Arithmetic in a recursive representation

April 27, 2021

Grading

This homework is graded as the *contrôle continu* of this course. You must send a written report (in a portable format) **detailing** your answers to the questions by the end of April, (2021-04-30T23:59+0200) to:

pierre.karpman@univ-grenoble-alpes.fr.

Unique exercise

The goal of this homework is to design a multiplication (and addition) algorithm for elements of $\mathbb{F}_{2^{2^n}}$ built from a recursive *Artin-Schreier extension tower* as:

$$\mathbb{F}_{2^{2^n}} \cong \mathbb{F}_2[X_1, \ldots, X_n]/\langle X_i^2 + X_i + \prod_{j<i} X_j \rangle_{1 \leqslant i \leqslant n}$$

The algorithm will itself be recursive, which means that we will reduce the multiplication in $\mathbb{F}_{2^{2^n}}$ to multiplication in $\mathbb{F}_{2^{2^{n-1}}}$, etc..

It can be shown that for all $n \geqslant 1$ the *Artin-Schreier polynomial* $X_n^2 + X_n + \prod_{j<n} X_j$ is irreducible in $\mathbb{F}_2[X_1, \ldots, X_n]/\langle X_i^2 + X_i + \prod_{j<i} X_j \rangle_{1 \leqslant i \leqslant n-1}$ (where we take the convention that for $n = 1$ the empty product equals 1), so one can build an extension of degree 2 of $\mathbb{F}_{2^{2^{n-1}}} \cong \mathbb{F}_2[X_1, \ldots, X_{n-1}]/\langle X_i^2 + X_i + \prod_{j<i} X_j \rangle_{1 \leqslant i \leqslant n-1}$ by adding one indeterminate X_n and the corresponding polynomial $X_n^2 + X_n + \prod_{j<n} X_j$ to the generators of the quotienting ideal.

Design an addition and multiplication algorithm for the above extension tower, and analyse their cost. More precisely, answer to the following:

1. How can you concisely represent "reduced" elements of $\mathbb{F}_2[X_1, \ldots, X_n]/\langle X_i^2 + X_i + \prod_{j<i} X_j \rangle_{1 \leqslant i \leqslant n}$ (*i.e.* polynomials in n variables where the degree in each indeterminate X_i is at most 1) 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 of $\mathbb{F}_{2^{2^n}}$ using this embedding? Is this easy to implement on a typical CPU, when vectors are mapped to binary strings?

4. Show how to compute the multiplication of two elements $P, Q \in \mathbb{F}_{2^{2^n}}$ from four* multiplications and one *Nim transform* in $\mathbb{F}_{2^{2^{n-1}}}$ by writing them as $P = P_0 + X_n P_1$, $Q = Q_0 + X_n Q_1$, $P_0, P_1, Q_0, Q_1 \in \mathbb{F}_{2^{2^{n-1}}}$, where the Nim transform in $\mathbb{F}_{2^{2^n}}$ is the linear mapping $\mathrm{NT} : \mathbb{F}_{2^{2^n}} \cong \mathbb{F}_2[X_1, \ldots, X_n]/\langle X_i^2 + X_i + \prod_{j<i} X_j \rangle_{1 \leqslant i \leqslant n} \to \mathbb{F}_{2^{2^n}}$, $P \mapsto P \cdot X_1 \cdots X_n$.

*Or three when using Karatsuba's algorithm.

5. Show how to recursively compute the Nim transform in $\mathbb{F}_{2^{2^n}}$ from Nim transforms in $\mathbb{F}_{2^{2^{n-1}}}$.

6. Using the same embedding into $\mathbb{F}_2^{2^n}$ as above, what is the (recursive) expression of the Nim transform as a matrix? (That is, express the matrix $\boldsymbol{A}_n$ of the Nim transform in $\mathbb{F}_{2^{2^n}}$ as a block matrix in function of $\boldsymbol{A}_{n-1}$, where $\boldsymbol{A}_0 := \big[0\big]$.)

7. What is the cost of this multiplication algorithm in $\mathbb{F}_{2^{2^n}}$ (using either the schoolbook or the Karatsuba algorithm in the above)? 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 to define "Nim arithmetic" over the integers (and beyond); notably this allows to endow $\mathbb{N}$ with a field structure.