

VISUALIZATION OF ALGORITHMS FOR GENERATING PSEUDORANDOM NUMBERS

Štěpán Zálešák¹, Radim Farana¹

¹Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic

DOI: https://doi.org/10.11118/ecos2026_044

ABSTRACT

This article deals with the design and implementation of an interactive web application intended for the visualization and comparison of two basic algorithms for generating pseudorandom numbers (PRNGs) — the Linear Congruential Generator (LCG) and the Linear Feedback Shift Register (LFSR). The work responds to the problem of the abstractness of teaching these algorithms and offers a tool that allows users to experiment with parameters, monitor the internal states of the generators, and analyze the quality of the resulting sequences using integrated analytical tools such as a histogram of value distribution, period detection, and a scatter plot to reveal the lattice structure.

Keywords: pseudorandom number, generator, algorithm visualization, web application

JEL Code: C63

1 INTRODUCTION

The generation of pseudorandom numbers is a critical part of many areas of computer science, including cryptography, computer simulations, the gaming industry, and machine learning. Unlike true random number generators (TRNGs), which utilize physical processes, pseudorandom generators (PRNGs) are deterministic algorithms. Based on an initial value (seed), they produce sequences that exhibit statistical properties of randomness, with their main advantages being reproducibility and high computational efficiency.

1.1 Linear Congruential Generator (LCG)

LCG is one of the oldest PRNG algorithms (Knuth 1998). It is defined by the recurrence relation:

$$x_i = (ax_{i-1} + c)(\text{mod } m) \quad (1)$$



This article is licensed under CC BY-SA 4.0 Attribution-ShareAlike 4.0 International license. <https://creativecommons.org/licenses/by-sa/4.0/deed.en>

where m is the modulus, a is the multiplier, and c is the increment, the choice of which affects the period and statistical quality of the generated sequence. Despite its simplicity, LCG suffers from inherent weaknesses, such as a lattice structure (n -tuples of values lying on hyperplanes) and a short period of lower bits for modules in the form of a power of two. From the literature, we know good LCG parameters for achieving a large period value and good statistical properties of the generated sequence of numbers (L'Ecuyer 1999).

1.2 Linear Feedback Shift Register (LFSR)

LFSR is based on bitwise operations. It consists of a register and a feedback function (usually XOR of selected bits — so-called taps). There are two basic variants: Fibonacci and Galois. The maximum period of an LFSR of length n is $2^n - 1$, which is achieved by using a primitive polynomial. LFSR is efficient, especially in hardware applications, but due to its linearity, it is not suitable for cryptography (Singer 2022).

2 APPLICATION DESCRIPTION

The resulting application is implemented using standard web technologies (HTML5, CSS3, JavaScript). The user interface is localized into Czech and English (Zálešák 2026).

2.1 Visualization Implementation

The LCG visualization displays the calculation process directly in the context of the mathematical formula, where variable values change dynamically in each step, respecting the standards for algorithm visualization (Naps *et al.* 2002). For LFSR, emphasis is placed on displaying the bit shift in the register and the feedback calculation, while the application supports both Fibonacci and Galois variants. The user has the option to step through the algorithm in both directions, even several times in a row, and thus correctly understand the behavior of both algorithms, see Fig. 1. After the user reaches the end of the algorithm, a statistical evaluation of the generated series of numbers is displayed, see Fig. 2 and Fig. 3.

2.2 Analytical Tools

To evaluate the quality of generators, the application contains:

- **Histogram:** Graphically represents the uniformity of the distribution of generated values, see Fig. 2. +
- **Period Detection:** Empirically determines the length of the period by generating values until the initial state is encountered, see Fig. 2.
- **Scatter Plot:** Displays pairs of consecutive numbers (x_i, x_{i+1}). This tool is key for visually detecting the lattice structure of LCG with an inappropriate choice of multiplier, see Fig. 3.

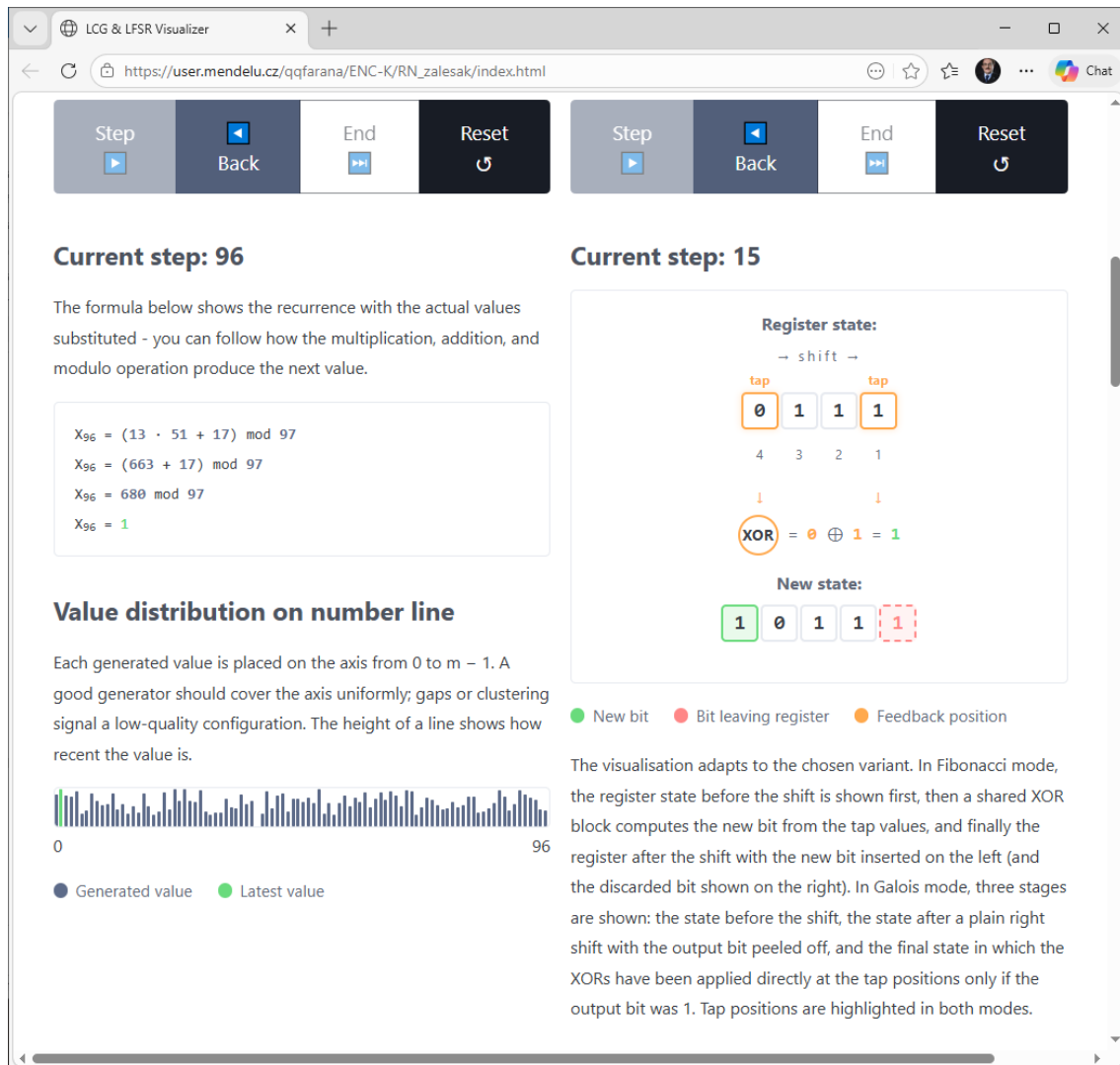


Fig. 1: Random numbers generation

3 TESTING RESULTS

The functionality and didactic benefit of the application were verified by user testing on a sample of computer science students. Testing took the form of a controlled scenario, after which respondents filled out a questionnaire focused on usability and clarity.

Tab. 1 Testing results

Evaluation Criterion	Average Rating (1-5, 5 is best)
Clarity of LCG visualization	4.7
Clarity of LFSR visualization	4.5
Contribution of analytical tools	4.8
Overall intuitiveness of the interface	4.6

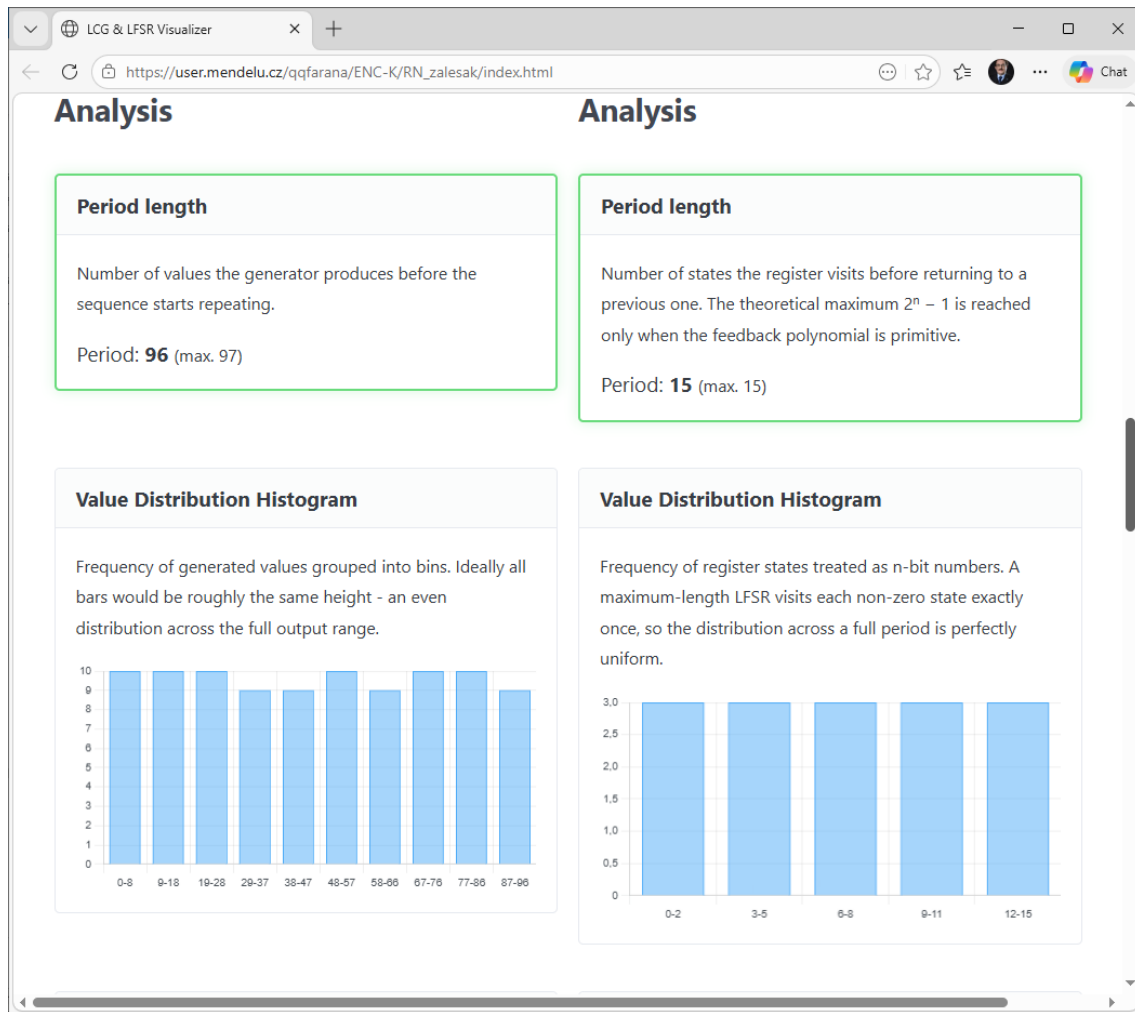


Fig. 2: Obtained period and histogram comparison

Users rated the period detection tool and the scatter plot most highly, as they allowed them to immediately see the influence of „bad“ parameters on the output quality. Identified problems included a higher difficulty in understanding the Galois variant of LFSR for beginners, which was subsequently addressed by adding context-sensitive help.

4 EVALUATION

The proposed application successfully demonstrates the differences in the principles and properties of LCG and LFSR. While LCG excels in integer arithmetic, LFSR offers efficient bitwise operations. Thanks to interactive comparison, users can observe that to achieve comparable quality and period, both algorithms require a specific, mathematically grounded choice of parameters. The application proved to be an effective didactic tool that supports the active involvement of students in the learning process.

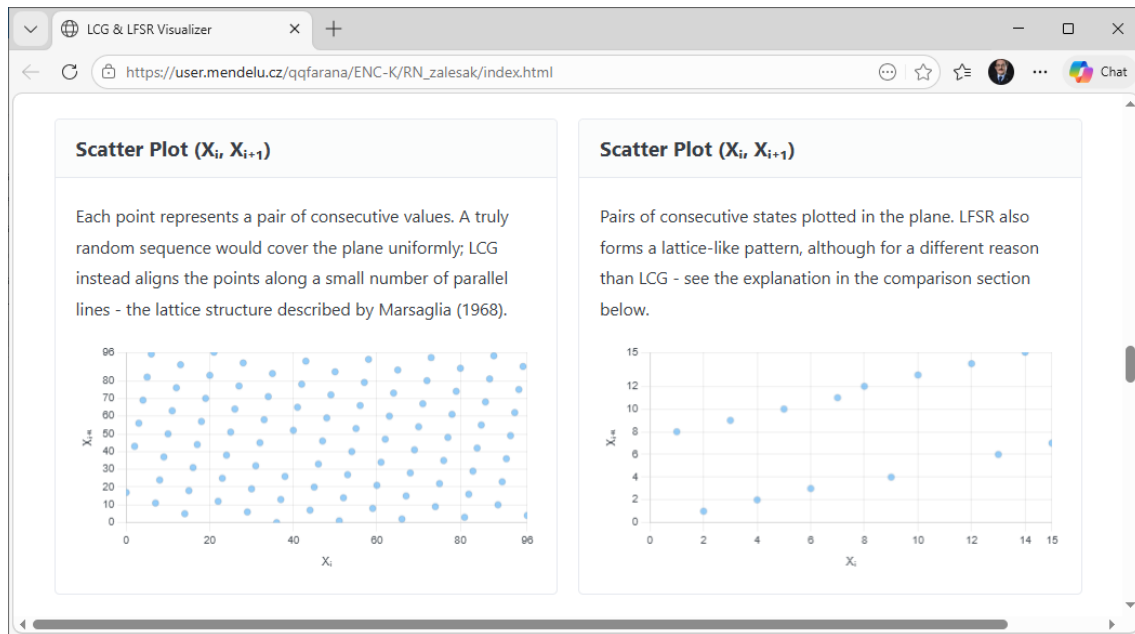


Fig. 3: Scatter plot for both generators

5 CONCLUSIONS

The created web application fulfills the set goals and provides a comprehensive environment for studying PRNG algorithms. The combination of step-by-step visualization and real-time statistical analysis represents an innovative approach to teaching this subject. Future expansions may include the implementation of other modern algorithms, such as Mersenne Twister, or the integration of more advanced statistical batteries of randomness tests. Thanks to its clarity, it can also help people who work with them and are not familiar with the algorithms used to understand the context associated with generating pseudorandom numbers.

Acknowledgements

This paper was supported by the project CZ.02.1.01/0.0/0.0/16_017/0002334 Research Infrastructure for Young Scientists, which is co-financed from the Operational Programme Research, Development and Education.

REFERENCES

- HULL, T. E., DOBELL, A. R. 1962. Random Number Generators. *SIAM Review*. 4(3), 230-254. ISSN 0036-1445. <https://doi.org/10.1137/1004061>
- KNUTH, D. E. 1998. *The Art of Computer Programming*. Volume 2: Seminumerical Algorithms. 3rd ed. Upper Saddle River, NJ: Addison-Wesley, c1998. ISBN 0-201-89684-2.
- L'ECUYER, P. 1999. Tables of Linear Congruential Generators of Different Sizes and Good Lattice Structure. *Mathematics of Computation*. 68(225), 249–260. ISSN 1088-6842. <https://doi.org/10.1090/S0025-5718-99-00996-5>

- NAPS, T. L. *et al.* 2002. Exploring the Role of Visualization and Engagement in Computer Science Education. In: *ITiCSE-WGR ,02: Working group reports from ITiCSE on Innovation and technology in computer science education*. New York: ACM, pp. 131-152. ISBN 978-1-4503-7449-1. <https://doi.org/10.1145/960568.782998>
- SINGER, D. G. 2022. Secrets of Linear Feedback Shift Registers. *Mathematics Magazine*. 95(4), 368-375. ISSN 0025-570X. <https://doi.org/10.1080/0025570x.2022.2094166>
- ZÁLEŠÁK, Š. 2026. *Web presentation and comparison of LCG and LFSR algorithms for generating pseudorandom numbers* [in Czech]. Bachelor's thesis, head: Farana, R. Brno: Mendel University in Brno.

Contact information

Štěpán Zálešák: e-mail: stepan.zalesak@outlook.com

Radim Farana: e-mail: radim.farana@mendelu.cz