# The Research and Improvement of SDT Algorithm for Historical Data in SCADA

Xu-Dong XU[1] and Yan-Ping FU[1,a]

[1] *Faulty of Information Technology, Beijing University of Technology, Beijing, 100124, China*

**Abstract.** With the rapid development of Internet of things and big data technology, the amount of data collected by SCADA(Supervisory Control And Data Acquisition)system is growing exponentially, which the traditional SDT algorithm can not meet the requirements of SCADA system for historical data compression. In this paper, ASDT(Advanced SDT) algorithm based on SDT algorithm is proposed and implemented in the Java language, which is based on the deep research of the data compression method, especially the Swing Door Trending. ASDT algorithm through the sine curve fitting data to achieve data compression, compared with the performance of the traditional SDT algorithm, which it can achieve better compression results. The experimental results show that compared with the traditional SDT algorithm, the ASDT algorithm can improve the compression ratio in the case of no significant increase in the compression error, and the compression radio is increased by nearly 50%.

## 1 Introduction

SCADA system is one of the process industry system based on configuration software, through the real-time detection and control of industrial equipment to achieve real-time data acquisition and storage of process data, the parameters of the device settings, real-time display of warning signals, and other functions.

With the development of Internet of things and cloud storage technology, amount of data collected by RTU is more and more large which acquisition cycle is also getting shorter and shorter [1]. Historical database constantly storing the historical data is an important part of the SCADA system, which can predict and analyze the system fault in advance to ensure the safe and stable operation of the SCADA system. Because the massive historical data in SCADA system has caused great burden to data storage and data processing, the efficient data compression technology can save storage space and improve the utilization rate of storage space. In order to ensure the efficient storage of mass data, it is very useful to data compression technology in the process of historical data storage in SCADA system.

In industry, there are two methods for data compression --- lossless compression and lossy compression [2]. For lossless compression, it is that the data is totally the same after and before the compression; for lossy compression, it is that the compressed data will be uncompressed, by which the uncompressed data will be deviated from the original data, however, the deviation will not affect the variation trend [3]. For SCADA system, there are three methods for compressing historical data ---

[a] Corresponding author: 958591548@qq.com

piecewise linear method, vector quantization and signal conversion method. For piecewise linear method, it includes boxcar method, backward slope method, SDT method and PLOT method [4]. SDT method is used for industrial flow most because it is easy and simple to be operated, as well as that it can be executed quickly. Signal conversion includes lots of methods, such as DCT (discrete cosine transform) and WT (Wavelet Transform)[5]. Today, WT is being widely used. However, compared with SDT, WT is more complicated.

## 2 Strategy for SCADA historical data compressing

### 2.1 Compressing quantity of state

For quantity of state signal, there are only two values -- 0 and 1, which will be collected by the gathering module at a fixed interval. In real situation, the data related to quantity of state will not change within a while, that is the state signal collected right now is the same as the signal collected last time. For compressing quantity of state, the method that once the data changes, the data shall be collected and saved shall be followed. That is only when the data changes can it be saved. See Figure1, the time t0-t7 has the quantity of state 1,0,0,1,0,0,0,0. Then according to this method, only the quantity of state of t0, t3 will be saved while others will not.

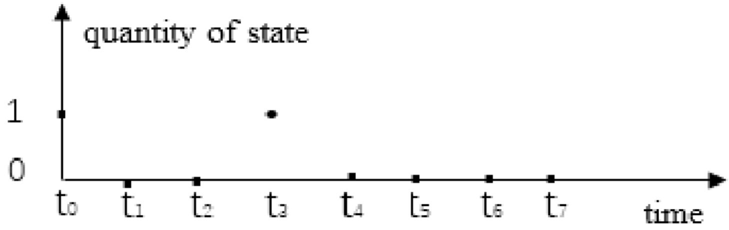

**Figure 1.** Compressing Quantity of State

### 2.2 Compressing analog quantity

#### *2.1.1 SDT algorithm*

SDT is a kind of data compression method based on linear fitting, which saves the data by establishing parallelogram while the data which is covered by the parallelogram will not be saved. That is only data out of the parallelogram will be saved [6]. See Figure2 for SDT steps. In Figure2, it is assumed that point a is the first data in the first compression duration, and the coordinate point (o, o') whose distance to point a is E, will be as the two supporting points for the revolving door. Once there is new data, the slope between o,o' and new data will be calculated. If there is only one data (a), o,o' and a is actually perpendicular to o,o'. The slope is 0, which means that the revolving door is closed; if there are more and more data, the revolving door will be gradually opened, whose slope will be changed at the same time, where we can find that the upper supporting point will have the maximal slope while the lower supporting point will have the minimal slope [7]. Within a while, once the door is opened, it cannot be closed until the slope of upper supporting point is the same as that of lower supporting point. As long as the two doors are not parallel to each other, or the sum of two interior angles is less than 180 degrees, t-he operation on revolving door will be continued. In Figure 2, the first compression duration is started from a to d, the straight line from a to d stands for data(a,b,c,d);Similarly, the second compression duration is started from e, the straigh line from e to h stands for the data (e, f, g, and h).

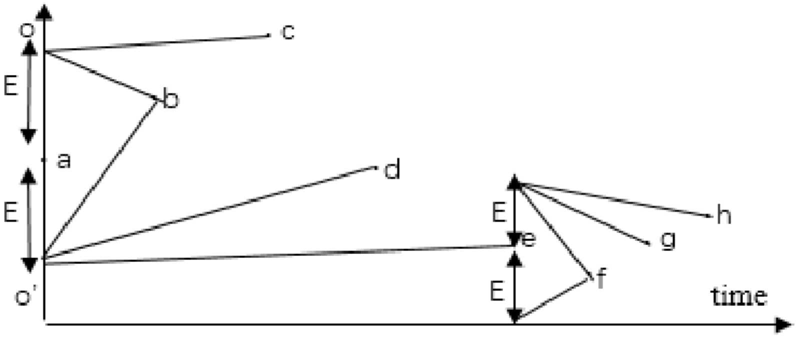

**Figure 2.** The basic steps of SDT algorithm

From what mentioned above (how SDT works), it can be found that there are some problems: first, the threshold value is the only controllable parameter, so E will directly affect the compression performance. Once E is defined, it cannot be changed during the compression. How to define E shall be realized via experimenting for a long time; if E is not proper, SDT performance will be greatly lowered. Second, if the data change is slow during a time, the time used by data compression will be long, which makes the instantaneity of linear trend lost [8].Third, in most situations, the data change is not linear. It is known that SDT is a method to fit data via linear trend.

## 3 Improved SDT

For the aforementioned problems, the researchers have proposed how to improve it. For example, Wang Ju [9], on the basis of the defect suffered by SDT -- E cannot be changed, proposed that E can be adjusted according to the initial threshold value; Yu Songtao [10] proposed an ideology --- dynamic regulation on allowance, that is the threshold value can be set dynamically according to the allowance. Qu Yilin,[11] depending on feedback control system, by which the allowance is able to have a dynamic change during the compression. Zhang Jingtao [12] proposed that the performance of data compression is largely determined by the compression deviation, so the compression deviation parameter shall be selected in a dynamic way. Duan Peiyong [13] proposed that a slope of a straight line cannot be used to show the change trend of several data, all the saved data shall be obtained by calculation but not by the compressed data. Ning Hainan [14] proposed that fitting of change trend shall be done via parabola, by which data compression can be realized; Zhang Jian [15] proposed that the abnormal points shall be inspected and removed, and the length of recording limit shall be set etc.

### 3.1 Basic principle of ASDT

For the improved SDT, the fitting is done between the straight line and compressed data; however, in real situation, the data change will not keep increasing or decreasing in a while. Even though there is fitting via curve, the change trend of data cannot be changed radically. It means that within a while, the data change is monotonous. Meanwhile, the improved SDT proposed that FSRL (Forced Storage-Recording Limit) can be set, which will make the calculation more complicated [16]. For ASDT, the data fitting is done via sine curve. So during a duration, the data will both increase and decrease, which is consistent with the change rule. The cycle of sine curve can be used to solve that how to set the recording limit.

### 3.2 Compression steps of ASDT

As for the setting of cycle of sine curve, it is required to consider the length of compression interval and compression ratio of process industry. For process industry, the compression ratio 10:1 is normally used. If the ratio exceeds this compression ratio, some useful details will be lost. Normally, one cycle of sine curve will be used as the compression interval. Therefore, the cycle shall be set as 10, which not only restricts the length of compression interval but also ensures the compression ratio. See Figure 3 for the flow diagram of ASDT, from which we can see that TL is the length of compression interval, E is the threshold, and the first data of each cycle is $A_0$. ($A_0$ shall be the benchmark of each sine cycle, which shall be fluctuated around $A_0$, for fluctuation, and the upper limit is toplimit as in (1) while the lower limit is lower limit as in (2)).

$$\text{toplimit} = A_0 + \max(\sin(\omega t + \phi) + E) \tag{1}$$

$$\text{lowerlimit} = A0 + \min(\sin(\omega t + \phi) - E) \tag{2}$$

Here below are the steps of ASDT:

Step1: initialization of parameters, TL and T shall be set as 10. $\omega = 2\pi/T = 2\pi/10 = \pi/5$.

Step2: judge that whether t is larger than TL, if not, move to step 4.

Step3: deal with the value of t, and $t = t\%TL$.

Step4: judge that whether the first time point of sine cycle (that is t) is equal to 1, if yes, move to step 7.

Step5: get the value of current data, figure out Emax as in (3) and Emin as in (4). During one sine cycle (10), when t is 2.5 and 7.5, the maximal value and minimal vlaue will be obtained.

$$\text{Emax} = A0 + \sin(0.2\pi * 2.5 + \phi) + E \tag{3}$$

$$\text{Emin} = A0 + \sin(0.2\pi * 7.5 + \phi) - E \tag{4}$$

Step6: judge that whether data is no more than the maximal value and no less than the minimal value, if yes, save the data. If no, move to step8.

Step7: SaveA0, which shall be the data of the first time point of sine cycle.

Step8: an operation that t pluses 1 shall be done.

Step9: judge that whether there is a new data, if yes, move to step2.

Step10: the data compression is ended.

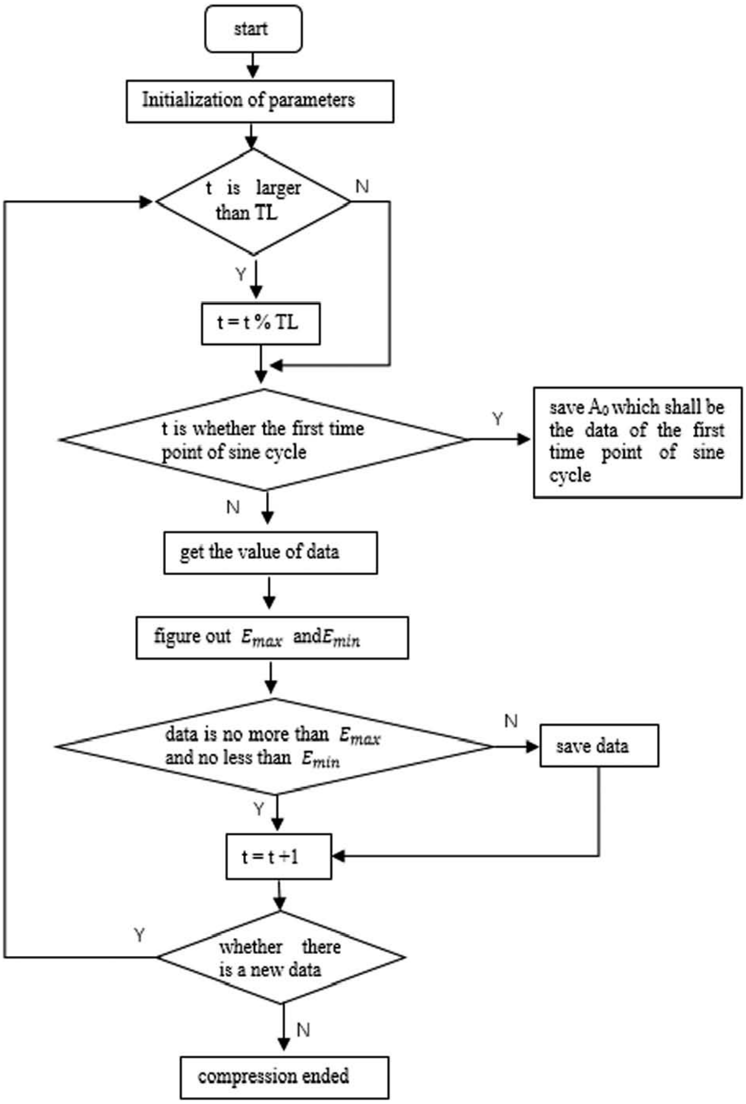
start
Initialization of parameters
t is larger than TL
N
Y
t = t % TL
t is whether the first time point of sine cycle
Y
save A₀ which shall be the data of the first time point of sine cycle
N
get the value of data
figure out $E_{max}$ and$E_{min}$
data is no more than $E_{max}$ and no less than $E_{min}$
N
save data
Y
t = t +1
Y
whether there is a new data
N
compression ended

**Figure 3.** Algorithm flow chart of ASDT

## 4 Experiment result analysis

There are two indexes to evaluate the performance of data compression -- compression ratio and compression error. It is assumed that there is a group data (there are n values), which is recorded as $y_1 \sim y_n$; before compression, the number of data is m(m<=n); after compression, there are n values, which are recorded as $\hat{y}_1 \sim \hat{y}_n$ here below is the definition of MSE (mean square error) as in (5) :

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{5}$$

For this experiment, the data is the remote temperature got by SCADA, 4 groups of data are selected. The digit quantity is 1000, 5000, 10000 and 50000 respectively. In Figure 4, there is a comparison on the compression ratio between SDT and ASDT. In Figure 5, there is a comparison on the compression error between SDT and ASDT. In Table 1, the data is about the experiment result; from the chart and the table, we can know that the compression error of SDT and ASDT almost the same; however, the compression ratio of ASDT is much higher than that of SDT.

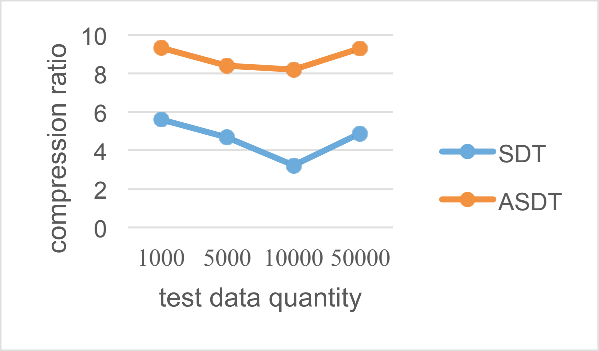

**Figure 4.** Comparison on the compression ratio between SDT and ASDT

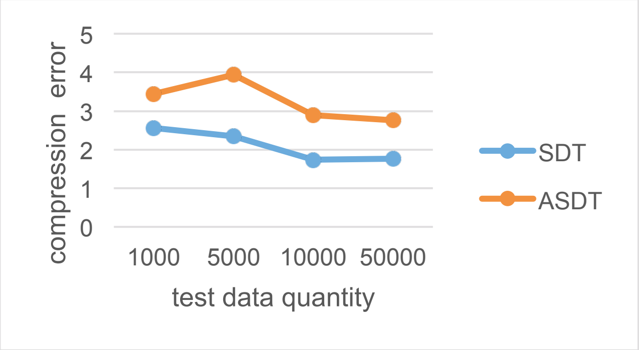

**Figure 4.** Comparison on the compression error between SDT and ASDT

**Table1.** performance analysis and comparison between SDT and ASDT

| **Compression Algorithm** | ***Test Data Quantity*** | ***Compression Ratio*** | ***Compression Error*** |
|---|---|---|---|
| SDT | 1000 | 5.6 | 2.57 |
| ASDT | 1000 | 9.33 | 3.45 |
| SDT | 5000 | 4.67 | 2.35 |
| ASDT | 5000 | 8.40 | 3.94 |
| SDT | 10000 | 3.23 | 1.74 |
| ASDT | 10000 | 8.20 | 2.89 |
| SDT | 50000 | 4.89 | 1.77 |
| ASDT | 50000 | 9.30 | 2.76 |

## Conclusion

In this thesis, it proposed an improved calculation based on the compression of revolving door, as well as an analog computation on the test data by compiling program. ASDT solves the problems suffered by traditional method, like fitting data via straight line does not consider the actual change happened during the data collection; most importantly, ASDT is no need to set the length of compression interval. By the experiments above, it compares SDT and ASDT, which concludes that ASDT can greatly enhance compression ratio without increasing error.

## References

1. S.Sivalingam and M.Hovd, ”Effect of data compression on controller performance monitoring, ”in Control Automation(MED), 2011 19th Mediterranean Conference on, pp.594-599(2011).
2. H.Vedam, V.Venkatasubramanian, M.Bhalodia. A B-spline based method for data compression, process monitoring and diagnosis[J]. Computers & Chemical Engineering, **22**(12):S827–S830(1998).
3. J.Pettersson, P.O.Gutman. Automatic tuning of the window size in the Box Car Back Slope data compression algorithm[J]. Journal of Process Control, **14**(4):431-439(2004).
4. Li X, Qiu H, Zhu Y. Research and Design of Lossy Compression Algorithm in Embedded Real-Time Database[M]. Proceedings of the Second International Conference on Mechatronics and Automatic Control. Springer International Publishing, 1027-1034(2015).
5. R.S.H.Mah, A.C.Tamhane, S.H.Tung. Process trending with piecewise linear smoothing[J]. Computers & Chemical Engineering, **19**(2):129-137(1995).
6. E.H.Bristol. Swing Door Trending: Adaptive Trend Recording[C]. National Conference Proceeding. Sydey: ISA,749-753(1990).
7. Zhang Wang, Chen Xinchu, Lu Dingxing. Research and Application of Improved Process Data Compression Algorithm-SDT [J]. Industrial Control Computer,**08**:1-3+6(2009).
8. Zhao Xudong, Ding Jiexiong, Bian Zhiyuan, Guan Lichao, Xie Gang, Du Li, Wang W-ei. Application of improved SDT algorithm in CNC monitoring system[J]. Manufacturing T-echnology & Machine Tool, **10**:155-159(2014).
9. Wang Ju, Chen Xiaojiang, Xing Tianzhang, Fang Ding Yi, Yuan Chang, Data Compression of Wireless Sensor Networks in the Heritage Monitor [J]. Journal of Xidian University, **01**:157-162(2012).
10. Yu Songtao, Wang Xiaokun, Zhao Liqiang, Yu Tao, Wang Jianlin. The Improved Al-gorithm for SDT Based on the Dynamic Adjustment of Tolerance[J]. Journal of Beijing U-niversity of Chemical Technology (NATURAL SCIENCE), **03**:109-113(2013).
11. Qu Yilin, Wang Wenhai. Automatic Parameter Control SDT Algorithm for Process D-ata Compression[J]. Computer Engineering, **22**:40-42(2010).
12. Zhang Jingtao, Wang Hua, Wang Hongan. Accessing and Compressing of Real-time Data[J]. Control and Instruments in Chemical Industry, **03**:47-50(2003).
13. Duan Peiyong, Zhang Mei, Tang Tongkui. A Swing Door Trending(SDT) Algorithm and its Application to Compressing Process Data Received by Local Operation Network Nodes [J]. Information and Control, **02**:132-135(2002).
14. Ning Hainan. A New Process Data Compression Algorithm Based on SDT Algorithm[J]. Computer Technology and Development, **01**:25-28(2010).
15. Zhang Jian, Liu Guangbin. An New Data Compression Based on ISDT Algorithm an-d its Performance Analysis[J]. Fire Control and Command Control, **02**:80-82+86(2007).
16. Xing Rui, Qi Qi, Zheng Tao. Improved SDT Algorithm [J]. Computer Engineering a-nd Design,**02**:515-518+528(2013).