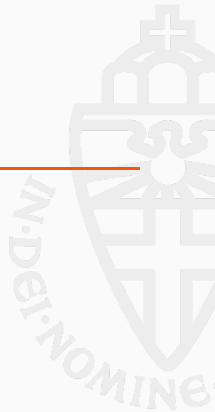




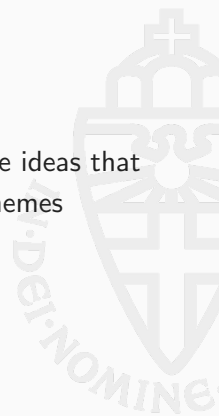
Common tools for modeling cryptographic problems algebraically

Lars Ran & Monika Trimoska

July 1, 2025, Summer School on real-world crypto and privacy



In modeling cryptographic problems as systems of equations, there are some ideas that are greatly re-usable and occur in multiple different attacks on different schemes

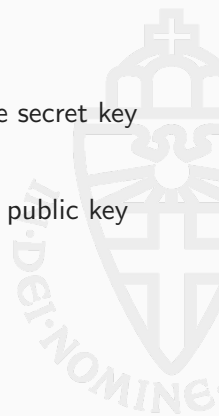


Recap of our goal



Given a public-key cryptographic scheme we would either like to recover the secret key or be able to forge signatures

We can often model this as a polynomial system of equations, whereby the public key defines the coefficients of the equations



Guessing variables



Suppose that we have a polynomial system with N solutions, all leading to a valid key / signature



Suppose that we have a polynomial system with N solutions, all leading to a valid key / signature

In FXL we already saw that we can enumerate k variables to increase the speed of solving the algebraic system at the cost of having multiple iterations



Systems with multiple solutions

Suppose that we have a polynomial system with N solutions, all leading to a valid key / signature

In FXL we already saw that we can enumerate k variables to increase the speed of solving the algebraic system at the cost of having multiple iterations

If we assume that the N solutions are uniformly distributed, we can do the same but this time with fewer iterations

$$\frac{q^k}{N}$$

Instead of guessing and/or enumerating a single variable, we can guess a linear constraint with the same probability

For example, for some λ_i and c we can guess

$$\lambda_1 x_1 + \cdots + \lambda_n x_n = c$$



Consider the system

$$\begin{aligned}(x + 5y + 7z)(3x + 2y) + 4x + 5y + 9z + 2 \\ (5x + 6y + 7z)(2x + 8y + 3z) + 1x + 9y + 6z + 3\end{aligned}$$



Consider the system

$$\begin{aligned}(x + 5y + 7z)(3x + 2y) + 4x + 5y + 9z + 2 \\ (5x + 6y + 7z)(2x + 8y + 3z) + 1x + 9y + 6z + 3\end{aligned}$$

If we guess $x + 5y + 7z = c$ we obtain another linear equation for free

$$(4 + 3c)x + (5 + 2c)y + 9z = 2$$

This makes the system easy to solve, since we now have 2 linear equations and 1 quadratic in 3 variables



Some examples where these degree drops occur in practice:

- ▶ Underdetermined MQ
- ▶ MQ-Sign
- ▶ PowAff2 (Biscuit)



Finite coefficient set equations



Consider the following trivial system for a secret $x \in \mathbb{F}_2$

$$x^6 + x^5 + x^3 + x^2 + x + 1$$



Consider the following trivial system for a secret $x \in \mathbb{F}_2$

$$x^6 + x^5 + x^3 + x^2 + x + 1$$

Since we know $x \in \mathbb{F}_2$ we know $x(x - 1) = 0$ and we can reduce the system to

$$x + 1$$



Consider the following trivial system for a secret $x \in \mathbb{F}_2$

$$x^6 + x^5 + x^3 + x^2 + x + 1$$

Since we know $x \in \mathbb{F}_2$ we know $x(x - 1) = 0$ and we can reduce the system to

$$x + 1$$

If we are looking for solutions in the finite field \mathbb{F}_q we can add the equations

$$x_i^q - x_i = 0 \quad \forall i$$

In general, if we know all, or some, variables x_i are members of a certain subset $S \subset \mathbb{F}_q$ we can introduce the equations

$$\prod_{s \in S} (x_i - s) = 0 \quad \forall i$$

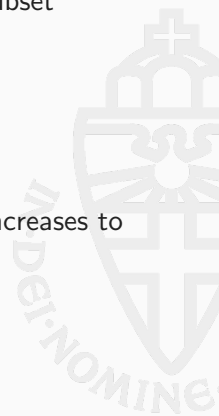


In general, if we know all, or some, variables x_i are members of a certain subset $S \subset \mathbb{F}_q$ we can introduce the equations

$$\prod_{s \in S} (x_i - s) = 0 \quad \forall i$$

Furthermore, when we start enumerating with FXL, our guess probability increases to

$$\frac{N}{|S|^k}$$



This does not apply well if q or S is too large. But there are plenty of examples where it works well

- ▶ General system solving over $\mathbb{F}_2, \mathbb{F}_3$
- ▶ Arora-Ge attack against LWE
- ▶ R-SDP (Cross)



Substituting variables



Consider a system of equations containing a linear equation $\sum_i \lambda_i x_i = c$. Then we can write

$$x_1 = \lambda_1^{-1} \left(c - \sum_{i>1} \lambda_i x_i \right)$$



Consider a system of equations containing a linear equation $\sum_i \lambda_i x_i = c$. Then we can write

$$x_1 = \lambda_1^{-1} \left(c - \sum_{i>1} \lambda_i x_i \right)$$

If we now substitute this into our other equations, then we remove one variable from our system

More importantly, the degrees of our other equations do not increase!

Can we do something similar with non-linear equations?

If we have a subset of variables $y_1, \dots, y_{n'}$ that only occurs linearly in each equation we can do something similar. Consider (over \mathbb{F}_7)

$$3x_1^2 + 5x_2^2 + 4x_1x_2 + 2x_1 + 3x_2 = 6y_1 + 4y_2$$

$$6x_1^2 + x_2^2 + 2x_1x_2 + x_1 = 2y_1 + y_2$$

$$x_1^2 + 4x_2^2 + 2x_1x_2 + 5x_1 + 2x_2 = 3y_1 + 5y_2$$



Can we do something similar with non-linear equations?

If we have a subset of variables $y_1, \dots, y_{n'}$ that only occurs linearly in each equation we can do something similar. Consider (over \mathbb{F}_7)

$$3x_1^2 + 5x_2^2 + 4x_1x_2 + 2x_1 + 3x_2 = 6y_1 + 4y_2$$

$$6x_1^2 + x_2^2 + 2x_1x_2 + x_1 = 2y_1 + y_2$$

$$x_1^2 + 4x_2^2 + 2x_1x_2 + 5x_1 + 2x_2 = 3y_1 + 5y_2$$

We can now reduce the y variables and obtain

$$2x_1^2 + 5x_2^2 + x_1x_2 + 6x_1 + 4x_2 = 0$$



Can we do something similar with non-linear equations?

If we have a subset of variables $y_1, \dots, y_{n'}$ that only occurs linearly in each equation we can do something similar. Consider (over \mathbb{F}_7)

$$3x_1^2 + 5x_2^2 + 4x_1x_2 + 2x_1 + 3x_2 = 6y_1 + 4y_2$$

$$6x_1^2 + x_2^2 + 2x_1x_2 + x_1 = 2y_1 + y_2$$

$$x_1^2 + 4x_2^2 + 2x_1x_2 + 5x_1 + 2x_2 = 3y_1 + 5y_2$$

We can now reduce the y variables and obtain

$$2x_1^2 + 5x_2^2 + x_1x_2 + 6x_1 + 4x_2 = 0$$

This is relevant, for example, in the model Monika presented yesterday

$$\mathbf{A}^{-1}\mathbf{D}_i = \mathbf{C}_i\mathbf{B}$$

Some examples where this is applied in practice

- ▶ Underdetermined MQ
- ▶ Matrix code equivalence
- ▶ Alternating Trilinear form equivalence



Bilinear systems



Consider a system in the variables x_1, x_2, x_3, y_1, y_2 such as

$$3x_1y_1 + 2x_1y_2 + 6x_2y_1 + 4x_2y_2 + x_3y_1 + 5x_3y_2 = 0$$

$$x_1y_1 + 3x_1y_2 + 2x_2y_1 + 6x_2y_2 + 4x_3y_1 + x_3y_2 = 0$$



Consider a system in the variables x_1, x_2, x_3, y_1, y_2 such as

$$3x_1y_1 + 2x_1y_2 + 6x_2y_1 + 4x_2y_2 + x_3y_1 + 5x_3y_2 = 0$$

$$x_1y_1 + 3x_1y_2 + 2x_2y_1 + 6x_2y_2 + 4x_3y_1 + x_3y_2 = 0$$

This is a bilinear system!

When constructing the Macaulay matrix we expect empty columns and a block-like structure

We can use this to our advantage and create smaller Macaulay matrices

We might have a system f_1, \dots, f_m that is bilinear in the x_1, \dots, x_n and $y_1, \dots, y_{n'}$ variables

We can multiply the polynomials by monomials of bi-degree (a, b) to get a polynomial with monomials in bi-degree $(a + 1, b + 1)$

This is a subset of the monomials of degree $a + 1 + b + 1$, so our Macaulay matrix is a lot smaller!

Some examples where this is applied in practice

- ▶ MinRank
- ▶ Rainbow Band separation
- ▶ Matrix code equivalence



Precisely compute the guessing probability for the best trade-off

Add the finite field equations in small fields

Remove linear-only variables

Use structure in your equations to your advantage!



Thanks for listening!

