



An efficient offline scheme to compute an FIR controller for active reduction of acoustic reflections in an anechoic chamber

R. Haasjes^{a,*}, A.P. Berkhoff^{a,b}

^a University of Twente, Faculty of Engineering Technology, Applied Mechanics and Data Analysis, Drienerlolaan 5, Enschede, 7522 NG, The Netherlands

^b TNO Acoustics and Sonar, Oude Waalsdorperweg 63, The Hague, 2597 AK, The Netherlands

ARTICLE INFO

Keywords:

Fixed gain control
Causality
Preconditioning
Acoustic anechoic chamber
Kirchhoff–Helmholtz integral

ABSTRACT

This paper presents an offline iterative scheme to efficiently obtain a fixed controller for larger-scale active noise control systems. The approach reduces the computational complexity of the scheme by applying preconditioning and prewhitening via the frequency domain, followed by conversion of the filters to the time domain using the inverse Fourier transform. Although the filters are not necessarily causal, causality is ensured by utilizing delay and truncation techniques. The use of these filters in the iterative scheme leads to improved convergence, lowering computational effort while also being memory efficient. Regularization is applied to maintain convergence while allowing for larger step sizes between iterations. The controller minimizes the reflected sound field, which is measured by the performance sensors. The signals of the performance sensors are computed with the Kirchhoff–Helmholtz integral. The results show that this allows for global control of the reflected sound field within the microphone area. A simulation is shown of a noise control setup with 12 primary sources, 12 secondary sources, 12 reference sensors, 12 error sensors and 37 performance sensors. Although the delay of the filters in the direct path is reflected in the filter coefficients, this delay can be truncated to obtain filter coefficients without any delay. Despite the truncation resulting in an average loss in performance of 1.6 dB, the control system without any delays is able to reduce the average reflected sound at the performance sensors by 13.8 dB, when the reference signal is known.

1. Introduction

Acoustic anechoic chambers are designed to absorb acoustic waves at its boundaries, effectively creating a free space environment that simulates an infinitely large chamber. These chambers are commonly used for transfer function measurements, sensor calibration, certification of machinery and more [1–3].

In the ideal case, all acoustic waves are absorbed at the walls, eliminating the presence of reflections. However, passive absorption measures have limitations, resulting in low-frequency sound waves reflecting off the walls [4,5], in some cases even above the cut-off frequency the chamber is designed for. This was demonstrated in a study by [4], where an anechoic chamber designed with a cut-off frequency of 75 Hz showed perturbations in the 110 to 160 Hz frequency range, increasing the cut-off frequency to about 200 Hz. Design principles of an acoustic anechoic chamber are described in [6–8], which highlight the design efforts required to lower the cut-off frequency.

* Corresponding author.

E-mail address: r.haasjes@utwente.nl (R. Haasjes).

<https://doi.org/10.1016/j.jsv.2023.118198>

Received 1 May 2023; Received in revised form 30 November 2023; Accepted 3 December 2023

Available online 4 December 2023

0022-460X/© 2024 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Active noise control is effective at lower frequencies, which makes this a promising addition to the passive wall absorption to extend the operable frequency range of an acoustic anechoic chamber. A detailed overview of research directions in active noise control is presented in [9]. The problem of input shaping for a known or estimated plant is solved in other relevant fields, examples are optimal tracking [10,11], model predictive control [12] and inverse system control [13]. A physical approach to expand a finite-size laboratory with a virtual domain using active boundary control, utilizing immersive boundary conditions is shown by [14]. However, this method relies on ideal sources, requiring calibration for which a method is not shown. The immersive boundary conditions have been extended to cloaking and holography [15] and elastic immersive wave experimentations [16,17]. Active control in an anechoic room is shown in [18], active control of scattered acoustic radiation in [19] and real-time active suppression of scattered acoustic radiation in [20]. The methods in [18–20] apply scattering filters to estimate the reflected sound field. The use of scattering filters introduces difficulties, for which solution strategies are suggested. Furthermore, in [19,20] it is stated that the performance is limited by the number of channels that is manageable by today's standards, and thus call for efficient methods to solve larger-scale systems.

The use of measured transfer functions eliminates a calibration stage and enables the design of a system with ordinary sources and sensors, thus reducing the cost. Assuming that the reference signals are stationary in time and that all involved systems are linear time invariant (LTI), a typical approach that incorporates measured transfer functions of the secondary path is to solve the Wiener filtering problem [21], to find an H_2 optimal fixed time domain filter. This requires solving a system of linear equations containing a Toeplitz-block matrix. Efficient algorithms exploit knowledge about the matrix structure to reduce the computational complexity. The Levinson algorithm [22] is well-known, which solves the system with $\mathcal{O}(n^2)$ complexity, where n is the size of the Toeplitz-block matrix.

Low-rank approximation methods such as sequentially semi-separable [23] and Hierarchically semi-separable [24] show promising efficiency enhancements for different structured matrices. However, it is unknown whether this can be generalized to block-structured Toeplitz matrices [23]. The optimal control problem can also be expressed in the frequency domain, which is computationally advantageous, but not necessarily causal [21].

Solving the Wiener filtering problem in an iterative manner with the preconditioned conjugate gradient method (PCG) is shown using circulant preconditioning for Toeplitz-block matrices [25]. For single-channel systems, this approach has a complexity of $\mathcal{O}(n \log(n))$. Other preconditioners have been introduced as well [26,27]. Despite being efficient, the PCG method requires storing the first columns and rows of the blocks within the Toeplitz-block matrix, resulting in increasing memory requirements for growing system sizes.

The filtered-reference least mean squares and the filtered-error least mean squares (FeLMS) algorithms [21] are well-known adaptive algorithms in the time domain. These algorithms assume that the reference signals are stationary in time and that all involved systems are LTI. Provided these adaptive algorithms are stable, they will converge to the optimal Wiener filter in the steady-state [28]. The memory consumption of these adaptive algorithms scales well with growing system sizes. Between these two, the FeLMS algorithm has a significant lower computational complexity in case of systems with multiple reference- or error-signals. The convergence speed of the FeLMS algorithm can be a challenge [28], requiring preconditioning methods. A method to prewhiten the reference signals and apply preconditioning and decoupling to the system using filters derived in the frequency domain is shown in [29]. Constructing these filters in the frequency domain results in a lower computational complexity, with the downside of non-guaranteed causality. To mitigate the causality problem, the filters are delayed and truncated in the time domain, ensuring causality. This approach is similar to an approach shown in [30] where a cyclic shift of the time domain samples is applied after the Fourier Transform. A modified FeLMS algorithm is shown in [31], which applies preconditioning using inner–outer factorization, regularization of the system and a method to compensate the delay for improved convergence. However, the inner–outer factorization is costly for systems with a higher number of channels. Furthermore, the inner–outer factorization requires a state–space model that is not always available.

The Kirchhoff–Helmholtz integral can be used to compute the sound field from the measured acoustic pressure and the particle velocity on an enclosed contour [32,33]. In case of a sound source positioned within the enclosed contour, the inverse Kirchhoff–Helmholtz integral should be used. On the other hand, if the sound source is positioned outside of the enclosed contour, the forward Kirchhoff–Helmholtz integral should be used [34], which is the case with sound waves reflecting from the walls. This method allows to separate the direct sound field from the reflected sound field, which is a necessity for active suppression of the sound reflected from the walls.

The contribution of this paper is a scheme in which the prewhitening and preconditioning filters are derived in the frequency domain, combining the work of [29,31]. Besides the application of a cross-talk cancellation system as shown in [29], this paper shows that the proposed frequency domain approach can also be applied in active control applications. This results in an algorithm with low computational complexity and small memory consumption to compute a finite impulse response (FIR) control filter, using measured transfer functions. The algorithm acquires the performance signals computed with the Kirchhoff–Helmholtz integral, which allows for global control of the reflected sound field within the microphone area. The proposed method is applied to a multichannel active noise control system to show both the achievable performance and the efficiency of the proposed scheme compared to existing algorithms. The simulation demonstrates a method to attenuate the reflected sound field, in an effort to work towards an active acoustic anechoic chamber.

M -by- $(L + M)$ to obtain $\mathbf{e}'(n)$. The signal $\mathbf{v}'(n)$ is a delayed version of $\mathbf{v}(n)$, written as:

$$\mathbf{v}'(n) = \Delta_{\text{ai}}(z)\mathbf{v}(n), \quad (1)$$

in which $\Delta_{\text{ai}}(z)$ is the simulation delay operator corresponding to the delay of Δ_{ai} samples in the filter $\hat{\mathbf{G}}_{\text{ai}}(z)$. Similarly, the signal $\mathbf{y}'(n)$ is a delayed version of the controller output, written as:

$$\mathbf{y}'(n) = \Delta_{\text{ai}}(z)\mathbf{C}(z)\mathbf{v}(n). \quad (2)$$

The secondary path $\mathbf{G}(e^{j\omega})$ is regularized to reduce the controller output at frequencies where the secondary path has small values to prevent large amplifications, resulting in the augmented filter $\bar{\mathbf{G}}(z)$ of dimensions $(L + M)$ -by- M .

The update rule for the controller is:

$$\mathbf{C}_i(n + 1) = \mathbf{C}_i(n) - \alpha \mathbf{e}''(n) \mathbf{v}^T(n - i), \quad i = 0, \dots, I - 1, \quad (3)$$

in which the superscript T denotes the matrix transpose, i denotes the tap-delay of the FIR coefficients, assuming a FIR filter length I and α is the convergence coefficient defined as:

$$\alpha = \frac{\alpha_0}{\bar{r}}, \quad (4)$$

such that α is normalized by the input power and α_0 should be ~ 1 . The normalization factor \bar{r} is computed as the norm of the filtered-reference signal $\mathbf{r}(n)$ following:

$$\bar{r} = \|\hat{\mathbf{G}}(z)\hat{\mathbf{G}}_{\text{mi}}(z)\hat{\mathbf{F}}_w(z)\mathbf{x}(n)\|^2. \quad (5)$$

After running the proposed algorithm, a controller $\mathbf{C}(z)$ is obtained, which is converted to the desired controller $\mathbf{W}(z)$ following:

$$\mathbf{W}(z) := \hat{\mathbf{G}}_{\text{mi}}(z)\mathbf{C}(z)\hat{\mathbf{F}}_w(z), \quad (6)$$

such that this controller can be implemented following the feedforward configuration as shown in Fig. 1.

2.1. Frequency domain formulation of the prewhitening and decorrelation filter

Starting from K discrete-time real-valued random reference signals which are assumed to be stationary but possibly correlated, being described by the matrix:

$$\mathbf{x}(n) = [x_1(n), \dots, x_k(n)]^T. \quad (7)$$

The correlation matrix of the signal $x(n)$ can be written as:

$$\mathbf{R}_{\text{xx}}(m) = E[\mathbf{x}(n + m)\mathbf{x}^T(n)], \quad -\infty < m < \infty. \quad (8)$$

Using the discrete-time Fourier transform, the correlation matrix can be written as the power spectral density (PSD) matrix [29]:

$$\mathbf{S}_{\text{xx}}(e^{j\omega}) = \sum_{m=-\infty}^{\infty} \mathbf{R}_{\text{xx}}(m)e^{-j\omega m}, \quad \mathbf{j} = \sqrt{-1}, \quad (9)$$

which can be computed using the Welch method. The K -by- K power spectral density matrix $\mathbf{S}_{\text{xx}}(e^{j\omega})$ is a positive semi-definite Hermitian matrix and therefore an eigenvalue decomposition (EVD) always exists [35], which can be computed for all frequencies:

$$\mathbf{S}_{\text{xx}}(e^{j\omega}) = \mathbf{Q}(e^{j\omega})\mathbf{D}(e^{j\omega})\mathbf{Q}^H(e^{j\omega}), \quad (10)$$

in which the superscript H denotes the Hermitian transpose, $\mathbf{D}(e^{j\omega})$ is a diagonal matrix containing the real eigenvalues and $\mathbf{Q}(e^{j\omega})$ contains the normalized eigenvectors. The prewhitening and decorrelation filter can be written as [29]:

$$\mathbf{F}_w(e^{j\omega}) = \mathbf{D}^{-\frac{1}{2}}(e^{j\omega})\mathbf{Q}^H(e^{j\omega}), \quad (11)$$

which applied to the reference signals gives:

$$\mathbf{v}(e^{j\omega}) = \mathbf{F}_w(e^{j\omega})\mathbf{x}(e^{j\omega}), \quad (12)$$

such that the signals $\mathbf{x}(e^{j\omega})$ are whitened and decorrelated.

The conditions for $\mathbf{D}^{-\frac{1}{2}}(e^{j\omega})$ in Eq. (11) to exist, are mentioned in [29]. To improve the conditioning of the matrix $\mathbf{D}(e^{j\omega})$, regularization is applied:

$$\bar{\mathbf{D}}(e^{j\omega}) = \mathbf{D}(e^{j\omega}) + \beta_F \mathbf{I}_K, \quad (13)$$

where \mathbf{I}_K is an identity matrix of size K -by- K , with the regularization parameter β_F . The regularization parameter β_F improves the conditioning of the matrix, but should typically be as small as possible because it changes the filter. Then the regularized filter is defined as:

$$\bar{\mathbf{F}}_w(e^{j\omega}) = \bar{\mathbf{D}}^{-\frac{1}{2}}(e^{j\omega})\mathbf{Q}^H(e^{j\omega}). \quad (14)$$

2.2. Frequency domain formulation of the system's preconditioning and decoupling filters

Similar to the prewhitening and decorrelation filter, a system can be decoupled by orthogonal transformations. Because a system matrix is possibly non-square and in general not Hermitian, a singular value decomposition (SVD) is used instead of the EVD [29]. Assume that a stable MIMO linear time-invariant system with M inputs and L outputs is described by the L -by- M frequency response matrix $\mathbf{G}(e^{j\omega})$. Application of regularization results in the $(L + M)$ -by- M augmented plant $\tilde{\mathbf{G}}(e^{j\omega})$ [31]:

$$\tilde{\mathbf{G}}(e^{j\omega}) = \begin{bmatrix} \mathbf{G}(e^{j\omega}) \\ \mathbf{G}_{\text{reg}}(e^{j\omega}) \end{bmatrix}, \quad (15)$$

where the regularizing transfer function is defined as:

$$\mathbf{G}_{\text{reg}}(e^{j\omega}) = \sqrt{\beta_G} \mathbf{I}_M, \quad (16)$$

in which β_G is the regularization parameter. The regularization parameter β_G prevents high gains in $\hat{\mathbf{G}}_{\text{mi}}(e^{j\omega})$ that might occur due to the inverse and therefore prevents saturation of the control signal [31]. This parameter should be chosen in the range of expected reduction levels.

Computing the singular value decomposition of the secondary path at each frequency:

$$\tilde{\mathbf{G}}(e^{j\omega}) = \tilde{\mathbf{U}}(e^{j\omega}) \tilde{\mathbf{\Sigma}}(e^{j\omega}) \tilde{\mathbf{V}}^H(e^{j\omega}), \quad (17)$$

where the dimensions of $\tilde{\mathbf{U}}(e^{j\omega})$ are $(L + M)$ -by- M , the dimensions of $\tilde{\mathbf{\Sigma}}(e^{j\omega})$ are M -by- M and the dimensions of $\tilde{\mathbf{V}}(e^{j\omega})$ are M -by- M .

The preconditioner of the direct path is written as:

$$\tilde{\mathbf{G}}_{\text{mi}}(e^{j\omega}) = \tilde{\mathbf{V}}(e^{j\omega}) \tilde{\mathbf{\Sigma}}^{-1}(e^{j\omega}), \quad (18)$$

which results in the preconditioned direct path:

$$\begin{aligned} \tilde{\mathbf{G}}(e^{j\omega}) \tilde{\mathbf{G}}_{\text{mi}}(e^{j\omega}) &= \\ \tilde{\mathbf{U}}(e^{j\omega}) \tilde{\mathbf{\Sigma}}(e^{j\omega}) \tilde{\mathbf{V}}^H(e^{j\omega}) \tilde{\mathbf{V}}(e^{j\omega}) \tilde{\mathbf{\Sigma}}^{-1}(e^{j\omega}) &= \\ \tilde{\mathbf{U}}(e^{j\omega}). \end{aligned} \quad (19)$$

The decoupling of the system is achieved by choosing

$$\tilde{\mathbf{G}}_{\text{ai}}(e^{j\omega}) = \tilde{\mathbf{U}}^H(e^{j\omega}), \quad (20)$$

such that

$$\tilde{\mathbf{G}}_{\text{ai}}(e^{j\omega}) \tilde{\mathbf{G}}(e^{j\omega}) \tilde{\mathbf{G}}_{\text{mi}}(e^{j\omega}) = \tilde{\mathbf{U}}^H(e^{j\omega}) \tilde{\mathbf{U}}(e^{j\omega}) = \tilde{\mathbf{I}}, \quad (21)$$

where $\tilde{\mathbf{I}} \approx \mathbf{I}$, such that the system is diagonalized.

2.3. Time domain formulation of the frequency domain filters

The frequency response matrices can be converted to impulse response matrices using the inverse discrete-time Fourier transform. The discrete Fourier transform could introduce wrap-around effects, and together with the truncation effects this might result in non-causality of the time domain filters. Causality of the filters must be ensured by introducing appropriate delay and truncation. These procedures are required for both the prewhitening and preconditioning filters.

• Prewhitening filter:

1. Having the frequency response matrix $\tilde{\mathbf{F}}_w(e^{j\omega})$, taking the inverse discrete Fourier transform results in the impulse response matrix $\tilde{\mathbf{F}}_w(z)$, with reference to the symmetry property of the frequency response matrix: $\tilde{\mathbf{F}}_w(e^{-j\omega}) = \tilde{\mathbf{F}}_w^*(e^{j\omega})$, in which the superscript $*$ denotes the complex conjugate.
2. Delay $\tilde{\mathbf{F}}_w(z)$ with an appropriate number of sample delays denoted by Δ_F to ensure causality of all entries of the impulse response matrix, to obtain a new impulse response matrix $\tilde{\tilde{\mathbf{F}}}_w(z)$.
3. Truncate the non-causal part of $\tilde{\tilde{\mathbf{F}}}_w(z)$ to obtain the coefficients of FIR filters $\hat{\tilde{\mathbf{F}}}_w(z)$ of length O .

• Preconditioning filters:

1. Convert the frequency response matrices $\tilde{\mathbf{G}}_{\text{mi}}(e^{j\omega})$ and $\tilde{\mathbf{G}}_{\text{ai}}(e^{j\omega})$ to the impulse response matrices $\tilde{\mathbf{G}}_{\text{mi}}(z)$ and $\tilde{\mathbf{G}}_{\text{ai}}(z)$ by taking the inverse discrete Fourier transform, with reference to the symmetry property of the frequency response matrices: $\tilde{\mathbf{G}}_{\text{mi}}(e^{-j\omega}) = \tilde{\mathbf{G}}_{\text{mi}}^*(e^{j\omega})$ and $\tilde{\mathbf{G}}_{\text{ai}}(e^{-j\omega}) = \tilde{\mathbf{G}}_{\text{ai}}^*(e^{j\omega})$.
2. Delay $\tilde{\mathbf{G}}_{\text{mi}}(z)$ with Δ_{mi} samples to obtain $\tilde{\tilde{\mathbf{G}}}_{\text{mi}}(z)$.
3. The delay Δ_{mi} should be incorporated in $\tilde{\mathbf{G}}_{\text{ai}}(z)$ but time-reversed, since it is based on the adjoint, and the Hermitian in the frequency domain, of the product $\tilde{\mathbf{G}}(e^{j\omega}) \tilde{\mathbf{G}}_{\text{mi}}(e^{j\omega})$.

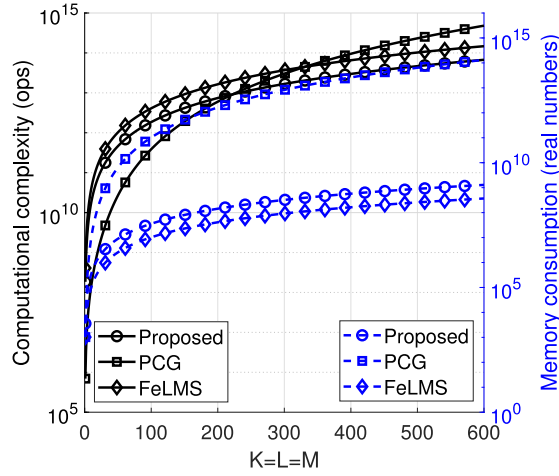


Fig. 4. Comparing the computational complexity and memory consumption of the proposed algorithm (circles) with the PCG method (squares) and the FeLMS algorithm (diamonds). Computational complexity is indicated by the solid lines (corresponding to the left axis), memory consumption is indicated by the dashed lines (corresponding to the right axis). It is assumed that the solution using the proposed algorithm can be obtained within $n_{\text{proposed}} = 4e4$ LMS iterations, using the PCG method within $n_{\text{PCG}} = 100$ iterations and using the FeLMS algorithm within $n_{\text{FeLMS}} = 4e5$ iterations. Furthermore it is assumed that $I = J = O = N_{\text{fit}} = 512$, with an assumed FFT cost of $n \log(n)$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1

Computational complexity of the proposed algorithm, compared to the PCG method with circulant preconditioning. The variable N_{fit} corresponds to the number of frequency bins in the frequency-response matrices and n indicates the number of iterations of the proposed algorithm. The cost of a Fourier Transform applied on a vector of length n is denoted as $FFT(n)$.

	Proposed algorithm	PCG with circulant preconditioning [25]	FeLMS [21]
Setup cost	$FFT(N_{\text{fit}})(K^2 + 3M^2 + 2LM) + FFT(n)K^2 + N_{\text{fit}}(K^2 + 3K^3 + 2M^2 + 3M^3 + LM + LM^2)$	$(MK)^2 I + (MK)^2 \cdot FFT(I) + MKI + I \cdot FFT(MK)$	None
Per-iteration cost	$OK^2 + 3MKI + JM^2 + 2J(L + M)M$	$MKI + 2I \cdot FFT(MK) + 2MK \cdot FFT(I)$	$(IK + JL)M$
Memory consumption	$OK^2 + MKI + JM^2 + 2J(L + M)M$	$(2I - 1)(MK)^2$	$MKI + JML$

4. Thereafter $\tilde{G}_{\text{ai}}(z)$ should be delayed by Δ_{ai} samples. Therefore, the filters can be written as:

$$\begin{aligned} \tilde{\tilde{G}}_{\text{mi}}(z) &= \Delta_{\text{mi}}(z)\tilde{G}_{\text{mi}}(z), \\ \tilde{\tilde{G}}_{\text{ai}}(z) &= \Delta_{\text{ai}}(z)\Delta_{\text{mi}}^-(z)\tilde{G}_{\text{ai}}(z), \end{aligned} \tag{22}$$

where Δ_{mi}^- represents a negative delay.

5. Truncate the non-causal parts of $\tilde{\tilde{G}}_{\text{mi}}(z)$ and $\tilde{\tilde{G}}_{\text{ai}}(z)$ to obtain the coefficients of FIR filters $\hat{G}_{\text{mi}}(z)$ and $\hat{G}_{\text{ai}}(z)$ of length J .

The computational complexity and memory consumption of the algorithm is shown in Table 1. A graph of these quantities with respect to the number of sources and sensors $K = L = M$ is shown in Fig. 4. The figure shows that from about $K = L = M = 200$, the proposed algorithm scales better than the PCG method in terms of computational complexity. The proposed algorithm scales better than the FeLMS algorithm for any number of K, L and M . Furthermore, it can be seen that the memory consumption of the proposed algorithm scales well with the problem size.

2.4. Computation of the reflected sound field using the Kirchhoff-Helmholtz integral

The Kirchhoff-Helmholtz integral is computed to obtain the reflected sound field. The integral can be derived by combining the acoustic wave equation and the Green's integral theorem. With reference to Fig. 5, considering a sound propagation in a volume V enclosed by a surface S , the solution to the inhomogeneous wave equation describes the full sound field as the sum of the direct sound field and the reflected sound field as [36]:

$$\begin{aligned} p(x, \omega) &= \int_V Q_{\text{vol}}(x_s, \omega)G(x|x_s, \omega)dV + \\ &\int_S (G(x|x_s, \omega)\nabla p(x_s, \omega) - p(x_s, \omega)\nabla G(x|x_s, \omega)) \cdot \mathbf{n}dS, \end{aligned} \tag{23}$$

in which x and x_s denote the position inside V and on S respectively, ω is the angular frequency, \mathbf{n} is the unit normal vector pointing into V , $G(x|x_s, \omega)$ and $\nabla G(x|x_s, \omega)$ are the Greens function and the gradient of the Greens function respectively, $p(x_s, \omega)$ and $\nabla p(x_s, \omega)$ are the acoustic pressure at S and the gradient of the acoustic pressure at S respectively. The direct sound field is

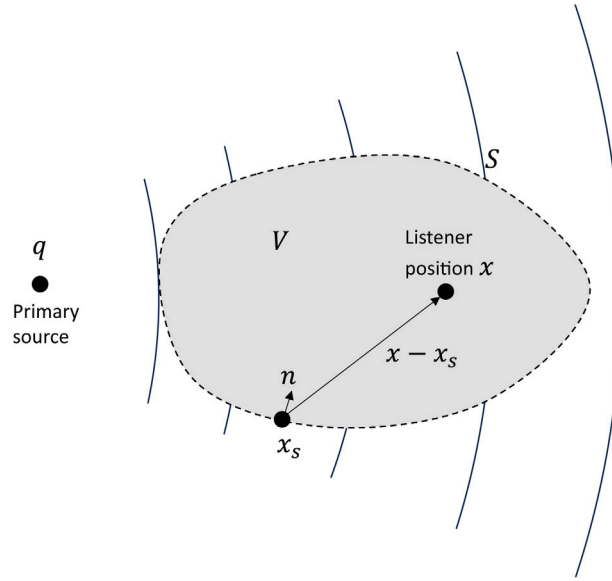


Fig. 5. Parameters used for the Kirchhoff-Helmholtz integral.

described by the first term containing the volume integral, and the reflected sound field is described by the second term containing the surface integral. By choosing the free-field Greens function and by setting $Q_{vol} = 0$ in Eq. (23), the integral will reduce to the Kirchhoff-Helmholtz integral, which will return the reflected sound field, even with the presence of a sound source within the volume V . This is shown in [36], by a derivation using reciprocity of the Greens function. Without the direct sound field, the Kirchhoff-Helmholtz integral computes the reflected sound field as [34]:

$$p(x, \omega) = \frac{1}{4\pi} \int_S \left(G(x|x_s, \omega) \frac{\partial p(x_s, \omega)}{\partial n} - p(x_s, \omega) \frac{\partial G(x|x_s, \omega)}{\partial n} \right) dS. \quad (24)$$

The partial derivative of the pressure to the normal direction at S can be written in terms of the particle normal velocity V_n at S following:

$$\frac{\partial p(x_s, \omega)}{\partial n} = j\rho c k V_n(x_s, \omega), \quad (25)$$

in which c is the speed of sound, $k = \omega/c$ is the wave number and ρ is the density of the acoustic medium.

In this case, the two-dimensional (2D) free-field Greens function is considered, defined as:

$$G(x|x_s, \omega) = -j\pi H_0^{(2)}(k|x - x_s|), \quad (26)$$

in which $H_a^{(b)}$ is the Hankel function of kind b and order a . Substitution of Eqs. (25) and (26) into Eq. (24) leads to the 2D forward Kirchhoff-Helmholtz integral:

$$p^{(2)}(x, \omega) = \frac{-jk}{4} \int_S \left(j\rho c V_n(x_s, \omega) H_0^{(2)}(k|x - x_s|) + p(x_s, \omega) H_1^{(2)}(k|x - x_s|) \cdot \mathbf{n} \right) dS, \quad (27)$$

Following the notation of Fig. 6, using a circular geometry with radius R , the equation is simplified and discretized [37]:

$$p^{(2)}(x, \omega) = \frac{-jk}{4} \sum_{i=1}^{N_L} \left(j\rho c V_n(x_s^i, \omega) H_0^{(2)}(k|x - x_s^i|) + p(x_s^i, \omega) H_1^{(2)}(k|x - x_s^i|) \cdot \mathbf{n} \right) R\Delta\theta, \quad (28)$$

in which $\Delta\theta$ is the angle between the sensors on the surface, assuming the sensors are equidistantly distributed.

The particle velocity in the radial outward direction can be obtained by a difference quotient between the sound pressure levels at $r_{s-h}^i = \|x_s^i\| - h$ and $r_{s+h}^i = \|x_s^i\| + h$ (in circular coordinates) following [34]:

$$V_n(x_s^i, \omega) = \frac{p(r_{s+h}^i, \omega) - p(r_{s-h}^i, \omega)}{2jh\omega\rho}, \quad (29)$$

in which h is the distance between the sensors. In this equation, the sign deviates from [34], which is due to the positive direction convention. The Kirchhoff-Helmholtz integral can be pre-computed such that it does not increase the real-time computational

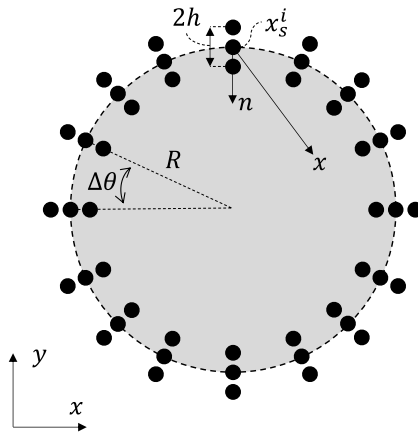


Fig. 6. Geometry used for the derivation of the Kirchhoff–Helmholtz integral in case of a circular microphone array.

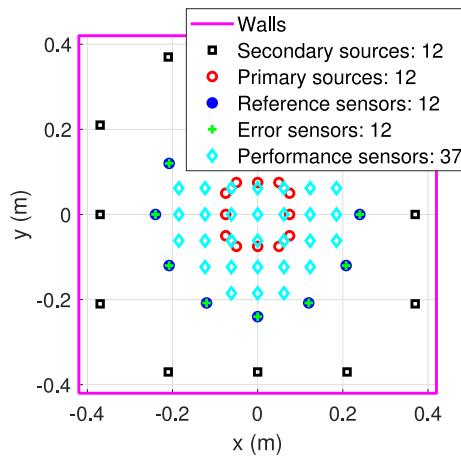


Fig. 7. 2-dimensional system configuration of the 12-channel active noise control setup to attenuate acoustic reflections from the walls.. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

complexity. The Kirchhoff–Helmholtz integral is computed at the augmented performance sensor locations. The signals of these performance sensors are the input of the control algorithm.

3. Simulation

A simulation is performed for a multichannel system in which $K = L = M = P = 12$, where all sources are independent. The number of sources and sensors is chosen equal, but do not need to be equal. The number of performance sensors is $L_{\text{perf}} = 37$, which are spatially distributed within the microphone surface with a distance of 0.06 m from each other, starting from the origin. The configuration of the system is shown in Fig. 7, with the objective to attenuate the reflected sound field. A summary of the simulation parameters is denoted in Table 2. The dimensions of the chamber and the number of sources and sensors in the simulation are based on a small-scale active anechoic chamber which is being prepared for experimental testing. However, if it is possible to place sensors directly on a sound radiating object in the room, this is favorable in terms of time-advantage and will probably result in a stronger correlation with the noise signal.

The speed of sound is $c = 343$ m/s and the density of the air is $\rho = 1.21$ kg/m³. The walls have the dimensions of 0.84 m square, modeled using an impedance of $Z = 10\rho c$. The primary and secondary sources are modeled by monopole sources. The distance between the secondary sources is 0.21 m and the primary sources is 0.075 m. The secondary sources are placed with a spacing of 0.05 m from the walls. The primary sources are placed at 0.12 m from the origin. The sensors are placed on a circle around the origin with a radius of $r = 0.24$ m. The reference sensors measure the pressure at the reference sensor locations. The performance sensors measure the reflected sound field, computed using the Kirchhoff–Helmholtz integral based on measurements of the error sensors.

The transfer functions between the source–sensor pairs are computed in Comsol 6.0 using at least 4 quadratic elements per wavelength. The reference sensor transfer function matrix $\mathbf{X}(e^{i\omega})$ is obtained by sampling in the frequency domain from 0 Hz to

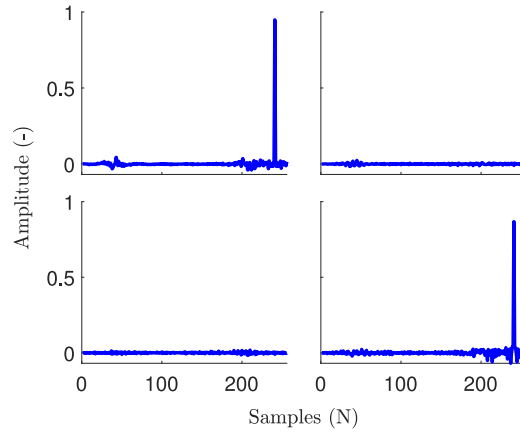


Fig. 8. First two channels of the decoupled plant impulse response matrix $\hat{\mathbf{G}}_{\text{ai}}(z)\hat{\mathbf{G}}(z)\hat{\mathbf{G}}_{\text{mi}}(z)$. The other channels are similar and thus omitted.

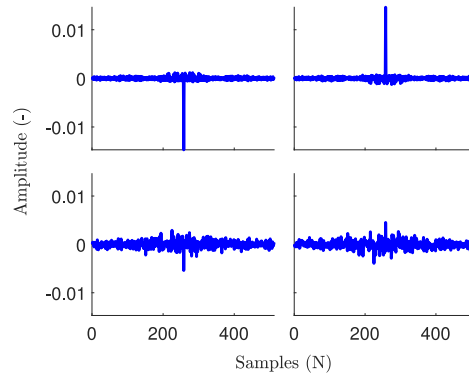


Fig. 9. Prewhitening filter $\hat{\mathbf{F}}_w(z)$ corresponding to the first two channels, referring to inputs 1,2 and outputs 1,2. The other channels are similar and thus omitted.

600 Hz, which is single-sided, using a frequency-resolution of 1 Hz. The acoustic pressure at the error sensors using $h = 0.01$ m, is obtained by sampling in the frequency domain from 0 Hz to 600 Hz, which is single-sided, using a frequency-resolution of 1 Hz, resulting in a sampling frequency of $f_s = 1200$ Hz. Following Eq. (29), the normal particle velocity is computed at the error sensors. Using the acoustic pressure and the normal particle velocity at the error sensors, the Kirchhoff–Helmholtz integral is computed, from which the primary- and secondary-path transfer function matrices $\mathbf{P}(e^{j\omega})$ and $\mathbf{G}(e^{j\omega})$ are computed. The matrix $\mathbf{P}(e^{j\omega})$ is the transfer function matrix from the primary sources to the performance sensors, and the matrix $\mathbf{G}(e^{j\omega})$ is the transfer function matrix from the secondary sources to the performance sensors. Both $\mathbf{P}(e^{j\omega})$ and $\mathbf{G}(e^{j\omega})$ are filtered with (1) a second-order low-pass filter having a cut-off frequency at 480 Hz and with (2) a second-order high-pass filter having a cut-off frequency at 120 Hz. These two filters are applied to minimize time domain artifacts that occur when transforming a pass-band with sharp cut-offs from the frequency- to the time domain. The primary path transfer function matrix is converted from the frequency- to the time domain using the Fourier transform resulting in the matrix of impulse responses $\mathbf{P}(z)$.

The secondary path is regularized such that $\sqrt{\beta_G}$ is at 20 dB below the largest frequency domain peak of $\mathbf{G}(e^{j\omega})$, and truncated after $J = 256$ samples such that $\tilde{\mathbf{G}}(z)$ is obtained. The preconditioning and decoupling filters are computed using $N_{fft} = 512$ Fourier Transform samples, resulting in $\tilde{\mathbf{G}}_{\text{mi}}(z)$ and $\tilde{\mathbf{G}}_{\text{ai}}(z)$. Adding a delay of $\Delta_{\text{mi}} = 200$ samples to $\tilde{\mathbf{G}}_{\text{mi}}(z)$ results in $\tilde{\tilde{\mathbf{G}}}_{\text{mi}}(z)$, which, after truncation to length J results in $\hat{\tilde{\tilde{\mathbf{G}}}}_{\text{mi}}(z)$. Similarly, $\tilde{\mathbf{G}}_{\text{ai}}(z)$ is delayed by $\Delta_{\text{mi}}^- = -200$ samples, followed by a delay of $\Delta_{\text{ai}} = 240$ samples, resulting in $\tilde{\tilde{\mathbf{G}}}_{\text{ai}}(z)$. Truncation of $\tilde{\tilde{\mathbf{G}}}_{\text{ai}}(z)$ to length J results in $\hat{\tilde{\tilde{\mathbf{G}}}}_{\text{ai}}(z)$. The decoupling of the plant is shown in Fig. 8, from which can be seen that the plant is effectively decoupled, having a diagonal response with a pure delay that corresponds to Δ_{ai} delay samples. The delays Δ_{mi} and Δ_{ai} correspond to the length of the secondary path impulse response J . In general, choosing $\Delta_{\text{mi}} \approx 0.8J$ and $\Delta_{\text{ai}} \approx 0.95J$ seems to be a good starting point, but this is dependent on the application. If those numbers are too far off from what they should be, the decoupling of the system is reduced.

The prewhitening and decorrelation filter is shown in Fig. 9. The filter is computed from the reference signals of length $n = 20000$ using Welch’s method with a Hamming window of length 80 samples having 50% overlap. A regularization is applied of $\sqrt{\beta_F}$ which is at 60dB below the largest eigenvalue of $\mathbf{D}(e^{j\omega})$ in the frequency domain, resulting in $\tilde{\mathbf{F}}_w(z)$. Because the PSD matrix is positive semi-definite symmetric, a delay of $\Delta_F = \frac{512}{2} + 1 = 257$ samples is added, resulting in $\tilde{\tilde{\mathbf{F}}}_w(z)$, followed by truncation to a length of 512 samples, resulting in $\hat{\tilde{\tilde{\mathbf{F}}}}_w(z)$.

Table 2
Summary of the simulation parameters.

Parameter	Quantity	Unit	Description
$K = L = M = P$	12	-	Number of sources and sensors
L_{perf}	37	-	Number of performance sensors
c	343	m/s	Speed of sound
ρ	1.21	kg/m ³	Density of air
Room size	0.84×0.84	m	Length x width
Z	$10\rho c$	pa s/m	Impedance at the walls
d_{ss}	0.21	m	distance between secondary sources
d_{ps}	0.075	m	distance between primary sources
$d_{\text{ss-wall}}$	0.05	m	distance between secondary sources and the wall
r_{sens}	0.24	m	Radius of the circle on the origin at which the sensors are placed
f_{start}	0	Hz	Start of range for frequency domain sampling
f_{step}	1	Hz	Frequency resolution
f_{end}	600	Hz	End of range for frequency domain sampling
f_s	1200	Hz	Sampling rate
$\sqrt{\beta_G}$	20	dB	Regularization factor for secondary path
J	256	samples	Length of plant model
N_{fft}	512	samples	Number of samples used in the Fourier transform
Δ_{mi}	200	samples	Delay in the $\hat{G}_{\text{mi}}(z)$ filter
Δ_{mi}^-	-200	samples	Delay in the $\hat{G}_{\text{ai}}(z)$ filter
Δ_{ai}	240	samples	Delay in the $\hat{G}_{\text{ai}}(z)$ filter
n_{iter}	$2e4$	samples	Number of iterations in the algorithm
$\sqrt{\beta_F}$	60	dB	Regularization factor of the prewhitening filter
Δ_F	257	samples	Delay in the $\hat{F}_w(z)$ filter
Δ_{art}	50	samples	Delay of the noise signal
I	64	samples	Control filter length
O	512	samples	Length of the prewhitening filter
α_0	4	-	Step size of the algorithm
n_{pwf}	$2e4$	-	Length of the reference signals to compute $\hat{F}_w(z)$
n_{window}	80	-	Length of the Hamming window to compute $\hat{F}_w(z)$
n_{overlap}	50%	-	Overlap between windows to compute $\hat{F}_w(z)$
f_{clip}	480	Hz	Cutoff frequency of the low-pass filter applied on $\mathbf{P}(e^{j\omega})$ and $\mathbf{G}(e^{j\omega})$.
f_{chp}	120	Hz	Cutoff frequency of the high-pass filter applied on $\mathbf{P}(e^{j\omega})$ and $\mathbf{G}(e^{j\omega})$.
h	0.01	m	Distance between microphones in radial direction.

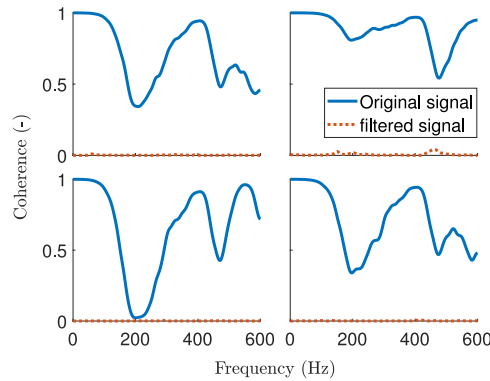


Fig. 10. Coherence between a selected subset of reference signals. Blue shows the original reference signals. Red shows the reference signals after filtering with $\hat{F}_w(z)$. Left upper figure: reference sensors 1 and 3, right upper figure: reference sensors 2 and 3, left lower figure: reference sensors 1 and 4, right lower figure: reference sensors 2 and 4. The other channels show similar results, and are therefore omitted.. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The coherence between the reference signals is shown in Fig. 10, showing that the signals are properly decorrelated.

The noise signal is a Gaussian white noise signal convolved with the primary path, which is filtered with an anti-aliasing filter having a cutoff-frequency at $f_{\text{Nyq}} = 600$ Hz. Upon running the proposed algorithm, the noise signal is delayed by $\Delta_d = \Delta_F + \Delta_{\text{mi}} + \Delta_{\text{art}}$ samples, with $\Delta_{\text{art}} = 50$ artificial delay samples. The performance of the proposed algorithm is shown in Fig. 11. The system is simulated using $n = 2e4$ time samples, with a desired filter length of $I = 64$ samples, requiring 12.2 s on a laptop with an Intel Core i7-10750H @ 2.60 GHz CPU and 16 GB of RAM. The algorithm converged to a reduction of 12.8 dB, averaged over all performance sensors. A step-size of $\alpha_0 = 4$ is used, which is the maximum allowable value for which the algorithm converges. Using the same settings, the normalized FeLMS algorithm converged to 3.3 dB averaged over all sensors. The normalized FeLMS algorithm required over $2e5$ iterations to obtain the same performance of 12.8 dB reduction averaged over all performance sensors, as compared to

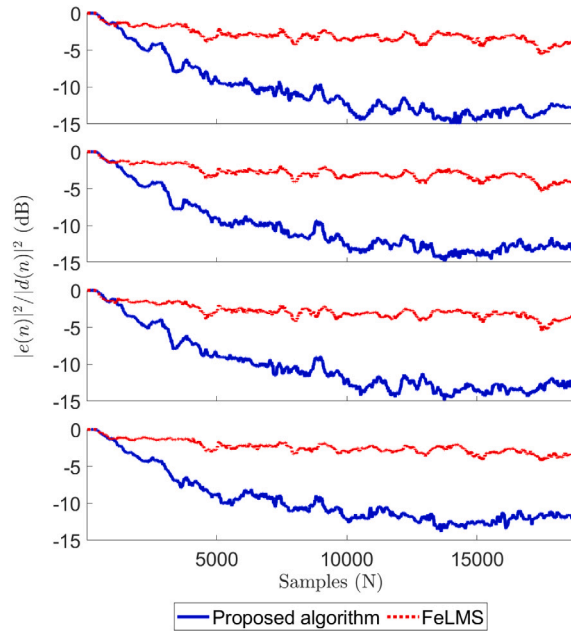


Fig. 11. Convergence of the first four channels of the proposed algorithm and the normalized FeLMS algorithm. The average reduction of the proposed algorithm is 12.8 dB, while the average reduction of the normalized FeLMS algorithm is 3.3 dB; the signals are averaged using a window of 500 samples. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

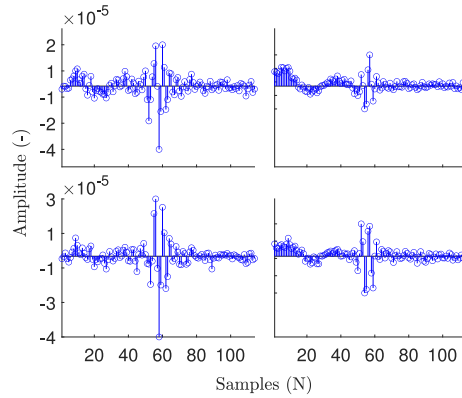


Fig. 12. Delayed controller filter coefficients $W_{\Delta}(z)$ of length $\bar{I} = \Delta_{art} + I$ which correspond to reference sensors 1,2 and secondary sources 1,2. The other channels show a similar delay, although the response could be different. The other channels are omitted for the sake of readability.

the proposed algorithm. This shows that the preconditioning filters reduce the number of iterations by about a factor 10 for this simulation with these parameters. Exporting the particle velocity directly from Comsol 6.0, without approximation using Eq. (29), allows for higher reduction levels. This shows that the approximation of Eq. (29) introduces a small error, but allows the use of microphones.

After running the iterative algorithm, the last updated set of filter coefficients $C(z)$ is converted to $W_{\Delta}(z)$ using Eq. (6), where subscript Δ is used to indicate that this controller includes a delay. The delays Δ_{mi} and Δ_F are truncated from the filter coefficients and the first two channels are shown in Fig. 12, still having a delay corresponding to Δ_{art} .

For the system to be applicable in the real world, Δ_{art} is truncated from the set of filter coefficients $W_{\Delta}(z)$, resulting in the desired set of filter coefficients $W(z)$ with length I , of which the first two channels are shown in Fig. 13.

Both sets of control filter coefficients $W_{\Delta}(z)$ and $W(z)$ are simulated, of which the frequency domain spectra are shown in Fig. 14. Running the controller with the set of filter coefficients $W_{\Delta}(z)$ with a noise signal that is delayed by Δ_{art} samples, shows an average performance of 15.4 dB. Running the controller with the set of filter coefficients $W(z)$ with a noise signal without any delay, shows an average performance of 13.8 dB. Removal of the delay from the filter coefficients, shows an average loss in reduction of 1.6 dB averaged at the performance sensors.

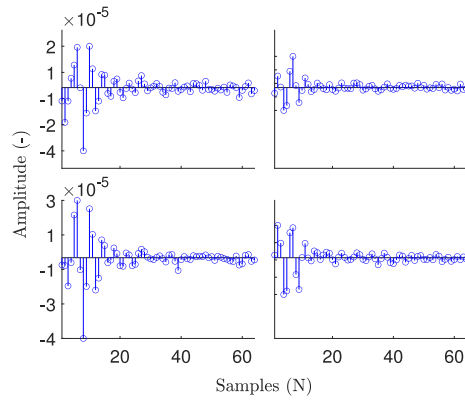


Fig. 13. Controller filter coefficients $W(z)$ truncated to length l which correspond to reference sensors 1,2 and secondary sources 1,2. The other channels are omitted for the sake of readability.

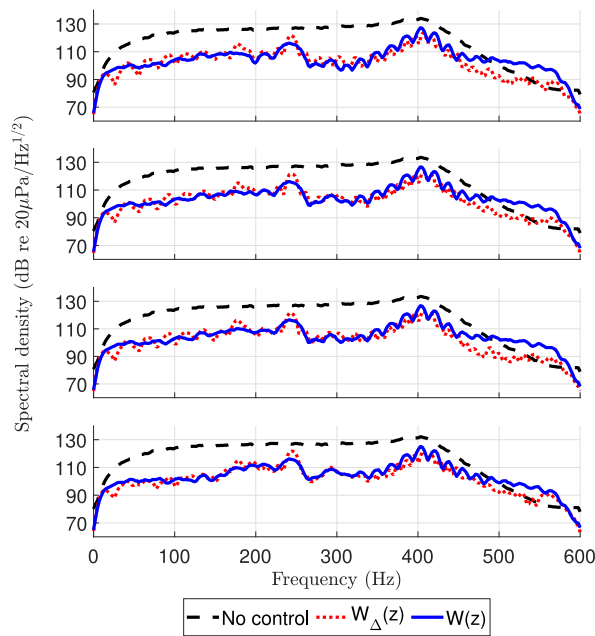


Fig. 14. Spectral densities of the first four performance sensors. These results are obtained after running the configuration as shown in Fig. 1 using the delayed controller $W_{\Delta}(z)$ with the noise signal that is delayed by 4_{arr} samples, compared to the controller $W(z)$ with a noise signal that is not delayed. The average reduction over all channels is 15.4 dB, using the filter coefficients $W_{\Delta}(z)$. The average reduction over all channels with the filter coefficients $W(z)$ is 13.8 dB; the signals are averaged using a window of 500 samples. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

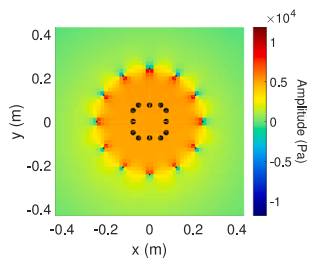


Fig. 15. Kirchhoff-Helmholtz integral at 56.2 Hz, with the primary sources active, which are located at the black dots. The result is only valid inside the contour. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

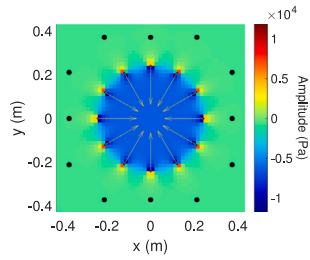


Fig. 16. As Fig. 15, but with the secondary sources active. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

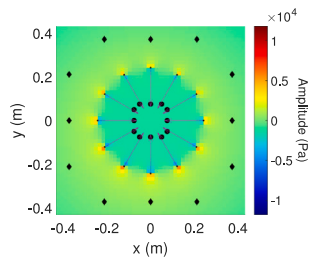


Fig. 17. Sum of Figs. 15 and 16. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

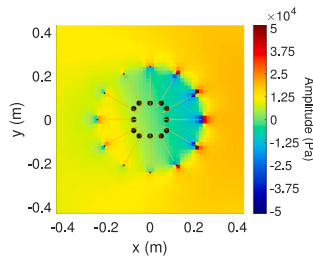


Fig. 18. As Fig. 15, but at 161.7 Hz. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

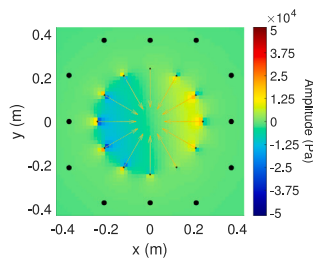


Fig. 19. As Fig. 15, but with the secondary sources active, at 161.7 Hz. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

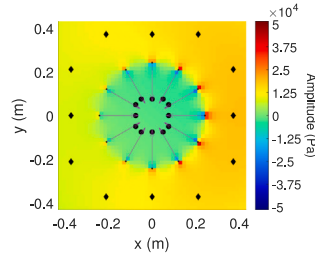


Fig. 20. Sum of Figs. 18 and 19. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

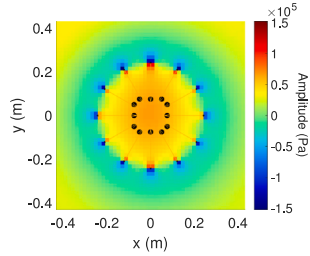


Fig. 21. As Fig. 15, but at 396.1 Hz. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

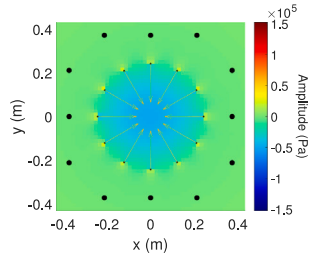


Fig. 22. As Fig. 15, but with the secondary sources active, at 396.1 Hz. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The Kirchhoff–Helmholtz integral is computed, and the results are shown at $f = 56.2$ Hz (Fig. 15 till Fig. 17), $f = 161.7$ Hz (Fig. 18 till Fig. 20) and $f = 396.1$ Hz (Fig. 21 till Fig. 23). It can be seen that at these frequencies, the sound field due to the secondary sources approaches the inverse of the sound field due to the primary sources, such that the sum of both fields is minimized by the controller. These figures also show that global control is achieved within the microphone area, which can be seen from the acoustic pressure in the field surrounded by the microphones. It should be noted that the results of the Kirchhoff–Helmholtz integral are not valid outside of the contour, but the full domain is shown for completeness.

4. Conclusion

A scheme is presented for the efficient computation of a fixed time domain controller. The scheme employs preconditioning filters to enhance convergence speed. The filters are computed in the frequency domain to reduce computational complexity, but may become non-causal after conversion to the time domain, necessitating the application of delay and truncation procedures. Despite the introduction of delays via the preconditioning filters in the direct path, it is possible to truncate the delay from the control filter coefficients. This allows for the computation of a fixed time domain control filter that is applicable on systems without any delay in the primary noise signal. To maintain convergence, regularization is incorporated which allows for larger step-sizes and improves reduction levels using a minimal number of iterations. The presented scheme is applied to the control of a square room in which the reflected sound field is attenuated. The reflected sound field is pre-computed with the Kirchhoff–Helmholtz integral using measurements at the error sensors. It is shown that using the performance sensors, global control of the reflected sound field within the microphone area is achieved. Furthermore, it is shown that truncation of the delays from the control filter coefficients enables implementation on the same system without delays. Although the truncation of the delay from the filter coefficients results in a loss of 1.6 dB in performance, a reduction of the reflected sound field of 13.8 dB averaged over all performance sensors is achieved.

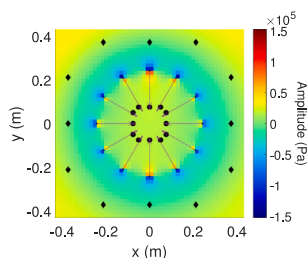


Fig. 23. (Color online) Sum of Figs. 21 and 22.

CRedit authorship contribution statement

R. Haasjes: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Software, Visualization, Writing – original draft, Writing – review & editing. **A.P. Berkhoff:** Funding acquisition, Methodology, Project administration, Software, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgments

This research is funded by TNO. The support and funding from TNO are gratefully acknowledged. The authors would like to thank the reviewers for taking the time and effort to review the manuscript. The comments and suggestions are sincerely appreciated and helped to improve the quality of the manuscript.

References

- [1] V. Kopicov, V. Palchikovskiy, I. Belyaev, Y.V. Bersenev, S.Y. Makashov, I. Khramtsov, I. Korin, E. Sorokin, O.Y. Kustov, Construction of an anechoic chamber for aeroacoustic experiments and examination of its acoustic parameters, *Acoust. Phys.* 63 (1) (2017) 113–124.
- [2] B.D. Patchett, B.E. Anderson, Nonlinear characteristics of high amplitude focusing using time reversal in a reverberation chamber, *J. Acoust. Soc. Am.* 151 (6) (2022) 3603–3614.
- [3] K. Sozanski, A. Sozanska, Low frequency loudspeaker measurements using an anechoic acoustic chamber, in: 2018 Signal Processing: Algorithms, Architectures, Arrangements, and Applications, SPA, Poznan, Poland, 2018, pp. 367–372.
- [4] S. Schneider, C. Kern, Acoustical behavior of the large anechoic chamber at the laboratoire de mécanique et d'acoustique in the low frequency range, *Acta Acust. United Acust.* 94 (2008) 141–147.
- [5] I. Belyaev, A.Y. Golubev, A.Y. Zverev, S.Y. Makashov, V. Palchikovskiy, A. Sobolev, V. Chernykh, Experimental investigation of sound absorption of acoustic wedges for anechoic chambers, *Acoust. Phys.* 61 (5) (2015) 606–614.
- [6] B. Cuyx, W. Desmet, W. Buyens, T. van Waterschoot, Design and validation of a low-cost acoustic anechoic chamber, *J. Audio Eng. Soc.* 149 (2020) 1–5.
- [7] M.S. Ressler, P.E. Wundes, Design of an acoustic anechoic chamber for application in hearing aid research, in: Proceedings of the 11th WSEAS International Conference on Acoustics & Music: Theory & Applications, World Scientific and Engineering Academy and Society (WSEAS), Iasi, Romania, 2010, pp. 18–23.
- [8] X. Duanqi, W. Zheng, C. Jinjing, Acoustic design of an anechoic chamber, *Appl. Acoust.* 29 (2) (1990) 139–149.
- [9] L. Lu, K. Yin, R. de Lamare, Z. Zheng, Y. Yu, X. Yang, B. Chen, A survey on active noise control in the past decade—Part I: Linear systems, *Signal Process.* 183 (2021) 108039.
- [10] M. Heertjes, D. Bruijnjen, MIMO FIR feedforward design for zero error tracking control, in: 2014 American Control Conference, Portland, OR, USA, 2014, pp. 2166–2171.
- [11] L. Mirkin, G. Tadmor, Imposing FIR structure on H2 preview tracking and smoothing solutions, in: Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, USA, 2006, pp. 1417–1422.
- [12] C.E. García, D.M. Pretti, M. Morari, Model predictive control: Theory and practice—A survey, *Automatica* 25 (3) (1989) 335–348.
- [13] B. Widrow, E. Walach, Adaptive Inverse Control: A Signal Processing Approach, second ed., Wiley-IEEE Press, New Jersey, USA, 2014.
- [14] T.S. Becker, D.-J. Van Manen, C.M. Donahue, C. Bärlocher, N. Börsing, F. Brogini, T. Haag, J.O.A. Robertsson, D.R. Schmidt, S.A. Greenhalgh, T.E. Blum, Immersive wave propagation experimentation: Physical implementation and one-dimensional acoustic results, *Phys. Rev. X* 8 (3) (2018).
- [15] N. Börsing, T.S. Becker, A. Curtis, D.-J. van Manen, T. Haag, J.O. Robertsson, Cloaking and holography experiments using immersive boundary conditions, *Phys. Rev. A* 12 (2019) 024011.
- [16] H.R. Thomsen, M. Molerón, T. Haag, D.-J. van Manen, J.O.A. Robertsson, Elastic immersive wave experimentation: Theory and physical implementation, *Phys. Rev. Res.* 1 (2019) 033203.
- [17] X. Li, J. Robertsson, D.-J. van Manen, Elastic immersive wave experimentation, *Geophys. J. Int.* 233 (1) (2022) 724–739.
- [18] D. Habault, E. Friot, P. Herzog, C. Pinhede, Active control in an anechoic room: Theory and first simulations, *Acta Acust. United Acust.* 103 (2017) 369–378.

- [19] E. Friot, R. Guillermin, M. Winninger, Active control of scattered acoustic radiation: a real-time implementation for a three-dimensional object, *Acta Acust. United Acust.* 92 (2006) 278–288.
- [20] E. Friot, C. Bordier, Real-time active suppression of scattered acoustic radiation, *J. Sound Vib.* 278 (3) (2004) 563–580.
- [21] S. Elliott, *Signal Processing for Active Control*, in: *Signal Processing and its Applications*, Academic Press, California, USA, 2001.
- [22] N. Levinson, The Wiener (root mean square) error criterion in filter design and prediction, *J. Math. Phys.* 25 (1–4) (1946) 261–278.
- [23] S. Chandrasekaran, N. Govindarajan, A. Rajagopal, Fast algorithms for displacement and low-rank structured matrices, in: *Proceedings of the 2018 ACM International Symposium on Symbolic and Algebraic Computation*, ACM, New York, USA, 2018, pp. 17–22.
- [24] Z. Sheng, P. Dewilde, S. Chandrasekaran, Algorithms to solve hierarchically semi-separable systems, in: *System Theory, the Schur Algorithm and Multidimensional Analysis*, Birkhäuser Basel, Basel, Switzerland, 2007, pp. 255–294.
- [25] T. Chan, J. Olkin, Circulant preconditioners for toeplitz-block matrices, *Numerical Algorithms* 6 (1992) 89–101.
- [26] M. Cai, X. Jin, Y. Wei, A generalization of t. Chan's preconditioner, *Linear Algebra Appl.* 407 (2005) 11–18.
- [27] X. Lv, T. Huang, Z. Ren, A modified t. Chan's preconditioner for toeplitz systems, *Comput. Math. Appl.* 58 (4) (2009) 693–699.
- [28] S. Elliott, Optimal controllers and adaptive controllers for multichannel feedforward control of stochastic disturbances, *IEEE Trans. Signal Process.* 48 (4) (2000) 1053–1060.
- [29] M. Bai, S. Elliott, Preconditioning multichannel adaptive filtering algorithms using EVD- and SVD-based signal prewhitening and system decoupling, *J. Sound Vib.* 270 (2004) 639–655.
- [30] H. Lollmann, P. Vary, Generalized filter-bank equalizer for noise reduction with reduced signal delay, in: *Interspeech 2005, ISCA, Lisbon, Portugal, 2005*, pp. 2105–2108.
- [31] A.P. Berkhoff, G. Nijssse, A rapidly converging filtered-error algorithm for multichannel active noise control, *J. Adapt. Control Signal Process.* 21 (2007) 556–569.
- [32] J. Ahrens, R. Rabenstein, S. Spors, The theory of wave field synthesis revisited, in: *Audio Engineering Society Convention 124*, Audio Engineering Society, Amsterdam, The Netherlands, 2008, pp. 1–19.
- [33] S. Spors, H. Teutsch, A. Kuntz, R. Rabenstein, Sound field synthesis, in: *Audio Signal Processing for Next-Generation Multimedia Communication Systems*, Springer US, Boston, MA, 2004, pp. 323–344.
- [34] P. Peretti, S. Cecchi, L. Romoli, F. Piazza, Performance evaluation of adaptive algorithms for wave field analysis/synthesis using sound field simulations, in: *Computational Simulations and Applications*, IntechOpen, Rijeka, Croatia, 2011, pp. 543–560.
- [35] B. Noble, J. Daniel, *Applied Linear Algebra*, in: *Featured Titles for Linear Algebra*, Prentice-Hall, New Jersey, USA, 1988.
- [36] P. Nelson, S. Elliott, *Active Control of Sound*, Academic Press, London, 1992.
- [37] E. Hulsebos, D. de Vries, E. Bourdillat, Improved microphone array configurations for auralization of sound fields by wave-field synthesis, *J. Audio Eng. Soc* 50 (10) (2002) 779–790.