MPRI 2.13.2 - Error-Correcting Codes and Cryptographic Applications

Final exam — 02/03/2017

You have 1h30. Any document including personal lecture notes is authorized.
You can answer either in French or in English.

General questions

1. **First line in the introduction:** explain why the know-plaintext attacks are the only attack scenario considered by the authors.
2. **Page 183:** It is assumed that $p < 0.5$. Explain how the attack must be modified if $p > 0.5$.
3. **Page 183:** Explain why the complexity of the classical algorithm (by Siegenthaler) is $\mathcal{O}\left(2^l \cdot l/C(p)\right)$.

On the definition of the underlying code

4. **Page 183:** It is assumed that the dimension of the code that must be decoded is equal to l, the degree of the LFSR feedback polynomial. Explain why the case where the dimension of the code is strictly less than the degree of the LFSR feedback polynomial is not relevant to practical applications.
5. **Page 184:** The authors claim that

$$h_i(D) = D^{i-1} \bmod g(D), \text{ for } 1 \leq i \leq N\ , \tag{E}$$

where g is the feedback polynomial of the target LFSR.

- Deduce from (E) the coefficients of h_{l+1} in terms of the feedback coefficients $c_0, \ldots, c_l$.
- Compare the value of x_{l+1} given by (2) with the first equation in (1), and deduce why the formula $h_i(D) = D^{i-1} \bmod g(D)$ is wrong.
- Modify (E) to get the correct expression of h_i, and give a proof of the modified formula.

On the new fast correlation attack

6. **Page 187, Line 11:** Why do the authors write that the length of the new code needs to be larger than the critical length n_0?
7. **Page 190, Lines 15-16**: Explain why the authors claim that the calculation of all parity checks for the $\mathcal{C}_2$ code is of order $\mathcal{O}(N \log N)$. Can you think of a method for improving this time complexity?
8. **Page 188, Algorithm 2:** Give the expression of the time complexity of the precomputation step for any t. Discuss the result.
9. **Page 193:** The authors compare two versions of the algorithm, with parameters $(t = 2, k = 23)$ and with parameters $(t = 3, k = 20)$. They write that the price of using the version with $t = 3$ is the increase in the precomputation complexity. Can you see another drawback of this version?

Potential improvements

10. **Precomputation step:** For $t \geq 5$, describe an algorithm for the precomputation step of Algorithm 2, which may offer a better trade-off between the time complexity and the memory.
11. **Decoding step:** Propose a faster algorithm for the decoding step of Algorithm 2, and discuss the pros and the cons compared to the algorithm presented in the paper.