IN SEARCH OF THE PERFECT ALGORITHM: THE INSOLUBLE RIDDLE OF COMPUTER TECHNOLOGY

Bielytskyi D.M., Yeromina N.S.
Kharkiv National University of Radio Electronics, Kharkiv, Ukraine
Ponomarenko O.M.
Educational and Scientific Institute "Ukrainian Engineering and Pedagogical Academy" of V. N. Karazin Kharkiv National University, Kharkiv, Ukraine

An ideal algorithm is a theoretical concept that represents the highest form of computing process capable of performing tasks with minimal resource consumption and high versatility. **The purpose of the report** is to analyze the concept of an ideal algorithm. **The report** will discuss the criteria for achieving an ideal algorithm.

The main criteria for ideal algorithm are minimal execution time, low memory consumption, ease of implementation, versatility, adaptability and stability, and for encryption algorithms, the highest level of security.

Attempts to combine these requirements in a single algorithm face key limitations: fast algorithms often need more resources, while memory-efficient methods may be slower. Single-threaded algorithms suit limited devices, while multi-threaded ones are better for powerful systems. Additionally, the pursuit of versatility can reduce efficiency or accuracy for certain tasks [1].

Examples of algorithms that try to get closer to the concept of the ideal are sorting algorithms such as QuickSort and TimSort. QuickSort is known for its fast execution speed due to its O(n\logn) time complexity in the average case, but it is not stable and requires a significant amount of memory when processing large data sets. This makes it less suitable for tasks where it is important to preserve the relative order of identical elements. TimSort, which combines merge and insert approaches, offers stability and adapts well to partially sorted arrays. However, it has a more complex implementation and may be slower on unsorted data than QuickSort, highlighting trade-offs between speed, stability, and adaptability. Analyzing existing methods—sorting, greedy algorithms, and multithreading—demonstrates the challenges in developing a universal algorithm [2].

However, modern approaches, such as artificial intelligence and quantum computing, open up new perspectives, allowing algorithms to adapt to real-time tasks and use parallel computing. While the concept of the perfect algorithm remains elusive, it is driving improvements in methods that bring algorithms closer to versatility and optimality in the face of increasingly complex data and computing platforms.

References

1. Chronaki K. Exploiting asymmetric multi-core systems with flexible system software : doctoral thesis. 2018. 138 p. URL: http://hdl.handle.net/10803/664032.

2. Combining Preference Elicitation with Local Search and Greedy Search for Matroid Optimization / N. Benabbou et al. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2021. Vol. 35, no. 14. P. 12233–12240. URL: https://doi.org/10.1609/aaai.v35i14.17452.