MODIFIED FILTERED-X AFFINE PROJECTION SIGN ALGORITHM FOR ACTIVE CONTROL OF IMPULSIVE NOISE

Longshuai Xiao, Ming Wu, Hongling Sun, Jun Yang and Jing Tian

Key Laboratory of Noise and Vibration Research, Institute of Acoustics, Chinese Academy of Sciences, Beijing, China 100190,
e-mail: xls.ioa@gmail.com

For active noise control problems, when noise sources are contaminated by impulsive noise, the classic FxLMS converges slowly or even diverges. In order to effectively control such impulsive noise sources, by minimizing the L1-norm of an error vector, a modified filtered-x affine projection sign algorithm (MFxAPSA) has been proposed. An exact efficient implementation can greatly reduce its number of multiplications per iteration from the order of projection order timing controller length to the order of adding them. Therefore, the efficiency improvement over the direct implementation is more impressive when the projection order is high. A variable step-size extension further improves the convergence rate of the MFxAPSA without increasing the computational cost. The superior performances of the proposed algorithms are verified by both synthesized and practical impulsive noise data.

1. Introduction

Active noise control (ANC) has wide applications in the cancellation of low frequency noise, based on the principle of destructive interference of propagating acoustic waves [1]. However, when the noise sources to be controlled are contaminated by impulsive noise, the classic ANC algorithm, filtered-x least mean square (FxLMS), converges slowly or even diverges [2–9]. Many common acoustical noise [10], such as impact noise [7], road traffic noise [8], audio signals [11] and noise in the office environments [12], is impulsive, which can be more accurately modeled by alpha stable distribution than Gaussian distribution. The characteristic function of the standard αS distribution is \( \phi(t) = \exp(-|t|^\alpha) \). A smaller \( \alpha \) indicates the heavier tail of the density function. As we know that, for the non-Gaussian stable distribution (\( \alpha < 2 \)), the second-order moment is infinite [13]. Therefore, the adaptive algorithms based on the second order moment, such as FxLMS, are severely degraded by impulsive interferences.

In order to make adaptive filters work well under the non-Gaussian impulsive environments, much work has been done, particularly for the ANC. The state-of-the-art Active impulsive noise control (AINC) algorithms can be classified into four types. The first type is based on lower order moment method, including filtered-x least mean p-norm (FxLMP) [2], which is obtained by minimizing fractional lower order statistics (FLOS) [13], and FxlogLMS [7], which uses zero order statistics (ZOS) [14]. FxLMP has better convergence performance than FxLMS for impulsive noise, but needs accurate \( \alpha \) estimates. The on-line estimation of such parameter can solve this problem.
[8], but it converges still slowly compared to other state-of-the-art AINC algorithms. FxlogLMS only works well for the very impulsive noise [9]. Based on robust statistics [15], by minimizing a modified M-estimator, filtered-x least mean M-estimate(FxLMM) [9] is an example of second type. The performance improvement of the FxLMM is very limited. The third type modifies samples/error signals, such as FxSunLMS [3] and FxLMM, both of which discard large amplitude samples, filtered-x modified least mean square (FxMLMS) [4] and FxMLMP [6], to replace large amplitude samples and error signals with threshold values. The modification method need an accurate estimate of the interval parameters, performed by off-line operations. Normalization method is the fourth type, including filtered-x modified normalized LMS (FxMNLMS) [5], FxMNLMP [6] and FxMNLMM [9]. The normalization method needs an additional L more multiplications. Our proposed algorithms are based on the minimization of the $L_1$-norm of an error vector instead of the $L_2$-norm. Hence, our algorithms should have robust performance under impulsive environments. Therefore, the proposed algorithms belong to the robust statistics method. And the MFxAPSA has normalization operations of input vectors.

Our work extends the affine projection sign algorithm (APSA) [16], its efficient implementation [17] and variable step-size version [18], to provide effective and efficient alternatives to the state-of-the-art AINC algorithms. The contribution is three-fold. Firstly, extending the APSA to the AINC results in the MFxAPSA. The extension is intractable, because of the inaccessible to the desired signal. Furthermore, for the AINC problems, both inputs and desired signals are impulsive instead of interferences. Secondly, large projection order accelerates the convergence rate of the MFxAPSA obviously with only a little steady-state performance degradation. And the efficient implementation makes the increase of cost is limited with larger projection order. Furthermore, the proposed algorithms still works well even when projection order is larger than the controller length. Thirdly, both the MFxAPSA and VSS-MFxAPSA outperform the state-of-the-art AINC algorithms for sources with severe impulsive corruptions.

2. The Proposed AINC Algorithms

2.1 The MFxAPSA

We extend the APSA [16] to the AINC problems. However, the desired signal $d(n)$ in the AINC can not be accessible directly, we use the modified (delay compensation) structure to estimate the desired signal at first. Furthermore, because of the existence of the secondary path, reference signal must be filtered first by an estimate of this path to update the weight. Based on the delay compensation and filtered-x structure, similar as the APSA, we minimize the $L_1$-norm of the an estimated a posterior error vector to derive the MFxAPSA. The model of the proposed algorithm is

$$
\min_{\mathbf{w}(n+1)} \| \mathbf{\hat{e}}_p(n) \|_1 \\
\text{s.t.} \| \mathbf{w}(n+1) - \mathbf{w}(n) \|_2^2 \leq \mu^2, \tag{1}
$$

where $\mathbf{\hat{e}}_p(n) = \mathbf{\hat{d}}(n) - \mathbf{X}_f^T(n)\mathbf{w}(n+1)$, and $T$ represents the transpose operation. Solving the model, we get the update equation of the proposed MFxAPSA

$$
\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \mathbf{x}_e(n) / \sqrt{\| \mathbf{x}_e(n) \|_2^2 + \epsilon}, \tag{2}
$$

where we use a prior error vector $\mathbf{\hat{e}}_a(n)$ instead of $\mathbf{\hat{e}}_p(n)$. Define

$$
\mathbf{\hat{e}}_a(n) = \mathbf{\hat{d}}(n) - \mathbf{X}_f^T(n)\mathbf{w}(n), \tag{3}
$$

$$
\mathbf{x}_e(n) = \mathbf{X}_f(n) \text{sgn}(\mathbf{\hat{e}}_a(n)), \tag{4}
$$
where \( \text{sgn} \) is a sign function. Define \( \hat{d}(n) = [\hat{d}(n), \hat{d}(n-1), \ldots, \hat{d}(n-P+1)]^T \), \( \mathbf{X}_f(n) = [\mathbf{x}_f(n), \mathbf{x}_f(n-1), \ldots, \mathbf{x}_f(n-P+1)] \), \( \mathbf{x}_f(n) = [x_f(n), x_f(n-1), \ldots, x_f(n-L+1)]^T \), \( \mathbf{x}_M(n) = [x(n), x(n-1), \ldots, x(n-M+1)]^T \), \( \mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-L+1)]^T \), \( y(n) = [y(n), y(n-1), \ldots, y(n-M+1)]^T \) and \( \hat{s} \) as an estimated secondary path, then

\[
x_f(n) = \hat{s}^T \mathbf{x}_M(n),
\]

\[
y(n) = \mathbf{x}^T(n) \mathbf{w}(n),
\]

\[
e(n) = d(n) - s^T y(n),
\]

\[
d(n) = e(n) + s^T y(n).
\]

For AINC problems, \( x(n) \) and \( d(n) \) are impulsive, according to the stability property of stable distribution [13], the linear combination \( \hat{e}_p(n) \) is impulsive and a stable vector process. As is well known [15], minimizing \( L_1 \)-norm rather than \( L_2 \)-norm can lead to robust estimates. So it’s quite reasonable for the proposed algorithms to be applicable for the AINC problems.

Furthermore, the sign function of the error vector and normalization of input vector make the weight gradient be less impulsive, thus the learning curve of the MFxAPSA is much smooth. Furthermore, thanks to the sign function of the error vector, the number of multiplications for this direct implementation is \( PL + 3L + 2M + 1 \), much less than that of the standard APA [19].

### 2.2 Exact Efficient Implementation of the MFxAPSA

The main computational cost of the MFxAPSA is \( \mathcal{O}(PL) \), which limits the projection order to be small for real-time applications. By using the time-shift structure of input vectors, Ni [17] reduces the complexity of the APSA to the first order of \( L \) and \( P \). Similarly, by this mathematically equivalent description, we get an exact efficient implementation of the MFxAPSA. We rewrite the weight update equation of the MFxAPSA at time \( n \) as

\[
\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(n) \mathbf{x}_e(n),
\]

\[
\mu(n) = \mu / \sqrt{\text{sgn}(\hat{e}_p(n))} \mathbf{R}_f(n) \text{sgn}(\hat{e}_a(n)) + \epsilon,
\]

\[
\mathbf{R}_f(n) = \mathbf{X}_f^T(n) \mathbf{X}_f(n)
\]

\[
= \begin{bmatrix}
[\mathbf{r}_f(n)]_{0} & [\mathbf{r}_f^T(n)]_{1:P-1} \\
[\mathbf{r}_f(n)]_{1:P-1} & [\mathbf{R}_f(n-1)]_{0:P-2,0:P-2}
\end{bmatrix},
\]

\[
[r_f(n)]_p = \mathbf{x}_f^T(n) \mathbf{x}_f(n-p), \quad p = 0, 1, \ldots, P
\]

\[
= [r_f(n-1)]_p + x_f(n)x_f(n-p) - x_f(n-L)x_f(n-p-L).
\]

Substituting \( \mathbf{w}(n) \) into \( \hat{e}_a(n) \) with its previous value, we get

\[
\hat{e}_a(n) = \left[ \hat{d}(n) - \mathbf{x}_f^T(n) \mathbf{w}(n-1) \right] - \mu(n-1) \mathbf{S}_f(n) \text{sgn}(\hat{e}_a(n-1)),
\]

\[
\mathbf{S}_f(n) = \mathbf{X}_f^T(n) \mathbf{X}_f(n-1) = \begin{bmatrix}
[\mathbf{r}_f(n)]_{1:P} \\
[\mathbf{R}_f(n-1)]_{0:P-2,0:P-1}
\end{bmatrix}.
\]

The computational cost of \( \hat{e}_a(n) \) has been greatly reduced from \( PL \) to \( L + P \) multiplications. Then the total number of multiplications of this exact efficient implementation is \( 3L + 2M + 3P + 3 \), which is the first-order complexity of controller length, secondary order and projection order. We will benefit from the efficient implementation further owing to the superior performance of the MFxAPSA shown in the simulation section.
Table 1. Number of Multiplications of AINC Algorithms

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>×</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard FxLMS</td>
<td>$2L + M$</td>
</tr>
<tr>
<td>Lower Order Moment Method</td>
<td>$2L + M$</td>
</tr>
<tr>
<td>Robust Statistics Method</td>
<td>$2L + M$</td>
</tr>
<tr>
<td>Modification Method</td>
<td>$3L + M$</td>
</tr>
<tr>
<td>Normalization Method</td>
<td>$3L + M$</td>
</tr>
<tr>
<td>Direct MFxAPSA</td>
<td>$(P + 3)L + 2M$</td>
</tr>
<tr>
<td>Efficient MFxAPSA</td>
<td>$3L + 2M + 3P^*$</td>
</tr>
<tr>
<td>VSS-MFxAPSA</td>
<td>$3L + 2M + 3P^*$</td>
</tr>
</tbody>
</table>

### 2.3 Variable Step-Size of the MFxAPSA

Shin [18] designed the optimal step-size for the APSA based on the minimization of the mean square deviation. We extend this result to the MFxAPSA, to accelerate its convergence speed. We assume the desired signal vector can be written as \( \mathbf{d}(n) = \mathbf{X}_f(n)\mathbf{w}_o + \mathbf{v}(n) \), where \( \mathbf{w}_o \) is the asymptotic optimal controller. We define the weight deviation vector as \( \mathbf{\tilde{w}}(n) = \mathbf{w}_o - \mathbf{w}(n) \). Then, the weight deviation update equation is given by

\[
\mathbf{\tilde{w}}(n+1) = \mathbf{\tilde{w}}(n) - \mu(n)\mathbf{x}_e(n).
\] (15)

Hence the mean square deviation equation is

\[
E\|\mathbf{\tilde{w}}(n+1)\|^2 = E\|\mathbf{\tilde{w}}(n)\|^2 - 2E\mu(n)\mathbf{\tilde{w}}^T(n)\mathbf{x}_e(n) + \mu^2,
\] (16)

where \( E \) is an expectation operation. The optimal step-size should maximize the cost function

\[
J(\mu) = 2E\mu(n)\mathbf{\tilde{w}}^T(n)\mathbf{x}_e(n) - \mu^2.
\] (17)

Solving the function, we approximate the optimal step-size, get

\[
\gamma(n) = \lambda\gamma(n-1) + (1-\lambda)\min\left(\frac{\|\mathbf{e}_a(n-1)\|_1}{\sqrt{\beta(n-1)}}, \gamma(n-1)\right),
\] (18)

\[
\beta(n) = \text{sgn}(\mathbf{e}_a^T(n))\mathbf{R}_f(n)\text{sgn}(\mathbf{e}_a(n)) + \epsilon,
\] (19)

\[
\mathbf{w}(n+1) = \mathbf{w}(n) + \tilde{\mu}(n)\mathbf{x}_e(n),
\] (20)

where \( \tilde{\mu}(n) = \gamma(n)/\sqrt{\beta(n)} \). Combining the exact efficient implementation of the MFxAPSA with this VSS version, the final computational cost of the VSS-MFxAPSA is \( 3L + 2M + 3P^* + 7 \) multiplications, nearly the same as that of the MFxAPSA.

### 2.4 Computational Complexity

The computational complexity (number of multiplications) results are shown in Table 1, where we ignore constant number of operations. For the lower order moment method, we neglect fractional power or logarithmic computations of error signals. And for the FxMNLMP, we count each fractional power of input sample as one multiplication for simplicity. The direct implementation of the MFxAPSA takes at least \( PL \) more multiplications than FxLMS and existing AINC algorithms. The exact efficient implementation of the fixed and variable step-size version reduces the additional cost to \( L + M + 3P \) or \( M + 3P^* \), which is less important considering the benefits of the proposed algorithms.
3. Simulation

The performance of the proposed MFxAPSA and VSS-MFxAPSA will be validated by simulation results. The independent identical impulsive noise sequences are generated by standard SαS distribution \cite{6, 9}. We choose $\alpha = 0.6$ for the very impulsive case \cite{14}, and $\alpha = 1.2$ for the general impulsive noise case. We use the primary and secondary path data measured in \cite{1}. The primary path, secondary path and controller are all modeled as finite impulse response (FIR) with the order of 256, 128 and 192, respectively. Assume the estimated path equals the true one. The averaged noise reduction (ANR) is used as the performance evaluation criterion. Define $ANR(dB) = 20\log_{10}(A_e(n)/A_d(n))$, where $A_e(n) = \lambda A_e(n) + (1 - \lambda)|e(n)|$ and $A_d(n) = \lambda A_d(n) + (1 - \lambda)|d(n)|$, $0.9 \leq \lambda \leq 1$. The ANR curves are averaged over 25 independent runs.

3.1 Parameters Selection

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1}
\caption{Parameter selections for the MFxAPSA and VSS-MFxAPSA. MFxAPSA, $\alpha = 1.2$: (a) $P = 64$; (b) $\mu = 5e^{-4}$. VSS-MFxAPSA: (c) $\alpha = 0.6$; (d) $\alpha = 1.2$;}
\end{figure}

On a trial-and-error basis, we run the algorithms with sets of possible parameters values. Results are shown in Fig. 1. For (a), the ANR curves of the MFxAPSA under different step-size $\mu$ values with projection order $P = 64$. Large step-size has faster convergence speed at the cost of pooper steady-state ANR performance. The suitable step-size of the MFxAPSA is $5e^{-4}$. For (b), ANR curves of the MFxAPSA vary with different $P$ values, ranging from 1 to 64 when $\mu = 5e^{-4}$. It is obvious that smaller projection order leads to slower ANR convergence curves just as we know. However, the interesting discovery is that the increased projection order won’t obviously degrade the
steady-state ANR performance with much improved convergence rate at the same time. Moreover, with the benefit of the exact efficient implementation of the MFxAPSA, the increase of the projection order won’t obviously increase the total complexity. The property still holds even when the projection order gets larger than the controller length. Thus, we let \( P = 64 \) for the MFxAPSA. The ANR curves of the VSS-MFxAPSA are shown in (c) and (d). (c) is for very impulsive case with \( \alpha = 0.6 \), the suitable projection order is 64. The result for \( \alpha = 1.2 \) is shown in (d), where the best performance is achieved when \( P = 8 \).

### 3.2 Performance Comparisons

![Graph](image1)

**Figure 2.** Compare the proposed algorithms with the state-of-the-art AINC algorithms for SoS noise sources. Original signal \( d(n) \) and residual signals of: A1, FxLMS; A2, FxLMP; A3, FxlogLMS; A4, FxLMM; A5, FxMLMP; A6, FxMNLM; A7, FxMNLM; A8, MFxAPSA; A9, VSS-MFxAPSA

We compare the performance of the proposed algorithms with state-of-the-art AINC algorithms, mentioned in the introduction section. The parameters values used lead to stable and fast convergence for these algorithms. For modification method, the percent of interval estimation is [0.0005, 0.9995] [4]. For the FxLMP, FxMLMP, FxMNLM, \( p = \alpha - 0.01 \). The results are shown in Fig. 2. Sub-figure (a) shows all ANR curves when \( \alpha = 0.6 \), the very impulsive noise case. All existing algorithms diverge or converge very slowly. The best of these is FxMNLM, significantly outperformed by the proposed algorithms. For the general impulsive input case (b), all the state-of-the-art AINC algorithms have improved convergence rate. In this case, the performance of the MFxAPSA is similar as that of the FxMNLM, still better than all other AINC algorithms. Again, VSS-MFxAPSA converges much faster than all the existing algorithms. Waveforms of error signals before and after control for different
Figure 3. Simulation results of pile driving noise. (a) Original noise. (b) Residual noise of the FxlogLMS algorithm. (c) Residual noise of the MFxAPSA. (d) Residual noise of the VSS-MFxAPSA.

algorithms when $\alpha = 1.2$ are shown in (c). Compared with primary noise $d(n)$ and other error signals, the error signals for the proposed algorithms have much less and smaller impulsive instant values.

Simulation results with pile driving noise [20] are also shown in Fig. 3. Compared with the FxlogLMS algorithm, the proposed MFxAPSA and VSS-MFxAPSA converge more quickly to smaller residual noise levels, where the performance of VSS-MFxAPSA is more impressive.

4. CONCLUSION

Minimizing the $L_1$-norm of the estimated a priori error vector, the MFxAPSA has been proposed to effectively control impulsive noise, modeled by the $\text{SoS}$ distribution. The exact efficient implementation can significantly reduce its complexity. The VSS-MFxAPSA can dramatically improve the convergence rate. For the MFxAPSA, much higher projection order, even higher than the controller length, can be adopted to greatly improve the convergence rate, with only a little degradation in the steady state performance, at a cost of a limited increase of complexity with the benefit of the exact efficient implementation. Compared to the state-of-the-art AINC algorithms, the MFxAPSA has outstanding performance for the very impulsive noise ($\alpha < 1$) and comparable performance for the general one ($\alpha \geq 1$). With proper projection order, the VSS-MFxAPSA can provide the overall best performance in the whole range of $\alpha$, with the same order of complexity as the MFxAPSA. Comparing the performance of the proposed algorithms with existing AINC algorithms using modified structure is a future work.

ACKNOWLEDGEMENTS

This research was supported by the National key Basic Research Program of China (973 Program) under Grant No. 2013CB632900.

References


