Research on observation data compression of hydraulic engineering based on swinging door trending algorithm

Lihuan Liu¹*, Jiang Ceng¹ and Chunhui Gu¹

¹Department of Water Resources, Hunan Institute of Water Resources and Hydropower, Changsha, Hunan Province, 410131, China

*Corresponding author’s e-mail: llhuan@slsdoa.com

Abstract. With the wide application of big data technology, there is a geometric series increase of the real-time data collection of water conservancy engineering, which causes a large number of storage units and memory. In order to make the huge amount of data easy to collect and process, an improved Swinging Door Trending algorithm is proposed to compress the data of water conservancy project based on the deep research of data compression algorithm. This work firstly analyzed the collected abnormal points, which identified and processed the points in time to realize the timely alarm of abnormal conditions according to the characteristics of water conservancy engineering data and acquisition requirements; then discussed the dynamic adjustment of tolerance, which is carried out according to the fluctuating state of water conservancy engineering data to store less real-time data; and finally put forward that the improved SDT algorithm has higher compression ratio and better compression efficiency than the traditional SDT method does by experiments.

1. Introduction

For a long time, the water conservancy administration has accumulated a large amount of business data in daily and emergency work, and has formed a relatively stable data growth system. The collection of water conservancy big data is the basis of storage and sharing research, but the data storage model of the water conservancy information system has not yet been unified, which has brought certain difficulties to the analysis and application of water conservancy big data. The quantity of real-time data in water conservancy project is large, and the statics are varying while being collected, resulting in the occupation of a large number of storage units and excessive consumption of memory, which is a heavy burden on the storage of the database; the hydraulic engineering data also needs to be monitored and recorded in real-time to achieve timely alarms and diagnosis of abnormal conditions [1]. In order to ensure that massive data can be stored and processed in real-time, reduce the data storage capacity as much as possible, save process data storage space, simplify and improve process operating conditions, and improve process data processing efficiency, it is necessary to perform fast and efficient compression processing of the data.

Data compression algorithms can be classified into lossy compression and lossless compression according to compression accuracy. Lossy compression determines whether to save data through certain conditions, and a part of the data is discarded during the compression process. These data are considered to have little impact on the recovery of the original data. Lossless compression can completely recover the original data. Of the two data compression algorithms, lossy compression has a higher compression rate by SDT algorithm[2-4]. Currently, in the real-time database field, due to a
large amount of original data, the data changes are relatively stable and can tolerate partial data loss. A lossy compression algorithm is usually used to obtain a higher compression ratio. SDT is a lossy compression algorithm used in real-time databases with high efficiency and high compression ratio. The advantages of simple implementation and controllable errors are widely used in real-time data compression. Xudong Xu et al.[5] fitted the data curve with a sine wave curve based on the fluctuation state of the data, and considered the abnormal data, which improves the compression performance of the SDT algorithm to a certain extent, but the applicability of the compression parameters of the algorithm should be determined according to the specific professional background. Jinsong Liu et al.[6] improved the SDT algorithm by implementing dynamic adjustment of tolerances, but did not consider the handling of abnormal data. Fayong Ma et al.[7] proposed an SDT algorithm based on an effective estimation, which can improve the data compression rate and reduce the time. However, the impact of abnormal data has not been considered.

When collecting hydraulic engineering data, it is often not sensitive to data that fluctuates near normal values, but rather care about abnormal values below or above a certain limit, so it is not efficient to use the same compression method for all data. A new improved SDT is proposed that adaptively determines abnormal points, dynamically adjusts tolerances based on the fluctuation state of the data, and saves points that have a large impact on compression accuracy based on the fitting error to reduce the compression error.

2. The principle of SDT algorithm

The SDT algorithm is a lossy compression algorithm for real-time data with strong compression ability and good compression effect. The basic principle is described as for the original data a, b, c ... h, the corresponding time points are t₀, t₁, t₂ ... ts; firstly, t store the initial point a, and use the two points above and below the distance ΔE as the fulcrum; let point a be the first data point of the first compression interval, t, save point a, and use the two points with a distance of ΔE (compression tolerance) as the fulcrum to establish two virtual doors. As the compression progresses, the two doors will rotate and open, and once opened, they can no longer be opened or closed. As long as the internal angle of the two doors is smaller than 180°, that is, when the doors are not parallel, the revolving door operation continues and the data points covered are discarded[8]. When the internal angle of the two doors is greater than or equal to 180° (as point d showed in Figure 1), the operation is stopped, data points that fall outside the compression interval are saved, and a new round of compression is performed with this point as the starting point of the next round of compression. The original data in Figure 1 is compressed by the SDT algorithm and becomes a, c, e 3 points, and adjacent data points are connected by line segments, and the line segments are used to replace other data points that are not saved. When decompressing, restore by linear interpolation Compressed points. The algorithm needs to record the length of each interval, the starting point data, and the ending point data, and the saved data of the previous section is the starting point of the next section.

![Figure 1. SDT algorithm schematic diagram](image-url)
The standard SDT algorithm has the following disadvantages: 1) Because SDT is a lossy compression algorithm, some data will be discarded during the compression process, and the points that are significantly different from others are retained. These abnormal data will affect the compression accuracy and compression performance of the data compression algorithm, but SDT only uses it as the starting point for a new round of compression, and does not do the identification and detection. 2) The compression tolerance \( \Delta E \) is a key factor to ensure the compression performance. However, its selection depends on experience. If the value of \( \Delta E \) is too large, the compression ratio will be increased, but some abnormal points will be discarded, resulting in different trends between the compressed data and the original data, which increases the compression error; if the value of \( \Delta E \) is too small, it will retain more data and reduce compression errors, but it will also reduce the compression ratio and reduce the compression effect.

3. Improved SDT algorithm

3.1. The principle improved SDT algorithm

In the field of water conservancy engineering, one of the main functions of informatization management is to judge abnormal situations and provide timely warnings. During the data compression process, some data have very large differences, which means some assignable changes such as topography or geological conditions and so on; there are also data that exceed established limits, which represent that the water regimen needs another level of regulation. In order to avoid the impact of abnormal data as much as possible, the multi-model optimization method is used instead of linear fitting to fit the interval line segments, so as to ensure that less hydraulic engineering data is stored without losing valid information and improve compression efficiency.

Compression ratio CR and compression error MSE are used as indicators to measure the quality of the improved SDT compression algorithm. The compression ratio refers to the ratio of the number of points of the original data to the number of points of the compressed data. The larger the compression ratio, the less storage space it takes.

\[
CR = \frac{n}{m}
\]  

(1)

There are many indicators for describing compression error, including relative error, absolute error, and standard deviation. The standard deviation indicates the degree of data dispersion and the size of the data fluctuation. The smaller the standard deviation, the closer the compressed and decompressed data is to the original data. The improved SDT algorithm uses standard deviation as the compression error.

\[
MSE = \left( \frac{1}{n} \sum (x_i - \mu)^2 \right)^{1/2}
\]  

(2)

Among them: \( n \) is the number of original data points, \( m \) is the number of data points after compression; \( x_i \) is the actual data value, and \( \mu \) is the data value after compression and decompression.

The main objectives of the improved algorithm are detecting abnormal data dynamically adjusting tolerances applying function classes to run the fitting of hydraulic engineering data to eliminate the impact of noise data. The steps of the algorithm are described in detail below.

3.2. Algorithm steps

Step 1: The intercept time range of the compression interval is \( T \), and the initial tolerance is set as \( \Delta E \) based on the empirical value of the actual data.

Step 2: Set the data change limit. Determine whether the data point being collected is abnormal. If it exceeds \( \Delta L \), the data is stored; if it does not exceed \( \Delta L \), the data is compressed by standard SDT.
Step 3: Perform standard SDT compression on the data in the compression interval, starting from the most recently saved data point and ending with the current data point, and construct a parallelogram with a tolerance of $\Delta E$. Use this parallelogram to determine whether the non-anomalous data has been saved. If the data point is not within the range of the parallelogram, then save and take it as the starting point of the new compression interval, and repeat this process.

Step 4: After the compression interval, fit the compressed data $y_1, y_2, \ldots, y_{m-1}$. Because many factors affect the hydraulic engineering data, and the data are diversified, function class $\{\phi_1, \phi_2, \phi_3, \ldots, \phi_s\}$ is used. The mathematical model in the model is subjected to least squares fitting, and the function with the smallest sum of squared errors is selected from it. It is recorded as $y = F(x)$, which satisfies the function:

$$\sigma^2 = \sum_{i=0}^{m-1} \left[ F(x_i - y_i) \right] = \min \sum_{i=0}^{m-1} \left[ F(x_i - y_i) \right]^2$$

(3)

Step 5: Calculate the standard deviation of the original data in this compressed interval:

$$\sigma = \left[ \frac{1}{n+1} \sum_{i=0}^{n} (y_i - u)^2 \right]^{1/2}$$

(4)

In the above function, $u$ is the average of the original data in the interval. If it is the initial compression interval, save the standard deviation $\sigma$ of this interval and compare with the standard deviation $\sigma' = \Delta E$ in the next compression interval. When $\sigma' \neq \sigma$, go to step 6; when $\sigma' = \sigma$, proceed to step 7;

Step 6: Redetermine $\Delta E$ according to the following conditions.

(1) $\sigma = 0$ means that the data in adjacent compression intervals are the same degree of dispersion, and then $\Delta E$ remains unchanged, $\sigma' = \Delta E$;

(2) $\sigma / \sigma' \geq 1$ means that the current compression interval is smaller than the next compression interval, and the data dispersion decreases. Enlarging $\Delta E$ increases compression ratio, and $\Delta E$ is assigned to the function:

$$\Delta E = \frac{\sigma}{\sigma'} \Delta E$$

(5)

(3) $\sigma / \sigma' \leq 1$ means that the current compression interval is larger than the next compression interval, and the data dispersion increase, decreasing $\Delta E$ reduces the compression error, and $\Delta E$ is assigned to the function (5).

Step 7: If there are uncompressed points and the time zone is greater than $T$, take step 2. Otherwise, the algorithm is ended.

The algorithm flow is shown in Figure 2. Compared with the standard SDT algorithm, the improved SDT can judge and record the abnormal points according to the characteristics of hydraulic engineering data, and can dynamically adjust the tolerance. Recording the abnormal points can be early-warned in time to ensure timely feedback on conditions related to the safe operation of water conservancy projects, and adjusting the tolerance can improve the compression ratio and improve the efficiency of data processing.
4. Verification

In order to affirm the performance of the improved algorithm, it is necessary to verify it. Firstly, 6 groups of data were used for comparative verification. Table 1 lists the daily rainfall data generation function in a reservoir area. Among them, the first three groups represent that the fluctuation of process data is not very obvious and does not have abnormal data; the latter three groups of data fluctuate more violently and have abnormal data $F(t)$, which can better simulate the actual process data. Table 2 shows the comparison of daily rainfall data in the reservoir area.

**Table 1** List of Producing Functions of Daily Rainfall Data in a Reservoir Area

| Data | Function | Remarks |
|------|----------|---------|
| 1    | $9+0.2x$ | $x=1,2,...,5000$ |
| 2    | $10+0.2x$ | $x=1,2,...,5000$ |
| 3    | $9.6+0.3x$ | $x=1,2,...,5000$ |
| 4    | $9+0.2x+F(t)$ | $x=1,2,...,5000$ |
| 5    | $10+0.2x+F(t)$ | $x=1,2,...,5000$ |
| 6    | $9.6+0.3x+F(t)$ | $x=1,2,...,5000$ |
In Table 2, the simulation calculation of data groups 1 and 2 shows that both the SDT algorithm and the improved SDT algorithm can adapt to the change of process trend. The simulation calculation of data 4, 5, and 6 shows that the process data with abnormal points will reduce the compression ratio and increase the compression error. The improved SDT can complete the data compression well. The calculation results show that the compression ratio of ISDT algorithm is CR 83, more than that of SDT algorithm on average. 5, while the compression error CE increased by only 3.7. One point that deserves special emphasis is that the improved SDT algorithm has the ability to identify and process abnormal points, while SDT algorithm does not. During the processing of actual process data, the use of improved SDT algorithm can reduce the requirement of storage space, reduce the possibility of blockage of Fieldbus network, and improve the overall performance of the control system. meanwhile, the ISDT compression error increases for a certain set of data when comparing the improved SDT algorithm and the SDT calculation results, but it does not mean that the improved SDT algorithm achieves an ideal compression ratio at the expense of compression accuracy.

The experiment selects 4 sets of data, the amount of which are 2000, 5000, 20000, and 50000. Figure 3 shows the comparison of compression ratios between standard SDT algorithm and improved SDT algorithm. Figure 4 shows the comparison of compression error between standard SDT algorithm and the improved SDT algorithm.

| Data | Algorithm       | Data points | Abnormal points | Compression ratio | Compression error |
|------|-----------------|-------------|-----------------|-------------------|-------------------|
| 1    | Standard SDT compression | 5000        | -               | 11                | 0.0               |
|      | Improved SDT compression | 0           | 11              | 0.0               |
| 2    | Standard SDT compression | 5000        | -               | 11                | 0.0               |
|      | Improved SDT compression | 0           | 11              | 0.0               |
| 3    | Standard SDT compression | 5000        | -               | 8.543             | 3.602 2E+003      |
|      | Improved SDT compression | 0           | 8.962           | 8.992 7E+003      |
| 4    | Standard SDT compression | 5000        | -               | 2.225             | 6.773 9E+003      |
|      | Improved SDT compression | 276         | 6.448           | 6.100 5E+003      |
| 5    | Standard SDT compression | 5000        | -               | 2.348             | 6.666 9E+003      |
|      | Improved SDT compression | 276         | 6.758           | 6.008 5E+003      |
| 6    | Standard SDT compression | 5000        | -               | 2.377             | 7.012 5E+003      |
|      | Improved SDT compression | 276         | 6.023           | 8.123 3E+003      |

Table 3 Comparison of algorithm performance

| Data Sample | Compression ratio | Compression error |
|-------------|-------------------|-------------------|
| Standard SDT compression | 2000 | 5.30 | 2.51 |
| Improved SDT compression | 2000 | 9.02 | 2.96 |
| Standard SDT compression | 5000 | 4.92 | 2.25 |
| Improved SDT compression | 5000 | 8.60 | 2.56 |
| Standard SDT compression | 10000 | 3.35 | 1.72 |
| Improved SDT compression | 10000 | 8.41 | 2.02 |
| Standard SDT compression | 50000 | 4.61 | 1.82 |
| Improved SDT compression | 50000 | 9.10 | 1.93 |
Figure 3. Comparison of Compression Ratio between Two Algorithms

Figure 4. Comparison of Compression Error between Two Algorithms

The data shows that the improved SDT algorithm improves the compression ratio, and increases the compression error at the same time, while the amplitude of the compression ratio increases greatly, and the compression ratio of the four data groups are all larger than 70%, but the amplitude of the compression error shows a downward trend.

5. Conclusion
In order to ensure the safety of water environment, the SDT algorithm is improved by combining with the optimized data collection method of hydraulic engineering to screen and compress the massive data of hydraulic engineering. Experiments have proved that the data processing of hydraulic engineering is optimized and the data processing efficiency is improved by the improved SDT algorithm.

Acknowledgments
The research is supported by Natural Science Foundation of Hunan Province. We would like to show our gratitude to the Hunan Hydropower Vocational College for their support in the research. We thank Ms. Zhao Aihua for her assistance.

References
[1] Chen J.F., Deng M.H., Wang H.M.(2007) Summary of water resources big data research.J.Advances in Water Science. 32: 622-631
[2] Salomon.D.(2007) Data compression-the complete reference, 4th edition.Springer Verlag Publishing, New York.
[3] Saha S. K., R ahaman M., (2007) Boosting the performance of LZW compression through block sorting for universal lossless data compression. In: International Conference on Computer and Information Technology. New York. pp. 1-5.
[4] Li Q, Yang J, Liu C. (2013) A real-time algorithm for fixed-length short data compression. International Conference on Fuzzy Systems and Knowledge Discovery. Paris. pp. 342-348.
[5] Xu X.D., Fu Y.P. (2018) Algorithm and its improvement for historical data in SCADA. Computer Engineering & Science. 6 999-1004.
[6] Liu J.S., Yu D., et al. (2018) Application of Improved SDT Algorithm in CNC Machine Monitoring Field. 2 81-83.
[7] Ma F.Y., Li Q.P., Ma Z.B., (2014) Historical Data Compression and Storage Strategy in Power Dispatching SCADA System. Power System Technology. 4 1109-1114.
[8] Bristol E. H. (1990) Swing door trending: Adaptive trend recording. Proc of National Conference. New York. pp.749-753.