P. Majdak, F. Zotter, F. Brinkmann, J. De Muynke, M. Mihocic, and M. Noisternig,
"Spatially Oriented Format for Acoustics 2.1: Introduction and Recent Advances"
J. Audio Eng. Soc., vol. 70, no. 7/8, pp. 565–584, (2022 July/August).
DOI: https://doi.org/10.17743/jaes.2022.0026

Spatially Oriented Format for Acoustics 2.1: Introduction and Recent Advances

PIOTR MAJDAK,[1] *AES Member*, **FRANZ ZOTTER,**[2] *AES Associate Member*,
(piotr.majdak@oeaw.ac.at) (zotter@iem.at)
FABIAN BRINKMANN,[3] *AES Associate Member*, **JULIEN DE MUYNKE,**[4, 5] *AES Member*,
(fabian.brinkmann@tu-berlin.de) (julien.demuynke@eurecat.org)
MICHAEL MIHOCIC,[1] *AES Member* **AND MARKUS NOISTERNIG,**[6] *AES Member*
(michael.mihocic@oeaw.ac.at) (markus.noisternig@ircam.fr)

[1]*Acoustics Research Institute, Austrian Academy of Sciences, Vienna, Austria*
[2]*Institute of Electronic Music and Acoustics, University of Music and Performing Arts, Graz, Austria*
[3]*Audio Communication Group, Technical University of Berlin, Germany*
[4]*Eurecat, Centre Tecnológic de Catalunya, Multimedia Technologies Group, Barcelona, Spain*
[5]*Sorbonne Université, CNRS, Institut Jean Le Rond d'Alembert, Paris, France*
[6]*Sciences et Technologies de la Musique et du Son, IRCAM, Sorbonne Université, CNRS, Paris, France*

Spatially oriented acoustic data can range from a simple set of impulse responses, such as head-related transfer functions, to a large set of multiple-input multiple-output spatial room impulse responses obtained in complex measurements with a microphone array excited by a loudspeaker array at various conditions. The spatially oriented format for acoustics (SOFA), which was standardized by *AES Standard 69*, provides a format to store and share such data. SOFA takes into account geometric representations of many acoustic scenarios, data compression, network transfer, and a link to complex room geometries and aims at simplifying the development of interfaces for many programming languages. With the recent advancement of SOFA, the format offers new continuous-direction representation of data by means of spherical harmonics and novel conventions representing many measurement scenarios, such as source directivity and multiple-input multiple-output spatial room impulse responses. This article reviews SOFA by first providing an introduction to SOFA and then describing examples that demonstrate the most recent features of the SOFA 2.1 (*AES Standard 69-2022*).

0 INTRODUCTION

Audio-related data often depends on spatial aspects. A very prominent example is the head-related transfer function (HRTF), which describes the acoustic filtering of a sound source by the listener's body acquired in free field at the listener's ear canal [1]. Usually, binaural sets of HRTFs are acquired for many sound-source positions introducing a strong spatial factor to an HRTF dataset [2]. HRTFs can also be measured in a room, and such data are usually called binaural room impulse responses (BRIRs) [3]. Another prominent example of spatially oriented data is the spatial room impulse response (SRIR), which describes the filtering of a sound source by a room recorded by microphones placed in the room [4]. In many applications, a set of SRIRs is acquired for various source positions and orientations, microphones, and acoustic conditions in a room [5]. A third example of spatially oriented data is the directivity of sound sources, such as loudspeakers or musical instruments [6]. All these and many other spatially oriented data result from measurements or numerical simulations (as summarized in Fig. 1) and thus urge for a file format describing the spatial nature of the data. The spatially oriented format for acoustics (SOFA), described in this article, aims at providing such a format.

Before the availability of SOFA, HRTFs were shared as databases for decades, but each laboratory used its own format. For example, in 1998, the AUDIS project released a compact disk with HRTFs stored in a proprietary format, requiring a binary converter application to access the data [7]. The first widely spread HRTF database with a well-defined format was the dataset of the Massachusetts

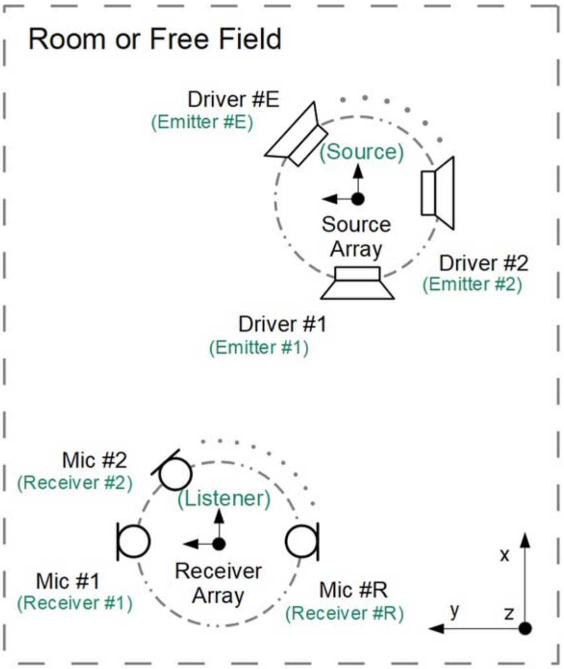


Fig. 1. Typical acoustic setup measuring or simulating spatial aspects. Corresponding names of spatially oriented format for acoustics (SOFA) objects are shown in brackets.

Institute of Technology (MIT) HRTF, which uses the waveform audio format to store the data [8]. Later, many laboratories started to use the MATLAB's file format (MAT) [9] to store HRTFs. For example, the database of the Center for Image Processing and Integrated Computing (CIPIC) [10] provides a file per listener in either plain text or MAT format. The directions are hard coded, i.e., the index of an HRTF corresponds to a predefined direction used in the measurements. The database of HRTFs in the horizontal plane for multiple distances [11] stores the HRTFs in a separate MAT file for each distance. Combined with the necessity to store a separate file for each listener, such a representation results in many files. Other databases, such as the LISTEN database of the Institute for Research and Coordination in Acoustics/Music (IRCAM) [12] and that of the Acoustics Research Institute (ARI) [2], consist of an HRTF matrix and additional matrices describing the spatial direction of the corresponding HRTF, thus allowing representation of HRTFs from any measured direction. Although these databases used the same MAT-file format as the numeric container, the HRTFs were represented using different indexing schemes, requiring a dedicated data parser for each of those HRTF databases.

The database of the Music and Audio Research LaboratoryNew York University (MARL-NYU) [13] was the first attempt to harmonize the various internal representations of CIPIC, LISTEN, MIT, and other common databases. By storing HRTFs from these databases in a single file format, the MARL-NYU database was a big step toward exchangeability of HRTFs across the databases—at the price of its limitation to HRTFs only. Approaching the problem from another perspective, Sound Description Interchange Format [14], a general format for storing audio-related data, has been adapted to HRTFs in order to store HRTFs of a single listener in a mixed text–based and binary representation. Unfortunately, the spatial aspects of HRTFs cannot be represented in Sound Description Interchange Format.

SRIR and BRIR datasets were publicly available before the development of SOFA. They were shared as MAT files or compressed waveform audio format files, for example, the SRIRs of the concert hall in Pori, Finland [15], and the Aachen Impulse Response database [16], respectively. By being available in different file formats and different internal representations, the exchangeability of these datasets was very limited.

This situation triggered the development of SOFA in 2012 by the group Aural Assessment By means of Binaural Algorithms, which is an intellectual group of scientists collaborating on the development and applications of models of human spatial hearing [17, 18]. The idea was quickly taken up by further European partners and the Audio Engineering Society (AES). The goal was to develop a file format capable of describing any existing and future spatially oriented data of acoustic systems. The following requirements were defined:

- The ability to describe an acoustic setup with arbitrary geometry, i.e., no limitation to special cases like a regular grid or constant distance;
- A consistent definition of a container with self-describing data, i.e., all the required information about the acoustic setup must be provided as metadata in the file;
- Flexibility to describe data of multiple conditions (listeners, rooms, spatial positions, etc.) in a single file;
- Available as a binary file with data compression for efficient storage and transfer;
- Support for network transfer, structured file hierarchy, and partial file access;
- Predefined description conventions for existing common acoustic setups (such as HRTFs and SRIRs) but open for any future setups without the need to re-define the format.

As a result, in 2015, SOFA 1.0 was defined [19] and approved by the AES as *AES Standard 69-2015* [20]. Since then, many institutions started to use SOFA to store HRTFs, SRIRs, BRIRs, and other spatial data, see for example [21] and APPENDIX A. SOFA underwent a major upgrade that is available as the updated *AES Standard 69-2020* [22], referred to as SOFA 2.0, and a minor update that is available as *AES Standard 69-2022*, referred to as SOFA 2.1.

This article reviews SOFA as a format to store spatially oriented acoustic data. SOFA is first introduced by defining SOFA objects; describing the numeric container, file structure, and geometric relation between the objects; and introducing the so-called SOFA conventions. Furthermore, examples are provided for the most recent features of SOFA 2.1, such as the continuous-direction representation of data, description of directivity of musical instruments and loudspeakers, and representation of multiple-input multiple-output (MIMO) data such as BRIRs or SRIRs. Finally, APPENDIX A.1 extends the article with a list of currently available SOFA materials, such as toolboxes, applications, and datasets.

1 BASICS OF SOFA

1.1 General

A typical acoustic measurement involves various objects representing acoustic sources, such as loudspeakers, and receivers, such as microphones (see Fig. 1). Often, sources and receivers are grouped to larger structures, such as loudspeaker arrays and microphone arrays. A dummy-head microphone, for example, can be seen as a group of two receivers placed at the entrance of the ear canals. In SOFA, these objects, which are essential to represent a general measurement situation, are defined. In this article, these SOFA objects are denoted as *Uppercase and italic*.

The object *Receiver* is defined as the single acoustic sensor, such as the microphone. The number of *Receivers* is not limited in SOFA, and multiple *Receivers* are jointly represented by the object *Listener* that incorporates all the *Receivers*. For HRTFs, *Listener* can be a human or artificial head. For SRIRs, *Listener* can be a microphone-array structure in the form of a sphere and a frame in the case of spherical and planar microphone arrays, respectively. Whereas the number of *Receivers* is not limited, there is only a single *Listener* object in SOFA. The definition of the *Listener* as a single logical object is important because in measurements, the spatial configuration of the *Listener* (such as its position and/or its orientation) usually varies without substantial changes in the relative configuration between the *Receivers*. For example, in BRIR measurements done for multiple listener positions in a room, the spatial relation of the head with respect to the room changes, but the relation between the head and microphones does not.

The object *Emitter* is defined as any acoustic excitation used in the measurement. The number of *Emitters* is not limited in SOFA, and multiple *Emitters* are jointly represented by the object *Source* that incorporates all *Emitters*. *Source* can be a multi-driver loudspeaker (with the particular drivers as *Emitters*), speaker array (with the particular speakers as *Emitters*), choir (with the particular human singers as *Emitters*), etc. Whereas the number of *Emitters* is not limited, only a single *Source* is used in SOFA.

Finally, the object *Room* is defined as a volume surrounding the measurement setup. *Room* can be any type of geometry including the special case of having no room at all, i.e., free-field conditions.

These five objects (*Emitter*, *Source*, *Receiver*, *Listener*, and *Room*) are the backbone of each SOFA file, see Fig. 1. They are described by their geometric, acoustic, and other properties. Furthermore, optional user-defined objects can be included in order to store any parameter relevant to the measurement, e.g., the torso-head relation in measurements with variable torso-head angle, temperature of the loudspeaker driver, tuning of the musical instrument, or even links to photographs of the acoustic scene to graphically represent the measurement setting.

1.2 Numeric Container

SOFA stores the information in a single file by serializing the data into a binary stream. For the serialization, SOFA relies on the numeric container developed by Unidata and called Network Common Data Form (NetCDF) [23, 24], which is a set of software libraries and data formats supporting the creation, access, and sharing of scientific data. It offers a structured representation of multidimensional data and metadata; is self-describing, network-transparent, and machine-independent; and supports huge files, partial access within a file, and data compression. The format NetCDF further relies on a more basic numeric container, the HDF5 [25]. The specifications of both formats are open access, are freely available, and include a complete definition and examples of various implementations. Pre-compiled libraries are available for programming languages C++, JAVA, and Fortran. Application-programming interfaces are available for many high-level programming languages, such as Ruby, Perl, and Python, and for scientific environments, such as MATLAB, Octave, and R. Thus, it is not surprising that NetCDF is an established format widely used across hundreds of research organizations in the field of climatology, meteorology, oceanography, geographic information, and numerical computation. For example, MATLAB's MAT files are actually NetCDF files [26].

By defining SOFA on top of NetCDF, custom development of a numeric container was avoided. In order to distinguish SOFA files from the general NetCDF files, SOFA files have the extension ".sofa."

Note that NetCDF comes in many variants: the classic format, 64-bit offset format, 64-bit data format, NetCDF-4 format, and NetCDF-4 classic model format. Historically, the classic format was the only format of NetCDF, developed between 1989 and 2004, and referred to as NetCDF-1. By 2008, further variants were developed, yielding to the introduction of NetCDF-4, which provides many additional features, some of them at the price of format complexity. SOFA uses the *NetCDF-4 classic model* format. By sticking to the classic model, data representation in SOFA remains simple. By using NetCDF-4, there is still access to the extended functionality of NetCDF-4, such as string type, parallel input/output, ample variable sizes, or Unicode names. From the SOFA user's perspective, when using the NetCDF-4 library for handling SOFA files, deeper knowledge of the NetCDF-format details is actually not required (more interested readers are referred to the NetCDF-4 User's Manual). For simplicity, this article refers to SOFA's numeric container as NetCDF.

1.3 File Structure

Fig. 2 shows the structure of a SOFA file. Following the NetCDF terminology, SOFA stores all data in variables and attributes. Variables are numeric or character matrices. Their size is defined by dimensions defined in a SOFA file. Variables can be further accompanied by attributes, which

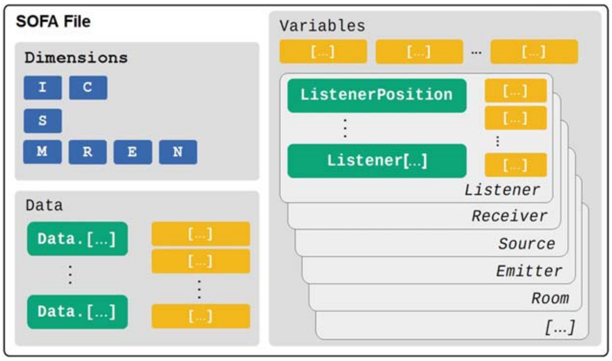


Fig. 2. Elements of a spatially oriented format for acoustics (SOFA) file structured by the Network Common Data Form (NetCDF) and SOFA terminology including SOFA *objects*. Blue: SOFA **`Dimensions`** stored as NetCDF dimensions. Green: NetCDF `variables`. Orange: NetCDF `attributes`, either linked to a NetCDF variable or stored as NetCDF global attributes. [. . .] = Placeholder for the actual strings.

are narrative descriptions belonging to a specific variable. Attributes provided at the global level of the file are called "global attributes" and provide narrative descriptions of the whole SOFA file. Note that although attributes are technically strings, in SOFA, they are not variables of the type string. In this article, SOFA variables and attributes are denoted as `monospaced words`.

In the SOFA terminology, differences between data and metadata are distinguished. Data is the acoustic information to be stored, i.e., information resulting from the measurement, such as impulse responses, transfer functions, etc. Metadata is the information required to interpret the acoustic data and can be, e.g., the spatial configuration of SOFA objects or other relevant information about the measurement. Both metadata and data are described by means of NetCDF variables and attributes.

The format NetCDF explicitly considers conventions, which are defined as sets of recommendations in a community on the naming and number of dimensions, variables, and attributes within a NetCDF file. Many conventions exist, mostly in the field of climate and geographical research. In SOFA, NetCDF conventions are defined specifically describing acoustical measurement setups, i.e., the so-called SOFA conventions, which are sets of recommendations on the number and naming of variables; their attributes and dimensions; and rules specific to commonly used acoustic setups are defined.

1.4 General Syntax

The names of the attributes and variables have to begin with a letter followed by letters or digits, with all names being case sensitive and using UTF-8 character coding. Underscores ("_") and the names "API," "GLOBAL," and "PRIVATE" are not allowed because they are reserved for internal purposes. The content has to be encoded by the UTF-8 character coding because it is backward compatible with ASCII and avoids complications of endianness and byte-order marks found in UTF-16 and UTF-32.

Grouping of metadata is done by using prefixes. For example, all metadata describing the object *Listener* have the prefix `Listener`. Accordingly, all the metadata concerning a change in the measurement setup have the prefix `Measurement`.

All variables are of the NetCDF type "double" or "character." Specifically, double is used to represent all numeric values in SOFA. Whereas NetCDF also supports other types of numeric representations, the restriction to double in SOFA simplifies the format. The disadvantage of having 8 bytes to represent each numeric value is negligibly small considering the compression available in NetCDF. The NetCDF type "character" is used to represent strings in variables.

SOFA uses the format *ISO Standard 8601-1:2019* [27] with the specification "yyyy-mm-dd HH:MM:SS" for all the information about the date and time stored as an attribute. When stored as a numeric variable, SOFA uses the number of seconds from 1970-01-01 00:00:00.

For all units, SOFA uses the International System of Units [28].

1.5 Dimensions

All NetCDF variables are organized as matrices of predefined dimensions. In SOFA, dimensions are single letters, and in this article, they are represented in **serif bold font**.

SOFA defines the dimensions **I** and **C** to represent the identity and spatial coordinates, respectively. Consequently, **I** = 1, and **C** = 3, always.

The dimension **S** is used to store matrices with alphanumeric data, i.e., strings. Consequently, **S** stores the size of the largest string. Note that SOFA represents **S** as an unlimited dimension in NetCDF, i.e., it can be extended easily without the need of internally reformatting the file.

Further dimensions are SOFA-object specific. The dimensions **R** and **E** represent the number of *Receivers* and *Emitters*, respectively, in the discrete-direction representation, or number of coefficients related to the *Receivers* and *Emitters*, respectively, in a continuous-direction representation (see Sec. 1.8 for more information on the geometric representation of objects).

The dimension **N** represents the number of data samples in a single measurement. In the time domain, **N** is the number of samples. In the frequency domain, **N** is the number of frequency bins. **N** is global for all data in a SOFA file, i.e., all impulse responses or spectra in a SOFA file have the same number of samples or frequency bins.

The dimension **M** represents the number of measurements, which can be the number of measured IRs or spectra. Usually a single measurement is given by a unique spatial configuration in the measurement. A change in that configuration, e.g., a change of the *Listener* position, can be reflected by a separate measurement, increasing **M**. Naturally, **R**, **E**, **N**, and **M** are positive non-zero integers.

1.6 Data

The acoustic information, i.e., data, is represented by a group of NetCDF variables with the prefix `Data.` (note the dot[1] at the end). The specific variable names depend on the data type used. In SOFA, three fundamental data types, which describe the data as a 3D matrix along the dimensions **M**, **R**, and **N**, are defined.

The data type "FIR" describes the data in the time domain, which is usually a set of finite IRs (FIRs) [29]. It uses `Data.IR`, `Data.Delay`, and `Data.SamplingRate` as variables. **N** defines the length of the IR in the units of the sampling interval in the time domain, i.e., the inverse of `Data.SamplingRate`. The variable `Data.Delay` allows the user to explicitly state an additional broadband delay for each of the IRs.

The data type "TF" describes the data in the frequency domain as a set of complex-valued spectra. It uses `Data.Real` and `Data.Imag` to store the real and imaginary parts of the spectra, respectively. **N** refers to the number of stored frequency bins in `Data.Real` and `Data.Imag`, in which particular frequencies can be arbitrary and are stored in the variable `N` (not to be confused with the dimension **N**). The variable `N` and its attributes provide the information about the frequency bins along the dimension **N**.

The data type "SOS" describes the data as a chain of second-order section (SOS) filters [30]. Such filters are also known as biquadratic filters widely used as shelving and peak audio filters [31]. SOS filters offer better computational efficiency because they represent the data as infinite IR (IIR) filters, usually requiring fewer calculations per audio sample [29]. Such an SOS filter is a chain of p SOS sections, each of which digital transfer function (in the z-domain) writes as

$$H(z) = \frac{B_1(z)}{A_1(z)} \cdot \frac{B_2(z)}{A_2(z)} \cdot \ldots \cdot \frac{B_p(z)}{A_p(z)}, \tag{1}$$

with $A_i(z) = a_{i,0} + a_{i,1}z^{-1} + a_{i,2}z^{-2}$ (usually normalized such that $a_{i,0} = 1$) and $B_i(z) = b_{i,0} + b_{i,1}z^{-1} + b_{i,2}z^{-2}$. The data type SOS was introduced with SOFA 2.0 (see SEC. 2.2 for an example) and uses `Data.SOS` to represent the filter coefficients along the dimension **N** ordered by

$$(b_{0,0}\ b_{0,1}\ b_{0,2}\ a_{0,0}\ a_{0,1}\ a_{0,2},\ b_{1,0}\ b_{1,1}\ \ldots). \tag{2}$$

The dimension **N** stores the total number of coefficients and is $6p$, i.e., 6-fold the total number of sections. Furthermore, `Data.SOS` is complemented by `Data.Delay` (in samples) and `Data.SamplingRate` (in Hertz).

These fundamental data types can be extended to more complex matrices. For example, the data type "FIR-E" extends FIR by the dimension **E** to a 4D matrix depending on **M**, **R**, **N**, and **E**.

1.7 Metadata

1.7.1 General

General metadata describe the general content of the SOFA file and are stored as NetCDF global attributes. The following global attributes are mandatory:

- `Conventions`: Always "SOFA," and labels the NetCDF files a SOFA file;
- `Version`: Version of SOFA, with "2.1" as the current version at the time of writing;
- `SOFAConventions`: Name of the SOFA convention used for the file (see SEC. 1.10);
- `SOFAConventionsVersion`: Version of the SOFA convention;
- `DataType`: Defines the data type used;
- `RoomType`: Defines the type of the room used;
- `Title`: A succinct description of the file's content;
- `DateCreated`: Date and time of file creation;
- `DateModified`: Date and time of file modification;
- `APIName`: Name of the API that created and/or edited the file;
- `APIVersion`: Version of the API;
- `AuthorContact`: Contact information, e.g., email address, of the author of the file;
- `Organization`: Legal name of the organization behind the file; and
- `License`: Legal license under which the file is provided.

There are more predefined optional global attributes, such as `Comment`, `History`, `References`, and `Origin`, that help to provide additional, optional information in a standardized way. Furthermore, any type of user-defined global attributes can be used for the user's convenience.

1.7.2 Object-Related Metadata

SOFA defines the metadata that describe the SOFA objects. These metadata consist of variables with the corresponding object's name as the prefix.

The metadata of *Listener*, *Receiver*, *Source*, and *Emitter* have similar definitions. They are described here with *Listener* as an example. The variable `ListenerPosition` defines the spatial position of the *Listener*. It is further described by the attributes `Type` and `Units`, stating the type of the coordinate system and units, respectively. Note that `ListenerPosition` can be a column vector of **C** elements for a stationary *Listener* or matrix of **M**-by-**C** for *Listener* with varying position. The variables `ListenerView` and `ListenerUp` optionally describe the spatial orientation of *Listener* using the View-Up vector pairs usually used in computer graphics to describe the gaze of a camera [32]. Both vectors are optional, but if `ListenerView` is provided, `ListenerUp` must be provided as well. The absence of both leaves the orientation in an undefined state, indicating an omnidirectional *Listener*. `ListenerView` must be accompanied by the attributes `Type` and `Units`, similarly to those used for `ListenerPosition`. A nar-

---

[1] Depending on the software package, this dot might be internally displayed as another character.

rative description and short name of the *Listener* can be provided in the optional global attributes `ListenerDescription` and `ListenerShortName`, respectively. Note that `ListenerView` and `ListenerUp` can be two column vectors of **`C`** elements for a *Listener* with a stationary orientation or two matrices of **`M`**-by-**`C`** for *Listener* with varying orientation.

All of this applies similarly to the objects *Receiver*, *Source*, and *Emitter*. *Room* has a different definition, and four room types are distinguished, described by the global attribute `RoomType`. The room type "free field" assumes an anechoic measurement condition and does not require any further description. The room type "reverberant" is used to describe some type of reverberation with a narrative description only. This can be provided in the global attribute `RoomDescription`. The room type "shoebox" is used to describe a room approximated by a rectangular cuboid defined by the coordinates of opposite corners. This information is provided in the variables `RoomCornerA` and `RoomCornerB`, accompanied by an arbitrary variable `RoomCorners` that has the attributes `Type` and `Units`, describing the type of the coordinate system and units used in the variables `RoomCornerA` and `RoomCornerB`. The room type "dae" is used to describe a room with a more complex geometry that is stored as a digital asset (DAE) file using the COLLAborative Design Activity format standardized as *ISO/PAS Standard 17506:2012* [33]. The link to that file is provided as a uniform resource identifiers (URIs) in the global attribute `RoomGeometry`. The URI syntax corresponds to RFC 3986 [34]. Note that `RoomShortName`, `RoomDescription`, and `RoomLocation` are optional global attributes, which can be used to give the room a short name, provide a narrative description of the room, and indicate the geographic location of the room, respectively, for any room type used.

1.8 Geometric Relation Between the Objects

SOFA uses the Cartesian and spherical coordinate systems to represent the geometrical relation between the objects. In the Cartesian system as used in SOFA, each point is described by three orthogonal linear values (x, y, z) as defined in *ISO Standard 80000-2:2019* [35]. In the spherical coordinate system as defined in *ISO Standard 80000-2:2019* and used by MPEG-H [36], each point is described by azimuth angle, polar angle, and radius. In SOFA, the polar angle is replaced by the elevation angle, which is measured from the horizontal x-y reference plane such that the elevation angle of zero is at the horizon. The azimuth angle still increases counter-clockwise, which is not to be confused with the opposite directions used by other coordinate systems, such as geographical, celestial, horizontal, or astronomical. Thus, when using the spherical coordinate system in SOFA, each point is described by azimuth angle, elevation angle, and radius (ϕ, θ, r) corresponding to

$$\phi = \arctan\frac{y}{x},\ \theta = \arctan\frac{z}{\sqrt{x^2+y^2}},\ r = \sqrt{x^2+y^2+z^2}. \tag{3}$$

The objects *Listener*, *Source*, and *Room* share the same global coordinate system. *Receivers* are defined in the local coordinate system of the *Listener*, i.e., the origin of the *Receivers*' local coordinate system is given relative to the *Listener*'s global coordinate system. Similar requirements apply to *Emitters* and the *Source*.

The orientation of the objects can be defined by the rotation of the respective coordinate systems described by two orthogonal vectors, the View and Up vectors. The View vector defines the direction of the positive x-axis of the respective coordinate system. The Up vector defines the direction of the positive z-axis of the respective coordinate system. Both View and Up vectors share the same type of coordinate system. When described in spherical coordinates, the View vector describes the azimuth and elevation angles of the object's orientation, and the Up vector describes the roll (as in the three rotational axes: pitch, yaw, and roll).

Note that for omnidirectional objects, information about the orientation has no meaning, and thus View and Up vectors are optional. However, Up needs to be specified when View is provided.

Furthermore, the geometric relation between multiple *Emitters* (and between multiple *Receivers*) can be transformed into a *continuous-direction* representation, in which the discrete-direction set of all *Emitters* (and all *Receivers*) are used to parameterize a finite series of spherical-harmonics (SH) coefficients $S_{l,m}(\phi, \theta)$ [37] corresponding to

$$S_{l,m}(\phi,\theta) = \begin{cases} \sqrt{2}N_{l,|m|}P_{l,|m|}(\sin\theta)\cos(|m|\,\phi), & m > 0 \\ N_{l,0}\ \ P_{l,0}\ \ (\sin\theta), & m = 0 \\ \sqrt{2}N_{l,|m|}P_{l,|m|}(\sin\theta)\sin(|m|\,\phi), & m < 0 \end{cases}, \tag{4}$$

with the order $l \in \{0, 1, \ldots, L\}$ and L, which define the maximal order of the series, and the degree $m \in \{-l, \ldots, 0, \ldots, l\}$. This representation is similar to Ambisonics widely used in the audio applications, e.g., [38, 39]. Note that the terms "degree" and "order" correspond to the common usage in the field of audio engineering, and their definitions may vary in other fields. For example, in quantum physics, the meaning of degree and order is the other way around. In order to calculate the SH coefficients, the Legendre-associated polynomials are defined by

$$P_{l,m}(x) = \frac{1}{2^l l!}\left(1-x^2\right)^{m/2}\frac{d^{l+m}}{dx^{l+m}}\left(x^2-1\right)^l, \tag{5}$$

and the normalization factor used is defined by

$$N_{l,m} = \sqrt{(2l+1)\cdot\frac{(l-m)!}{(l+m)!}}. \tag{6}$$

This normalization, often called "4π-normalized" [40], is widely used in the area of geodesy and spectral analysis. In the field of audio engineering, it is known as "N3D" and used in the Audio Definition Model (ADM) [41] and MPEG-H framework [36, 42].

Fig. 3 shows the visual representations of the first 16 real-valued SH functions according to Eqs. (4)–(6), vertically

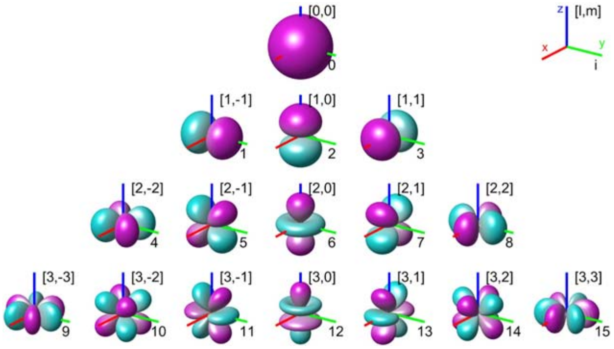


Fig. 3. Representation of the first 16 real-valued spherical harmonics according to Eq. (4) (magenta denotes positive and cyan denotes negative amplitudes). Columns: Degree m. Rows: Order l up to the order $L = 3$. The local coordinate system of each representation is shown in the inset. The numbering in the brackets shows $[l, m]$. The numbering in the subscript shows the index i of the harmonic in their linear order according to Eq. (7).

ordered by order l up to the order $L = 3$ and horizontally ordered by the degree m. Note that L as the maximal order of the SH series and defines the directional accuracy of the representation.

In SOFA, the SH coefficients are stored in a linear arrangement given by l and m such that the ith data element corresponds to

$$i = l(l+1) + m, \tag{7}$$

which is the same as the Ambisonic Channel Number used in audio formats storing SH coefficients, such as the ADM [41] and AmbiX [43].

For a given maximal order L, there are

$$\#i = (L+1)^2 \tag{8}$$

coefficients to be stored. As an example, Fig. 3 shows all $\#i = 16$ harmonics possible in a representation with the maximal order of $L = 3$.

1.9 Relation Between the Files

It might be necessary to store multiple different representations of the same acoustic data or store results of processing an existing acoustic data. For example, HRTFs of a listener can be processed by headphone filters, with both data provided in separate SOFA files, yielding a third file as the result.

In SOFA, a link to other SOFA files can be provided in order to create a structured hierarchy of data linking original data (i.e., parent) with processed data. Such a dependence in a hierarchical structure can be represented by the optional global attribute `Parents`, which can contain a list of URIs pointing to the files containing the original data. In the case of multiple entries, each URI is separated by the end-of-line token (hexadecimal $0A_{16}$).

As an example of such a hierarchy, consider a file "HRTF" storing the HRTFs of a listener and a file "EQ" storing the equalization filters of a given headphone. In order to create equalized HRTFs, the files HRTF and EQ are used to create file "EQUED." Furthermore, consider a file "ROOM" containing room impulse responses, which is then used together with the equalized HRTFs to create auralization filters stored in the file "AURA." HRTF and EQ are parents of EQUED, that is, together with ROOM the parents of AURA. Note that when creating the file EQUED, files HRTF and EQ are not modified—they require read access only.

1.10 SOFA Conventions

A SOFA convention is a set of rules for naming mandatory metadata, their dimensions and default values as well as of suggestions for naming optional metadata. Conventions are developed to meet the requirements of particular application fields. They comply with and extend the general SOFA specifications described in the previous section. In order to prevent predicting the future, conventions are developed only for known measurement setups and existing datasets. In the standard, conventions are described in an Annex, enabling a quicker reaction to new developments without the necessity to revise the main SOFA standard. In this article, conventions are denoted in bold.

As of SOFA 2.1, the following conventions are standardized:

- **General**: For a general representation of data as any type, storing data with only the required general metadata being pre-defined. This is the most general SOFA convention.
- **GeneralFIR**: For a general representation of data as FIR filters, storing 3D sets of IRs depending on the measurement, *Receiver*, and time.
- **GeneralTF**: For a general representation of data as transfer functions, storing 3D sets of complex spectra depending on the measurement, *Receiver*, and frequency.
- **GeneralSOS**: For a general representation of data as SOS filters, storing 3D sets of biquad filters depending on the measurement, *Receiver*, and SOS index.
- **GeneralFIR-E**: For a general representation of data as FIR filters, storing 4D sets of IRs depending on the measurement, *Receiver*, time, and *Emitter*.
- **GeneralTF-E**: For a general representation of data as transfer functions, storing 4D sets of complex spectra depending on the measurement, *Receiver*, frequency, and *Emitter*.
- **SimpleFreeFieldHRIR**: For representing discrete-direction free-field head-related impulse responses (HRIRs) of a single subject stored as sets of IRs. Each HRIR position is represented by a unique representation of the *Source*.
- **SimpleFreeFieldHRTF**: For representing discrete-direction free-field HRTFs of a single subject stored as sets of complex spectra. Each HRTF position is represented by a unique representation of the *Source*.
- **SimpleFreeFieldHRSOS**: For representing discrete-direction free-field HRTFs of a single subject stored as sets of SOSs. Each HRTF position

is represented by a unique representation of the *Source*.

- **FreeFieldHRIR**: Extension of the convention **SimpleFreeFieldHRIR** to continuous-direction representations. The HRIRs are stored as 4D sets of IRs depending on measurement, *Receiver*, time, and *Emitter*. In contrast to **SimpleFreeFieldHRIR**, each HRIR position is represented by a unique representation of the *Emitter* (see SEC. 2.1 for an example).
- **FreeFieldHRTF**: Extension of the convention **SimpleFreeFieldHRTF** to continuous-direction representations. Other aspects as in **FreeFieldHRIR**.
- **SimpleHeadphoneIR**: For storing IRs with a one-to-one correspondence between *Emitter* and *Receiver*, with the main application being to store headphone impulse responses [headphone IRs (HpIRs), see SEC. 2.5 for an example].
- **SingleRoomSRIR**: For representing SRIRs as FIR filters measured in a single room with a single *Source* (e.g., a loudspeaker) and *Listener* containing an arbitrary number of (per default omnidirectional) *Receivers* (e.g., a microphone array), with both the *Source* and *Listener* potentially varying (see SEC. 2.3 for an example).
- **SingleRoomMIMOSRIR**: For representing SRIRs as FIR filters measured in a single room with a single *Source* containing an arbitrary number of (per default omnidirectional) *Emitters* (e.g., a loudspeaker array) and *Listener* containing an arbitrary number of (per default omnidirectional) *Receivers* (e.g., a microphone array), with both the *Source* and *Listener* potentially varying (e.g., SEC. 2.3).
- **FreeFieldDirectivityTF**: For storing directivities as complex spectra of a single *Source*, such as instruments, loudspeakers, singers, or talkers, obtained by multiple *Receivers* from potentially multiple measurements, such as various musical notes or playing various styles (see SEC. 2.4 for an example).

Within each convention, application-specific metadata can be provided by the user. They can be of arbitrary dimension, but they must not use metadata names from the respective convention set. In the case of extending an already specified attribute (e.g., to provide a more-specific description along a specific dimension), a plural version of the attribute's name is required, and both the attribute and variable are provided. For example, the attribute `ReceiverDescription` can be extended as an **`M`**-dependent variable called `ReceiverDescriptions`. Furthermore, on top of the standardized conventions, users can propose and discuss new conventions by contributing to the SOFA website [44].

2 EXAMPLES OF ADVANCES IN SOFA 2.1

This section describes features of SOFA as of version 2.1 in more detail.

2.1 Continuous-Direction Representations of HRTFs

SOFA 2.0 introduced the continuous-direction representation of the data that can be used to represent HRTFs without any spatial discretization of the sound-source direction. To this end, conventions such as **FreeFieldHRTF** and **FreeFieldHRIR** are provided to store HRTFs in both discrete and continuous ways. The continuous-direction representation can be used to easily interpolate between HRTF directions.

As an example, the conversion from discrete directions in **SimpleFreeFieldHRIR** (as an established convention in SOFA 1.0) to continuous directivity in **FreeFieldHRTF** (introduced with SOFA 2.0) is explained. Then, the capabilities of directionally continuous representations are demonstrated by spatially sampling the continuous HRTF representation to new directions in a dense discrete grid.

The most widely used convention to represent HRTFs, **SimpleFreeFieldHRIR**, stores HRTFs measured at discrete directions in space. The left column of Fig. 4 shows the left-ear HRTFs from `HRIR_L2354.sofa` of the THK database [45]. The top and bottom rows show the logarithmic IRs in the horizontal plane and logarithmic amplitude spectra in the median plane, respectively. Recall that in **SimpleFreeFieldHRIR**, each measurement corresponds to a measured sound-source direction, thus **`M`** represents the number of measured directions (represented as **d** in Table 1), `SourcePosition` has the dimension **`M`**-by-**`C`**, and the data matrix has the dimension **`M`**-by-**`R`**-by-**`N`** (with **`N`** being the number of filter taps and usually with **`R`**=2), The two most-left columns of Table 1 show the corresponding summary.

In the first step, the data is transformed to a spectral representation by means of Fourier transform applied on the data matrix, for each **`M`** independently. The real and imaginary parts of each spectrum are stored in `Data.Real` and `Data.Imag`, respectively. Furthermore, the convention changes to **SimpleFreeFieldHRTF**, and **`N`** becomes frequency. To this end, the variable `N` is created as a vector of **`N`** (note the difference between **`N`** and `N`) and filled with a series of frequencies beginning with 0 and ending at half the `Data.SamplingRate`.

In order to represent these HRTFs in a continuous way, they need to be converted to **FreeFieldHRTF** but still use the discrete-direction representation. To this end, each source position of **SimpleFreeFieldHRTF** is treated as a separate *Emitter* of a virtual loudspeaker array centered at the position of the *Listener*. The data matrices `Data.Real` and `Data.Imag` are extended by the dimension of *Emitter*, i.e., they have the dimension **`M`**-by-**`R`**-by-**`N`**-by-**`E`**, with **`E`** being the number of measured positions and **`M`** being reduced to 1. The content of `SourcePosition` is copied to `EmitterPosition`, which becomes an **`E`**-by-**`C`** matrix. Then, `SourcePosition`, being an **`M`**-by-**`C`** matrix, is reduced to a 1-by-**`C`** matrix representing a single position of the virtual loudspeaker array located in the center of the *Listener* (compare Table 1, the center column). Note that this conversion does not change the data content.

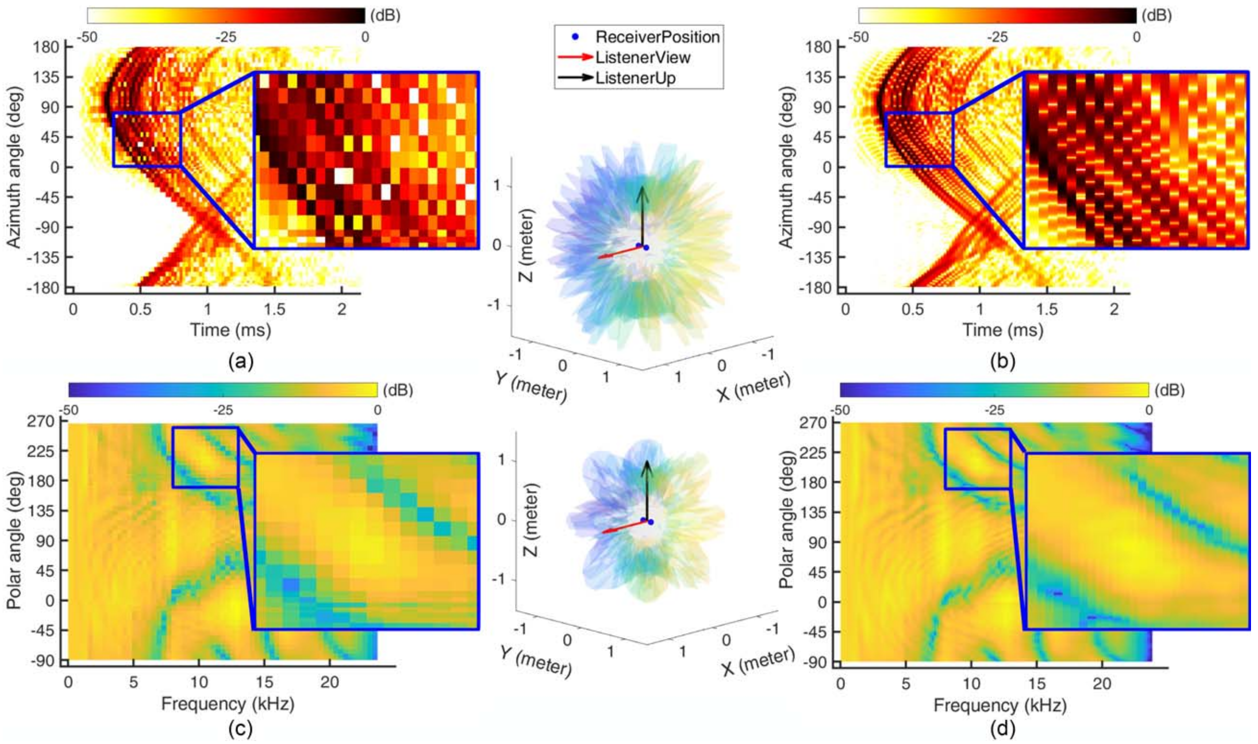


Fig. 4. Spatially continuous representation in **FreeFieldHRTF** demonstrated by spatial interpolation. Left: Spatially discrete head-related transfer functions (HRTFs) represented in **SimpleFreeFieldHRIR** at an average resolution of 8° shown as energy time curves along the horizontal plane (a) and as amplitude spectra along the median plane (c). Center: Spatially continuous HRTFs (see text) represented in **FreeFieldHRTF**, obtained by converting **SimpleFreeFieldHRIR** to the spherical-harmonics (SH) domain and shown as an example for orders 40 (top) and 4 (bottom). Right: Spatially discrete HRTFs represented in **SimpleFreeFieldHRIR**, obtained by spatial sampling **FreeFieldHRTF** at a resolution of 1°. The energy time curve is shown in (b) and the amplitude spectra along the median plane in (d).

Having those preparations done, the discrete-direction *Emitter* positions are used to transform the data to the continuous-direction representation. To this end, for a given order L, SH coefficients are calculated by applying Eqs. (4) and (5) on all *Emitter* positions, forming the SH matrix. Then, for each ear (i.e., **`R`**) and frequency (i.e., **`N`**), the continuous-direction data is calculated as the product of the inverse SH matrix and discrete-direction data for all emitters. This is done separately for `Data.Real` and `Data.Imag`. The data matrices become the size of 1-by-**`R`**-by-**`N`**-by-$\#i$, with $\#i$ as in Eq. (8). Finally, the type of the coordinate system for the *Emitters* is adapted (i.e., the `Type` attribute of `EmitterPosition` is "spherical harmonics") and `EmitterPosition` becomes a 1-by-**`C`** matrix, with no actual meaning, e.g., set to the center position of the *Listener* (compare Table 1, most-right column). Within the example, the center of Fig. 4 shows as an example SH coefficients for two SH orders.

With that representation, HRTFs can be directionally post-hoc sampled for an arbitrary direction. Within the example, the right column of Fig. 4 shows the left-ear continuous HRTFs, sampled at a resolution of 1°. The finer resolution is visualized in the insets. These sampled HRTFs can be stored directly in **FreeFieldHRTF** or converted to

Table 1. Relation between **SimpleFreeField** and **FreeField**.

	SimpleFreeField HRIR/HRTF	**FreeField HRIR/HRTF**	
	Discrete	**Discrete**	**Continuous**
`M`	**d**	1	1
`E`	1	**d**	**#i**
`SourcePosition`	**d** × 3	1 × 3	1 × 3
`EmitterPosition`	**1** × 3	**d** × 3	**1** × 3
Data matrix	**d** × **`R`** × **`N`**	**1** × **`R`** × **`N`** × **d**	**1** × **`R`** × **`N`** × #**i**

Note. **d** = number of measured HRTF directions; HRIR = head-related impulse response; HRTF = head-related transfer function; **#i** = number of SH coefficients, depending on L [see Eq. (8)]; SH = spherical harmonics. Significant changes are shown in bold.

SimpleFreeFieldHRTF for legacy with SOFA 1.0. This example and parts of Fig. 4 can be reconstructed with the file `demo_FreeFieldHRTF.m` from the SOFA Toolbox 2.0 [46].

2.2 HRTFs Represented by SOS Filters

Binaural audio rendering systems mostly often require computationally efficient models of HRTFs. IIR filters provide the opportunity to approximate the magnitude of HRTFs with fewer coefficients than direct FIR representations [47–49]. Although direct forms of IIR filters are simple to construct and have small computational requirements, they are very sensitive to parameter variations. A simple approach to improve their numerical stability is to represent them as a series of low-order filters, e.g., [50, 51]. Especially the series of SOSs is widely used in audio engineering as biquadratic filters, e.g., [52], and has found application in the field of binaural rendering via HRTFs [53, 54].

An example is the conversion of HRIRs stored in **SimpleFreeFieldHRIR** to a cascade of SOS filters stored in **SimpleFreeFieldHRSOS** introduced in SOFA 2.0. The modeling of HRTFs is typically split into an all-pass stage, which models the broadband interaural differences, and minimum-phase filter. The minimum-phase filter is then designed as a chain of SOS. The total filter design process is as follows:

- Computation of the minimum-phase spectrum;
- Flattening of the lower-frequency and higher-frequency regions;
- Critical-band smoothing;
- Frequency warping;
- Conversion back to the time domain;
- IIR modeling using a time-domain algorithm for least-squares approximation of desired HRIRs for IIR digital filters [55] by minimizing the error between the targeted and modeled HRIRs;
- Conversion from IIR representation to a cascade of SOSs, e.g., [56]; and
- Frequency unwarping;

In **SimpleFreeFieldHRSOS**, the obtained coefficients are stored in `Data.SOS` ordered following Eq. (2). Fig. 5 shows as an example the amplitude spectra of an HRTF modeled with a cascade of 6, 12, and 18 sections. The example is based on a free-field HRIR from the BiLi database [57] of the human subject “1100” [58]. For the reference, the solid line shows the amplitude spectrum of the underlying original HRTF [59].

2.3 Multi-Perspective SRIRs

SRIRs aim at capturing the spatial characteristics of the acoustics of a room through a number of RIRs measured at various positions and/or orientations of *Source* and *Listener*. Both, *Source* and *Listener* can be compact arrays consisting of multiple acoustic channels, i.e., transducers and microphones, respectively.

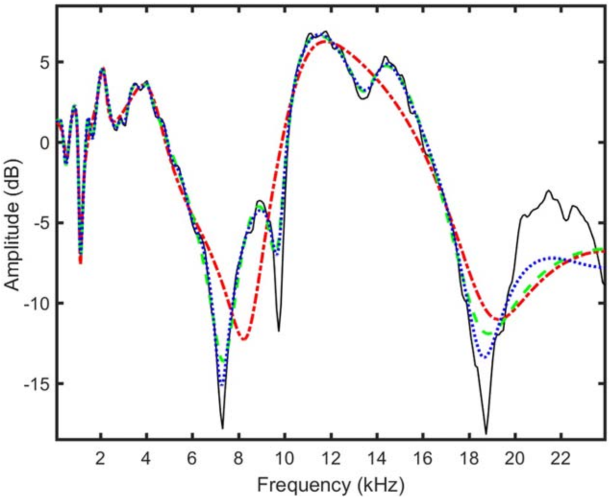


Fig. 5. Head-related transfer functions (HRTFs) represented in **SimpleFreeFieldHRSOS** as an example for the data type second-order section (SOS). Dash lines: Amplitude spectra obtained with 6 (red, dash-dotted), 12 (green, dashed), and 18 (blue, dotted) SOSs, represented in **SimpleFreeFieldHRSOS**. Solid black line: Amplitude spectrum of the original HRTF for the reference.

SingleRoomSRIR[2] and **SingleRoomMIMOSRIR** aim at storing such SRIRs obtained from measurement or simulation setups along with a convenient indexing scheme of the positions and/or orientations of the *Source* and *Listener*, as well as of the multiple channels of the *Listener* (compact microphone array) and *Source* (compact source array) in the **SingleRoomMIMOSRIR** case.

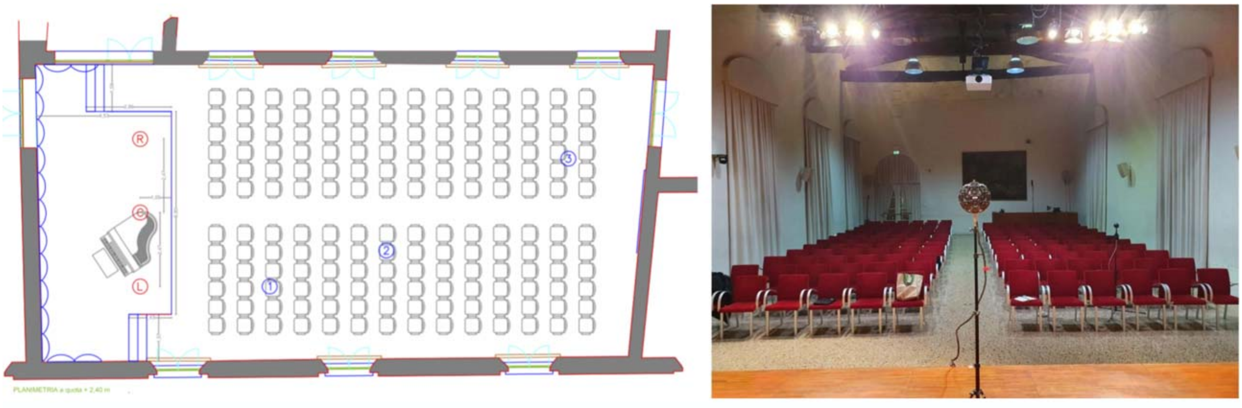
These SOFA conventions are well-suited for spatial audio applications, such as room acoustics analysis and reproduction [15, 60, 61]. Additionally, the option of multiple *Emitters* supported by **SingleRoomMIMOSRIR** makes it possible to simulate arbitrary source directivity, e.g., [62].

SingleRoomSRIR is intended for SRIRs measured with an arbitrary number of *Receivers* (**`R`** $>$ 1, such as a microphone array), but it is limited to a source having a single *Emitter* with arbitrary directivity (**`E`** $=$ 1). The SRIRs can be provided for multiple positions and/or orientations of the *Listener* and/or *Source* in a single room. Consequently, in such cases, `ListenerPosition` and `SourcePosition` are **`M`**-by-**`C`** matrices. If *Listener* uses varying orientations, `ListenerView` and `ListenerUp` are **`M`**-by-**`C`** matrices (applying to `SourceView` and `SourceUp` for the Source manipulations). Note that only by providing the actual coordinates of the *Receivers* (in `ReceiverPosition`, in the local coordinate system centered on the *Listener*), the geometric relations of the measurement scenario can be reproduced, which is important for users, e.g., applying their own SH transform on the data.

As an extension to represent a Source with multiple *Emitters*, **SingleRoomMIMOSRIR** is meant for storing SRIRs

---

[2] As of SOFA 2.0, **SingleRoomSRIR** is the standardized version of **SingleRoomDRIR** proposed by the community.



(a) (b)

Fig. 6. Measurements of spatial room impulse responses (SRIRs) in Sala dei Concerti of Casa della Musica, Parma. (a) Measurement grid. Blue: Positions of the *Listener* (compact microphone array). Red: Positions of the *Source* (compact loudspeaker array). (b) Photography of the setup with *Source-Listener* position combination C-1.

measured with a compact *Source* with an arbitrary number of *Emitters* of arbitrary directivity (**E** > 1, such as a multi-driver loudspeaker) for multiple positions and/or orientations of the *Listener* and/or *Source* in a single room. To this end, the data matrix is a 4D representation with the dimensions **M**, **R**, **N**, and **E**. The other properties of **SingleRoomSRIR**, e.g., a compact *Listener* containing an arbitrary number of *Receivers*, are propagated unchanged to **SingleRoomMIMOSRIR**. Note that only by providing the actual coordinates of both *Receivers* (in `ReceiverPosition`, in the local coordinate system centered on the *Listener*) and *Emitters* (in `EmitterPosition`, in the local coordinate system centered on the *Source*), the geometric relations of the scenario can be reproduced.

Both conventions consider measurement setups comprising different models of Listeners, Receivers, Sources, and Emitters. In such cases, `ListenersDescription`, `ReceiversDescription`, `SourcesDescription`, and `EmittersDescription`, respectively, can be used as variables to describe the different objects (note the plural in the name).

An example for **SingleRoomMIMOSRIR** is the set of SRIR data measured at the Sala dei Concerti of Casa della Musica in Parma as shown in Fig. 6. These measurements were carried out with a 32-channel compact microphone array and 32-channel compact loudspeaker array (the loudspeaker drivers were used independently and sequentially).

In SOFA, the microphone array used in that campaign is represented as a *Listener* with 32 *Receivers*, i.e., **R** = 32. The loudspeaker array is represented as *Source* with 32 *Emitters*, i.e., **E** = 32. Moreover, the measurements were done for three *Listener* positions and three *Source* positions [see Fig. 6(a)], resulting in nine measurements, i.e., **M** = 9. The 4D data matrix containing the SRIRs is thus 9 × 32 × **N** × 32, resulting in 9,216 SRIRs, each of size **N** taps. Note that *Listener* and *Source* had the same orientation at all considered positions, although **SingleRoomMIMOSRIR** supports different orientations for each measurement.

In the next example, the discrete *Receiver* and *Emitter* positions are transformed to the continuous representation of a spherical harmonics expansion. For third-order Ambisonics, the 4D matrix containing the spherical harmonics–encoded SRIRs is thus 9 × 16 × **N** × 16. The attributes `Type` of `ReceiverPosition` and `EmitterPosition` are set to "spherical harmonics." Then, `ReceiverPosition` and `EmitterPosition` each become a 1-by-**C** matrix, with no actual meaning.

These measurements were done in the scope of the Sipario project [63], and more details on the measurement campaign and data processing can be found in [64]. Examples of SOFA files using the described conventions **SingleRoomSRIR** and **SingleRoomMIMOSRIR** are available for download from the SOFA examples website [65].

Note that SOFA files of type **SingleRoomSRIR** or **SingleRoomMIMOSRIR** can be of large size depending on the number of *Emitters* (**E**), number of *Receivers* (**R**), total number of positions (**M**), and length of the individual SRIRs (**N**). When saving such files, enabling data compression available in NetCDF offers an efficient storage.

2.4 Acoustical Directivity

The source directivity describes the directional distribution of the radiated sound energy. Although the directivity of sources such as loudspeakers can be described straightforwardly, describing the directivity of natural sources, such as singers or musical instruments, is more complicated. For example, identical musical notes can have different directivities—think of a guitar on whose on which the same musical note can be played on different strings. The convention **FreeFieldDirectivityTF** (introduced in SOFA 2.0) aims at organizing the description of such directivities.

FreeFieldDirectivityTF uses the data type TF to provide much flexibility to the users. This way not only directivities recorded for specific frequencies can be stored, but also those for musical notes can be described by providing the frequency information about the notes, complete spectra with linearly spaced frequencies can be converted to time-domain signals (by means of the inverse Fourier transform), non-linear frequency spacing can be used to represent directivities averaged (e.g., across one-third–octave frequency

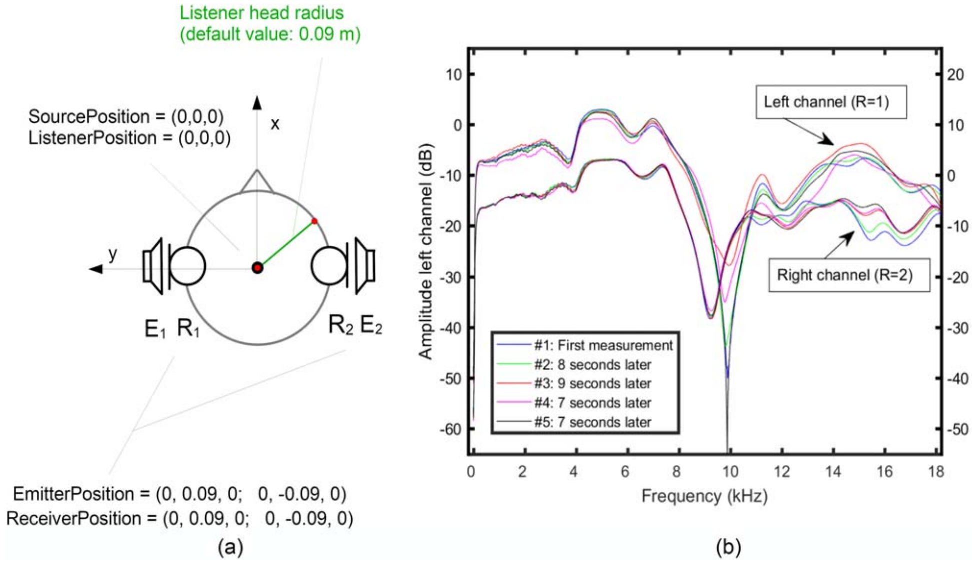


Fig. 7. **SimpleHeadphoneIR** for the representation of headphone impulse responses (HpIRs). (a) Schematics of the acoustic setup with the metadata and default values used in **SimpleHeadphoneIR**. (b) Binaural amplitude spectra of HpIRs from the Acoustics Research Institute (ARI) database for the subject NH5 obtained from the file hpir_nh5.sofa. For clarity, the channels were vertically separated by 10 dB.

bands), or musical spectra can be stored as a fundamental frequency with the directivity of the overtones.

FreeFieldDirectivityTF also defines the mandatory metadata to make sure the data can be understood by others. To this end, it requires the attributes `SourceType` and `SourceManufacturer`, both being the narrative descriptions of the acoustic source and its manufacturers, respectively. It requires `ListenerView` and `SourceView`, assuming a directional *Listener* and *Source*, respectively. Furthermore, it requires the attributes `Reference` to `SourcePosition`, `SourceView`, and `SourceUp` as narrative descriptions of the spatial reference of the *Source* position and orientation. Examples for such a reference of the `SourcePosition` can be "the bell" for a trumpet or "on the front plate between the low- and high-frequency driver" for a loudspeaker. Examples for such a reference of the `SourceView` can be "viewing direction of the bell" for a trumpet or "perpendicular to the front plate" for a loudspeaker. Examples for such a reference of the `SourceUp` can be "along the keys, keys up" for a trumpet or "perpendicular to the top plate" for a loudspeaker.

As an example for **FreeFieldDirectivityTF**, Fig. 8 shows the directivity of the fundamental frequency of the note A4 played on a trumpet. The note was recorded in the anechoic chamber, processed as described in [66], and interpolated to a $1^\circ \times 1^\circ$ resolution by using first-order splines. In SOFA, the metadata specifies the orientation of the trumpet in the coordinate system, played note, and tuning frequency, among various other information.

This example can be reconstructed with the file `demo_DirectivitiyTF.m` from the SOFA Toolbox 2.0 [46] in combination with the AKTools 1.2.0 [67].

2.5 HpIRs

The convention **SimpleHeadphoneIR** was standardized to enable an easy way for storing IRs with a one-to-one correspondence between *Emitter* and *Receiver*. The main application for this convention is to store HpIRs measured for each side of the head.

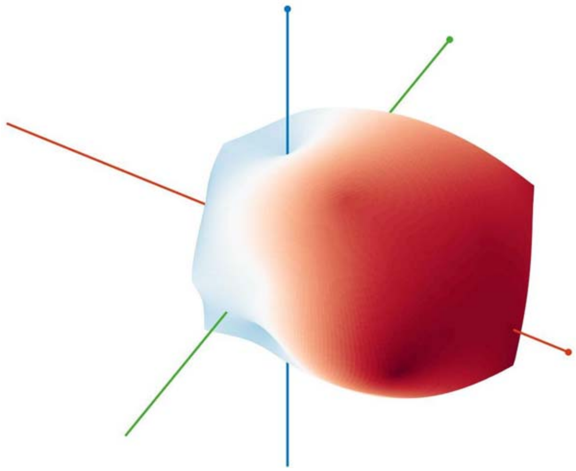


Fig. 8. Directivity of the fundamental frequency of the note A4 played on a trumpet. Sound pressure (in decibels) is indicated by radius and color. The red, green, and blue lines denotes the x, y, and z axis, respectively, with the dots pointing toward the positive directions.

Fig. 7(a) shows an example of such an acoustic setup: the headphones are placed on a human or artificial head, and the response of each driver is acquired at the corresponding (potentially simulated) ear canal. In terms of SOFA objects, the IRs between *Emitters* and *Receivers* are represented. Note that although the typical headphone measurement setup deals with two *Emitters* and two *Receivers*, **SimpleHeadphoneIR** is not limited to that—the only limitation is the requirement to have the same number of *Emitters* and *Receivers*. The support of multiple measurements for the same headphone and/or listener makes **SimpleHeadphoneIR** interesting.

In fact, **SimpleHeadphoneIR** can store HpIRs acquired under various conditions in a single SOFA file. As an example, Fig. 7(b) shows five amplitude spectra per ear stored in a single file. The dataset represents repeated measurements recorded with single headphones for the same listener, repeated by taking the headphones off and then putting them back on the head. In the SOFA file, this dataset is represented as the data type FIR, i.e., as an **`M`**-by-**`R`**-by-**`N`** matrix of binaural IRs (here, with the 256-tap IRs, the `Data.IR` is a matrix of $5 \times 2 \times 256$ elements), complemented with the metadata `MeasurementDate` storing five elements, each of them representing the time stamp of the measurement **`M`**. This example can be reconstructed with the file `demo_HeadphoneIR.m` from the SOFA Toolbox 2.0 [46].

2.6 Conventions for General Data (Not Fitting Specific Conventions)

SOFA can represent any type of spatially oriented data. In SOFA 1.0, users had to create their own convention to do so, and over the course of time, two conventions have been established (**GeneralFIR** and **GeneralTF**). In SOFA 2.0, these and three further conventions (**GeneralFIR-E**, **GeneralTF-E**, and their full generalization **General**) have been standardized to ease access to any type of spatial data.

The most general convention is called **General**. It has minimum requirements on the information to be provided, namely, the mandatory global attributes, mandatory object-related metadata, and the data, with the data type being one of those standardized or defined by the user. This convention is thought to represent any existing and/or future acoustic setup, at the price of being least specific in its description.

Conventions **GeneralFIR-E**, **GeneralTF-E**, **General-FIR**, and **GeneralTF** restrict the convention **General** to the data type to be FIR-E, TF-E, FIR, and TF, respectively. Users with their data in one of these data types are encouraged to use one of these conventions to store their spatially oriented data that does not fit into the more-specific conventions.

3 SUMMARY

With the update of the *AES Standard 69-2020*, SOFA 2.0 introduced many exciting possibilities to represent spatial data. An important addition is the method to represent continuous-direction data by means of spherical harmonics, in the audio community known as Ambisonics.

Although remaining backward compatible to the original *AES Standard 69-2015* (as known as SOFA 1.0), SOFA 2.0 introduced new conventions to describe a variety of spatial configurations, e.g., a convention describing the directivity of musical instruments and loudspeakers, with flexibility not covered by other AES standards; two conventions describing SRIR measurements enabling complex interaction between sources and listeners (such as MIMO data and multi-perspective representations); or conventions for a more comprehensive and flexible description of HRTFs, SRIRs, and equivalent data. Finally, general conventions for a comprehensive representation of any spatial acoustic setup are provided as a basis for being further tailored down by the user.

All those details have been further fine-tuned in the update *AES Standard 69-2022* (as known as SOFA 2.1), which is described and available to the users at the SOFA website [44]. Many datasets have been provided by contributors from all around the world, being available at the SOFA repository [65]. Data stored in the continuous-directional representation of SOFA can be directly integrated in the ADM [41] and/or in MPEG-H [36]. Other organizations, such as the European Telecommunications Standards Institute, proposed directly SOFA for binaural rendering within the virtual-reality services of the telecommunication standard 5G [68]. Within the field of virtual reality and 360° video, SOFA has been proposed to be an essential part of the reproduction system [69]. SOFA is a registered media subtype [70], offering both application developers and users a convenient container for storage, transfer, and exchange of spatially oriented data.

4 ACKNOWLEDGMENT

The authors thank all the SOFA contributors for providing the datasets and creating toolboxes to the community. The authors also thank David Ackermann for providing the directivity data described in SEC. 2.4 and Angelo Farina for providing the SRIRs used as examples in SEC. 2.3.

This work was supported by the European Union (RIA action of Horizon 2020, project SONICOM, grant 101017743 to P.M.; Horizon 2020, project MOBIUS, grant 957185 to J.D.M; and Joint Programming Initiative on Cultural Heritage, project PHE, grant ANR-20-JPIC-0002 to J.D.M) and by the French National Research Agency (project RASPUTIN, grant ANR-18-CE38-0004 to M.N.).

5 REFERENCES

[1] H. Møller, M. F. Sørensen, D. Hammershøi, and C. B. Jensen, "Head-Related Transfer Functions of Human Subjects," *J. Audio Eng. Soc.*, vol. 43, no. 5, pp. 300–321 (1995 May).

[2] P. Majdak, M. J. Goupell, and B. Laback, "3-D Localization of Virtual Sound Sources: Effects of Visual Environment, Pointing Method, and Training," *Atten. Percept. Psychophys.*, vol. 72, no. 2, pp. 454–469 (2010 Feb.). http://doi.org/10.3758/APP.72.2.454.

[3] B. G. Shinn-Cunningham, N. Kopco, and T. J. Martin, "Localizing Nearby Sound Sources in a Classroom: Binaural Room Impulse Responses," *J. Acoust. Soc. Am.*, vol. 117, no. 5, pp. 3100–3115 (2005 May). https://doi.org/10.1121/1.1872572.

[4] D. Khaykin and B. Rafaely, "Acoustic Analysis by Spherical Microphone Array Processing of Room Impulse

Responses," *J. Acoust. Soc. Am.*, vol. 132, no. 1, pp. 261–270 (2012 Jul.). https://doi.org/10.1121/1.4726012.

[5] P. Massé, T. Carpentier, O. Warusfel, and M. Noisternig, "A Robust Denoising Process for Spatial Room Impulse Responses With Diffuse Reverberation Tails," *J. Acoust. Soc. Am.*, vol. 147, no. 4, pp. 2250–2260 (2020 Apr.). http://doi.org/10.1121/10.0001070.

[6] F. Otondo and J. H. Rindel, "The Influence of the Directivity of Musical Instruments in a Room," *Acta Acust. united Acust.*, vol. 90, no. 6, pp. 1178–1184 (2004 Nov.).

[7] J. Blauert, M. Brüggen, K. Hartung, et al., "The AUDIS Catalog of Human HRTFs," in *Proceedings of 16th International Congress on Acoustics (ICA)*, vol. 1, pp. 2901–2902 (Seattle, WA) (1998 Jun.). https://doi.org/10.1121/1.422910.

[8] W. G. Gardner and K. D. Martin, "HRTF Measurements of a KEMAR," *J. Acoust. Soc. Am.*, vol. 97, no. 6, pp. 3907–3908 (1995 Jun.). http://doi.org/10.1121/1.412407.

[9] MATLAB, *Version 7.10.0 (R2010a)* (MathWorks Inc., Natick, MA, 2010).

[10] V. R. Algazi, R. O. Duda, D. M. Thompson, and C. Avendano, "The CIPIC HRTF database," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 99–102 (New York, NY) (2001 Oct.). http:/doi.org/10.1109/ASPAA.2001.969552.

[11] H. Wierstorf, M. Geier, A. Raake, and S. Spors, "A Free Database of Head Related Impulse Response Measurements in the Horizontal Plane With Multiple Distances," presented at the *130th Convention of the Audio Engineering Society* (2011 May), e-Brief 6. https://www.aes.org/e-lib/browse.cfm?elib=16564.

[12] O. Warusfel, Institut de recherche et coordination acoustique/musique, "LISTEN HRTF Database," http://recherche.ircam.fr/equipes/salles/listen/ (2003 May).

[13] A. Andreopoulou and A. Roginska, "Towards the Creation of a Standardized HRTF Repository," presented at the *131st Convention of the Audio Engineering Society* (2011 Oct.), paper 8571. https://www.aes.org/e-lib/online/browse.cfm?elib=16096.

[14] D. Schwarz and M. Wright, "SDIF Sound Description Interchange Format," http://sdif.sourceforge.net/ (2000).

[15] J. Pätynen, S. Tervo, and T. Lokki, "Analysis of Concert Hall Acoustics via Visualizations of Time-Frequency and Spatiotemporal Responses," *J. Acoust. Soc. Am.*, vol. 133, no. 2, pp. 842–857 (2013 Feb.). http://doi.org/10.1121/1.4770260.

[16] Institute for Communication Systems, "Aachen Impulse Response (AIR) Database," http://www.iks.rwth-aachen.de/forschung/tools-downloads/databases/aachen-impulse-response-database/ (accessed Nov. 29, 2021).

[17] J. Blauert, J. Braasch, J. Buchholz, et al., "Aural Assessment by Means of Binaural Algorithms – The AABBA Project," in *Proceedings of the International Symposium on Auditory and Audiological Research*, vol. 2, pp. 113–124 (Helsingor, Denmark) (2009 Aug.). https://proceedings.isaar.eu/index.php/isaarproc/article/view/2009-12.

[18] Institut für Schallforschung, "AABBA: Aural Assessment by Means of Binaural Algorithms," http://oeaw.ac.at/isf/aabba (2021).

[19] 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.

[20] AES, "AES Standard for File Exchange - Spatial Acoustic Data File Format," *AES Standard 69-2015* (2015).

[21] R. Sridhar, J. G. Tylka, and E. Choueiri, "A Database of Head-Related Transfer Functions and Morphological Measurements," presented at the *143rd Convention of the Audio Engineering Society* (2017 Oct.), paper 357. https://www.aes.org/e-lib/browse.cfm?elib=19308.

[22] AES, "AES Standard for File Exchange - Spatial Acoustic Data File Format," *AES Standard 69-2020* (2020 Dec.).

[23] R. Rew and G. Davis, "NetCDF: An Interface for Scientific Data Access," *IEEE Comput. Graph. Appl.*, vol. 10, no. 4, pp. 76–82 (1990 Jul.). http://doi.org/10.1109/38.56302.

[24] R. Rew, D. Glenn, and S. Emmerson, *NetCDF User's Guide: An Interface for Data Access*, (1993 Apr.). http:/doi.org/10.1109/38.56302.

[25] Q. Koziol and D. Robinson, "HDF5," *Publicly Accessible Repository* (2018 Mar.). http:/doi.org/10.11578/dc.20180330.1.

[26] MathWorks, "MAT-File Versions," https://de.mathworks.com/help/matlab/import_export/mat-file-versions.html (accessed Nov. 29, 2021).

[27] ISO, "Date and Time—Representations for Information Interchange—Part 1: Basic Rules," *ISO Standard 8601-1:2019* (2019 Feb.). https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/07/09/70907.html.

[28] International Bureau of Weights and Measures, *The International System of Units (SI)* (BIPM, Sèvres, France, 2019), 9th ed.

[29] A. Oppenheim and R. Schafer, *Discrete-Time Signal Processing* (Pearson Education Ltd, Harlow, UK, 2013), 3rd ed.

[30] P. P. Vaidyanathan, "Robust Digital Filter Structures," in S. K. Mitra and J. F. Kaiser (Eds.), *Handbook for Digital Signal Processing* (John Wiley & Sons, New York, NY, 1993).

[31] S. A. White, "Design of a Digital Biquadratic Peaking or Notch Filter for Digital Audio Equalization," *J. Audio Eng. Soc.*, vol. 34, no. 6, pp. 479–483 (1986 Jun.). http://www.aes.org/e-lib/browse.cfm?elib=5262.

[32] Y. Jin, L. Chen, Y. Yin, H. Ren, and M. Zhao, "The Vector View-Up in Computer Graphics," in T. Xiao, L. Zhang, and M. Fei (Eds.), *AsiaSim*, Communications in Computer and Information Science, vol. 325, pp. 167–177 (Springer, Berlin, Germany, 2012). http://doi.org/10.1007/978-3-642-34387-2_20.

[33] ISO, "Industrial Automation Systems and Integration—COLLADA Digital Asset Schema Specification for 3D Visualization of Industrial Data," *ISO/PAS*

Standard 17506:2012 (2012 Jul.). https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/05/99/59902.html.

[34] "Uniform Resource Identifier (URI): Generic Syntax," https://datatracker.ietf.org/doc/html/rfc3986 (accessed Dec. 07, 2021).

[35] ISO, "Quantities and Units—Part 2: Mathematics," *ISO Standard 80000-2:2019* (2019 Aug.). https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/49/64973.html.

[36] J. Herre, J. Hilpert, A. Kuntz, and J. Plogsties, "MPEG-H Audio—The New Standard for Universal Spatial/3D Audio Coding," *J. Audio Eng. Soc.*, vol. 62, no. 12, pp. 821–830 (2014 Dec.). http://doi.org/10.17743/jaes.2014.0049.

[37] M. A. Blanco, M. Flórez, and M. Bermejo, "Evaluation of the Rotation Matrices in the Basis of Real Spherical Harmonics," *J. Mol. Struct.: THEOCHEM.*, vol. 419, no. 1, pp. 19–27 (1997 Dec.). http://doi.org/10.1016/S0166-1280(97)00185-1.

[38] P. B. Fellgett, "Ambisonic Reproduction of Directionality in Surround-Sound Systems," *Nature*, vol. 252, pp. 534–538 (1974 Dec.). http://doi.org/10.1038/252534b0.

[39] F. Zotter and M. Frank, "All-Round Ambisonic Panning and Decoding," *J. Audio Eng. Soc.*, vol. 60, no. 10, pp. 807–820 (2012 Oct.). https://www.aes.org/e-lib/browse.cfm?elib=16554.

[40] M. A. Wieczorek and M. Meschede, "SHTools: Tools for Working With Spherical Harmonics," *Geochem. Geophys. Geosyst.*, vol. 19, no. 8, pp. 2574–2592 (2018 May). http://doi.org/10.1029/2018GC007529.

[41] ITU-R, "Audio Definition Model," *ITU-R Recommendation BS.2076-2* (2019 Oct.). https://www.itu.int/dms_pubrec/itu-r/rec/bs/R-REC-BS.2076-2-201910-I!!PDF-E.pdf.

[42] ISO, "Information Technology—High Efficiency Coding and Media Delivery in Heterogeneous Environments—Part 3: 3D Audio," *ISO/IEC Standard 23008-3:2019* (2019 Feb.). https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/07/44/74430.html.

[43] M. Chapman, W. Ritsch, T. Musil, et al., "A Standard for Interchange of Ambisonic Signal Sets," in *Proceedings of the Ambisonics Symposium*, paper 27 (Graz, Austria) (2009 Jun.). https://iem.kug.ac.at/fileadmin/media/iem/projects/2009/ambixchange09.pdf.

[44] The SOFA Team, "SOFA (Spatially Oriented Format for Acoustics)," http://sofaconventions.org (accessed Dec. 23, 2021).

[45] The SOFA Team, "Index of /Data/Database/Thk," https://sofacoustics.org/data/database/thk/ (accessed Dec. 23, 2021).

[46] The ARI Team, "SOFA: Spatially Oriented Format for Acoustics," http://sourceforge.net/projects/sofacoustics (accessed Jun. 28, 2016).

[47] J. Huopaniemi, N. Zacharov, and M. Karjalainen, "Objective and Subjective Evaluation of Head-Related Transfer Function Filter Design," *J. Audio Eng. Soc.*, vol. 47, no. 4, pp. 218–239 (1999 Apr.). http://www.aes.org/e-lib/browse.cfm?elib=12109.

[48] A. Kulkarni and H. S. Colburn, "Infinite-Impulse-Response Models of the Head-Related Transfer Function," *J. Acoust. Soc. Am.*, vol. 115, no. 4, pp. 1714–1728 (2004 Apr.). http://doi.org/10.1121/1.1650332.

[49] J. Mackenzie, J. Huopaniemi, V. Valimaki, and I. Kale, "Low-Order Modeling of Head-Related Transfer Functions Using Balanced Model Truncation," *IEEE Signal Proc. Lett.*, vol. 4, no. 2, pp. 39–41 (1997 Feb.). http://doi.org/10.1109/97.554467.

[50] J. Dattorro, "The Implementation of Recursive Digital Filters for High-Fidelity Audio," *J. Audio Eng. Soc.*, vol. 36, no. 11, pp. 851–878 (1988 Nov.). https://www.aes.org/e-lib/browse.cfm?elib=5125.

[51] R. Wilson, "Filter Topologies," presented at the *AES UK 7th Conference: Digital Signal Processing (DSP)* (1992 Sep.), paper DSP-09. https://www.aes.org/e-lib/browse.cfm?elib=6169.

[52] G. Ramos and J. J. López, "Filter Design Method for Loudspeaker Equalization Based on IIR Parametric Filters," *J. Audio Eng. Soc.*, vol. 54, no. 12, pp. 1162–1178 (2006 Dec.). https://www.aes.org/e-lib/online/browse.cfm?elib=13893.

[53] H. Hasegawa, M. Kasuga, S. Matsumoto, and A. Koike, "Simply Realization of Sound Localization Using HRTF Approximated by IIR Filter," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E83-A, no. 6, pp. 973–978 (2000 Jun.).

[54] G. Ramos and M. Cobos, "Parametric Head-Related Transfer Function Modeling and Interpolation for Cost-Efficient Binaural Sound Applications," *J. Acoust. Soc. Am.*, vol. 134, no. 3, pp. 1735–1738 (2013 Sep.). http://doi.org/10.1121/1.4817881.

[55] K. Steiglitz and L. McBride, "A Technique for the Identification of Linear Systems," *IEEE Trans. Autom. Control*, vol. 10, no. 4, pp. 461–464 (1965 Oct.). http://doi.org/10.1109/TAC.1965.1098181.

[56] L. B. Jackson, "Discrete-Time Networks," in L. B. Jackson, *Digital Filters and Signal Processing: With MATLAB Exercises*, pp. 95–137 (Springer, Boston, MA, 1996). http://doi.org/10.1007/978-1-4757-2458-5_5.

[57] T. Carpentier, H. Bahu, M. Noisternig, and O. Warusfel, "Measurement of a Head-Related Transfer Function Database With High Spatial Resolution," in *Proceedings of the 7th Forum Acusticum*, pp. 1–6 (Krakow, Poland) (2014 Sep.). https://hal.archives-ouvertes.fr/hal-01247583.

[58] IRCAM, "Contents of SimpleFreeFieldSOS/BILI/COMPENSATE," http://opendap.ircam.fr/hyrax/SimpleFreeFieldSOS/BILI/COMPENSATED_INTERPOLATED/96000/contents.html (accessed Dec. 23, 2021).

[59] Institut de recherche et coordination acoustique/musique, "BILI HRTF Database" (FreeFieldHRIR/BILI/COMPENSATE dataset), http://opendap.ircam.fr/hyrax/SimpleFreeFieldHRIR/BILI/COMPENSATED_INTERPOLATED/96000/contents.html (accessed Dec. 23, 2021).

[60] L. McCormack, V. Pulkki, A. Politis, O. Scheuregger, and M. Marschall, "Higher-Order Spatial Impulse Response Rendering: Investigating the Perceived Effects of Spherical Order, Dedicated Diffuse Rendering, and Frequency Resolution," *J. Audio Eng. Soc.*, vol. 68, no. 5, pp. 338–354 (2020 May). http://doi.org/10.17743/jaes.2020.0026.

[61] J. Merimaa and V. Pulkki, "Spatial Impulse Response Rendering I: Analysis and Synthesis," *J. Audio Eng. Soc.*, vol. 53, no. 12, pp. 1115–1127 (2005 Dec.). http://www.aes.org/e-lib/browse.cfm?elib=13401.

[62] M. T. Neal and M. C. Vigeant, "The CHOR-Database: A Twenty-One Concert Hall Spherical Microphone and Loudspeaker Array Measurement Database," in *Proceedings of the 23rd International Congress on Acoustics*, pp. 7863–7870 (Aachen, Germany) (2019 Sep.). http://doi.org/10.18154/RWTH-CONV-239853.

[63] M. Dolci, "Sipario – Il Suono: Arte Intangibile Delle Performing Arts – Ricerca Su Teatri Italiani per l'Opera," https://www.progettosipario.org/homepage/ (accessed Dec. 23, 2021).

[64] A. Farina and L. Chiesi, "Measuring Spatial MIMO Impulse Responses in Rooms Employing Spherical Transducer Arrays," in *Proceedings of the AES International Conference on Sound Field Control* (2016 Jul.), paper 7-2.

[65] The SOFA Team, "Index of /Data," http://sofacoustics.org/data (accessed Dec. 23, 2021).

[66] N. R. Shabtai and M. Vorländer, "Acoustic Centering of Sources With High-Order Radiation Patterns," *J. Acoust. Soc. Am.*, vol. 137, no. 4, pp. 1947–1961 (2015 Apr.). https://doi.org/10.1121/1.4916594.

[67] F. Brinkmann and S. Weinzierl, "AKtools—An Open Software Toolbox for Signal Acquisition, Processing, and Inspection in Acoustics," presented at the *142nd Convention of the Audio Engineering Society* (2017 May), e-Brief 309. https://www.aes.org/e-lib/browse.cfm?elib=18685.

[68] ETSI, "5G; Virtual Reality (VR) Profiles for Streaming Applications," *ETSI Technical Specification 126 118 V16.2.1* (2021 Jan.). https://www.etsi.org/deliver/etsi_ts/126100_126199/126118/15.03.00_60/ts_126118v150300p.pdf.

[69] L. Reed and P. Phelps, "Audio Reproduction in Virtual Reality Cinemas – Position Paper," in *Proceedings of the IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 1513–1516 (Osaka, Japan) (2019 Mar.). http://doi.org/10.1109/VR.2019.8797904.

[70] P. Majdak, "Media Type: Audio/Sofa," https://www.iana.org/assignments/media-types/audio/sofa (accessed Dec. 7, 2021).

[71] The HDF Group, "HDFView 3.x User's Guide," https://confluence.hdfgroup.org/display/HDFVIEW/HDFView+3.x+User%27s+Guide (accessed Jan. 1, 2022).

[72] The ARI Team, "SOFA Matlab/Octave API," https://github.com/sofacoustics/SOFAtoolbox (accessed Jul. 22, 2022).

[73] F. Brinkmann and M. Berzborn, "pyfar/sofar: Maybe the Most Complete Python Package for SOFA Files so Far," https://github.com/pyfar/sofar (accessed Dec. 13, 2021).

[74] Sofacoustics, "API_Cpp," https://github.com/sofacoustics/API_Cpp (accessed Dec. 13, 2021).

[75] A. P. López, "pysofaconventions," http://github.com/andresperezlopez/pysofaconventions (accessed Dec. 13, 2021).

[76] J. Encke, "pySOFA," https://github.com/Jencke/pySOFA (accessed Dec. 13, 2021).

[77] I. Laghidze, "SOFASonix," https://github.com/OneBadNinja/SOFASonix (accessed Dec. 13, 2021).

[78] C. Hoene, "libmysofa," https://github.com/hoene/libmysofa (accessed Dec. 13, 2021).

[79] C. W. Budde, "WebSofa," https://github.com/CWBudde/WebSofa (accessed Dec. 13, 2021).

[80] D. Johnson and H. Lee, "APL-Huddersfield/SOFA-for-Max: SOFA For Max," *Zenodo* (2019 Jul.). http://doi.org/10.5281/zenodo.3269271.

[81] T. Carpentier, M. Noisternig, and O. Warusfel, "Twenty Years of Ircam Spat: Looking Back, Looking Forward," in *Proceedings of the 41st International Computer Music Conference (ICMC)*, pp. 270–277 (Denton, TX) (2015 Sep.). https://hal.archives-ouvertes.fr/hal-01247594.

[82] IRCAM, "Spat," https://forum.ircam.fr/projects/detail/spat/ (2021) (accessed Dec. 13, 2021).

[83] Sofacoustics, "SOFAlizer-for-VLC-Player," https://github.com/sofacoustics/SOFAlizer-for-VLC-Player (accessed Dec. 13, 2021).

[84] M. Brandner, M. Frank, and D. Rudrich, "DirPat—Database and Viewer of 2D/3D Directivity Patterns of Sound Sources and Receivers," presented at the *144th Convention of the Audio Engineering Society* (2018 May), e-Brief 425. https://www.aes.org/e-lib/browse.cfm?elib=19538.

[85] 3D Tune-In, "3D Tune-In," https://www.3d-tune-in.eu/ (accessed Dec. 13, 2021).

[86] 3DTune-In, "3dti AudioToolkit," https://github.com/3DTune-In/3dti_AudioToolkit (accessed Dec. 13, 2021).

[87] D. Poirier-Quinot and B. F. G. Katz, "The Anaglyph Binaural Audio Engine," presented at the *144th Convention of the Audio Engineering Society* (2018 May), e-Brief 431. https://www.aes.org/e-lib/browse.cfm?elib=19544.

[88] 3DJ, "[WIP] Binaural Audio," https://binaural-audio.slite.com/p/note/2VyRD1D63DjKAWw1VxNuKt (accessed Dec. 13, 2021).

[89] D. Carvalho, "Individualized HRTF Synthesis," https://github.com/davircarvalho/Individualized_HRTF_Synthesis (accessed Dec. 13, 2021).

[90] P. Majdak and M. Mihocic, "HRTF-Database," https://www.oeaw.ac.at/isf/das-institut/software/hrtf-database (accessed Dec. 13, 2021).

[91] P. Majdak, B. Masiero, and J. Fels, "Sound Localization in Individualized and Non-Individualized Crosstalk Cancellation Systems," *J. Acoust. Soc. Am.*, vol. 133, no. 4, pp. 2055–2068 (2013 Apr.). http://doi.org/10.1121/1.4792355.

[92] RIEC, "The RIEC HRTF Dataset," http://www.riec.tohoku.ac.jp/pub/hrtf/index.html (accessed Dec. 13, 2021).

[93] K. Watanabe, Y. Iwaya, Y. Suzuki, S. Takane, and S. Sato, "Dataset of Head-Related Transfer Functions Measured With a Circular Loudspeaker Array," *Acoust. Sci. Technol.*, vol. 35, no. 3, pp. 159–165 (2014 May). http://doi.org/10.1250/ast.35.159.

[94] R. Bomhardt, M. de la Fuente Klein, and J. Fels, "A High-Resolution Head-Related Transfer Function and Three-Dimensional Ear Model Database," *Proc. Mtgs. Acoust.*, vol. 29, no. 1, paper 050002 (2016 Nov.). http://doi.org/10.1121/2.0000467.

[95] F. Brinkmann, M. Dinakaran, R. Pelzer, et al., "A Cross-Evaluated Database of Measured and Simulated HRTFs Including 3D Head Meshes, Anthropometric Features, and Headphone Impulse Responses," *J. Audio Eng. Soc.*, vol. 67, no. 9, pp. 705–718 (2019 Sep.). http://doi.org/10.17743/jaes.2019.0024.

[96] S. Ghorbal and R. Séguier, "Computed HRIRs and Ears Database for Acoustic Research," https://sofacoustics.org/data/database/chedar/documentation.pdf (2020 Feb.).

[97] S. Harder, R. R. Paulsen, M. Larsen, et al., "A Framework for Geometry Acquisition, 3-D Printing, Simulation, and Measurement of Head-Related Transfer Functions With a Focus on Hearing-Assistive Devices," *Comput. Aided Des.*, vol. 75–76, pp. 39–46 (2016 Jun.). http://doi.org/10.1016/j.cad.2016.02.006.

[98] C. Pörschmann, J. M. Arend, and R. Gillioz, "How Wearing Headgear Affects Measured Head-Related Transfer Functions," in *Proceedings of the EAA Spatial Audio Signal Processing Symposium*, pp. 49–54 (Paris, France) (2019 Sep.). http://doi.org/10.25836/sasp.2019.27.

[99] G. Yu, R. Wu, Y. Liu, and B. Xie, "Near-Field Head-Related Transfer-Function Measurement and Database of Human Subjects," *J. Acoust. Soc. Am.*, vol. 143, no. 3, pp. EL194–EL198 (2018 Mar.). http://doi.org/10.1121/1.5027019.

[100] H. Wierstorf, M. Geier, A. Raake, and S. Spors, "A Free Database of Head-Related Impulse Response Measurements in the Horizontal Plane With Multiple Distances," *Zenodo* (2016 Jun.). http://doi.org/10.5281/zenodo.55418.

[101] F. Brinkmann, A. Lindau, S. Weinzierl, et al., "A High Resolution and Full-Spherical Head-Related Transfer Function Database for Different Head-Above-Torso Orientations," *J. Audio Eng. Soc.*, vol. 65, no. 10, pp. 841–848 (2017 Oct.). http://doi.org/10.17743/jaes.2017.0033.

[102] R. Greff and B. F. G. Katz, "Round Robin Comparison of HRTF Simulation Results: Preliminary Results," presented at the *123rd Convention of the Audio Engineering Society* (2007 Oct.), paper 7188.

[103] S. Spagnol, K. B. Purkhús, R. Unnthórsson, and S. K. Björnsson, "The Viking HRTF Dataset," presented at the *16th Sound and Music Computing Conference (SMC)* (Málaga, Spain) (2019 May). http://doi.org/10.5281/zenodo.3249252.

[104] H. S. Braren and J. Fels, "A High-Resolution Head-Related Transfer Function Data Set and 3D-Scan of KEMAR," *Tech. Rep. RWTH-2020-11307* (2020). http://doi.org/10.18154/RWTH-2020-11307.

[105] T. Qu, Z. Xiao, M. Gong, et al., "Distance-Dependent Head-Related Transfer Functions Measured With High Spatial Resolution Using a Spark Gap," *IEEE Trans. Audio Speech Lang. Process.*, vol. 17, no. 6, pp. 1124–1132 (2009 Aug.). http://doi.org/10.1109/TASL.2009.2020532.

[106] P. Majdak, M. J. Goupell, and B. Laback, "Two-Dimensional Localization of Virtual Sound Sources in Cochlear-Implant Listeners," *Ear Hear.*, vol. 32, no. 2, pp. 198–208 (2011 Mar.). http://doi.org/10.1097/AUD.0b013e3181f4dfe9.

[107] C. Guezenoc and R. Séguier, "A Wide Dataset of Ear Shapes and Pinna-Related Transfer Functions Generated by Random Ear Drawings," *J. Acoust. Soc. Am.*, vol. 147, no. 6, pp. 4087–4096 (2020 Jun.). http://doi.org/10.1121/10.0001461.

[108] B. B. Boren, M. Geronazzo, P. Majdak, and E. Choueiri, "PHOnA: A Public Dataset of Measured Headphone Transfer Functions," presented at the *137th Convention of the Audio Engineering Society* (2014 Oct.), paper 9126. http://www.aes.org/e-lib/browse.cfm?elib=17449.

[109] M. Geronazzo, F. Granza, S. Spagnol, and F. Avanzini, "A Standardized Repository of Head-Related and Headphone Impulse Response Data," presented at the *134th Convention of the Audio Engineering Society* (2013 May), paper 8902. https://www.aes.org/e-lib/browse.cfm?elib=16802.

[110] F. Brinkmann, R. Roden, A. Lindau, and S. Weinzierl, "Audibility and Interpolation of Head-Above-Torso Orientation in Binaural Technology," *IEEE J. Sel. Top. Signal Process.*, vol. 9, no. 5, pp. 931–942 (2015 Aug.). http://doi.org/10.1109/JSTSP.2015.2414905.

[111] H. Kayser, S. D. Ewert, J. Anemüller, et al., "Database of Multichannel In-Ear and Behind-the-Ear Head-Related and Binaural Room Impulse Responses," *EURASIP J. Adv. Sig. Process.*, paper 298605 (2009 Jul.). http://doi.org/10.1155/2009/298605.

[112] V. Erbes, M. Geier, S. Weinzierl, and S. Spors, "Database of Single-Channel and Binaural Room Impulse Responses of a 64-Channel Loudspeaker Array," presented at the *138th Convention of the Audio Engineering Society* (2015 May), e-Brief 189. https://secure.aes.org/forum/pubs/ebriefs/index.cfm?elib=17624.

[113] D. Satongar, Y. W. Lam, and C. Pike, "Measurement and Analysis of a Spatially Sampled Binaural Room Impulse Response Dataset," in *Proceedings of the 21st International Congress on Sound and Vibration* (Beijing, China) (2014 Jul.). https://core.ac.uk/reader/19725910.

[114] P. Stade, B. Bernschütz, and M. Rühl, "A Spatial Audio Impulse Response Compilation Captured at the WDR Broadcast Studios," *Zenodo* (2012 May). http://doi.org/10.5281/zenodo.3820043.

[115] C. Armstrong, L. Thresh, D. Murphy, and G. Kearney, "A Perceptual Evaluation of Individual and Non-Individual HRTFs: A Case Study of the SADIE II

Database," *Appl. Sci.*, vol. 8, no. 11, paper 2029 (2018 Oct.). http://doi.org/10.3390/app8112029.

[116] T. McKenzie, S. J. Schlecht, and V. Pulkki, "Acoustic Analysis and Dataset of Transitions Between Coupled Rooms," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 481–485 (Toronto, Canada) (2021 Jun.). http://doi.org/10.1109/ICASSP39728.2021.9415122.

[117] T. McKenzie, L. McCormack, and C. Hold, "Dataset of Spatial Room Impulse Responses in a Variable Acoustics Room for Six Degrees-of-Freedom Rendering and Analysis," *Zenodo* (2021 Nov.). http://doi.org/10.5281/zenodo.5720724.

A.1 TOOLBOXES, APPLICATIONS, AND DATA

Here, a snapshot of important toolboxes, applications, and datasets related to SOFA at the time of writing are listed. Most of them are listed and described on the SOFA website [44]. The datasets are available for download from the SOFA repository [65]. References to documentation are provided where available.

1 TOOLBOXES

- HDFView: Generic viewer for Hierarchical Data Format (HDF5) files running in Java for low-level inspection of SOFA files [71].
- SOFA Toolbox 2.0: Comprehensive toolbox for MATLAB and Octave, with the previous versions known as the SOFA API for MATLAB/Octave [46, 72].
- sofar: Comprehensive interface for SOFA in Python including verification of SOFA files [73].
- SOFA API for C++: Interface for handling SOFA files in C++ [74].
- pysofaconventions, pySOFA, SOFASonix: Various lightweight interfaces in Python [75–77].
- libmysofa: Lightweight read-only API in C without nearly any dependencies, suitable for embedded devices and part of Debian Linux [78].
- WebSofa: API based on the libmysofa aiming for applications in JavaScript environments [79].
- sofa~ : A collection of objects for using and creating SOFA files in Max [80].

2 APPLICATIONS

- Spat: Suite for spatialization of sound signals in real-time, intended for musical creation, post production, and live performances [81, 82].
- SOFAlizer plug-in for VideoLANClient (VLC) player: A simple demo of an audio engine as a plugin for the VLC player [83].
- DirPat: Tools for the analysis and visualization of the directivity of acoustic sources [84].
- 3D Tune-In Toolkit: A C++ library aiming at spatialization and simulation of hearing loss within real-time virtual environments [85, 86].
- Anaglyph Virtual Studio Technology (VST): A VST plugin for binaural rendering within digital-audio workstations [87].
- Binaural Audio: Web application site introducing HRTFs and providing audio samples online [88].
- Auralization Engine: A MATLAB-based app for auralization and HRTF personalization [89].

3 DATASETS

3.1 Free-Field HRTFs of Human Listeners

- ARI: HRTFs of over 220 listeners, in two bandwidths (300-Hz and 50-Hz lower frequency) [90, 91].
- CIPIC: HRTFs of 45 listeners with partially available anthropometric data [10].
- BILI (IRCAM): HRTFs of 54 listeners measured at 1,680 directions and sampling rate of 96 kHz.
- CROSSMOD (IRCAM): HRTFs of 24 listeners measured at 651 directions and sampling rate of 44.1 kHz.
- LISTEN (IRCAM): HRTFs of 50 listeners measured at 187 directions and sampling rate of 44.1 kHz.
- RIEC: HRTFs of over 100 listeners [92, 93].
- Aachen: HRTFs of 48 listeners with anthropometric data and 3D ear models [94].
- HUTUBS: HRTFs of 96 listeners including anthropometric data, headphone impulse responses, and 3D head models [95].
- CHEDAR: Numerically calculated HRTFs with 3D meshes and anthropometric data [96].
- 3D3A: HRTFs of 38 listeners with partially 3D head and torso scans available [21].

3.2 Free-Field HRTFs of Artificial Heads

- MIT: HRTFs of the Knowles Electronics Manikin for Auditory Research (KEMAR) [8].
- ARI: HRTFs of a printed listener head measured with the same setup as for human listeners [97].
- BILI: HRTFs of three mannequins measured at IRCAM using the same setup as for human listeners.
- HRTFs, raw and reference data of the dummy head Neumann KU 100.
- HRTFs, raw and reference data of the dummy head Brüel & Kjaer type 4100D with and without pinna.
- THK: High spatial resolution dataset consisting of 1) far-field HRTFs of Neumann KU100, 2) near-field HRTFs of Neumann KU100 for various distances, and 3) HRTFs of Neumann KU100 and HEAD acoustics HMS II while mounted various head gears [98].
- SCUT: Near-field HRTFs of the KEMAR measured for several distances [99].
- TU-Berlin: HRTFs of the KEMAR measured for several distances [100] and of the FABIAN mannequin acoustically measured and numerically calculated [101].

- Club Fritz: HRTFs of Neumann KU 100 measured by many institutions [102].
- VIKING: HRTFs of the KEMAR mannequin with 20 different pairs of artificial silicone pinnae for 1,513 different directions including 3D scans [103].
- Aachen: High-resolution HRTFs of the KEMAR mannequin combined with a 3D model [104].
- PKU-IOA: High spatial resolution HRTFs of the KEMAR mannequin measured for several distances [105].

3.3 Special General Purpose Free-Field Data

- Behind-the-ear HRTFs: Dataset of HRTFs measured with hearing-aid devices worn behind the ear [106].
- Pinna-related transfer functions (PRTFs): Widespread dataset consisting of 1,005 ear shapes and numerically simulated PRTFs created by randomly weighting the principle components of the pinna representation obtained from an analysis of 119 pinna meshes of actual listeners [107].
- Directivities: A dataset of a three-way loudspeaker (low, middle, and high units) in 10° resolution and two datasets of a trumpet, recorded with a 32-channel microphone array.
- HpIRs ARI: Datasets of HpIRs of over 100 human listeners wearing a single headphone, measured five times with headphone repositioning [108].
- HpIRs BT-DEI: HpIRs of 16 human listeners measured for three headphones [109].
- HpIRs FABIAN: HpIRs of the FABIAN mannequin acoustically measured for 34 headphones [110].

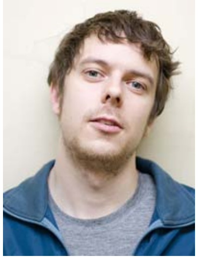
3.4 (Spatial) Room Impulse Responses ((S)RIRs)

- Oldenburg: RIRs measured in an office under several conditions [111].
- TuBuRo: RIRs (from omnidirectional mic) and BRIRs (from KEMAR manequin) recorded with a 64-channel loudspeaker array in a room under various absorbing conditions [112].
- SBSBRIR: BRIRs from the Salford-BBC dataset measured in a recording room for 12 loudspeakers, each for 15 head orientations [113].
- THK: SRIRs and BRIRs measured at the Westdeutscher Rundfunk Köln (WDR) broadcast studios with various microphone arrays [114].
- SADIE: BRIRs and HRTFs measured for the same 20 listeners, including HpIRs and anthropometrics [115].
- Room transition: SRIRs capturing the transition between coupled rooms with 101 positions and four coupled room pairs [116].
- Variable room acoustics: SRIRs measured in a variable acoustics room with two spherical microphone arrays [117].

THE AUTHORS


Piotr Majdak


Franz Zotter


Fabian Brinkmann


Julien De Muynke


Michael Mihocic


Markus Noisternig

Piotr Majdak studied electrical and audio engineering at the University of Technology (TUG) and the University of Music and Performing Arts (KUG), both in Graz, Austria. In 2008, he received his Ph.D. degree in psychoacoustics and signal processing. He works at the Acoustics Research Institute (ARI) of the Austrian Academy of Sciences to get a better understanding of the mechanisms underlying spatial hearing. His keys to reaching out are computer algorithms, toolboxes, and reproducible research. (amtoolbox.org and sofaconventions.org). Piotr is a member of the Acoustical Society of America (ASA), Association for Research in Otolaryngology, and Austrian and German Acoustic Associations (DEGA); the chair of the technical committee for psychological and physiological acoustics of the European Acoustics Association (EAA); and the president of the Austrian section of the AES.

•

Franz Zotter received an M.Sc. degree in electrical and audio engineering from the University of Technology (TUG) in 2004 and Ph.D. degree in natural science in 2009 from the University of Music and Performing Arts (KUG) in Graz, Austria. He joined the Institute of Electronic Music an Acoustics (IEM) of KUG in 2004 as Research Assistant, became Senior Scientist in 2008, and became Tenure-Tracked Assistant Professor in 2019. He co-authored the book *Ambisonics* with Matthias Frank, and his interests include spherical array signal processing and spatial audio applications in music and virtual reality. Franz is a member of the AES, German Acoustical Society (DEGA), and German Tonmeister Society (VDT). In 2012, Franz was awarded DEGA's Lothar Cremer medal.

•

Fabian Brinkmann received an M.A. degree in communication sciences and technical acoustics in 2011 and Dr. rer. nat. degree in 2019 from the Technical University of Berlin, Germany. He focuses on the fields of signal processing and evaluation approaches for spatial audio. Fabian is a member of the AES, German Acoustical Society (DEGA), and the European Acoustics Association (EAA) technical committee for psychological and physiological acoustics.

•

Julien De Muynke received a Master's Degree in electrical engineering from École Nationale Supérieure de l'Électronique et de ses Applications (ENSEA), Cergy, France, in 2005, with his graduation project in the Acoustics Department of Aalborg University. He then worked as an audio-signal processing engineer in the Consumer Electronics industry sector, and in 2015, he joined the spatial audio research group at Eurecat, Centre Tecnológic de Catalunya, Barcelona. Since 2020, Julien has been a Ph.D. student at the Institute Jean le Rond d'Alembert/Sorbonne Université in the field of heritage room acoustics. His fields of interest include immersive audio, virtual and augmented reality, room acoustics, and binaural hearing.

•

Michael Mihocic received a graduate engineer degree (Diplom-Ingenieur (FH)) in electronics, with a focus on audio and video engineering in 2005 and an M.Sc. degree in innovation and technology management in 2014. In 2005, he joined the Acoustics Research Institute (ARI), where he is the main laboratory technician. Michael Mihocic is the secretary of the Austrian section of the AES.

•

Markus Noisternig studied audio and electrical engineering at the University of Technology (TUG) and computer-music composition at the University of Music and Performing Arts (KUG). He obtained his Ph.D. degree in acoustics, audio signal processing, and psychoacoustics from KUG. He is a researcher at the STMS Lab of the Institute for Research and Coordination in Acoustics/Music (IRCAM), the French National Centre for Scientific Research (CNRS), and the Sorbonne University (SU), where he is also head of the Music Research Group. His fields of interest include acoustics and audio signal processing, immersive audio for virtual and augmented reality, and room acoustics. Markus is member of the AES, ASA, German Acoustical Society (DEGA), French Acoustics Association (SFA), and council member of the TC for audio signal processing of the European Acoustics Association (EAA).