Advanced cryptology (GBX9SY06)
*
Information set decoding project

Pierre Karpman

2018-12

1 Description

The goal of this project is to implement an information set decoding algorithm to search for low-weight codewords of *binary* codes, using the SageMath software (https://www.sagemath.org/). There is no particular requirement as to which algorithm to implement, but overall efficiency will be rewarded.

Performance evaluation

Your submission will be evaluated on a series of codes (defined below) on version 8.2 of SageMath running on the "Retour d'Est" (retourdest.imag.fr) server. This is a single-processor system with 256 GB of memory and running an Intel(R) Xeon(R) Gold 6126 CPU @ 2.60GHz, that features 12 (resp. 24) physical (resp. logical) cores. You are encouraged to take advantage of this architecture, for instance by developping a multi-threaded implementation.

The codes used for evaluation are the following, all defined in the provided file https://www-ljk.imag.fr/membres/Pierre.Karpman/cry_adv2018_codes_desc.sage.

Known codes.

— A binary BCH code of parameters $[511, 448, 15]$. The evaluation criterion will be the ability to find a minimum-weight codeword. This should take (far) less than a minute.

— A binary BCH code of parameters $[511, 421, 21]$. The evaluation criterion will be the average time (over three runs) needed to find a minimum-weight codeword. This may take some time, but an efficient implementation should need (far) less than an hour on a single thread.

— A binary BCH code of parameters $[511, 385, 29]$. The evaluation criterion will be the smallest weight of a found codeword after one run of at most 30 minutes. A bonus will be awarded if a word of weight 30 is found. An extra bonus will be awarded if a word of weight 29 is found.

Random codes.

— A random code of parameters $[768, 384]$. The evaluation criterion will be the smallest weight of a found codeword after one run of at most 15 minutes.

— A random code of parameters $[2048, 128]$. The evaluation criterion will be the smallest weight of a found codeword after one run of at most 15 minutes.

You are advised to use a slightly verbose implementation, that for instance prints the time required to find a codeword with a new lowest weight.

2 Instructions

Working in teams of two is allowed and encouraged. You must send your implementation to:

pierre.karpman@univ-grenoble-alpes.fr

by Monday 2019-01-14T23:59+0100. No separate report is required, but your code must be well-commented and include a small textual description of your algorithmic and implementation choices. If your algorithm is parameterized, be sure to mention which parameters to use for every one of the evaluation codes. You are also advised to provide some numbers about your own performance evaluation of your implementation. Finally, you must prepare a short ($\sim$ 10 minutes for a single-person team; $\sim 2 \times 7$ minutes for a two-person one) presentation of your work for Thursday 2019-01-17.

References

[CC98] Anne Canteaut and Florent Chabaud. A New Algorithm for Finding Minimum-Weight Words in a Linear Code: Application to McEliece's Cryptosystem and to Narrow-Sense BCH Codes of Length 511. *IEEE Trans. Information Theory*, 44(1):367–378, 1998.