RACSAM
Rev. R. Acad. Cien. Serie A. Mat.
VOL. **98 (**1), 2004, pp. 239–258
Álgebra / Algebra

P-adic Root Isolation

Thomas Sturm and Volker Weispfenning

Abstract. We present an implemented algorithmic method for counting and isolating all p-adic roots of univariate polynomials f over the rational numbers. The roots of f are uniquely described by p-adic isolating balls, that can be refined to any desired precision; their p-adic distances are also computed precisely. The method is polynomial space in all input data including the prime p. We also investigate the uniformity of the method with respect to the coefficients of f and the primes p. Our method thus provides information analogous to that provided by well-established real methods as, e.g., Cauchy bounds and Sturm sequences over the reals.

Separación de raíces P-ádicas

Resumen. Presentamos un método algorítmico implementado para contar y aislar todas las raíces p-ádicas de polinomios f en una variable con coeficientes racionales. Las raíces de f se describen unívocamente mediante bolas p-ádicas que las aíslan y que pueden ser refinadas hasta cualquier precisión que se desee; sus distancias p-ádicas son también computadas con precisión. El método posee complejidad polinomial en todos los datos de entrada, incluído el primo p. También investigamos la uniformidad del método con respecto a los coeficientes de f y los números primos p. Nuestro método suministra información análoga a la que suministran métodos bien establecidos para los números reales, como por ejemplo, las cotas de Cauchy y la sucesión de Sturm.

1. Introduction

The field $\mathbb{R}$ of real numbers and the fields $\mathbb{Q}_p$ of p-adic numbers for a prime p are similar in many respects: Both are completions of the rational field $\mathbb{Q}$ with respect to some multiplicative norm; both have decidable first-order theories, allow quantifier-elimination in suitable languages and have natural complete first-order axiomatizations [2, 3, 10, 11, 6, 15, 21, 22, 23, 7]. For the reals $\mathbb{R}$ the crucial axioms are the Intermediate Value Theorem for polynomials of arbitrary degree; for the p-adics $\mathbb{Q}_p$ the crucial axioms are Hensel's Lemma for polynomials of arbitrary degree [12, 16, 6, 15, 21, 22, 23]. In both fields each zero of a univariate polynomial can be described uniquely: In the case of the reals $\mathbb{R}$ either by isolating intervals or via Thom's Lemma [5, 4]; in the case of the p-adics $\mathbb{Q}_p$ as Hensel zeros with given residue class [6, 7, 22, 23].

The most famous fundamental algorithmic result for the reals is Sturm's Theorem [19, 4], which provides an explicit formula for counting the number of real zeros of a real univariate polynomial in a given interval. Together with its generalizations by Sylvester and Habicht [4] it is the basis of most algorithmic methods for real polynomials. In particular it allows an algorithmic construction of a system of pairwise disjoint isolating intervals for the real zeros of a given polynomial $f \in \mathbb{Q}[X]$ and the sign-evaluation of a

Presentado por Luis M. Laita.
Recibido: December 5, 2003. *Aceptado*: Ocotber 13, 2004.
Palabras clave / Keywords: P-adic Fields, Root Isolation, Root Counting, Algorithm, Implementation.
Mathematics Subject Classifications: 12J12, 68W30.

further polynomial $g \in \mathbb{Q}[X]$ at all real zeros of f. These algorithmic methods for the reals $\mathbb{R}$ have been implemented and optimized for many years in several computer algebra systems and packages.

By the way of contrast, the corresponding algorithmic methods for the p-adics $\mathbb{Q}_p$ have remained largely theoretical [2, 3, 10, 11, 6, 15, 21, 22, 23, 7]. A notable exception is the REDLOG package of REDUCE, which provides a quantifier elimination and decision method for first-order linear problems over the p-adics [8, 9, 20]. The computer algebra system Magma contains an efficient numerical p-adic root finder, which computes approximations of the p-adic zeros of a univariate polynomial with rounded p-adic coefficients to any prescribed p-adic precision; see [17] for the algorithm used there. Just like approximative numerical zero finding algorithms in the reals this approach is not well-suited for our goals.

Even on the theoretical side, however, there appears to be no stringent analogue for the theorems of Sturm and Sylvester on root counting and root localization with side conditions for the p-adics. While it is clear from the general algorithmic quantifier elimination and decision methods that such an analogue must exist, an explicit formulation, let alone an implementation, had been missing so far. An early theoretical algorithm for determining, whether a multivariate polynomial has at least one p-adic zero appears in [16].

In the present paper we fill this gap:

- We provide an algorithm for computing isolating p-adic balls for all p-adic zeros of a given polynomial $f \in \mathbb{Q}[X]$.
- We provide an algorithm for refining the isolating p-adic balls obtained by our algorithm above to arbitrary precision.
- The output of our algorithms explicitly provides the distances between the different p-adic zeros of f.
- From our presented algorithms one can straightforwardly derive an algorithm for computing the exact number of p-adic zeros of f in a prescribed p-adic ball.

Thus we have complete analogues of the algorithmic method of Sturm for the p-adics. Our algorithms have been implemented by the first author in a REDUCE package named PROOTS. This package is available for download and will be included in the next release of REDUCE. A number of explicit computational examples illustrates the range of possible applications.

On the theoretical side, we provide explicit upper bounds on the asymptotic complexity of our algorithms for large polynomial degrees and/or large primes p. Furthermore, we investigate the uniformity of the algorithms with respect to variations of the prime p and/or variations in the coefficients of the input polynomials f and g.

For values of the prime p that are large in comparison to the coefficients of the input polynomials the situation is particularly easy and pleasant. Here we can formulate a very close analogue of Sturm's Theorem.

As a further analogue to real algebra [13], we prove a theorem that bounds the number of p-adic zeros of f solely in terms of the number of monomials of f, the prime p, and a certain p-adic value computed from the coefficients of f independently of the degree of f.

2. Possible Values of Zeros

Our initial idea is that the strict triangle inequality for v imposes surprisingly hard restrictions on the possible value of zeros. Let $f \in \mathbb{Q}[X]$ be a nonzero polynomial

$$f = \sum_{i=0}^{n} a_i X^i \quad \text{with} \quad a_n \neq 0.$$

Consider $c \in \mathbb{Q}_p$ with $c \neq 0$ and $f(c) = 0$. Then $v\big(\sum_{i=0}^{n} a_i c^i\big) = \infty$. Assume for a contradiction that there is exactly one $a_i c^i$ of minimal value in this sum. Then we can conclude by the strict triangle inequality

$$v\left(\sum_{i=0}^{n} a_i c^i\right) = v(a_i c^i) < \infty.$$

So there must be at least two such summands. These informal considerations motivate the following definitions, which exploit them without talking about c.

A *balancable pair* for f is a pair $(a_j X^j, a_k X^k)$ of monomials of f, where $j < k$, $a_j \neq 0$, $a_k \neq 0$, and

$$\frac{v(a_j) - v(a_k)}{k - j} \in \mathbb{Z}.$$

This integer $\frac{v(a_j)-v(a_k)}{k-j}$ is called the *balancing value* of the balancable pair $(a_j X^j, a_k X^k)$. A balancable pair $(a_j X^j, a_k X^k)$ is *critical* for f if

$$\frac{kv(a_j) - jv(a_k)}{k - j} = \min_{i \in \{1,\ldots,n\}} v(a_i) + i\frac{v(a_j) - v(a_k)}{k - j}.$$

Its balancing value is then called a *critical value* for f.

Lemma 1 (Possible Values of Zeros) *Let $f = \sum_{i=0}^{n} a_i X^i \in \mathbb{Q}[X]$ with $f \neq 0$. Let $c \in \mathbb{Q}_p$ with $c \neq 0$ such that $f(c) = 0$. Then $v(c)$ is a critical value for f.*

PROOF. To start with, we observe that from $f(c) = 0$, it follows that $v(f(c)) = \infty$. Define

$$\mu = \min_{i \in \{1,\ldots,n\}} v(a_i c^i) \quad \text{and} \quad M = \big\{\, i \in \{0, \ldots, n\} \;\big|\; v(a_i c^i) = \mu \,\big\}.$$

We may conclude from $f \neq 0$ and $c \neq 0$ that $\mu < \infty$. This implies in turn that $a_i \neq 0$ for $i \in M$. Next, we have that $|M| \geq 2$, because $M = \{i\}$ for some $i \in \{0, \ldots, n\}$ would by the strict triangle inequality yield the contradiction

$$\infty = v(f(c)) = v(a_i c^i) = \mu < \infty.$$

We can thus choose j, $k \in M$ with $j < k$, and consider the pair $(a_j X^j, a_k X^k)$. ¿From the equality $v(a_j c^j) = \mu = v(a_k c^k)$ we equivalently obtain

$$v(a_j) + jv(c) = v(a_k) + kv(c) \iff v(a_j) - v(a_k) = (k - j)v(c) \iff \frac{v(a_j) - v(a_k)}{k - j} = v(c).$$

Since $v(c) \in \mathbb{Z}$, this shows in particular that $(a_j X^j, a_k X^k)$ is a balancable pair. Assume for a contradiction that $(a_j X^j, a_k X^k)$ is not critical for f. Then there is $i \in \{0, \ldots, n\}$ such that

$$v(a_i c^i) = v(a_i) + i\frac{v(a_j) - v(a_j)}{k - j} < \frac{kv(a_j) - jv(a_k)}{k - j} = v(a_j) + j\frac{v(a_j) - v(a_j)}{k - j} = v(a_j c^j) = \mu,$$

which contradicts the choice of μ. ■

Notice that the number of critical values for f is bounded by the number of critical balancable pairs, which is in turn bounded by $n(n-1)/2$. Hence there are only finitely many possible values for zeros of f, and a finite superset of these values can be computed from f.

For illustrating our notions by means of examples, we make various choices for the valuation v_p, and then consider the polynomial

$$f = 81X^4 - 6X + 5.$$

For $p = 2$, the pair $(5, -6X)$ is balancable with balancing value -1. It is, however, not critical, because $v(5) = v(-6) - 1 > v(81) - 4$. The pair $(5, 81X^4)$ is balancable with balancing value 0. This one is critical, because $v(5) = v(81) \leq v(-6)$. The pair $(-6X, 81X^4)$ is not balancable. The critical values for f are -1 and 0.

For $p = 3$, all the pairs $(5, -6X)$, $(5, 81X^4)$, and $(-6X, 81X^4)$ are balancable with the same balancing value -1. They are all critical, because $v(5) = v(-6) - 1 = v(81) - 4$. There is only one critical value for f, which is -1.

For $p = 5$, the pair $(5, -6X)$ is balancable with balancing value 1. This is a critical pair, because $v(5) = v(-6) + 1 \leq v(81) + 4$. The pair $(5, 81X^4)$ is not balancable. The pair $(-6X, 81X^4)$ is balancable with balancing value 0. It is also critical, because $v(5) > v(-6) = v(81)$. The critical values for f are 1 and 0.

3. Normalization

In the previous section we have computed finitely many critical values, which are possible values of zeros. We are now going to derive for each such value ν and an associated critical pair $(a_k X^j, a_k X^k)$ a transformed polynomial $f_{j,k}$ from the original polynomial. The idea is that roots of value zero of $f_{j,k}$ correspond to roots of value ν of f. In addition, we take care that $f_{j,k} \in \mathbb{Z}_p[X]$.

As in the previous section let $f = \sum_{i=0}^{n} a_i X^i \in \mathbb{Q}[X]$ with $f \neq 0$. Let $(a_j X^j, a_k X^k)$ be a critical balancable pair for f. Then the *j-k-normalization* of f is defined as

$$f_{j,k} = p^{-\frac{kv(a_j)-jv(a_k)}{k-j}} \sum_{i=0}^{n} a_i \left(p^{\frac{v(a_j)-v(a_k)}{k-j}} X\right)^i = \sum_{i=0}^{n} a_i p^{\frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j}} X^i.$$

As an example choose $p = 5$ and reconsider our polynomial $f = 81X^4 - 6X + 5$ from the previous section. We obtain

$$f_{0,1} = 5^{-1}\big(81(5X)^4 - 6(5X) + 5\big) = 10125X^4 - 6X + 1 \quad \text{and} \quad f_{1,4} = f.$$

Lemma 2 (Zeros Under j-k-Normalization) *Let $f \in \mathbb{Q}[X]$ with $f \neq 0$. Let $(a_j X^j, a_k X^k)$ be a critical balancable pair for f. Then the following hold:*

(i) For $z \in \mathbb{Q}_p$ we have $f_{j,k}(z) = 0$ if and only if $f\left(p^{\frac{v(a_j)-v(a_k)}{k-j}} z\right) = 0$.

(ii) For $z \in \mathbb{Q}_p$ we have $(f_{j,k})'(z) = 0$ if and only if $f'\left(p^{\frac{v(a_j)-v(a_k)}{k-j}} z\right) = 0$.

PROOF. To start with, observe that since $(a_j X^j, a_k X^k)$ is critical, we have $\frac{kv(a_j)-jv(a_k)}{k-j} < \infty$ and thus $p^{-\frac{kv(a_j)-jv(a_k)}{k-j}} \neq 0$.

(i) By definition of $f_{j,k}$ we have

$$f_{j,k} = p^{-\frac{kv(a_j)-jv(a_k)}{k-j}} \sum_{i=0}^{n} a_i \left(p^{\frac{v(a_j)-v(a_k)}{k-j}} X\right)^i.$$

¿From this together with our initial observation, the assertion in Part (i) follows immediately.

(ii) Using the definition of $f_{j,k}$ as in Part (i) and the chain rule, the derivative of $f_{j,k}$ is obtained as

$$(f_{j,k})' = \left(p^{-\frac{kv(a_j)-jv(a_k)}{k-j}} f\left(p^{\frac{v(a_j)-v(a_k)}{k-j}} X\right)\right)' = p^{-\frac{kv(a_j)-jv(a_k)}{k-j}} p^{\frac{v(a_j)-v(a_k)}{k-j}} f'\left(p^{\frac{v(a_j)-v(a_k)}{k-j}} X\right).$$

By the definition of the balancable pair (a_jX^j, a_kX^k) we have $\frac{v(a_j)-v(a_k)}{k-j} \in \mathbb{Z}$ and thus certainly $p^{\frac{v(a_j)-v(a_k)}{k-j}} \neq 0$. From this together with our initial observation, the assertion in Part (ii) follows immediately. ■

Lemma 3 (Properties of the j-k-Normalization) *Let $f = \sum_{i=0}^{n} a_iX^i \in \mathbb{Q}[X]$ with $f \neq 0$. Let (a_jX^j, a_kX^k) be a critical balancable pair for f. Then the following hold:*

(i) $f_{j,k}$ is square-free if and only if f is square-free.

(ii) All the coefficients of $f_{j,k}$ are p-adic integers: $f_{j,k} \in \mathbb{Z}_p[X]$.

(iii) Both the j-th and k-th coefficient of $f_{j,k}$ have value 0.

(iv) If $f \in \mathbb{Z}[X]$, then $f_{j,k} \in \mathbb{Z}[X]$.

PROOF.

(i) This is an immediate consequence of Lemma 2.

(ii) Making use of the fact that (a_jX^j, a_kX^k) is critical for f, we obtain for $i \in \{1, \ldots, n\}$ for the value of the i-th coefficient

$$\begin{aligned} v\Big(a_ip^{\frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j}}\Big) &= v(a_i) + \frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j} \\ &= \Big(v(a_i) + i\frac{v(a_j)-v(a_k)}{k-j}\Big) - \frac{kv(a_j)-jv(a_k)}{k-j} \\ &\geq 0. \end{aligned}$$

(iii) For the values of the j-th and k-th coefficients we obtain

$$v\Big(a_jp^{\frac{(j-k)v(a_j)-(j-j)v(a_k)}{k-j}}\Big) = v(a_j) + \frac{(j-k)v(a_j)-(j-j)v(a_k)}{k-j} = v(a_j) - v(a_j) = 0$$

and

$$v\Big(a_kp^{\frac{(k-k)v(a_j)-(k-j)v(a_k)}{k-j}}\Big) = v(a_k) + \frac{(k-k)v(a_j)-(k-j)v(a_k)}{k-j} = v(a_k) - v(a_k) = 0,$$

respectively.

(iv) Let $i \in \{1, \ldots, n\}$. Since $a_i \in \mathbb{Z}$, there are $a_i' \in \mathbb{Z}$ and $m \in \mathbb{N}$ such that $a_i = a_i'p^m$ with $p \nmid a_i'$. In these terms, the i-th coefficient of $f_{j,k}$ can be rewritten as follows:

$$a_ip^{\frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j}} = a_i'p^mp^{\frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j}} = a_i'p^{m+\frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j}}.$$

According to Part (i) we have

$$m + \frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j} = v\Big(a_ip^{\frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j}}\Big) \geq 0$$

and thus $a_i'p^{m+\frac{(i-k)v(a_j)-(i-j)v(a_k)}{k-j}} \in \mathbb{Z}$. ■

4. Roots of Value Zero

When counting the different p-adic zeros of a univariate polynomial with integer coefficients we may restrict our attention to the case, where f is square-free. Otherwise we pass from f to the square-free part of f by dividing f by the greatest common divisor of f and f'.

By the results of the previous section all non-zero p-adic roots of f have critical p-adic value. Moreover for any fixed critical value $\nu = \frac{v(a_j)-v(a_k)}{k-j}$, the p-adic roots of f of value ν are in one-to-one correspondence with the roots of p-adic value zero of normalizations $f_{j,k}$ of f with respect to critical balancable pairs $(a_j X^j, a_k X^k)$ of value ν. The correspondence is effected by a simple multiplication by the corresponding power p^ν of the prime p.

So from now on we study only zeros of p-adic value zero of polynomials such as the $f_{j,k}$ with p-adic integer rational coefficients of minimal p-adic value zero.

Given $g \in \mathbb{Z}_p[X]$, our localization of all p-adic roots of g with value zero is based on Theorem 1 below. This theorem make use of Hensel's Lemma in a form introduced by the second author as Lemma 2.2 in [22]. We restate this result here for the sake of a self-contained description:

Lemma 4 (Hensel's Lemma) *Let $f \in \mathbb{Z}_p[X]$, let $a \in \mathbb{Q}_p$ with $v(a) \geq 0$, and let $\alpha \in \mathbb{Z}$ such that*

$$v\big(f'(a)\big) \leq \alpha \quad \textit{and} \quad 2\alpha < v\big(f(a)\big).$$

Then there is $z \in \mathbb{Q}_p$ with $v(z) = 0$ such that $f(z) = 0$ and $v(z-a) > \alpha$. $\square$

Theorem 1 (Main Theorem) *Let $g \in \mathbb{Z}_p[X]$, let $\beta \in \mathbb{Z}$ with $\beta \geq 0$, and let $c \in \mathbb{Q}_p$ with $v(c) = 0$.*

(i) Let $d \in \mathbb{Q}_p$ such that $v(c-d) > \beta$. Then

$$v\big(g(c)\big) \leq \beta \quad \textit{iff} \quad v\big(g(d)\big) \leq \beta,$$

and in the positive case we even have $v\big(g(c)\big) = v\big(g(d)\big)$.

(ii) Let $z \in \mathbb{Q}_p$ such that $v(c-z) > \beta$, let $g(z) = 0$, and let $v\big(g'(c)\big) \leq \beta$ or $v\big(g'(z)\big) \leq \beta$. Then

$$v\big(g'(c)\big) = v\big(g'(z)\big) \leq \beta \quad \textit{and} \quad \beta < v\big(g(c)\big) = v\big(g'(z)\big) + v(c-z).$$

(iii) Let $v\big(g'(c)\big) \leq \beta$ and $2\beta < v\big(g(c)\big)$. Then there exists one and only one $z \in \mathbb{Q}_p$ with $v(z) = 0$ such that

$$g(z) = 0 \quad \textit{and} \quad v(z-c) > \beta.$$

PROOF. Let $g = \sum_{i=0}^{n} a_i X^i$ with $a_n \neq 0$. Then the Taylor expansion of g at point $z \in \mathbb{Z}_p$ is of the form

$$g = g(z) + \sum_{i=1}^{n} b_i (X-z)^i \quad \text{with} \quad b_i = \frac{g^{(i)}(z)}{i!}.$$

Note that we have $\frac{g^{(i)}(X)}{i!} \in \mathbb{Z}_p[X]$ and thus $v(b_i) \geq 0$; in particular $v\big(g'(z)\big) = v(b_1) \geq 0$.

(i) Considering the Taylor expansion of g at point d and evaluating this expansion at point c we obtain

$$v\big(g(c)-g(d)\big) = v\left(\sum_{i=1}^{n} \frac{g^{(i)}(d)}{i!}(c-d)^i\right) \geq \min_{i\in\{1,\ldots,n\}} \left(v\left(\frac{g^{(i)}(d)}{i!}\right) + iv(c-d)\right) > \beta.$$

Let $v\big(g(c)\big) \leq \beta$, and assume for a contradiction that $v\big(g(d)\big) > \beta$. Then

$$v\big(g(c)-g(d)\big) = \min\big(v\big(g(c)\big), v\big(g(d)\big)\big) = v\big(g(c)\big) \leq \beta,$$

a contradiction. Analogously, it follows from $v\big(g(d)\big) \leq \beta$ that $v\big(g(c)\big) \leq \beta$.

Consider now the positive case that both $v\big(g(c)\big) \leq \beta$ and $v\big(g(d)\big) \leq \beta$. Assume for a contradiction that $v\big(g(c)\big) \neq v\big(g(d)\big)$, wlog. $v\big(g(c)\big) < v\big(g(d)\big)$. Then again

$$v\big(g(c) - g(d)\big) = \min\big(v\big(g(c)\big), v\big(g(d)\big)\big) = v\big(g(c)\big) \leq \beta.$$

(ii) By Part (i) it follows from $v(c - z) > \beta$ in combination with $v\big(g'(c)\big) \leq \beta$ or $v\big(g'(z)\big) \leq \beta$ that $v\big(g'(c)\big) = v\big(g'(z)\big) \leq \beta$.

On our assumption that $g(z) = 0$ we obtain the following Taylor expansion of g at point z evaluated at c:

$$g(c) = g'(z)(c - z) + (c - z)^2 \sum_{i=2}^{n} b_i (c - z)^{i-2} \quad \text{with} \quad b_i = \frac{g^{(i)}(z)}{i!}.$$

We are going to estimate the values of the two summands: From $v\big(g'(z)\big) \leq \beta$ and $v(c - z) > \beta$, we straightforwardly obtain

$$v\big(g'(z)(c - z)\big) = v\big(g'(z)\big) + v(c - z) < 2v(c - z).$$

For the other summand we recall that $v(b_i) \geq 0$ and observe that $(i - 2)v(c - z) > (i - 2)\beta \geq 0$, where $i \in \{2, \ldots, n\}$. We thus obtain

$$v\left((c - z)^2 \sum_{i=2}^{n} b_i (c - z)^{i-2}\right) \geq 2v(c - z) + \min_{i \in \{2,\ldots,n\}} \big(v(b_i) + (i - 2)v(c - z)\big) \geq 2v(c - z).$$

Hence by the strict triangle inequality

$$v\big(g(c)\big) = v\big(g'(z)(c - z)\big) = v\big(g'(z)\big) + v(c - z) \geq v(c - z) > \beta.$$

(iii) The existence of z is guaranteed by Hensel's Lemma 4. As for the uniqueness, let $z' \in \mathbb{Q}_p$ such that $g(z') = 0$ and $v(z' - c) > \beta$. It follows that also

$$v(z' - z) = v\big((z' - c) - (z - c)\big) \geq \min\big(v(z' - c), v(z - c)\big) > \beta.$$

Using Part (ii) we obtain

$$\infty = v\big(g(z')\big) = v\big(g'(z)\big) + v(z' - z) \leq \beta + v(z' - z).$$

Hence $v(z' - z) = \infty$, i.e. $z' = z$. ■

Similar to Hensel's Lemma above, Part (ii) of the following lemma is a citation from the literature, namely from a textbook by Akritas [1].

Lemma 5 (Properties of the Discriminant Value) *Let $g \in \mathbb{Z}_p[X]$ be square-free, and denote by $\operatorname{discr}(g)$ the discriminant of g.*

(i) $0 \leq v\big(\operatorname{discr}(g)\big) < \infty$

(ii) $\operatorname{discr}(g)$ *has a representation in the form* $\operatorname{discr}(g) = rg + sg'$ *with* $r, s \in \mathbb{Z}_p[X]$.

(iii) Let $c \in \mathbb{Q}_p$ with $v(c) = 0$ such that $v\big(\operatorname{discr}(g)\big) < v\big(g(c)\big)$. Then

$$v\big(g'(c)\big) \leq v\big(\operatorname{discr}(g)\big).$$

This holds in particular if even $2v\big(\operatorname{discr}(g)\big) < v\big(g(c)\big)$; notice that then c and $\operatorname{discr}(g)$ satisfy the requirements in Hensel's Lemma 4.

(iv) Let $z \in \mathbb{Q}_p$ with $v(z) = 0$ such that $v\big(g(z)\big) = 0$. Then

$$v\big(g'(z)\big) \le v\big(\mathrm{discr}(g)\big) \quad \textit{and} \quad 2v\big(\mathrm{discr}(g)\big) < v\big(g(z)\big) = \infty.$$

That is, every zero of g with value 0 satisfies the requirements in Hensel's Lemma 4 for all choices $\alpha \ge v\big(\mathrm{discr}(g)\big)$.

PROOF.

(i) $0 \le v\big(\mathrm{discr}(g)\big)$ follows from the fact the discriminant, as a particular resultant, is a polynomial form in the coefficients of g. Since g is square-free, we have $\mathrm{discr}(g) \neq 0$ and thus $v\left(\mathrm{discr}(g)\right) < \infty$.

(ii) This has been proved as Theorem 5.2.4 in [1].

(iii) According to Part (ii) we have

$$v\big(\mathrm{discr}(g)\big) = v\big(r(c)g(c) + s(c)g'(c)\big) \ge \min\big(v\big(r(c)\big) + v\big(g(c)\big), v\big(s(c)\big) + v\big(g'(c)\big)\big).$$

Since $v\big(r(c)\big) \ge 0$, $v\big(s(c)\big) \ge 0$, and $v\left(g(c)\right) > v\big(\mathrm{discr}(g)\big)$, we can conclude that

$$v\big(r(c)\big) + v\big(g(c)\big) > v\big(\mathrm{discr}(g)\big) = v\big(s(c)\big) + v\big(g'(c)\big) \ge v\big(g'(c)\big).$$

(iv) This is an immediate consequence of Part (iii). ■

Corollary 1 (to the Main Theorem) *Let $g \in \mathbb{Z}_p[X]$ be square-free, let $\alpha = v\big(\mathrm{discr}(g)\big)$, and let $c \in \mathbb{Q}_p$ with $v(c) = 0$.*

(i) Let $z \in \mathbb{Q}_p$ such that $v(c - z) > \alpha$. Let $g(z) = 0$. Then

$$v\big(g'(c)\big) = v\big(g'(z)\big) \le \alpha \quad \textit{and} \quad \alpha < v\big(g(c)\big) = v\big(g'(z)\big) + v(c - z).$$

(ii) Let $2\alpha < v\big(g(c)\big)$. Then there exists one and only one $z \in \mathbb{Q}_p$ with $v(z) = 0$ such that

$$g(z) = 0 \quad \textit{and} \quad v(z - c) > \alpha.$$

PROOF.

(i) By Lemma 5(iv), we have $v\left(g'(z)\right) \le \alpha$. We can thus apply Theorem 1(ii).

(ii) By Lemma 5(iii) we have $v\left(g'(z)\right) \le \alpha$. We can now apply Theorem 1(iii). ■

4.1. Root Isolation

Let us say that $c_1, c_2 \in \mathbb{Z}_p$ with $v(c_1) = v(c_2) = 0$ are *β-close* if $v(c_1 - c_2) > \beta$. This corresponds to membership $c_1 - c_2 \in I_\beta$ in the ideal $I_\beta = \{\, x \in \mathbb{Z}_p \mid v(x) > \beta \,\}$. Thus β-closeness is a congruence relation on $\mathbb{Z}_p$. In terms of the p-adic absolute value β-closeness states that $|c_1 - c_2| < p^{-\beta}$.

4.1.1. Brute Force Algorithm

In terms of the definition above, Corollary 1 now guarantees the following for $c \in \mathbb{Q}_p$ with $v(c) = 0$:

(i) If c is 2α-close to a root of g, then we will observe that that $v\big(g(c)\big) > 2\alpha$.

(ii) Whenever we make this observation $v\big(g(c)\big) > 2\alpha$, then we can be sure that c is at least α-close to a root of g. Moreover this root is then the only one that is α-close to c.

On the other hand, we can get 2α-close to each root with value zero of g by trying one representative of each nonzero congruence class in $\mathbb{Z}_p/I_{2\alpha}$ for the ideal $I_{2\alpha} = \{\, x \in \mathbb{Z}_p \mid v(x) > 2\alpha \,\}$. This gives rise to the following brute force algorithm:

Algorithm 1 (Brute Force Root Isolation)

Input $g \in \mathbb{Z}_p[X]$.

Output *A finite set* $\{(c_1, \alpha), \ldots, (c_k, \alpha)\} \subset \mathbb{Z} \times \mathbb{Z}$ *such that for* $j \in \{1, \ldots, k\}$*:*

1. $\alpha \geq 0$,
2. $v(c_j) = 0$,
3. *each* c_j *is* α*-close to exactly one root* $z \in \mathbb{Q}_p$ *with* $v(z) = 0$ *of* g.

Vice versa, each root $z \in \mathbb{Q}_p$ *with* $v(z) = 0$ *of* g *is* α*-close to* c_j *for exactly one* $j \in \{1, \ldots, k\}$.

Method

```
1  procedure isolatebf0(g)
2      α := v(discr(g))
3      R := ∅
4      for i ∈ {1, ..., p^(2α+1) − 1} do
5          if v(i) = 0 and v(g(i)) > 2α then
6              if not exists (i', α) ∈ R with v(i' − i) > α then
7                  R := R ∪ {(i, α)}
8      return R
9  end  □
```

Let us choose $p = 2$ and consider a call `isolatebf0`(g) with $g = X^2 - 1$. In Line 2 we obtain $\alpha = 2$. So in the for-loop in Line 4 we have i ranging from 1 to 31. The condition in Line 5 becomes true for 1 and 15, where the latter is α-close to -1. Later on, the condition in Line 5 becomes true also for 17 and 31, both of which are duplicate hits that do not pass the test in Line 6: $v(-1 - 17) > \alpha$ and $v(-15 - 31) > \alpha$.

In fact, 17 is α-close to 1, and 31 is α-close to -1. The following lemma generalizes this observation and thus clarifies the correctness of the test in Line 6:

Lemma 6 *Let* $\alpha \geq 0$*, and let* c*,* c'*,* $z \in \mathbb{Z}_p$*. Let* $v(c - z) > \alpha$*. Then* $v(c' - z) > \alpha$ *if and only if* $v(c' - c) > \alpha$.

PROOF. Let $v(c' - z) > \alpha$, and assume for a contradiction that $v(c' - c) \leq \alpha$. Then

$$v(c - z) = v\big(c' - z - (c' - c)\big) = \min\big\{v(c' - z), v(c' - c)\big\} \leq \alpha$$

contradicting $v(c - z) > \alpha$. The converse follows analogously. ■

It is noteworthy that in our little example above, we have $g' = 2X$ and thus $v\big(g'(1)\big) = 1$. This shows that the choice of α for the Hensel conditions is relevant even for exact zeros.

4.1.2. Smart Algorithm

We now turn to an alternative and much more sophisticated recursive algorithm, which exploits not only our Corollary 1 but the more general underlying Main Theorem 1:

Algorithm 2 (Root Isolation)

Input $g \in \mathbb{Z}_p[X]$.

Output *A finite set* $\{(c_1, \beta_1), \ldots, (c_k, \beta_k)\} \subset \mathbb{Z} \times \mathbb{Z}$ *such that for* $j \in \{1, \ldots, k\}$*:*

1. $\beta_j \geq 0$,
2. $v(c_j) = 0$,
3. *each* c_j *is* β_j*-close to exactly one root* $z \in \mathbb{Q}_p$ *with* $v(z) = 0$ *of* g.

Vice versa, each root $z \in \mathbb{Q}_p$ *with* $v(z) = 0$ *of* g *is* β_j*-close to* c_j *for exactly one* $j \in \{1, \ldots, k\}$.

Method

```
1  procedure isolate0(g)
2      α := v(discr(g))
3      R := ∅
4      for i ∈ {1, ..., p − 1} do
5          R := R ∪ isorefine(g, α, i, 0)
6      return R
7  end

8  procedure isorefine(g, α, i, β)
9      if v(g'(i)) ≤ β then
10         return isorefine1(g, α, i, β)
11     S := ∅
12     if β < α then
13         for k ∈ {0, ..., p − 1} do
14             S := S ∪ isorefine(g, α, i + kp^(β+1), β + 1)
15     return S
16 end

17 procedure isorefine1(g, α, i, β)
18     if not β < v(g(i)) then
19         return ∅
20     if 2β < v(g(i)) then
21         return {(i, β)}
22     S := ∅
23     if β < α then
24         for k ∈ {1, ..., p − 1} do
25             S := S ∪ isorefine1(g, α, i + kp^(β+1), β + 1)
26     return S
27 end  □
```

The basic idea is that for checking β-closeness it is sufficient to consider I_β. To catch the idea of the algorithm it is most helpful to understand the refinements in terms of the unique p-adic expansion of $x \in \mathbb{Z}_p$:

$$x = \sum_{k=0}^{\infty} x_k p^k =: 0.x_0 x_1 x_2 \ldots \quad \text{where} \quad x_i \in \{0, \ldots, p-1\}.$$

By the strict triangle inequality we have that $v(x)$ is the smallest index k such that $x_k \neq 0$. Accordingly, c_1 is β-close to c_2 iff $c_1 - c_2 \in I_\beta$ iff $(c_1 - c_2)_0 = \cdots = (c_1 - c_2)_\beta = 0$. Within the procedure `isorefine`, the argument i has a finite representation $0.i_0 \ldots i_\beta$, i.e., $i_k = 0$ for all $k > \beta$. Each recursive call in either Line 15 or 26 constructs one possible next digit $i_{\beta+1}$.

Corollary 2 *Let β, $\beta' \in \mathbb{Z}$ with $0 \leq \beta < \beta'$. Consider $0.x_0 \ldots x_\beta$, $0.x_0 \ldots x_{\beta'} \in \mathbb{Z}_p$.*

(i) Let $z \in \mathbb{Z}_p$. Then $0.x_0 \ldots x_\beta$ is β-close to z iff $0.x_0 \ldots x_{\beta'}$ is β-close to z.

(ii) For $g \in \mathbb{Z}_p[X]$ we have $v\big(g'(0.x_0 \ldots x_\beta)\big) \leq \beta$ iff $v\big(g'(0.x_0 \ldots x_{\beta'})\big) \leq \beta$.

PROOF. To start with, it is easy to see that $0.x_0 \ldots x_\beta$ is β-close to $0.x_0 \ldots x_{\beta'}$:

$$0.x_0 \ldots x_\beta - 0.x_0 \ldots x_{\beta'} = \sum_{k=0}^{\beta} x_k p^k - \sum_{k=0}^{\beta'} x_k p^k = -\sum_{k=\beta+1}^{\beta'} x_k p^k = -0.0 \ldots 0 x_{\beta+1} \ldots x_{\beta'}.$$

(i) This follows from the observation above and the fact that β-closeness is an equivalence relation.

(ii) By the observation above have $v(0.x_0 \ldots x_\beta - 0.x_0 \ldots x_{\beta'}) > \beta$. We can thus apply Theorem 1(i) to $0.x_0 \ldots x_\beta$, $0.x_0 \ldots x_{\beta'}$, and g'. ■

Algorithm 2 starts by recursively constructing in a a depth-first manner all representatives for $\mathbb{Z}_p/\beta$ where β is increased towards α. Concerning the test in Line 12 (and also later on in Line 23) notice that

$$\beta < \alpha \iff 2(\beta+1) \leq 2\alpha \iff p^{2(\beta+1)} < p^{2\alpha+1} \iff p^{2(\beta+1)} \leq p^{2\alpha+1} - 1.$$

As soon as in Line 9 within `isorefine` there is discovered that $v\big(g'(i)\big) \leq \beta$, the recursion continues in `isorefine1` instead of `isorefine`. At this point, we know on the one hand from Corollary 2(ii) that this condition is going to hold as well for all refinements of i. It thus need not be checked anymore. On the other hand, the validity of this condition allows us to possibly exploit our Main Theorem 1 in two ways:

1. If we discover in Line 18 within `isorefine1` that we have *not* $\beta < v\big(g(i)\big)$, then according to Theorem 1(ii) there cannot be any root of g with value 0 that is β-close to our considered i. By Corollary 2(i) the same holds for all refinements of i.

2. If we discover in Line 20 within `isorefine1` that we have even $2\beta < v\big(g(i)\big)$, then according to Theorem 1(iii) there exactly one root of g with value 0 that is β-close to i. Although we have possibly not yet reached α, there is no need for further refinement.

The reader might have noticed that we have not provided any results on the quality of the bound $\alpha = v\big(\mathrm{discr}(g)\big)$. In view of Algorithm 2 this is actually not relevant at all. The sole purpose of α here is to guarantee termination.

4.2. Refinement of the Isolation

The algorithms described in the previous section yield an analogue to a system of isolating intervals for the zeros of real univariate polynomials. As in the real case any such system can be refined to a system of arbitrarily good approximations of the roots:

Algorithm 3 (Root Refinement)

Input *1.* $g \in \mathbb{Z}_p[X]$,
2. $(c, \beta) \in \mathbb{Q} \times \mathbb{Z}$ *obtained by Algorithm 1 or Algorithm 2,*

3. $\beta' > \beta$.

Output $(c', \beta') \in \mathbb{Q} \times \mathbb{Z}$ *with* $v(c') = 0$ *such that* c' *is* β'*-close to* z.

Method

```
1  procedure refine0(g, (c, β), β')
2      if not β < v(g(c)) then
3          return ⊥
4      if β ≥ β' and 2β' < v(g(c)) then
5          return (c, β)
6      for k ∈ {0, ..., p − 1} do
7          w := refine0(g, (c + kp^{β+1}, β + 1), β')
8          if w ≠ ⊥ then
9              return w
10 end  □
```

5. Lifting

We are now going to combine our algorithms for roots with value zero that we have developed in the previous section with our results on possible values of zeros and corresponding normalizations from Section 2**.** and Section 3**.**, respectively.

We extend our notion of β-closeness, which we have introduced in Section 4**.**1**.**, to numbers of arbitrary value: Let $\gamma \in \mathbb{Z}$. Then d_1, $d_2 \in \mathbb{Q}_p$ are γ*-close* if $v(d_1 - d_2) > \gamma$. In terms of the p-adic absolute value, γ-closeness states that $|d_1 - d_2| < p^{-\gamma}$. Notice that γ-closeness is an equivalence relation but not a congruence relation. For convenience we agree that every number is ∞*-close* to itself.

Algorithm 4 (Root Isolation)

Input $f \in \mathbb{Q}[X]$ *with* $f \neq 0$.

Output *A finite set* $\{(d_1, \gamma_1), \ldots, (d_k, \gamma_k)\} \subset \mathbb{Q} \times \mathbb{Z}$ *such that for each* $j \in \{1, \ldots, k\}$ *we have that* d_j *is* γ_j*-close to exactly one root* $z \in \mathbb{Q}_p$ *of* f*. Vice versa, each root* $z \in \mathbb{Q}_p$ *of* f *is* γ_j*-close to* d_j *for exactly one* $j \in \{1, \ldots, k\}$.

Method

```
1  procedure isolate(f)
2      f := the square-free part of f
3      if f(0) = 0 then
4          S' := {(0, ∞)}
5          f := f/X
6      else
7          S' := ∅
8      if f ∈ ℚ then
9          return S'
10     C := a set containing one critical balancable pair for each critical value for f
11     for (a_j X^j, a_k X^k) ∈ C do
12         κ := (v(a_j) − v(a_k))/(k − j)
13         f_{j,k} := the j-k-normalization of f
14         S := isolate0(f_{j,k})
15         for (c, β) ∈ S do
16             S' := S' ∪ {(p^κ c, β + κ)}
```

17 **return** S'
18 **end**

Remark *A corresponding brute-force variant can be obtained by replacing the call to* `isolate0` *in Line 14 by a corresponding call to* `isolatebf0`*. In the sequel, we are going to refer to this variant as* `isolatebf`*.* □

Let us consider $S' = \texttt{isolate}(f)$ and ignore the trivial cases which lead to returning in Line 9. In view of our previous results it is straightforward that there is a one-to-one correspondence between the roots of f and the pairs in S'. It remains to verify the assertions on γ-closeness made in the specification of the algorithm.

Let $(d, \gamma) \in S'$. Then there is (c, β) with $v(c) = 0$ such that c is β-close to a root z of some $f_{j,k}$, i.e, $v(c - z) > \beta$. It is easy to see that $v(d) = \gamma - \beta$. Accordingly, $d = p^{\gamma-\beta}c$, and by Lemma 2 we have that $p^{\gamma-\beta}z$ is a root of f. It follows that

$$v(d - p^{\gamma-\beta}z) = v(p^{\gamma-\beta}c - p^{\gamma-\beta}z) = \gamma - \beta + v(c - z) > \gamma.$$

Let $p^{\gamma-\beta}z'$ be another root of f with value $\gamma-\beta$, and assume for a contradiction that $v(d-p^{\gamma-\beta}z') > \gamma$. It follows that

$$\gamma < v(d - p^{\gamma-\beta}z') = v(p^{\gamma-\beta}c - p^{\gamma-\beta}z') = \gamma - \beta + v(c - z')$$

and thus $v(c - z') > \beta$, a contradiction to the specification of the Algorithm `isoloate`. Next, let $p^{\delta}z''$ be another root of f with $v(p^{\delta}z'') = \delta > \gamma - \beta$. Then

$$v(d - p^{\delta}z'') = v(p^{\gamma-\beta}c - p^{\delta}z'') = v(p^{\gamma-\beta}c) = \gamma - \beta \leq \gamma.$$

Let finally $p^{\varepsilon}z'''$ be another root of f with $v(p^{\varepsilon}z''') = \varepsilon < \gamma - \beta$. Then

$$v(d - p^{\varepsilon}z''') = v(p^{\gamma-\beta}c - p^{\varepsilon}z''') = v(p^{\varepsilon}z''') = \varepsilon < \gamma.$$

We have shown that each d with $(d, \gamma) \in S'$ is γ-close to exactly one root of f. By the above-mentioned one-to-one correspondence it follows that vice versa each root of f is γ-close to exactly one d with $(d, \gamma) \in S'$.

Algorithm 5 (Root Refinement)

Input *1.* $f \in \mathbb{Q}[X]$,
2. $(d, \gamma) \in \mathbb{Q} \times \mathbb{Z}$*, obtained by Algorithm 4,*
3. $\gamma' > \gamma$.

Output $(d', \gamma') \in \mathbb{Q} \times \mathbb{Z}$ *with* $v(d') = v(d)$ *such that* d' *is* γ*-close to* z.

Method

1 **procedure** `refine`$(f, (d, \gamma), \gamma')$
2 $(a_j, X^j, a_k X^k) :=$ a critical balancable pair for f with critical value $v(d)$
3 $f_{j,k} :=$ the j-k-normalization of f
4 $c := d/p^{v(d)}$
5 $\beta := \gamma - v(d)$
6 $\beta' := \gamma' - v(d)$
7 $(c', \beta') :=$ `refine0`$(f_{j,k}, (c, \beta), \beta')$
8 $d' := p^{v(d)}c'$
9 **return** (d', γ')
10 **end** □

The existence of the pair in Line 2 is guaranteed by Lemma 1. Denote $z_0 = p^{-v(d)}z$; then we have by Lemma 2(i) that this z_0 is a root of $f_{j,k}$. From the γ-closeness $v(d-z) > \gamma$ it follows that c computed in Line 4 is β-close to z_0, where β is computed in Line 5:

$$v(c - z_0) = v\big(p^{-v(d)}d - p^{-v(d)}z\big) = v(d-z) - v(d) > \gamma - v(d) = \beta.$$

Sine $\gamma' > \gamma$, we know in Line 6 that $\beta' > \beta$. By the specification of Algorithm 3 (`refine0`) we obtain in Line 7 some c' that is β'-close to z_0. So we obtain for our d' of Line 8 that

$$v(d' - z) = v\big(p^{v(d)}c' - p^{v(d)}z_0\big) = v(c' - z_0) + v(d) > \beta' + v(d) = \gamma'.$$

6. The Distance of Roots

The output of Algorithm 4 (`isolate`) provides most precise information on the distance between all p-adic roots of the input polynomial f:

Lemma 7 (Exact Distance of Roots) *Let $f \in \mathbb{Q}[X]$, and let (d,γ), $(d',\gamma') \in$ `isolate`(f). Let $z \in \mathbb{Q}_p$ and $z' \in \mathbb{Q}_p$ be the roots of f that are approximated by (d,γ) and (d',γ'), respectively. Then $v(z-z') = v(d-d')$.*

PROOF. By the specification of Algorithm 4 (`isolate`), we know $v(d) = v(z)$ and $v(d-z) > \gamma$ and, correspondingly, $v(d') = v(z')$ and $v(d'-z') > \gamma'$. Let wlog. $v(d) \le v(d')$ and thus $v(z) \le v(z')$.

Consider the case $v(d) = v(d')$ and thus $v(z) = v(z')$. Let wlog. $\gamma \le \gamma'$, and assume for a contradiction that $v(d-d') > \gamma$. Then

$$v(d-z') = v\big((d-d') + (d'-z')\big) \ge \min\big\{v(d-d'), v(d'-z')\big\} > \gamma.$$

This contradicts the specification of Algorithm 4, by which z is the only root of f that is γ-close to d. So we now know that $v(d-d') \le \gamma$. This implies in turn

$$v(z-z') = v\big((d'-z') + (d-d') - (d-z)\big) = \min\big\{v(d'-z'), v(d-d'), v(d-z)\big\} = v(d-d').$$

Consider now the complementary case that $v(d) < v(d')$ and thus $v(z) < v(z')$. Then it follows as well that

$$v(z-z') = \min\big\{v(z), v(z')\big\} = v(z) = v(d) = \min\big\{v(d), v(d')\big\} = v(d-d'). \quad \blacksquare$$

7. The Maximal Number of Roots

As an easy consequence of our root counting algorithm we get a universal upper bound for the maximal number of p-adic zeros of polynomials in $\mathbb{Q}_p[X]$ that does *not* depend on the degree:

Corollary 3 (Number of Different Roots) *Let $f = \sum_{i=1}^{n} a_i X^i \in \mathbb{Q}_p[X]$. Let $m \le n+1$ be the number of non-zero monomials in f. Let α be the maximum of all $v\big(\mathrm{discr}(f_{jk})\big)$ for all critical balancable pairs (a_jX^j, a_kX^k) of f. Then*

$$\big|\{\, z \in \mathbb{Q}_p \mid f(z) = 0 \,\}\big| \le \frac{m(m-1)}{2}(p^{\alpha+1} - p^{\alpha}).$$

PROOF. The number of balancable pairs of f is bounded by $\frac{m(m-1)}{2}$. For fixed critical balancable pair (a_jX^j, a_kX^k) the number of p-adic zeros of f_{jk} is by our Algorithm 1 bounded by the number of residue classes i of $\mathbb{Z}$ modulo $p^{\alpha+1}$ with the additional property that $v(i) = 0$. Since the number of residue classes j modulo $p^{\alpha+1}$ with $v(j) > 0$ is p^{α}, it follows that the number of residue classes i of $\mathbb{Z}$ modulo $p^{\alpha+1}$ with $v(i) = 0$ is exactly $p^{\alpha+1} - p^{\alpha}$. $\blacksquare$

This result is reminiscent of corresponding bounds for real zeros within the *Fewnomial* framework by Khovanskii [13].

A corresponding theorem for fixed prime p was proved in [14] and generalized to the multivariate case in [18]. Note that our bound is a more general result, because it is *uniform* in p. In particular, we have shown that for fixed polynomial f the number of roots is polynomial in p.

8. Uniformity

There are two types of uniformity of our algorithms to consider:

1. uniformity in the prime p for fixed polynomial f,
2. uniformity in the polynomial f for fixed prime p.

As for the first type of uniformity the following immediate consequence of our algorithms is essentially well-known:

Corollary 4 *For fixed square-free polynomial $f \in \mathbb{Q}[X]$ let P_f be the set of all primes p such that all coefficients of f as well as the discriminant of f have value zero with respect to v_p. Then P_f consists of almost all primes, and for all $p \in P_f$ the following hold:*

(i) For each $i \in \{1, \ldots, p-1\}$ with $v\big(f(i)\big) > 0$ there is exactly one zero z of f with $v(i-z) > 0$. For each $i \in \{1, \ldots, p-1\}$ with $v\big(f(i)\big) = 0$ there is no zero z of f with $v(i-z) > 0$.

(ii) The zeros of f are in one-to-one correspondence with the integers $i \in \{1, \ldots, p-1\}$ with $v\big(f(i)\big) > 0$.

(iii) The following direct analogue of Sturm's Theorem for real zeros holds: The number of p-adic zeros of f equals

$$\big|\{\, i \in \{1, \ldots, p-1\} \mid v(f(i)) > 0 \,\}\big|. \quad \square$$

For the second type of uniformity we fix the prime p and vary the coefficients of the polynomial f obtaining a new polynomial f^* in such a way that our results on the p-adic zeros of f do not change. When stepping from f to f^* we have to ensure the following:

1. $(a_j^* X^j, a_k^* X^k)$ is a critical balancable pair of f^* if and only if $(a_j X^j, a_k X^k)$ is a critical balancable pair of f.
2. For each j-k-normalization the value of the discriminant does not change:

$$v\big(\mathrm{discr}(f_{j,k}^*)\big) = v\big(\mathrm{discr}(f_{j,k})\big).$$

3. Let $\alpha = v\big(\mathrm{discr}(f_{j,k})\big) = v\big(\mathrm{discr}(f_{j,k}^*)\big)$. For all $i \in \{1, \ldots, p^{2\alpha} - 1\}$ we must have

$$2\alpha < v\big(f_{j,k}^*(i)\big) \quad \text{iff} \quad 2\alpha < v\big(f_{j,k}(i)\big).$$

In order to satisfy all these conditions we consider the equivalence relations $\sim_\beta$ for natural numbers β introduced by the second author in [23]. They are defined as follows:

$$a^* \sim_\beta a \quad \text{iff} \quad v(a^*) = v(a) \quad \text{and} \quad v(a^* - a) > \beta + v(a).$$

Note that equivalence is preserved when both sides are multiplied by the same non-zero p-adic number.

Let α be the maximum of all $v\big(\mathrm{discr}(f_{jk})\big)$ for all critical balancable pairs $(a_j X^j, a_k X^k)$ of f. If we fix the equivalence classes of all coefficients of f and of the discriminant of f modulo $\sim_{2\alpha}$, then all the conditions required above are indeed satisfied.

9. Complexity

We want to compute upper bounds for the asymptotic complexity of our algorithms for locating all p-adic zeros of a univariate rational polynomial $f = \sum_{i=0}^{n} a_i X^i \in \mathbb{Q}[X]$ of degree n. For this purpose we review the single steps of the algorithms:

Possible Values of Zeros and Normalization The computation of each critical balancable pair, its critical value and its normalization involves only arithmetical operations on the coefficients of f and on p-powers p^γ, where γ is a linear combination of values of coefficients of f. All these p-powers are divisors of powers of coefficients of f with exponents bounded by the degree of f. So if the input polynomial f is given in dense representation, then the operations are performed in polynomial time. Moreover the total number of critical balancable pairs is bounded by $\frac{n(n-1)}{2}$. So the computation of *all* normalizations f_{jk} of f is performed in polynomial time.

Roots of Value Zero Let us first consider the Brute Force Algorithm 1: For each given normalization $g := f_{jk}$ we compute the zeros of g of value zero by testing representatives $i \in \{1, \ldots, p^{2\alpha+1} - 1\}$, where $\alpha = v\big(\mathrm{discr}(g)\big)$. Since $\mathrm{discr}(g)$ is a polynomial in the coefficients of g, and p^α is a divisor of $\mathrm{discr}(g)$, it follows that $p^{2\alpha+1} = p \cdot (p^\alpha)^2$ is computed in polynomial time from the coefficients of g, and hence also in polynomial time from the coefficients of f. Moreover α is bounded by the binary size of $\mathrm{discr}(g)$. Nevertheless, the number of representatives i to test is exponential in the input size. Each single test involves an evaluation of $v\big(g'(i)\big)$ and of $v\big(g(i)\big)$ and hence is performed in polynomial time. This yields exponential time in the bit size of g. Note that $\{1, \ldots, p^{2\alpha} - 1\}$ can be straightforwardly enumerated such that the algorithm requires only polynomial space.

We now turn to the more sophisticated isolation Algorithm 2. It recursively constructs a tree T of height $2\alpha + 1$ and out-degree p. This is done in a depth-first manner by storing at each time only a single branch of T. The space required for this is polynomial in the size of g, i, and α. It is hence polynomial in the size of the input polynomial f. In particular the whole testing procedure is performed in space polynomial in f. In particular, the space is independent of p. This independence is due to the same phenomenon that yields the uniformity in p: for p that do not divide all coefficients of f and all discriminants $\mathrm{discr}(f_{jk})$ all p-adic values required during the computations are zero.

Lifting The overall number of complex roots and thus that of p-adic roots is bounded by the degree n of f. Hence, in analogy to the discussion of the normalization above, the lifting step for all roots of all $f_{j,k}$ is in polynomial time.

Consequently the complete localization of all p-adic zeros of f can be performed in space polynomial in the bit size of f represented as dense polynomial.

10. Implementation and Computation Examples

The methods described throughout this paper are implemented in a REDUCE [1] package PROOTS[2]. All examples discussed throughout this section have been computed with this package using 128 MB RAM on a 2 GHz Pentium 4 machine running Linux.

10.1. Roots of Unity

We consider polynomials $f^{(n)} = X^n - 1$ for $n \in \mathbb{N}$. These polynomials are square-free. For any $n \in \mathbb{N}$ the pair $(-1, X^n)$ is balancable with balancing value 0, and the 0-n-normalization $f_{0,n}^{(n)}$ is equal to the original polynomial $f^{(n)}$.

[1] Information on the computer algebra system REDUCE can be fount at `http://www.zib.de/Symbolik/reduce/`.
[2] The package is available for download at `http://www.fmi.uni-passau.de/~reduce/proots/`.

We first consider a series where we choose p and set $n = p-1$. There are then exactly n different zeros of $f^{(n)}$. For all $n \in \mathbb{N}$ we obtain $\alpha = v\big(\mathrm{discr}(f^{(n)})\big) = 0$. There are thus only residue classes modulo I_0 to be considered. As a consequence we cannot expect the `isolate` variant of our Algorithm 4 to perform better than the brute-force variant `isolatebf`.

					`isolatebf`	`isolate`
n	p	α	$p^{2\alpha+1}-1$	result	time (ms)	time (ms)
1	2	0	1	$\{(1,0)\}$	< 10	< 10
102	103	0	102	$\{(1,0),\ldots,(102,0)\}$	10	< 10
1008	1009	0	1008	$\{(1,0),\ldots,(1008,0)\}$	770	1310
2002	2003	0	2002	$\{(1,0),\ldots,(2002,0)\}$	5400	9690
3000	3001	0	3000	$\{(1,0),\ldots,(3000,0)\}$	16000	28430
4000	4001	0	4000	$\{(1,0),\ldots,(4000,0)\}$	38600	69990

In fact, we observe that the brute-force variant is approximately twice as fast. That is because it does not plug the candidate numbers i into the derivative g' but only into g.

We next consider a slightly different series, where we set $n = p$. Except for $n = p = 2$ there is only one root then, which is 1. In this series we obtain nonzero discriminant values $\alpha = v\big(\mathrm{discr}(f^{(n)})\big) = n$. We can now see that the `isolate` variant of Algorithm 4 is extremely superior over the brute-force variant `isolatebf`:

				`isolatebf`		`isolate`	
n	p	α	$p^{2\alpha+1}-1$	result	time (ms)	result	time (ms)
2	2	2	31	$\{(1,2),(15,2)\}$	< 10	$\{(1,1),(3,1)\}$	< 10
3	3	3	2186	$\{(1,3)\}$	< 10	$\{(1,1)\}$	< 10
5	5	5	48828124	$\{(1,5)\}$	107180	$\{(1,1)\}$	< 10
7	7	7	4747561509942	–	–	$\{(1,1)\}$	< 10
103	103	103	$103^{207}-1$	–	–	$\{(1,1)\}$	1200
211	211	211	$211^{423}-1$	–	–	$\{(1,1)\}$	20560
307	307	307	$307^{615}-1$	–	–	$\{(1,1)\}$	87840
401	401	401	$401^{803}-1$	–	–	$\{(1,1)\}$	257030

A rough guess for the running time of `isolatebf` for $n = 7$ would be, extrapolating from $n = 5$, $(4747561509942/48828124) \cdot 107180$ ms, which is about 120 days.

10.2. Non-Radical Roots

The polynomial $X^5 - 4X + 2$ has Galois group S_5. Hence its zeros cannot be expressed by radicals. For $p = 2$, the only balancable pair is $(2, -4X)$ with balancing value -1, but this is not critical. Let now $p > 2$. All the pairs $(2, -4X)$, $(2, X^5)$, $(-4X, X^5)$ are critical with critical value 0. It follows that our polynomial $X^5 - 4X + 2$ is then equal to its 0-1-normalization, and we obtain $\alpha = v\big(\mathrm{discr}(X^5 - 4X + 2)\big) = 0$. Again, `isolate` has no advantage over `isolatebf`, and again we observe the factor 2 in speed:

p	α	$p^{2\alpha+1}-1$	result	`isolatebf` time (ms)	`isolate` time (ms)
2	–	–	$\emptyset$	< 10	< 10
3	0	2	$\emptyset$	< 10	< 10
5	0	4	$\{(4,0)\}$	< 10	< 10
7	0	6	$\emptyset$	< 10	< 10
11	0	10	$\{(9,0)\}$	< 10	< 10
13	0	12	$\{(2,0),(5,0)\}$	< 10	< 10
103	0	102	$\emptyset$	< 10	< 10
1009	0	1008	$\{(43,0),(577,0)\}$	< 10	< 10
10007	0	10006	$\{(6872,0)\}$	20	40
100003	0	100002	$\{(2222,0)\}$	240	420
1000003	0	1000002	$\{(101947,0),(140688,0),(424568,0)\}$	2510	4640
10000019	0	10000018	$\{(3465137,0),(5835443,0)\}$	26060	47760

10.3. Illustrating Example Revisited

We revisit our example polynomial $f = 81X^4 - 6X + 5$ from Section 2. and Section 3.. For $p = 2$ we have to consider $f_{0,4} = f$, for $p = 3$ we have $f_{0,1} = X^4 - 2X + 5$, and for $p = 5$ we have $f_{0,1} = 10125X^4 - 6X + 1$ and $f_{1,4} = f$.

p	α	$p^{2\alpha+1}-1$	`isolatebf` result	`isolatebf` time (ms)	`isolate` result	`isolate` time (ms)
2	4	511	$\{(233,4)\}$	< 10	$\{(1,1)\}$	< 10
3	0	2	$\emptyset$	< 10	$\emptyset$	< 10
5	9, 0	19073486328124, 4	–	–	$\{(5,1),(1,0)\}$	< 10

10.4. Refinement

We consider the root $(1,1)$ for $p = 2$ in the previous example. We can refine this within 60 ms to

$$(4686275122178049460444579759902607416056735440119802796400 41, 100),$$

within 4580 ms to precision 500, and within 35080 ms to precision 1000.

11. Conclusions

We have presented algorithmic methods for isolating all p-adic zeros of a given univariate polynomial. The isolating balls can be refined to any desired precision. This makes root counting and the determination of the p-adic distances between all roots straightforward. We have thus a perfect explicit p-adic analogue to Sturm's results for the reals. We have analyzed various aspects of uniformity and complexity of our methods. All our methods discussed here are implemented in a REDUCE package PROOTS which is freely available. We have demonstrated the application range of this package by means of various computation examples and benchmark series.

References

[1] Alkaviadis G. Akritas. *Elements of Computer Algebra*. A Wiley-Interscience publication. John Wiley & Sons, New York, 1988.

[2] James Ax and Simon Kochen. Diophantine problems over local fields. *American Journal of Mathematics*, 87:605–648, 1965. Parts I and II.

[3] James Ax and Simon Kochen. Diophantine problems over local fields. *Annals of Mathematics*, 83:437–456, 1966. Part III.

[4] Saugata Basu, Richard Pollack, and Marie-Francoise Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer, 2003.

[5] Bob F. Caviness and Jeremy R. Johnson, editors. *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Texts and Monographs in Symbolic Computation, Linz, 1993. Springer, Wien, New York, 1998.

[6] Paul J. Cohen. Decision procedures for real and p-adic fields. *Communications in Pure and Applied Logic*, 25:213–231, 1969.

[7] Jan Denef. P-adic semi-algebraic sets and cell decomposition. *Journal für die reine und angewandte Mathematik*, 369:154–166, 1986.

[8] Andreas Dolzmann and Thomas Sturm. Redlog: Computer algebra meets computer logic. *ACM SIGSAM Bulletin*, 31(2):2–9, June 1997.

[9] Andreas Dolzmann and Thomas Sturm. P-adic constraint solving. In Sam Dooley, editor, *Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation (ISSAC 99), Vancouver, BC*, pages 151–158. ACM Press, New York, NY, July 1999.

[10] Yuri L. Ershov. On elementary theories of local fields. *Algebra i Logika Sem.*, 4(2):5–30, 1965.

[11] Yuri L. Ershov. On elementary theories of maximal valued fields III. *Algebra i Logika Sem.*, 6:31–73, 1967.

[12] Kurt Hensel. *Theorie der algebraischen Zahlen*. Teubner, Leipzig, 1908.

[13] Askold G. Khovanskii. *Fewnomials*, volume 88 of *Translations of Mathematical Monographs*. AMS, Providence, RI, 1991.

[14] H. W. Lenstra. On the factorization of lacunary polynomials. In *Number Theory in Progress*, volume I, pages 277–291. Walter de Gruyter, 1999.

[15] Angus Macintyre. On definable subsets of p-adic fields. *Journal of Symbolic Logic*, 41(3):605–610, September 1976.

[16] A. Nerode. A decision method for p-adic integral zeros of Diophantine equations. *Bulletin of the American Mathematical Society*, 69:513–517, 1963.

[17] Peter N. Panayi. *Computation of Leopoldt's p-adic Regulator*. PhD thesis, University of East Anglia, 1995.

[18] Maurice Rojas. Finiteness for arithmetic fewnomial systems. In E. Green, S. Hosten, R. Laubenbacher, and V. Powers, editors, *AMS-IMS-SIAM Joint Summer Research Conference Proceedings of "Symbolic Computation: Solving Equations in Algebra, Geometry, and Engineering" (June 11–15, 2000, Mount Holyoke College)*, volume 286 of *Contemporary Mathematics*, pages 107–114. AMS Press, 2001.

[19] Jacques Charles François Sturm. Mémoire sur la résolution des équations numériques. In *Mémoires présentés par divers Savants étrangers à l'Académie royale des sciences, section Sc. math. phys.*, volume 6, pages 273–318, 1835.

[20] Thomas Sturm. Linear problems in valued fields. *Journal of Symbolic Computation*, 30(2):207–219, August 2000.

[21] Volker Weispfenning. *Elementary Theories of Valued Fields*. Ph.D. thesis, Universität Heidelberg, 1971.

[22] Volker Weispfenning. On the elementary theory of Hensel fields. *Annals of Mathematical Logic*, 10:59–93, 1976.

[23] Volker Weispfenning. Quantifier elimination and decision procedures for valued fields. In G. H. Müller and M. M. Richter, editors, *Models and Sets (Aachen, 1983)*, volume 1103 of *Lecture Notes in Mathematics*, pages 419–472. Springer-Verlag, Berlin, Heidelberg, 1984.

Thomas Sturm and Volker Weispfenning
Fakultät für Mathematik und Informatik
Universität Passau
Innstraße 33
94032 Passau
Germany
sturm@uni-passau.de,weispfen@algebra.uni-passau.de