Selected Areas in Cryptology - Part 1: Post-quantum cryptography

Exercise sheet 1, 12 February 2024

1. Write a recursive depth-first traversal (DFT) algorithm that enumerates all possible assignments to a solution vector $\mathbf{x} = (x_1, \ldots, x_n)$. The algorithm should assign values to variables with an instruction such as $x_1 \leftarrow 1$ and print out $\mathbf{x}$ on each leaf. This exercise can be done on paper, in Magma (by completing the demo script, for instance), SageMath, or any language of your choice, but without using language-specific syntax or functions.

2. Argument why DFT exhaustive search and Gray-code exhaustive search do not have the same complexity when we account for polynomial factors.

   Hint: For instance, by calculating both and comparing them, plus explaining where the difference comes from.

3. Consider a bilinear system of m equations in (s, p) variables: $\mathbf{x} = (x_1, \ldots, x_s)$ and $\mathbf{y} = (y_1, \ldots, y_p)$. See an example below with $s = 2$, $p = 3$ and $m = 5$.

$$\begin{aligned}
&x_1y_1 + x_2y_1 + x_1y_2 + x_2y_2 + x_2y_3 = 0\\
&x_2y_1 + x_2y_2 + x_1y_3 + x_2 + 1 = 0\\
&x_1y_1 + x_2y_1 + x_2y_3 = 0\\
&x_2y_1 + x_1y_2 + x_1 + x_2 = 0\\
&x_1y_1 + x_2y_2 + x_1y_3 + x_2 + 1 = 0
\end{aligned}$$

   (a) How would you perform an (almost) exhaustive search on a bilinear system, and what is the complexity of your approach?

   (b) How would you modify the *Simple* algorithm to take advantage of the bilinearity? What is the complexity in this case?

4. What is the number of monomials of degree at most D in a system of equations

   (a) defined over $\mathbb{F}_2$;

(b) defined over $\mathbb{F}_q$, with $q > 2$;

5. What is the degree of regularity of a system of $m = 6$ equations in $n = 4$ variables?

   Hint: To check your answer, you can compute the *Hilbert series* in SageMath as follows

```
R.<t> = PowerSeriesRing(ZZ)
hs = ((1-t^2)^(m)) / (1-t)^(n)
print(hs)
```

   In the following, you should use this code any time you need to find the degree of regularity.

6. If the system in the previous exercise is solved using a Gröbner-based algorithm, how big is the Macaulay matrix that we build, in MB?

   (a) Answer the same question for a system of $m = 146$ equations in $n = 73$ variables.

7. Let us consider a system over $\mathbb{F}_3$ of m equations in n variables, where $m = 2n$.

   (a) What is minimum number of variables for which a Gröbner-based algorithm outperforms the *Simple* algorithm, ignoring the memory requirements and polynomial factors in the complexity?

   (b) Answer the same question for a system over $\mathbb{F}_7$.

   (c) If we do take into account the memory requirements of the algorithms and compare their performance starting from a fixed budget for the overall attack, would the value of n increase or decrease in your answer?