UDC 004.62:004.72

Oleshchenko L.M., assoc. prof., PhD; Ziborov D.Y., student

National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”

ADAPTIVE DATA COMPRESSION METHOD AND SOFTWARE FOR OPTIMIZING DATA TRANSMISSION

Анотація

К.т.н., доцент Олещенко Л. М., магістрант Зіборов Д.Ю.
Метод адаптивного ущільнення даних та програмне забезпечення для оптимізації передачі даних

У статті проаналізовано використання алгоритмів ущільнення в динамічних мережевих середовищах з гетерогенними типами даних. Запропоновано адаптивний метод ущільнення, який автоматично обирає оптимальний алгоритм на основі ентропійного аналізу даних, параметрів мережі та доступності обчислювальних ресурсів. Розроблено модульну архітектуру програмного забезпечення, яка зменшує час передачі даних порівняно зі статичними підходами. Проведено оцінку результатів у порівнянні з традиційними методами стиснення та надано рекомендації щодо подальшого вдосконалення запропонованого методу.

Introduction

The dynamic growth of digital data creates critical load on network infrastructure. Traditional compression algorithms use fixed parameters without considering data types and transmission conditions, leading to inefficient resource utilization. For structured data, the system does not adapt to its properties; for random sequences, it wastes resources without results. It is essential to develop an adaptive method that selects the optimal algorithm based on entropy analysis, network parameters, and available resources to minimize end-to-end latency.

The research subject is adaptive software data compression algorithms that automatically select the optimal compression method based on entropy analysis of the input stream, fragment repetition coefficient, and network environment parameters. The object of research is the process of transmission, storage, and processing of digital data in computer systems using adaptive data compression methods aimed at minimizing end-to-end latency.

Terminology

Data entropy – an indicator of the degree of randomness and structure of information in a stream.

End-to-end latency – the aggregate time required for compression, transmission, and decompression of data.

Proposed method and software architecture

The proposed method minimizes transmission latency by accounting for compression time, transmission time, and decompression time. Unlike traditional approaches that maximize the compression ratio regardless of conditions, this method achieves an optimal balance between three latency components. If computational resources are limited, even a highly efficient algorithm can prove suboptimal due to lengthy processing. If the network is slow, high compression is critically more important than processing speed. The system selects a strategy that minimizes the sum of all three components, rather than maximizing any single one.

To implement this approach, a modular software architecture with five interconnected components has been developed (Fig. 1). The stream dispatcher intercepts network traffic in transparent mode, presents data in logical blocks, and coordinates interaction between modules. The monitoring module continuously collects information about system state: CPU utilization, available memory, effective network bandwidth, latency, and packet loss ratio.

The analysis module examines a representative set of input data, computing Shannon entropy, sequence repetition coefficient, and trial compression results. The obtained characteristics are passed to the decision-making module, which based on a cost model calculates the predicted latency for each available strategy and automatically selects the optimal one. The compression module implements the selected algorithm (LZ4, Zstandard, Brotli, GZIP) through a unified interface, ensuring the possibility of easily extending the system with new compression methods. This approach provides flexibility, scalability, and practical efficiency in dynamic network environments. The system ensures that algorithm selection always corresponds to current conditions, minimizing data delivery time.

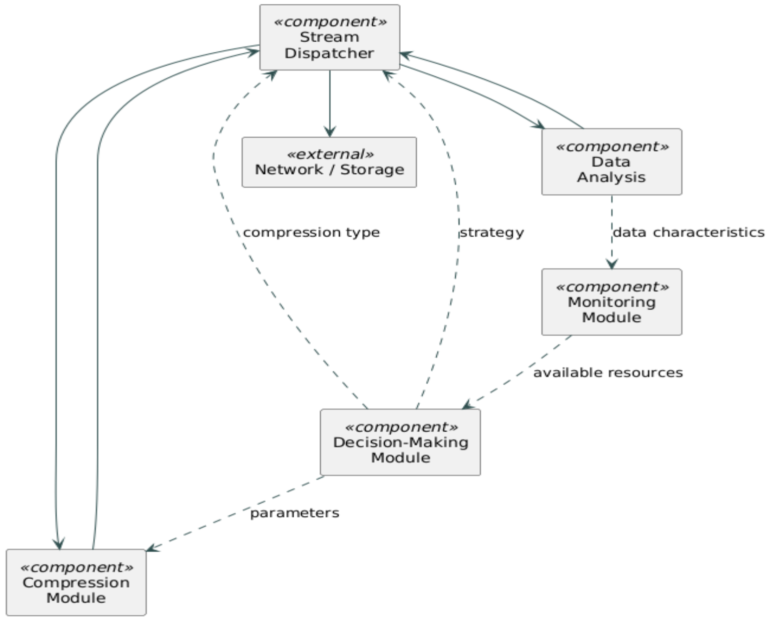

Fig. 1. Architecture of module interaction

Research results

The conducted computational experiment aimed to evaluate the effectiveness of the proposed adaptive compression method under diverse real-world conditions. Four representative data categories – repetitive, structured, text, and media, were selected to reflect different entropy levels and compression potentials, with an additional category of pre-compressed data serving as a control case. For each category, the system dynamically selected the optimal compression strategy based on real-time analysis of entropy, system load, and network conditions.

The analysis demonstrated a substantial dependence of adaptive compression effectiveness on data entropy (Table 1).

Table 1.

Adaptive compression results by data category

Category	Entropy	Compression Ratio (%)	Volume Reduction (%)	Transmission Time Reduction (%)
Repetitive	0.0 – 1.5	95 – 99	94 – 98	90 – 96
Structured	2.5 – 3.5	80 – 90	88 – 97	75 – 85
Text	4.5 – 5.5	55 – 75	52 – 73	30 – 45
Media	6.0 – 7.0	10 – 20	9 – 19	5 – 15
Pre-compressed	7.5 – 8.0	0 – 5	0 – 5	0 – 2

For low entropy the system achieved maximum performance: 95–99% compression and 90–96% reduction in transmission time. For structured formats, the ratio was 80–90% with 75–85% time savings; for text data, 55–75% compression and 30–45% time reduction. For lossless media, the system provided 10–20% additional compression. Critically important is that for pre-compressed files, the system recognized high entropy and refused further processing, saving computational resources. This demonstrates the adaptivity of the method – it does not waste resources on futile compression but selects the optimal strategy for each data type. Related works [1-2] focus on the analysis of Internet traffic structure and data compression methods in cloud systems, while proposed method is distinguished by its scientific novelty due to the adaptive selection of the compression algorithm based on a cost model that simultaneously takes into account the time of data compression, transmission, and decompression, ensuring minimal overall latency in real time. Average efficiency gains ranged from 50–80% compared to static methods, confirming the practical value of the adaptive approach.

Conclusions

The proposed adaptive data compression method and software demonstrate substantial advantage over traditional static approaches. Proposed system provides 50–80% reduction in data transmission time, decreased load on network channels, and rational use of computational resources. The modular software architecture allows supplementation of the system with new compression algorithms and its integration into modern information systems of various purposes. The developed method has practical value for application in content delivery networks, cloud services, database management systems, and other systems with high requirements for data transmission efficiency. Future research perspectives include integration of machine learning algorithms for predicting compression effectiveness, development of adaptive dictionaries, and expansion of support for specific data types.

References

1. Kaium Hossain, Mizanur Rahman, Shanto Roy. 2019. International Journal of Cloud Applications and Computing. IoT Data Compression and Optimization Techniques in Cloud Storage. https://doi.org/10.4018/IJCAC.2019040103.
2. Ege Cem Kirci, Ayush Mishra, Laurent Vanbever. 2025. Five Blind Men and the Internet: Towards an Understanding of Internet Traffic https://arxiv.org/abs/2509.06515.