

Factoring Quantum-Plane Skew Polynomials over $\mathbb{Q}(\omega)(t)$

Mark Giesbrecht

Cheriton School of Computer Science, Faculty of Mathematics, University of Waterloo, Canada
mwg@uwaterloo.ca

Abstract. We study algorithms for exact factorization in the quantum plane of (dilation) skew polynomials over a function field of a cyclotomic field:

$$R = K(t)[x; \sigma], \quad K = \mathbb{Q}(\omega), \quad \sigma(t) = \omega t,$$

where $\omega \in \mathbb{C}$ is a primitive m -th root of unity. As is standard, we work through central elements: the centre $\mathbb{C}$ of R is the univariate PID $\mathbb{C} = K(t^m)[x^m]$. For a monic skew polynomial $f \in K[t][x; \sigma]$, the first stage of the algorithm stays in characteristic zero: it computes the monic bound, or minimal central left multiple $\varphi \in \mathbb{C}$ of f , factors φ in $\mathbb{C}$, and uses right gcd computations to split off all pieces with reducible bounds. For the remaining factors with irreducible non-trivial bounds we first apply degree-bound and norm tests for irreducibility, and then propose a two-level modular approach: specialize the single central parameter $\mathcal{T}$ to good algebraic values, study the resulting cyclic algebras over number fields, and reduce further at good inert primes so that fast finite-field skew-factorization algorithms apply. Every factor returned by this procedure is certified by exact division. Over $K(t)$ the irreducible-bound stage is therefore a sound certifying search, but neither completeness nor a complexity estimate is known.

We next consider the effect of extending the field of constants from K to $\bar{\mathbb{Q}}$, that is, of factoring over $\bar{\mathbb{Q}}(t)[x; \sigma]$. This changes the factorization problem, since irreducibility over $K(t)$ and over $\bar{\mathbb{Q}}(t)$ need not agree and the output may lie over a larger constant field. In the algebraically closed-constants setting, Tsen's theorem forces the irreducible-bound algebra to split, which yields a complete factorization algorithm in the standard exact algebraic model based on finite extensions and root adjunction. We then discuss the corresponding picture in the Blum–Shub–Smale (BSS) model over $\mathbb{C}$: bare BSS decides reducibility, while exact factor output requires a univariate root-choice primitive.

1 Introduction

This paper studies factorization of skew polynomials over a function field over a cyclotomic field with a natural dilation automorphism. This is a natural characteristic-zero analogue of the Ore polynomial setting over $\mathbb{F}_q(t)$ that underlies earlier skew factorization algorithms. In the root-of-unity quantum-plane case, the large univariate centre makes a “bound first” approach especially attractive.

Let $K = \mathbb{Q}(\omega)$, where $\omega \in \mathbb{C}$ is a primitive m -th root of unity, and let

$$R = K(t)[x; \sigma], \quad xa = \sigma(a)x \ (a \in K(t)), \quad \sigma(t) = \omega t.$$

Because σ has finite order m , its fixed field is $K(t^m)$ and the centre of R is the commutative principal ideal domain

$$C = K(t^m)[x^m] = K(\mathcal{T})[X], \quad \mathcal{T} = t^m, \quad X = x^m.$$

Thus the centre is a polynomial ring in the single central variable X over the rational function field $K(\mathcal{T})$. For our algorithms we take as input a monic polynomial

$$f \in K[t][x; \sigma] \subset R, \quad \text{and seek a factorization } f = f_1 f_2 \dots f_k$$

into irreducibles $f_1, \dots, f_k \in R$. Such factorizations need not be unique, though the Jordan–Hölder theory fixes the multiset of similarity classes and hence the degree sequence of irreducible factors. Denominators in t may necessarily occur in the factors even when the input has coefficients in $K[t]$.

Following [Jacobson \(1943\)](#), we define the **bound** of f as the unique monic $\varphi \in C$ of minimal degree in x , or equivalently minimal degree in X amongst central polynomials after the normalization above, such that $\varphi = uf \in C$ for some $u \in R$. In other words, $\varphi \in C$ is the monic central left multiple of f of minimal degree. Such a bound exists, is unique, and can be computed by linear algebra; more generally, the algorithms of [Gómez-Torrecillas et al. \(2019\)](#) compute bounds for Ore polynomials.

The overall algorithmic approach initially follows that of [Giesbrecht \(1998\)](#) and subsequent algorithms. First we compute a monic bound φ . If φ factors in C , then right gcd computations split f into a rough factorization whose pieces have irreducible central bounds; the multiplicities of central factors are handled recursively. The genuinely difficult case is therefore a factor whose bound is central and irreducible. This is the same point at which the general theory over $\mathbb{F}_q(t)$ passes from commutative central factorization to simple algebras and zero divisors ([Giesbrecht and Zhang, 2003](#); [Gómez-Torrecillas et al., 2015, 2019](#)). The same obstruction appears here over $K(t)$.

Our proposed method for this case is a two-level modular approach. We proceed in characteristic zero long enough to exploit the centre, the reduced norm, and the cyclic-algebra structure attached to an irreducible bound. We then specialize the single central parameter $\mathcal{T} = t^m$ to a good algebraic value a , so that the problem descends to a cyclic algebra over a number field. Finally, we reduce further at good inert primes, factor the resulting finite-field images using the algorithms of [Giesbrecht \(1998\)](#) or successors, such as [Caruso and Le Borgne \(2017\)](#), and lift certified factors back first to the specialized number field and then to $K(t)$.

Our contribution over $K(t)$ is a complete central decomposition algorithm, two inexpensive irreducibility filters, and a two-level certifying modular search for the remaining irreducible-bound case. The search may return failure, but every factor it returns is verified by exact division in the original ring.

If we allow ourselves to enlarge the constants from K to $\overline{\mathbb{Q}}$, then we obtain stronger results, albeit for a different factorization problem whose output may be much larger. In this setting over an algebraically closed field, the irreducible-bound algebra splits, by Tsen’s theorem, and full factorization becomes decidable in the standard exact algebraic model based on finite extensions and root adjunction. We make no efficiency claims here. We also discuss the corresponding factorization problem in the Blum–Shub–Smale

(BSS) model over $\mathbb{C}$, more precisely in BSS augmented with a univariate root-choice primitive.

Section 2 surveys previous relevant work. Sections 3–8 develop our proposed algorithm over $K(t)$. Section 9 records complexity and practical remarks, Section 10 looks at the history and related literature, while Section 11 identifies the parts that are currently algorithmic over $K(t)$ and isolates the remaining obstacles to decidability and efficiency. Section 12 describes our prototype implementation in SageMath. In Section 13 proves decidability over $\bar{\mathbb{Q}}(t)$ and explains the corresponding BSS interpretation.

2 Background and previous work

Ore (1933) introduced skew polynomial rings and their basic algebraic structure. Over finite coefficient fields, Giesbrecht (1998) gave the first polynomial-time factorization algorithm for $\mathbb{F}_q[x; \sigma]$, and Caruso and Le Borgne (2017) later developed algorithms with improved complexity and a good implementation. Over $\mathbb{F}_q(t)$, Giesbrecht and Zhang (2003) established the centre/eigenring viewpoint that underlies the present paper, although the general irreducible-bound case requires subroutines whose general availability was overstated. The obstruction is identified sharply in Gómez-Torrecillas et al. (2015): once the bound is central and irreducible, one has to find zero divisors in simple algebras over finite extensions of the centre. The bound-computation side of the method is developed further in Gómez-Torrecillas et al. (2019).

In the present paper we consider the univariate quantum plane over a rational function field at a root of unity. This is exactly the setting where x^m and t^m are central, so the centre is a univariate polynomial ring over $K(\mathcal{T})$. Low-degree factorization in quantum planes was studied by Coulibaly and Price (2006), who give irreducibility criteria for quadratics and described prime elements in the root-of-unity case. We are not aware of a general factorization algorithm in the root-of-unity quantum-plane setting considered here.

3 Quantum planes, bounds, and factor degrees

We begin with the structural results that govern every monic element $f \in R$ whose monic bound is an irreducible central polynomial $\pi \in C$.

The standard structure theory gives the degree pattern in this irreducible-bound case. See Giesbrecht and Zhang (2003) and Gómez-Torrecillas et al. (2019).

Fact 3.1. *Assume that the monic bound of $f \in R$ is an irreducible polynomial $\pi \in C$ with $\pi \neq X = x^m$. Set $s = \deg_X(\pi)$, $F = C/(\pi)$, and $A = R/R\pi$. Since π is central, $R\pi = \pi R$ is a two-sided ideal. The algebra A is a central simple F -algebra of degree m . Write $A \cong M_r(D)$, with D a central division F -algebra of degree d ; thus $rd = m$. Since any nonconstant right factor of f has monic bound dividing π , and π is irreducible, every such factor also has bound π . Since $\deg_x \pi = ms$, every irreducible factor of f in R has degree $e = sd$ in x . In particular, every complete factorization of f has the form*

$$f = h_1 h_2 \cdots h_\ell, \quad \deg_x h_i = sd \quad (1 \leq i \leq \ell), \quad \ell = (\deg_x f)/(sd),$$

and the only possible irreducible factors are of degree sd , with $d \mid m$.

Note that the exceptional central prime $\mathcal{X} = x^m$ is elementary: x is a normal irreducible element, and if x is a right factor of f , then one first strips the maximal right power of x .

The mechanism behind Fact 3.1 is as follows. The field $F = \mathbb{C}/(\pi)$ contains the image of $\mathcal{X}$, and the hypothesis $\pi \neq \mathcal{X}$ ensures that this image is nonzero and hence invertible. The base change

$$L_F := K(t) \otimes_{K(\mathcal{T})} F$$

is a cyclic étale F -algebra of degree m , with the induced order- m action of σ . Then

$$A \cong (L_F/F, \sigma, \mathcal{X})$$

is the corresponding cyclic crossed product, with relation $x^m = \mathcal{X}$. This is a central simple F -algebra of degree m . The factor-degree statement is then the usual module-theoretic statement. Since $R\pi \subseteq Rf$, the cyclic module $M_f := R/Rf$ is a left A -module. If S is a simple left A -module, then $M_f \cong S^\ell$, $\ell = \frac{\deg_x f}{sd}$, and $\text{End}_A(M_f) \cong M_\ell(D^{\text{op}})$. Thus irreducible right factors correspond to simple A -subquotients of M_f . Since such a simple module has F -dimension md , the corresponding skew factor has x -degree sd .

The following inexpensive irreducibility criterion is an immediate consequence.

Corollary 3.2. *Let $f \in R$ be monic and assume that its monic bound is an irreducible polynomial $\pi \in \mathbb{C}$ with $\pi \neq \mathcal{X}$. If $\deg_x(\pi) = m \deg_x(f)$, then f is irreducible.*

4 Central decomposition

We first separate the part of the input that is already controlled by the centre from the genuinely irreducible-bound part. The normal factor x is handled explicitly before the central recursive step.

Algorithm 4.1 Central decomposition stage

Input: a monic bounded polynomial $f \in R$.

Output: a rough product decomposition $f = g_1 g_2 \cdots g_\ell$, in which each g_j is either the irreducible factor x or is monic with irreducible monic bound $\pi_j \in \mathbb{C}$, $\pi_j \neq \mathcal{X}$.

- (1) Strip the maximal right power of x : write $f = f_0 x^\varepsilon$, with $\varepsilon \geq 0$ and either $f_0 = 1$ or $x \nmid_r f_0$. If $f_0 = 1$, return ε copies of x .
- (2) Compute the monic bound $\varphi \in \mathbb{C}$ of f_0 in $R = K(t)[x; \sigma]$, for example by linear algebra or by the algorithm of Gómez-Torrecillas et al. (2019). Since $x \nmid_r f_0$, the bound φ is not divisible by $\mathcal{X}$.
- (3) If φ is irreducible in $\mathbb{C}$, return the one-term factorization $f_0 = g_1$, followed by ε copies of x .
- (4) Otherwise choose a monic irreducible factor $\pi \neq \mathcal{X}$ of φ and write $\varphi = \rho\pi$, with $\rho, \pi \in \mathbb{C}$ monic and nonconstant.
- (5) Compute the monic right gcd in R : $v := \text{gcd}(f_0, \pi)$.
- (6) Compute the exact right quotient $u \in R$ satisfying $f_0 = uv$.
- (7) Apply the present algorithm recursively to u . If the recursive call returns

$$u = u_1 u_2 \cdots u_r, \quad \text{then return} \quad f = u_1 u_2 \cdots u_r v x^\varepsilon.$$

Algorithm 4.1 is the standard recursive rough-factor stage of the bound method: at each step one peels off a factor attached to a single irreducible central divisor and then recurses on the exact cofactor. The word “rough” is important: the output pieces have irreducible central bounds, but they need not yet be irreducible skew polynomials.

Proposition 4.2. *Algorithm 4.1 terminates. Apart from the explicitly returned copies of x , each returned factor is monic and has an irreducible monic bound different from X .*

Proof (Sketch). If $\text{gcd}(f_0, \pi) = 1$, a left Bézout identity shows that ρ is a smaller central left multiple of f_0 , contradicting minimality of φ ; hence $v = \text{gcd}(f_0, \pi)$ is nonconstant. If $v = f_0$, then π is itself a smaller central left multiple of f_0 , again impossible. Thus the exact cofactor u has smaller positive x -degree. Since π is a central left multiple of v , the monic bound of v divides π , hence equals π because π is irreducible. The case $\pi = X$ is excluded by the initial stripping step. Finally, if φ were divisible by X , then $\varphi = w f_0$ would imply $w \equiv 0 \pmod{RX}$, so φ/X would be a smaller central left multiple of f_0 . Therefore the recursion proceeds on smaller x -degree and preserves the product identity by exact right division. $\square$

This approach to the computation just described is already well supported by the literature (Giesbrecht and Zhang, 2003; Gómez-Torrecillas et al., 2019). The hard input class is therefore when $g \in R$ has an irreducible monic bound $\pi \in C$. The remainder of this article will focus on this case.

4.1 Norms and cyclic algebras

Our algorithm above and the subsequent discussions below are driven primarily by central bounds, their factorizations, and the remaining irreducible-bound case. Alternatively, *reduced norms* give another commutative invariant, and have been used algorithmically in skew factorization over finite fields and in the finite-order setting (Caruso and Le Borgne, 2017; Pumplün and Thompson, 2022). This norm is another central invariant controlled by the same central prime as the irreducible-bound case. We include it here because it is natural, relatively inexpensive to compute when m is fixed, and sometimes gives a useful preliminary test before the specialization stage.

Let E be the field of fractions $K(\mathcal{T})(X)$ of C and $L_{\text{gen}} := K(t)(X)$. Localizing R at the nonzero central elements gives

$$A_{\text{gen}} := E \otimes_C R \cong (L_{\text{gen}}/E, \sigma, X),$$

the generic cyclic algebra of degree m . Here L_{gen}/E is the cyclic extension obtained from $t^m = \mathcal{T}$, the automorphism is induced by $\sigma(t) = \omega t$, and the crossed-product relation is $x^m = X$.

For $g \in R$, collect its image in A_{gen} modulo powers of $x^m = X$ and write it on the right L_{gen} -basis $1, x, \dots, x^{m-1}$:

$$g = a_0 + x a_1 + \dots + x^{m-1} a_{m-1}, \quad a_i \in L_{\text{gen}}.$$

Equivalently, a left-coefficient expression $\sum b_i x^i$ is converted by using $b_i x^i = x^i \sigma^{-i}(b_i)$. Let M_g be the $m \times m$ matrix of left multiplication by g on this right L_{gen} -basis. We define

$$\text{Norm}(g) := \det(M_g).$$

This determinant is the reduced norm of g in A_{gen} , not the determinant of left multiplication on the full E -vector space A_{gen} , which would be a power of the reduced norm. Although the matrix entries lie in L_{gen} , the determinant lies in E ; for $g \in R$, the finite-order skew-polynomial norm results of [Pumplün and Thompson \(2022, Theorem 3, Lemma 5, Theorem 6, Corollary 7\)](#) imply that in fact

$$\text{Norm}(g) \in C = K(\mathcal{T})[X].$$

Moreover $\text{Norm}(g)$ is a central left multiple of g . Thus, the monic bound of g divides the monic associate of $\text{Norm}(g)$ in C .

With classical linear algebra, after collecting coefficients modulo $x^m = X$, the determinant above can be computed using $O(m^3)$ arithmetic operations over L_{gen} . The bit complexity also depends on the degrees and heights of the coefficient rational functions; we do not analyze those parameters here.

By the finite-order norm construction and irreducibility criterion of [Pumplün and Thompson \(2022\)](#), we obtain the following.

Proposition 4.3. *Let $g \in R$ be monic. If the monic associate of $\text{Norm}(g)$ is irreducible in C , then g is irreducible in R .*

Thus, following Algorithm 4.1, it is fast and convenient to compute the reduced norm on factors in the rough factorization (with irreducible bounds). If Proposition 4.3 applies we may be able to certify the factor irreducible. This test is sufficient but not necessary, and is not a complete irreducibility test, but could be helpful in an implementation.

5 Good characteristic-zero specializations $\mathcal{T} \mapsto a$

We now turn to the problem of factoring an irreducible-bound input $g \in R$ with monic bound

$$\pi \in C, \quad s = \deg_X \pi, \quad n_g = \deg_x g.$$

Fix for the remainder of this section an exact left quotient $u \in R$ with $\pi = ug$. Our first modular level specializes the single central parameter $\mathcal{T} = t^m$ to an algebraic integer $a \in O_K$, where O_K is the ring of all algebraic integers in K . To make specialization explicit, write every coefficient of g in the $K(\mathcal{T})$ -basis $1, t, \dots, t^{m-1}$:

$$K(t) = \bigoplus_{j=0}^{m-1} K(\mathcal{T})t^j,$$

so that

$$g = \sum_{i=0}^{n_g} \left(\sum_{j=0}^{m-1} c_{ij}(\mathcal{T})t^j \right) x^i, \quad c_{ij}(\mathcal{T}) \in K(\mathcal{T}).$$

The same applies to the coefficients of u and of π . Clearing finitely many denominators in these expansions gives a finite set of bad values of $\mathcal{T}$ that must be avoided for specialization even to be defined. The irreducibility requirements below are separate Hilbert/Kummer conditions, not merely finite exclusions.

Definition 5.1. An algebraic integer $a \in \mathcal{O}_K$ is called a **good central specialization** for (g, π) , or simply **good for** (g, π) , if all of the following hold:

- (S1) no denominator occurring in the $\mathcal{T}$ -coefficients of g , u , or π vanishes at $\mathcal{T} = a$;
- (S2) the specialization of π has the same X -degree as π , and $\text{disc}_X(\pi(a, X)) \neq 0$;
- (S3) the specialized central polynomial

$$\pi_a(X) := \pi(a, X) \in K[X]$$

is irreducible and satisfies $\pi_a \neq X$;

- (S4) the Kummer polynomial $y^m - a \in K[y]$ is irreducible.

Condition (S2) is mostly automatic from monicity once a fixed presentation has been cleared of denominators, but it is useful to record explicitly because discriminants and leading coefficients will also be excluded in later reductions. Condition (S4) implies that

$$L_a = K(\theta_a), \quad \theta_a^m = a,$$

has degree m over K . Since K contains the group μ_m of m -th roots of unity, the map

$$\sigma_a(\theta_a) = \omega \theta_a$$

extends to a K -automorphism of L_a of order m , and its fixed field is K . Thus, the centre of $R_a := L_a[x; \sigma_a]$ is $Z(R_a) = K[x^m] = K[X]$.

For the correctness statements below we only use that any chosen a satisfying Definition 5.1 behaves as claimed. Hilbert irreducibility is the natural tool for proving the existence of many a satisfying the irreducibility conditions (S3) and (S4). Chebotarev density type tools are more relevant later, when one asks for good inert primes for a fixed specialization.

Let $\Delta(\mathcal{T}) \in K[\mathcal{T}]$ be a common denominator for all coefficients of g , u , and π when these are written in the $K(\mathcal{T})$ -basis $1, t, \dots, t^{m-1}$. Set

$$B_\Delta = K[\mathcal{T}, \Delta(\mathcal{T})^{-1}, t] / (t^m - \mathcal{T}) \subset K(t).$$

For a satisfying (S1), define the specialization homomorphism

$$\text{sp}_a : B_\Delta \longrightarrow L_a, \quad \sum_{j=0}^{m-1} r_j(\mathcal{T}) t^j \longmapsto \sum_{j=0}^{m-1} r_j(a) \theta_a^j,$$

where each $r_j(\mathcal{T})$ is regular at $\mathcal{T} = a$. Thus sp_a is not a homomorphism from all of $K(t)$, but from the localized coefficient ring on which the chosen input data are defined.

Applying this map coefficientwise to g and u produces specialized skew polynomials

$$g_a, u_a \in R_a = L_a[x; \sigma_a]$$

satisfying $\pi_a = u_a g_a$. Because g is monic in x , the specialization g_a is also monic and nonconstant.

In practice one may sample rational integers $a \in \mathbb{Z} \subset \mathcal{O}_K$, but the definition of goodness uses exact tests. In particular, irreducibility of $y^m - a$ is tested in $K[y]$ using standard commutative factorization algorithms (see, e.g., (von zur Gathen and Gerhard, 2013)). The specialized irreducible-bound problem remains a cyclic-algebra problem, but now over a number field.

Proposition 5.2. *Let a be good for (g, π) . Then the monic bound of $g_a \in R_a = L_a[x; \sigma_a]$ is exactly π_a . Put*

$$F_a = K[X]/(\pi_a), \quad A_a = R_a/R_a\pi_a.$$

Then A_a is a central simple F_a -algebra of degree m . If

$$A_a \cong M_{r_a}(D_a),$$

with D_a a central division F_a -algebra of degree d_a , so $r_a d_a = m$, then every irreducible factor of g_a in R_a has x -degree $s d_a$.

Proof. The identity $\pi = ug$ specializes to $\pi_a = u_a g_a$ in R_a . Since $\pi_a \in K[X] = Z(R_a)$, it is a central left multiple of g_a . Therefore the monic bound of g_a divides π_a in $K[X]$.

The specialization g_a is still monic in x , hence nonconstant. Thus, its bound is nonconstant. By (S3), π_a is irreducible in $K[X]$ and $\pi_a \neq X$, so the only possible nonconstant monic divisor of π_a is π_a itself. Hence the monic bound of g_a is exactly π_a .

Since π_a is central, $R_a\pi_a = \pi_a R_a$ is a two-sided ideal. The ring R_a has centre $K[X]$, and the image of X in F_a is nonzero because $\pi_a \neq X$; hence it is invertible. Thus the same cyclic-algebra structure used in Fact 3.1 applies to the quotient $A_a = R_a/R_a\pi_a$: it is a central simple F_a -algebra of degree m .

Finally, (S2) gives $\deg_X(\pi_a) = \deg_X(\pi) = s$. Applying the same argument as in Fact 3.1 in the specialized ring R_a yields that every irreducible factor of g_a has degree $s d_a$ in x . $\square$

The integer $d_a = \text{ind}(A_a)$ is, in principle, decidable by known cyclic-algebra methods over number fields. In particular, Hanke (2007) reduces splitting and isomorphism questions for cyclic algebras to field-theoretic tasks including norm equations. Thus the specialization $\mathcal{T} \mapsto a$ has two purposes: it preserves characteristic zero, and it converts the unknown generic obstruction into a number-field cyclic-algebra problem.

The most useful interpretation of d_a is as a block size. If $d_a = 1$, then g_a factors into pieces of degree s . If $d_a > 1$, then g_a factors into pieces of degree $s d_a$. These are exact statements for the specialized problem. They may or may not agree with the generic factor degrees of g over $\mathbb{R}$, so we use them only to organize the search, not to certify irreducibility or reducibility upstairs!

6 Good prime reductions of a specialization

After a good characteristic-zero specialization, the second modular level reduces the specialized problem to finite fields, where we have effective and efficient algorithms.

We assume throughout this section that we have fixed a good specialization $a \in \mathcal{O}_K$ and the number field L_a/K . Because L_a/K is a cyclic Kummer extension of degree m generated by θ_a , the preferred primes are the inert ones.

Definition 6.1. *A nonzero prime ideal $\mathfrak{q} \subset \mathcal{O}_K$ is called good for the specialization a if all of the following hold:*

(P1) $\mathfrak{q}$ is unramified and inert in L_a/K ;

- (P2) $\mathfrak{q}$ avoids the denominators, leading coefficients, and discriminants arising in g_a , u_a , and π_a ;
- (P3) the reduction $\bar{\pi}_{a,\mathfrak{q}}$ of π_a modulo $\mathfrak{q}$ is irreducible in $k_{\mathfrak{q}}[X]$ and satisfies $\bar{\pi}_{a,\mathfrak{q}} \neq X$, where $k_{\mathfrak{q}} = O_K/\mathfrak{q}$.

As in the specialization step, we do not prove, and do not need, that infinitely many good inert primes exist for every good a . Correctness of the returned factors uses only the behaviour of primes satisfying Definition 6.1; completeness would require an additional Chebotarev-type supply theorem. The search for such primes is therefore part of the heuristic modular stage: if no such primes are found within the prescribed budget, the stage returns failure.

If $\mathfrak{q}$ is inert, there is a unique prime $\mathfrak{p}$ of L_a above it, and the residue field $\lambda_{a,\mathfrak{q}} = O_{L_a}/\mathfrak{p}$ has degree m over $k_{\mathfrak{q}}$. The automorphism σ_a preserves $\mathfrak{p}$ and therefore descends to an automorphism $\bar{\sigma}_{a,\mathfrak{q}} \in \text{Gal}(\lambda_{a,\mathfrak{q}}/k_{\mathfrak{q}})$ of order m . Thus we obtain a finite-field skew polynomial ring $\bar{R}_{a,\mathfrak{q}} = \lambda_{a,\mathfrak{q}}[x; \bar{\sigma}_{a,\mathfrak{q}}]$. Let $\bar{g}_{a,\mathfrak{q}}$ denote the reduction of g_a .

Proposition 6.2. *Let a be good for (g, π) and let $\mathfrak{q}$ be good for a . Then:*

- (a) *the monic bound of $\bar{g}_{a,\mathfrak{q}}$ is the reduction $\bar{\pi}_{a,\mathfrak{q}}$ of π_a ;*
- (b) *every irreducible factor of $\bar{g}_{a,\mathfrak{q}}$ in $\bar{R}_{a,\mathfrak{q}}$ has degree exactly s .*

Proof. Reducing the identity $\pi_a = u_a g_a$ modulo $\mathfrak{q}$ gives

$$\bar{\pi}_{a,\mathfrak{q}} = \bar{u}_{a,\mathfrak{q}} \bar{g}_{a,\mathfrak{q}}.$$

Thus $\bar{\pi}_{a,\mathfrak{q}}$ is a central multiple of $\bar{g}_{a,\mathfrak{q}}$, so the monic bound of $\bar{g}_{a,\mathfrak{q}}$ divides $\bar{\pi}_{a,\mathfrak{q}}$ in $k_{\mathfrak{q}}[X]$. Because $\bar{\pi}_{a,\mathfrak{q}}$ is irreducible and $\bar{\pi}_{a,\mathfrak{q}} \neq X$ by (P3), it follows that the monic bound of $\bar{g}_{a,\mathfrak{q}}$ is exactly $\bar{\pi}_{a,\mathfrak{q}}$. This proves (a).

For (b), the quotient $\bar{A}_{a,\mathfrak{q}} = \bar{R}_{a,\mathfrak{q}}/\bar{R}_{a,\mathfrak{q}}\bar{\pi}_{a,\mathfrak{q}}$ is a central simple algebra over the finite field $\bar{F}_{a,\mathfrak{q}} := k_{\mathfrak{q}}[X]/(\bar{\pi}_{a,\mathfrak{q}})$. Since $\text{Br}(\bar{F}_{a,\mathfrak{q}}) = 0$, this algebra is split, so its index is 1. Fact 3.1 in the specialized finite-field setting therefore gives factor degree s . $\square$

Proposition 6.2 is the reason factorization over a finite field is efficient here. Even when the specialized characteristic-zero problem has block size $d_a > 1$, the finite reductions split all the way down to degree- s irreducible modular factors. The price is that these modular factors are usually too small and must be recombined.

6.1 Factoring a specialization by inert prime reductions

For fixed a , the practical task is now clear: factor g_a over $L_a[x; \sigma_a]$ by repeatedly reducing to $\bar{R}_{a,\mathfrak{q}}$, factoring there, and lifting exact blocks of degree sd_a .

Algorithm 6.3 is deliberately stated at the level of a *certified heuristic*. The finite-field factorizations are fast and exact; the heuristic part is the candidate generation and reconstruction search. In particular, Step (2c) is only a way to produce candidate blocks from the modular factors; it does not assert that one fixed finite-field factorization enumerates all right divisors of degree sd_a . Every accepted specialized factor is nevertheless correct because it is certified by exact division in the specialized ring.

Algorithm 6.3 Factoring one specialization

Input: a good specialization a for an irreducible-bound input g , together with the block size $d_a = \text{ind}(A_a)$.

Output: a certified factorization of g_a in $L_a[x; \sigma_a]$, or failure after a prescribed search budget.

- (1) Choose several good inert primes $q_1, \dots, q_r$ of K for the specialization a .
- (2) For each q_i :
 - (a) construct the finite field λ_{a, q_i} and the reduced skew ring $\bar{R}_{a, q_i}$;
 - (b) factor $\bar{g}_{a, q_i}$ completely by [Caruso and Le Borgne \(2017\)](#) or [Giesbrecht \(1998\)](#);
 - (c) generate candidate monic right divisors of degree sd_a from ordered products, or other right-divisor recombinations, of the degree- s modular factors.
- (3) For a chosen compatible collection of modular blocks, and after determining a common denominator for the candidate coefficients, use Chinese remaindering across the primes $p_i \subset \mathcal{O}_{L_a}$ above q_i to reconstruct candidate coefficient data in an integral basis of $\mathcal{O}_{L_a}$.
- (4) Recover a monic candidate factor $\hat{h}_a \in L_a[x; \sigma_a]$ of degree sd_a by rational reconstruction in L_a , and certify it by exact right division of g_a .
- (5) If certification succeeds, then $\hat{h}_a$ is irreducible; by [Proposition 5.2](#), every irreducible factor of g_a has x -degree exactly sd_a , while $\deg_x(\hat{h}_a) = sd_a$. Recurse on the exact cofactor, and continue until g_a is completely factored.
- (6) If no candidate survives, enlarge the set of inert primes or the search budget and continue. If the budget is exhausted, return failure.

7 Lifting from several specializations back to $K(t)$

The finite-field factorizations obtained from several specializations must then be reconciled upstairs. Assume now that for several good values $a_1, a_2, \dots, a_r \in \mathcal{O}_K$ we have obtained certified specialized factors of g_{a_i} in $L_{a_i}[x; \sigma_{a_i}]$.

Write a candidate specialized factor in the basis $1, \theta_{a_i}, \dots, \theta_{a_i}^{m-1}$:

$$h_{a_i} = \sum_{\ell=0}^e \left(\sum_{j=0}^{m-1} c_{\ell j}(a_i) \theta_{a_i}^j \right) x^\ell, \quad c_{\ell j}(a_i) \in K,$$

where e is the candidate degree upstairs. A generic factor upstairs has the form

$$h = \sum_{\ell=0}^e \left(\sum_{j=0}^{m-1} c_{\ell j}(\mathcal{T}) t^j \right) x^\ell, \quad c_{\ell j}(\mathcal{T}) \in K(\mathcal{T}).$$

Thus, each good specialization supplies the values

$$c_{\ell j}(\mathcal{T})|_{\mathcal{T}=a_i} = c_{\ell j}(a_i).$$

This is now a *commutative* interpolation problem over K .

Definition 7.1. For fixed degree e , a family of specialized factors $h_{a_1}, \dots, h_{a_r}$ is called **B -compatible** if, for every lower coefficient pair (ℓ, j) , with $\ell < e$ and $0 \leq j < m$, the data $(a_1, c_{\ell j}(a_1)), \dots, (a_r, c_{\ell j}(a_r))$ interpolate a rational function in $K(\mathcal{T})$ with numerator and denominator degree at most B .

Algorithm 7.2 Lifting from specializations**Input:** an irreducible-bound input g , together with specialized factorizations of $g_{a_1}, \dots, g_{a_r}$.**Output:** a certified factorization of g in $\mathbb{R}$, or failure after a prescribed search budget.

- (1) For each divisor d of m with $sd \mid n_g$ and $sd < n_g$, set $e = sd$.
- (2) For each specialization a_i , build a candidate list of monic degree- e right divisors of g_{a_i} from selected products of the exact irreducible factors found in Algorithm 6.3 (noting that a specialized divisor of degree e must be a product of $e/(sd_{a_i})$ specialized irreducible factors, so we only use a_i with $sd_{a_i} \mid e$).
- (3) Search, among the specializations retained for this e , for tuples of degree- e divisors $(h_{a_1}, \dots, h_{a_r})$ that are B -compatible.
- (4) For each B -compatible tuple, choose a rational reconstruction of the lower coefficient functions, that is, functions $\widehat{c}_{\ell j}(\mathcal{T}) \in \mathbb{K}(\mathcal{T})$ for $\ell < e$ and $0 \leq j < m$, each with numerator and denominator degree at most B , whose values at the used specializations agree with the data $c_{\ell j}(a_i)$. From these, reconstruct a monic candidate

$$\widehat{h} = x^e + \sum_{\ell=0}^{e-1} \left(\sum_{j=0}^{m-1} \widehat{c}_{\ell j}(\mathcal{T}) t^j \right) x^\ell \in \mathbb{R}.$$

- (5) Compute the exact right remainder $\text{rrem}(g, \widehat{h})$ in $\mathbb{R}$.
- (6) If the remainder is zero, compute the exact cofactor q with $g = q\widehat{h}$, recurse on q and on $\widehat{h}$ until the current factor is completely factored, and return the resulting certified factorization.
- (7) If no candidate survives certification for any admissible degree e , enlarge the search budget by using more specializations a_i , larger height bound B , or more candidate blocks inside the specialized factorizations, and continue.
- (8) If the prescribed search budget is exhausted before a complete factorization is obtained, return failure.

Algorithm 7.2 is the characteristic-zero analogue of combining modular factors in commutative factorization. The difference is that the local factors come from *two* modular levels: inert-prime factorization inside each specialization, and then interpolation across several characteristic-zero specializations. As in Algorithm 6.3, Step (2) is a candidate-generation device rather than a claim that one fixed specialized factorization lists every right divisor of degree e .

8 A certifying two-level algorithm for factorization over $\mathbb{K}(t)$

We can now collect the previous stages into one certifying procedure over $\mathbb{K}(t)$.

Theorem 8.2. *Every factorization returned by Algorithm 8.1 is correct, i.e., it multiplies to f , and every factor declared irreducible has been certified by one of the stated irreducibility tests.*

Proof. At the central stage, every factor is certified by exact right gcd computations. The irreducibility declarations made before specialization are certified by exact verification

Algorithm 8.1 Certifying two-level modular factorization in the quantum plane**Input:** a monic skew polynomial $f \in K[t][x; \sigma]$.**Output:** either a complete certified factorization of f in $K(t)[x; \sigma]$, or failure after the prescribed search budget.

- (1) Normalize f : remove its scalar content in $K[t]$, clear denominators if necessary, and make it monic in x .
- (2) Apply Algorithm 4.1 to split f into factors with central-irreducible bounds.
- (3) For each irreducible-bound factor g of f :
 - (a) Check the conditions of Corollary 3.2 (degree of irreducible bound) and Proposition 4.3 (irreducible norm); if either test proves irreducibility, keep g and continue to the next current factor.
 - (b) otherwise choose good central specializations $a_1, a_2, \dots$;
 - (c) for each a_i , compute the specialized block size $d_{a_i} = \text{ind}(A_{a_i})$ using cyclic-algebra methods over number fields;
 - (d) attempt to factor g_{a_i} by Algorithm 6.3;
 - (e) from any successful specialized factorizations, apply Algorithm 7.2 to reconstruct certified factors of g upstairs.
- (4) If every current factor has been proved irreducible or completely factored, return the resulting decomposition; otherwise return failure.

of the hypotheses of Corollary 3.2 or Proposition 4.3. In the modular stage, specialized factors are certified by exact right division in the specialized rings, and lifted factors are certified by exact right division in the original ring $K(t)[x; \sigma]$. Therefore the algorithm may fail to find a factorization, but every factorization it returns is correct. $\square$

Algorithm 8.1 is not claimed to be complete over $K(t)$: in the irreducible-bound stage it is a certifying search procedure and may return failure. What is *not* proved is completeness or a general complexity bound. The unresolved part is the same one that appears over $\mathbb{F}_q(t)$: the irreducible-bound case is governed by a simple algebra. The present algorithm circumvents that algebra heuristically by passing to specializations and finite inert primes, but does not eliminate the underlying difficulty.

9 Complexity and practical remarks

Only part of the method currently comes with a plausible cost analysis. A full bit-complexity discussion must include the coefficient-height parameter of the input. Let H denote a height bound for the coefficients of f when they are written in the basis $1, t, \dots, t^{m-1}$ over $K(\mathcal{T})$.

The part of the cost picture that is currently under control is the following.

- (C1) Computing the bound and the reducible-bound stage are polynomial-time in the algebraic operations model, and are effective in the bit model assuming standard commutative factorization over number fields; a full height analysis is not attempted here.

- (C2) The maximal-bound-degree test is negligible once the bound is known.
- (C3) For each good specialization a , computing the specialized block size $d_a = \text{ind}(A_a)$ invokes cyclic-algebra algorithms over number fields. This step is effective in principle, but its cost is not analyzed here.
- (C4) For fixed specialization a and inert prime q , the factorization of the finite-field image of degree n is handled by [Caruso and Le Borgne \(2017\)](#) or [Giesbrecht \(1998\)](#). This is the fastest stage of the algorithm.
- (C5) Reconstruction over one specialization uses Chinese remaindering in the Dedekind domain $\mathcal{O}_{L_a}$ and linear algebra in an integral basis of L_a .
- (C6) Lifting from several specializations uses rational interpolation over K in the single central variable $\mathcal{T}$.

The difficult part is the search. There are in fact *two* search spaces:

- (S1) the block search inside each specialized finite-field factorization, where degree- s modular factors must be grouped into blocks that lift to degree- sd_a factors over L_a ;
- (S2) the compatibility search across different values of a , where products of specialized factors must interpolate to coefficient functions in $K(\mathcal{T})$.

Both searches are helped by exact certification and by aggressive pruning through height bounds, but the cost of neither is governed by a strong theorem. In particular, we make no complexity claim for the irreducible-bound stage over $K(t)$.

The method appears computationally attractive because the centre is a univariate PID, central specializations $\mathcal{T} \mapsto a$ are easy to sample, and the expensive noncommutative work is pushed into finite-field images where fast algorithms already exist.

10 Historical context and related work

Our strategy combines four earlier viewpoints: the centre-and-bound philosophy of [Giesbrecht and Zhang \(2003\)](#) and [Gómez-Torrecillas et al. \(2019\)](#), the root-of-unity quantum-plane phenomena studied by [Coulibaly and Price \(2006\)](#), norm methods in finite-order skew-polynomial rings ([Pumplün and Thompson, 2022](#)), and algorithms for cyclic and central simple algebras over number fields ([Hanke, 2007](#); [Ivanyos et al., 2012](#)). The key point here is that finite reductions alone do not preserve the arithmetic obstruction attached to an irreducible bound; the characteristic-zero specialization step is done to expose that cyclic-algebra structure before passing to finite fields for speed.

The polynomial quantum plane also lies within the general class of quasi-commutative G -algebras supported by the noncommutative subsystem PLURAL of SINGULAR ([Levandovskyy, 2006](#)); this framework provides PBW-type normal forms, noncommutative Gröbner bases, elimination, and related algorithms for broad classes of noncommutative polynomial algebras. The excellent `ncfactor.lib` library of [Levandovskyy and Heinle \(2018\)](#) implements ansatz-driven factorization methods for several such algebras, building on finite factorization results for G -algebras and related classes. This work is therefore important adjacent infrastructure, especially for polynomial quantum-plane computations. The problem considered here is different in two respects. First,

we factor in the localized Ore ring $K(t)[x; \sigma]$, not only in the polynomial G -algebra $K\langle t, x \rangle / (xt - \omega tx)$, so denominators in t may occur in factors. Second, the algorithm exploits the root-of-unity centre $K(t^m)[x^m]$, through the additional structure of central bounds, right-gcd splitting, irreducible-bound cyclic algebras, good specialization, inert-prime finite-field reduction, modular recombination, and interpolation.

11 What is presently algorithmic over $K(t)$?

We now separate what is truly computable, what is only recursively enumerable, and what remains open over $K(t)$.

Proposition 11.1. *The following tasks are computable (without claim of efficiency):*

- (a) *Normalization of the input, computation of the monic bound φ , commutative factorization of φ in $\mathbb{C} = K(\mathcal{T})[X]$, and the rough factorization of f into central-irreducible pieces by right gcd computations.*
- (b) *The maximal-bound-degree test of Corollary 3.2 and the norm-based irreducibility test of Proposition 4.3.*
- (c) *For each chosen good specialization $\mathcal{T} \mapsto a$ and each chosen good inert prime $\mathfrak{q}$, the construction of the specialized skew ring, the finite-field image, the factorization of that image, and the certification of any reconstructed factor by exact right division.*
- (d) *Interpolation of candidate coefficient functions from finitely many characteristic-zero specializations, together with exact verification of the resulting candidate factors in the original ring.*

Proof (Discussion). Parts (a)–(d) are immediate from the constructions in Sections 4–7, together with the cited commutative and finite-field skew-factorization routines. $\square$

Because $K(t)$ is countable and effectively presented, one can enumerate all monic candidates of smaller x -degree in $K(t)[x; \sigma]$ and test them by exact right division. Thus reducibility over $K(t)$ is recursively enumerable. What is missing is the converse semidecision: there is no known procedure that certifies irreducibility in the generic irreducible-bound case over $K(t)$.

11.1 The missing ingredient for decidability over $K(t)$

The missing step is still the generic irreducible-bound algebra.

Proposition 11.2. *Assume that for every monic irreducible-bound input $g \in K(t)[x; \sigma]$ with monic bound $\pi \in K(\mathcal{T})[X]$, $\pi \neq X$, there is an algorithm which constructs the finite-dimensional algebra*

$$E_g := \text{End}_{A_\pi}(M_g), \quad F_\pi = K(\mathcal{T})[X]/(\pi), \quad A_\pi = R/R\pi, \quad M_g = R/Rg,$$

and either proves that E_g is a division algebra or returns a nontrivial idempotent of E_g . Then full factorization in $K(t)[x; \sigma]$ is decidable.

Proof (Sketch). Apply Algorithm 4.1 to reduce to irreducible-bound pieces. Let g be one such piece, with bound π . Since $\pi \in Rg$, the containment $R\pi \subseteq Rg$ gives $M_g = R/Rg$ the structure of a left A_π -module. Write $A_\pi \cong M_r(D)$, where D is a central division F_π -algebra of degree d . If S is a simple left A_π -module, then

$$M_g \cong S^\ell, \quad \ell = \frac{\deg_x g}{sd}, \quad s = \deg_\chi \pi,$$

and therefore $E_g = \text{End}_{A_\pi}(M_g) \cong M_\ell(D^{\text{op}})$. Thus g is irreducible if and only if $\ell = 1$, equivalently if and only if E_g is a division algebra.

If the assumed algorithm proves that E_g is a division algebra, then g is irreducible. Otherwise it returns a nontrivial idempotent $e \in E_g$. The image eM_g is a proper nonzero A_π -submodule of M_g . Taking its inverse image under $R \rightarrow M_g$ gives an intermediate left ideal $Rg \subsetneq I \subseteq R$. Since R is a left principal ideal domain, write $I = Rh$. Then $g \in Rh$, so $g = qh$ for some $q \in R$, and h is a proper right factor of g . Exact right division verifies the factorization. Recursing on the exact cofactors gives a complete factorization. $\square$

Proposition 11.2 isolates the genuinely arithmetic obstruction: the unresolved object is not merely the central simple algebra A_π attached to the central prime π , but the A_π -module $M_g = R/Rg$, or equivalently its eigenring $E_g = \text{End}_{A_\pi}(M_g)$. A zero divisor in A_π alone does not necessarily split the particular polynomial g .

11.2 Questions governing efficiency over $K(t)$

Even if one had a proof of decidability, a useful complexity theory for the modular stage would still require quantitative control of the search. The main missing ingredients are the following.

Proposition 11.3. *A proof of completeness with effective running-time control for the irreducible-bound stage over $K(t)$ would require, at minimum, satisfactory answers to the following questions.*

- (i) *Specialization control. Is there a Hilbert- or Chebotarev-type specialization theorem showing that finitely many good values a and finitely many good inert primes $\mathfrak{q}$ suffice to recover the generic factorization pattern of an irreducible-bound input?*
- (ii) *Block recombination. Can one control how the local degree- s modular factors in the finite-field images recombine into the correct specialized factors of degree sd_a over the number fields L_a ?*
- (iii) *Reconstruction bounds. Is there an explicit normalization and corresponding height bound on the coefficient functions in $\mathcal{T}$ that makes the interpolation stage finite and quantitatively controllable?*

Proof (Discussion). Questions (i)–(iii) correspond exactly to the outer specialization/primes search, the inner block recombination search, and the final interpolation step back to $K(t)$. $\square$

Propositions 11.2 and 11.3 explain the role of the present modular algorithm. It is not meant to replace the generic irreducible-bound algebra; rather, it is a practical surrogate for it. The modular stages are the parts of the search that are currently effective. The unresolved issue is whether they always recover the generic algebra information in finite time, and if so at what cost.

12 SageMath implementation and computational examples

We have made a prototype implementation of Algorithm 8.1 and its subroutines in SageMath 10.8 (The Sage Developers, 2025). All code and sample runs are available for trial.* The implementation represents elements of $K(t)[x; \sigma]$, $\sigma(t) = \omega t$, by writing coefficients in the $K(\mathcal{T})$ -basis $1, t, \dots, t^{m-1}$. It implements skew multiplication, exact right division, right gcds, and the conversion between central polynomials $K(\mathcal{T})[X]$ and skew polynomials. The bound of a monic input is computed by solving for the minimal monic central left multiple. The reducible-bound part then factors this bound in the commutative ring $K(\mathcal{T})[X]$ and applies the right-gcd splitting procedure of Algorithm 4.1. For a piece with irreducible bound, the code first applies the degree criterion of Corollary 3.2 and the norm prefilter of Proposition 4.3. If these do not settle the reducibility of the polynomial, the implementation enters the two-level modular stage: it searches for good specializations $\mathcal{T} \mapsto a$, constructs the Kummer extension $L_a = K(\theta_a)$, $\theta_a^m = a$, searches for good inert primes, reduces to a finite-field skew-polynomial ring, and invokes Sage's finite-field skew factorization backend, using the provided code of Caruso and Le Borgne (2017). Candidate factors are lifted by Chinese remaindering, rational reconstruction, and interpolation in the single central parameter $\mathcal{T}$. As in Algorithms 6.3 and 7.2, every accepted factor is certified by exact right division in the original ring. The prototype also contains a small exact sparse search, used only as a certifying fallback for deliberately constructed examples such as $(x^e - t)^2$ or $X^e - \mathcal{T}$. This fallback is not a replacement for the missing generic irreducible-bound algorithm identified in Section 11; it only makes these test cases reproducible while preserving exact certification.

Table 1 summarizes one run of the example suite on an Apple Mac Mini M2 Pro with 32GB RAM. The examples were chosen to exercise different parts of the method. The central examples C1–C3 and S1, S3 are products of linear skew factors and are solved completely by central bound factorization and right gcds. The mixed example S2 is still central-stage-complete, but has irreducible central factors of degrees 2 and 4 as well as six linear central factors. The examples I1 and I2 have irreducible bounds

$$X^2 - \mathcal{T}, \quad X^2 - \mathcal{T}(\mathcal{T} + 1)^3,$$

respectively, and factor as squares of degree-two skew polynomials. The examples B1 and S5 factor the irreducible central polynomials themselves: for $m = 3$,

$$X^e - \mathcal{T} = x^{3e} - t^3 = (x^e - t)^3$$

* https://cs.uwaterloo.ca/~mwg/archive/quantum_plane_factorization

Table 1. Timings for the SageMath example suite. The column “bound” lists the degrees of the irreducible factors of the central bound $\varphi \in K(\mathcal{T})[X]$. The column “output” lists the x -degrees of the certified skew factors. Here C means central factorization only, C+DB means central factorization plus degree-bound irreducibility tests, M/S means modular attempt plus exact sparse certification, and CI/S means factorization of an irreducible central polynomial by exact sparse certification.

ID	stage	m	$\deg_x f$	$\deg_X \varphi$	bound	output	time (s)
C1	C	2	2	2	1×2	1×2	0.012
C2	C	2	5	5	1×5	1×5	0.069
C3	C	3	5	5	1×5	1×5	0.396
I1	M/S	3	4	2	2	2×2	0.850
I2	M/S	3	4	2	2	2×2	0.403
B1	CI/S	3	6	2	2	2×3	0.031
S1	C	3	8	8	1×8	1×8	2.145
S2	C+DB	3	12	12	$1 \times 6, 2, 4$	$1 \times 6, 2, 4$	148.660
S3	C	5	6	6	1×6	1×6	0.972
S4	M/S	3	8	4	4	4×2	599.347
S5	CI/S	3	15	5	5	5×3	0.049
Total wall time for the selected examples							752.934

for $e = 2$ and $e = 5$. Finally, S4 is the larger irreducible-bound square $(x^4 - t)^2$, with bound $X^4 - \mathcal{T}$. The timings should be read as implementation data rather than complexity claims: in particular, S2 and S4 show that either central splitting or irreducible-bound search can dominate the running time on constructed inputs.

All rows in Table 1 ended with an exact product check in $K(t)[x; \sigma]$. Thus the examples are not evidence of completeness for the irreducible-bound stage, but they do try to demonstrate that the presently algorithmic components of Proposition 11.1 can be assembled into a working certifying implementation.

Note that we found no SageMath or other software for the characteristic-zero quantum-plane factorization problem considered here. SageMath provides effective routines for univariate Ore/skew polynomial rings, finite-order central invariants such as bounds and reduced norms, and an excellent finite-field skew-polynomial factorization backend implementing the Caruso–Le Borgne algorithm. These tools are important ingredients in our implementation. An implementation of characteristic-zero algorithms over $\mathbb{Q}(\omega)(t)[x; \sigma]$, including central decomposition, good specialization $\mathcal{T} \mapsto a$, inert-prime reduction, modular factor recombination, interpolation in $\mathcal{T}$, and final exact certification, does not seem to be available in any publicly available software.

13 Decidability over $\overline{\mathbb{Q}}(t)$ in the exact algebraic model

We next consider the same factorization problem after enlarging the field of constants from K to the algebraic closure $\overline{\mathbb{Q}}$. The analogous picture over $\mathbb{C}$ will be discussed separately in Subsection 13.3. In this setting the generic irreducible-bound algebra always splits, and the remaining problem is constructive rather than arithmetic. Set

$$\tilde{R} = \overline{\mathbb{Q}}(t)[x; \sigma], \quad \sigma(t) = \omega t,$$

with centre

$$\tilde{\mathbb{C}} = \overline{\mathbb{Q}}(\mathcal{T})[\mathcal{X}], \quad \mathcal{T} = t^m, \mathcal{X} = x^m.$$

By the *standard exact algebraic model* over $\overline{\mathbb{Q}}$ we mean the usual symbolic computation model in which every input is given over some explicit finite extension $E_0/\mathbb{Q} \subset \overline{\mathbb{Q}}$, represented by a primitive element, and computations are then carried out exactly in E_0 and in further finite extensions obtained by adjoining roots as needed. Arithmetic, equality testing, gcd computations, and commutative polynomial factorization over each such finite extension are exact. In this sense $\overline{\mathbb{Q}}$ is handled as the directed union of the finite extensions of $\mathbb{Q}$. See [Steel \(2010\)](#) for an effective algorithmic approach to computing in algebraic closures.

13.1 Irreducible-bound pieces over algebraically closed constants

Over an algebraically closed field of constants the simple algebra attached to an irreducible bound becomes just a matrix algebra over its centre. We say that such an algebra *splits* over a field F , meaning simply that it is isomorphic to a full matrix algebra $M_r(F)$ for some r . This removes the division-algebra obstruction, so the factor degree in Fact 3.1 drops to the minimal possible value s .

Theorem 13.1. *Let $g \in \tilde{\mathbb{R}}$ be monic with irreducible bound $\pi \in \tilde{\mathbb{C}}$, $\pi \neq \mathcal{X}$, and write $s = \deg_{\mathcal{X}}(\pi)$, $\tilde{F} = \tilde{\mathbb{C}}/(\pi)$, and $A = \tilde{\mathbb{R}}/\tilde{\mathbb{R}}\pi$. Then A is a split central simple algebra over $\tilde{F}$. Thus, every irreducible factor of g in $\tilde{\mathbb{R}}$ has degree exactly s in x . In particular, g is irreducible if and only if $\deg_x(g) = s$.*

Proof. The field $\tilde{F}$ is the function field of a curve over the algebraically closed field $\overline{\mathbb{Q}}$. By Tsen's theorem, $\tilde{F}$ is a C_1 -field, and since C_1 -fields have trivial Brauer group, we get $\text{Br}(\tilde{F}) = 0$. See ([Gille and Szamuely, 2006](#), Theorem 6.2.8 and Proposition 6.2.3). Since A is a finite-dimensional central simple $\tilde{F}$ -algebra, it follows that A is split, i.e., $A \cong M_m(\tilde{F})$. Thus the index in Fact 3.1 is $d = 1$, and every irreducible factor degree is $sd = s$. $\square$

13.2 A complete exact factorization algorithm over $\overline{\mathbb{Q}}(t)$

The splitting theorem immediately turns the irreducible-bound case into an exact, if very inefficient, search problem.

Theorem 13.2. *In the exact algebraic model over $\overline{\mathbb{Q}}$, factorization in $\overline{\mathbb{Q}}(t)[x; \sigma]$ is decidable. More precisely, there is an exact algorithm that returns a complete factorization of any monic input in $\overline{\mathbb{Q}}(t)[x; \sigma]$.*

Proof. Let $f \in \overline{\mathbb{Q}}(t)[x; \sigma]$ be monic. Choose a finite extension $E_0/\mathbb{Q}$ inside $\overline{\mathbb{Q}}$ that contains K and all coefficients of f . Let φ be the monic bound of f . In the exact algebraic model over $\overline{\mathbb{Q}}$, factor φ completely in $\tilde{\mathbb{C}} = \overline{\mathbb{Q}}(\mathcal{T})[\mathcal{X}]$. Since φ has finitely many coefficients, the coefficients of its central irreducible factors lie in some finite extension $E_1/\mathbb{Q}$ obtained from E_0 by adjoining finitely many algebraic constants. Replacing E_0 by E_1 , apply the same central decomposition as Algorithm 4.1 over $E_1(t)[x; \sigma]$ to split

f into factors whose bounds are irreducible in $\widetilde{\mathbb{C}}$. It therefore suffices to treat one such irreducible-bound piece g with bound π .

Set $s = \deg_X(\pi)$. If $\deg_x(g) = s$, then g is irreducible by Theorem 13.1. Assume therefore that $\deg_x(g) > s$. Then Theorem 13.1 implies that g has a nontrivial right factor of x -degree exactly s .

We now enumerate finite extensions $E/\mathbb{Q}$ containing E_0 , represent each one exactly by a primitive element, and inside each $E(t)$ enumerate all tuples of rational functions that can occur as the lower coefficients of a monic degree- s skew polynomial

$$h = x^s + h_{s-1}x^{s-1} + \cdots + h_0.$$

For each candidate h , compute the exact right remainder $\text{rrem}(g, h)$. Because an actual right factor of degree s exists and uses only finitely many algebraic coefficients, it lies in $E(t)[x; \sigma]$ for some finite extension $E/\mathbb{Q}$ containing E_0 , so enumeration eventually finds it. At that stage the right remainder is zero and we have found an exact factor of g .

Compute the exact cofactor by right division and recurse on both pieces. Since the x -degree drops at each recursive step, the process terminates and returns a complete factorization of f . $\square$

Theorem 13.2 is a decidability result, not an efficiency theorem: after algebraic closure of the constants, the irreducible-bound obstruction is no longer a division-algebra obstruction but only a constructive search problem. A more structured search using norm witnesses in the associated split cyclic algebra is also possible in principle (see Gräter and Weese (2004)).

13.3 Factoring in the BSS model over $\mathbb{C}$

The algebraically closed constant-field picture may also be interpreted in the Blum–Shub–Smale model over $\mathbb{C}$ (Blum et al., 1998). If one replaces $\overline{\mathbb{Q}}$ by $\mathbb{C}$, Theorem 13.1 still shows that an irreducible-bound input is reducible if and only if its degree in x is strictly larger than the X -degree of its bound. Thus bare BSS suffices to decide reducibility of irreducible-bound inputs. What bare BSS does *not* provide is exact algebraic choice. As emphasized by Feferman (2013) and Braverman (2005), exact root-selection functions are not available in the bare model. This obstruction is already visible in the constant-coefficient commutative subclass $\mathbb{C}[y] \subset \mathbb{C}(t)[y]$.

Proposition 13.3. *Let $\text{RC}_{\mathbb{C}} : \mathbb{C}[y] \setminus \mathbb{C} \rightarrow \mathbb{C}$ denote a univariate root-choice operation, meaning that for every nonconstant $q(y) \in \mathbb{C}[y]$ the value $\text{RC}_{\mathbb{C}}(q)$ is some $\lambda \in \mathbb{C}$ with $q(\lambda) = 0$. Then:*

- (1) *Any procedure that outputs a complete exact factorization of every commutative polynomial $p(y) \in \mathbb{C}(t)[y]$ computes $\text{RC}_{\mathbb{C}}$ by restriction to the constant-coefficient subclass $\mathbb{C}[y] \subset \mathbb{C}(t)[y]$.*
- (2) *$\text{RC}_{\mathbb{C}}$ is not computable in the bare BSS model over $\mathbb{C}$.*
- (3) *In BSS augmented by $\text{RC}_{\mathbb{C}}$, complete exact factorization of every polynomial $q(y) \in \mathbb{C}[y]$ is computable.*

Hence univariate root choice is already forced by the constant commutative subproblem inside factorization over $\mathbb{C}(t)$.

Proof. For (1), restrict any exact factor-output procedure for $\mathbb{C}(t)[y]$ to the subclass $\mathbb{C}[y]$: since $\mathbb{C}$ is algebraically closed, one reads off a linear factor $y - \lambda$ and hence a root of the input polynomial, computing $\text{RC}_{\mathbb{C}}$. For (2), if bare BSS computed $\text{RC}_{\mathbb{C}}$, then for fixed $d \geq 2$ the total map $r(z) := \text{RC}_{\mathbb{C}}(y^d - z)$ would satisfy $r(z)^d = z$ on $\mathbb{C}$. Along each halting path the output is a rational function in z , and because the constructible path domains cover $\mathbb{C}$, some path is taken on a cofinite subset. Hence $r(z) = R(z)$ there for some $R(z) \in \mathbb{C}(z)$, forcing $R(z)^d = z$ in $\mathbb{C}(z)$, impossible because z is not a d th power in $\mathbb{C}(z)$. For (3), one repeatedly applies $\text{RC}_{\mathbb{C}}$ to choose a root and divides by the corresponding linear factor until the degree is zero. $\square$

14 Open problems

Our algorithm over $K(t)$ and comparison with $\overline{\mathbb{Q}}(t)$ suggests the following questions.

- (O1) Give a direct algorithm for the generic irreducible-bound algebra over the univariate function field $F = K(\mathcal{T})[\mathcal{X}]/(\pi)$, thereby turning Proposition 11.2 into a full decidability theorem over $\mathbb{Q}(\omega)(t)$.
- (O2) Prove a Hilbert- or Chebotarev-type specialization theorem showing that finitely many good values a and finitely many good inert primes $\mathfrak{q}$ suffice to recover the generic factorization pattern of an irreducible-bound input.
- (O3) Develop effective recombination and reconstruction bounds for the two-level modular search, including a useful normalization theory for candidate factors over $K(t)$.
- (O4) Determine whether the exact decidability over $\overline{\mathbb{Q}}(t)$ can be turned into an effective descent method back to $\mathbb{Q}(\omega)(t)$.

15 Conclusion

The root-of-unity quantum plane considered here has a univariate central PID, so the bound-first approach is especially natural: compute the bound, factor it centrally, and use right gcds to split off all reducible-bound pieces. On the remaining irreducible-bound pieces, the practical strategy developed here is the two-level modular method through characteristic-zero specializations and finite inert primes.

The resulting method is exact on the reducible-bound stage and certifying on the irreducible-bound stage. What is still missing over $\mathbb{Q}(\omega)(t)$ is a complete algorithm for the generic irreducible-bound algebra over $K(\mathcal{T})[\mathcal{X}]/(\pi)$, together with quantitative control of the specialization, recombination, and reconstruction searches. After enlarging the constants to $\overline{\mathbb{Q}}$, Tsen's theorem removes that division-algebra obstruction and yields decidability in the standard exact algebraic model; over $\mathbb{C}(t)$, the corresponding BSS picture shows that exact factor output already forces a univariate root-choice primitive.

Bibliography

- Lenore Blum, Felipe Cucker, Michael Shub, and Steve Smale. *Complexity and Real Computation*. Springer, New York, 1998. doi:10.1007/978-1-4612-0701-6.
- Mark Braverman. On the complexity of real functions. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 155–164, Los Alamitos, CA, 2005. IEEE Computer Society. doi:10.1109/SFCS.2005.58.
- Xavier Caruso and J  r  my Le Borgne. A new faster algorithm for factoring skew polynomials over finite fields. *J. Symb. Comp.*, 79(2):411–443, 2017. doi:10.1016/j.jsc.2016.02.016.
- Romain Coulibaly and Kenneth L. Price. Factorization in quantum planes. *Missouri J. Mathematical Sciences*, 18(3):197–205, 2006. doi:10.35834/2006/1803197.
- Solomon Feferman. About and around computing over the reals. In B. Jack Copeland, Carl J. Posy, and Oron Shagrir, editors, *Computability: Turing, G  del, Church, and Beyond*, pages 55–76. MIT Press, Cambridge, MA, 2013.
- J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 3 edition, 2013.
- Mark Giesbrecht. Factoring in skew-polynomial rings over finite fields. *J. Symb. Comp.*, 26(4):463–486, 1998. doi:10.1006/jsc.1998.0224.
- Mark Giesbrecht and Yang Zhang. Factoring and decomposing Ore polynomials over $\mathbb{F}_q(t)$. In *Proceedings of the 2003 International Symposium on Symbolic and Algebraic Computation (ISSAC 2003)*, pages 127–134, New York, 2003. ACM. doi:10.1145/860854.860888.
- Philippe Gille and Tam  s Szamuely. *Central Simple Algebras and Galois Cohomology*. Cambridge University Press, Cambridge, 2006.
- Jos   Gomez-Torrecillas, F. J. Lobillo, and Gabriel Navarro. Factoring Ore polynomials over $\mathbb{F}_q(t)$ is difficult, 2015. doi:10.48550/arXiv.1505.07252.
- Jos   Gomez-Torrecillas, F. J. Lobillo, and Gabriel Navarro. Computing the bound of an Ore polynomial. applications to factorization. *J. Symb. Comp.*, 92:269–297, 2019. doi:10.1016/j.jsc.2018.04.018.
- J. Gr  ter and M. Weese. On the norm equation over function fields. *J. London Mathematical Society*, 69(3):575–592, 2004. doi:10.1112/S0024610704005290.
- Timo Hanke. The isomorphism problem for cyclic algebras and an application. In *Proceedings of the 2007 International Symposium on Symbolic and Algebraic Computation (ISSAC 2007)*, pages 181–186, New York, 2007. ACM. doi:10.1145/1277548.1277574.
- G  bor Ivanyos, Lajos R  nyai, and Josef Schicho. Splitting full matrix algebras over algebraic number fields. *J. Algebra*, 354(1):211–223, 2012. doi:10.1016/j.jalgebra.2012.01.008.
- N. Jacobson. *The Theory of Rings*. American Math. Soc., New York, 1943.
- Viktor Levandovskyy. Plural, a non-commutative extension of singular: Past, present and future. In *Mathematical Software – ICMS 2006*, volume 4151 of *Lecture Notes in Computer Science*, pages 144–157. Springer, 2006. doi:10.1007/11832225_13.
- Viktor Levandovskyy and Albert Heinle. A factorization algorithm for G -algebras and its applications. *J. Symb. Comp.*, 85:188–205, 2018. doi:10.1016/j.jsc.2017.06.005.
- Oystein Ore. Theory of non-commutative polynomials. *Annals of Mathematics*, 34(3):480–508, 1933.
- Susanne Pumpl  n and Daniel Thompson. The norm of a skew polynomial. *Algebras and Representation Theory*, 25:869–887, 2022. doi:10.1007/s10468-021-10051-z.
- Allan K. Steel. Computing with algebraically closed fields. *J. Symb. Comp.*, 45(3):342–372, 2010. doi:10.1016/j.jsc.2009.09.005.
- The Sage Developers. *SageMath, the Sage Mathematics Software System*, 2025. URL <https://www.sagemath.org>. doi:10.5281/zenodo.8042260.