Efficiently factoring polynomials modulo p^4

Ashish Dwivedi * Rajat Mittal † Nitin Saxena ‡

Abstract

Polynomial factoring has famous practical algorithms over fields– finite, rational and p-adic. However, modulo prime powers, factoring gets harder because there is non-unique factorization and a combinatorial blowup ensues. For example, $x^2 + p \bmod p^2$ is irreducible, but $x^2 + px \bmod p^2$ has exponentially many factors! We present the first randomized poly$(\deg f, \log p)$ time algorithm to factor a given univariate integral $f(x)$ modulo p^k, for a prime p and $k \leq 4$. Thus, we solve the open question of factoring modulo p^3 posed in (Sircana, ISSAC'17).

Our method reduces the general problem of factoring $f(x) \bmod p^k$ to that of *root finding* in a related polynomial $E(y) \bmod \langle p^k, \varphi(x)^\ell \rangle$ for some irreducible $\varphi \bmod p$. We can efficiently solve the latter for $k \leq 4$, by incrementally transforming $E(y)$. Moreover, we discover an efficient refinement of Hensel lifting to lift factors of $f(x) \bmod p$ to those mod p^4 (if possible). This was previously unknown, as the case of repeated factors of $f(x) \bmod p$ forbids classical Hensel lifting.

2012 ACM CCS concept: Theory of computation– Algebraic complexity theory, Problems, reductions and completeness; Computing methodologies– Algebraic algorithms, Hybrid symbolic-numeric methods; Mathematics of computing– Number-theoretic computations.
Keywords: efficient, randomized, factor, local ring, prime-power, Hensel lift, roots, p-adic.

1 Introduction

Polynomial factorization is a fundamental question in mathematics and computing. In the last decades, quite efficient algorithms have been invented for various fields, e.g., over rationals [LLL82], number fields [Lan85], finite fields [Ber67, CZ81, KU11] and p-adic fields [Chi87, CG00]. Being a problem of fundamental theoretical and practical importance, it has been very well studied; for more background refer to surveys, e.g., [Kal92, vzGP01, FS15].

The same question over *composite* characteristic rings is believed to be computationally hard. For instance it is related to integer factoring [Sha93, Kli97]. What is less understood is factorization over a local *ring*; especially, ones that are the residue class rings of $\mathbb{Z}$ or $\mathbb{F}_q[z]$. A natural variant is as follows.

*CSE, Indian Institute of Technology, Kanpur, `ashish@cse.iitk.ac.in`
†CSE, Indian Institute of Technology, Kanpur, `rmittal@cse.iitk.ac.in`
‡CSE, Indian Institute of Technology, Kanpur, `nitin@cse.iitk.ac.in`

Problem: Given a univariate integral polynomial $f(x)$ and a prime power p^k, with p prime and $k \in \mathbb{N}$; output a nontrivial factor of $f \bmod p^k$ in randomized poly($\deg f, k \log p$) time.

Note that the polynomial ring $(\mathbb{Z}/\langle p^k\rangle)[x]$ is *not* a unique factorization domain. So $f(x)$ may have many, usually *exponentially* many, factorizations. For example, $x^2 + px$ has an irreducible factor $x + \alpha p \bmod p^2$ for each $\alpha \in [p]$ and so $x^2 + px$ has exponentially many (wrt $\log p$) irreducible factors modulo p^2. This leads to a total breakdown in the classical factoring methods.

We give the first randomized polynomial time algorithm to non-trivially factor (or test for irreducibility) a polynomial $f(x) \bmod p^k$, for $k \leq 4$.

Additionally, when $f \bmod p$ is power of an irreducible, we provide (and count) all the lifts $\bmod p^k$ ($k \leq 4$) of any factor of $f \bmod p$, in randomized polynomial time.

Usually, one factors $f(x) \bmod p$ and tries to "lift" this factorization to higher powers of p. If the former is a coprime factorization then Hensel lifting [Hen18] helps us in finding a non-trivial factorization of $f(x) \bmod p^k$ for any k. But, when $f(x) \bmod p$ is power of an irreducible then it is not known how to lift to some factorization of $f(x) \bmod p^k$. To illustrate the difficulty let us see some examples (also see [vzGH96]).

Example. [coprime factor case] Let $f(x) = x^2 + 10x + 21$. Then $f \equiv x(x + 1) \bmod 3$ and Hensel lemma lifts this factorization uniquely mod 3^2 as $f(x) \equiv (x + 1 \cdot 3)(x + 1 + 2 \cdot 3) \equiv (x + 3)(x + 7) \bmod 9$. This lifting extends to any power of 3.

Example. [power of an irreducible case] Let $f(x) = x^3 + 12x^2 + 3x + 36$ and we want to factor it mod 3^3. Clearly, $f \equiv x^3 \bmod 3$. By brute force one checks that, the factorization $f \equiv x \cdot x^2 \bmod 3$ lifts to factorizations mod 3^2 as: $x(x^2+3x+3)$, $(x+6)(x^2+6x+3)$, $(x+3)(x^2+3)$. Only the last one lifts to mod 3^3 as: $(x+3)(x^2+9x+3)$, $(x+12)(x^2+3)$, $(x+21)(x^2+18x+3)$.

So the big issue is: efficiently determine which factorization out of the exponentially many factorizations mod p^j will lift to mod p^{j+1}?

1.1 Previously known results

Using Hensel lemma it is easy to find a non-trivial factor of $f \bmod p^k$ when $f \bmod p$ has two coprime factors. So the hard case is when $f \bmod p$ is power of an irreducible polynomial. The first resolution in this case was achieved by [vzGH98] assuming that k is "large". They assumed k to be larger than the maximum power of p dividing the discriminant of the integral f. Under this assumption (i.e. k is large), they showed that factorization modulo p^k is well behaved and it corresponds to the unique p-adic factorization of f (refer p-adic factoring [Chi87, Chi94, CG00]). To show this, they used an extended version of Hensel lifting (also discussed in [BS86]). Using this observation they could also describe *all* the factorizations modulo p^k, in a compact data structure. The complexity of [vzGH98] was improved by [CL01].

The related questions of root finding and root counting of $f \bmod p^k$ are also of classical interest, see [NZM13, Apo13]. Root counting has interesting applications in arithmetic algebraic-geometry, for instance to compute Igusa zeta function of a univariate integral polynomial [ZG03, DH01]. A recent result by [BLQ13, Cor.4] resolves these problems in randomized polynomial time. Again, it describes *all* the roots modulo p^k, in a compact data structure. [NRS17] improved the time complexity of [BLQ13]. Very recently, [KRRZ19] also found a randomized poly-time algorithm which counts all the roots of $f \bmod p^k$.

Derandomizing root counting problem remained open until very recently. A partial derandomization of root counting algorithm has been obtained by [CGRW18] last year; which runs in deterministic poly-time when $k = O(\log \log p)$. Finally, [DMS19] gave a deterministic poly$(\deg(f), k \log p)$-time algorithm for the problem, which also generalizes to count all the basic irreducible factors of $f \bmod p^k$; taking a step closer towards irreducibility testing of $f \bmod p^k$.

Going back to factoring $f \bmod p^k$, [vzGH96] discusses the hurdles when k is small. The factors could be completely unrelated to the corresponding p-adic factorization, since an irreducible p-adic polynomial could reduce mod p^k when k is small. We give an example from [vzGH96].

Example. Polynomial $f(x) = x^2 + 3^k$ is irreducible over $\mathbb{Z}/\langle 3^{k+1} \rangle$ and so over 3-adic field. But, it is reducible mod 3^k as $f \equiv x^2 \bmod 3^k$.

von zur Gathen and Hartlieb also pointed out that the distinct factorizations are completely different and not nicely related, unlike the case when k is large. An example taken from [vzGH96] is,

Example. $f = (x^2+243)(x^2+6)$ is an irreducible factorization over $\mathbb{Z}/\langle 3^6 \rangle$. There is another completely unrelated factorization $f = (x + 351)(x + 135)(x^2 + 243x + 249) \bmod 3^6$.

Many researchers tried to solve special cases, especially when k is constant. The only successful factoring algorithm is by [Săl05] over $\mathbb{Z}/\langle p^2 \rangle$; it is actually related to Eisenstein criterion for irreducible polynomials. The next case, to factor modulo p^3, is unsolved and was recently highlighted in [Sir17].

1.2 Our results

We saw that even after the attempts of last two decades we do not have an efficient algorithm for factoring mod p^3. Naturally, we would like to first understand the difficulty of the problem when k is constant. In this direction we make significant progress by devising a unified method which solves the problem when $k = 2, 3$ or 4 (and sketch the obstructions we face when $k \geq 5$). Our first result is,

Theorem 1. *Let p be prime, $k \leq 4$ and $f(x)$ be a univariate integral polynomial. Then, $f(x) \bmod p^k$ can be factored (and tested for irreducibility) in randomized poly$(\deg f, \log p)$ time.*

Remarks. (1) The procedure to factor $f \bmod p^4$ also factors mod p^3 and mod p^2 (and tests for irreducibility) in randomized poly$(\deg f, \log p)$ time. This solves the open question of efficiently factoring $f \bmod p^3$ [Sir17] and generalizes [Săl05].

(2) Our method can as well be used to factor a 'univariate' polynomial $f \in \left(\mathbb{F}_p[z]/\langle \psi^k \rangle\right)[x]$, for $k \leq 4$ and irreducible $\psi(z) \bmod p$, in randomized poly$(\deg f, \deg \psi, \log p)$ time.

Next, we do more than just factoring f modulo p^k for $k \leq 4$. Given that f is power of an irreducible mod p (hard case for Hensel lemma). We show that our method works in this case to give all the lifts $g(x) \bmod p^k$ (possibly exponentially many) of any given factor $\tilde{g}$ of $f \bmod p$, for $k \leq 4$.

Theorem 2. *Let p be prime, $k \leq 4$ and $f(x)$ be a univariate integral polynomial such that $f \bmod p$ is a power of an irreducible polynomial. Let $\tilde{g}$ be a given factor of $f \bmod p$. Then, in randomized poly($\deg f, \log p$) time, we can compactly describe (and count) all possible factors of $f(x) \bmod p^k$ which are lifts of $\tilde{g}$ (or report that there is none).*

Remark. Theorem 2 can be seen as refinement of Hensel lifting method (Lemma 16) to $\mathbb{Z}/\langle p^k\rangle$, $k \leq 4$. To lift a factor f_1 of $f \bmod p$, Hensel lemma relies on a cofactor f_2 which is coprime to f_1. Our method needs no such assumption and it directly lifts a factor $\tilde{g}$ of $f \bmod p$ to (possibly exponentially many) factors $g(x) \bmod p^k$.

1.3 Proof technique– Root finding over local rings

Our proof involves two main techniques which may be of general interest.

Technique 1: Known factoring methods mod p work by first reducing the problem to that of root finding mod p. In this work, we efficiently reduce the problem of factoring $f(x)$ modulo the principal ideal $\langle p^k\rangle$ to that of finding roots of some polynomial $E(y) \in (\mathbb{Z}[x])[y]$ modulo a *bi-generated* ideal $\langle p^k, \varphi(x)^\ell\rangle$, where $\varphi(x)$ is an irreducible factor of $f(x) \bmod p$. This technique works for all $k \geq 1$.

Technique 2: Next, we find a root of the equation $E(y) \equiv 0 \bmod \langle p^k, \varphi(x)^\ell\rangle$, assuming $k \leq 4$. With the help of the special structure of $E(y)$ we will efficiently find all the roots y (possibly exponentially many) in the local ring $\mathbb{Z}[x]/\langle p^k, \varphi(x)^\ell\rangle$.

It remains open whether this technique extends to $k = 5$ and beyond (even to find a single root of the equation). The possibility of future extensions of our technique is discussed in Appendix D.

1.4 Proof overview

Proof idea of Theorem 1: Firstly, assume that the given degree d integral polynomial f satisfies $f(x) \equiv \varphi^e \bmod p$ for some $\varphi(x) \in \mathbb{Z}[x]$ which is irreducible mod p. Otherwise, using Hensel lemma (Lemma 16) we can efficiently factor $f \bmod p^k$.

Any factor of such an $f \bmod p^k$ must be of the form $(\varphi^a - py) \bmod p^k$, for some $1 \leq a < e$ and $y \in (\mathbb{Z}/\langle p^k\rangle)[x]$. In Theorem 8, we first reduce the problem of finding such a factor $(\varphi^a - py)$ of $f \bmod p^k$ to finding roots of some $E(y) \in (\mathbb{Z}[x])[y]$ in the local ring $\mathbb{Z}[x]/\langle p^k, \varphi^{ak}\rangle$. This is inspired by the p-adic power series expansion of the quotient $f/(\varphi^a - py)$. On going mod p^k we get a polynomial in y of degree $(k-1)$; which we want to be divisible by φ^{ak}.

The root y of $E(y) \bmod \langle p^k, \varphi^{ak}\rangle$ can be further decomposed into coordinates $y_0, y_1, \ldots$, $y_{k-1} \in \mathbb{F}_p[x]/\langle \varphi^{ak}\rangle$ such that $y =: y_0 + py_1 + \ldots + p^{k-1}y_{k-1} \bmod \langle p^k, \varphi^{ak}\rangle$. When we take $k = 4$, it turns out that the root y only depends on the coordinates y_0 and y_1 (i.e. y_2, y_3 can be picked arbitrarily).

Next, we reduce the problem of root finding of $E(y_0 + py_1)$ in the ring $\mathbb{Z}[x]/\langle p^4, \varphi^{4a}\rangle$ to root finding in characteristic p; of some $E'(y_0, y_1)$ in the ring $\mathbb{F}_p[x]/\langle \varphi^{4a}\rangle$ (Lemma 11). We make use of a subroutine ROOT-FIND given by [BLQ13] which can efficiently find all the roots of a univariate $g(y)$ in the ring $\mathbb{Z}/\langle p^j\rangle$. In fact, we need a slightly generalized version of it, to find all the roots of a given $g(y)$ in the ring $\mathbb{F}_p[x]/\langle\varphi(x)^j\rangle$ (Appendix B).

Note that y_0, y_1 are in the ring $\mathbb{F}_p[x]/\langle\varphi^{4a}\rangle$ and so they can be decomposed as $y_0 =: y_{0,0} + \varphi y_{0,1} + \ldots + \varphi^{4a-1}y_{0,4a-1}$ and $y_1 =: y_{1,0} + \varphi y_{1,1} + \ldots + \varphi^{4a-1}y_{1,4a-1}$, with all $y_{i,j}$'s in the field $\mathbb{F}_p[x]/\langle\varphi\rangle$.

To get $E'(y_0, y_1) \bmod \langle p, \varphi^{4a}\rangle$ the idea is: to first divide by p^2, and then to go modulo the ideal $\langle p, \varphi^{4a}\rangle$. Apply Algorithm ROOT-FIND to solve $E(y_0 + py_1)/p^2 \equiv 0 \bmod \langle p, \varphi^{4a}\rangle$. This allows us to fix some part of y_0, say $a_0 \in \mathbb{F}_p/\langle\varphi^{4a}\rangle$, and we can replace it by $a_0 + \varphi^{i_0}y_0$, $i_0 \geq 1$. Thus, $p^3 | E(a_0 + \varphi^{i_0}y_0 + py_1) \bmod \langle p^4, \varphi^{4a}\rangle$ and we divide out by this p^3 (and change the modulus to $\langle p, \varphi^{4a}\rangle$). In Lemma 11 we show that when we go modulo the ideal $\langle p, \varphi^{4a}\rangle$ (to find a_0), we only need to solve a univariate in y_0 using ROOT-FIND. So we only need to fix some part of y_0, that we called a_0, and y_1 is irrelevant. Finally, we get $E'(y_0, y_1)$ such that $E'(y_0, y_1) := E(a_0 + \varphi^{i_0}y_0 + py_1)/p^3 \bmod \langle p, \varphi^{4a}\rangle$. Importantly, the process yields at most *two* possibilities of E' (resp. a_0) to deal with.

Lemma 11 also shows that the bivariate $E'(y_0, y_1)$ is a special one of the form $E'(y_0, y_1) \equiv E_1(y_0) + E_2(y_0)y_1 \bmod \langle p, \varphi^{4a}\rangle$, where $E_1(y_0) \in (\mathbb{F}_p[x]/\langle\varphi^{4a}\rangle)[y_0]$ is a cubic univariate polynomial and $E_2(y_0) \in (\mathbb{F}_p[x]/\langle\varphi^{4a}\rangle)[y_0]$ is a linear univariate polynomial. We exploit this special structure to represent y_1 as a rational function of y_0, i.e., $y_1 \equiv -E_1(y_0)/E_2(y_0) \bmod \langle p, \varphi^{4a}\rangle$. The important issue is that we can calculate y_1 only when, on some specialization $y_0 = a_0$, the division by $E_2(a_0)$ is well defined. So we guess each value of $0 \leq r \leq 4a$ and ensure that the valuation (with respect to the powers of φ) of $E_1(y_0)$ is at least r but that of $E_2(y_0)$ is *exactly* r. Once we find such a y_0, we can efficiently compute y_1 as $y_1 \equiv -(E_1(y_0)/\varphi^r)/(E_2(y_0)/\varphi^r) \bmod \langle p, \varphi^{4a-r}\rangle$.

To find y_0, we find common solution of two equations: $E_1(y_0) \equiv E_2(y_0) \equiv 0 \bmod \langle p, \varphi^r\rangle$, for each guessed value r, using Algorithm ROOT-FIND. Since the polynomial $E_2(y_0)$ is linear, it is easy for us to filter all y_0's for which valuation of $E_2(y_0)$ is *exactly* r (Lemma 13). Thus, we could efficiently find all (y_0, y_1) pairs that satisfy the equation $E'(y_0, y_1) \equiv 0 \bmod \langle p, \varphi^{4a}\rangle$.

Proof idea of Theorem 2: If $f \equiv \varphi^e \bmod p$ then any lift $g(x)$ of a factor $\tilde{g}(x) \equiv \varphi^a \bmod p$ of $f \bmod p$ will be of the form $g \equiv (\varphi^a - py) \bmod p^k$. So basically we want to find all the y's mod p^{k-1} that appear in the proof idea of Theorem 1 above. This can be done easily, because Algorithm ROOT-FIND (Appendix B) [BLQ13] describes all possible y_0's in a compact data structure. Moreover, using this, a count of all y's can be provided as well.

2 Preliminaries

Let $R(+, .)$ be a ring and S be a non-empty subset of R. The product of the set S with a scalar $a \in R$ is defined as $aS := \{as \mid s \in S\}$. Similarly, the sum of a scalar $u \in R$ with the set S is defined as $u + S := \{u + s \mid s \in S\}$. Note that the product and the sum operations used inside the set are borrowed from the underlying ring R. Also note that if S is the empty set then so are aS and $u + S$ for any $a, u \in R$.

Representatives. The symbol '*' in a ring R, wherever appears, denotes all of ring R. For example, suppose $R = \mathbb{Z}/\langle p^k \rangle$ for a prime p and a positive integer k. In this ring, we will use the notation $y = y_0 + py_1 + \ldots + p^i y_i + p^{i+1}*$, where $i+1 < k$ and each $y_j \in R/\langle p \rangle$, to denote a set $S_y \subseteq R$ such that

$$S_y = \{y_0 + \ldots + p^i y_i + p^{i+1} y_{i+1} + \ldots + p^{k-1} y_{k-1} \mid \forall y_{i+1}, \ldots, y_{k-1} \in R/\langle p \rangle\}.$$

Notice that the number of elements in R represented by y is $|S_y| = p^{k-i-1}$.

We will sometimes write the set $y = y_0 + py_1 + \ldots + p^i y_i + p^{i+1}*$ succinctly as $y = v + p^{i+1}*$, where $v \in R$ stands for $v = y_0 + py_1 + \ldots + p^i y_i$.

In the following sections, we will add and multiply the set $\{*\}$ with scalars from the ring R. Let us define these operations as follows ($*$ is treated as an unknown)

- $u + \{*\} := \{u + *\}$ and $u\{*\} := \{u*\}$, where $u \in R$.
- $c + \{a + b*\} = \{(a+c) + b*\}$ and $c\{a + b*\} = \{ac + bc*\}$, where $a, b, c \in R$.

Another important example of the $*$ notation: Let $R = \mathbb{F}_p[x]/\langle \varphi(x)^k \rangle$ for a prime p and an irreducible φ mod p. In this ring, we use the notation $y = y_0 + \varphi y_1 + \ldots + \varphi^i y_i + \varphi^{i+1}*$, where $i+1 < k$ and each $y_j \in R/\langle \varphi \rangle$, to denote a set $S_y \subseteq R$ such that

$$S_y = \{y_0 + \ldots + \varphi^i y_i + \varphi^{i+1} y_{i+1} + \ldots + \varphi^{k-1} y_{k-1} \mid \forall y_{i+1}, \ldots, y_{k-1} \in R/\langle \varphi \rangle\}.$$

Zero-Divisors. Let $R[x]$ be the ring of polynomials over $R = \mathbb{Z}/\langle p^k \rangle$. The following lemma about zero divisors in $R[x]$ will be helpful.

Lemma 3. *A polynomial $f \in R[x]$ is a zero divisor iff $f \equiv 0 \bmod p$. Consequently, for any polynomials $f, g_1, g_2 \in R[x]$ and $f \not\equiv 0 \bmod p$, $f(x)g_1(x) = f(x)g_2(x)$ implies $g_1(x) = g_2(x)$.*

Proof. If $f \equiv 0 \bmod p$ then $f(x)p^{k-1}$ is zero, and f is a zero divisor.

For the other direction, let $f \not\equiv 0 \bmod p$ and assume $f(x)g(x) = 0$ for some non-zero $g \in R[x]$. Let

- i be the biggest integer such that the coefficient of x^i in f is non-zero modulo p,
- and j be the biggest integer such that the coefficient of x^j in g has minimum valuation with respect to p.

Then, the coefficient of x^{i+j} in $f \cdot g$ has same valuation as the coefficient of x^j in g, implying that the coefficient is nonzero. This contradicts the assumption $f(x)g(x) = 0$.

The consequence follows because $f \not\equiv 0 \bmod p$ implies that f cannot be a zero divisor. □

Quotient ideals. We define the quotient ideal (analogous to division of integers) and look at some of its properties.

Definition 4 (Quotient Ideal)**.** *Given two ideals I and J of a commutative ring R, we define the quotient of I by J as,*

$$I : J := \{a \in R \mid aJ \subseteq I\}.$$

It can be easily verified that $I : J$ is an ideal. Moreover, we can make the following observations about quotient ideals.

Claim 5 (Cancellation)**.** *Suppose I is an ideal of ring R and a, b, c are three elements in R. By definition of quotient ideals, $ca \equiv cb \bmod I$ iff $a \equiv b \bmod I : \langle c \rangle$.*

Claim 6. *Let p be a prime and $\varphi \in (\mathbb{Z}/\langle p^k \rangle)[x]$ be such that $\varphi \not\equiv 0 \bmod p$. Given an ideal $I := \langle p^l, \phi^m \rangle$ of $\mathbb{Z}[x]$,*

1. *$I : \langle p^i \rangle = \langle p^{l-i}, \phi^m \rangle$, for $i \leq l$, and*
2. *$I : \langle \phi^j \rangle = \langle p^l, \phi^{m-j} \rangle$, for $j \leq m$.*

Proof. We will only prove part (1), as proof of part (2) is similar. If $c \in \langle p^{l-i}, \varphi^m \rangle$ then there exists $c_1, c_2 \in \mathbb{Z}[x]$, such that, $c = c_1 p^{l-i} + c_2 \varphi^m$. Multiplying by p^i,

$$p^i c = c_1 p^l + c_2 p^i \varphi^m \in I \Rightarrow c \in I : \langle p^i \rangle.$$

To prove the reverse direction, if $c \in I : \langle p^i \rangle$ then there exists $c_1, c_2 \in \mathbb{Z}[x]$, such that, $p^i c = c_1 p^l + c_2 \varphi^m$. Since $i \leq l$ and $p \not| \varphi$, we know $p^i | c_2$. So,
$c = c_1 p^{l-i} + (c_2/p^i)\varphi^m \ \Rightarrow \ c \in \langle p^{l-i}, \phi^m \rangle$. □

Lemma 7 (Compute quotient)**.** *Given a polynomial $\varphi \in \mathbb{Z}[x]$ not divisible by p, define I to be the ideal $\langle p^l, \phi^m \rangle$ of $\mathbb{Z}[x]$. If $g(y) \in (\mathbb{Z}[x])[y]$ is a polynomial such that $g(y) \equiv 0 \bmod \langle p, \phi^m \rangle$, then $p | g(y) \bmod I$ and $g(y)/p \bmod I : \langle p \rangle$ is efficiently computable.*

Proof. The equation $g(y) \equiv 0 \bmod \langle p, \phi^m \rangle$ implies $g(y) = pc_1(y) + \varphi^m c_2(y)$ for some polynomials $c_1(y), c_2(y) \in \mathbb{Z}[x][y]$. Going modulo I, $g(y) \equiv pc_1(y) \bmod I$. Hence, $p | g(y) \bmod I$ and $g(y)/p \equiv c_1(y) \bmod I : \langle p \rangle$ (Claim 5).

If we write g in the reduced form modulo I, then the polynomial $g(y)/p$ can be obtained by dividing each coefficient of $g(y) \bmod I$ by p. □

3 Main Results: Proof of Theorems 1 and 2

Our task is to factor a univariate integral polynomial $f(x) \in \mathbb{Z}[x]$ of degree d modulo a prime power p^k. Without loss of generality, we can assume that $f(x) \not\equiv 0 \bmod p$. Otherwise, we can efficiently divide $f(x)$ by the highest power of p possible, say p^l, such that $f(x) \equiv p^l \tilde{f}(x) \bmod p^k$ and $\tilde{f}(x) \not\equiv 0 \bmod p$. In this case, it is equivalent to factor $\tilde{f}$ instead of f.

To simplify the input further, write $f \bmod p$ (uniquely) as a product of powers of coprime irreducible polynomials. If there are two coprime factors of f, using Hensel lemma (Lemma 16), we get a non-trivial factorization of $f \bmod p^k$. So we can assume that f is a power of a monic irreducible polynomial $\varphi \in \mathbb{Z}[x] \bmod p$. In other words, we can efficiently write $f \equiv \varphi^e + pl \bmod p^k$ for a polynomial l in $(\mathbb{Z}/\langle p^k \rangle)[x]$. We have $e \cdot \deg \varphi \leq \deg f$, for the integral polynomials f and φ.

3.1 Factoring to Root-finding

By the preprocessing above, we only need to find factors of a polynomial f such that $f \equiv \varphi^e + pl \bmod p^k$, where φ is an irreducible polynomial modulo p. Up to multiplication by units, any nontrivial factor h of f has the form $h \equiv \varphi^a - py$, where $a < e$ and y is a polynomial in $(\mathbb{Z}/\langle p^k\rangle)[x]$.

Let us denote the ring $\mathbb{Z}[x]/\langle p^k, \varphi^{ak}\rangle$ by R. Also, denote the ring $\mathbb{Z}[x]/\langle p, \varphi^{ak}\rangle$ by R_0. We define an auxiliary polynomial $E(y) \in R[y]$ as

$$E(y) := f(x)(\varphi^{a(k-1)} + \varphi^{a(k-2)}(py) + \ldots + \varphi^a(py)^{k-2} + (py)^{k-1}).$$

Our first step is to reduce the problem of factoring $f(x) \bmod p^k$ to the problem of finding roots of the univariate polynomial $E(y)$ in R. Thus, we convert the problem of finding factors of $f(x) \in \mathbb{Z}[x]$ modulo a principal ideal $\langle p^k\rangle$ to root finding of a polynomial $E(y) \in (\mathbb{Z}[x])[y]$ modulo a bi-generated ideal $\langle p^k, \varphi^{ak}\rangle$.

Theorem 8 (Reduction theorem)**.** *Given a prime power p^k; let $f(x), h(x) \in \mathbb{Z}[x]$ be two polynomials of the form $f(x) \equiv \varphi^e + pl \bmod p^k$ and $h(x) \equiv \varphi^a - py \bmod p^k$. Here y, l are elements of $(\mathbb{Z}/\langle p^k\rangle)[x]$ and $a \le e$. Then, h divides f modulo p^k if and only if*

$$E(y) \;=\; f(x)(\varphi^{a(k-1)} + \varphi^{a(k-2)}(py) + \ldots + \varphi^a(py)^{k-2} + (py)^{k-1}) \;\equiv\; 0 \bmod \langle p^k, \varphi^{ak}\rangle.$$

Proof. Let Q denote the *ring of fractions* of the ring $(\mathbb{Z}/\langle p^k\rangle)[x]$. Since φ is not a zero divisor, $(E(y)/\varphi^{ak}) \in Q$.

We first prove the reverse direction. If $E(y) \equiv 0 \bmod \langle p^k, \varphi^{ak}\rangle$, then $(E(y)/\varphi^{ak})$ is a polynomial over $(\mathbb{Z}/\langle p^k\rangle)[x]$. Multiplying h with $(E(y)/\varphi^{ak}) \bmod p^k$, we write,

$$(\varphi^a - py)((f/\varphi^{ak})\Sigma_{i=0}^{k-1}\varphi^{a(k-1-i)}(py)^i) \;\equiv\; (f/\varphi^{ak})(\varphi^{ak} - (py)^k) \equiv f \cdot \varphi^{ak}/\varphi^{ak} \equiv f \bmod p^k.$$

Hence, h divides f modulo p^k.

For the forward direction, assume that there exists some $g(x) \in (\mathbb{Z}/\langle p^k\rangle)[x]$, such that, $f(x) \equiv h(x)g(x) \bmod p^k$. We get two factorizations of f in Q,

$$f(x) = h(x)g(x) \quad \text{and} \quad f(x) = h(x)(E(y)/\varphi^{ak}).$$

Subtracting the first equation from the second one,

$$h(x)\left(g(x) - (E(y)/\varphi^{ak})\right) = 0.$$

Notice that $h(x)$ is not a zero divisor in $(\mathbb{Z}/\langle p^k\rangle)[x]$ (by Lemma 3) and is thus invertible in Q. So, $E(y)/\varphi^{ak} = g(x)$ in Q. Since $g(x)$ is in $(\mathbb{Z}/\langle p^k\rangle)[x]$, we deduce the equivalent divisibility statement: $E(y) \equiv 0 \bmod \langle p^k, \varphi^{ak}\rangle$. □

The following two observations simplify our task of finding roots y of polynomial $E(y)$.

- First, due to symmetry, it is enough to find factors $h \equiv \varphi^a \bmod p$ with $a \le e/2$. The assertion follows because $f \equiv hg \bmod p^k$ implies, at least one of the factor (say h) must be of the form $\varphi^a \bmod p$ for $a \le e/2$. By Lemma 3, for a fixed $h \equiv (\varphi^a - py) \bmod p^k$, there is a unique $g \equiv (\varphi^{e-a} - py') \bmod p^k$ such that $f \equiv hg \bmod p^k$. So, to find g, it is enough to find h.

- Second, observe that any root $y \in R$ (of $E(y) \in R[y]$) can be seen as $y = y_0 + py_1 + p^2 y_2 + \ldots + p^{k-1} y_{k-1}$, where each $y_i \in R_0$ for all i in $\{0, \ldots, k-1\}$. The following lemma decreases the required precision of root y.

Lemma 9. *Let $y = y_0 + py_1 + p^2 y_2 + \ldots + p^{k-1} y_{k-1}$ be a root of $E(y)$, where $k \ge 2$ and $a \le e/2$. Then, all elements of set $y = y_0 + py_1 + p^2 y_2 + \ldots + p^{k-3} y_{k-3} + p^{k-2} *$ are also roots of $E(y)$.*

Proof. Notice that the variable y is multiplied with p in $E(y)$, implying y_{k-1} is irrelevant. Similar argument is applicable for the variable y_{k-2} in any term of the form $(py)^i$ for $i \ge 2$. The only remaining term containing y_{k-2} is $f\varphi^{a(k-2)}(py)$. The coefficient of y_{k-2} in this term is $\varphi^{a(k-2)} f p^{k-1}$. This coefficient vanishes modulo $\langle p^k, \varphi^{ak} \rangle$ too, because

$$\varphi^{a(k-2)} f \;\equiv\; \varphi^{a(k-2)} \varphi^e \;\equiv\; \varphi^{ak} \varphi^{e-2a} \;\equiv\; 0 \bmod \langle p, \varphi^{ak} \rangle \; . \qquad \square$$

Root-finding modulo a principal ideal. Finally, we state a slightly modified version of the theorem from [BLQ13, Cor.4], showing that all the roots of a polynomial $g(y) \in R_0[y]$ can be efficiently described. They gave their algorithm to find (all) roots in $\mathbb{Z}/\langle p^n \rangle$; we modify it in a straightforward way to find (all) roots in $\mathbb{F}_p[x]/\langle \varphi^{ak} \rangle = R_0$ (Appendix B). Any root in R_0 can be written as $y = y_0 + \varphi y_1 + \cdots + \varphi^{ak-1} y_{ak-1}$, where each y_j is in the field $R_0/\langle \varphi \rangle$.

Let $g(y)$ be a polynomial in $R[y]$, then a set $y = y_0 + \varphi y_1 + \ldots + \varphi^i y_i + \varphi^{i+1} *$ will be called a *representative root* of g iff

- All elements in $y = y_0 + \varphi y_1 + \ldots + \varphi^i y_i + \varphi^{i+1} *$ are roots of g.

- Not all elements in $y' = y_0 + \varphi y_1 + \ldots + \varphi^{i-1} y_{i-1} + \varphi^i *$ are roots of g.

We will sometimes represent the set of roots, $y = y_0 + \varphi y_1 + \ldots + \varphi^i y_i + \varphi^{i+1} *$, succinctly as $y = v + \varphi^{i+1} *$, where $v \in R$ stands for $y = y_0 + \varphi y_1 + \ldots + \varphi^i y_i$. Such a pair, $(v, i+1)$, will be called a *representative pair*.

Theorem 10. *[BLQ13, Cor.4] Given a bivariate $g(y) \in R_0[y]$ where $R_0 = \mathbb{Z}[x]/\langle p, \varphi^{ak} \rangle$, let $Z \subseteq R_0$ be the root set of $g(y)$. Then Z can be expressed as the disjoint union of at most $\deg_y(g)$ many representative pairs (a_0, i_0) ($a_0 \in R_0$ and $i_0 \in \mathbb{N}$).*

These representative pairs can be found in randomized poly($\deg_y(g), \log p, ak \deg \varphi$) time.

For completeness, Algorithm Root-Find(g, R_0) is given in Appendix B.

We will fix $k = 4$ for the rest of this section. Similar techniques (even simpler) work for $k = 3$ and $k = 2$. The barriers for $k > 4$ will be discussed in Appendix D.

3.2 Reduction to root-finding modulo a principal ideal of $\mathbb{F}_p[x]$

In this subsection, the task to find roots of $E(y)$ modulo the bi-generated ideal $\langle p^4, \varphi^{4a}\rangle$ of $\mathbb{Z}[x]$ will be reduced to finding roots modulo the principal ideal $\langle \varphi^{4a}\rangle$ (of $\mathbb{F}_p[x]$).

Let us consider the equation $E(y) \equiv 0 \bmod \langle p^4, \varphi^{4a}\rangle$. We have,

$$f(\varphi^{3a} + \varphi^{2a}(py) + \varphi^a(py)^2 + (py)^3) \equiv 0 \bmod \langle p^4, \varphi^{4a}\rangle\,. \tag{1}$$

Using Lemma 9, we can assume $y = y_0 + py_1$,

$$f(\varphi^{3a} + \varphi^{2a}p(y_0 + py_1) + \varphi^a p^2(y_0^2 + 2py_0y_1) + (py_0)^3) \equiv 0 \bmod \langle p^4, \varphi^{4a}\rangle\,. \tag{2}$$

The idea is to first solve this equation modulo $\langle p^3, \varphi^{4a}\rangle$. Since $f \equiv \varphi^e \bmod p$, $e \geq 2a$, variable y_1 is redundant while solving this equation modulo p^3. The following lemma finds all representative pairs (a_0, i_0) for y_0, such that, $E(a_0 + \varphi^{i_0}y_0 + py_1) \equiv 0 \bmod \langle p^3, \varphi^{4a}\rangle$ for all $y_0, y_1 \in R$. Alternatively, we can state this in the polynomial ring $R[y_0, y_1]$. Dividing by p^3, we will be left with an equation modulo the principal ideal $\langle \varphi^{4a}\rangle$ (of $\mathbb{F}_p[x]$).

Lemma 11 (Reduce to char=p)**.** *We efficiently compute a unique set S_0 of all representative pairs (a_0, i_0), where $a_0 \in R_0$ and $i_0 \in \mathbb{N}$, such that,*

$$E((a_0 + \varphi^{i_0}y_0) + py_1) \;=\; p^3E'(y_0, y_1) \;\bmod \langle p^4, \varphi^{4a}\rangle$$

for a polynomial $E'(y_0, y_1) \in R_0[y_0, y_1]$ (it depends on (a_0, i_0)). Moreover,

1. *$|S_0| \leq 2$. If our algorithm fails to find E' then Eqn. 2 has no solution.*
2. *$E'(y_0, y_1) =: E_1(y_0) + E_2(y_0)y_1$, where $E_1(y_0) \in R_0[y_0]$ is cubic in y_0 and $E_2(y_0) \in R_0[y_0]$ is linear in y_0.*
3. *For every root $y \in R$ of $E(y)$ there exists $(a_0, i_0) \in S_0$ and $(a_1, a_2) \in R \times R$, such that $y = (a_0 + \varphi^{i_0}a_1) + pa_2$ and $E'(a_1, a_2) \equiv 0 \bmod \langle p, \varphi^{4a}\rangle$.*

We think of E' as the quotient $E((a_0 + \varphi^{i_0}y_0) + py_1)/p^3$ in the polynomial ring $R_0[y_0, y_1]$; and would work with it instead of E in the root-finding algorithm.

Proof. Looking at Eqn. 2 modulo p^2,

$$f\varphi^{2a}(\varphi^a + py_0) \;\equiv\; 0 \;\bmod \langle p^2, \varphi^{4a}\rangle.$$

Substituting $f = \varphi^e + ph_1$, we get $(\varphi^e + ph_1)(\varphi^{3a} + \varphi^{2a}py_0) \;\equiv\; 0 \bmod \langle p^2, \varphi^{4a}\rangle$. Implying, $ph_1\varphi^{3a} \;\equiv\; 0 \bmod \langle p^2, \varphi^{4a}\rangle$. Using Claim 6 the above equation implies that,

$$h_1 \;\equiv\; 0 \;\bmod \langle p, \varphi^a\rangle\,, \tag{3}$$

is a necessary condition for y_0 to exist.

We again look at Eqn. 2, but modulo p^3 now: $f(\varphi^{3a} + \varphi^{2a}py_0 + \varphi^a p^2 y_0^2) \;\equiv\; 0 \;\bmod \langle p^3, \varphi^{4a}\rangle$.

Notice that y_1 is not present because its coefficient: $p^2 f \varphi^{2a} \equiv 0 \bmod \langle p^3, \varphi^{4a} \rangle$. Substituting $f = \varphi^e + ph_1$, we get,

$$(\varphi^e + ph_1)(\varphi^{3a} + \varphi^{2a} p y_0 + \varphi^a p^2 y_o^2) \equiv 0 \bmod \langle p^3, \varphi^{4a} \rangle.$$

Removing the coefficients of y_0 which vanish modulo $\langle p^3, \varphi^{4a} \rangle$,

$$\varphi^{e+a} p^2 y_0^2 + \varphi^{3a} p h_1 + \varphi^{2a} p^2 h_1 y_0 \equiv 0 \mod \langle p^3, \varphi^{4a} \rangle.$$

From Eqn. 3, h_1 can be written as $ph_{1,1} + \varphi^a h_{1,2}$, so

$$p^2 \left(\varphi^{e+a} y_0^2 + \varphi^{3a} h_{1,2} y_0 + \varphi^{3a} h_{1,1} \right) \equiv 0 \mod \langle p^3, \varphi^{4a} \rangle.$$

We can divide by $p^2 \varphi^{3a}$ using Claim 6 to get an equation modulo φ^a in the ring $\mathbb{F}_p[x]$. This is a quadratic equation in y_0. Using Theorem 10, we find the solution set S_0 with at most two representative pairs: for $(a_0, i_0) \in S_0$, every $y \in a_0 + \varphi^{i_0} * + p*$ satisfies,

$$E(y) \equiv 0 \mod \langle p^3, \varphi^{4a} \rangle .$$

In other words, on substituting $(a_0 + \varphi^{i_0} y_0 + p y_1)$ in $E(y)$,

$$E(a_0 + \varphi^{i_0} y_0 + p y_1) \equiv p^3 E'(y_0, y_1) \mod \langle p^4, \varphi^{4a} \rangle,$$

for a "bivariate" polynomial $E'(y_0, y_1) \in R_0[y_0, y_1]$. This sets up the correspondence between the roots of E and E'.

Substituting $(a_0 + \varphi^{i_0} y_0 + p y_1)$ in Eqn. 2, we notice that $E'(y_0, y_1)$ has the form $E_1(y_0) + E_2(y_0) y_1$ for a linear E_2 and a cubic E_1.

Finally, this reduction is constructive, because of Lemma 7 and Theorem 10, giving a randomized poly-time algorithm. □

3.3 Finding roots of a *special* bi-variate $E'(y_0, y_1)$ modulo $\langle p, \varphi^{4a} \rangle$

The final obstacle is to find roots of $E'(y_0, y_1)$ modulo $\langle \varphi^{4a} \rangle$ in $\mathbb{F}_p[x]$. The polynomial $E'(y_0, y_1) = E_1(y_0) + E_2(y_0) y_1$ is *special* because $E_2 \in R_0[y_0]$ is *linear* in y_0.

For a polynomial $u \in \mathbb{F}_p[x][\mathbf{y}]$ we define *valuation* $\mathrm{val}_\varphi(u)$ to be the largest r such that $\varphi^r | u$. Our strategy is to go over all possible valuations $0 \le r \le 4a$ and find y_0, such that,

- $E_1(y_0)$ has valuation at least r.
- $E_2(y_0)$ has valuation exactly r.

From these y_0's, y_1 can be obtained by 'dividing' $E_1(y_0)$ with $E_2(y_0)$. The lemma below shows that this strategy captures all the solutions.

Lemma 12 (Bivariate solution). *A pair $(u_0, u_1) \in R_0 \times R_0$ satisfies an equation of the form $E_1(y_0) + E_2(y_0) y_1 \equiv 0 \bmod \langle p, \varphi^{4a} \rangle$ if and only if $val_\varphi(E_1(u_0)) \ge val_\varphi(E_2(u_0))$.*

Proof. Let r be $\mathrm{val}_\varphi(E_2(u_0))$, where r is in the set $\{0,1,\ldots,4a\}$. If $\mathrm{val}_\varphi(E_1(u_0)) \geq \mathrm{val}_\varphi(E_2(u_0))$ then set $u_1 \equiv -(E_1(u_0)/\varphi^r)/(E_2(u_0)/\varphi^r) \bmod \langle p, \varphi^{4a-r}\rangle$. The pair (u_0, u_1) satisfies the required equation. (Note: If $r = 4a$ then we take $u_1 = *$.)

Conversely, if $r' := \mathrm{val}_\varphi(E_1(u_0)) < \mathrm{val}_\varphi(E_2(u_0)) \leq 4a$ then, for every u_1,
$\mathrm{val}_\varphi(E_1(u_0) + E_2(u_0)u_1) = r' \ \Rightarrow\ E_1(u_0) + E_2(u_0)u_1 \not\equiv 0 \bmod \langle p, \varphi^{4a}\rangle$. □

We can efficiently find all representative pairs for y_0, at most three, such that $E_1(y_0)$ has valuation at least r (using Theorem 10). The next lemma shows that we can efficiently filter all y_0's, from these representative pairs, that give valuation *exactly* r for $E_2(y_0)$.

Lemma 13 (Reduce to a unit E_2). *Given a linear polynomial $E_2(y_0) \in R_0[y_0]$ and an $r \in [4a-1]$, let (b, i) be a representative pair modulo $\langle p, \varphi^r\rangle$, i.e., $E_2(b+\varphi^i *) \equiv 0 \bmod \langle p, \varphi^r\rangle$. Consider the quotient $E_2'(y_0) := E_2(b + \varphi^i y_0)/\varphi^r$.*

If $E_2'(y_0)$ does not vanish identically modulo $\langle p, \varphi\rangle$, then there exists at most one $\theta \in R_0/\langle\varphi\rangle$ such that $E_2'(\theta) \equiv 0 \bmod \langle p, \varphi\rangle$, and this θ can be efficiently computed.

Proof. Suppose $E_2(b+\varphi^i y_0) \equiv u + v y_0 \equiv 0 \bmod \langle p, \varphi^r\rangle$. Since y_0 is formal, we get $\mathrm{val}_\varphi(u) \geq r$ and $\mathrm{val}_\varphi(v) \geq r$. We consider the three cases (wrt these valuations),

1. $\mathrm{val}_\varphi(u) \geq r$ and $\mathrm{val}_\varphi(v) = r$: $E_2'(\theta) \not\equiv 0 \bmod \langle p, \varphi\rangle$, for all $\theta \in R_0/\langle\varphi\rangle$ except $\theta = (-u/\varphi^r)/(v/\varphi^r) \bmod \langle p, \varphi\rangle$.
2. $\mathrm{val}_\varphi(u) = r$ and $\mathrm{val}_\varphi(v) > r$: $E_2'(\theta) \not\equiv 0 \bmod \langle p, \varphi\rangle$, for all $\theta \in R_0/\langle\varphi\rangle$.
3. $\mathrm{val}_\varphi(u) > r$ and $\mathrm{val}_\varphi(v) > r$: $E_2'(y_0)$ vanishes identically modulo $\langle p, \varphi\rangle$, so this case is ruled out by the hypothesis.

There is an efficient algorithm to find θ, if it exists; because the above proof only requires calculating valuations which entails division operations in the ring. □

3.4 Algorithm to find roots of $E(y)$

We have all the ingredients to give the algorithm for finding roots of $E(y)$ modulo ideal $\langle p^4, \varphi^{4a}\rangle$ of $\mathbb{Z}[x]$.

Input: A polynomial $E(y) \in R[y]$ defined as $E(y) := f(x)(\varphi^{3a} + \varphi^{2a}(py) + \varphi^a(py)^2 + (py)^3)$.

Output: A set $Z \subseteq R_0$ and a *bad* set $Z' \subseteq R_0$, such that, for each $y_0 \in Z - Z'$, there are (efficiently computable) $y_1 \in R_0$ (Theorem 14) satisfying $E(y_0 + py_1) \equiv 0 \bmod \langle p^4, \varphi^{4a}\rangle$. These are exactly the roots of E.

Also, both sets Z and Z' can be described by $O(a)$ many representatives (Theorem 14). Hence, a $y_0 \in Z - Z'$ can be picked efficiently.

Algorithm 1 Finding all roots of $E(y)$ in R

1: Given $E(y_0 + py_1)$, using Lemma 11, get the set S_0 of all representative pairs (a_0, i_0), where $a_0 \in R_0$ and $i_0 \in \mathbb{N}$, such that $p^3 | E((a_0 + \varphi^{i_0} y_0) + py_1) \bmod \langle p^4, \varphi^{4a}\rangle$.
2: Initialize sets $Z = \{\}$ and $Z' = \{\}$; seen as subsets of R_0.

3: **for** each $(a_0, i_0) \in S_0$ **do**

4: Substitute $y_0 \mapsto a_0 + \varphi^{i_0} y_0$, let $E'(y_0, y_1) = E_1(y_0) + E_2(y_0)y_1 \bmod \langle p, \varphi^{4a} \rangle$ be the polynomial obtained from Lemma 11.

5: **If** $E_2(y_0) \not\equiv 0 \bmod \langle p, \varphi \rangle$ **then** find (at most one) $\theta \in R_0/\langle \varphi \rangle$ such that $E_2(\theta) \equiv 0 \bmod \langle p, \varphi \rangle$. Update $Z \leftarrow Z \cup (a_0 + \varphi^{i_0} *)$ and $Z' \leftarrow Z' \cup (a_0 + \varphi^{i_0}(\theta + \varphi *))$.

6: **for** each possible valuation $r \in [4a]$ **do**

7: Initialize sets $Z_r = \{\}$ and $Z_r^{'} = \{\}$.

8: Call Root-Find(E_1, φ^r) to get a set S_1 of representative pairs (a_1, i_1) where $a_1 \in R_0$ and $i_1 \in \mathbb{N}$ such that $E_1(a_1 + \varphi^{i_1} y_0) \equiv 0 \bmod \langle p, \varphi^r \rangle$.

9: **for** each $(a_1, i_1) \in S_1$ **do**

10: Analogously consider $E_2'(y_0) := E_2(a_1 + \varphi^{i_1} y_0) \bmod \langle p, \varphi^{4a} \rangle$.

11: Call Root-Find(E_2', φ^r) to get a representative pair (a_2, i_2) ($\because E_2'$ is linear), where $a_2 \in R_0$ and $i_2 \in \mathbb{N}$ such that $E_2'(a_2 + \varphi^{i_2} y_0) \equiv 0 \bmod \langle p, \varphi^r \rangle$.

12: **if** $r = 4a$ **then**

13: Update $Z_r \leftarrow Z_r \cup (a_1 + \varphi^{i_1}(a_2 + \varphi^{i_2} *))$ and $Z_r^{'} \leftarrow Z_r^{'} \cup \{\}$.

14: **else if** $E_2'(a_2 + \varphi^{i_2} y_0) \not\equiv 0 \bmod \langle p, \varphi^{r+1} \rangle$ **then**

15: Get a $\theta \in R_0/\langle \varphi \rangle$ (Lemma 13), if it exists, such that $E_2'(a_2 + \varphi^{i_2}(\theta + \varphi y_0)) \equiv 0 \bmod \langle p, \varphi^{r+1} \rangle$. Update $Z_r' \leftarrow Z_r' \cup (a_1 + \varphi^{i_1}(a_2 + \varphi^{i_2}(\theta + \varphi *)))$.

16: Update $Z_r \leftarrow Z_r \cup (a_1 + \varphi^{i_1}(a_2 + \varphi^{i_2} *))$.

17: **end if**

18: **end for**

19: Update $Z \leftarrow Z \cup (a_0 + \varphi^{i_0} Z_r)$ and $Z' \leftarrow Z' \cup (a_0 + \varphi^{i_0} Z_r^{'})$.

20: **end for**

21: **end for**

22: Return Z and Z'.

We prove the correctness of Algorithm 1 in the following theorem.

Theorem 14. *The output of Algorithm 1 (set $Z - Z'$) contains exactly those $y_0 \in R_0$ for which there exist some $y_1 \in R_0$, such that, $y = y_0 + py_1$ is a root of $E(y)$ in R. We can easily compute the set of y_1 corresponding to a given $y_0 \in Z - Z'$ in poly($\deg f, \log p$) time.*

Thus, we efficiently describe (and exactly count) the roots $y = y_0 + py_1 + p^2 y_2$ in R of $E(y)$, where $y_0, y_1 \in R_0$ are as above and y_2 can assume any value from R.

Proof. The algorithm intends to output roots y of equation $E(y) \equiv f(x)(\varphi^{3a} + \varphi^{2a}(py) + \varphi^a(py)^2 + (py)^3) \equiv 0 \bmod \langle p^4, \varphi^{4a} \rangle$, where $y = y_0 + py_1 + p^2 y_2$ with $y_0, y_1 \in R_0$ and $y_2 \in R$. From Lemma 9, y_2 can be kept as $*$, and is independent of y_0 and y_1.

Using Lemma 11, Algorithm 1 partially fixes y_0 from the set S_0 and reduces the problem to finding roots of an $E'(y_0, y_1) \bmod \langle p, \varphi^{4a} \rangle$. In other words, if we can find all roots (y_0, y_1) of $E'(y_0, y_1) \bmod \langle p, \varphi^{4a} \rangle$, then we can find (and count) all roots of $E(y) \bmod \langle p^4, \varphi^{4a} \rangle$. This is accomplished by Step 1. From Lemma 11, $|S_0| \leq 2$, so loop at Step 3 runs only for a constant number of times.

Using Lemma 11, $E'(y_0, y_1) \equiv E_1(y_0) + E_2(y_0)y_1 \bmod \langle p, \varphi^{4a} \rangle$ for a cubic polynomial $E_1(y_0) \in R_0[y_0]$ and a linear polynomial $E_2(y_0) \in R_0[y_0]$.

We find all solutions of $E'(y_0, y_1)$ by going over all possible valuations of $E_2(y_0)$ with respect to φ. The case of valuation 0 is handled in Step 5 and valuation $4a$ is handled in Step 12. For the remaining valuations $r \in [4a-1]$, Lemma 12 shows that it is enough to find $(z_0, z_1) \in R_0 \times R_0$ such that $\varphi^r | E_1(z_0)$ and $\varphi^r || E_2(z_0)$.

Notice that the number of valuations is bounded by $4a = O(\deg f)$. At Step 6, the algorithm guesses the valuation r of $E_2(y_0) \in R_0[y_0]$ and subsequent conputation finds all representative roots $b + \varphi^i *$ efficiently (using Theorem 10), such that,

$$E_1(b + \varphi^i y_0) \equiv E_2(b + \varphi^i y_0) \equiv 0 \bmod \langle p, \varphi^r \rangle .$$

The representative root $b + \varphi^i *$ is denoted by $a_1 + \varphi^{i_1}(a_2 + \varphi^{i_2} *)$ in Steps 13 and 16 of Algorithm 1.

Finally, we need to filter out those y_0's for which $E_2(b + \varphi^i y_0) \equiv 0 \bmod \langle p, \varphi^{r+1} \rangle$. This can be done efficiently using Lemma 13, where we get a unique $\theta \in R_0/\langle \varphi \rangle$ for which,

$$E_2(b + \varphi^i(\theta + \varphi y_0)) \equiv 0 \bmod \langle p, \varphi^{r+1} \rangle.$$

We store partial roots in two sets Z_r and Z'_r, where Z'_r contains the bad values filtered out by Lemma 13 as $b + \varphi^i(\theta + \varphi *)$ and Z_r contains all possible roots $b + \varphi^i *$. So, the set $Z_r - Z'_r$ contains exactly those elements z_0 for which there exists $z_1 \in R_0$, such that, the pair (z_0, z_1) is a root of $E'(y_0, y_1) \bmod \langle p, \varphi^{4a} \rangle$.

Note that size of each set S_1 obtained at Step 9 is bounded by three using Theorem 10 (E_1 is at most a cubic in y_0). Again using Theorem 10, we get at most one pair (a_2, i_2) at Step 11 for some $a_2 \in R_0$ and $i_2 \in \mathbb{N}$ (E'_2 is linear in y_0).

Now, for a fixed $z_0 \in Z_r - Z'_r$ we can calculate all z_1's by the equation

$$z_1 \equiv \tilde{z}_1 := -(C(y_0)/L(y_0)) \bmod \langle p, \varphi^{4a-r} \rangle.$$

Here $C(y_0) := E_1(z_0)/\varphi^r \bmod \langle p, \varphi^{4a-r} \rangle$ and $L(y_0) := E_2(z_0)/\varphi^r \bmod \langle p, \varphi^{4a-r} \rangle$. So, $z_1 \in R_0$ comes from the set $z_1 \in \tilde{z}_1 + \varphi^{4a-r} *$. This can be done efficiently in $\mathrm{poly}(\deg f, \log p)$ time.

Finally, sets $Z = a_0 + \varphi^{i_0} Z_r$ and $Z' = a_0 + \varphi^{i_0} Z'_r$, for $(a_0, i_0) \in S_0$ and corresponding valid $r \in \{0, \ldots, 4a-1\}$, returned by Algorithm 1, describe the y_0 for the roots of $E(y_0 + py_1) \bmod \langle p^4, \varphi^{4a} \rangle$. The number of representatives in each of these sets is $O(a)$, since $|S_0| \le 2$ and sizes of Z_r and Z'_r are only constant.

Since we can efficiently describe these y_0's and corresponding y_1's, and we know their precision, we can count all roots $y = y_0 + py_1 + p^2 * \subseteq R$ of $E(y) \bmod \langle p^4, \varphi^{4a} \rangle$. □

Remark 1 (Root finding for $k = 3$ and $k = 2$). *Algorithm 1 can as well be used when $k \in \{2, 3\}$ (the algorithm simplifies considerably).*

For $k = 3$, by Lemma 9, the only relevant coordinate is y_0. Moreover, we can directly call algorithm Root-Find *to find all roots of $E(y)/p^2$.*

For $k = 2$, using Lemma 9 again, we see that there are only two possibilities: $y_0 = *$, or there is no solution. This can be determined by testing whether $E(y)/p^2 \bmod \langle \varphi^{2a} \rangle$ exists.

3.5 Wrapping up Theorems 1 and 2

Proof of Theorem 1. We prove that given a general univariate $f(x) \in \mathbb{Z}[x]$ and a prime p, a non-trivial factor of $f(x)$ modulo p^4 can be obtained in randomized poly(deg $f, \log p$) time (or the irreducibility of $f(x)$ mod p^4 gets certified).

If $f(x) \equiv f_1(x)f_2(x) \bmod p$, where $f_1(x), f_2(x) \in \mathbb{F}_p[x]$ are two coprime polynomials, then we can efficiently lift this factorization to the ring $(\mathbb{Z}/\langle p^4\rangle)[x]$, using Hensel lemma (Lemma 16), to get non-trivial factors of $f(x)$ mod p^4.

For the remaining case, $f(x) \equiv \varphi^e \bmod p$ for an irreducible polynomial $\varphi(x)$ modulo p. The question of factoring f mod p^4 then reduces to root finding of a polynomial $E(y)$ mod $\langle p^4, \varphi^{4a}\rangle$ by Reduction theorem (Theorem 8). Using Theorem 14, we get all such roots and hence a non-trivial factor of $f(x)$ mod p^4 is found. If there are no roots $y \in R$ of $E(y)$, for all $a \leq e/2$, then the polynomial f is irreducible (by symmetry, if there is a factor for $a > e/2$ then there is a factor for $a \leq e/2$). □

We observe that our efficient algorithm (Appendix C) outputs *all* the factors of f mod p^3 in a compact way.

Proof of Theorem 2. We will prove the theorem for $k = 4$, case of $k < 4$ is similar.

We are given a univariate $f(x) \in \mathbb{Z}[x]$ of degree d and a prime p, such that, $f(x)$ mod p is a power of an irreducible polynomial $\varphi(x)$. So, $f(x)$ is of the form $\varphi(x)^e + ph(x) \bmod p^4$, for an integer $e \in \mathbb{N}$ and a polynomial $h(x) \in (\mathbb{Z}/\langle p^4\rangle)[x]$ of degree $\leq d$ (also, $\deg \varphi^e \leq d$). By unique factorization over the ring $\mathbb{F}_p[x]$, if $\tilde{g}(x)$ is a factor of $f(x)$ mod p then, $\tilde{g}(x) \equiv \tilde{v}\varphi(x)^a \bmod p$ for a unit $\tilde{v} \in \mathbb{F}_p$.

First, we show that it is enough to find all the lifts of $\tilde{g}(x)$, such that, $\tilde{g}(x) \equiv \varphi(x)^a \bmod p$ for an $a \leq e$. If $\tilde{g}(x) \equiv \tilde{v}\varphi(x)^a \bmod p$, then any lift has the form $g(x) \equiv v(x)(\varphi^a - py) \bmod p^4$ for a unit $v(x) \in (\tilde{v} + p*) \subseteq (\mathbb{Z}/\langle p^4\rangle)[x]$. Any such $g(x)$ maps uniquely to a $g_1(x) := \tilde{v}^{-1}g(x) \bmod p^4$, which is a lift of $\varphi(x)^a \bmod p$. So, it is enough to find all the lifts of $\varphi(x)^a \bmod p$.

We know that any lift $g(x) \in (\mathbb{Z}/\langle p^4\rangle)[x]$ of $\tilde{g}(x)$, which is a factor of $f(x)$, must be of the form $\varphi(x)^a - py(x) \bmod p^4$ for a polynomial $y(x) \in (\mathbb{Z}/\langle p^4\rangle)[x]$. By Reduction theorem (Theorem 8), we know that finding such a factor is equivalent to solving for y in the equation $E(y) \equiv 0 \bmod \langle p^4, \varphi^{4a}\rangle$. By Theorem 14, we can find all such roots y in randomized poly$(d, \log p)$ time, for $a \leq e/2$.

If $a > e/2$ then we replace a by $b := e - a$, as $b \leq e/2$, and solve the equation $E(y) \equiv 0 \bmod \langle p^4, \varphi^{4b}\rangle$ using Theorem 14. This time the factor corresponding to y will be, $g(x) \equiv f/(\varphi^b - py) \equiv E(y)/\varphi^{4b} \bmod p^4$, using Reduction theorem (Theorem 8).

The number of lifts of $\tilde{g}(x)$ which divide f mod p^4 is the count of y's that appear above. This is efficiently computable. □

4 Conclusion

The study of [vzGH98, vzGH96] sheds some light on the behaviour of the factoring problem for integral polynomials modulo prime powers. It shows that for "large" k the problem is similar to the factorization over p-adic fields (already solved efficiently by [CG00]). But, for "small" k the problem seems hard to solve in polynomial time. We do not even know a practical algorithm.

This motivated us to study the case of constant k, with the hope that this will help us invent new tools. In this direction, we make significant progress by giving a unified method to factor $f \bmod p^k$ for $k \leq 4$. We also refine Hensel lifting for $k \leq 4$, by giving all possible lifts of a factor of $f \bmod p$, in the classically hard case of $f \bmod p$ being a power of an irreducible.

We give a general framework (for any k) to work on, by reducing the factoring in a big ring to root-finding in a smaller ring. We leave it open whether we can factor $f \bmod p^5$, and beyond, within this framework.

We also leave it open, to efficiently get all the solutions of a *bivariate* equation, in $\mathbb{Z}/\langle p^k \rangle$ or $\mathbb{F}_p[x]/\langle \varphi^k \rangle$, in a compact representation. Surprisingly, we know how to achieve this for univariate polynomials [BLQ13]. This, combined with our work, will probably give factoring mod p^k, for any k.

Acknowledgements. We thank Vishwas Bhargava for introducing us to the open problem of factoring $f \bmod p^3$. N.S. thanks the funding support from DST (DST/SJF/MSA-01/2013-14). R.M. would like to thank support from DST through grant DST/INSPIRE/04/2014/001799.

References

[Apo13] Tom M Apostol. *Introduction to analytic number theory*. Springer Science & Business Media, 2013. 2

[Ber67] Elwyn R Berlekamp. Factoring polynomials over finite fields. *Bell System Technical Journal*, 46(8):1853–1859, 1967. 1

[BLQ13] Jérémy Berthomieu, Grégoire Lecerf, and Guillaume Quintin. Polynomial root finding over local rings and application to error correcting codes. *Applicable Algebra in Engineering, Communication and Computing*, 24(6):413–443, 2013. https://link.springer.com/article/10.1007/s00200-013-0200-5. 2, 3, 5, 9, 16, 19

[BS86] Zenon I Borevich and Igor R Shafarevich. *Number theory*, volume 20. Academic press, 1986. 2

[CG00] David G Cantor and Daniel M Gordon. Factoring polynomials over p-adic fields. In *International Algorithmic Number Theory Symposium*, pages 185–208. Springer, 2000. 1, 2, 16

[CGRW18] Qi Cheng, Shuhong Gao, J Maurice Rojas, and Daqing Wan. Counting roots of polynomials over prime power rings. In *Thirteenth Algorithmic Number Theory Symposium, ANTS-XIII*. Mathematical Sciences Publishers, 2018. arXiv:1711.01355. 3

[Chi87] Alexander L Chistov. Efficient factorization of polynomials over local fields. *Dokl. Akad. Nauk SSSR*, 293(5):1073–1077, 1987. 1, 2

[Chi94] Alexander L Chistov. Algorithm of polynomial complexity for factoring polynomials over local fields. *Journal of mathematical sciences*, 70(4):1912–1933, 1994. 2

[CL01] Howard Cheng and George Labahn. Computing all factorizations in $\mathbb{Z}_N[x]$. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation, ISSAC'01*, pages 64–71, 2001. 2

[CZ81] David G Cantor and Hans Zassenhaus. A new algorithm for factoring polynomials over finite fields. *Mathematics of Computation*, pages 587–592, 1981. 1, 18

[DH01] Jan Denef and Kathleen Hoornaert. Newton polyhedra and Igusa's local zeta function. *Journal of number Theory*, 89(1):31–64, 2001. 2

[DMS19] Ashish Dwivedi, Rajat Mittal, and Nitin Saxena. Counting basic-irreducible factors mod p^k in deterministic poly-time and p-adic applications. *Computational Complexity Conference*, 2019. https://www.cse.iitk.ac.in/users/nitin/papers/basic-irred-mod-pk.pdf. 3

[FS15] Michael A Forbes and Amir Shpilka. Complexity theory column 88: Challenges in polynomial factorization. *ACM SIGACT News*, 46(4):32–49, 2015. 1

[Hen18] Kurt Hensel. Eine neue theorie der algebraischen zahlen. *Mathematische Zeitschrift*, 2(3):433–452, Sep 1918. 2, 19

[Kal92] Erich Kaltofen. Polynomial factorization 1987–1991. In *Latin American Symposium on Theoretical Informatics*, pages 294–313. Springer, 1992. 1

[Kli97] Adam Klivans. Factoring polynomials modulo composites. Technical report, Carnegie-Mellon Univ, Pittsburgh PA, Dept of CS, 1997. 1

[KRRZ19] Leann Kopp, Natalie Randall, Joseph Rojas, and Yuyu Zhu. Randomized polynomial-time root counting in prime power rings. *Mathematics of Computation*, page 1, 02 2019. 3

[KU11] Kiran S Kedlaya and Christopher Umans. Fast polynomial factorization and modular composition. *SIAM Journal on Computing*, 40(6):1767–1802, 2011. 1

[Lan85] Susan Landau. Factoring polynomials over algebraic number fields. *SIAM Journal on Computing*, 14(1):184–195, 1985. 1

[LLL82] Arjen K Lenstra, Hendrik W Lenstra, and László Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982. 1

[NRS17] Vincent Neiger, Johan Rosenkilde, and Éric Schost. Fast computation of the roots of polynomials over the ring of power series. In *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*, pages 349–356. ACM, 2017. 3

[NZM13] Ivan Niven, Herbert S Zuckerman, and Hugh L Montgomery. *An introduction to the theory of numbers*. John Wiley & Sons, 2013. 2

[Săl05] Ana Sălăgean. Factoring polynomials over $\mathbb{Z}_4$ and over certain galois rings. *Finite fields and their applications*, 11(1):56–70, 2005. 3

[Sha93] Adi Shamir. On the generation of multivariate polynomials which are hard to factor. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, pages 796–804. ACM, 1993. 1

[Sir17] Carlo Sircana. Factorization of polynomials over $\mathbb{Z}/(p^n)$. In *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*, pages 405–412. ACM, 2017. 3

[vzGH96] Joachim von zur Gathen and Silke Hartlieb. Factorization of polynomials modulo small prime powers. Technical report, Paderborn Univ, 1996. 2, 3, 15

[vzGH98] Joachim von zur Gathen and Silke Hartlieb. Factoring modular polynomials. *Journal of Symbolic Computation*, 26(5):583–606, 1998. 2, 15

[vzGP01] Joachim von zur Gathen and Daniel Panario. Factoring polynomials over finite fields: A survey. *Journal of Symbolic Computation*, 31(1-2):3–17, 2001. 1

[Zas69] Hans Zassenhaus. On hensel factorization, I. *Journal of Number Theory*, 1(3):291–311, 1969. 19

[ZG03] WA Zuniga-Galindo. Computing Igusa's local zeta functions of univariate polynomials, and linear feedback shift registers. *Journal of Integer Sequences*, 6(2):3, 2003. 2

A Preliminaries

The following theorem by Cantor-Zassenhaus [CZ81] efficiently finds all the roots of a given univariate polynomial over a finite field.

Theorem 15 (Cantor-Zassenhaus). *Given a univariate degree d polynomial $f(x)$ over a given finite field $\mathbb{F}_q$, we can find all the irreducible factors of $f(x)$ in $\mathbb{F}_q[x]$ in randomized poly($d, \log q$) time.*

Currently, it is a big open question to derandomize the preceding theorem.

Below we state a lemma, originally due to Hensel [Hen18], for $\mathcal{I}$-adic lifting of *coprime* factorization for a given univariate polynomial. Over the years, it has acquired many forms in different texts; the version being presented here is due to Zassenhaus [Zas69].

Lemma 16 (Hensel lemma and lift [Hen18])**.** *Let R be a commutative ring with unity, and let $\mathcal{I} \subseteq R$ be an ideal. Given a polynomial $f(x) \in R[x]$, let $g(x), h(x), u(x), v(x) \in R[x]$ be polynomials, such that, $f(x) = g(x)h(x) \bmod \mathcal{I}$ and $g(x)u(x) + h(x)v(x) = 1 \bmod \mathcal{I}$.*

Then, for any $l \in \mathbb{N}$, we can efficiently compute $g^*, h^*, u^*, v^* \in R[x]$ such that

$$f \;=\; g^* h^* \bmod \mathcal{I}^l \qquad (\textit{called}\ \text{lift of the factorization})$$

where $g^* = g \bmod \mathcal{I}$, $h^* = h \bmod \mathcal{I}$ and $g^*u^* + h^*v^* = 1 \bmod \mathcal{I}^l$.

Moreover, g^* and h^* are unique upto multiplication by a unit.

B Root finding modulo $\varphi(x)^i$

Let us denote the ring $\mathbb{F}_p[x]/\langle \varphi^i \rangle$ by R_0 (for an irreducible $\varphi(x) \bmod p$). In this section, we give an algorithm to find all the roots y of a polynomial $g(y) \in R_0[y]$ in the ring R_0. The algorithm was originally discovered by [BLQ13, Cor.4] to find roots in $\mathbb{Z}/\langle p^i \rangle$, we adapt it here to find roots in R_0.

Note that $R_0/\langle \varphi^j \rangle = \mathbb{F}_p[x]/\langle \varphi^j \rangle$, for $j \le i$, and $R_0/\langle \varphi \rangle =: \mathbb{F}_q$ is the finite field of size $q := p^{\deg(\varphi \bmod p)}$. The structure of a root y of $g(y)$ in R_0 is

$$y \;=\; y_0 + \varphi y_1 + \varphi^2 y_2 + \ldots + \varphi^{i-1} y_{i-1},$$

where $y \in R_0$ and each $y_j \in \mathbb{F}_q$ for all $j \in \{0, \ldots, i-1\}$. Also, recall the notation of $*$ (given in Section 2) and representative roots (in Section 3.1).

The **output** of this algorithm is simply a set of at most $(\deg g)$ many representative roots of $g(y)$. This bound of $\deg g$ is a curious by-product of the algorithm.

Algorithm 2 Root-finding in ring R_0

procedure ROOT-FIND($g(y)$, φ^i)
- **If** $g(y) \equiv 0$ in $R_0/\langle \varphi^i \rangle$ **return** $*$ (every element is a root).
- Let $g(y) \equiv \varphi^\alpha \tilde{g}(y)$ in $R_0/\langle \varphi^i \rangle$, for the unique integer $0 \le \alpha < i$ and the polynomial $\tilde{g}(y) \in R_0/\langle \varphi^{i-\alpha} \rangle[y]$, s.t., $\tilde{g}(y) \not\equiv 0$ in $R_0/\langle \varphi \rangle$ and $\deg(\tilde{g}) \le \deg(g)$.
- Using Cantor-Zassenhaus algorithm find all the roots of $\tilde{g}(y)$ in $R_0/\langle \varphi \rangle$.
- **If** $\tilde{g}(y)$ has no root in $R_0/\langle \varphi \rangle$ then **return** $\{\}$. (Dead-end)
- Initialize $S = \{\}$.
- **for** each root a of $\tilde{g}(y)$ in $R_0/\langle \varphi \rangle$ **do**
  - Define $g_a(y) := \tilde{g}(a + \varphi y)$.
  - $S' \leftarrow$ROOT-FIND($g_a(y)$, $\varphi^{i-\alpha}$).
  - $S \leftarrow S \cup (a + \varphi S')$.

```
11:     end for
12:     return S.
13: end procedure
```

Note that in Step 9 we ensure: $\varphi | g_a(y)$. So, in every other recursive call to Root-find the second argument reduces by at least one. The key reason why $|S| \leq \deg g$ holds: The number of representative roots of $g_a(y)$ are upper bounded by the multiplicity of the root a of $\tilde{g}(y)$.

C Finding all the factors modulo p^3

We will give a method to efficiently get and count all the distinct factors of $f \bmod p^3$, where $f(x) \in \mathbb{Z}[x]$ is a univariate polynomial of degree d.

Theorem 17. *Given $f(x) \in \mathbb{Z}[x]$, a univariate polynomial of degree d and a prime $p \in \mathbb{N}$, we give (and count) all the distinct factors of $f \bmod p^3$ of degree at most d in randomized poly($d, \log p$) time.*

Note: We will not distinguish two factors if they are same up to multiplication by a unit. We will only find monic (leading coefficient 1) factors of $f(x) \bmod p^3$ and assume that f is monic.

Proof of Theorem 17. By Theorem 15 and Lemma 16 we write:

$$f(x) \equiv \prod_{i=1}^{n} f_i(x) \equiv \prod_{i=1}^{n} (\varphi_i^{e_i} + ph_i) \bmod p^3$$

where $f_i(x) \equiv (\varphi_i^{e_i} + ph_i) \bmod p^3$ with $\varphi_i \bmod p^3$ being monic and irreducible mod p, $e_i \in \mathbb{N}$, and $h_i(x) \bmod p^3$ of degree $< e_i \deg(\varphi_i)$, for all $i \in [n]$.

Thus, without loss of generality, consider the case of $f \equiv \varphi^e + ph$.

By Reduction theorem (Theorem 8) finding factors of the form $\varphi^a - py \bmod p^3$ of $f \equiv \varphi^e + ph \bmod p^3$, for $a \leq e/2$, is equivalent to finding all the roots of the equation

$$E(y) \equiv f(x)(\varphi^{2a} + \varphi^a(py) + (py)^2) \equiv 0 \bmod \langle p^3, \varphi^{3a} \rangle.$$

Consider $R := \mathbb{Z}[x]/\langle p^3, \varphi^{3a} \rangle$ and $R_0 := \mathbb{Z}[x]/\langle p, \varphi^{3a} \rangle$, analogous to those in Section 2.

Using Lemma 9, we know that all solutions of the equation $E(y) \equiv 0 \bmod \langle p^3, \varphi^{3a} \rangle$ will be of the form $y = y_0 + p* \in R$, for a $y_0 \in R_0$. On simplifying this equation we get $E(y) \equiv ph\varphi^{2a} + (p^2 h \varphi^a) y_0 + (p^2 \varphi^e) y_0^2 \equiv 0 \bmod \langle p^3, \varphi^{3a} \rangle$.

Reducing this equation mod $\langle p^2, \varphi^{3a} \rangle$, we get that $h \equiv 0 \bmod \langle p, \varphi^a \rangle$ is a necessary condition for a root y_0 to exist. So, we get

$$E(y) \equiv p^2 g_2 \varphi^{2a} + (p^2 g_1 \varphi^{2a}) y_0 + (p^2 \varphi^e) y_0^2 \equiv 0 \bmod \langle p^3, \varphi^{3a} \rangle,$$

where $h := \varphi^a g_1 + p g_2$ for unique $g_1, g_2 \in \mathbb{F}_p[x]$.

This equation is already divisible by p^2 as well as φ^{2a} and so using Claim 6 we get that, finding factors of the form $\varphi^a - py \bmod p^3$ of $f \equiv \varphi^e + ph \bmod p^3$, for $a \le e/2$, is equivalent to finding all the roots of the equation

$$g_2 + g_1 y_0 + \varphi^{e-2a} y_0^2 \equiv 0 \bmod \langle p, \varphi^a \rangle .$$

We find all the roots of this equation using one call to Root-find in randomized poly$(d, \log p)$ time. Note that any output root u_0 lives in $\mathbb{F}_p[x]/\langle \varphi^a \rangle$ and so its degree in x is $< a \deg(\varphi)$. This yields *monic* factors of $f \bmod p^3$ (with $0 \le a \le e/2$).

For $e \ge a > e/2$, we can replace a by $b := e - a$ in the above steps. Once we get a factor $\varphi^b - py \bmod p^3$, we output the cofactor $f/(\varphi^b - py)$ (which remains monic).

Counting these factors can be easily done in poly-time.

In the general case, if N_i is the number of factors of $f_i \bmod p^3$ then, $\prod_{i=1}^n N_i$ is the count on the number of distinct monic factors of $f \bmod p^3$. □

D Barriers to extension modulo p^5

The reader may wonder about polynomial factoring when k is greater than 4. In this section we will discuss the issues in applying our techniques to factor $f(x) \bmod p^5$.

Given $f(x) \equiv \varphi^e \bmod p$, finding one of its factor $\varphi^a - py \bmod p^5$, for $a \le e/2$ and $y \in (\mathbb{Z}/\langle p^5 \rangle)[x]$, is reduced to solving the equation

$$E(y) := f(x)(\varphi^{4a} + \varphi^{3a}(py) + \varphi^{2a}(py)^2 + \varphi^a(py)^3 + (py)^4) \equiv 0 \bmod \langle p^5, \varphi^{5a} \rangle \qquad (4)$$

By Lemma 9, the roots of $E(y) \bmod \langle p^5, \varphi^{5a} \rangle$ are of the form $y = y_0 + py_1 + p^2 y_2 + p^3 *$ in R, where $y_0, y_1, y_2 \in R_0$ need to be found.

First issue. The first hurdle comes when we try to reduce root-finding modulo the bi-generated ideal $\langle p^5, \varphi^{5a} \rangle \subseteq \mathbb{Z}[x]$ to root-finding modulo the principal ideal $\langle \varphi^{5a} \rangle \subseteq \mathbb{F}_p[x]$. In the case $k = 4$, Lemma 11 guarantees that we need to solve at most two related equations of the form $E'(y_0, y_1) \equiv 0 \bmod \langle p, \varphi^{4a} \rangle$ to find exactly the roots of $E(y) \bmod \langle p^4, \varphi^{4a} \rangle$. Below, for $k = 5$, we show that we have exponentially many candidates for $E'(y_0, y_1, y_2) \in R_0[y_0, y_1, y_2]$ and it is not clear if there is any compact efficient representation for them.

Putting $y = y_0 + py_1 + p^2 y_2$ in Eqn. 4 we get,

$$E(y) =: E_1(y_0) + E_2(y_0) y_1 + E_3(y_0) y_2 + (f \varphi^{2a} p^4) y_1^2 \equiv 0 \bmod \langle p^5, \varphi^{5a} \rangle, \qquad (5)$$

where $E_1(y_0) := f\varphi^{4a} + f\varphi^{3a} p y_0 + f\varphi^{2a} p^2 y_0^2 + f\varphi^a p^3 y_0^3 + f p^4 y_0^4$ is a quartic in $R[y_0]$, $E_2(y_0) := f\varphi^{3a} p^2 + f\varphi^{2a} 2p^3 y_0 + f\varphi^a 3p^4 y_0^2$ is a quadratic in $R[y_0]$ and $E_3(y_0) := f\varphi^{3a} p^3 + f\varphi^{2a} 2p^4 y_0$ is linear in $R[y_0]$.

To divide Eqn. 5 by p^3, we go mod $\langle p^3, \varphi^{5a} \rangle$ obtaining

$$E(y) \equiv E_1(y_0) \equiv f\varphi^{4a} + f\varphi^{3a} p y_0 + f\varphi^{2a} p^2 y_0^2 \equiv 0 \bmod \langle p^3, \varphi^{5a} \rangle,$$

a univariate quadratic equation which requires the whole machinery used in the case $k = 3$. We get this simplified equation since $E_3(y_0) \equiv 0 \bmod \langle p^3, \varphi^{5a}\rangle$ and $E_2(y_0) \equiv f\varphi^{3a}p^2 \equiv \varphi^{e-2a}\varphi^{2a+3a}p^2 \equiv 0 \bmod \langle p^3, \varphi^{5a}\rangle$.

But, to really reduce Eqn. 5 to a system modulo the principal ideal $\langle \varphi^{5a}\rangle \subseteq \mathbb{F}_p[x]$, we need to divide it by p^4. So, we go mod $\langle p^4, \varphi^{5a}\rangle$:

$$E(y) \;\equiv\; E_1^{'}(y_0) + E_2^{'}(y_0)y_1 \;\equiv\; 0 \;\bmod \langle p^4, \varphi^{5a}\rangle$$

where $E_1^{'}(y_0) \equiv E_1(y_0) \bmod \langle p^4, \varphi^{5a}\rangle$ is a cubic in $R[y_0]$ and $E_2^{'}(y_0) \equiv E_2(y_0) \bmod \langle p^4, \varphi^{5a}\rangle$ is linear in $R[y_0]$. This requires us to solve a special bivariate equation which requires the machinery used in the case $k = 4$.

Now, the problem reduces to computing a solution pair $(y_0, y_1) \in (R_0)^2$ of this bivariate. We can apply the idea used in Algorithm 1 to get all valid y_0 efficiently, but since y_1 is a function of y_0, we need to compute exponentially many y_1's. So, there seem to be exponentially many candidates for $E'(y_0, y_1, y_2)$, that behaves like $E(y)/p^4$ and lives in $(\mathbb{F}_p[x]/\langle \varphi^{5a}\rangle)[y_0, y_1, y_2]$. At this point, we are forced to compute all these E's, as we do not know which one will lead us to a solution of Eqn. 5.

Second issue. Even if we resolve the first issue and get a valid E', we are left with a trivariate equation to be solved mod $\langle p, \varphi^{5a}\rangle$ (Eqn. 5 after shifting y_0 and y_1 then dividing by p^4). We could do this when k was 4, because we could easily write y_1 as a function of y_0. Though, it is unclear how to solve this trivariate now as the equation is *nonlinear* in both y_0 and y_1.

For $k > 5$ the difficulty will only increase because of the recursive nature of Eqn. 4 with more and more number of unknowns (with higher degrees).