

## Research Article

# An Algorithm of Auto-Update Threshold for Singularity Analysis of Pipeline Pressure

**Jinhai Liu, Jian Feng, Liang Tan, and Dazhong Ma**

*School of Information Science and Engineering, Northeastern University, Shenyang, Liaoning Province 110819, China*

Correspondence should be addressed to Jinhai Liu; [jh\\_lau@126.com](mailto:jh_lau@126.com)

Received 23 November 2012; Revised 15 February 2013; Accepted 26 February 2013

Academic Editor: Pak-Kin Wong

Copyright © 2013 Jinhai Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A precise auto-update threshold algorithm (AUTA) which imitates the short-term memory of human brain is proposed to search singularities in pipeline pressure signal. According to the characteristics of the pressure signal, the pressure can be divided into two states known as nonsteady state and steady state. The AUTA can distinguish these two states and then choose corresponding method to calculate the dynamic thresholds of pressure variation in real time. Then, the parameters of AUTA are analyzed to determine their values or ranges. Finally, in the simulations to the actual pressure signal from oil pipelines, we verified the effectiveness of AUTA in estimating the dynamic threshold value of pressure.

## 1. Introduction

Pipeline transportation is the fifth transport style following the road, water, rail, and air, and that is the main means for the transportation of oil and gas. Take China as an example, pipeline has transmitted more than 70% of oil and 99% natural gas transportation [1].

There are two class researches on pipeline transportation, one is the running control of pipeline [2–7], and the other is running security. We focus on the security of pipeline in the paper. The leakage due to theft, corrosion, and natural disasters is the main security problem of the pipeline transportation. There are few analysis methods to real time find leakage of pipeline, in which pressure analysis is the primary analysis method. Literature [8] proposed the leak analysis method of wavelet and the maximum likelihood statistic. Reference [9, 10] found the leak points according to wavelet transformation of the pressure signals which improved the accuracy of leak location. In Reference [11], according to the time-domain characteristics, a new fuzzy classifier finding leakage was training by the signals of flow, pressure, and temperature. Reference [12] applied historical pressure to train BP neural network weights and then completed the pipeline real-time leak detection by neural network. Reference [13, 14] used expert systems to

further enhance the robustness and accuracy of pipeline leak detection. Reference [15] employed the supervisory control and data acquisition (SCADA) system with object linking and embedding for process control (OPC) interface to collect data from multisensor, then designed a real-time pipeline leaks fault diagnosis method based on a comprehensive analysis of the data; the SCADA can detect leakage of 1% flow and the location error is less than 0.98% of the total length of pipeline. All previously mentioned methods are named as pressure-wave analysis method (PWAD).

PWAD has been widely studied and applied. However, there are two shortcomings that limited the application of PWAD in real-time leak detection. First, many methods in PWAD, such as wavelet, would produce large amounts of false alarms because they are hard to distinguish the signal changes caused by leak from noise in the pipeline system. Second, some methods in PWAD, like neural network, are not ideal for processing pressure signal in real-time just due to the time consuming fact. To overcome the two shortcomings of PWAD, we have to find the pressure abnormal signal before applying PWAD. In other words, we need to set accuracy threshold to tell pressure changes from noise while minimizing the number of the false tell. It is easy to judge this kind of change when large leakage happens, but hard to distinguish leak from noise when weak leakage happens. It

is an effective way to find leakage of pipeline transportation through detecting the changes of pressure state. However, it is hard to inspect the accurate change of pressure in practice because of the dynamic noise in pressure.

Some researches improved this situation by adding a threshold. The classic methods of threshold include fixed threshold and root-mean-square (AUTA) [16–20]. These two methods are simple and rapid, but their level of accuracy is low. Therefore, there are some cognitive methods of threshold setting been proposed. Chang et al. [21] adopted genetic algorithm to yield threshold, then applied an artificial neural network model to forecast the future trading signals. Pendharkar [22] proposed a bisection method for varying classification threshold value for cost sensitive neural network learning. Matsuda and Yamaguchi [23] proposed an adaptive threshold for joint approximate diagonalization under a probabilistic framework. This method is assumed that the errors in nondiagonal elements in cumulant matrices are given by the exponential distribution. Reference [24] used information entropy-based dynamic threshold to improve the performance of artificial immune algorithm. Reference [25] proposed threshold measurement method based on dynamic neural network models. Reference [26] used unbiased expert system to determine an accurate threshold. The previous cognitive algorithms of threshold setting are accurate but not suitable for real-time computation because of lots of computation.

To our knowledge, few researches focus on the threshold of pressure signal. And it is a key part to find the optimal threshold for the pipeline pressure signal. Therefore, the AUTA that imitates the short-term memory of human brain is proposed according to the characters of noise in pressure signal. In AUTA, the maximum values and minimum values of pressure signal in short term are recorded, which is used for threshold of next short term. In the course of recording, some parameters are auto updated that guarantee the validity of threshold. These parameters can be self-adaption even if the amplitude of noise changes because they are chosen by the noise frequency. The AUTA can adapt the change of noise rapidly and accuracy, which is suitable for dealing real-time pressure signal.

## 2. Problem Descriptions

Typically, threshold of pressure signal is fixed value. Due to the changeable state of pressure, the fixed threshold is hard to approximate/stand for the actual situation long-timely and accurately. Figures 1 and 2 are the pipeline pressure at stopping operating condition and running operating condition, respectively. We apply the standard deviation to characterize the data of the previous two cases. The results are  $2.1548e-004$  and  $6.1962e-004$ , respectively, which represents the noise level. In other words, the noise level at running operating condition is almost trebled (the actual is 2.9 times) than that at stopping operating condition. Besides, when the working conditions change, the noise also varies.

Therefore, judging pressure singularities at different work conditions with a fixed threshold will lead to mistaken

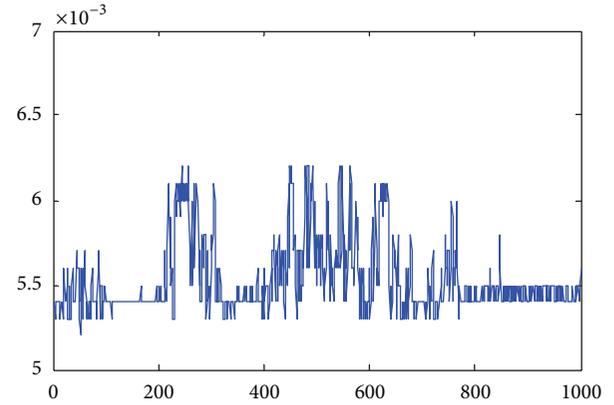


FIGURE 1: Pressure data of transportation pipeline at stop operational state.

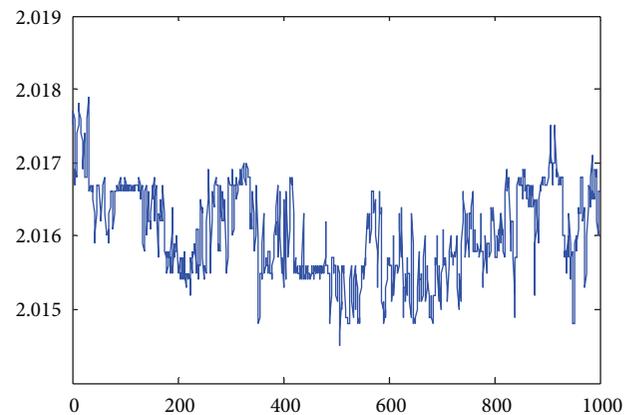


FIGURE 2: Pressure data of transportation pipeline at running operational state.

judgment or decreased judgment sensitivity, which makes it difficult for the following pressure data to be processed. If we can track pressure noise level real time by some means and calculate the dynamic threshold along with it, then we can solve the drawback which the fixed threshold brings.

## 3. Auto-Update Algorithm for Singularity Threshold of Pressure

Based on the analysis of large amount of pressure data, we found that the threshold of pressure signal is only relevant to previous pressure data. Therefore, we designed AUTA that could imitate the mechanism of cognizance and short-term memory of human brain.

The AUTA is described as follows.

*Step 1.* (1) Initialize the pressure threshold  $E$ . The threshold  $E$  is dynamic, which can update automatically in real time. A proper initial value of  $E$  can eliminate the impact of noise on the threshold to a certain degree. The physical meaning of  $E$  indicates the peak of a period of pressure data. The initial value of  $E$  should be set a little larger. Otherwise, it will cause false judgment of pressure state, which may lead to failure

application of the later algorithm. During the operation of AUTA,  $E$  will decrease rapidly. It is usually set to 1% of the amplitude of the normal pressure. For example, if the normal pressure amplitude is 1 Mpa, then  $E$  should be set to 0.01 Mpa.

- (2) Initialize  $S$ .  $S$  controls the switch of AUTA between stable and unstable pressure state.
- (3) Set two counters,  $c1$  and  $c2$ .  $c1$  is the counter of stable state of pressure signal, and the initial value is set to 0;  $c2$  is the counter of unstable state of pressure signal, and the initial value is also set to 0. Generally, the sampling frequency of pressure signal is 1 Hz, and we set 1 second as the count interval for  $c1$  and  $c2$ .

*Step 2.* Calculate the absolute difference of the current moment  $E$  with that of previous moment to judge whether the pressure is stable. When the pressure is stable, the noise in pressure is small and stable. In this case, fast method is chosen to obtain the real-time threshold. Otherwise, we need more operations to avoid the wrong setting of threshold value. The details are as follows.

Suppose that  $X = \{x(n - k + 1), x(n - 1), \dots, x(n)\}$  is pressure data used to judge the pressure state, where  $x(n)$  represents data values of the current moment,  $x(n - 1)$  represents data values of previous moment, and  $k$  is the length of the data segment.

Then,

$$\delta = |\bar{X}_1 - \bar{X}_0|, \quad (1)$$

where

$$\bar{X}_0 = \frac{x(n) + x(n - 1) + \dots + x(n - m + 1)}{k/2} \quad (2)$$

represents average of  $k/2$  data at the end of the data segment  $X$ ;

$$\bar{X}_1 = \frac{x(n - k + 1) + x(n - k + 2) + \dots + x(n - k + m)}{k/2} \quad (3)$$

represents average of  $k/2$  data at the beginning of the data segment  $X$ .

If  $d > S$ , it means that pressure is unstable; go to Step 3.

If  $d < S$ , it means that pressure is stable; go to Step 6.

*Step 3.* Empty all data of  $X$ , and set counter  $c1 = 0$ . The reason to do so is that when the switch from stable state to unstable state happens, the existed data segment  $X$  will result in abnormal larger threshold value which will make  $E$  fail to adjust itself with the noise change.

Then, record unstable pressure data in another data-array  $Y$ , and counter  $c2$  is the length of  $Y$ .

*Step 4.* If  $c2$  does not reach the setting value yet, go to Step 2. The larger setting of  $c2$  will lose the real-time capability and the smaller setting of  $c2$  will lose the learning capability. The setting procedure of  $c2$  will be discussed in detail in Section 4.

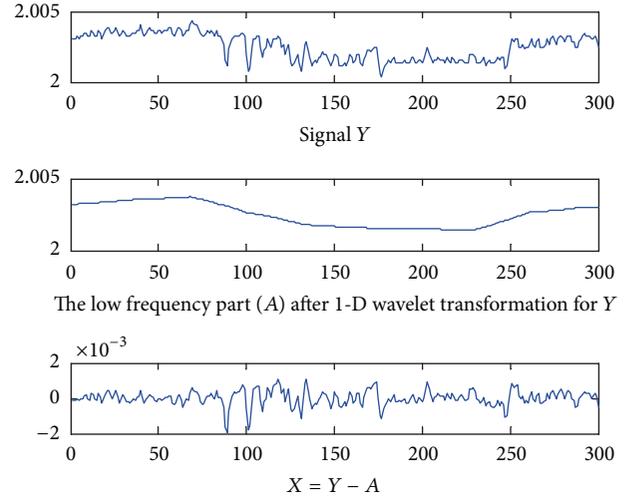


FIGURE 3: The calculating procedure of noise information ( $X$ ).

If  $c2$  reaches the setting value, we transform data-array  $Y$  by discrete-wavelet transformation and put low frequency information of the transformation into array  $A$ . Then, let  $X = Y - A$ , where  $X$  is noise information. Figure 3 shows the relations among  $Y$ ,  $A$ ,  $X$ , where the mother wavelet is “db3”  $A$  is the low frequency information after five layers decomposition. The procedure is to give the low frequency approximation of the unstable pressure by wavelet. Then, go to Step 5.

*Step 5.* Calculate  $P$  which is the difference of maximum and minimum values of  $X$ ; namely,  $P = \max(X) - \min(X)$ . If  $P > E$ , then let  $E = P$ , and  $c2 = 0$ . Go to Step 2.

When the pressure is unstable, stop calculating  $P$ . Only after  $X$  is filled with data or unstable state is over,  $P$  is calculated. This is because the wavelet algorithm needs a segment of data; otherwise, frequent calculation will decrease the real-time capability of calculating threshold.

*Step 6.* Calculate the other extreme value  $p$  ( $p = \max(X) - \min(X)$ ) of  $X$ , then compare  $p$  with the existing threshold  $P$ . If  $P < p$ , replace  $P$  by  $p$ ; Conversely, do nothing.

The reason for calculating  $p$  firstly rather than  $P$  is that can improve timeliness of acquiring the threshold in stable state.

*Step 7.* Determine whether  $c1$  reaches the setting value. When the update-time arrives, then the algorithm assigns  $P$  to  $E$  and clear the counter  $c1$  ( $c1 = 0$ ), then go back to Step 2.

$c2$  is the time counter that imitates the recognition and memory of human beings. AUTA will find the extreme value in  $c2$  and set it as the new threshold meanwhile disposing the old one.

The overall flowchart of the algorithm is shown in Figure 4.

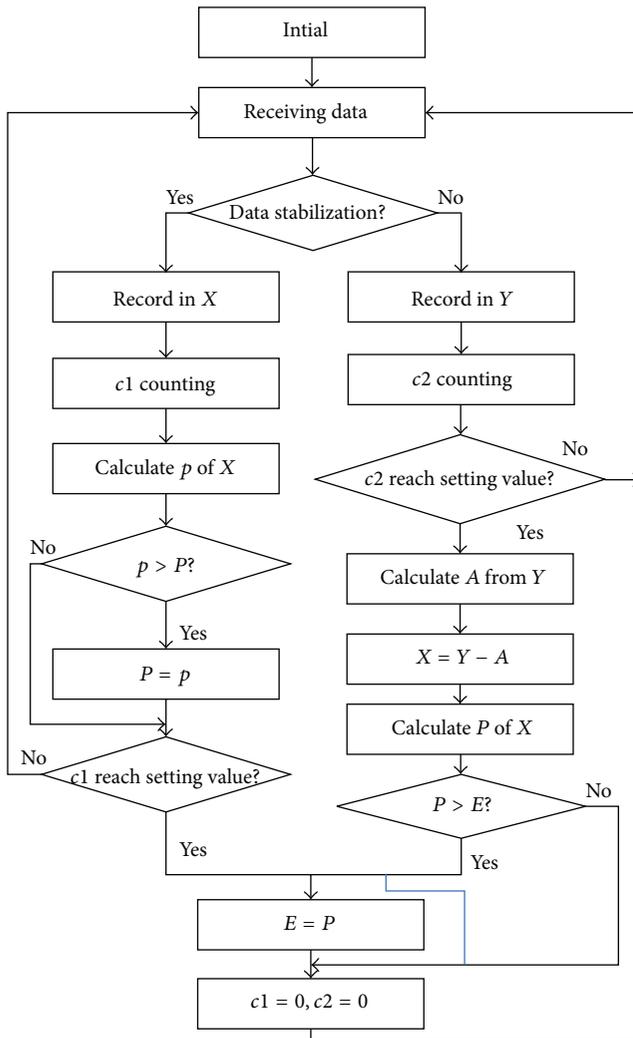


FIGURE 4: Flow chart of auto-update of threshold of pressure data.

#### 4. Settings of Algorithm Parameters

The parameters of AUTA include  $S$ ,  $k$ , counter parameters, and the boundary of  $E$  mentioned in Section 2. These four parameters play a key role in calculating pressure threshold accurately. Next, we will discuss how to select values of these parameters.

**4.1. Parameter  $S$ .**  $S$  can judge whether the operation of pipeline is adjusting. When operation of pipeline is adjusting, AUTA will not calculate threshold value. Otherwise, threshold deviating from real value will lead to omission of abnormal pressure. The effectiveness of  $S$  on AUTA can be demonstrated by a simulation. Figures 5 and 6 show the dynamic threshold curves of normal pressure data and abnormal pressure data as  $S = 0.001$ ,  $S = 0.005$ , and  $S = 0.01$ , respectively. The average values of the corresponding threshold are 0.0043, 0.0054, and 0.0057, respectively. From the figure, we can see that when  $S = 0.001$ , the threshold update automatically but slowly, and it is difficult to track

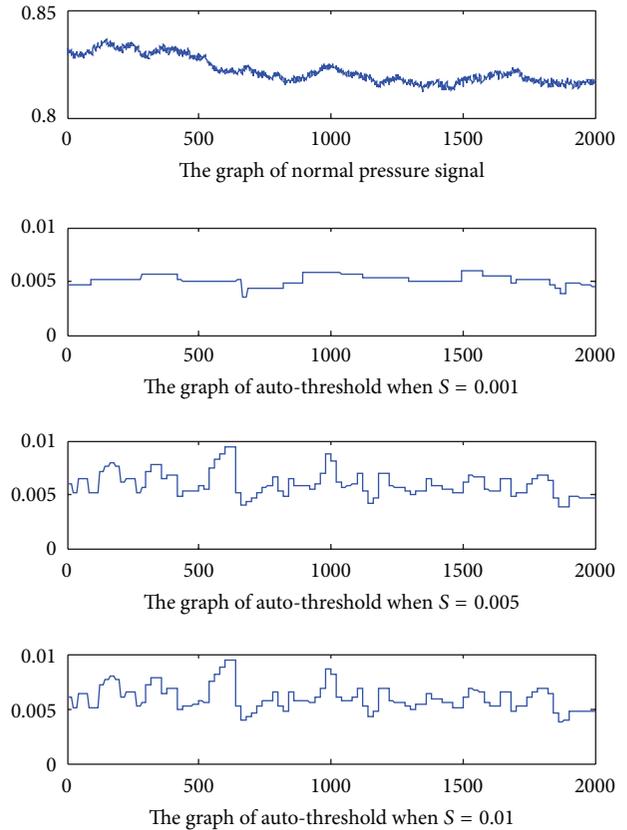


FIGURE 5: The comparison of different  $S$  at normal pressure state.

changes of the threshold in real time; when  $S = 0.01$ , real-time tracking performs well, but it leads to a bigger average threshold and sensitivity decrease; when  $S = 0.005$ , real-time tracking capability and sensitivity of the threshold are both in a better level. We found that it is ideal to set the value to 0.5% of the average of a stable pressure from repetitious times of experiments. Therefore, during the practical application,  $S$  is often set to 0.5% of the normal pressure.

**4.2. The Length of Data Segment  $k$ .**  $k$  controls the length of the data segment  $X$ . The length of the  $k/2$  data can judge whether the pressure is adjusting or not; so, its value should be the least common multiple of the contained noise cycle. As in practical applications, it is difficult to accurately determine all of the frequency of noise; so, the value is equal to the maximum cycle length of the noise. In actual situations, the measured cycle noise is 0.01 Hz in frequency, its cycle period is 100 seconds, and then we have  $k/2 = 100$ , and  $k = 200$ .

**4.3. The Setting of the Counters  $c1$  and  $c2$ .** Counter selection should also comply with certain rules. The pressure recorded by the counters can be real-time response to changes of the pressure state on the one hand. On the other hand, it would not update the threshold frequently as far as possible, otherwise, it will increase the amount of computation of AUTA. Then, the value should be the greatest common divisor of all noise cycle to meet the previous conditions; In practice, it is

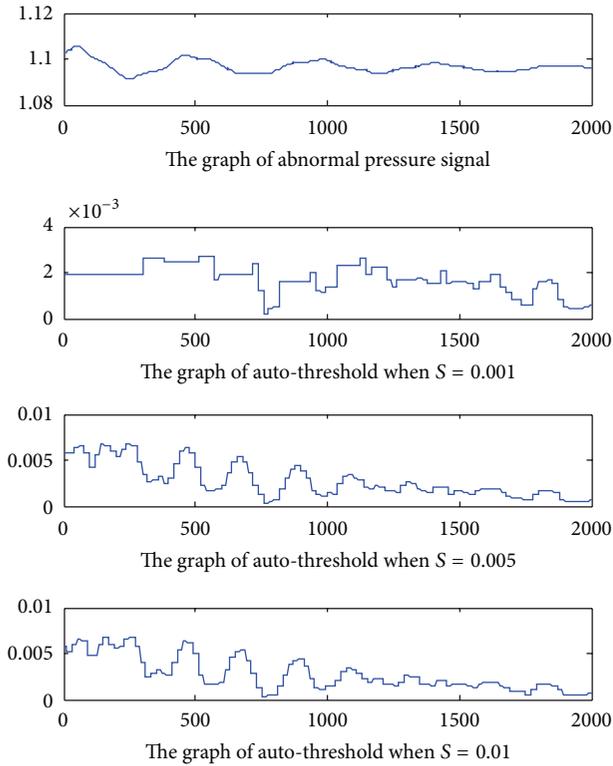


FIGURE 6: The comparison of different S at abnormal pressure state.

difficult to find all the periodic noise as mentioned earlier. Figure 7 depicts the corresponding pressure threshold curves when tested by different counter  $c1, c2$  values. The simulation figure shows that the average threshold rises at the beginning and declines when the setting value is set to be more than 20 seconds. Based on actual data test, it is appropriate to set the counter setting value to 20 seconds.

**4.4. Threshold E.** Pressure threshold  $E$  can automatically update according to the characters of pressure. However, in order to prevent incapable-receiving data or overlooking the fluctuation of pressure,  $E$  must automatically update within a certain range. For example, if no pressure data was acquired, then it will make  $E$  equal to 0 which will cause AUTA generating false judgments. If data is acquired with an exceptional pressure, then it will make  $E$  larger than its real value, which will cause omission within a period of time. Therefore, the boundary of  $E$  is set to 0.1%–1% of the normal pressure in this paper. For instance, if normal pressure is 1 Mpa,  $E$  will range from 0.001 to 0.01.

### 5. Simulation and Validation

**5.1. Performance of Different Methods.** A real pressure dataset was used to verify the performance of the fixed threshold, RMS, and AUTA. The value of fixed threshold was set as 1.03. RMS shared the same architecture with AUTA. The initialization values of AUTA, and RMS were  $S = 0.0045$ ;  $k = 200$  and the counters  $c1$  and  $c2$  were both set to 20

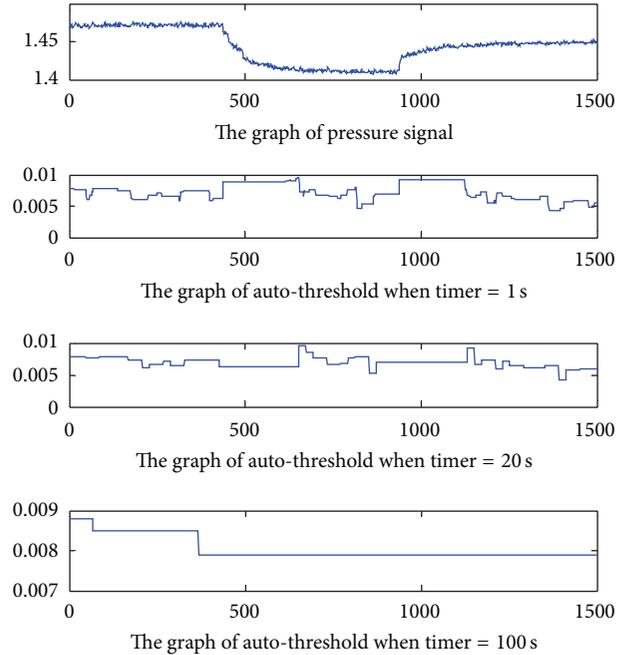


FIGURE 7: The comparison at different counter values.

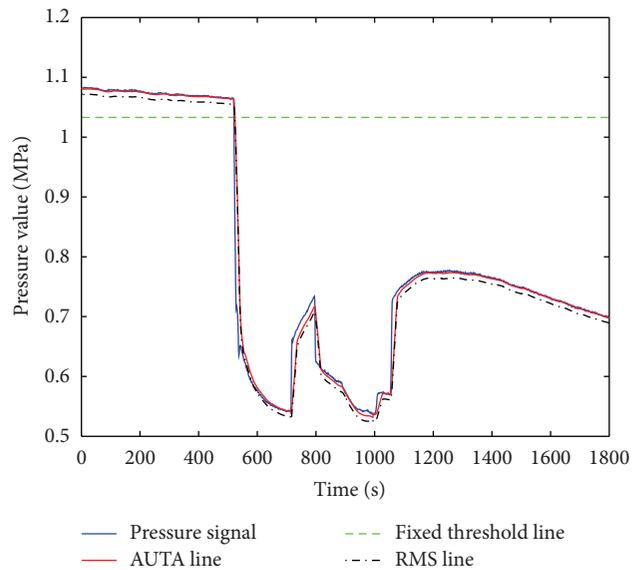


FIGURE 8: The comparison of performance of different methods.

seconds, and boundary of  $E$  ranges from 0.0001 to 0.01. The result is shown in Figure 8. We notice that the fixed threshold algorithm cannot catch the first two pressure singularity. With cost of sensitivities, fixed threshold algorithm can catch all the abnormal points by decreasing the fixed value which is not appropriate when the pressure changes frequently and drastically. The standard deviation of RMS is 0.0021, and its variance is 0.0029. The standard deviation of AUTA is 0.0017, and its variance is 0.0025. The time is 0.2 second for RMS and 0.11 second for AUTA, respectively. Thus, we know that AUTA is more accurate and faster.

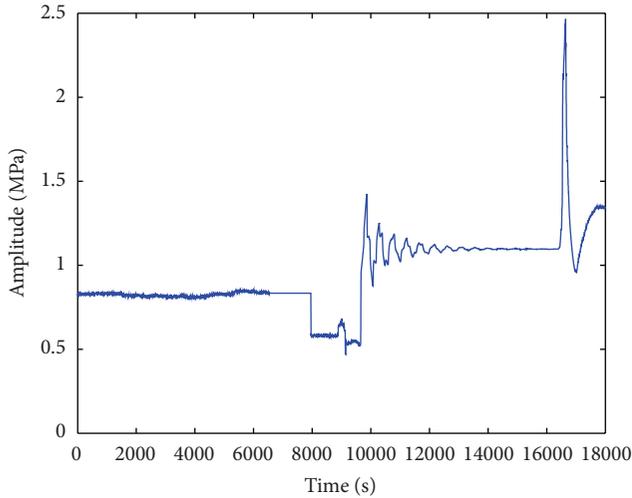


FIGURE 9: The tendency chart of pressure signal which is used to simulate.

TABLE 1: Performance of different methods without miss judgements.

Items	Fixed	RMS	AUTA
Time consumed (S)	0.501	10.935	1.205
False reporting	68	18	5
Average deviation	$1.0 \times 10^{-3}$	$3.5 \times 10^{-3}$	$6.4 \times 10^{-3}$
Standard deviation	$1.9 \times 10^{-3}$	$8.6 \times 10^{-4}$	$7.5 \times 10^{-4}$
$c1$ and $c2$	NA	5	18

5.2. *Simulation without Miss Judgements.* We chose a large scale pressure signal from product pipeline in North China to test the performance differences among these methods, in which the sampling frequency is 1 Hz and the number of singularity is 34. The pressure signal was plot as Figure 9.

Set the parameters to ensure there was no miss judgement in AUTA, RMS, and fixed threshold. The maximum threshold value for fixed threshold was 0.1 in step of 0.0001. The suitable value of fixed threshold was 0.0012. The initialization values of AUTA and RMS were  $S = 0.0045$ ;  $k = 200$  and the counters  $c1$  and  $c2$  were both set to 20 seconds, boundary of  $E$  ranged from 0.00085 to 0.0085, reducing the counter value in step of 1 second until all the singularity was found out. Table 1 shows the time of the three algorithms consumed at the state of all parameters optimized. The false reporting indicates the number of false judgements. The average deviation indicates the average deviation of every methods. The results show that fixed threshold can not be put into application because of its 68 times of false judgements, when compared with 18 times in RMS and 5 times in AUTA. And the rate of AUTA is only 12.82%. The test shows that RMS and fixed threshold value could not follow with the trend of signals by decreasing the threshold value. The computation complexity of AUTA is more than fixed threshold and much less than RMS.

5.3. *Simulation without False Judgements.* This simulation used the same data as Section 5.2. First, search the parameters

TABLE 2: Performance of different methods without false judgements.

Items	Fixed	RMS	AUTA
Time consumed (S)	0.523	2.46	1.18
False reporting	17	7	2
Average deviation	$4.9 \times 10^{-2}$	$7.4 \times 10^{-3}$	$6.9 \times 10^{-3}$
Standard deviation	$9.7 \times 10^{-3}$	$1.0 \times 10^{-3}$	$7.5 \times 10^{-4}$
$c1$ and $c2$	NA	24	20

to ensure there was no false judgement in AUTA, RMS, and fixed threshold. The initialization values of AUTA and RMS were  $S = 0.0045$ ;  $k = 200$  and the counters  $c1$  and  $c2$  were both set to 20 seconds, and boundary of  $E$  ranged from 0.00085 to 0.0085. Record the counter value in step of 1 second before the first false alarm coming. The results are shown in Table 2. From this simulation, AUTA performs best with the rate of false alarm of 6.25 while that of fixed threshold is up to 50. The consumed time relation is fixed threshold  $>$  AUTA  $>$  RMS threshold.

5.4. *Application.* In May 2012, AUTA has been applied in the oil pipeline network in North China for inspecting leak. The network consists of 23 segments and a total length of 2100 kilometers. The running pressure value ranged from 0.1 Mpa to 8 Mpa. During application of four months, the accuracy of the alarm reports of the detection system increases 17% by AUTA.

## 6. Conclusion

This paper proposes an auto-update algorithm for abnormal pressure threshold, called AUTA, which divides the calculation of the threshold into stable and unstable cases to ensure timeliness and sensitivity of the threshold. Then, the selection principles of the various parameters in AUTA are analyzed. The simulations show that this algorithm can provide an effective and real-time threshold for judgment of abnormal pressure. In future work, we can further apply other methods, like fuzzy [27, 28], ELM [29, 30], SVM [31, 32], and fault diagnosis technologies [33], to diagnose and locate leak points.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (61104021, 61034005, and 61203086), the National High Technology Research and Development Program of China (2012AA040104), and the Fundamental Research Funds for the Central Universities of China (N100304007).

## References

- [1] M. Pu, "Our new progress of the fluid in 2009," *International Petroleum Economics*, vol. 3, pp. 14–20, 2010.

- [2] Z. Huaguang and Q. Yongving, "Modeling, identification, and control of a class of nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 2, pp. 349–354, 2001.
- [3] H. Zhang, Y. Luo, and D. Liu, "Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints," *IEEE Transactions on Neural Networks*, vol. 20, no. 9, pp. 1490–1503, 2009.
- [4] H. Zhang, Q. Gong, and Y. Wang, "Delay-dependent robust  $H_\infty$  control for uncertain fuzzy hyperbolic systems with multiple delays," *Progress in Natural Science*, vol. 18, no. 1, pp. 97–104, 2008.
- [5] H. Zhang and Y. Wang, "Guaranteed cost control for uncertain stochastic fuzzy systems with time delay," *Progress in Natural Science*, vol. 17, no. 1, pp. 95–101, 2007.
- [6] H. Zhang, M. Li, J. Yang, and D. Yang, "Fuzzy model-based robust networked control for a class of nonlinear systems," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 39, no. 2, pp. 437–447, 2009.
- [7] P. K. Wong, Q. S. Xu, C. M. Vong, and H. C. Wong, "Rate-dependent hysteresis modeling and control of a piezostage using on-line support vector machine and relevance vector machine," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 4, pp. 1988–2001, 2012.
- [8] M. Daneti, "A practical preprocessing treatment for pipeline leak locating improving," in *Proceedings of the IEEE International Conference on Digital Object Identifier*, pp. 9–12, September 2008.
- [9] M. Ferrante and B. Brunone, "Pipe system diagnosis and leak detection by unsteady-state tests. 2. Wavelet analysis," *Advances in Water Resources*, vol. 26, no. 1, pp. 107–116, 2003.
- [10] C. Ge, H. Yang, H. Ye, and G. Wang, "A fast leak locating method based on wavelet transform," *Tsinghua Science and Technology*, vol. 14, no. 5, pp. 551–555, 2009.
- [11] V. Sima and M. Behzad, "Multiphase pipeline leak detection based on fuzzy Classification," *AIP Conference Proceedings*, vol. 1159, no. 1, pp. 72–80, 2009.
- [12] J. Liu, H. Zhang, J. Feng, and H. Yue, "A new fault detection and diagnosis method for oil pipeline based on rough set and neural network," *Lecture Notes in Computer Science*, vol. 4493, no. 3, pp. 561–569, 2007.
- [13] D. L. Xu, J. Liu, J. B. Yang et al., "Inference and learning methodology of belief-rule-based expert system for pipeline leak detection," *Expert Systems with Applications*, vol. 32, no. 1, pp. 103–113, 2007.
- [14] H. V. da Silva, C. K. Morooka, I. R. Guilherme, T. C. da Fonseca, and J. R. P. Mendes, "Leak detection in petroleum pipelines using a fuzzy system," *Journal of Petroleum Science and Engineering*, vol. 49, no. 3–4, pp. 223–238, 2005.
- [15] L. B. Zhang, X. Y. Qin, Z. H. Wang, and W. Liang, "Designing a reliable leak detection system for West Products Pipeline," *Journal of Loss Prevention in the Process Industries*, vol. 22, no. 6, pp. 981–989, 2009.
- [16] H. L. Chan, M. A. Lin, T. Wu, S. T. Lee, Y. T. Tsai, and P. K. Chao, "Detection of neuronal spikes using an adaptive threshold based on the max-min spread sorting method," *Journal of Neuroscience Methods*, vol. 172, no. 1, pp. 112–121, 2008.
- [17] P. H. Thakur, H. Lu, S. S. Hsiao, and K. O. Johnson, "Automated optimal detection and classification of neural action potentials in extra-cellular recordings," *Journal of Neuroscience Methods*, vol. 162, no. 1–2, pp. 364–376, 2007.
- [18] K. S. Guillory and R. A. Normann, "A 100-channel system for real time detection and storage of extracellular spike waveforms," *Journal of Neuroscience Methods*, vol. 91, no. 1–2, pp. 21–29, 1999.
- [19] P. T. Watkins, G. Santhanam, K. V. Shenoy, and R. R. Harrison, "Validation of adaptive threshold spike detector for neural recording," in *Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '04)*, pp. 4079–4082, September 2004.
- [20] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, 1995.
- [21] P. C. Chang, T. W. Liao, J. J. Lin, and C. Y. Fan, "A dynamic threshold decision system for stock trading signal detection," *Applied Soft Computing Journal*, vol. 11, no. 5, pp. 3998–4010, 2011.
- [22] P. C. Pendharkar, "A threshold varying bisection method for cost sensitive learning in neural networks," *Expert Systems with Applications*, vol. 34, no. 2, pp. 1456–1464, 2008.
- [23] Y. Matsuda and K. Yamaguchi, "An adaptive threshold in joint approximate diagonalization by assuming exponentially distributed errors," *Neurocomputing*, vol. 74, no. 11, pp. 1994–2001, 2011.
- [24] Q. Zhang, X. Xu, and Y. C. Liang, "An improved artificial immune algorithm with a dynamic threshold," *Journal of Bionic Engineering*, vol. 3, no. 2, pp. 93–97, 2006.
- [25] J. V. Ringwood, "Anaerobic threshold measurement using dynamic neural network models," *Computers in Biology and Medicine*, vol. 29, no. 4, pp. 259–271, 1999.
- [26] A. Botros, B. van Dijk, and M. Killian, "AutoNRT: an automated system that measures ECAP thresholds with the Nucleus Freedom cochlear implant via machine intelligence," *Artificial Intelligence in Medicine*, vol. 40, no. 1, pp. 15–28, 2007.
- [27] J. H. Zhai, "Fuzzy decision tree based on fuzzy-rough technique," *Soft Computing*, vol. 15, no. 6, pp. 1087–1096, 2011.
- [28] M. Z. Jahromi and M. Taheri, "A proposed method for learning rule weights in fuzzy rule-based classification systems," *Fuzzy Sets and Systems*, vol. 159, no. 4, pp. 449–459, 2008.
- [29] J. H. Zhai, H. Y. Xu, and X. Z. Wang, "Dynamic ensemble extreme learning machine based on sample entropy," *Soft Computing*, vol. 16, no. 9, pp. 1493–1502, 2012.
- [30] W. W. Zong, G. B. Huang, and Y. Q. Chen, "Weighted extreme learning machine for imbalance learning," *Neurocomputing*, vol. 101, pp. 229–242, 2013.
- [31] P. K. Wong, H. C. Wong, and C. M. Vong, "Online time-sequence incremental and decremental least squares support vector machines for engine air-ratio prediction," *International Journal of Engine Research*, vol. 13, no. 1, pp. 28–40, 2012.
- [32] C. Sudheer, N. Anand, B. K. Panigrahi, and S. Mathur, "Stream-flow forecasting by SVM with quantum behaved particle swarm optimization," *Neurocomputing*, vol. 101, pp. 18–23, 2013.
- [33] C. M. Vong, P. K. Wong, and W. F. Ip, "A new framework of simultaneous-fault diagnosis using pairwise probabilistic multi-label classification for time-dependent patterns," *IEEE Transactions on Industrial Electronics*, 2013.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

