

RESEARCH

Open Access



A variable step size least mean p -power adaptive filtering algorithm based on multi-moment error fusion

Boyu Zhu¹, Biao Wang^{1*} , Banggui Cai¹, Yunan Zhu¹, Peng Chao¹ and Zide Fang¹

*Correspondence:
wangbiao@just.edu.cn

¹ Ocean College, Jiangsu
University of Science
and Technology,
Zhenjiang 212100, Jiangsu
Province, China

Abstract

For the channel estimation problem under α -stable distributed impulse interference, the traditional fixed-step adaptive filtering cannot satisfy the fast convergence speed and low steady-state error at the same time, whereas the variable-step method is able to effectively solve this contradiction. This paper proposes an improved variable step-size least mean p -power adaptive algorithm that offers good robustness against impulsive noise. The proposed algorithm takes into account the linkage between the errors and uses the adjustment of the step size based on the errors of the current moment and the previous k moments, thus overcoming the problems of poor anti-noise performance and large steady-state fluctuations of the fixed-step size algorithm. This algorithm ensures that the step size does not change abruptly when the system is disturbed by impulse noise and can achieve a lower steady-state error. The simulation results show that the algorithm has better convergence than the traditional fixed step-size algorithm and the existing variable step-size algorithm under the interference of impulsive noise.

Keywords: Impulsive noise, α -stable distribution, Adaptive filtering algorithm, Least mean p -power (LMP), Multi-moment error

1 Introduction

In many cases where adaptive algorithms are used for channel estimation, fixed step sizes are often designed to facilitate engineering applications. Under this case, a compromise is often required between fast adaptation to overcome input variations and slow adaptation to cope with system noise in order to obtain the best performance [1]. For this problem, a large strand of research has made improvements on the basis of the traditional fixed step-size least mean square (LMS) adaptive algorithm, so that the algorithm can change the step size with the change of the mean square error, thus solving the problem faced previously and achieving better results [2, 3]. However, the above methods are applied in the environment of Gaussian noise, but in practical engineering applications the system is often disturbed by a large amount of non-Gaussian noise. The literature has shown that the most coastal areas around the world are flooded with impulsive interference in the hydroacoustic channel due to the presence of a large number of drum

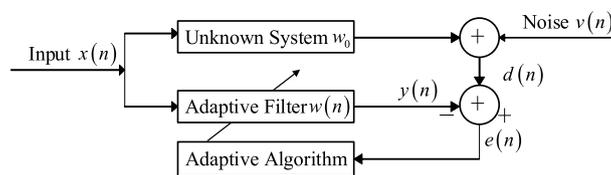


Fig. 1 Adaptive filter system block

shrimp family organisms (which emit strong pulses by opening and closing the knuckles of their large cheeks) [4]. The traditional LMS algorithm only takes into account the second-order statistics of data, and so it works well under Gaussian noise interference but performs generally when it encounters non-Gaussian noise such as impulse noise [5]. To overcome this problem, researchers have proposed the least mean p -power (LMP) algorithm by using lower-order statistics of the data to replace the second-order statistics [6, 7]. LMP algorithm is a nonlinear filtering algorithm that uses generalized correntropy as the cost function. Compared with traditional adaptive filtering algorithms, LMP algorithm has better robustness in dealing with impulsive interference, sparse channels or abrupt channel changes. Therefore, this paper improves the LMP algorithm when estimating the underwater acoustic channel under impulse interference. The system block diagram of the general adaptive filtering algorithm to implement channel estimation is shown in Fig. 1.

2 Proposed algorithm

The step size of the traditional LMP algorithm is a fixed value, which cannot meet the requirements of faster convergence speed and lower steady-state error at the same time. Therefore, it is necessary to speed up convergence speed in the initial stage of adaptation and reduce system error after reaching a steady state. Variable step size is an effective method to solve the algorithm's convergence speed and steady-state error, but the traditional variable step-size algorithm only considers the current update error and ignores the influence of errors at other moments. In an actual simulation, it is found that although the convergence effect is better than those of other algorithms, it does not meet expectations. Therefore, based on the traditional variable step-size LMP (VSS-LMP) algorithm, this paper considers the influence of the error of the previous several times on the current state and constructs the error function about the previous k times as the step-size update basis of the adaptive algorithm. The block diagram of proposed algorithm appears in Fig. 2.

One effective way to improve the convergence speed is to design a formula with variable step size. The error can be used to update the step size. Thus, a larger step size can be given by the variable step-size function when the error between the desired signal and the output signal is large, which can accelerate the convergence process. In the presence of impulsive noise disturbance that produces results far from the actual situation, the connection between the previous and the generated step size is hard to establish because of the large error signal.

As such, some scholars construct a deformation Gaussian function to resolve the effect of convergence due to the sudden change of error [8]. The formula is:

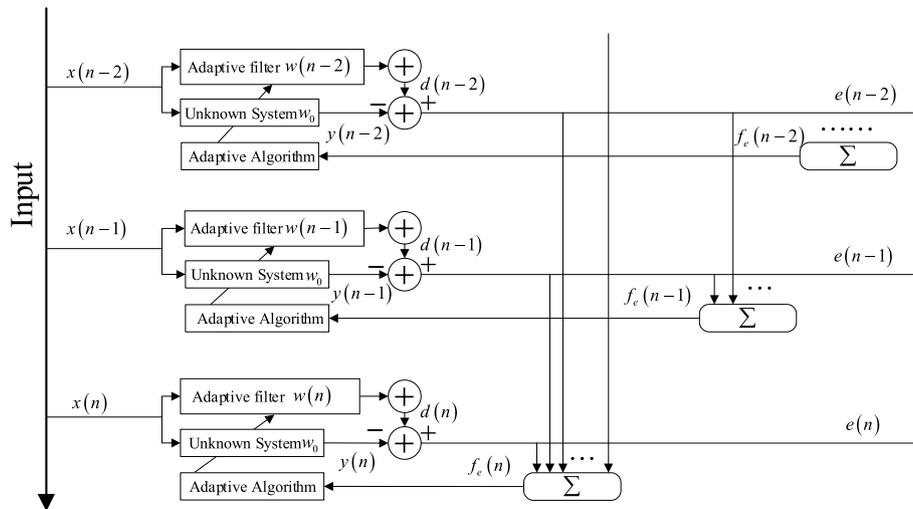


Fig. 2 System block of proposed algorithm

$$\mu(e(n + 1)) = \theta\mu[e(n)] + (1 - \theta)\alpha|e(n + 1)|^2 e^{-\beta|e(n+1)|^2} \tag{1}$$

This is a step-update function, thus achieving anti-pulse interference. However, when studying the algorithm, it is found that the effect is not ideal in the case of large impulsive noise. To further improve its conversion effect, we enhance the algorithm and propose an Improved Variable Step-Size Least Mean p -Power (IVSS-LMP) algorithm, by introducing the equation:

$$f_e(n) = k_{ap} \times \sqrt{\sum_{i=n-k+1}^n |e(i)|^2} \tag{2}$$

where $k_{ap} > 0$. We use this formula to correct the step size, which makes the algorithm give a better convergence effect. Since the error of the previous k terms is introduced, the value of the parameter k cannot be large in order to ensure a real-time update. Hence, choosing a suitable value of the parameter k is crucial for this algorithm.

The weighted moving average method is often used in mathematics to predict future values. Considering the correlation between steps, we also use the weighted moving average method to build the performance stable. The formula is

$$\mu(f_e(n + 1)) = \theta\mu[f_e(n)] + (1 - \theta)\alpha f_e(n + 1) e^{-\beta f_e(n+1)} \tag{3}$$

This weighs the steady-state misalignment and convergence speed. The pseudo-code process of proposed algorithm is shown in Table 1.

3 Convergence analysis

To prove the stability of the algorithm, it is necessary to analyze the convergence score of the algorithm by first defining the initial conditions, $f_e(0) = 0$ and $\mu(0) = 0$. We modify Eqs. (2) and (3) to:

Table 1 The proposed algorithm

Proposed algorithm
Initialization
$w = \bar{0}$
Beginning computation
Iterate for $n > k$
$d(n) = h^T x(n) + v(n)$
$e(n) = d(n) - w^T x(n)$
$f_e(n) = k_{ap} \sqrt{\sum_{i=n-k+1}^n e(i) ^2}$
$\mu(f_e(n+1)) = \beta \mu[f_e(n)] + (1 - \beta) \alpha f_e(n+1) e^{-\beta f_e(n+1)}$
$w(n+1) = w(n) + \mu[f_e(n)] e(n) ^{p-1} \text{sgn}(n) x(n)$
end

$$\begin{aligned} \mu(f_e(n+1)) &= \theta \mu[f_e(n)] + (1 - \theta) \alpha f_e(n+1) e^{-\beta f_e(n+1)} \\ &= \sum_{i=1}^n \theta^{n-i} (1 - \theta) \alpha f_e(i) \exp(-\beta f_e(i)) + \theta^n \mu(f_e(0)) \\ &= (1 - \theta) \alpha \sum_i \theta^{n-i} f_e(i) \exp(-\beta f_e(i)) \end{aligned} \tag{4}$$

Here,

$$f_e(n) = 0, \quad n < 0 \tag{5}$$

$$f_e(i) \exp(-\beta f_e(i)) = \frac{\beta f_e(i) \exp(-\beta f_e(i))}{\beta} \tag{6}$$

We constructing the function $g(y)$:

$$g(y) = ye^{-y} \tag{7}$$

By calculating the derivative of this function, we can see that the function has a maximum value e^{-1} at $y = 1$. Therefore, Eq. (6) can be written as:

$$f_e(i) \exp(-\beta f_e(i)) = \frac{\beta f_e(i) \exp(-\beta f_e(i))}{\beta} \leq \frac{e^{-1}}{\beta} \tag{8}$$

Bringing Eq. (8) into (4), we get:

$$\begin{aligned} \mu(f_e(n+1)) &= (1 - \theta) \alpha \sum_{i=1}^n \theta^{n-i} f_e(i) \exp(-\beta f_e(i)) \\ &\leq (1 - \theta) \alpha \sum_{i=1}^n \theta^{n-i} \frac{e^{-1}}{\beta} = \frac{\alpha(1 - \theta)e^{-1}}{\beta} \cdot \frac{1 - \theta^n}{1 - \theta} \end{aligned} \tag{9}$$

When $0 < \theta < 1$, we get the following inequality:

$$\mu(f_e(n+1)) \leq \frac{\alpha e^{-1}}{\beta} \tag{10}$$

The formula for updating the weights of the IVSS-LMP algorithm is:

$$w(n+1) = w(n) + \mu[f_e(n)]|e(n)|^{p-2}e(n)x(n) \tag{11}$$

Here, $\mu[f_e(n)]|e(n)|^{p-2}$ can be considered as the total step length of the LMS algorithm, The algorithm converges only when the step size in the LMS algorithm satisfies $0 < \mu < \frac{2}{3\text{tr}(R)}$ [9]. Therefore, the IVSS-LMP algorithm step size should satisfy:

$$0 \leq \mu[f_e(n)]|e(n)|^{p-2} \leq \frac{2}{3\text{tr}(R)} \tag{12}$$

Here, $\text{tr}(\cdot)$ denotes the trace of the matrix, $R = E[x^T(n)x(n)]$ is the auto-correlation matrix of the input signal, and $0 < p < \alpha$ (α is the parameter of impulsive noise). Thus, $|e(n)|^{p-2} \leq 1$.

Bringing Eq. (9) into (12) yields the inequality:

$$\frac{\alpha}{\beta} \leq \frac{2e}{3\text{tr}(R)} \tag{13}$$

Therefore, the proposed LMP algorithm converges when the parameters satisfy the above formula.

4 Simulation analysis

This study applies the proposed IVSS-LMP algorithm to channel estimation under α -stable distributed impulsive noise interference, where the input signal $x(n)$ obeys a Gaussian distribution with mean 0 and variance 1. The weight vector w_0 of the unknown system also obeys a Gaussian distribution with mean 0. The filter order is $M=128$, and the number of sampling points performed is 20,000 points each time. In order to reflect the performance of the algorithm in the case of abrupt changes in the system, the system is abruptly changed at 10,000 iterations (using the method of changing the weight vector to the opposite of the original). The final simulation results are then averaged over 20 Monte Carlo simulations to obtain a more easily comparable simulation [10].

To better measure the performance of the algorithm in channel estimation, this paper applies Normalized Mean Square Deviation (NMSD) of the weights to reflect the effect of the algorithm. The estimated channel parameters are compared with the original simulated channel parameters by normalizing the difference, so as to reflect the effectiveness of the algorithm more intuitively and accurately. A smaller NMSD value indicates that the convergence of the algorithm is better. The expression is:

$$\text{NMSD} = 10 \log_{10} \left[\frac{\|w_0 - w(n)\|^2}{\|w_0\|^2} \right] \tag{14}$$

In the simulation experiments, α -stable distribution is used as the impulsive noise model. The steady-state distribution noise is a class of random noise with linear spikes

and trailing effect. It does not have a uniform probability density function, but rather a unified characteristic function [11], which can be expressed as:

$$\varphi(t) = \exp \{jat - \gamma |t|^\alpha [1 + j\beta \text{sgn}(t)\omega(t, \alpha)]\} \tag{15}$$

Here, $\text{sgn}(\cdot)$ is the sign function:

$$\omega(t, \alpha) = \begin{cases} \tan(\alpha\pi)/2, & \alpha \neq 1 \\ (2/\pi) \log |t| & \alpha = 1 \end{cases} \tag{16}$$

Since impulsive noise does not have second-order statistics such as variance and correlation functions, the traditional signal-to-noise ratio function is ineffective under the α -stable distributed noise. Hence, the following equation can be used to measure the signal-to-noise ratio of impulsive noise to the useful signal [12]:

$$\text{SNR} = 10 \log (\sigma_s^2/\gamma) \tag{17}$$

Here, σ_s^2 is the variance of the input signal, and γ is the dispersion coefficient of the α -stable distribution.

4.1 Comparison with unimproved algorithm

By comparing the error functions before and after the improvement, we can visually analyze the difference between the improved algorithm and the one before the improvement. The following involves parameter θ taking the value of 0.98, parameter k_{ap} taking the value of 1, parameter α taking the value of 0.0007, parameter β taking the value of 0.004, and parameter k taking the value of 3. In order to compare the effect of two error functions more intuitively, the error function is treated logarithmically, as shown in Fig. 3.

In this figure the variance of the error function of VSS-LMP is 543.4234, and the variance of the error function of IVSS-LMP is 191.2481. From the figure we also find in the face of impulsive noise that the error function constructed by IVSS-LMP has smaller fluctuations than the direct use of error; i.e., better robustness against impulsive noise.

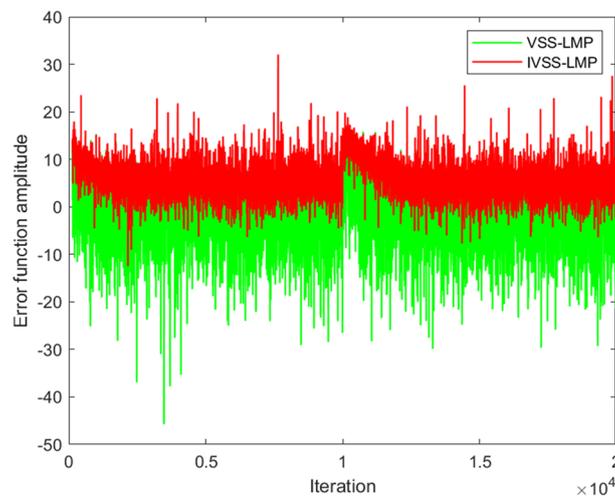


Fig. 3 Comparison of different error functions

Table 2 the parameters of different cases

Case	θ	k_{ap}	α	β	k
1	0.98	0.9	0.0007	0.004	3
2	0.9	0.8	0.0008	0.003	3
3	0.98	0.7	0.0006	0.004	4

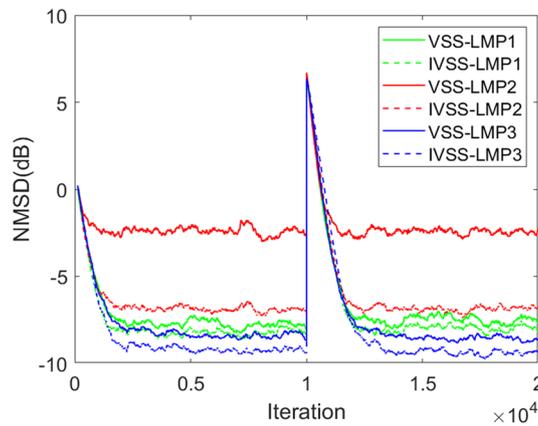


Fig. 4 Comparison of the algorithms under random parameters

Using this property, the algorithm can achieve a lower normalized mean square error at convergence.

From a comparison with the algorithm before the improvement, the following is the result of the simulation with the same parameters except for the newly added variables. Here, 123 represents the three parameter cases respectively, as shown in Table 2.

The simulation of VSS-LMP and the improved IVSS-LMP are compared according to the parameters in the table. The comparison results appear in Fig. 4.

A comparison of the performance under random parameters shows that the errors at convergence of the IVSS-LMP algorithm are lower than those of the VSS-LMP algorithm before the improvement. The remaining parameters are guaranteed to be unchanged except for the newly added parameters.

4.2 Parameter analysis

To investigate the effect of parameter k on the performance of the algorithm, parameter k_{ap} is taken as 1, smoothing factor θ is taken as 0.97, parameter α is taken as 0.001, parameter β is taken as 0.0038, and algorithm parametric number p is taken as 1.15. The above parameters are selected by trial-and-error method under the condition of step stability to obtain the best value [13]. Parameter k is taken as 2, 3, 4, and 5 respectively, in order to compare the effect of parameter k on the algorithm more intuitively. The algorithm before improvement is also added as a reference. The simulation pulsive noise parameters are $N = [1.5, 0, 0.04, 0]$, and the noise signal is shown in Fig. 5.

Under the simulation pulse noise, the algorithm performance curve is shown in Fig. 6.

Analysis of the graph shows that the NMSD values when the algorithm converges are -26.82 dB, -23.33 dB, -22.08 dB, and -20.14 dB when the values of k are taken

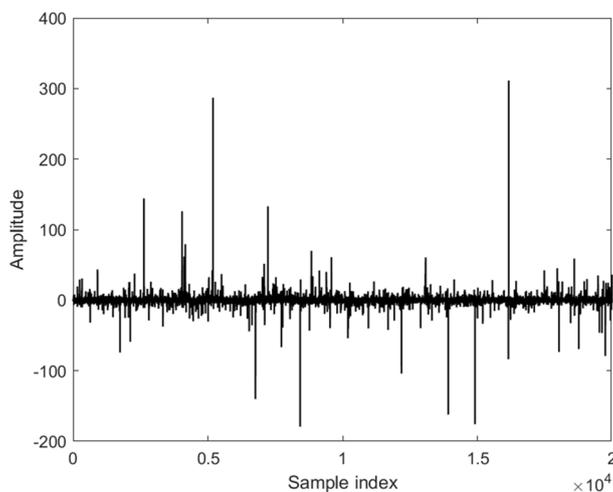


Fig. 5 Reference impulsive noise

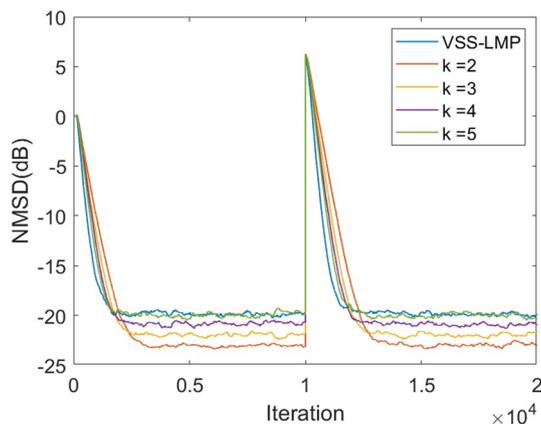


Fig. 6 Comparison under different parameter k

as 1, 2, 3, 4, and 5, respectively. Compared to VSS-LMP (i.e., when no forward multiple error correction is made, with an increasing value of k , the convergence rate gradually decreases. However, after convergence in order to achieve a lower NMSD, for the variable hydroacoustic channel environment, a low error is particularly important, and from the simulation it can be seen that the algorithm with forward error correction has a significant improvement in the convergence effect. The difference in the steady state achieved for values of k greater than 3 is not significant. Moreover, the performance of the improved algorithm decreases compared to the original algorithm for values of parameter k greater than or equal to 5. Combining the time required to reach steady state with the value of NMSD, the simulation is relatively better at parameter $k = 3$.

To research the effect of parameter θ on the performance of the algorithm, the value of parameter k is taken as 3, the value of θ is taken as 0.86, 0.87, 0.88, 0.89, 0.96, 0.97, 0.98, and 0.99, and other parameters remain constant. The performance of the algorithm is shown in Fig. 7.

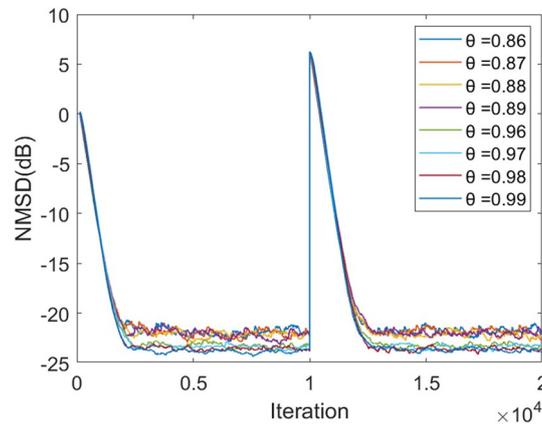


Fig. 7 Comparison under different parameter θ

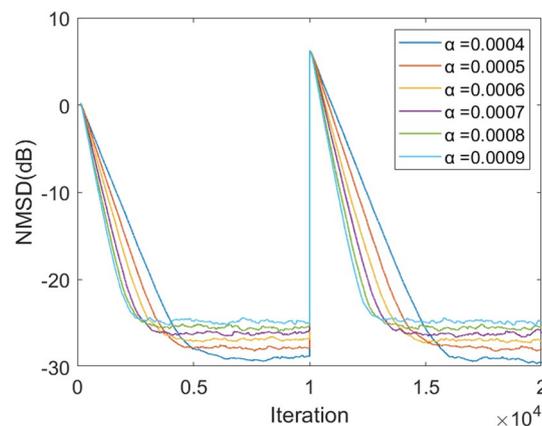


Fig. 8 Comparison under different parameter α

By analyzing the graph, we see that as the value of θ becomes larger, the time required to reach steady state increases while still being able to reach a lower steady state. The simulation figure shows that θ of the algorithm can affect system convergence. With all other parameters being the same, the convergence is better when θ is close to 1. In the case of changing only θ , the better the convergence effect is, the more times it takes to reach steady-state convergence. At $\theta=0.98$, the fluctuation of NMSD after reaching steady state is smaller, and so the value of parameter θ is taken as 0.98.

To study the effect of parameter α on the performance of the algorithm, the value of parameter θ is taken as 0.98, the value of parameter α is taken as 0.0004, 0.0005, 0.0006, 0.0007, 0.0008, and 0.0009, and other parameters remain constant. The performance of the algorithm appears as the Fig. 8.

Analyzing this figure, the value of parameter α affects convergence speed as well as the convergence effect when the steady state is reached. As the value of α increases from 0.0004 to 0.0009, the number of algorithm convergence steps are respectively 5467, 4399, 3940, 3520, 3043, and 2621, and the convergence speed keeps getting faster. After convergence of the algorithm, NMSD values are -27.43 dB, -26.11 dB, -25.45 dB, -24.71 dB, -24.08 dB, and -23.18 dB, respectively. From the data, we see that the gain is very

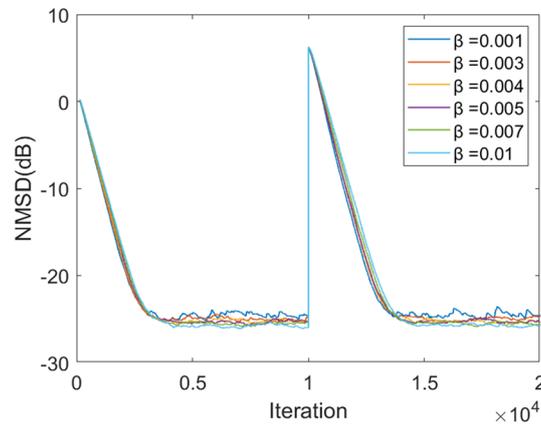


Fig. 9 Comparison under different parameter β

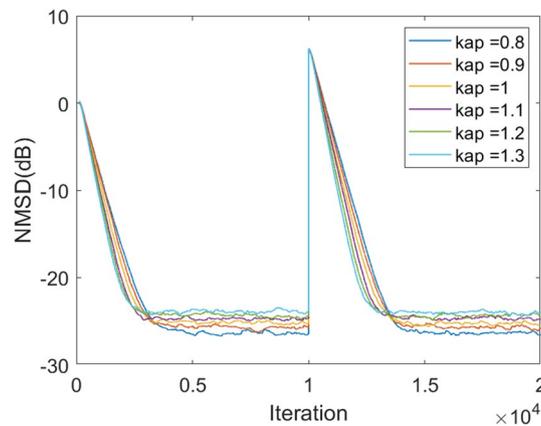


Fig. 10 Comparison under different parameter k_{ap}

high at the beginning when the value of α decreases; i.e., sacrificing less convergence speed in exchange for better convergence. However, as α decreases further, the gains start to decline, even sacrificing more than double the previous number of steps while gaining the same convergence effect. Therefore, considering the two factors, parameter α is taken as 0.0006.

To study the effect of parameter β on the performance of the algorithm, the value of parameter α is taken as 0.0007 and the value of parameter β is taken as 0.001, 0.003, 0.004, 0.005, 0.007, and 0.01. The simulation results appear as the Fig. 9.

Analyzing this figure, the steady-state fluctuation is larger when β is small. As β increases, the convergence speed decreases, but the steady-state error turns higher. The magnitude of fluctuation after reaching the steady state is integrated, and parameter β at 0.004 in this algorithm is better.

To analyze the effect of parameter k_{ap} on the algorithm, the parameter β value is 0.004 and the parameter k_{ap} values are set to 0.8, 0.9, 1.0, 1.1, 1.2, and 1.3. The simulation yields in Fig. 10.

The figure shows that as the k_{ap} value increases, the convergence speed becomes faster, the convergence error increases, and better gains can be obtained by increasing the k_{ap}

value when the k_{ap} value is low (getting faster at the expense of the same NMSD). We also find when the parameter k_{ap} value is greater than 1 that the ratio of convergence speed to the number of iterations changes rapidly. In the case of k_{ap} values of 1.1 and 1.2, the convergence effect is similar, but in the case of k_{ap} value of 1.1, there is less fluctuation after convergence. Taking this into account, the value of k_{ap} in this algorithm is 1.1.

4.3 Comparison of simulated channel algorithms

We now introduce the improved sigmoid function-based [14] into the LMP algorithm to form the sigmoid variable step-size least mean p -power (SVS-LMP) algorithm. We then add the improved inverse hyperbolic tangent function [15] into the LMP algorithm to form the inverse hyperbolic tangent variable step-size LMP (IHTVS-LMP) algorithm. Lastly, we present a normal distribution curve [16] to construct a normal distribution curve step-size LMP (NDCS-LMP) algorithm. Here, each algorithmic parameter is adjusted to the appropriate value, as shown in Table 3.

The simulation is shown in Fig. 11.

Table 3 the parameters of different algorithm under simulated channel

Algorithm	Step	α	β	θ	k_{ap}
LMS	0.005				
LMP	0.005				
SVS-LMP	$\theta \left[\frac{1}{1 + \exp(-\alpha e(n))} - 0.5 \right]$	0.4		0.009	
IHTVS-LMP	$\theta \times \operatorname{artanh}(\alpha \times e(n) ^\beta)$	0.02	0.1	0.18	
NDCS-LMP	$\theta [1 - \exp(-\alpha e(n) ^\beta)]$	0.1	2	0.007	
VSS-LMP	$\mu = \theta \mu(e(n-1)) + (1-\theta)\alpha e(n) ^2 \exp(-\beta e(n) ^2)$	0.0006	0.004	0.98	
IVSS-LMP	$f = k_{ap} \times \sqrt{\sum_{i=n-k+1}^n e(i) ^2}$ $\mu = \theta \mu f_e(n-1) + (1-\theta)\alpha f_e(n) \exp(-\beta f_e(n))$	0.0006	0.004	0.98	1.1

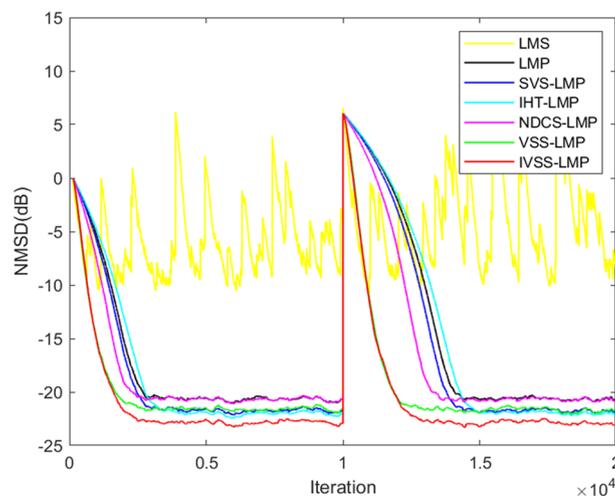


Fig. 11 Comparison of different algorithms under simulated channel

Comparing the simulation plots of different algorithms in the literature with a signal-to-noise ratio of 24 dB, the LMS algorithm suffers from poor convergence performance when faced with impulse interference. Therefore, we adopt the LMP algorithm to improve the convergence performance under impulse interference. Through comparison of the convergence performance, we show that the proposed algorithm achieves better convergence performance with the same number of iterations, and has lower steady-state error than the existing algorithms. Moreover, the proposed algorithm exhibits significant improvement in scenarios where the channel undergoes abrupt changes.

4.4 Comparison of actual channel algorithms

To further investigate the performance of each algorithm, this section applies IVSS-LMP to an actual channel for comparison. Under impulsive noise, the measured hydroacoustic channel impulse response is identified as a weight vector in this section, and the estimation performance of each algorithm is simulated. The actual measured channel impact response at a given moment in Norwegian waters [17] is shown in Fig. 12.

The impulse response of the actual hydroacoustic channel is sampled at 1 kHz, and the length of the channel is 256 ms. Thus, this section selects it as the unknown system response to be identified. The number of sampling points is 2×10^4 , and at 1×10^4 sampling moments, the system undergoes a sudden change; i.e., the channel impulse response is inverted, where the parameters of the comparison signal are shown in Table 4.

The convergence curves of each algorithm are shown in Figs. 13 and 14.

The data in Fig. 13 shows that the LMS algorithm fails to converge in the presence of impulse interference in a realistic underwater acoustic channel. By contrast, the experimental data we find in Fig. 14 demonstrates that the proposed algorithm can obtain better convergence in channel estimation of real hydroacoustic channels under impulsive interference with the same number of iterations when comparing the algorithm before improvement and the LMP algorithm using other criteria.

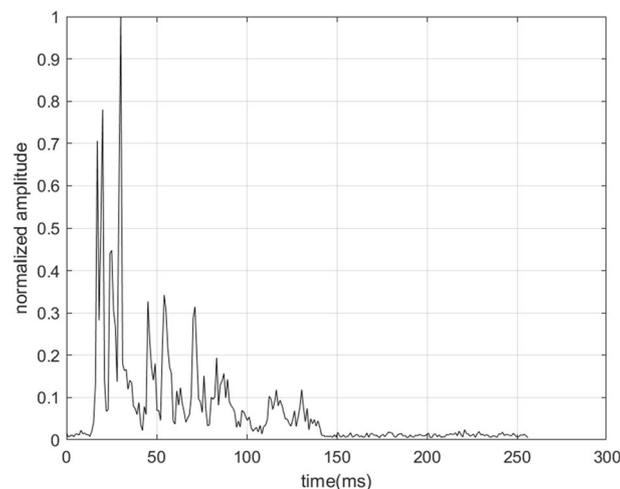


Fig. 12 The actual channel impact response

Table 4 the parameters of different algorithm under actual channel

Algorithm	Step	α	β	θ	k_{ap}
LMS	0.0035				
LMP	0.0035				
SVS-LMP	$\theta \left[\frac{1}{1 + \exp(-\alpha e(n))} - 0.5 \right]$	0.4		0.009	
IHTVS-LMP	$\theta \times \arctan \text{ch}(\alpha \times e(n) ^\beta)$	0.02	0.1	0.18	
NDCS-LMP	$\theta [1 - \exp(-\alpha e(n) ^\beta)]$	0.4	1.15	0.0068	
VSS-LMP	$\mu = \theta\mu(e(n-1)) + (1-\theta)\alpha e(n) ^2 \exp(-\beta e(n) ^2)$	0.0006	0.01	0.98	
IVSS-LMP	$f = k_{ap} \times \sqrt{\sum_{i=n-k+1}^n e(i) ^2}$ $\mu = \theta\mu f_e(n-1) + (1-\theta)\alpha f_e(n) \exp(-\beta f_e(n))$	0.0007	0.01	0.98	0.9

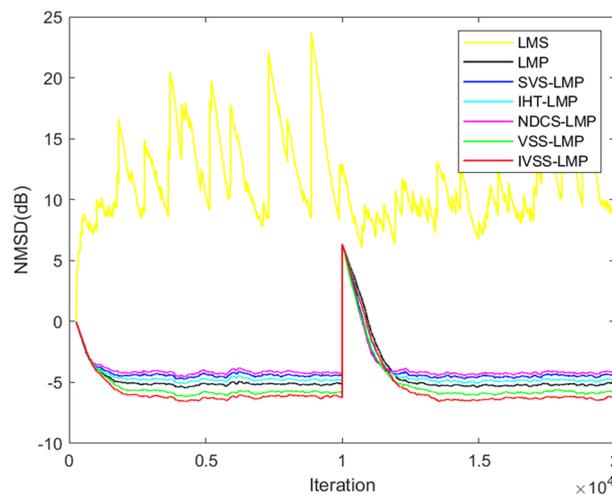


Fig. 13 Comparison of LMS and LMP under actual channel

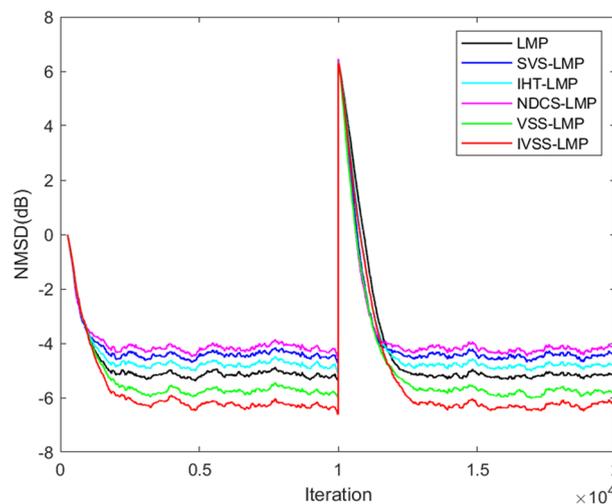


Fig. 14 Comparison of different algorithms under actual channel

5 Conclusion

Under the background of α -stable impulsive noise, this paper considers the influence of the error of the current moment and several previous moments on the convergence effect of the algorithm. The study then optimizes the algorithm on the basis of the improved Gaussian function, constructs a variable-step function by using the moving average method, and proposes an improved variable-step LMP adaptive filtering algorithm with robustness to impulse noise. Simulation experiments show that the IVSS-LMP algorithm has faster convergence and better system tracking capability than the fixed-step LMP algorithm and existing variable-step algorithms. The proposed IVSS-LMP algorithm also achieves faster convergence while guaranteeing low steady-state error through actual hydroacoustic channel identification experiments.

Abbreviations

LMS	Least mean square
LMP	Least mean p -power
VSS-LMP	Variable step-size least mean p -power
IVSS-LMP	Improved variable step-size least mean p -power
NMSD	Normalized mean square deviation
SVS-LMP	Sigmoid variable step-size least mean p -power
IHTVS-LMP	Inverse hyperbolic tangent variable step-size least mean p -power
NDCS-LMP	Normal distribution curve step-size least mean p -power

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 52071164 and in part by the Postgraduate Research & Practice Innovation Program of Jiangsu Province under Grant KYCX22_3844.

Author contributions

Methodology, BYZ, BGC, PC; software, BYZ, ZDF, PC; formal analysis, BYZ, BGC, BW, YNZ; data curation, BYZ, BW, YNZ; writing—original draft preparation, BYZ, BW; writing—review and editing, BYZ, BW; funding acquisition, BW. All authors read and approved the final manuscript.

Funding

This research received no external funding.

Availability of data and materials

Publicly available dataset were analyzed in this study. This data can be found here: <https://www.ffi.no/forskning/prosjekter/watermark>.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 4 April 2023 Accepted: 26 June 2023

Published online: 04 July 2023

References

1. B. Widrow, J.M. McCool, M.G. Larimore, C.R. Johnson, Stationary and nonstationary learning characteristics of the LMS adaptive filter. *Proc. IEEE* **64**(8), 1151–1162 (1976). <https://doi.org/10.1109/PROC.1976.10286>
2. R.H. Kwong, E.W. Johnston, A variable step size LMS algorithm. *IEEE Trans. Signal Process.* **40**(7), 1633–1642 (1992). <https://doi.org/10.1109/78.143435>
3. M.T. Akhtar, M. Abe, M. Kawamata, A new variable step size LMS algorithm-based method for improved online secondary path modeling in active noise control systems. *IEEE Trans. Audio Speech Lang. Process.* **14**(2), 720–726 (2006). <https://doi.org/10.1109/TSA.2005.855829>
4. W.W.L. Au, K. Banks, The acoustics of the snapping shrimp *Synalpheus parneomeris* in Kaneohe Bay. *J. Acoust. Soc. Am.* **103**(1), 41–47 (1998). <https://doi.org/10.1121/1.423234>
5. H. Zhao, B. Liu, P. Song, Variable step-size affine projection maximum correntropy criterion adaptive filter with correntropy induced metric for sparse system identification. *IEEE Trans. Circuits Syst. II Express Briefs* **67**(11), 2782–2786 (2020). <https://doi.org/10.1109/TCSII.2020.2973764>
6. M. Shao, C.L. Nikias, Signal processing with fractional lower order moments: stable processes and their applications. *Proc. IEEE* **81**(7), 986–1010 (1993). <https://doi.org/10.1109/5.231338>

7. O. Arikan, A. Enis Cetin, E. Erzin, Adaptive filtering for non-Gaussian stable processes. *IEEE Signal Process. Lett.* **1**(11), 163–165 (1994). <https://doi.org/10.1109/97.335063>
8. B. Wang, H. Li, C. Wu, C. Xu, M. Zhang, Variable step-size correntropy-based affine projection algorithm with compound inverse proportional function for sparse system identification. *IEEJ Transactions Elec Engng* **17**(3), 416–424 (2022). <https://doi.org/10.1002/tee.23533>
9. S.B. Gelfand, Y. Wei, J.V. Krogmeier, The stability of variable step-size LMS algorithms. *IEEE Trans. Signal Process.* **47**(12), 3277–3288 (1999). <https://doi.org/10.1109/78.806072>
10. N.J. Bershad, On error saturation nonlinearities for LMS adaptation in impulsive noise. *IEEE Trans. Signal Process.* **56**(9), 4526–4530 (2008). <https://doi.org/10.1109/TSP.2008.926103>
11. Z. Yin-bing, Z. Jun-wei, G. Ye-cai, L. Jin-ming, A constant modulus algorithm for blind equalization in α -stable noise. *Appl. Acoust.* **71**(7), 653–660 (2010). <https://doi.org/10.1016/j.apacoust.2010.02.007>
12. X. Zhu, W.-P. Zhu, B. Champagne, Spectrum sensing based on fractional lower order moments for cognitive radios in α -stable distributed noise. *Signal Process* **111**, 94–105 (2015). <https://doi.org/10.1016/j.sigpro.2014.12.022>
13. W. Yang, H. Liu, Y. Li, Y. Zhou, Joint estimation algorithms based on LMS and RLS in the presence of impulsive noise. *Xi'an Dianzi Keji Daxue Xuebao* **44**, 165–170 (2017). <https://doi.org/10.3969/j.issn.1001-2400.2017.02.028>
14. F. Huang, J. Zhang, S. Zhang, A family of robust adaptive filtering algorithms based on sigmoid cost. *Signal Process.* **149**, 179–192 (2018). <https://doi.org/10.1016/j.sigpro.2018.03.013>
15. J. Sun, X. Li, K. Chen, W. Cui, M. Chu, A novel CMA+DD_LMS blind equalization algorithm for underwater acoustic communication. *Comput. J.* **63**(6), 974–981 (2020). <https://doi.org/10.1093/comjnl/bxaa013>
16. W. Pingbo, M.A. Kai, W.U. Cai, Segmented variable-step-size LMS algorithm based on normal distribution curve. *J. Natl. Univ. Def. Technol.* **42**(5), 16–22 (2020)
17. P. van Walree, R. Otnes, and T. Jensenud, Watermark: a realistic benchmark for underwater acoustic modems, in 2016 *IEEE Third Underwater Communications and Networking Conference (UComms)*, (2016), pp. 1–4. <https://doi.org/10.1109/UComms.2016.7583423>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
