$t_1(x) = t_{-1}(x) - q_1(x)t_0(x) = 2 + 2x + x^2$

Output: Primitive polynomial $q(x) = t_1(x) = 2 + 2x + x^2$, initial state of the register $h(x) = s_1(x) = 1$, linear complexity $\lambda = \deg(q(x)) = 2$

As one can see we found that the linear complexity $\lambda = 2$ is equal the expected value and the algorithm finished at its first step. The primitive polynomial that was found with the algorithm is equivalent to the one that was used to generate the sequence.

Tests with different sequences, i.e. different Galois Fields, primitive polynomials and lengths of pLFSRs, were made, all reconfirming the results.

As the aim of this paper is to investigate the non-linearity that is introduced to the output sequence of the pGSSG by the self-shrinking rule of the *p*-ary register's output, sets of such output sequences will be investigated with algorithm 1 in order to calculate their linear complexity.

The equation (7) sets the boundaries of the linear complexity of a *p*-ary sequence, i.e. $0 \le \lambda_S \le T$. The value $\lambda_S = 0$ is assumed to be the linear complexity of the zero sequence $S = (0, 0, \ldots, 0)$. The rule resulting from the Berlekamp-Massey algorithm says that for determining the linear complexity λ of a sequence S, $2.\lambda$ consecutive elements of S are needed. Thus, the algorithm will be tested as its input will be fed with sequenced with length $2n$, that corresponds to the maximal possible linear complexity λ.

Table 1. *Results for linear complexity of output pGSSG sequences*

GF(p^n)	Count	Period	Linear complexity	
			Min.	Max.
GF(3^2)	2	6	4	5
GF(3^3)	4	18	15	16
GF(3^4)	8	54	48	51
GF(3^5)	22	162	152	158
GF(3^6)	48	486	478	481
GF(5^2)	4	20	18	19
GF(5^3)	20	100	76	98
GF(5^4)	48	500	492	497
GF(7^2)	8	42	35	41
GF(7^3)	36	294	288	292
GF(7^4)	160	2058	2050	2055

More than 360 tests are conducted using the algorithm implementation for evaluation the linear complexity of the output sequences generated by *pGSSG* in different fields GF(p^n). Table 1 shows a summary of the test results including the number of possible output pGSSG sequences (Count), their period T, and the minimum and maximum results for the calculated linear complexities.

As it was shown in section 2, the linear complexity of the underlying pLSFR in pGSSG is equal to its length L. We can notice in Table 1 that the expected maximum value of the linear complexity of the non-linear pGSSG generator is

$$\lambda_{\max} = T - (L - 1) = (p - 1)p^{L-1} - (L - 1). \quad (51)$$

Therefore, the proposed non-linear method for self-shrinking in the pGSSG increases the linear complexity with the following coefficient

$$K_\lambda = \frac{(p - 1)p^{L-1} - (L - 1)}{L}. \quad (52)$$

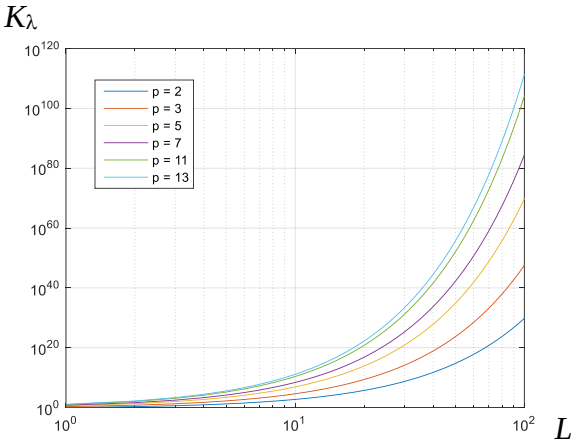

Fig. 2. *Coefficient for increasing the linear complexity for pGSSG*

As it is shown in [13] when $p = 2$ the generator *pGSSG* is converted into the classic *SSG* [10]. Having that in mind figure 2 represents the linear complexity enhancement dependency of *pGSSG* over *SSG* (shown in blue color) on the length L of the used *pLFSR* with different prime p.

5 Conclusion

In this paper we have investigated the linear complexity of *p*-ary pseudo random sequences produced by *p*-ary generalized self-shrinking generator. We have mathematically justified that the extended Euclidian algorithm can be applied to find the linear complexity of *p*-ary pseudorandom sequences. Using this algorithm the linear complexity of output sequence of the pGSSG has been calculated. The more than 360 tests that have been performed show that the linear complexity of pGSSG output sequences is close to the maximum theoretical.

Acknowledgements

This paper is a result of a project supported by the National Science Fund, Ministry of Education and Science, Bulgaria via FINANCIAL SUPPORT FOR PROJECT OF JUNIOR RESEARCHERS – 2016 [Grant Number DM07/5 – 15.12.2016]

References:

[1] Berlekamp, E. R. (1968). Algebraic coding theory. McGraw-Hill Book Co., New York-Toronto, Ont.-London.

[2] Buchanan, W. J., Li, S., & Asif, R. (2017). Lightweight cryptography methods. Journal of Cyber Security Technology, 1(3-4), 187-201.

[3] Edemskiy, V., & Minin, A. (2016). About the linear complexity of the almost perfect sequences. International Journal of Communications, 1, 223-226.

[4] ISO/IEC 29192-3:2012. International standard for lightweight cryptographic methods, ISO/IEC, 2012.

[5] Manifavas, C., Hatzivasilis, G., Fysarakis, K., & Papaefstathiou, Y. (2016). A survey of lightweight stream ciphers for embedded systems. Security and Communication Networks, 9(10), 1226-1246.

[6] Massey, J. L. (1969). Shift-register synthesis and BCH decoding. IEEE Trans. Information Theory, IT-15, 122–127.

[7] Massey, James L., and Shirlei Serconek. „Linear complexity of periodic sequences: a general theory.“ In *Advances in cryptology—CRYPTO'96*, pp. 358-371. Springer Berlin Heidelberg, 1996.

[8] McKay, K. A., Bassham, L., Turan, M. S., & Mouha, N. (2017). NISTIR 8114 report on lightweight cryptography. National Institute of Standards and Technology (NIST), Gaithersburg.

[9] Meidl, W., Winterhof, A. (2013). Linear complexity of sequences and multisequences, *Handbook of finite fields*. Chapter 10.4, pp. 318-330. Chapman and Hall/CRC.

[10] Meier, W., Staffelbach, O.: The self-shrinking generator. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 205–214. Springer, Heidelberg (1995).

[11] Randrianarisoa, T. (2018). Coding Theory using Linear Complexity of Finite Sequences. arXiv preprint arXiv:1802.10034.

[12] Rueppel R.A. (1986) Linear Complexity and Random Sequences. In: Pichler F. (eds) Advances in Cryptology — EUROCRYPT' 85. EUROCRYPT 1985. Lecture Notes in Computer Science, vol 219. Springer, Berlin, Heidelberg

[13] Tasheva, A., Tasheva, Zh., Milev, A., Generalization of the Self-Shrinking Generator in the Galois Field GF(p^n), Advances in Artificial Intelligence, vol. 2011, Article ID 464971, 10 pages, 2011.

[14] Tasheva, A., Savova-Tasheva, Zh., Petrov, B., Stoykov, K., Determining the Feedback Multipliers in a p-ary Linear Feedback Shift Registers, WSEAS Transactions on Systems and Control, Volume 13, 2018, Art. #45, pp. 420-424

[15] Venkateswarlu, A. (2007). Studies on error linear complexity measures for multisequences (Doctoral dissertation).

[16] Wang, Q., Jiang, Y., & Lin, D. (2015). Linear complexity of binary generalized cyclotomic sequences over GF (q). Journal of Complexity, 31(5), 731-740.

[17] Winterhof, A., Linear complexity and related complexity measures, in Selected Topics in Information and Coding Theory, vol. 7. Hackensack, NJ, USA: World Scientific, 2010, pp. 3–40.