unconditional density of X is $f(x) = -\log x$ (the *continuous mixture of uniforms*).

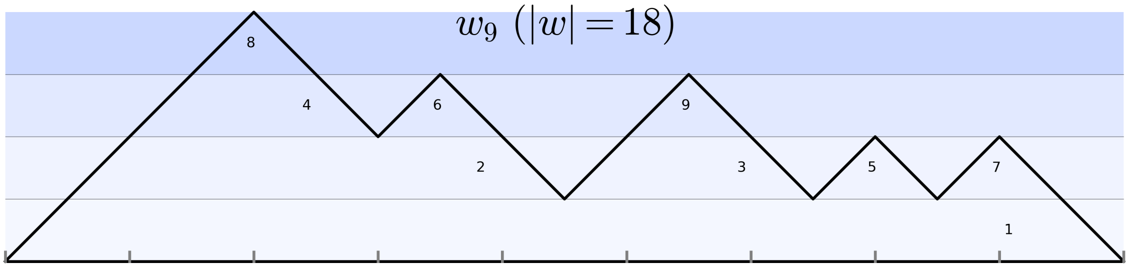

Figure 10: Dyck path for $n = 9$ with numbers at corresponding levels

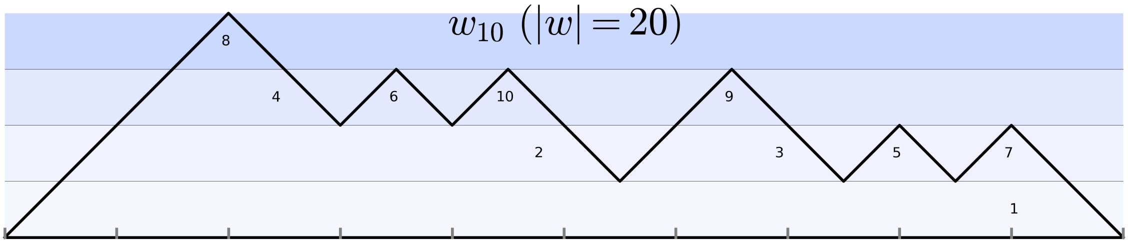

Figure 11: Dyck path for $n = 10$ with numbers at corresponding levels

6 Commutativity of admissible insertions and confluence to T_N

We recall our convention: for $m \geq 2$, $P_1(m)$ denotes the largest prime divisor of m, with $P_1(1) := 1$. For $N \in \mathbb{N}$, the finite tree T_N has vertex set $\{1, 2, \dots, N\}$, root 1, and edges

$$m \longrightarrow mp \qquad \text{for primes } p \text{ such that } \; P_1(m) \leq p \leq \left\lfloor \frac{N}{m} \right\rfloor,$$

with the children of m ordered increasingly by p.

6.1 Parent, child set, and prime rank

For $m \geq 2$ we set

$$\mathrm{par}(m) \; := \; \frac{m}{P_1(m)} \in \mathbb{N}. \tag{2}$$

Then in T_N the vertex m is *exactly* the child of $\mathrm{par}(m)$, and the ordered children of $\mathrm{par}(m)$ are

$$\mathrm{Ch}_N(\mathrm{par}(m)) \; = \; \Big\{ \; \mathrm{par}(m) \cdot p : \; p \text{ prime}, \quad P_1(\mathrm{par}(m)) \leq p \leq \left\lfloor \frac{N}{\mathrm{par}(m)} \right\rfloor \; \Big\}, \tag{3}$$

sorted by increasing p.

It is convenient to record the *prime rank* of m among its siblings:

$$\mathrm{rank}_N(m) \; := \; 1 + \#\Big\{ q \text{ prime} : \; P_1(\mathrm{par}(m)) \; \leq \; q \; < \; P_1(m) \Big\}. \tag{4}$$

Thus $\mathrm{rank}_N(m)$ is the index of the prime $P_1(m)$ within the sorted list of allowable primes for the parent $\mathrm{par}(m)$; note that $\mathrm{rank}_N(m)$ does not depend on the upper cutoff N as long as $N \geq m$.

6.2 Admissible insertion at a fixed horizon

Fix a finite rooted ordered tree U that is a valid truncation of some T_M (for some M), and fix a target horizon $N \geq 1$. For $x \in \mathbb{N}$, we say that x *is N-admissible in U* and write $x \in \mathrm{Adm}_N(U)$ if

$$\mathrm{par}(x) \in U \qquad \text{and} \qquad x \leq N. \tag{5}$$

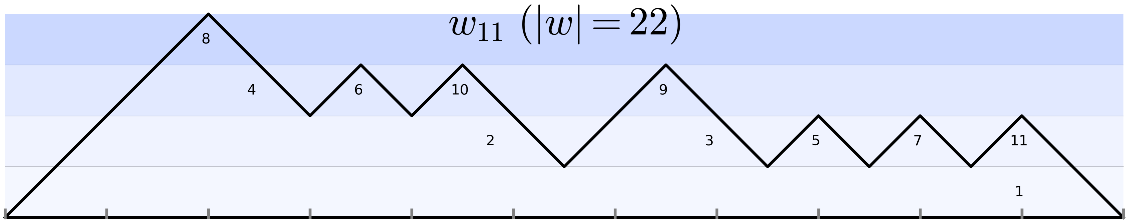

Figure 12: Dyck path for $n = 11$ with numbers at corresponding levels

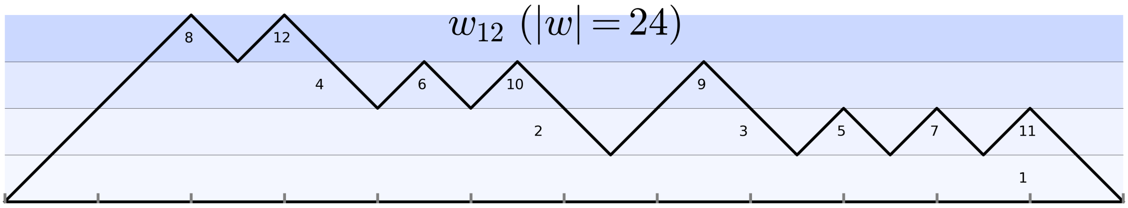

Figure 13: Dyck path for $n = 12$ with numbers at corresponding levels

In that case we define the *insertion* operation

$$U \oplus_N x$$

to be the rooted ordered tree obtained from U by adjoining the new vertex x as the child of $\mathrm{par}(x)$ and placing it among the children of $\mathrm{par}(x)$ at the unique position prescribed by the prime rank (4), i.e. so that the child order remains increasing in the underlying prime parameter.

Remark 6.1. If $U = T_n$ and $x = n+1$ then x is $(n+1)$-admissible and $T_n \oplus_{n+1} (n+1) = T_{n+1}$. For general $x > n$ one must raise the horizon to $N \geq x$; the operation $T_n \oplus_N x$ is well-defined but (unless $N = x$ and $x = n+1$) does not yield the full T_N by itself, because other admissible nodes $y \leq N$ might still be missing.

6.3 Local commutativity for admissible insertions

We first show that two admissible insertions that do not stand in an ancestor–descendant relation *commute*.

Definition 6.2 (Ancestor partial order)**.** For $u, v \in \mathbb{N}$ we write $u \preccurlyeq v$ ("u is an ancestor of v") if v can be reached from u by a (possibly empty) sequence of valid prime multiplications with nondecreasing primes, i.e. if there exists $k \geq 0$ and primes $q_1 \leq \cdots \leq q_k$ such that

$$v \;=\; u \cdot \prod_{i=1}^{k} q_i \quad \text{and} \quad q_1 \;\geq\; P_1(u),$$

with the convention that $k = 0$ means $v = u$. We write $u \prec v$ for $u \preccurlyeq v$ and $u \neq v$.

Lemma 6.3 (Local commutativity)**.** *Let U be a valid finite truncation and fix N. Suppose $a, b \in \mathrm{Adm}_N(U)$, i.e. $\mathrm{par}(a), \mathrm{par}(b) \in U$ and $a, b \leq N$, and assume $a \not\preccurlyeq b$ and $b \not\preccurlyeq a$ (incomparable in the ancestor order). Then both $b \in \mathrm{Adm}_N(U \oplus_N a)$ and $a \in \mathrm{Adm}_N(U \oplus_N b)$, and*

$$(U \oplus_N a) \oplus_N b \;=\; (U \oplus_N b) \oplus_N a. \tag{6}$$

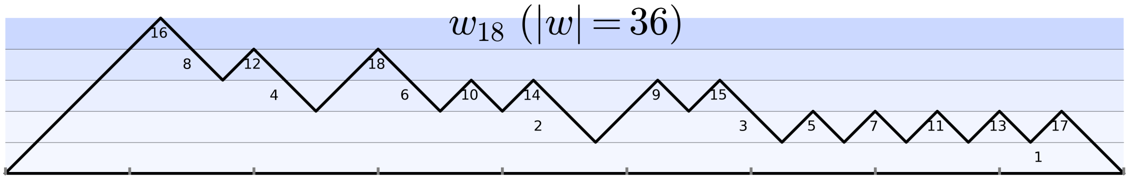

Figure 14: Dyck path for $n = 18$ with numbers at corresponding levels

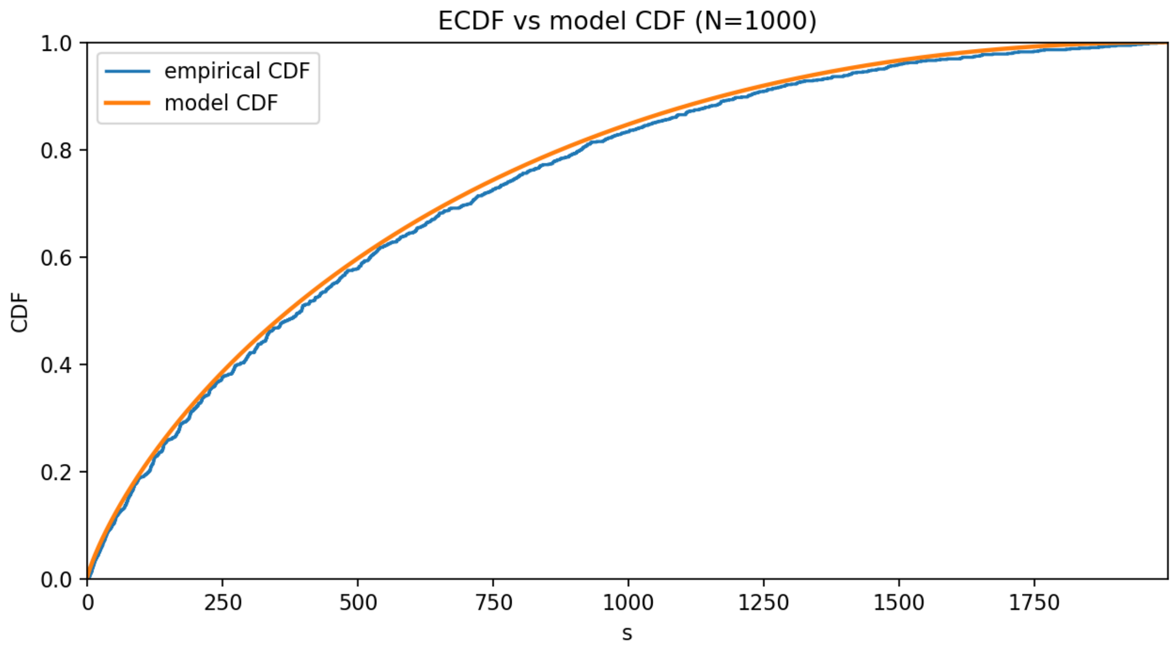

Figure 15: Empirical CDF (pooled indices S for $2 \leq n \leq 1000$) versus the discrete mixture-of-uniforms model CDF from Proposition 5.1.

Proof. We consider two cases.

Case 1: $\mathrm{par}(a) \neq \mathrm{par}(b)$. Inserting a only modifies the child list of $\mathrm{par}(a)$ by appending a at the rank (4); in particular it does not create or destroy $\mathrm{par}(b)$ nor affect its children. Hence b remains N-admissible after inserting a, and symmetrically a remains admissible after inserting b. Moreover, the two operations affect disjoint parts of the tree (two different child lists), so the final tree does not depend on the order: (6) holds.

Case 2: $\mathrm{par}(a) = \mathrm{par}(b) =: m$. Let $p_a := P_1(a)$ and $p_b := P_1(b)$. By admissibility we have $P_1(m) \leq p_a, p_b \leq \lfloor N/m \rfloor$. The insertion $U \oplus_N a$ places a into the sorted child list of m at the unique position determined by p_a; inserting b afterwards places b into the *same* sorted list at the position determined by p_b. Thus the final child list of m is the same multiset of children $\{m \cdot p : \ p \in \mathcal{P}\}$ with $\mathcal{P}$ the set of primes in $[P_1(m), \lfloor N/m \rfloor]$ augmented by $\{p_a, p_b\}$ and ordered increasingly; hence the result does not depend on whether we insert a first or b first. All other vertices and child lists are unaffected in either order, so (6) follows. □

Remark 6.4 (Necessity of incomparability)**.** If $a \prec b$ then $\mathrm{par}(b)$ does *not* belong to U unless a (and all its ancestors up to $\mathrm{par}(b)$) has been inserted; hence b may fail to be admissible before inserting a. In this sense, the partial order $\preccurlyeq$ encodes precedence constraints that must be respected by any valid insertion schedule.

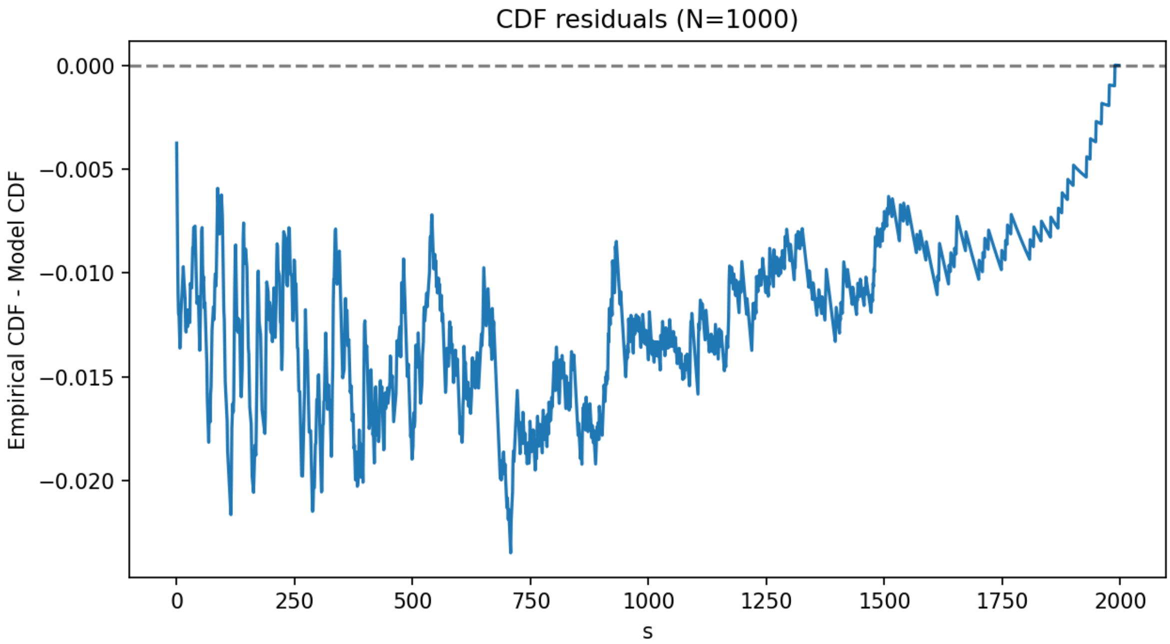

Figure 16: CDF residuals: empirical CDF minus model CDF for the pooled S at N=1000.

6.4 Global confluence: any linear extension yields T_N

Fix $n < N$ and set

$$A \;:=\; \{n+1, n+2, \ldots, N\}. \tag{7}$$

Equip A with the restriction of the ancestor partial order $\preccurlyeq$ of Definition 6.2. A *valid schedule* is a bijection $\sigma : \{1, \ldots, |A|\} \to A$ such that

$$u \preccurlyeq v \ \text{ in } A \quad \Rightarrow \quad \sigma^{-1}(u) \le \sigma^{-1}(v),$$

i.e. σ is a linear extension of $(A, \preccurlyeq)$. Given such σ, define the iterative insertion

$$U_0 := T_n, \qquad U_k := U_{k-1} \oplus_N \sigma(k) \quad (k = 1, \ldots, |A|). \tag{8}$$

Theorem 6.5 (Confluence to T_N)**.** *For every valid schedule σ as above, the tree $U_{|A|}$ defined by (8) is equal (as an ordered rooted tree) to T_N. In particular, if σ and τ are two linear extensions of $(A, \preccurlyeq)$, then the outcomes coincide:*

$$\mathrm{Fold}(T_n, \sigma) \;=\; \mathrm{Fold}(T_n, \tau) \;=\; T_N.$$

Proof. We proceed by induction on $|A|$.

Base $|A| = 0$. Then $N = n$ and $U_0 = T_n = T_N$.

Inductive step. Assume the claim for all target sets of size $< |A|$. Let M be the set of $\preccurlyeq$-minimal elements of A. Every valid schedule starts with some $x \in M$. Fix any $x \in M$. Then $\mathrm{par}(x) \in T_n$ (because x has no strict ancestor in A), hence x is N-admissible in T_n and $U_1 = T_n \oplus_N x$ is well-defined.

Consider the reduced target set $A' := A \setminus \{x\}$ with the induced partial order. Any linear extension σ of $(A, \preccurlyeq)$ with $\sigma(1) = x$ restricts to a linear extension σ' of $(A', \preccurlyeq)$, and the iterative fold (8) for $k \ge 2$ coincides with the fold that starts from U_1 and inserts the $\sigma'(1), \ldots, \sigma'(|A'|)$ in that order. By the induction hypothesis applied to U_1 and the target A', this results in T_N.

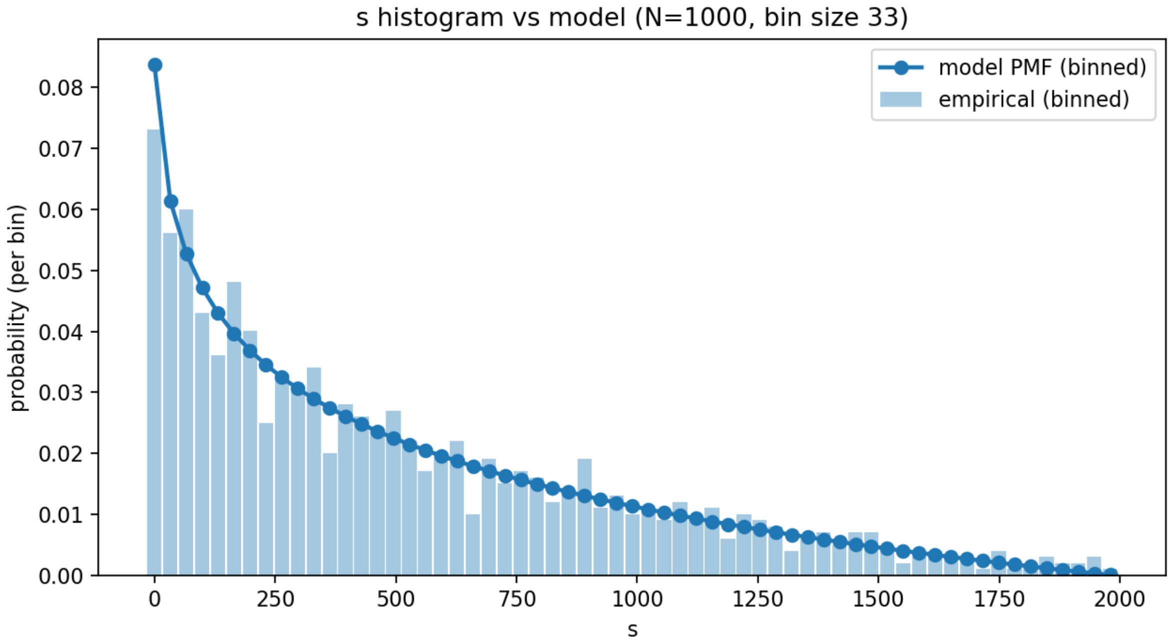

Figure 17: Histogram of pooled S (bin size 33) versus the binned model PMF from Proposition 5.1 (N=1000).

It remains to check that the particular choice of the first minimal element does not matter. If $x, y \in M$ are two distinct minimal elements, then x and y are incomparable in $\preccurlyeq$; by Lemma 6.3 we have local commutativity

$$(T_n \oplus_N x) \oplus_N y \;=\; (T_n \oplus_N y) \oplus_N x.$$

Therefore any two valid schedules that differ only by swapping the first two entries $x, y \in M$ lead to the same U_2; iterating this argument and using induction on the remaining $|A| - 2$ insertions shows that *all* valid schedules yield the same final tree, which must be T_N by vertex count and the defining edge/ordering rules. This completes the induction. □

6.5 Worked example: swapping 5 and 6 at horizon $N = 6$

Start from T_4 (vertices $\{1, 2, 3, 4\}$). We have

$$\mathrm{par}(5) = \frac{5}{P_1(5)} = \frac{5}{5} = 1, \qquad P_1(5) = 5,$$

so 5 is a new child of the root. Also

$$\mathrm{par}(6) = \frac{6}{P_1(6)} = \frac{6}{3} = 2, \qquad P_1(6) = 3,$$

so 6 is a new child of 2. Both parents 1 and 2 already belong to T_4, hence at horizon $N = 6$ we have $\{5, 6\} \subset \mathrm{Adm}_6(T_4)$, and 5 and 6 are incomparable in the ancestor order. Lemma 6.3 yields

$$(T_4 \oplus_6 6) \oplus_6 5 \;=\; (T_4 \oplus_6 5) \oplus_6 6.$$

By Theorem 6.5 the common result is T_6; concretely, inserting 6 first modifies only the child list of 2 by appending $2 \cdot 3$, while inserting 5 first modifies only the child list of 1 by appending the prime child 5. The two modifications are independent and the final ordered child lists coincide with those in T_6.

7 The limit tree T_∞

In this section we formalize the category in which our prime–multiplication trees live, construct the direct (colimit) limit object T_∞, relate it to the finite trees T_n, and finally give an explicit set–theoretic description of T_∞.

7.1 A category of rooted ordered trees and embeddings

Definition 7.1 (Category $\mathsf{Tree}_{\mathrm{emb}}$)**.** An object of $\mathsf{Tree}_{\mathrm{emb}}$ is a (possibly countably infinite) rooted, ordered tree $T = (V, E, \rho, \preceq)$ where

- V is a set of vertices, $E \subset V \times V$ is a set of directed edges forming a tree oriented away from the root $\rho \in V$;
- for each vertex $v \in V$, the set of children $\mathrm{Ch}(v) := \{w : (v, w) \in E\}$ is endowed with a total order $\preceq_v$ ("plane"/ordered tree).

A morphism $f : T \to T'$ in $\mathsf{Tree}_{\mathrm{emb}}$ is an *embedding*, i.e. an injective map $f : V \to V'$ such that

(i) $f(\rho) = \rho'$ (root-preserving);

(ii) $(v, w) \in E \iff (f(v), f(w)) \in E'$ (edge- and ancestor-preserving);

(iii) for each $v \in V$, the order on $\mathrm{Ch}(v)$ is preserved: if $x \preceq_v y$ then $f(x) \preceq'_{f(v)} f(y)$.

Remark 7.2. In this category, directed systems of trees with embedding transition maps have concrete colimits given by directed unions: since the morphisms are injective and order/edge preserving, the universal object is obtained by taking the union of vertex and edge sets and the induced child orders.

7.2 The direct limit tree and its relation to T_n

Recall T_n is the rooted ordered tree on vertices $\{1, 2, \ldots, n\}$ with root 1, where a vertex m has as children the numbers $mp \le n$ for primes p satisfying $p \ge P_1(m)$ (with the convention $P_1(1) = 1$), ordered by increasing p.

Proposition 7.3 (Directed system (T_n))**.** *For each n, the inclusion of vertex sets $\{1, \ldots, n\} \hookrightarrow \{1, \ldots, n+1\}$ induces an embedding*

$$i_{n,n+1} : T_n \hookrightarrow T_{n+1}$$

in $\mathsf{Tree}_{\mathrm{emb}}$. Hence $(T_n, i_{n,n+1})_{n\ge1}$ is a directed system in $\mathsf{Tree}_{\mathrm{emb}}$.

Definition 7.4 (Direct limit T_∞)**.** The *limit tree* T_∞ is the colimit of the directed system $(T_n, i_{n,n+1})$ in $\mathsf{Tree}_{\mathrm{emb}}$:

$$T_\infty := \varinjlim (T_n, i_{n,n+1}).$$

Concretely (Remark above), T_∞ is obtained by taking the directed union of the vertex and edge sets and the induced child orders, and the canonical embeddings $j_n : T_n \to T_\infty$ are the inclusions.

Proposition 7.5 (Relationship between T_∞ and T_n)**.** *For each n, T_n is precisely the induced finite rooted ordered subtree of T_∞ on the vertex set $\{1, 2, \ldots, n\}$. Equivalently, $j_n : T_n \to T_\infty$ identifies T_n with that induced subtree. Moreover, $(T_\infty, (j_n))$ satisfies the universal property of the colimit: for any cocone (S, ϕ_n) with $\phi_{n+1} \circ i_{n,n+1} = \phi_n$, there exists a unique morphism $u : T_\infty \to S$ with $u \circ j_n = \phi_n$.*

7.3 An explicit set–theoretic description of T_∞

Definition 7.6 (Explicit model of T_∞)**.** Let $\mathbb{P}$ denote the set of prime numbers and $P_1 : \mathbb{N} \to \mathbb{N}$ the map sending $m \geq 2$ to its largest prime factor and $P_1(1) := 1$. Define the rooted ordered tree

$$T_\infty = (V, E, \rho, \preceq)$$

as follows:

- *Vertices:* $V = \mathbb{N} = \{1, 2, 3, \dots\}$.
- *Root:* $\rho = 1$.
- *Edges:* For each $m \in \mathbb{N}$, its (possibly infinite) set of children is

  $$\mathrm{Ch}(m) \ := \ \{\, mp : \ p \in \mathbb{P}, \ p \geq P_1(m)\,\},$$

  and $E = \{(m, mp) : \ p \in \mathbb{P}, \ p \geq P_1(m)\}$.
- *Child order:* The children $\mathrm{Ch}(m)$ are totally ordered by the value of the prime, i.e. if $p < q$ then $mp \prec mq$.

Proposition 7.7. *The explicit tree of Definition 7.6 is isomorphic (in* $\mathsf{Tree}_{\mathrm{emb}}$*) to the colimit* T_∞ *of Definition 7.4. Under this identification, for each n the finite tree T_n is the induced ordered subtree of T_∞ on the vertex set* $\{1, \dots, n\}$*.*

Proof sketch. By construction, T_n has vertex set $\{1, \dots, n\}$ and edges (m, mp) whenever $mp \leq n$ and $p \geq P_1(m)$. Taking the directed union in n yields precisely the edge set in Definition 7.6, and the child orders are compatible (increasing primes) and hence induce the stated total orders on $\mathrm{Ch}(m)$. The universal property follows from the standard colimit property for nested embeddings. □

Remark 7.8 (Local finiteness and truncations)**.** Note that T_∞ has countably infinite outdegree at every vertex (all sufficiently large primes are allowed), hence it is not locally finite. For any finite cutoff n, the truncation obtained by deleting vertices $> n$ recovers T_n; for any prime cutoff B, the induced subtree on edges with child-prime $\leq B$ is finite over each vertex and stabilizes as $n \to \infty$.

8 Abstract properties of the limit tree T_∞

Recall the explicit model of the limit tree $T_\infty = (V, E, \rho, \preceq)$ from Definition 7.6: $V = \mathbb{N}$, $\rho = 1$, and for each $m \in \mathbb{N}$ the set of children is

$$\mathrm{Ch}(m) = \{\, mp : \ p \text{ prime}, \ p \geq P_1(m)\,\},$$

where $P_1(m)$ denotes the largest prime factor of m (with the convention $P_1(1) = 1$); the children are ordered by increasing prime p.

8.1 Unique path property: arithmetic determinism

Proposition 8.1 (Unique path from the root)**.** *For every node $m \in \mathbb{N}$, there exists a* unique *simple path in T_∞ from the root 1 to m.*

Proof. *Existence.* Write the prime factorization of m as

$$m = \prod_{i=1}^{k} p_i \quad \text{with} \quad p_1 \le p_2 \le \cdots \le p_k.$$

Define the chain $m_0 := 1$, $m_i := \prod_{j=1}^{i} p_j$ for $1 \le i \le k$. Then $m_i \to m_{i+1} = m_i \cdot p_{i+1}$ is an edge of T_∞ because $P_1(m_i) = p_i$ and $p_{i+1} \ge p_i$ by construction. Hence $1 = m_0 \to m_1 \to \cdots \to m_k = m$ is a valid path.

Uniqueness. Let $1 = n_0 \to n_1 \to \cdots \to n_\ell = m$ be any path in T_∞. By definition of edges, for each i there exists a prime q_{i+1} such that $n_{i+1} = n_i \cdot q_{i+1}$ and $q_{i+1} \ge P_1(n_i)$. Therefore $m = \prod_{j=1}^{\ell} q_j$, so the multiset $\{q_1, \dots, q_\ell\}$ equals the multiset of prime factors of m. Moreover the monotonicity constraint implies $q_{i+1} \ge P_1(n_i) = \max\{q_1, \dots, q_i\}$, hence $q_1 \le q_2 \le \cdots \le q_\ell$. Since there is only one nondecreasing listing of a fixed multiset of primes, the path is uniquely determined and coincides with the one constructed above. $\square$

Corollary 8.2 (Acyclicity and connectedness)**.** *T_∞ is a connected acyclic graph; in particular it is a (rooted, ordered) tree.*

Proof. By Proposition 8.1, every vertex is joined to the root by a path (connectedness) and there is a unique simple path between any two vertices (no cycles). $\square$

8.2 Recursive self-similarity

For $m \in \mathbb{N}$, write $T_\infty(m)$ for the full rooted ordered subtree of T_∞ induced by the descendants of m.

Proposition 8.3 (Self-similarity via prime filtering)**.** *Let $r := P_1(m)$. Consider the auxiliary rooted ordered tree $T'_\infty(r)$ defined by:*

- *vertices $V' = \{\, k \in \mathbb{N} : \text{ every prime factor of } k \text{ is } \ge r \,\}$ with root 1;*
- *edges $k \to k \cdot p$ for primes $p \ge \max\{r, P_1(k)\}$, ordered by increasing p.*

Then the map

$$\phi : T'_\infty(r) \longrightarrow T_\infty(m), \qquad \phi(k) = m \cdot k,$$

is an isomorphism of rooted ordered trees.

Proof. *Well-definedness and bijectivity.* If $k \in V'$, then every prime factor of k is $\ge r = P_1(m)$, hence $P_1(mk) = \max\{P_1(m), P_1(k)\} = P_1(k)$, so $\phi(k) = mk$ is a descendant of m. Conversely, any descendant x of m has the form $x = m \cdot k$ with all prime factors of $k \ge P_1(m)$, thus $k \in V'$. Hence ϕ is a bijection $V' \to V(T_\infty(m))$.

Edge preservation. In $T'_\infty(r)$, we have an edge $k \to k \cdot p$ iff $p \ge \max\{r, P_1(k)\}$. Since $r = P_1(m)$ and all primes of k are $\ge r$, we have $P_1(mk) = P_1(k)$. In $T_\infty(m)$, there is an edge $\phi(k) = mk \to \phi(k \cdot p) = mk \cdot p$ iff $p \ge P_1(mk) = P_1(k)$. Thus the edge condition is identical under ϕ.

Order preservation. Children are ordered by increasing p in both trees; ϕ leaves p unchanged, hence preserves the child order. $\square$

8.3 Order structure vs. divisibility

Let $u \preccurlyeq v$ denote the ancestor relation in T_∞ (i.e. u lies on the unique root-to-v path).

Proposition 8.4 (Ancestor implies divisibility; strictness)**.** *If $u \preccurlyeq v$ in T_∞, then $u \mid v$. The converse fails in general: there exist $u \mid v$ with $u \not\preccurlyeq v$.*

Proof. If $u \preccurlyeq v$, the unique path $u = n_0 \to n_1 \to \cdots \to n_t = v$ multiplies by primes at each step, hence $v = u \cdot \prod_{i=1}^{t} q_i$ and $u \mid v$.

For failure of the converse, take $u = 3$ and $v = 12$. Although $3 \mid 12$, the unique path to 12 is $1 \to 2 \to 4 \to 12$ (primes $2, 2, 3$ in nondecreasing order). There is no edge $3 \to 6$ using prime 2 because the edge rule requires a prime $\geq P_1(3) = 3$. Thus $3 \not\preccurlyeq 12$. □

The ancestor relation admits a precise arithmetic characterization.

Proposition 8.5 (Characterization of ancestry)**.** *Let $u, v \in \mathbb{N}$. Write $v = u \cdot t$ if $u \mid v$, and list the prime factors of t in nondecreasing order as $q_1 \leq \cdots \leq q_s$ (with $t = 1$ interpreted as $s = 0$). Then*

$$u \preccurlyeq v \iff u \mid v \text{ and } q_1 \geq P_1(u) \text{ (vacuously true if } s = 0).$$

Equivalently, $u \preccurlyeq v$ iff $u \mid v$ and every prime factor of v/u is at least $P_1(u)$.

Proof. $\Rightarrow$: If $u \preccurlyeq v$, then along the unique path from u to v we multiply by a nondecreasing sequence of primes each $\geq P_1(u)$; hence $u \mid v$ and all primes dividing v/u are $\geq P_1(u)$.

$\Leftarrow$: If $u \mid v$ and every prime factor of $t = v/u$ is $\geq P_1(u)$, list them as $q_1 \leq \cdots \leq q_s$. Then the chain $u \to uq_1 \to uq_1q_2 \to \cdots \to v$ is a valid path because at each step the edge rule requires multiplying by a prime at least as large as the current largest prime factor, and $q_1 \geq P_1(u)$ while $q_{i+1} \geq q_i$. By Proposition 8.1, this is the unique path; hence $u \preccurlyeq v$. □

8.4 Topological and degree properties

Proposition 8.6 (Basic graph-theoretic properties)**.** *The tree T_∞ is connected and acyclic (Corollary 8.2). Moreover, for every vertex m one has $|\mathrm{Ch}(m)| = \aleph_0$, i.e. the out-degree of every vertex is countably infinite. In particular T_∞ is not locally finite.*

Proof. Connectedness and acyclicity were proved in Corollary 8.2. For the degree, fix m. Then $\mathrm{Ch}(m) = \{ mp : p \text{ prime}, \ p \geq P_1(m) \}$. By Euclid's theorem there are infinitely many primes exceeding any given bound, hence $\mathrm{Ch}(m)$ is infinite; in fact it is countably infinite. Since every vertex has infinite out-degree, the graph is not locally finite. □

Remark 8.7 (Consequences for random processes)**.** The failure of local finiteness implies that certain probabilistic processes (e.g. simple random walk started at the root) behave differently than on locally finite trees; for instance, transition probabilities out of a vertex are not normalizable by uniform choice over children. One can nevertheless build natural dynamics using size-biased or intensity-biased selections over primes $p \geq P_1(m)$.

9 An abstract divisor–ordered setting for the factorization tree

We now isolate the minimal axioms under which our "prime-multiplication tree" construction (§**??**) still works. The key point is that we only ever used: (i) divisibility, (ii) a notion of "prime atom", (iii) a total order on those atoms, (iv) and the rule that you are only allowed to multiply by atoms that are not smaller than your current largest atom.

This can be formulated abstractly without reference to the integers.

9.1 Ordered unique-factorization monoids

Definition 9.1 (Ordered UFM)**.** An *ordered unique-factorization monoid* (ordered UFM) is a quadruple

$$(X, \ \cdot, \ 1, \ \leq)$$

satisfying the following axioms:

(A1) **Commutative cancellative monoid.** $(X, \cdot, 1)$ is a commutative monoid with identity 1. Cancellation holds: if $a \cdot b = a \cdot c$ then $b = c$.

(A2) **Atoms, unique factorization.** There is a distinguished subset $\mathcal{P} \subset X \setminus \{1\}$ of *atoms* (think: "primes") such that: every $x \in X$ can be written as a finite product

$$x \; = \; \prod_{p \in \mathcal{P}} p^{e_p(x)},$$

where $e_p(x) \in \mathbb{N}$ and $e_p(x) = 0$ for all but finitely many p. Moreover, this multiset $\{(p, e_p(x)) : p \in \mathcal{P}\}$ is unique. In particular, 1 is the empty product (all $e_p(1) = 0$).

(A3) **Divisibility.** Define $a \mid b$ iff $\exists t \in X$ with $b = a \cdot t$. This induces a partial order (the divisibility poset).

(A4) **Covers are "multiply by one atom".** If $x \prec y$ is a cover in the divisibility poset (i.e. $x \mid y$, $x \neq y$, and no z satisfies $x \mid z \mid y$ except $z = x, y$), then

$$y = x \cdot p \quad \text{for a } \textit{unique} \text{ atom } p \in \mathcal{P}.$$

Conversely, for every $x \in X$ and atom $p \in \mathcal{P}$, $x \prec x \cdot p$ is a cover.

(A5) **Total order on atoms.** We are given a total order $\leq$ on $\mathcal{P}$. We extend notation and also write $\leq$ for this order on atoms.[1]

(A6) **Least element.** 1 is the unique $|$-minimal element of X (and thus the root we will use).

Remark 9.2. Axioms (A2) and (A4) say abstractly: the divisibility poset $(X, |)$ looks like a free commutative monoid on the atom set $\mathcal{P}$, and each cover $x \prec y$ corresponds to "multiply x by exactly one more copy of a single atom". Axiom (A5) gives us a *total* order on atoms, so we can meaningfully say "the largest atom dividing x".

Definition 9.3 (Largest atom dividing an element)**.** For $x \neq 1$, define

$$P_1(x) \; := \; \max_{\leq} \{\, p \in \mathcal{P} : e_p(x) \geq 1 \,\} \; \in \mathcal{P},$$

i.e. the $\leq$-largest atom that divides x. By convention set $P_1(1) := 1$, where 1 is regarded as "smaller than every atom". This is well-defined because each x has only finitely many atoms with $e_p(x) > 0$, and $\leq$ is a total order on $\mathcal{P}$.

9.2 The abstract prime-restricted growth tree

Definition 9.4 (Global tree T_X)**.** Let $(X, \cdot, 1, \leq)$ be an ordered UFM. We define the rooted, ordered tree

$$T_X = (V, E, \rho, \preceq)$$

as follows:

- Vertices: $V := X$.
- Root: $\rho := 1$.
- Directed edges: for each $x \in X$, for each atom $p \in \mathcal{P}$ with $p \geq P_1(x)$ (in the total order on atoms), we draw a directed edge

$$x \longrightarrow x \cdot p.$$

[1] We do *not* require $\leq$ to be compatible with multiplication globally. We only require it to totally order the atoms.

- Child order: for a fixed parent x, we order its children $x \cdot p$ by the total order on atoms p. So "smaller atom first" means "left child first."

This is exactly the same rule we used over the integers: from x you are only allowed to multiply by atoms p that are *not smaller* than your current largest atom $P_1(x)$, and among those children we sort by p.

9.3 Uniqueness of paths and basic properties

Proposition 9.5 (Canonical nondecreasing-atom path)**.** *For every $x \in X$ there is a unique simple directed path in T_X from 1 to x.*

Proof. **Existence.** By unique factorization (A2), write

$$x \;=\; \prod_{p\in\mathcal{P}} p^{e_p(x)}.$$

List each atom p exactly $e_p(x)$ times, and sort these atoms in nondecreasing order with respect to $\leq$:

$$p_1 \;\leq\; p_2 \;\leq\; \cdots \;\leq\; p_k.$$

Now define

$$x_0 := 1, \qquad x_i := \Big(\prod_{j=1}^{i} p_j\Big) \quad (1 \leq i \leq k).$$

We claim $x_{i-1} \to x_i$ is an edge of T_X. Indeed, $x_i = x_{i-1} \cdot p_i$. Since $p_1 \geq P_1(1) = 1$, the first edge $1 \to p_1$ is allowed. For $i > 1$, the largest atom dividing x_{i-1} is exactly p_{i-1} because by construction we multiplied by atoms in nondecreasing order. Therefore $P_1(x_{i-1}) = p_{i-1} \leq p_i$. Thus $p_i \geq P_1(x_{i-1})$, so $x_{i-1} \to x_i$ is allowed by Definition 9.4. Hence

$$1 = x_0 \to x_1 \to \cdots \to x_k = x$$

is a directed path in T_X.

Uniqueness. Suppose we have *any* directed path in T_X,

$$1 = y_0 \to y_1 \to \cdots \to y_\ell = x.$$

By the edge rule, for each step there is some atom q_i such that

$$y_i = y_{i-1} \cdot q_i \quad \text{and} \quad q_i \geq P_1(y_{i-1}).$$

In particular q_i divides y_i. Since $P_1(y_{i-1})$ is the largest atom dividing y_{i-1}, and $q_i \geq P_1(y_{i-1})$, we get

$$q_1 \;\leq\; q_2 \;\leq\; \cdots \;\leq\; q_\ell \quad \text{in the total order on } \mathcal{P}.$$

Multiplying along the path,

$$x \;=\; y_\ell \;=\; q_1 q_2 \cdots q_\ell.$$

By unique factorization, the multiset $\{q_1, \ldots, q_\ell\}$ (counting multiplicities) must equal the multiset of atoms $\{p_1, \ldots, p_k\}$ from the sorted factorization of x above. Moreover, because both (q_i) and (p_i) are *nondecreasing* lists of the same multiset, they must agree *as sequences*, not just as multisets. Therefore $k = \ell$, and $y_i = x_i$ for all i. So the path we constructed is the *only* possible path. ☐

Corollary 9.6. *T_X is a connected acyclic directed graph with distinguished root 1 and a unique root-to-x path for each $x \in X$. In particular, T_X is an (infinite) rooted ordered tree.*

Proposition 9.7 (Ancestry vs. divisibility)**.** *Let $u, v \in X$. Then u is an ancestor of v in T_X (i.e. u lies on the unique path from 1 to v) if and only if*

$$u \mid v \quad \textit{and} \quad \textit{every atom dividing } v/u \textit{ is } \geq P_1(u).$$

Equivalently, write $v = u \cdot q_1 q_2 \cdots q_s$ with the atoms $q_1 \leq \cdots \leq q_s$. Then u is an ancestor of v iff $q_1 \geq P_1(u)$.

Proof. Exactly as in the integer setting. ($\Rightarrow$) If u is on the path $1 \to \cdots \to v$, then by concatenating the segment from u to v we see $v = u \cdot q_1 \cdots q_s$ for a weakly nondecreasing sequence of atoms q_i, each satisfying $q_i \geq P_1(\text{current node}) \geq P_1(u)$. Thus each prime factor of v/u is $\geq P_1(u)$. ($\Leftarrow$) Conversely, suppose $u \mid v$ and every atom of v/u is $\geq P_1(u)$. Sort those atoms nondecreasingly as $q_1 \leq \cdots \leq q_s$. Then, starting at u and multiplying successively by $q_1, \ldots, q_s$, the edge rule is always satisfied, so we build a directed path $u \to \cdots \to v$. By uniqueness of the root-to-v path, this shows u is on it. $\square$

9.4 Finite truncations and Dyck encoding

In applications we work not with all of X but with a *finite* "horizon"—for the integers, this was $\{1, 2, \ldots, n\}$.

Definition 9.8 (Finite horizon / truncation)**.** A finite subset $B \subset X$ is called a *divisibility ideal* if whenever $x \in B$ and $y \mid x$, then $y \in B$. Given such a B, we define $T_X[B]$ to be the induced rooted ordered subtree of T_X with vertex set B, keeping only edges $x \to x \cdot p$ that stay inside B.

For $X = \mathbb{N}$ and $B = \{1, \ldots, n\}$, this is precisely the finite tree T_n we studied. For any such finite rooted ordered tree, a standard depth-first traversal ("write 1 when you enter a node, write 0 when you finish its subtree") produces a Dyck word of length $2|B|$, and conversely any Dyck word of length $2|B|$ reconstructs that ordered rooted tree in $O(|B|)$ time. All proofs of correctness carry over verbatim because they use only tree structure, not arithmetic of $\mathbb{N}$.

9.5 Examples

We now list two concrete families that satisfy the axioms above and which can be visualized in computer algebra systems.

Example 1: The classical integer case. Take

$$X = \mathbb{N}_{\geq 1}, \quad \cdot = \text{ordinary multiplication}, \quad 1 = 1, \quad \mathcal{P} = \{\text{prime numbers}\},$$

and let $\leq$ be the usual order on primes ($2 < 3 < 5 < 7 < \cdots$). Then Axioms (A1)–(A6) hold: commutative cancellative monoid, unique factorization into primes, divisibility is the usual divisibility, covers are $m \prec mp$ for a prime p, the primes are totally ordered by their numeric size, and 1 is the global minimum. The resulting T_X is exactly the prime-multiplication tree T_∞ defined earlier: each node m has children $m \cdot p$ for primes $p \geq P_1(m)$, ordered increasingly by p.

Finite truncations $B = \{1, 2, \ldots, n\}$ give the finite trees T_n from §**??**.

Example 2: Monomials in two variables. Let X be the set of all monomials in two commuting variables x, y:

$$X = \{x^a y^b : a, b \in \mathbb{N}\}, \qquad (x^a y^b) \cdot (x^c y^d) := x^{a+c} y^{b+d}, \qquad 1 := x^0 y^0.$$

This is a commutative cancellative monoid. The atoms are

$$\mathcal{P} = \{x,\ y\}.$$

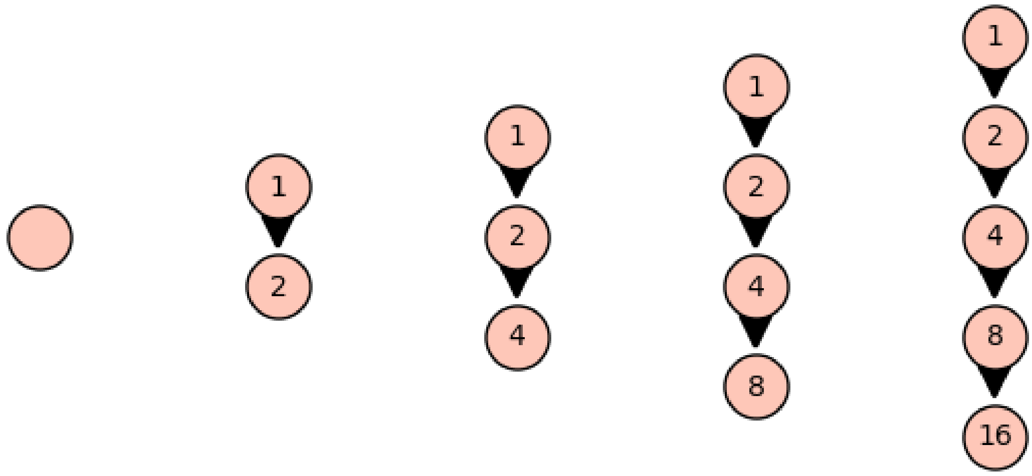

Figure 18: Trees at $\{2\}$

Every monomial $x^a y^b$ factors uniquely as $x^a y^b$, so unique factorization holds. Divisibility is

$$x^a y^b \mid x^c y^d \quad \iff \quad a \le c \text{ and } b \le d,$$

and covers are exactly "multiply by one more x" or "multiply by one more y". To make this an *ordered* UFM, we choose a total order on $\{x, y\}$. For instance, declare

$$x \;<\; y.$$

Then $P_1(x^a y^b)$ is:

- $P_1(1) = 1$;
- $P_1(x^a) = x$ if $b = 0$;
- $P_1(x^a y^b) = y$ if $b > 0$, because $y > x$.

The edge rule in T_X says: from $x^a y^b$ you may multiply by any atom $\ge P_1(x^a y^b)$. Concretely:

- From 1: $P_1(1) = 1$, so both x and y are allowed. Children: $1 \to x$, $1 \to y$, ordered as $x \prec y$.
- From x: $P_1(x) = x$, so you may multiply by any atom $\ge x$, i.e. x or y. Children: $x \to x^2$, $x \to xy$.
- From y: $P_1(y) = y$, so you may only multiply by atoms $\ge y$, i.e. only y. Child: $y \to y^2$.
- From xy: $P_1(xy) = y$, so only y is allowed. Child: $xy \to xy^2$.

This produces an infinite rooted ordered tree where "once you have used y, you are forever locked to y only," but along the x-only branch you can still branch into x or y.

A finite horizon is, for example,

$$B_d := \{x^a y^b : a + b \le d\},$$

which is clearly downward closed under divisibility. Then $T_X[B_d]$ is a finite ordered rooted tree you can draw in Sage.

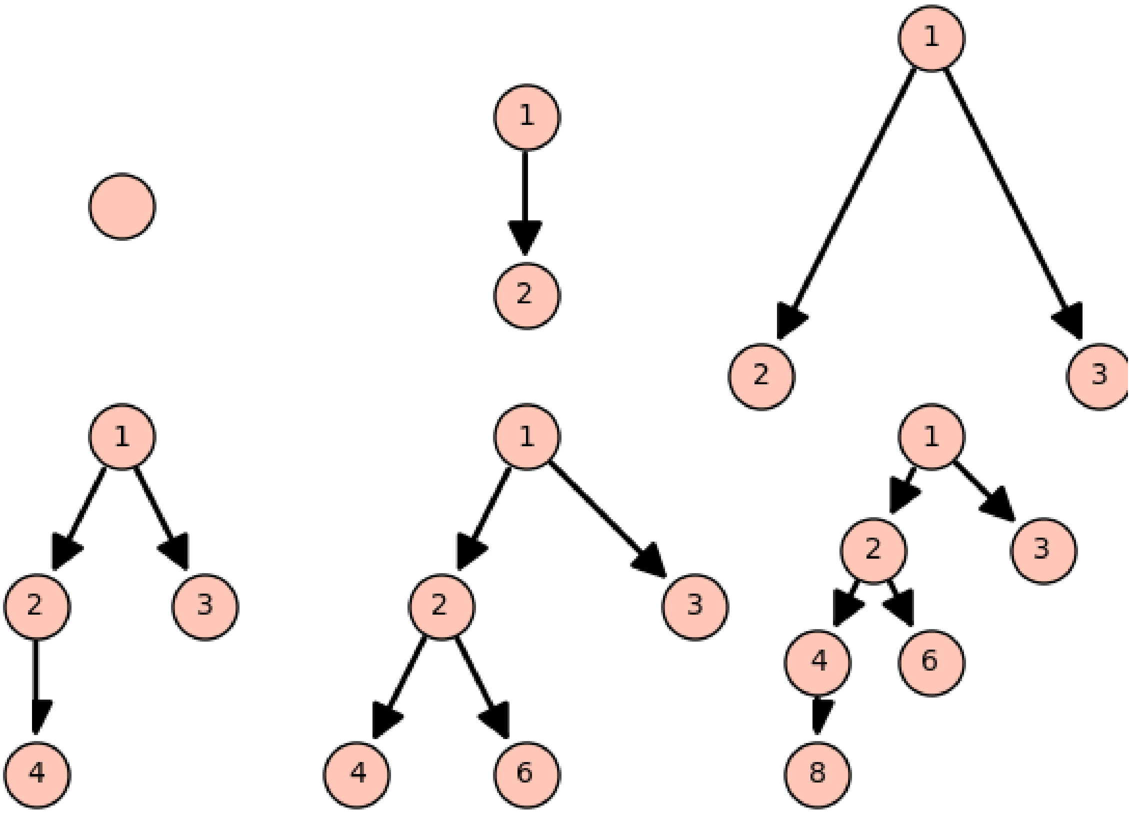

Figure 19: Trees at $\{2, 3\}$

Example 3: Monomials in three variables. Similarly, take $X = \{x^a y^b z^c : a, b, c \in \mathbb{N}\}$, $\mathcal{P} = \{x, y, z\}$, and impose a total order $x < y < z$. Then $P_1(x^a y^b z^c)$ is:

$$P_1(x^a y^b z^c) = \begin{cases} x & \text{if } b = c = 0, \\ y & \text{if } c = 0 \text{ but } b > 0, \\ z & \text{if } c > 0, \end{cases}$$

and edges from a monomial allow you to multiply only by atoms $\geq P_1$(that monomial). Again, choosing a finite "degree ball"

$$B_d := \{x^a y^b z^c : a + b + c \leq d\}$$

gives a finite tree.

In both monomial examples:

- Axiom (A1): obvious (commutative cancellative monoid under multiplication of monomials).
- (A2): unique factorization into variables x, y (or x, y, z) with exponents.
- (A3): divisibility is coordinatewise $\leq$.
- (A4): covers correspond to multiplying by exactly one more factor of one variable.
- (A5): we impose a total order on the variables, e.g. $x < y < z$.
- (A6): $1 = x^0 y^0 z^0$ is minimal under divisibility.

Therefore these monomial worlds are *valid ordered UFMs* in our sense, and T_X is well-defined and has unique root-to-node paths, just like in the integer case.

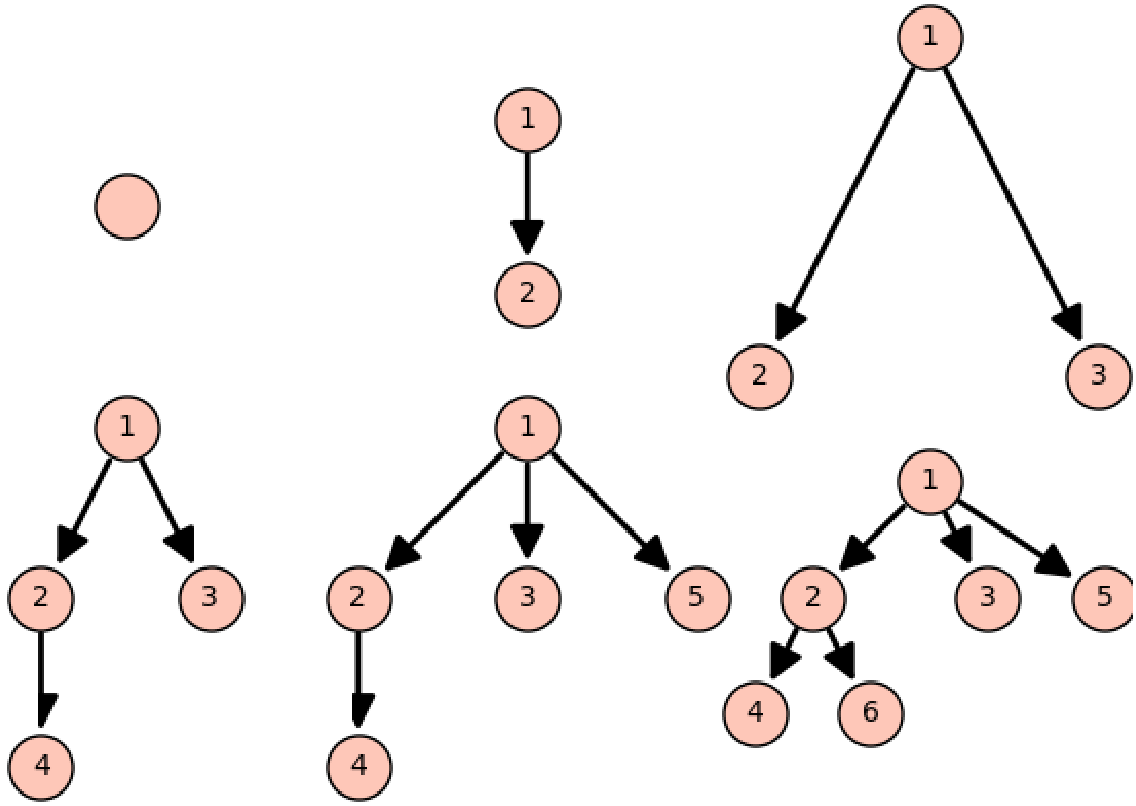

Figure 20: Trees at $\{2,3,5\}$

10 Rational functions attached to the trees

We work with the prime-multiplication tree described earlier. For a node v with numeric value $\mathrm{val}(v) \in \mathbb{N}$, and for $x > 0$, define recursively

$$p(v;x) \;:=\; \begin{cases} x^{\mathrm{val}(v)}, & v \text{ leaf}, \\ \dfrac{x^{\mathrm{val}(v)}}{\displaystyle\sum_{v \to w} p(w;x)}, & \text{otherwise}, \end{cases} \qquad p(T_n, x) := p(\mathrm{root}; x), \qquad p_\infty(x) := \lim_{n\to\infty} p(T_n, x).$$

We prove that $p_\infty(x) = 0$.

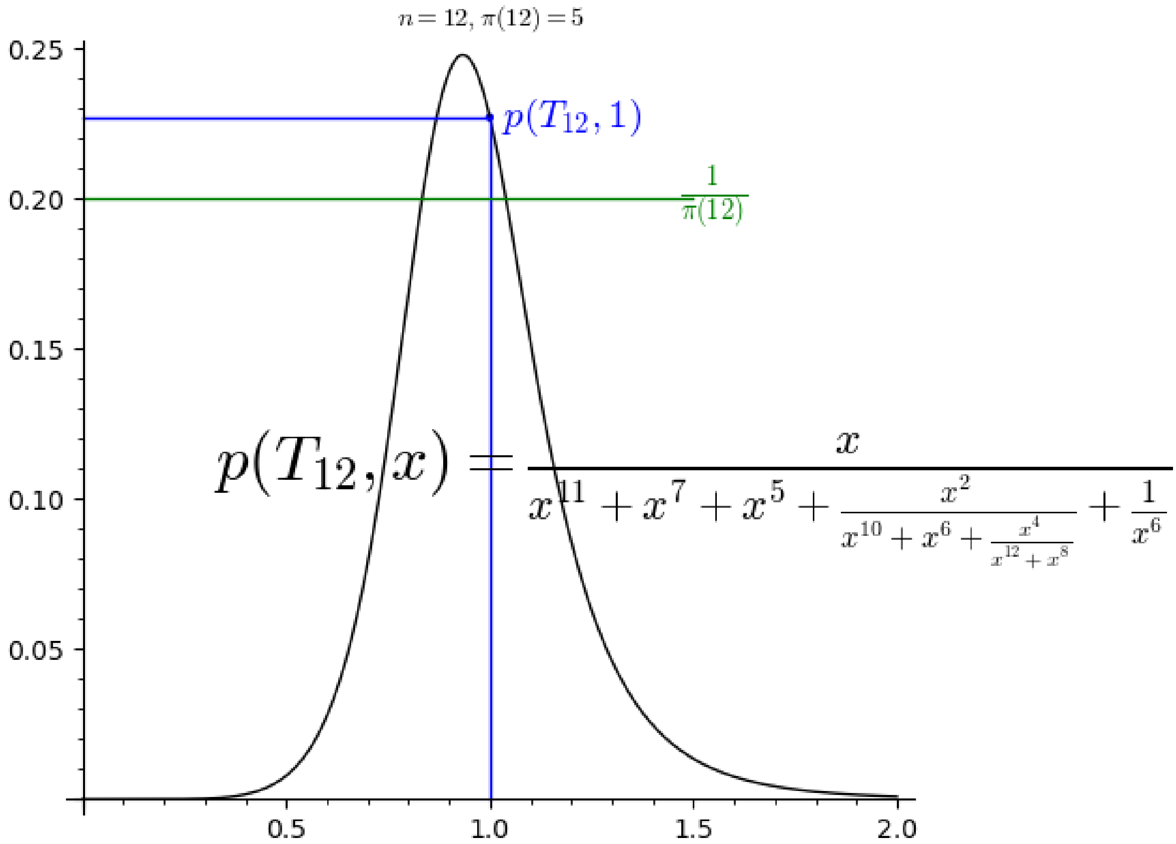

Figure 21: Plot of one function

10.1 Example values of $p(T_n, x)$

We list the exact expressions for $p(T_n, x)$ for $n = 1, \ldots, 20$:

$$n = 1 : x$$

$$n = 2 : \frac{1}{x}$$

$$n = 3 : \frac{x}{x^3 + x^2}$$

$$n = 4 : \frac{x}{x^3 + \frac{1}{x^2}}$$

$$n = 5 : \frac{x}{x^5 + x^3 + \frac{1}{x^2}}$$

$$n = 6 : \frac{x}{x^5 + x^3 + \frac{x^2}{x^6+x^4}}$$

$$n = 7 : \frac{x}{x^7 + x^5 + x^3 + \frac{x^2}{x^6+x^4}}$$

$$n = 8 : \frac{x}{x^7 + x^5 + x^3 + \frac{x^2}{x^6+\frac{1}{x^4}}}$$

$$n = 9 : \frac{x}{x^7 + x^5 + \frac{x^2}{x^6+\frac{1}{x^4}} + \frac{1}{x^6}}$$

$$n = 10 : \frac{x}{x^7 + x^5 + \frac{x^2}{x^{10}+x^6+\frac{1}{x^4}} + \frac{1}{x^6}}$$

$$n = 11 : \frac{x}{x^{11} + x^7 + x^5 + \frac{x^2}{x^{10}+x^6+\frac{1}{x^4}} + \frac{1}{x^6}}$$

$$n = 12 : \frac{x}{x^{11} + x^7 + x^5 + \frac{x^2}{x^{10}+x^6+\frac{x^4}{x^{12}+x^8}} + \frac{1}{x^6}}$$

$$n = 13 : \frac{x}{x^{13} + x^{11} + x^7 + x^5 + \frac{x^2}{x^{10}+x^6+\frac{x^4}{x^{12}+x^8}} + \frac{1}{x^6}}$$

$$n = 14 : \frac{x}{x^{13} + x^{11} + x^7 + x^5 + \frac{x^2}{x^{14}+x^{10}+x^6+\frac{x^4}{x^{12}+x^8}} + \frac{1}{x^6}}$$

$$n = 15 : \frac{x}{x^{13} + x^{11} + x^7 + x^5 + \frac{x^3}{x^{15}+x^9} + \frac{x^2}{x^{14}+x^{10}+x^6+\frac{x^4}{x^{12}+x^8}}}$$

$$n = 16 : \frac{x}{x^{13} + x^{11} + x^7 + x^5 + \frac{x^3}{x^{15}+x^9} + \frac{x^2}{x^{14}+x^{10}+x^6+\frac{x^4}{x^{12}+\frac{1}{x^8}}}}$$

$$n = 17 : \frac{x}{x^{17} + x^{13} + x^{11} + x^7 + x^5 + \frac{x^3}{x^{15}+x^9} + \frac{x^2}{x^{14}+x^{10}+x^6+\frac{x^4}{x^{12}+\frac{1}{x^8}}}}$$

$$n = 18 : \frac{x}{x^{17} + x^{13} + x^{11} + x^7 + x^5 + \frac{x^3}{x^{15}+x^9} + \frac{x^2}{x^{14}+x^{10}+\frac{x^4}{x^{12}+\frac{1}{x^8}}+\frac{1}{x^{12}}}}$$

11 Factorization trees, the $p(T_n, x)$ recursion, and the link to $\pi(n)$

This section distills the content of the MathOverflow question *"Factorization trees and (continued) fractions?"* and the answer by Weber into a self-contained proof of two statements:

(a) a recursion in n for the root weight $p(T_n, x)$ together with the branched "continued–fraction" representation coming from the tree structure;

(b) the exact identity $c(n) := \frac{1}{p(T_n,1)} = \sum_{p \le n} p(T_{n,p}, 1)$ and the bounds $\pi(n) - \pi(\sqrt{n}) \le c(n) \le \pi(n)$, which imply $c(n) \sim \pi(n)$.

11.1 Definitions and basic setup

For $m \ge 2$ let $P_1(m)$ denote the largest prime divisor of m, with the convention $P_1(1) = 1$. For integers $1 \le m \le n$ we define the rooted ordered tree $T_{n,m}$ as follows: the root is m; its children are the numbers mp with p prime and $P_1(m) \le p \le \lfloor n/m \rfloor$, ordered by increasing p. If no such prime exists, m is a leaf. We write $T_n := T_{n,1}$. For a node v with numeric value $\mathrm{val}(v)$ and parameter $x > 0$ set

$$p(T_{n,v}, x) \;:=\; \begin{cases} x^{\mathrm{val}(v)}, & \text{if } v \text{ is a leaf,} \\ \dfrac{x^{\mathrm{val}(v)}}{\displaystyle\sum_{v \to w} p(T_{n,w}, x)}, & \text{otherwise,} \end{cases} \qquad p(T_n, x) := p(T_{n,1}, x).$$

This is exactly the weighting used in the MO discussion.

11.2 Part (a): recursion in n and the branched continued fraction

Step 1: a locality lemma for the update $n \mapsto n{+}1$

Let $n \ge 1$ and write the prime factorization of $n{+}1$ in nondecreasing order as $n{+}1 = p_1 p_2 \cdots p_r$ ($r \ge 1$). The unique root-to-($n{+}1$) path in T_{n+1} is

$$1 = v_0 \to v_1 = p_1 \to v_2 = p_1 p_2 \to \cdots \to v_r = p_1 \cdots p_r = n{+}1.$$

Consequently, from T_n to T_{n+1} the *only* structural change in the child list of the root occurs *either* by appending the prime child $n{+}1$ (when $r = 1$), *or* by modifying the subtree rooted at the single prime child p_1 (when $r \ge 2$). No other root child is affected.

Step 2: the n–recursions for $p(T_n, x)$

Let

$$S_n(x) \;:=\; \sum_{\substack{p \le n \\ p \text{ prime}}} p(T_{n,p}, x), \qquad \text{so that} \qquad p(T_n, x) = \frac{x}{S_n(x)}.$$

Then:

- $n{+}1$ *prime.* A new root child $n{+}1$ appears and is a leaf, hence $S_{n+1}(x) = S_n(x) + x^{n+1}$ and therefore

$$\boxed{p(T_{n+1}, x) \;=\; \frac{x}{\frac{x}{p(T_n,x)} + x^{n+1}}\;.} \tag{9}$$

- $n+1$ *composite.* Let $s := \mathrm{spf}(n+1)$ be the smallest prime dividing $n+1$. By the locality lemma only the subtree at the root child s changes; all other prime children are unchanged. Thus

$$S_{n+1}(x) \;=\; S_n(x) - p(T_{n,s},x) + p(T_{n+1,s},x),$$

and hence

$$\boxed{\; p(T_{n+1},x) \;=\; \frac{x}{\frac{x}{p(T_n,x)} + \big(p(T_{n+1,s},x) - p(T_{n,s},x)\big)}\;.\;} \tag{10}$$

Both formulas follow directly from the definition of $p(\cdot)$ and the description of which child sums change when n increases by one.

Step 3: "branched continued–fraction" representation

The defining rule for an internal node m, $p(T_{n,m},x) = x^m / \sum_{m\to v} p(T_{n,v},x)$, together with the leaf rule $p(T_{n,m},x) = x^m$, shows by induction on the number of nodes in a finite subtree that $p(T_{n,m},x)$ is obtained from the leaf monomials by repeatedly applying the operations *(finite sum over children)* and *(take reciprocal and multiply by x^m).* Equivalently:

Lemma 11.1 (branched S-fraction along the tree)**.** *For every finite rooted subtree $U \subseteq T_n$, $p(U,x)$ is a rational expression built from the leaves $x^{\mathrm{val}(v)}$ by alternating finite sums and reciprocals according to the tree structure. In particular $p(T_n,x)$ admits a nested (branched) Stieltjes continued–fraction expansion dictated by T_n.*

11.3 Part (b): the identity for $c(n)$ and bounds by $\pi(n)$

Define

$$c(n) \;:=\; \frac{1}{p(T_n,1)}.$$

At $x=1$ the root formula reads $p(T_n,1) = 1/\sum_{p\le n} p(T_{n,p},1)$, because the root children are precisely the primes $\le n$. Therefore

$$\boxed{\; c(n) \;=\; \sum_{\substack{p\le n\\ p \text{ prime}}} p(T_{n,p},1)\,.\;} \tag{11}$$

Upper bound $c(n) \le \pi(n)$

We show $p(T_{n,m},1) \le 1$ for every non-leaf node m. By Bertrand's postulate there is a prime q with $\frac{n}{2m} < q \le \frac{n}{m}$. Then mq is a child of m and satisfies $mq > n/2$, hence mq has no further children (any additional prime factor would push it above n), i.e. mq is a leaf and $p(T_{n,mq},1) = 1$. Since the denominator of $p(T_{n,m},1)$ is a sum over the children and contains at least this 1, we get $p(T_{n,m},1) \le 1$. Applying this at the root to the children $m = p$ (primes) gives

$$c(n) = \sum_{p\le n} p(T_{n,p},1) \;\le\; \sum_{p\le n} 1 \;=\; \pi(n).$$

Lower bound $c(n) \ge \pi(n) - \pi(\sqrt{n})$

If $p > \sqrt{n}$ is prime, then p is a leaf of T_n (since $p^2 > n$), hence $p(T_{n,p},1) = 1$. Counting such primes yields

$$c(n) \;\ge\; \#\{\, p \le n :\; p > \sqrt{n}\,\} \;=\; \pi(n) - \pi(\sqrt{n}).$$

Asymptotics and an n–recurrence for $c(n)$

Combining the bounds,

$$\pi(n) - \pi(\sqrt{n}) \ \le\ c(n)\ \le\ \pi(n),$$

and using $\pi(\sqrt{n}) = o(\pi(n))$ (e.g. by the prime number theorem) gives $c(n) \sim \pi(n)$. Moreover, from (11) we read off the exact increment:

- if $n+1$ is prime, a new root child (a leaf) of weight 1 is added, so $c(n+1) = c(n) + 1$;
- if $n+1$ is composite with $s = \mathrm{spf}(n+1)$, then the only changed root subtree is that rooted at s, hence

$$\boxed{c(n+1) - c(n) \ =\ p(T_{n+1,s}, 1) - p(T_{n,s}, 1)}.$$

In summary, (a) establishes the precise n–recursions (9)–(10) and the branched continued–fraction structure of $p(T_n, x)$; (b) proves the identity (11) together with the bounds sandwiching $c(n)$ between $\pi(n)$ and $\pi(n) - \pi(\sqrt{n})$, and derives the exact update rule for c.

Acknowledgments

The definition of the factorization tree and the observation about the fast factorization were done by the first author, while all the rest by the second author.

A SymPy/Numpy/Matplotlib script replicating the experiments

What it does. Builds w_n incrementally via the `nextWord` logic, records the insertion index s_n, constructs the stable separating permutation, computes $C(\sigma)$, then $Z = |w| - C(\sigma)$, and plots/prints normalized histograms.

```
# prime_tree_experiments.py  (pure SymPy/NumPy/Matplotlib)

from sympy import factorint, isprime, primerange, nextprime
from sympy.combinatorics.permutations import Permutation
import numpy as np
import math
import random
import matplotlib.pyplot as plt

# ---------- helpers ----------
def P1(n: int) -> int:
    """Largest prime divisor (P1(1)=1)."""
    if n == 1:
        return 1
    f = factorint(n)
    return max(f.keys())

def sorted_prime_divisors(n: int):
    """Prime divisors with multiplicity, sorted nondecreasing."""
    f = factorint(n)
    out = []
    for p in sorted(f.keys()):
        out.extend([p] * f[p])
    return out

def encode(n: int, start: int = 1) -> str:
```

```python
    """Reference encoder of the tree T_n as a Dyck word (1=open, 0=
        close)."""
    # children are mp where p runs over primes in [P1(start), n//start]
    low = P1(start)
    high = n // start
    children = [start * p for p in primerange(low, high + 1)]
    if not children:
        return "10"
    w = "".join(encode(n, start=c) for c in children)
    return "1" + w + "0"

# ---------- incremental word update (inserting "10") ----------
def nextWord(wn: str) -> str:
    """Given w_n (as '01' string), return w_{n+1} by inserting '10' at
        index s_n."""
    if wn == "":
        return "10"
    n = len(wn) // 2
    sp = sorted_prime_divisors(n + 1)
    Mp = sp[-1]
    myM = (n + 1) // Mp

    # simulate the DFS queue (m, current_prime_bound)
    qq = [(+1, 2)]
    cnt = 0
    wn_list = list(wn)
    s = None

    for ch in wn_list[1:]:
        cnt += 1
        if ch == '1':
            m, p = qq[0]
            qq.insert(0, (m * p, p))
        else:  # ch == '0'
            m, p = qq.pop(0)
            if qq:
                qq[0] = (qq[0][0], nextprime(qq[0][1]))
            if myM == m and s is None:
                s = cnt - 1
                break

    if s is None:
        # fallback: append (should not happen in typical traces)
        s = len(wn) - 1

    wn1 = wn[: (s + 1)] + "10" + wn[(s + 1) :]
    return wn1

# --------- separation via stable sort and cycle count ---------
def stable_separator_cycle_count(w: str) -> int:
    """
    Build the stable sorting permutation that moves all '0's first,
        then '1's,
    preserving relative order within each block; return number of
        cycles.
    """
    m = len(w)
    # current positions are 0..m-1
```

```
    zeros = [i for i, ch in enumerate(w) if ch == '0']
    ones  = [i for i, ch in enumerate(w) if ch == '1']
    target_order = zeros + ones  # desired positions in separated word
    # Permutation in SymPy uses images in one-line notation: p(i) =
        images[i]
    # We want permutation pi such that pi maps new index j to old index
         target_order[j].
    images = list(range(m))
    for newpos, oldpos in enumerate(target_order):
        images[newpos] = oldpos
    pi = Permutation(images)  # maps j -> images[j]
    return len(pi.cyclic_form)

# ---------- main experiment ----------
def run_experiment(N=350, show_plots=True):
    wn = ""
    Z_list = []
    s_norm = []

    for n in range(1, N + 1):
        s = 0
        if wn != "":
            wn_next = nextWord(wn)
            # index of first difference:
            for i in range(len(wn)):
                if wn[i] != wn_next[i]:
                    s = i
                    break
        wn = nextWord(wn)
        # sanity: optional equality check with full encode(n)
        # assert wn == encode(n), "Mismatch at n={}".format(n)

        # build separator permutation and measure cycles
        C = stable_separator_cycle_count(wn)
        Z = len(wn) - C                     # our statistic
        Z_list.append(Z - 2 * n)            # centered at 2n as in the
            question
        s_norm.append(s / max(1, 2 * n - 1))

        if n % 100 == 0:
            print(n, Z - 2 * n)

    # normalize Z_list to mean 0, variance 1 for hist
    Z_arr = np.array(Z_list, dtype=float)
    mu = Z_arr.mean()
    sd = Z_arr.std(ddof=0)
    if sd == 0.0:
        sd = 1.0
    Z_norm = (Z_arr - mu) / sd

    print("Z - 2n: mean =", mu, " stddev =", sd)
    print("s_norm: mean =", np.mean(s_norm), " stddev =", np.std(s_norm
        , ddof=0))

    if show_plots:
        fig, ax = plt.subplots(1, 2, figsize=(10,4))
        ax[0].hist(Z_norm, bins=30, density=True)
        ax[0].set_title("Normalized (Z-2n)")
```

```
        ax[1].hist(s_norm, bins=30, density=True)
        ax[1].set_title("Normalized␣insertion␣index␣s_n/(2n)")
        plt.tight_layout()
        plt.show()

if __name__ == "__main__":
    run_experiment(N=350, show_plots=True)
```