



Multi-channel ANC System with Online Secondary Path Modeling for Turboprop Aircraft Cabin

Hao Shen¹ · Qing Xue^{2,3} · Ningjuan Dong² · Yixiao Chen² · Xing Shen^{1,3}

Received: 11 September 2023 / Accepted: 12 March 2024 / Published online: 23 April 2024
© Australian Acoustical Society 2024

Abstract

The engineering implementation of the multi-channel active noise control (MCANC) system for turboprop aircraft cabin is seriously hampered by its enormous computational complexity. This paper proposes the variable-P-sequential-partial-update filtered-x least mean square (VP-SPUFxLMS) algorithm, which achieves noise reduction performance comparable to that of the multi-channel FxLMS (MCFxLMS) algorithm while significantly reducing the computational complexity. Additionally, considering the time-varying nature of the secondary paths in practical applications, the Eriksson online secondary path modeling (OSPM) method is extended from single-channel to multi-channel, the problems that may be faced when the method is applied to MCANC systems are analyzed, and an improved alternative online secondary path modeling (AOSPM) method is proposed to address the above problems, which exhibits great online modeling capabilities without introducing excessive computational load. Simulation and experiment results validate the noise control performance of the proposed method, and the ANC experiment has achieved an average reduction of more than 15 dB in the sound pressure level (SPL) of the four channels, which fully demonstrates its broad engineering application prospects.

Keywords Adaptive filters · Active noise control · Multi-channel FxLMS algorithm · Online secondary path modeling

1 Introduction

In recent years, there has been a growing focus on the comfort of air travel, prompting aircraft manufacturers to actively seek ways to enhance the comfort of flight. Bouwens et al. [1] have identified several crucial environmental factors that significantly influence the comfort of passengers in air travel, including anthropometrics, noise, odors, climate, vibrations, and light, ranked in descending order of importance. A short test flight study with 94 participants revealed that noise was mentioned as a major contributor to discomfort by over 91%

of the participants [2]. Notably, among the various types of aircraft used for passenger transportation, turboprop aircraft face a particularly severe noise problem. Consequently, it becomes imperative to find an effective solution to the problem of cabin noise in turboprop aircraft.

Turboprop noise is the main noise source of turboprop aircraft, characterized by a typical spectrum consisting of line spectrum noise superimposed on a broadband noise base. The main energy is distributed on the blade pass frequency (BPF) and its harmonic components, usually not exceeding 500 Hz. There are two main methods for noise control: passive noise control (PNC), which utilizes acoustic materials and structures to convert noise energy into thermal energy, such as sound insulation and vibration isolation, and ANC, which applies the superposition principle to counteract the noise by emitting sound waves with the same amplitude as the noise but with the opposite phase from a secondary sound source. PNC technology, due to its inherent principles, exhibits limited effectiveness in addressing low-frequency noise with longer wavelengths. Conversely, ANC method has proven to be effective in suppressing low-frequency noise, while also offering the advantages of smaller system volumes and

✉ Xing Shen
shenx@nuaa.edu.cn

Hao Shen
shenhao@nuaa.edu.cn

¹ State Key Laboratory of Mechanics and Control of Aerospace Structures, Nanjing University of Aeronautics and Astronautics, Nanjing Yudao, 210016, China

² AVIC Aircraft Strength Research Institute of China, Xi'an 710065, China

³ National Key Laboratory of Strength and Structural Integrity, Xi'an 710065, Shaanxi, China

weights [3–5]. Therefore, for noise control in the cabin of turboprop aircraft, the ANC method is a superior choice.

As soon as the ANC method was proposed, it was applied to the cabin of turboprop aircraft. Nelson and Elliott from the University of Southampton in the UK conducted ANC flight experiments on the cabin of the BAE748 twin propeller aircraft in the early 1990s. The experiment used 32 error microphones to collect sound signals, and 16 speakers were employed as secondary sound sources to control the noise generated by the turboprop. The experiment achieved good results: the cabin noise was measured at the BPF of the turboprop at 88 Hz and its two harmonic frequencies at 176 Hz and 264 Hz being reduced by 13 dB, 9 dB, and 6 dB, respectively [6, 7]. On this basis, ANC was soon applied to turboprop aircrafts, such as Dornier 328, ATR42/47, A400M, Q4000, etc., some of which are still in service today [8]. However, decades later, the commercial application of ANC technology to aircraft has not expanded, which is mainly due to the fact that in order to effectively reduce noise in the cabin of airplanes with large space, it is necessary to expand the scale of the ANC system, and the computational complexity of the system increases drastically, which leads to difficulties in the implementation of ANC algorithms.

To alleviate the computational complexity, numerous algorithms have been proposed to reduce the computations involved in the adaptation process [9–15]. The block FxLMS algorithm reduces the update frequency of filter weight coefficients [16–19], while the frequency-domain FxLMS algorithm uses fast Fourier transform (FFT) to convert the convolution operation in the time domain into the multiplication operation in the frequency domain [20, 21]. Both methods can effectively reduce the amount of calculation, but the introduction of latency deteriorates the performance of ANC system. The sign-FxLMS algorithm reduces multiplications in the weight update equation by replacing the filtered reference value with its sign, but this improvement comes at the cost of slower convergence and compromised noise reduction performance [22]. The block coordinate descent MCFxLMS algorithm takes each control filter as a block coordinate group to update alternatively during the control progress, resulting in a substantial reduction in computational load but poor convergence performance [23]. The partial-update FxLMS algorithm (PUFxLMS) updates only a portion of the filter weight coefficients within one iteration cycle and it also suffers from slow convergence [24, 25].

These algorithms demonstrate various trade-offs between computational complexity and noise control performance. To strike a balance, this paper proposes the Variable-P-Sequential-Partial-Update FxLMS (VP-SPUFxLMS) algorithm, which aims to update more control filter weight coefficients during the early stages of system operation and fewer weights as the system converges. By dynamically adjusting the update strategy according to the state of the

system, the VP-SPUFxLMS algorithm achieves a favorable trade-off between computational complexity and noise control performance.

In addition, accurate modeling of the secondary path is crucial for the implementation of the FxLMS algorithm and its variations, as they rely on the estimation of the secondary path to facilitate the updating of adaptive noise control filters [26, 27]. Typically, a secondary path includes a sequence of components, such as a DAC (digital-to-analog converter), reconstruction (low pass) filter, power amplifier, loudspeaker, acoustic path, error sensor, preamplifier, anti-aliasing (low pass) filter, and an ADC (analog-to-digital converter) [28]. In practical ANC implementations, the secondary path is time varying, necessitating online modeling to ensure the performance of ANC systems based on FxLMS-type algorithms.

Eriksson [29] proposed the OSPM method in 1989, where he used additional random white noise as the modeling signal to model the secondary path online. This method has a drawback where the secondary modeling signal and the control signal can interfere with each other, leading to potential system instability, and the algorithm may diverge in severe cases. To address the limitation, Ming et al. [30, 31] proposed a cross-update secondary channel modeling method using an additional filter, effectively eliminating interactions between the noise active control and the secondary path modeling sessions. Akhtar [32, 33] introduced the variable step size LMS algorithm and proposed a method for continuous control of additional random white noise power, which serves to mitigate the effect of the secondary modeling signal on the noise active control session. Taking advantage of fractional signal processing, Aslam et al. [34] developed a new adaptive procedure using a multidirectional step size strategy in fractional LMS algorithm and applied it to weight adaptation of modeling filters, which demonstrates distinct advantages in terms of model accuracy and convergence rate. Ma et al. [35] used a decoupling filter consisting of multiple band-pass filters to separate the broadband noise component from the narrowband noise component for narrowband ANC. The remaining sinusoidal noise from the linear prediction filter output is used to update the controller and scale the auxiliary white Gaussian noise, while the linear prediction filter error signal is used as the desired signal for the OSPM subsystem. This approach effectively minimizes the coupling between the controller and the OSPM subsystem.

In fact, the current practical application of ANC systems rarely employs online modeling method. There are several challenges when applying the online modeling method to MCANC systems, including the coupling between the OSPM filters and the control filters, interchannel coupling effect, and the introduction of enormous computational complexity. This paper introduces a method that alternately updates the secondary paths corresponding to each group of secondary sources, effectively addressing these issues. Table 1 provides

Table 1 Abbreviations used in this paper

| Abbreviations | Expansion |
|---------------|--|
| 1 MCANC | Multi-channel ANC |
| 2 MCFxLMS | Multi-channel filtered-x LMS |
| 3 OSPM | Online secondary path modeling |
| 4 PUFxLMS | Partial-update FxLMS |
| 5 SPUFxLMS | Sequential-partial-update FxLMS |
| 6 VP-SPUFxLMS | Variable-P-sequential-partial-update FxLMS |
| 7 AOSPM | Alternative online secondary path modeling |

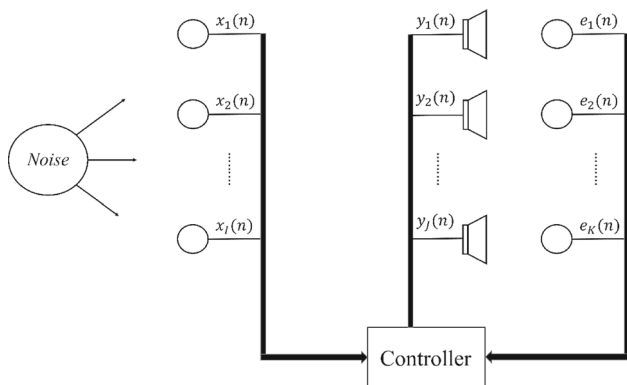


Fig. 1 Diagram of MCANC system

the abbreviations used in this paper along with their counterparts.

The sections are organized as follows: Section 2 describes the VP-SPUFxLMS algorithm. The proposed online secondary path modeling method is presented in Sect. 3. The computational complexity analysis is given in Sect. 4. Sections 5 and 6 exhibit the simulation and real-time experiment results of the proposed method, respectively. Finally, Sect. 7 gives the concluding remarks.

2 Variable-P-Sequential-Partial-Update FxLMS (VP-SPUFxLMS) Algorithm

It is assumed that the MCANC system consists of I reference microphones, J secondary sources, and K error microphones, as shown in Fig. 1. There is a primary path between each reference microphone and each error microphone, resulting in a total of IK secondary paths, where P_{ik} denotes the primary path between the i th reference microphone and the k th error microphone. Correspondingly, there is also a secondary path between each secondary source and each error microphone, resulting in a total of JK secondary paths. The secondary path between the j th secondary source and the k th

error microphone is denoted as $S_{jk}(n)$, and $\hat{S}_{jk}(n)$ represents the secondary path modeling estimation of $S_{jk}(n)$, where n represents the time.

The i th reference signal vector is denoted by $\mathbf{x}_i(n) = [x_i(n)x_i(n-1)\cdots x_i(n-L+1)]^T$, and $\mathbf{W}_{ij}(n) = [w_{ij1}(n)w_{ij2}(n)\cdots w_{ijL}(n)]^T$ represents the adaptive filter weight vector corresponding to the i th reference signal and the j th speaker with the length of L , where T is the transpose operation. The cost function of the MCFxLMS algorithm is the sum of the mean squares of the K error signals, as shown below

$$J(n) = \sum_{k=1}^K E[e_k^2(n)] \tag{1}$$

Actually, it is impossible to record the error signal data at each moment, so the objective function is usually replaced by the sum of the squares of the instantaneous errors [36]:

$$J(n) = \sum_{k=1}^K e_k^2(n) \tag{2}$$

where $e_k(n)$ represents the error signal captured by the k th error microphone, which can be expressed as:

$$e_k(n) = d_k(n) - \sum_{j=1}^J y_j(n) * S_{jk}(n) \tag{3}$$

where $*$ is the linear convolution operator. $y_j(n) = [y_j(n)y_j(n-1)\cdots y_j(n-M+1)]^T$ indicates the output signal of the j th secondary source with the length of M , which can be expressed as:

$$y_j(n) = \sum_{i=1}^I \mathbf{W}_{ij}^T(n) \bullet \mathbf{x}_i(n) \tag{4}$$

By substituting (3) and (4) into (1), the gradient of (1) in the terms of $\mathbf{W}_{ij}(n)$ can be derived as:

$$\nabla_{ij} J(n) = -2 \sum_{k=1}^K e_k(n) \bullet [x_i(n) * S_{jk}(n)] \tag{5}$$

The adaptive filter weight coefficients get updated by LMS algorithm as

$$\mathbf{W}_{ij}(n+1) = \mathbf{W}_{ij}(n) + 2\mu \sum_{k=1}^K e_k(n) \bullet [x_i(n) * S_{jk}(n)] \tag{6}$$

where μ stands for the step size. Since the secondary path is unknown, its modeling estimation result $\hat{S}_{jk}(n)$ is used as a

substitute in the calculations. Note that the filtered- x signal is $\mathbf{x}'_{ijk}(n) = \mathbf{x}_i(n) * \widehat{\mathbf{S}}_{jk}(n)$, then Eq. (6) can be rewritten as:

$$\mathbf{W}_{ij}(n+1) = \mathbf{W}_{ij}(n) + 2\mu \sum_{k=1}^K e_k(n) \bullet \mathbf{x}'_{ijk}(n) \quad (7)$$

When the number of channels and the length of the control filters increase, the computational burden of the MCFxLMS algorithm increases rapidly, which poses a significant challenge to the real-time implementation of the algorithm. To tackle this challenge, the SPU-MCFxLMS algorithm is introduced, based on the core concept of assuming gradual changes in the statistical characteristics of the reference signal. It divides the weight coefficients of the control filter into P equal parts and updates the corresponding subset of coefficients at each sampling cycle while leaving the remaining coefficients unchanged. As the number of iterations increases, all weight coefficients are updated after P iterations. The weight coefficients update equation of the algorithm can be expressed as:

$$\mathbf{W}_{ijl}(n+1) = \begin{cases} \mathbf{W}_{ijl}(n) + 2\mu \sum_{k=1}^K e_k(n) \cdot x_{ijk}(n-l), & (n-l)\%P = 0 \\ \mathbf{W}_{ijl}(n), & \text{otherwise} \end{cases} \quad (8)$$

where $\%$ represents the remainder operation, and l represents the l th element in the control filter weight coefficient array. From (8), it can be observed that the algorithm does not need to calculate the filtered- x signal at each iteration, but only needs to recalculate the new filtered- x signal every P iterations. It is worth noting that this algorithm not only reduces the computational complexity of the weight coefficient updating process, but also reduces the computational complexity of the filtered- x calculation, since it is no longer necessary to compute the corresponding filtered- x values for the weight coefficients that have not been updated. As a result, the SPU-MCFxLMS algorithm provides a significant reduction in computation.

However, compared to the MCFxLMS algorithm, the SPU-MCFxLMS algorithm requires more time to complete a full update of all weight coefficients, and the weight coefficients are not entirely up-to-date at each moment, which results in slower convergence speed and a delayed response to external changes [24]. Specifically, the larger the P -value, the fewer the weight coefficients updated in each iteration, the slower the convergence speed; the smaller the P -value, the more the weight coefficients updated in each iteration, the faster the convergence speed. To solve this problem, the SPU-MCFxLMS algorithm is improved by updating more weight

coefficients, or even all of them, at the beginning of convergence and gradually updating fewer weight coefficients as the algorithm stabilizes. The value of P , which determines how many weight coefficients are updated, is calculated as follows:

$$P(n) = \text{floor}[\rho(n)P_{\min} + (1 - \rho(n))P_{\max}] \quad (9)$$

where floor denotes the floor operation, P_{\min} and P_{\max} can be determined by experiments, as long as their values do not cause the system to diverge, and $\rho(n)$ is determined by the power of the error signal $e_k(n)$ and the output signal $y_j(n)$ as follows:

$$\rho(n) = P_e(n)/(P_e(n) + P_y(n)) \quad (10)$$

It is evident that the range of $\rho(n)$ is from 0 to 1. The calculation of $P_e(n)$ and $P_y(n)$ is smoothed to mitigate the interference of external noise, particularly impulse noise.

$$P_e(n) = \lambda P_e(n-1) + (1 - \lambda) \sum_{k=1}^K e_k^2(n) \quad (11)$$

$$P_y(n) = \lambda P_y(n-1) + (1 - \lambda) \sum_{j=1}^J y_j^2(n) \quad (12)$$

where λ is the forgetting factor and typically close to 1, which determines the extent to which the previous power value is considered in the current increment.

Actually, $\rho(n)$ can be approximated as the ratio of the power of error signal power $e_k(n)$ to the power of primary noise signal $d_k(n)$, since the power of $d_k(n)$ is nearly equal to the sum of the power of $e_k(n)$ and $y_k(n)$. Therefore, the adjustment of P can be seen as being based on the amount of reduction in noise power. When the system converges, $\rho(n)$ is close to 0 and P is close to its maximum value, while when the residual noise is high, $\rho(n)$ is close to 1 and P is close to its minimum value. By dynamically adjusting the value of P , the proposed algorithm not only enhances convergence performance but also ensures a significant reduction in computational complexity.

3 Alternative Online Secondary Path Modeling (AOSPM) Method

The offline modeling approach estimates the secondary paths before the system starts, and once the results are imported into the control system, the model of the secondary paths will remain constant throughout the control process. However, when there are modeling errors, the FxLMS algorithm converges to a biased solution, which is inevitable in practical applications because the secondary paths are subject

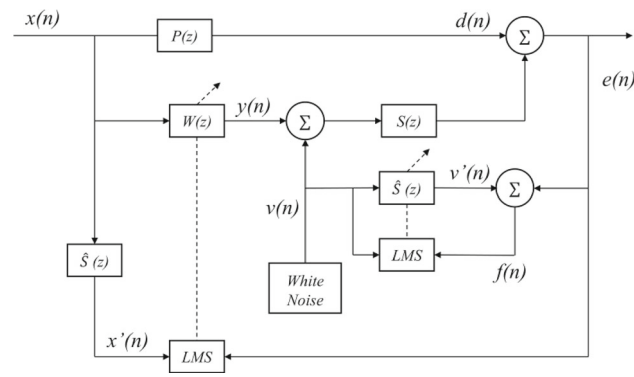


Fig. 2 Block diagram of Eriksson’s method

to constant changes that can be attributed to factors such as component aging, thermal variations, and environmental modifications [37].

The presence of modeling errors will lead to inaccurate updates of the control filter weight coefficients, thereby affecting the performance of the ANC system and leading to control failure in more severe cases [26]. OSPM is preferred to solve the problem. The fundamental method for ANC systems with OSPM was proposed by Eriksson et al. [29], as shown in Fig. 2.

Eriksson introduces auxiliary white noise and another adaptive filter for secondary path modeling. The error signal $e(n)$ comprises auxiliary noise filtered by the secondary path and the residual noise of the ANC system, where the auxiliary noise cannot be canceled out. To enhance the modeling accuracy, it is necessary to increase the energy of white noise. However, the error signal used to update the control filter coefficients contains this component, resulting in a deviation in the direction of coefficient updating and subsequently leading to a decline in control performance. In addition, when the OSPM is extended to MCANC, there is not only coupling between the OSPM filters and the control filters but also coupling between the different secondary paths. Actually, this cross-coupling effect occurs for both online and offline modeling. As an example, a $1 \times 2 \times 2$ ANC system is used to briefly illustrate the cause of the problem, as shown in Fig. 3.

Define the error component attributed to the original noise as

$$e_{p1}(n) = d_1(n) - y_1(n) * S_{11}(n) - y_2(n) * S_{21}(n) \quad (13)$$

The control error and the modeling error can be expressed as follows:

$$e_1(n) = e_{p1}(n) + v(n) * S_{11}(n) + v(n) * S_{21}(n) \quad (14)$$

$$f_1(n) = e_{p1}(n) + v(n) * (S_{11}(n) - \hat{S}_{11}(n)) + v(n) * (S_{21}(n) - \hat{S}_{21}(n)) \quad (15)$$

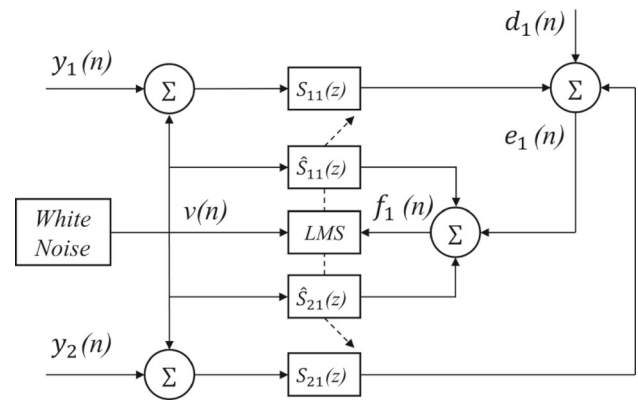


Fig. 3 Secondary path modeling for a $1 \times 2 \times 2$ ANC system

where $v(n)$ represents zero-mean white noise that is uncorrelated with the reference signal $x(n)$. The coefficients of the modeling filter are updated by the LMS algorithm, which can be expressed as follows:

$$\begin{aligned} \hat{S}_{11}(n+1) &= \hat{S}_{11}(n) + \mu_s v(n) f_1(n) = \hat{S}_{11}(n) + \mu_s v(n) \\ &\times [e_{p1}(n) + v(n) * (S_{11}(n) - \hat{S}_{11}(n)) \\ &+ v(n) * (S_{21}(n) - \hat{S}_{21}(n))] \end{aligned} \quad (16)$$

Ideally, the LMS algorithm can accurately estimate the secondary paths $S_{11}(n)$ and $S_{21}(n)$ that can be represented as

$$S_{11}(n) = \hat{S}_{11}(n) + \mu_s v(n) [v(n) * (S_{11}(n) - \hat{S}_{11}(n))] \quad (17)$$

$$S_{21}(n) = \hat{S}_{21}(n) + \mu_s v(n) [v(n) * (S_{21}(n) - \hat{S}_{21}(n))] \quad (18)$$

By substituting (17) and (18) into (16), $\hat{S}_{11}(n+1)$ can be rewritten as:

$$\hat{S}_{11}(n+1) = S_{11}(n) + (S_{21}(n) - \hat{S}_{21}(n)) + \mu_s v(n) e_{p1}(n) \quad (19)$$

Since $v(n)$ is uncorrelated with $v(n)$, and $e_{p1}(n)$ is also uncorrelated with $v(n)$, the LMS solution will not be influenced by $e_{p1}(n)$. In Eq. (19), $\hat{S}_{11}(n)$ will converge to $S_{11}(n) + (S_{21}(n) - \hat{S}_{21}(n))$ instead of the expected value $S_{11}(n)$. When the number of channels increases, this coupling effect will be more serious, and the accuracy of modeling will further decrease.

Furthermore, the additional computational complexity associated with OSPM is often considered unacceptable. The number of secondary paths is determined by both the number of secondary sources and the number of error signals. As the scale of MCANC systems expands from $1 \times 2 \times 2$ to $1 \times 4 \times 4$, the number of secondary paths increases from 4 to 16, indicating that online modeling necessitates 16 adaptive filters,

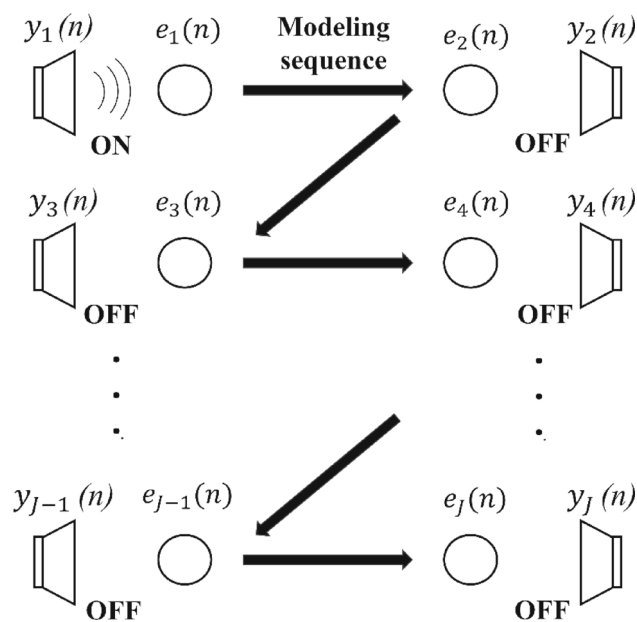


Fig. 4 The proposed secondary path modeling strategy (A group contains a secondary source)

while the number of control filters is only 4. In practice, the length of the secondary path model is often longer than that of the control filter, resulting in the computational burden of online modeling frequently exceeding that of the active control part, which severely increases the overall computational load of the MCANC system.

To address these problems, this paper proposes the AOSPM method, which employs the key strategy of grouping several secondary sources and alternately updating the secondary path models corresponding to each group online. Figure 4 provides an illustration of the modeling process.

The specific process of this method is as follows: Firstly, the system is turned on, and the first group of secondary sources emits white noise to model its corresponding secondary paths. After a certain duration, the first group ceases emitting white noise, and the next group emits white noise to model the corresponding secondary paths, and then, the process is repeated. It can be summarized as follows:

$$\hat{S}_{jk}(n+1) = \begin{cases} \hat{S}_{jk}(n) + \mu_s v(n) f_k(n), & j \in \text{group}(n) \\ \hat{S}_{jk}(n), & \text{others} \end{cases} \quad (20)$$

where $\text{group}(n)$ represents the group of secondary sources that should emit white noise to model the corresponding secondary paths at time n . The duration of modeling for each group can be determined based on preliminary experiments. The principle of grouping secondary sources is to minimize the mutual interference of white noise emitted by the secondary sources in the group at the error sensors, which is mainly achieved by selecting secondary sources that are

positioned at a considerable distance from each other. This method attenuates or even directly eliminates (when each group contains only one secondary source) the mutual coupling between secondary paths within the group and greatly reduces the computational complexity.

4 Computational Complexity Analysis

The proposed algorithm and secondary path modeling method, along with a detailed analysis of their computational complexity per iteration, are summarized in Table 2. Assuming that the MCANC system has I reference sensors, J secondary sources, and K error sensors, the length of the control filters is L , the length of the secondary path models is M , the partial update coefficient is P , and the secondary sources are divided into G groups. The calculations involving '%' and 'ε' can be accomplished using conditional statements without additional calculation, and the computational complexity of division and conditional statements is not considered because they account for a small percentage.

In order to intuitively compare the computational complexity of these methods, Fig. 5 shows the computational complexity of the MCANC system with a scale of $1 \times 16 \times 16$, where $I = 1$, $J = 16$, $K = 16$, $L = 128$, $M = 128$, $P = 4$, $G = 8$.

It is evident that the proposed method, which directly reduces the computational complexity of the key steps: calculating x'_{ijk} , updating W_{ijl} , calculating v'_{jk} , and updating \hat{S}_{jk} , greatly reduces the computational complexity of the system by nearly several times. In addition, the sustained large value of P after system stabilization contributes to an even lower computational load, which is advantageous for maintaining low power consumption during the long-term operation of the ANC system. This prevents excessive heat generation in the hardware, which could result in performance degradation and further cause the real-time requirements of the algorithm to be unsatisfied.

5 Simulation Results

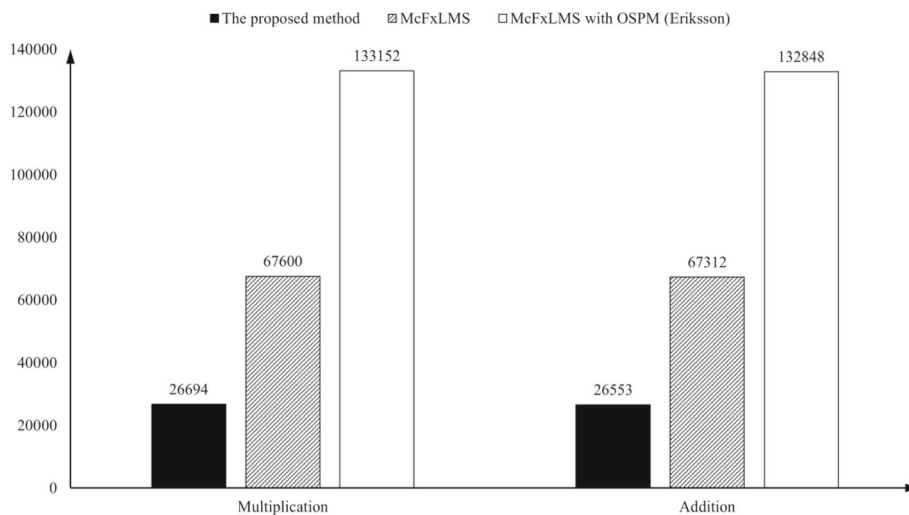
In this section, simulations were conducted to evaluate the performance of the proposed method. The scale of the simulated MCANC system is $1 \times 2 \times 2$, and its two primary paths are shown in Fig. 6. The sampling frequency for all simulations is 5 kHz and the primary and secondary paths are the same.

A certain type of turboprop aircraft adopts a dual turboprop engine, and its noise energy is primarily concentrated in the blade passing frequency of 108 Hz and its first- and second-order harmonic components of 216 Hz and 324 Hz. As long as these frequencies of noise can be controlled, the

Table 2 Algorithm description and detailed computational complexity analysis of the proposed method, MCFxLMS and MCFxLMS with OSPM (Eriksson)

| Calculation step | Multiplications (×) | Additions (+) |
|--|---|---|
| $\mathbf{x}'_{ijk}(n) = \mathbf{x}_i(n) * \widehat{\mathbf{S}}_{jk}(n)$ | $IJKM/P$ | $IJK(M - 1)/P$ |
| $\mathbf{W}_{ij}(n+1) = \mathbf{W}_{ij}(n) + \mu \sum_{k=1}^K e_k(n) \bullet \mathbf{x}'_{ijk}(n)$ | $IJKL/P + K$ | $IJKL/P$ |
| $y_j(n) = \sum_{i=1}^I \mathbf{W}_{ij}^T(n) \bullet \mathbf{x}_i(n) + v_j(n)$ | IJL | $J(IL - 1) + J/G$ |
| $\widehat{\mathbf{S}}_{jk}(n+1) = \widehat{\mathbf{S}}_{jk}(n) + \mu_s \mathbf{v}(n) f_k(n)$ | $JKM/G + K$ | JKM/G |
| $f_k(n) = e_k(n) - v'_{jk}(n)$ | 0 | K/G |
| $v'_{jk}(n) = \mathbf{v}_j(n) * \widehat{\mathbf{S}}_{jk}(n)$ | JKM/G | $JK(M - 1)/G$ |
| $P(n) = \text{floor}[\rho(n)P_{\min} + (1 - \rho(n))P_{\max}]$ | 2 | 2 |
| $\rho(n) = P_e(n)/(P_e(n) + P_y(n))$ | 0 | 1 |
| $P_e(n) = \lambda P_e(n - 1) + (1 - \lambda) \sum_{k=1}^K e_k^2(n)$ | $K + 2$ | $K + 1$ |
| $P_y(n) = \lambda P_y(n - 1) + (1 - \lambda) \sum_{j=1}^J y_j^2(n)$ | $J + 2$ | $J + 1$ |
| Total | $\frac{IJKM+IJKL}{P} + IJL + \frac{2JKM}{G} + 3K + J + 6$ | $\frac{IJK(L+M-1)}{P} + IJL + K + \frac{J+K+JK(2M-1)}{G} + 5$ |
| MCFxLMS | $IJKM + IJKL + K + IJL$ | $IJK(L + M - 1) + IJL - J$ |
| MCFxLMS with OSPM (Eriksson) | $IJKM + IJKL + IJL + 2JKM + 2K$ | $IJK(L + M - 1) + IJL - J + K + JKM$ |

Fig. 5 The computational complexity of the MCANC system with a scale of $1 \times 16 \times 16$, where $I = 1, J = 16, K = 16, L = 128, M = 128, P = 4, G = 8$

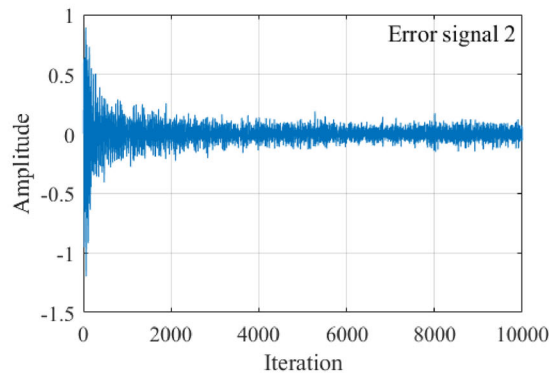
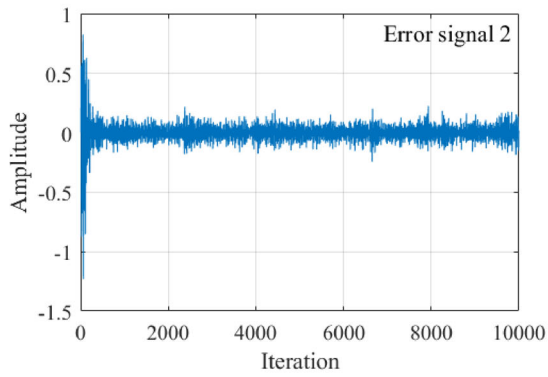
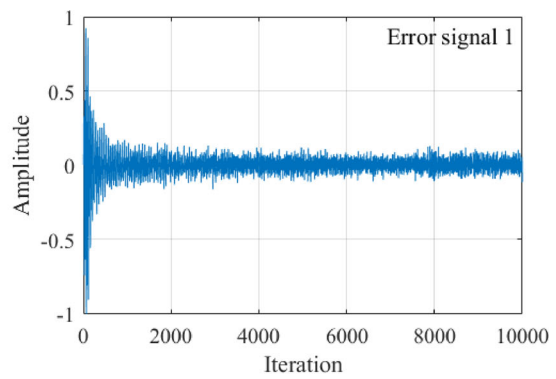
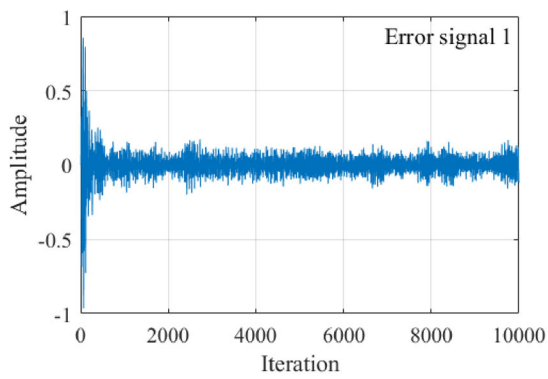
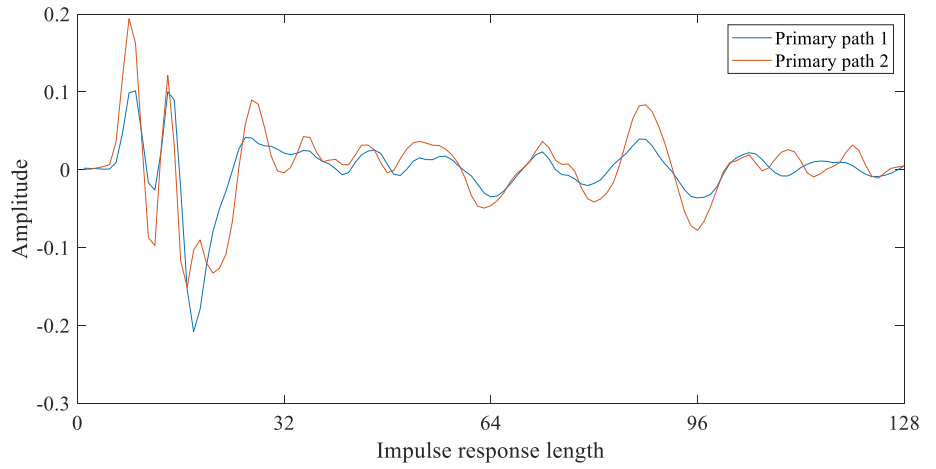


overall noise level can be significantly reduced. Therefore, the primary noise for simulation consists mainly of the above three specified frequencies with a certain amount of white noise superimposed to approximate reality, the spectrum of which can be seen in Fig. 11.

Three sets of simulations were used to compare and evaluate the performance of the VP-SPUFxLMS algorithm and the

AOSPM method. In Case 1, the convergence speed between the VP-SPUFxLMS algorithm and the MCFxLMS algorithm was compared, and the ability of the adaptation strategy of P to handle sudden changes in noise was also verified; In Case 2, the AOSPM method was compared with the OSPM (Eriksson) method; in Case 3, the reduction in sound pressure level after convergence of the MCFxLMS algorithm,

Fig. 6 Primary paths



(a) MCFxLMS

(b) VP-SPUFxLMS

Fig. 7 The waveforms of the MCFxLMS algorithm and the VP-SPUMCFxLMS algorithm running for 2 s

Fig. 8 The variation of P over time

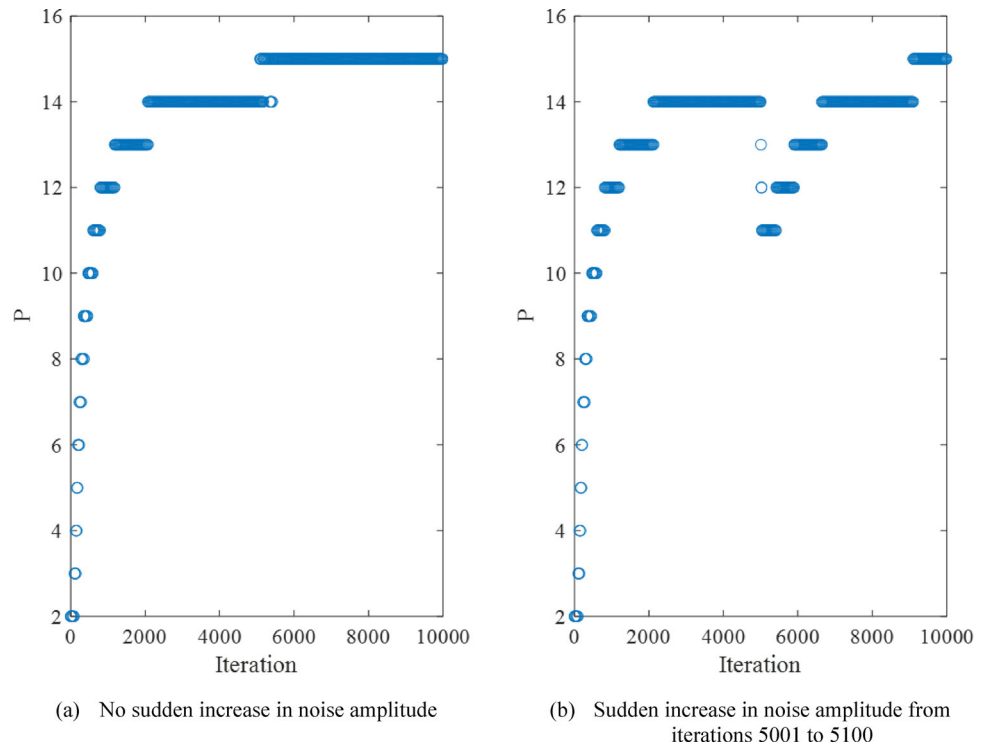


Fig. 9 Modeling error. S_{ij} represents the secondary path from speaker i to error sensor j

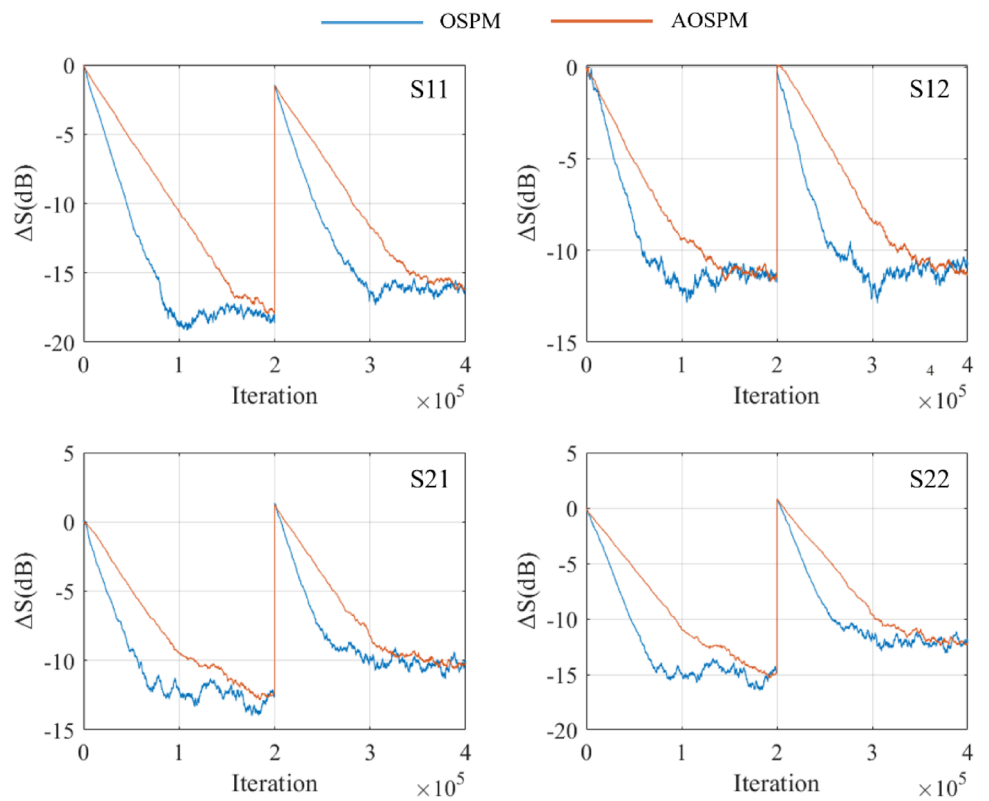
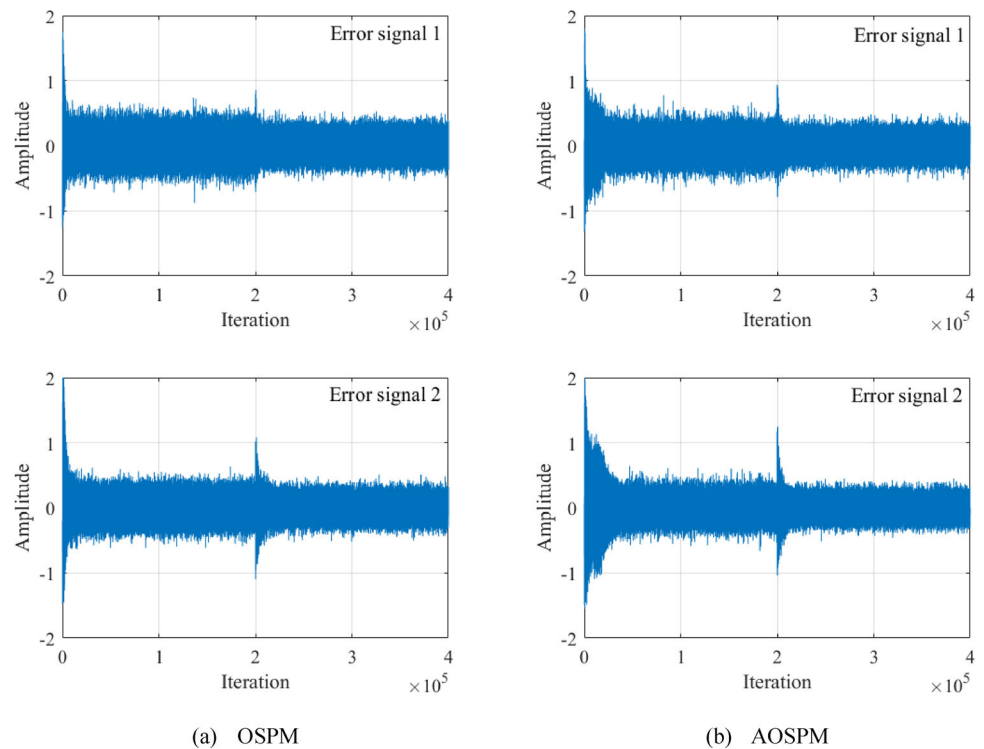


Fig. 10 The waveform of the OSPM and AOSPM running for 40 s



the MCFxLMS algorithm with OSPM (Eriksson) and the VP-SPUFxLMS algorithm with AOSPM was analyzed.

5.1 Case 1

In this case, it is assumed that all the estimates of the secondary paths are accurate. The step size μ is set to 0.0001; P_{\min} and P_{\max} are set to 2 and 16, respectively. The forgetting factor λ is selected as 0.997. Figure 7 shows the time-domain waveforms of the VP-SPUFxLMS algorithm and the MCFxLMS algorithm running for 2 s.

From Fig. 7a, b, it can be observed that the VP-SPUFxLMS algorithm has a slightly slower convergence speed compared to the MCFxLMS algorithm, but both algorithms can reach a stable state within two seconds. The computational complexity of the VP-SPUFxLMS algorithm can be roughly estimated by observing the value of P , and the variation of P is depicted in Fig. 8a.

In Fig. 8b, the variation of P is shown when a sudden external disturbance occurs which is simulated by adding a square wave with an amplitude of + 5 to the error signals from iterations 5001 to 5100. It can be seen that P is not significantly affected by this disturbance; its value briefly decreases from 14 to 11 and then increases back to 14 immediately. Therefore, the P adaptation strategy of the VP-SPUFxLMS algorithm is effective in resisting strong external noise interference, ensuring low computational load during

system convergence, and enabling sustained operation with low power consumption.

5.2 Case 2

In this case, the online modeling performance of AOSPM and OSPM is compared. The initial values of the secondary path models are set to $\mathbf{0}$, and it is assumed that the secondary paths undergo changes after 20,000 iterations. The step sizes μ and μ_s are set to 0.00001 and 0.01, respectively. The white noise with a variance of 0.005 is used for secondary path modeling, and the switching time of AOSPM is 0.1 s. The relative modeling error is defined as (21), its variation curve is shown in Fig. 9, and the convergence of two error signals can be seen in Fig. 10.

$$\Delta S = 10 \log_{10} \left\{ \frac{\sum_{i=1}^M [s_i(n) - \hat{s}_i(n)]^2}{\sum_{i=1}^M [s_i(n)]^2} \right\} \quad (21)$$

From the above results, it is evident that both online modeling strategies successfully completed the modeling of secondary paths and effectively track the changes in secondary paths, and AOSPM exhibited a slightly slower convergence speed compared to OSPM. However, in practice, AOSPM benefits from not considering the coupling issue, which allows it to utilize white noise with a higher energy for modeling, potentially enhancing its convergence speed.

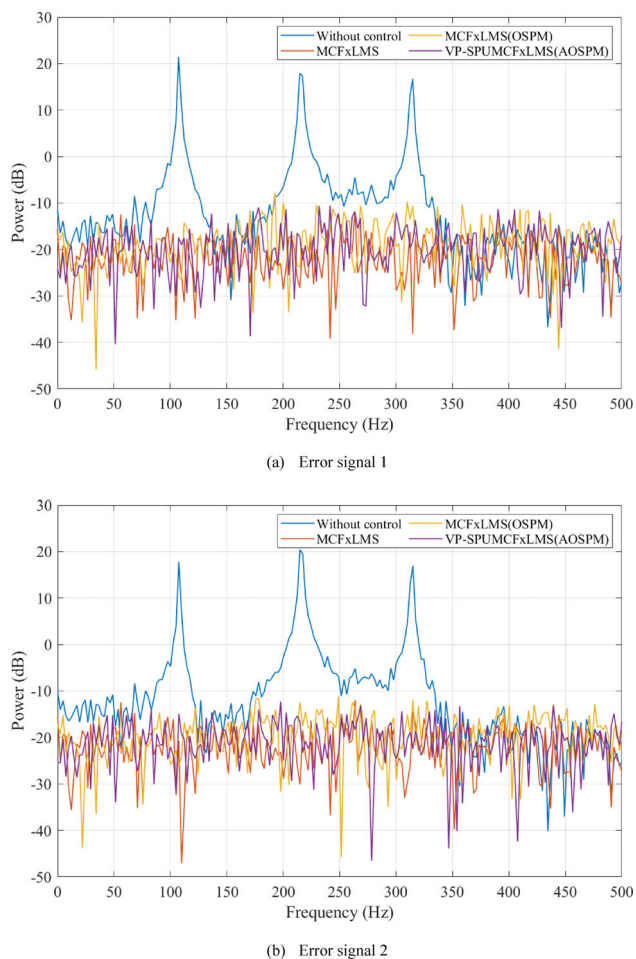


Fig. 11 Power spectrums of the two error signals



Fig. 12 Experimental platform

5.3 Case 3

In this case, the steady-state performance of the MCFxLMS algorithm, the MCFxLMS algorithm with OSPM, and the proposed method are compared. It is assumed that the MCFxLMS algorithm has accurately estimated secondary path models, while the other two methods start modeling the secondary paths from $\mathbf{0}$. The power spectra of the two residual noises for the three methods are shown in Fig. 11.

It can be seen that all three algorithms successfully attenuate the primary noise and have similar noise power spectra. The MCFxLMS algorithm shows slightly better performance, primarily due to its assumption of an accurately estimated secondary path model, which is impossible in practice. In contrast, the other two methods introduce additional white noise, which affects the noise power spectrum.

In summary, with greatly reduced computational complexity, the proposed method offers a convergence speed and steady-state performance that is close to the MCFxLMS algorithm, while it also demonstrates comparable capability to the OSPM method in terms of secondary path modeling.

6 Experimental Results

To further validate the performance of the proposed method in practical applications, a $1 \times 4 \times 4$ MCANC system was constructed based on the NI-PXI platform, as shown in Fig. 12. The controller is PXIe-7862, which connects 16 analog input channels and 8 analog output channels directly to the Kintex-7325 T FPGA, each with a dedicated analog-to-digital converter (ADC) for independent timing and triggering. All algorithms have been coded by the FPGA module in the LabVIEW 2020 software.

The step sizes μ and μ_s are set to 0.00001 and 0.01, respectively, P_{\min} and P_{\max} are set to 2 and 16, respectively, the forgetting factor λ is selected as 0.997, and the sampling frequency is 5 kHz. The length of control filters and the secondary path modeling filters are both 128 taps. The primary noise is generated as a combination of multiple frequency components, including the frequencies of 108 Hz, 216 Hz, and 324 Hz, which is consistent with the simulation. Only model the secondary paths corresponding to one secondary source at a time, and the switching time of AOSPM is 2 s. The time-domain waveforms of the MCANC system based on the MCFxLMS, MCFxLMS with OSPM, and VP-SPUFxLMS with AOSPM in a steady state are shown in Fig. 13.

Fig. 13 The waveforms of four error signals, when MCANC, based on the MCFxLMS, MCFxLMS with OSPM, and VP-SPUMCFxLMS with AOSPM is in steady-state

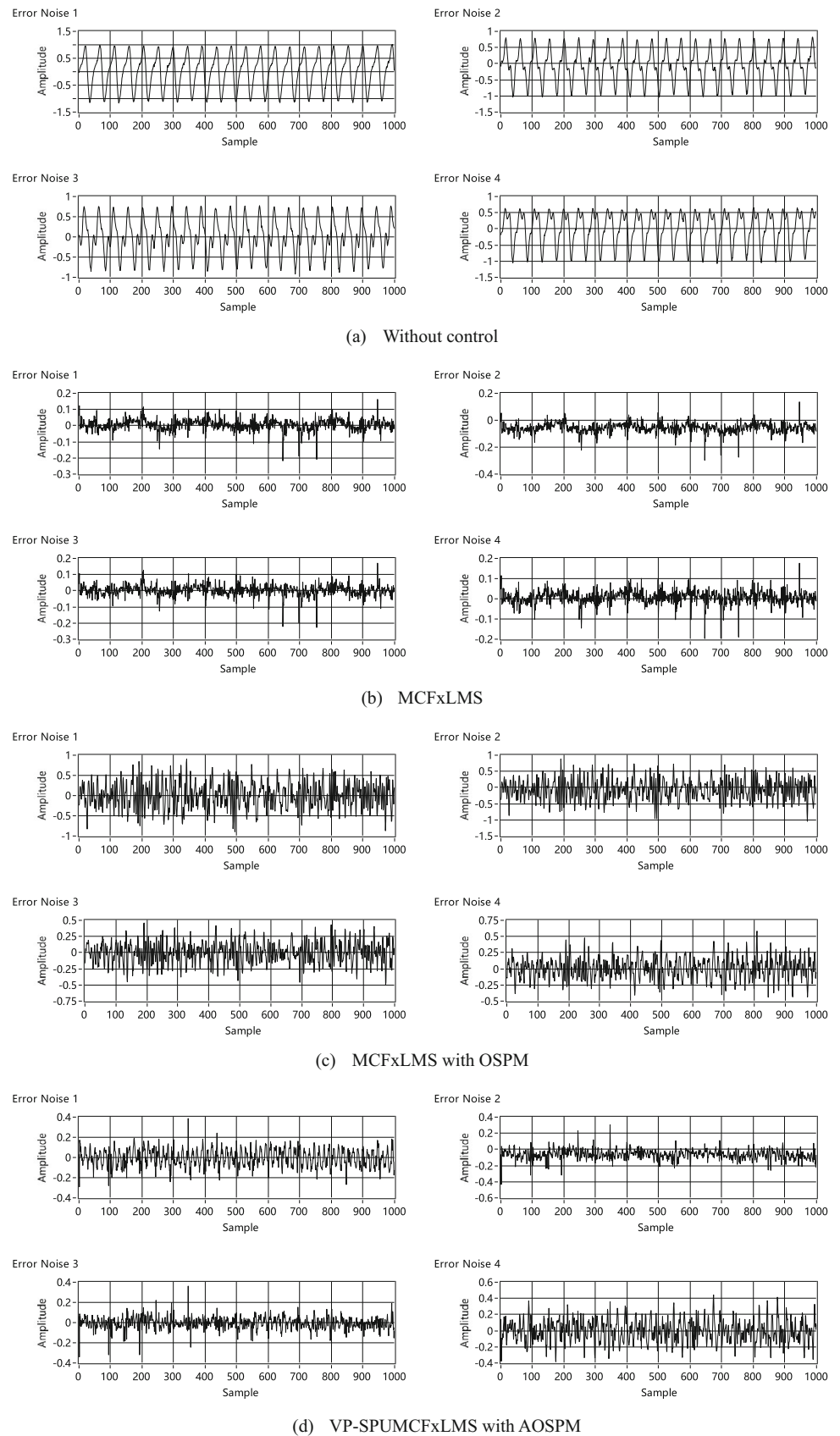


Table 3 Total sound pressure-level comparison

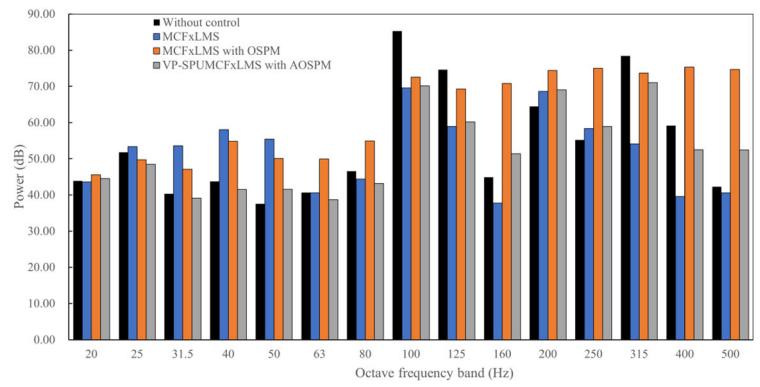
| Position | 1 (dB) | 2 (dB) | 3 (dB) | 4 (dB) | Average noise reduction (dB) |
|--------------------------|--------|--------|--------|--------|------------------------------|
| Control off | 89.02 | 86.64 | 88.17 | 86.18 | |
| MCFxLMS | 72.89 | 66.62 | 71.57 | 68.79 | 17.58 |
| MCFxLMS with OSPM | 80.69 | 79.76 | 77.67 | 75.65 | 9.06 |
| VP-SPUMCFxLMS with AOSPM | 75.24 | 68.04 | 72.61 | 73.82 | 15.075 |

In the experiment, it was found that OSPM tends to diverge when the energy of white noise is low, necessitating an increase in the white noise energy, which will lead to a deterioration of the overall noise reduction performance. On the other hand, AOSPM, which emits white noise from only one secondary sound source at a time, had a relatively smaller impact on the final noise reduction performance. From Fig. 13, it can be observed that all three methods significantly reduce the amplitude of the waveforms compared to the uncontrolled waveforms. It is noteworthy that in the OSPM, the waveforms tend to be rougher with higher amplitude, while in the AOSPM this phenomenon is only more pronounced in the waveform of the error noise 4, which is closest to the secondary source emitting white noise. The steady-state performance of the three methods also demonstrates similar characteristics, as illustrated in Table 3 and Fig. 14. The noise reduction performance of MCFxLMS with OSPM is the poorest, with an average noise reduction level of 9.06 dB at the four positions. On the other hand, VP-SPUFxLMS with AOSPM and MCFxLMS show similar performance, with average noise reduction levels of 17.58 dB and 15.075 dB, respectively. Among the four positions, AOSPM only exhibits slightly inferior performance at position 4, where the fourth speaker emits white noise, while achieving almost the same level of noise reduction levels at the other positions.

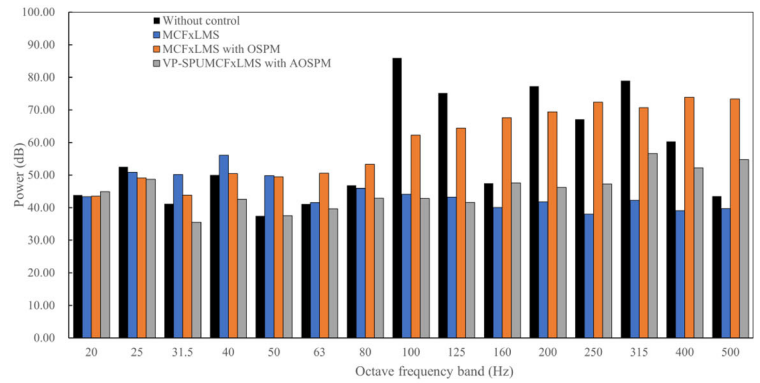
7 Conclusion

This paper proposed the VP-SPUFxLMS algorithm, which significantly reduces the computational complexity of the MCANC system while maintaining similar noise reduction performance to the traditional MCFxLMS algorithm. Additionally, the AOSPM method was introduced to overcome the challenges encountered in applying online modeling in MCANC systems. This method avoids excessive computational burden, mitigates the coupling between secondary paths modeling, and reduces the influence of modeling white noise on residual noise. The simulation results validated the performance of the proposed method. Finally, a $1 \times 4 \times 4$ MCANC system was built, and the results showed excellent performance with an average total sound pressure level reduction of 15 dB at the four test positions, which further demonstrates the outstanding potential for the proposed method in engineering applications.

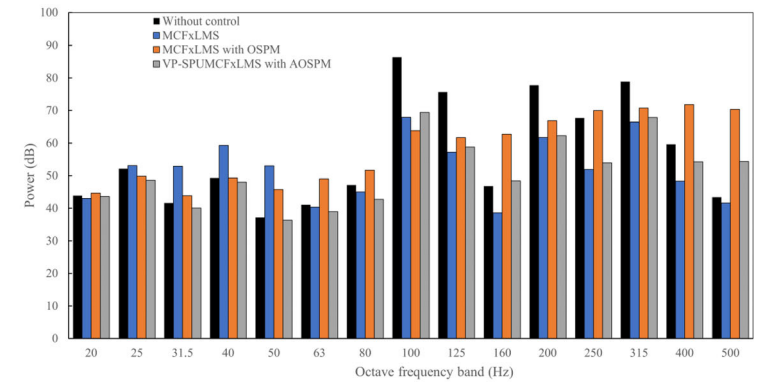
Fig. 14 The 1/3 octave power spectrum the ANC control results, when ANC, based on the MCFxLMS, MCFxLMS with OSPM, and VP-SPUMCFxLMS with AOSPM, is turned off and on



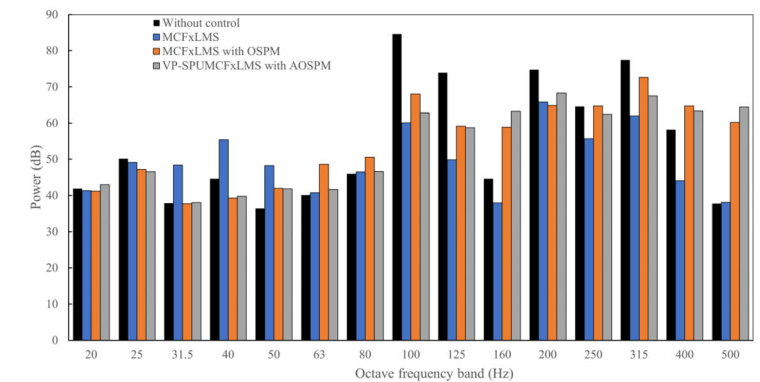
(a) Error signal 1



(b) Error signal 2



(c) Error signal 3



(d) Error signal 4

Acknowledgements The work was supported by the National Natural Science Foundation of China under Grant No.12111540251, National Key Laboratory of Strength and Structural Integrity under Grant ASSIKFJJ202303001.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bouwens, J., Mastrigt, H.V., Vink, P.: Ranking of human senses in relation to different in-flight activities contributing to the comfort experience of airplane passengers. *Int. J. Aviat. Aeronaut.* (2018). <https://doi.org/10.15394/ijaaa.2018.1228>
2. Vink, P., Vledder, G., Song, Y., Herbig, B., Reichherzer, A.S., Mansfield, N.: Aircraft interior and seat design: priorities based on passengers' opinions. *Int. J. Aviat. Aeronaut.* (2022). <https://doi.org/10.15394/ijaaa.2022.1679>
3. Kuo, S. M., Kuo, K., Gan, W. S.: Active noise control: open problems and challenges. In: The 2010 international conference on green circuits and systems (2010). <https://doi.org/10.1109/ICGCS.2010.5543076>
4. Meng, H., Chen, S.: A modified adaptive weight-constrained FxLMS algorithm for feedforward active noise control systems. *Appl. Acoust.* **164**, 107227 (2020). <https://doi.org/10.1016/j.apacoust.2020.107227>
5. Carneal, J.P., Giovanardi, M., Fuller, C.R., Palumbo, D.: Re-active passive devices for control of noise transmission through a panel. *J. Sound Vib.* **309**, 495–506 (2008). <https://doi.org/10.1016/j.jsv.2007.07.059>
6. Elliott, S.J., Nelson, P.A.: Active noise control. *IEEE Signal Proc. Mag.* **10**, 12–35 (1993). <https://doi.org/10.1109/79.248551>
7. Elliott, S.J., Nelson, P.A.: The active control of sound. *Electron. Commun. Eng. J.* **2**, 127–136 (1990). <https://doi.org/10.1049/ecej:19900032>
8. Shi, D., Lam, B., Gan, W. S., Cheer, J., Elliott, S. J.: Active noise control in the new century: The role and prospect of signal processing. In: INTER-NOISE and NOISE-CON Congress and Conference Proceedings. Institute of Noise Control Engineering pp. 5141–5151. (2023). https://doi.org/10.3397/IN_2023_0730
9. Shi, D., Gan, W.S., Lam, B., Wen, S.: Feedforward selective fixed-filter active noise control: algorithm and implementation. *IEEE-ACM Trans. Audio Speech Lang. Process.* **28**, 1479–1492 (2020). <https://doi.org/10.1109/TASLP.2020.2989582>
10. Wen, S., Gan, W.S., Shi, D.: Using empirical wavelet transform to speed up selective filtered active noise control system. *J. Acoust. Soc. Am.* **147**, 3490–3501 (2020). <https://doi.org/10.1121/10.0001220>
11. Xiao, L., Wu, M., Yang, J.: A new efficient filtered-x affine projection sign algorithm for active control of impulsive noise. *Signal Process.* **120**, 456–461 (2016). <https://doi.org/10.1016/j.sigpro.2015.09.015>
12. Lorente, J., Ferrer, M., de Diego, M., Gonzalez, A.: The frequency partitioned block modified filtered-x NLMS with orthogonal correction factors for multichannel active noise control. *Digit. Signal Process.* **43**, 47–58 (2015). <https://doi.org/10.1016/j.dsp.2015.05.003>
13. Gersho, A.: Adaptive filtering with binary reinforcement. *IEEE T. Inform. Theory* **30**, 191–199 (1984). <https://doi.org/10.1109/TIT.1984.1056890>
14. Xue, P., Liu, B.: Adaptive equalizer using finite-bit power-of-two quantizer. *IEEE Trans. Acoust. Speech Signal Process.* **34**, 1603–1611 (1986). <https://doi.org/10.1109/TASSP.1986.1164999>
15. Sethares, W.A., Johnson, C.R.: A comparison of two quantized state adaptive algorithms. *IEEE Trans. Acoust. Speech Signal Process.* **37**, 138–143 (1989). <https://doi.org/10.1109/29.17515>
16. Yang, F., Cao, Y., Wu, M., Albu, F., Yang, J.: Frequency-domain filtered-x LMS algorithms for active noise control: a review and new insights. *Appl. Sci.* **8**, 2313 (2018). <https://doi.org/10.3390/app8112313>
17. Yang, F., Yang, J.: Mean-square performance of the modified frequency-domain block LMS algorithm. *Signal Process.* **163**, 18–25 (2019). <https://doi.org/10.1016/j.sigpro.2019.04.030>
18. Rout, N.K., Das, D.P., Panda, G.: Computationally efficient algorithm for high sampling-frequency operation of active noise control. *Mech. Syst. Signal Process.* **56**, 302–319 (2015). <https://doi.org/10.1016/j.ymsp.2014.10.009>
19. Paillard, B., Le Donh, C.T., Berry, A., Nicolas, J.: Accelerating the convergence of the filtered-x LMS algorithm through transform-domain optimisation. *Mech. Syst. Signal Process.* **9**, 445–464 (1995). <https://doi.org/10.1006/mssp.1995.0035>
20. Feintuch, P.L., Bershad, N.J., Lo, A.K.: A frequency domain model for “filtered” LMS algorithms-stability analysis, design, and elimination of the training mode. *IEEE T. Signal Proces.* **41**, 1518–1531 (1993). <https://doi.org/10.1109/78.212728>
21. Shen, Q., Spanias, A.S.: Time and frequency domain X-block LMS algorithms for active noise control. *Noise Control Eng. J.* **44**, 281–293 (1996). <https://doi.org/10.3397/1.2828414>
22. Morgan, D.R.: History, applications, and subsequent development of the FXLMS Algorithm [DSP History]. *IEEE Signal Process. Mag.* **30**, 172–176 (2013). <https://doi.org/10.1109/MSP.2013.2242394>
23. Shi, D., Lam, B., Gan, W.S., Wen, S.: Block coordinate descent based algorithm for computational complexity reduction in multichannel active noise control system. *Mech. Syst. Signal Process.* **151**, 107346 (2021). <https://doi.org/10.1016/j.ymsp.2020.107346>
24. Godavarti, M., Hero, A.O.: Partial update LMS algorithms. *IEEE Trans. Signal Process.* **53**, 2382–2399 (2005). <https://doi.org/10.1109/TSP.2005.849167>
25. Douglas, S.C.: Adaptive filters employing partial updates. *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.* **44**, 209–216 (1997). <https://doi.org/10.1109/82.558455>
26. Ardekani, I.T., Abdulla, W.H.: Theoretical convergence analysis of FxLMS algorithm. *Signal Process.* **90**, 3046–3055 (2010). <https://doi.org/10.1016/j.sigpro.2010.05.009>
27. Ardekani, I.T., Abdulla, W.H.: On the convergence of real-time active noise control systems. *Signal Process.* **91**, 1262–1274 (2011). <https://doi.org/10.1016/j.sigpro.2010.12.012>
28. Kuo, S.M., Morgan, D.R.: *Active Noise Control Systems*. Wiley, New York (1996)
29. Eriksson, L.J., Allie, M.C.: Use of random noise for on-line transducer modeling in an adaptive active attenuation system. *J. Acoust. Soc. Am.* **85**, 797–802 (1989). <https://doi.org/10.1121/1.397552>
30. Zhang, M., Lan, H., Ser, W.: Cross-updated active noise control system with online secondary path modeling. *IEEE Trans. Speech Audio Process.* **9**, 598–602 (2001). <https://doi.org/10.1109/89.928924>
31. Zhang, M., Lan, H., Ser, W.: A robust online secondary path modeling method with auxiliary noise power scheduling strategy and norm constraint manipulation. *IEEE Trans. Speech Audio Process.* **11**, 45–53 (2003). <https://doi.org/10.1109/TSA.2003.805643>
32. Akhtar, M.T., Abe, M., Kawamata, M.: 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**, 720–726 (2006). <https://doi.org/10.1109/TSA.2005.855829>

33. Ahmed, S., Akhtar, M.T., Zhang, X.: Robust auxiliary-noise-power scheduling in active noise control systems with online secondary path modeling. *IEEE Trans. Audio Speech Lang. Process.* **21**, 749–761 (2012). <https://doi.org/10.1109/TASL.2012.2234112>
34. Aslam, M.S., Raja, M.A.Z.: A new adaptive strategy to improve online secondary path modeling in active noise control systems using fractional signal processing approach. *Signal Process.* **107**, 433–443 (2015). <https://doi.org/10.1016/j.sigpro.2014.04.012>
35. Ma, Y., Xiao, Y., Ma, L., Khorasani, K.: A robust feedforward hybrid active noise control system with online secondary-path modelling. *IET Signal Process.* **17**, e12183 (2023). <https://doi.org/10.1049/sil2.12183>
36. Kuo, S.M., Morgan, D.R.: Active noise control: a tutorial review. *Proc. IEEE* **87**, 943–973 (1999). <https://doi.org/10.1109/5.763310>
37. Tobias, O.J., Seara, R.: Mean weight behavior of the FXAFA LMS algorithm. *IEEE Trans. Signal Process.* **54**, 801–804 (2006). <https://doi.org/10.1109/TSP.2005.861789>

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.