INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 15, Issue: 5, Month: October, Year: 2020
Article Number: 3920, https://doi.org/10.15837/ijccc.2020.5.3920

Solving a Modified Syndrome Decoding Problem using Integer Programming

V.-F. Drăgoi, P.-L. Cayrel, B. Colombier, D. Bucerzan, S. Hoară

Vlad-Florin Drăgoi*
Aurel Vlaicu University of Arad
310330 Arad, Elena Dragoi, 2, Romania
*Corresponding author: vlad.dragoi@uav.ro

Pierre-Louis Cayrel
Univ Lyon, UJM-Saint-Etienne, CNRS, Laboratoire Hubert Curien UMR 5516
F-42023, SAINT-ETIENNE, France
pierre.louis.cayrel@univ-st-etienne.fr

Brice Colombier
Univ Lyon, UJM-Saint-Etienne, CNRS, Laboratoire Hubert Curien UMR 5516
F-42023, SAINT-ETIENNE, France
b.colombier@univ-st-etienne.fr

Dominic Bucerzan
Aurel Vlaicu University of Arad
310330 Arad, Elena Dragoi, 2, Romania
dominic.bucerzan@uav.ro

Sorin Hoară
IOSUD, Department of Computer Science Politehnica University of Timisoara
Vasile Parvan Street, No. 2 300223 Timisoara, Romania
sorin-horatiu.hoara@student.upt.ro

Abstract

In this article, we model a variant of the well-known syndrome decoding problem as a linear optimization problem. Most common algorithms used for solving optimization problems, e.g. the simplex algorithm, fail to find a valid solution for the syndrome decoding problem over a finite field. However, our simulations prove that a slightly modified version of the syndrome decoding problem can be solved by the simplex algorithm. More precisely, the algorithm returns a valid error vector when the syndrome vector is an integer vector, i.e.,the matrix-vector multiplication, is realized over $\mathbb{Z}$, instead of $\mathbb{F}_q$.

Keywords: Syndrome decoding, integer linear programming, simplex algorithm.

1 Introduction

In 1978, the Coset Weight Problem (CWP) and the Subspace Weight Problem (SWP), usually known by the name of Syndrome Decoding Problem (SDP), have been proven to be $\mathcal{NP}$-complete for random binary codes by Berlekamp, McEliece and van Tilborg [6]. The hardness of those two problems is central in code-based cryptography. In fact, the security of certain cryptographic applications like the public-key cryptosystems of McEliece [32] and Niederreiter [39] relies directly on the difficulty of solving an instance of SDP. The idea of McEliece had a great succes in the code-based cryptographic community, as a plethora of variants were proposed. Mainly aiming to reduce the key size as well as to have tight security reductions, these variantions consisted in changing the private code or adding additional structure. In Table 1 we present a non-exhaustive list of candidates with references, as well as existing/possible attacks. SDP is not only present in public-key encryptiion schemes (PKE), it is also a key ingredient for code-based hash functions [1], as well as code-based signature schemes [12].

Table 1: McEliece variants and structural attacks

Code family	PKE scheme	Attacks	Variants on sub-codes	Attacks on sub-codes	Wieschebrink's technique	Attacks on Wieshbrink's tch.	Weak keys
Goppa	[32]						[30]
GRS	[39]	[48]	[5]	[51, 52]	[50]	[13]	
Reed-Muller	[47]	[11, 33]			[23]	[40]	
Concatenated	[44]	[45]					
Algebraic Geometry	[25]	[14, 20, 48]	[14]	[14]			
LDPC	[36]	[36]					
QC-LDPC	[2]	[41]					
Wild Goppa	[7, 8]	[15, 16]					
Srivastava	[42]	[19]					
Convolutional	[31]	[28]					
MDPC	[34]				[37][1]	[17]	[4]
Polar	[46]	[3, 18]	[24]				

Integer Linear Programming (ILP) has been widely used in the literature to reformulate and solve various economic, logistic, computer science and many more real life problems [22, 43, 49]. It was also used in cryptographic context, mainly for the study of stream ciphers. In [38] Mixed Integer Linear Programming (MILP) is used to prove security bounds against both differential and linear cryptanalysis. The authors in [38] implement MILP-based methods to obtain practical results for Enocoro-128v2, as well as for calculating the number of active S-boxes for AES. In [9, 10] the authors illustrate how MILP can be used in the case of the Trivium stream cipher, and the lightweight block cipher Ktantan. For doing so, the authors in [10] reformulate a non-linear multivariate Boolean equation system into a MILP problem and use the CPLEX solver to find a solution.

Algorithms for solving ILP problems (ILPP) have been developed for more than 40 years. The well-known book in Discrete Optimizations by [26] offer a detailed view of the major techniques, e.g., polyhedral or cutting-planes methods, Chvátal-Gomory cuts, lattice basis reductions, and many others. Even though ILPPs with binary constrains are $\mathcal{NP}$-hard in general [27], there are several particular instances which are solvable in polynomial time, such as totally uni-modular matrices [35], or fixed number of variables [29]. For a recent review on the evolution of ILPPs see [21]. We will see that in our case the ILPP does not fall in any of these cases. However, we obtain practical results that point out to a polynomial time complexity.

Our contribution We show, in this article, that if the SDP is modified, it becomes in practice much easier to solve. Our main contribution is to formalise a modified variant of the SDP into an ILPP. Our strategy is the following. We take the original SDP and replace the multiplication in $\mathbb{F}_2$ by the usual integer multiplication. Hence, the syndrome vector is no longer a binary vector, but an integer vector. Furthermore, we will formalise this modified SDP as an ILPP and prove that if there is an unique vector $\boldsymbol{x}^*$ solution to the ILPP, satisfying the Hamming weight condition in the SDP, then $\boldsymbol{x}^*$ is the optimum solution of the ILPP. Hence, to find the error vector, solution to the SDP, one needs to solve the ILPP and output $\boldsymbol{x}^*$. We test our method in Maple software using two functions: `simplex` and `LPSolve`. Our simulations point out that the modified SDP is always solvable by the corresponding ILPP. Moreover, the empirical performance of the LP solvers points out to a polynomial-time algorithm.

The article is organized as follows. In Section 2, we introduce a short technical background from coding theory necessary for the rest of the article. Section 3 gives a brief description of the ILPP. In the same section we describe the modified SDP as an ILPP. In Section 4, we propose a series of simulations to sustain the efficiency of this method.

2 Syndrome decoding problem

Throughout this article, $\mathbb{F}_q$ denotes the finite field with q elements, $\mathcal{M}_{k,n}(\mathbb{F})$ denotes the set of $k \times n$ matrices over a field $\mathbb{F}$. An $[n, k]$ linear code $\mathscr{C}$ over $\mathbb{F}_{q^m}$ is a linear subspace of dimension k of the vector space $\mathbb{F}_{q^m}^n$. Any element in $\mathscr{C}$ is called a *codeword.* A *generator matrix* for a $[n, k]$ linear code is a $k \times n$ matrix (often denoted by $\boldsymbol{G}$) whose rows form a basis for the code. The *dual* of $\mathscr{C}$, denoted by $\mathscr{C}^{\perp}$, is the linear code which consists of all vectors $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ such that $\forall\ \boldsymbol{c} \in \mathscr{C} \quad \boldsymbol{y} \cdot \boldsymbol{c}^T = 0$. A *parity-check matrix* of $\mathscr{C}$ is a generator matrix of its dual. It is also an $(n - k) \times n$ matrix $\boldsymbol{H}$ of full rank that satisfies $\boldsymbol{H}\boldsymbol{c}^T = \boldsymbol{0}$ for all $\boldsymbol{c} \in \mathscr{C}$. The Hamming distance between two vectors is the number of coordinates on which they differ. This distance induces a norm known as the Hamming weight $\mathtt{wt}$. One of the main attributes of an error-correcting code, exploited by any code-based cryptosystem, is the decoding capability of a code $\mathscr{C}$. A decoding function of a code $\mathscr{C}$ takes as input any vector $\boldsymbol{y}$ from the ambient space and outputs the most likely codeword $\boldsymbol{c} \in \mathscr{C}$, i.e., the $\boldsymbol{c}$ that maximizes the probability of receiving $\boldsymbol{y}$ given that $\boldsymbol{c}$ was sent over the communication channel. The majority of the code-based schemes use the Binary Symmetric Channel framework. In this case, the decoding problem can be solved by the well-known closest vector problem. One possible solution is the SDP.

Definition 1 (Decisional SDP)**.**
Instance: *A full rank matrix* $\boldsymbol{H} \in \mathcal{M}_{n-k,n}(\mathbb{F}_2)$*, a vector* $\boldsymbol{s} \in \mathbb{F}_2^{n-k}$
and an integer $\omega > 0$.
Question: *Is there a vector* $\boldsymbol{x} \in \mathbb{F}_2^n$ *of weight* $\leq \omega$*, such that* $\boldsymbol{H}\boldsymbol{x} = \boldsymbol{s}$*?*

This problem can be defined over any finite extension field. However, the most common one is over $\mathbb{F}_2$.

3 Modified syndrome decoding problem

3.1 Integer Linear Programming

Any ILPP can be written either in canonical form or in standard form.

Definition 2 (ILPP)**.** *Let* n, m *be two positive integers. Let* $\boldsymbol{c} \in \mathbb{Z}^n, \boldsymbol{b} \in \mathbb{Z}^m$*,* $\boldsymbol{s} \in \mathbb{Z}^m$ *and* $\boldsymbol{A} \in \mathcal{M}_{m,n}(\mathbb{Z})$*. Define*

- *the canonical form of an ILPP*

$$\max\{\boldsymbol{c}^t\boldsymbol{x} | \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{x} \in \mathbb{Z}^n, \boldsymbol{x} \geq 0\}; \tag{1}$$

- *the standard form of an ILPP*

$$\max\{\boldsymbol{c}^t\boldsymbol{x} | \boldsymbol{A}\boldsymbol{x} + \boldsymbol{d} = \boldsymbol{b}, \boldsymbol{x}, \boldsymbol{d} \in \mathbb{Z}^n, \boldsymbol{x} \geq 0, \boldsymbol{d} \geq 0\}. \tag{2}$$

Notice that in the standard form the vector $\boldsymbol{d}$ can be seen as a quantification of the difference between $\boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{b}$, compared to the canonical form. Indeed, in the canonical form, $\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}$ might be equal to any positive vector, where as in the standard form this vector is fixed and given. Any vector $\boldsymbol{x}$ satisfying $\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$ is called a feasible solution. If a feasible solution $\boldsymbol{x}^*$ satisfies the maximum condition in (1) then $\boldsymbol{x}^*$ is optimal. A particular ILPP, that we will consider here, is when equality holds in Equation (1), i.e., $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, which is equivalent to the standard form (2) with $\boldsymbol{d} = \boldsymbol{0}$. Considering the dual problem, we have:

Definition 3 (Primal-Dual ILPP)**.** *Let n, m be two positive integers and $\boldsymbol{c} \in \mathbb{Z}^n, \boldsymbol{b} \in \mathbb{Z}^m, \boldsymbol{s} \in \mathbb{Z}^m$ and $\boldsymbol{A} \in \mathcal{M}_{m,n}(\mathbb{Z})$. Define the primal ILPP as*

$$\max\{\boldsymbol{c}^t\boldsymbol{x}|\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \in \mathbb{Z}^n, \boldsymbol{x} \geq 0\}. \tag{3}$$

Then the dual problem of (3) *is defined as*

$$\min\{\boldsymbol{b}^t\boldsymbol{x}|\boldsymbol{A}^t\boldsymbol{x} = \boldsymbol{c}, \boldsymbol{x} \in \mathbb{Z}^n, \boldsymbol{x} \geq 0\}. \tag{4}$$

3.2 The modified syndrome decoding problem

We define the Modified SDP over $\mathbb{Z}$ as:

Definition 4 ($\mathbb{Z}$-MSDP)**.**

Instance: $\boldsymbol{H} \in \mathcal{M}_{n-k,n}(\mathbb{Z})$ *with* $h_{i,j} \in \{0, 1\}$ *for all* i, j *a vector* $\boldsymbol{s} \in \mathbb{Z}^{n-k}$ *and an integer* $\omega > 0$.

Question: *Is there a vector* $\boldsymbol{x} \in \{0, 1\}^n$ *with* $\mathsf{wt}(\boldsymbol{x}) \leq \omega$, *s.t.* $\boldsymbol{H}\boldsymbol{x} = \boldsymbol{s}$*?*

Remark 3.1. *The difference between SDP and $\mathbb{Z}$-MSDP is that the matrix-vector multiplication is no longer performed in $\mathbb{F}_2$, but in $\mathbb{Z}$. This is why the syndrome vector now has entries in $\mathbb{Z}$ instead of $\mathbb{F}_2$.*

In the rest of article we restrict our analysis to the case where $\mathsf{wt}(\boldsymbol{x}) = \omega$. We also suppose that the $\mathbb{Z}$-MSDP has a solution, i.e., the syndrome vector $\boldsymbol{s}$ was not generated at random, but was obtained as a valid instance of the matrix-vector multiplication.

Theorem 1. *Let us suppose that there exists a unique vector $\boldsymbol{x}^* \in \{0, 1\}^n$ with $\mathsf{wt}(\boldsymbol{x}^*) = \omega$, solution to the $\mathbb{Z}$-MSDP. Then $\boldsymbol{x}^*$ is the optimum solution of an ILPP.*

Proof. Suppose that such an $\boldsymbol{x}^*$ exists and is unique, i.e., $\boldsymbol{H}\boldsymbol{x}^* = \boldsymbol{s}$ with $\boldsymbol{s} \in \mathbb{Z}^{n-k}$ and $\mathsf{wt}(\boldsymbol{x}^*) = \omega$. We will construct an ILPP for which $\boldsymbol{x}^*$ is the optimum solution. For that, we simply set $\boldsymbol{A}^t = \boldsymbol{H}, \boldsymbol{c} = \boldsymbol{s}$, and $\boldsymbol{b}^t = (1, \ldots, 1)$ in (4). Since $\boldsymbol{x} \in \{0, 1\}^n$ $\mathsf{wt}(\boldsymbol{x}) = \sum_{i=1}^{n} x_i = (1, \ldots, 1) \cdot \boldsymbol{x}$, but this is equal to $\boldsymbol{b}^t\boldsymbol{x}^*$. The ILPP we need to solve can now be defined

$$\min\{\boldsymbol{b}^t\boldsymbol{x}|\boldsymbol{A}^t\boldsymbol{x} = \boldsymbol{c}, \boldsymbol{x} \in \{0, 1\}^n\}. \tag{5}$$

This implies that $\boldsymbol{x}^*$ is a feasible solution to (5), and as $\boldsymbol{x}^*$ is the unique vector satisfying $\boldsymbol{A}^t\boldsymbol{x}^* = \boldsymbol{c}$ with $\mathsf{wt}(\boldsymbol{x}^*) \leq \omega$, $\boldsymbol{x}^*$ is optimum for the minimum weight condition. □

Remark 3.2. *The condition from* (5), *more exactly $\boldsymbol{x} \in \{0, 1\}^n$, could possibly be replaced with the more general $\boldsymbol{x} \in \mathbb{Z}^n$. In this case, Theorem 1 still holds as long as there is no other integer feasible solution $\boldsymbol{x}' \in \mathbb{Z}^n$ with $(1, \ldots, 1) \cdot \boldsymbol{x}' < \omega$.*

4 Simulations

We simulated the algorithm that solves the $\mathbb{Z}$-MSDP on a regular laptop with the Linux operating system running on a 2-core processor at 2.60 gigahertz. We are running our script using the Maple software and its optimization packages, more precisely, the `simplex minimize` and `LPSolve` functions.

Remark 4.1. *Throughout our simulations we noticed that, in nearly all the cases, the `LPSolve` function was much faster than the `simplex` function. That is why we will describe and discuss our variant of the algorithm that uses the `LPSolve` function.*

Our script is composed of two steps, the `Generation` function and the `Solve` function.

The `Generation` function takes as input the integers n, k, ω and outputs an error vector $\boldsymbol{e}$ of weight ω and an integer syndrome vector $\boldsymbol{s}$.

1. Firstly, it generates a random binary matrix of length n and dimension k using the `RandomMatrix` function and it checks that all rows/columns are pairwise distinct;
2. Secondly, it generates a random binary vector of fixed Hamming weight ω ;
3. Thirdly, it computes the corresponding integer syndrome $\boldsymbol{s}$.

The `Solve` step takes as input the matrix and the syndrome and outputs the estimated error.

1. Firstly, it creates the system using the binary matrix and the syndrome $\boldsymbol{s}$;
2. Secondly, it defines the constraints on the solution, i.e., Hamming weight is minimized;
3. Thirdly, it calls the `LPSolve` function with the aforementioned parameters. The output of the `LPSolve` function is a vector of rational numbers and the corresponding Hamming weight.
4. There is also a verification step, where we check the equality of the two vectors, i.e., between the $\boldsymbol{e}$ and the output vector of the `Solve` function.

We conducted two types of experiments, i.e., verification and performance. The first type of simulations were done to check the correctness of the solution. More exactly, we counted how many verification steps were successful for a given number N of trials. Our results showed that with probability 1 the `LPSolve` function recovers the exact error vector. The experiments were conducted for values of n up to 200 and $k \leq \frac{n}{2}$. We repeated each experiment for 10000 different vectors for a fixed matrix, and repeat this for 1000 random binary matrices.

Remark 4.2. *We noticed that our algorithm retrieved the correct error vector for any value of ω. This fact implies that for some given binary matrix $\boldsymbol{H}$ and integer vector $\mathbf{s}$ there is only one binary solution, and on top of that it has the minimum Hamming weight.*

Moreover, even when the condition on $\boldsymbol{x}$ to be binary was relaxed to $\boldsymbol{x} \in \mathbb{Z}^n$, the solution to the problem found by our algorithm was the binary one. In other words, for the given matrix and syndrome, there were no feasible integer solutions satisfying $(1, \ldots, 1) \cdot \boldsymbol{x} = \omega$.

If we slightly modified one of the components of the syndrome $\boldsymbol{s}$, i.e., the new syndrome $\boldsymbol{s}^* = \boldsymbol{s} + \Big(0, \ldots, 0, \epsilon_j, 0, \ldots, 0\Big)^t$, and apply the `Solve` function to the initial matrix and $\boldsymbol{s}^*$, we noticed two different cases:

- when $\epsilon_j \geq 0$ the algorithm finds a solution, which is a rational vector different from $\boldsymbol{e}$;
- when $\epsilon_j < 0$ the algorithm does not find a solution with Hamming weight $\leq \omega$.

Table 2: Timings (in seconds) for the `LPSolve` function for two types of parameters. On the left hand side for $k = n/2$ and on the right hand side for $k = \frac{n}{3}$. Also two different types of values for ω, $\omega = \sqrt{n}$ (upper part of the table) and $\omega = \log_2(n)$ (lower part of the table).

k	n	ω/n	Timings [s]	k	n	ω/n	Timings [s]
250	500	0.044	1.74	167	500	0.044	1.03
500	1000	0.031	16.92	333	1000	0.031	9.65
750	1500	0.025	63.8	500	1500	0.025	34.38
250	500	0.017	1.72	167	500	0.017	0.98
500	1000	0.010	16.24	333	1000	0.010	8.34
750	1500	0.007	63.41	500	1500	0.007	34.14

The second type of simulations were evaluating the performance of the algorithm. The `LPSolve` function performed extremely well for small values of n, e.g., for $n = 100$ and $k = 50$ it took less than 25 milliseconds to find the solution for any $\omega \leq n/3$. For $n = 200$ and $k = 100$ it took less than 0.1 second for any $\omega \leq \sqrt{n}$.

For large values of n, we illustrate in Table 2 the performance of the `LPSolve` function. It is worth noting that for $n = 1500, k \leq n/2$ and $\omega \leq \sqrt{n}$ we need less than 60 seconds to find the solution. Also, we can observe that from $k = n/2$ to $k = n/3$ the timings are almost divided by 2.

5 Conclusion and Perspectives

We have shown in this article that, using linear programming, we can efficiently solve a modified version of the syndrome decoding problem (the $\mathbb{Z}-$MSDP). Our simulations reveal interesting properties, that we intend to investigate in the near future from a theoretical point of view. Among the most significant ones, we mentioned the uniqueness of the binary solution, even under less restrictive conditions, and the fact that the observed work factor of the algorithm was polynomial (more precisely cubic in the parameters). It would be interesting to see if other modifications of other similar hard problems can be solved more efficiently using linear programming.

Funding

This work was supported by a grant of the Romanian Ministry of Education and Research, CNCS - UEFISCDI, project number PN-III-P1-1.1-PD-2019-0285, within PNCDI III.

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Augot, D.; Finiasz, M.; Sendrier, N. (2005). A family of fast syndrome based cryptographic hash functions. In *Ed Dawson, Serge Vaudenay (editors). Progress cryptology-Mycrypt First international conference on cryptology Malaysia,* Springer LNCS, 3715, 64–83, Kuala Lumpur, Malaysia, Sept. 2005.

[2] Baldi, M.; Chiaraluce, F. (2007). Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, 2591–2595, Nice, France, June 2007.

[3] Bardet, M.; Chaulet, J.; Dragoi, V.; Otmani, A.; Tillich, J.-P. (2016). Cryptanalysis of the McEliece public key cryptosystem based on polar codes. In *Post-Quantum Cryptography 2016*, Springer LNCS, 9606, 118–143, Fukuoka, Japan, Feb. 2016.

[4] Bardet, M.; Dragoi, V.; Luque, J.; Otmani, A.(2016). Weak keys for the quasi-cyclic MDPC public key encryption scheme. In *Progress in Cryptology - AFRICACRYPT 2016 - 8th International Conference on Cryptology in Africa,* Springer LNCS, 9646, 346–367, Fes, Morocco, April 13-15, 2016.

[5] Berger, T.P.; Loidreau, P.(2005). How to mask the structure of codes for a cryptographic use. *Des. Codes Cryptogr.*, 35(1),63–79, 2005.

[6] Berlekamp, E.; McEliece, R.; van Tilborg, H.(1978). On the inherent intractability of certain coding problems. *IEEE Trans. Inform. Theory*, 24(3),384–386, May 1978.

[7] Bernstein, D.J. ; Lange, T.; Peters, C.(2010). Wild McEliece. In A. Biryukov, G. Gong, D. Stinson, editors, *Selected Areas in Cryptography*, Springer LNCS, 6544, 143–158, 2010.

[8] Bernstein, D.J. ; Lange, T.; Peters, C.(2011). Wild McEliece Incognito. In B.-Y. Yang, editor, *Post-Quantum Cryptography 2011*, Springer LNCS, 7071, 244–254. Springer Berlin Heidelberg, 2011.

[9] Borghoff, J.(2012). Mixed-integer linear programming in the analysis of trivium and ktantan. *IACR Cryptology ePrint Archive*, 2012. URL:https://eprint.iacr.org/2012/676.pdf

[10] Borghoff, J.; Knudsen, L.R.; Stolpe, M.(2009). Bivium as a mixed-integer linear programming problem. In *Cryptography and Coding*, Springer LNCS, 5921, 133–152. Springer Berlin Heidelberg, 2009.

[11] Chizhov, I.V.; Borodin, M.A.(2014). Effective attack on the McEliece cryptosystem based on Reed-Muller codes. *Discrete Math. Appl.*, 24(5),273–280, 2014.

[12] Courtois, N.; Finiasz, M.; Sendrier, N.(2001). How to achieve a McEliece-based digital signature scheme. In *Advances in Cryptology - ASIACRYPT 2001*, Springer LNCS 2248, 157–174, 2001. Springer.

[13] Couvreur, A.; Gaborit, P.; Gauthier-Umaña, V.; Otmani, A., Tillich, J.-P. (2014). Distinguisher-based attacks on public-key cryptosystems using Reed-Solomon codes. *Des. Codes Cryptogr.*, 73 (2),641–666, 2014.

[14] Couvreur, A.; Márquez-Corbella, I.; Pellikaan, R.(2014). A polynomial time attack against algebraic geometry code based public key cryptosystems. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT 2014*, 1446–1450, June 2014.

[15] Couvreur, A.; Otmani, A.; Tillich, J.-P.(2013). New identities relating wild Goppa codes. *Finite Fields Appl.*, 29,178–197, 2014.

[16] Couvreur, A.; Otmani, A.; Tillich, J.-P.(2014). Polynomial time attack on wild McEliece over quadratic extensions. In P. Q. Nguyen and E. Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014*, Springer LNCS, 8441, 17–39. Springer Berlin Heidelberg, 2014.

[17] Dragoi, V.; Talé Kalachi, H.(2018). Cryptanalysis of a public key encryption scheme based on QC-LDPC and QC-MDPC codes. *IEEE Communications Letters*, 22(2),264–267, Feb 2018.

[18] Drăgoi, V.; Beiu, V.; Bucerzan, D.(2019). Vulnerabilities of the mceliece variants based on polar codes. In J.-L. Lanet and C. Toma, editors, *Innovative Security Solutions for Information Technology and Communications*, Springer LNCS,11359, 376–390, Cham, 2019. Springer International Publishing.

[19] Faugère, J.-C. ; Otmani, A.; Perret, L.; de Portzamparc, F.; Tillich, J.-P. (2016). Folding alternant and Goppa Codes with non-trivial automorphism groups. *IEEE Trans. Inform. Theory*, 62(1), 184–198, 2016.

[20] Faure, C.; Minder, L.(2008). Cryptanalysis of the McEliece cryptosystem over hyperelliptic curves. In *Proceedings of the eleventh International Workshop on Algebraic and Combinatorial Coding Theory*, 99–107, Pamporovo, Bulgaria, June 2008.

[21] Ganian, R.; Ordyniak, S.(2019). Solving integer linear programs by exploiting variable-constraint interactions: A survey. *Algorithms*, 12(12),248, 2019.

[22] B.Gou (2008). Generalized integer linear programming formulation for optimal pmu placement. *IEEE transactions on Power Systems*, 23(3),1099–1104, 2008.

[23] Gueye, C.T. ; Mboup, E.H.M. (2013). Secure cryptographic scheme based on modified Reed Muller codes. *International Journal of Security and Its Applications*, 7(3),55–64, 2013.

[24] Hooshmand, R.; Shooshtari, M.K. ; Eghlidos, T.; Aref, M.(2014). Reducing the key length of McEliece cryptosystem using polar codes. In *2014 11th International ISC Conference on Information Security and Cryptology (ISCISC)*, 104–108. IEEE, 2014.

[25] Janwa, H.; Moreno, O. (1996). McEliece public key cryptosystems using algebraic-geometric codes. *Des. Codes Cryptogr.*, 8(3),293–307, 1996.

[26] Johnson, E.L. ; Nemhauser, G.L. ; Savelsbergh, M.W.(2000) . Progress in linear programming-based algorithms for integer programming: An exposition. *Informs journal on computing*, 12(1), 2–23, 2000.

[27] Karp, R.M. (1972). Reducibility among combinatorial problems. In *Complexity of computer computations*, 85–103. Springer, 1972.

[28] Landais, G.; Tillich, J.-P. (2013). An efficient attack of a McEliece cryptosystem variant based on convolutional codes. In P. Gaborit, editor, *Post-Quantum Cryptography'13*, Springer LNCS, 7932, 102–117. Springer, June 2013.

[29] Lenstra Jr, H.W.(1983). Integer programming with a fixed number of variables. *Mathematics of operations research*, 8(4),538–548, 1983.

[30] Loidreau, P.; Sendrier, N.(2001). Weak keys in the McEliece public-key cryptosystem. *IEEE Trans. Inform. Theory*, 47(3),1207–1211, 2001.

[31] Löndahl, C.; Johansson, T.(2012). A new version of McEliece PKC based on convolutional codes. In *Information and Communications Security, ICICS*, Springer LNCS, 7168, 461–470. Springer, 2012.

[32] McEliece, R.J. (1987). *A Public-Key System Based on Algebraic Coding Theory*, pages 114–116. Jet Propulsion Lab, 1978. DSN Progress Report 44.

[33] Minder,L.; Shokrollahi,A.(2007). Cryptanalysis of the Sidelnikov cryptosystem. In *Advances in Cryptology - EUROCRYPT 2007*, Springer LNCS, 4515, 347–360, Barcelona, Spain, 2007.

[34] Misoczki, R.; Tillich, J.-P.;Sendrier, N.; Barreto, P.S. L.M.(2013). MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, 2069–2073, 2013.

[35] Mizuno, S.(2016). The simplex method using tardos' basic algorithm is strongly polynomial for totally unimodular lp under nondegeneracy assumption. *Optimization Methods and Software*, 31 (6),1298–1304, 2016.

[36] Monico, C.; Rosentha, J.l; Shokrollahi, A.A.(2000). Using low density parity check codes in the McEliece cryptosystem. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, 215, Sorrento, Italy, 2000.

[37] Moufek, H.; Guenda, K. Gulliver, T.A. (2017). A new variant of the mceliece cryptosystem based on qc-ldpc and qc-mdpc codes. *IEEE Communications Letters*, 21(4),714–717, April 2017.

[38] Mouha, N.; Wang, Q.; Gu, D.; Preneel, B.(2012). Differential and linear cryptanalysis using mixed-integer linear programming. In C.-K. Wu, M. Yung, and D. Lin, editors, *Information Security and Cryptology*, 57–76. Springer Berlin Heidelberg, 2012.

[39] Niederreiter, H.(1985). A public-key cryptosystem based on shift register sequences. In *Advances in Cryptology - EUROCRYPT 1985*, Springer LNCS, 219, 35–39, 1985.

[40] Otmani, A.; Kalachi, H.T.(2015). Square code attack on a modified sidelnikov cryptosystem. In *Codes, Cryptology, and Information Security*, 173–183. Springer, 2015.

[41] Otmani, A.; Tillich, J.-P. ; Dallot, L.(2008). Cryptanalysis of McEliece cryptosystem based on quasi-cyclic LDPC codes. In *Proceedings of First International Conference on Symbolic Computation and Cryptography*, 69–81, Beijing, China, Apr. 28-30 2008. LMIB Beihang University.

[42] Persichetti, E.(2012). Compact McEliece keys based on quasi-dyadic Srivastava codes. *J. Math. Cryptol.*, 6(2),149–169, 2012.

[43] Richards, A.; How, J.P.(2002). Aircraft trajectory planning with collision avoidance using mixed integer linear programming. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)*, 3, 1936–1941. IEEE, 2002.

[44] Sendrier, N.(1994). On the structure of a randomly permuted concatenated code. In *EUROCODE'94*, 169–173, 1994.

[45] Sendrier, N.(1998). On the concatenated structure of a linear code. *Appl. Algebra Eng. Commun. Comput. (AAECC)*, 9(3),221–242, 1998.

[46] Shrestha, S.R.; Kim, Y.-S.(2014). New McEliece cryptosystem based on polar codes as a candidate for post-quantum cryptography. In *2014 14th International Symposium on Communications and Information Technologies (ISCIT)*, 368–372. IEEE, 2014.

[47] Sidelnikov, V.M.(1994). A public-key cryptosytem based on Reed-Muller codes. *Discrete Math. Appl.*, 4(3),191–207, 1994.

[48] Sidelnikov, V.M. ; Shestakov, S.(1992). On the insecurity of cryptosystems based on generalized Reed-Solomon codes. *Discrete Math. Appl.*, 1(4),439–444, 1992.

[49] Wagner, H.M.(1959). An integer linear-programming model for machine scheduling. *Naval Research Logistics Quarterly*, 6(2),131–140, 1959.

[50] Wieschebrink, C.(2006). Two NP-complete problems in coding theory with an application in code based cryptography. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, 1733–1737, 2006.

[51] Wieschebrink, C.(2006). An attack on a modified Niederreiter encryption scheme. In M. Yung, Y. Dodis, A. Kiayias, and T. Malk, editors, *Public-Key Cryptography - PKC 2006*, Springer LNCS, 3958, 14–26. Springer, 2006.

[52] Wieschebrink, C.(2009). Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. IACR Cryptology ePrint Archive, Report 2009/452, 2009.

C O P E
Member since 2012
JM08090

This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:
Drăgoi, V.-F.; Cayrel, P.-L. ; Colombier, B.; Bucerzan, D.; Hoară, S. (2020). Solving a Modified Syndrome Decoding Problem using Integer Programming, *International Journal of Computers Communications & Control*, 15(5), 3920, 2020. https://doi.org/10.15837/ijccc.2020.5.3920