Crypto Engineering
Finite fields extensions

2020-09-22

Exercise 1: AES field

Most of the elementary operations used in the definition of the AES block cipher are defined over $\mathbb{F}_{2^8}$, represented as $\mathbb{F}_2[X]/X^8 + X^4 + X^3 + X + 1$.

We define the following C function:

```
uint8_t xtime(uint8_t a)
{
        uint8_t m = a & 0x80 ? 0x1B : 0;

        return ((a << 1) ^ m);
}
```

Q.1: What does this function do?

Q.2: Write your own variant of `xtime` for a different representation of $\mathbb{F}_{2^8}$ (for instance using the polynomial $X^8 + X^6 + X^5 + X^4 + X^3 + X + 1$, which is irreducible over $\mathbb{F}_2[X]$).

Q.3: Write a multiplication function `mul8` for the AES representation of $\mathbb{F}_{2^8}$.

Exercise 2: Multiplication by a constant in $\mathbb{F}_{2^8}$

Let $P = \sum_{i=0}^{7} p_i X^i$ be an arbitrary polynomial of $\mathbb{F}_2[X]$ of degree < 8.

Q.1: Compute (symbolically) the result of the multiplication of P by X modulo $Q := X^8 + X^4 + X^3 + X + 1$.

Q.2: Considering that P can be written as a row vector $\begin{pmatrix} p_0 & \cdots & p_7 \end{pmatrix}$ of $\mathbb{F}_2^8$, write the multiplication of the previous question as a vector-matrix product and give the matrix $\boldsymbol{M}_{\texttt{0x2}}$ of the right multiplication by X modulo Q.

Remark. $\boldsymbol{M}_{\texttt{0x2}}$ is called the *companion matrix* of Q

Q.3: Compute $\boldsymbol{M}_{\texttt{0x4}} := \boldsymbol{M}_{\texttt{0x2}}^2$ and $\boldsymbol{M}_{\texttt{0x8}} := \boldsymbol{M}_{\texttt{0x2}}^3$. What is $\boldsymbol{M}_{\texttt{0xB}}$, the matrix of the right multiplication by $X^3 + X + 1$ modulo Q?

Exercise 3: Artin-Schreier extension towers ⋆

The goal of this exercise is to define a multiplication algorithm for elements of $\mathbb{F}_{2^{2^n}}$ built from a recursive *Artin-Schreier extension tower* as $\mathbb{F}_{2^{2^n}} \cong \mathbb{F}_2[x_1, \ldots, x_n]/\langle x_i^2 + x_i + \prod_{j<i} x_j \rangle_{1 \le i \le n}$. It can be shown that for all $n \ge 1$ the *Artin-Schreier polynomial* $x_n^2 + x_n + \prod_{j<n} x_j$ is irreducible over $\mathbb{F}_2[x_1, \ldots, x_n]/\langle x_i^2 + x_i + \prod_{j<i} x_j \rangle_{1 \le i \le n-1}$ (where we take the convention that for $n = 1$ the empty product equals 1), so one can build an extension of degree 2 of $\mathbb{F}_{2^{2^{n-1}}} \cong \mathbb{F}_2[x_1, \ldots, x_{n-1}]/\langle x_i^2 + x_i + \prod_{j<i} x_j \rangle_{1 \le i \le n-1}$ by adding one indeterminate x_n and the corresponding polynomial $x_n^2 + x_n + \prod_{j<n} x_j$ to the quotienting ideal.

In the following we only consider fields represented using the above extension tower.

Q.1:

1. How can you concisely represent elements of $\mathbb{F}_{2^{2^n}}$ as vectors of $\mathbb{F}_2^{2^n}$?
2. Give the vector corresponding to $x_1 + x_2 + x_1x_3 + x_2x_3$ when $n = 3$. Same question for $n = 4$.
3. How can you add together two elements using this embedding? Is this easy to implement on a typical CPU, when vectors are mapped to bit strings?

Q.2: Show how to compute the multiplication of two elements of $p, q \in \mathbb{F}_{2^{2^n}}$ from four* multiplications and one *Nim transform* in $\mathbb{F}_{2^{2^{n-1}}}$ by writing them as $p = p_0 + x_np_1$, $q = q_0 + x_nq_1$, $p_0, p_1, q_0, q_1 \in \mathbb{F}_{2^{2^{n-1}}}$, where the Nim transform is the linear mapping $\mathrm{NT} : \mathbb{F}_{2^{2^n}} \cong \mathbb{F}_2[x_1, \ldots, x_n]/\langle x_i^2 + x_i + \prod_{j<i} x_j \rangle_{1 \le i \le n} \to \mathbb{F}_{2^{2^n}}$, $p \mapsto p \cdot x_1 \ldots x_n$.

Q.3:

1. Show how to compute the Nim transform over $\mathbb{F}_{2^{2^n}}$ recursively from Nim transforms over $\mathbb{F}_{2^{2^{n-1}}}$.
2. Using the same embedding as in **Q.1**, what is the (recursive) expression of the Nim transform as a matrix? (That is, express the matrix $\boldsymbol{A}_n$ of the Nim transform over $\mathbb{F}_{2^{2^n}}$ as a block matrix in function of $\boldsymbol{A}_{n-1}$, where $\boldsymbol{A}_0 := \big[0\big]$.)

Q.4: What is the complexity of this multiplication algorithm in $\mathbb{F}_{2^{2^n}}$ (using either the schoolbook or the Karatsuba algorithm in **Q.2**)? How does this compare with the addition?

Hint: Use the "Master theorem" to analyse the recursivity (`https://en.wikipedia.org/wiki/Master_theorem_(analysis_of_algorithms)`).

Remark. Artin-Schreier extension towers play an important role (among others) in additive Fast Fourier Transform algorithms (Cantor, 1989, etc.), especially useful in characteristic two. Conway also used the above tower over $\mathbb{F}_2$ to define "Nim arithmetic" over the integers (and beyond); notably this allows to endow $\mathbb{N}$ with a field structure.

*Or three when using Karatsuba's algorithm.