OCEAN: AN OPEN LOW-COST EMBEDDED PLATFORM FOR ACTIVE NOISE CONTROL RESEARCH AND EDUCATION

Alessandro Fortino[1], Lukas Lübbert[1]

[1]*Department of Mechanical Engineering, University of Applied Sciences and Arts Dortmund, Germany*

Abstract

Active noise control (ANC) is well established in research and in series products, yet hands-on experimentation remains rare in engineering education and difficult to reproduce across laboratories. Conventional ANC test benches rely on rapid control prototyping platforms and measurement front-ends whose cost is of the order of several thousand euros, so that most students encounter adaptive noise control only in simulation. This contribution presents OCEAN (Open Control Environment for Active Noise Control), an open-source hardware and software platform that lowers the entry barrier to a component cost below 150 €. The system combines a modular 3D-printed duct of 90 mm inner diameter, two loudspeakers acting as primary and secondary source, two digital MEMS microphones connected over I^{2}S, and an STM32F767 development board carrying a class-D amplifier expansion. A model-based workflow generates the real-time code of a filtered-x LMS controller with feedback-path neutralisation directly from Simulink. The paper describes the mechanical and electrical design, the sampling rate and latency concept of the embedded signal chain, and the identification procedure for the secondary and feedback paths. Simulation results obtained with the published reference models quantify the influence of step size, control filter length and secondary path mismatch.

Keywords: *active noise control, FxLMS, embedded systems, open-source hardware, engineering education*

1. Introduction

Active noise control exploits destructive interference between an unwanted primary sound field and a secondary field radiated by a controlled actuator. The one-dimensional duct is the canonical configuration of the field: below the cut-on frequency of the first higher-order mode only plane waves propagate, the primary and secondary paths are well conditioned, and the filtered-x least mean squares (FxLMS) algorithm converges reliably [1]–[4]. Duct ANC is therefore the textbook example in courses on active acoustics and the basis of applications such as ventilation silencers and exhaust systems [5], [6], and it remains an active research field [7].

Practical experimentation with ANC nevertheless remains the exception in teaching. A conventional bench combines a rapid control prototyping system, measurement microphones with conditioning amplifiers, a power amplifier and a signal analyser, and easily reaches a five-figure investment. A department therefore owns a single bench, so that students observe a demonstration instead of building and debugging a controller themselves. Published experiments are also hard to repeat elsewhere, because neither the geometry nor the transducers nor the controller implementation are available to others.

Meanwhile the hardware has become inexpensive. Arm Cortex-M7 microcontrollers offer a single-precision FPU and DSP instructions above 200 MHz, digital MEMS microphones deliver an I^{2}S bitstream without analogue conditioning, and digital class-D amplifiers accept the same serial interface. Additive manufacturing removes the workshop dependency: the duct becomes a file. What has been missing is an integrated, documented and openly licensed combination of these elements, although open-source hardware has proven its value for research equipment in general [8].

This contribution presents OCEAN, an Open Control Environment for Active Noise Control: (i) a complete duct ANC demonstrator with a component cost of 92 € and a total system cost below 150 €; (ii) a model-based FxLMS controller with feedback-path neutralisation, generated from Simulink and executed on an STM32F767 target; (iii) a pre-training procedure that identifies the secondary and feedback paths with the

Corresponding author: *alessandro.fortino@fh-dortmund.de*

on-board transducers, so that no external measurement equipment is required; and (iv) the release of all design files, models, scripts and course material under open licences.

2. System design

2.1 Requirements

Five requirements guided the design. The component cost should stay below 150 € so that one setup per student group is affordable. The mechanical parts must be printable on a FDM machine and assembled with hand tools. The basic experiments must not require external measurement equipment, which means the control microphones also serve as the measurement chain. The complete signal path must remain visible and modifiable rather than hidden in a black box. Finally, the target hardware should be representative of embedded platforms used in series applications.

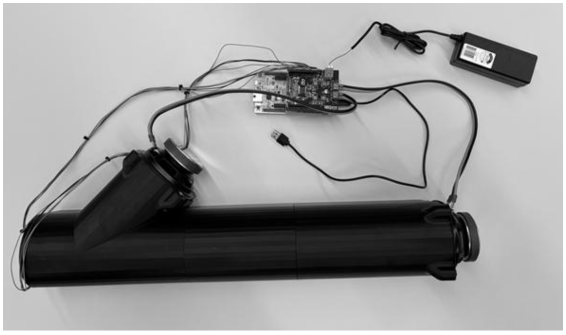

Figure 1. Assembled OCEAN demonstrator: 3D-printed duct with primary source (right), inclined branch carrying the secondary source (left), and the Nucleo board with the amplifier expansion.

2.2 Acoustic duct

The duct is assembled from three printed segments: a primary source mount, an extension of 105 mm, and a branch section of 227 mm carrying the secondary loudspeaker on an inclined stub. All segments share an inner diameter of D = 90 mm and a wall thickness of about 7 mm, and are joined by flanges with a printed microphone insert clamped between them. The configuration in Fig. 1 is about 0.54 m long; further extensions change the primary path delay, one of the parameters students are asked to vary.

The inner diameter fixes the usable bandwidth. In a circular duct the first higher-order mode cuts on at $f_1 = 1.8412\ c\ /\ (\pi\ D)$, which for D = 90 mm and c = 343 m/s gives 2.23 kHz. Below this frequency the field is a superposition of plane waves, so the one-dimensional assumptions behind the classical FxLMS derivation hold and the lecture material matches the observed behaviour. Above it, the modal field makes single-channel control ineffective, which is itself instructive.

The secondary source sits on an inclined branch rather than coaxially. This directs its radiation downstream and attenuates, without eliminating, the acoustic feedback to the reference microphone; the residual feedback is handled in the controller. Both microphones sit on the duct axis on a cross-shaped insert (Fig. 2a) that obstructs only a small fraction of the cross section.

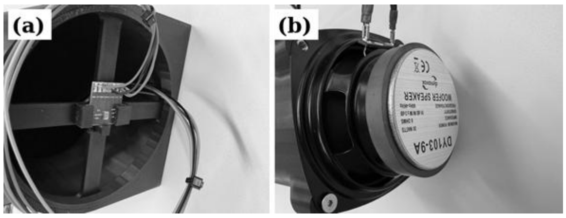

Figure 2. Construction details: (a) digital MEMS microphone on the printed cross-shaped insert clamped between two duct segments; (b) loudspeaker on the printed mount.

2.3 Electronics

The controller runs on a Nucleo-F767ZI board (STM32F767ZI, Cortex-M7 at 216 MHz with single-precision FPU, 2 MB flash, 512 kB RAM). An X-NUCLEO-CCA01M1 expansion carries a digital class-D amplifier that receives audio over I²S and is configured over I²C at start-up. Two digital MEMS microphones on evaluation boards share one I²S bus as the left and right channel of a stereo stream, so reference and error signal are sampled synchronously by construction. Table 1 lists the components.

Table 1. Bill of materials of the OCEAN demonstrator.

Component	Qty	Price
Nucleo-F767ZI controller board	1	35 €
X-NUCLEO-CCA01M1 amplifier	1	39 €
Digital MEMS microphone, I²S	2	4 €
Dynavox DY103-9A loudspeaker, 4″	2	5 €
Total		**92 €**

The printed parts need roughly 400 g of filament; with a 12 V supply and wiring the total stays below 150 €. In a conventional bench the microphones alone exceed this figure by an order of magnitude.

2.4 Signal chain and sampling rates

The serial audio interface is configured for a nominal 48 kHz. Because the audio PLL cannot realise this rate exactly with the given clock tree, the actual rate is 46.875 kHz, a deviation of −2.34 %. This is a property of the platform students should be aware of; the rates quoted below are nominal.

The input is read in blocks of 384 samples at 24 bit and converted to single precision. A polyphase FIR decimator reduces the rate by D = 24, so the control loop runs at about 2 kHz with a block size of 16 samples; the anti-noise signal is interpolated by the same factor and written at 16 bit. Three arguments motivate this two-rate architecture. The control rate gives a Nyquist frequency of about 977 Hz, safely below the duct cut-on at 2.23 kHz, and therefore covers the physically controllable band. An 80-tap filter at 2 kHz spans 41 ms of impulse response, enough for the delays and reflections in a duct of 0.5 m, whereas the same coverage at 48 kHz would need about 1900 taps. And

the computational load drops by more than an order of magnitude.
The price is the group delay of the rate conversion filters, which acts in series with the acoustic secondary path. It is not compensated explicitly but is simply part of the path identified in pre-training. The practical consequence, emphasised in the exercise instructions, is that the same rate conversion settings must be used during pre-training and during control operation. Table 2 summarises the implementation parameters.

Table 2. Parameters of the embedded implementation.

Parameter	Value
Target	STM32F767ZI, 216 MHz
Audio interface	I²S, 24 bit in / 16 bit out
Sampling rate	48 kHz nom. / 46.875 kHz
Input block size	384 samples
Decimation factor	24
Control rate	≈ 2 kHz, block 16
Control filter	80 taps, normalised LMS
Step size, leakage	0.002, 1.0
Soft start	0.1 s output, 0.2 s adapt.

3. Control structure and implementation

3.1 FxLMS with feedback-path neutralisation

Let u(n) be the signal at the reference microphone, y(n) the anti-noise signal and e(n) the residual at the error microphone. The primary path P(z) runs from the primary source to the error microphone, the secondary path S(z) covers the complete electro-acoustic chain from the controller output back to the error microphone, and the feedback path F(z) describes the unwanted propagation from the secondary source to the reference microphone. Since the reference microphone picks up the anti-noise signal, the measured reference is corrected with an estimate $\hat{F}(z)$ of the feedback path [11]:

$$x(n) = u(n) - \hat{f}^{\,T}(n)\, y(n) \tag{1}$$

The control filter W(z) of length L produces the anti-noise signal from the neutralised reference, and the residual follows from the superposition of the primary disturbance d(n) and the filtered output:

$$y(n) = w^{T}(n)\, x(n), \quad e(n) = d(n) - s^{T}\, y(n) \tag{2}$$

Because the output passes through S(z) before reaching the error microphone, gradient descent on $e^2(n)$ requires the reference to be filtered by an estimate $\hat{S}(z)$, which is the defining step of FxLMS [1], [9], [10]:

$$x'(n) = \hat{s}^{T}\, x(n) \tag{3}$$

The update is a leaky normalised LMS recursion, which bounds the step size independently of the signal level and prevents coefficient drift in the absence of excitation:

$$w(n+1) = \lambda\, w(n) + \mu\, e(n)\, x'(n) \,/\, (\varepsilon + ||x'(n)||^{2}) \tag{4}$$

Fig. 3 shows this structure superimposed on the geometry of the demonstrator, making the correspondence between physical paths and algorithm blocks explicit. In the course this single figure resolves most of the confusion about which transfer function is measured, which is modelled and which is adapted.

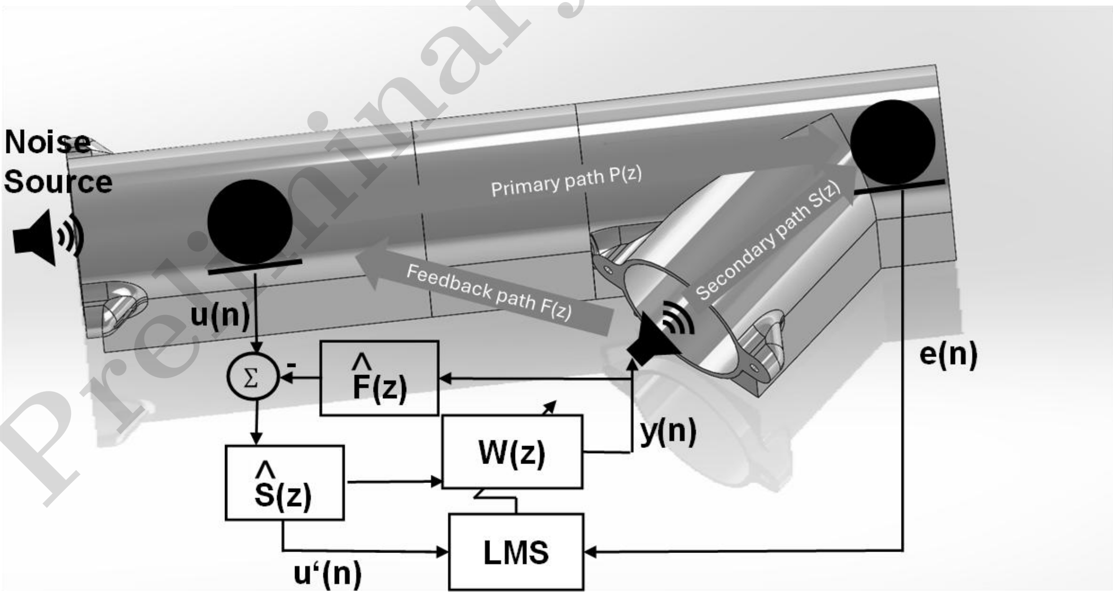

Figure 3. Control structure of the OCEAN demonstrator. The primary, secondary and feedback paths are shown on the duct geometry; the adaptive control filter W(z), the LMS update and the fixed models $\hat{S}(z)$ and $\hat{F}(z)$ obtained from pre-training form the controller.

3.2 Identification of the secondary and feedback paths

$\hat{S}(z)$ and $\hat{F}(z)$ are determined in a separate pre-training run. White noise is emitted through the secondary loudspeaker while two normalised LMS filters adapt in parallel, one driven by the error microphone and one by the reference microphone [12]. After convergence their weights are a discrete approximation of the secondary and the feedback path, and they are loaded as fixed FIR coefficients into the control model.

These estimates contain not only the acoustic propagation but the entire chain — amplifier, loudspeaker, duct, microphone, serial interface and the group delay of the rate converters. This is why an ANC controller cannot be designed from an acoustic model alone, and why identification must be repeated whenever the geometry or the signal chain changes.

3.3 Embedded implementation

The controller is implemented in Simulink with the STM32 hardware support blocks (Fig. 4). Fixed-point samples from the I²S interface are converted to single precision and scaled, and the anti-noise signal is scaled back to 16-bit integers before transmission. A delay of 16 samples in the neutralisation branch accounts for the block processing of the rate converters. Two on-delays implement a soft start: the output is enabled 0.1 s and the adaptation 0.2 s after the model starts, avoiding the loud transient that otherwise occurs while the filters are still empty. Code is generated and flashed directly from the model, and external mode allows the error signal and the filter weights to be observed on the host while the controller runs on the target.

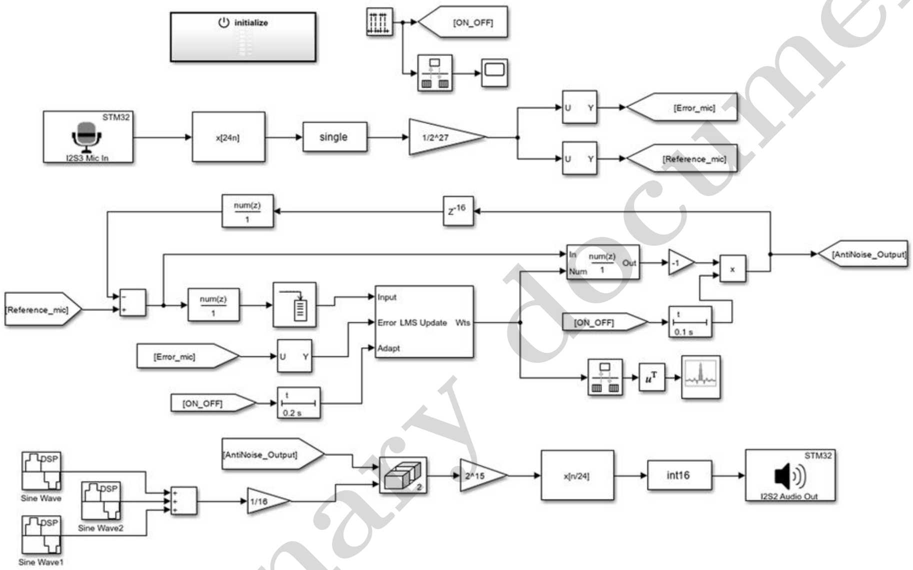

Figure 4. Simulink model of the embedded FxLMS controller with feedback-path neutralisation. The upper part reads the two microphone channels from the I²S interface, the centre implements the adaptive filter and the LMS update, and the lower part generates the tonal disturbance and writes the anti-noise signal to the amplifier.

4. Simulation results

4.1 Setup

To provide a quantitative reference that any user can reproduce without the hardware, the control structure was simulated with the plant models distributed with the OCEAN exercise material. The primary path is P(z) = A(z) B(z) with:

A = [0.5, 0.5, −0.3, −0.3, −0.2, −0.2] and

B = [0.1, −0.1, 0.2, −0.2, 0.3, −0.3, 0.15, −0.15].

The secondary path is S(z) = A(z), and the controller uses:

Ŝ = [0.466, 0.533, −0.257, −0.274, −0.231, −0.175], i.e. it carries a deliberate modelling error of a few percent.

The construction is didactically deliberate: since S(z) = A(z), the optimal control filter is W_opt(z) = P(z)/S(z) = B(z), an FIR filter of exactly eight taps. An exact solution exists, so the penalty of choosing the control filter too short or unnecessarily long becomes directly visible. The sampling rate is 12 kHz as in the exercise, the reference is band-limited to 2 kHz to reflect the plane-wave range of the duct, and a white noise floor 45 dB below the disturbance is added at the error microphone to represent MEMS microphone self-noise. Without such a floor the attainable attenuation would be limited only by numerical precision.

4.2 Convergence and attenuation

For the band-limited broadband disturbance the residual falls by 35 dB, but convergence extends over several seconds (Fig. 5a). For a two-tone disturbance at 200 Hz and 443 Hz the controller reaches 44.8 dB within about 50 ms, i.e. it hits the microphone noise floor. The contrast demonstrates why narrowband ANC is so much easier: for a periodic disturbance the filter only has to match two amplitudes and two phases and the reference is perfectly coherent, whereas broadband control must approximate an entire transfer function and is limited by the causality constraint between reference and error sensor. The spectra in Fig. 5b confirm that the broadband reduction is distributed over the whole controlled band, without spectral enhancement outside it.

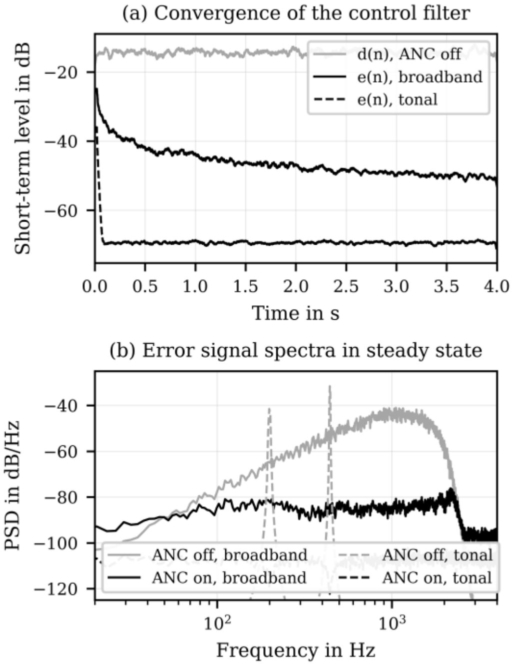

Figure 5. Simulated performance with the published reference plant models: (a) short-term level of disturbance and residual for broadband and tonal excitation; (b) power spectral densities in steady state. L = 32, μ = 0.05.

4.3 Parameter study

The attenuation after two seconds of adaptation rises from 20 dB at $\mu = 0.002$ to 39 dB at $\mu = 0.4$, and the algorithm diverges for $\mu \geq 0.8$ (Fig. 6a). Within a fixed adaptation time a larger step size simply converges further; the practical lesson is that the useful range ends abruptly rather than gradually. Fig. 6b confirms the effect of filter length. Four taps are clearly insufficient and yield 6.5 dB; the optimum lies near twelve taps, slightly above the eight of the exact solution because the mismatch of $\hat{S}(z)$ must be compensated; beyond that the attenuation decreases to 25 dB at 128 taps, because every extra coefficient contributes gradient noise without contributing to the solution. The common student assumption that a longer adaptive filter is always better is refuted by a single curve. Fig. 6c addresses stability. A pure delay error was applied to $\hat{S}(z)$ while the true secondary path was left unchanged. Attenuation remains near 32 dB for delay errors between −1.25 and +0.75 samples, degrades by about 6 dB up to +1.75 samples, and the loop diverges from +2 samples onwards. At 12 kHz two samples correspond to 167 μs, a phase error of 120° at the upper edge of the control band, consistent with the classical result that FxLMS tolerates secondary path phase errors up to roughly 90° [13]. The narrowness of this margin explains why pre-training cannot be skipped.

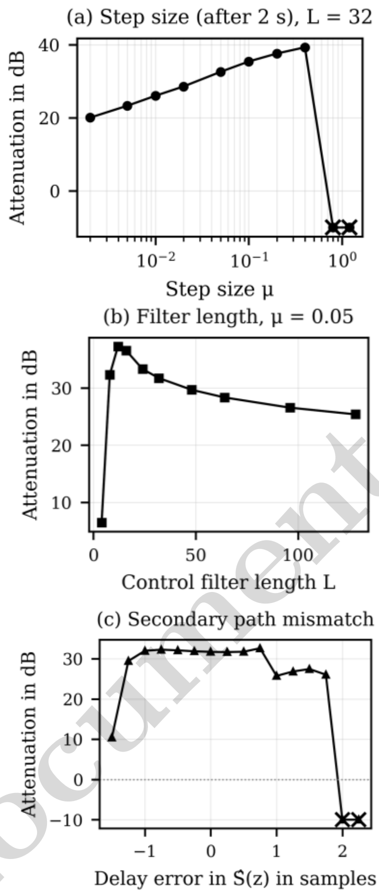

Figure 6. Influence of the controller parameters on broadband attenuation: (a) step size after 2 s of adaptation; (b) control filter length; (c) delay error of the secondary path estimate.

5. Integration into teaching

OCEAN is used in the master module Active Acoustics of the mechanical engineering programme at the University of Applied Sciences and Arts Dortmund. Five exercises build on one another: audio signal processing in Python, digital filters in MATLAB, real-time audio in Simulink, FxLMS simulation in Simulink, and ANC on embedded hardware. The first four are purely computational; the fifth uses the demonstrator and is structured in three stages.

First, the students drive the two loudspeakers with sine signals and adjust phase and amplitude manually until the microphone level is minimal. There is no adaptive filter; the purpose is physical intuition. They find the phase of maximum cancellation, observe that it is never complete, and explain why. Second, they run the pre-training model, observe the convergence of the two identification filters and export the coefficients. Third, the full FxLMS controller is deployed and they vary step size, filter length, leakage and disturbance frequency while listening to the result.

The learning outcomes of the third stage are precisely the questions Fig. 6 answers in simulation, but now with audible consequences. Combining simulation and hardware experiment in one exercise (Fig. 7) makes the transition from model to physical system explicit, including the effects the model does not contain.

Figure 7. OCEAN setups during the laboratory course in the winter term 2025/26.

The economic consequence is decisive. A course of twenty students in groups of three needs seven setups, an investment of roughly 1000 € including printing. This is an order of magnitude below a single conventional ANC bench, and it changes the format of the course: every group builds, commissions and debugs its own controller.

6. Open-source release

The platform is published in a public repository [14] containing the CAD models in native and neutral formats with print-ready meshes, the bill of materials with supplier part numbers, the STM32CubeMX project defining the peripheral configuration, the Simulink models for the three stages of the hardware exercise, the simulation script of Section 4, the exercise sheets and lecture material, and the assembly and commissioning documentation. Licences are chosen per part: permissive for source code and models, an open hardware licence for the mechanical design files, and a Creative Commons attribution licence for documentation and teaching material. Tagged releases carry a persistent identifier so that a specific hardware and software revision can be cited unambiguously, which is the prerequisite for reproducibility.

Two limitations should be stated plainly. The model-based workflow requires MATLAB and Simulink licences; a reference implementation in plain C that removes this dependency is the first item on the roadmap. Furthermore, the results in Section 4 are simulation results obtained with the published reference plant models. A measurement campaign on the hardware, covering the identified secondary path, the attenuation as a function of frequency and the convergence behaviour under realistic disturbances, is in preparation and will be published with the first tagged release. Further planned extensions are a hybrid feedforward and feedback structure, a multi-channel variant, and automatic re-identification of the secondary path during operation [12].

7. Conclusions

OCEAN demonstrates that a fully functional duct ANC experiment, including transducers, real-time controller and the complete model-based toolchain, can be built for a component cost of 92 €. A 3D-printed duct, digital MEMS microphones, a digital class-D amplifier and a Cortex-M7 microcontroller together remove the cost barrier and the workshop dependency that have restricted hands-on ANC to well-equipped laboratories. Beyond cost, the open release addresses a reproducibility problem. When geometry, transducers, controller and identification procedure are all available as files, an experiment reported by one group can be repeated by another. The authors invite other laboratories to build the platform and to contribute duct variants, control algorithms and course material.

8. Acknowledgments

The development of the teaching concept was supported by the internal teaching funding programme (HiLF) of the University of Applied Sciences and Arts Dortmund.

References

[1] S. M. Kuo and D. R. Morgan, “Active noise control: a tutorial review,” Proc. IEEE, vol. 87, no. 6, pp. 943–973, 1999.

[2] S. J. Elliott and P. A. Nelson, “Active noise control,” IEEE Signal Process. Mag., vol. 10, no. 4, pp. 12–35, 1993.

[3] P. A. Nelson and S. J. Elliott, Active Control of Sound. London, U.K.: Academic Press, 1992.

[4] S. M. Kuo and D. R. Morgan, Active Noise Control Systems: Algorithms and DSP Implementations. New York, NY, USA: Wiley, 1996.

[5] C. H. Hansen and S. D. Snyder, Active Control of Noise and Vibration. London, U.K.: E & FN Spon, 1997.

[6] M. L. Munjal, Acoustics of Ducts and Mufflers, 2nd ed. Chichester, U.K.: Wiley, 2014.

[7] Y. Kajikawa, W.-S. Gan, and S. M. Kuo, “Recent advances on active noise control: open issues and innovative applications,” APSIPA Trans. Signal Inf. Process., vol. 1, e3, 2012.

[8] J. M. Pearce, “Building research equipment with free, open-source hardware,” Science, vol. 337, no. 6100, pp. 1303–1304, 2012.

[9] E. Bjarnason, “Analysis of the filtered-X LMS algorithm,” IEEE Trans. Speech Audio Process., vol. 3, no. 6, pp. 504–514, 1995.

[10] S. J. Elliott, Signal Processing for Active Control. London, U.K.: Academic Press, 2001.

[11] S. M. Kuo and D. Vijayan, “A secondary path modeling technique for active noise control systems,” IEEE Trans. Speech Audio Process., vol. 5, no. 4, pp. 374–377, 1997.

[12] L. J. Eriksson and M. C. Allie, “Use of random noise for on-line transducer modeling in an adaptive active attenuation system,” J. Acoust. Soc. Am., vol. 85, no. 2, pp. 797–802, 1989.

[13] S. D. Snyder and C. H. Hansen, “The effect of transfer function estimation errors on the filtered-x LMS algorithm,” IEEE Trans. Signal Process., vol. 42, no. 4, pp. 950–953, 1994.

[14] A. Fortino and L. Lübbert, “OCEAN – Open Control Environment for Active Noise Control,” https://github.com/OCEAN-ANC, 2026.