

WEB APPLICATION FOR VISUALIZATION AND COMPARISON OF MINIMUM-REDUNDANCY CODES

Ivan Kolesnychenko¹, 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/10.11118/10.11118/ecos2026_014

ABSTRACT

This article presents the results of the development and testing of a specialized web application designed for teaching and comparing lossless data compression algorithms based on minimum redundancy codes. The work focuses on two fundamental pillars of statistical coding: the Shannon-Fano (Shannon 1948) and Huffman algorithms (Huffman 1952). The main contribution of the created solution is an interactive environment that allows users to monitor calculation steps and the creation of tree structures in real-time on identical inputs, which supports an intuitive understanding of the differences in the efficiency of both methods. The technical solution is built on modern Vue 3 and D3.js technologies. Validation took place in the form of user testing, which proved the high didactic potential of the application and its benefit for students of technical fields.

Keywords: lossless compression, Shannon-Fano coding, Huffman coding, entropy, algorithm visualization, web application

JEL Code: C63

1 INTRODUCTION

Data compression is a fundamental process in the field of modern computer science and telecommunications (Cover and Thomas 2006). With the constantly increasing volume of transmitted data, effective coding methods are becoming key to optimizing storage space and network throughput. Despite their practical ubiquity, the teaching of basic principles of lossless compression at universities is often limited to static theoretical interpretation, which may not sufficiently reflect the dynamics of the algorithms themselves. The motivation for this work was the need to create an interactive and visually attractive tool that would allow students to experiment with input data and observe how changes in the probability of symbol occurrence are reflected in the structure of the resulting code. The presented text describes the design of such an application, its technological implementation, and an evaluation of its contribution to the educational process (Kolesnychenko 2026).



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>

2 CURRENT STATE AND ANALYSIS

2.1 Theoretical Foundations

Lossless compression is based on the principle of eliminating redundancy while maintaining information integrity. A key metric in this context is source entropy, which determines the minimum average code length per symbol required for error-free reconstruction (Farana 2014).

$$H(X) = - \sum_{x \in X} p(x) \log_2 p(x) \quad (1)$$

Where $p(x)$ represents the probability of a symbol's occurrence in the source alphabet. Shannon-Fano and Huffman coding then represent prefix codes that attempt to approach this theoretical limit by assigning shorter code words to symbols with higher probability (Salomon 2007). A minimum-redundancy code is better the closer its average codeword length is to the entropy.

2.2 Existing Solutions

During the analysis, a number of available visualization tools were examined. It was found that although general platforms exist (e.g., Algorithm Visualizer), they often lack a specific focus on information theory or do not allow a direct comparison of two methods over the same text. Educational resources such as the Purdue University tutorial provide quality theory but lack interactivity. These findings led to the definition of requirements for a new solution that would combine these aspects (Mishra 2023).

3 APPLICATION DESIGN

The goal of the design was to create an environment that minimizes the user's cognitive load and maximizes the didactic effect. The application was designed as a single-page application (SPA) with an emphasis on clarity and responsiveness.

3.1 Functional Requirements

Lossless compression is based on the principle of eliminating redundancy while maintaining information integrity. A key metric in this context is source entropy, which determines the minimum average code length per symbol required for error-free reconstruction.

- **Input Data:** Option to enter a custom string or select from predefined test sets (e.g., homogeneous distribution vs. extremely unbalanced frequencies).
- **Visualization:** Generation of binary trees using the D3.js library, which dynamically react to the algorithm's stepping.
- **Comparison:** Parallel display of Shannon-Fano and Huffman algorithm progress side by side.
- **Metrics:** Immediate calculation of entropy, average code length, and efficiency for both methods.

4 APPLICATION DESCRIPTION AND IMPLEMENTATION

The application was developed using the modern JavaScript ecosystem. The core consists of the Vue 3 framework using the Composition API, which allows for efficient state management and reactivity. The entire code is written in TypeScript, which significantly facilitated the implementation of complex algorithm logic and reduced the risk of errors when manipulating tree structures.

4.1 Visualization Logic

A key element is the generation of intermediate algorithm steps. Each step is saved as a “snapshot” of the current state of the data structures. For Huffman coding, the phases of joining two nodes with the lowest probability are visualized, while for the Shannon-Fano algorithm, the application displays the process of dividing the symbol list into two parts with the closest possible sum of probabilities, see Fig. 1.

4.2 Graphical Interface

The D3.js library was used for tree rendering, enabling smooth transitions and node animations. Thanks to this, the user “sees” how the nodes group or divide over time. The interface is localized into Czech and English using the vue-i18n module, which expands the potential target group.

5 TESTING RESULTS

User testing was conducted to verify the functionality and benefits of the application. It was attended by 15 respondents from the ranks of students (both IT and non-IT oriented) and the general public. The methodology included solving specific tasks in the application, followed by a structured interview with ratings of individual aspects on a scale of 1–10.

From the results, it is clear that the application was rated very highly in all monitored categories. The most significant success was recorded in the “suitability for teaching” category, where it reached an average value of 9.27.

Tab. 1 Average rating of selected categories

Respondent Group	Understanding of Principles	Suitability for Teaching	Overall Rating
IT students	9.4	9.6	8.8
Non-IT students	9.2	8.0	9.6
Non-students	9.0	8.0	8.6
Total average	8.47	9.27	9.00

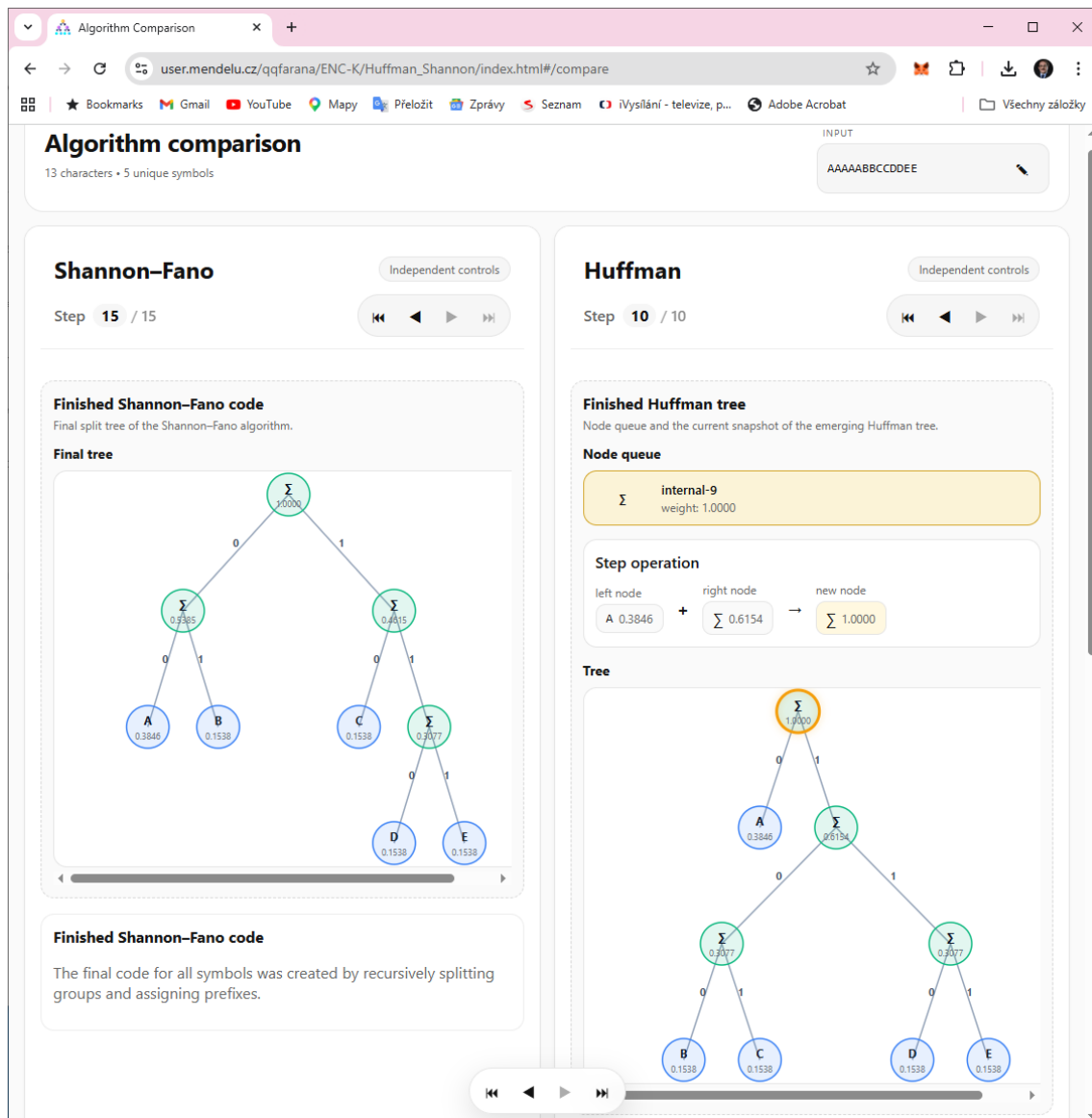


Fig. 1: Algorithm comparison

6 EVALUATION

The feedback analysis confirmed that the visual representation of the algorithm's progress is much more understandable for students than studying pseudocode or static diagrams. Users particularly appreciated the ability to step back and compare the results of both methods on identical data. Higher ability to interpret efficiency results was observed in the IT student group, while the lay public primarily appreciated the aesthetic and technical side of the application. The created solution outperforms analyzed competition mainly in the areas of interactivity and didactic clarity. The ability to step through the algorithm both forwards and backwards, along with comments explaining each step, contributed most to the correct understanding of both algorithms. This allowed the user to go back and forth through the problematic part of the algorithm several times.

By comparing the results of both algorithms, the user will also verify the validity of the empirical finding that the Shannon-Fano algorithm generates codes of worse average length, at most the same as the Huffman code.

7 CONCLUSION

The project successfully demonstrated that modern web technologies can serve as an effective platform for teaching complex algorithmic concepts. The web application for lossless compression visualization represents a functional tool that helps students overcome the barrier of theoretical complexity and gain intuitive insight into the issues of information theory. Future development of the application could move toward integrating other coding methods (e.g., LZW, arithmetic coding) or expanding the theoretical part with interactive quizzes to further consolidate the knowledge gained through visualization. The created tool is ready for deployment in teaching as supplementary material for subjects focused on data processing and information theory.

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

- COVER, T. M., THOMAS, J. A. 2006. *Elements of Information Theory*. 2nd ed. Hoboken: Wiley-Interscience. ISBN 978-0-471-24195-9
- FARANA, R. 2014. *Data Coding and Compression* [in Czech]. Ostrava: VŠB-TU Ostrava. ISBN 978-80-248-3564-8
- HUFFMAN, D. A. 1952. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*. 40(9), 1098–1101, ISSN 0096-8390. <https://doi.org/10.1109/JRPROC.1952.273898>
- KOLESNYCHENKO, I. 2026. *Web Application for Visualization and Comparison of Lossless Compression Algorithms* [in Czech]. Bachelor's thesis, head: Farana, R. Brno: Mendel University in Brno.
- MISHRA, A. 2023. Shannon-Fano Algorithm for Data Compression. *Scaler*. [Accessed: 2026-04-12]. <https://www.scaler.com/topics/shannon-fano-algorithm-for-data-compression/>
- SALOMON, D. 2007. *Data Compression: The Complete Reference*. 4th ed. London: Springer. ISBN 978-1-84628-602-5
- SHANNON, C. E. 1948. A mathematical theory of communication. *Bell System Technical Journal*. 27(3), 379–423, ISSN 0005-8580. <https://doi.org/10.1002/j.1538-7305.1948.tb01338.x>

Contact information

Ivan Kolesnychenko: e-mail: ikolesny@seznam.cz

Radim Farana: radim.farana@mendelu.cz