Multivariate Ore Polynomials as a Primitive for Cryptographic Protocols

Reinhold Burger
supervised by George Labahn

Albert Heinle*
supervised by Mark Giesbrecht

Symbolic Computation Group – University of Waterloo

Introduction

Ore polynomials rings [4] acquired the attention of a large number of researchers, since they form algebraic abstractions of e.g. the differential and the shift operators, which are indispensable in many aspects of theoretical and applied science. Boucher et al.have presented a Diffie-Hellman like key exchange protocol based on the difficulty of factoring in univariate Ore polynomial rings over finite fields [1]. One weakness of their approach, among others, is that the rings they chose are Euclidean domains. We will present a way to construct Ore polynomials rings which are both feasible in terms of computability, as well as secure to be used in cryptographic protocols.

Ore Polynomials – A Crash Course

Let R be any ring, and σ be a ring endomorphism of R. Furthermore, let δ be a **σ-derivation** of R, i.e. an additive endomorphism on R where

$$\forall r, s \in R : \delta(rs) = \sigma(r)\delta(s) + \delta(r)s.$$

We can construct a polynomial ring $R[\partial; \sigma, \delta]$ with

- the usual addition rules for polynomials and
- the non-commutative relation $\partial r = \sigma(r)\partial + \delta(r)$.

We call this an **Ore polynomial ring over R**. This process can be iterated with different suitable σ and δ to construct **multivariate Ore polynomial rings**.

Identified Suitable Rings

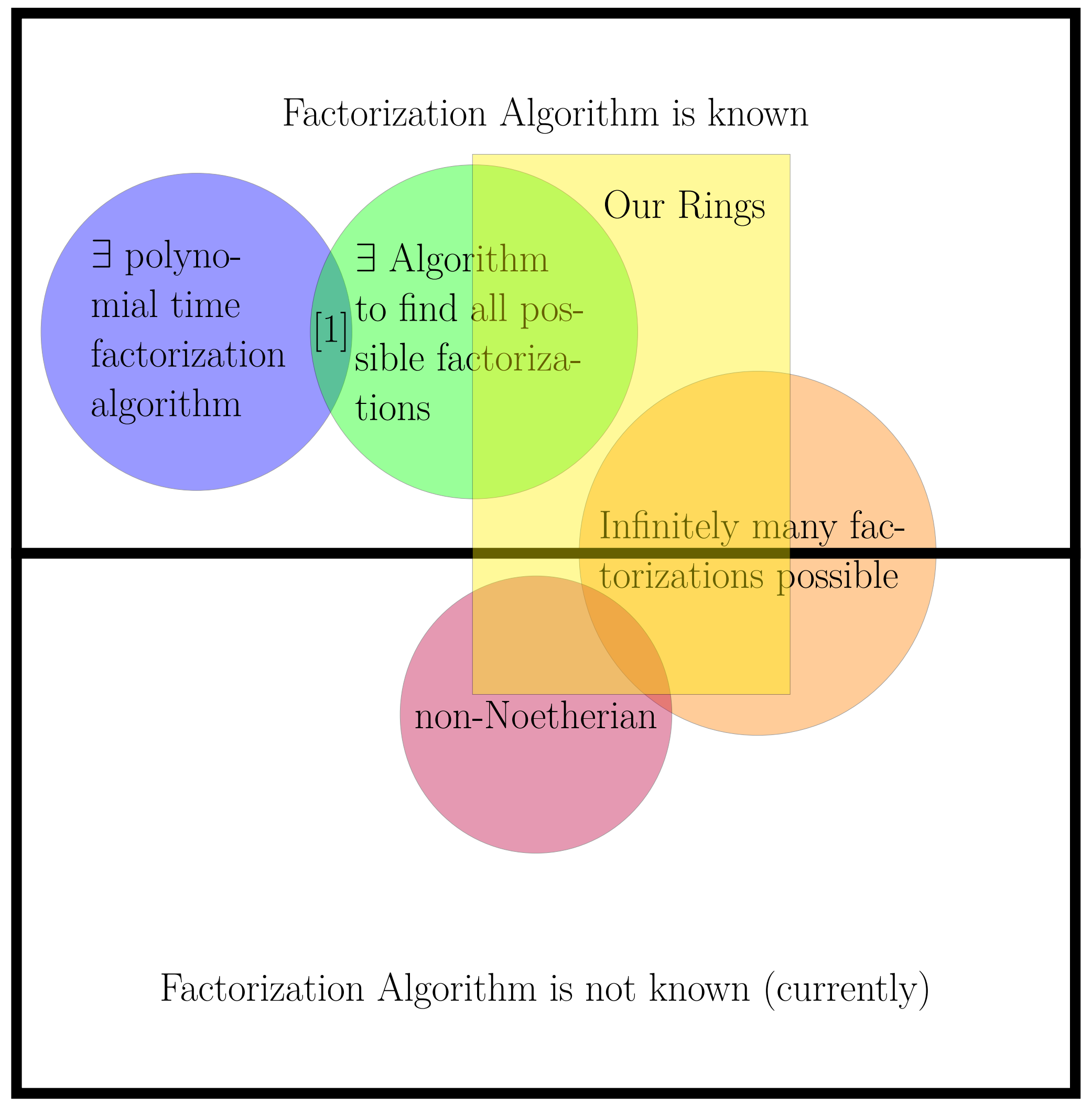
The rings that we identified as being suitable for use in cryptographic protocols are of the form

$$S := R[\partial_1; \sigma_1, \delta_1][\partial_2; \sigma_2, \delta_2] \dots [\partial_n; \sigma_n, \delta_n], \quad (1)$$

where $\mathbb{N} \ni n > 1$, R **is a domain with identity element**, and for all $i \in \{1, \dots, n\}$, **either σ_i is the identity map, or δ_i is the zero map**. These assumptions guarantee that

- arithmetics in S will have a polynomial time complexity and
- S is not a Euclidean domain.

Ore Polynomials – Status Quo


Factorization Algorithm is known
∃ polynomial time factorization algorithm
[1]
∃ Algorithm to find all possible factorizations
Our Rings
Infinitely many factorizations possible
non-Noetherian
Factorization Algorithm is not known (currently)

Proposed Key Exchange Protocol

```
1: $A$ and $B$ publicly agree on a ring $S$ of type (1), a security parameter $\nu \in \mathbb{N}$ representing the size of the elements to be picked from $S$ in terms of total degree and coefficients, a non-central element $L \in S$, and two multiplicatively closed, commutative subsets of $\mathcal{C}_l, \mathcal{C}_r \subset S$, whose elements do not commute with $L$.
2: $A$ chooses a tuple $(P_A, Q_A) \in \mathcal{C}_l \times \mathcal{C}_r$.
3: $B$ chooses a tuple $(P_B, Q_B) \in \mathcal{C}_l \times \mathcal{C}_r$.
4: $A$ sends the product $A_{\text{part}} := P_A \cdot L \cdot Q_A$ to $B$.
5: $B$ sends the product $B_{\text{part}} := P_B \cdot L \cdot Q_B$ to $A$.
6: $A$ computes $P_A \cdot B_{\text{part}} \cdot Q_A$.
7: $B$ computes $P_B \cdot A_{\text{part}} \cdot Q_B$.
8: $P_A \cdot P_B \cdot L \cdot Q_B \cdot Q_A = P_B \cdot P_A \cdot L \cdot Q_A \cdot Q_B$ is the shared secret key of $A$ and $B$.
```

In practice, we define the commuting subsets C_l, C_r as follows: Pick $P, Q \in S$, such that neither of these elements commutes with L. We furthermore assume that there exists a non-trivial $\tilde{R} \subsetneq R$, which lies in the center of S. Then

$$\mathcal{C}_l := \left\{ f(P) \mid f = \sum_{i=0}^{m} f_i X^i \in \tilde{R}[X], m \in \mathbb{N}, f_0 \neq 0 \right\},$$

$$\mathcal{C}_r := \left\{ f(Q) \mid f = \sum_{i=0}^{m} f_i X^i \in \tilde{R}[X], m \in \mathbb{N}, f_0 \neq 0 \right\} \quad (2)$$

fulfill our requirements.

Remark: Our primitive is not bound to this protocol, but can be applied to other cryptographic paradigms, e.g. a three-pass-protocol and a zero-knowledge-proof protocol [2].

Example

Let S be the third Weyl algebra A_3 over the finite field $\mathbb{F}_{71}$, upon which A and B agree. Let

$$L := 3x_2^2 - 5\partial_2^2 - x_2\partial_3 - x_3 - \partial_2,$$
$$P := -5x_3^2 - 2x_1\partial_3 + 34, \quad \text{and}$$
$$Q := x_2^2 + x_1x_3 - \partial_3^2 + \partial_3,$$

where L, P and Q are non-central. The polynomials P, Q define the sets $\mathcal{C}_l$ and $\mathcal{C}_r$ as in (2).
Suppose A chooses polynomials

$$f_A(X) = 48X^2 + 22X + 27, \quad g_A(X) = 58X^2 + 5X + 52,$$

while B chooses

$$f_B(X) = 3X^2 + X + 31, \quad g_B(X) = 24X^2 + 4X + 11.$$

Then the private tuples are

$$(P_A,\ Q_A) = (f_A(P),\ g_A(Q)) \text{ and } (P_B,\ Q_B) = (f_B(P),\ g_B(Q)).$$

As described in the protocol, A subsequently sends the product $A_{\text{part}} := P_A \cdot L \cdot Q_A$ to B, while B sends $B_{\text{part}} := P_B \cdot L \cdot Q_B$ to A, and their secret key is $P_A \cdot P_B \cdot L \cdot Q_B \cdot Q_A = P_B \cdot P_A \cdot L \cdot Q_A \cdot Q_B$.

Difficult Problem

Difficult Problem
Given:

- a ring S,
- a security parameter ν,
- two sets $\mathcal{C}_l, \mathcal{C}_r$ of multiplicatively closed, commutative subsets of S,
- $P_A \cdot L \cdot Q_A$ and $P_B \cdot L \cdot Q_B$ for some (P_A, Q_A), $(P_B, Q_B) \in \mathcal{C}_l \times \mathcal{C}_r$.

Compute

$$P_B \cdot P_A \cdot L \cdot Q_A \cdot Q_B \; (= P_A \cdot P_B \cdot L \cdot Q_B \cdot Q_A).$$

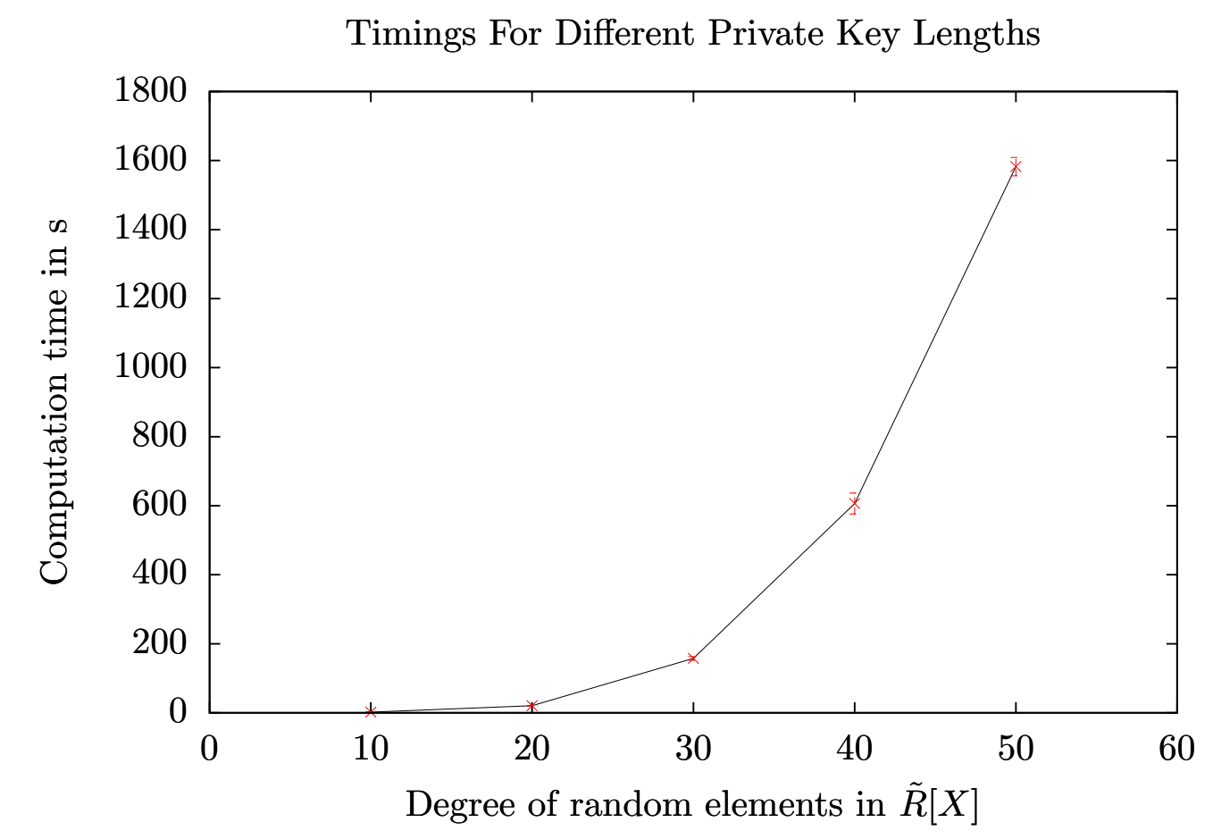
Ways of solving the difficult problem:

- Finding the correct factorization of the product $P_A \cdot L \cdot Q_A$ (resp. $P_B \cdot L \cdot Q_B$).
- Setting up an ansatz for the coefficients of P_A, Q_A (resp. P_B, Q_B) and solving a non-linear system of equations.
- Brute-Force.

All these problems are infeasible for large choices of $P_B, P_A, Q_B, Q_A \in S$ at the current state of research.
Remark: For a practical choice of the ring R, which is e.g. $R = \mathbb{F}_{2^k}$, we determined in [2] that key-sizes of about 700kB size (naive representation) lead to sufficient security. This is similar to secure key-sizes for the McEliece Cryptosystem [3].

Implementation

- Commodity computer algebra systems appeared to have too slow implementations of non-commutative rings. Hence we wrote our own experimental implementation in `C`[a].
- We used a naive, i.e. dense, representation of polynomials. The timings are promising.



- We created challenges for breaking the system[b].

What if...

Due to the connections between Ore polynomials and operator equations, someone breaking our system would lead to new insights about operator equations. This is a Win-Win situation.

References

[1] D. Boucher, P. Gaborit, W. Geiselmann, O. Ruatta, and F. Ulmer.
Key Exchange and Encryption Schemes Based on Non-Commutative Skew Polynomials.
In *Post-Quantum Cryptography*, pages 126–141. Springer, 2010.

[2] R. Burger and A. Heinle.
A Diffie-Hellman-like Key Exchange Protocol Based on Multivariate Ore Polynomials.
arXiv preprint arXiv:1407.1270, 2014.

[3] R. J. McEliece.
A Public-Key Cryptosystem Based on Algebraic Coding Theory.
DSN progress report, 42(44):114–116, 1978.

[4] O. Ore.
Theory of Non-Commutative Polynomials.
Annals of mathematics, 34:480–508, 1933.

Acknowledgements

We thank Konstantin Ziegler for his valuable remarks, comments and suggestions on our paper [2].
We are grateful to Alfred Menezes for the fruitful discussion we had with him.

[a] `https://github.com/ioah86/diffieHellmanNonCommutative`
[b] `https://cs.uwaterloo.ca/~aheinle/miscellaneous.html`