D. Stefani, M. Binelli, A. Farina, and L. Turchet,
"Improved Real-Time Six-Degrees-of-Freedom Dynamic Auralization
Through Nonuniformly Partitioned Convolution,"
J. Audio Eng. Soc., vol. 73, no. 10, pp. 671–681 (2025 Oct.).
https://doi.org/10.17743/jaes.2022.0224.

Improved Real-Time Six-Degrees-of-Freedom Dynamic Auralization Through Nonuniformly Partitioned Convolution

DOMENICO STEFANI,[1,*] MARCO BINELLI,[2] ANGELO FARINA,[2] *AES Fellow* **AND**
(domenico.stefani@unitn.it) (marco.binelli@unipr.it) (angelo.farina@unipr.it)

LUCA TURCHET,[1] *AES Member*
(luca.turchet@unitn.it)

[1]*Department of Information Engineering and Computer Science, University of Trento, Trento, Italy*
[2]*Department of Engineering and Architecture, University of Parma, Parma, Italy*

Recent years have witnessed an increasing interest from the academic and industrial research community toward software for dynamic auralization and six-degrees-of-freedom (6DoF) navigation of immersive audio environments. Some existing tools rely on the convolution of source sounds with Ambisonics impulse responses (IRs) recorded in real spaces. However, despite advancements in computing power of modern central processing units, convolution remains a demanding computation to perform, especially with many channels and in real time. Moreover, efficient computation schemes often used in single-IR matrix tools have not made their way into open-source 6DoF spatial audio plugins. This paper presents MCFX-6DoFconv, an open-source 6DoF convolution plugin combining the efficient convolution engine of the MCFX-Convolver plugin with the 6DoF navigation features of SPARTA 6DoFconv, along with functional and interface improvements. Compared with the original SPARTA 6DoFconv, the proposed plugin yields a considerable increase in computing efficiency throughout a wide range of IR lengths, number of channels, and audio buffer sizes, up to a 3.7-fold improvement. This enables real-time auralization with longer IRs and multiple source rendering with more plugin instances. Moreover, the proposed plugin enables instant listener-position updates, eliminating delays up to two buffer sizes and removing the audio latency caused by internal buffering.

0 INTRODUCTION

Sound spatialization and virtual acoustics have become increasingly vital in numerous applications, from virtual reality (VR) [1] to gaming [2] and acoustic preservation [3, 4]. The ability to render spatial audio through dynamic auralization with six degrees-of-freedom (DoF) [5], allowing listeners to move and rotate freely within a virtual acoustic space, represents a significant advancement in creating immersive audio experiences. This is particularly valuable as virtual and augmented reality technologies continue to evolve and demand more sophisticated spatial audio rendering solutions. Moreover, this technology enables the documentation, virtual reconstruction, and experiential preservation of acoustics of heritage spaces, allowing future generations to experience the sonic characteristics of historically significant environments [3].

*To whom correspondence should be addressed, email: domenico.stefani@unitn.it

A common approach for achieving realistic 6DoF spatial audio involves convolving audio from a source with Ambisonics [6] impulse responses (IRs) captured in actual spaces [7]. Through this process, a source audio signal is encoded in the Ambisonics B-format with the virtual acoustic properties contained in the IR matrices. This method is particularly valuable for acoustic reproduction because it can preserve acoustic characteristics captured from an original space. Although this approach can produce spatially accurate results, given the use of sufficiently high Ambisonics orders [8], it presents substantial computational challenges in real-time applications where multiple audio channels must be processed simultaneously. Despite the steady increase in computing power of modern systems, real-time convolution remains a significant technical hurdle, particularly when implementing it within accessible, open-source tools.

Current open-source solutions for 6DoF dynamic auralization often prioritize either functionality or performance but rarely achieve an optimal balance between the two. This has created a notable gap in the tool kit available to researchers, developers, and audio professionals. Notably, the main open-source tools for convolution-based Ambisonics dynamic auralization with a moving listener (i.e., SPARTA 6DoFconv [7] and RoomZ [9]) offer different sets of functionalities with some overlap, but they rely on partitioned convolution schemes that can be computationally expensive, limiting the duration of IRs, their Ambisonics order, or the number of sources that can be rendered in real time. More efficient convolution partitioning schemes [10, 11] are used in convolution tools for stereo reverberation, binaural rendering, and Ambisonics auralization, but have not been applied in open-source or free tools for moving-listener dynamic auralization with Ambisonics IRs.

To bridge these gaps, this paper introduces MCFX-6DoFconv, an open-source virtual studio technology (VST) plugin for convolution-based dynamic auralization combining the 6DoF navigation features of the SPARTA 6DoFconv plugin, the efficient nonuniform partitioned convolution engine of MCFX-Convolver, and several additional improvements. The original SPARTA 6DoFconv is a VST that can load multiple Ambisonic IR matrices measured at different points of a room relative to a single source. The plugin can then convolve a monophonic audio signal with the IR matrix that is closest to the listener's coordinates. The listener can move around the room, and getting closer to a different IR matrix causes the plugin to switch, handling potential audio discontinuities by applying crossfade to the signals convolved with the previous and current IRs. After convolution, listener rotation is applied by means of Ambisonic scene rotation. The plugin can be followed by an auralization method such as binaural rendering.

The proposed MCFX-6DoFconv is open-source and made available online, along with the comparison benchmark.[1] Here, the aim is to reduce the central processing unit (CPU) load of the SPARTA 6DoFconv to enable real-time execution with smaller audio buffers, longer room IRs (RIRs), and multiple plugin instances for multisource rendering. Moreover, the aim is to reduce delays in the change of IR matrices on listener movement, reduce audio latency, and improve ease of use through features such as automatic IR resampling. Finally, it is intended to ease future integration of multisource auralization by using the multiple-in-multiple-out MCFX-Convolver engine.

The remainder of the paper is organized as follows. SEC. 1 provides background on convolution-based dynamic auralization tools. In SEC. 2, the proposed system is described. SEC. 3 presents the result of a comparative evaluation of the software with its original version. Finally, SEC. 4 discusses the findings, and conclusions are drawn in SEC. 5.

1 BACKGROUND

Real-time auralization tools can be used to experience virtual acoustic spaces, particularly in virtual and augmented reality applications. Among real-time auralization tools, those that can perform real-time convolution with custom RIRs are particularly valuable but relatively scarce. Even fewer support dynamic auralization with source or listener movement, which requires both a dynamic convolution engine and user interface controls for movement and rotation. A review of available software revealed only two freely available audio plugins that offer real-time dynamic auralization with custom RIRs: SPARTA 6DoFconv and RoomZ. Both tools implement different feature sets and interfaces. Tools for auralization based on other techniques are not taken into account because they are fundamentally different and used for different purposes (e.g., virtual room acoustic modelers [12]).

The following sections provide an overview of these two tools and their limitations. Moreover, SEC. 1.3 presents MCFX-Convolver, a plugin for multichannel convolution with no dynamic navigation capabilities but with an efficient nonuniform partitioned convolution algorithm and with the possibility of sending an open sound control (OSC) command that forces the loading of a new finite IR filter matrix corresponding to the new position of the source or of the receiver.

1.1 SPARTA 6DoFConv

SPARTA 6DoFconv [7] is an open-source VST plugin from the SPARTA plugin collection [13] for convolution with B-format/Ambisonic IRs with 6DoF listener movement. The plugin implements a uniformly partitioned convolution scheme in the frequency domain (see SEC. 1.4), combined with an IR selection method based on nearest neighbors (NN) search and cross-fading. The plugin supports full 6DoF navigation, enabling both 3D listener movement and three degrees of rotation through Ambisonic scene rotation. Three-DoF navigation is available with single-in-multiple-out IR matrix other than Ambisonics. IR matrices are loaded as spatially oriented format for acoustics (SOFA) files [14], which can include any metadata needed (e.g., 3D IR measurement coordinates, source coordinates). SPARTA 6DoFconv can also be controlled through OSC messages, enabling listener movement and rotation via external interfaces, head-tracking sensors, or VR systems. For these reasons, SPARTA 6DoFconv has been used for real-time auralization and 6DoF navigation of VR environments, reproducing the acoustics of heritage sites [4]. SPARTA 6DoFconv supports Ambisonic RIRs up to the tenth order (121 channels). Fig. 1 shows the interface of the SPARTA 6DoFconv plugin.

1.2 RoomZ

RoomZ [9] is a freeware VST plugin that supports both listener and source movement in 1D or 2D spaces and implements an advanced interpolation mechanism where three selected IRs contribute to the final rendered audio at each point. For IR selection, RoomZ offers two methods: K-NN (k = 1,2,3) search and Delaunay triangulation [15]. This approach enables smoother transitions between positions compared with single-NN methods [16], although at

[1] www.github.com/CIMIL/6DoF-SpatialAudioConvolver

Fig. 1. Interface of the SPARTA 6DoFconv plugin. The left part of the interface allows users to load a SOFA file with single-in-multiple-out IRs, shows IR information, and allows users to change the listener position along three axes and three rotation angles. The plugin shows a map of the room.

the cost of increased computational load (tripled with k=3 [16]). However, RoomZ only supports 2D movement and does not provide listener rotation, limiting its applicability in full 6DoF scenarios. When using Ambisonics IRs, rotation can be provided by an external Ambisonics scene rotator plugin, achieving 5DoF movement. Furthermore, RoomZ does not integrate OSC control. RoomZ scenario configurations are based on XML metadata and waveform audio (WAV) files for IRs, and it does not support SOFA. Lastly, RoomZ is a closed-source project, where the plugin binaries are made available as freeware for commercial and noncommercial uses.

1.3 MCFX-Convolver

Although not designed for dynamic auralization, MCFX-Convolver [17] represents an important development in efficient multichannel convolution. MCFX-Convolver is an open-source multichannel convolution VST that implements an efficient nonuniform partitioned convolution algorithm based partially on Gardner's work [10], with significant improvements made by Fons Adriaensen in his JConv open-source software for Linux [18]. The plugin allows users to load a single IR matrix with up to 128 × 128 channels and perform static multichannel convolution, making it suitable for tasks such as Ambisonics-to-binaural decoding with custom head-related transfer functions and Ambisonics reverberation. Its efficient convolution engine makes it particularly effective for processing long IRs compared with uniform partitioning schemes. Differently from the previous tools, MCFX-Convolver does not support dynamic navigation or real-time IR switching, and changing the IR matrix stops the audio stream and clears the audio buffer. The interface of the most recent version of the MCFX-Convolver plugin is presented in Fig. 2.

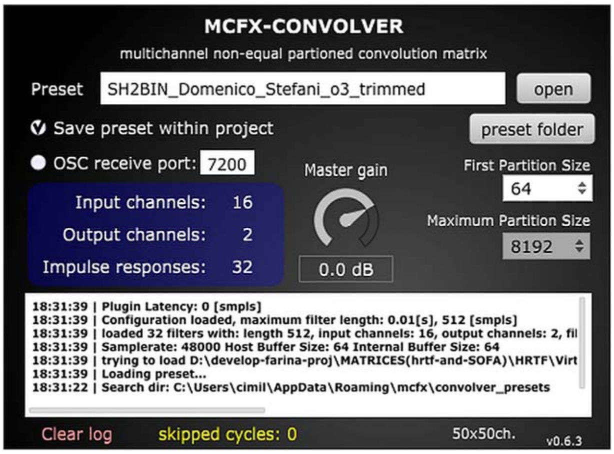


Fig. 2. Interface of the MCFX-Convolver plugin.

1.4 Current Limitations and Challenges

The existing dynamic auralization tools present different trade-offs between computational efficiency and rendering quality. SPARTA 6DoFconv's simpler NN approach requires denser RIRs for high-quality rendering [16]. Dense RIR grids can be achieved through offline interpolation of coarse IR measurements [7], which does not impact CPU usage during real-time processing, at the cost of higher memory usage. Instead, RoomZ's sophisticated interpolation method can be applied to coarser IR measurement grids but faces increased computational costs in real-time [9, 16]. Additionally, both tools rely on uniform partitioned convolution schemes, which can be computationally expensive with longer IRs or multiple sources. Convolution partitioning [19] consists of cutting long IRs into smaller partitions, allowing users to quickly convolve small blocks of input audio with early partitions while delaying convolutions with IR tails. IR partitioning can therefore enable low-latency real-time convolution. In detail, uniform partitioning refers to cutting IRs into equally sized partitions. Beyond uniform partitioning, considerably more computationally efficient *nonuniform* partitioning schemes [19] have been available for two decades now [10, 11].

Despite these efficient schemes being used in various "static" real-time convolution software [20, 17], including MCFX-Convolver, the available dynamic auralization tools described still rely on simpler uniform partitioned convolution. However, real-time dynamic auralization tools stand among the applications most dependent on efficient computation, given their need for high-channel convolutions and interpolation between multiple convolution lines. This creates a notable gap in available solutions.

For these reasons, along with the free and open-source software (FOSS) general public license version 3 (GPLv3) license of both SPARTA 6DoFconv and MCFX-Convolver, a plugin is proposed here that is based on the interface and navigation capabilities of SPARTA 6DoFconv and the efficient convolution engine of MCFX-Convolver. Table 1 presents an overview of the features of the mentioned auralization tools, with the addition of the 6DoF plugin proposed in this paper.

Table 1. Features of prominent freely available real-time multichannel convolution-based auralization plugins. MCFX-6DoFconv (bold) is the plugin proposed here.

Plugin	DoF	License	Interpolation	IR partitioning	Receiver	Source	IR format	OSC
RoomZ	2	Closed-source freeware	1–3 NN or triangle	Uniform	Multiple, moving	Multiple, moving	Wav + XML	No
SPARTA 6DoFconv	6	FOSS (GPLv3)	1-NN selection	Uniform	Single, moving	Single, static	SOFA	Yes
MCFX	0	FOSS (GPLv3)	NO	Nonuniform	Single, static	Multiple, static	Wav matrix	Minimal
MCFX-6DoFconv proposed	6	FOSS (GPLv3)	1-NN selection	Nonuniform	Single, moving	Single, static	SOFA	Yes

Note: XML = extensible markup language.

2 SYSTEM DESCRIPTION

The proposed spatial audio plugin represents a significant redesign of the SPARTA 6DoFconv tool, introducing multiple performance, functional, and interface improvements across critical signal processing and user interaction domains. The primary objectives of the redesign were to address computational efficiency and reduce processing latency.

2.1 Convolution Engine

The most significant architectural modification introduced in the proposed plugin is the integration of the MCFX-Convolver engine, which employs a nonuniform partitioned convolution approach. The MCFX-Convolution engine replaced the previous *tvconv*, which is a single-thread uniformly partitioned convolution implementation based on the SPARTA Multiconv plugin.

The IR partitioning scheme of the MCFX begins with four initial partitions of equal size, corresponding by default to the host buffer size (hBS). The parameter is exposed in the graphical user interface (GUI) as `first partition size`. Then, the following four partitions are double the size of the first four. Doubling the partition size in groups of four continues until the IR length is reached or a maximum partition size is reached. The maximum size is initialized to 8,192 samples and can be modified from the plugin GUI along with the first partition size.

The main interface to the convolver is through its `ConvolverData` and `MtxConvMaster` classes, where the first holds the IR matrices and the latter performs partitioning and directs computation for each partition. The frequency domain convolution for all partitions of different sizes is performed in a multithreaded manner through objects of the `MtxConvSlave` class. Convolution of the first partition is always performed immediately in the main audio thread. By construction, the MCFX-Convolver introduces no delay to the buffered processing of the digital audio workstation (DAW). Forward and inverse discrete Fourier transform computations are performed with the FFTW library [21].

Because the MCFX-Convolver plugin is not intended for dynamic auralization, its engine had to be modified to fit the new use case. In particular, the original MCFX convolution engine allocates memory for the IR partitions whenever the IR matrices are changed. Because in the proposed plugin, this is bound to happen during audio playback whenever the listener moves through IR measurement positions, the engine was modified to allocate memory only at SOFA load time and to only replace IR data onto the preallocated structures at runtime, therefore avoiding unbounded time operations in the audio thread and resulting audio artifacts.

Additionally, the strategy for smooth IR swap on listener position change had to be reimplemented because it was tightly integrated within the single *tvconv* engine in SPARTA 6DoFconv. In the original plugin, whenever the IR matrix was changed, the process routine of *tvconv* would perform convolution twice, one for each matrix, and a one-buffer linear cross-fade was applied to the two outputs.

The same one-block cross-fade strategy was applied to the MCFX-Convolver by having two instances of `MtxConvMaster` alternate on IR swap and cross-fading their outputs. The option of more tight and efficient integration of cross-fade into the MCFX-Convolver (e.g., in frequency domain [19]) was discarded to allow future improvements of MCFX to be more easily merged to MCFX-6DoFconv. For the same reason, priming, cross-fading, and the remaining modifications to the MCFX-Convolution engine were applied through a C++ wrapper layer that matched the *tvconv* API and that of the new engine.

2.1.1 Impulse Response Change Delay

Analyzing the code of the original plugin revealed that *tvconv* introduced a delay of up to two audio buffers between receiving a change of position[2] that caused a change of IR matrix and the beginning of the cross-fade operation; this was likely done to prevent audible signal discontinuities while avoiding costly flush and reinitialization operations to the convolution line. However, introducing a position-change delay can potentially affect the user's feeling of immersion in head tracking–based simulation and VR environments with tightly spaced IR grids. Position-change

[2]Position change instructions are imparted through either VST parameter automation or OSC messages.

delay can compound with additional audio latency and head tracker processing delays [22], potentially crossing thresholds of noticeable latency [23].

For this reason, and having obtained positive preliminary efficiency measures on the new convolver, the proposed approach employs a more costly but nearly instant change of IR matrix. The new strategy consists of keeping a *priming buffer* with several of the most recent input audio blocks and using it to prepare the convolver for the new position upon change. Whenever a listener movement causes the selection of a new IR matrix, the inactive convolver instance is prompted to perform convolution on all blocks of past audio contained in the priming buffer, disregarding the output produced. This ensures that all partition buffers in the engines are filled with the results of the convolution of the most recent audio with the new IRs selected, preventing them from causing signal discontinuities when convolved with the new input blocks. When convolution with all blocks of the priming buffer is completed, the engine can convolve the current input signal block with the new IR matrix, and the output is cross-faded with that of the convolver for the previous position.

The listener position is, as in the original plugin, updated at the beginning of the processing for each audio buffer. As a consequence, as soon as the callback starts processing after a change in position, the change of position and cross-fade are started.

Currently, the size of the priming buffer is set to the length of the loaded IRs. Therefore, priming requires *IR length/hBS* calls to the convolver's process routine. A theoretically lower computational cost could be achieved by modifying how the MCFX-Convolver handles the flushing and initialization of the `MtxConvSlave` convolvers for the different-size partitions. However, once again, this option was discarded to allow future improvements of the MCFX engine to be more easily integrated into MCFX-6DoFconv. Perceptual validation of the extent of the impact of position-change delays will be addressed in future work.

2.1.2 Audio Latency

By default, MCFX-6DoFconv is set to initialize the size of the first IR partition to the hBS communicated by the DAW. Additionally, the first partition of the IR is always immediately convolved with the input audio buffers. As a consequence, the proposed plugin does not introduce any delay to the input audio beyond the audio buffering inherent to the operation of the host DAW. Although, in principle, the original SPARTA 6DoFconv could operate in the same manner, its convolver forces a lower limit of 512-sample buffers to its internal processing routine (and therefore partition size) and introduces its own buffering to store the input signal between calls to the processing routine. As it is understood here, this was to limit costly convolution operations with the uniform partitioning scheme and small partition sizes.

In SPARTA 6DoFconv, the internal buffering is performed in its processing routine, where an *input* first-in-first-out multichannel queue is progressively filled with multichannel frames from the input buffer. At the same time, output frames are taken from an *output* queue and written to the plugin's output buffer. Whenever the internal buffer size (iBS) value is reached, convolution is performed on the values in the input queue and output to the second queue. Frames in the output queue will only be consumed on the following call to the audio processing routine. As a consequence, SPARTA 6DoFconv introduces a constant audio delay that corresponds to its iBS. Additionally, this affects the aforementioned IR-change delay because it delays the output signal in relation to the moment in which IR matrices are changed. An analysis and comparison of the two behaviors is addressed in the evaluation (see SEC. 3).

2.2 Functional and Interface Enhancements

The main functional improvement introduced in the proposed plugin is automatic IR resampling. When adding IRs to MCFX-6DoFconv, automatic resampling is performed by the `ConvolverData` objects through the `ResamplingAudioSource` class of the JUCE framework. Although the original plugin halted on sample-rate mismatch, prompting users to change its value from the host side, MCFX-6DoFconv can function with resampled IRs. Nevertheless, it notifies users through the plugin's interface of the operation performed to avoid potential host misconfiguration from going unnoticed.

Moreover, minor additions were made to the plugin's interface (see Fig. 3). First, the axis representation of the graphical room visualization on the screen was changed from horizontal X and vertical Y to horizontal Y and vertical X to conform the reference system to the ISO 2631 standard [24] in the top view and rear view. Furthermore, grid computation was modified by replacing the previous fixed grid spacing with a spacing that is relative to the size of the room, reducing graphics computational complexity for large rooms. Finally, a status notification has been added to the GUI to notify whether the SOFA 1) is not initialized, 2) is loading, 3) has been loaded successfully, 4) is invalid, 5) contains unexpected dimensions, or 6) has an unexpected format.

3 EVALUATION

In this section, the evaluation of three key aspects of the proposed plugin is described: 1) the audio latency, 2) the IR position-change delay, and 3) the computational efficiency of the proposed plugin, comparing it with the original SPARTA 6DoFconv. All the measurement data and code are available in the project's repository.[1]

3.1 Audio Latency

Audio latency was assessed through source code analysis and a simple input-output test with 10-Hz impulses. Through a set of hBS values ranging from 64 to 2,048, it was found that the proposed MCFX-6DoFconv plugin introduces no delay beyond the DAW's inherent buffering process, whereas the original SPARTA 6DoFconv consistently introduces a latency of one of its internal buffers.

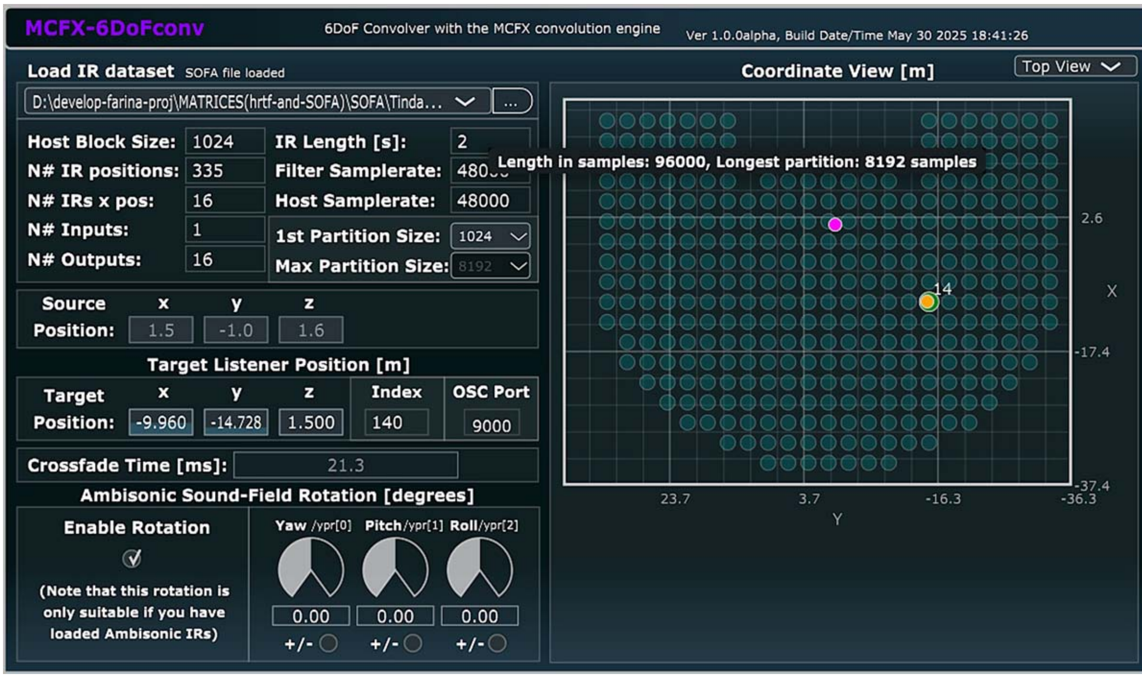


Fig. 3. Interface of the new MCFX-6DoFconv plugin. With respect to SPARTA 6DoFconv, the new plugin presents additional information displays (e.g., position crossover time in milliseconds) and controls (e.g., first and maximum partition size).

Table 2. Audio and IR-change latencies measured for the original SPARTA 6DoFconv plugin and the proposed MCFX-6DoFconv.

				IR-change latency	
Plugin	hBS	iBS	AL	[samples]	[ms]
SPARTA 6DoFConv	64	512	512	769	16.0
	128	512	512	833	17.3
	256	512	512	897	18.7
	512	512	512	1,025	21.4
	1,024	1,024	1,024	2,050	42.7
	2,048	2,048	2,048	4,097	85.4
MCFX-6DoF	64	64	0	0	0.0
	128	128	0	0	0.0
	256	256	0	0	0.0
	512	512	0	0	0.0
	1,024	1,024	0	0	0.0
	2,048	2,048	0	0	0.0

Note: All measures are in samples unless stated otherwise. All the reported latency measures exclude the 1-buffer-size delay introduced by the DAW's inherent buffering process. hBS = host-buffer-size; iBS = internal-buffer size; AL = audio latency.

SPARTA 6DoFconv's iBS corresponds to the hBS for all values greater than or equal to 512 and is fixed to 512 for all hBS values below (See Table 2).

3.2 Position-Change Latency

The position-change latency was measured for both the original and the proposed plugin through a custom patch with the Plogue Bidule Software. For the test, a stereo audio file containing a 10-kHz sine tone on the left channel and

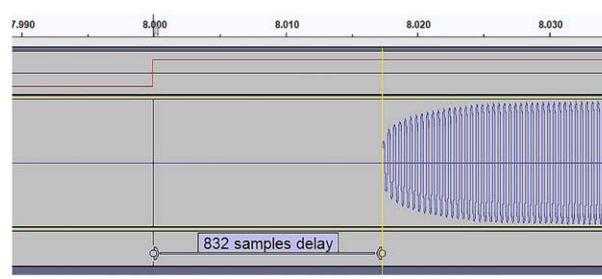


Fig. 4. Position-change delay visualized in Audacity.

a 1-Hz square wave on the right was prepared. Through Bidule's audio file player, the left channel was sent to an instance of the 6DoF plugin, and the right channel was sent to a parameter modulator object that was linked to the listener_y coordinate of the plugin. A test SOFA with two measurement positions was created for the occasion, where one position contained real IRs from a third-order Ambisonics SOFA, and the second contained IRs filled with zeros. The low-frequency input channel was rescaled to prompt a sharp switch of the position on each wave front, causing the plugin to select one or the other IR matrix. The low-frequency channel was also rescaled and multiplied to the left channel, which compensates for Bidule's internal processing delays, and the output was recorded along with the output of the 6DoF plugin.

The process was repeated for the original and proposed plugin with different audio buffer size values (64, 128, 256, 1,024, 2,048) and a 48-kHz sampling rate. After all the tests were recorded, the delay between the control change and the start of the cross-fade was measured with sample-accurate labels in the Audacity software (Fig. 4), thanks to the clear distinction between the sections of audio con-

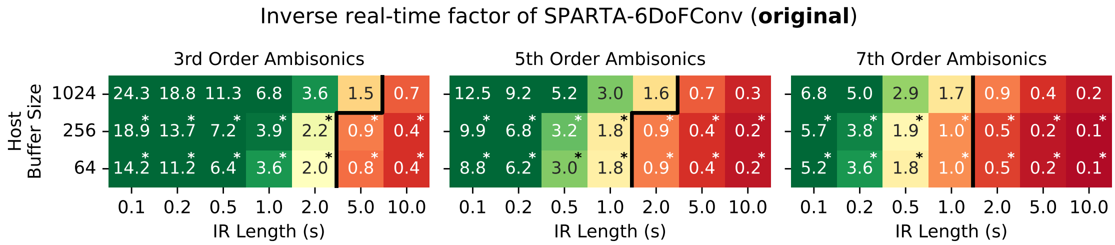


Fig. 5. The iRTF of the original SPARTA 6DoFconv plugin with different hBSs, IR length, and IR Ambisonics order. Solid lines divide configurations where real-time processing requires less time than the allotted time budget (left) to those where it takes more time (right), rendering real-time processing unfeasible. Asterisks mark all configurations where the audio latency introduced by the plugin is buffer-independent because it is fixed to 512 samples.

volved with, respectively, the real and zeroed IR matrices. The average results are presented in Table 2. Although the proposed plugin introduced no delay between the control message and the change of IR matrix, the original SPARTA 6DoFconv plugin was confirmed to introduce a delay that depends on the point at which the position is changed relative to the hBS and iBS and whose average corresponds to the following equation:

$$\text{Avg. IRswitch Delay} = hBS + 0.5 \times (hBS + iBS), \quad (1)$$

which factors in the audio latency reported in the previous section (i.e., $1 \times hBs$).

3.3 Computational Efficiency

To evaluate the computational performance of the proposed MCFX-6DoFconv, an extensive series of measurements was conducted that compared the plugin's efficiency against the original SPARTA 6DoFconv. Real-time factor (RTF) was adopted as the primary metric as a proxy for computational efficiency, defined as the time required to offline render a known input file divided by its actual duration.

Measurements were performed using the Cockos Reaper DAW under controlled conditions. Specifically, the measured plugin was hosted in a single multichannel track, and Reaper was instructed to render a known input file (white noise) at full speed offline, maintaining the selected buffer size without applying any offline optimization techniques. This approach ensured that the rendering process closely mimicked real-time processing conditions. Moreover, throughout all the tests, the listener position was automated through a 1-Hz triangle wave oscillator along the y-axis, with the full wave cycle switching through three distinct IR measurement positions on the test SOFA files described below.

The evaluation was repeated by varying the following three load-critical parameters for a total of 63 configurations per plugin:

1. *Ambisonics IR Order*: third, fifth, and seventh order. (i.e., 16, 36, and 64 channels).
2. *IR Duration*: ranging from 100 ms to 10 s (100 ms, 200 ms, 500 ms, 1 s, 2 s, 5 s, 10 s).
3. *hBS*: 64, 256, and 1024 samples.

All parameter ranges cover areas of feasible computation, with upper limits that stressed real-time processing. In particular, although both plugins support tenth Ambisonic order, seventh-order IRs were found to stress computational limits sufficiently, with IRs not being processed in real-time with the original plugin.

Measurements for each configuration were performed twice and averaged. The SOFA files for the test were generated by modifying a SingleRoomSRIR[3] SOFA file, containing third-order 48-kHz IR matrices measured at different positions of the Greek-Roman theater of Tyndaris (Sicily, Italy) [25] with a single source in the center of the stage. Starting from a SOFA file with third-order, 2-s-long Ambisonics IRs, 20 additional SOFA files were created by virtually reducing or inflating the number of channels and length of the IRs, matching the aforementioned configurations.

Measurements were performed on a 2020 Dell XPS 15 9500 laptop with the Windows 10 operating system and an Intel Core i7-10750H CPU with six cores and 32 gigabytes of RAM. Most background processes were disabled, including networking and Bluetooth.

We report the multiplicative inverse (i.e., reciprocal) of RTF (iRTF) measured via Reaper, defined as input duration/render time, because it shows how many instances of the same plugin could be hosted while processing in real-time on average. The iRTF for both the original SPARTA 6DoFconv and the proposed MCFX-6DoFconv are shown in Figs. 5 and 6. Different audio and position change latencies identified in Secs. 3.2 and 3.1 are marked in all the affected configurations of Fig. 5. For a direct comparison, Fig. 7 presents a heatmap of the speedup factor between the two plugins, measured as the iRTF of the proposed MCFX-6DoFconv divided by that of the original SPARTA 6DoFconv.

The results show the superior efficiency of the proposed plugin in all real-time configurations and generally superior or equivalent performance in 60 configurations out of 63. SPARTA 6DoFconv performs slightly better than the proposed plugin in only three cases, which correspond to 1,024 samples hBS and 10-s long IRs. This can be attributed to

[3]SOFA convention for Spatial room IRs measured in a single room with a single excitation source and a listener containing an arbitrary number of receiver channels.

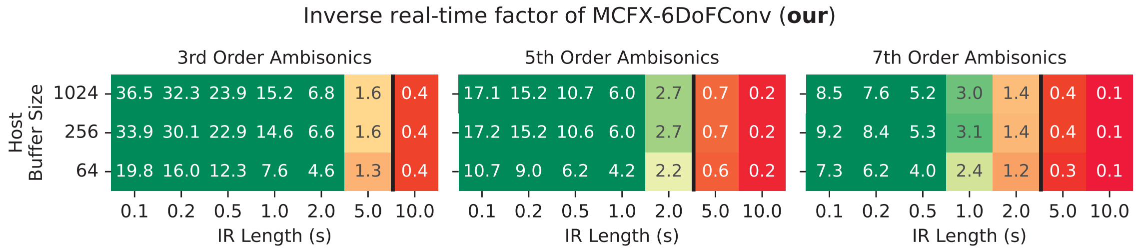


Fig. 6. The iRTF of the proposed MCFX-6DoFconv plugin. In contrast to the original, the proposed plugin does not introduce audio latency. In all but three configurations, the proposed plugin performs better or comparably, and real-time division lines are closer to the right of each heatmap, indicating how more IR configuration can be convolved in real-time on the test laptop.

the IR-change operation becoming more complex with the long IRs.

Additionally, during the test, it was noticed that the original plugin showed severe lagging and long plugin load times with longer IRs (2 s and over) and channel count. Furthermore, the DAW stalled for several dozens of seconds when asked to render a file. These issues with the original plugin were not present in one used here and can be attributed to the partition creation and discrete Fourier transform computation phase because it seems to be repeated when the audio context is stopped and restarted. Load times with the new plugin were always reasonable ($<$10 s).

After measuring the performance of both plugins as they are, a fairer comparison was conducted without the fixed audio latency of SPARTA 6DoFconv. To do so, the original plugin was modified by removing the lower cap to the *iBS* of the convolver (therefore allowing for smaller IR partition sizes). As a result, the latency of SPARTA 6DoFconv was reduced from the previous iBS to 1 hBS. Subsequently, the measurements were repeated with hBS values smaller than 512 on the modified SPARTA 6DoFconv plugin.

The iRTF measures on the modified plugin are reported in Fig. 8, and the speedup factor of the proposed plugin compared with the former is presented in Fig. 9. As expected, the performance of SPARTA 6DoFconv is greatly degraded when using smaller partition sizes owing to the struggle of the uniform partitioned convolution engine to scale to a high count of small-sized partitions. The number of SOFA configurations that can be rendered in real time with the modified SPARTA 6DoFconv is greatly degraded, with hundreds of milliseconds being the size of the longest IRs that can be convolved safely with the 64-sample buffer size.

4 DISCUSSION

The results of the evaluation demonstrate significant improvements in computational efficiency and latency reduction achieved by the proposed plugin. The integration of MCFX's nonuniform partitioned convolution engine has yielded substantial performance benefits that are particularly evident in configurations with longer IRs and higher Ambisonics orders.

The computational efficiency gains are found to be most pronounced with smaller buffer sizes and longer IRs, where the proposed plugin outperforms the original SPARTA 6DoFconv by up to 3.7 times while operating with considerably lower audio and IR latency. This improvement enables real-time processing of longer IRs and higher-order Ambisonics, expanding the practical applications of dynamic auralization. The performance advantage is particularly relevant for virtual acoustic rendering of highly reverberant spaces, where longer IR lengths are necessary to capture the full acoustic character of the environment. A clear indication of the efficiency improvement can be seen through the real-time division lines in Fig. 7, which show how seven IR configurations that could not be processed in real-time with the original plugin can now be handled with the proposed plugin.

The elimination of the audio latency related to internal buffering represents a significant advancement for applications requiring minimal processing delay, such as live

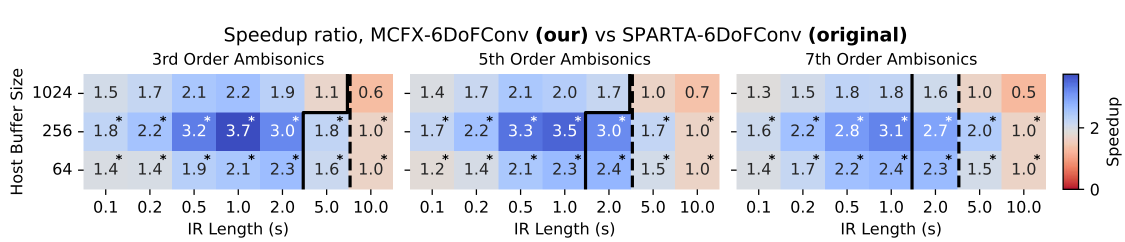


Fig. 7. Speedup ratio between the new MCFX-6DoFconv plugin over SPARTA 6DoFconv. The newer plugin processes the same input audio file in a shorter-or-equal time with all configurations apart from 10-second IRs with a hBS of 1024 samples. The solid line marks real time division of the original plugin; the dashed line marks the division for the proposed plugin (real time to the left and non–real time to the right). Asterisks mark all configurations where the iBS for SPARTA-6DoF is limited at a minimum 512 samples, introducing fixed audio latency.

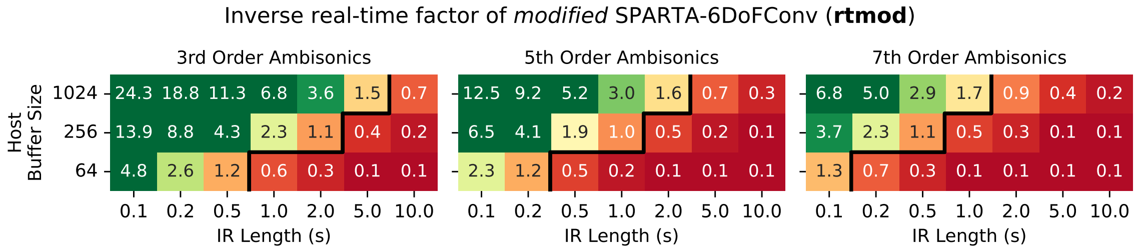


Fig. 8. The iRTF of the modified (rtmod) version of the original SPARTA 6DoFconv, where the lower limit to the convolver's buffer size of 512 samples was removed. The average performance of the convolver is considerably degraded compared with the original (Fig. 5) because its uniform partitioning scheme now requires the IR partition size to match even the smallest hBSs (<512 samples), resulting in a great number of partitions to be created and a sizable increase of operations to be performed for each audio buffer. Real-time division lines moved to the left as a consequence.

auralization. Although the original SPARTA 6DoFconv plugin's forced internal buffering provided computational stability at the cost of its latency, the proposed implementation demonstrates that efficient real-time processing is achievable without this compromise, provided an efficient convolution algorithm is employed. Despite "live" 6DoF auralization currently being a niche application, the drastic reduction in input-output audio latency can facilitate the exploration of these possibilities.

When comparing the efficiency of the proposed plugin against the modified SPARTA 6DoFconv (i.e., without a lower limit on IR partition size and with comparable audio), the performance improvement is even more pronounced (Fig. 9), with MCFX-6DoFconv being up to 23.7 times faster with 64-channel, 1-s-long IRs. This comparison highlights the advantages of the new nonuniform partitioned convolution engine with multithreading support. It is also worth noting that, despite the modified version of SPARTA 6DoFconv having comparable audio latency to MCFX-6DoFconv, the former still exhibits a position-change delay of two buffers (although, now iBS and hBS correspond) that is removed in the proposed plugin (Table 2).

The removal of position-change delay is the most relevant latency improvement because it can enhance the responsiveness of the virtual acoustic environment in all scenarios (i.e., playback with a moving listener or live auralization), especially in sensor-controlled movement applications such as VR. In the literature, noticeable head-tracker to sound latency thresholds have been found to be approximately 60 to 70 ms [22], and although the motion-to-sound latency of SPARTA 6DoFconv was measured to be 36 ms in [26], this study has shown how such delay depends in a nonlinear manner on the chosen buffer size. Moreover, compounding latency factors, such as sensor processing delays, could cause the total motion-to-sound latency to cross noticeable thresholds. Although the IR change delay is drastically cut in the proposed plugin, the extent of its repercussions on perceptual quality remains an open question that warrants further investigation through formal listening tests.

Some limitations of the current implementation should be acknowledged. The priming buffer approach for position changes, although effective, could be optimized further through tighter integration with the MCFX engine. Additionally, the performance with very long IRs (10 s) at large buffer sizes shows slightly lower efficiency compared with the original plugin, suggesting room for optimization in these extreme cases.

5 CONCLUSIONS

This paper has presented MCFX-6DoFconv, an open-source VST plugin that significantly advances the state of real-time dynamic auralization tools through several key innovations. Through the integration of a nonuniform partitioned convolution engine, the plugin achieves up to 3.7 times improvement in computational efficiency compared with its predecessor. Audio latency besides inherent DAW

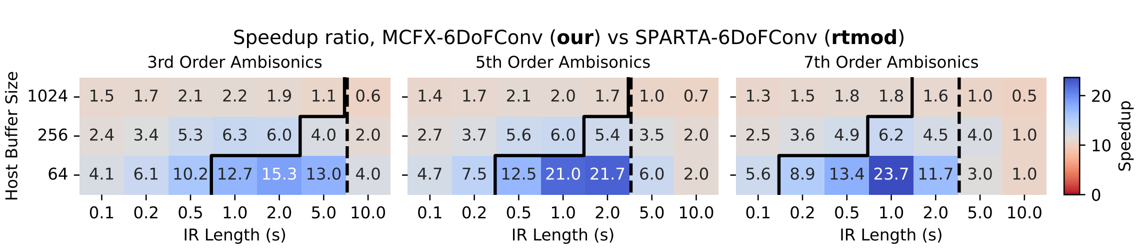


Fig. 9. Speedup ratio between the new MCFX-6DoFconv plugin over the modified version of the original plugin. The proposed plugin can process the same input signals up to 23.7 times faster than the modified version of SPARTA 6DoFconv, indicating a greatly reduced use of computational resources when processing real-time inputs. The solid line marks real time division of the original plugin; the dashed line marks the division for the proposed plugin (real time to the left and non–real time to the right).

buffering has been eliminated for all buffer sizes and removed position change delays have been removed and automatic IR resampling has been added. The user interface has been enhanced with improved status feedback and visualization.

The substantial performance improvements enable real-time processing of longer IRs and higher-order Ambisonics, expanding the practical applications of dynamic auralization in virtual and augmented reality, acoustic heritage preservation, and multichannel audio production.

Future work will focus on perceptual evaluation to assess the impact of position-change latency on user experience. Furthermore, future work will investigate additional features of the new convolution engine, such as the potential for multisource dynamic auralization in a single plugin instance, which could reduce the load overhead that is currently incurred by using multiple instances of the plugin. The plugin is made available as open-source software, enabling further development by the academic and professional audio communities and fostering the reproducibility of the results presented in this work.

6 ACKNOWLEDGMENT

This work has been supported in part by the MUSMET project funded by the European Innovation Council (Grant Number 101184379) and in part by the PR-FESR 2021-27 project funded by Regione Emilia-Romagna–Priority 1–Action 1.1.2, CUP E97G22000580007, titled “AGORA.” Views and opinions expressed are those of the authors only and do not necessarily reflect those of the European Union or the European Innovation Council.

Neither the European Union nor the European Innovation Council can be held responsible for them. This work is dedicated to Angelo Farina, who passed away in March 2025 shortly after submission.

7 REFERENCES

[1] J. Yang, A. Barde, and M. Billinghurst, “Audio Augmented Reality: A Systematic Review of Technologies, Applications, and Future Research Directions,” *J. Audio Eng. Soc.*, vol. 70, no. 10, pp. 788–809 (2022 Oct.). https://doi.org.10.17743/jaes.2022.0048>.

[2] B. Cowan and B. Kapralos, “Spatial Sound for Video Games and Virtual Environments Utilizing Real-time GPU-Based Convolution,” in *Proceedings of the Conference on Future Play: Research, Play, Share*, pp. 166–172 (Toronto, Canada) (2008 Nov.). https://doi.org/10.1145/1496984.1497012.

[3] B. F. G. Katz, D. Murphy, and A. Farina, “The Past Has Ears (PHE): XR Explorations of Acoustic Spaces as Cultural Heritage,” in *Proceedings of the 7th International Conference on Augmented Reality, Virtual Reality, and Computer Graphics*, pp. 91–98 (Lecce, Italy) (2020 Sep.). https://doi.org/10.1007/978-3-030-58468-9_7.

[4] L. Lavagna, A. Astolfi, L. Shtrepi, and A. Farina, “Esplorare il Passato Attraverso l’Auralizzazione Dinamica dei Beni Culturali: Il Caso Studio del Teatro Greco-Romano di Tindari,” in *Proceedings of Atti del 50° Convegno Nazionale AIA* (Taormina, Italy) (2024 May).

[5] J. M. Arend, S. V. A. Garí, C. Schissler, F. Klein, and P. W. Robinson, “Six-Degrees-of-Freedom Parametric Spatial Audio Based on One Monaural Room Impulse Response,” *J.Audio Eng. Soc.*, vol. 69, no. 7/8, pp. 557–575 (2021 Jul./Aug.). https://doi.org/10.17743/jaes.2021.0009.

[6] M. A. Gerzon, “Periphony: With-Height Sound Reproduction,” *J. Audio Eng. Soc.*, vol. 21, no. 1, pp. 2–10 (1973 Feb.).

[7] T. McKenzie, N. Meyer-Kahlen, R. Daugintis, L. McCormack, S. Schlecht, and V. Pulkki, “Perceptually Informed Interpolation and Rendering of Spatial Room Impulse Responses for Room Transitions,” in *Proceedings of the 24th International Congress on Acoustics*, pp. 1–11 (Gyeongiu, Korea) (2022 Oct.).

[8] I. Engel and L. Picinali, “Chapter 5: Reverberation and Its Binaural Reproduction: The Trade-off Between Computational Efficiency and Perceived Quality,” in B. F. G. Katz and P. Majdak Eds., *Advances in Fundamental and Applied Research on Spatial Audio*, (Intechopen, London, UK, 2022). https://doi.org/10.5772/intechopen.101940.

[9] D. Poirier-Quinot, P. Stitt, and B. F. Katz, “RoomZ: Spatial Panning Plugin for Dynamic RIR Convolution Auralisations,” in *Proceedings of the AES International Conference on Spatial and Immersive Audio* (2023 Aug.), paper 19.

[10] W. G. Gardner, “Efficient Convolution Without Input-Output Delay,” *J. Audio Eng. Soc.*, vol. 43, no. 3, pp. 127–136 (1995 Mar.).

[11] G. Garcia, “Optimal Filter Partition for Efficient Convolution With Short Input/Output Delay,” presented at the *113th Convention of the Audio Engineering Society* (2002 Oct.), paper 5660.

[12] D. Poirier-Quinot, B. F. Katz, and M. Noisternig, “EVERTims: Open Source Framework for Real-Time Auralization in Architectural Acoustics and Virtual Reality,” in *Proceedings of the 20th International Conference on Digital Audio Effects (DAFx-17)* , pp. 323–325 (Edinburgh, UK) (2017 Sep.).

[13] L. McCormack and A. Politis, “SPARTA & COMPASS: Real-Time Implementations of Linear and Parametric Spatial Audio Reproduction and Processing Methods,” in *Proceedings of the AES International Conference on Immersive and Interactive Audio* (2019 Mar.), paper 111.

[14] P. Majdak, Y. Iwaya, T. Carpentier et al., “Spatially Oriented Format for Acoustics: A Data Exchange Format Representing Head-Related Transfer Functions,” presented at the *134th Convention of the Audio Engineering Society* (2013 May), paper 8880.

[15] D. T. Lee and B. J. Schachter, “Two Algorithms for Constructing a Delaunay Triangulation,” *Int. J. Comput. Inf. Sci.*, vol. 9, no. 3, pp. 219–242 (1980 Jun.). https://doi.org/10.1007/BF00977785.

[16] J. De Muynke, D. Poirier-Quinot, and B. F. G. Katz, “Effect of Interpolation Artifacts on Perceived Stability of Nearby Sources in a Navigable Reverberant Virtual Environment,” *J. Audio Eng. Soc.*, vol. 72, no. 10, pp. 664–678 (2024 Oct.). https://doi.org/10.17743/jaes.2022.0158.

[17] M. Kronlachner, "Plug-In Suite for Mastering the Production and Playback in Surround Sound and Ambisonics," presented at the *136th AES Convention, Gold-Awarded Contribution to AES Student Design Competition* (2014 Apr.).

[18] F. Adriaensen, "Fast Convolution Engine," https://github.com/brugal/gtrfx/tree/master/jconv. (accessed March 3, 2025).

[19] F. Wefers, *Partitioned Convolution Algorithms for Real-Time Auralization*, Ph.D. thesis, RWTH Aachen University, Aachen, Germany (2014 Sep.).

[20] F. Adriaensen, "Design of a Convolution Engine Optimised for Reverb," in *Proceedings of the 4th International Linux Audio Conference*, pp. 49–53 (Karlsruhe, Germany) (2006 Apr.).

[21] M. Frigo and S. Johnson, "FFTW: An Adaptive Software Architecture for the FFT," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 1381–1384 (Seattle, WA) (1998 May). https://doi.org/10.1109/ICASSP.1998.681704.

[22] D. S. Brungart, A. J. Kordik, and B. D. Simpson, "Effects of Headtracker Latency in Virtual Audio Displays," *J. Audio Eng. Soc.*, vol. 54, no. 1-2, pp. 32–44 (2006 Jan./Feb.).

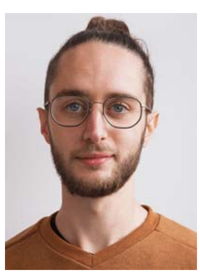
[23] A. Lindau, "The Perception of System Latency in Dynamic Binaural Synthesis," in *Proceedings of the35th German Annual Conference on Acoustics (DAGA)*, pp. 1063–1066 (Rotterdam, the Netherlands) (2009 Mar.).

[24] "Mechanical Vibration and Shock - Evaluation of Human Exposure to Whole-Body Vibration," Standard 2631-1:1997 (1997 May).

[25] L. Lavagna, L. Shtrepi, A. Farina, A. Bevilacqua, and A. Astolfi, "Acoustic Design Optimization Through the Use of Auralization: How Does It Sound Like?" in *Proceedings of the 2nd Symposium: The Acoustics of Ancient Theatres*, pp. 215–217 (Verona, Italy) (2022 Jul.).

[26] N. Meyer-Kahlen, M. Kastemaa, S. J. Schlecht, and T. Lokki, "Measuring Motion-to-Sound Latency in Virtual Acoustic Rendering Systems," *J. Audio Eng. Soc.*, vol. 71, no. 6, pp. 390–398 (2023 Jun.). https://doi.org/10.17743/jaes.2022.0089.

THE AUTHORS



Domenico Stefani

Marco Binelli

Angelo Farina

Luca Turchet

Domenico Stefani is a postdoctoral researcher at the Department of Information Engineering and Computer Science of the University of Trento, Italy. He received an M.S. in computer science from the University of Trento, Italy, and a Ph.D. from the same institution in 2024. His research interests include musical audio processing, music information retrieval, and embedded systems.

•

Marco Binelli received B.S. and M.S. degrees in electronic engineering from the University of Parma, Italy, in 2003 and 2006, respectively, and a Ph.D. degree from the University of Parma, in 2010, with a thesis on analysis and equalization techniques in automotive acoustics. Since 2010, he has been a research fellow with the University of Parma. His research interests include equalization, microphone and loudspeaker arrays, spatial audio psychoacoustics, and active noise control.

•

Angelo Farina received his M.S. in civil engineering (1982) and Ph.D. in technical physics (1987) from the University of Bologna, Italy. He became a full professor in environmental applied physics at the University of Parma in 2005. Author of over 300 scientific articles, his research focused on applied acoustics, concert hall acoustics, simulation software, and advanced measurement systems. He received an Audio Engineering Society fellowship in 2008 for pioneering work on electroacoustic measurements and the AES Silver Medal in 2024 for outstanding contributions to audio engineering and spatial audio.

•

Luca Turchet is an associate professor at the Department of Information Engineering and Computer Science of the University of Trento, Italy. He is the chair of the Institute of Electrical and Electronic Engineers (IEEE) Emerging Technology Initiative on the Internet of Sounds and is the founding president of the Internet of Sounds Research Network.