Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

December 2019 Volume XI Number 4 ISSN 2061-2079

MESSAGE FROM THE GUEST EDITORS

SPECIAL ISSUE

PAPERS FROM OPEN CALL

CALL FOR PAPERS / PARTICIPATION

ADDITIONAL

Technically Co-Sponsored by

IEEE ComSoc™
IEEE Communications Society

hte

IEEE
HUNGARY SECTION

HTE for 70 years

Editorial Board

Indexing information

Infocommunications Journal is covered by Inspec, Compendex and Scopus.
Infocommunications Journal is also included in the Thomson Reuters – Web of ScienceTM Core Collection, Emerging Sources Citation Index (ESCI)

Infocommunications Journal

Technically co-sponsored by IEEE Communications Society and IEEE Hungary Section

Supporters

FERENC VÁGUJHELYI – president, National Council for Telecommunications and Information Technology (NHIT)
GÁBOR MAGYAR – president, Scientic Association for Infocommunications (HTE)

Editorial Office (Subscription and Advertisements)**:**
Scientic Association for Infocommunications
H-1051 Budapest, Bajcsy-Zsilinszky str. 12, Room: 502
Phone: +36 1 353 1027
E-mail: info@hte.hu • Web: www.hte.hu

Articles can be sent also to the following address:
Budapest University of Technology and Economics
Department of Telecommunications and Media Informatics
Phone: +36 1 463 4189, Fax: +36 1 463 3108
E-mail: pvarga@tmit.bme.hu

Subscription rates for foreign subscribers: 4 issues 10.000 HUF + postage

Publisher: PÉTER NAGY

HU ISSN 2061-2079 • Layout: PLAZMA DS • Printed by: FOM Media

Special Issue on Cryptology – Guest Editorial

Václav (Vashek) Matyáš, Pavol Zajac, Jan Hajný and Marek Sýs

Abstract—**This special issue brings selected papers from the 2019 Central European Conference on Cryptology, held in Telč, June 12-14, 2019.**

This special issue focuses on the area of applied cryptography, bringing up selected papers from the 2019 Central European Conference on Cryptology, covering various aspects of cryptology. All accepted papers went through two rounds of reviews and the authors duly incorporated the feedback in their revised papers.

The first paper of Michal Andrzejczak and Wladyslaw Dudzic "SAT Attacks on ARX Ciphers with Automated Equations Generation" investigates a new approach to algebraic attacks on block ciphers with SAT solvers. Authors try to find encryption keys for ciphers SIMON and SPECK by solving a specific system of equations. The equations are converted to satisfiability problem, and solved with standard SAT solvers. The novel approach of the authors is not to model so-called key expansion algorithm, producing a smaller system, but with a possibility of finding invalid keys. Probability of invalid keys is reduced by using multiple input-output pairs, which however increases the system.

The second paper of Mithilesh Kumar et al. "Reducing Lattice Enumeration Search Trees" deals with the security of post-quantum lattice-based schemes. In particular, the paper focuses on algorithms solving the shortest vector problem (SVP). Two optimized methods are proposed in the paper. The first method (hybrid enumeration) is based on finding suitable permutations, the second (sign-based pruning) is based on the estimation of coefficient signs. The paper also presents the experimental results provided for both methods and the comparison with standard techniques.

The third paper "The search of square m-sequences with maximum period via GPU and CPU" of Paweł Augustynowicz and Krzysztof Kanciak is concerned with the efficient parallel search of square m-sequences on modern CPUs and GPUs. The authors come up with the idea to exploit particular vector processor instructions, with the aim to utilize the advantages of the Single Instruction Multiple Data and Single Instruction Multiple Threads execution patterns. The authors also present the early abort sieving strategy based on the application of SAT-solvers. The paper shows that the proposed solution can exhaustively search m-sequences up to the degree 32.

The last paper "A New Type of Signature Scheme Derived from a MRHS Representation of a Symmetric Cipher" of Pavol Zajac and Peter Špaček introduces a fundamentally new idea of a post-quantum signature scheme. The scheme is defined by Multiple-Right-Hand-Side (MRHS) equations representing the entire SPN of the given cipher. The paper describes key procedures of the algorithm (key generation, signature generation, signature verification) and provides simplified examples for some critical steps of the algorithm. The security of the scheme is based on the difficulty of solving MRHS equations, or equivalently on the difficulty of the decoding problem (both are NP-hard).

Václav (Vashek) Matyáš is a Professor at Masaryk University, Brno, CZ, acting as the Vice-Dean for Industrial and Alumni Relations at the Faculty of Informatics. His research interests are related to applied cryptography and security; he has published well over 150 peer-reviewed papers and articles and has co-authored several books. He worked in the past with Red Hat Czech, CyLab at Carnegie Mellon University, as a Fulbright-Masaryk Visiting Scholar at the Center for Research on Computation and Society of Harvard University, Microsoft Research Cambridge, University College Dublin, Ubilab at UBS AG, and as a Royal Society Postdoctoral Fellow with the Cambridge University Computer Lab. Vashek also worked on the Common Criteria and in ISO/IEC JTC1 SC27. Vashek is a member of the Editorial Board of the Infocommunications Journal and can be contacted at matyas AT fi.muni.cz.

Pavol Zajac is a Professor of Applied Computer Science at Slovak University of Technology in Bratislava, Slovakia. His research interests are related to mathematical cryptography and information security. Nowadays he works mostly with post-quantum cryptography and related algebraic problems. Pavol can be contacted pavol.zajac AT stuba.sk.

Jan Hajný works as an Associate Professor at the Faculty of Electrical Engineering and Communication at Brno University of Technology. He is the head of the Advanced Cybersecurity group, member of the faculty's Scientific Committee and the person responsible for the Information Security study programs. The scientific activities of prof. Hajný include research into modern cryptography and privacy protection. Prof. Hajný is the principal investigator of many projects, including Czech grants (GAČR, TAČR) and international projects (Horizon 2020). He is also active in the contractual research for major Czech companies and in international collaboration with institutions visited as a visiting researcher, i.e., KU Leuven, BE; IBM Research Zurich, CH; and University of Minnesota, USA.

Marek Sýs is an Assistant Professor at Masaryk University, Brno, CZ. His research interests are related to applied cryptography where he published 12 peer-review papers. His collaborative work received Real-World Award at ACM CCS 2017 for discovering ROCA vulnerability. He worked in the past as Postdoctoral Research Associate at Masaryk University and Assistant Professor at Technical University, Bratislava. He received his PhD degree from the Technical University, Bratislava, and can be contacted at syso AT mail.muni.cz.

SAT Attacks on ARX Ciphers with Automated Equations Generation

Michal Andrzejczak[1] and Wladyslaw Dudzic[2]

***Abstract*— We propose a novel and simple approach to algebraic attack on block ciphers with the SAT-solvers. As opposed to a standard approach, the equations for key expansion algorithms are not included in the formulas that are converted to satisfiability problem. The lack of equations leads to finding the solution much faster. The method was used to attack a lightweight block ciphers - SIMON and SPECK. We report the timings for round-reduced versions of selected ciphers and discuss the potential factors affecting the execution time of our attack.**

***Index Terms*—SAT attacks, cryptanalysis, ARX ciphers, SIMON, SPECK, FPGA.**

I. Introduction

Block ciphers are the essential elements in communication security, providing secrecy of exchanged information at expected security level. From time to time, a new cipher is proposed, usually offering some new functionality. For example, authenticated encryption, or a new property like smaller hardware requirements. The lightweight cryptography is a recently growing area of research. It'-s aim is to deliver secure ciphers suitable for embedded device development, especially Internet of Things (IoT) devices. Due to hardware limitations, lightweight cipher usually consists mostly of basic logic operations.

The proper cipher cryptanalysis is a key part of cipher development, as well as vital for proving the targeted security strength of the cipher. To date, many techniques for cryptanalysis have been introduced. Thus, it takes time to prove the security of the cipher, since every new cipher must resist against all known attacks.

Previously, several ciphers were broken by a new kind of attack, which was not known at the time of cipher development. For example, Data Encryption Standard (DES) was broken by algebraic cryptanalysis with SAT solvers [1]. Thus, effort put into searching for new methods and accelerations to known ones may be beneficial.

The recent development and improvements in SAT solvers have led to a new algebraic attack on block ciphers. The current state of SAT solvers lowers the total time of the key recovery attack for a new set of ciphers directly affecting cipher's security, especially the lightweight ones.

A. Contribution

In this paper we present our novel approach to constructing an SAT attack on lightweight block ciphers. We report the results for an SAT attack with our method on ARX ciphers: SIMON and SPECK [2]. The method for obtaining equations for describing the cipher, which are required for an algebraic attack is presented with several sources of equations being listed. We describe our attack approach, considering many factors influencing the time of the attack — several equations, the form of equations, the number of known and used plaintex-ciphertext pairs and the used SAT solver [3]. By analyzing our results, we propose the best approach to break SIMON and SPECK with algebraic cryptanalysis. This approach can be also applied to other ciphers.

II. Previous work

SIMON and SPECK resistance against differential and linear cryptanalysis has been thoroughly investigated in [4], [5] and [6]. SAT attacks are widely used in cryptography, often as a supporting method for other classical attacks like linear or differential cryptanalysis.

SIMON and SPECK have gained cryptanalyst's attention and as a result, several papers about security of the mentioned ciphers were published. In 2014, Curtois et.al [7] presented an algebraic attack combined with a truncated differentials attack on SIMON. They were able to conduct a practical and successful attack on nine rounds of SIMON. However, the attack is more distinguished and requires additional time spent on searching for proper truncated differentials. Found differentials are provided to a system of equations as a plaintext-ciphertext pairs. In the next step, an SAT attack is conducted.

The most recent results have been published by Ren and Chen [8]. They report the first zero-correlation linear attack and integral attack on 11 rounds of SPECK. To conduct the attack, they have used an SAT-based model to search for impossible differentials and zero-correlation linear hulls.

III. Algebraic cryptanalysis

Algebraic cryptanalysis is an attack method on a large subset of ciphers [9]. It consist of two main steps. The first step relies on converting the cipher into a system of polynomial equations, usually over $GF(2)$, but not limited to this particular ring. In the second step, the system of equations is being solved to obtain a proper secret key used for encrypting the provided plaintext-ciphertext pair. There

This work was not supported by any organization.

[1,2] Institute of Mathematics and Cryptology Military University of Technology Warsaw, Poland.

[1] E-mail: michal.andrzejczak@wat.edu.pl

[2] E-mail: wladyslaw.dudzic@wat.edu.pl

DOI: 10.36244/ICJ.2019.4.1

are several approaches for solving the system of equations, ranging from XL algorithm and Gröbner basis [10] to SAT-solvers. A brief overview of algebraic cryptanalysis is provided by Bard [11]. This technique is also used for cryptanalysis of the hash function [12].

IV. SIMON AND SPECK CIPHERS

SIMON and SPECK [2] ciphers are members of the ARX ciphers family. The only operations used in ARX ciphers are simple logic operations: AND, rotate and XOR. The ciphers are lightweight and suitable for implementation on constrained devices. SIMON and SPECK both are Feistel type ciphers.

A. SIMON

SIMON's round function is as follows. For a round key k, the round function is a two stage Feistel map $R_k : GF(2)^n \times GF(2)^n \leftarrow GF(2)^n \times GF(2)^n$ defined as:

$$R_k(x, y) = (y \oplus f(x) \oplus k, x), \quad (1)$$

where $f(x) = (Sx \& S^8 x) \oplus S^2 x$, and S^j is a left circular shift by j elements. The SIMON cipher family has several possible data block and key sizes. For each possible combination the number of rounds also varies. All versions with parameters of SIMON ciphers are listed in Tab. I.

B. SPECK

The SPECK family of block ciphers is constructed only from bitwise XOR, addition modulo 2^n and similar to SIMON family, left circular shift S^j by j positions. The round function is the map $R_k : GF(2)^n \times GF(2)^n \leftarrow GF(2)^n \times GF(2)^n$ defined as:

$$R_k(x, y) = ((S^{-\alpha} x + y) \oplus k, S^{\beta} y \oplus (S^{-\alpha} + y) \oplus k), \quad (2)$$

where k is a round key and $\alpha = 7$ and $\beta = 2$ for block size $n = 32$ and $\alpha = 8$ and $\beta = 3$ otherwise. The specification of all block ciphers from SPECK family is presented in Tab. I

TABLE I
POSSIBLE VARIANTS OF SIMON AND SPECK BLOCK CIPHER

block size	key size	word size	# of SIMON rounds	# of SPECK rounds
32	64	16	32	22
48	72	24	36	22
	96		36	23
64	96	32	42	26
	128		44	27
96	128	48	52	28
	144		54	29
128	128	64	68	32
	192		69	33
	256		72	34

SIMON and SPECK have a low multiplicative complexity, which is a one of the measurement units of non-linearity [13]. Thus, algebraic cryptanalysis seems to be promising.

V. SAT ATTACK ON SIMON AND SPECK

We present a known plaintext attack for lightweight block ciphers. The attack partially belongs to the algebraic cryptanalysis family. The proposed attack starts with obtaining proper algebraic equations describing a chosen cipher, which is necessary for a key recovery attack. Classical algebraic cryptanalysis methods try to solve given equations with XL algorithm [14] or Grobner basis [10]. There are also attempts to minimize the number of variables by using external tools [15] [16]. In these algorithms, the attacker usually tries to gather some additional information from equations or tries to decrease the degree of equations and number of variables before solving the system of equations. In our approach, we do not solve the equations directly. Instead, we convert the equations to a satisfiability problem and we try to find a key's values that are valid for used pairs of plaintext and ciphertext. There are several factors affecting the execution time and probability of success. We consider them to find the best approach for attacking SIMON and SPECK with SAT—solvers. Compared with other SAT attacks on SIMON and SPECK our method does not require puting an additional effort into selecting a proper plaintext and does not require any additional tools for minimizing the number of variables. Used pairs are picked up at random and the equations are taken as they were produced by different compilers.

A. Attack model

In our attack we use two different approaches for constructing the attack model.

In the first scenario, equations for the cipher and key expansion algorithms are used by an SAT-solver. The found keys are usually valid if the number of used pairs is larger than two.

For the second scenario, equations for the key expansion algorithm are not included in the system of equations converted to a satisfiability problem. With this approach, the round keys are found in reduced time, compared to the first scenario, but the round keys might be unrelated. The unrelated keys are not valid, and they are a random keys that can not be a result of a key expansion algorithm. The probability of finding a valid key depends mostly on the number of plaintext-ciphertext pairs.

B. Number of pairs

The solution found by an SAT-solver in the second scenario is not always the valid key, used for encryption. The probability of success depends mostly on the number of used pairs. In theory, only one pair should be needed to solve the equations. This is true, the solution for an SAT-problem is found even for one pair and it is done very quickly. However, without equations for a key expansion included to the model, it is possible to find invalid unrelated round keys that solve the satisfiability problem. The probability of finding a valid key increases with the number of used pairs.

The number of pairs also affects the execution time. In Fig. 1 , Fig. 2 and Fig. 3 the time required for solving an SAT problem depens on the number of used pairs. For simple problems, additional pairs extended the running time. There is a visible cross point, where adding more pairs extend the time required for finding solution. For five rounds of SPECK, the

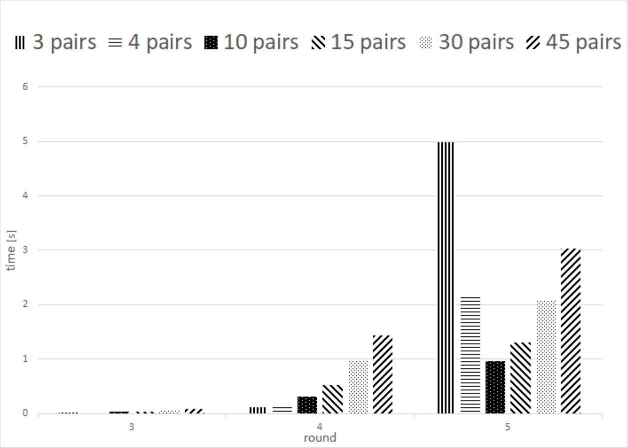


Fig. 1. Average time of attack using Lingeling and different number of pairs of SPECK 32/64 (with key expand algorithm and handmade equations)

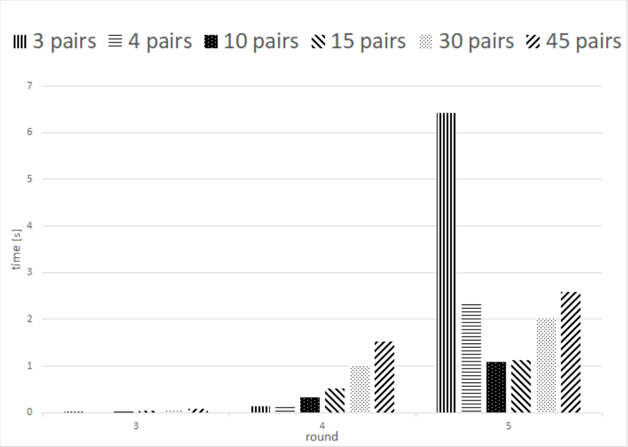


Fig. 2. Average time of attack using Lingeling and different number of pairs of SPECK 32/64 (without key expand algorithm and handmade equations)

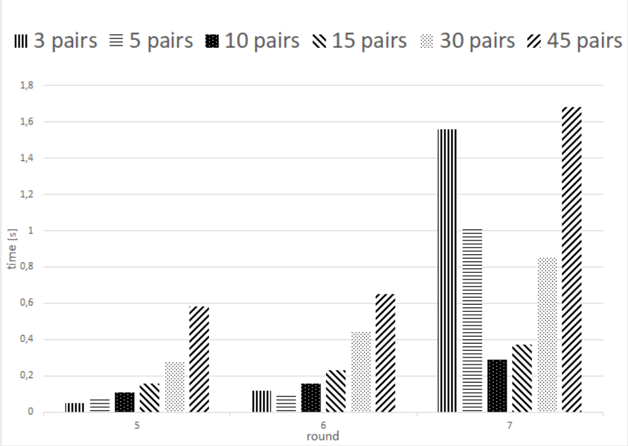


Fig. 3. Average time of attack using Lingeling and different number of pairs of SIMON 32/64 (without key expand algorithm and handmade equations)

attack works the best for only 10 pairs. For complex problems, the situation is the opposite.

Large systems of equations can be solved more efficiently with more pairs available. This dependency works for both attack models.

This is probably the case where some of the SAT-solvers are able to utilize additional information taken from a larger number of pairs, which results in finding the solution in a shorter timeframe. There are also solvers not able to utilize this additional information and their running time increases with the number of pairs

C. Equations type

The same cipher can be described by sets of a different algebraic equations. The main differences between two sets are: the number of equations, the maximal algebraic degree and the number of clauses. The equations can be obtained automatically by proper software tools. We have developed a method for extracting algebraic equations from software and hardware implementations. Thus, we are able to process different equations describing the same cipher. In the experiments we used equations from several sources including: handwritten, generated by hardware synthesis tools, generated by C compiler and generated by Cryptol. Handwritten equations seems to be the most natural and readable for human. The hardware equations are describing every logic cell in the implementation of the cipher in FPGAs, so the structure of the cipher is hidden. It is similar to equations from C and Cryptol, where the cipher is translated by compiler to computer readable format.

The idea of using an equation taken from hardware tools was earlier explored by Courtois et al. [1] to conduct an SAT attack on DES block cipher. In 2012, during SHA-3 competition, Homsirikamol et al. [15] developed a similar tool to obtain hardware equations describing SHA-3 final candidates and evaluating their security margin.

D. Numbers of clauses and variables

Fig.4 presents the increase of the number of clauses for SPECK cipher with every additional round. Fig.5 presents

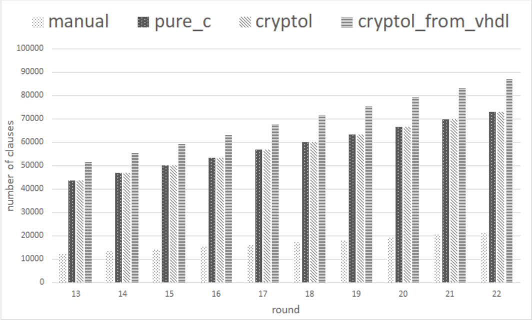


Fig. 4. Number of clauses in CNF equation, depending on the type of equations generator and number of SPECK rounds (without key expand algorithm)

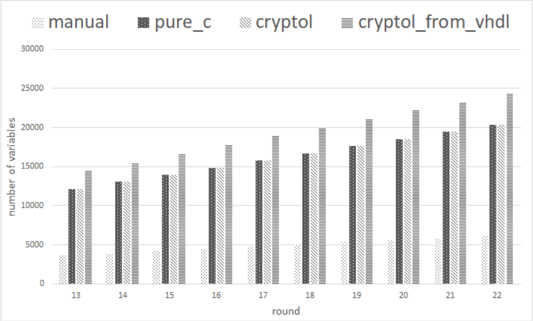


Fig. 5. Number of variables in CNF equation, depending on the type of equations generator and number of SPECK rounds for two pairs (without key expand algorithm)

the number of variables obtained from conversion tools to describe the round reduced SPECK cipher. Both numbers have a linear dependency on the number of rounds. The numbers describing the hardware and software equations increase more than handwritten ones. However, according to our experiments, the system of equations with larger number of variables and clauses can be solved faster than smaller ones. In some cases, an SAT solver can utilize the additional information hidden under the equations.

VI. RESULTS

We report the best results obtained among three SAT—solvers: Cadical, Lingeling and Treengeling using AMD FX(tm)-8300 CPU clocked with 3531 MHz. The results were obtained by attacking the smallest ciphers from the SIMON and SPECK families with limited computation time set to maximum one hour. The hardware equations are taken from synthesizing the cipher design into an FPGA board by a free-of-charge version of Intel Quartus 18.1. For every logic cell in the FPGA design, an appropriate equation is given. Our targeted FPGA family was Intel Cyclone V. Equations from C implementation are obtained by translating the and-inverted-graphs (AIG) [17] produced by clang version 3.6.0 C compiler. Equations from Cryptol [18] implementation are obtained in a similar way. The reported results are taken as average of 40 runs of each experiment. To make the comparison fair, the used key was random and the plaintext-ciphertext pairs were the same in every experiment.

# of rounds\# of pairs	3	4	>= 5
5	∼0,23	∼0.62	∼1,00
6	∼0,05	∼0.47	∼1,00
7	∼0,00	∼0.62	∼1,00
8	∼0,00	∼0.59	∼1,00
9	∼0,00	∼0.58	∼1,00
10	∼0,00	∼0.61	∼1,00
$\geq$ 11	∼0,00	?	?

TABLE II
EXPERIMENTAL PROBABILITY OF SUCCESS ATTACK FOR SIMON 32/64 (WITHOUT KEY EXPAND ALGORITHM)

All of our experiments took less than one hour. After one hour of computation, the tasks were terminated. With this approach, we failed to perform a successful attack on six round of SPECK and 8 rounds of SIMON with the first attack model, where key expansion algorithm is included into system of equations.

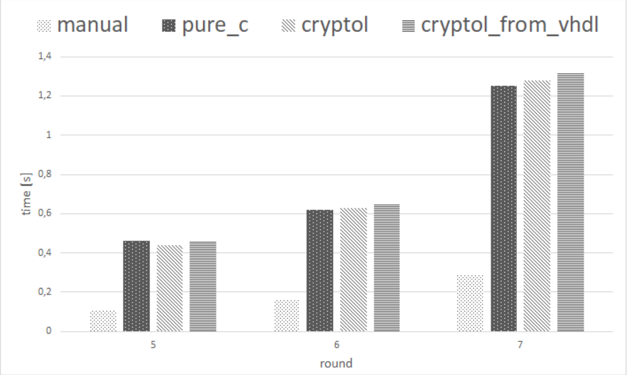


Fig. 6. Average time of attack using Lingeling for ten pairs of SIMON (without key expand algorithm)

In Tab. II and Tab. IV we report probability of success for SAT attack on SIMON and SPECK when equations from the key expand algorithm are not included. In Tab. III we report probability of success for SAT attack on SPECK when equations from key expand algorithm are included.

For the SIMON algorithm, with less than 4 pairs, the probability of finding the valid key is decreasing with every additional round of encryption and becomes negligible even for a small number of rounds. Extending the system of equations with additional pairs increases the probability of success. Moreover, only 5 pairs are required to find the valid key in the scenario, where equations for a key expansion are not included into the system of equations. This is an important observation that allows for the successful execution of a known plaintext attack with limited access to an encryption device.

Our attack for SIMON works the best for 10 pairs. The best obtained results were for handwritten equations. The equations taken from software and hardware compilers were more than three times higher.

The worst results were obtained for equations taken from HDL implementation. This might come from a high complexity of compilers and a specific form of logic element build, where only several pins of input can be mapped to up to two pins of output.

Similar results for SPECK are shown in Fig. 11. However, in Fig. 10 the results for an attack on SPECK with four pairs is presented. For a given setting, the best results were obtained for the equations taken from hardware implementation and the hand written equations provided the worst results. This is the opposite to most of the other experiments. Hardware equations were also the best for an attack on SIMON with 30

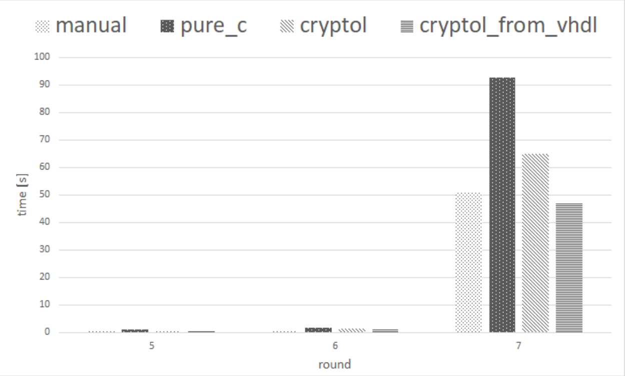


Fig. 7. Average time of attack using Cadical for thirty pairs of SIMON (without key expand algorithm)

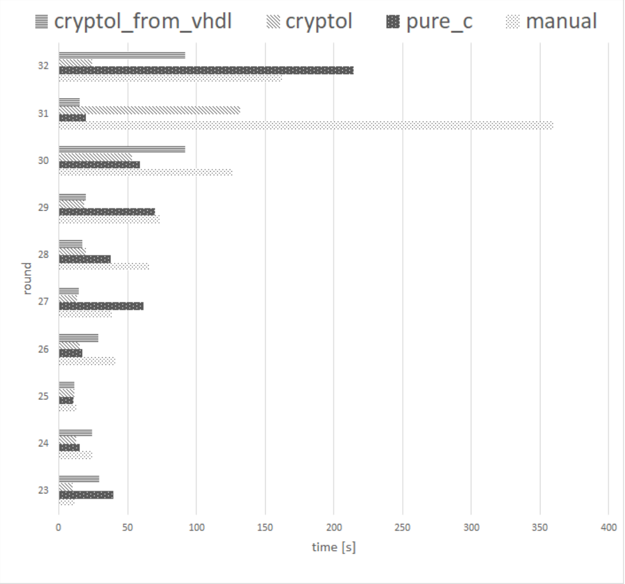


Fig. 8. Average time of attack using Cadical for three pairs of SIMON (without key expand algorithm)

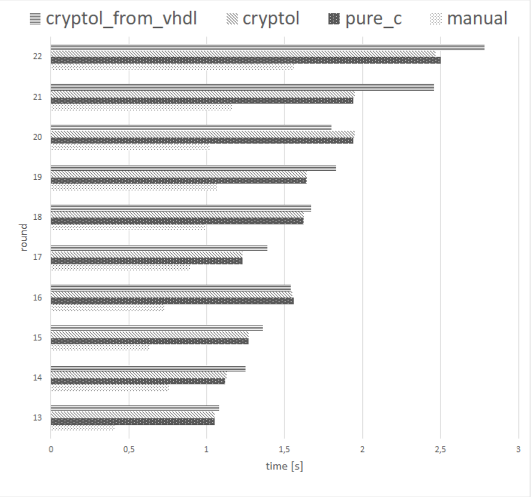


Fig. 9. Average time of attack on SPECK using Cadical for two pairs (without key expand algorithm)

pairs. The results of this attack are presented in Fig. 7. The common aspect of the two attacks is the Cadical SAT solver used for finding a solution. The proper solver and equations type selection might be crucial for obtaining the best results.

Our attack for SPECK also works the best for 10 pairs. We were able to attack up to 6 rounds with 10 pairs using Lingeling SAT to obtain a valid key in less than 600 seconds.

In Fig. 9 the time of attack for two pairs on full SPECK with Cadical is presented. The same as for SIMON, the Cadical SAT—solver was the best for two pairs and Lingeling is the best for more than 10 pairs.

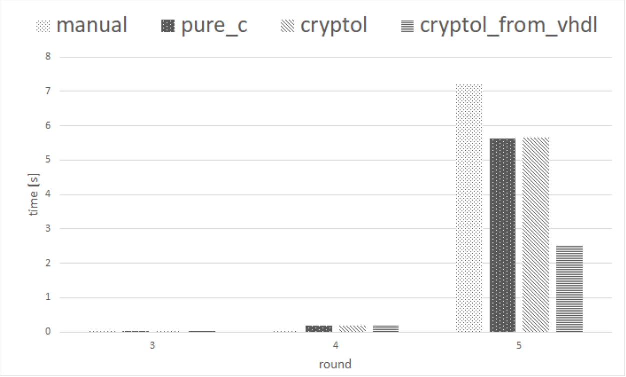


Fig. 10. Average time of attack on SPECK using Cadical for four pairs (without key expand algorithm)

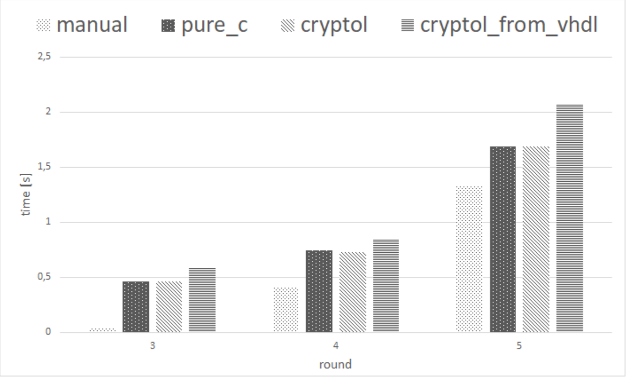


Fig. 11. Average time of attack on SPECK using Lingeling for ten pairs (without key expand algorithm)

# of rounds\# of pairs	1	2	>= 3
3	∼0,00	∼0.05	∼1,00
4	∼0,00	∼0.12	∼1,00
5	∼0,00	∼0.68	∼1,00
6	∼0,00	?	?
7	∼0,00	?	?
≥ 8	?	?	?

TABLE III
EXPERIMENTAL PROBABILITY OF SUCCESS ATTACK FOR SPECK 32/64 (WITH KEY EXPAND ALGORITHM)

# of rounds\# of pairs	2	3	>= 4
3	∼0,68	∼0.95	∼1,00
4	∼0,07	∼0.85	∼1,00
5	∼0,00	∼0.88	∼1,00
6	∼0,00	∼0.86	∼1,00
≥ 7	∼0,00	?	?

TABLE IV
EXPERIMENTAL PROBABILITY OF SUCCESS ATTACK FOR SPECK 32/64 (WITHOUT KEY EXPAND ALGORITHM)

VII. CONCLUSION

As for now, we have focused on research about the best parameters to set for an attack, rather than attacking the highest number of rounds. We have checked the influence of several parameters on the solving time of an satisfiability problem.

Considering all mentioned factors, we were able to successfully break up to 6 rounds of SPECK cipher and up to 10 rounds of SIMON cipher using our novel approach on a single-core CPU. In Tab. V we report our best results. For aforementioned ciphers, the best approach is to use the second attack scenario, where the key expansion algorithm is not included.

	number of rounds	number of pairs	equations type	SAT	time [s]
SPECK	6	10	cryptol	Lingeling	451,29
SPECK	6	10	vhdl	Lingeling	588,74
SIMON	10	10	cryptol	Lingeling	8,98
SIMON	10	10	manual	Lingeling	5,78

TABLE V
OUR THE BEST RESULTS OF SAT ATTACK ON SPECK 32/64 AND SIMON 32/64 (WITHOUT KEY EXPAND ALGORITHM)

Presented results are only a fraction of those obtained from experiments. We have shown that many factors affect the success rate and time required for SAT attack on SPECK and SIMON ciphers. Further exploration of mentioned factors like source of equations, used SAT—solver, number of pairs, number of rounds and attack model may lead to even better results affecting the claimed security of the aforementioned ciphers or even to a full break of these ciphers.

Our new approach to an SAT attack offers a significant speed-up when compared to the standard method. It also decreases the amount of work required for preparing the attack. Moreover, the attack can be applied to other ciphers and the preparation costs of the attack are very low. Thus, our tool seems to be a good choice for a plug-and-play attack on the initial security strength evaluation. Our method can be also combined with other types of attacks as reported in [7], [19]. Using pairs selected with linear or differential cryptanalysis for an SAT attack without key expansion algorithm is a promising idea for further research and obtaining even better results.

REFERENCES

[1] Nicolas T. Courtois and Gregory V. Bard. Algebraic cryptanalysis of the data encryption standard. In Steven D. Galbraith, editor, *Cryptography and Coding*, pages 152–169, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg, DOI: 10.1007/978-3-540-77272-9_10.

[2] J. Smith S. Treatman-Clark B. Weeks L. Wingers R. Beaulieu, D. Shors. The simon and speck families of lightweight block ciphers. 2016, DOI: 10.1145/2744769.2747946.

[3] Armin Biere. *CADICAL, LINGELING, PLINGELING, TREENGELING and YALSAT Entering the SAT Competition 2017*. 2017.

[4] Alex Biryukov, Arnab Roy, and Vesselin Velichkov. Differential analysis of block ciphers simon and speck. pages 546–570, 03 2014, DOI: 10.1007/978-3-662-46706-0_28.

[5] Farzaneh Abed, Eik List, Jakob Wenzel, and Stefan Lucks. Differential cryptanalysis of round-reduced simon and speck. 03 2014, DOI: 10.1007/978-3-662-46706-0_27.

[6] Qingju Wang, Zhiqiang Liu, Kerem Varici, Yu Sasaki, Vincent Rijmen, and Yosuke Todo. Cryptanalysis of reduced-round simon32 and simon48. pages 143–160, 12 2014, DOI: 10.1007/978-3-319-13039-2_9.

[7] Nicolas Courtois, Theodosis Mourouzis, Guangyan Song, Pouyan Sepehrdad, and Petr Susil. Combined Algebraic and Truncated Differential Cryptanalysis on Reduced-round Simon:. In *Proceedings of the 11th International Conference on Security and Cryptography*, pages 399–404. SCITEPRESS - Science and and Technology Publications, DOI: 10.5220/0005064903990404.

[8] J. Ren and S. Chen. Cryptanalysis of Reduced-Round SPECK. 7:63045–63056, DOI: 10.1109/ACCESS.2019.2917015.

[9] Martin Albrecht. Algorithmic algebraic techniques and their application to block cipher cryptanalysis. 4:120–141, 04 2011, DOI: 10.1007/978-3-642-22497-3_9.

[10] Carlos Cid and Ralf-Philipp Weinmann. Block ciphers: Algebraic cryptanalysis and grobner bases. In Massimiliano Sala, Shojiro Sakata, Teo Mora, Carlo Traverso, and Ludovic Perret, editors, *Grobner Bases, Coding, and Cryptography*, pages 307–327. Springer Berlin Heidelberg, DOI: 10.1007/978-3-540-93806-4_17.

[11] Gregory V. Bard. Algebraic cryptanalysis. DOI: 10.1007/978-0-387-88757-9.

[12] Ilya Mironov and Lintao Zhang. Applications of sat solvers to cryptanalysis of hash functions. volume 2006, pages 102–115, 01 2006, DOI: 10.1007/11814948_13.

[13] Joan Boyar, Magnus Find, and Rene Peralta. Four Measures of Nonlinearity. In Paul G. Spirakis and Maria Serna, editors, *Algorithms and Complexity*, pages 61–72. Springer Berlin Heidelberg, DOI: 10.1007/978-3-642-38233-8_6.

[14] Gregory Bard. Algorithms for solving linear and polynomial systems of equations over finite fields with applications to cryptanalysis.

[15] Ekawat Homsirikamol, Pawel Morawiecki, Marcin Rogawski, and Marian Srebrny. Security Margin Evaluation of SHA-3 Contest Finalists through SAT-Based Attacks. In Agostino Cortesi, Nabendu Chaki, Khalid Saeed, and Slawomir Wierzchon, editors, *Computer Information Systems and Industrial Management*, pages 56–67. Springer Berlin Heidelberg, DOI: 10.1007/978-3-642-33260-9_4.

[16] Nicolas T. Courtois, Pouyan Sepehrdad, Petr Susil, and Serge Vaudenay. ElimLin Algorithm Revisited. In Anne Canteaut, editor, *Fast Software Encryption*, volume 7549, pages 306–325. Springer Berlin Heidelberg, DOI: 10.1007/978-3-642-34047-5_18.

[17] Biere Armin. *The AIGER And-Inverter Graph (AIG) Format*. 2007.

[18] Inc Galois. *Cryptol The Language of Cryptography*. 2018.

[19] Ludovic Perret Jean-Charles Faugere and Pierre-Jean Spaenlehauer. Algebraic-differential cryptanalysis of DES. 07 2010,

Michal Andrzejczak In 2016 he obtained a master's degree in computer science with a specialization in cryptology from the Military University of Technology. He is currently a PhD student on MUT and his research interests include FPGAs and its modern applications to cryptography.

Wladyslaw Dudzic In 2015 he obtained a master's degree in computer science with a specialization in cryptology from the Military University of Technology. He is currently a PhD student on MUT and his research interests include cryptography and its modern applications, design of cryptographic algorithms (in particular block ciphers), linear and diffenerial cryptanalysis, SAT solvers and their applications in algebraic cryptanalysis.

Reducing Lattice Enumeration Search Trees

Mithilesh Kumar, Håvard Raddum, and Srimathi Varadharajan

***Abstract*—We revisit the standard enumeration algorithm for finding the shortest vectors in a lattice, and study how the number of nodes in the associated search tree can be reduced. Two approaches for reducing the number of nodes are suggested. First we show that different permutations of the basis vectors have a big effect on the running time of standard enumeration, and give a class of permutations that give relatively few nodes in the search tree. This leads to an algorithm called hybrid enumeration that has a better running time than standard enumeration when the lattice is large. Next we show that it is possible to estimate the signs of the coefficients yielding a shortest vector, and that a pruning strategy can be based on this fact. Sign-based pruning gives fewer nodes in the search tree, and never missed the shortest vector in the experiments we did.**

***Index Terms*—Lattices, SVP problem, enumeration, pruning**

I. Introduction

A *lattice* in $\mathbb{R}^n$ is the set of all integer combinations of m linearly independent vectors $b_1, b_2, ..., b_m$ in $\mathbb{R}^n$. In this work we assume $m = n$, but all results can easily be generalized. One of the most basic computational problems concerning lattices is the *shortest vector problem* (SVP): given a lattice basis as an input the task is to find a nonzero lattice vector of smallest norm.

It is known that SVP is NP-hard under randomized reductions [1]. With the current interest in post-quantum cryptography, lattice based cryptographic primitives are among the most promising candidates for achieving secure and efficient quantum safe crypto.

There are two main algorithmic techniques for the lattice problems. The first technique is called *lattice reduction*, and the best known algorithms are the famous LLL algorithm [2] and BKZ algorithm [3]. Both of these algorithms work by applying successive transformations to the input basis in an attempt to make the basis vectors short and as orthogonal as possible. A second and more basic approach, which is the focus of our work, is the *enumeration technique* which is simply an exhaustive search for finding the integer combinations of basis vectors whose norm is small enough.

The search can be seen as a depth-first search tree where internal nodes correspond to the partial assignments of the integer coefficients and the leaves correspond to the lattice points.

Previous results: In the 1980's Fincke, Pohst and Kannan studied how to improve the complexity of the standard algorithm for solving SVP at the time [4], [5], [6]. These algorithms are deterministic and based on exhaustive search of lattice points within a small convex set. In general, the running time of an enumeration algorithm heavily depends on the quality of the input basis. So, suitably pre-processing the input lattice using a basis reduction algorithm is essential before starting a lattice enumeration method.

Recently there have been other approaches using sieving and discrete pruning techniques, see [7], [8], [9], [10]. For a survey paper on lattice reduction algorithms, see [11].

In the 90's Schnorr, Euchner and Hörner introduced the *pruning* technique, by which these algorithms obtained substantial speedups [12], [13]. The rough idea is to prune away sub-trees where the probability of finding the desired lattice vector is small. This restricts the exhaustive search to a subset of all solutions. Although there is a chance of missing the desired vector, the probability of this is small compared to the gain in running time.

The pruning strategy was later studied more rigorously by Gama, Nguyen and Regev in [14] in 2010, introducing what they called *extreme pruning*. Very large parts of the search tree is cut away with extreme pruning. This makes the search very fast, but the probability of finding the shortest vector on a given run is very small. However, the authors show that the search tree is reduced more than the probability of finding the shortest vector, so one obtains a speed-up by just permuting the basis and repeating the process a number of times. The algorithm using extreme pruning is the fastest known, and today's state of the art when it comes to enumeration.

Our contribution: In this paper we propose two new ideas, and show their benefit in speeding up lattice enumeration. First, we propose a new enumeration algorithm called *hybrid enumeration* for computing intervals for the coefficients v_i. Second, we provide an algorithm for estimating the signs (+ or -) of the coefficients $v_1, v_2, ..., v_n$ in the lattice basis $\sum_{i=1}^{n} v_i \boldsymbol{b}_i$. Both these algorithms aims at reducing the size of search tree, thereby providing faster enumeration to find the shortest vector.

One disadvantage with the standard enumeration technique is that the algorithm depends on the computed Gram-Schmidt (GS) orthogonal basis for computing the intervals where the v_i-coefficients can be found. Once the GS orthogonal basis is computed, it fixes the order of the coefficients to be guessed.

In our paper, the hybrid enumeration takes a new approach by computing the intervals in a way that does not depend on GS orthogonalization. This means the basis vectors are not bound by any particular order and we are free to choose which of the untried coefficients v_i to guess on at any given point in the search tree. We show that dynamically changing the order of the guessed v_i's significantly lowers the number of nodes in the search tree compared to the standard enumeration algorithm.

The price to pay for this flexibility is increased work at each node of the search tree. Hence the actual time taken to enumerate a lattice using the new method may be longer than the time taken by the standard GS enumeration. Therefore we

All authors are with Simula UiB, Bergen, Norway

DOI: 10.36244/ICJ.2019.4.2

only propose to use the new enumeration technique at the nodes on the highest levels in the search tree, and then switch to standard GS enumeration for levels lower than that. This still leads to a significant reduction in the number of nodes in comparison with the standard enumeration method, depending on type of lattice and the level where we switch to standard GS enumeration.

The second technique we provide is to estimate the signs of each v_i. The main idea behind the algorithm is to exploit the dot product function which contains information about the length and angle between the basis vectors. Given two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, if the angle between them is less than 90 degrees then their sum $\boldsymbol{a}+\boldsymbol{b}$ is longer than both $\boldsymbol{a}$ and $\boldsymbol{b}$ and $\boldsymbol{a}-\boldsymbol{b}$ will be shorter than at least one of $\boldsymbol{a}$ and $\boldsymbol{b}$. To get a short vector we need to subtract one from another which implies that the sign of these vectors are opposite with respect to each other. Similarly, when the angle between them is more than 90 degrees, then addition gives a short vector, so their relative signs should be the same.

We generalize this observation on n vectors, developing a method for estimating the signs of each v_i together with a confidence measure for each estimate. We then give a pruning strategy where the interval computed for each v_i is cut down using the estimate of the sign and confidence factor. Unlike other pruning methods, this leads to a one-sided pruning where we only cut away a portion of possible v_i values where the sign is believed to be wrong. A useful fact is that our sign-based pruning can be applied on top of any other pruning strategy.

II. Preliminaries

Throughout the paper, we will denote all vectors in boldface type, all matrices as capital letters, and all scalars in lower case italics. Given a linearly independent set of vectors $\{\boldsymbol{b}_1, \boldsymbol{b}_2, ..., \boldsymbol{b}_n\}$ in $\mathbb{R}^n$, the lattice $\mathcal{L}$ generated by them is the set

$$\mathcal{L} = \Big\{ \sum_{i=1}^{n} v_i \boldsymbol{b}_i | v_i \in \mathbb{Z} \Big\}$$

of integer linear combination of $\boldsymbol{b}_i$'s. The set of vectors $\{\boldsymbol{b}_1, \boldsymbol{b}_2, \dots, \boldsymbol{b}_n\}$ is called the *lattice basis*.

The *inner product* of two vectors $\boldsymbol{a} = (a_1, \dots, a_n)$ and $\boldsymbol{b} = (b_1, \dots, b_n)$ is defined as

$$\boldsymbol{a} \cdot \boldsymbol{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n.$$

The *Euclidean norm* of a vector $\boldsymbol{a}$ is defined as $\sqrt{\boldsymbol{a} \cdot \boldsymbol{a}}$ and is denoted $\|\boldsymbol{a}\|$. The vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ are said to be *orthogonal* if $\boldsymbol{a} \cdot \boldsymbol{b} = 0$. Given a basis $B = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \dots, \boldsymbol{b}_n\}$ of a lattice $\mathcal{L}$, B is said to be orthogonal if for every pair of distinct vectors $\boldsymbol{b}_i$ and $\boldsymbol{b}_j$ in B are orthogonal.

A lattice $\mathcal{L}$ contains non-zero vectors of shortest length with respect to the Euclidean norm. This parameter is denoted by $\lambda_1(\mathcal{L})$. A vector of norm $\lambda_1(\mathcal{L})$ is called a *shortest vector* of $\mathcal{L}$.

A. Gram-Schmidt orthogonalization

In general, a basis B for a lattice is not orthogonal. The Gram-Schmidt process is a method for orthogonalizing a set of vectors in an n-dimensional Euclidean space $\mathbb{R}^n$. The *projection* of a vector $\boldsymbol{a}$ onto a vector $\boldsymbol{b}$ is defined as

$$P_{\boldsymbol{b}}(\boldsymbol{a}) = \left(\frac{\boldsymbol{b} \cdot \boldsymbol{a}}{\boldsymbol{b} \cdot \boldsymbol{b}} \right) \boldsymbol{b}. \tag{1}$$

The Gram-Schmidt process can then be described via the following equations:

$$\begin{aligned}
\boldsymbol{b}_1^* &= \boldsymbol{b}_1 \\
\boldsymbol{b}_2^* &= \boldsymbol{b}_2 - P_{\boldsymbol{b}_1^*}(\boldsymbol{b}_2) \\
\boldsymbol{b}_3^* &= \boldsymbol{b}_3 - P_{\boldsymbol{b}_1^*}(\boldsymbol{b}_3) - P_{\boldsymbol{b}_2^*}(\boldsymbol{b}_3) \\
&\vdots \\
\boldsymbol{b}_n^* &= \boldsymbol{b}_n - \sum_{j=1}^{n-1} P_{\boldsymbol{b}_j^*}(\boldsymbol{b}_n)
\end{aligned}$$

The set $\{\boldsymbol{b}_1^*, \boldsymbol{b}_2^*, \dots, \boldsymbol{b}_n^*\}$ is an orthogonal basis for the same space as that spanned by $\{\boldsymbol{b}_1, \boldsymbol{b}_2, \dots, \boldsymbol{b}_n\}$. More generally, for any $1 \le i \le n$ the subspace spanned by $\{\boldsymbol{b}_1^*, \boldsymbol{b}_2^*, \dots, \boldsymbol{b}_i^*\}$ is the same as that spanned by $\{\boldsymbol{b}_1, \boldsymbol{b}_2, \dots, \boldsymbol{b}_i\}$.

B. Projections

We can generalize the projection given in (1) to apply to a larger space. Let the space V be given by the basis $V = \{\boldsymbol{b}_1, \dots, \boldsymbol{b}_k\}$. The projection of a vector $\boldsymbol{a}$ onto the space V is then given by

$$P_V(\boldsymbol{a}) = P_{\boldsymbol{b}_1^*}(\boldsymbol{a}) + \cdots + P_{\boldsymbol{b}_k^*}(\boldsymbol{a}),$$

where the $\boldsymbol{b}_i^*$ form the orthogonal basis of V, giving a vector that lies inside the space V.

Lemma 1. *Let V be a subspace of $\mathbb{R}^n$. For any $\boldsymbol{a} \in \mathbb{R}^n$, the vector $\boldsymbol{a} - P_V(\boldsymbol{a})$ is perpendicular to every vector in V.*

Proof. We start with a basis $B_V = \{\boldsymbol{b}_1, \dots, \boldsymbol{b}_k\}$ for V and expand it to a basis for the entire space by adding some vectors $\boldsymbol{b}_{k+1}, \dots, \boldsymbol{b}_n$. We then apply the Gram-Schmidt process to get an orthogonal basis $K = \{\boldsymbol{b}_1^*, \dots, \boldsymbol{b}_n^*\}$ for $\mathbb{R}^n$. Then K is the concatenation of the two bases $\{\boldsymbol{b}_1^*, \dots, \boldsymbol{b}_k^*\}$ and $\{\boldsymbol{b}_{k+1}^*, \dots, \boldsymbol{b}_n^*\}$, and by the GS property, $\{\boldsymbol{b}_1^*, \dots, \boldsymbol{b}_k^*\}$ is a basis for V. Any vector $\boldsymbol{a} \in \mathbb{R}^n$ can be written as $\boldsymbol{a} = r_1 \boldsymbol{b}_1^* + \cdots + r_n \boldsymbol{b}_n^*$ where $r_i = \frac{\boldsymbol{a} \cdot \boldsymbol{b}_i^*}{\boldsymbol{b}_i^* \cdot \boldsymbol{b}_i^*}$.

By definition we have $P_V(\boldsymbol{a}) = r_1 \boldsymbol{b}_1^* + \cdots + r_k \boldsymbol{b}_k^*$. Hence, $\boldsymbol{a} - P_V(\boldsymbol{a}) = r_{k+1} \boldsymbol{b}_{k+1}^* + \cdots + r_n \boldsymbol{b}_n^*$. Since any vector $\boldsymbol{u}$ in V is a linear combination of the vectors in $\{\boldsymbol{b}_1^*, \dots, \boldsymbol{b}_k^*\}$, we have $\boldsymbol{u} \cdot (\boldsymbol{a} - P_V(\boldsymbol{a})) = 0$. □

The following lemma provides us with a way to compute the projection of a vector onto a space V, without needing to orthogonalize the basis for V.

Lemma 2. *Let $\boldsymbol{a}$ be a vector in $\mathbb{R}^n$ and let V be a subspace of $\mathbb{R}^n$ with basis $B_V = \{\boldsymbol{b}_1, \dots, \boldsymbol{b}_k\}$. Let A be the matrix with $\boldsymbol{b}_1, \dots, \boldsymbol{b}_k$ as columns. Then*

$$P_V(\boldsymbol{a}) = c_1 \boldsymbol{b}_1 + \cdots + c_k \boldsymbol{b}_k$$

where the c_i are the entries of the vector $(A^TA)^{-1}A^T\boldsymbol{a}$. In particular, the projection can be computed as $P_V(\boldsymbol{a}) = A(A^TA)^{-1}A^T\boldsymbol{a}$.

Proof. Since the columns of A are the basis vectors for V, we can write $P_V(\boldsymbol{a}) = c_1\boldsymbol{b}_1 + \cdots + c_k\boldsymbol{b}_k = A\boldsymbol{c}$ for some values c_i. By Lemma 1, the vector $\boldsymbol{a} - P_V(\boldsymbol{a})$ is orthogonal to every vector in V. Hence $A^T(\boldsymbol{a} - P_V(\boldsymbol{a})) = \boldsymbol{0}$ since the matrix/vector multiplication is just taking the inner product of basis vectors with $(\boldsymbol{a} - P_V(\boldsymbol{a}))$. Substituting for $P_V(\boldsymbol{a})$, we get $A^T\boldsymbol{a} - A^TA\boldsymbol{c} = \boldsymbol{0}$ which implies that $\boldsymbol{c} = (A^TA)^{-1}A^T\boldsymbol{a}$. Hence, $P_V(\boldsymbol{a}) = A(A^TA)^{-1}A^T\boldsymbol{a}$. □

C. The standard enumeration algorithm

Let $\mathcal{L}$ be a lattice whose shortest vector v is unique up to the sign. Assume we are given the basis $\{\boldsymbol{b}_1, \boldsymbol{b}_2, ..., \boldsymbol{b}_n\}$ of $\mathcal{L}$ and an upper bound R on $\lambda_1(\mathcal{L})$ such that we need to find all vectors $\boldsymbol{w}$ in the lattice $\mathcal{L}$ that satisfy $\|\boldsymbol{w}\| \leq R$.

The shortest vector $\boldsymbol{s} \in \mathcal{L}$ can be written as $\boldsymbol{s} = v_1\boldsymbol{b}_1 + v_2\boldsymbol{b}_2 + ... + v_n\boldsymbol{b}_n$ where the $v_i's$ are unknown integers and $\boldsymbol{b}_i = \sum_{j=1}^{i-1}\mu_{i,j}\boldsymbol{b}_j^*$, where $\mu_{i,j} = (\boldsymbol{b}_i \cdot \boldsymbol{b}_j^*)/(\boldsymbol{b}_j^* \cdot \boldsymbol{b}_j^*)$ are the Gram-Schmidt coefficients. Our goal is to find $\boldsymbol{s}$.

To find $\pm\boldsymbol{s}$, the enumeration goes through an enumeration tree formed by the subspace spanned by the vectors whose norm is at most R. The enumeration tree is a depth first search tree of depth n. Each internal node in the tree is associated with a particular v_i and each outgoing edge represents an assignment of an integer value (obtained from a range) to v_i. In particular the root of the tree is the zero vector, while the leaves are all the vectors of $\mathcal{L}$ whose norm is at most R.

At any node, the enumeration algorithm selects an index i not yet branched for, obtains a set of integers (interval range) I_i for the possible values v_i can take and for each integer $t \in I_i$ the algorithm calls itself recursively to compute the interval for the next level. The length bound here remains constant throughout the algorithm. For $1 \leq k \leq n$, the following inequality (see [14]) needs to be satisfied, essentially defining the interval I_k:

$$\Big(v_k + \sum_{i=k+1}^{n}\mu_{i,k}v_i\Big)^2\|b_k^*\|^2 + \sum_{j=k+1}^{n}\Big(v_j + \sum_{i=j+1}^{n}\mu_{i,j}v_i\Big)^2\|b_j^*\|^2 \leq R^2 \qquad (2)$$

By the inequality above, for each $1 \leq k \leq n$ the interval range I_k for v_k can be obtained if v_j is known for each $j \in \{k+1, k+2, ..., n\}$. This implies that in the enumeration algorithm, the indices i can only be chosen in the order starting from $n, n-1, ..$ down to 1. In the rest of the paper we refer to the root node of the search tree being at level n, the second highest level being level $n-1$, etc. That is, if a node is at level l in the search tree, then only the coefficient v_l can be selected for branching at that node.

III. HYBRID ENUMERATION

In this section we study how permutations of the basis vectors of a lattice affects the running time of enumeration. Based on this we present a good strategy for selecting an order of the basis vectors that results in relatively small search trees when doing enumeration. This can help speed up extreme pruning, by only selecting permutations that give small search trees when iterating the extremely pruned enumeration runs.

nodes in search tree	BKZ-10	BKZ-20
minimum	60.934.596	4.059.025
average	424.300.658	52.886.123
maximum	1.180.735.200	194.214.522
std. deviation	361.710.571	40.202.374

TABLE I: Number of nodes to fully enumerate the BKZ-reduced SVP40 challenge lattice for 20 random permutations of the basis. The number of nodes in a search tree is highly dependent on the particular permutation.

A. Variations in Enumeration Complexity from Basis Permutations

As far as we know, there have been no studies of how the complexity of standard enumeration varies when the vectors in the input basis are permuted. To motivate the work that follows, we first present the results of some experiments showing that the number of nodes in the search tree when doing full enumeration is highly sensitive with respect to the order of the basis vectors.

The lattice we use for the demonstration is Darmstadt's SVP40 challenge [15], generated from seed 0. The experiment was done as follows: First, we ran two BKZ-reductions on the SVP40 lattice, one with block size 10 and one with block size 20. Then we did full enumeration of each of the two BKZ-reduced lattices, counting the number of nodes in the search tree. Next we randomized the two BKZ-reduced bases 20 times each, and ran full enumeration on all of them. The average number of nodes in the search trees for the randomized bases are shown in Table I, together with the maximum and minimum numbers observed and the standard deviation.

From Table I we see that the order of the basis vectors has a big impact on the size of the enumeration search tree. The standard deviation is of similar size as the average, showing that the sizes of the search trees vary greatly with the permutation.

Another interesting thing we observed is that the order of the reduced basis as given straight out of BKZ is particularly good for enumeration. Enumerating the SVP40 challenge with the basis order given by BKZ-10 gives a tree with 5.968.085 nodes, and the order given by BKZ-20 gives a tree with 1.232.737 nodes, significantly smaller than the numbers observed for any of the random permutations.

B. Intervals for coefficients

Given a length bound R, basic enumeration will search exhaustively for all lattice vectors of length less than or equal to R. Assume that $\boldsymbol{s} = v_1\boldsymbol{b}_1 + v_2\boldsymbol{b}_2 + \ldots + v_n\boldsymbol{b}_n$ is a vector such that $\|\boldsymbol{s}\| \leq R$. Before the enumeration can start, the μ-matrix $[\mu_{i,j}]$ of Gram-Schmidt coefficients and the orthogonal basis vectors $\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_n^*$ must be computed. The μ-matrix is dependent on the particular order of the basis vectors, and once it is computed this order remains fixed throughout the standard enumeration routine.

The actual enumeration starts by computing an interval I_n such that $\|\boldsymbol{s}\| \leq R$ implies $v_n \in I_n$. The algorithm then fixes an integer value in I_n for v_n, and based on the choice computes an interval I_{n-1} such that $\|\boldsymbol{s}\| \leq R$ implies $v_{n-1} \in I_{n-1}$. Then an integer is selected from I_{n-1} and assigned to v_{n-1}, and the interval where v_{n-2} must be found is computed. This continues until a selection for v_1 can be made, in which case we find a lattice vector with length less than R, or until an interval I_j that contains no integers is computed.

Intervals are computed recursively in the order $I_n, I_{n-1}, \ldots, I_2, I_1$, and all values from all intervals must be tried to do a complete search that guarantees that a shortest vector will be found. In the following, we denote the length of an interval I_i by $|I_i|$.

Basic enumeration assumes the μ-matrix is computed once and for all before actual enumeration starts, but this is not strictly necessary. We can set every basis vector $\boldsymbol{b}_i$ in the basis as the last one, recompute the μ-matrix, and find the interval of possible coefficients for I_i. Doing this allows us to make a choice of which vector to first fix the coefficient for. For instance, we may select the basis vector giving the shortest interval as the first one to branch for.

This strategy can be generalized and done at any point during enumeration: Assume v_j for $j \in J \subseteq \{1, \ldots, n\}$ have been fixed, where $|J| = k$. All remaining basis vectors $\boldsymbol{b}_i$ for $i \in (\{1, \ldots, n\} \setminus J)$ can be tried by placing them successively in position $n-k$ in the basis. The μ-matrix and the coefficient intervals are re-computed for every choice, and the vector giving the shortest interval is selected as the next one to branch for. In this way we may dynamically change the order of which basis vector to branch for, while the enumeration algorithm is running.

Remark: To compute the smallest interval at a given node, we do not need to re-compute the full μ-matrix. We only need to re-compute the entries in μ from the point where we have changed the order of the basis vectors. For example, if we are computing the interval for v_j, only the rows of μ with indices higher than j needs to be updated when setting $\boldsymbol{b}_j$ last.

C. Strategy for selecting order for basis vectors

The strategy we use for choosing the order of basis vectors to branch for follows a greedy approach: We always choose the next v_i to try as the one with the shortest interval I_i. The rationale for this strategy can be explained via the following lemma, basically saying that the interval for one v_i shortens, when more of the other coefficients are fixed.

Lemma 3. *Let $J_1 \subseteq J_2 \subseteq (\{1, \ldots, n\} \setminus \{i\})$. Let $I_i(J_1)$ be the interval for v_i after values of $v_j, j \in J_1$ have been fixed, and let $I_i(J_2)$ be the interval for v_i after some additional v_j's, $j \in J_2 \setminus J_1$ have been fixed. Then $|I_i(J_1)| \geq |I_i(J_2)|$.*

Proof. From Equation (2) we see that the length of $I_i(J_1)$ is determined by the sum

$$\sum_{j=k+1}^{n} \left(v_j + \sum_{i=j+1}^{n} \mu_{i,j} v_i \right)^2 \left\| b_j^* \right\|^2, \tag{3}$$

while the center of the interval is determined by

$$\sum_{i=k+1}^{n} \mu_{i,k} v_i.$$

When we branch in an unspecified order, (3) can be written as

$$\sum_{j \in J_1} t_j^2,$$

where the t_j's are terms decided by the specific order in which the indices in J_1 were chosen. The larger this sum becomes, the smaller $|I_i(J_1)|$ will be. The terms in the sum are all positive, so expanding with the extra terms to create the sum $\sum_{j \in J_2} t_j^2$ before branching for v_i can only decrease the length of I_i. Hence $|I_i(J_1)| \geq |I_i(J_2)|$. □

Lemma 3 shows that the longer we wait to select a particular v_i to branch for, the shorter its interval I_i will become. The idea for the branching strategy is that intervals that are long when few v_j's have been selected will become short by the time the algorithm is forced to branch on them. This will lead to relatively small search trees.

One way to more easily see this is in the case when one I_i becomes empty after fixing the v_j's for $j \in J$, for some J. Say the branching order has been fixed from the start, the values of v_j, $j \in J$ have been fixed, and that I_i is empty, but v_i is only to be branched for after another 10 v_k's have been fixed. Even though it is clear (if we compute I_i) that all choices of values for the v_k's will lead to a dead end, the traditional enumeration algorithm will try all of them before backtracking away from this sub-tree. By always selecting the next v_i to branch for as the one with the shortest interval, v_i will be selected as soon as $|I_i| = 0$ (the shortest length possible), and backtracking into the v_j's where $j \in J$ will start immediately.

D. Cost vs effect for minimizing intervals

The drawback of checking which of the remaining indices to branch for is the extra work done in each node. If we compute an interval I_i using the μ-matrix of Gram-Schmidt coefficients, we in general have to recompute the μ-matrix as part of the process. The complexity for computing this matrix for one index is $\mathcal{O}(n^3)$ multiplications, and doing this for every remaining index not yet branched for gives overall complexity of $\mathcal{O}(n^4)$ in each node.

Computing an interval I_i using the projection method involves inverting a matrix, which also has complexity $\mathcal{O}(n^3)$. Repeating for all unbranched indices again gives an overall complexity of $\mathcal{O}(n^4)$ for the work done in each node. These complexities are quite high considering they have to be done for each node. However, they are still polynomial and the number of nodes in a search tree is super-exponential in n, so if the reduction in the number of nodes is big enough it is still worthwhile.

As we saw in Table I, the number of nodes in a search tree without using any minimizing strategy depends heavily on the order of the basis vectors. The order of the basis vectors does not matter when using the minimizing strategy as the vectors will be sorted as part of the enumeration routine. Hence it is hard to say anything in general about how large the effect of minimizing intervals will have, since it depends on how ”lucky” the initial order of the vectors is.

We have tested the minimizing strategy on random lattices of relatively small dimensions ($10 \leq n \leq 20$), and compared the number of nodes in these search trees with the number of nodes in the search trees using standard enumeration. The minimizing strategy indeed leads to search trees with much fewer nodes, on the average the reduction is approximately by a factor n for the small dimensions we looked at. As the increase in workload in each node is by a factor $\mathcal{O}(n^4)$, applying the minimizing strategy in every node is not worth the extra effort.

E. Hybrid enumeration

When values for many v_j's have been assigned (for $j \in J$), the effect of minimizing intervals for the relatively few remaining indices in $\{1, \ldots, n\} \setminus J$ is small. On the other hand, applying the minimizing strategy on the very first v_j's to be fixed has a much greater effect. The number of large sub-trees rooted high up in the full tree when no ordering strategy is applied, become significantly smaller when minimizing intervals. In the extreme case of some interval becoming empty, the whole sub-tree gets pruned away.

Thus we propose to only apply the minimizing strategy on the relatively few nodes at the highest levels of the search tree. This has the benefit of a relatively low cost for a high effect. We call enumeration with the strategy of minimizing intervals for the first few levels of the tree for *hybrid enumeration*.

One parameter for hybrid enumeration is the level in the tree where we switch from finding an optimal order based on minimizing intervals to classic enumeration where the basis is in some given and fixed order. We call this parameter the *switch level*.

More precisely, when we reach a node at the switch level we do the following: We compute the interval lengths for remaining indices one last time, and permute the remaining basis vectors according to these lengths. Indices with the shortest intervals will be branched for first. Then we do normal enumeration for the sub-tree rooted at the current node, using this fixed order for the whole sub-tree. Pseudo code for hybrid enumeration is given in Algorithm 1.

For $B = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$, we regard the root node of the tree (at the top) to be at level n, and the short vectors of $\mathcal{L}(B)$ will be found at level 0. Note that we can run basic enumeration of the lattice by calling HybridEnumerate($B, R, n+1, n$). Calling HybridEnumerate(B, R, n, n) will also run basic enumeration, but the basis is first permuted according to the strategy of minimizing intervals. This makes it easy to compare the benefit of using hybrid enumeration over basic enumeration.

Algorithm 1 HybridEnumerate(B, R, sl, l)

```
Input: The basis vectors $B = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ of a lattice $\mathcal{L}$, a length bound $R$, the current level $l$, and the switch level $sl$.
Output: All vectors $\boldsymbol{s} \in \mathcal{L}$ with $\|s\| \leq R$

if $l > sl$ then
    $I_i \leftarrow$ shortest interval for $\boldsymbol{b}_i \in B$
    for $v_i \in I_i$ do
        $r \leftarrow$ min. length added to $\|\boldsymbol{s}\|$ due to choice of $v_i$
        HybridEnumerate($B \setminus \{\boldsymbol{b}_i\}, R - r, sl, l - 1$)
    end for
end if
if $l = sl$ then
    Compute intervals $I_j, \forall \boldsymbol{b}_j \in B$
    Sort $B$ according to $|I_j|$, basis vectors on bottom of $B$ has shortest intervals
    HybridEnumerate($B, R, sl, l - 1$)
end if
if $l < sl$ then
    Run standard enumeration on $B$ with length bound $R$
end if
```

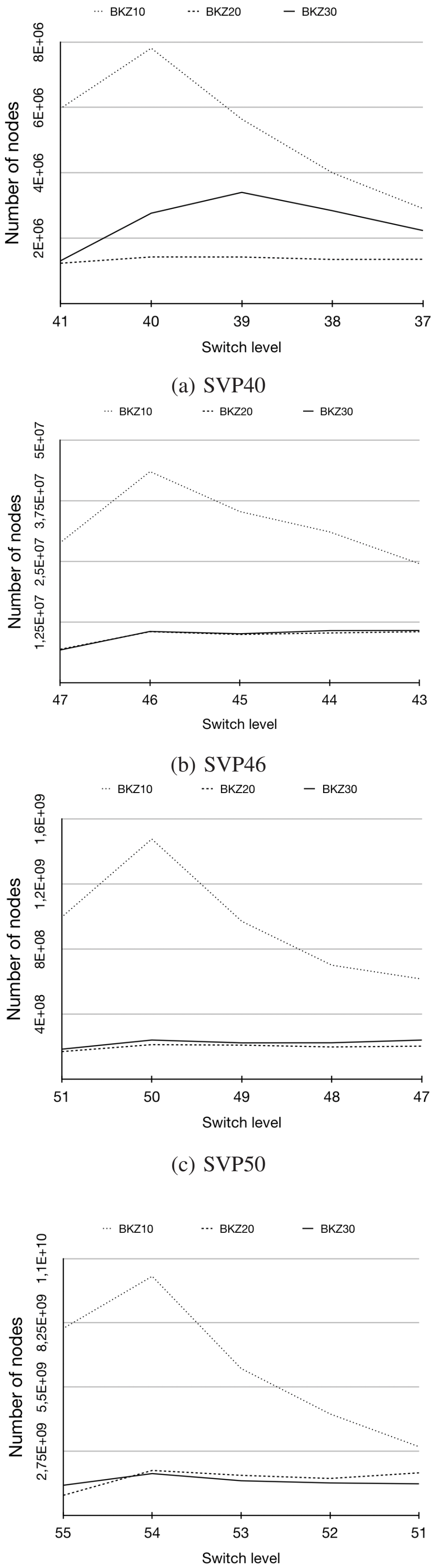
F. Experiments

We have tested hybrid enumeration on several of the SVP challenges of [15] and counted the number of nodes hybrid enumeration gives for different switch levels. The lattice bases were first reduced by running BKZ-β on them, for $\beta \in \{10, 20, 30\}$. For each reduced lattice, we ran hybrid enumeration with switch levels ranging from $n+1$, equivalent to standard enumeration, to $n-4$, counting the nodes in each search tree. The results are shown as plots in Figure 1.

We see a few trends from these plots. First, there is not much difference between BKZ-20 and BKZ-30 regarding the quality of the bases. Both of them give search trees with approximately the same number of nodes, and applying the strategy of minimizing intervals does not change this by much. Also, the order of the basis vectors given by hybrid enumeration yields search trees approximately as small as the order given by BKZ. This is in contrast to the random orders used for computing the numbers in Table I, that shows a large increase in the number of nodes. Hence the strategy of sorting the basis vectors according to interval lengths clearly is a good approach.

For the BKZ-10 reduced bases, we see a much bigger effect. First, we see that BKZ-10 gives a significantly weaker reduction than BKZ-20 or BKZ-30, leading to larger enumeration search trees. The order as given by BKZ-10 is still good for enumeration, and doing one initial sorting of the basis according to interval lengths (switch level n) increases the search tree. However, lowering the switch level has a clear impact and significantly reduces the number of nodes in the search tree, beyond the low number of nodes given by the initial BKZ-order.

Of course, what matters in the end for a lattice enumeration algorithm is its complexity, measured in the actual time taken. We recorded the times taken in all the experiments, to see if the extra work done in the nodes at and above the switch level is worth the effort. For the enumeration of BKZ-20 and BKZ-30 reduced bases it is clearly not worth the effort as the number of nodes stay almost the same for the various switch



(a) SVP40

(b) SVP46

(c) SVP50

(d) SVP54

Fig. 1: Number of nodes using hybrid enumeration on lattice bases pre-processed with BKZ-β for $\beta \in \{10, 20, 30\}$.

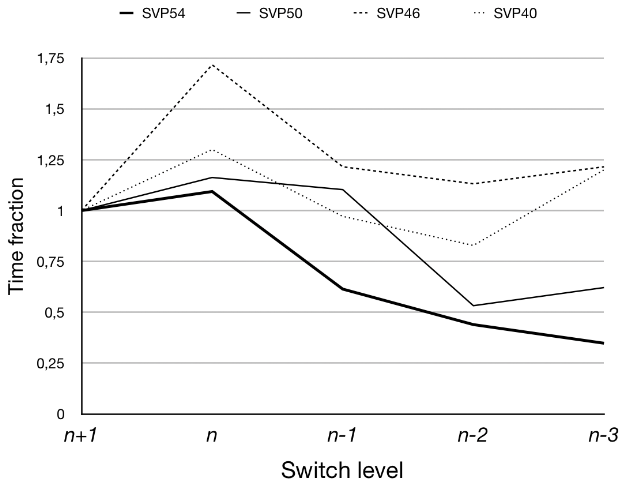


Fig. 2: Fraction of time taken for doing full hybrid enumeration on BKZ-10 reduced lattice bases, compared to time taken for standard enumeration.

levels. For enumerating the lattices only reduced by BKZ-10, there is a significant decrease in the number of nodes as the switch level decreases. Is this enough to weigh up for the $\mathcal{O}(n^4)$ work done in each node at and above the switch level?

In Figure 2 we have plotted the fraction of time taken for enumerating the four lattices we have used, compared to standard enumeration (switch level $n + 1$). The typical time taken for running these instances ranged from about one minute for the SVP46 basis, to about 24 hours for the SVP54, both pre-processed with BKZ-10. The experiments were run on a DELL computer running Linux with two 2.8 GHz AMD EPYC 7451 24-Core processors and 188 GB of RAM.

We observe a few things from Figure 2. First, except for SVP46, the time it takes to do hybrid enumeration is less than the time for doing standard enumeration, for some switch level. Using switch level n gives an increase in time because of an increase in the number of nodes. For deeper switch levels the reduction in the number of nodes is actually worth the extra work done in the few nodes at the top. Second, for the bigger lattices, the time saving is largest, with full hybrid enumeration for SVP54 using switch level 51 only taking 34.8% of the time it takes to do full standard enumeration. Third, we also see there is an optimal switch level. For SVP40 and SVP50, hybrid enumeration takes longer for switch level $n - 3$ than for $n - 2$, even though the number of nodes is less for switch level $n - 3$. The reduction in the number of nodes is then not worth the extra work for all nodes on level $n - 3$.

Figure 2 is only for BKZ-10 reduced bases, and for better BKZ reductions we do not demonstrate an improvement in running time. However, the lattices we are able to do full enumeration for in practice have dimensions in the range 40 - 60, and a block size of 20 and 30 when running BKZ is then a large portion of that. We see in the plots that there is not much difference between BKZ-20 and BKZ-30 reduced bases, and there is hardly any improvement to be done for these cases. They appear to be quite optimal from the start.

We conjecture that for higher dimensions, like $n = 150$, BKZ-30 would not give an optimally reduced basis, and that hybrid enumeration then would show the same improvements as we see with the BKZ-10 reduced bases in our experiments. All in all, we claim that if one wants to do full enumeration on large lattices that are not optimally reduced, then hybrid enumeration will be faster than standard enumeration.

IV. Sign-based pruning

Going back to the expansion of a shortest vector in terms of the basis vectors $\boldsymbol{s} = \sum_{i=1}^{n} v_i \boldsymbol{b}_i$, an enumeration algorithm computes possible values for each coefficient v_i. The equation for computing the coefficients v_i indicates that the range I_i for v_i is likely to contain both positive and negative values. As both $\boldsymbol{s}$ and $-\boldsymbol{s}$ are shortest vectors, we are content in finding either of those. If we could a priori *know* the sign of these integers (that is whether $v_i \leq 0$ or $v_i \geq 0$), we could discard appropriate values from I_i, making the enumeration tree smaller. Effectively, this would provide us with another strategy for pruning. In this section, we describe an algorithm for making educated guesses for the signs of these coefficients and how to use them for pruning. In the following we assume that the lattice basis has been reduced, and that the lengths of the basis vectors are of low variance.

A. Sign-estimation

First we show how to compute the signs of the coefficients of the shortest vector when the dimension on the given lattice is only 2. Let us consider a lattice in 2 dimensions with basis vectors $\{\boldsymbol{b}_1, \boldsymbol{b}_2\}$. If $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ are obtuse to each other (i.e. the angle between them is more than 90°), then a shortest vector $\boldsymbol{s} = v_1\boldsymbol{b}_1 + v_2\boldsymbol{b}_2$ can only be obtained if the signs of v_1 and v_2 are the same. Similarly, if they are acute (angle less than 90°) to each other, a shortest vector can only be obtained if the signs of v_1 and v_2 are opposite to each other. It is easy to see this, as a (positive) sum of two vectors pointing in approximately the same direction can only increase in length.

To extend this observation to higher dimensions we define the dot-product matrix M, where $M_{ij} = \boldsymbol{b}_i \cdot \boldsymbol{b}_j$. Two vectors have a positive dot product when the angle between them is less than 90° and a negative dot product when the angle between them is larger than 90°. Moreover, the magnitude of $\boldsymbol{b}_i \cdot \boldsymbol{b}_j$ relative to the product of the lengths of $\boldsymbol{b}_i$ and $\boldsymbol{b}_j$ is a measure of how parallel or anti-parallel $\boldsymbol{b}_i$ and $\boldsymbol{b}_j$ are.

The algorithm for computing the sign of coefficients is shown in Algorithm 2. The algorithm computes a vector $\boldsymbol{\sigma}$ of signs with entries $+1$ or -1. The sign for the coefficients v_i are computed one at a time, and the estimated sign of v_i depends on the signs of coefficients that have already been computed. Intuitively, the algorithm compares each basis vector with some reference vector to estimate the sign of the corresponding coefficient.

The sign of the first basis vector $\boldsymbol{b}_1$ is set to be positive by default, so $\sigma_1 = +1$. This is without loss of generality since both $\boldsymbol{s}$ and $-\boldsymbol{s}$ are shortest vectors and at least one of them must have non-negative v_1. The vector $\boldsymbol{b}_1$ is set as the reference vector $\boldsymbol{a}$ for the next basis vector. The first row of M contains the inner product of $\boldsymbol{b}_1 (= \boldsymbol{a})$ with all the other basis vectors. The basis vector with the largest inner product in absolute value is both a relatively long vector, and makes an angle close to 0° or 180° with $\boldsymbol{b}_1$. Let $\boldsymbol{b}_i$ be this basis vector. Then the sign of v_i is set to -1 if $M_{1i} > 0$, otherwise σ_i is set to $+1$. The reference vector is updated to $\boldsymbol{a} = \boldsymbol{a} + \sigma_i \boldsymbol{b}_i$.

Now we want to find a basis vector which is *most* parallel or anti-parallel to $\boldsymbol{a}$. For this we look at the largest entry in the vector $D = M_1 + \sigma_i M_i$, where M_1 is the top row of M and M_i is the i'th row. The largest entry in absolute value in D (except for index 1 and i) indicates the third vector, say $\boldsymbol{b}_j$, to estimate the sign for. If $D_j > 0$ then $\sigma_j = -1$, and if $D_j \geq 0$, $\sigma_j = +1$. The vector $\sigma_j \boldsymbol{b}_j$ is added to $\boldsymbol{a}$ and D is updated to $D = D + \sigma_j M_j$. The algorithm continues like this until all basis vectors have had their signs estimated.

Algorithm 2 ComputeSign(B)

```
Input: The basis vectors $B$ of the lattice $\mathcal{L}$.
Output: A vector $\boldsymbol{\sigma}$ that contains the estimated sign of each coefficient $v_i$ in $\boldsymbol{s} = \sum_i v_i \boldsymbol{b}_i$ where $\boldsymbol{s}$ is a shortest vector, and a vector $\boldsymbol{\gamma}$ of real values indicating confidence for each estimate.

Compute dot-product matrix $M$ such that $M_{ij} = \boldsymbol{b}_i \cdot \boldsymbol{b}_j$.
Initialize $D := M_1$ where $M_1$ is the top row of $M$.
Set $\sigma_1 = +1$ and $\gamma_1 = 1$.
Set reference lattice vector $\boldsymbol{a} := \boldsymbol{b}_1$
Set the counter $n_s := 1$.
while $n_s \leq n$ do
    Let $i$ be the index of $\max\{|D_j|  |j$ is not among already fixed signs$\}$.
    if $D_i > 0$ then
        Set $\sigma_i = -1$
    else
        Set $\sigma_i = +1$
    end if
    Set $\boldsymbol{a} = \boldsymbol{a} + \sigma_i \boldsymbol{b}_i$
    Set $D = D + \sigma_i M_i$
    Compute $\gamma_i = |\frac{\boldsymbol{a} \cdot \boldsymbol{b}_i}{\|\boldsymbol{a}\| \|\boldsymbol{b}_i\|}|$
    Set $n_s = n_s + 1$
end while
```

The signs computed in Algorithm 2 are not necessarily correct for a shortest vector. For each variable v_i, we compute a number $0 \leq \gamma_i \leq 1$ to denote how confident we are that the computed σ_i is correct. When $\gamma_i = 1$ we are certain that the corresponding σ_i is correct and $\gamma_i = 0$ means we have no knowledge whether the sign for v_i should be positive or negative. We compute the confidence values of the estimated signs as follows: Let $J \subset \{1, \ldots, n\}$ be the set of indices for which values have been fixed and let the reference vector be $\boldsymbol{a} = \sum_{j \in J} \sigma_j \boldsymbol{b}_j$. Then the confidence value for the σ_i estimate is given as $\gamma_i = |\frac{\boldsymbol{a} \cdot \boldsymbol{b}_i}{\|\boldsymbol{a}\| \|\boldsymbol{b}_i\|}|$.

The intuition behind this measure for confidence is that if two vectors are very close to being parallel, then having the same sign on the coefficients of these vectors will always lead to a longer vector as their sum, pointing approximately in the same direction as the other two. In order to be part of a short

Lattice	Pre-processing	node fraction	shortest vector found
SVP40	BKZ10	0.670	yes
SVP40	BKZ20	0.745	yes
SVP40	BKZ30	0.665	yes
SVP46	BKZ10	0.682	yes
SVP46	BKZ20	0.750	yes
SVP46	BKZ30	0.800	yes

TABLE II: Measure of effect of sign-based pruning. The node fraction is the number of nodes in pruned search tree compared to the number of nodes in the full enumeration search tree.

vector $\boldsymbol{s}$, the other basis vectors must be able to offset this long vector. If the signs of the coefficients are opposite, a sum of the two approximately parallel basis vectors would be much shorter. It is easier to sufficiently offset a short vector than a long one, in order to find the shortest vector overall.

When two vectors are close to being parallel then $\frac{\boldsymbol{a}\cdot\boldsymbol{b}_i}{\|\boldsymbol{a}\|\|\boldsymbol{b}_i\|}$ is close to being 1, and when two vectors are close to being anti-parallel $\frac{\boldsymbol{a}\cdot\boldsymbol{b}_i}{\|\boldsymbol{a}\|\|\boldsymbol{b}_i\|}$ is close to being -1. In both cases $\gamma_i \approx 1$.

On the other hand, when $\boldsymbol{a}$ and $\boldsymbol{b}_i$ are close to orthogonal (i.e. $\boldsymbol{a}\cdot\boldsymbol{b}_i \approx 0$), then $\boldsymbol{a}+\boldsymbol{b}_i$ and $\boldsymbol{a}-\boldsymbol{b}_i$ will be of almost equal lengths, and we can only to a little extent distinguish which of the two cases that will be most easily offset by the other basis vectors. The confidence value will therefore be close to 0 in this case.

We now turn to how we use the confidence values to prune intervals in the search tree.

B. Pruning intervals based on sign estimation

We can use the sign estimations and their confidence values to shorten the intervals computed for enumeration, while still maintaining a high probability we do not prune away all shortest vectors.

For a node in the search tree where possible values for v_i are tried, let I_i be the interval computed for v_i. Let $I_i^+ := I_i \cap [0,\infty)$ and $I_i^- := I_i \cap (-\infty, 0]$. For an interval $I := [l, m]$ and a positive number $\alpha \in \mathbb{R}$, let us define the interval αI to be $[\alpha l, \alpha m]$. If $\sigma_i = -1$, then I_i is pruned to $I_i = (1-\gamma_i)I_i^+ \cup I_i^-$. If $\sigma_i = +1$, then I_i is pruned to $I_i = (1-\gamma_i)I_i^- \cup I_i^+$. In other words, we cut away a portion of the interval where we believe a correct value for v_i will not be found. The portion cut away is proportional to the confidence we have in our estimate.

An advantage of this pruning strategy is that it can be put on top of any other pruning strategy. The sign-based pruning does not depend on how the intervals are computed. This pruning strategy reduces the search tree as long as the given intervals are non-empty and cuts away integer values that are opposite in sign to the predicted sign.

C. Experiments

We have used a few of the SVP challenge lattices to test the sign-based pruning strategy. We measured both the reduction in the number of nodes in the search tree, and whether the pruning failed to find the shortest vector. The results are summarized in Table II.

What we see in Table II is that in the experiments we never failed to find the shortest vector, and that the reduction in the number of nodes is by a modest but still significant fraction. One explanation for this is that we cut away the ends of the intervals, which only takes away small subtrees from the whole enumeration tree. The v_i-values found at the ends of the intervals are those that consume much of the length limit R when selected, probably quickly leading to dead ends anyway. Cutting away these values may not prune away very large parts of the search tree. Still, it is worthwhile to apply the sign-based pruning as it costs practically nothing in terms of extra complexity. The actual run times are cut down by almost the same fraction as the reduction in the number of nodes.

V. CONCLUSIONS

Public key encryption schemes based on lattices are one of the most promising approaches for achieving quantum safe crypto, and it is important to understand the hardness of the SVP problem on which they are based. Lattice enumeration plays a central role in the best known methods for solving SVP, so studying how to speed up lattice enumeration is important for assessing the security of lattice-based encryption. In this paper we have explored two different ideas for speeding up lattice enumeration.

First we looked at how permutations of the basis vectors of a lattice affect the running time of the standard enumeration algorithm. We demonstrate that the particular order of the basis vectors have a big impact on the number of nodes in the search tree and the running time. Next we identified particular permutations that give relatively small search trees. Dynamically finding the best permutations has a high cost on its own. However, if the lattice dimension is big enough and the pre-processing does not leave a strongly reduced basis, it is well worth the effort to apply the strategy in the relatively few nodes at the top of the search tree. We call this type of enumeration for hybrid enumeration.

Secondly, we looked at the possibility of estimating the signs of the coefficients giving a shortest vector. We can only estimate the signs with some degree of confidence, but the estimates and the confidence values lead directly to a pruning strategy. Unlike other pruning strategies that cuts away values from both ends of the interval where a coefficient v_i can be found, sign-based pruning only cuts values from one side of the interval (the side where the values have the ”wrong” sign). Sign-based pruning can therefore be applied together with any other pruning strategy one may use.

The experiments of sign-based pruning give a reduction in the number of nodes in the search tree compared to standard enumeration, but the reduction is not great. However, we never failed to find the shortest vector using sign-based pruning. This may indicate that the pruning we employed from the confidence measure is not aggressive enough, and that larger parts of the intervals could be cut away without sacrificing too much accuracy in solving the SVP. Further studies of sign-based pruning is topic for future work.

References

[1] M. Ajtai, "The Shortest Vector Problem in L2 is NP-hard for Randomized Reductions (Extended Abstract)," in *Proceedings of the Thirtieth 9 Annual ACM Symposium on Theory of Computing*, ser. STOC '98. New York, NY, USA: ACM, 1998, pp. 10–19, doi: 10.1145/276698.276705.

[2] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, pp. 515–535, 1982, doi: 10.1007/BF01457454.

[3] C. P. Schnorr, "A hierarchy of polynomial time lattice basis reduction algorithms," *Theoretical Computer Science*, vol. 53, pp. 201–224, 1987, doi: 10.1016/0304-3975(87)90064-8.

[4] U. Fincke and M. Pohst, "A procedure for determining algebraic integers of given norm," in *Proceedings of the European Computer Algebra Conference on Computer Algebra*, ser. EUROCAL '83. London, UK, UK: Springer-Verlag, 1983, pp. 194–202, doi: 10.1007/3-540-12868-9_103.

[5] R. Kannan, "Improved algorithms for integer programming and related lattice problems," in *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, ser. STOC '83. New York, NY, USA: ACM, 1983, pp. 193–206, doi: 10.1145/800061.808749.

[6] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computation*, pp. 463 – 471, 1985, doi: 10.2307/2007966.

[7] T. Laarhoven and A. Mariano, "*Progressive lattice sieving," in Post-Quantum Cryptography*, T. Lange and R. Steinwandt, Eds. Springer International Publishing, 2018, pp. 292–311, doi: 10.1007/978-3-319-79063-3_14.

[8] M. Schneider, "Sieving for shortest vectors in ideal lattices," in *Progress in Cryptology – AFRICACRYPT 2013*, A. Youssef, A. Nitaj, and A. E. Hassanien, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 375–391, doi: 10.1007/978-3-642-38553-7_22.

[9] Y. Aono and P. Q. Nguyen, "Random sampling revisited: Lattice enumeration with discrete pruning," in *Advances in Cryptology – EUROCRYPT 2017*, J.-S. Coron and J. B. Nielsen, Eds. Springer International Publishing, 2017, pp. 65–102, doi: 10.1007/978-3-319-56614-6_3.

[10] Y. Aono, P. Q. Nguyen, and Y. Shen, "Quantum lattice enumeration and tweaking discrete pruning," in *Advances in Cryptology – ASIACRYPT 2018*, T. Peyrin and S. Galbraith, Eds. Springer International Publishing, 2018, pp. 405–434, doi: 10.1007/978-3-030-03326-2_14.

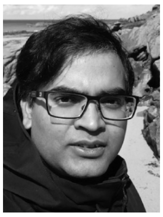
[11] P. Q. Nguyen, "Lattice reduction algorithms: Theory and practice," in *Advances in Cryptology – EUROCRYPT 2011*, K. G. Paterson, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 2–6, doi: 10.1007/978-3-642-20465-4_2.

[12] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Math. Program.*, vol. 66, no. 2, pp. 181–199, Sep. 1994, doi: 10.1007/BF01581144.

[13] C. P. Schnorr and H. H. Hörner, "Attacking the chor-rivest cryptosystem by improved lattice reduction," in *Proceedings of the 14th Annual International Conference on Theory and Application of Cryptographic Techniques*, ser. EUROCRYPT'95. Berlin, Heidelberg: Springer-Verlag, 1995, pp. 1–12, doi: 10.1007/3-540-49264-X_1.

[14] N. Gama, P. Q. Nguyen, and O. Regev, "Lattice enumeration using extreme pruning," in *Advances in Cryptology – EUROCRYPT 2010*, H. Gilbert, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 257–278, doi: 10.1007/978-3-642-13190-5_13.

[15] N. Schneider and N. Gama, "SVP Challenge," https://www.latticechallenge.org/svp-challenge/index.php.



Mithilesh Kumar was born in India. He has done masters in Physics from IIT Kanpur and masters in computer science from CMI Chennai. In 2014 he started his PhD studies in algorithms at the University of Bergen, and he received his PhD degree from this university in 2017. His primary interests are graph theory, algorithms, lattices and quantum computation.

Håvard Raddum was born in Bergen, Norway and received his master degree from the Department of mathematics at the University of Bergen in 1999. In 2005 he received a PhD in cryptography from the University of Bergen. He has done research on the cryptanalysis of ciphers, and is interested in algebraic aspects of cryptographic primitives. Håvard currently leads the cryptography research group at Simula UiB.

Srimathi Varadharajan is a PhD student at Simula UiB, working on the mathematics of cryptography. She got a master degree in mathematics from Royal Holloway University of London in 2015.

The search of square m-sequences with maximum period via GPU and CPU

Paweł Augustynowicz[1] and Krzysztof Kanciak[2]

***Abstract*—This paper deals with the efficient parallel search of square *m*-sequences on both modern CPUs and GPUs. The key idea is based on applying particular vector processor instructions with a view to maximizing the advantage of *Single Instruction Multiple Data* (SIMD) and *Single Instruction Multiple Threads* (SIMT) execution patterns. The developed implementation was adjusted to testing for the maximum-period of *m*-sequences of some particular forms. Furthermore, the early abort sieving strategy based on the application of SAT-solvers were presented. With this solution, it is possible to search *m*-sequences up to degree 32 exhaustively.**

I. INTRODUCTION

Feedback Shift Registers (FSR) are used to generate cryptographically applicable binary sequences. They have many proponents due to their simplicity, both software and hardware effectiveness and well-known properties. In particular, stream ciphers designers use them to construct invertible mappings with internal state. The strongly desirable property of stream ciphers is their long period. Therefore, the FSR used in them should also have this feature. Informally, the period of mapping is the length of the most extended cycle in its state transition graph.

In recent years, many cryptographic algorithms such as stream ciphers (for example GRAIN which is NIST standard [9], Trivium [3] or Achterbahn [2]), lightweight block ciphers and sponge-based generators [4, 10] have used NLFSR for providing both security and efficiency. In most cases, NLFSRs have much greater linear-complexity than LFSRs of the same period, which is directly connected with the security of cryptographic algorithms [12].

Computationally efficient methods for construction of cryptographically strong NLFSRs remains unknown. The most critical NLFSR related problem is finding a systematic procedure for constructing NLFSRs with a long confirmed period. Available algorithms either consider some individual cases or apply to low order NLFSRs only [7, 14, 16]. Nikolay Poluyanenko developed the most efficient method. However, it was not sufficient to obtain applicable NLFSR of degree 30 or higher [13]. Moreover, it requires the usage of special-purpose Field-Programmable Gate Arrays (FPGA) hardware, which is not commonly available.

If we look at the above-mentioned subject from another point of view, NLFSRs are also known as de Bruijn sequences. In a de Bruijn series of order n, all 2^n different binary n-tuples appear precisely once. A modified de Bruijn sequence is obtained from a proper de Bruijn sequence by removing tuple containing zero elements only.

Another essential sequence type, which statistical and structural properties were examined, are so-called m-sequences. Boolean functions that generate the m-sequence can by constructed by introducing nonlinear disturbances into linear functions[11]. Unfortunately, complexity of this approach is extremely high for orders greater than 8. As a result in this paper we address the problem of efficient searching for m-sequences with a guaranteed full period by exhaustively search for the NLFSR with the following form of feedback function:

$$f(x_0, x_1, \ldots, x_{n-1}) = g(x_0, x_1, \ldots, x_{n-1}) + x_i + x_i \cdot x_j$$

for which $i \neq j$, $1 \leq i, j \leq n-1$ and $g(x_0, x_1, \ldots, x_{n-1})$ is defined by a primitive polynomial over $\mathbf{F}_2$. Owing to the large number of candidate feedback functions, the search was conducted on GPUs and special strategy of early abort via SAT solvers' detection of short cycles were applied.

The aforementioned computational experiment allows obtaining an extensive, complete list of n-bit NLFSR ($n < 31$) with a maximum period for the considered form of feedback functions. The previous research in the investigated area has resulted in maximum period NLFSR up to degree 27 [6] on Central Processing Units (CPU) and up to degree 29 on FPGA [13]. We have enumerated all m-sequences up to degree 31. Obtained results suggest the dependency between the Hamming weight of feedback functions and the period of NLFSR generated by that function was observed (see Table VII).

II. BASIC NOTATIONS AND DEFINITIONS

Definition 1: Binary Feedback Shift Register of order n is a mapping $\mathbf{F}_2^n \rightarrow \mathbf{F}_2^n$ of the form:

$$(x_0, x_1, ..., x_{n-1}) \rightarrow (x_1, x_2, ..., x_{n-1}, f(x_0, x_1, ..., x_{n-1})),$$

where:

- f is a boolean function of n variables;
- x_{n-1} is an output bit.

Depending on the type of feedback function two main types of shift registers are concerned:

[1,2] Military University of Technology Institute of Cybernetics, gen. Sylwestra Kaliskiego 2, 00-908, Warsaw, Poland.

[1] E-mail: pawel.augustynowicz@wat.edu.pl

[2] E-mail: krzysztof.kanciak@wat.edu.pl

DOI: 10.36244/ICJ.2019.4.3

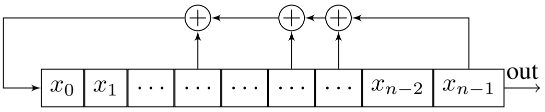


Fig. 1: A structure of Feedback Shift Register.

- linear if the feedback function is linear;
- nonlinear if the feedback functions has degree equal two or higher.

The period of an FSR is the length of the longest cyclic output sequence it generates.

Definition 2: A de Bruijn sequence of order n is a cyclic sequence of length 2^n of elements of $\mathbf{F}_2$ in which all different n-tuples appear exactly once.

Definition 3: A modified de Bruijn sequence of order n is a sequence of length $2^n - 1$ obtained from a de Bruijn sequence of order n by removing one zero from the tuple of n consecutive zeros.

From the cryptographic or random number generation perspective, it is strongly desirable that NLFSR of order n should generate a de Bruijn sequence of order n. Furthermore, due to practical reasons, the following conditions should be fulfilled:

- the number of feedback function's linear and nonlinear terms should remain as small as possible;
- the algebraic degree of feedback function should be the lowest possible;
- the feedback function should be easy to generate.

So-called square m-sequences achieve all the considered restrictions.

Definition 4: A square m-sequence is a bit sequence generated by a shift register with a feedback function with the following form:

$$f(x_0, x_1, \ldots, x_{n-1}) = \sum_{0 \le i \le j \le n-1} a_{i,j} x_i x_j.$$

Moreover, square m-sequences can be described by a very concise form of the recurrence, which can by formulated as:

$$\forall_{k \ge 0} : s_{n+k} = \sum_{0 \le i \le j \le n-1} a_{i,j} s_i s_j,$$

where s_i denotes the i-th position in the sequence s. It is well-known, that square m-sequences can be algorithmically generated by introducing nonlinear disturbances into linear functions, for example by the following form:

$$f(x_0, x_1, \ldots, x_{n-1}) = g(x_0, x_1, \ldots, x_{n-1}) + x_i + x_i \cdot x_j,$$

where $i \neq j$, $0 \le i \le j \le n-1$, and $g(x_0, x_1, \ldots, x_{n-1})$ is linear functions whose LFSR generates maximum-period sequence. From the theory of de Bruijn sequences [15], it can be concluded that $g(x_0, x_1, \ldots, x_{n-1})$ must be defined by a primitive polynomial in $\mathbf{F}_2[x]$.

III. MASSIVELY PARALLEL ALGORITHM

Due to overhelming number of possible feedback functions (see Table I), we have constructed massively parallel algorithm for the search of square m-sequences. It examines the provided functions' period completeness by enumerating the following states and checking for their uniqueness. In practice, it satisfies to prove that their initial states will be generated after exactly $2^n - 1$ steps.

Degree	26	27	28	29	30
log_2(#Candidates)	29,05	30,45	30,73	32,79	32,85

TABLE I: The number of square m-sequences candidates to be examined by computational experiment.

For accurate description and outline of the feedback function examining algorithm, consider the subsequent data labels:

LFSR – bit representation of the linear component of the feedback function;

NLFSR– bit representation of the nonlinear component of the feedback function;

N – the order of the shift register;

For example for the primitive polynomial of form x_9+x_4+1 and nonlinear part of function with the form $x_3 \cdot x_2$, its bit representation of the linear component has following form in hex: `0x211` whereas the nonlinear one: $0x00c$. Its length N is naturally equal 9.

```
Input : LFSR , NLFSR - ,N - length of register;
i_state = $0x01$;
for $i = 1, \ldots, 2^n - 1$ do
    b_LFSR = (popcount(i_state and LFSR)) mod 2;
    b_NLFSR = (popcount(i_state and NLFSR)) mod 2;
    bit = b_LFSR xor b_NLFSR;
    i_state = (i_state rot_left 1) xor bit;
    if i_state == $0x01$ then
        return $false$;
    end
end
if i_state == $0x01$ then
    return $true$;
else
    return $false$;
end
```

Algorithm 1: The period examination algorithm of NLFSR's feedback function.

For the sake of completeness of the specifications considered in the algorithm 1, it should be completed that **popcount** indicates an operation of returning the number of ones in the given integer and **mod** – an instruction of a division with the remainder. The algorithm 1 considered above can be implemented on all kinds of Graphical Processing Units (GPU) resulting in efficiency advantage over modern CPUs. It is strongly recommended to take advantage of SIMD (Single Instruction Multiple Data), a parallel execution model of modern CPUs, to achieve maximum possible efficiency. With the application of SIMD vector instructions, simple calculations, such as xor, bit shifts or counting ones in a word can be performed even on eight words by one thread. For example the concurrent rotation of 8 32-bits words can be realized by the Intel processor intrinsic `_mm256_mullo_epi32`, whearas **xor** can be computed via `_mm256_xor_si256`.

As far as GPU implementation and its SIMT (Single Instruction Multiple Threads) parallel model are concerned, the most significant factor is to determine the number of ones in the given integer effectively. It can be realized by generating ptx code or exploiting `popcntq` instruction on NVIDIA graphics cards and their CUDA (Compute Unified Device Architecture) development tools. Nevertheless, it is impossible to achieve similar performance on OpenCL (Open Computing Language) implementations. Moreover we have observed that usign one thread per one i_state strategy is obviously optimal. Unluckily performing conditional instructions on GPU is exceptionally inefficient. Consequently, the inner if condition should be omitted in these kind of implementations. As a result the developed algorithm posses no early abort strategy on GPU platform, which would allow to efficient filtration of short-period NLFSRs. However, it can be realized on CPU by the usage of SAT-solvers.

IV. Application of SAT solvers

For polynomials up to degree 31, GPU exhaustive cycle verification method works well since we can examine thousands of registers at once. For polynomials of higher degrees, we found FPGA and CPU implementations much more convenient. Furthermore, before full-cycle FPGA or GPU exhaustive verification, we strongly recommend to check for short cycle existence by solving a Boolean satisfiability problem. It can be realized automatically with the help of some open source tools. Firstly, it is required to translate our for example C programming language implementation to And-Inverter Graphs (AIG), which are intermediate states only of Algebraic Normal Form generation (ANF). This step can be done by usage of ABC: System for Sequential Logic Synthesis and Formal Verification and SAW The Software Analysis Workbench. From ANF, there is the well-known path to Conjunctive Normal Form (CNF), which is finally inputted to SAT-solver (Cadical and Lingeling work well and much better than other more popular SAT-solvers in this case [1]). We do not know NLFSR cycles structure, but the majority of polynomials of degree higher than 31 can be quickly eliminated by SAT searching of cycles shorter than 16 states. FPGA or CPU based full cycle verification is being performed in case of UNSAT (no model found) result of prior SAT short cycle check. SAT-based pre-phase works entirely on the CPU, which gives us tremendous resources utilization rate of the entire computing system. The proposed approach is inspired by the work of Elena Dubrova and Maxim Teslenko [8], [5]. It is worth mentioning that the first application of SAT solvers to NLFSR was motivated by the search of short cycles in stream ciphers [8].

V. Example application of SAT-solver filtering results

Short cycle existence of polynomial can be checked during filtration phase in seconds. For instance polynomial $x_0+x_3+x_{31}+x_1+x_1x_2$ is being checked for consecutive cycle lenghts:

1) cycle lengths equal from 2 to 5 — gives us UNSAT result in miliseconds which means that there is no 2,3,4,5-step cycle
2) cycle lenght equal to 6 — gives us SAT result in less than 3 seconds and bits assignment is equal to 10011010011010011010011010011010

Next polynomial $x_0 + x_3 + x_{31} + x_1 + x_1x_3$ has 2-step cycle and SAT solver returned the assignment 10101010101010101010101010101010 in less than 2 seconds. The exact distribution of cycle lengths remains unknown. Nevertheless, the vast majority of polynomials has cycles shorter than 32-steps and can be easily eliminated in seconds without using extensive computing power. It is estimated that the rejection ratio is approximately about 70% of rejected polynomials for NLFSR degree 31 and checking time less than 60 seconds. Further extension of checking time or the length of the short cycles probably will not result in a performance gain.

VI. Performance Evaluation

A fair comparison of the efficiency of various computing platforms is a very troublesome task due to their completely diverse characteristic. Therefore we simplify the comparison by analyzing only the most important efficiency indicators such as:

- the time of one n-bit full-cycle NLFSR enumeration T_{cycle},
- the number of simultaneously tested NLFSRs,
- the estimated time of enumeration of 1 GB pack of n-bit NLFSRs excluding memory transactions T_{1GB}.

The time of one n-bit full-cycle NLFSR enumeration T_{cycle} is based on the measurement of search time of 10^5 possible NLFSRs for CPU and GPU. The computations were conducted on following computing platforms:

- Intel Core i7 6700K, 4.0 GHz CPU, MSI GeForce GTX 1080 8GB GDDR5 with 32 GB of RAM,
- 2 x Xeon 2699 v3, Tesla K80 with 32 GB of RAM,
- Xeon2699 v4, Tesla P100 with 32 GB of RAM.

As it can be seen from the Tables III and IV GPUs are very efficient for small NLFSRs, but they tend to lose efficiency with the growth of NLFSR order. Consequently, as it can be concluded from Figure 2 and Tables III and IV for degrees higher than 31, it is inefficient to take advantage of GPU computing platform.

i7-6700	Xeon2699v4	Tesla P100	Tesla K80	1080GTX
8	44	3584	2496	2560

TABLE II: The number of parallel computing units for different platforms.

n	Tesla P100	Tesla K80	1080GTX
23	0,0012	0,0021	0,0010
24	0,0029	0,0074	0,0022
25	0,0068	0,0117	0,0051
26	0,0149	0,0242	0,0102
27	0,0286	0,0519	0,0200
28	0,0551	0,1399	0,0410
29	0,1087	0,1870	0,0813
30	0,2244	—	0,1644
31	0,9736	—	—

TABLE III: The time of one n-bit full-cycle NLFSR enumeration T_{cycle} for different GPU.

n	Tesla P100	Tesla K80	1080GTX
23	124	314	139
24	287	1059	304
25	648	1607	688
26	1371	3197	1316
27	2541	6612	2486
28	4719	17194	4910
29	8987	22192	9401
30	17926	—	18387
31	75272	—	—

TABLE IV: The estimated time of enumeration of 1 GB pack of n-bit NLFSRs for GPUs.

n	i7-6700	Xeon2699v3	Xeon2699v4
23	0,0001	0,0001	0,0002
24	0,0002	0,0002	0,0003
25	0,0004	0,0004	0,0007
26	0,0008	0,0008	0,0014
27	0,0016	0,0017	0,0027
28	0,0033	0,0033	0,0055
29	0,0065	0,0066	0,0110
30	0,0130	0,0132	0,0220
31	0,0261	0,0263	0,0439
32	0,0546	0,0527	0,0877

TABLE V: The time of one n-bit full-cycle NLFSR enumeration T_{cycle} for different CPU.

n	i7-6700	2×Xeon2699v3	Xeon2699v4
23	1140	380	363
24	2458	667	695
25	4456	1165	1335
26	8570	2073	2566
27	16263	3884	4943
28	31364	7230	9575
29	60564	13760	18490
30	116873	26409	35707
31	225571	50972	69188
32	437658	98532	133976

TABLE VI: The estimated time of enumeration of 1 GB pack of n-bit NLFSRs for CPUs.

The algorithm 1 scales well on CPUs (see Tables V and VI), which means that we can search for NLFSR's of degrees higher than 32 without any performance drop (this is a currently ongoing process).
GPU devices have a significant advantage in NLFSR's searching up to degree 30. For higher degrees, we can see the performance drop. It is caused by switching the arithmetic of integers from 32-bit word length to 64-bit one and cannot be avoided. One of the possible solutions to the problem mentioned above is applying a bit-slicing method to the algorithm implementation, although it was not the case in our application.

VII. Most significant results

With the usage of algorithm 1 all the NLFSR with feedback function of considered form up to degree 31 were enumerated. Examples of aforementioned feedback functions are given below:

- $n = 30$:
  $x_0 + x_1 + x_3 + x_5 + x_7 + x_8 + x_9 + x_{15} + x_{16} + x_{17} + x_{18} + x_{22} + x_{27} + x_{29} + x_4 \cdot x_{20}$;
  $x^0 + x^1 + x^3 + x^{10} + x^{12} + x^{15} + x^{16} + x^{17} + x^{20} + x^{22} + x^{23} + x^{25} + x^{29} + x^{24} \cdot x^8$;
  $x^0 + x^1 + x^2 + x^3 + x^{10} + x^{11} + x^{16} + x^{18} + x^{19} + x^{23} + x^{24} + x^{25} + x^{27} + x^5 \cdot x^{28}$
- $n = 29$:
  $x_0 + x_6 + x_7 + x_{13} + x_{21} + x_{22} + x_{23} + x_{24} + x_{25} + x_{27} + x_{28} + x_1 \cdot x_{17}$;
  $x_0 + x_4 + x_6 + x_7 + x_9 + x_{10} + x_{11} + x_{12} + x_{16} + x_{17} + x_{21} + x_{25} + x_{26} + x_{17} \cdot x_{21}$;
  $x^0 + x^1 + x^3 + x^9 + x^{10} + x^{11} + x^{14} + x^{16} + x^{18} + x^{19} + x^{20} + x^{21} + x^{22} + x^{26} + x^{28} + x^6 \cdot x^3$;
- $n = 28$:
  $x_0 + x_2 + x_5 + x_{15} + x_{16} + x_{17} + x_{19} + x_{22} + x_{27} + x_8 \cdot x_{18}$;
  $x_0 + x_1 + x_2 + x_3 + x_4 + x_5 + x_8 + x_{10} + x_{11} + x_{16} + x_{17} + x_{19} + x_{20} + x_{21} + x_{22} + x_{24} + x_6 \cdot x_{26}$; $x_0 + x_3 + x_4 + x_5 + x_7 + x_8 + x_{11} + x_{15} + x_{19} + x_{20} + x_{26} + x_{27} + x_{10} \cdot x_{24}$;
- $n = 27$:
  $x_0 + x_2 + x_4 + x_5 + x_6 + x_9 + x_{10} + x_{11} + x_{13} + x_{14} + x_{16} + x_{17} + x_{18} + x_{21} + x_{22} + x_{24} + x_{26} + x_1 \cdot x_8$;
  $x_0 + x_4 + x_6 + x_7 + x_9 + x_{10} + x_{12} + x_{13} + x_{14} + x_{15} + x_{18} + x_{22} + x_{23} + x_1 \cdot x_{25}$;
  $x_0 + x_4 + x_{11} + x_{12} + x_{13} + x_{14} + x_{15} + x_{16} + x_{17} + x_{22} + x_{23} + x_{25} + x_2 \cdot x_{21}$;
- $n = 26$:
  $x_0 + x_3 + x_4 + x_6 + x_7 + x_8 + x_9 + x_{12} + x_{15} + x_{17} + x_{19} + x_{21} + x_{22} + x_{23} + x_{25} + x_1 \cdot x_{15}$;
  $x_0 + x_2 + x_3 + x_5 + x_6 + x_7 + x_9 + x_{10} + x_{11} + x_{12} + x_{21} + x_{24} + x_1 \cdot x_5$;
  $x_0 + x_2 + x_5 + x_7 + x_9 + x_{11} + x_{12} + x_{14} + x_{16} + x_{18} + x_{19} + x_{20} + x_{24} + x_{25} + x_1 \cdot x_{16}$.

Moreover, it has been observed that the feedback functions with around half of non-zero coefficients are more likely to occur than the others. What is more, the most desirable polynomials with the low number of non-zero coefficients are extremely rare in practice or even impossible to find for high degrees. This observation has been illustrated in Table VII. Taking all of the above into consideration, it can be concluded that the construction of ideal cryptographic NLFSR with maximum-period and very concise form is still an open problem.

VIII. Conclusion and Future Work

In this article, the problem of construction of applicable NLFSR was addressed. The exhaustive search of particular

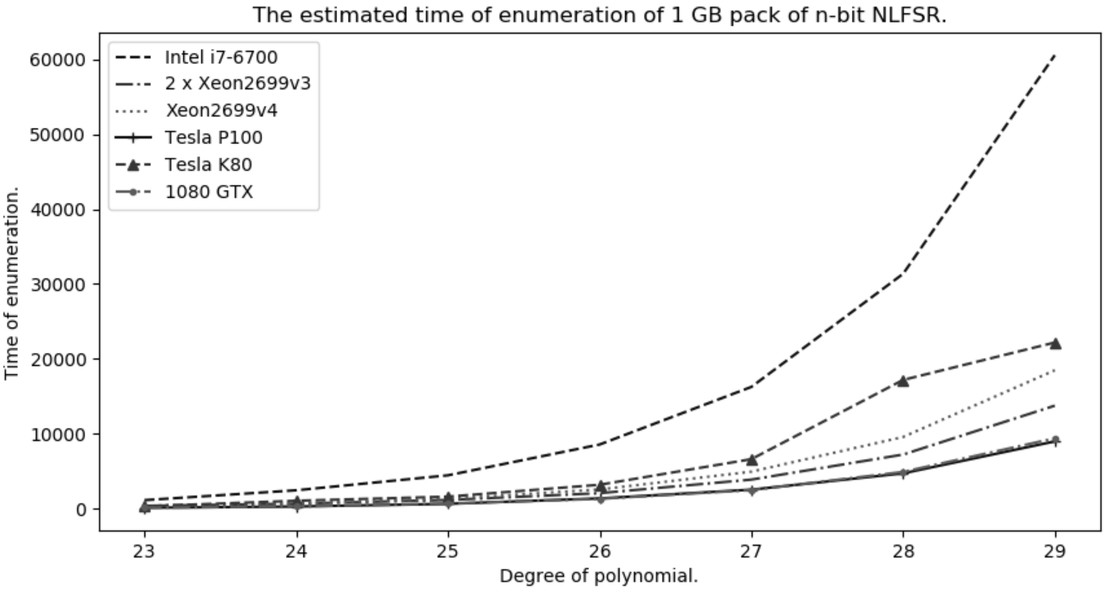


Fig. 2: Comparison of the estimated enumeration time of 1 GB pack of n-bit NLFSRs.

deg	<7	9	11	13	15	17	19	21	23
26	0	24	26	32	48	22	2	0	0
27	0	6	28	58	56	36	4	0	0
28	0	4	10	26	42	32	12	0	2
29	0	4	24	49	72	32	24	8	0
30	0	0	17	37	32	27	13	5	2

TABLE VII: The number of feedback functions with certain number of non-zero coefficients.

form of NLFSRs was conducted, and the results were discussed. Following conclusions can be drawn:

1) the search of NLFSR can be realized both on modern CPU and GPU by adjusting the enumeration algorithm to SIMD and SIMT parallel execution models;
2) square m-sequences have certain cryptographic and practical properties, that are desirable, especially very concise form and lowest possible algebraic degree;
3) the number of square m-sequences with lesser number of terms decreases with the degree of the NLFSR.
4) the GPU implementation should be altered for the 31-degree NLFSR due to the efficiency decrease. One possible solution is to apply the bit-slicing methodology in order to avoid a costly switch to 64-bit arithmetic.
5) SAT solvers sieving can contribute to the fast rejection of short period NLFSR and in consequence, reduce the reasonable time of computational experiments.

It is planned to continue the search of square m-sequences up to degree 32 on FPGA platform and GPU after modification of algorithm 1 via bit-slicing methodology. Moreover, it is necessary to improve the sieving ratio of short period NLFSR due to numerous possible m-sequences of degrees 31 and 32 (approximately $2^{34,9}$ and 2^{35} respectively).

REFERENCES

[1] Biere A. “CaDiCaL, Lingeling, Plingeling, Treengeling, YalSAT Entering the SAT Competition 2017”. In: *Proc. of SAT Competition 2017 – Solver and Benchmark Descriptions*. Ed. by Tomáš Balyo, Marijn Heule, and Matti Järvisalo. Vol. B-2017-1. Department of Computer Science Series of Publications B. University of Helsinki, 2017, pp. 14–15.

[2] Gammel B. M., Gottfert R., and Kniffler O. *The Achterbahn Stream Cipher, Project*. 2005.

[3] De Cannière C. and Preneel B. “TRIVIUM Specifications”. In: eSTREAM, ECRYPT Stream Cipher Project, 2006.

[4] De Cannière C., Dunkelman O., and Miroslav Knežević. “KATAN and KTANTAN — A Family of Small and Efficient Hardware-Oriented Block Ciphers”. In: *Cryptographic Hardware and Embedded Systems - CHES 2009: 11th International Workshop Lausanne, Switzerland, September 6-9, 2009 Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 272–288. DOI: 10.1007/978-3-642-04138-9_20.

[5] Elena Dubrova and Maxim Teslenko. “An efficient SATbased algorithm for finding short cycles in cryptographic algorithms”. In: Apr. 2018, pp. 65–72. DOI: 10.1109/HST.2018.8383892.

[6] Dubrova E. *A List of Maximum Period NLFSRs*. Cryptology ePrint Archive, Report 2012/166. http://eprint.iacr.org/2012/166. 2012.

[7] Dubrova E. *A Scalable Method for Constructing Galois NLFSRs with Period 2^{nd} – 1 using Cross-Join Pairs*. Cryptology ePrint Archive, Report 2011/632. http://eprint.iacr.org/2011/632. 2011.

[8] Dubrova E. and Teslenko M. *A SAT-Based Algorithm for Finding Short Cycles in Shift Register Based Stream Ciphers*. Cryptology ePrint Archive, Report 2016/1068. https://eprint.iacr.org/2016/1068. 2018.

[9] Zhang H. and Wang X. *Cryptanalysis of Stream Cipher Grain Family*. Cryptology ePrint Archive, Report 2009/109. http://eprint.iacr.org/2009/109. 2009.

[10] Aumasson J. P. et al. “Quark: A Lightweight Hash”. In: *Journal of Cryptology*. Vol. 26. 2. Apr. 2013, pp. 313–339. DOI: 10.1007/s00145-012-9125-6.

[11] Mykkeltveit J. and Szmidt J. “On cross joining de Bruijn sequences”. In: *Contemporary Mathematics, 2015*, vol.63. 2015, pp. 335–346.

[12] Courtois N. T. and W. Meier. “Algebraic Attacks on Stream Ciphers with Linear Feedback”. In: *Advances in Cryptology — EUROCRYPT 2003: International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4–8, 2003 Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 345–359. DOI: 10.1007/3-540-39200-9_21.

[13] Poluyanenko N. *Development of the search method for Non-Linear Shift Registers using hardware, implemented on Field Programmable Gate Arrays*. Jan. 2017. DOI: 10.21303/2461-4262.2017.00271.

[14] Dabrowski P. et al. *Searching for Nonlinear Feedback Shift Registers with Parallel Computing*. Cryptology ePrint Archive, Report 2013/542. http://eprint.iacr.org/2013/542. 2013.

[15] Golomb S. *Shift Register Sequences*. Portions coauthored with Lloyd R. Welch, Richard M. Goldstein, and Alfred W. Hales. 1967, pp. xiv + 224.

[16] Rachwalik T. et al. *Generation of Nonlinear Feedback Shift Registers with special-purpose hardware*. Cryptology ePrint Archive, Report 2012/314. http://eprint.iacr.org/2012/314. 2012.

Paweł Augustynowicz is a PhD student at Military University of Technology in Warsaw, Poland. His research interests are related to efficient computation, parallel computing and cryptography. Nowadays he works mostly with construction and search methods of irreducible polynomials over finite fields of prime characteristic.

Krzysztof Kanciak is a Asistant Professor at Military University of Technology in Warsaw, Poland. His research interests are related to modern cryptography, cryptographic protocols and formal verification methods.

A New Type of Signature Scheme Derived from a MRHS Representation of a Symmetric Cipher

Pavol Zajac, and Peter Špaček

Abstract— **We propose a new concept of (post-quantum) digital signature algorithm derived from a symmetric cipher. Key derivation is based on a system of Multiple-Right-Hand-Sides equations. The source of the equations is the encryption algorithm. Our trapdoor is based on the difficulty of creating a valid transcript of the encryption algorithm for a given plaintext (derived from the signed message): the signer can use the encryption algorithm, because he knows the secret key, and the verifier can only check that the solution of the equation system is correct. To further facilitate the verification, we use techniques from coding theory. Security of the system is based on the difficulty of solving MRHS equations, or equivalently on the difficulty of the decoding problem (both are NP hard).**

Index Terms—**Signature scheme, Substitution-Permutation Network, MRHS equation system, post-quantum.**

I. Introduction

We propose a new concept of (post-quantum) digital signature algorithm derived from a symmetric cipher. There are already some signature algorithms that use symmetric primitives as their basis: hash-based signatures, that use one-way property of the underlying hash function (e.g.,SPHINGS+ [1]), and generic schemes based on non-interactive proofs and multiparty computation (e.g., Picnic [2]).

Main innovation of our design is that it does not use underlying cipher as a black-box, but instead as a white-box. This might seem similar to white-box cryptography [3], but our goal is different. While white-box cryptography models the user as a potential attacker, we use white-box version of the cipher to provide a secret algorithm for signatures for a legitimate owner of a secret key. The recipient that verifies the signature does not have access to the white-box, but is instead provided a public key that is created from the cipher representation.

Our design is mostly related to multivariate signatures [4]: Public key is essentially a system of equations, that only the signer can solve (with the help of the secret key). Unlike multivariate case, we use a different representation of equation systems, so called Multiple-Right-Hand-Sides equations (MRHS, [5]). The source of our equations is the encryption algorithm. Our trapdoor is based on the difficulty of creating a valid transcript of the encryption algorithm for a given plaintext (derived from the signed message): the signer can use the encryption algorithm, because he knows the secret key, and the verifier can only check that the solution of the equation system is correct. To further facilitate the verification, we use techniques from coding theory.

In Section II, we summarize the notation, basic definitions and notions required to understand the proposed scheme. The scheme itself is specified in Section III. We provide a simplified example of some steps of the algorithm in Section IV. In Section V we discuss the correctness of the scheme, as well as its efficiency. Finally, in Section VI we analyse the security of the proposed scheme. The security of the system is based on the difficulty of solving MRHS equations, or equivalently on the difficulty of special decoding problem. Both of the problems are NP-hard (in generic version), and should be difficult to solve with quantum computer as well.

Our original goal was to base the signature scheme directly on the symmetric encryption standard AES. Main advantage would be prevalent existence of hardware and software implementations of AES on essentially any platform. As our security analysis shows, it is not clear, whether the scheme can be made secure with the underlying design of AES, which is strongly structured, and this structure might leak the used trapdoor. A more suitable underlying cipher might be LowMC [6] or similar designs. We believe that to construct a fully secure signature scheme, a new type of symmetric cipher should be designed in a way that will facilitate our trapdoor type. We leave these questions open for further research.

II. Definitions

We presume that the reader is acquainted with basic cryptographic definitions such as cryptosystem, (symmetric) cipher, public-key encryption, signature scheme, hash function, etc., as well as the related security notions. We also suppose that the reader is familiar with basic notions of coding theory, such as generator and parity-check matrix.

A. Notation

In this article we will use the following notation:

- All bit operations are represented in algebraic way over field $GF(2)$, shortened to $\mathbb{F}_2$. Note that standard operator $+$ in this field corresponds to a logic operation XOR.
- Sets are denoted by block form, e.g. $\mathbb{R} \subset \mathbb{F}_2^m$.
- Integers are represented by simple notation $i, j, n \in \mathbb{Z}$.
- Vectors of bits are denoted by bold variables such as $\mathbf{u}, \mathbf{x}$. The dimension of the vector depends on the context, and is introduced when defining the vector, e.g., $\mathbf{x} \in \mathbb{F}_2^n$. In our paper, all vectors are always row vectors.

Slovak University of Technology in Bratislava, Slovakia.
e-mail: pavol.zajac@stuba.sk

This research was sponsored in parts by the NATO Science for Peace and Security Programme under grant G5448, and by Slovak Republic under grant VEGA 1/0159/17.

DOI: 10.36244/ICJ.2019.4.4

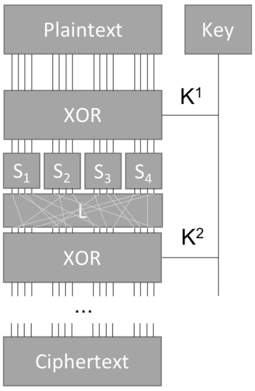


Fig. 1. An example of a subsitution-permutation network.

- Matrix is represented by a bold uppercase letter, with dimensions either depended on the context, or directly defined in the definition of the matrix: $\mathbf{M} \in \mathbb{F}_2^{(n\times m)}$.
- Functions are denoted by uppercase letters, such as F, H, R, S, e.g., $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$.
- Greek notation, such as π, is reserved for permutations of numbers $1, 2, \ldots, n$ (for some n defined in the context).

B. Substitution-permutation network

Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. In cryptographic context, F is called an S-box, if n is relatively small, and F is a highly non-linear function used in cipher design. In our paper, we will denote S-boxes always with S, or S_i if we need to number them.

Substitution-permutation network (SPN) is a type of symmetric cipher, in which the encryption algorithm consists of multiple rounds (number of rounds will be denoted by r). Each round consists of three basic steps:

1) key addition: $\mathbf{y} = \mathbf{x} + \mathbf{k}^i$, where $\mathbf{k}^i$ is a round subkey;
2) non-linear layer of S-boxes: $\mathbf{y}_{i\cdots j} = S(\mathbf{x}_{i\cdots j})$;
3) a linear diffusion layer (in basic case just a permutation of bits): $\mathbf{y} = \mathbf{x}\mathbf{M}$.

In the last round of SPN, linear layer is typically replaced by another key addition. Round subkeys are derived from the main key $\mathbf{k}$ by a specific algorithm call a key schedule, i.e., $(\mathbf{k}^1, \mathbf{k}^2, \ldots, \mathbf{k}^{(r+1)}) = KS(\mathbf{k})$.

Sequence of internal bits that is used during the encryption (of required granularity, e.g., inputs to S-boxes) is denoted as a transcript of the encryption. If we know the encryption key and the cipher input, the whole transcript can be reproduced by following the computation steps of the SPN. For the attacker, the knowledge of the transcript is typically equivalent to the knowledge of the key. In SPN, if the attacker knows the transcript, he can easily compute round keys (using inverse of the key addition operation), and use the round keys instead of the original key (or the attacker can derive the original key if the key schedule allows it).

Advanced Encryption Standard (AES) is a specific instance of SPN-like cipher Rijndael [7], with linear layer defined by two operations ($ShiftRows$, $MixColumns$, with a specific last round). In our concrete instantiation of the signature scheme, we will use version AES-128, which has 128-bit key and block size, 10 rounds. Each round uses 16 bijective S-boxes on groups of 8 bits. Further details of the AES encryption process are not required to understand the paper.

C. MRHS equations

In our system, we create a non-linear Boolean equation system derived from the selected SPN. While such a system can be written in multiple forms (such as ANF for Gröbner basis method, or CNF for SAT solvers), for our purpose a specific form of a MRHS system [5] is preferable.

Definition 1: [8] Let $\mathbb{F}$ be a finite field. Let $\mathbf{M} \in \mathbb{F}^{(n\times m)}$ be an $(n \times m)$ matrix. Let $\mathbb{R}$ be a set of vectors from $\mathbb{F}^m$. MRHS equation is defined by an inclusion:

$$\mathbf{x}\mathbf{M} \in \mathbb{R}.$$

Vector $\mathbf{x} \in \mathbb{F}^n$ is a solution of MRHS equation, if the inclusion holds for this particular value of $\mathbf{x}$.

MRHS equations related to SPN are centered on S-boxes. Let $\mathbf{x}$ be a vector of variables (e.g. selected unknown transcript bits during encryption with SPN). Suppose that each vector of input bits of an S-box S can be expressed as a linear combination of unknown transcript bits: $\mathbf{u} = \mathbf{x}\mathbf{U}$. Similarly let $\mathbf{v} = \mathbf{x}\mathbf{V}$ be the output vector of the same S-box S. We can then write an MRHS equation for S-box S as

$$\mathbf{x}(\mathbf{U}|\mathbf{V}) \in \left\{(\mathbf{u}, S(\mathbf{u})); \mathbf{u} \in \mathbb{F}_2^k\right\}.$$

MRHS (equation) system is a set of MRHS equations, each of which must be satisfied simultaneously for some $\mathbf{x}$ (the particular $\mathbf{x}$ is then a solution of the MRHS system). MRHS system can be written in a similar form to a simple MRHS equation by using a Cartesian product:

$$\mathbf{x}\mathbf{M} \in \mathbb{R}_1 \times \mathbb{R}_2 \times \cdots \times \mathbb{R}_l,$$

where system matrix $\mathbf{M} = (\mathbf{M}_1|\mathbf{M}_2|\cdots|\mathbf{M}_l)$ is composed of matrices of individual MRHS equations.

MRHS system for an SPN is then a set of MRHS equations for each S-box in the system. Unknowns $\mathbf{x}$ in the system must be selected in such a way, that each input and output of the S-box can be expressed as a linear combination of bits of $\mathbf{x}$. Note that we can create a virtual "variable" $\mathbf{1}$ that can express the addition of a constant (0 or 1) when creating the system. After transcribing the system with variable $\mathbf{1}$:

$$(\mathbf{x}, \mathbf{1}) \cdot \begin{pmatrix} \mathbf{M} \\ \mathbf{c} \end{pmatrix} \in \mathbb{R}_1 \times \mathbb{R}_2 \times \cdots \times \mathbb{R}_l,$$

we can rewrite it in extended form as follows:

$$\mathbf{x}\mathbf{M} + \mathbf{c} \in \mathbb{R}_1 \times \mathbb{R}_2 \times \cdots \times \mathbb{R}_l.$$

Constant $\mathbf{c}$ can be transferred to the right-hand side by adding corresponding parts of $\mathbf{c}$ to each vector in $\mathbb{R}_i$.

If MRHS equation has a small number of right-hand sides (members of $\mathbb{R}$), it is easy to solve such a system by repeatedly solving a linear system of equations. Unlike individual equations, an MRHS system has exponentially many right-hand sides (if we try to express the Cartesian product directly). For a general MRHS system, a question of existence of a solution (MRHS problem) is an NP-hard problem [9].

III. ALGORITHM DESCRIPTION

Let us have a symmetric block cipher with encryption algorithm defined by function $Enc : \mathbb{F}_2^n \times \mathbb{K} \rightarrow \mathbb{F}_2^n$, which is itself an SPN network with r rounds. Let $S : \mathbb{F}_2^k \rightarrow \mathbb{F}_2^k$ be an S-box of the SPN network. We suppose that non-linear part of each of the r rounds of SPN consists of n/k applications of the same S-box S (in parallel). The condition that each S-box is the same is important for the security of the scheme. We will denote the total number of S-box applications by $m = rn/k$. Let $KS : \mathbb{K} \rightarrow \mathbb{F}_2^{n(r+1)}$ be a key schedule algorithm.

We define a signature scheme derived from this SPN with the following algorithms: $KeyGen$, $Sign$, and $Verify$.

A. Key generation algorithm

Let $\mathbf{k} \in \mathbb{K}$ be a randomly selected secret key (both of the symmetric cipher, and as a part of a private key of our signature scheme). Furthermore, let π be a randomly selected secret permutation of numbers $1, 2, \ldots, m$, which does not change the order of the initial n/k elements. *Private key* for creating signatures consists of the pair $(\mathbf{k}, \pi)$.

To construct the public key, we must do the following:

1) *Expand the key.* Given $\mathbf{k} \in \mathbb{K}$, we compute the SPN's key schedule: $\hat{\mathbf{k}} = (\mathbf{k}^1, \mathbf{k}^2, \ldots, \mathbf{k}^{(r+1)}) = KS(\mathbf{k})$.
2) *Prepare the MRHS system.* Let $\mathbf{x} \in \mathbb{F}_2^{n(r+1)}$ denote a vector of unknowns corresponding to the inputs of the S-boxes during the SPN evaluation plus one set of S-box outputs in the last round. Inner outputs of the S-boxes can be expressed as linear combinations of variables from $\mathbf{x}$, cipher constants, and constants based on the expanded key $\hat{\mathbf{k}}$. Given linear expressions for the inputs and outputs of the S-boxes, we can construct a MRHS system (as described in section II-C) in the form
$$\mathbf{xM} + \mathbf{c} \in \bigotimes \left\{(\mathbf{u}, S(\mathbf{u}))\,;\mathbf{u} \in \mathbb{F}_2^k\right\}. \qquad (1)$$
Constant $\mathbf{c}$ is derived from the constants of the algorithm and bits of the expanded key $\hat{\mathbf{k}}$. Matrix $\mathbf{M}$ has dimensions $n(r+1) \times 2mk$.
Note that this step can be precomputed up to computation of the final constant $\mathbf{c}$ that depends on $\mathbf{k}$.
3) *Apply masking permutation.* Matrix $\mathbf{M}$ from equation (1) can be written as $\mathbf{M} = (\mathbf{M}_1|\mathbf{M}_2|\cdots\mathbf{M}_m)$. Blocks $\mathbf{M}_i$ of dimension $(n(r+1) \times 2k)$ correspond to linear functions that construct inputs and outputs of S-boxes from variables $\mathbf{x}$. Similarly, split $\mathbf{c}$ into blocks of size $2k$ denoted by $(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_m)$.
Apply the secret permutation π on the order of blocks of $\mathbf{M}$ and c. A permuted form of the system is given as:
$$\begin{aligned}&\mathbf{x} \cdot (\mathbf{M}_{\pi(1)}|\mathbf{M}_{\pi(2)}|\cdots\mathbf{M}_{\pi(m)})\\&\quad + (\mathbf{c}_{\pi(1)}, \mathbf{c}_{\pi(2)}, \ldots, \mathbf{c}_{\pi(m)}) \in\\&\bigotimes \left\{(\mathbf{u}, S(\mathbf{u}))\,;\mathbf{u} \in \mathbb{F}_2^k\right\}.\end{aligned} \qquad (2)$$
Let us denote the joint matrix of system (2) by $\mathbf{M}_\pi$, and similarly let us denote be permuted vector $\mathbf{c}$ by $\mathbf{c}_\pi$.
4) *Parity check matrix.* Compute **systematic**[1] parity check matrix $\mathbf{H} = (\mathbf{I}|\mathbf{Q})$, such that $\mathbf{M}_\pi\mathbf{H}^T = \mathbf{0}$.
5) *Syndrome.* Compute $\mathbf{q} = \mathbf{c}_\pi\mathbf{H}^T$.
6) *Final public key* The public key consist of the pair $(\mathbf{Q}, \mathbf{q})$.

When applicable, public key should be augmented by "domain parameters" describing the used SPN. These consist of a triplet (n, m, k, S), in order: the block size, the number of S-boxes, the S-box size, and the S-box itself.

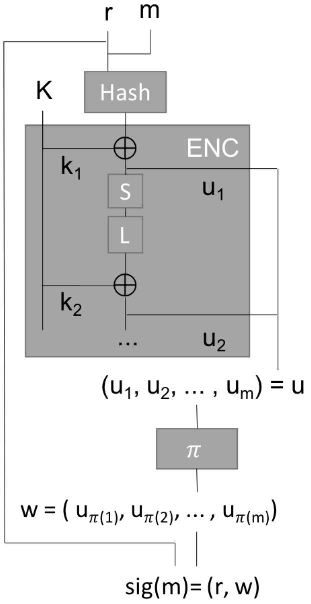


Fig. 2. An overview of the signature algorithm.

B. Signature algorithm

Let $m \in \mathbb{F}_2^*$ be a message we want to sign. Let $H : \mathbb{F}_2^* \rightarrow \mathbb{F}_2^n$ be a cryptographically secure hash function[2]. To sign message m do the following:

1) Generate random (nonce) $\mathbf{r} \in \mathbb{F}_2^n$.
2) Let $\mathbf{p} = H(\mathbf{r}|\mathbf{m}) + \mathbf{k}^1$. Here $\mathbf{k}^1$ is the first subkey derived by KS. Note that value $\mathbf{p}$ is constructed in such a way that $H(\mathbf{r}|\mathbf{m})$ is the vector of inputs to the first layer of S-boxes in the first round of the SPN.
3) Compute $c = Enc(\mathbf{p}, \mathbf{k})$ using SPN algorithm. During encryption, store a sequence of S-box inputs as vector $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m)$.
4) Apply secret permutation π to the order of blocks of $\mathbf{u}$, and compute vector $\mathbf{w} = (\mathbf{u}_{\pi(1)}, \mathbf{u}_{\pi(2)}, \ldots, \mathbf{u}_{\pi(m)})$.
5) Signature of m is pair of vectors $(\mathbf{r}, \mathbf{w})$.

[1] We can compute systematic parity check matrix by linear algebra. Simple algorithm is to use (modified) Gaussian elimination to get $\mathbf{M}_\pi$ to form $(\mathbf{Q}^T|\mathbf{I})$. Note that we cannot change the order of columns during the Gaussian elimination (if no pivots are available, we need to restart the algorithm, or change π to swap blocks as required).

[2] We require that H is one-way and collision resistant.

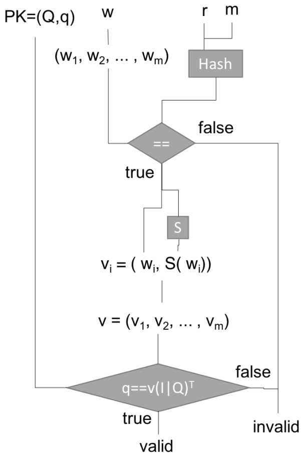


Fig. 3. An overview of the verification algorithm.

C. Verification algorithm

Let $m' \in \mathbb{F}_2^*$ be a message with a supposed signature $(\mathbf{r}, \mathbf{w})$. Let $(\mathbf{Q}, \mathbf{q})$ be the corresponding public key of the signer. To verify whether the signature is valid, perform the following steps.

1) Split $\mathbf{w}$ into m blocks of size k, $\mathbf{w} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_m)$.
2) Let $\mathbf{h} = H(\mathbf{r}|\mathbf{m})$. Verify that $\mathbf{h} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{n/k})$. If not, signature is invalid.
3) Construct m vectors $\mathbf{v}_i$ by using specified S-box S to compute $\mathbf{v}_i = (\mathbf{w}_i, S(\mathbf{w}_i))$.
4) Concatenate $\mathbf{v}_i$ to get vector $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m)$.
5) Verify that $\mathbf{q} = \mathbf{v}(\mathbf{I}|\mathbf{Q})^T$. If not, signature is invalid.

IV. Example

In this section, we provide (simplified) examples for some of the critical steps of the algorithm. In our example, we will work with SPN introduced in Section II-B (see Figure 1). To demonstrate MRHS equation building step, imagine first a simple SPN with 2-bit ”S-boxes” given by permutation 1230. Suppose we denote S-box inputs by x_1, x_2, and outputs by y_1, y_2. MRHS equation with solutions corresponding to valid I/O pairs for S-box can be written as

$$(x_1, x_2, y_1, y_2) \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \begin{Bmatrix} (0,0,0,1), \\ (0,1,1,0), \\ (1,0,1,1), \\ (1,1,0,0) \end{Bmatrix}$$

If we wanted to demonstrate an MRHS equation related to a 4-bit S-box, we would need four input variables, four output variables, 8×8 identity matrix, and a set of sixteen 8-bit vectors on the right-hand side.

Now let us construct an MRHS system for Figure 1. We have selected that our unknowns are S-box inputs. We do not use variables of type $y_1, y_2, \ldots$ as in the previous example. Instead, we express y variables as linear combinations of x variables and subkey bits. We use SPN network as denoted in Figure 1. Its linear layer is represented as a multiplication with matrix $\mathbf{L}$. We can write

$$(\mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_7, \mathbf{x}_8) = (\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4) \cdot \mathbf{L} + (\mathbf{k}_5, \mathbf{k}_6, \mathbf{k}_7, \mathbf{k}_8),$$

which leads to

$$(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4) = (\mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_7, \mathbf{x}_8) \cdot \mathbf{L}^{-1} + (\mathbf{k}_5, \mathbf{k}_6, \mathbf{k}_7, \mathbf{k}_8) \cdot \mathbf{L}^{-1}$$

We can extract equations for separate y_i variables (one for each S-box) by splitting matrix $\mathbf{L}$ into blocks $\mathbf{L}_{i,j}$ of size 4×4 bits ($k \times k$ in general). We get the following system (simplified for the first two S-boxes only):

$$(\mathbf{x}_1, \ldots, \mathbf{x_8}, 1) \cdot \left(\begin{array}{cc|cc|c} \mathbf{I} & 0 & 0 & 0 & \cdots \\ 0 & 0 & \mathbf{I} & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots \\ 0 & \mathbf{L}_{11}^{-1} & 0 & \mathbf{L}_{12}^{-1} & \cdots \\ 0 & \mathbf{L}_{21}^{-1} & 0 & \mathbf{L}_{22}^{-1} & \cdots \\ 0 & \mathbf{L}_{31}^{-1} & 0 & \mathbf{L}_{32}^{-1} & \cdots \\ 0 & \mathbf{L}_{41}^{-1} & 0 & \mathbf{L}_{42}^{-1} & \cdots \\ 0 & \mathbf{c}_1 & 0 & \mathbf{c}_2 & \cdots \end{array}\right) \in \mathbb{R} \times \mathbb{R},$$

where $\mathbb{R}$ is a set of sixteen 8-bit vectors $\mathbb{R} = \{(\mathbf{x}, S(\mathbf{x})), \mathbf{x} \in \mathbb{F}_2^4\}$. Constants are computed from subkeys:

$$(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4) = (\mathbf{k}_5, \mathbf{k}_6, \mathbf{k}_7, \mathbf{k}_8) \cdot \mathbf{L}^{-1},$$

and can be extracted from MRHS equation matrix to get an extended form $\mathbf{x} \cdot \mathbf{M} + \mathbf{c} \in \mathbb{R}$, as described in Section II-C.

The system is expanded for each round in a similar way. However, for the last round, we need to add variables that denote S-box outputs, as there are no other S-box inputs. In four round SPN, we get a system in form

$$(\mathbf{x}_1, \ldots, \mathbf{x}_{20}) \cdot \begin{pmatrix} \mathbf{A}_1 & 0 & 0 & 0 \\ \mathbf{A}_2 & \mathbf{A}_3 & 0 & 0 \\ 0 & \mathbf{A}_4 & \mathbf{A}_5 & 0 \\ 0 & 0 & \mathbf{A}_6 & \mathbf{A}_7 \\ 0 & 0 & 0 & \mathbf{A}_8 \end{pmatrix} + \mathbf{c} \in \mathbb{R} \times \cdots \times \mathbb{R},$$

where each matrix $\mathbf{A}_i$ is 16×16 bit matrix, each $\mathbf{x}_i$ is an unknown 4-bit vector ($\mathbf{x}_1$ to $\mathbf{x}_{16}$ are S-box inputs, $\mathbf{x}_{17}$ to $\mathbf{x}_{20}$ are last-round S-box outputs). Constant c is an 80-bit vector (computed from subkeys), and there are 16 (identical) right-hand side sets $\mathbb{R}$ in the Cartesian product.

The system matrix and the constant $\mathbf{c}$ can be rewritten into 16 blocks of 8 bits, corresponding to each S-box used during the encryption. I.e., we understand the system to be

$$\mathbf{x} \cdot (\mathbf{M}_1 | \cdots | \mathbf{M}_{16}) + (\mathbf{c}_1, \ldots, \mathbf{c}_{16}) \in \mathbb{R} \times \cdots \times \mathbb{R}.$$

Select a secret 16-element permutation with first 4 elements fixed, e.g. $\pi = (1, 2, 3, 4, 12, \cdots, 7)$. To get the public key, we construct

$$\mathbf{M}_\pi = (\mathbf{M}_1 | \cdots | \mathbf{M}_4 | \mathbf{M}_{12} | \cdots | \mathbf{M}_7),$$

and compute the corresponding systematic parity check matrix $\mathbf{H}$. Size of the matrix $\mathbf{M}_\pi$ is 80×128 bits, thus $\mathbf{H}$ has dimensions 48×128. Finally, we compute 48-bit public vector

$$\mathbf{q} = (\mathbf{c}_1, \ldots, \mathbf{c}_4, \mathbf{c}_{12}, \ldots, \mathbf{c}_7) \cdot \mathbf{H}^T.$$

Having constructed private and public keys for the signature scheme, following the steps of the signature and verification algorithms is relatively simple. We do not provide concrete bit values, as the number of bits involved is quite large even for the SPN-based demo. To provide a working demonstration of the system, we have prepared a proof of concept implementation based on simple SPN with 16-bit block and 4-bit S-boxes. The source code is available on GitHub at https://github.com/zajacpa/SPNsig. The demonstration code requires SAGE linear algebra system [10] to run.

Note that for a simpler implementation we store inputs and outputs of S-boxes in different order: first we store sequence of all S-box inputs, and then the sequence of S-box outputs. While not exactly corresponding to the theoretical MRHS instance, it is easy to see that the algorithm still works regardless of the bit order. We only need to ensure that the corresponding columns of $\mathbf{M}$ (with correct $\mathbf{H}$) and $\mathbf{c}$ are arranged in the same way as the desired order of the bits in the transcript of the encryption.

V. Signature scheme properties

After the description of the signature scheme, we devote this short section to the discussion of the correctness and efficiency of the proposed scheme.

A. Correctness

Verification algorithm has two steps where it can reject the signature: verification of the hash $\mathbf{h}$, and verification of the syndrome of $\mathbf{v}$.

During the signature, inputs to SPN encryption were chosen in such a way that $\mathbf{h}$ is the input to the first layer of S-boxes in the first round of SPN. Recall that during the key generation we require that the order of these S-boxes remains fixed by permutation π. This means the verifier in the second step just checks whether plaintext $\mathbf{p}$ was constructed from message $\mathbf{m}$ and provided randomness $\mathbf{r}$.

Signature process is based on SPN encryption modelled by MRHS system (1). The pairs $(\mathbf{u}_i, S(\mathbf{u}_i))$ are corresponding right-hand sides providing a valid solution of this system. They are published in modified order, which is obtained as vector $\mathbf{v}$ by the verifier. The following (overdetermined) linear equation system thus has a solution:

$$\mathbf{x}\mathbf{M}_\pi + \mathbf{c}_\pi = \mathbf{v}.$$

If we multiply this by $\mathbf{H}^T = (\mathbf{I}|\mathbf{Q})^T$ we get:

$$\mathbf{x}\mathbf{M}_\pi\mathbf{H}^T + \mathbf{c}_\pi\mathbf{H}^T = \mathbf{v}\mathbf{H}^T,$$

or equivalently

$$\mathbf{q} = \mathbf{v}(\mathbf{I}|\mathbf{Q})^T.$$

B. Efficiency

The performance of the scheme depends on the chosen underlying SPN. However, in comparison with number theoretic or other post-quantum signature schemes the algorithm is extremely simple. In the following we consider the part of the algorithm without message hashing (which is required regardless of the signature algorithm).

The signature algorithm is equivalent to a single symmetric encryption accompanied with a simple permutation of a vector of transcript bits. Similarly, the verification algorithm requires evaluation of S-boxes plus vector-matrix multiplication. The number of nonlinear operations is the same as the number of operations required during a single SPN encryption. While the complexity of the linear part can be higher than required in SPN encryption, this operation is quite simple and can be efficiently implemented.

The size of the public key and signature is related to the chosen SPN. Private key consists of an l-bit symmetric key and a permutation of m numbers. Public key is a pair consisting of a vector $\mathbf{q}$ of size $2mk-(r+1)n$ bits and a matrix $\mathbf{Q}$ of size $(2mk-(r+1)n) \times mk$ bits. Each signature is also a sequence of mk bits.

Let us consider instantiation of the scheme with AES algorithm. In this case, we use $m = 160$ S-boxes of size $k = 8$ bits. This means that each signature is 1280 bits long (160 bytes). This is comparable to a standard RSA signature. Public key contains vector $\mathbf{q}$ of size 1280 bits, and matrix $\mathbf{Q}$ of size 1638400 bits. Together, this is approximately 200kB.

VI. Security

As mentioned in the introduction, the proposed signature scheme is a provided as a new (and hopefully interesting) concept. Security of the scheme is related to the difficulty of solving MRHS equations and the decoding problem, but also to the security of the underlying symmetric encryption scheme. To be able to fully instantiate the (modified version of the) scheme in a provably secure manner requires a deeper research of the proposed scheme, and the related security questions. In this section we focus on security aspects of the scheme and the potential attacks that can compromise the security of the scheme.

Informally, a signature system is secure, if no (poly-time) attacker is able to forge signatures on new messages. To formally prove the security we would have to provide suitable security reduction to some computationally difficult problem. While we believe this might be possible with a suitable instantiation of the SPN and further tweaks of the design, the current scheme as described is not (provably) secure.

The *security level* of the scheme is limited to $n/2$, where n is the block size of the used SPN. This is due to the use of initial hashing step: If the attacker can find a hash collision in the form $h(r|m_1) = h(r|m_2)$, he can use the same signature for two different messages, breaking the non-repudiation property of the signatures. Thus, for our scheme instantiated with AES we can guarantee at most 64-bit security. It is possible to use general Rijndael algorithm with 256-bit block, and with 256-bit hash function (e.g., SHA-2) to extend the presumed security level to 128-bits. In this case $m = 448$ (Rijndael with 256-bit block and 128-bit key, 14 rounds, 32 S-boxes in each), the signature size is 3584 bits, and public key size is 1568 kB. If we take into account Grover's algorithm and extend the key to 256-bits as well, we get $m = 576$ (18 rounds), i.e., 576B signature, and 2592kB public key, respectively.

Regardless of generic attacks on the hash function, there are multiple other ways that attacker can try to attack the signature algorithm:

1) Try to submit false vector $\mathbf{w}'$ that will satisfy the verification algorithm. First part of the verification (hash of message with nonce r should be the prefix of $\mathbf{w}'$) is easily satisfied. Thus the attacker is only concerned with providing false $\mathbf{w}'$ that will satisfy $\mathbf{q} = \mathbf{v}(\mathbf{I}|\mathbf{Q})^T$.
2) Try to derive the private key from the public key.
3) Try to derive the private key, or a new signature from the public key and (chosen) signatures.

To prevent these attacks we rely on the following (presumably) difficult problems:

1) Decoding problem/MRHS problem. These problems are in general NP-hard, and both are related as discussed in [11]. Parameters of the proposed scheme (if instantiated by AES/Rijndael) are comparable (or even stronger) to parameters of proposed code-based cryptosystems with the same expected security level. Main concern for our scheme is whether our specific type of decoding/MRHS instances derived from SPN representation by MRHS system are still sufficiently hard (in relation to random instances of the problem).
2) Given a set of permuted transcripts of the SPN encryption (with known, but not chosen, input to the first layer of S-boxes), can the attacker find the key, or at least provide any new transcript (permuted in the same way)? This question is related to the security of the underlying SPN, and the efficiency of the proposed masking (random permutation of S-box inputs).

We will now discuss these attack vectors in more details.

A. Decoding/MRHS attacks

Verification algorithm consists of verifying the identity $\mathbf{v} \cdot \mathbf{H}^T = \mathbf{q}$. There is an exponentially large number of solutions $\mathbf{v}'$, but only some of them represent a valid signature. Given valid signature $\mathbf{v}$, such that $\mathbf{v}\mathbf{H}^T = \mathbf{q}$, attacker can compute any other valid $\mathbf{v}'$ by adding a codeword $\mathbf{u}$ of the code generated by $\mathbf{M}_\pi$ (which can be computed from public $\mathbf{Q}$). If our signature system was just based on the Niederreiter-like code-based system [12], attacker could just use any prefix $h(\mathbf{m}', \mathbf{r}')$ and find a valid $\mathbf{v}' \cdot \mathbf{H}^T = \mathbf{q}$ by solving a linear system of equations.

However, our scheme has an additional property: only one half of the vector $\mathbf{v}$ is provided in the signature, second half is computed using SPN's non-linear S-boxes. This means that valid signatures form only a (very small and non-linear) subset of the code coset. Each valid signature is a solution of the MRHS system given by equation 2. If the attacker can forge signatures, he can solve this non-linear MRHS system, and vice-versa.

Note that underlying SPN is a block cipher, and thus a permutation for each secret key. This means that there is a unique transcript of the encryption and thus a unique signature for each message hash.

As mentioned, the MRHS problem problem is related to the decoding problem. We can use a specific decoding algorithm to solve MRHS problem [11]. In our system there is also a specific case where the attacker can apply the decoding algorithm: given public key $(\mathbf{q}, \mathbf{Q})$, attacker tries to find the original constant $\mathbf{c}_\pi$ which was used to define code coset defined by syndrome $\mathbf{q}$. If the attacker obtains $\mathbf{c}_\pi$ it might be possible to reconstruct the original key: Attacker knowns a permuted set of subkeys (except the first and the last one), how difficult it is obtain original key?

However, to get $\mathbf{c}$ attacker needs to solve the decoding problem first. Depending on the structure of the cipher he can presume that $\mathbf{c}_\pi$ is a sparse vector (in contrast to signatures $\mathbf{v}$) with a specific structure (i.e., subkeys are only used to compute outputs of S-boxes from the S-box inputs of the next round). It is not clear whether this information can be sufficient to simplify the decoding problem, as the expected weight of $\mathbf{c}$ is still too large (compared to expected minimum code weight).

B. Structural attacks on signatures

We believe that the security of the signature scheme with respect to decoding/MRHS problem is sufficient with respect to generic solving methods. However, there are more critical attacks that exploit the internal structure of the used MRHS problem to circumvent the need to apply generic solution methods, or to assist the generic methods. We will call these attacks structural attacks (similar to terminology used in code-based crypto).

First, let us note that it is necessary that attacker cannot recover permutation π. If the attacker knowns π and a single signature $\mathbf{w}$, he can reconstruct the exact sequence of S-box inputs in SPN. As S-boxes are public, he also knows the corresponding S-box outputs. Let $\mathbf{x}$ denote a vector of outputs of S-boxes in round $i-1$, and let $\mathbf{y}$ denote a vector of inputs of S-boxes in round i. Let $\mathbf{L}$ represent a (known) matrix of the linear diffusion layer of SPN. Then $\mathbf{k}^i = \mathbf{y} + \mathbf{L}(\mathbf{x})$. This means that knowledge of π leads to a knowledge of the sequence of subkeys. This sequence is sufficient to forge signatures (regardless of the key schedule).

Note that specific key schedules can have an adverse effect on the security. In AES (and many other ciphers), main encryption key can be derived from any single subkey (by running the key schedule algorithm in reverse). This means that the attacker only needs to find matching S-box outputs/inputs between arbitrary rounds $i-1$ and i. The easiest case is to find the correct inputs to S-boxes in round 2 (as round 1 is known, due to fixed part of π). This means that attacker needs to correctly place a sequence of $b = n/k$ blocks out of $(r-1)b$ blocks. Complexity of exhaustive search in this case is

$$N = \frac{((r-1)\,b)!}{((r-2)\,b)!}.$$

In case of AES-128, we get $N \approx 2^{113}$, which is much higher than expected security of 64 bits. Similarly for Rijndael-256-256, we get $N \approx 2^{289}$, which is again much higher than security level of 128 bits.

On the other hand, attacker is not limited to an exhaustive search. In the first version of the signature scheme, message $\mathbf{m}$ (of fixed length given by the block size) was used directly as an input of the first layer of S-boxes. Suppose that SPN was AES-128, and attacker used some signature oracle to obtain signatures for two messages that differ in a single byte. From the properties of AES diffusion layer we know that in second round there are exactly 4 non-zero differences and 12 zero differences. Attacker marks those bytes that are unchanged between signatures. Repeating this with different bytes, he can disclose the positions of the round-2 S-box inputs (in 4 byte groups, whose order can be quickly searched to completly disclose the subkey).

To prevent the class of chosen plaintext attacks on SPN, we have added a randomized hashing[3] as the first step of the algorithm. The attacker still knows the inputs to the first layer of S-boxes, but cannot select them in arbitrary way. E.g., if the attacker was to reproduce the previous attack, he requires two hashes that coincide in each byte except one. This is slightly easier than the security level (in AES-128 expected complexity is 2^{60} hashes), but it is then followed by an attack with a difficulty higher than the reduction obtained while looking for collisions.

While the randomized hashing restricts the efficiency of some of the structural attacks, it is not clear whether it is sufficient to prevent all attacks of this type (and how serious is the security level reduction). Efficiency of these types of attacks are related to the concrete structure of the used SPN. It is not sufficient that the cipher under consideration has a sufficient number of rounds to resist some type of cryptanalysis: as we have seen the goal of the attacker can be different, as in some cases he only needs to distinguish which blocks are used in some specific round.

C. Structural attacks on public key

In previous section we have discussed some ways that the attacker can try to recover secret permutation π from some known or chosen signatures. There is still another attack vector related to the public key, namely parity check matrix $\mathbf{H} = (\mathbf{I}|\mathbf{Q})$.

System matrix $\mathbf{M}$ reflects the structure of the used SPN, and is very sparse. Essentially, each block $\mathbf{M}_i$ can be expressed as

$$\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

where $\mathbf{I}$ is an identity matrix corresponding to S-box inputs in round i. Matrix $\mathbf{L}$, along with constant $\mathbf{c}$ reflects the linear layer of SPN, and represent the affine transformation applied to S-box inputs in round $i+1$ to compute S-box outputs in round i.

This structure of the matrix is in essence preserved when π is applied. It is however not clear, whether this structure is always present in matrix $\mathbf{H}$. Note that there are multiple possible matrices $\mathbf{H}$: any base of the dual space is suitable for signature verification. For efficiency reasons, we have chosen to restrict $\mathbf{H}$ to have a systematic form (so we only need to store $\mathbf{Q}$). Another advantage of the systematic form is that during the computation of systematic $\mathbf{H}$ we hope to hide the sparsity and structure of the original $\mathbf{M}$, similarly to LDPC- and MDPC-based cryptosystems [13].

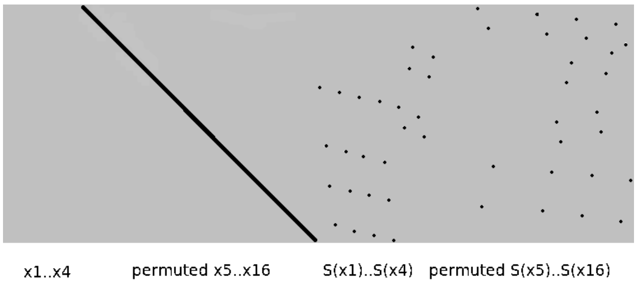


Fig. 4. Graphical depiction of the systematic parity check matrix obtained in one of the experiments with small 16-bit SPN.

Our experiments with the small SPN show that in this case a computation of the systematic parity check matrix is not enough to mask the structure of the system. In Figure 4, we show an example of the matrix $\mathbf{H}$ obtained from the SPN demo. Structure of the original $\mathbf{M}_\pi$ is clearly preserved, and by matching ones on corresponding lines attacker can obtain permutation π.

We have not implemented a full version of AES-based scheme, thus it is not clear whether this problem is also affecting this larger scheme. While AES-based matrix is less sparse due to MixColumns operations, it is still significantly structured. We believe that in this case, systematic form of $\mathbf{H}$ would still not be enough to hide the structure sufficiently.

We might ask, whether the signature scheme be ever made secure in view of this structural attack? Unfortunately, we do not have a definitive answer, but we propose some possible solutions to hide the system structure:

- Change the basis of variables in the system. Instead of defining $\mathbf{x}$ as a vector of S-box inputs, define $\mathbf{x}$ as a vector of arbitrary linearly independent variables $\mathbf{y}$, from which S-box inputs can be computed with linear algebra as $\mathbf{x} = \mathbf{yA}$. In the key generation, we get modified system $\mathbf{yAM}$, and we compute parity check matrix for the code generated by $\mathbf{AM}_\pi$. Note that this is actually the same code as the one generated by $\mathbf{M}_\pi$. Our hope is to obtain different matrix $\mathbf{H}$ that is less sparse and structured as before. It is not clear whether this matrix can really be obtained, and even if it is obtained, whether attacker cannot find different $\mathbf{H}$ that will show him the system structure.
- Use SPN with secret random linear layer. We are inspired by cipher family LowMC [6], but in the signature scheme we do not publish the linear layer, but make it part of the secret key. In this case we lose some efficiency and flexibility in symmetric encryption: linear layer is now asymmetric secret, thus the cipher cannot be reused for symmetric encryption between network participants. This can be remedied by using two versions of the cipher: one for encryption, with fixed (simple) linear layer, and one

[3] We thank one of the anonymous reviewers of the Central European Conference on Cryptology 2019 for the general idea.

(complex/random) linear layer for signatures with secret linear layer. Note that while this eliminates some of the sparsity from the system, the block structure related to cipher rounds remains the same.

- Decompose larger S-boxes to individual AND gates, and call these AND gates new S-boxes. In this case we create a MRHS system with right-hand sides consisting of sets related to AND gates, consisting of triples: $\{000, 010, 100, 111\}$. Signer produces a vector of bits of length $2m$, and verifier just appends bits, that are products of successive pairs, and verifies the coset. System matrix in this case is more complex, as individual I/O bits on individual AND gates need to be expressed as linear combinations of variables. This is also the main disadvantage of the system: system matrix becomes excessively large, increasing the size of the public key.
- Use a different cipher design. E.g., a wide cipher with a large branch number could have a complex enough linear layer in each round. If we study Figure 4, we can see that the main problem is that S-box inputs from one round are only connected to S-box outputs from the previous rounds. If we do not need the encryption algorithm to be reversible (e.g. for a use in stream cipher mode), we can propagate some internal bits to multiple rounds. Note that this must be done carefully to avoid enabling linear or differential attacks on the underlying cipher.
- Would it be possible to hide some information? E.g., we might consider, what would happen if we only include odd numbered rounds in the signature. The MRHS system is still present, and the signature system works correctly. Main difference is that here are now fewer blocks of $\mathbf{M}$, and parity check matrix has smaller dimension (if we remove half of blocks, dimension becomes zero!). This means there would be false solutions, and multiple signatures per message hash. It is not clear whether there is a suitable trade-off when removing selected blocks would actually improve the security.

Implementing some of these solutions can also solve problems with structural attacks based on known signatures. However, we believe that provably secure scheme can only be obtained with a carefully designed block cipher with a goal of providing signatures (through our general scheme) along with symmetric encryption.

VII. Conclusions

In this paper we have presented a new concept of signature scheme based on symmetric cipher design, whose signature and verification algorithm are comparable in complexity to symmetric encryption. Parameters of the system, and its connection to symmetric ciphers, are quite favourable to consider it for future use.

The proposed design should be not be considered a secure signature scheme, as our assumptions are heuristic. The signature scheme relies on hardness of the decoding problem / MRHS problem. Moreover, if the signature system, as presented here, is instantiated by current cipher designs (such as AES), it would presumably not attain the required security due to structural attacks. We have proposed some options on how to further hide the inner structure of the encryption system, but all of these options require further research. We believe that the most promising direction is to design a specific symmetric cipher that will support solid security arguments for the proposed scheme.

Acknowledgment

The authors would like to thank anonymous reviewers of the Central European Conference on Cryptology for the comments on the first proposal of this scheme, that led to identification of some types of attacks on the scheme.

References

[1] A. Hulsing, D. J. Bernstein et al., “Sphincs+,” 2018. [Online]. Available: https://sphincs.org/

[2] G. Zaverucha, M. Chase *et al.*, “The Picnic signature algorithm,” 2018. [Online]. Available: https://microsoft.github.io/Picnic/

[3] S. Chow, P. Eisen, H. Johnson, and P. C. Van Oorschot, “White-box cryptography and an aes implementation,” in *International Workshop on Selected Areas in Cryptography*. Springer, 2002, pp. 250–270. doi: 10.1007/3-540-36492-7_17

[4] J. Ding and B.-Y. Yang, “Multivariate public key cryptography,” in *Postquantum cryptography*. Springer, 2009, pp. 193–241. doi: 10.1007/978-3-540-88702-7_6

[5] H. Raddum and I. Semaev, “Solving multiple right hand sides linear equations,” *Design, Codes and Cryptography*, vol. 49, no. 1, pp. 147–160, 2008. doi: 10.1007/s10623-008-9180-z

[6] M. R. Albrecht, C. Rechberger, T. Schneider, T. Tiessen, and M. Zohner, “Ciphers for mpc and fhe,” in *Advances in Cryptology–EUROCRYPT 2015*. Springer, 2015, pp. 430–454. doi: 10.1007/978-3-662-46800-5_17

[7] J. Daemen and V. Rijmen, *The design of Rijndael: AES-the advanced encryption standard*. Springer Science & Business Media, 2013.

[8] H. Raddum and P. Zajac, “MRHS solver based on linear algebra and exhaustive search,” *Journal of Mathematical Cryptology,* vol. 12, no. 3, pp. 143–157, 2018. doi: 10.1515/jmc-2017-0005

[9] P. Zajac, “MRHS equation systems that can be solved in polynomial time,” *Tatra Mountains Mathematical Publications*, vol. 67, no. 1, pp. 205–219, 2016. doi: 10.1515/tmmp-2016-0040

[10] The Sage Developers, *SageMath, the Sage Mathematics Software System (Version 7.2)*, 2016, http://www.sagemath.org.

[11] P. Zajac, “Connecting the complexity of MQ-and code-based cryptosystems,” *Tatra Mountains Mathematical Publications*, vol. 70, no. 1, pp. 163–177, 2017. doi: 10.1515/tmmp-2017-0025

[12] N. T. Courtois, M. Finiasz, and N. Sendrier, “How to achieve a McEliece-based digital signature scheme,” in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 2001, pp. 157–174. doi: 10.1007/3-540-45682-1_10

[13] M. Baldi, *QC-LDPC code-based cryptography*. Springer Science & Business, 2014.

Pavol Zajac received PhD. in applied mathematics from Slovak University of Technology in Bratislava in 2008. His research interest is mathematical cryptography, see https://scholar.google.com/citations?user=kutD0ZsAAAAJ for a list of publications. Currently he is a full professor at Slovak University of technology working on problems related to post-quantum cryptography.

Peter Špaček is a PhD. student under supervision of Pavol Zajac. His research focus is on post-quantum cryptography and its adaptation into real world applications.

Cross-platform Identity-based Cryptography using WebAssembly

Ádám Vécsi, Attila Bagossy, and Attila Pethő

Abstract—**The explosive spread of the devices connected to the Internet has increased the need for efficient and portable cryptographic routines. Despite this fact, truly platform-independent implementations are still hard to find. In this paper, an Identitybased Cryptography library, called CryptID is introduced. The main goal of this library is to provide an efficient and opensource IBC implementation for the desktop, the mobile, and the IoT platforms. Powered by WebAssembly, which is a specification aiming to securely speed up code execution in various embedding environments, CryptID can be utilized on both the client and the server-side. The second novelty of CrpytID is the use of structured public keys, opening up a wide range of domain-specific use cases via arbitrary metadata embedded into the public key. Embedded metadata can include, for example, a geolocation value when working with geolocation-based Identity-based Cryptography, or a timestamp, enabling simple and efficient generation of singleuse keypairs. Thanks to these characteristics, we think, that CryptID could serve as a real alternative to the current Identitybased Cryptography implementations.**

Index Terms— **Pairing-based Cryptography, Identity-based Cryptography, WebAssembly**

I. Introduction

Identity-based cryptography (IBC) is an important branch of public-key cryptography. Although its foundations were established in 1985 by Shamir [1], who managed to build an identity-based signature (IBS) scheme, identity-based encryption (IBE) remained an open problem until Boneh and Franklin [2] created their pairing-based scheme in 2001, which was fast enough for practical use.

IBC's uniqueness lies in the fact that its public key is a string clearly identifying an individual or organization in a certain domain. Such a string can be an email address or a username. The core purpose behind the IBC was to simplify the certificate management and eliminate the need for certification authorities. In a standard scenario, when employing the public key infrastructure (PKI), the key is bound to its user's identity with a public key certificate, however with IBC the user's identity is the public key itself, thus there is no need for a certificate. Despite this advantage, IBC still requires trusted third-party servers as private key generation and distribution can only be done by a so-called private key generator (PKG).

One can find several IBC implementations on the Internet [3], [4], [5], [6]. However, most of these libraries are focused on a single platform as their target. Unfortunately, applications developed for one specific platform can not be directly adapted for different use. In our opinion, this is a disadvantage of these libraries, because nowadays there is an increasing number of mobile devices connecting to the Internet making use of apps or web-based services. Our motivation was to create a cross-platform, portable IBC solution targeting a large pool of diverse devices that are capable to maintain an internet connection.

One popular technology for development with such goals is JavaScript. One early library of IBC is WebIBC [3], which was developed in 2008 using JavaScript. The authors of the paper concluded that the web browsers and the JavaScript environment were not powerful enough, to implement a standard IBC library, which is based on pairing, because it is ”too complex and overkill”. Instead, they built a combined scheme, which requires much less computation-power and yet, the performance on a desktop was barely satisfying (1.5-2.5 seconds on average for encryption, using a 192-bit integer as the key).

Of course, since 2008, the performance of JavaScript engines significantly increased. An article written by a developer of the V8 JavaScript engine [7] points out that the performance of V8 quadrupled over the last ten years, which may inspire us to give a chance to implement a standard IBC library with the listed goals, using JavaScript. However, over the years a new technology, called WebAssembly came into the picture, which seems even more promising.

Developed by the W3C WebAssembly Community Group since 2015, WebAssembly is a virtual instruction set architecture, aiming to provide a basis for fast computations on the web, while also giving a solution which is embeddable into any environment [8]. Albeit being a quite young technology, already 86% of the internet users have a compatible browser enjoying its benefits [9]. Therefore, we can consider this technology as a promising choice for the development of a cross-platform IBC solution.

In this paper, we will introduce our open source solution for a cross-platform IBC implementation using WebAssembly. The source code is available at https://github.com/cryptid-org, while a consumable NPM package can be downloaded from https://www.npmjs.com/package/@cryptid/cryptid-js. Our solution, called CryptID, is on the one hand small enough to be stored on devices with limited storage capacity, while, on the other hand, its performance is acceptable even on devices with limited computational power. Our experiments proved

Manuscript submitted October 30, 2019. This work was partially supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund. Research of Á. Vécsi and A. Pethő was partially supported by the 2018-1.2.1-NKP-2018-00004 *Security Enhancing Technologies for the Internet of Things* project.

The authors are with the Department of Computer Science, University of Debrecen, H-4028 Debrecen, Kassai str. 26. (emails in order: vecsi.adam@inf.unideb.hu; bagossy.attila@inf.unideb.hu; Petho.Attila@inf.unideb.hu)

DOI: 10.36244/ICJ.2019.4.5

that public key solutions based on IBC are similarly efficient, as those which are based on the PKI. However, the former provides additional possibilities thanks to the properties of the public key.

The rest of the paper is organized as follows. Section II contains some discussion about the relevance of IBC and WebAssembly and also describes Pairing-based cryptography and the standard IBC scheme. Our IBC implementation and its novelties are presented in Section III. Afterward, Section IV outlines the performance of our library on multiple platforms. Section V gives a conclusion and contains some future development plans.

II. Preliminaries

A. Pairing

For $q = p^k$ with a prime p denote $\mathbb{F}_q$ the finite field with q elements. Let $E = E(\mathbb{F}_q)$ be an elliptic curve over $\mathbb{F}_q$. For the subgroup $G_1 \subseteq E$ the mapping $e : G_1 \times G_1 \mapsto \mathbb{F}_{q^\ell}$ with $\ell \geq 1$ is called *pairing* if

bilinear: For all $P, Q \in G_1$ and for all $a, b \in \mathbb{Z}_q^*$ we have $e(aP, bQ) = e(P, Q)^{ab}$.

non degenerated: If P is a non-zero element of G_1 then $e(P, P)$ generate $\mathbb{F}_{q^\ell}$.

Pairing is a rich theory and has numerous applications in cryptography, see the book of Cohen et al [10]. Its first celebrated application is due to Menezes, Okamoto and Vanstone [11], who proved that for supersingular elliptic curves the discrete elliptic logarithm problem can be reduced in polynomial time to a discrete logarithm problem. To prove this result they used the efficiently computable Weil pairing. To avoid technical difficulties we do not define the Weil pairing, but refer to the paper of Boneh and Franklin [2]. There you may find not only the exact definition of the Weil pairing, but also its application to the identity based cryptography.

B. Identity-based Cryptography

In a public-key cryptography system, one very important task is key management. Nowadays, it is mostly handled by the PKI, which seems to work well, however, it has some shortcomings. In the white paper published by Micro Focus International plc [12] six important requirements are specified for enterprise key management.

- Deliver encryption keys.
- Authenticate users and deliver decryption keys.
- Jointly manage keys with partners.
- Deliver keys to trusted infrastructure components.
- Recover keys.
- Scale for growth.

The paper also clearly points out the shortcomings of the PKI. In many ways, it is difficult to use, implement and manage. This difficulty mainly comes from the need of maintaining enormous databases, which can be compromised or damaged, leading to severe data breaches or data loss. Additionally, maintaining such databases can get very expensive.

IBC may offer an obvious solution to these problems. IBC is a type of public-key cryptography in which the public key is a string clearly identifying an individual or organization in a certain domain. It is important to mention that not just the identifier can be arbitrary, but also the domain which specifies the scope of the identifier. This domain can be a global or even a local one, with only a few people in it.

The attractiveness of IBC comes from the previously mentioned properties of the public key, making it possible to establish systems without certification authorities and with simpler key management. Thus, IBC satisfies all six requirements in a cost-effective and user-accessible way.

From the point of this paper, two applications of IBC are relevant, encryption (IBE) and digital signature creation (IBS). Figure 1 shows how a standard IBE scheme works. The main participants are as follows: those want to exchange encrypted messages with each other and a third party, which handles the authentication and the private key generation. For authentication purposes, any already deployed resource can be reused, since this aspect is not limited by the scheme itself. The private key generation is performed by a trusted third party called the Private Key Generator (PKG).

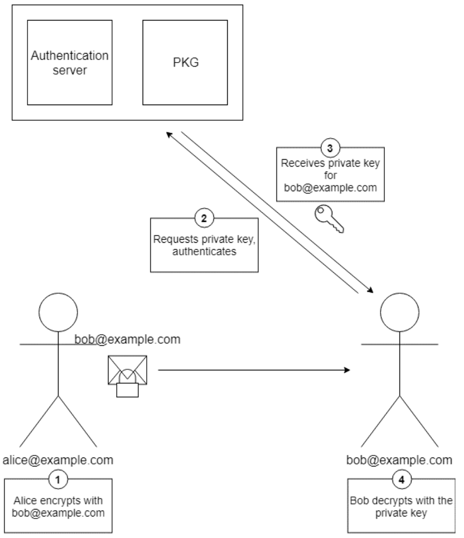

Authentication server
PKG
3
Receives private key for bob@example.com
2
Requests private key, authenticates
bob@example.com
alice@example.com
1
Alice encrypts with bob@example.com
bob@example.com
4
Bob decrypts with the private key

Fig. 1: How IBE works

The IBE scheme is based on four algorithms.

- *Setup*. Responsible for the initialization of the system. It generates the public parameters of the system and the master secret.
- *Extract*. This is the algorithm for calculating the private key from the public parameters, the user's identity, and the master secret.
- *Encrypt*. The algorithm for message encryption. It produces a ciphertext from the public parameters of the system, the public key and the plaintext message.
- *Decrypt*. The algorithm for message decryption. It uses the public parameters of the system, a private key generated by the PKG and an encrypted message.

It should be noted, that *Encrypt* and *Decrypt* are the inverse of each other. This means, if the message space is $\mathcal{M}$, then $\forall M \in \mathcal{M}$: *Decrypt*(*Encrypt*(M, ID), sQ_{ID}) $= M$, where

ID is the user's identity and sQ_{ID} the corresponding secret key.

There are already multiple types of implementations of the IBE. The most popular variants are based on factoring, discrete logarithm or pairing. The first implementation worth mentioning is the Cocks IBE [13], which is based on integer factorization and the quadratic residuosity problem. Unfortunately, this solution produces long ciphertexts and suffers from long runtimes, rendering it inadequate for practical use. The real break-through came with the Boneh-Franklin IBE [2], which is based on pairing. This scheme is appropriate for practical use, but there exist other well-known IBE schemes with better performance, such as the Boneh-Boyen IBE [14], Sakai-Kasahara IBE [15] and TinyIBE [4].

The standard IBS scheme is similar to the IBE scheme by it's structure. The first schemes were based on factoring or RSA [1], [16], [17], but these were not practical. Nowadays the ones based on pairing seem to be the preferred schemes, for instance [18], [19] to name a few.

To assess the security provided by the schemes, Bellare, Namprempre and Neven compared most of the existing IBS schemes with their framework [20].

C. WebAssembly

In this section, we would like to provide a short overview of WebAssembly, which is the key technology behind our work.

During the past decades, the Web has become a ubiquitous application platform, allowing developers to target a huge audience of users in a platform-independent manner. Thus, one can see more and more use cases for the Web platform, even in computation-intensive niches such as games, computer-aided design or audio and video manipulation software. On the other hand, efficient and at the same time, secure code execution remained an issue: technologies such as ActiveX [21], PNaCl [22] or asm.js [23] failed to consistently deliver these properties.

Therefore, a new Web specification, WebAssembly has born, aiming to securely speed up code execution on the Web [24]. The design goals of WebAssembly revolve around two key points, semantics and representation [8].

1) Design Goals: Regarding semantics, WebAssembly aims to execute code with near-native performance in a safe, sandboxed environment, while being hardware-, language- and platform-independent. As WebAssembly can be seen as a compilation target, language-independence means the lack of a privileged programming or object model.

Considering representation, WebAssembly offers a compact, modular binary format, that can be efficiently decoded, validated and compiled. The specification also considers streaming and parallel compilation of modules.

2) Targeting WebAssembly: Several popular programming languages offer WebAssembly as a compilation target, such as C/C++, Rust or C#. As our library is written in C, here we would only cover targeting facilities for that case.

Emscripten is a compiler toolchain built on top of LLVM, that can create WebAssembly modules from C and C++ source files [25] [26]. It should be noted, however, that Emscripten is capable of much more than simply emitting WebAssembly modules. As the browser (which is the main target of Emscripten) is a vastly different environment than the one assumed by most C applications, Emscripten offers the Emscripten Runtime Environment including, for example, a virtual file system, libc and libcxx implementations and tailored input and output handling.

3) Embedding WebAssembly: Despite its name, WebAssembly was designed considering server-side deployments from the ground up. The specification explicitly states openness as one of its design goals, which is achieved by providing a small, well-defined interface between the host environment and the WebAssembly semantics. Since the birth of the specification, several server-side runtimes have appeared, such as Lucet [27] or Wasmer [28].

Regarding Emscripten, we have previously highlighted, that it provides its own runtime environment in the browser. As the WebAssembly specification does not cover interfacing with system resources (such as files), currently each host has to define its own, incompatible runtime environment. In the future, this is going to change, since the WebAssembly System Interface (WASI) specification aims to cover this area [29].

III. CryptID

In most of the cases, it is not obvious how to implement a reliable cryptosystem, even if a mathematically proved secure cryptography protocol is available to build on. During the implementation, it is easy to make mistakes that open vulnerabilities. These vulnerabilities could come from programming negligence (incorrect input validation), or mathematical inattention, ignorance (using unsafe elliptic curves).

Most of the mistakes could be prevented, by using the standards during the implementation. In the case of IBC, multiple standards assist and guide the implementation [30], [31], [32], [33], [34], [35].

This section of the paper is about our solution, called CryptID, which is, in brief, an IBC implementation based on the RFC 5091. Nevertheless, it is not just a usual implementation, its novelty can be approached from two directions.

The novelty in the implementation is that CryptID is based on WebAssembly. Thanks to this property, CryptID is able to work on both the server-side and the client-side, or even completely separated from the web, providing a truly cross-platform and efficient IBC solution.

The novelty in the IBC scheme can be found in the public key. CryptID uses structured public keys, which may contain any kind of metadata with the identity string. This opens up many kinds of domain-specific opportunities. For example, if the current time is part of the metadata, then the keypair is devised for one-time use.

A. Cross-platform operation

With CryptID we wanted to create a library which provides efficient client-side IBC mainly targeting web browsers. Furthermore, we intended to implement a solution that can be used on the server-side, and even on IoT and alike. The motivation behind this was that we did not know about any open source

IBC implementation which is out-of-the-box compatible with these platforms.

Earlier, the only technology which was able to serve these needs, was JavaScript. Unfortunately, JavaScript is far from ideal regarding the performance of computation-intensive tasks, which is just made worse by the fact that every browser has different optimizations, meaning, if something runs fast in one browser, it may be slow in the other. One solution for this problem is asm.js [23], which is a carefully chosen, easy-to-optimize subset of JavaScript. However, asm.js is not a well-established standard.

WebAssembly, on the other hand, is a great choice for projects like CryptID, because it is designed from the ground up as a performance and secure target platform. Moreover, WebAssembly made it possible for us to use GMP [36] as our arbitrary-precision arithmetic library, providing a stable, thoroughly tested foundation to our library.

B. Structured public key

As was written earlier, the essence of IBC is that the public key clearly identifies an individual in a certain domain. Furthermore, Boneh and Franklin in their work [2] mentioned that the public keys are expandable with any kind of metadata. It can be, for example, a year, which assigns a limited period of validity to the public and private keys. In that paper they simply concatenate the metadata to the identifier:

”bob@company.com || current-year”

In our opinion, this is a brilliant idea, with one serious flaw. By using concatenation, flexibility suffers greatly: everything needs to be in a fixed order. Of course, this cannot be changed, as the public key needs to be the same on the bit-level both at encryption and extraction time. This could be a possible point of failure, especially if there are plenty of metadata concatenated.

One solution to this is to add an extra step to the protocol before we use the public key. If we convert the public key to JSON, we can accept arbitrary-ordered JSON documents from the clients. The idea is simple: as the order of keys in a JSON object does not carry any meaning, we are free to reorder them. When given a public key, we always use the same key-sorting algorithm, making it possible to feed bit-accurate public keys to the rest of the protocol. Thus, the clients of CryptID do not need to worry about the way they structure the public key.

Of course, this solution is only applicable to standard IBC protocols. There are schemes, that are using more flexible public keys and do not require complete bit-accuracy. The first work related to this idea was presented by Sahai and Waters [37]. This idea opened an entire branch of protocols, which are focusing on the idea, that decryption should be possible for users who own a public key, that is not bit-accurate to the key used for encryption, but satisfies some kind of rules. This way, it is possible to target a group of users with single encryption. The branch is called Attribute-based Encryption, which has two papers containing the fundamentals and both approaches the problem from different ways. One is Key Policy Attribute-based Encryption [38] and the other is Ciphertext Policy Attribute-based Encryption [39]. Besides, there are some IBC protocols too, with the same essentials [40], [41].

Unfortunately, most of the protocols that are targeting multiple users with a single encryption suffer from the same problem. The more flexibility the encryption provides, the more computation is required by the clients, which results in slower encryption and/or decryption.

C. Library structure

CryptID can be divided into two main components: CryptID.wasm, which is a WebAssembly module, containing the IBC routines. The source code is written in C and is compiled to WebAssembly via Emscripten. The second part of the library is CryptID.js, which is a wrapper on top of the WebAssembly module, written in JavaScript. It provides an easy to use interface for the developers.

The library formed by these parts can, in turn, be divided into five smaller layers, shown in Figure 2.

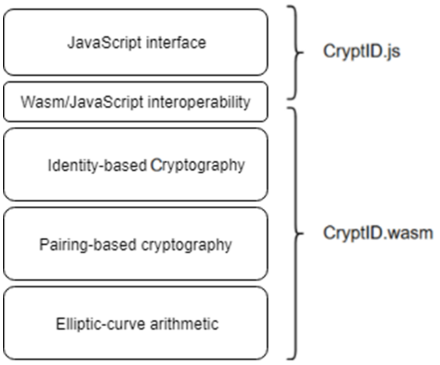

JavaScript interface
Wasm/JavaScript interoperability
Identity-based Cryptography
Pairing-based cryptography
Elliptic-curve arithmetic
CryptID.js
CryptID.wasm

Fig. 2: The structure of CryptID.

Elliptic-curve arithmetics: Most of the popular IBC schemes are based on elliptic-curve cryptography, so the core part of our library is the elliptic-curve arithmetics. The reason behind writing our own implementation is that we could not find a third-party solution that is well-tested and compilable to WebAssembly. As our routines were designed and tested with WebAssembly in mind, we could be sure that they would operate correctly in this environment.

This layer is optimized to Type-1 curves, as recommended in the RFC 5091. The class of curves of Type-1 is defined as the class of all elliptic curves of equation $E(\mathbb{F}_p) : y^2 = x^3 + 1$ for all primes $p \equiv 11 (\text{mod } 12)$. This class forms a subclass of the class of supersingular curves.

To represent big numbers, we are using the GMP [36] arithmetic library. The elliptic-curve points are represented on the affine plane. The layer contains only the necessary methods, namely point doubling, point addition and point scalar multiplication.

Pairing-based cryptography: The majority of IBC protocols are using the pairing operation. As the RFC 5091 recommends, this layer provides a custom implementation of the Tate pairing. The implementation maps two points of $E(\mathbb{F}_p)$ to an element of $\mathbb{F}_{p^2}$. The implementation is based on Lynn's dissertation [42] and Martin's book [43].

Identity-based cryptography: The IBC routines are embedded into this layer. It includes the implementation of the Boneh-Franklin IBE [2] and the Hess IBS [18]. It also contains some miscellaneous helper functions, like an SHA implementation based on the RFC 6234 standard [44], and other hash functions based on SHA.

Wasm/JavaScript interoperability: The previous three layers together form an IBC implementation, which can even be consumed by native applications, without the need for WebAssembly compilation. However, when targeting the Web, these layers are hidden behind a JavaScript interface, as a WebAssembly module.

This abstraction is not absolutely necessary, because the WebAssembly embedding environment allows to call the functions of the module directly. However, CryptID is designed for those embedding environments, which grant the WebAssembly module calls via JavaScript. Such environments include web browsers or Node.js. This way CryptID can be called like any other JavaScript library rendering WebAssembly a simple implementation detail for clients.

To support this approach, it is crucial to implement an interoperability layer, which is responsible for the following tasks.

- Wrapping the C functions. To make the C functions callable from JavaScript, they need to be wrapped with the *cwrap* function of Emscripten's *Module* object.
- Conversion between datatypes. In our case two conversions were necessary, in both cases back and forth. The first one is the converson of GMP's *mpz_t* big number type to JavaScript strings. The other one is the conversion betwen raw C byte arrays and JavaScript *ArrayBuffer*s.
- Bidirectional dataflow. It is not possible to use complex types like structs as parameters or return values, so the only way to exchange values is to copy between the isolated memory spaces accessible to JavaScript and to WebAssembly.

JavaScript interface: The JavaScript interface is a group of functions and datatypes which are public for outside clients. While previous layers can all be seen as implementation details, this layer is the actual interface that clients may consume.

Besides being a facade to lower layers, this layer has further responsibilities.

- Input validation. Being the public interface of the library, this is the only point where invalid input may enter into the system. Such values include *null* values or objects and strings with incorrect structure. Thanks to the validation performed by this layer, malformed values cannot form the basis of any computation.
- Key conversion. One of CryptID's novelties was the structured public key, which is able to contain any kind of metadata. Currently, structured public keys are handled in the interoperability layer, as JSON values. As there can be multiple JSON representations of the semantically same information, it is an important task to always convert JSON strings with the same content to the same bitstream. Our solution is to first create a new JavaScript object from the JSON string, with keys added in alphabetical order. Afterwards, `JSON.stringify` is called on this object to produce a new JSON string. Since `JSON.stringify` is guaranteed to preserve the original key addition order when producing JSON documents, we will always get the same bitstream from documents with the same keys. Thanks to this solution the lower layers do not need to know anything about the structure of the public key.

IV. Performance

In the next section, the performance of the CryptID library is covered. We ran several benchmarks in multiple different environments while exercising the most performance-critical parts of the codebase. Where appropriate, we also compared the performance of our solution with the native version of other, well-established libraries. The IBS scheme is based on the same algorithms as the IBE protocol, so it's performance evaluation is not included in the paper, but the main results are identical.

First, we briefly outline the benchmark environments, which is followed by a detailed description of the performed experiments and their results.

A. Environments

Proving the platform-independent nature of our library, we aimed to benchmark it on a variety of platforms and WebAssembly embedders. On the desktop, we performed experiments in three different embedders (Mozilla Firefox, Google Chrome, Node.js), and we also included the performance of the native version of the library as a baseline result. The exact hardware and software specifications can be seen in Table I. Regarding the mobile, we executed measurements in a single embedder (Google Chrome). Detailed specifications are available in Table II.

On all platforms, we used the Google Benchmark library [45] for our experiments. An experiment comprises twenty performance tests, where each test contains multiple executions of the same code on the same input. The result of the experiment is calculated as the average of the execution times. Inputs were chosen randomly for the four RFC defined security levels shown in Table III. Here, p is the order of the base finite field, the elliptic curve is defined over, while k stands for the RSA keylength providing comparable security [46].

Parameter	Value
Model	Dell Inspiron 5567 (2017)
CPU	i7-7500U, 2,7 GHz
OS	Ubuntu 16.04.4 LTS
emscripten	1.38.8
gcc	5.4.0 20160609
Node.js	v8.9.1
Firefox Quantum	62.0.3

TABLE I: Desktop hardware and software configuration.

Parameter	Value
Model	Nokia 6.1 TA-1043
CPU	Qualcomm Snapdragon 630, 2.2 GHz
OS	Android 8.1.0 - Kernel 4.4.78-perf+
Chrome for Mobile	68.0.3440.91

TABLE II: Mobile hardware and software configuration.

Security Level	p bitlength	k
LOWEST	512	1024
LOW	1024	2048
MEDIUM	1536	3072
HIGH	3840	7680

TABLE III: Security levels and values for appropriate parameters as stated in RFC 5091.

B. Benchmark Results

1) Elliptic Curve Scalar Multiplication: Considering the elliptic curve arithmetics, the scalar multiplication of elliptic curve points is a key operation to IBE. This operation can be found in a vast number of libraries, from which we chose MIRACL [5] and PARI [47] for our benchmarks because these are well-known and thoroughly tested solutions. In our experiments, we compared the performance of four different configurations: native MIRACL, native PARI, native CryptID and CryptID WebAssembly (Node.js).

In Figure 3, we graphed the benchmark results on a logarithmic scale, where the vertical axis represents the runtime in nanoseconds. It is clear that MIRACL is several magnitudes faster than any other solution. On the other hand, we would like to highlight, that the native version of CryptID is just a little behind PARI, which is promising. Being this close results from the same choice of algorithm (double-and-add) and arithmetic library (GMP). In the case of the WebAssembly version, a somewhat consistent performance penalty can be noticed, compared to the native version. As the specification and the implementations mature, we expect this gap to decrease gradually.

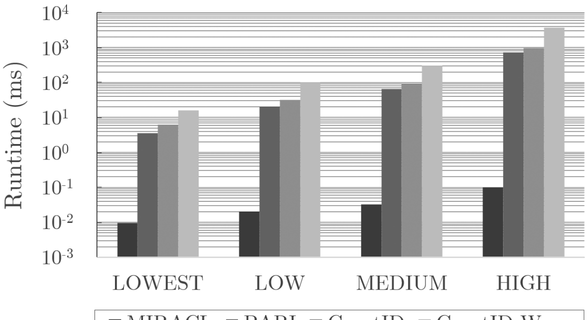


Fig. 3: Performance comparison of elliptic curve scalar multiplication solutions.

2) Encrypt: CryptID was primarily designed for client-side use cases, where encryption and decryption take place on the user's device. Optimizing the performance of these operations is crucial, as we expect them to executed be frequently in a wide variety of browsers and devices. Thus, we first measured the runtime of the encrypt method in four different configurations: desktop Node.js, desktop Firefox browser, desktop Chrome browser, and again, desktop native as a baseline.

The logarithmically graphed results can be seen in Figure 4, where the vertical axis represents the runtime in milliseconds. On the desktop, the same sized performance gap is present between the native and WebAssembly versions, as in the case of elliptic curve scalar multiplication. Executing the WebAssembly code is consistently three to four times slower than the native program. However, we were quite surprised to discover, that our experiments took approximately the same time to finish in Node.js and Firefox, considering the difference between the WebAssembly runtimes of these environments.

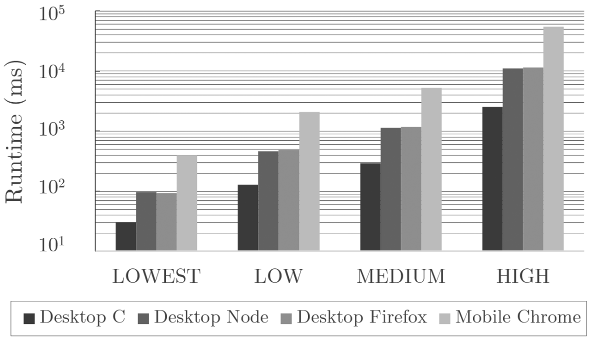


Fig. 4: Performance comparison of the *encrypt* function in across various environments.

Unfortunately, encryption on the mobile is around four to five times slower than on the desktop. The difference results from the gap between desktop and mobile computational power. In spite of that, execution time of the `low` and `medium` security level can still be considered acceptable in practice.

3) Components of Encrypt: After outlining and comparing the performance of the encrypt operation in different environments, we would also like to further break this operation down into smaller components. The pie charts of Figure 5 show the results of profiling an experiment on a `high` input in the desktop Firefox environment.

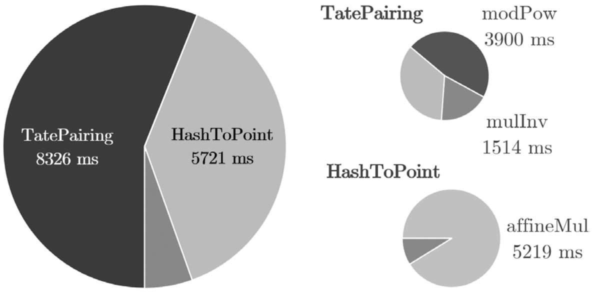


Fig. 5: Profiling results of a single *encrypt* execution on HIGH input in the desktop Firefox environment.

The run time of encrypt is impacted by the performance of the Tate pairing and the *HashToPoint* function. As it can be seen on the bottom right chart, the execution time of the latter is approximately equivalent to that of a single elliptic curve scalar multiplication. Regarding the Tate pairing, modular exponentiation and multiplicative inverse have the largest influence on the performance.

4) Decrypt: We also performed experiments on the decrypt function in the same configurations as in the case of encryption. Based on the previous tests, we already had an approximate expectation regarding the performance of this function.

The actual results are shown in Figure 6, graphed on a logarithmic scale. Just as expected, the native desktop version has the best performance, while desktop Firefox is on par with Node.js. The performance gaps are of the same size as observed in the case of encryption.

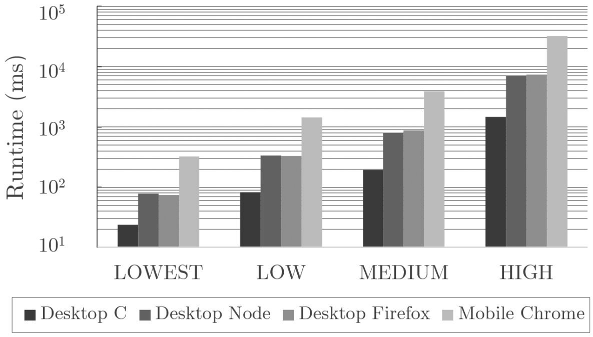


Fig. 6: Performance comparison of the *decrypt* function in across various environments.

Comparing the performance of decryption and encryption, one can notice, that *decrypt* runs for two-thirds of the run time as of encrypt. This is caused by the fact, that decrypt is equal to a single execution of the Tate pairing.

V. Conclusion and future work

We created a novel IBC library, which serves as a real alternative to the current implementations. One of the library's unique characteristics lies in its portability. Thanks to WebAssembly's platform-independent nature, CryptID offers an IBC solution on desktop, mobile, and IoT. The portability is combined with simple integrability, which makes for an appealing application development experience. CryptID also extends the already appealing identity-based public keys with optional metadata, in a structured, easy-to-manage way, giving an opportunity for many domain-specific use cases.

Several applications can be built on the above-mentioned novelties of the library. The client-side execution of the cryptographic functions provides a secure use of IBC, for example as a secure e-mail service. Besides the domain-specific options, structured public keys can also be used for the creation of single-use public keys, which is another important potential application of the library.

Considering future work, there are multiple possibilities to improve the performance of the library. With the implementation of better-performing arithmetic functions, we can optimize CryptID. Our main goal is to improve the elliptic-curve arithmetic layer, with the implementation of the Heuberger-Mazzoli scalar multiplication [48], and with some useful tricks, like precalculations.

Furthermore, another direction is the binary size reduction. Even though our bare library itself is lightweight, our dependency on GMP increases the linked binary size to a few hundred kilobytes. Unfortunately, dropping this dependency would cost us a lot of work, thus we are thinking of different approaches. Such an approach, for example, is called *tree-shaking*, which means the disposal of the unused code, potentially reducing the size of the linked binary even further.

Acknowledgment

We would like to thank the anonymous referees for their thoughtful comments and efforts towards improving our manuscript.

Competing interests

The authors declare that they have no competing interests.

References

[1] A. Shamir, "Identity-based Cryptosystems and Signature Schemes," in *Proceedings of CRYPTO 84 on Advances in Cryptology*. New York, NY, USA: Springer-Verlag New York, Inc., 1985, pp. 47–53. [Online]. Available: http://dl.acm.org/citation.cfm?id=19478.19483

[2] D. Boneh and M. K. Franklin, "Identity-Based Encryption from the Weil Pairing," in *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '01. London, UK, UK: Springer-Verlag, 2001, pp. 213–229. [Online]. Available: http://dl.acm.org/citation.cfm?id=646766.704155

[3] Z. Guan, Z. Cao, X. Zhao, R. Chen, Z. Chen, and X. Nan, "WebIBC: Identity based cryptography for client side security in web applications," in *2008 The 28th International Conference on Distributed Computing Systems*. IEEE, Jun. 2008, DOI: 10.1109/icdcs.2008.24.

[4] P. Szczechowiak and M. Collier, "TinyIBE: Identity-based encryption for heterogeneous sensor networks," in *2009 International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*. IEEE, 2009, DOI: 10.1109/issnip.2009.5416743.

[5] (2019) MIRACL Cryptographic SDK. [Online]. Available: https://github.com/miracl/MIRACL

[6] L. Ducas, V. Lyubashevsky, and T. Prest, "Efficient identity-based encryption over NTRU lattices," in *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2014, pp. 22–41, DOI: 10.1007/978-3-662-45608-8_2.

[7] M. Bynens. (2018) Celebrating 10 years of V8. [Online]. Available: https://v8.dev/blog/10-years

[8] (2018) WebAssembly Specification. WebAssembly Community Group. [Online]. Available: https://webassembly.github.io/spec/core/ download/ WebAssembly.pdf

[9] (2018) Can I use WebAssembly? [Online]. Available: https://caniuse.com/#feat=wasm

[10] H. Cohen, G. Frei, R. Avanzi, C. Doche, T. Lange, K. Nguyen, and F. Vercauteren, Eds., *Handbook of Elliptic and Hyperelliptic Curve Cryptography, Second edition*. Chapman & Hall/CRC, 2012.

[11] A. Menezes, T. Okamoto, and S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," *IEEE Transactions on Information Theory*, vol. 39, pp. 1639 – 1646, 1993, DOI: 10.1109/18.259647.

[12] (2018) The Identity-Based Encryption Advantage. [Online]. Available: https://www.microfocus.com/media/white-paper/the_identity_based_encryption_advantage_wp.pdf

[13] C. Cocks, "An identity based encryption scheme based on quadratic residues," in *In IMA International Conference*. Springer-Verlag, 2001, pp. 360–363, DOI: 10.1007/3-540-45325-3_32.

[14] D. Boneh and X. Boyen, "Efficient selective-ID secure identity-based encryption without random oracles," in *Advances in Cryptology - EUROCRYPT 2004*. Springer Berlin Heidelberg, 2004, pp. 223–238, DOI: 10.1007/978-3-540-24676-3_14.

[15] R. Sakai and M. Kasahara, "Id based cryptosystems with pairing on elliptic curve," 2003. [Online]. Available: http://eprint.iacr.org/2003/054

[16] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," in *Advances in Cryptology — CRYPTO' 86*. Springer Berlin Heidelberg, 2000, pp. 186–194, DOI: 10.1007/3-540-47721-7_12.

[17] T. Okamoto, "Provably secure and practical identification schemes and corresponding signature schemes," in *Advances in Cryptology — CRYPTO' 92*. Springer Berlin Heidelberg, 2001, pp. 31–53, DOI: 10.1007/3-540-48071-4_3.

[18] F. Hess, "Efficient identity based signature schemes based on pairings," in *Selected Areas in Cryptography*. Springer Berlin Heidelberg, 2003, pp. 310–324, DOI: 10.1007/3-540-36492-7_20.

[19] P. S. L. M. Barreto, B. Libert, N. McCullagh, and J.-J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2005, pp. 515–532, DOI: 10.1007/11593447_28.

[20] M. Bellare, C. Namprempre, and G. Neven, "Security proofs for identity-based identification and signature schemes," *Journal of Cryptology*, vol. 22, no. 1, pp. 1–61, Aug. 2008, DOI: 10.1007/s00145-008-9028-8.

[21] (2017) Introduction to ActiveX Controls. [Online]. Available: https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/platform-apis/aa751972%28v%3dvs.85%29

[22] (2017) NaCl and PNaCl. [Online]. Available: https://developer.chrome.com/native-client/nacl-and-pnacl

[23] (2014) asm.js Specification. [Online]. Available: http://asmjs.org/spec/latest/

[24] A. Haas, A. Rossberg, D. L. Schuff, B. L. Titzer, M. Holman, D. Gohman, L. Wagner, A. Zakai, and J. Bastien, "Bringing the web up to speed with webassembly," *SIGPLAN Not.*, vol. 52, no. 6, pp. 185–200, jun 2017, DOI: 10.1145/3140587.3062363.

[25] A. Zakai, "Emscripten: An LLVM-to-JavaScript Compiler," in *Proceedings of the ACM International Conference Companion on Object Oriented Programming Systems Languages and Applications Companion*, ser. OOPSLA '11. New York, NY, USA: ACM, 2011, pp. 301–312, DOI: 10.1145/2048147.2048224.

[26] A. Zakai. (2015) Big Web App? Compile It! [Online]. Available: kripken.github.io/mloc_emscripten_talk/

[27] (2019) Lucet, the Sandboxing WebAssembly Compiler. [Online]. Available: https://github.com/fastly/lucet

[28] (2019) Wasmer – Universal WebAssembly Runtime. [Online]. Available: https://wasmer.io/

[29] L. Clark. (2019) Standardizing WASI: A system interface to run WebAssembly outside the web. [Online]. Available: https://hacks.mozilla.org/2019/03/standardizing-wasi-a-webassembly-system-interface/

[30] X. Boyen and L. Martin, "Identity-based cryptography standard (IBCS) #1: Supersingular curve implementations of the BF and BB1 cryptosystems," *RFC*, vol. 5091, pp. 1–63, 2007, DOI: 10.17487/RFC5091.

[31] G. Appenzeller, L. Martin, and M. Schertler, "Identity-based encryption architecture and supporting data structures," Tech. Rep., Jan. 2009, DOI: 10.17487/rfc5408.

[32] L. Martin and M. Schertler, "Using the boneh-franklin and boneh-boyen identity-based encryption algorithms with the cryptographic message syntax (CMS)," Tech. Rep., Jan. 2009, DOI: 10.17487/rfc5409.

[33] "IEEE standard for identity-based cryptographic techniques using pairings," 2013, DOI: 10.1109/ieeestd.2013.6662370.

[34] "Iso/iec 18033-5:2015: Information technology - security techniques - encryption algorithms - part 5: Identity-based ciphers," 2015. [Online]. Available: https://www.iso.org/standard/59948.html

[35] "Iso/iec 15946-1:2016: Information technology - security techniques - cryptographic techniques based on elliptic curves - part 1: General," 2016. [Online]. Available: https://www.iso.org/standard/65480.html

[36] T. Granlund and the GMP development team. (2016) GNU MP: The GNU Multiple Precision Arithmetic Library. [Online]. Available: http://gmplib.org/

[37] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Lecture *Notes in Computer Science*. Springer Berlin Heidelberg, 2005, pp. 457–473, DOI: 10.1007/11426639_27.

[38] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM conference on Computer and communications security - CCS '06*. ACM Press, 2006, DOI: 10.1145/1180405.1180418.

[39] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *2007 IEEE Symposium on Security and Privacy (SP '07)*. IEEE, May 2007, DOI: 10.1109/sp.2007.11.

[40] M. Abdalla, D. Catalano, A. W. Dent, J. Malone-Lee, G. Neven, and N. P. Smart, "Identity-based encryption gone wild," in *Automata, Languages and Programming*. Springer Berlin Heidelberg, 2006, pp. 300–311, DOI: 10.1007/11787006_26.

[41] O. Blazy, P. Germouty, and D. H. Phan, "Downgradable identity-based encryption and applications," in *Topics in Cryptology – CT-RSA 2019*. Springer International Publishing, 2019, pp. 44–61, DOI: 10.1007/978-3-030-12612-4_3.

[42] B. Lynn, "On the implementation of pairing-based cryptosystems," Ph.D. dissertation, Stanford University, Stanford, CA, USA, 6 2007. [Online]. Available: https://crypto.stanford.edu/pbc/thesis.pdf

[43] L. Martin, *Introduction to Identity-based Encryption*. Boston: Artech House, 2008.

[44] D. Eastlake and T. Hansen, "US secure hash algorithms (SHA and SHA-based HMAC and HKDF)," Tech. Rep., May 2011, DOI: 10.17487/rfc6234.

[45] (2019) Google Benchmark – A microbenchmark support library. [Online]. Available: https://github.com/google/benchmark

[46] (2009) The Case for Elliptic Curve Cryptography. National Security Agency. [Online]. Available: https://web.archive.org/web/20090117023500/http://www.nsa.gov/business/programs/elliptic_curve.shtml

[47] (2018) PARI/GP version 2.11.0. The PARI Group. Univ. Bordeaux. [Online]. Available: http://pari.math.u-bordeaux.fr/

[48] C. Heuberger and M. Mazzoli, "Symmetric digit sets for elliptic curve scalar multiplication without precomputation," *Theoretical Computer Science*, vol. 547, pp. 18–33, aug 2014, DOI: 10.1016/j.tcs.2014.06.010.

Ádám Vécsi graduated as a computer scientist at the Faculty of Informatics, University of Debrecen in 2019. He started his Ph.D in the same year at the Doctoral School of Informatics, University of Debrecen. His current research interests include various aspects of cryptography, mainly Identity-based Cryptography. You can contact him: vecsi.adam@inf.unideb.hu.

Attila Bagossy graduated as a computer scientist at the Faculty of Informatics, University of Debrecen in 2019. He started his Ph.D in the same year at the Doctoral School of Informatics, University of Debrecen. His current research interests include unconventional models of computation, cryptography and WebAssembly. You can contact him: bagossy.attila@inf.unideb.hu.

Attila Pethő is a professor and was the head of the Department of Computer Science, Faculty of Informatics, University of Debrecen, Hungary. He got PhD degree in mathematics from the Lajos Kossuth University. He is an ordinary member of the Hungarian Academy of Sciences. His research interest are number theory and cryptography. You can contact him: petho.attila@inf.unideb.hu.

Performance Analysis of Communication System with Fluctuating Beckmann Fading

Zakir Hussain, Asim ur Rehman Khan, Haider Mehdi and Aamir Ali

Abstract—**In this paper, performance of device-to-device (D2D) communication system over Fluctuating Beckmann (FB) fading channels is analyzed. FB fading model is a novel generalized fading model that unifies various fading models such as Rayleigh, Nakagami, one-sided Gaussian, Rician, Rician shadowed, *κ-μ*, *κ-μ* shadowed, *η-μ* and Beckmann. The considered D2D system is assumed to be affected by various FB faded co-channel interferers. Using the characteristic function (CF) approach outage probability and success probability expressions are given. These expressions are functions of D2D and interference path-loss exponents, distance between the D2D devices, distances between interferers and the D2D receiver and, interference and D2D fading channel conditions. Maximum ratio combining (MRC) and selection combining (SC) based diversity schemes are considered to mitigate channel fading effects. D2D communication system under various conditions of channel fading and interference is numerically analyzed and discussed.**

Index Terms— **Co-channel Interference, Device-to-Device, Fluctuating Beckmann, Outage Probability, Success Probability**

I. Introduction

An explosive growth in high data rate demand on cellular communication systems is expected in near future. This demand is due to the popularity of online gaming, HD video streaming and other social media services. Device-to-device (D2D) communication system has emerged as one of the promising technologies to overcome this problem [1-2]. It is considered to be one of the dynamic techniques of the 5th generation (5G) cellular communication standard. D2D communication is defined as the direct communication of devices with each other, when are in proximity, without involvement of base station (BS). D2D system can enhance the data rate, spectrum utilization and, the energy efficiency of the user devices and the networks [3-4]. Despite many advantages, D2D communication brings some challenges as well. Due to insufficient wireless channel bandwidth and the loss of coordination between wireless devices, co-channel interference (CCI) problem arises [5]. Therefore, effects of CCI should be considered for the analysis of D2D systems. In this paper, performance of the D2D communication system is analyzed with the help of outage probability and success probability. Outage probability performance of D2D communication system using stochastic geometry is studied by authors in [6]. Authors in [6], have not considered any diversity scheme for the system. In [7], authors studied outage performance of D2D system under optimal spectrum allocation strategy. Authors has discussed resource sharing method for user in the system. Authors in [8], discussed outage probability of D2D communication in a cellular network from a general threshold-based perspective. Success probability of D2D communication system over Rician fading channel under the distributed random-access control scheme is analyzed by authors in [9].

Outage and success performance analysis of D2D system in the presence of CCI is the aim of this work. The co-channel interference signals are considered to originate from any wireless device with which the system has lost coordination. The channel for the D2D and CCI signals are assumed to follow Fluctuating Beckmann (FB) distribution. FB fading model is a generalized fading model which includes many fading models as special case [10]. The one-sided Gaussian, Rayleigh, Nakagami, *η-μ*, *κ-μ*, *κ-μ* shadowed, Rician, Rician Shadowed and the Beckmann distribution are the special cases of FB model [10]. The *κ-μ* shadowed fading model [11] manages to capture propagation conditions like clustering and LoS fluctuation. However, it fails when there is power imbalance in the LoS and NLoS components. Fluctuating Beckmann (FB) fading model which is an extended *κ-μ* shadowed model is introduced in [10]. FB model effectively captures such scenarios. FB model is a generalization of Beckmann fading model by considering the effects of line-of-sight (LoS) fluctuation and clustering. It also takes into account the effects of power imbalance in the LoS and non-LoS components [10]. CCI effects on the performance of D2D communication system is analyzed. Use of generalized FB model has enabled us to present analytical expressions that can be used to analyze various fading conditions. The SC and MRC based diversity schemes are also considered to combat fading effects. Path loss conditions are also included in the analysis. Outage and success probability expressions for the non-identically distributed D2D and CCI signals are presented. Numerical analysis under various channel and CCI conditions are presented and discussed. Numerical results from these expressions are obtained with the help of MATLAB. The rest of paper is structured as follows: the system model is discussed in Section II. Also analytical expressions for outage probability and success probability are presented in Section II. In Section III, numerical results are presented and discussed. Finally, paper is concluded in Section IV.

DOI: 10.36244/ICJ.2019.4.6

II. SYSTEM MODEL

A device-to-device (D2D) communication system in an interference limited environment is considered here. The system model is illustrated in Fig. 1. There are *N* co-channel interferers that are affecting the D2D communication system. The D2D signals and co-channel interference (CCI) signals are considered to be independent and non-identically distributed. The co-channel interferers are assumed to be at different distances from the receiver of the D2D pair. The channels for D2D and CCI signals are assumed to be Fluctuating Beckmann (FB) distributed. Path-loss is a significant factor in performance degradation of any communication system. In this paper, a simplified path-loss model is considered [12]. To reduce the effects of fading maximal ratio combining (MRC) and selection combining (SC) based diversity techniques with *D* branches are considered in the system.

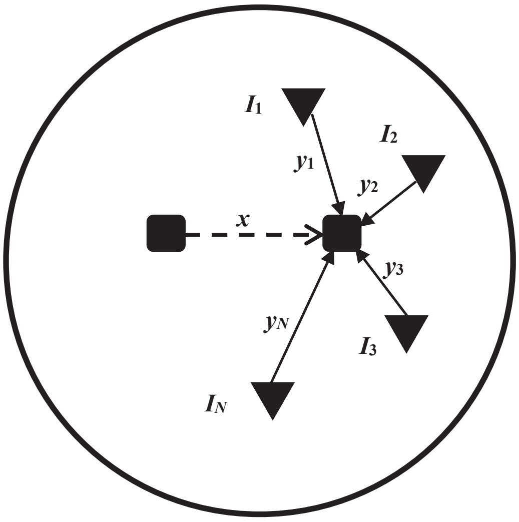


Fig. 1. System Model

D2D device	■
Desired D2D Signal	- - - - ->
Distance between D2D devices	x
n-th Co-channel Interferer	▼
Co-channel Interference Signal	→
Distance between the *n-th* Co-channel Interferer and the D2D Receiver	y_n

A. Selection Combining (SC) Scheme

The signal-to-interference ratio (SIR) at the *d*-th diversity branch of the selection combining (SC) based D2D system is

$$\frac{S_{S-SC,d}}{S_I} = \frac{P_1\left(\dfrac{{x_0}^{u-2}}{x^u}\right)h_d}{\sum_{n=1}^{N} P_{I,n}\left(\dfrac{{y_{0,n}}^{v_n-2}}{{y_n}^{v_n}}\right)\beta_n} \tag{1}$$

where $S_{S\text{-}SC,d}$ is the received power at the *d*-th diversity branch of D2D receiver, P_1 is the power of the D2D signal, *x* is the distance between desired D2D devices, *u* is path-loss exponent for the D2D signal, x_0 is the reference distance and h_d is an independent FB fading variable in the *d*-th diversity branch. Similarly, S_I is the received power of CCI signals, $P_{I,n}$ is the power of the *n*-th CCI signal which is originated from a device at a distance y_n from the D2D receiver, $y_{0,n}$ is the reference distance, v_n is the path-loss exponent of the *n*-th co-channel interferer and β_n is an independent FB fading variable of the *n*-th CCI signal. The outage probability is defined as the probability that the instantaneous SIR of the communication system falls below a predefined threshold *R*. The outage probability for the D2D system is

$$P_{out,SC} = \Pr\left(RS_I > S_{S-SC,MAX}\right) \tag{2}$$

where $S_{S-SC,MAX} = \max_{d=1,\ldots,D}\left(S_{S-SC,d}\right)$ and Pr(.) is probability. Based on (2), a decision variable ∂ is defined as

$$\partial = RS_I - S_{S,SC,MAX} \tag{3}$$

For a successful reception, the value of ∂ must be less than zero, otherwise outage will occur. Mathematically,

$$\partial\begin{cases} >0 & \text{Outage} \\ \leq 0 & \text{Successful Transmission} \end{cases} \tag{4}$$

In this study, a characteristic function (CF) based approach is used for the outage analysis. The CF expression for the decision variable is obtained with the help of [10] and is given as,

$$\begin{aligned}\phi_\partial(\omega) = {} & \frac{\left[1+\dfrac{1}{m_d}\left(\dfrac{j\omega C_d}{G_d + j\omega E_d}+\dfrac{j\omega F_d}{G_d + j\omega H_d}\right)\right]^{-m_d}}{\left[\left(1+j\omega A_d\right)\left(1+j\omega B_d\right)\right]^{\frac{\mu_d}{2}}} \\ & \times\prod_{n=1}^{N}\left[\frac{\left[1-\dfrac{1}{m_n}\left(\dfrac{j\omega C_n}{G_n - j\omega E_n}+\dfrac{j\omega F_n}{G_n - j\omega H_n}\right)\right]^{-m_n}}{\left[\left(1-j\omega A_n\right)\left(1-j\omega B_n\right)\right]^{\frac{\mu_n}{2}}}\right]\end{aligned} \tag{5}$$

where $A_d = \dfrac{2\eta_d\Omega_d\delta_d}{\mu_d\left(1+\eta_d\right)\left(1+\kappa_d\right)}$, $B_d = \dfrac{2\Omega_d\delta_d}{\mu_d\left(1+\eta_d\right)\left(1+\kappa_d\right)}$,

$C_d = \mu_d\kappa_d\left(\dfrac{{\zeta_d}^2}{1+{\zeta_d}^2}\right)\left(1+\eta_d\right)\Omega_d\delta_d$, $\qquad E_d = 2\eta_d\Omega_d\delta_d$,

$G_d = (1+\eta_d)(1+\kappa_d)\mu_d$, $H_d = 2\Omega_d\delta_d$, $F_d = \mu_d\kappa_d\left(\frac{1}{1+\zeta_d^{\,2}}\right)(1+\eta_d)\Omega_d\delta_d$ and $A_n = \frac{2\eta_n\Omega_n\delta_n}{\mu_n(1+\eta_n)(1+\kappa_n)}$, $B_n = \frac{2\Omega_n\delta_n}{\mu_n(1+\eta_n)(1+\kappa_n)}$, $C_n = \mu_n\kappa_n\left(\frac{\zeta_n^{\,2}}{1+\zeta_n^{\,2}}\right)(1+\eta_n)\Omega_n\delta_n$, $E_n = 2\eta_n\Omega_n\delta_n$, $F_n = \mu_n\kappa_n\left(\frac{1}{1+\zeta_n^{\,2}}\right)(1+\eta_n)\Omega_n\delta_n$, $H_n = 2\Omega_n\delta_n$, $G_n = (1+\eta_n)(1+\kappa_n)\mu_n$, where μ_d and μ_n represent number of clusters of the d-th branch D2D signal and n-th CCI signal, respectively. κ_d and κ_n are related to the strength of line-of-sight (LoS) component, η_d and η_n are the in-phase/quadrature power imbalance in the non-LoS components of the d-th branch D2D signal and n-th CCI signal, respectively. ζ_d^2 and ζ_n^2 are the in-phase/quadrature power imbalance in the LoS components, m_d and m_n accounts the fluctuation in LoS components, and, Ω_d and Ω_n are average power of the d-th branch D2D signal and n-th CCI signal, respectively. Moreover, $\delta_d = P_1\left(\frac{x_0^{\,u-2}}{x^u}\right)$ and $\delta_n = RP_{I,n}\left(\frac{y_{0,n}^{\,v_n-2}}{x^{v_n}}\right)$. Outage probability of the D2D system is obtained by using the identity, $Pout = \frac{1}{2}+\frac{1}{\pi}\int_0^\infty \frac{\mathrm{Im}(\phi_\partial(\omega))}{\omega}\,d\omega$, where Im (.) gives the imaginary part. The outage probability for the SC diversity based D2D system is

$$P_{out,SC} = \prod_{d=1}^{D}\left[\frac{1}{2}+\frac{1}{\pi}\int_0^\infty \frac{\mathrm{Im}(\phi_\partial(\omega))}{\omega}\,d\omega\right] \tag{6}$$

The outage expression in (6) is for the independent but non-identically distributed D2D and CCI signals. The outage probability for independent and identically distributed case is

$$P_{out,SC} = \left[\frac{1}{2}+\frac{1}{\pi}\int_0^\infty \frac{\mathrm{Im}(\phi_\partial(\omega))}{\omega}\,d\omega\right]^D \tag{7}$$

Success probability is defined as the probability that the SIR of the communication system remains above a predefined threshold R. The expression for the success probability of SC diversity based D2D system is

$$P_{S,SC} = 1-\prod_{d=1}^{D}\left[\frac{1}{2}+\frac{1}{\pi}\int_0^\infty \frac{\mathrm{Im}(\phi_\partial(\omega))}{\omega}\,d\omega\right] \tag{8}$$

The success probability expression presented in (8) is for the independent but non-identically distributed D2D and CCI signals. The success probability expression for the independent and identically distributed case is

$$P_{S,SC} = 1-\left[\frac{1}{2}+\frac{1}{\pi}\int_0^\infty \frac{\mathrm{Im}(\phi_\partial(\omega))}{\omega}\,d\omega\right]^D \tag{9}$$

B. *Maximal Ratio Combining (MRC) Scheme*

The SIR of the D2D system at the output of D branches MRC combiner is

$$\frac{S_{S,MRC}}{S_I} = \frac{P_1\left(\frac{x_0^{\,u-2}}{x^u}\right)\sum_{d=1}^{D} h_d}{\sum_{n=1}^{N} P_{I,n}\left(\frac{y_{0,n}^{\,v_n-2}}{y_n^{\,v_n}}\right)\beta_n}. \tag{10}$$

where $S_{S,MRC}$ is the received power of the D2D signal. The outage probability for the MRC based D2D system is

$$P_{out,MRC} = \Pr\left(RS_I > S_{S,MRC}\right) \tag{11}$$

Using the expression given in (11), a decision variable θ is defined as

$$\theta = RS_I - S_{S,MRC} \tag{12}$$

For a satisfactory desired D2D signal quality, the value of θ has to be negative. Otherwise, outage will happen. Mathematically,

$$\theta\begin{cases} >0 & \text{Outage} \\ \le 0 & \text{Acceptable Transmission}\end{cases} \tag{13}$$

The CF of the decision variable θ is given as

$$\phi_\theta(\omega) = \prod_{d=1}^{D}\left[\frac{\left[1+\frac{\left(\frac{j\omega C_d}{G_d+j\omega E_d}+\frac{j\omega F_d}{G_d+j\omega H_d}\right)}{m_d}\right]^{-m_d}}{\left[(1+j\omega A_d)(1+j\omega B_d)\right]^{\frac{\mu_d}{2}}}\right]$$

$$\times\prod_{n=1}^{N}\left[\frac{\left[1-\frac{\left(\frac{j\omega C_n}{G_n-j\omega E_n}+\frac{j\omega F_n}{G_n-j\omega H_n}\right)}{m_n}\right]^{-m_n}}{\left[\left(1-j\omega A_n\right)\left(1-j\omega B_n\right)\right]^{\frac{\mu_n}{2}}}\right] \tag{14}$$

Based on (13) and (14), the outage probability expression for independent but non-identically distributed D2D and CCI signals is

$$P_{out,MRC}=\frac{1}{2}+\frac{1}{\pi}\int_0^\infty\frac{\mathrm{Im}\left(\phi_\theta\left(\omega\right)\right)}{\omega}d\omega \tag{15}$$

The outage probability expression for independent and identically distributed system is

$$P_{out,MRC}=\frac{1}{2}+\frac{1}{\pi}\int_0^\infty\frac{\mathrm{Im}\left(\left[X_d\right]^D\left[Y_n\right]^N\right)}{\omega}d\omega \tag{16}$$

where

$$X_d=\frac{\left[1+\frac{1}{m_d}\left(\frac{j\omega C_d}{G_d+j\omega E_d}+\frac{j\omega F_d}{G_d+j\omega H_d}\right)\right]^{-m_d}}{\left(1+j\omega A_d\right)^{\frac{\mu_d}{2}}\left(1+j\omega B_d\right)^{\frac{\mu_d}{2}}}$$

and

$$Y_n=\frac{\left[1-\frac{1}{m_n}\left(\frac{j\omega C_n}{G_n-j\omega E_n}+\frac{j\omega F_n}{G_n-j\omega H_n}\right)\right]^{-m_n}}{\left(1-j\omega A_n\right)^{\frac{\mu_n}{2}}\left(1-j\omega B_n\right)^{\frac{\mu_n}{2}}}$$

The success probability expression for the independent and non-identically distributed system is

$$P_{S,MRC}=\frac{1}{2}-\frac{1}{\pi}\int_0^\infty\frac{\mathrm{Im}\left(\phi_\theta\left(\omega\right)\right)}{\omega}d\omega \tag{17}$$

The success probability expression for the independent and identically distributed case is

$$P_{S,MRC}=\frac{1}{2}-\frac{1}{\pi}\int_0^\infty\frac{\mathrm{Im}\left(\left[X_d\right]^D\left[Y_n\right]^N\right)}{\omega}d\omega \tag{18}$$

III. Numerical Results and Analysis

In this section, the performance of FB faded D2D system based on the expressions derived in Section II is presented. The reference distances x_0 and $y_{0,n}$ are assumed to be 1 meter. Fig. 2 shows the outage performance of D2D system with the various number of diversity branches for both SC and MRC schemes. The values for D2D signals parameters for number of diversity branches $D = 2$ and 3, and CCI signals parameters are shown in Table 1.

Table 1.

Parameters	Values	Parameters	Values
u	3.4	$P_{I,n}$	[16.98, 17.78, 18.45, 19.03, 20] dBm
x	15 meters	v_n	[2.7, 2.8, 2.9, 3.0, 2.6]
μ_d	[10,7] & [10, 7, 1]	y_n	[45, 40, 35, 30, 25] meters
κ_d	[10, 3] & [10, 3, 8]	μ_n	[1, 3, 7, 3, 4]
ζ_d^2	[0.1, 0.01] & [0.1, 0.01, 1]	ζ_n^2	[0.01, 0.1, 0.001, 1, 2]
η_d	[10, 5] & [10, 5, 6]	κ_n	[1, 5, 6, 4, 5]
m_d	[3, 2] & [3, 2, 7]	η_n	[1, 2, 7, 2, 3]
R	16.98	m_n	[1, 3, 4, 2, 6]

From the figure, it is clear that as the number of branches is increased for the SC and MRC schemes outage performance improves. It is because of the improved SIR conditions of the system. Furthermore, it is seen that the increase in power of the D2D signal improves the outage performance of the system. Analytical and simulation of Outage performance of D2D communication system with various values μ_d and x is shown in Fig. 3. The values for D2D signals parameters and CCI signals parameters are shown in Table 2.

Table 2.

Parameters	Values	Parameters	Values
P_1	20 dBm	v_n	[3, 3.1, 3.2, 3.3, 3]
u	2.6	μ_n	[1, 3, 7, 3, 4]
κ_d	[0, 0]	κ_n	[0, 0, 0, 0, 0]
η_d	[0, 0]	η_n	[0, 0, 0, 0, 0]
m_d	[0, 0]	m_n	[0, 0, 0, 0, 0]
ζ_d^2	[0, 0]	ζ_n^2	[0, 0, 0, 0, 0]
$P_{I,n}$	[20, 23.01, 24.77, 26.02, 26.99] dBm	y_n	[45, 40, 35, 30, 30] meters

From the figure, it is observed that the outage probability of the system is less for higher values fading parameter μ_d. It is because of the fading condition of D2D signals which results in an improved outage performance of the D2D system. Moreover, the increase in distance between D2D devices degrades the outage performance of the system. It is because of the path-loss phenomena. Outage performance of D2D system for various values of v_n and x is shown in Fig. 4. The values for the three branches of MRC based D2D signal parameters and CCI signal Parameters are given in Table 3.

Table 3.

Parameters	Values	Parameters	Values
P_1	20 dBm	$P_{I,n}$	[16.98, 17.78, 18.45, 19.03, 20] dBm
u	3.2	μ_n	[1, 3, 4, 3, 2]
κ_d	[10, 3, 8]	κ_n	[1, 2, 3, 4, 1]
η_d	[10, 5, 6]	η_n	[1, 2, 3, 2, 3]
m_d	[6, 5, 7]	m_n	[1, 3, 4, 2, 3]
ζ_d^2	[0.1, 0.01, 1]	ζ_n^2	[0.01, 0.1, 0.001, 0.5, 1]
μ_d	[10, 7, 1]	y_n	[25, 30, 35, 40, 45] meters

From the figure, it is observed that the outage probability of the system is less for higher values CCI path-loss exponents. It is because of the weakening of CCI signals which results in an improved outage performance of the D2D system. Moreover, the increase in distance between D2D devices degrades the outage performance of the system.

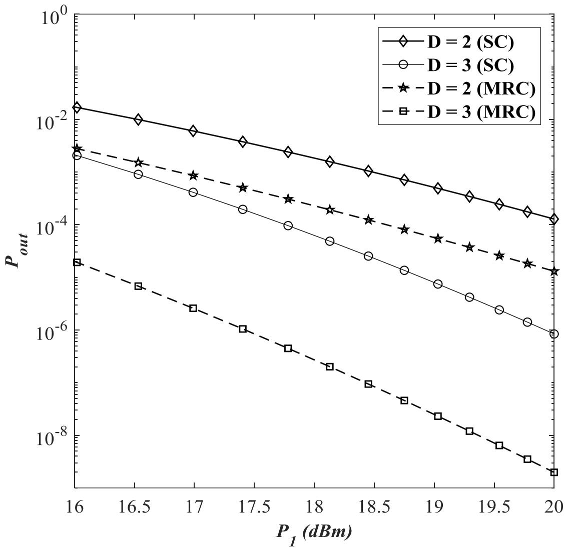


Fig. 2. Outage probability for various numbers of SC and MRC branches

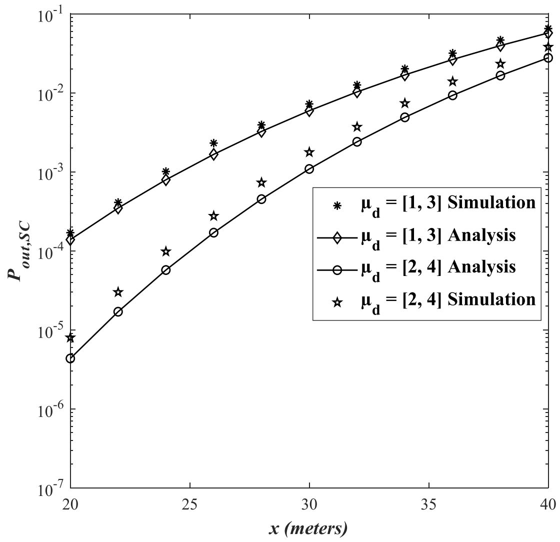


Fig. 3. Outage Probability for various values of fading parameter

In Fig. 5, outage performance of D2D system for various values of u and m_d is shown. The values for the three branches of SC based D2D signal parameters and CCI signal parameters are given in Table 4.

Table 4.

Parameters	Values	Parameters	Values
P_1	20 dBm	$P_{I,n}$	[16.98, 17.78, 18.45, 19.03, 20] dBm
x	19 meters	μ_n	[1, 3, 7, 3, 4]
κ_d	[10, 3, 8]	κ_n	[1, 5, 6, 4, 5]
η_d	[10, 5, 6]	η_n	[1, 2, 7, 2, 3]
μ_d	[10, 7, 1]	m_n	[1, 3, 4, 2, 3]
ζ_d^2	[0.1, 0.01, 1]	ζ_n^2	{0.01, 0.1, 0.001, 1, 2}
v_n	[2.7, 2.8, 2.9, 3, 2.6]	y_n	[25, 30, 35, 40, 45] meters

From the figure, it is observed that the outage probability of the system is worse for the higher values of the u. It is due to the weakened D2D signal due to the path-loss effects. Moreover, from the figure it can also be seen that the outage performance of the system improves as the values of m_d is increased. It is because of the better shadowing conditions of the D2D communication channel.

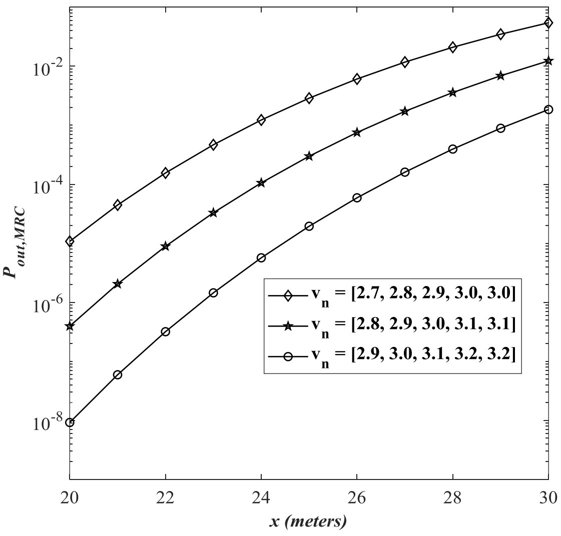


Fig. 4. Outage performance of MRC based D2D system with varying path-loss exponents of CCI signals

Fig. 6 presents outage performance of D2D communication system with varying values of m_n. The values for P_1, $P_{I,n}$, x, y_n, u, v_n, μ_d, μ_n, κ_d, κ_n, η_d, η_n, ζ_d^2, ζ_n^2 and R are considered to be 20 dBm, 16.98 dBm, 15 meters, 35 meters, 3.3, 2.7, 2, 2, 5, 4, 0.1, 0.1, 0.01, 0.01 and 20 dBm, respectively. From the figure it is observed that the outage performance is almost insensitive to the variations of m_n. Furthermore, it is also observed that the outage performance improves as the values of m_d is increased.

Outage performance of D2D system for the varying values of ζ_n^2 and η_n in a scenario with $\mu_n = 1$ and μ_d is shown in Fig. 7. The values for P_1, $P_{I,n}$, x, y_n, u, v_n, κ_d, κ_n, η_d, m_d, m_n, ζ_d^2 and R are set to be 20 dBm, 16.98 dBm, 15 meters, 40 meters, 3.5, 2.5, 10, 1, 10, 3, 5, 0.1, and 16.98 dBm, respectively. From the figure it is observed that outage performance of the system is better for the higher values of η_n. Moreover, from the figure it can be seen that the outage performance improves as the number of clusters of desired D2D signal increases.

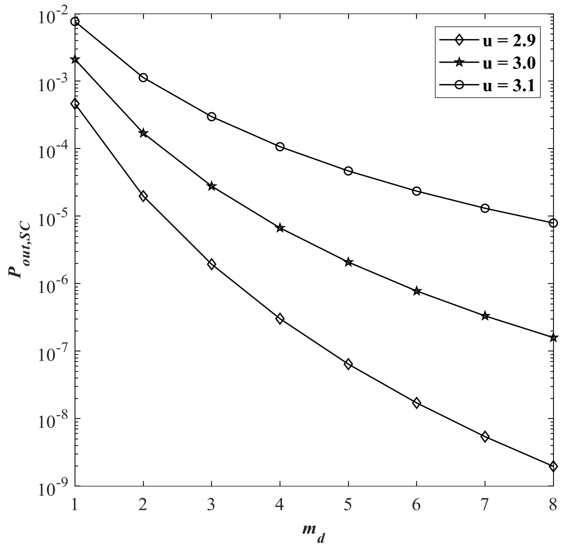


Fig. 5. Outage performance of SC based D2D system for various values of path-loss exponent u

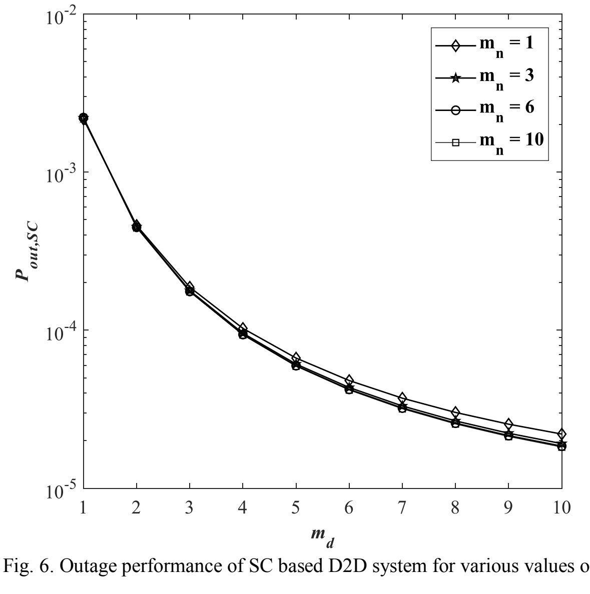


Fig. 6. Outage performance of SC based D2D system for various values of m_n

Success probability performance of D2D system for κ_n and κ_d is shown in Fig. 8. The values for P_1, $P_{I,n}$, x, y_n, u, v_n, η_d, η_n, m_d, m_n, μ_d, μ_n, ζ_d^2, ζ_n^2 and R are set to be 20 dBm, 16.98 dBm, 15 meters, 30 meters, 3.8, 2.5, 2, 2, 3, 1, 2, 2, 0.01, 0.01 and 16.98 dBm, respectively. From the figure it is observed that success probability performance of the system is better for the lower values κ_n. It is because of degraded CCI signals which results in an improved the SIR performance of the system, and hence better success probability performance of the system. It can also be seen that the success probability of the system increases as the values of κ_d increases. In Fig. 9, success probability performance of D2D communication system with varying values of and is shown. P_1, $P_{I,n}$, x, y_n, u, v_n, κ_d, κ_n, η_d, η_n, m_d, m_n, μ_d, μ_n and R are fixed at 20 dBm, 16.98 dBm, 16 meters, 25 meters, 3.5, 2.5, 1, 5, 10, 2, 1, 1, 3, 3 and 20 dBm, respectively. From the figure, it can be seen that the success probability does not show much variation when ζ_n^2 is varied. Furthermore, it is also observed that the success probability performance of the system deteriorates as ζ_d^2 is increased.

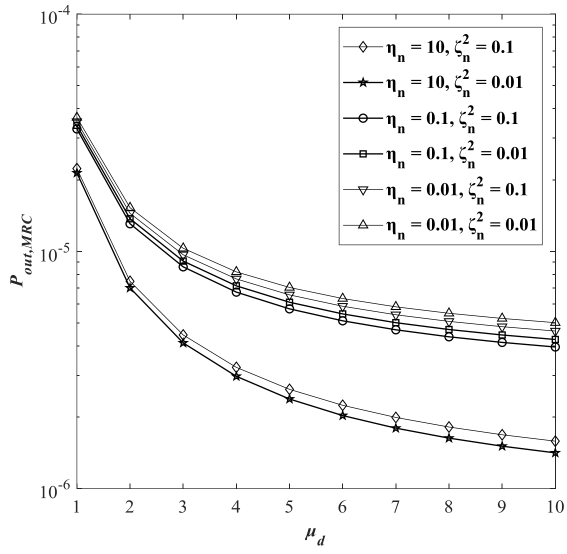


Fig. 7. Outage performance of MRC based D2D system for various values of η_n and ζ_n^2

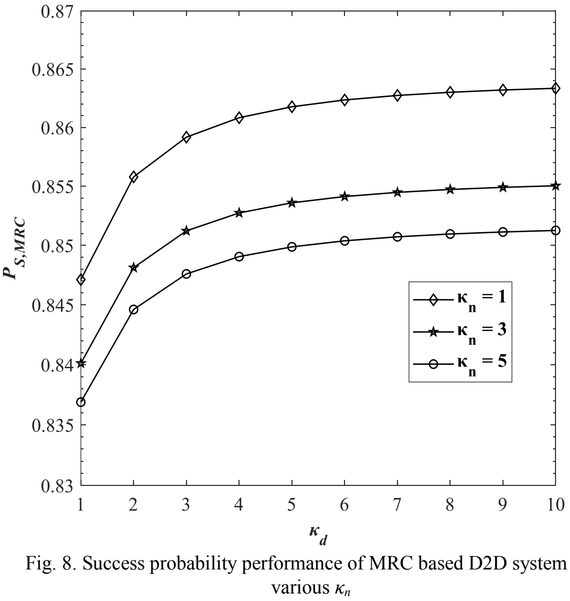


Fig. 8. Success probability performance of MRC based D2D system for various κ_n

Success probability performance of the D2D system with varying values μ_n and the distance y is analyzed in Fig. 10. Here, the values for parameters P_1, $P_{I,n}$, x, u, v_n, κ_d, κ_n, η_d, η_n, m_d, m_n, $\zeta_d^{\,2}$, $\zeta_n^{\,2}$ and μ_n are fixed at 20 dBm, 17.78 dBm, 16 meters, 3.5, 2.5, 1, 5, 10, 5, 1, 10, 10, 0.1 and 2, respectively. From the figure, it is evident that the success probability performance of the system is almost insensitive to the change in the values of μ_n. Moreover, from the figure it is also seen that the success probability of the system improves as the distance y is increased. It is because of the improved SIR conditions of the system due to weakening of CCI signals by the path-loss.

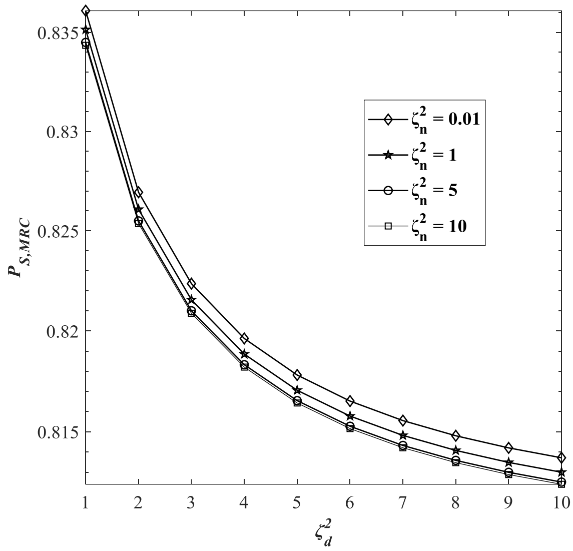


Fig. 9. Success probability performance of MRC based D2D system with varying values of $\zeta_n^{\,2}$

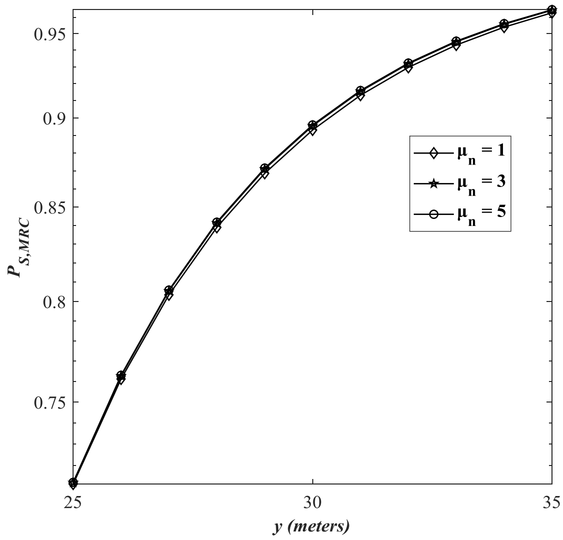


Fig. 10. Success Probability of MRC based D2D system with varying values of μ_n

IV. Conclusion

In this paper, outage and success performances of a D2D communication system over a Fluctuating Beckmann (FB) fading channel in an interference limited scenario is analyzed. The FB distribution generalizes various distributions. Expressions of outage and success probabilities using characteristic function (CF) based approach is presented. Effects of co-channel interference, FB channel conditions and the path-loss on the outage and success probabilities of the system are presented and discussed. MRC and SC diversity schemes are also incorporated to mitigate fading conditions. It is observed that path-loss significantly effects performance of D2D system. It is also observed that the system performance improves as the number of clusters in the D2D signal is increased. However, system performance is almost insensitive to the variations in the number of clusters of CCI.

Appendix A
Proof of Eq. No. 5

$$\partial = RS_I - S_{S,SC,MAX}$$

$$\begin{aligned}
\phi_\partial(\omega) &= E\left(e^{\,j\omega\left(R\,S_I - S_{S,SC,MAX}\right)}\right) \\
&= E\left(e^{j\omega\left(R\,S_I\right)} e^{j\omega\left(-S_{S,SC,MAX}\right)}\right) \\
&= E\left(e^{j\omega\left(\sum_{n=1}^{N}\delta_n\beta_n\right)} e^{j\omega\left(-\delta_d h_d\right)}\right) \\
&= E\left(e^{j\omega\left(\sum_{n=1}^{N}\delta_n\beta_n\right)}\right) E\left(e^{j\omega\left(-\delta_d h_d\right)}\right) \\
&= E\left(e^{j\omega\left(-\delta_d h_d\right)}\right) \prod_{n=1}^{N} E\left(e^{j\omega\,\delta_n\beta_n}\right) \\
&= \phi_d\left(-\omega\,\delta_d\,h_d\right) \prod_{n=1}^{N} \phi_I\left(\omega\,\delta_n\,\beta_n\right)
\end{aligned}$$

References

[1] You, D. and Dong H. K., "Hybrid STBC–SM Suitable for Multi-link Device-to-Device Communication in Cellular Networks," Wireless Personal Communications, vol. 96, no. 1, pp. 1507-1518., Sept. 2017, [https://doi.org/10.1007/s11277-017-4253-9].

[2] Sharifi, S. and Mohammad F., "Underlay device to device communication with imperfect interference channel knowledge," Wireless Personal Communications, vol. 101, no. 2, pp. 619-634, July 2018, [https://doi.org/10.1007/s11277-018-5707-4].

[3] Schreck, J., Jung P. and Stańczak, S., "Compressive Rate Estimation With Applications to Device-to-Device Communications," in IEEE Transactions on Wireless Communications, vol. 17, no. 10, pp. 7001-7012, Oct. 2018, [https://doi.org/10.1109/TWC.2018.2865347].

[4] Wang, J., Huang, Y., Jin, S., Schober, R., You X. and Zhao, C., "Resource Management for Device-to-Device Communication: A Physical Layer Security Perspective," in IEEE Journal on Selected Areas in Communications, vol. 36, no. 4, pp. 946-960, April 2018, [https://doi.org/10.1109/JSAC.2018.2825484].

[5] Liu, F., Hou, X. and Liu, Y., "Capacity Improvement for Full Duplex Device-to-Device Communications Underlaying Cellular Networks," in IEEE Access, vol. 6, pp. 68373-68383, 2018, [https://doi.org/10.1109/ACCESS.2018.2879472].

[6] Huq, K. M. S., Mumtaz, S. and Rodriguez, J., "Outage probability analysis for device-to-device system," 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, 2016, pp. 1-5, [https://doi.org/10.1109/ICC.2016.7510678].

[7] Khuntia, P. and Hazra, R., "Resource sharing for device-to-device communication underlaying cellular network," 2018 4th International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, 2018, pp. 1-5 [https://doi.org/10.1109/RAIT.2018.8389093].

[8] Liu, J., Nishiyama, H., Kato, N. and Guo, J., "On the Outage Probability of Device-to-Device-Communication-Enabled Multichannel Cellular Networks: An RSS-Threshold-Based Perspective," in IEEE Journal on Selected Areas in Communications, vol. 34, no. 1, pp. 163-175, Jan. 2016, [https://doi.org/10.1109/JSAC.2015.2452492].

[9] Peng, M., Li, Y., Quek, T. Q. S. and Wang, C., "Device-to-Device Underlaid Cellular Networks under Rician Fading Channels," in IEEE Transactions on Wireless Communications, vol. 13, no. 8, pp. 4247-4259, Aug. 2014, [https://doi.org/10.1109/TWC.2014.2314115].

[10] Ramirez-Espinosa, P., Lopez-Martinez, F. J., Paris, J. F. , Yacoub, M. D. and Martos-Naya, E., "An Extension of the $\varkappa$-μ Shadowed Fading Model: Statistical Characterization and Applications," in IEEE Transactions on Vehicular Technology, vol. 67, no. 5, pp. 3826-3837, May 2018 [https://doi.org/10.1109/TVT.2017.2787204].

[11] J.F. Paris, "Statistical Characterization of $\varkappa$-μ Shadowed Fading,", in IEEE Transactions on Vehicular Technology, vol. 63, no. 2, pp. 518-526, Feb. 2014, [https://doi.org/10.1109/TVT.2013.2281213].

[12] Goldsmith, A., Wireless communications, Cambridge university press, 2005.

Zakir Hussain received M.S. degree in Electrical Engineering from National University of Computer and Emerging Sciences (NUCES), Karachi Campus in 2017. Research areas includes Device-to-Device communication, interference analysis in wireless communication systems, digital communication systems, signal processing.

Asim ur Rehman Khan received BSc degree in Electrical Engineering (EE) from UET Lahore, Pakistan in 1981, MS EE from South Dakota State University, South Dakota, USA in 1987, and PhD EE from Polytechnic University (now NYU) in Brooklyn, New York, 8 USA in 1993. His main area of research is AI, Computer Vision, and Pattern Recognition. He has worked in various telecom projects in Pakistan. Since 2002, he is Professor of Electrical Engineering at the Karachi campus of National University of Computer; Emerging Sciences (NUCES), Pakistan.

Haider Mehdi research interests include Device-to-Device communication, interference analysis in wireless communication systems, digital communication systems, digital signal processing, and signal-processing techniques for the communication systems.

Aamir Ali received the B.S. and M.S. degrees in Electrical Engineering from National University of Computer and Emerging Sciences in 2015 and 2019, respectively. He has been associated with the faculty of Electrical Engineering Department at National University of Computer and Emerging Sciences since Jan 2016. He has been author of many research papers. His research interest include Robotics and Telecommunication.

Quantum Optimization of Resource Distribution Management for Multi-Task, Multi-Subtasks

Sara El Gaily and Sándor Imre

***Abstract*— This paper proposes a new optimization strategy for resource distribution management based on a quantum algorithm, as a way to reduce the computational complexity in finding the optimum deployment scenario, taking into consideration the required conditions and constraints of the resource distribution system. We show that the quantum method computes the results in minimum time and outperforms on the other classical algorithms in terms of computational complexity.**

***Index Terms* —quantum computing; resource distribution management; *quantum extreme value searching algorithm*; quantum existence testing; computational complexity.**

I. INTRODUCTION

A. Motivation

The first question that comes to the mind of the reader is how the quantum optimization methods may increase the performance system of resource distribution management process and how it will be used in resource management as a way to reduce the computational complexity in finding the optimum deployment scenario. What are the fundamental differences between a classical computer and quantum computer which can lead to choosing the quantum strategy as a future alternative solution for the resource distribution management model?

B. Quantum Computing Overview

In fact, quantum computer's functionality and conception work based on the laws of quantum mechanics. There is a large list of differences between the quantum computer and classical computer. First of all, classical computer functionality works based on the laws of classical mechanics, it performs calculations relying on the basic unit of information zeros and ones (0 and 1), while quantum computer uses qubits which can take superposition of states at the same time [1], furthermore, quantum computer outperforms with high speed than the binary computer, as well as it can solve computational problems with low computational complexity, maximum accuracy, and short circumstance).

If we assume that a large number of binary computers can combine their efforts and overcome this gap, they cannot reach the performance level of a quantum computer. Quantum computing and information have important quantum algorithms that solve important computational problems which do not seem to be possibly solved by a classical method, for example, the most known actually are the quantum Fourier transform which is used to solve factoring and discrete logarithm problems, and its fascinating advantage to make the communication over a quantum channel more secure, and the quantum search algorithm [1][2] the so-called Grover's algorithm [3][4] which uses fewer steps than its classical counterpart to find a certain entry in an unsorted data with more accuracy, speed and less time.

C. Establishing context and the importance of the research topic

Resource distribution management must be designed to be highly reacting fast with maximum accuracy performance to any unpredictable task workload, as it is known, for tasks with fixed running time require more computation in a real-time system, since they are executed at a constant rate. In order to rationally use resource computing as a way to reduce the computational complexity in finding the best optimum deployment scenarios under the imposed constraints, we resorted to handling this problem by using an approach based on the quantum method. This study provides an important opportunity to improve the efficiency of using system resources by exploiting quantum computing methods and concepts.

Suppose all classical machines that are working on classical laws of classical physics will disappear from our world and be replaced by quantum computers, so there will be an intensive need for developing new adaptive models. From a resource distribution management point of view, if the real-time decision-maker will be replaced by a quantum method, so, the way becomes open to think how to implement a resource distribution management model with the new device since new hardware technique requires a new resource distribution process modeling.

D. A Brief Synopsis of the Relevant Literature

It is difficult to relate the proposed strategy to other work in the literature because the proposed resource distribution management based on quantum optimization is a new contribution. So, we will try to give approximately and generally

A preliminary version of this paper has been presented at the 2019 42nd International Conference on Telecommunications and Signal Processing (TSP) [15].

The research was partially supported by the National Research Development and Innovation Office of Hungary (Project No.2017-1.2.1-NKP-2017-00001), by the Hungarian Scientific Research Fund. The research reported in this paper was supported by the BME- Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI/SC)".

Department of Networked Systems and Services, Budapest University of Technology and Economics. / Műegyetem. 3-9, Budapest, 1111, HUNGARY. Email: {elgaily, imre}@hit.bme.hu.

DOI: 10.36244/ICJ.2019.4.7

the recent works that have been proposed in this field. Periodic activities have the major computational demand in many real-time applications since they provide a simple way to enforce timing constraints through rate control [5], there has been a greater interest in proposing new techniques appropriate in using system resources of fixed real-time tasks. In [6], the Quality of service-based resource distribution which addresses the problem of distributing a bandwidth portion among services merged with the distribution algorithm in order to decrease computational complexity. In [7], a proposed solution for finding the optimal task periods for practical problems with a remarkable speedup by exploiting the concept of the exact feasibility region of the space. In [5]-[8], the elastic task model (ETM) was taken as an efficient mechanism for controlling the quality of service of the system as a function of the current load, the ETM is extremely useful for supporting both multimedia systems and control applications in which the execution rate of some computational activities have to be tuned as a function of the current system state. On the other hand, some recent works were using the Hungarian combinatorial algorithm as a tool for assigning tasks, for example, in [9] for multi-task to multi-worker allocation based on the demand distribution model, or, in [10], a decentralized task allocation algorithm based on the Hungarian approach. In [11], for channel allocation problem over a frequency-selective channel. Moreover, for a multicasting problem, two heuristics algorithms [12], Farthest First and Nearest First based were applied to minimize the number of used wavelengths. In [13], the orchestration algorithm was used in a heterogonous cloud environment to minimize the usage of computing resources.

This study is an extension of the previously published work, in [14] we have been started from a simple resource distribution management model, for one task generator: (a) demonstrating analytically that the quantum solution is more efficient by comparing the computational complexity and distribution uniformity of the quantum solution with the randomized, exhaustive and sequence methods, (b) showing the importance of the quantum solution, a simulation environment of the proposed optimization of distribution system was constructed and compared to two reference distribution systems which follow the randomized and sequence strategies. In [15], we have set up carefully the system parameters of the quantum algorithm with respect to the proposed resource distribution model (it contains one task generator). Furthermore, we discussed the most important parameters and derived the appropriate approximation formulas if different computation units are allowed in the system.

E. Contribution

This paper provides a new and comprehensive study on reducing the computational complexity of a distribution problem, using a system of multiple task generators which dissociate each task to several subtasks, integrating resource distribution model in the quantum system-level framework is not straight-forward and may need to a careful configuration of its parameters. The quantum approach will improve the speed of computation as well as the accuracy in selecting the best result, allowing the movement from an $O(d)$ computational complexity to $O\left(log_2(T)log_2{}^3(\sqrt{d})\right)$.

The main questions addressed in this paper are:

- From a computational complexity point of view, how can we use the quantum searching method in resource distribution management? And how much is it efficient?
- From an engineering point of view, how can we set up the stochastic parameters of the quantum logarithm search according to the given resource distribution model?

F. Organization

The remaining part of this paper is organized as follows: Section II begins by describing a resource distribution management model with multi-task and multi-subtasks handling with one optimization metric. Then we will discuss how can we apply the quantum optimization algorithm to improve the efficiency of system resources from a computational complexity point of view, after that, we will demonstrate how the quantum approach is an efficient computational infrastructure tool by comparing it with the classical approach method, finally we will set up the system parameters of the quantum algorithm of the resource distribution management model. Section III concludes the paper.

II. Quantum Resource Distribution Management Optimization in Multi-Task and Multi-Subtasks

In the resource distribution model, the uniformity distribution metric is a perfect standard measurement for checking whether resource utilization is balanced or not, in our study, we will rely fundamentally on the relative load variance of the system for measuring the uniformity distribution degree. For a large number of resources, it is difficult to compute classically the overall possible deployments which fit the optimum distribution, the solution is to exploit the power of quantum approach which will guarantee a high result in computational complexity reduction as well as accuracy performance.

In order to not confuse the reader, before explaining how the quantum method works in the resource distribution management model, first, we will give an overview of the quantum algorithm.

A. Quantum Extreme Value Searching Algorithm Optimization

The quantum extreme value searching algorithm QEVSA [16] combines the well-known logarithmic binary search algorithm which is originally intended for searching a given item in a sorted database [17] with the quantum existence testing, it is represented in the algorithm as QET [18]-[19]. Quantum existence testing is a special case of quantum counting, it focuses on checking the existence of a given entry in the database rather than in determining the number of existent entries. The QEVSA aims to find the extreme value (minimum or maximum point) of a point called cost function or database. Moreover, the power of quantum existence testing is derived from the quantum phase estimation which makes it outperforms better than the other algorithms, this technique produces an algorithm which keeps the efficiency of the binary search while processing an unsorted database. The proposed algorithm is introduced in [16].

B. *Resource Distribution Management Model with multi-task and multi-subtasks*

For a fixed reservation requirement, i.e., the contracted capacity amount used for the task, any available capacity left unreserved cannot be reused by other tasks. maximizing and improving the resource utilization requires using multi-subtask model, i.e., any task has subtasks. As aforementioned, the high utilization of computing resources and huge demand for computation leads to a search for efficient and less operational costs with respect to the required quality of service, for this purpose, we proposed a resource allocation model for running the workload. In order to model the general resource distribution management system, we divided the functionalities into three main blocks:

Multiple task generator: Let u denote the number of task type generators, each task generator has its own task arrival time distribution and produce identical tasks, let the number of subtasks generated by the l^{th} task generator be denoted by g_l and the total number of subtasks type generated by all the task generators is denoted by W. Note that every subtask type v has a fixed running time p_v and the memory requirement for the subtask type v is Δ_v.

Decision maker: It is responsible for the deployment of the subtasks among the computing units, later we will explain the role of the quantum approach in reducing the computational complexity of the task deployment.

Computing units: Let c be the number of computing units used to serve the subtasks, the computing units may have different theoretical capacities, such as the i^{th} unit which has s_i as a theoretical capacity, let the number of running subtasks from type v on-the i^{th} unit is denoted by N^b_{iv}. *Fig.1* represents the resource distribution management structure.

In compliance with what has been already discussed, we have chosen to distribute uniformly the tasks among the computing units, the variance of the relative load in the system is used as a measurement metric for uniformity performance. In the case of optimal task distribution, if the variance of the relative load tends to zero, then the resources are distributed uniformly, otherwise, they are not, the formula of the relative variance is as follows,

$$\sigma^2 = \frac{1}{c}\sum_{i=1}^{c}\left(\bar{b} - \frac{\sum_{v=1}^{W} N_{iv}\Delta_v}{s_i}\right)^2, \qquad (1)$$

Where $\bar{b}$ is the average of the relative load of the system. Later, we will see how the variance is employed in the quantum algorithm for searching the optimum deployment scenario.

C. *How to Use Quantum Extreme Value Searching in Resource Distribution Management*

The reader should bear in mind that for the proposed distribution model, this study did not discuss the resource requirement scheme implementation because the quantum existing testing is a special form of quantum phase estimation and quantum gate circuit structure is well known, in this paper, the modification will be only in the quantum extreme value searching algorithm.

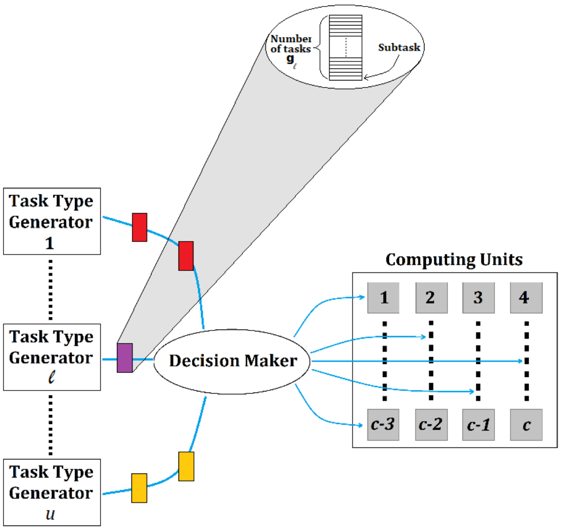


Fig.1: Resource distribution management architecture

Conserving the uniformity load of the system implies finding the optimum scenario that corresponds to the minimum variance, thus, in this case, we use the quantum extreme value searching algorithm as a minimum searching algorithm. What makes this proposed quantum approach special, is that it handles the database as a function, i.e., variance. The corresponding quantum algorithm according to the desired resource distribution model is given as follows,

1. We start with $S = 0$: ${\sigma^2}_{min\,1} = {\sigma^2}_{min\,0}$, ${\sigma^2}_{max\,1} = {\sigma^2}_{max\,0}$, and $\Delta\sigma^2 = {\sigma^2}_{max\,0} - {\sigma^2}_{min\,0}$
2. $S = S + 1$
3. ${\sigma^2}_{med\,S} = {\sigma^2}_{min\,S} + \left[\frac{{\sigma^2}_{max\,S} - {\sigma^2}_{min\,S}}{2}\right]$
4. $flag = QET\,({\sigma^2}_{med\,S})$:

- If $flag = Yes,$ then ${\sigma^2}_{max\,S+1} = {\sigma^2}_{med\,S}, {\sigma^2}_{min\,S+1} = {\sigma^2}_{min\,S}$
- Else ${\sigma^2}_{max\,S+1} = {\sigma^2}_{max\,S}$, ${\sigma^2}_{min\,S+1} = {\sigma^2}_{med\,S}$

5. If S < $log_2\,(T)$, then go to 2, else stop and $y_{opt} = {\sigma^2}_{med\,S}$

The maximum number of the necessary steps to run the logarithm search T depends on two parameters which are the step size of the search which is according to the proposed distribution model is the minimum distance between variances of two scenarios α as presented in (2) and the size of the region of the variance's values $\Delta\sigma^2 = {\sigma^2}_{max} - {\sigma^2}_{min}$, the expression of T is illustrated in (3),

$$\alpha = \min_{\forall V_i, V_j} \left|(\boldsymbol{\sigma}^2_{V_i} - \boldsymbol{\sigma}^2_{V_j})\right|, \qquad (2)$$

$$T = \frac{{\sigma^2}_{max} - {\sigma^2}_{min}}{\alpha}, \qquad (3)$$

Where V_i and V_j referred to two different assignment scenarios. Note that the stochastic variable α depends on many parameters such as the number of presented subtasks, processing time of each arrival task type distribution, the number of presented computing resources, etc.

Note that, integrating the resource distribution model in the framework of quantum system-level is not straight-forward and may need a careful configuration to its parameters. We are interested in providing a rigorous mathematical demonstration for bounding α based on the performance specifications of the proposed resource distribution model, as it is required in real physical implementation.

In the current subsection, we answered the fundamental question on how to apply the quantum method in the resource deployment system. In the next subsection, we will present an analytical comparison between the proposed quantum strategy and the classical counterpart.

D. Analytical comparison between the computational complexity of the quantum and the classical strategy.

The quantum minimum searching algorithm is used as a tool to reduce the computational complexity for selecting the optimum deployment scenario, the time complexity of the quantum method of the entire system is $O\left(log_2(T)log_2{}^3(\sqrt{d})\right)$, it depends on the computational complexity of the quantum existence testing function $log_2{}^3(\sqrt{d})$ and the logarithm search of the quantum algorithm $log_2(T)$, where *d* refers to the number of possible deployment scenarios, this quantum technique uses fewer steps than the other searching methods like heuristic and randomized algorithms [20][21], etc.

To calculate the maximum number of steps T which are necessary to run the logarithm search of the quantum algorithm for every bunch of new coming subtasks to the system, the real problem lies in calculating properly the value α at every task arrival (which means that the value of α changes for every new coming task), so in order to not confuse the reader we will denote this repeatedly computed value of α by α_{actual}. Finding the value of α_{actual} requires determining the minimum distance for any two different load distributions, mathematically expressed as $\alpha_{actual}(V_i, V_j) = \min_{\forall V_i,V_j} \left|(\boldsymbol{\sigma}^2_{V_i} - \boldsymbol{\sigma}^2_{V_j})\right|$. The computational complexity for finding α_{actual} is $O\left(\frac{d(d-1)}{2}\right)$, in the worst case, $d = c^{g_l}$, where g is the number of subtasks in the arrived task from type *l*, it is noticeable that computing the value of α_{actual} is computationally hard.

Instead of calculating α_{actual} at each arrival task, the alternative solution is to compute in advance the global non-zero minimum of α, denoted by α_{global}, before starting the operation of the system, such that $\alpha_{global} = \min_{\forall S^i_{ch}, S^j_{ch}} \alpha_{local}(S^i_{ch}, S^j_{ch})$, where α_{local} means minimization over all possible load configurations of changed unit sets S^i_{ch} and S^j_{ch} belonging to the distributions V_i and V_j. Because a certainly changed unit set fits many distributions, it is enough to define the previous formula with *i* and *j*. The expression of the value of α_{local} is influenced only by a load of changed computing units, as we will see the proof later in the next subsection E, while unchanged computing units have no effects on α_{local}.

Investigating all unit set pairs to calculate α_{global} we need to compute all α_{local} which are related to unit sets, assuming that the number of computing unit types is θ, taking advantage of the previous statement of α_{global}, we conclude that the number of possible distributions $d^{'}$ is less or equal than $\theta^{g_{max}}$ where g_{max} is the number of arrival subtasks, which means that the computational complexity at this stage is $O\left(\frac{d'(d'-1)}{2}\right)$, this computation complexity is significantly less than the computation complexity of α_{local}, the disadvantage of this minimization will create an increase in the maximum number of steps, but the quantum approach can handle the logarithm complexity of a large number of scale values because it will not increase significantly the complexity of the system.

The present subsection covers a comparison between the classical and the quantum approach. The question was how to reduce the computational complexity for the setup of the repeated changing value of α for any coming task, The alternative solution was to compute the global value α_{global} by exploiting the value of α_{local}. The next subsection addresses the formulation of α_{local} taking into consideration the resource distribution parameters.

E. Setting up the system parameter of the resource distribution management model with multi-task and multi-subtasks

As already mentioned, in order to fully exploit the potential of the quantum minimum searching algorithm, it is necessary to configure properly the parameters of the quantum method according to the characteristics of the resource allocation model. As aforementioned, the parameter *T* depends on $\Delta\sigma^2$, in the worst case, the value of $\sigma_{max}{}^2$ is 0.25 and the value of $\sigma_{min}{}^2$ will be always 0. Thus $\Delta\sigma^2 = \sigma_{max}{}^2$.

A more interesting approach addressed in this paper consist of providing concrete configuration of the quantum algorithm according to the mathematical formula given in (2), for the sake of the minimum distance between variances of two scenarios α_{local}, so, it is necessary to find a manageable expression of $(\boldsymbol{\sigma}^2_{V_i} - \boldsymbol{\sigma}^2_{V_j})$.

Let x_k, b_k and $\bar{b}$ be respectively the total of the kth unit load before receiving new subtask types, the relative load of the kth unit, and relative load average as it is shown respectively in (4), (5) and (6),

$$\boldsymbol{x_k} = \sum_{v=1}^{W} N^b_{kv}\,\Delta_v, \tag{4}$$

where N^b_{kv} refers to the number of subtask type v in the kth computing unit,

$$\boldsymbol{b_k} = \frac{\boldsymbol{x_k}}{\boldsymbol{s_k}}, \tag{5}$$

$$\overline{\boldsymbol{b}} = \frac{\mathbf{1}}{\boldsymbol{c}} \sum_{\boldsymbol{k}=\mathbf{1}}^{\boldsymbol{c}} \boldsymbol{b}_{\boldsymbol{k}}. \tag{6}$$

The problem was in formulating a general expression of the minimum distance between variances of two scenarios ($\boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_i} - \boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_j}$), i.e., for a given distribution scenario which fits to assigning only one subtask to a given unit. For this purpose, we considered that the load status of the computing units before and after the task deployment is given respectively by the V^b (7) and V^a matrices, their rows represent the computing units and the columns denote the subtasks type (8). In order to find the expression of the minimum variance between two different scenarios, the load of the new subtasks type should be taken into account, let's denote the set of computing units receiving the new subtasks by S_{ch} and the remaining set of computing units by S_{un}, such that $S_{ch} \cup S_{un} = S_T$. For the sake of a simple notation that describes all the possible deployment scenarios of the new subtasks, we use V_i matrix, its rows represent the computing units and the columns denote the subtask types as it is expressed in (8),

$$V^b = \begin{bmatrix} N^b_{11} & \cdots & \cdots & \cdots & \cdots & N^b_{1W} \\ \vdots & \ddots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \cdots & \vdots \\ \vdots & \cdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \cdots & \ddots & \ddots & \vdots \\ N^b_{c1} & \cdots & \cdots & \cdots & \cdots & N^b_{cW} \end{bmatrix}, \tag{7}$$

$$V_i = \begin{bmatrix} N^{V^i}_{11} & \cdots & \cdots & \cdots & \cdots & N^{V^i}_{1W} \\ \vdots & \ddots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \cdots & \vdots \\ \vdots & \cdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \cdots & \ddots & \ddots & \vdots \\ N^{V^i}_{c1} & \cdots & \cdots & \cdots & \cdots & N^{V^i}_{cW} \end{bmatrix}. \tag{8}$$

The relation between the load status of the computing units before and after the task deployment is $V^b + V_i = V^a$.
The relative load of the set of computing units receiving the new subtasks S_c is denoted by $\boldsymbol{b}^{\boldsymbol{V}^i}_{k \in S_{ch}}$ and the relative load of the remaining computing units $\boldsymbol{b}_{k \in S_{un}}$ is formulated as follows,

$$\begin{cases} b^{V^i}_{k \in S_{ch}} = b_{k \in S_{ch}} + \boldsymbol{V}_{\boldsymbol{i}}\Delta \\ b_{k \in S_{un}} = b_k \end{cases} \tag{9}$$

Taking into consideration the new deployment subtasks, the average of the relative load of the resource model is expressed by the formula (10), note that $\boldsymbol{P} = [s_1 \quad \cdots \quad s_W]$ and $\Delta = [\Delta_1 \quad \cdots \quad \Delta_W]^t$,

$$\overline{b_{V_\iota}} = \frac{1}{c} \sum_{k=1}^{c} b_k + \frac{1}{c} \boldsymbol{P}\boldsymbol{V}_{\boldsymbol{i}}\Delta. \tag{10}$$

The relative load of the variance of the new deployment scenario is given as follows,

$$\boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_i} = \frac{\mathbf{1}}{\boldsymbol{c}} \sum_{\boldsymbol{k}=\mathbf{1}}^{\boldsymbol{c}} \left(\overline{\boldsymbol{b}_{\boldsymbol{V}_\iota}} - \boldsymbol{b}^{\boldsymbol{V}_i}_{\boldsymbol{k}}\right)^{\mathbf{2}}. \tag{11}$$

In the end, we end up with the corresponding formula, which is expressed as follows,

$$\begin{aligned} &\boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_i} - \boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_j} \\ &= \frac{\mathbf{1}}{\boldsymbol{c}} \Bigg[\sum_{k \in S^{V^i}_{ch}} \left(\frac{\sum_v^W N^{V^i}_{kv}}{S_k} \right)^2 - \sum_{k \in S^{V^j}_{ch}} \left(\frac{\sum_v^W N^{V^j}_{kv}}{S_k} \right)^2 \\ &+ 2 \Bigg(\sum_{k \in S^{V^i}_{ch}} \left(\sum_{v=1}^{W} N^b_{kv} \right) \left(\frac{\sum_v^W N^{V^i}_{kv}}{{S_k}^2} \right) \\ &- \sum_{k \in S^{V^j}_{ch}} \left(\sum_{v=1}^{W} N^b_{kv} \right) \left(\frac{\sum_v^W N^{V^j}_{kv}}{{S_k}^2} \right) \Bigg) \Bigg] \end{aligned} \tag{12}$$

The value of α denotes the smallest distance between two different scenarios among all the possible scenarios in a database. α is illustrated in Fig.2

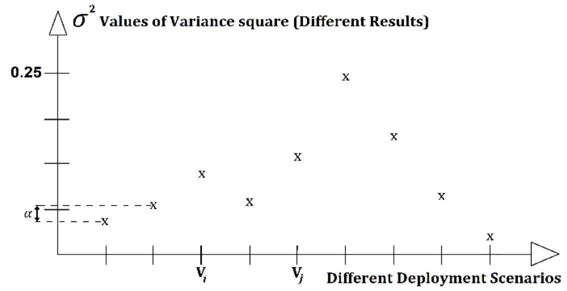


Fig.2: The horizontal axis presents all the possible deployment scenarios, while the vertical axis presents the borders of the variance square function (different results), each possible scenario corresponds to a variance value. Computing the value of α requires selecting the minimum distance between variances of two deployment scenarios V_i and V_j.

It is important to mention that for any distribution, the difference between variances of two different scenarios depends only on the set of the computing units S_{ch} that have been assigned a workload, not on all computing units.

Note that we considered that the number of incoming subtasks for both scenarios V^i and V^j are considered as fixed parameters for the system distribution, another important remark that we want to investigate is the non-zero α. Furthermore, it is clearly noticeable that $(\boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_i} - \boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_j})^2 \geq 0$, in order to find the minimum places, it is enough to investigate $(\boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_i} - \boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_j})^2 = 0$, this expression derives an important property of the minimum points (i.e., the variables N^b_{kv}, $v \in \{1, \ldots, W\}$ are linearly dependent), in compliance with this result we conclude that the minimum places of the function $(\boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_i} - \boldsymbol{\sigma}^{\mathbf{2}}_{\boldsymbol{V}_j})^2$ are situated in a hyperplane.

In the general case, to determine the desired α, it is needed to fulfill a certain number of restrictions determined by choosing the suitable range of the variables N^b_{kv}, $v \in \{1, \ldots, W\}$. The function $(\boldsymbol{\sigma}^2_{\boldsymbol{V_i}} - \boldsymbol{\sigma}^2_{\boldsymbol{V_j}})$ is continuous, however, from resource distribution management point of view the variables N^b_{kv}, $v \in \{1, \dots, W\}$ must be integers. The first step is to assign an integer number different than zero to all the variables except one value, which corresponds to the subtask $\boldsymbol{v} = \omega$, this value could be $N^b_{k\omega}$ / $k \in S^{V^i}_{ch}$ (or $k \in S^{V^j}_{ch}$), then computing the value of $N^b_{k\omega}$ / $k \in S^{V^i}_{ch}$ (or $k \in S^{V^j}_{ch}$), that corresponds to the remaining variables, here, at this stage, we have two options whether the value of $N^b_{k\omega}$ / $k \in S^{V^i}_{ch}$ (or $k \in S^{V^j}_{ch}$) is an integer or not.

1st case: if $N^b_{k\omega}$ / $R \in S^{V^i}_{ch}$ (or $R \in S^{V^j}_{ch}$) is an integer

The minimum distance between the variances of two scenarios α equals to zero and the values of the variables $N^b_{k\omega}$, $v \in \{1, \dots, W\}$ are not appropriate solutions for α, so, in this case, we can modify $N^b_{k\omega}$ / $k \in S^{V^i}_{ch}$ (or $R \in S^{V^j}_{ch}$), by increasing or decreasing $N^b_{k\omega}$, and assigning the value of $N^b_{k\omega} + \mathbf{1}$ or $N^b_{k\omega} - \mathbf{1}$ to $N^b_{k\omega}$ / $k \in S^{V^i}_{ch}$, (or assigning the value of $N^b_{k\omega} + \mathbf{1}$ or $N^b_{k\omega} - \mathbf{1}$ to $N^b_{k\omega}$ / $k \in S^{V^j}_{ch}$).

The most important thing is to choose only one assignment which results in the minimum value of $\boldsymbol{\alpha}$, let us investigate the value of $f = (\boldsymbol{\sigma}^2_{\boldsymbol{V_i}} - \boldsymbol{\sigma}^2_{\boldsymbol{V_j}})^2$ when we assign the value of $N^b_{k\omega} + \mathbf{1}$ or $N^b_{k\omega} - \mathbf{1}$ to $N^b_{k\omega}$, (or assigning the value of $N^b_{k\omega} + \mathbf{1}$ or $N^b_{k\omega} - \mathbf{1}$ **to** $N^b_{k\omega}$ / $k \in S^{V^j}_{ch}$), using the previous result which stated that $(\boldsymbol{\sigma}^2_{\boldsymbol{V_i}} - \boldsymbol{\sigma}^2_{\boldsymbol{V_j}})^2 \geq 0$, we get to the following result,

$$f_{N^b_{k\omega}+1} = f_{N^b_{k\omega}-1}. \qquad (13)$$

The above expression states that the cross-section of the hyperplane is symmetric on the minimum places of $(\sigma^2_{V_i} - \sigma^2_{V_j})^2$ in its dimensional space.

2nd case: if $N^b_{k\omega}$ / $k \in S^{V^i}_{ch}$ (or $R \in S^{V^j}_{ch}$) is not an integer

The function $(\boldsymbol{\sigma}^2_{\boldsymbol{V_i}} - \boldsymbol{\sigma}^2_{\boldsymbol{V_j}})$ has monotonous nature, consequently, the solution is to assign to $N^b_{k\omega}$ / $k \in S^{V^i}_{ch}$ (or $k \in S^{V^j}_{ch}$) the nearest integer, in this case, α does not equal to zero.

In this section, a mathematical framework was developed, which jointly determines the minimum distance between the variances of two scenarios α.

F. Simulation

We have developed a simulator to show the efficiency of the quantum method compared to the randomized method. We will apply the optimized strategy for two metrics independently, the resource distribution system contains three elements; they are defined as follows:

One task generator: The tasks are exponentially generated, all the tasks have the same memory and energy requirement.

Decision-Maker: This component is responsible for selecting the best placement of the task, so, we will have two simulations, the first one is to assign tasks randomly to the computing units and the second one is to use the quantum algorithm.

Resources: The system contains 30 computing units, there are 3 types of these computing units, the following table presents the initial energy consumption of every computing unit type and their theoretical capacity, i.e., the energy consumed from computing units when it is working and not serving any task. The service time of each task is fixed, it equals 5s and its memory requirement is also fixed, it equals 2 Kbit, the amount of the necessary power consumption to complete the serving of each task is 10 Watts.

Units type Characteristics	**Type 1**	**Type 2**	**Type 3**
Initial Energy	150	300	450
Theoretical capacity	10	20	30
Number of computing units	30	40	30

Table.1: the characteristics of computing units

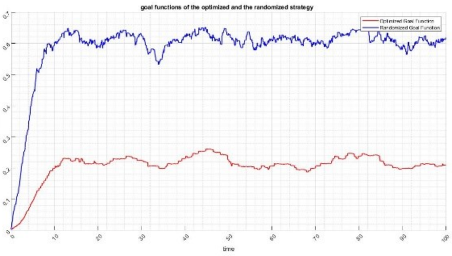


Fig.3: The overall energy consumption curves of the optimized (red line) and the randomized strategies (blue line), in case setting up the following parameters, the mean = 0.05.

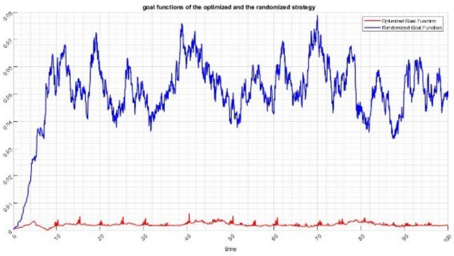


Fig.4: The variances curves of the optimized (red line) and the randomized strategies (blue line), in case setting up the following parameters, the mean = 0.05.

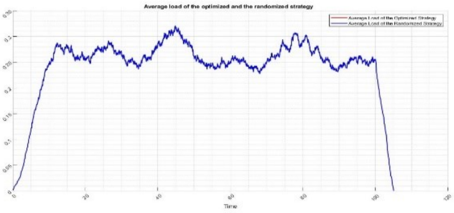


Fig.5: The average load curve of the optimized and the randomized strategies, in case setting up the following parameters, the mean = 0.05.

According to *Fig.3* and *Fig.4*, the curves representing the total power consumption of the randomized algorithm are larger than the optimized method. As a conclusion, the optimized strategy has better results in finding the minimum overall power consumption and the best uniformity distribution than the randomized method during the whole simulation process.

III. Conclusion

This work addresses the problem of distributing task portions, i.e., subtasks, among different resource computing as a means to alleviate the computational complexity of selecting the optimum distribution scenarios, the main objective was to conserve the uniformity distribution load. The quantum minimum searching was our best choice for achieving optimal deployment results, which is dramatically influences in reducing the computational complexity of the system.

Also, we handled a general case that handles multi-optimization metrics, as well as testing the proposed quantum approach by the simulation environment.

In the future work, the quantum algorithm will treat a model which handle more general case, multi-optimization metrics, as well as testing the proposed quantum approach by simulation environment, at the same time increasing the complexity problem by defining its nonlinear combination function, furthermore, we will try to implement constraints to our quantum algorithm.

References

[1] S. Imre, F. Balázs. Quantum Computing and Communications – An Engineering Approach. John Wiley, England, 2005, 283 pp. doi: 10.1002/9780470869048.

[2] S. Imre, L. Gyongyosi. Advanced Quantum Communications: An Engineering Approach", 2012, doi: 10.1002/9781118337462.

[3] Lov.K. Grover: "Quantum Computers Can Search Arbitrarily Large Databases by a Single Query" Vol 79, No 23, 1997. doi: https://doi.org/10.1103/PhysRevLett.79.4709

[4] Lov.K. Grover: "A fast quantum mechanical algorithm for database search", InProc. STOC '96 Proc. of the Twenty-eighth annual ACM symposium on Theory of Computing, Philadelphia, Pennsylvania, USA, 1996, pp. 212-2019, doi: 10.1145/237814.237866.

[5] Buttazzo G.C., Lipari G., Caccamo M., L. Abeni:" Elastic scheduling for flexible workload management". IEEE Transactions on Computers, Vol: 51, 2002. doi: 10.1109/12.990127.

[6] Marau R., Lakshmanan K., Pedreiras P., Almeida L., Rajkumar R.:" Efficient Elastic Resource Management for Dynamic Embedded Systems", 2011 IEEE 10th International Conference on Trust, Security, and Privacy in Computing and Communications, 2011. doi: 10.1109/TrustCom.2011.135.

[7] Bini E., Natale M. Di: "Optimal task rate selection in fixed priority systems", 26th IEEE International Real-Time Systems Symposium (RTSS'05), 5-8 December. 2005, doi: 10.1109/RTSS.2005.32.

[8] Buttazzo G.C., Lipari G., Abeni L.: "Elastic task model for adaptive rate control", InProc. 19th IEEE Real-Time Systems Symposium (Cat. No.98CB36279), 1998. doi: 10.1109/REAL.1998.739754.

[9] Yu D.; Zhou Z., Wang Y.: "Crowdsourcing Software Task Assignment Method for Collaborative Development", IEEE Access, Vol 7, pp. 35743 – 35754. doi: 10.1109/ACCESS.2019.2905054.

[10] Ismail S.; Sun L.: "Decentralized Hungarian-based approach for fast and scalable task allocation". International Conference on Unmanned Aircraft Systems (ICUAS), Miami, FL, USA, 2017, doi: 10.1109/ICUAS.2017.7991447

[11] Bistritz I., Leshem A.: "Efficient and asymptotically optimal resource block allocation", IEEE Wireless Communications and Networking Conference (WCNC), 2018, doi: 10.1109/WCNC.2018.8376960.

[12] Le, D. D, Molnár, M. and Palaysi, J., " Multicast Routing in WDM Networks without Splitters,", Infocommunication Journal, vol. 5, no. 2, pp. 1–10, June. 2013. doi: 10.1109/MCOM.2014.6852098

[13] Szabo, M., Hajay, D. and Szalayz, M., " Cost-Efficient Resource Allocation Method for Heterogeneous," Infocommunication Journal, vol 10, no 1, March 2018.

[14] S. El Gaily, S. Imre. EVALUATION OF RESOURCE OPTIMIZATION BASED ON QUANTUM SEARCH. Hungarian Journal of industry and chemistry. 2019. doi: 10.33927/hjic-2019-03.

[15] El Gaily S., Imre S., "Derivation of Parameters of Quantum Optimization in Resource Distribution Management". 42nd International Conference on Telecommunications and Signal Processing. doi: 10.1109/TSP.2019.8769092.

[16] S. Imre: „Extreme Value Searching in Unsorted Databases Based on Quantum Computing". International Journal of Quantum Information. World Scientific 2005. Vol. 3. No. 1, pp. 171-176. doi: 10.1142/S0219749905000700.

[17] Imre S: Quantum Communications – Explained for Communication Engineers IEEE COMMUNICATIONS MAGAZINE 51: 8 pp. 28-35., 8 p. (2013). doi: 10.1109/MCOM.2013.6576335

[18] Donald K. (1998). Sorting and searching. The Art of Computer Programming. 3 (2nd ed.). Reading, MA: Addison-Wesley Professional. doi: 10.1090/s0002-9904-1973-13173-8.

[19] Imre S., 'Quantum Existence Testing and Its Application for Finding Extreme Values in Unsorted Databases'. IEEE Transactions on Computer, Vol:56, no. 5, 2007. doi: 10.1109/TC.2007.1032

[20] Ariffin N., Zin M., Norul S., Sheikh H., Faridatul N., Zainal A.: A Comparison of Exhaustive, Heuristic and Genetic Algorithm for Travelling Salesman Problem in PROLOG, International Journal on Advanced Science Engineering Information Technology. Vol 2, no.6, 2012. doi: 10.18517/ijaseit.2.6.244.

[21] Hooker J.N.: Unifying Local and Exhaustive Search, Carnegie Mellon University, 2005. doi: 10.1111/itor.12020

Sara El Gaily, a Ph.D. student at the Department of Networked Systems and services at the Budapest University of Technology and Economics (BME). She obtained her Master's degree in electrical systems and renewable energies at Hassan II University in 2017 in Casablanca, Morocco. She obtained her bachelor's degree in computer and industrial electronics in 2014. Her research interests are quantum computing and resource distribution management.

Sándor Imre [M'93] professor and Head of the Department of Networked Systems and services at the Budapest University of Technology and Economics (BME). He obtained dr. univ. degree in probability theory and statistics in 1996, a Ph.D. degree in 1999 and DSc degree from the Hungarian Academy of Sciences in 2007. He was elected the corresponding member of HAS in 2019. He acts as supervisor in the High-Speed Networks Laboratory since 1999. He is a member of the Doctoral Council of HAS. He was invited to join the eMobile Innovation Center of BME as an R&D director in 2005. His research interests include mobile and wireless systems, quantum computing and communications. Especially he has contributions to different wireless access technologies, mobility protocols and their game-theoretical approaches, reconfigurable systems, quantum computing-based algorithms, and protocols.

Call for Papers

Follow us on twitter Find us on Facebook /tspconf

IEEE Conference Record: #49548

Mark Your Calendar

tsp 2020

July 7-9, 2020

region 8 IEEE IEEE Italy Section IEEE IEEE Czechoslovakia Section

2020
43rd International Conference on Telecommunications and Signal Processing (TSP)

Milan, Italy

SUBMIT SPECIAL SESSION AND WORKSHOPS:
February 1, 2020
FULL PAPER SUBMISSIONS:
February 15, 2020

PARTNERS & ORGANIZERS

PROCEEDINGS INDEXED BY

The *2020 43rd International Conference on Telecommunications and Signal Processing (TSP)* will be held during *July 7-9, 2020* in *Milan, Italy*. In cooperation with the ***IEEE Region 8 (Europe, Middle East and Africa)***, ***IEEE Italy Section***, ***Italy Section SP Chapter***, ***Italy Section VT/COM Joint Chapter***, ***IEEE Czechoslovakia Section***, ***IEEE Czechoslovakia Section SP/CAS/COM Joint Chapter***, and ***Scientific Association for Infocommunications***, a ***Sister Society of the IEEE*** and the ***IEEE Communications Society***, the TSP 2020 is organized by eighteen universities for academics, researchers, and developers and it serves as a premier annual international forum to promote the exchange of the latest advances in telecommunication technology and signal processing. We look forward to your innovative contributions in any of the following areas:

Telecommunications:

- Information Systems
- Network Services
- Network Technologies
- Telecommunication Systems
- Modelling, Simulation and Measurement

Signal Processing:

- Analog Signal Processing
- Audio, Speech and Language Processing
- Biomedical Signal Processing
- Digital Signal Processing
- Image and Video Signal Processing

PROCEEDINGS

All accepted papers will be published in the TSP 2020 Conference Proceedings issued online. The Proceedings with presented papers will be submitted for indexing in *IEEE Xplore® Digital Library* - ***IEEE Conference Record #49548***, *Conference Proceedings Citation Index (CPCI) of Thomson Reuters*, *SCOPUS*, *DBLP*, and *Google Scholar* databases. *Selected papers will be published in* ***Special Issues of journals*** *Applied Sciences* ***(ISSN 2076-3417; Web of Science Impact Factor: 2.217)*** *and* *International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems (IJATES)* ***(ISSN: 1805-5443).***

IMPORTANT DATES

Paper Submission: *February 15, 2020*

Notification of Acceptance: *April 15, 2020*

Authors' Registration: *May 20, 2020*

CONTACTS

E-mail: tsp@feec.vutbr.cz

Web: http://tsp.vutbr.cz/

FOLLOW US

FACEBOOK /tspconf

TWITTER /tspconf

ieee-sensors2020.org

IEEE SENSORS 2020

WTC, Rotterdam, The Netherlands October 25-28, 2020

CALL FOR PAPERS

Organizers

General Co-Chair
Paddy French
TU Delft

General Co-Chair
Troy Nagle
NC State University

Technical Program Co-Chair
Gijs Krijnen
University of Twente

Technical Program Co-Chair
Rolland Vida
Budapest University of Technology and Economics

Important Dates

May 6, 2020
Proposals for Tutorials

May 20, 2020
Proposal for Focused Sessions

June 18, 2020
4 pages (max)
3 pages of text (max)
+ 1 page of references (max)

August 10, 2020
Notification of Paper Acceptance

August 31, 2020
Submission of Final Papers

Please visit
ieee-sensors2020.org

IEEE SENSORS 2020 is intended to provide a forum for research scientists, engineers, and practitioners throughout the world to present their latest research findings, ideas, and applications in the area of sensors and sensing technology.

IEEE SENSORS 2020 will include keynote addresses and invited presentations by eminent scientists and engineers. The conference solicits original state-of-the-art contributions as well as review papers.

Topics for IEEE SENSORS 2020 include

- Sensor Phenomenology, Modeling and Evaluation
- Sensor Materials, Processing and Fabrication (including Printing)
- Chemical, Electrochemical and Gas Sensors
- Microfluidics and Biosensors
- Optical Sensors
- Physical Sensors - Temperature, Mechanical, Magnetic and Others
- Acoustic and Ultrasonic Sensors
- Sensor Packaging (including on Flexible Materials)
- Emerging Sensor Applications
- Sensor Networks (including IoT and related areas)
- Sensor Systems: Signals, Processing and Interfaces
- Actuators and Sensor Power Systems
- Sensors in Industrial Practices

  This track is for traditional papers where the focus is on the industrial applications of different sensors. This track is only for industry, i.e. first author must be from industry.
- Live Demonstration of Sensors and Sensing Technologies

Focused Sessions

IEEE SENSORS 2019 will have focused sessions on emerging sensor-related topics. Details related to the Call For Focused Sessions is on the conference website.

Publication of Papers

Presented papers will be included in the Proceedings of IEEE SENSORS 2020 and in IEEE Xplore pending author requirements being met. Authors may submit extended versions of their paper to the IEEE Sensors Journal.

Exhibition Opportunities

The Conference exhibit area will provide your company or organization with the opportunity to inform and display your latest products, services, equipment, books, journals, and publications to attendees from around the world.

For further information,
contact Rachel Brockhoff,
rbrockhoff@conferencecatalysts.com

Industry Day

A special track designed to encourage industry participation will include industry showcase, industry networking, and an industry panel discussion. Special flexible one-day registration will be available to facilitate industry participation.

Visit the website for the most up to date information relating to abstract submission, tutorials, and special sessions information and deadlines.

Special Issue in the IEEE Sensors Journal

If your paper is accepted for publication at this conference, you may receive an invitation to submit an extended manuscript to be considered for publication in the IEEE Sensors Journal. Best papers presented at the conference will be invited to participate via the call for papers for this special issue of the journal.

IEEE Advancing Technology for Humanity

IEEE Sensors Council

Guidelines for our Authors

Format of the manuscripts

Original manuscripts and final versions of papers should be submitted in IEEE format according to the formatting instructions available on

https://journals.ieeeauthorcenter.ieee.org/
Then click: "IEEE Author Tools for Journals"
- "Article Templates"
- "Templates for Transactions".

Length of the manuscripts

The length of papers in the aforementioned format should be 6-8 journal pages.
Wherever appropriate, include 1-2 figures or tables per journal page.

Paper structure

Papers should follow the standard structure, consisting of *Introduction* (the part of paper numbered by "1"), and *Conclusion* (the last numbered part) and several *Sections* in between.
The Introduction should introduce the topic, tell why the subject of the paper is important, summarize the state of the art with references to existing works and underline the main innovative results of the paper. The Introduction should conclude with outlining the structure of the paper.

Accompanying parts

Papers should be accompanied by an *Abstract* and a few *index terms (Keywords)*. For the final version of accepted papers, please send the short cvs and *photos* of the authors as well.

Authors

In the title of the paper, authors are listed in the order given in the submitted manuscript. Their full affiliations and e-mail addresses will be given in a footnote on the first page as shown in the template. No degrees or other titles of the authors are given. Memberships of IEEE, HTE and other professional societies will be indicated so please supply this information. When submitting the manuscript, one of the authors should be indicated as corresponding author providing his/her postal address, fax number and telephone number for eventual correspondence and communication with the Editorial Board.

References

References should be listed at the end of the paper in the IEEE format, see below:

a) Last name of author or authors and first name or initials, or name of organization
b) Title of article in quotation marks
c) Title of periodical in full and set in italics
d) Volume, number, and, if available, part
e) First and last pages of article
f) Date of issue
g) Document Object Identifier (DOI)

[11] Boggs, S.A. and Fujimoto, N., "Techniques and instrumentation for measurement of transients in gas-insulated switchgear," IEEE Transactions on Electrical Installation, vol. ET-19, no. 2, pp.87–92, April 1984. DOI: 10.1109/TEI.1984.298778

Format of a book reference:

[26] Peck, R.B., Hanson, W.E., and Thornburn, T.H., Foundation Engineering, 2nd ed. New York: McGraw-Hill, 1972, pp.230–292.

All references should be referred by the corresponding numbers in the text.

Figures

Figures should be black-and-white, clear, and drawn by the authors. Do not use figures or pictures downloaded from the Internet. Figures and pictures should be submitted also as separate files. Captions are obligatory. Within the text, references should be made by figure numbers, e.g. "see Fig. 2."
When using figures from other printed materials, exact references and note on copyright should be included. Obtaining the copyright is the responsibility of authors.

Contact address

Authors are requested to submit their papers electronically via the EasyChair system. The link for submission can be found on the journal's website:

www.infocommunications.hu/for-our-authors

If you have any question about the journal or the submission process, please do not hesitate to contact us via e-mail:

Pál Varga – Editor-in-Chief:
pvarga@tmit.bme.hu

Rolland Vida – Associate Editor-in-Chief:
vida@tmit.bme.hu

CALL FOR PAPERS

On behalf of the European Association for Signal Processing (EURASIP), it is a great pleasure of the organizing committee to invite you to the 28th European Signal Processing Conference, EUSIPCO 2020, to be held in Amsterdam, The Netherlands.

EUSIPCO is the flagship conference of EURASIP and offers a comprehensive technical program addressing all the latest developments in research and technology for signal processing. EUSIPCO 2020 will feature worldclass speakers, oral and poster sessions, plenaries, exhibitions, demonstrations, tutorials, and satellite workshops, and is expected to attract many leading academic researchers and people from industry from all over the world.

TECHNICAL SCOPE

We invite the submission of original, unpublished technical papers on topics including but not limited to:

- Audio and acoustic signal processing
- Speech and language processing
- Image and video processing
- Multimedia signal processing
- Signal processing theory and methods
- Sensor array and multichannel signal processing
- Signal processing for communications
- Radar and sonar signal processing
- Signal processing over graphs and networks
- Nonlinear signal processing
- Statistical signal processing
- Compressed sensing and sparse modelling
- Optimization methods
- Machine learning
- Bio-medical image and signal processing
- Signal processing for computer vision and robotics
- Computational imaging / spectral imaging
- Information forensics and security
- Signal processing for power systems
- Signal processing for education
- Bioinformatics and genomics
- Signal processing for big data
- Signal processing for the internet of things
- Design/implementation of signal processing systems

Accepted papers will be included in IEEE Xplore©. EURASIP enforces a “no-show” policy. Procedures to submit papers, proposals for special sessions, tutorials and satellite workshops can be found on the website.

EURASIP

TU Delft

IEEE Signal Processing Society
TECHNICAL CO-SPONSOR

SCAN ME

COMMITTEES

GENERAL CO-CHAIRS
Richard Heusdens, TU Delft, The Netherlands
Cédric Richard, University of Nice Sophia-Antipolis, France

TECHNICAL PROGRAM CO-CHAIRS
Alle-Jan van der Veen, TU Delft, The Netherlands
Pina Marzilano, EPFL, Switzerland
Toon van Waterschoot, KU Leuven, Belgium

FINANCIAL CHAIR
Richard C. Hendriks, TU Delft, The Netherlands

TUTORIALS CO-CHAIRS
Béatrice Pesquet, Thales LAS, France
Sundeep Chepuri, Indian Institute of Science, India

SPECIAL SESSIONS CO-CHAIRS
Helmut Bölcskei, ETH Zürich, Switzerland
Tulay Adali, UMBC, USA
Sharon Gannot, Bar-Ilan University, Israel

PLENARY CO-CHAIRS
Geert Leus, TU Delft, The Netherlands
Mario A.T. Figueiredo, University of Lisboa, Portugal

SPONSOR CHAIR
Ton Kalker, Xperi Corporation, USA

AWARDS CO-CHAIRS
Mads G. Christensen, Aalborg University, Denmark
Timo Gerkmann, Hamburg University, Germany

PUBLICITY CHAIR
Elvin Isufi, TU Delft, The Netherlands

STUDENT ACTIVITIES CO-CHAIRs
Andreas Jakobsson, Lund University, Sweden
Jesper Rindom Jensen, Aalborg University, Denmark

SATELLITE WORKSHOPS CO-CHAIRS
Zekeriya Erkin, TU Delft, The Netherlands
Odette Scharenborg, TU Delft, The Netherlands

PUBLICATIONS CO-CHAIRS
Antonio Marques, King Juan Carlos University, Spain
Borbala Hunyadi, TU Delft, The Netherlands

PCO
Nicole Fontein, BleuBoxEvents, The Netherlands

IMPORTANT DATES

Special Session Proposals— December 6, 2019
Tutorial Proposals — January 17, 2020
Satellite Workshop Proposals — January 24, 2020
Full Paper Submission — February 21, 2020
Notification of Acceptance — May 29, 2020
Final Manuscript Submission — June 12, 2020

HTTP://2020.EUSIPCO.ORG

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

hte

FOUNDED IN 1949

SCIENTIFIC
ASSOCIATION FOR
INFOCOMMUNICATIONS
HUNGARY

Who we are

Founded in 1949, the Scientific Association for Infocommunications (formerly known as Scientific Society for Telecommunications) is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals working in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

Besides its 1000 individual members, the Scientific Association for Infocommunications (in Hungarian: Hírközlési és Informatikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society.

What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

Contact information

President: **GÁBOR MAGYAR, PhD** • *elnok@hte.hu*

Secretary-General: **ERZSÉBET BÁNKUTI** • *bankutie@ahrt.hu*

Operations Director: **PÉTER NAGY** • *nagy.peter@hte.hu*

International Affairs: **ROLLAND VIDA, PhD** • *vida@tmit.bme.hu*

Address: H-1051 Budapest, Bajcsy-Zsilinszky str. 12, HUNGARY, Room: 502
Phone: +36 1 353 1027
E-mail: *info@hte.hu*, Web: *www.hte.hu*