# Minimal Elements for the Prime Numbers

Curtis Bright
School of Computer Science
University of Waterloo
Waterloo, ON N2L 3G1
Canada
cbright@uwaterloo.ca

Raymond Devillers
Département d'Informatique, CP 212
Université Libre de Bruxelles
B-1050 Bruxelles
Belgium
rdevil@ulb.ac.be

Jeffrey Shallit
School of Computer Science
University of Waterloo
Waterloo, ON N2L 3G1
Canada
shallit@cs.uwaterloo.ca

January 22, 2015

**Abstract**

We say a string of symbols $s$ is *minimal* for a language $L$ if $s$ is a member of $L$, and it is not possible to obtain another member of $L$ by striking out one or more symbols from $s$. Although the set $M(L)$ of minimal strings is necessarily finite, determining it explicitly for a given $L$ can be a difficult computational problem. We use some number-theoretic heuristics to compute $M(L)$, where $L$ is the language of base-$b$ representations of the prime numbers, for $2 \leq b \leq 30$.

# 1 Introduction

Problems about the digits of prime numbers have a long history, and many of them are still unsolved. For example, are there infinitely many primes, all of whose base-10 digits are `1`? Currently, there are only five such “repunits” known [16], corresponding to $(10^p - 1)/9$ for $p \in \{2, 19, 23, 317, 1031\}$. It seems likely that four more are given by $p \in \{49081, 86453, 109297, 270343\}$, but this has not yet been rigorously proven.

Another problem on the digits of primes was introduced by the third author [15]. To describe it, we need some definitions. We say that a string $x$ is a *subword* of a string $y$, and we write $x \triangleleft y$, if one can strike out zero or more symbols of $y$ to get $x$. For example, `string` is a subword of `Meistersinger`. (In the literature, this concept is sometimes called a “scattered subword” or “substring” or “subsequence”.) A *language* is a set of strings. A string $s$ is *minimal* for $L$ if (a) $s \in L$ and (b) if $x \in L$ and $x \triangleleft s$, then $x = s$. The set of all minimal strings of $L$ is denoted $M(L)$.

In this paper, we describe a heuristic technique for determining $M(L_b)$ in the case where $L_b$ consists of the representations, in base $b$, of the the prime numbers $\{2, 3, 5, \ldots\}$. We obtain a complete characterization of $M(L_b)$ for bases $2 \leq b \leq 16$ and $b = 18$, 20, 22, 24, 30. For the remaining bases $b = 17$, 19, 21, 23, and $25 \leq b \leq 29$, we obtain results that allow us to “almost” completely characterize this set.

The same technique can also be applied to find minimal sets for subsets of prime numbers. For example, we were able to determine the minimal set for primes of the form $4n + 1$ represented in base 10 (and similarly for those of the form $4n+3$). This successfully completes the sequences A111055 and A111056 in the *Encyclopedia of Integer Sequences* [12], which had been incomplete since their introduction to the Encyclopedia in 2005.

Finally, we also show that determining the minimal set for the composite numbers represented in base $b$ is a significantly easier problem, and explicitly compute the minimal set for all $2 \leq b \leq 30$.

## 1.1 Notation

In what follows, if $x$ is a string of symbols over the alphabet $\Sigma_b := \{\mathtt{0}, \mathtt{1}, \ldots, b-1\}$ we let $[x]_b$ denote the evaluation of $x$ in base $b$ (starting with the most significant digit). This is extended to languages as follows: $[L]_b := \{\, [x]_b : x \in L \,\}$. We use the convention that $\mathtt{A} := 10$, $\mathtt{B} := 11$, and so forth, to conveniently represent strings of symbols in base $b > 10$. We let $(x)_b$ be the canonical representation of $x$ in base-$b$, that is, the representation without leading zeroes. Finally, as usual, for a language $L$ we let $L^n := \overbrace{LL \cdots L}^{n}$ and $L^* := \bigcup_{i \geq 0} L^i$.

# 2 Why minimal sets are interesting

One reason why the minimal set $M(L)$ of a language $L$ is interesting is because it allows us to compute two natural and related languages, defined as follows:

$$\begin{aligned}\mathrm{sub}(L) &\coloneqq \{\, x \in \Sigma^* : \text{there exists } y \in L \text{ such that } x \triangleleft y \,\};\\ \mathrm{sup}(L) &\coloneqq \{\, x \in \Sigma^* : \text{there exists } y \in L \text{ such that } y \triangleleft x \,\}.\end{aligned}$$

An amazing fact is that $\mathrm{sub}(L)$ and $\mathrm{sup}(L)$ are always regular. This follows from the following classical theorem due to Higman [9] and Haines [8].

**Theorem 1.** *For every language $L$, there are only finitely many minimal strings.*

Indeed, we have $\mathrm{sup}(L) = \mathrm{sup}(M(L))$ and $\Sigma^* - \mathrm{sub}(L) = \mathrm{sup}(M(\Sigma^* - \mathrm{sub}(L)))$, and the superword language of a finite language is regular, since

$$\mathrm{sup}\big(\{w_1, \ldots, w_n\}\big) = \bigcup_{i=1}^{n} \Sigma^* w_{i,1} \Sigma^* \cdots \Sigma^* w_{i,|w_i|} \Sigma^*$$

where $w_i = w_{i,1} \cdots w_{i,|w_i|}$ with $w_{i,j} \in \Sigma$.

# 3 Why the problem is hard

Determining $M(L)$ for arbitrary $L$ is in general unsolvable, and can be difficult even when $L$ is relatively simple [6, 7].

The following is a "semi-algorithm" that is guaranteed to produce $M(L)$, but it is not so easy to implement:

```
(1) $M \coloneqq \emptyset$
(2) while ($L \neq \emptyset$) do
    (3) choose $x$, a shortest string in $L$
    (4) $M \coloneqq M \cup \{x\}$
    (5) $L \coloneqq L - \mathrm{sup}(\{x\})$
```

In practice, for arbitrary $L$, we cannot feasibly carry out step (5). Instead, we work with $L'$, some regular overapproximation to $L$, until we can show $L' = \emptyset$ (which implies $L = \emptyset$). In practice, $L'$ is usually chosen to be a finite union of sets of the form $L_1 L_2^* L_3$, where each of $L_1$, $L_2$, $L_3$ is finite. In the case we consider in this paper, we then have to determine whether such a language contains a prime or not.

However, it is not even known if the following simpler decision problem is recursively solvable:

**Problem 2.** Given strings $x$, $y$, $z$, and a base $b$, does there exist a prime number whose base-$b$ expansion is of the form $x\overbrace{yy\cdots y}^{n}z$ for some $n \geq 0$?

An algorithm to solve this problem, for example, would allow us to decide if there are any additional Fermat primes (of the form $2^{2^n}+1$) other than the known ones (corresponding to $n = 0, 1, 2, 3, 4$). To see this, take $b := 2$, $x := \mathtt{1}$, $y := \mathtt{0}$, and $z := \mathtt{0}^{16}\mathtt{1}$. Since if $2^n + 1$ is prime then $n$ must be a power of two, a prime of the form $[xy^*z]_b$ must be a new Fermat prime.

Therefore, in practice, we are forced to try to rule out prime representations based on heuristics such as modular techniques and factorizations. This is discussed in the next section.

# 4 Some useful lemmas

It will be necessary for our algorithm to determine if families of the form $[xL^*z]_b$ contain a prime or not. We use two different heuristic strategies to show that such families contain no primes.

In the first strategy, we mimic the well-known technique of "covering congruences" [3], by finding some finite set $S$ of integers $N > 1$ such that every number in a given family is divisible by some element of $S$. In the second strategy, we attempt to find a difference-of-squares or difference-of-cubes factorization.

## 4.1 The first strategy

We start with the simplest version of the idea: to find an $N > 1$ that divides each element of the family $[xL^*z]_b$. At first glance, this would require checking that $N$ divides $xL^nz$ for $n = 0, 1, 2, \ldots$. However, the following lemma shows that it is only necessary to check the two cases $n = 0$ and 1. Although divisibility based on digital considerations has a long history (e.g., [4, Chap. XII]), we could not find these kinds of results in the literature.

**Lemma 3.** *Let $x, z \in \Sigma_b^*$, and let $L \subseteq \Sigma_b^*$. Then $N$ divides all numbers of the form $[xL^*z]_b$ if and only if $N$ divides $[xz]_b$ and all numbers of the form $[xLz]_b$.*

*Proof.* Let $y = y_1 \cdots y_n \in L^*$, where $y_1, \ldots, y_n \in L$. By telescoping we have

$$[xyz]_b - [xz]_b = \sum_{i=1}^{n} ([xy_iy_{i+1} \cdots y_nz]_b - [xy_{i+1} \cdots y_nz]_b).$$

Cancelling the final $|y_{i+1} \cdots y_nz|$ base-$b$ digits in the summand difference — which are identical — this becomes

$$[xyz]_b = [xz]_b + \sum_{i=1}^{n} b^{|y_{i+1}\cdots y_nz|}([xy_i]_b - [x]_b).$$

But $b^{|z|}([xy_i]_b - [x]_b) = [xy_iz]_b - [xz]_b$ by adding and subtracting $[z]_b$, so we have

$$[xyz]_b = [xz]_b + \sum_{i=1}^{n} b^{|y_{i+1}\cdots y_n|}([xy_iz]_b - [xz]_b).$$

Since $N \mid [xz]_b$ and $N \mid [xy_iz]_b$ for each $1 \le i \le n$, it follows that $N \mid [xyz]_b$.

The other direction is clear, since $[xz]_b$ and numbers of the form $[xLz]_b$ are both of the form $[xL^*z]_b$. □

In practice, our algorithm employs this lemma with $L \coloneqq \{y_1, \ldots, y_n\} \subseteq \Sigma_b$, and all numbers of the form $[xL^*z]_b$ are shown to be composite with the following corollary.

**Corollary 4.** *If $1 < \gcd([xz]_b, [xy_1z]_b, \ldots, [xy_nz]_b) < [xz]_b$ then all numbers of the form $[x\{y_1, \ldots, y_n\}^*z]_b$ are composite.*

*Proof.* By Lemma 3, we know that $N \coloneqq \gcd([xz]_b, [xy_1z]_b, \ldots, [xy_nz]_b) > 1$ divides all numbers of the form $[x\{y_1, \ldots, y_n\}^*z]_b$. By the size condition $N$ is strictly less than each such number, and so is a nontrivial divisor. □

**Example 5.** Since $\gcd(49, 469) = 7$, every number with base-10 representation of the form $\texttt{46}^*\texttt{9}$ is divisible by 7. Since $1 < 7 < 49$, each such number is composite.

We also generalize this to the following corollary in the case where a single divisor does not divide each number in the family.

**Corollary 6.** *Let $L \coloneqq \{y_1, y_2, \ldots, y_n\}$. If*

$$N_0 \coloneqq \gcd\bigl(\{[xz]_b\} \cup [xL^2z]_b\bigr)$$

*and*

$$N_1 \coloneqq \gcd\bigl([xLz]_b \cup [xL^3z]_b\bigr)$$

*lie strictly between 1 and $[xz]_b$, then all numbers of the form $[xL^*z]_b$ are composite.*

*Proof.* By Lemma 3 applied to $[x(L^2)^*z]_b$, we know that $N_0$ divides all numbers of the form $[xL^*z]_b$ in which an even number of $y_i$ appear. By Lemma 3 on $[xy_i(L^2)^*z]_b$ for each $1 \le i \le n$, we know that $N_1$ divides all numbers of the form $[xL^*z]_b$ for which an odd number of $y_i$ appear. By the size conditions, $N_0$ and $N_1$ are nontrivial divisors. □

**Example 7.** Since $\gcd([\texttt{6}]_9, [\texttt{611}]_9) = 2$, every number with base-9 representation of the form $\texttt{61}^*$ of odd length is divisible by 2. Since $\gcd([\texttt{61}]_9, [\texttt{6111}]_9) = 5$, every number with base-9 representation of the form $\texttt{61}^*$ of even length is divisible by 5. Since these numbers lie strictly between 1 and 6, every number with base-9 representation of the form $\texttt{61}^*$ is composite.

We also note that it is simple to generalize Corollary 6 to apply to check if there are divisors $N_0$, $N_1$, …, $N_{k-1}$ such that $N_i$ divides all numbers of the form $[x\{y_1, \ldots, y_n\}^*z]_b$ in which the number of $y_i$ appearing is congruent to $i$ mod $k$.

**Example 8.** Let $b \coloneqq 16$. Then 7 divides $[\texttt{8A01}]_b$ and $[\texttt{8A0AAA1}]_b$. Furthermore, 13 divides $[\texttt{8A0A1}]_b$ and $[\texttt{8A0AAAA1}]_b$, and 3 divides $[\texttt{8A0AA1}]_b$ and $[\texttt{8A0AAAAA1}]_b$. Thus all numbers with base-16 representation of the form $\texttt{8A0A}^*\texttt{1}$ are divisible by either 7, 13, or 3, depending on their length mod 3.

A version of Lemma 3 which applies to the most general kind of family we need to consider ($x_1L_1^* \cdots x_mL_m^*$, where we allow the case $L_m^* = \emptyset$) is formulated in Lemma 9.

**Lemma 9.** *Let $x_1, \ldots, x_m \in \Sigma_b^*$, and $L_1, \ldots, L_m \subseteq \Sigma_b^*$. Then $N$ divides all numbers of the form $[x_1L_1^*x_2L_2^* \cdots x_mL_m^*]_b$ if and only if $N$ divides $[x_1 \cdots x_m]_b$ and all numbers of the form $[x_1L_1x_2x_3 \cdots x_m]_b, \ldots, [x_1 \cdots x_{m-1}x_mL_m]_b$.*

*Proof.* Say $w \in x_1L_1^*x_2L_2^* \cdots x_mL_m^*$; then there exist $y_{i,1}, \ldots, y_{i,n_i} \in L_i$ such that

$$w = x_1y_{1,1} \cdots y_{1,n_1}x_2y_{2,1} \cdots y_{2,n_2} \cdots x_my_{m,1} \cdots y_{m,n_m}$$

for $1 \leq i \leq m$. As in the proof of Lemma 3, we have that

$$[w]_b = [x_1 \cdots x_m]_b + \sum_{i=1}^{m} \sum_{j=1}^{n_i} b^{|y_{i,j+1} \cdots y_{m,n_m}|}([x_1 \cdots x_iy_{i,j}x_{i+1} \cdots x_m]_b - [x_1 \cdots x_m]_b)$$

from which the claim follows. □

As in Lemma 3, we typically apply this lemma in the case where each $L_i \subseteq \Sigma_b$ and show that all numbers of the form $[x_1L_1^*x_2L_2^* \cdots x_mL_m^*]_b$ have a divisor.

**Example 10.** Take $(L_1, L_2, L_3) := (\{\mathtt{0}\}, \{\mathtt{0}\}, \emptyset)$ and $(x_1, x_2, x_3) := (\mathtt{9}, \mathtt{8}, \mathtt{1})$. Since 9 divides 981, 9081, and 9801, it follows that 9 divides every number with base-10 representation of the form $\mathtt{90}^*\mathtt{80}^*\mathtt{1}$.

More generally, if a single divisor doesn't work for every number, Lemma 9 can also be applied in the case where all numbers of the form $[x_1L_1^* \cdots x_i(L_i^2)^* \cdots x_mL_m^*]_b$ have one divisor, and all numbers of the form $[x_1L_1^* \cdots x_iL_i(L_i^2)^* \cdots x_mL_m^*]_b$ have another divisor.

**Example 11.** Let $b := 11$. Since 3 divides each of $[\mathtt{44A1}]_b$, $[\mathtt{44A111}]_b$, $[\mathtt{440A1}]_b$ , it follows that every number of the form $[\mathtt{440}^*(\mathtt{11})^*\mathtt{1}]_b$ is composite. Since 2 divides each of $[\mathtt{44A11}]_b$, $[\mathtt{44A1111}]_b$, $[\mathtt{440A11}]_b$, we know every number of the form $[\mathtt{440}^*(\mathtt{11})^*\mathtt{11}]_b$ is composite. It follows that all numbers of the form $[\mathtt{440}^*\mathtt{A1}^*\mathtt{1}]_b$ are composite.

Lemma 9 can also be applied to the case when all even-length strings under consideration have one divisor, and all the odd-length strings have another divisor. One such case is, for example, if numbers of the form $[x_1(L_1^2)^*x_2(L_2^2)^*x_3]_b$ and $[x_1L_1(L_1^2)^*x_2L_2(L_2^2)^*x_3]_b$ have one divisor, and numbers of the form $[x_1L_1(L_1^2)^*x_2(L_2^2)^*x_3]_b$ and $[x_1(L_1^2)^*x_2L_2(L_2^2)^*x_3]_b$ have another divisor.

**Example 12.** Let $b := 9$. Since 2 divides each of $[\mathtt{6}]_b$, $[\mathtt{116}]_b$, $[\mathtt{611}]_b$, $[\mathtt{161}]_b$, $[\mathtt{11161}]_b$, $[\mathtt{16111}]_b$, every odd-length string of the form $\mathtt{1}^*\mathtt{61}^*$ is composite. Since 5 divides each of $[\mathtt{16}]_b$, $[\mathtt{1116}]_b$, $[\mathtt{1611}]_b$, $[\mathtt{61}]_b$, $[\mathtt{1161}]_b$, $[\mathtt{6111}]_b$, every even-length string of the form $\mathtt{1}^*\mathtt{61}^*$ is composite.

### 4.2 The second strategy

A second way of proving that families of the form $xL^*z$ do not contain a prime is via algebraic factorizations, such as a difference-of-squares factorization.

**Lemma 13.** *Let $x$, $y$, $z \in \Sigma_b^*$, and let $g \coloneqq \gcd([y]_b, b-1)$, $X \coloneqq ([y]_b + (b-1)[x]_b)/g$, and $Y \coloneqq (b^{|z|}[y]_b - (b-1)[z]_b)/g$. If $b$, $X$, and $Y$ are all squares and $\sqrt{b^{|z|}X} - \sqrt{Y} > (b-1)/g$, then all numbers of the form $[xy^*z]_b$ are composite.*

*Proof.* Evaluating the base-$b$ expansion of $xy^nz$, we get

$$\begin{aligned}[xy^nz]_b &= b^{|z|+n}[x]_b + b^{|z|}\frac{b^n-1}{b-1}[y]_b + [z]_b \\ &= \frac{b^{|z|+n}X - Y}{(b-1)/g}.\end{aligned}$$

Since $b$, $X$, and $Y$ are all squares the numerator factors as a difference of squares. By the size condition both factors are strictly larger than the denominator, and so the factorization is nontrivial. □

**Example 14.** Let $b \coloneqq 16$, $x \coloneqq \mathtt{4}$, $y \coloneqq \mathtt{4}$, and $z \coloneqq \mathtt{1}$. Then $g = 1$, $X = 8^2$, $Y = 7^2$, and

$$[\mathtt{44}^n\mathtt{1}]_b = \frac{(4^{n+1}\cdot 8 + 7)(4^{n+1}\cdot 8 - 7)}{15}.$$

Since $4 \cdot 8 - 7 > 15$, this factorization is nontrivial and no number of the form $[\mathtt{44}^*\mathtt{1}]_b$ is prime.

It is also possible to combine Lemma 9 with Corollary 6 to construct a test which also applies to bases which are not squares.

**Corollary 15.** *Using the same setup as in Lemma 9, if $b^{|z|}X$ and $Y$ are squares, $\sqrt{b^{|z|}X} - \sqrt{Y} > (b-1)/g$, and $1 < \gcd([xyz]_b, [xy^3z]_b) < [xz]_b$, then all numbers of the form $[xy^*z]_b$ are composite.*

*Proof.* Say $n = 2m$ is even. Then from the factorization in Lemma 9,

$$[xy^nz]_b = \frac{(b^m\sqrt{b^{|z|}X} + \sqrt{Y})(b^m\sqrt{b^{|z|}X} - \sqrt{Y})}{(b-1)/g}$$

which is nontrivial by the size condition.

Alternatively, if $n$ is odd then as in Corollary 6 we have that $\gcd([xyz]_b, [xy^3z]_b)$ divides $[xy^nz]_b$, and by the size condition this divisor is nontrivial. □

**Example 16.** Let $b \coloneqq 17$, $x \coloneqq \mathtt{19}$, $y \coloneqq \mathtt{9}$, and $z \coloneqq \mathtt{9}$. Then $g = 1$, $b^{|z|}X = 85^2$, $Y = 3^2$, and

$$[xy^{2n}z]_b = \frac{(17^n \cdot 85 + 3)(17^n \cdot 85 - 3)}{16}.$$

Since $85 - 3 > 16$ this factorization is nontrivial. Furthermore, all numbers of the form $[xy^{2n+1}z]_b$ are even, so all numbers of the form $[\mathtt{199}^*\mathtt{9}]_b$ are composite.

Finally, we present a variant of Lemma 13 which applies to a difference-of-cubes factorization.

**Lemma 17.** *Let $x$, $y$, $z \in \Sigma_b^*$, and let $g \coloneqq \gcd([y]_b, b-1)$, $X \coloneqq ([y]_b + (b-1)[x]_b)/g$, and $Y \coloneqq (b^{|z|}[y]_b - (b-1)[z]_b)/g$. If $b$, $X$, and $Y$ are all cubes and $\sqrt[3]{b^{|z|}X} - \sqrt[3]{Y} > (b-1)/g$, then all numbers of the form $[xy^*z]_b$ are composite.*

*Proof.* As in Lemma 13, we have

$$[xy^nz]_b = \frac{\big((b^{|z|+n}X)^{1/3} - Y^{1/3}\big)\big((b^{|z|+n}X)^{2/3} + (b^{|z|+n}XY)^{1/3} + Y^{2/3}\big)}{(b-1)/g}.$$

The second factor is at least as large as the first (except in the single case $b^{|z|+n}X = 1$ and $Y = -1$, which is not possible by construction of $X$ and $Y$), so by the size condition both factors are strictly larger than the denominator, and the factorization is nontrivial. □

**Example 18.** Let $b \coloneqq 8$, $x \coloneqq \mathtt{1}$, $y \coloneqq \mathtt{0}$, and $z \coloneqq \mathtt{1}$. Then $g = 7$, $X = 1$, $Y = -1$, and

$$[\mathtt{10}^n\mathtt{1}]_b = (2^{n+1} + 1)(4^{n+1} - 2^{n+1} + 1).$$

Since $2 - (-1) > 1$, this factorization is nontrivial and no number of the form $[\mathtt{10}^*\mathtt{1}]_b$ is prime.

# 5 Our heuristic algorithm

As previously mentioned, in practice to compute $M(L_b)$ one works with an underapproximation $M$ of $M(L_b)$ and an overapproximation $L$ of $L_b - \sup(M)$. One then refines such approximations until $L = \emptyset$ from which it follows that $M = M(L_b)$.

For the initial approximation, note that every minimal prime in base $b$ with at least 4 digits is of the form $xY^*z$, where $x \in \Sigma_b - \{\mathtt{0}\}$, $z \in \Sigma_b$, and

$$Y \coloneqq \Sigma_b - \{\, y : (p)_b \triangleleft xyz \text{ for some prime } p \,\}.$$

Making use of this, our algorithm sets $M$ to be the set of base-$b$ representations of the minimal primes with at most 3 digits (which can be found simply by brute force) and $L$ to be $\bigcup_{x,z} xY^*z$, as described above.

All remaining minimal primes are members of $L$, so to find them we explore the families in $L$. During this process, each family will be decomposed into possibly multiple other families. For example, a simple way of exploring the family $xY^*z$ where $Y \coloneqq \{y_1, \dots, y_n\}$ is to decompose it into the families $xY^*y_1z$, …, $xY^*y_nz$. If the smallest member (say $xy_iz$) of any such family happens to be prime, it can be added to $M$ and the family $xY^*y_iz$ removed from consideration. Furthermore, once $M$ has been updated it may be possible to simplify some families in $L$. In this case, $xY^*y_jz$ (for $j \neq i$) can be simplified to $x(Y - \{y_i\})^*y_jz$ since no minimal prime contains $xy_iz$ as a proper subword.

Another way of decomposing the family $xY^*z$ is possible if one knows that a digit of $Y$ can only occur a certain number of times. For example, if $xy_iy_iz$ has a proper prime subword then the digit $y_i$ can occur at most once in any minimal prime of the form $xY^*z$, and we can split $xY^*z$ into the two families

$$x(Y-\{y_i\})^*z \qquad \text{and} \qquad x(Y-\{y_i\})^*y_i(Y-\{y_i\})^*z.$$

Lastly, the family $xY^*z$ can be decomposed by considering digits of $Y$ which are mutually incompatible, i.e., they cannot occur simultaneously in a minimal prime. For example, if $xy_iy_jz$ and $xy_jy_iz$ ($i \neq j$) both have proper prime subwords then the digits $y_i$ and $y_j$ cannot occur simultaneously in any minimal prime of the form $xY^*z$, and we can split $xY^*z$ into the two families

$$x(Y-\{y_i\})^*z \qquad \text{and} \qquad x(Y-\{y_j\})^*z.$$

Sometimes it is not possible to show two digits are mutually incompatible, but it is possible to know that one digit must appear before the other. For example, if $xy_iy_jz$ has a proper prime subword then the digit $y_j$ must appear before $y_i$ in any minimal prime of the form $xY^*z$, and we can replace $xY^*z$ with the family

$$x(Y-\{y_i\})^*(Y-\{y_j\})^*z.$$

Similarly, if $xy_jyy_iz$ has a proper prime subword then we can split $xY^*yY^*z$ into the two families

$$x(Y-\{y_i\})^*yY^*z \qquad \text{and} \qquad xY^*y(Y-\{y_j\})^*z.$$

We now formulate these arguments for the most general kind of family we need to consider, namely $x_1L_1^*\cdots x_mL_m^*$. For simplicity, we only specify the decompositions as applying to $L_1 \coloneqq \{y_1,\ldots,y_n\}$, but it is straightforward to generalize these decompositions to also apply to $L_i$ for any $1 \leq i \leq m$.

**Lemma 19.** *Every minimal prime of the form $x_1L_1^*\cdots x_mL_m^*$ must also be of the form $x_1x_2L_2^*\cdots x_mL_m^*$ or $x_1y_iL_1^*x_2L_2^*\cdots x_mL_m^*$ for some $1 \leq i \leq n$.*

*Proof.* Follows from the fact that $L_1^* = \{\epsilon\} \cup \bigcup_{i=1}^n y_iL_1^*$. □

Similarly, one can generalize Lemma 19 to apply to adding characters to the right of $L_1$ rather than to the left.

**Example 20.** The family $\texttt{10}\{\texttt{0},\texttt{1}\}^*\texttt{61}^*\texttt{1}$ splits into the three families $\texttt{1061}^*\texttt{1}$, $\texttt{10}\{\texttt{0},\texttt{1}\}^*\texttt{061}^*\texttt{1}$, and $\texttt{10}\{\texttt{0},\texttt{1}\}^*\texttt{161}^*\texttt{1}$ by exploring $\{\texttt{0},\texttt{1}\}^*$ on the right.

**Lemma 21.** *If $x_1y_i^kx_2\cdots x_m$ contains a prime proper subword (for some $k \geq 1$) then every minimal prime of the form $x_1L_1^*\cdots x_mL_m^*$ is of the form*

$$x_1(L_1-\{y_i\})^*(y_i(L_1-\{y_i\})^*)^jx_2L_2^*\cdots x_mL_m^*$$

*for some $0 \leq j < k$.*

*Proof.* If $w \in x_1L_1^* \cdots x_mL_m^*$ then $w \in x_1yx_2L_2^* \cdots x_mL_m^*$ for some $y \in L_1^*$. If $y$ contains $k$ or more instances of $y_i$ then by assumption it follows that $w$ contains a proper prime subword, and therefore is not a minimal prime. So if $w$ is a minimal prime then $y$ must contain less than $k$ instances of $y_i$, i.e., $y$ must be of the form $(L_1 - \{y_i\})^*(y_i(L_1 - \{y_i\})^*)^j$ for some $0 \leq j < k$, from which the claim follows. □

**Example 22.** The string `661` represents a prime in base 9, and is a proper subword of `10661`. It follows that the family $\texttt{10}\{\texttt{0},\texttt{1},\texttt{6}\}^*\texttt{1}$ splits into the families $\texttt{10}\{\texttt{0},\texttt{1}\}^*\texttt{1}$ and $\texttt{10}\{\texttt{0},\texttt{1}\}^*\texttt{6}\{\texttt{0},\texttt{1}\}^*\texttt{1}$ in base 9.

Lemma 21 is especially useful when it can be applied with $k = 1$, since in that case the family $x_1L_1^* \cdots x_mL_m^*$ is replaced by a single strictly simpler family, in contrast to the other lemmas we will describe.

**Lemma 23.** *If $x_1y_iy_jx_2 \cdots x_m$ and $x_1y_iy_jx_2 \cdots x_m$ contain prime proper subwords (where $i \neq j$) then every minimal prime of the form $x_1L_1^* \cdots x_mL_m^*$ is of the form*

$$x_1(L_1 - \{y_i\})^*x_2L_2^* \cdots x_mL_m^* \qquad \textit{or} \qquad x_1(L_1 - \{y_j\})^*x_2L_2^* \cdots x_mL_m^*.$$

*Proof.* If $w \in x_1L_1^* \cdots x_mL_m^*$ then $w \in x_1yx_2L_2^* \cdots x_mL_m^*$ for some $y \in L_1^*$. If $y$ contains both $y_i$ and $y_j$ then by assumption it follows that $w$ contains a proper prime subword, and therefore is not a minimal prime. So if $w$ is a minimal prime then $y$ cannot contain $y_i$ and $y_j$ simultaneously, i.e., $y$ must be of the form $(L - \{y_i\})^*$ or $(L - \{y_j\})^*$, from which the claim follows. □

**Example 24.** The string `4611` represents a prime in base 8, and is a proper subword of `446411` and `444611`. It follows that the family $\texttt{44}\{\texttt{4},\texttt{6}\}^*\texttt{11}$ splits into the families $\texttt{444}^*\texttt{11}$ and $\texttt{446}^*\texttt{11}$ in base 8.

**Lemma 25.** *If $x_1y_iy_jx_2 \cdots x_m$ contains a prime proper subword (where $i \neq j$) then every minimal prime of the form $x_1L_1^* \cdots x_mL_m^*$ is of the form*

$$x_1(L_1 - \{y_i\})^*(L_1 - \{y_j\})^*x_2L_2^* \cdots x_mL_m^*.$$

*Proof.* If $w \in x_1L_1^* \cdots x_mL_m^*$ then $w \in x_1yx_2L_2^* \cdots x_mL_m^*$ for some $y \in L_1^*$. If $y$ contains $y_i$ before $y_j$ then by assumption it follows that $w$ contains a proper prime subword, and therefore is not a minimal prime. So if $w$ is a minimal prime then $y$ cannot contain $y_i$ before $y_j$, i.e., $y$ must be of the form $(L - \{y_i\})^*(L - \{y_j\})^*$, from which the claim follows. □

**Example 26.** The string `10` represents a prime in base 11, and is a proper subword of `90101`. It follows that the family $\texttt{90}\{\texttt{0},\texttt{1},\texttt{9}\}^*\texttt{1}$ splits into the family $\texttt{90}\{\texttt{0},\texttt{9}\}^*\{\texttt{1},\texttt{9}\}^*\texttt{1}$ in base 11.

**Lemma 27.** *If $x_1y_ix_2y_{2,j}x_3 \cdots x_m$ contains a prime proper subword (where $y_{2,j} \in L_2$) then every minimal prime of the form $x_1L_1^* \cdots x_mL_m^*$ is of the form*

$$x_1(L_1 - \{y_i\})^*x_2L_2^* \cdots x_mL_m^* \qquad \textit{or} \qquad x_1L_1x_2(L_2 - \{y_{2,j}\})^*x_3L_3^* \cdots x_mL_m^*.$$

*Proof.* If $w \in x_1L_1^* \cdots x_mL_m^*$ then $w \in x_1yx_2y'x_3L_3^* \cdots x_mL_m^*$ for some $y \in L_1^*$ and $y' \in L_2^*$. If $y$ contains $y_i$ and $y'$ contains $y_{2,j}$ then by assumption it follows that $w$ contains a proper prime subword, and therefore is not a minimal prime. So if $w$ is a minimal prime then either $y$ cannot contain $y_i$ or $y'$ cannot contain $y_{2,j}$, i.e., $y$ is of the form $(L_1 - \{y_i\})^*$ and $y'$ is of the form $L_2^*$, or $y$ is of the form $L_1^*$ and $y'$ is of the form $(L_2 - \{y_{2,j}\})^*$, from which the claim follows. □

**Example 28.** The string 60411 represents a prime in base 8, and is a proper subword of 604101. It follows that the family $\texttt{60}\{\texttt{0},\texttt{4}\}^*\texttt{10}^*\texttt{1}$ splits into the families $\texttt{600}^*\texttt{10}^*\texttt{1}$ and $\texttt{60}\{\texttt{0},\texttt{4}\}^*\texttt{11}$ in base 8.

We call families of the form $xy^*z$ (where $x$, $z \in \Sigma_b^*$ and $y \in \Sigma_b$) *simple* families. Our algorithm then proceeds as follows:

1. $M := \{\text{minimal primes in base } b \text{ of length} \leq 3\}$
   $L := \bigcup_{x,z\in\Sigma_b} xY^*z$, where $x \neq \texttt{0}$ and $Y$ is the set of digits $y$ such that $xyz$ has no subword in $M$

2. While $L$ contains non-simple families:

   (a) Explore each family of $L$ by applying Lemma 19, and update $L$.

   (b) Examine each family of $L$:

      i. Let $w$ be the shortest string in the family. If $w$ has a subword in $M$, then remove the family from $L$. If $w$ represents a prime, then add $w$ to $M$ and remove the family from $L$.

      ii. If possible, simplify the family by applying Lemma 21 with $k = 1$.

      iii. Using the techniques of Section 4, check if the family can be proven to only contain composites, and if so then remove the family from $L$.

   (c) Apply Lemmas 21, 23, 25, and 27 to the families of $L$ as much as possible and update $L$; after each split examine the new families as in (b).

The process of exploring/examining/splitting a family can be concisely expressed in a tree of decompositions. A sample tree of decompositions, for the family $\texttt{1}\{\texttt{0},\texttt{1},\texttt{6}\}^*\texttt{1}$ in base 9, is displayed in Figure 1. When applying Lemma 19 on a family with only one nonempty $L_i$ we don't show the first decomposition (simply removing $L_i^*$) since that results in the shortest word in the family, which is always implicitly checked for primality/minimal subwords at each step anyway.

## 5.1 Implementation

An implementation of the above algorithm was written in C using the GMP library [5] by the first author [2]. Independently, the second author wrote an implementation using the same ideas and derived the same results (except for a less extensive search for primes in the simple families).

Note that when exploring the families in line (a), one should not always apply Lemma 19 directly as stated by adding characters to the left of $L_1$; in our implementation we alternate between adding characters to the left and right of $L_{1+k \bmod m}$, where $k$ is the number of times we have previously passed through line (a). Also, the application of Lemma 25 tended to initially produce an excess number of cases, so it was useful to only apply it following a certain number of iterations (6 in our implementation). It also led to duplicate families after simplifying, for example families of the form $xy^*y^ny^*z$ for varying $n$, but these could be detected and removed.

At the conclusion of the algorithm described, $L$ will consist of simple families (of the form $xy^*z$) which have not yet yielded a prime, but for which there is no obvious reason why there can't be a prime of such a form. In such a case, the only way to proceed is to test the primality of larger and larger numbers of such form and hope a prime is eventually discovered.

The numbers in simple families are of the form $(ab^n + c)/d$ for some fixed $a$, $b$, $c$, $d$ where $d \mid ab^n + c$ for all $n$. Except in the special case $c = \pm 1$ and $d = 1$, when $n$ is large the known primality tests for such a number are too inefficient to run. In this case one must resort to a pseudoprimality test such as a Miller–Rabin test, unless a divisor of the number can be found. Since we are testing many numbers in an exponential sequence, it is possible to use a sieving process to find divisors rather than using trial division.

To do this, we made use of Geoffrey Reynolds' `srsieve` software [14]. This program uses the baby-step giant-step algorithm to find all primes $p$ which divide $ab^n + c$ where $p$ and $n$ lie in a specified range. Since this program cannot handle the general case $(ab^n + c)/d$ when $d > 1$ we only used it to sieve the sequence $ab^n + c$ for primes $p \nmid d$, and ititialized the list of candidates to not include $n$ for which there is some prime $p \mid d$ for which $p \mid (ab^n + c)/d$. The program had to be modified slightly to remove a check which would prevent it from running in the case when $a$, $b$, and $c$ were all odd (since then $2 \mid ab^n + c$).

Once the numbers with small divisors had been removed, it remained to test the remaining numbers using a pseudoprimality test. For this we used the software `LLR` by Jean Penné [13]. Although undocumented, it is possible to run this program on numbers of the form $(ab^n+c)/d$ when $d \neq 1$, so this program required no modifications. A script was also written which allowed one to run `srsieve` while `LLR` was testing the remaining candidates, so that when a divisor was found by `srsieve` on a number which had not yet been tested by `LLR` it would be removed from the list of candidates.

In the cases where the elements of $M(L_b)$ could be proven prime rigorously, we employed `PRIMO` by Marcel Martin [11], an elliptic curve primality proving implementation.

# 6 Results

A summary of the results of our algorithm is presented in Figure 2; it was able to completely solve all bases up to 30 except for bases 26, 28, and the odd bases larger than 16. The results in base 29 required some additional strategies, as described in Section 7.

For every solved base, we give the size $|M(L_b)|$ and the "width" $\max_{x \in M(L_b)} |x|$ of the

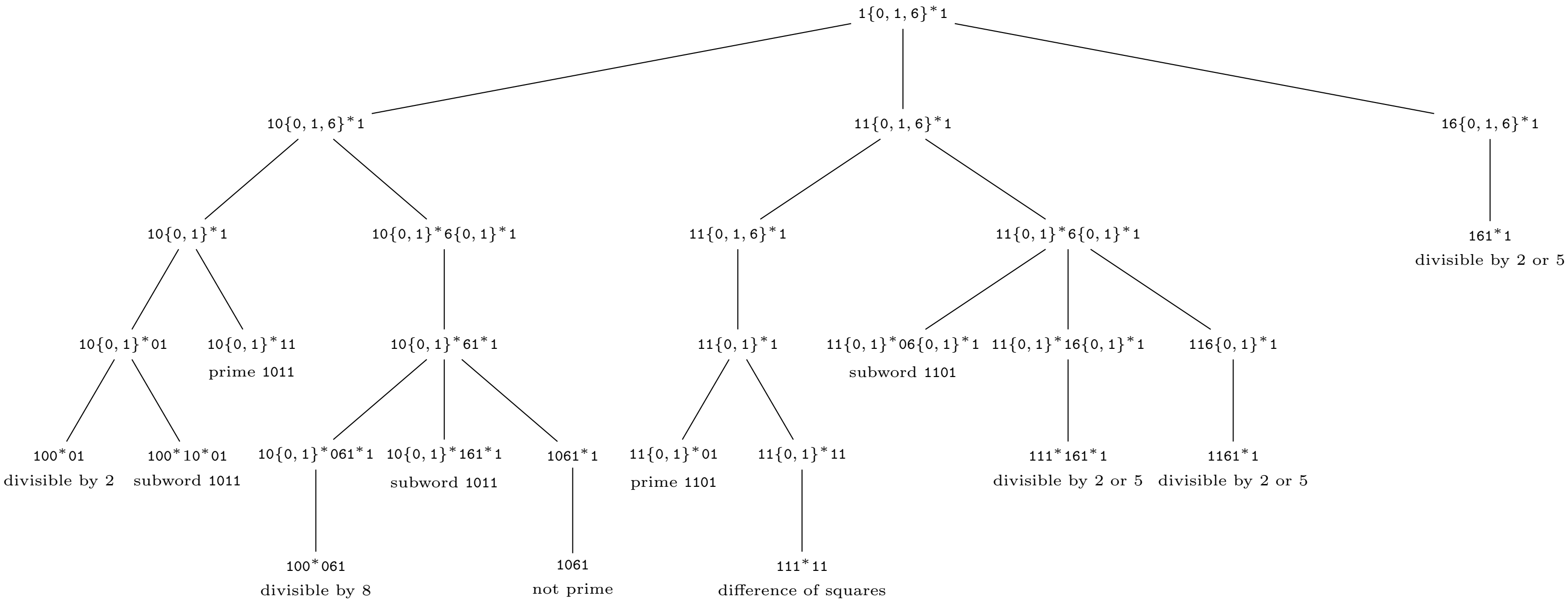

Figure 1: Tree of decompositions for $\texttt{1\{0,1,6\}}^*\texttt{1}$ in base 9.

corresponding family. For the unsolved bases, we give a lower bound on the size and width of $M(L_b)$, along with the number of families of the form $xy^*z$ for which no prime member could be found, nor could the family be ruled out as only containing composites. Since such simple families can contain at most one minimal prime, an upper bound on $|M(L_b)|$ is given by the sum of the lower bound for $|M(L_b)|$ and the number of unsolved families. Additionally, we give the height to which the simple families were searched for primes; if there are any more primes in $M(L_b)$ they must have at least this many digits in base $b$.

The family $\texttt{80}^*\texttt{1}$ in base 23, corresponding to the generalized Proth numbers $8{\cdot}23^n{+}1$, was already known to be prime for minimal $n = 119215$ in the process of solving the generalized Sierpiński conjecture in base 23 [1].

## 6.1 Unsolved families

We now explicitly list the 40 families for which we were unable to determine if they contain a prime or not.

Base 17: $\texttt{F19}^*$
Base 19: $\texttt{EE16}^*$
Base 21: $\texttt{CF}^*\texttt{0K}$, $\texttt{G0}^*\texttt{FK}$
Base 23: $\texttt{9E}^*$
Base 25: $\texttt{6MF}^*\texttt{9}$, $\texttt{CM1}^*$, $\texttt{EE1}^*$, $\texttt{E1}^*\texttt{E}$, $\texttt{EFO}^*$, $\texttt{F1}^*\texttt{F1}$, $\texttt{F0}^*\texttt{KO}$, $\texttt{F0K}^*\texttt{O}$, $\texttt{LOL}^*\texttt{8}$, $\texttt{M1}^*\texttt{F1}$, $\texttt{M10}^*\texttt{8}$, $\texttt{OL}^*\texttt{8}$
Base 26: $\texttt{A}^*\texttt{6F}$, $\texttt{I}^*\texttt{GL}$
Base 27: $\texttt{80}^*\texttt{9A}$, $\texttt{999G}^*$, $\texttt{CL}^*\texttt{E}$, $\texttt{EI}^*\texttt{F8}$, $\texttt{F}^*\texttt{9FM}$
Base 28: $\texttt{OA}^*\texttt{F}$
Base 29: $\texttt{1A}^*$, $\texttt{68LO}^*\texttt{6}$, $\texttt{AMP}^*$, $\texttt{C}^*\texttt{FK}$, $\texttt{F}^*\texttt{OPF}$, $\texttt{FKI}^*$, $\texttt{F}^*\texttt{OP}$, $\texttt{LPO9}^*$, $\texttt{OOPS}^*\texttt{A}$, $\texttt{OO}^*\texttt{FPL}$, $\texttt{PC}^*$, $\texttt{PPPL}^*\texttt{O}$, $\texttt{Q}^*\texttt{GL}$, $\texttt{Q}^*\texttt{LO}$, $\texttt{RM}^*\texttt{G}$

## 6.2 Primes of the form $4n + 1$ and $4n + 3$

The heuristic algorithm described in Section 5 may also be easily modified to apply to recursive subsets of prime numbers, e.g., those congruent to 1 mod 4 (alternatively, 3 mod 4). The only modifications necessary are to the initialization of $M$ in line 1, and to update the check "$w$ represents a minimal prime" to also check that $w$ is in the subset under consideration.

The lemmas of Section 4 can be used without modification, since a family which contains only composites of course does not contain any primes of a special form, either. For simplicity the lemmas of Section 5 are stated to apply when the minimal set consists of primes, but actually apply to general minimal sets and also need no modification.

When run on the primes of the form $4n + 1$ represented in base 10, our implementation successfully computed the minimal set consisting of 146 elements, the largest of which contained 79 digits. The *Encyclopedia of Integer Sequences* [12] contains the elements for this minimal set in entry A111055. Prior to our work, the longest 11 elements were missing from this listing.

When run on the primes of the form $4n+3$ represented in base 10, our implementation successfully computed the minimal set consisting of 113 elements, the largest of which contained 19153 digits. Because of its size, the largest element has not yet been subjected to a rigorous primality test. The second-largest element has 50 digits, so the remaining elements can easily be proved prime. The *Encyclopedia of Integer Sequences* [12] contains the elements for this minimal set in entry A111056. Prior to our work, the longest 10 elements were missing from this listing.

# 7 Some additional strategies

The strategies discussed so far suffice to restrict the possible forms of minimal primes to a finite number of simple families in all bases $2 \le b \le 28$. However, as $b$ increased, in addition to the calculations becoming more costly, it was found to be necessary to use increasingly complicated strategies. We now describe some additional strategies which we found sufficient to solve all non-simple families in base 29.

**Lemma 29.** *If every number of the form $x_1(L_1 - y_i)^* x_2 L_2^* \cdots x_m L_m^*$ is composite, then every minimal prime of the form $x_1 L_1^* \cdots x_m L_m^*$ must also be of the form $x_1 L_1^* y_i L_1^* x_2 L_2^* \cdots x_m L_m^*$.*

*Proof.* If $w \in x_1 L_1^* \cdots x_m L_m^*$ then $w \in x_1 y x_2 L_2^* \cdots x_m L_m^*$ for some $y \in L_1^*$. If $y$ does not contain a $y_i$ then $w$ is composite by assumption. Therefore if $w$ is a prime then $y$ contains a $y_i$, i.e., $y \in L_1^* y_i L_1^*$, from which the result follows. (Note that one could improve this result via $y \in (L_1 - y_i)^* y_i L_1^* \cup L_1^* y_i (L_1 - y_i)^*$, but this was found to be unnecessary for our purposes.) □

**Example 30.** The numbers represented by the family $\texttt{F}\{\texttt{0},\texttt{9},\texttt{F}\}^*\texttt{F}$ in base 29 are divisible by 3, so every minimal prime of the form $\texttt{F}\{\texttt{0},\texttt{9},\texttt{F},\texttt{P}\}^*\texttt{F}$ must also be of the form $\texttt{F}\{\texttt{0},\texttt{9},\texttt{F},\texttt{P}\}^*\texttt{P}\{\texttt{0},\texttt{9},\texttt{F},\texttt{P}\}^*\texttt{F}$.

**Lemma 31.** *If $x_1 y_i y_j y_i x_2 \cdots x_m$ contains a prime proper subword (where $i \ne j$) then every minimal prime of the form $x_1 L_1^* \cdots x_m L_m^*$ is of the form*

$$x_1(L_1 - \{y_i\})^*(L_1 - \{y_j\})^*(L_1 - \{y_i\})^* x_2 L_2^* \cdots x_m L_m^*.$$

*Proof.* If $w \in x_1 L_1^* \cdots x_m L_m^*$ then $w \in x_1 y x_2 L_2^* \cdots x_m L_m^*$ for some $y \in L_1^*$. If $y$ contains $y_i y_j y_i$ then by assumption it follows that $w$ contains a proper prime subword, and therefore is not a minimal prime. So if $w$ is a minimal prime then either $y$ does not contain a $y_j$, or $y$ contains a $y_j$ and all $y_i$s in $y$ either come before or after the $y_j$. In each case, $y$ is of the form $(L - \{y_i\})^*(L - \{y_j\})^*(L - \{y_i\})^*$, from which the claim follows. □

**Example 32.** The string $\texttt{QLQ}$ represents a prime in base 29, and is a proper subword of $\texttt{LQLQL}$. It follows that the family $\texttt{L}\{\texttt{L},\texttt{Q}\}^*\texttt{L}$ splits into the family $\texttt{LL}^*\texttt{Q}^*\texttt{L}^*\texttt{L}$ in base 29.

This rule may also be generalized to apply to the case when $x_1 y_i y_j y_i y_j x_2 \cdots x_m$ contains a prime proper subword (where $i \ne j$).

| $b$ | $\lvert M(L_b)\rvert$ | $\max_{x\in M(L_b)} \lvert x\rvert$ | # unsolved families | searched height |
|---|---|---|---|---|
| 2 | 2 | 2 | 0 | – |
| 3 | 3 | 3 | 0 | – |
| 4 | 3 | 2 | 0 | – |
| 5 | 8 | 5 | 0 | – |
| 6 | 7 | 5 | 0 | – |
| 7 | 9 | 5 | 0 | – |
| 8 | 15 | 9 | 0 | – |
| 9 | 12 | 4 | 0 | – |
| 10 | 26 | 8 | 0 | – |
| 11 | 152 | 45 | 0 | – |
| 12 | 17 | 8 | 0 | – |
| 13$^*$ | 228 | 32021 | 0 | – |
| 14 | 240 | 86 | 0 | – |
| 15 | 100 | 107 | 0 | – |
| 16 | 483 | 3545 | 0 | – |
| 17$^*$ | $\geq 1279$ | $\geq 111334$ | 1 | 740000 |
| 18 | 50 | 33 | 0 | – |
| 19$^*$ | $\geq 3462$ | $\geq 110986$ | 1 | 523000 |
| 20 | 651 | 449 | 0 | – |
| 21$^*$ | $\geq 2599$ | $\geq 47336$ | 2 | 260000 |
| 22 | 1242 | 764 | 0 | – |
| 23$^*$ | $\geq 6020$ | $\geq 119216$ | 1 | 661000 |
| 24 | 306 | 100 | 0 | – |
| 25$^*$ | $\geq 17597$ | $\geq 136967$ | 12 | 187000 |
| 26$^*$ | $\geq 5662$ | $\geq 8773$ | 2 | 254000 |
| 27$^*$ | $\geq 17210$ | $\geq 109006$ | 5 | 203000 |
| 28$^*$ | $\geq 5783$ | $\geq 94538$ | 1 | 351000 |
| 29$^*$ | $\geq 57282$ | $\geq 123420$ | 15 | 139000 |
| 30 | 220 | 1024 | 0 | – |

$^*$Data based on results of pseudoprimality tests.

Figure 2: Summary of results for the prime numbers for each base $b$.

**Example 33.** The string LL9L9LQL represents a prime in base 29. It follows that the family LL{9,L}*Q*QL splits into the family LLL*9*L*9*Q*QL in base 29.

In Section 4.1 we described a number of strategies for determining if every member of a family has a divisor, but for some families divisors exist which will not be found using those tests. The following lemma can help one discover when this is the case; in particular when every member of a family is divisible by some small prime. For a language $L$ we use the notation $[L]_b \bmod N$ to denote the finite set of residues $\{\, [x]_b \bmod N : x \in L \,\}$.

**Lemma 34.** *If* $1 < \gcd(n, N)$ *for every* $n \in [L]_b \bmod N$ *and* $p \notin [L]_b$ *for every prime* $p$ *which divides* $N$*, then all numbers of the form* $[L]_b$ *are composite.*

*Proof.* Let $x$ be an arbitrary member of $L$. Then $\gcd([x]_b, N) = \gcd([x]_b \bmod N, N) > 1$ by assumption, so $[x]_b$ cannot be prime unless $[x]_b = \gcd([x]_b, N)$. But in that case $[x]_b$ would be a prime which divides $N$, and so $[x]_b \notin [L]_b$ by the second assumption, in contradiction to $x \in L$. □

To make use of this lemma, we need to be able to compute $[x_1 L_1^* \cdots x_m L_m^*]_b \bmod N$. To do this, we can make use of the following relations:

$$[Lx]_b \bmod N = (b \cdot ([L]_b \bmod N) + [x]_b) \bmod N$$

$$[L\{y_1, \ldots, y_n\}]_b \bmod N = \bigcup_{i=1}^{n} [Ly_i]_b \bmod N$$

$$[L\{y_1, \ldots, y_n\}^*]_b \bmod N = \bigcup_{i=0}^{\infty} [L\{y_1, \ldots, y_n\}^i]_b \bmod N$$

In the final case, the union may be taken to be finite over $i < k$ where $k$ is chosen such that $\bigcup_{i=0}^{k} [L\{y_1, \ldots, y_n\}^i]_b \bmod N = \bigcup_{i=0}^{k-1} [L\{y_1, \ldots, y_n\}^i]_b \bmod N$. Using these relations, we can compute $[x_1 L_1^* \cdots x_m L_m^*]_b \bmod N$ progressively, working left to right and starting from $[\emptyset]_b \bmod N = \{0\}$. To solve base 29 it was sufficient to use $N = 2 \cdot 3 \cdot 5$.

**Example 35.** Let $b \coloneqq 29$ and $N \coloneqq 30$. Then $[\texttt{L1*61*LK*K}]_b \bmod N = \{4, 5, 6, 14, 15, 16, 25\}$, so if $n$ is in this set then $\gcd(n, N) \in \{2, 5, 6, 15\}$ and $\gcd(n, N) > 1$. Since 2, 3, $5 \notin [\texttt{L1*61*LK*K}]_b$, by Lemma 34 all numbers of the form L1*61*LK*K are composite.

# 8 Composite numbers

One can also consider the companion problem of determining the minimal elements for the composite numbers $\{4, 6, 8, 9, 10, 12, \ldots\}$. Here, in contrast with the primes, we have

**Theorem 36.** *The following decision problem is recursively solvable: given a base* $b$ *and a DFA* $M$ *accepting a language* $L$ *of strings in a base-*$b$ *canonical representation, does* $M$ *accept the base-*$b$ *representation of a composite number?*

This follows immediately from

**Theorem 37.** *Suppose $L$ is a regular language, accepted by a deterministic finite automaton $M$ of $n$ states. Then if $M$ accepts a composite number expressed in base $b$, it must accept one whose base-$b$ representation has at most $n(b^{2n}+1)$ digits.*

*Proof.* If $L$ is finite, then the longest string accepted by $M$ has at most $n-1$ digits, and $n-1 < n(b^{2n}+1)$.

Otherwise $L$ is infinite. Then $M$ accepts a string of length $\ell$ with $n < \ell \leq 2n$ that can be pumped (as in the pumping lemma). That is, there exists $x \in L$ such that $x = uvw$ with $|uv| \leq n$. Then a classical proof of the non-regularity of the prime numbers (e.g., [10, Example 3.2, p. 57]) shows that either $x$ is the representation of a composite number (in which case $|x| \leq 2n \leq n(b^{2n}+1)$) or $uv^pw$ is composite, where $p = [x]_b$. Our bound now follows. □

Write $S_b \coloneqq \{\, (n)_b : n \geq 4 \text{ is composite} \,\}$. For our purposes, the following theorem gives a better bound.

**Theorem 38.** *Every minimal element of $S_b$ is of length at most $b+2$.*

*Proof.* Consider any word $w$ of $S_b$ of length $\geq b+3$. Since there are only $b$ distinct digits, some digit $d$ is repeated at least twice, so that $dd \triangleleft w$. If $d > \mathtt{1}$, the number $[dd]_b$ is composite, as it is divisible by $[\mathtt{11}]_b$ but not equal to it. If $d = \mathtt{0}$, then some nonzero digit $c$ precedes it in $w$, so $c\mathtt{00} \triangleleft w$ and $[c\mathtt{00}]_b$ is divisible by $b^2$, which is composite. Finally, if no digit other than $\mathtt{1}$ is repeated, then $\mathtt{1111} \triangleleft w$, and $[\mathtt{1111}]_b = [\mathtt{11}]_b \cdot [\mathtt{101}]_b$, and hence is composite. □

Theorem 38 turns the computation of the minimal elements for $S_b$ into a finite search. Our results are given in Figure 3.

# 9 Conclusion and perspectives

We have (sometimes partially) solved the problem of finding the minimal elements for the prime numbers represented with the first bases. However, the problem rapidly becomes very complex: the number of members increases rapidly, some bases need sophisticated strategies, and some members are very long. Hence, to complete the cases where some (simple) families resisted our analysis, and to go further in the bases, we need better exploration strategies, and better primality tests for numbers expressed by a simple family in some base.

Another problem left open is to delineate the asymptotic behaviour of the size and width of those sets $M(L_b)$ when $b$ increases. A rough estimation is that the size increases exponentially, but this increase is lower when the base has many different factors.

To illustrate once more the difficulty of computing $M(L)$, we recall an open problem from [15]:

**Problem 39.** Let $L \coloneqq \{\mathtt{1}, \mathtt{2}, \mathtt{4}, \mathtt{8}, \mathtt{16}, \mathtt{32}, \mathtt{64}, \ldots\}$, the base-10 representation of the powers of 2. Is it the case that

$$M(L) = \{\mathtt{1}, \mathtt{2}, \mathtt{4}, \mathtt{8}, \mathtt{65536}\}?$$

| $b$ | $\lvert M(S_b)\rvert$ | $\max\limits_{x\in M(S_b)} \lvert x\rvert$ |
|---|---|---|
| 2 | 3 | 4 |
| 3 | 4 | 3 |
| 4 | 9 | 3 |
| 5 | 10 | 3 |
| 6 | 19 | 4 |
| 7 | 18 | 3 |
| 8 | 26 | 3 |
| 9 | 28 | 2 |
| 10 | 32 | 3 |
| 11 | 32 | 3 |
| 12 | 46 | 4 |
| 13 | 43 | 3 |
| 14 | 52 | 3 |
| 15 | 54 | 2 |
| 16 | 60 | 3 |
| 17 | 60 | 3 |
| 18 | 95 | 4 |
| 19 | 77 | 3 |
| 20 | 87 | 3 |
| 21 | 90 | 2 |
| 22 | 94 | 3 |
| 23 | 97 | 3 |
| 24 | 137 | 4 |
| 25 | 117 | 2 |
| 26 | 111 | 3 |
| 27 | 115 | 2 |
| 28 | 131 | 3 |
| 29 | 123 | 3 |
| 30 | 207 | 4 |

Figure 3: Summary of results for the composite numbers for each base $b$.

This would follow, for example, if we could prove that every power of 16 greater than 65536 contained at least one of the digits $\{\texttt{1},\texttt{2},\texttt{4},\texttt{8}\}$. But this seems beyond current capabilities.

## References

[1] G. Barnes. Sierpinski conjectures and proofs, http://www.noprimeleftbehind.net/crus/Sierp-conjectures.htm.

[2] C. Bright. MEPN implementation, https://github.com/curtisbright/mepn.

[3] S. L. G. Choi. Covering the set of integers by congruence classes of distinct moduli. *Math. Comp.* **25** (1971), 885–895.

[4] L. E. Dickson. *History of the Theory of Numbers.* Volume I: Divisibility and Primality. Chelsea Publishing Company, New York, 1952.

[5] T. Granlund et al. The GNU Multiple Precision Arithmetic Library, http://gmplib.org/.

[6] H. Gruber, M. Holzer, and M. Kutrib. The size of Higman-Haines sets. *Theoret. Comput. Sci.* **387** (2007), 167–176.

[7] H. Gruber, M. Holzer, and M. Kutrib. More on the size of Higman-Haines sets: effective constructions. *Fundam. Inform.* **91** (2009), 105–121.

[8] L. H. Haines. On free monoids partially ordered by embedding. *J. Combinatorial Theory* **6** (1969), 94–98.

[9] G. Higman. Ordering by divisibility in abstract algebras. *Proc. London Math. Soc.* (3) **2** (1952), 326–336.

[10] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, 1979.

[11] M. Martin. Primo for Linux, http://www.ellipsa.eu/public/primo/primo.html.

[12] OEIS Foundation Inc. The On-Line Encyclopedia of Integer Sequences, http://oeis.org.

[13] J. Penné. LLR Version 3.8.9, http://jpenne.free.fr/index2.html.

[14] G. Reynolds. Sierpinski/Riesel Sieve Version 0.6.17, https://sites.google.com/site/geoffreywalterreynolds/programs/srsieve.

[15] J. Shallit. Minimal primes. *J. Recreational Math.* **30** (2) (2001), 113–117.

[16] H. C. Williams and H. Dubner. The primality of $R_{1031}$. *Math. Comput.* **47** (1986), 703–711.