Algorithmic Construction and Structure of Prime Number Subsets

Volodymyr Tkachenko

July 6, 2021

Dedicated to my mother, Antonina Grigorievna Tkachenko, a radiologist,
my 5th-grade teacher Valentina Petrovna Golovko,
and my son Ilya Tkachenko, who survived the Russian occupation of 2022.

Abstract

This work introduces a novel algorithm for constructing the entire set of prime numbers $\mathbb{P}$ by uncovering deep structural connections between their subsets. The algorithm operates recursively, without a single division operation, providing a new perspective on the nature of primes.

The method partitions the set of natural numbers into intervals defined by an infinite sequence of primorials. Within each such interval, a finite subset of prime numbers, termed a "brigade," is generated. The core of the approach is a proof that these brigades can be constructed entirely from monomials formed by previously discovered primes. This demonstrates that all prime numbers can be generated in an infinite, self-sustaining cycle.

By operating with a significantly smaller set of monomials compared to the number of integers in each interval, the algorithm offers a computationally attractive alternative to traditional sieve-based methods. This structural approach, which leverages the inherent polynomial relationships among primes first demonstrated by Matiyasevich, provides new insights into their organization on the number line.

This theory can serve as an accessible and intuitive framework for university students in mathematics, bridging classical number theory with an algorithmic perspective. This research was conducted independently by the author. The algorithm has been fully implemented and verified by a working computer program written in JavaScript, easily accessible in web browsers and on smartphones, available upon request.

1 Introduction

In this paper, we introduce the concept of brigades (Brig), which are subsets of prime numbers residing within intervals on the number line defined by successive primorials. These connections are sufficiently strong to allow the generation of the entire set of prime numbers $\mathbb{P}$ without division.

Several established tools for studying prime numbers include:

1. The **Euler-Riemann zeta function**, introduced by Leonhard Euler and later extended to the complex domain by Bernhard Riemann in 1859:

$$\zeta(s) = \sum_{n=1}^{\infty} n^{-s} = \prod_{p \in \mathbb{P}} \left(1 - p^{-s}\right)^{-1},$$

   where p ranges over all prime numbers.

2. Positive values of a multivariable polynomial derived from the solution to Hilbert's 10th problem, established in 1970 by **Davis, Putnam, Robinson, and Matiyasevich**.

The existence of the Robinson-Matiyasevich polynomial demonstrates inherent connections within the set of prime numbers.

This paper elucidates these connections through an algorithmic approach.

2 Preliminaries and Definitions

Definition 1.

- *i - **infinite loop number** (the loop operation starts from $i = 3$).*
- ***Prime number sequence**: A prime p_i is the i-th smallest prime.*
  For convenience, we will assume that $p_0 := 1$.
  This article assumes that if $p_s \in \mathbb{P}, p_t \in \mathbb{P}$, then $p_s \leq p_t \iff s \leq t$.
  For example, $p_1 = 2$, $p_5 = 11$.
- ***Primorial ($\mathbf{p}_i\#$)**: The multiplication of the first i primes:*

$$p_0\# := 1, \quad p_i\# := \prod_{k=1}^{i} p_k.$$

  For example, $p_4\# \ = \ p_1 \cdot p_2 \cdot p_3 \cdot p_4 \ = \ 2 \cdot 3 \cdot 5 \cdot 7 = 210$.
- ***allMp**$_n$: is $\{p_1, p_2, \ldots, p_n\}$ for $n \in \mathbb{N}$*
- ***Interval of natural numbers (NInt$_i$)**: Defined as:*

$$\mathrm{NInt}_1 := [1, 2], \quad \mathrm{NInt}_i := [p_{i-1}\# + 1, \ p_i\#] \quad \textit{for } i \geq 2.$$

  Example: $\mathrm{NInt}_2 = [3, 6]$, $\mathrm{NInt}_3 = [7, 30]$, $\mathrm{NInt}_4 = [31, 210]$, $\mathrm{NInt}_5 = [211, 2310]$.
- ***Subset of primes (Brig$_i$)**: The primes in* NInt_i.
  For example, $NInt_3 = [7, 30]$ and $Brig_3 = \{7, 11, 13, 17, 19, 23, 29\} = allMp_{10} \setminus allMp_3$.
  First we'll show how Brig_i is constructed, and then we'll prove that it is indeed the entire set of prime sets from the NInt_i.
- ***ActivePrimes**$_i$ -current active set of prime numbers in the algorithm*

$$\mathrm{ActivePrimes}_i = \mathrm{ActivePrimes}_{i-1} \cup \mathrm{Brig}_{i-1} \setminus \{p_{i-1}\}$$

  or

$$\mathrm{ActivePrimes}_i = \left(\bigcup_{k=1}^{i-1} Brig_k \right) \setminus \mathrm{allMp}_{i-1}$$

- ***setMonom$_i$ = monomPowerGeZero(ActivePrimes$_i$, hiLimit$_i$)** - procedure, which creates all possible monomials setMonom$_i$ from an ascending ordered array of prime numbers* $\mathrm{ActivePrimes}_i$ *creates all possible monomials setMonom$_i$, whose value is less than hiLimit$_i$ and the sum of powers of each monomial ≥ 0 and returns this set of monomials.*
  It follows that $1 \in$ setMonom$_i$ (power=0), $\mathrm{ActivePrimes}_i \subset$ *setMonom$_i$ (power=1).*
- ***BuildingRow (BuildingRow$_i$)**: A set for building the table:*
  BuildingRow$_i$ = monomPowerGeZero($\mathrm{ActivePrimes}_i$, $p_{i-1}\#$*),*
  or

$$\mathrm{BuildingRow}_i = \left\{ r = \prod_{j=i}^{m} p_j^{k_j} \;\middle|\; k_j \geq 0, \, r < p_{i-1}\# \right\},$$

  where m is the largest index of a prime number from $\cup_{k=1}^{i-1} \mathrm{Brig}_k$.
  or $\mathrm{BuildingRow}_i$ *make*

$$\mathrm{BuildingRow}_i = \mathrm{monomPowerGeZero}(\mathrm{ActivePrimes}_i, \ p_{i-1}\#)$$

  Example: $\mathrm{BuildingRow}_3 = \{1, 5\}$, $\mathrm{BuildingRow}_4 = \{1, 7, 11, 13, 17, 19, 23, 29\}$,

$\text{BuildingRow}_5 = \ \{1\} \sqcup \text{Brig}_4 \ \sqcup \{p_5 \cdot p_5,\ p_5 \cdot p_6,\ p_5 \cdot p_7,\ p_6 \cdot p_6\} \setminus p_4 \ =,$
$\{1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209\}$
$|\text{BuildingRow}_5| = 48$, *where* Brig_4 *is described below.*
Note that all BuildingRow_i*, then* $i \geq 5$ *also have composite numbers.*

- ***Candidate table (Candid$_i$):*** *A set of numbers within* NInt_i *constructed as potential primes:* ***Definition (tabl()).*** *Operator tabl() pass set BuildingRow$_i$ and prime number p_i in set Candid$_i$:*

$$\text{Candid}_i := \bigsqcup_{k=1}^{p_i - 1} (\text{BuildingRow}_i + k \cdot p_{i-1}\#).$$

This sign $\bigsqcup$ *denotes a disjunctive union.* Candid_i *can be considered as a table with* $|\text{BuildingRow}_i|$ *elements in a row and* $p_i - 1$ *rows in total.*

To each element of the $\{\text{BuildingRow}_i\}$ *is added* $k \cdot p_{i-1}\#$*,*

where $k \in \{1, 2, \ldots, p_i - 1\}$*,* k *is the row number.*

Example: For Candid_4*, see below.*

- ***setMonom$_i$ = monomPowerGeTwo(ActivePrimes$_i$, hiLimi$_i$)*** *- procedure, which creates all possible monomials setMonom$_i$ from an ascending ordered array of prime numbers* ActivePrimes_i *creates all possible monomials setMonom$_i$, whose value is less than hiLimit$_i$ and the sum of powers of each monomial* ≥ 2 *and returns this set of monomials.*

- ***Filter (Filter$_i$):*** *A subset of* Candid_i *containing composite numbers:*

$$\text{Filter}_i := \left\{ r \in \text{Candid}_i \;\middle|\; r = \prod_{j=i}^{m} p_j^{k_j} \;\middle|\; k_j \geq 0,\ \sum_{j=i}^{m} k_j \geq 2,\ r < p_i\# \right\}$$

or

$$\text{Filter}_i := \{r \in \text{monomPowerGeTwo}(\text{ActivePrimes}_i, p_i\#) \ | \ r \in \text{Candid}_i\}.$$

Note that Filter_i *consists of composite numbers (Power* ≥ 2*).*

Also. The condition that Filter_i *is nested within* Candid_i *is optional. The computer program works without this requirement. But if the condition that* Filter_i *is nested in* Candid_i *is fulfilled, then a division operation will be required - determining the remainder r when dividing by the factorial* $p_{i-1}\#$ *and finding out whether r belongs to* BuildingRow_i*. hiLimi$_i$ = $p_i\#$.*

Note that here the upper limit is larger than it was before.

Example: $\text{Filter}_4 = \{49, 77, 91, 119, 121, 133, 143, 161, 169, 187, 203, 209\}$
where m is the largest index of a prime number from Brig_{i-1}*.*
Indeed they are composite numbers, because the sum of powers ≥ 2*,*
and the prime numbers used are the same as those used to get the elements for Candid_i*.*
Below we will prove that :

- *(It will be proved below) Primes in* NInt_i*: Obtained by removing* Filter_i *from* Candid_i*:*

$$\text{Brig}_i := \text{Candid}_i \setminus \text{Filter}_i.$$

- *(It will be proved below)* $\mathbb{P}$*: The set of all prime numbers,* $\mathbb{P} := \bigsqcup_{i=1}^{\infty} \text{Brig}_i$*, where* Brig_i *denotes the primes obtained in the i-th step of the algorithm.*

- *If number m is mutually simple with number n, then we write*

$$simp(m,\ n)$$

Let us say that the set setM is mutually simple with number n, then any element of setM is mutually simple with n

$$simp(setM,\ n)$$

Let us define the procedure setMonom(setPrimes, hiLimit) as a set of monomials built on the basis of elements from the finite set of prime numbers setPrimes with the number hiLimit restricting the monomials from above

$$setMonom(setPrimes,\ hiLimit)$$

$\mathrm{pBase}(n)$ *- is the minimal set of some prime numbers* $\{q_1, q_2, \ldots, q_k\}$ *like this, that the product of their powers* $a1, a2, \ldots, ak$ *equally* n

or

pBase *for number* n *we will call such a minimal set from which it is possible to construct a monomial number equal to* n

$$n = q_1^{a1} \cdot q_2^{a2} \cdot \ldots \cdot q_k^{ak}$$

$\mathrm{pBase}(1) \stackrel{\text{def}}{=} \emptyset$

- $\mathrm{pBase}(a) \cap \mathrm{pBase}(b) = \{p \in \mathbb{P} \ \mid \ p \in \mathrm{pBase}(a) \ \textit{and} \ p \in \mathrm{pBase}(b)\}$
- $\mathrm{pBase}(a) \cup \mathrm{pBase}(b) = \{p \in \mathbb{P} \ \mid \ p \in \mathrm{pBase}(a) \ \textit{or} \ p \in \mathrm{pBase}(b)\}$
- $\mathrm{pBase}(a) \setminus \mathrm{pBase}(b) = \{p \in \mathbb{P} \ \mid \ p \in \mathrm{pBase}(a) \ \textit{and} \ p \notin \mathrm{pBase}(b)\}$

  We will write $sipl(a, b)$ *when the numbers* a *and* b *are mutually simple.*

3 Main Theorem and Proof

Theorem 1. Brig_i *is the set of prime numbers in* NInt_i*.*

Proof. Here is a list of the basic properties of $\mathrm{simp}(), \mathrm{pBase}()$, some of the properties with proofs and others proofs are omitted due to their elementary nature.

1. $\mathrm{simp}(a, b) \Longleftrightarrow \mathrm{simp}(b, a)$

   - symmetry

2. $simp(a,\ b),\ simp(c,\ b) \Longrightarrow simp(a \cdot c,\ \ b)$

   By Bezu's theorem (which is based on the Euclidean division algorithm)

   it follows that for $simp(a,\ b),\ simp(c,\ b)$

   there exist such integers $x1, y1$ and $x2, y2$ that

   $a \cdot x1 + b \cdot y1 = 1,$

   $c \cdot x2 + b \cdot y2 = 1.$

   Multiply the left and right sides of the equations.

   $a \cdot c \cdot x1 \cdot x2 + a \cdot b \cdot x1 \cdot y2 + b \cdot c \cdot y1 \cdot x2 + b^2 \cdot y1 \cdot y2 = 1$ or

   $a \cdot c \cdot (x1 \cdot x2) + b \cdot (a \cdot x1 \cdot y2 + c \cdot x2 \cdot y1 + b \cdot y1 \cdot y2) =$

   $a \cdot c \cdot x3 + b \cdot y3),$

   where $x3 = x1 \cdot x2, \quad y3 = a \cdot x1 \cdot y2 + c \cdot x2 \cdot y1 + b \cdot y1 \cdot y2$

3. $simp(a \cdot c,\ \ b),\ simp(c,\ b) \Longrightarrow simp(a,\ b)$

4. $\mathrm{simp}(a, b) \Longleftrightarrow \mathrm{pBase}(a) \cap \mathrm{pBase}(b) = \emptyset$

5. $\mathrm{pBase}(p_i) = \{p_i\}\ for\ i \in \mathbb{N}$

6. $\mathrm{pBase}(a^n) = \mathrm{pBase}(a)\ for\ n \in \mathbb{N}$

7. $\mathrm{simp}(a, b) \Longleftrightarrow$

   $\mathrm{pBase}(a + b) \cap \mathrm{pBase}(a) = \emptyset \ \mid \ \mathrm{pBase}(a + b) \cap \mathrm{pBase}(b) = \emptyset$

   Note, for proof to use $a \cdot b + a \cdot c = a \cdot (b + c)$

8. $n \in \mathbb{N} \Longrightarrow n = monomM(n)$

   It's about the basic theorem of arithmetic.

9. $p_k \in \mathbb{P} \mid k > i \Longrightarrow \text{simp}(p_k,\, p_i\#)$

10. $\text{pBase}(\mathbb{N}) = \text{pBase}(\mathbb{P})$ if you broaden the definition pBase()

11. $1 \leq k < p_i \Longrightarrow \text{pBase}(p_{i-1}\# \cdot k) = \text{pBase}(p_{i-1}\#)$

    In this case $\text{pBase}(k) \subset \text{pBase}(p_{i-1}\#)$

12. $r \in \text{BuildingRow}_i \mid 1 \leq k < p_i \Longrightarrow$

    $p_{i-1} <$ any divisor of the number $(p_{i-1}\# \cdot k + r)$ and $< p_j$, where j - maximal computed prime number after i-1-th cycle.

Let NI_s be the interval from 1 to the primorial $p_s\#$, i.e.
$NI_s = [1, p_s\#]$ and p_t is the greatest prime of this interval, i.e.
$p_t \in NI_s,\ p_s\# < p_{t+1}$.
Then all numbers of this interval can be represented as monomials from allMp_t ,
i.e. $pBase(NI_s) = \text{allMp}_t$ and $NI_s = monomPowerGeZero(allMp_t, p_s\#)$.
Proof
Necessity
$1 = p_1^0$
Let u as
$1 \leq u < t + 1 \Longrightarrow p_u \in NI_s \mid p_u = p_u^1$ - monomial.
Sufficiency follows from the basic theorem of arithmetic.
End proof
Let j be the maximum index of a prime number not included in NInt_i.
$M_i := \{p_{j+1}, \ldots, p_{j+k}\}$, then $p_{j+k} < p_{i-1}\# < p_{j+k+1}$.
Number $p_{i-1}\#$ is mutually simple set $\text{pBase}(M_i)$.
The statement follows from the fact that the p-bases do not intersect.
$\text{pBase}(p_{i-1}\#) \cap M_i = \emptyset$
Indeed, $\text{pBase}(p_{i-1}\#) = \text{allMp}_{i-1}$
does not intersect with $\{p_{j+1}, \ldots, p_{j+k}\}$.
$setMonom(\{p_{j+1}, \ldots, p_{j+k},\ p_{i-1}\#\})$ is set mutually simple with $p_{i-1}\#$
We will assume that the set $setM_i := \{p_{j+1}, \ldots, p_{j+k}\}$,
where $p_{j+k} < p_{i-1}\# < p_{j+k+1}$ contains all prime numbers, belonging to the interval $NInt_i$.

It is clear that on the basis of the set allMp_{j+k}. is constructed as monomials of the entire set $\{1, 2, \ldots, p_j\#\}$ and, in particular, $NInt_i$.

New prime elements cannot form monomials with old ones - this is how the selection was done in $Candid_i$. New simple elements cannot form monomials with themselves because the square of the smallest of them exceeds the the upper limit of the interval. Thus, the new prime goes by itself, not mixed with others. The numbers from $NInt_i \setminus setMonom(M_i)$ must have elements from $\{p_1, \ldots, p_j\}$, and hence be composite.

The set $Filter_i$ is constructed as containing monomials based on $\{p_{i-1}, \ldots, p_j$ and their $power \geq 2$. $Filter_i$ is a composite set. Therefore, when filtering the set $Candid_i$ by the set $Filter_i$, only prime numbers - new prime numbers - will remain.

We have a set of prime numbers $Mp(1, s) = \{p_1, p_2, ..., p_s\}$.
Let i as: $0 < i < s$.
Let us split the set $Mp(1, s)$ into two: $Mp(1, s) = Mp(1, i) \cup Mp(i + 1, s)$
$Mp(1, i) := \{p_1, p_2, ..., p_i\}$ and $M_(i + 1, j) := \{p_{i+1}, p_{i+2}, ..., p_s\}$.
Let us partition the set of monomials constructed on $Mp(1, s)$
into two using the primorial $p_i\#$:

1. $notSinpMonP_i$ the set of monomials not mutually simple with $p_i\#$,
2. $SinpMonP_i$ the set of monomials mutually simple with $p_i\#$.

If a monomial contains at least one prime from $Mp(1, i)$, then the monomial is included in the $notSinpMonP_i$, and if the monomial consists only of simplices from $Mp(i + 1, s)$, then it is in the $SinpMonP_i$.

Therefore $NInt_i \setminus \text{Candid}_i$ consists of composite numbers and all new prime numbers from $NInt_i$ are monomials,

and the new prime numbers have no coefficients.
Conclusion: Candid_i consists of

- prime numbers and,
- composite numbers which belong to Filter_i.

Therefore, the set $\mathrm{Candid}_i \setminus \mathrm{Filter}_i$ is the set of prime numbers from NInt_i and $\mathrm{Brig}_i = \mathrm{Candid}_i \setminus \mathrm{Filter}_i$.
Therefore, Brig_i contains all prime numbers in NInt_i.
Consequently, $\mathbb{P} := \bigsqcup_{i=1}^{\infty} \mathrm{Brig}_i$. □

4 Example for $i = 4$

In this case $p_4 = 7$.
$p_3\# = 30$, and $p_4\# = 210$, consequently, $\mathrm{NInt}_4 = [31, 210]$.
The set $\mathrm{BuildingRow}_4$ contains numbers of the form $\prod_{j=3}^{m} p_j^{k_j}$,
where $k_j \geq 0$ and the multiplication is less than $p_3\# = 30$:
$\mathrm{BuildingRow}_4 = \{1, 7, 11, 13, 17, 19, 23, 29\}$, $|\mathrm{BuildingRow}_4| = 8$
The set Candid_4 is formed as $\mathrm{Candid}_4 = \bigsqcup_{k=1}^{6}(\mathrm{BuildingRow}_4 + k \cdot 30)$.
The elements of Candid_4 are shown in Table 1, where rows correspond to values of k:

Table 1: Elements of Candid_4

k	$\mathrm{BuildingRow}_4 + k \cdot 30$							
1	31	37	41	43	47	49	53	59
2	61	67	71	73	77	79	83	89
3	91	97	101	103	107	109	113	119
4	121	127	131	133	137	139	143	149
5	151	157	161	163	167	169	173	179
6	181	187	191	193	197	199	203	209

The set Filter_4 contains the composite numbers from Candid_4:
$\mathrm{Filter}_4 = \{49, 77, 91, 119, 121, 133, 143, 161, 169, 187, 203, 209\} =$
$\{7 \cdot 7,\ 7 \cdot 11,\ 7 \cdot 13,\ 7 \cdot 17,\ 11 \cdot 11,\ 7 \cdot 19,\ 11 \cdot 13,\ 7 \cdot 23,\ 13 \cdot 13,\ 11 \cdot 17,\ 7 \cdot 29,\ 11 \cdot 19)$
$|\mathrm{Filter}_4| = 12$

5 Algorithm for Construction of $\mathbb{P}$

6 Properties of the Algorithm

1. Without the division operation, relying on connections between finite subsets of prime numbers Brig_i (Brigades), the algorithm constructs the entire set of prime numbers $\mathbb{P}$ in an infinite loop.
2. Every prime number belongs to a Candidate set Candid_i and can be uniquely represented as the sum of two monomials of the aforementioned kind.
3. If i is the ordinal index of a prime number p, it directly influences other prime numbers from the cycle following its occurrence up to the i-th cycle in the algorithm's calculations. In particular, in the infinite loop of the algorithm, each prime number is directly used a finite number of times.
4. The distribution of prime numbers on the axis of natural numbers has a wave-like nature.
5. Finally, the relations between subsets of prime numbers $\{Brig_i\}_{i=1}^{\infty}$ which are described in this paper.

Algorithm 1 Algorithm GeneratePrimes

Initialize $Pset \leftarrow \{2, 3\}$
$i \leftarrow 3$
while True **do**
  $NInt_i \leftarrow [p_{i-1}\# + 1,\ p_i\#]$
  m is max index $p_m \in \mathrm{Brig}_{i-1}$
  $BuildingRow_i \leftarrow \{r = \prod_{j=i}^{m} p_j^{k_j} \mid j \geq i,\ k_j \geq 0,\ r < p_{i-1}\#\}$
  $Candid_i \leftarrow \{\}$
  for $k \leftarrow 1$ **to** $p_i - 1$ **do**
    for each $r \in BuildingRow_i$ **do**
      $Candid_i \leftarrow Candid_i \cup \{r + k \cdot p_{i-1}\#\}$
    end for
  end for
  $Filter_i \leftarrow \{r \in Candid_i \mid r = \prod_{j=i}^{m} p_j^{k_j} | j \geq i, k_j \geq 0, \sum_{j=i}^{m} k_j \geq 2, r < p_i\#\}$
  $Brig_i \leftarrow Candid_i \setminus Filter_i$
  $Pset \leftarrow Pset \cup Brig_i$
  $i \leftarrow i + 1$
end while

7 Applications and Open Questions

1. Prove that every row or column in Candid_i contains at least one prime.

2. Investigate the existence of long arithmetic sequences in the initial columns of sets Candid_i.

3. Given that the algorithm places all prime numbers into Candidate sets, are there any useful approaches to this fact from the perspective of graph theory? We propose to define inheritance by columns.

4. Investigate the relevance of the algorithm's monomial constructions and ordered sequence generation to the study of commutative rings with order.

5. Hopefully the property in (2) will help simplify some existing proofs and perhaps help in new ones. And a special hope for the fundamental connections between the Brigades.

8 Keywords

Prime number subsets; monomials; brigade structure; subset relations; algorithmic number theory; prime number construction; computational number theory.

References

[1] Hardy, G.H., & Wright, E.M. *An Introduction to the Theory of Numbers*. Oxford University Press, 2008.

[2] Crandall, R., & Pomerance, C. *Prime Numbers: A Computational Perspective*. Springer, 2008.

[3] Matiyasevich, Y.V. *Hilbert's Tenth Problem*. MIT Press, 1993.