
I-Simpa Documentation

Release 1.3.4

Picaut J.

Oct 15, 2019

I-SIMPA PRESENTATION

1	I-Simpa overview	3
2	I-Simpa features	5
2.1	Geometry	5
2.2	Surface material	6
2.3	Sound sources	6
2.4	Acoustic receivers	7
2.5	Acoustic calculation	7
2.6	Other features	8
3	Study of a teaching room	11
3.1	Scene creation	12
3.2	Calculation	13
3.3	Results	14
3.4	Learn more	15
4	Study of the Elmia hall	17
4.1	Geometry import	18
4.2	Define sound sources	20
4.3	Define punctual receivers	21
4.4	Define a plane receiver	21
4.5	Define surface materials	21
4.6	SPPS calculation	21
4.7	Exploring the results	22
5	Study of an industrial room	25
5.1	Import a geometry	25
5.2	Define a machine as a sound source	26
5.3	Duplicate the machine	27
5.4	Define a scene volume as a fitting zone	27
5.5	Define a parallelepipedic volume as a fitting zone	28
5.6	Define surface materials	29
5.7	Insert a line of punctual receivers	30
5.8	Define a plane receiver	31
5.9	SPPS calculation	31
5.10	Evaluate the effect of changing one surface material	32
6	(Un)Setup	35
6.1	Installation	35
6.2	Un-installation	35

7	GUI Presentation	37
8	Menus	39
8.1	Menu ‘File’	39
8.2	Menu ‘Edition’	41
8.3	Menu ‘Simulation’	41
8.4	Menu ‘View’	42
8.5	Menu ‘Windows’	43
8.6	Menu ‘Help’	44
9	Toolbars	45
9.1	Toolbar ‘Project’	45
9.2	Toolbar ‘View/Camera’	45
9.3	Toolbar ‘Meshing’	45
9.4	Toolbar ‘Selection’	46
9.5	Toolbar ‘Simulation’	46
10	‘Project’ window	47
10.1	Presentation	47
10.2	‘Scene’ tab	49
10.3	‘Calculation’ tab	57
10.4	‘Results’ tab	62
11	‘Properties’ window	67
12	‘Console’ window	69
13	‘Main’ window	71
14	Project database	73
14.1	Directivities	73
14.2	Materials	73
14.3	Spectrum	76
15	Import options	77
15.1	‘Average model remesh’	77
15.2	‘Repair model’	78
15.3	‘Surface meshing’	78
15.4	‘Keep existing group’	78
16	Recommendations to import a 3D scene	79
17	Define and use spectrum	81
17.1	General information	81
17.2	Definition	81
18	Define a position using the pointer on the 3D view	83
19	Surface selection	85
20	Manipulating sources and receivers	87
20.1	Rotate a group of sources/receivers	87
20.2	Translate a group of sources/receivers	87
20.3	Create a grid of receivers	87
21	Using directivity	89

22	Creating charts	91
23	Chart Options	93
24	Presentation	95
25	Using SPPS within I-Simpa	97
25.1	Frequency bands	97
25.2	Meshing	97
25.3	Calculation parameters	98
25.4	Computational time optimization	99
26	General principle	101
26.1	Sound particle concept	101
26.2	Sound particles vs. sound rays	101
26.3	Code principle	102
26.4	Spatial meshing	102
26.5	Temporal meshing	103
27	Modelling of physical phenomena in SPPS	105
27.1	Sound source modelling	105
27.2	Acoustic propagation modelling	109
27.3	Wall modelling	119
27.4	Calculation of sound levels at observation points	128
28	Presentation	133
29	Using TCR within I-Simpa	135
29.1	Meshing	135
29.2	Calculation parameters	136
30	Standards	137
30.1	Acoustics	137
30.2	Other	137
31	Glossary	139
32	References	141
33	Indices and tables	143
	Bibliography	145
	Index	147



I-Simpa is a [new generation of noise prediction software for research, engineering and education](#).

I-Simpa is a graphical user interface (GUI) developed to host three-dimensional numerical codes for the modeling of sound propagation in complex geometrical domains, and propose many [features](#). Although I-Simpa is well adapted for energetic models (ray-tracing, sound-particle tracing, theory of reverberation...), it can be extended to use ondulatory approaches.

I-Simpa is distributed with two codes (TCR based on the classical theory of reverberation and SPSS based on particle tracing approach). Classical applications are room and building acoustics, environmental noise and industrial noise, but it can be easily extended to other applications concerning the sound propagation in 3D environments (interior of vehicle, sound in cavities...).

This is the **official I-Simpa User Guide**, including the SPSS and the TCR numerical codes (embedded within the I-Simpa software).

- for **more information** on I-Simpa, [visit the official I-Simpa website](#)
- for **contributing to I-Simpa** from the source code, follows the [instructions](#)
- for **contacting the development team** uses the email: i-simpa@ifsttar.fr

Important:

- The guide is currently not complete. Additions are underway.
- If the present documentation is the 'Offline documentation' you may refer to the online version at <http://i-simpa-wiki.readthedocs.io/en/latest/> for an up-to-date documentation.

Note:

- Some illustrations may refer to previous versions of I-Simpa.
 - Depending on your OS, screenshots may differ.
 - Some texts and translations in I-Simpa may have changed.
 - If you observe some mistakes or errors, please contact us at i-simpa@ifsttar.fr.
 - You can also [contribute to the documentation](#).
-

Note: The official documentation is available in English only.

I-SIMPA OVERVIEW

I-Simpa alone is **not a calculation software**, but is equivalent to a pre and post-processor for acoustic codes. One or more numerical codes must be added in order to obtain a fully functional system.

At the present time, **I-Simpa is distributed with two codes:**

- **TCR code** based on the classical theory of reverberation
- **SPPS code** based on particle tracing approach

The resume of **all code characteristics** are given [here](#)

Classical applications are room and building acoustics, environmental noise and industrial noise, but it can be easily extend to other applications concerning the sound propagation in 3D environments (interior of vehicle, sound in cavities...).

I-Simpa has been initially developed as a research tool (i.e. for research laboratories), but can also be a very efficient tool for a professional/commercial use, as well as for education.

Functionnal Elements and components are organized in tree structures, to easily access to all information, parameters and properties. Many features are proposed for helping users

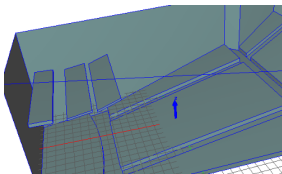
Open system All information and data are organized in spreadsheets that can be displayed, exported, copied... User can use their own software for processing data and graphical representation.

Open software User can integrate their own numerical propagation code, develop their own functionalities within the interface, using Python scripts... All have been done in order you can extend I-Simpa easily.

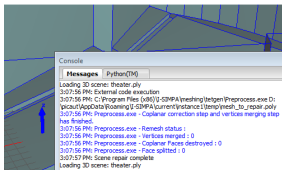
Open source project The source code of I-Simpa is freely available on the GitHub platform (<https://github.com/Ifsttar/I-Simpa>). There are many way for the community to contribute to the development of I-Simpa.

I-SIMPA FEATURES

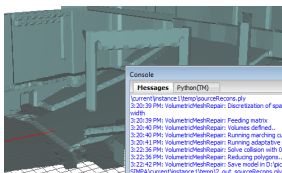
2.1 Geometry



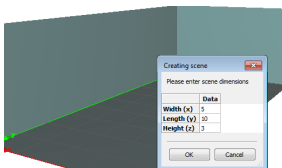
Import 3D files Import 3D files with common extensions: 3DS (3D Studio), STL (stereolithography), PLY (Stanford) and POLY (TetGen).



Geometry correction Automatic tools for model corrections and repairs during file importation. Model detection error.

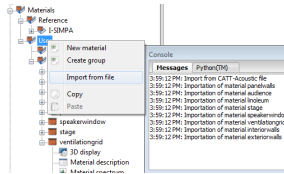


Geometry approximation Algorithm for model approximation by the marching cube method. This can be useful for original 3D scene with many problems.

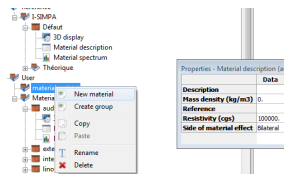


Parallelepipedic geometry Built-in design of parallelepipedic model. Just gives the dimensions of the scene.

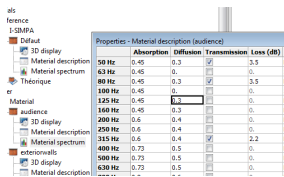
2.2 Surface material



Import data material Import data from file (import from CATT-Acoustic™ and Odeon room acoustics software).

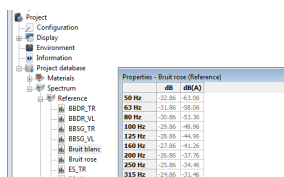


Reference and User material database Use the built-in database (Reference) or create your own materials (User database). Organize and copy material between (sub-)groups of material.

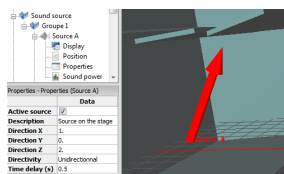


Material properties Define acoustic parameters by frequency band: absorption, transmission loss, scattering, scattering reflection law. Add material information (description, physical parameters. . .).

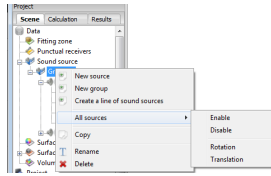
2.3 Sound sources



Reference and User spectrum database Use the built-in database (Reference) or create your own source spectrum (User database).

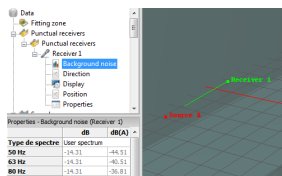


Sources properties Define source properties: spectrum, position, directivity, delay, activate/desactivate sources.

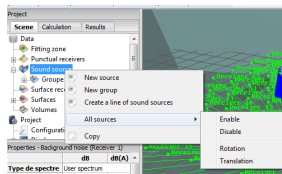


Sources manipulation Organize and copy sources between (sub-)groups of material. Translation and rotation of (sub-)groups of sound sources.

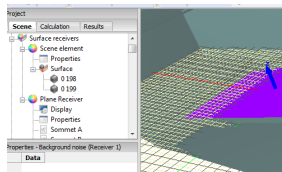
2.4 Acoustic receivers



Punctual sound sources Defines source properties: position, directivity, orientation to a point or (with a dynamic link) to a source, background noise.

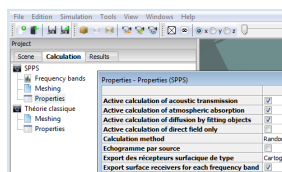


Receivers manipulation Organize and copy receivers between (sub-)groups of material. Translation and rotation of (sub-)groups of receivers.

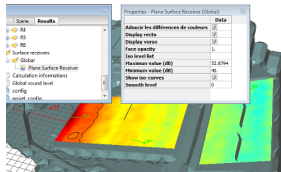


Surface receivers Definition of a surface receiver by selecting surfaces of the 3D model or by creating cut planes.

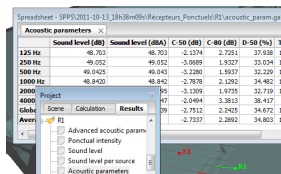
2.5 Acoustic calculation



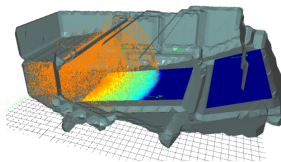
Calculation code Built-in acoustic calculation code: sound particle tracing (SPPS); Classical theory of reverberation for room acoustics (TCR).



Sound map Noisemap (surface receivers) in stationary stationary state, time-varying state, time-cumulated state. Iso-contours.



Acoustics parameters *EN ISO 3382-1* room acoustics parameters: Reverberation times, Clarity, Definition, Early support, Centre time, Strength, Early Lateral Energy Fraction, Late Lateral Sound Level.

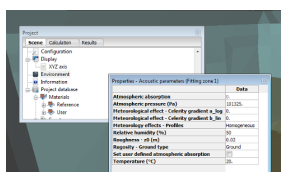


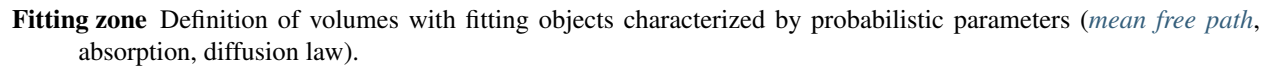
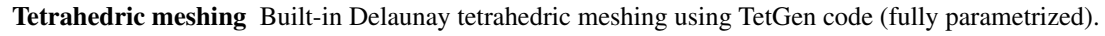
Animation Intensity vectors animation for punctual receivers. Ray-tracing and particle-tracing animation.

2.6 Other features

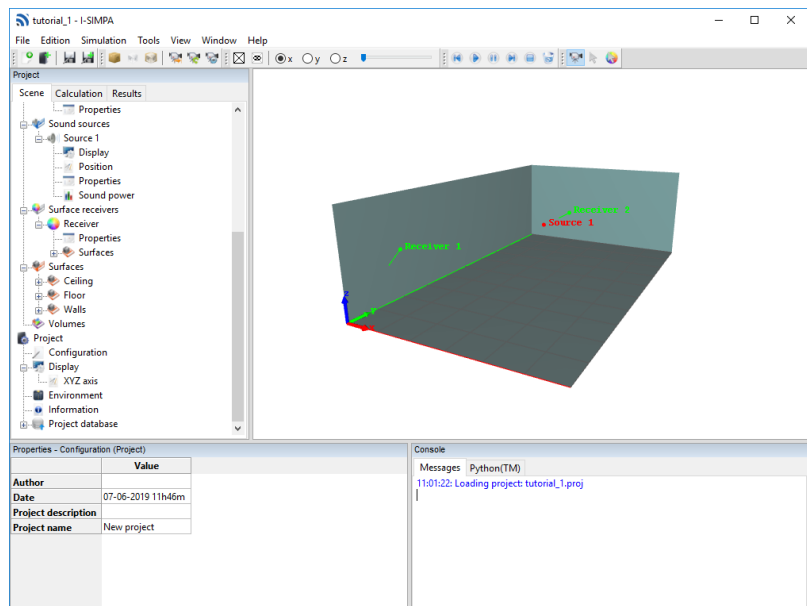
```
4 from libsimpa import *
5
6
7 def GetMixedLevel(folderwid):
8     #folder devient l'objet dossier
9     folderwid = folderwid
10    #dans un tableau on place les indices des
11    recplist=folder.getallElementtype(ui.el)
```

Python™ scripting Extension of I-Simpa functionalities using Python scripts. Creations of Toolbox.





STUDY OF A TEACHING ROOM



This tutorial concerns the simple study of a teaching room. The geometry is rectangular of size (6m x 10m x 3m). The goal of this tutorial is to:

- manipulate face elements
- manipulate surface material
- manipulate punctual receivers
- manipulate omnidirectional sound sources
- perform calculations with the TCR and SPPS codes
- display results

Resources for this tutorial are located in the following folder:

```
<I-Simpa installation folder>\doc\tutorial\tutorial 1
```

This folder contains several files:

- tutorial_1.proj: I-Simpa project of the present tutorial (with TCR and SPPS results)
- Screenshot_tutorial_1.PNG: I-Simpa screenshot of tutorial 1
- additional PNG images (screenshots)

3.1 Scene creation

3.1.1 Create a new geometry

The first step is to create the geometry. As we consider a simple rectangular geometry with the same material of each main face, the built-in option for creating a rectangular room can be used.

Note: Remember that for more complicate room shapes or for rectangular room with different materials on a same main face, user will have to create the geometry with a CAD software.

1. **Start** I-Simpa
2. When starting I-Simpa, the default project is the last saved project. To clear I-Simpa, **select** 'New project' in the File menu or **click** on the 'New project' icon on the 'File' toolbar to clear I-Simpa
3. In the menu file, **select** 'New scene'.
4. **Enter** the given size of the room (width: 6m, length: 10m and height: 3m), then **click** on 'OK'.
5. The 'Loading 3D scene' windows appears. Leave the default values (you can also deselect all options; this will not change the result) and **click** on 'OK'.

3.1.2 Define surfaces

The second step is to affect one material for each main faces. Here, we will consider 3 surface groups: the 'Floor', the 'Walls' and the 'Ceiling'.

1. **Select** the mode 'Face selection' on the toolbar 'Selection'
2. **Hold** the CTRL key and **double-click** on the floor in the '3D view':
 - this will select all co-planar faces (i.e. both triangular faces of the floor);
 - the 2 corresponding elements will be underlined in the 'Surface' folder of the tree 'Data';
3. **Uses** the right click on one of the 2 selected 'Surface' elements in the surfaces group in the Scene tree: a contextual menu will appear, and **Select** 'Send to a new face group'
4. a new surface group, named 'group', is created.

Note: A new group is created with the default name 'group'. Group surfaces can have the same name, , since each group is identified with its own internal Id.

5. **Right click** on the new corresponding group and **Select** 'Rename' to rename the group with the name 'floor' (you can also use the Left click on the group name to change the name)

Tip: Instead of using the 'Face selection' mode, you can create 2 new groups ('Floor' and 'Ceiling' for example), select the corresponding surfaces within the 'Group' folder and send them to a new face group. You can also directly move the corresponding surfaces to their corresponding surfaces group using the pointer.

6. **Once the surface groups are defined, Left click on a group and select a material in the Material list element in the Property**
 - Ceiling: '30% absorbing'

- Floor: '10% absorbing'
- Walls: '20% absorbing'

3.1.3 Define a sound source

1. **Right click** on the 'Sound sources' root folder in the Data tree of the Scene tab and **Select** 'New source'. A new source is created.
2. **Select** the 'Position' element and gives the values of the source coordinates (3, 5, 1.8).
3. **Select** the 'Sound power' element and **Choose** the 'White noise' spectrum in the Spectrum list (it should be the default spectrum).
4. In the 'Global' element of the 'Sound power' properties, **Defines** the value '80' for the 'Global' value of the 'Lw' tab element. This will define a global sound power of 80 dB.

3.1.4 Define a surface receiver

1. **Right click** on the 'Surface receiver' root folder in the Data tree of the Scene tab and **Select** 'New scene receiver'. A new surface receiver is created.
2. In the 'Surfaces' root folder, **Select** the 'Floor' group (**Left click** and **Hold**) and move it in the 'Surfaces' element of the surface receiver. It will define a surface receiver on the room floor.

3.1.5 Define two punctual receivers

1. **Right click** on the 'Punctual receiver' root folder in the Data tree of the Scene tab and **Select** 'New receiver'. A new punctual receiver is created.
2. **Select** the 'Position' element and define the coordinates (x,y,z) of the punctual receiver. You can choose (1.,1.,1.8) for example.
3. **Repeat** the two first step for the second receiver at location (3.,7.,1.8) for example.

3.2 Calculation

3.2.1 TCR code

1. **Select** the 'Meshing' element in the 'Classical theory of reverberation' code.
2. **Enable** the 'Surface receivers constraint'.
3. In the 'Surface receivers constraint (m²)' element, **Define** the value '0.1'. This allows to resample the corresponding surfaces for the surface receiver. In the other hand (without defining a surface receiver constraint), only the two initial surface element of the floor will be used as receiver, leading to a very poor definition.
4. **Right click** on the 'Classical theory of reverberation' and **Select** 'Run calculation' to start the simulation.

3.2.2 SPPS code

- 1-3. **Repeat** the same 1-3 steps as for the TCR code.
4. **Right click** on the 'SPPS' and **Select** 'Run calculation' to start the simulation.

3.3 Results

Note: The ‘Results’ tab correspond exactly on the image of the data on the hard disk. You can open the file explorer by a **Right click** on a result folder and by selecting ‘Open folder’. You can delete a folder or a file, either using the file explorer or the ‘Delete’ action (**Right click** on the folder/element in the results folder) within I-Simpa.

Note: For each new simulation, a specific folder is created in the corresponding simulation folder (*i.e.* SPPS or TCR), with a name that is defined from the starting time and date of the simulation.

Tip: Most of results are displayed in table. You can select cells in the table, right click on the selection and select ‘New diagram’ to create a graphical display of the results. You can also export data to a csv file by selecting ‘Save data as...’.

3.3.1 TCR code

1. **Unfold** the ‘Results’ folder in the ‘Classical theory of reverberation’ folder.
2. **Unfold** the ‘Punctual receiver’ folder, and **Double left click** on one element (*i.e.* one of the two receivers that have been created). A new window is displayed, showing the SPL at the receiver for the direct field and the total field (diffuse field + direct field) according to the Sabine and Eyring formulae.
3. **Unfold** the ‘Total field’ folder (Sabine or Eyring), **Select** the surface receiver, **Select** the frequency band and **Right click** on the ‘Sound level’ element and **Choose** ‘Load animation’, then ‘Cumulating sound level’. It displays the sound map for the corresponding surface receiver on the 3D view. You can remove the colormap on the 3D view by clicking on the ‘Dash’ icon on the ‘Simulation’ toolbar.
4. **Double left click** on the ‘Main results’. It opens a new display with some general results for the whole room (global reverberation time, SPL of the diffuse sound field and equivalent absorption according to the Sabine and Eyring formulae)

3.3.2 SPPS code

1. **Unfold** the ‘Results’ folder in the ‘SPPS’ folder.
2. **Unfold** the ‘Punctual receiver’ folder, **Unfold** the folder for a given punctual receiver and **Left click** on one element to display the corresponding results (for example: ‘Sound level’). A new window is displayed, showing the results.
3. **Right click** on the ‘Sound level’ element and **Select** ‘Calculate acoustics parameters’. It opens a new window: **Keep** the default values, and **Select** ‘OK’. It creates two new elements ‘Schroeder curves’ and ‘Acoustic parameters’. **Select** the new elements to display the corresponding results in a spreadsheet.
4. **Right click** on the ‘Advanced sound level’ element and **Select** ‘Calculate acoustics parameters’. It opens a new window: **Keep** the default values, and **Select** ‘OK’. It creates a new element ‘Advanced acoustic parameters’. **Select** the element to display the corresponding results in a spreadsheet.
5. **Unfold** the surface receiver folder, **Select** the frequency band and **Right click** on the ‘Sound level’ element and **Choose** ‘Load animation’, then **Choose** an option (for example: ‘Instantaneous sound level’). It displays the (animated) sound map for the corresponding surface receiver on the 3D view. You can remove the colormap

on the 3D view by clicking on the ‘Dash’ icon, or if required, pause/resume/stop... the animation using the ‘Simulation’ toolbar.

- **‘Instantaneous sound level’** Shows the time varying sound level (animation)
- **‘Cumulative instantaneous sound level’** Shows the time varying sound level by cumulating all previous time step (animation)
- **‘Cumulating sound level’** Shows the cumulating sound level (no animation)

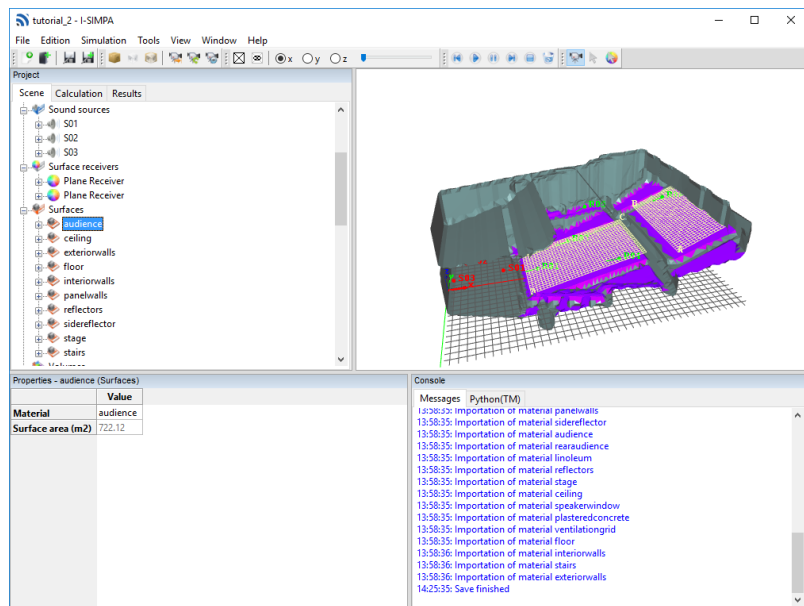
Note: You may change the [properties of the ‘Sound level’ element](#), for example the ‘Maximum value (dB)’ and the ‘Minimum value (dB)’ for displaying the colormap in a good way.

5. **Unfold** the surface receiver folder, **Select** the frequency band and **Right click** on the ‘Sound level’ element and **Choose** ‘Acoustic parameters’, then **Choose** a given parameters from the list. It creates one or more new elements (acoustic parameters). **Right click** on a given acoustic parameter and **Select** ‘Load animation’ to display the corresponding parameter on the 3D view.

3.4 Learn more

- Change frequency bands of calculation in ‘SPPS’ calculation [by setting the ‘Frequency bands’ element](#)

STUDY OF THE ELMIA HALL



In this example, we are interested in the case of a theater. This hall, the Elmia hall, has been the subject of measurements and a comparative study of several room acoustics software (Round Robin II). The user can refer to the following reference for more information: [A Comparison of Room Simulation Software - The 2nd Round Robin on Room Acoustical Computer Simulation](#), Bork Ingolf, Acta Acustica united with Acustica, Volume 86, Number 6, November/December 2000, pp. 943-956(14).

The goal of this tutorial is to:

- manipulate face elements
- manipulate surface material
- manipulate punctual and plane receivers
- manipulate omnidirectional sound sources
- generate an approximate 3D model
- perform calculations with the SPPS codes
- display results

Resources for this tutorial are located in the following folder:

```
<I-Simpa installation folder>\doc\tutorial\tutorial 2
```

This folder contains several files:

- `tutorial_2.proj`: I-Simpa project of the present tutorial (without Results)
- `Elmia.ply`: 3D room geometry
- `elmia_stage.jpg`: room picture
- `elmiahall.jpg`: room picture
- `speaker_s1.jpg`: room picture
- `elmia_coordinates.xls`: coordinates of the 3D scene corners
- `material_catt.txt`: surface material definition (CATT-Acoustic software format)
- `Auswertung_Phase_II.xls`: All information (software results, measurements, comparisons) of the Round Robin 2 (source: PTB file at <https://www.ptb.de/cms/ptb/fachabteilungen/abt1/fb-16/ag-163/round-robin-in-room-acoustics.html>)
- additional PNG images (screenshots)

Important: If not already done, we suggest you to follow the two previous tutorials, before the present tutorial:

- [Study of a teaching room](#)
-

4.1 Geometry import

The 3D model that we have at our disposal presents numerous irregularities (surfaces in intersection in particular). According to the calculation code used, such as the SPPS code, it is possible that this geometry can not give rise to calculations. In this case, the use of the advanced model correction tools proposed by I-Simpa, at the scene importation, makes it possible to create a new model.

4.1.1 Import the ‘bad’ 3D model

The ‘bad’ 3D model is already divided into 10 surface groups corresponding to the materials to be combined (audience, ceiling, floor, stages, stairs...). The first step is to load this model without performing geometric approximation processing in order to recover the existing surface groups:

1. **Create** a new project.
2. In the main menu, **Select** ‘File’ and ‘Import new scene’ and **Select** the `Elmia.ply` file located in resource folder. The “Loading 3D Scene” window opens.
3. **Uncheck** all options in the window and **Select** ‘OK’. The model is displayed in the 3D view, some misaligned faces give the impression that surfaces are missing. You can notice that 10 groups of surfaces are visible in the tree of the ‘Scene’ tab. This first operation was only intended to load these groups of surfaces in order to automatically re-associate each surface from the model that will be generated in the next task.

4.1.2 Create an approximated 3D model

1. **Choose** ‘Import new scene’ in the ‘File’ menu and **Select** the same 3D scene `Elmia.ply`. The “Loading 3D Scene” window opens.
2. **Select** the ‘Average model remesh’ option and **Set** ‘1.’ (1 m) for the parameter ‘Association maximum distance’. Valid with ‘OK’. The ‘Mesh correction computation’ window is displayed.

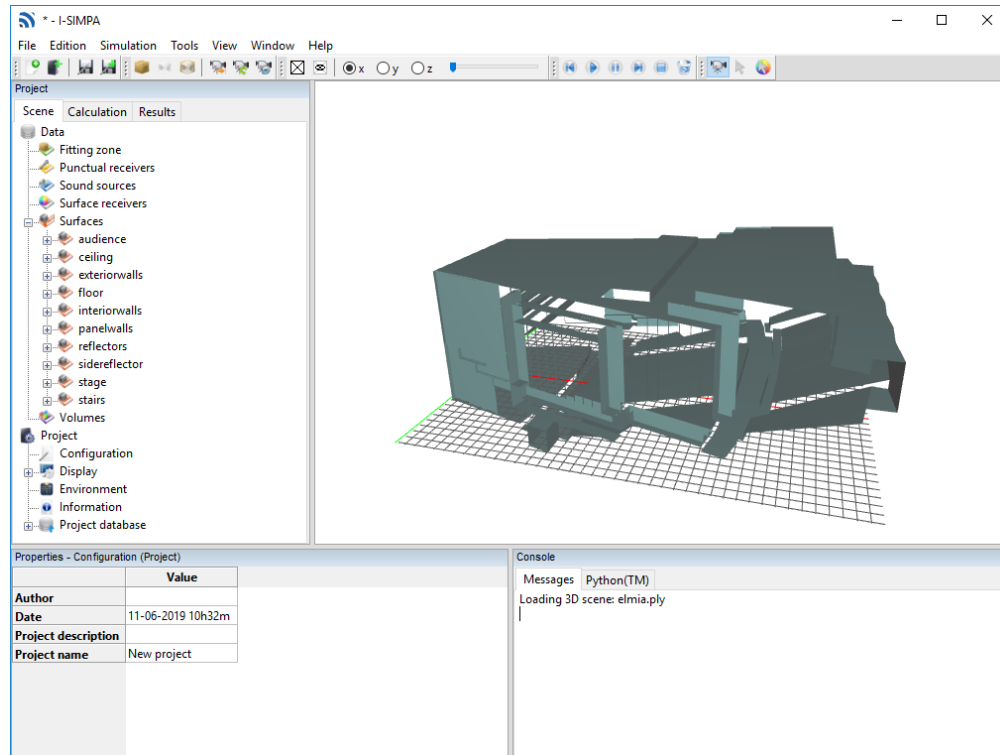


Fig. 1: Import original geometry in I-Simpa

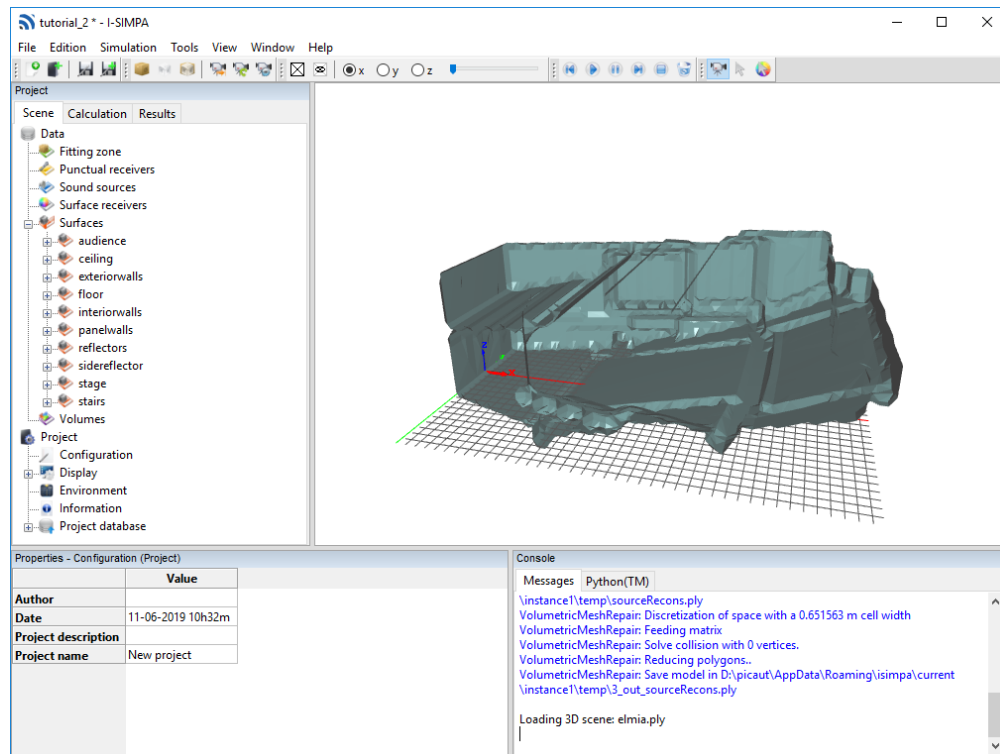


Fig. 2: Generate approximate geometry in I-Simpa

3. **Set** to 6 for the ‘Model solving’ parameter. Using this value, the original 3D scene will be filled with 2^6 cubes and **Valid** all options with ‘Next’. A new window is displayed, showing all closed 3D volume in the scene.
4. **Select** the volume that corresponds to the interior volume of the scene. In the present example, select the volume ‘9624 m³’ and valid with ‘Finish’. A new 3D scene, an approximate scene, is created and displays in the 3D view.

Warning: The Average model remesh operation can only be used simultaneously on a single volume of the scene. If the scene contains several closed volumes, the user must retain only one.

5. To verify that the model correction has been effective, you can generate a meshing of the scene: in the ‘Calculation’ tab, **Open** the properties of the ‘Meshing’ element of the SPPS code and **Uncheck** the ‘Scene correction before meshing’ (since the 3D scene is already optimized). In the ‘Meshing’ toolbar, **Click** on the specific icon to generate the meshing. **Choose** SPPS in the new window that is displayed and **Valid** with ‘OK’. The meshing is displayed on the 3D view: you can check the ‘x’, ‘y’ or ‘z’ boxes in the Meshing toolbar to choose a axis and use the ‘Slider’ cursor to move the meshing.

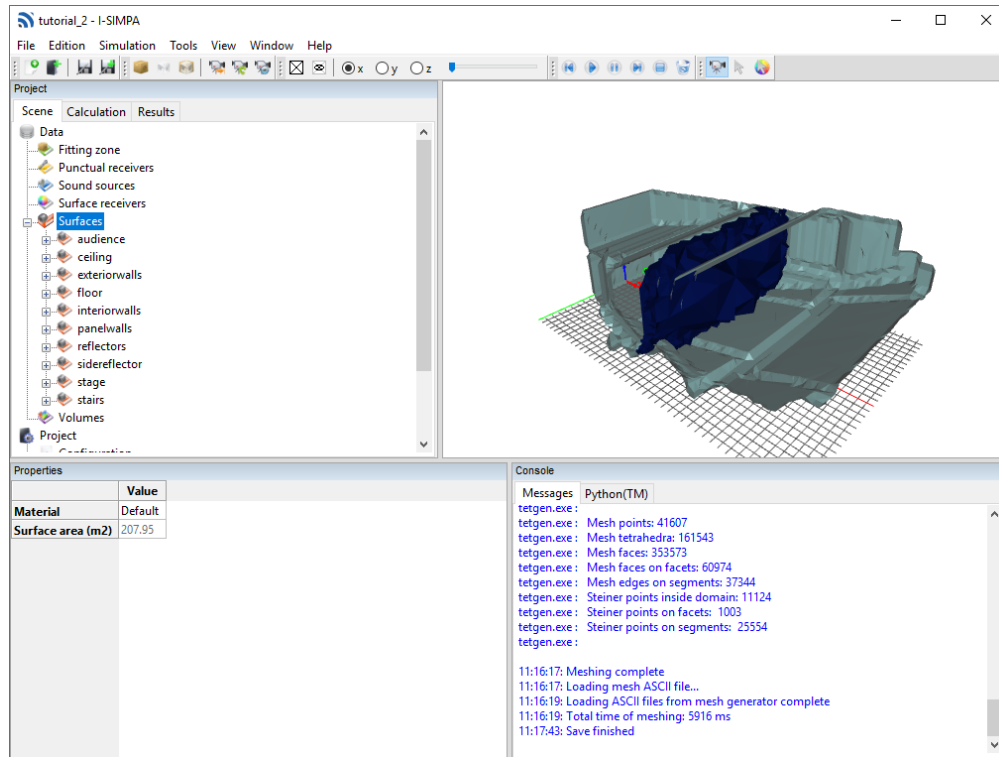


Fig. 3: Generate a meshing in I-Simpa to verify the consistency of the 3D model

4.2 Define sound sources

1. Following the same procedure as for the tutorial ‘Study of a teaching room’ (define sound sources), **Create** three omnidirectional sound sources at S01(8.5,0.0,2.1), S02(3.0,3.5,2.1), S03(1.0,1.0,0.9), with an arbitrary global sound power (for example a white noise @80dB).

4.3 Define punctual receivers

1. Following the same procedure as for the tutorial ‘Study of a teaching room’ ([define punctual receivers](#)), **Create** one or more punctual receivers, for example at R01(13.8,0.0,1.45), R02(12.9,10.5,5.3), R03(19.9,5.1,2.7), R04(25.5;-4.9,4.35), R05(24.8,11.9,5.7), R06(37.8,6.4,8.45).

4.4 Define a plane receiver

The two main locations of the audience will be used to define two plane receivers with a height of 0.50 m from the audience surface:

1. For the first plane, **Click** on the ‘Scene’ tab and **Select** the ‘New plane receiver’ action from the context menu of the ‘Surface receivers’ root folder. A new plane receiver is added and a grid appears in the 3D view.
2. The 3 points defining the plane receiver must then be placed at the level of the two main locations of the audience, defined by the surfaces group “audience”. To do this, **Use** the ‘Define via 3D view’ action of the context menu associated with the ‘Vertex A’, ‘Vertex B’ and ‘Vertex C’ elements of the surface receiver property. **Set** the positions of the vertices of the plane receiver to: A(11.93,-5.66,1.16), B(11.93,5.73,1.13), C(29.56,5.79,5.25).
3. For the second plane receiver, **Repeat** the previous 1-2 steps, with A(33.25,-8.22,6.26), B(33.17,8.58,6.23), C(40.73,8.58,9.47).

.... note:

You can change the location **and** orientation of the plane by changing the location of **one** **or** more vertex (A, B, C). If a part of the plane receiver **is** outside the 3D **scene**, this **is not** a problem **for** the SPPS code.

4.5 Define surface materials

The present study from the Round Robin II, provides the materials in CATT-Acoustic format. The corresponding `.txt` file is available in the same folder as the 3D scene: `material_catt.txt`.

1. **Right click** on the ‘Project/Project Database/Materials/User’ element and **Select** ‘Import from File’.
2. **Select** the corresponding format from the drop-down list and **Open** the file on your computer. A new material folder is created named with the material file name.
3. **Assign** each material to the corresponding surfaces group following the same procedure as for the tutorial ‘Study of a teaching room’ ([define surfaces](#)).

4.6 SPPS calculation

1. Imported materials are only given for octaves between 125 Hz and 4000 Hz. The calculation must therefore be limited to these frequencies. In the ‘Calculation’ tab of the project, **Unfold** the ‘SPPS’ element, then **Right click** the ‘Frequency bands’ element and **Select** the option ‘Automatic selection’, ‘Octave’, ‘Building/Road [125-4000] Hz’.
2. In the ‘Meshing’ element, **Uncheck** the element ‘Scene correction before meshing’.
3. **In the ‘Properties’ element, Set:**
 - a. ‘0.005’ to the ‘Time step (s)’ element;

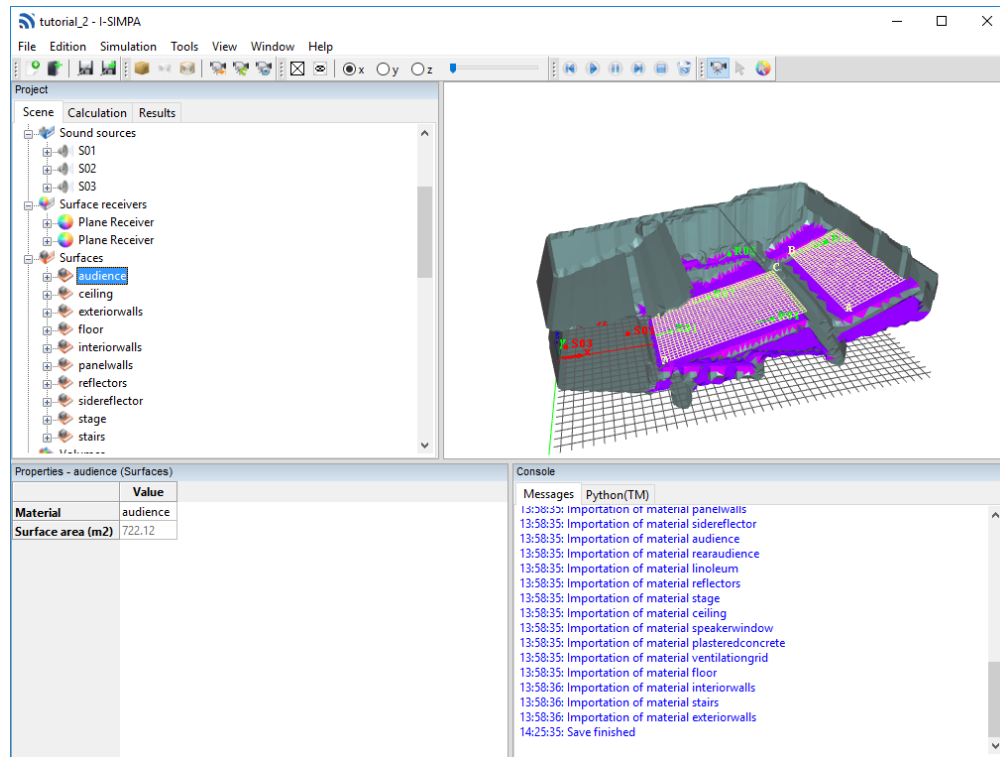


Fig. 4: 3D Model in I-Simpa: sources, punctual receivers, surfaces receivers, material affectation

- b. 'Energetic' to the 'Calculation' element;
 - c. '100 000' to the 'Number of sound particles per source'.
4. **Uncheck** the 'Export surface receivers for each frequency band' (to limit the size of the resulting files).
5. **Right click** on the 'SPPS' and **Select** 'Run calculation' to start the simulation.

Note: The computation time depends mainly of the number of particles, the number of sound sources and frequency bands, as well as the choice of the calculation method ('Energetic' need more computation time than 'Probilistic'). In the present tutorial, using 100 000 particles allows to obtain quick results. However, it should not be enough to obtain detailed results. Increasing the number of particles to 1 million could give more pertinent results.

4.7 Exploring the results

Follow the same procedures as the 'Study of a teaching room' tutorial for [exploring the results](#).

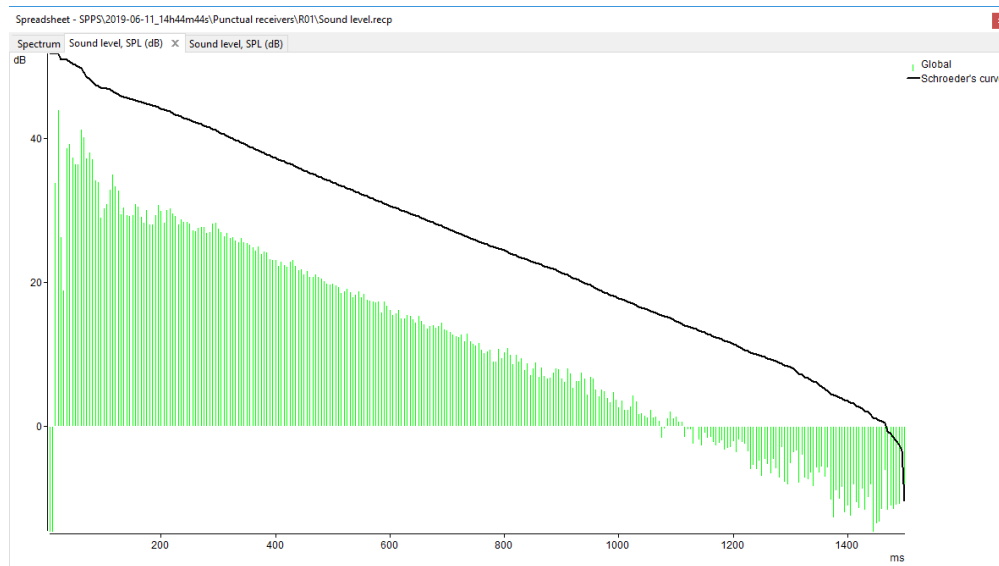


Fig. 5: Echogramm at receiver R01 (1 million particles per source)

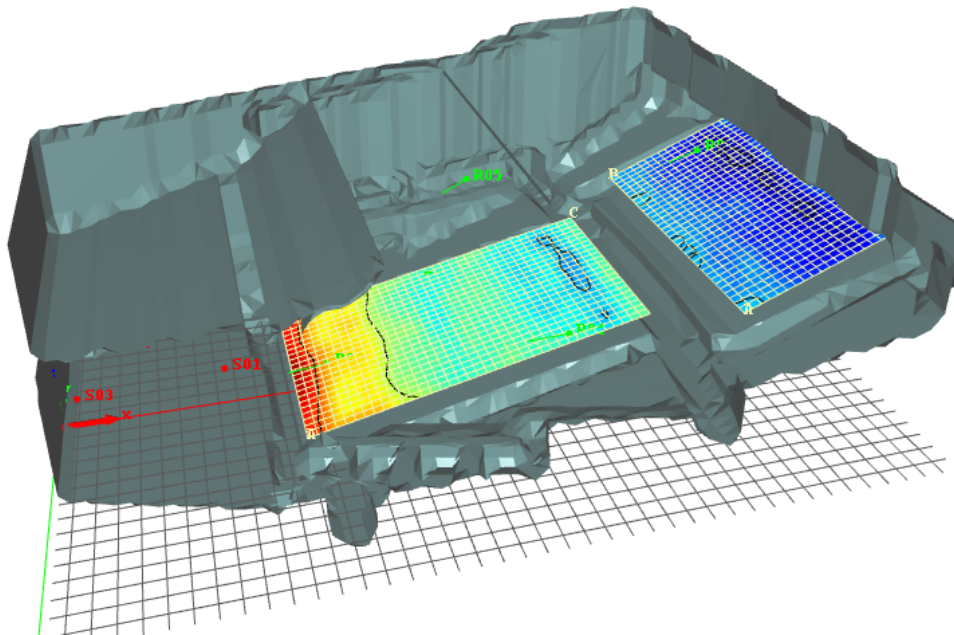


Fig. 6: Noisemap at surface receivers 1 and 2 (1 million particles per source)

STUDY OF AN INDUSTRIAL ROOM

In this example, we are interested in the case of an industrial hall coupled with other rooms. This hall is composed of two identical noisy machines and two fitting zones (zones made of many objects that act as acoustics scatterers). The purpose of this tutorial is to:

- illustrate the creation of machines
- illustrate the creation of fitting zones
- create new surface Materials
- create a line of receivers
- create a plane receiver
- perform calculations using the SPPS code
- study the effect of changing a material on a wall of the industrial hall, on the sound field

Resources for this tutorial are located in the following folder:

```
<I-Simpa installation folder>\doc\tutorial\tutorial 3
```

This folder contains several files:

- `tutorial_3.proj`: I-Simpa project of the present tutorial (without Results)
- `Industrial_hall.ply`: 3D hall geometry
- additional PNG images (screenshots)

Important: If not already done, we suggest you to follow the two previous tutorials, before the present tutorial:

- [Study of a teaching room](#)
 - [Study of the Elmia hall](#)
-

5.1 Import a geometry

The first step is to import the geometry of the project. We are interested here in a large industrial hall coupled to a corridor through a door in a wall, this corridor being itself coupled to another room. We assume that walls and doors allow acoustic transmission.

1. In the 'File' menu, **Select** 'Import new scene', and open the file `LocalIndustrial.ply` located in the corresponding tutorial folder.
2. In the next windows, **Select** 'OK' without changing the default values.

Note: When importing the model, several surfaces groups are created. These groups were defined at the 3D scene creation with the CAD software.

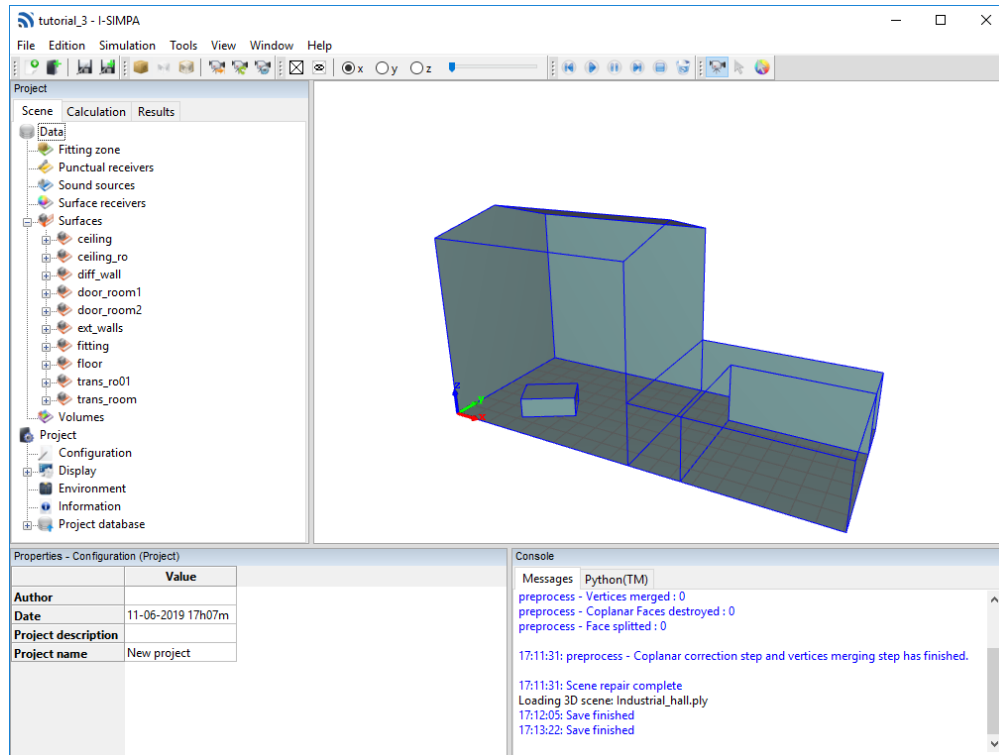


Fig. 1: 3D scene import in I-Simpa

5.2 Define a machine as a sound source

At this step, we want to create the sound sources. In industrial environments, sound sources are often machines or workstations made up of several source points. Since the hall can be made up of several identical machines located at different locations, it is simpler to define a machine in the form of a group of sound sources, which group can then be copied and moved within the room. In our example, we will define machines composed of three sound sources, which will be duplicated. Each sound source will be defined by a pink noise spectrum, with a global level of 80 dB.

1. **Right-click** the 'Data/Sound Sources' element in the 'Scene' tab of the 'Project' windows, and **Select** 'New Group'. By default, the name of the new group is 'Sound sources'.
2. **Rename** the group as the following 'Milling machine';
3. **Right-click** on the 'Milling Machine' group and select 'New Source'. **Perform** this operation three times (to create three sources), named 'Source 1', 'Source 2' and 'Source 3'. It is possible to modify the name of the sources by applying the same method as for the name of a group;
4. For each source, **Unfold** the source properties, then **Click** on the 'Position' item, and **Assign** the following coordinates (x,y,z) for the source 1 (2,7,1.2), the source 2 (2,6,0.6) and the source 3 (1,7,0.75).
5. For each source, **Select** the 'Sound power' element, **Choose** 'Pink Noise', and then **Set** the global level 'Lw' to 80dB.

5.3 Duplicate the machine

The previous machine can then be duplicated and moved into the room.

1. **Right-click** on the ‘Milling Machine’ group and **Select ‘Copy’**;
2. **Right-click** the ‘Sound Sources’ root folder, then **Select ‘Paste’**: an item named ‘Copy of Sound sources’ is created.

Note: The sound sources within the new group have exactly the same properties as the original sources, including the names of the sources. However there is no possible confusion between two sources with the same name, each source having its own internal identifier.

3. **Rename** the group created in ‘Milling machine 2’;
4. Since the new machine is located in the same place as the original machine, it is necessary to move it. To perform a translation of the new machine, **Right-click** on the ‘Milling Machine 2’ group and **Select All sources** → *Translation item* menu;
5. **Set** the translation values in each direction (x,y,z) to [5,-2,0], and **Click** on ‘OK’. The sources group is translated.

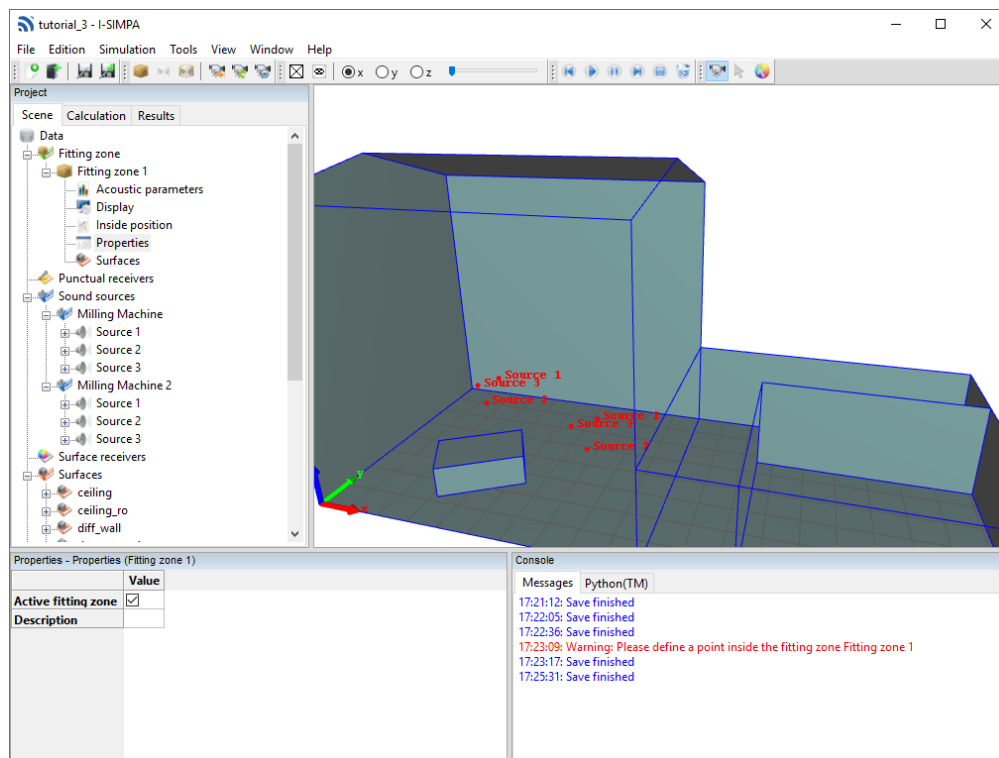


Fig. 2: Sound sources creation (2 identical machines made of 3 punctual sources)

5.4 Define a scene volume as a fitting zone

There are two methods of creating a fitting zone in the room:

- when creating the scene with the CAD software. In this case, a closed volume is modeled in the 3D geometric model. It is then a 'Scene fitted zone'
- either in I-Simpa, in the form of a parallelepipedal zone. It is then a 'Rectangular fitted zone'

In our example, a parallelepipedal zone has already been provided at the creation of the 3D scene, in order to be assigned to a 'Scene fitted zone'.

1. In the 'Scene' tree, **Right-click** on the 'Fitting zone' root folder and **Select** 'Define scene fitted zone';
2. Now you have to define the surfaces of the scene for the corresponding zone. **Select** the surface selection mode on the 'Selection' toolbar; **Hold** Ctrl keyboard key and **Double-click** on each face element of the fitting zone. Each face element is selected in red (or another color depending of the I-Simpa Settings). At the same time, the corresponding face elements are also highlighted in the folder 'Surfaces' of the 'Data' tree.

Note: Holding the Ctrl keyboard key allow to select all coplanar face element by a double click. You can also use the same procedure, but with one single click, to select each face element independently.

3. In the folder 'Surfaces' of the 'Data' tree, **Select** all highlighted face elements, and **drag/drop** them to the 'Surfaces' element of the fitting zone. All face elements are then duplicated in this folder.

Note: In this example, all face elements were already identified in a given folder 'fitting' of the 'Data' tree, because this volume was build when preparing the 3D geometry. In this case, it was not necessary to follow the step 2 of this procedure. One can directly drag and drop all face elements of the corresponding surface folder.

4. One must also select the face elements of the fitting zone that are located on the ground. **Open** the 'floor' surface group and find the two face elements that correspond to the floor of the fitting zone (**Select** each face element of the group and **Identify** the ones that correspond to the fitting zone). Once the two face elements are identified, **Select** them and drag/drop them to the 'Surfaces' element of the fitting zone.
5. **Define** the 'Acoustic parameters' of the fitted zone, with 0.25 for 'Alpha' (absorption coefficient of the obstacles), 0.5 m for 'Lambda' (mean free path in the fitted zone) and 'Uniform reflection'. See the SPSS documentation for more information about acoustic parameters of fitted zones.

Tip: When you define value in a spreadsheet (for example 'Alpha' in the example above), you can duplicate the value in all the column (i.e. frequency bands) by selecting :*Set the same value* → *To the column* after a Right-Click on the corresponding value.

You can also define an average value by setting a value to 'Average' (last row of the spreadsheet), which will define the same value for all the rows (i.e. frequency bands).

5.5 Define a parallelepipedic volume as a fitting zone

One may create a fitting zone as a parallelepipedic volume directly within I-Simpa. Such volume is defined by creating the two opposite corners of the volume. In the following example, one consider a fitting zone made of chairs.

1. **Right-Click** on the 'Fitted zone' element of the 'Data' tree, and **Select** 'Define rectangular fitted zone'. A new fitted zone is created.
2. **Define** the 'Position' element (coordinates) of the opposite corners of the fitting zone, 'Origin volume' and 'Destination volume', as (13,4,0) and (18,1,1.2) respectively.

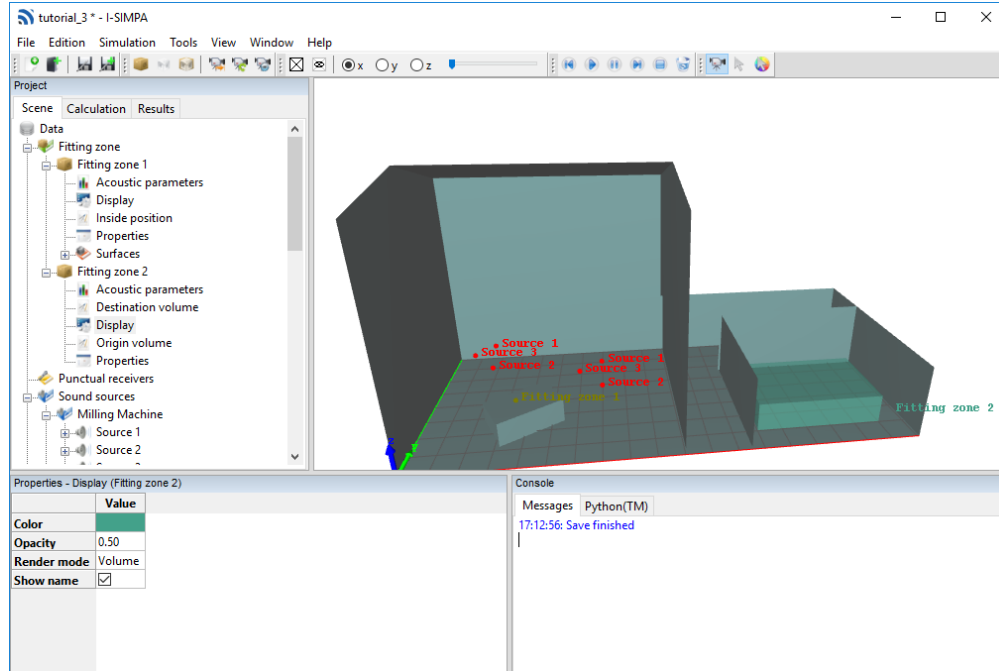


Fig. 3: Fitting zones representation (scene and parallelepipedic fitting zones)

3. **Define** the ‘Acoustic parameters’ of the fitted zone, with 0.15 for ‘Alpha’ (absorption coefficient of the obstacles), 0.3 m for ‘Lambda’ (mean free path in the fitted zone) and ‘Uniform reflection’.

5.6 Define surface materials

User can define specific material for the project.

1. **Unfold** the ‘Materials’ element of the ‘Project database’ element of the ‘Project’ tree and **Right-Click** on the ‘User’ element. **Select** ‘New material’. A ‘New material’ is created. **Rename** the name of the corresponding material to ‘Trans_material’.
2. **Select** the ‘Material spectrum’ element to display the acoustic parameters of the corresponding material. **Set** the values of each parameters according to the following table.

Table 1: Acoustical properties of the ‘Trans_material’ material

Frequency bands	Absorption	Diffusion	Transmission	Loss (dB)	Diffusion law
125 Hz	0.11	0.3	Check	10	Lambert
250 Hz	0.08	0.4	Check	11	Lambert
500 Hz	0.07	0.5	Check	12	Lambert
1000 Hz	0.06	0.6	Check	13	Lambert
2000 Hz	0.09	0.7	Check	14	Lambert
4000 Hz	0.05	0.8	Check	15	Lambert

3. **Go to** the ‘Surfaces’ folder in the ‘Data’ tree. **Select** the ‘trans_ro1’ surface to display the corresponding ‘Properties’ and **set** the ‘Material’ parameter to the ‘Trans_material’ material. **Repeat** this procedure for the ‘trans_room’ surface group.
4. **Create** a new material ‘Open_door’ using the same procedure as for ‘Trans_material’, using the following parameters, and **Set** this material to the ‘door_room1’ and ‘door_room2’ surface groups.

Table 2: Acoustical properties of the ‘Open_door’ material

Frequency bands	Absorption	Diffusion	Transmission	Loss (dB)	Diffusion law
125 Hz	1.0	0	Check	0	Can not be defined
250 Hz	1.0	0	Check	0	Can not be defined
500 Hz	1.0	0	Check	0	Can not be defined
1000 Hz	1.0	0	Check	0	Can not be defined
2000 Hz	1.0	0	Check	0	Can not be defined
‘4000’ Hz	1.0	0	Check	0	Can not be defined

5. For all other surfaces groups, **Set** the material to ‘30% absorbing’ from the ‘Reference materials’ database.

Tip: For all these surface groups, since the surface material is the same, user can move all surface elements to a given surface group and set the material to this surface group only.

Note: The surface elements that have been defined as fitting surfaces (i.e. ‘fitting’ surface group) are consider as perfectly transparent by default. CAUTION

Tip: In order to verify that material have been correctly set, you can display the 3D by selecting the ‘Material’ option in the :View → *Material color* menu.

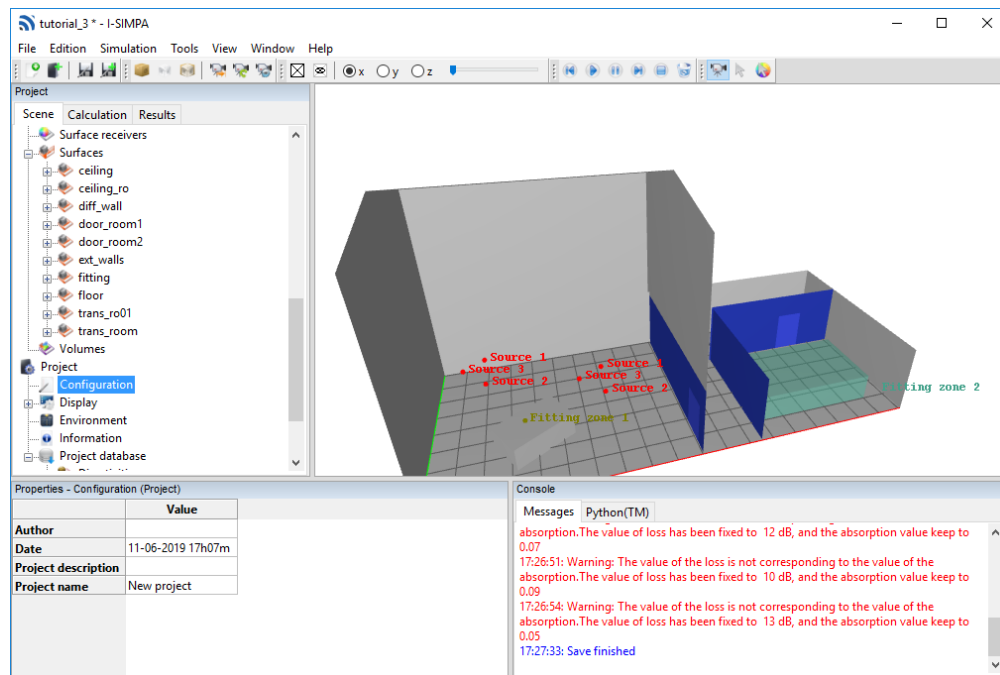


Fig. 4: Representation of material of scene surfaces

5.7 Insert a line of punctual receivers

Instead of creating punctual receivers individually, you can directly create a line (or an grid) of punctual sound sources.

1. **Go to** the ‘Punctual receivers’ folder in the ‘Data’ tree and **Right-Click** on it. Thus, **Select** the ‘Create a receiver grid’ option.
2. A spreadsheet is displayed to define the receiver grid parameters. See [the corresponding documentation for more information concerning these parameters](#). **Set** the following values:

Table 3: Receiver grid parameters

Parameter	Value
‘Col step x (m)’	1
‘Col step y (m)’	1
‘Col step z (m)’	0
‘Number of rows’	5
‘Number of cols’	1
‘Row step x (m)’	2
‘Row step y (m)’	0
‘Row step z (m)’	0
‘Starting position x (m)’	0
‘Starting position y (m)’	1
‘Starting position z (m)’	1.6

Tip: To create a grid of receivers instead of a line of receivers in the example above, change the value of the ‘Number of cols’ parameter to 2 or more.

5.8 Define a plane receiver

1. Following the same procedure than for the [Elmia tutorial](#), **Create** a ‘New plane receiver’, using the default parameters.

5.9 SPPS calculation

1. **Go to** the calculation tab of the ‘Project’ windows, **Unfold** the ‘SPPS’ code and **Select** the ‘Properties’ element. It displays a spreadsheet with all SPPS calculation parameters.
2. **Set** the parameters to the corresponding values (see [the corresponding documentation for more information about these parameters](#)):

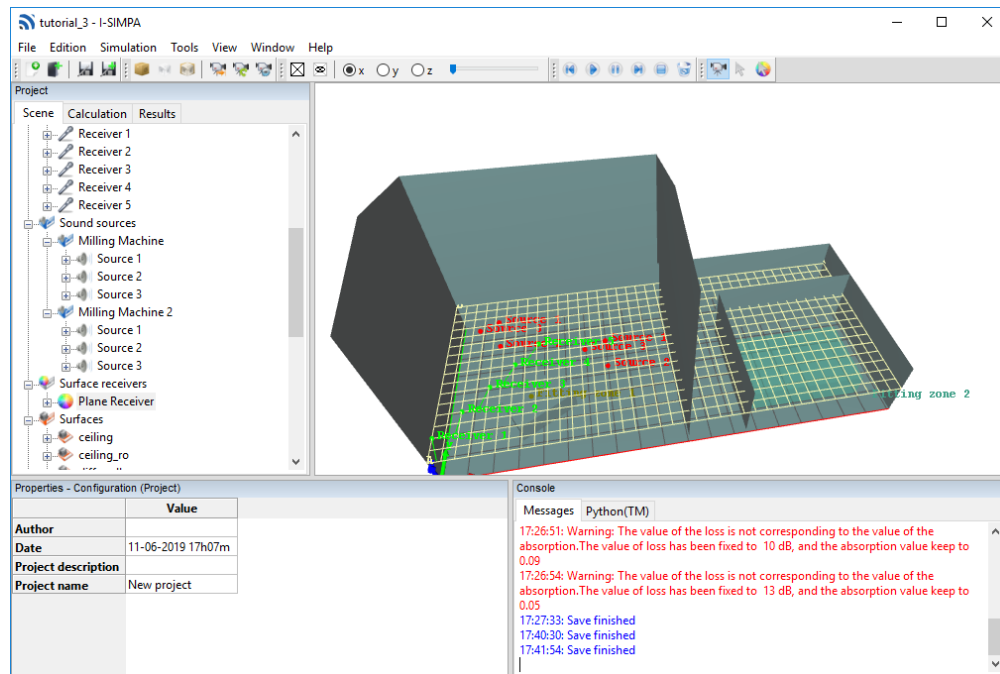


Fig. 5: Line of punctual receivers (green) and plane surface receiver (mesh)

Table 4: SPPS calculation parameters (tutorial 3)

Parameter	Value
'Active calculation of atmospheric absorption'	Check
'Active calculation of diffusion by fitting objects'	Check
'Active calculation of direct field only'	Uncheck
'Active calculation transmission'	Check
'Calculation method'	Energetic
'Echogram per source'	Uncheck
'Export surface receivers for each frequency band'	Uncheck
'Limit value of the particle extinction: ratio 10^n '	5
'Number of sound particles per source'	150 000
'Number of sound particles per source (display)'	0
'Random initialization number'	0
'Receiver radius (m)'	0.31
'Simulation length (s)'	2
'Surface receiver export'	Soundmap: SPL
'Time step (s)'	0.002

3. **Right click** the 'Frequency bands' element and **Select** the option 'Automatic selection', 'Octave', 'Building/Road [125-4000] Hz'.
4. **Run** the calculation code by right-clicking on the 'SPPS' element and selecting 'Run calculation'

5.10 Evaluate the effect of changing one surface material

One can be interesting to evaluate the effect of changing one surface material. I-Simpa includes a tool that allows to compare two results on a surface material. For such operation it is important that the surface receiver is not changed

between the tow simulation.

1. **Create** a new material named 'Absorbing_material' using the following data:

Table 5: Acoustical properties of the 'Absorbing_material' material

Frequency bands	Absorption	Diffusion	Transmission	Loss (dB)	Diffusion law
125 Hz	0.60	0	Uncheck	0	Specular
250 Hz	0.62	0	Uncheck	0	Specular
500 Hz	0.77	0	Uncheck	0	Specular
1000 Hz	0.74	0	Uncheck	0	Specular
2000 Hz	0.80	0	Uncheck	0	Specular
'4000' Hz	0.81	0	Uncheck	0	Specular

2. **Set** the 'Diff_wall' surface to the new material 'Absorbing_material'.
3. **Run** a new SPPS calculation using the same calculation parameters. At this step, two simulation results are available in the 'Results' tab of the SPPS code.
4. **Goto** to *Tools* → *'File results process'* → *'Compare surface receivers'* menu
5. In the list, **Select** the data 'rs_cut' in the *'Surface receiver'* → *'Global'* list element of the surface receivers used as a 'reference' (should be the first calculation, but you can choose the second calculation instead), then **Select** 'Next'.
6. In the list, **Select** the data 'rs_cut' in the *'Surface receiver'* → *'Global'* list element of the surface receiver that must be compared to the reference (i.e. the 'other simulation', should be the second calculation, but you can choose the first calculation instead). At this step, you can defined a name for the difference between the two results ('reference' - 'other simulation') (by default 'Power gain') and create a new data file by checking the corresponding box. Then **Select** 'Finish'. The corresponding result is created in the folder *'Surface receiver'* → *'Global'* of the second result.
7. **Right-click** on the 'Power gain' result and **choose** *'Load animation'* → *'Cumulating sound level'* to display a sound map of the difference between the two results.

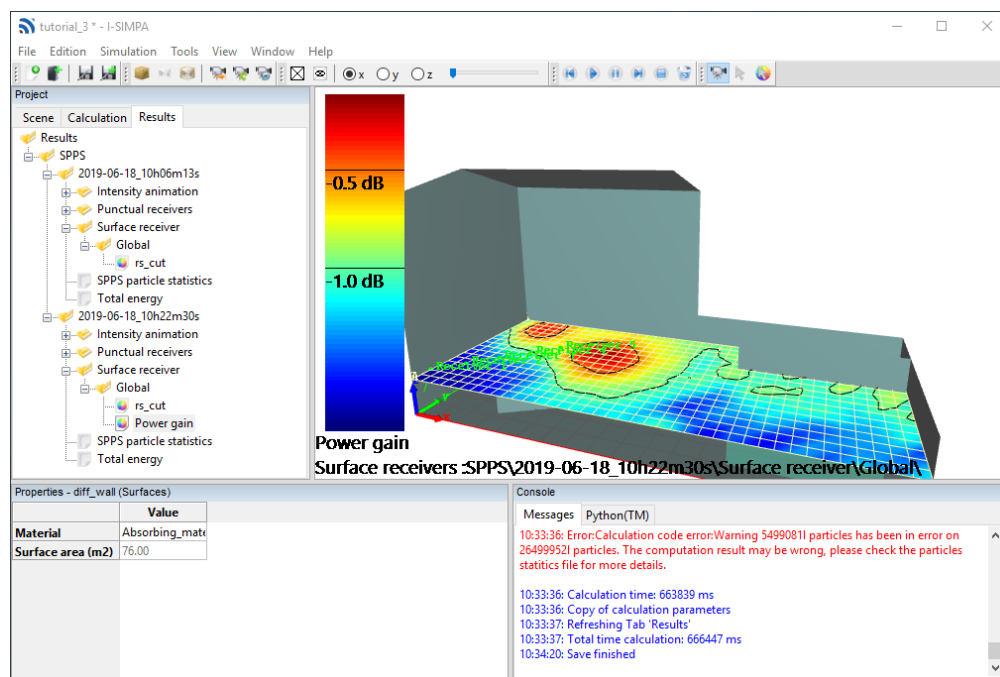


Fig. 6: SPL sound map showing the difference between two calculations.

(UN)SETUP

6.1 Installation

- **Download I-Simpa** from the web site <http://i-simpa.ifsttar.fr>
- **Run** the installation file
- **Select the language** of installation (language can be changed within I-Simpa)
- **Agree** (or not) with the terms of the I-Simpa license agreement
- **Choose the components to install** at the same time (a functional Python installation is required. If not already install on your computer, choose the 'Python' component)
- **Choose to install I-Simpa for you** or for anyone
- **Choose the install location**
- **Wait** during the installation procedure
- **Launch I-Simpa**

The installation procedure will create:

- an icon on the desktop
- a folder in the start menu folder

6.2 Un-installation

Warning: By removing I-Simpa, you will remove all the installation folder (including all files and all subfolders). If you have have modified or installed new files (scripts, graphics, data...), if necessary, please backup all files before removing I-Simpa.

- **Select the uninstall** tool in the start menu of I-Simpa
- **Wait** during the uninstallation procedure
- I-Simpa is removed from your computer

GUI PRESENTATION

The graphic user interface (GUI) is composed by the following components:

- a menu bar ('File', 'Edition', 'Simulation', 'View', 'Windows', 'Help') on the main window
- several independent 'Toolbars' (dockable/undockable)
- a 'Project' window (dockable/undockable), with three tabs 'Scene', 'Calculation', 'Results' (tabs can be re-ordered within the windows)
- a 'Properties' window (dockable/undockable)
- a 'Console' window (dockable/undockable) with two tabs 'Message' and 'Python' (tabs can be re-ordered within the windows)
- a 'Main' window with the 3D view

Windows properties:

- User can **rearrange and resize** all components in the computer screen by selecting and moving the corresponding component.
- You can come back to the **default view** by selecting the option 'Reinitialize interface' in the 'Windows' menu.

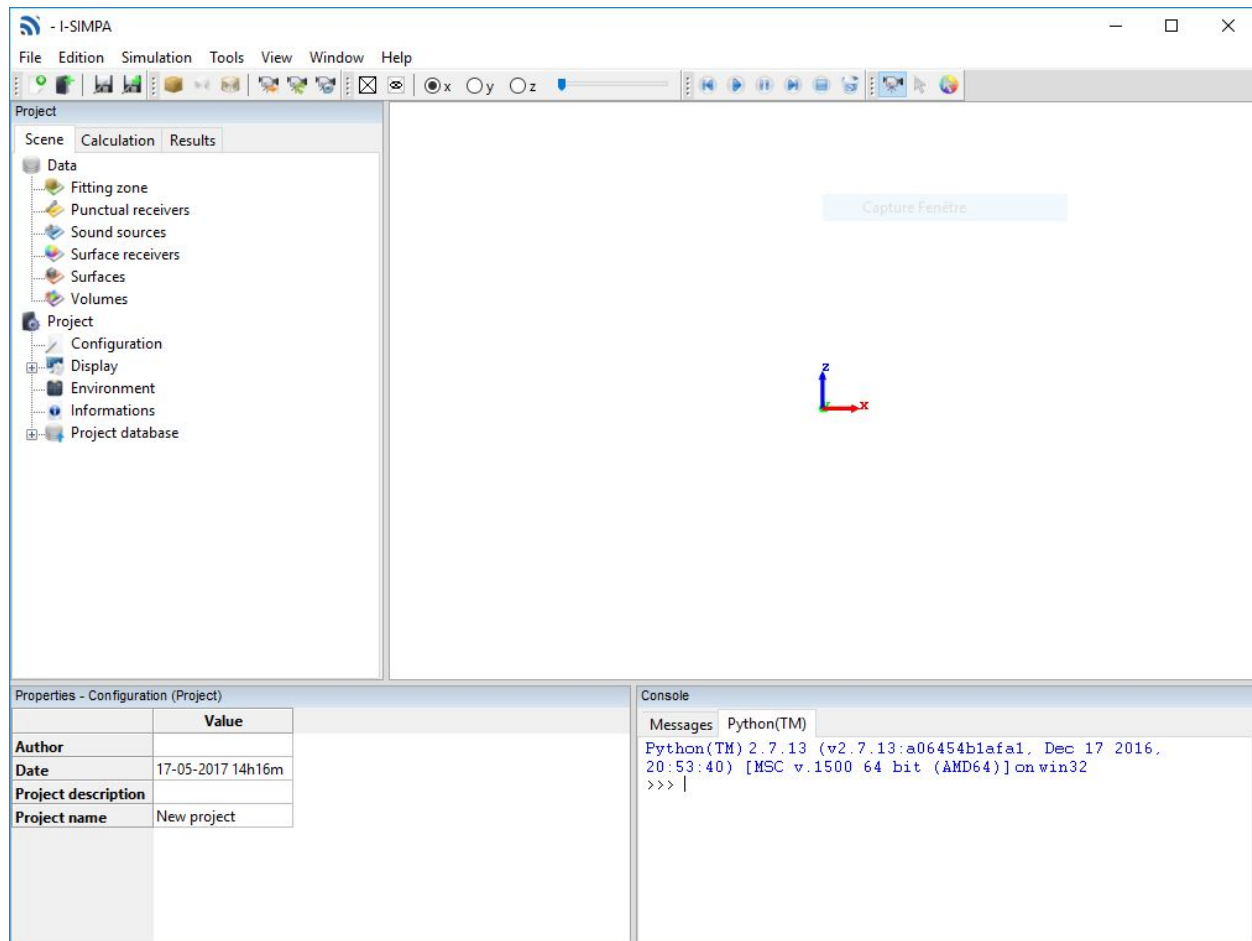


Fig. 1: I-Simpa user interface.

8.1 Menu ‘File’

‘New project’ Create a new project (empty project)

‘Open project’ Open an existing project with the I-Simpa file format *.proj

‘Import new scene’ Import a new 3D geometry in the existing project. Some **Import options** are available at this step.

Import file formats:

- **.3DS**: keep all surface groups and texture
- **.PLY**: ASCII and BIN format, keep all surface groups
- **.POLY**
- **.STL**

Tip: When you import a new scene, except the geometry, all previous data (sources, receivers, materials...) of the current project are unchanged. You can use this option if you want to keep all current data, and just import a new geometry. If you want to remove all data, you need to create a new project.

Note: Numerical codes within I-Simpa (like SPPS) are very sensitive to the quality of the import geometry. Although I-Simpa can correct some defaults during importation of a geometry, it is recommended to pay a particular attention to the creation of the geometry. [Read the recommendations here.](#)

‘Export scene’ Export the current 3D geometry from the current project (without additional data)

Export file formats:

- **.PLY**
- **.MAT.PLY**: same as PLY, with surface groups
- **.POLY**
- **.MESH**
- **.NFF**
- **.BIN**: I-Simpa file format
- **.ASC**

‘New scene’ This option able to create a parallelepipedic scene. This is the simple way to create a 3D scene with I-Simpa. This can be useful for starting with I-Simpa. For more realistic scenes, users have to build 3D scenes using a CAD software, and then, importing the geometry using one the possible import file format (see **‘Import new scene’** in the file menu).

Note: Using the built-in option for creating a rectangular geometry, each main face of the geometry (*i.e.* there are 6 main faces), is split into 2 triangular face elements. This mean that, when defining the acoustical material for each face, you will only have the possibility to affect two different materials for each main face (one on each triangular face). Most of times, users will consider a single material for each group.

‘Save project’ Save the current project with the current filename.

‘Save project as’ Save the current project with a new/given filename.

‘Save copy of project’ Save the current project with a new filename, by default, in the same folder as the original project.

‘Recent projects’ Show the more recent projects. Load a project by selecting the file in the list.

‘Settings’ Defines I-Simpa settings.

- **‘Options’** Defines general options.
 - **‘3D rendering’** Defines 3D rendering options.
 - * **‘Animations’** Defines 3D animation options.
 - **‘Label color’** Color of the legend concerning animations.
 - **‘Particles display’** Show/hide particle animation.
 - **‘Render frequency’** Set frequency of animations (Hz). the speed of animations increases with frequency.
 - **‘Surface receiver display’** Show/hide surface receiver animations.
 - * **‘Legend’** Defines legend options.
 - **‘Legend font’** Legend font (Text Dialog box).
 - **‘Text background color’** Background color of the text legend (Color Dialog box).
 - **‘Text color’** Legend text color (Color Dialog box).
 - **‘Transparent text background’** Enable/disable the transparency of the legend background. If selected, the option ‘Text background color’ has no effect.
 - * **‘General’** Defines general 3D options.
 - **‘Default color’** Default color of an surface element of the geometry (Color Dialog box).
 - **‘Scene background’** Background color of the 3D view (Color Dialog box).
 - **‘Scene lines’** Color of lines for the geometry (Color Dialog box).
 - **‘Selection’** Color of a face selection (Color Dialog box).
 - * **‘Particles’** Defines particles options.
 - **‘Color of particles’** Color of particles (Color Dialog box).
 - * **‘Sound map’** Defines sound map options.

- **‘Iso-lines color’** Color of lines used for displaying iso-contour representation (Color Dialog box).
- **‘Iso-levels color’** Select the colormap for sound map representation. Colormaps can be added by users.
- **‘Main preferences’** Defines the main preferences of the interface.
 - * **‘History’** Defines history options.
 - **‘Keep modification history’** Enable/disable ‘Undo’/‘Redo’ options.
- **‘Language’** Open a dialog box for selecting the interface language.

Note: I-Simpa must be restarted in order to apply the selected language.

- **‘Change application data folder’** Allows to change the default folder for the Application Data
- ‘Exit’** Exit I-Simpa.

Note: When starting I-Simpa, the last project (if available) will be automatically loaded.

8.2 Menu ‘Edition’

Warning: The ‘Undo’ and ‘Redo’ options acts only on action realized in the three ‘Project’. By activating the ‘Undo’ and ‘Redo’ options, I-Simpa can be slowed down. Since, these options are not essential, it is recommended to disable these options (see the ‘Settings’ of I-Simpa).

‘Undo’ Cancel the last actions (recursive option).

‘Redo’ Restore the last cancellation with ‘Undo’.

8.3 Menu ‘Simulation’

‘Play’ Play an animation within I-Simpa (see also Toolbar ‘Animation’). This is functional only if compatible data have been loaded.

‘Pause’ Pause an animation within I-Simpa (see also Toolbar ‘Animation’). This is functional only if compatible data have been loaded.

‘Stop’ Stop an animation within I-Simpa (see also Toolbar ‘Animation’). This is functional only if compatible data have been loaded.

‘Meshing’ Generate a meshing within I-Simpa using the mesh generator *TetGen*.

Note: This option is functional only if the 3D geometry is *compatible* (see [Import options](#)).

8.4 Menu ‘View’

This menu proposes several display modes for the 3D view of the geometry.

8.4.1 ‘Face’

‘Outside’ Show faces oriented outside the geometry. If all faces are well oriented, only the exterior of the geometry should be visible.

‘Inside’ Show faces oriented inside the geometry. If all faces are well oriented, only the interior of the geometry should be visible.

‘None’ Hide all faces.

8.4.2 ‘Material color’

‘Original’ Show faces with their original color/texture (*i.e.* following the CAD exportation)

‘Material’ Show faces with color associated with materials (*i.e.* following I-Simpa material assignation).

8.4.3 ‘Lines’

‘All’ All lines are visible.

‘Contour’ Show lines around co-planar surfaces (co-planar faces are merged, only for the display).

‘None’ Hide all lines.

8.4.4 ‘Hide meshing’

If a mesh has been generated, this option allows to hide the mesh representation in the 3D view.

8.4.5 ‘Reinitialize camera’

Return to the default 3D view.

8.4.6 ‘Camera’

Change the camera mode (use the mouse and/or the arrow keys to change the view).

‘First person’ mode: Users visualize the 3D scene with its own eyes:

- Left clic + motion (left, right, up, down) head motion (left, right, up, down)
- Right clic + motion (left, right) head motion (left, right)
- Right clic + motion (up, down) zoom + and zoom -

‘Rotation/Zoom’ mode:

- Left clic + motion (up, down) zoom + and zoom -
- Left clic + motion (left, right) zoom + and zoom -
- Right clic + motion (left, right, up, down) rotation (left, right, up, down)

- Center clic: move scene

8.4.7 ‘XY grid’

Select this options to display the XY grid. The corresponding grid can also be displayed using the ‘XYZ axis’ properties of ‘Display’ element of the tree ‘Scene’).

8.4.8 ‘XZ grid’

Select this options to display the XZ grid. The corresponding grid can also be displayed using the ‘XYZ axis’ properties of ‘Display’ element of the tree ‘Scene’).

8.4.9 ‘YZ grid’

Select this options to display the YZ grid. The corresponding grid can also be displayed using the ‘XYZ axis’ properties of ‘Display’ element of the tree ‘Scene’).

8.4.10 ‘Export 3D view’

Allow to save the 3D view in a file (PNG, JPG, BMP).

‘To file’ Open a File Dialog box for saving the 3D view in a file with a specific format and a default resolution (800×600).

‘To a file with a specified resolution’ Open a File Dialog box for saving the 3D view in a file with a specific format and a given resolution.

8.5 Menu ‘Windows’

‘Reinitialize interface’ Restore the default configuration of the interface.

8.5.1 ‘3D View’

Define the location of the 3D view:

‘Main windows’ Set the 3D view as the main windows of the interface

‘Floating 3D view’ Set the 3D view as an independent windows. This option is useful when using two screen, for example one with the interface and the other for the 3D rendering.

8.5.2 ‘Console’

‘Export to file’ Export all information in the ‘Messages’ tab of the ‘Console’ windows to a ‘.TXT’ file (open a File Dialog box).

‘Delete windows’ Delete all information in the ‘Messages’ tab of the ‘Console’ windows.

8.6 Menu ‘Help’

‘Website’ Link to the I-Simpa website: i-simpa.ifsttar.fr.





‘Online documentation’ Link to the online documentation.

‘License’ Open the I-Simpa license file.




‘About I-Simpa’ Open a message box with general information concerning I-Simpa.

TOOLBARS



9.1 Toolbar ‘Project’

-  Create new project;
-  Open new project;
-  Save the project with the current filename;
-  Save the project with a new filename.

9.2 Toolbar ‘View/Camera’



- **3D display:**
 -  3D with faces (interior);
 -  3D without faces, all lines;
 -  3D with faces and all lines;
- **Camera:**
 - ◡ Reinitialize camera;
 - ◡ Mode ‘Rotation/Zoom’;
 - ◡ Mode ‘First person’;

9.3 Toolbar ‘Meshing’







-  Generate and display the meshing (open a Dialog box for selecting the meshing parameters from a calculation code);
-  Remove the meshing from the 3D view;
- Radio button ‘x’, ‘y’, ‘z’: Select the planes X, Y, Z for displaying the meshing with the cursor;

- The slider allows to move the meshing along the selected plane.

9.4 Toolbar ‘Selection’

- Mode ‘Camera’: see [menu ‘View’](#);
-  Mode ‘Face selection’: enable the face selection with the pointer (selected faces are colored on the 3D view and on the ‘Surfaces’ group on the tree ‘Data’ of the window ‘Project’). [The procedure for the face selection is presented here.](#)
-  Mode ‘Data extraction’: enable to extract a value on a colormap, with the pointer. The corresponding data is displayed in the window ‘Message’.

9.5 Toolbar ‘Simulation’

-  Reward to the last time step;
-  Play from the current time step;
-  Pause the animation;
-  Forward to the next time step;
-  Stop the animation (and return at the first time step);
-  Remove the animation (remove data animation from memory).

‘PROJECT’ WINDOW

10.1 Presentation

10.1.1 Principle

The **‘Project’** window allows to organize the set of useful data on three tabs *‘Scene’ tab*, *‘Calculation’ tab* and *‘Results’ tab*. It follows a natural procedure, for a classical acoustical study, which consists in:

1. Giving all information (geometry, material, sound sources, receivers...),
2. Gtarting the calculation (using a specific code),
3. Representing the results.

In the definition and the realisation of a calculation, **users should follow the natural procedure:**

1. Define all information/data in the tab *‘Scene’ tab*
2. Choose and parametrize the calculation code in the tab *‘Calculation’ tab*
3. Explore, process and represent all results in the tab *‘Results’ tab*

10.1.2 Definitions

All data, in each tab, are organized in **trees**, each tree containing **folders** (eventually with subfolders), each folder with **elements**. Each element is defined by several parameters that are showed in the **‘Properties’** window. These elements can be data, check boxes, character chains, scrolling lists, trees...

Warning: Data defined in the **‘Scene’** tree depend on the numerical code that will be used in the tab **‘Calculation’** (i.e. depending on the calculation code, it is not necessary to defined all elements. See the user guide/manual of the code for more information).

Note: Using Python scripts, users can add folders and elements in trees. This can be useful for integrating new numerical codes or specific processes. See the scripting guide of I-Simpa for more information.

10.1.3 Common features

Collapse/Expand a tree

A tree/branch can be collapsed (or expanded) by a:

- click on the - (or the +) symbol in front of a folder in a tree;
- double-click on the folder.

Copy/Paste function in a tree

The copy/paste function is valid in many folders and elements of trees. This can be used to duplicate elements and folders in a tree, or to copy elements and folders in a text editor (the format of elements is 'XML'). Use the right click of the pointer ('Copy' and 'Paste' actions) to copy/paste elements.

Copy/Paste function in a spreadsheet

All data concerning an element are described in a spreadsheet in the '**Properties**' window. All cells of the spreadsheet (for a given element) can be copy/paste to another spreadsheet for an equivalent element. Copy/Paste is also functional to export data to an external spreadsheet software.

Rename an element in a tree

Excepted for predefined folders and elements, all new folders and elements created by users can be renamed, either when creating the element/folder, either after creation using the '**Rename**' option (right click of the pointer on the element/folder to be renamed).

Remove an element from a tree

Excepted for predefined folders and elements, all new folders and elements created by users can be removed, using the '**Delete**' action (right click of the pointer on the element/folder to be renamed).

Create charts from tabular data

Creating charts

Most values represented as tables in a 'Properties' window can be displayed in a chart form, configurable by the user. The procedure consists of selecting cells in a table of data, then, to click with the right mouse button on the corresponding selection and to specify options representation:

'Tab name' Name of the tab containing the created chart.

'Name of X axis' Name of the x -axis.

'Name of Y axis' Name of the y -axis.

'Data alignment' Choose the orientation of

'Text label for X' Display labels of the x -axis in a 'text mode' using a linear space, or in 'value mode' using the real x value.

'Default Style' Curves representation.

Note: The choice of colors (for curves, fills, dots...) in the representation of graphics is random.

Note: The labels of the data series (in tables) are automatically considered in charts.

Chart Options

Once the chart is created, several actions are possible, using the contextual menu on the chart.

‘Original Zoom’ Reset to the initial display. Use the mouse (button left) to zoom in/out.

‘Export’

- ‘Export to image file’: export the chart as an image file (PNG, JPG, BMP).
- ‘Export to clipboard’: user can paste the chart in another application.

‘Show/Hide curves’ Select the curves to be displayed or ‘Show’/‘Hide’ all curves.

‘Display parameters’ Configure all the chart properties.

10.2 ‘Scene’ tab

In the ‘Data’ tree of the ‘Project’ window, user defines all needed information, in an acoustic point of view, in order to realize an acoustical calculation. This means, surface material, sound sources, receivers (punctual and plane), as well as other needed data in function of the numerical code (for example, the SPPS code can also take into account fitting zones).

In the ‘Project’ tree of the ‘Project’ window, user defines all general information (environmental data, configuration, information...) and common database (material and spectrum).

Warning: All data defined in the ‘Scene’ tree depend on the numerical code that will be used in the tab ‘Calculation’ (*i.e.* depending on the calculation code, it is not necessary to define all elements. See the user guide/manual of the corresponding code for more information).

Tip: For data that depend on frequency, users need to define only the data for the frequency bands of interest, *i.e.* the frequency bands that will be used for the calculation.

10.2.1 ‘Data’ tree

Fitting zone

The element ‘Fitting zone’ allows to define one or more volumes of the geometry that contain scattering objects, producing acoustics diffusion. This phenomenon can have an important effect on the sound level distribution, as well as on other acoustical parameters (like reverberation time), in rooms such as in industrial halls, with many objects on the floor.

Most of time, the diffusion process is modelled using the concept of mean free path (noted Lambda or $[\text{m}]$) with a statistical approach defined by a diffusion law). On the same time, acoustical absorption (noted Alpha or α , between 0 and 1) can also occurred on the scattering object.

Within I-Simpa, the diffusion in a fitting zone, can be defined using two methods (**Right click** on the folder ‘Fitting zone’ and **Select** the option in the contextual menu):

- by creating a rectangular volume in the 3D scene (**‘Define rectangular fitted zone’**) and by setting the diffusion parameters in the corresponding zone;
- by detecting one or more volumes in the 3D scene (**‘Define scene fitted zone’**) and by setting the diffusion parameters in the corresponding zones. This can also be done handly, by selecting all the face elements around a given volume and a point inside the volume.

Right click on the root ‘Fitting zone’ folder to define a fitting zone:

- **‘Define scene fitted zone’** Defines a fitted zone from the face elements of the 3D scene.

Warning: The volume must be a closed space.

- **‘Surfaces’** Contains the scene surfaces that define the fitting zone.
 - * **‘Surface area’** Total surface (in m^2) of all face elements in the folder.

The selection of faces can be made:

- * by using the 3D view. See the section [Surface selection](#) for understanding how to select surface elements
- * by drag & drop of all surfaces of a ‘Volume’ element;
- * by a direct conversion of a ‘Volume’ element to a fitting zone (see section [Volumes](#)).

- **‘Properties’** Defines the properties of the element.
 - * **‘Active fitting zone’** (Un)check to (Des)activate the fitting zone.

Note: Depending of the parameters of the selected calculation code, all the fitting zones can also be desactivated at the calculation step.

- * **‘Description’** Gives a short text description of the element.
- **‘Inside position’** Defines a point of coordinates (x,y,z) inside the volume. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.
- **‘Display’** Defines the parameters for displaying the element on the 3D view
 - * **‘Color’:** Defines the color of the fitting zone.
 - * **‘Show name’:** Check to display the name of the fitting zone on the 3D view.
- **‘Acoustics parameters’** Defines the acoustics parameters of the fitting zone, by frequency band.
 - * **‘Alpha’** *Absorption coefficient* of the scatterers.
 - * **‘Lambda’** *Mean free path* (unit: m).
 - * **‘Diffusion law’** Reflection law.
- **‘Define rectangular fitted zone’** Defines a parallelepipedic fitted zone.
 - **‘Acoustics parameters’** Similar as for ‘Define scene fitted zone’.

- ‘Display’
 - * ‘Color’ Defines color of the fitting zone in the 3D view.
 - * ‘Opacity’ Defines opacity of the fitting zone in the 3D view.
 - * ‘Render mode’ In the 3D view, the fitting zone can be represented by a zone with face (‘Volume’) or a contour (‘Borders’).
 - * ‘Show name’ Check to display the name of the fitting zone on the 3D view.
- ‘Properties’ Similar as for ‘Define scene fitted zone’.
- ‘Origin’ Defines the (x,y,z) coordinates of the first point of the rectangular zone. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.
- ‘Destination’ Defines the (x,y,z) coordinates of the point opposite the ‘Origin’ point of the rectangular zone. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.

Punctual receivers

Several acoustic indicators must be associated to punctual receivers in the volume of propagation. Besides their position and direction of view, it may also be necessary to define other acoustic properties, such as directivity, background noise... The receivers can be grouped within source groups. A group can contain other groups.

Right click on the root folder ‘Punctual receivers’ to create new receivers or groups of receivers.

- ‘New receiver’ Creates a new receiver.
- ‘New Group’ Creates a new group of receivers.
- ‘Create a receiver grid’ Creates a grid of receivers in a new group.
 - ‘Number of receivers (column)’ Number of receivers along a column.
 - ‘Number of receivers (row)’ Number of receivers along a row.
 - ‘Column vector’ Defines the (x,y,z) coordinates of the vector along the column.
 - ‘Row vector’ Defines the (x,y,z) coordinates of the vector along the row.
 - ‘Grid origin’ Defines the (x,y,z) coordinates of the grid origin.
- ‘Rotation’ Rotate all receivers in a given group (See [Manipulating sources and receivers](#) for description).
 - ‘Angle (degree)’ Defines the rotation angle in degree.
 - ‘Rotation center’ Defines the (x,y,z) coordinates of the point of rotation.
 - ‘Rotation vector’ Defines the (x,y,z) coordinates of the vector of rotation.
- ‘Translation’ Translate all receivers in a given group.
 - ‘Translation vector’ (See [Manipulating sources and receivers for description](#)) Defines the (x,y,z) coordinates of the vector of translation.
- ‘Orientation’ Orient all receivers in a given group to a target point.
 - ‘Target point’ Defines the (x,y,z) coordinates of the target point.

Receiver parameters:

- **‘Background noise’** Defines the existing background noise at the receiver location. This parameters can be useful, for example, to calculate some acoustics indicator for speech intelligibility. This parameter is defined by a background noise ‘Spectrum’. See the section [Using spectrum](#) for understanding how to use a spectrum in the application.
- **‘Direction’** Defines the orientation of the receivers, using a vector of coordinates (x,y,z). This parameters can be useful, for example, to associate a directivity to the receiver. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view. The receiver can also be automatically oriented to a given sound source with a dynamic link: choose the corresponding sound source in the list.
- **‘Display’** Defines the parameters for displaying the element on the 3D view.
 - **‘Color’**: Defines the color of the receiver.
 - **‘Show name’**: Check to display the name of the receiver on the 3D view.
- **‘Position’** Defines the (x,y,z) coordinate of the receiver. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.
- **‘Properties’** Defines the properties of the element.
 - **‘Description’** Gives a short text description of the element.
 - **‘Direction x’** Coordinate x of the orientation vector (unit vector).
 - **‘Direction y’** Coordinate y of the orientation vector (unit vector).
 - **‘Direction z’** Coordinate z of the orientation vector (unit vector).
 - **‘Directivity’** Directivity of the receiver. See the [Directivity](#) section for details.

Sound sources

This element allows to define the acoustic properties of sound sources, such as the acoustic power, the directivity, the position of the source and its orientation. The sound sources can be grouped within source groups. A group can contain other groups.

Right click on the root folder ‘Sound source’ to create new sound sources or groups of sound sources

- **‘New source’** Creates a new source.
- **‘New Group’** Creates a new group of receivers.
- **‘Create a line of sound source’** Creates a line of sound source in a new group.
- **‘All sources’** Possible actions on a group of sound sources.
 - **‘Enable’** Enable all sound surces in a group.
 - **‘Disable’** Disable all sound surces in a group.
 - **‘Rotation’** (See [Manipulating sources and receivers for description](#)) Same action as for sound sources.
 - **‘Translation’** (See [Manipulating sources and receivers for description](#)) Same action as for sound sources.

Sound source parameters:

- **‘Display’** Defines the parameters for displaying the element on the 3D view.

- **‘Position’** Defines the (x,y,z) coordinate of the sound source. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.
- **‘Properties’** Defines the properties of the element.
 - **‘Active source’** (Un)check to (Des)activate the sound source.
 - **‘Description’** Gives a short text description of the element.
 - **‘Direction x’** Coordinate x of the orientation vector (unit vector).
 - **‘Direction y’** Coordinate y of the orientation vector (unit vector).
 - **‘Direction z’** Coordinate z of the orientation vector (unit vector).
 - **‘Directivity’** Selects the directivity of the sound source in a list. Several ‘theoretical’ directivities are proposed (omnidirectionnal, unidirectionnal, XY, YZ, ZY) as well as ‘balloon’ directivities (*i.e.* as measured). In this last case, the ‘Balloon’ parameter must be given.
 - **‘Directivity Balloon’** Selects the directivity balloon of the sound source in a list. This parameter is enable only if the ‘Balloon’ option is selected in the ‘Directivity’ parameter.
 - **‘Time delay’** Defines a delay (in second) for activating the sound source.
- **‘Sound power’** Defines the sound power of the sound source. This parameter is defined by a power ‘Spectrum’. See the section [Using spectrum](#) for understanding how to use a spectrum in the application.

Surface receivers

It is common in a acoustics study to focus on the distribution of sound levels on a surface (noise mapping). The concept of mapping can also be generalized to the surface representation of any acoustic parameter, like reverberation time, clarity... The I-Simpa interface enables to define surfaces for the representation of maps.

By right clicking on the root ‘Surface receiver’ folder, it allows define two types of surface receiver:

- **‘New scene receiver’** Surface elements of the 3D scene can be selected.
 - **‘Surfaces’** Defines the surface element of the 3D scene to be considered for the surface receiver. See the section [Surface selection](#) for understanding how to select surface elements.
 - * **‘Surface area’** Total surface (in m²) of all face elements in the folder.
 - **‘Properties’** Defines the properties of the element.
 - * **‘Description’** Gives a short text description of the element.
- **‘New plan receiver’** The surface receiver is defined by a grid cutting plane, made of 3 vertices A, B and C.
 - **‘Display’** Defines the parameters for displaying the element on the 3D view.
 - * **‘Grid color’** Defines the color for displaying the grid.
 - * **‘Show grid’** Check to display the grid on the 3D view.
 - * **‘Show vertice names’** Check to display the name of the vertices A, B, C.
 - **‘Properties’** Defines the properties of the element.
 - * **‘Description’** Gives a short text description of the element.
 - * **‘Enable’** Check to enable the plan receiver in the calculation.
 - * **‘Grid resolution’** Defines the resolution of the grid (in m).

- **‘Vertex A’** Defines the (x,y,z) coordinates of the vertex A. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.
- **‘Vertex B’** Defines the (x,y,z) coordinates of the vertex B. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.
- **‘Vertex C’** Defines the (x,y,z) coordinates of the vertex C. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.

Surfaces

This element contains all surfaces of the scene. These surfaces are either created within I-Simpa in the case of a the creation of a parallelepipedic geometry or imported. In the last case, depending on the software used to create the 3D scene, it is possible to import faces that are already organized into groups. This group organisation allows to easily affect specific material (with specific acoustic properties) for each surface group.

Right click on the root folder ‘Surfaces’ to organize the face elements:

- **‘Add a group’** Creates a new group.

Right click on a group to perform actions:

- **‘Inverse face normal’** Change the orientation of the normal of all face elements within the group. This can useful if, at the importation of 3D scene, some original faces have a wrong orientation.
- **‘Properties’** Defines the properties of the element
 - **‘Material’** Select the material to affect to the surfaces group, from a list. The list is defined in the ‘Material database’ of the ‘Project tree’.
 - **‘Surface area’** Total surface (in m²) of all face elements in the folder.

Volumes

The volume element allows to define volumes of interest in the 3D scene. You can ‘handly’ create a volume from face elements in the 3D scene or automatically detect all existing volume in the 3D scene. This can useful to create or detect fitting zones.

Right click on the root folder ‘Volumes’ to manage ‘Volumes’:

- **‘Create a volume’** Creates a volume from the face elements of the 3D scene.
 - **‘Display’** Defines the parameters for displaying the element on the 3D view.
 - * **‘Domain color’** Defines the color for displaying the volume.
 - **‘Inside position’** Defines a point of coordinates (x,y,z) inside the volume. The position can be defined by filling the form or using the pointer on the 3D view. See the section [Defines position](#) for understanding how to define a location with the 3D view.
 - **‘Properties’** Defines the properties of the element
 - * **‘Calculate the mean free path’** Calculate the *mean free path* using the classical formulae from statistical room acoustics ($=4V/S$, with V the volume and S the surface of the volume).
 - * **‘Mean free path’** Value (in m) of the calculated mean free path.

- **‘Volume auto-detect’** Automatically detect all volumes within the 3D scene. Same parameters as for ‘Create a volume’.

10.2.2 ‘Project’ tree

Configuration

This element of the ‘Project’ tree defines some useful information concerning the project. These elements are not used for the calculation.

Property elements: these elements are all optional.

- **‘Author’** Name or information concerning the author of the project.
- **‘Date’** Date of the project. By default, it is the date of creation of the project (text).
- **‘Project description’** Quick description of the project.
- **‘Project name’** Name of the project, given by the author.

Display

This element of the ‘Project’ tree defines some useful parameter for the 3D display.

Property elements:

- **‘XYZ axis’** Define some parameters for the graphical representation of the geometric coordinates.
- **‘Arrow color (x)’** Open a dialog box for selecting the color of the arrow of the x-axis.
- **‘Arrow color (y)’** Open a dialog box for selecting the color of the arrow of the y-axis.
- **‘Arrow color (z)’** Open a dialog box for selecting the color of the arrow of the z-axis.
- **‘Arrow length’** Length (in m) of the arrow of the xyz-axis.
- **‘Arrow width’** Width (in m) of the arrow of the xyz-axis.
- **‘Grid size’** Size (in m) of a grid element.
- **‘Grid color’** Open a dialog box for selecting the color of the XYZ grid.
- **‘XY Grid’** Enable/disable the display of the XY grid.
- **‘XZ Grid’** Enable/disable the display of the XZ grid.
- **‘YZ Grid’** Enable/disable the display of the YZ grid.

Environment

This element of the ‘Project’ tree defines some useful environment data.

Warning: The use of these information depends on the calculation code. See the reference guide of the numerical code for more details. The code can enable the calculation of the atmospheric absorption and of the meteorological effect (considering a log₁₀ celerity gradient) with a ground description using the roughness parameters.

Property elements:

- **‘Atmospheric absorption - User defined’** Enable/disable the selection of the value of the atmospheric absorption.
- **‘Atmospheric absorption - Value’** User value of the atmospheric absorption. This option is available only if the option ‘Atmospheric absorption - User defined’ is enable.
- **‘Atmospheric pressure (Pa)’** Atmospheric pressure(Pa).
- **‘Meteorological effect - Celerity gradient a_log’** Value of the a_log parameter of the log_lin profile celerity. The default value is function of the option given by the ‘Meteorological effect - Profile’, but can be modified by the user.
- **‘Meteorological effect - Celerity gradient b_lin’** Value of the b_lin parameter of the log_lin profile celerity. The default value is function of the option given by the ‘Meteorological effect - Profile’, but can be modified by the user.
- **‘Meteorological effect - Profile’** Defines a log_lin gradient profile. By selecting one of the following options, it sets the ‘Meteorological effect - Celerity gradient a_log’ and ‘Meteorological effect - Celerity gradient b_lin’ parameters. However, these values can be modified by the user.
 - **‘Very favorable’** a_log=+1, b_lin=+0.12
 - **‘Favorable’** a_log=+0.4, b_lin=+0.04
 - **‘Homogenous’**: a_log=0, b_lin=0
 - **‘Unfavorable’**: a_log=-0.4, b_lin=-0.04
 - **‘Veryfavorable’**: a_log=-1, b_lin=-0.12
- **‘Relative humidity (%)’** Relative humidity of air (%).
- **‘Roughness - z0 (m)’** Value of the ground roughness (m).
- **‘Roughness - Ground type (m)’** Defines the ground type. By selecting one of the following options, it sets the ‘Roughness - z0 (m)’ parameter. However, this value can be modified by the user.
 - **‘Water’** z0=0.006
 - **‘Ground’** z0=0.02
 - **‘Short lawn’** z0=0.001
 - **‘Dense lawn’** z0=0.02
 - **‘Wheat (1m height)’** z0=0.16
 - **‘Sparse habitat (farm, trees, hedges)’** z0=0.6
 - **‘Low concentration Suburb (residential areas, gardens)’** z0=1.2
 - **‘Dense urban’** z0=10
 - **‘Dense suburb’** z0=1.8
- **‘Temperature (°C)’** Temperature of air (°C).

Informations

This element of the ‘Project’ tree gives some useful information concerning the I-Simpa project.

Property elements:

- **‘Model face count’** Number of face elements.
- **‘Number of active fitting zones’** Number of active fitting zones.

- **‘Number of active sound sources’** Number of active sound sources.
- **‘Number of fitting zones’** Number of fitting zones.
- **‘Number of punctual receivers’** Number of punctual receivers.
- **‘Number of sound sources’** Number of sound sources.
- **‘Number of surface groups’** Number of surface groups.
- **‘Scene surface (m2)’** Scene surface (m²).
- **‘Scene volume (m3)’** Scene volume (m³).

Project database

See a detail description of the [Project database](#)

10.3 ‘Calculation’ tab

This tab allows to choose and to define the acoustic calculation. This tab contains all calculation codes that are linked to I-Simpa. By default, two numerical codes are given:

- **‘TCR calculation code’_** Apply the Classical Theory of Reverberation for room acoustics. It applies Sabine and Eyring formulae. This code must be used only for closed spaces.
- **‘SPPS calculation code’_** This code is based on a particle tracing method, in respect with the geometrical and energetical hypothesis. This code can be applied in most of situations in room acoustics, as well as in open field, including urban area.

Important: Read the corresponding documentation before using these codes.

10.3.1 Using SPPS within I-Simpa

SPPS is already embedded in I-Simpa. Follow the next instructions to run a calculation with the SPPS code in I-Simpa.

Double left click on the TCR calculation code in I-Simpa to display all calculation parameters:

- *Frequency bands* for the selection of the frequency band to be used for the calculation
- **‘Meshing’_** for the meshing configuration
- **‘Calculation parameters’_** for the calculation parameters

Right click on the selected calculation code to display all possible actions:

- Run calculation
- Job list

Frequency bands

This allows to define the frequency band that will be used for the calculation. This can be done:

- by *hand*, by checking/unchecking the frequency to be considered;

- *automatically* ('Automatic selection'), by using the contextual menu on the element 'Frequency band' of the chosen code:
 - 'Unselect all' Uncheck all frequency bands.
 - 'Select all' Check all frequency bands.
 - 'Octave' Check third octave bands that are centred on the corresponding octave bands.
 - * 'All' All octave bands on the whole frequency range
 - * 'Building/Road' All octave bands between 125Hz and 4000Hz.
 - 'Third octaves' Check the third octave bands, considering.
 - * 'All' All third octave bands on the whole frequency range.
 - * 'Building/Road' All octave bands between 100Hz and 5000Hz.

Meshing

Some calculation codes may require a meshing of the domain (SPPS for example). I-Simpa used the [TetGen meshing code \(A Quality Tetrahedral Mesh Generator and a 3D Delaunay Triangulator\)](#). See the documentation for more information.

Within I-Simpa, TetGen can be parametrized:

- 'debugmode' Check/uncheck for activate the debug mode of TetGen;

Warning: With the debug mode, the meshing is not realized.

- 'minratio' Defines the ratio between the radius of the sphere that would contain a mesh and the length of the mesh. This parameter could be useful in order to avoid long meshes.
- 'userdefineparams' Defines new parameters by replacing default parameters.
- 'appendparams' Add parameters to the default parameters.
- 'preprocess' Check/uncheck for trying to repair the 3D scene if possible.
- 'maxvol' Maximum accepted value of a mesh volume (m3). The value can be modified only if the 'ismaxvol' parameter is checked.
- 'ismaxvol' Check/uncheck for using a given maximum value of the mesh volume 'maxvol'.
- 'constraintrecepteurss' Maximum accepted value of the mesh surface (m2) for a surface receiver. The value can be modified only if the 'isareaconstraint' parameter is checked.
- 'isareaconstraint'. Check/uncheck for using a given value of the mesh surface 'constraintrecepteurss'.

Calculation parameters

- 'Active calculation of atmospheric absorption' Enables calculation of atmospheric absorption.
- 'Active calculation of diffusion by fitting objects' Enables calculation of calculation of diffusion by fitting objects.

Important: If fitting zones are enabled, by checking this option, it disables the calculation of all fitting zones. If you uncheck this option, the individual configuration of each fitting zone is preserved.

- **‘Active calculation of direct field only’** Enables calculation of direct field only. There is no calculation of the reverberant field.

Tip: This can be useful if you want to realize a first calculation, as a reference, of the direct field at receivers. The second calculation (direct and reverberant fields) can thus be calculated and compared to the direct field (first calculation).

- **‘Active calculation transmission’** Enables calculation of transmission through walls.
- **‘Calculation method’** Selects the calculation method.
 - **‘Random’** Select the Random method.
 - **‘Energetic’** Select the Energetic method.
- **‘Echogram per source’** Check to calculation an echogram for each sound source, for a given receiver. If unchecked, a global echogram is calculated by summing all source contributions.
- **‘Export surface receivers for each frequency band’** Check to export the surface receiver results, for each frequency band. If unchecked, only the global value is exported.
- **‘Limit value of the particle extinction: ratio 10^n ’** For the ‘Energetic’ mode only. It defines a limit value of the particle energy. If the decrease of the particle energy is less than this value (*i.e.* the particle energy is very low), the particle is removed from the domain. For example, if $n=60$, it means that all particles whose energy will decrease to 60dB, will be removed.
- **‘Number of sound particles per source’** Defines the number of sound particles that are generated by the source.

Warning: The computational time depends on the number of particles. If you increase the total number of particles, you drastically increase the computational time.

- **‘Number of sound particles per source (display)’** Defines the number of particles that are used for the particle animation.

Note: Most of time, you need to consider only few hundreds or thousands particles for the animation. Increasing this number, will decrease the memory resources.

- **‘Random initialization number’** Initialize the random number series. If you select a number that is different from ‘0’, the random number series will always be the same. The starting number of the random number series will depend on the number you will consider for this parameter.

Warning: In a multithread simulation, I-Simpa/SPPS can not control the generation of random numbers. It means that this parameter will have no effect. Multithread simulation occurs when several frequency bands are considered in the simulation. To avoid multithreading, consider only one frequency band calculation.

- **‘Receiver radius (m)’** Defines the receiver radius (in m).
- **‘Simulation length (s)’** Defines the duration (in s) of the simulation.
- **‘Surface receiver export’** Select the kind of results that is exported.
 - **‘Soundmap: intensity’** Intensity level (in dB).

- ‘**Soundmap: SPL**’ Sound pressure level (in dB).
- ‘**Time step (s)**’ Time step (in s) for the calculation.

Computational time optimization

The computational time depends mainly of the **Number of sound particles per source**. Thus, in the SPPS code, each sound particle is followed, at each ‘Time step’ during its propagation inside the 3D domain, experiencing absorption, reflection, diffusion... and so on, during the ‘Simulation length’ and/or until the particles disappears when its energy is too small.

Considering a small number of sound particles may not produce relevant results to proceed an interesting acoustic analysis of the 3D model. Thus, it is necessary to consider a *large* number of sound particles. Assigning the good value for *large* is difficult and depends mainly on the scene size and the absorbent nature of the scene (surface absorption and volume absorption).

Few recommendations to evaluate the computational time:

- the number of sound particles defined in SPPS (‘Number of sound particles per source’ parameter) is defined for **each sound source**. Considering N sound particles per source and M sound sources will generate $N \times M$ sound particles in the domain;
- one calculation is done for **each frequency band**: considering F frequency bands will be equivalent to a calculation with $F \times N$ sound particles for one source (or $F \times N \times M$ for M sources). Note that, by default, the computation is parallelized on the processor core, for each frequency band (a frequency band for a given core). So the increase of computational time with the number of frequency band is not linear;
- in the ‘**Energetic**’ mode, each sound particle is ‘alive’ during its propagation until its energy is below a ratio of the initial energy (see “Limit value of the particle extinction: ratio 10^n ” parameter). **Increasing n** will keep the sound particles longer in the domain, which increases the computational time;
- in the ‘**Random**’ mode, each physical phenomena (absorption, diffusion) is applied by considering probabilistic approaches. Using this mode, the computational time is drastically decreased, but the quality of the results is also decreased. It is suggested to consider the ‘Random’ mode at the initial step of the study, and then to change to the ‘Energetic’ mode in order to obtain the final result.

10.3.2 Using TCR within I-Simpa

TCR is already embedded in I-Simpa. Follow the next instructions to run a calculation with the TCR code in I-Simpa.

Double left click on the TCR calculation code in I-Simpa to display all calculation parameters:

- ‘**Meshing**’_ for the meshing configuration
- ‘**Calculation parameters**’_ for the calculation parameters

Right click on the selected calculation code to display all possible actions:

- Run calculation
- Job list

Meshing

Some calculation codes may require a meshing of the domain (SPPS for example). I-Simpa used the [TetGen meshing code](#) (A Quality Tetrahedral Mesh Generator and a 3D Delaunay Triangulator). See the documentation for more information.

Within I-Simpa, TetGen can be parametrized:

- **‘debugmode’** Check/uncheck for activate the debug mode of TetGen;

Warning: With the debug mode, the meshing is not realized.

- **‘minratio’** Defines the ratio between the radius of the sphere that would contain a mesh and the lenght of the mesh. This parameter could be useful in order to avoid long meshes.
- **‘userdefineparams’** Defines new parameters by replacing default parameters.
- **‘appendparams’** Add parameters to the dafault parameters.
- **‘preprocess’** Check/uncheck for trying to repair the 3D scene if possible.
- **‘maxvol’** Maximun accepted value of a mesh volume (m3). The value can be modified only if the ‘ismaxvol’ parameter is check.
- **‘ismaxvol’** Check/uncheck for using a given maximum value of the mesh volume ‘maxvol’.
- **‘constraintrecepteurss’** Maximun accepted value of the mesh surface (m2) for a surface receiver. The value can be modified only if the ‘isareaconstraint’ parameter is checked.
- **‘isareaconstraint’**. Check/uncheck for using a given value of the mesh surface ‘constraintrecepteurss’.

Calculation parameters

- **‘Active calculation of atmospheric absorption’** Enables calculation of atmospheric absorption.
- **‘Export surface receivers for each frequency band’** Check to export the surface receiver results, for each frequency band. If uncheck, only the global value is exported.

10.3.3 Run calculation

Starts the calculation with the selected calculation code.

10.3.4 Job list

Instead of running the calculation, manually one after the other, you can create a list of calculation and, thus, run all calculations at once.

Important: To use a job list, you have to create a specific project for each job. If you keep the same project, but if you change only some parameters, you will repeat exactly the same calculation.

- **‘Add calculation in the job list’** Add the calculation of the given project in the job list.
- **‘Clear job list’** Clear the job list.
- **‘Run job list’** Run the job list.
- **‘Show job list’** Show the job list in the tab ‘Messages’ of the ‘Console’.

10.4 ‘Results’ tab

The ‘Results’ tab contains all results produced by the calculation codes. When a project is loaded into I-Simpa the outcome data are organized as a tree structure that exactly correspond to the folder/files tree on the hard disk (*i.e.* the *physical* tree). It is possible to directly access to the contents of a folder from the associated contextual menu, by choosing the ‘Open folder’ option. If changes are made in the *physical* tree, it is possible to refresh the results tree by setting the action ‘Refresh folder’.

Warning: Deleting a folder or a file from the I-Simpa interface or from an extern file explorer is definitive.

The results of the calculations are sequentially added to the ‘Results’ tree with a folder name that is automatically generated from the date of the calculation (the name can be modified by the user). For each calculation, two XML files are automatically added to the results folder:

- the *config.xml* file is generated by I-Simpa and contains all the information that are needed for the calculation (according to the calculation code used).
- the *projet_config.xml* file contains all information in the 3 tabs ‘Scene’, ‘Calculation’ and ‘Results’. This file can be used as a ‘memory’ of the calculation (archive).

In addition, each calculation codes generates specific files that can be interpreted by I-Simpa, depending of their format (with specific contents and file extensions).

10.4.1 File formats

I-Simpa can interpret two types of file:

- **File formats recognized by the Operating System**, such as *txt*, *xml*, *html*, *pdf*... for example. Double left click on the file starts the associated program within the operating system.
- **Native I-Simpa files**: these file are created by the calculation code and that can be associated with specific actions and treatments within I-Simpa. These files and associated actions are detailed below.
 - **File *.recp*** Punctual receiver file
 - **File *.gap*** Punctual receiver file
 - **File *.csbin*** Surface receiver file
 - **File *.gabe*** Tabulated data file
 - **File *.rpi*** Animated intensity file
 - **File *.pbin*** Animated data file

10.4.2 Punctual receiver file *.recp*

This file contains the temporal evolution, by frequency band, of a given quantity that is homogeneous to instantaneous sound squared pressure (p^2). Most of time, it should correspond to a punctual eceiver at the point. Typically, this type of file can represent an impulse response (RI). The actions associated with I-Simpa arise directly from the file format.

Important: The operations performed by the interface I-Simpa with this file type make sense only if the data in the file are equivalent to an instantaneous squared sound pressure (p^2).

Data Display

Double left click on a *.recp* file open a new window with three tabs

- **‘Sound Level SPL (dB)’ tab** It contains the temporal evolution of the quantity, for each frequency band, as a data table.
 - This tab is opened by default.
 - The values are represented in sound pressure levels (SPL).
 - The **Global** value (*i.e.* the sum of all frequency bands) is calculated and displayed at the bottom of each column.
 - The **Total** value (*i.e.* the sum of all time step) is calculated and displayed at the end of each row.
- **‘Sound Level SPL (dB)’ tab** It provides a graphic display of the **Global** value.
 - The temporal evolution of the **Global** value is displayed, as an echogram.
 - The cumulative quantity of the **Global** value is displayed, according to the Schroeder’s backward integration [Sch65].
- **‘Spectrum’ tab** It displays a spectrum at the punctual receiver.

Room acoustics parameters

Given the nature of the file (*i.e.* an echogram), the contextual menu that is associated to this file allows to calculate several room acoustic parameters:

- Sound Pressure Level (SPL) in dB
- Sound Pressure Level (SPL) in dB(A)
- Clarity C (in dB)
- Definition D (in %)
- Central Time Ts (in ms)
- Reverberation time RT (in s)
- Early decay time EDT (in s)
- Stage Support ST (in dB)

Several parameters can be given by the user in order to calculate user-values of some room acoustics parameters:

- Clarity: fix the value of the temporal limit of integration, usually 50 ms
- Definition: fix the value of the temporal limit of integration, usually 50 ms
- Reverberation time: fix the value of the sound level limit of integration, usually 30dB

Tip: Multiple calculations are allowed for each parameter, by using the semicolon ‘;’ between parameters.

After the calculation parameters, two files are created in the corresponding folder:

- **‘Acoustic parameters’** This file provides access to the room acoustics parameters in the form of a data table.
 - The parameters are given for each frequency band of interest.
 - When allowed, the **Global** value (*i.e.* the sum of all frequency bands) is calculated and displayed at the bottom of each column.

- When allowed, the **Average** value (*i.e.* the mean value on all frequency bands) is calculated and displayed at the bottom of each column.
- **‘Schroeder curves’** It displays the temporal evolution of the Schroeder’s curves [Sch65], for each frequency band, as a data table.

Tip: User can create charts for representing data from the data table. See [charts creation](#)

10.4.3 Punctual receiver file **.gap**

In addition to classical room acoustics parameters, some *advanced* parameters are also calculated, and displayed in the ‘Advanced sound level’ file. This file display the temporal evolution of the impulse response, for each frequency band, as a data table, weighted by $\cos \theta$ (LFC and LG) and $\cos^2 \theta$ (LF and LG).

The contextual menu that is associated to this file allows to calculate several room acoustic parameters:

- Early lateral energy fraction LFC (in %)
- Early lateral energy fraction LF (in %)
- Early lateral energy LF (in dB)
- Strength G (in dB)

Data display:

- The **Global** value (*i.e.* the sum of all frequency bands) is calculated and displayed at the bottom of each column.
- The **Total** value (*i.e.* the sum of all time step) is calculated and displayed at the end of each row.

Several parameters can be given by the user in order to calculate user-values of advanced parameters:

- LF: fix the value of the temporal limit of integration, usually 80 ms
- LFC: fix the value of the temporal limit of integration, usually 80 ms

Tip: Multiple calculations are allowed for each parameter, by using the semicolon ‘;’ between parameters.

10.4.4 Surface receiver file **.csbin**

A **.csbin** file contains the temporal evolution of a quantity that represents an acoustic intensity on a surface.

Depending of the calculation parameters, one can obtain, for each surface receiver, by default, on **Global** value (in the ‘Global’ folder) and, in addition, the result for each frequency band in a corresponding folder.

A right click on a surface receiver file opens the contextual menu, with specific actions:

- **‘Acoustic parameters’** Allows to compute relevant room acoustics parameters on the surface. Depending of the selection, it creates specific files on the corresponding folder.
 - Clarity C (in dB)
 - Definition D (in %)
 - Central Time Ts (in ms)
 - Reverberation time RT (in s)
 - Early decay time EDT (in s)

- Stage Support ST (in dB)
- **‘Load animation’** It allows to represent the spatial variation of the indicators that is selected. If the indicators contains some time dependent value, it can displayed an animation. On can interact on the animation with the **‘Simulation’** toolbar.
 - **‘Instantaneous value’** Show the value of the given indicator at each time step.
 - **‘Cumulative instantaneous value’** Show the value of the given indicator at each time step, by cumulating all past steps.
 - **‘Total value’** Show the total value of the given indicator. No animation.

10.4.5 Tabulated data file *.gabe*

10.4.6 Animated intensity file *.rpi*

10.4.7 Animated data file *.pbin*

‘PROPERTIES’ WINDOW

This windows display the properties of a given element.

In order to display the element properties in the ‘Properties’ windows, users can either:

- **double-click** on an element in a tree;
- **right click** on the element and select ‘Properties’ in the corresponding contextual menu.

‘CONSOLE’ WINDOW

The ‘Console’ window contains two tabs:

- the ‘Message’ tab displays messages for the user, with a color code:
 - Information messages (black) according to the operations performed by the user
 - Warning messages (red): operations to be performed by the user
 - Information messages (blue) from internal or external programs (computer codes, a correction tool...)

Note: All messages are time stamped.

Note: For the meaning of the messages from external programs, refer to the documentation of the corresponding program.

Note: The information in the tab ‘Message’ can be saved in a text format using the ‘Console/Export to file’ option of the ‘Windows’ menu.

Note: The console can be reset using the ‘Console/Delete windows’ option of the ‘Windows’ menu.

- the ‘Python’ tab allows you to run Python scripts, both to perform treatments on data from I-Simpa or to develop new features. Refers to the scripting examples for the development of scripts in I-Simpa and to the [Python documentation](#).

‘MAIN’ WINDOW

The ‘3D View’ window contains the 3D model of the scene. Depending on the Pointer mode, several actions are possible:

- face selection (See the section [Surface selection](#) for understanding how to select surface elements)
- navigation in the scene (See [Camera mode](#) for details)
- data extraction

Tip: The main window is (un)dockable. You can display the Main window on a second screen (see menu ‘Windows > 3D View’ options)

PROJECT DATABASE

The Project database is an element of the 'Project' tree. This database is composed of three database:

- *Directivities* for sound source and receiver
- *Materials* for the scene boundaries
- *Spectrum* for receiver and sound sources

For each database, two sub-database are given:

- a 'Reference' database that contains built in information, which can not be edited by the user
- a 'User' database that can be edited by the user to create new elements

Tip: Note that it is possible to copy an element from the 'Reference' database to the 'User' database. Copy/Paste is also possible for folder.

14.1 Directivities

This feature is experimental and is not documented.

14.2 Materials

14.2.1 Contextual menu

Right click on the 'User' folder to define a new material or a group of material.

- **'New Material'** Creates a new material in the corresponding group.
- **'Create group'** Creates a new group in the corresponding group.
- **'Import from a file'** Allows to import materials from a file. Several formats are supported:

Importing from CATT-Acoustic Format File used by the CATT-Acoustic software: text file with extension '.txt' (see [CATT-Acoustic website](#)).

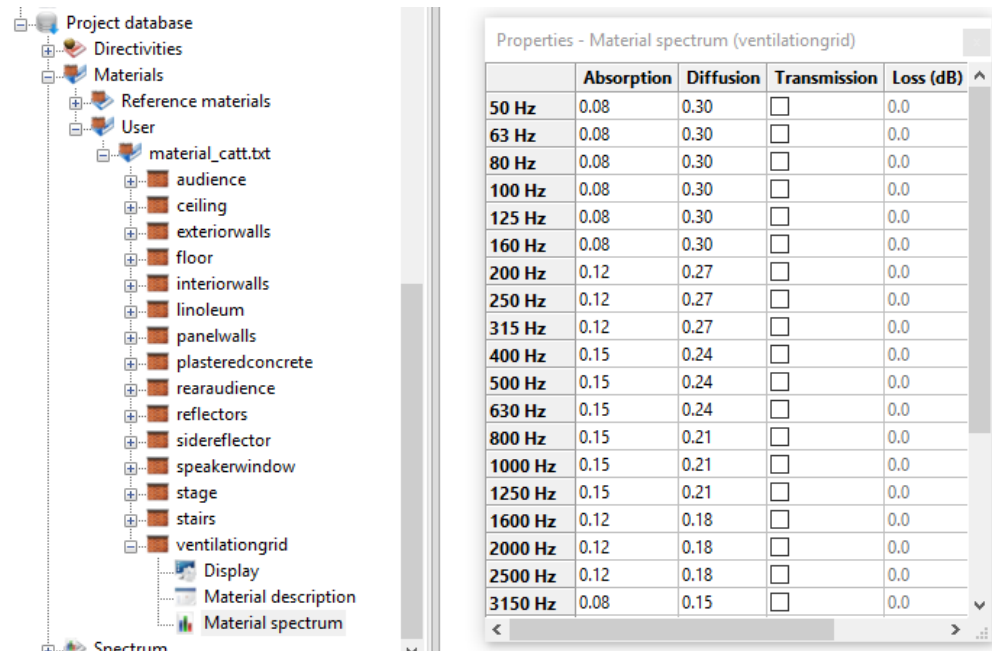
CATT-Acoustic material file example:

```

panelwalls = < 21 12 9 6 9 13 > L < 15 13 11 9 7 5 >
sidereflector = < 20 12 6 4 7 10 > L < 15 20 25 35 45 50 >
audience = < 45 60 73 80 75 64 > L < 30 40 50 60 65 70 >
rearaudience = < 50 66 80 88 83 70 > L < 30 40 50 60 65 70 >
linoleum = < 2 3 3 4 6 5 > L < 30 27 24 21 18 15 >
reflectors = < 12 10 4 3 3 2 >
stage = < 15 8 6 6 6 6 > L < 15 13 11 9 7 5 >
ceiling = < 20 15 10 8 4 2 > L < 15 13 11 9 7 5 >
speakerwindow = < 2 6 3 3 2 2 > L < 30 27 24 21 18 15 >
plasteredconcrete = < 2 2 3 3 4 6 > L < 15 13 11 9 7 5 >
ventilationgrid = < 8 12 15 15 12 8 > L < 30 27 24 21 18 15 >
floor = < 15 8 6 6 6 6 > L < 15 13 11 9 7 5 >
interiorwalls = < 20 12 6 4 7 10 > L < 15 13 11 9 7 5 >
stairs = < 15 8 6 6 6 6 > L < 15 13 11 9 7 5 >
exteriorwalls = < 20 12 6 4 7 10 > L < 15 20 25 35 45 50 >

```

CATT-Acoustic material file import example in I-Simpa



	Absorption	Diffusion	Transmission	Loss (dB)
50 Hz	0.08	0.30	<input type="checkbox"/>	0.0
63 Hz	0.08	0.30	<input type="checkbox"/>	0.0
80 Hz	0.08	0.30	<input type="checkbox"/>	0.0
100 Hz	0.08	0.30	<input type="checkbox"/>	0.0
125 Hz	0.08	0.30	<input type="checkbox"/>	0.0
160 Hz	0.08	0.30	<input type="checkbox"/>	0.0
200 Hz	0.12	0.27	<input type="checkbox"/>	0.0
250 Hz	0.12	0.27	<input type="checkbox"/>	0.0
315 Hz	0.12	0.27	<input type="checkbox"/>	0.0
400 Hz	0.15	0.24	<input type="checkbox"/>	0.0
500 Hz	0.15	0.24	<input type="checkbox"/>	0.0
630 Hz	0.15	0.24	<input type="checkbox"/>	0.0
800 Hz	0.15	0.21	<input type="checkbox"/>	0.0
1000 Hz	0.15	0.21	<input type="checkbox"/>	0.0
1250 Hz	0.15	0.21	<input type="checkbox"/>	0.0
1600 Hz	0.12	0.18	<input type="checkbox"/>	0.0
2000 Hz	0.12	0.18	<input type="checkbox"/>	0.0
2500 Hz	0.12	0.18	<input type="checkbox"/>	0.0
3150 Hz	0.08	0.15	<input type="checkbox"/>	0.0

Importing from Odeon Format File used by the software Odeon: text file with extension ‘.li8’ (see [Odeon website](#));

Warning: The CATT-Acoustic and Odeon data files are defined octave bands, while I-Simpa uses third octave bands. During the importation, values per octave band are attributed to the corresponding third octaves. Furthermore, since I-Simpa takes into account upper and lower frequency bands than the last software, data are extrapolated from the closest frequency band data.

Note: During importation, all fields of the original file are not imported from the data files.

14.2.2 Properties

- **‘Display’** Opens a dialog box to assign a color to the material, for the representation of the material in the scene (see [Menu view](#)).
- **‘Description’**
 - **‘Description’** String to describe the material.
 - **‘Mass density (kg/m3)’** Density of the material (real).
 - **‘Reference’** Reference associated with the material (text).
 - **‘Resistivity (kN.mm-4)’** Resistivity of the material (real value).
 - **‘Side of material effect’** List for choosing how the material properties are considered in relation to the direction of propagation relative to the normal to the material:
 - * **‘Bilateral’** The material has the same properties on both sides of the surface to which it is associated.
 - * **‘Unilateral’** The material is transparent in one direction. The acoustic properties of the material are considered only in a propagation direction opposite to the normal to the face. See the guide of the numerical code to know how this property is used.
- **‘Spectrum’** Defines the acoustic properties of the material for each frequency band.
- **‘Absorption’** Absorption coefficient of the material (real value between 0 and 1).
- **‘Scattering’** Scattering coefficient of the material (real value between 0 and 1).
- **‘Transmission’** Check/uncheck for taking acoustic transmission into account.
- **‘Loss’** Transmission loss of the material (dB).
- **‘Diffusion law’** Reflection law for the diffuse part of the reflection;
 - **‘Lambert’** Diffuse Lambert reflection law (cosinus of the incident angle).
 - **‘Specular’** Specular reflection law.
 - **‘Uniform’** Uniform reflection law, (*i.e.* ‘W’ reflection law that corresponds to \cos^2 of the incident angle).
 - **‘W2’** ‘W2’ reflection law (\cos^2 of the incident angle).
 - **‘W3’** ‘W3’ reflection law (\cos^3 of the incident angle).
 - **‘W4’** ‘W4’ reflection law (\cos^4 of the incident angle).

Note: For a scattering coefficient set to 0 (no diffuse reflection), the ‘Diffusion law’ is set to ‘Specular’, and can not be modified.

Note: For a scattering coefficient set to 1 (full diffuse reflection), the ‘Diffusion law’ is set to ‘Specular’ by default, but can be changed.

Warning: Absorption and transmission loss are linked together. Since transmission is a part of the absorption, the transmission loss can not larger than the absorption coefficient. For example, for a given absorption coefficient of 0.2, the transmission loss must be smaller than 20 dB (*i.e.* $0.2=10^{(20/10)}$).

- If the transmission loss value is larger than the given absorption, the absorption coefficient is automatically adjusted.
- If the absorption coefficient is changed, the transmission loss can be automatically adjusted if necessary.

14.3 Spectrum

Spectrum are can be used either for the definition of the source emission or for the definition of the background noise at a receiver. See the section [Using spectrum](#) for understanding how to use a spectrum in the application.

14.3.1 Contextual menu

Right click on the ‘User’ folder to define a new sepctrum.

14.3.2 Properties

User has to define the spectrum value for each frequency band, either in ‘dB’ or in ‘dB(A)’.

Note: Values in in ‘dB’ and in ‘dB(A)’ are linked together. Changing one value, for example in dB (respectively in dB(A)), will change the dB(A) value (respectively the dB value).

Note: Changing the ‘Global’ value will affect the value for each frequency band: all values are shifted in order to respect the global value.

IMPORT OPTIONS

15.1 ‘Average model remesh’

Allow to build an approximate model from the 3D scene which is imported.

Indeed, due to the sensitivity of some numerical codes within I-Simpa in regards with the ‘quality’ of the 3D geometries, it may be possible that some calculation codes will not be functional. A possibility that is offered by I-Simpa, is to use the ‘original’ 3D geometry to create an approximate 3D model, with a ‘quality’ in accordance with the calculation code.

Note: Please note, that research works on the sound field modelling in room acoustics,

have shown that it is preferable to use a simple geometry with a good evaluation of the acoustical properties of surface materials, instead of a complicated geometry. In other word, if you need to build an approximate geometry, you don’t need to choose a high level of approximation.

Warning: Depending of the chosen options and the size of the volume, the ‘Average model remesh’ option can require large computational times.

The realization of the approximate model is based on the marching cube technique, and will try to propose a model as close as possible from the original one. When choosing the ‘Average model remesh’ options, two dialog boxes will be successively proposed:

15.1.1 Mesh correction options:

- **‘Model solving’** This option defines the resolution n of the approximation. The volume of the original model is decomposed in 2^n elements in each 3D axis. For example, for $n=7$, we will have $128 \times 128 \times 128$ cubic cells in the original volume.
- **‘Triangulation method’** This option defines the type of method used for approximating model. There is currently only one possible option ‘Marching cube’.
- **‘Remesh adaptation’** When cubic cells are created in the original volume, you can decide to ‘deform’ the cells close to the surface of the volume in order to fit as well as possible the delimiting surface of the original model (option ‘Vertices translation’). If deformation is required, choose the ‘None’ option. important The ‘Vertices translation’ option can require large computational time and can also produced some unwanted effects.
- **‘Reduction of the number of surfaces’** This option able to reduce the total number of faces in the approximate model, by evaluating the quality of each face. There is currently only one possible option ‘Edge collapse’.

This method is based on the size and the shape of each face, reduced to a ‘Quality Constraint’, noted *q*, and on the ‘Approximation constraint’. See the following [book](#) for details: PLTMG: A Software Package for Solving Elliptic Partial Differential Equations by Randolph E. Bank.

15.2 ‘Repair model’

With this option, I-Simpa try to repair some elements of the 3D model (fusion of vertices, correction of non-planar surfaces...). This will assure a better compatibility with the numerical code within I-Simpa, but is still limited.

15.3 ‘Surface meshing’

This option allows to enhance the quality of surface elements, by re-meshing original surfaces. Default parameter are proposed (it seems to offer a good quality), but can be changed. This procedure is based on the [TetGen](#) mesh generator.

Warning: By using this option, it increases the numbers of face elements. The assignation of materials within I-Simpa will be more time consumer. This option should be used only if the mesh generation within I-Simpa can not be realized due to a poor quality of surface elements.

15.4 ‘Keep existing group’

This option should be used only when importing a new 3D geometry in the current project.

This allows to associate *new* faces of the new geometry to *old* surface groups. The method compares the location of the *new* surface elements from the *old* one, in regard with a given precision (‘Association maximum distance’).

Tip: This can be useful, if users want to load a modified geometry (with minor changes from the previous one) without re-affecting all surface materials within I-Simpa.

RECOMMENDATIONS TO IMPORT A 3D SCENE

I-Simpa is very sensitive to the quality of 3D files. It will not accept a ‘bad’ geometry, with faces intersections, holes between faces... (in fact, the model will be imported, but some codes like SPPS will not be functional).

It is recommended that users try to export their model in I-Simpa during the different steps of the model building:

1. **Start to build your 3D model** with your CAD software (the main close surface of the model);
2. **Export the model** in a 3DS file with your CAD software;
3. **Import the 3D model** within I-Simpa;
4. **Generate a meshing** by clicking on the ‘Meshing’ button on the icon toolbar: this will verify that your model is correctly built;
5. if user can see the mesh inside the model, it means that the model is correct for I-Simpa and that the calculation with SPPS will be functional;
6. if the model is not correct, return to the CAD software and try to see what is the problem (the problem is due to something in the last step of the built of the 3D model);
7. if the model is correct, return to your CAD software (step 0) and continue your 3D modelling, by adding objects, faces... Then, continue using the same procedure.

Note: At the importation step, I-Simpa can correct some errors. During importation check the ‘Repair model’ [Import options](#) (if not already done).

Note: If you still have trouble during importation, you can also create an approximate model by using the [Import options](#) ‘Average model remesh’. It will create a simple (or simplified) model.

Tip: Note that, when creating a 3D model for acoustic calculations, it is not necessary to create a very complex 3D model with too many faces. You will have better results with the simple shape of your 3D model, and by adjusting correctly all acoustical characteristics of the material on the face. Make simple.

DEFINE AND USE SPECTRUM

Important: It is extremely important that users experiment the manipulation of spectrum before using I-Simpa. The interdependence of sound level in dB and in dB(A), as well as the interdependence of global sound level and sound level for each frequency band, although it is conformed to physical principles, is not obvious.

17.1 General information

- Spectrum are defined in the same way for receivers and sound sources
- Spectrum are defined only within in 'Spectrum' database folder of 'Project database' folder, in the 'Project' tree,
- the 'Spectrum' database folder contains 2 subfolders, the 'Reference' subfolder and the 'User' subfolder:
- Spectrum within the 'Reference' subfolders can not be modified. This folder contains only reference built-in spectrum.
- User can create spectrums in the 'User' subfolder. Spectrum can be modified later if necessary.

17.2 Definition

17.2.1 Spectrum creation in the 'Project' tree

In the 'Spectrum' database folder, spectrum are defined simultaneously by a sound level in decibels (dB) or in decibels with the A-weighted (dB(A)), for each frequency band of interest.

Very important rules:

- at the spectrum creation, the global sound level (in dB) is set to 0 dB;
- modifying the global sound level (in dB) updates automatically the ponderate global sound level (in dB(A));
- modifying the ponderate global sound level (in dB(A)) updates automatically the global sound level (in dB);
- modifying the global sound level, either in dB and in dB(A), updates automatically the value for all frequency band, in dB and in dB(A), by applying an uniform ponderation on all frequency band;
- modifying one or more values for a given frequency band, affects automatically the global sound level, both in dB and in dB(A);
- the global sound level, both in dB and in dB(A), are calculated over the 27 frequency bands between 50 and 20000 Hz.

Warning: User can however select only few frequency bands for the calculation. In this case, the user must set the correct sound level values for each corresponding frequency band. The global sound level that will be calculated by I-Simpa (for source spectrum) will not be the global sound level that should be obtained over the selected frequency bands. However, in the results data, since the calculation will be done with the selected frequency bands, the global sound level at a receiver will be correct.

Examples:

1. A first spectrum is created:

- By default, the global sound level is automatically set to 0 dB;
- By default, the global sound level is distributed over all frequency bands (27 third octave bands, from 50 Hz to 20 000 Hz), which gives -14.31 dB for each band (*i.e.* $-10\log(27)=-14.31$ dB);
- The A-weighted is automatically calculated for each frequency band, leading a global sound level of -2.38 dB(A).

2. The global sound level is set to 100 dB:

- The sound level for each frequency band is uniformly modified and sets to 85.69 dB (*i.e.* $100-14.31$ dB);
- The ponderate sound level in dB(A) is automatically modified, both for each frequency band and for the global sound level.

3. The ponderate sound level is set to 100 dB(A):

- The ponderate sound level for each frequency band is modified, according to the pattern of the A ponderation;
- The ponderate sound level in dB is automatically modified, both for each frequency band and for the global sound level.

4. from the last spectrum, the sound level is modified to 50 dB for the third octave band between 500Hz and 2000Hz, and then the global sound level is set to 80 dB:

- All sound level, both in dB and dB(A) are automatically updated.

17.2.2 Using spectrum in the ‘Data’ tree

In I-Simpa, spectrum are used either for punctual receivers (element ‘Background noise’) and for sound sources (element ‘Sound power’).

- in the ‘Properties’ window, **select** the ‘Spectrum’ in the list (the list contains all ‘Reference’ and ‘User’ spectrums)
- **define** the total sound power ‘Lw’ (in dB);
- if necessary, **add** an ‘Attenuation’ (in dB) either in global or in frequency band: this attenuation will be applied on the corresponding values of the sound power of the source.

Tip: The ‘Attenuation’ value can be useful to simulate an attenuation device on the source, without changing the source spectrum, just by adjusting the attenuation in global or in frequency band.

Warning: A global attenuation of 0 dB will apply an uniform attenuation of -14.31 dB for each frequency band. In you do not want to take an attenuation into account, the attenuation must be set to 0 dB for each frequency band.

DEFINE A POSITION USING THE POINTER ON THE 3D VIEW

Many elements of the tree 'Data' require to define a 'Position' (x,y,z) in the 3D geometry. Data can be defined:

- by filling the corresponding cells (x, y and z) in the corresponding 'Properties' window;
- **by using the pointer. In this last case, user has to point the 'Position' element with the pointer in the tree:**
 - **Right click** on the 'Position' element;
 - **Select** the 'Define in the 3D view' option in the contextual menu;
 - Then, in the 3D view, users point a position on a visible face; this will affect the values (x,y,z) to the 'Position' element. Users can then adjust values in the corresponding cell of the spreadsheet in the 'Properties' window.

SURFACE SELECTION

To select one or more face elements on the 3D scene, select the pointer mode 'Face selection' on the Toolbar 'Selection':

- **Left click** on a surface to select a face element
- **Press CTRL button** and **Double left click** on a surface to select all coplanar face elements
- **Press CTRL button** and **Select** multiple face elements

When one or more surfaces are selected:

- the selected face elements are highlighted on the 3D view;
- the selected face elements are highlighted on the corresponding 'Surfaces' group in the 'Data' tree of the 'Scene' tab;
- once the face elements are selected, they can be moved from a 'Surfaces' group to another (within the 'Data' tree), using the pointer or using available actions after a right click on one of the selected faces.

MANIPULATING SOURCES AND RECEIVERS

20.1 Rotate a group of sources/receivers

I-Simpa can automatically rotate a group of punctual receivers or sound sources. Users have to specify:

- **‘Rotation angle (degree)’**: angle of rotation [degree];
- **‘Rotation center’**: center of the rotation (position (x,y,z) in [m]);
- **‘Rotation vector’**: axis of rotation (x,y,z) in [m].

Example: Below, an example of rotation of a receivers grid in a plane XY:

- **‘Rotation angle (degree)’**=45°
- at center position **‘Rotation center’**=[1.,1.,0.]
- around axis **‘Rotation vector’**=[0.,0.,1.]

20.2 Translate a group of sources/receivers

I-Simpa can automatically translate of punctual receivers or sound sources. Users have to specify:

- **‘Direction (m)’**: define the direction (vector) of translation.

Example: Below, an example of a translation of a receivers grid, in the direction [3.,0.,0.]:

20.3 Create a grid of receivers

I-Simpa can creates automatically 2D grids of punctual receivers. Users have to specify:

- a column (col) vector and a row vector to define the grid plane, as well as, the spatial step between two receivers
- the starting position (origin) of the grid,
- and the number of rows and columns of the grid.

Grid Parameters:

- **‘Col step x (m)’**: define the x coordinate of the column vector;
- **‘Col step y (m)’**: define the y coordinate of the column vector;
- **‘Col step z (m)’**: define the z coordinate of the column vector;
- **‘Number of cols’**: define the number of columns;

- **‘Number of rows’**: define the number of rows;
- **‘Row step x (m)’**: define the x coordinate of the row vector;
- **‘Row step y (m)’**: define the y coordinate of the row vector;
- **‘Row step z (m)’**: define the z coordinate of the row vector;
- **‘Starting position x (m)’**: define the x coordinate of the origin of the grid;
- **‘Starting position y (m)’**: define the y coordinate of the origin of the grid;
- **‘Starting position z (m)’**: define the z coordinate of the origin of the grid;

USING DIRECTIVITY

This feature is experimental and is not documented.

CREATING CHARTS

Most values represented as tables in a 'Properties' window can be displayed in a chart form, configurable by the user. The procedure consists of selecting cells in a table of data, then, to click with the right mouse button on the corresponding selection and to specify options representation:

'Tab name' Name of the tab containing the created a chart.

'Name of X axis' Name of the x -axis.

'Name of Y axis' Name of the y -axis.

'Data alignment' Choose the orientation of

'Text label for X' Display labels of the x -axis in a 'text mode' using a linear space, or in 'value mode' using the real x value.

'Default Style' Curves representation.

Note: The choice of colors (for curves, fills, dots. . .) in the representation of graphics is random.

Note: The labels of the data series (in tables) are automatically considered in charts.

CHART OPTIONS

Once the chart is created, several actions are possible, using the contextual menu on the chart.

‘Original Zoom’ Reset to the initial display. Use the mouse (button left) to zoom in/out.

‘Export’

- ‘Export to image file’: export the chart as an image file (PNG, JPG, BMP).
- ‘Export to clipboard’: user can paste the chart in another application.

‘Show/Hide curves’ Select the curves to be displayed or ‘Show’/‘Hide’ all curves.

‘Display parameters’ Configure all the chart properties.

PRESENTATION

SPPS is a sound particles-tracing code, based on geometrical, energetical and probabilistic approaches.

- Visit the [official I-Simpa website](#) for more information about SPPS code.
- See the main [characteristics of the code](#) that is embedded in I-Simpa

The simulation principle of the SPPS code (from French “Simulation de la Propagation de Particules Sonores”) relies upon tracking sound particles, carrying a amount of energy and emitted from a sound source, within a 3D-domain. Each particle propagates along a straight line between two time steps t (the whole trajectory may be curved), until collision with an object. At each collision, sound particles may be absorbed, reflected, scattered, diffused, transmitted, depending on the nature of the object.

Two algorithms can be considered:

1. The first approach (**Energetic**) is to consider that the energy of the particle is constant. In function of the phenomena, the particle may disappear from the domain or follows its propagation: the number of sound particles decreases along the time.
2. In the second approach (**Random**), the particle energy is varying according to the physical phenomena occurring during the propagation. In this case, the number of particles in the domain should be constant along the time. Since, in both cases, physical phenomena can be modeled according to probabilistic laws, both approaches are equivalent to Monte-Carlo methods. The accuracy of prediction is then mainly dependent of the initial number of particles.

USING SPPS WITHIN I-SIMPA

SPPS is already embedded in I-Simpa. Follow the next instructions to run a calculation with the SPPS code in I-Simpa.

Double left click on the TCR calculation code in I-Simpa to display all calculation parameters:

- *Frequency bands* for the selection of the frequency band to be used for the calculation
- *Meshing* for the meshing configuration
- *Calculation parameters* for the calculation parameters

Right click on the selected calculation code to display all possible actions:

- Run calculation
- Job list

25.1 Frequency bands

This allows to define the frequency band that will be used for the calculation. This can be done:

- by *hand*, by checking/unchecking the frequency to be considered;
- *automatically* ('Automatic selection'), by using the contextual menu on the element 'Frequency band' of the chosen code:
 - **'Unselect all'** Uncheck all frequency bands.
 - **'Select all'** Check all frequency bands.
 - **'Octave'** Check third octave bands that are centred on the corresponding octave bands.
 - * **'All'** All octave bands on the whole frequency range
 - * **'Building/Road'** All octave bands between 125Hz and 4000Hz.
 - **'Third octaves'** Check the third octave bands, considering.
 - * **'All'** All third octave bands on the whole frequency range.
 - * **'Building/Road'** All octave bands between 100Hz and 5000Hz.

25.2 Meshing

Some calculation codes may require a meshing of the domain (SPPS for example). I-Simpa used the [TetGen meshing code](#) (A Quality Tetrahedral Mesh Generator and a 3D Delaunay Triangulator). See the documentation for more information.

Within I-Simpa, TetGen can be parametrized:

- **‘debugmode’** Check/uncheck for activate the debug mode of TetGen;

Warning: With the debug mode, the meshing is not realized.

- **‘minratio’** Defines the ratio between the radius of the sphere that would contain a mesh and the lenght of the mesh. This parameter could be useful in order to avoid long meshes.
- **‘userdefineparams’** Defines new parameters by replacing default parameters.
- **‘appendparams’** Add parameters to the default parameters.
- **‘preprocess’** Check/uncheck for trying to repair the 3D scene if possible.
- **‘maxvol’** Maximum accepted value of a mesh volume (m3). The value can be modified only if the ‘ismaxvol’ parameter is checked.
- **‘ismaxvol’** Check/uncheck for using a given maximum value of the mesh volume ‘maxvol’.
- **‘constraintrecepteurss’** Maximum accepted value of the mesh surface (m2) for a surface receiver. The value can be modified only if the ‘isareaconstraint’ parameter is checked.
- **‘isareaconstraint’**. Check/uncheck for using a given value of the mesh surface ‘constraintrecepteurss’.

25.3 Calculation parameters

- **‘Active calculation of atmospheric absorption’** Enables calculation of atmospheric absorption.
- **‘Active calculation of diffusion by fitting objects’** Enables calculation of calculation of diffusion by fitting objects.

Important: If fitting zones are enabled, by checking this option, it disables the calculation of all fitting zones. If you uncheck this option, the individual configuration of each fitting zone is preserved.

- **‘Active calculation of direct field only’** Enables calculation of direct field only. There is no calculation of the reverberant field.

Tip: This can be useful if you want to realize a first calculation, as a reference, of the direct field at receivers. The second calculation (direct and reverberant fields) can thus be calculated and compared to the direct field (first calculation).

- **‘Active calculation transmission’** Enables calculation of transmission through walls.
- **‘Calculation method’** Selects the calculation method.
 - **‘Random’** Select the Random method.
 - **‘Energetic’** Select the Energetic method.
- **‘Echogram per source’** Check to calculation an echogram for each sound source, for a given receiver. If uncheck, a global echogram is calculated by summing all source contributions.
- **‘Export surface receivers for each frequency band’** Check to export the surface receiver results, for each frequency band. If uncheck, only the global value is exported.

- **‘Limit value of the particle extinction: ratio 10^n ’** For the ‘Energetic’ mode only. It defines a limit value of the particle energy. If the decrease of the particle energy is less than this value (*i.e.* the particle energy is very low), the particle is removed from the domain. For example, if $n=60$, it means that all particles whose energy will decrease to 60dB, will be removed.
- **‘Number of sound particles per source’** Defines the number of sound particles that are generated by the source.

Warning: The computational time depends on the number of particles. If you increase the total number of particles, you drastically increase the computational time.

- **‘Number of sound particles per source (display)’** Defines the number of particles that are used for the particle animation.

Note: Most of the time, you need to consider only few hundreds or thousands of particles for the animation. Increasing this number will decrease the memory resources.

- **‘Random initialization number’** Initialize the random number series. If you select a number that is different from ‘0’, the random number series will always be the same. The starting number of the random number series will depend on the number you will consider for this parameter.

Warning: In a multithread simulation, I-Simpa/SPPS can not control the generation of random numbers. It means that this parameter will have no effect. Multithread simulation occurs when several frequency bands are considered in the simulation. To avoid multithreading, consider only one frequency band calculation.

- **‘Receiver radius (m)’** Defines the receiver radius (in m).
- **‘Simulation length (s)’** Defines the duration (in s) of the simulation.
- **‘Surface receiver export’** Select the kind of results that is exported.
 - **‘Soundmap: intensity’** Intensity level (in dB).
 - **‘Soundmap: SPL’** Sound pressure level (in dB).
- **‘Time step (s)’** Time step (in s) for the calculation.

25.4 Computational time optimization

The computational time depends mainly on the **Number of sound particles per source**. Thus, in the SPPS code, each sound particle is followed, at each ‘Time step’ during its propagation inside the 3D domain, experiencing absorption, reflection, diffusion... and so on, during the ‘Simulation length’ and/or until the particles disappear when their energy is too small.

Considering a small number of sound particles may not produce relevant results to proceed an interesting acoustic analysis of the 3D model. Thus, it is necessary to consider a *large* number of sound particles. Assigning the good value for *large* is difficult and depends mainly on the scene size and the absorbent nature of the scene (surface absorption and volume absorption).

Few recommendations to evaluate the computational time:

- the number of sound particles defined in SPPS ('Number of sound particles per source' parameter) is defined for **each sound source**. Considering N sound particles per source and M sound sources will generate $N \times M$ sound particles in the domain;
- one calculation is done for **each frequency band**: considering F frequency bands will be equivalent to a calculation with $F \times N$ sound particles for one source (or $F \times N \times M$ for M sources). Note that, by default, the computation is paralleleized on the processor core, for each frequency band (a frequency band for a given core). So the increase of computational time with the nimber of frequency band is not linear;
- in the '**Energetic**' mode, each sound particle is 'alive' during its propagation untill its energy is below a ratio of the initial energy (see "'Limit value of the particle extinction: ratio 10^n ' parameter). **Increasing** n will keep the sound particles longer in the domain, which increases the computational time;
- in the '**Random**' mode, each physical phenomena (absorption, diffusion) is applied by considering probabilistic approaches. Using this mode, the computational time is drastically decreased, but the quality of the results is aslo decrease. It is suggested to consider the 'Random' mode at the initial step of the study, and then to change to the 'Energetic' mode in order to obtain the final result.

GENERAL PRINCIPLE

26.1 Sound particle concept

In contrast to classical acoustic theory, where the study of the sound field is based on the propagation of a wave in a continuous material medium, the approach used in the **SPPS** code is geometric, and more particularly particulate: (joyce1974). In a field of complex propagation, the sound field is then decomposed into a multitude of elementary particles, called sound particles or phonons, without mutual interaction and carrying an infinitesimal energy and constant over time. These particles propagate at the speed of sound, either in straight line (homogeneous atmosphere), or with curved trajectories (in the presence of a velocity profile, atmospheric turbulence...), between two successive shocks with the obstacles and limits of the propagation medium. In a collision with an obstacle or a boundary of the propagation domain, the particles can be absorbed or reflected in a new direction of propagation. Geometric acoustics thus become a special case of particle dynamics, so that a sound field can be likened to a gas of sound particles. Under these conditions, the distribution of the energy of the sound field is assimilated to the distribution of the sound particles. Since the local density of sound energy is proportional to the local density of phonons, the only difficulty lies in determining the distribution of these sound particles over time. Although analytical approaches may be considered under certain conditions (lepolles2003), we have chosen an all-digital approach here. With the evolution of computing powers, the numerical simulation of a system composed of several hundred thousand particles is no longer a major difficulty.

26.2 Sound particles vs. sound rays

This concept is relatively similar to the traditional methods of sound beam tracing, implemented in most current closed or open environment noise prediction software. Nevertheless, even if in form these two methods are comparable, the major differences lie in the management of the sound energy carried by sound particles and sound rays. For the latter method, each sound ray carries an intensity whose amplitude decreases proportionally with the square of the propagation distance, thus simulating the acoustic radiation of a spherical source (geometric dispersion). On the other hand, in the concept of sound particles, each particle carries an elementary energy ϵ , whose amplitude does not vary according to the propagation distance.

On the other hand, all the physical phenomena occurring during propagation (reflection, absorption, diffusion, atmospheric absorption) are not considered in the same way in the two approaches. In the “sound rays” approach, these phenomena occur in the form of a weighting applied to the sound intensity carried by a ray. For example, when in contact with an absorption coefficient wall α , the intensity of the beam after reflection will be weighted by the coefficient $(1 - \alpha)$. In the “sound particles” approach, some of these physical phenomena can be considered probabilistically. When in contact with an absorption coefficient wall α , the particle could for example have a probability $(1 - \alpha)$ of being reflected, and a probability α of being absorbed. In the latter case, it disappears from the propagation domain.

One of the major advantages of the sound particle concept lies above all in the consideration of diffuse reflections on the walls and diffusion by encumbrances. In sound ray methods, wall reflections are determined by image source

methods that allow only specular reflections, perfectly deterministic, and prohibit the consideration of diffuse reflection phenomena. In the particulate approach, and as we will see below, these diffuse reflections are considered in a probabilistic way, the angle of reflection being able to be chosen according to any arbitrary law of reflection. The same applies to the consideration of diffusion by objects distributed in the propagation medium (machines in industrial premises for example). In the sound ray approach, it is necessary to know exactly the shape and position of the object in order to calculate in a deterministic way the angle of reflection of a sound ray on it. In the particle approach, scattering objects can be considered statistically, with no knowledge other than their spatial distribution and average shape, which significantly reduces costs in terms of calculating and defining the field of study. This statistical (or probabilistic) description, which can be generalized to most of the physical phenomena involved in sound propagation, is therefore the main strength of the sound particles approach compared to the “sound rays” approach.

26.3 Code principle

The principle of simulations is therefore based on the tracking of sound particles, carrying an initial energy ϵ , emitted from one or more sound sources, in a volume chamber V totally or partially closed. Each particle is propagated along rectilinear or curved paths, until it collides with a wall or scattering object. At each collision, the sound particle (or part of its energy) can be absorbed, reflected or transmitted, depending on the absorption and transmission coefficient of the wall or object. As the processing of certain physical phenomena can be performed by random number draws, this simulation procedure can therefore be likened to a Monte Carlo method: *raw-latex: cite [burns1990]*. The physical phenomena simulated by these draws of random numbers will be all the better respected as the number of random draws will be large, that is to say that the initial number N of sound particles will also be very large. Nevertheless, more N will be large, longer will be the duration of the simulations. The choice of N is therefore a compromise between the computation time and the accuracy of the results, but is also a function of the geometry and acoustic characteristics of the propagation domain. For example, the more the propagation domain will be absorbing (at the walls, scattering objects, atmospheric absorption), the more it will be necessary to consider sound particles to have a satisfactory description of the physical phenomena.

In the current version of the “SPPS” code, two calculation modes are proposed:

- **Random Modeling:** In this mode, the energy of the particle is constant. The phenomena of absorption and atmospheric absorption are considered statistically: depending on the values of the atmospheric absorption and the absorption coefficients of the materials, the particles can be made to disappear completely from the propagation domain, or to remain in the domain with the same energy. The other physical phenomena (diffusion by a congestion, diffuse reflection) are also treated statistically. As the number of sound particles decreases over time, the calculation time decreases gradually. Moreover, the density of sound energy at a point of the domain is then proportional to the number of sound particles at the same point.
- **Energetic modeling:** In this mode, the energy of the particle is weighted according to the values of the atmospheric absorption and the absorption coefficients of the materials. The other physical phenomena (diffusion by congestion, diffuse reflection) are also treated randomly. Since in this mode, the number of sound particles is constant, the duration of the numerical simulations is longer than for the first mode. Moreover, the sound energy density at a point of the domain is then proportional to the sum of the energy of the sound particles at this same point.

26.4 Spatial meshing

Unlike conventional calculation codes based on ray and sound beam throws, the SPPS code is based on a volumetric mesh of the propagation domain.

Benefits:

- the mesh of the propagation domain allows to optimize the propagation of sound particles in the domain. Indeed, a pre-processing makes it possible to determine for each mesh of the domain its position with respect to the limits

of the domain, its position inside or outside a space requirement, as well as general properties, such as the speed of sound in the mesh and the meteorological conditions (*via* the height with respect to the ground for example). The tracking of a sound particle in the domain is therefore accelerated by the knowledge of the properties of the meshes adjacent to the mesh containing the particle at the current time step;

- the volume mesh allows to realize a surface mesh of the domain boundaries, which can be used to build surface receivers, on any boundary of the domain;
- the realization of a volumetric mesh requiring that the geometric model be perfectly closed, thus ensuring that no sound particles can disappear from the propagation domain due to an opening (faces of the model not perfectly joined for example).

Disadvantages:

- the realization of the mesh requiring that the geometric model is perfectly closed, this supposes that the model is perfectly constructed. Calculation codes based on sound rays do not have this constraint, since they “allow” a ray to “leave” the domain and propagate to infinity.

26.5 Temporal meshing

In parallel, the monitoring of sound particles is carried out in constant time steps, and not in continuous time. It is therefore necessary to choose a sufficiently small time step so that the physical phenomena present in the sound emission and acoustic propagation processes can be applied to a given time step.

Mesh supported by the code SPPS:

- tetrahedral mesh;

MODELLING OF PHYSICAL PHENOMENA IN SPPS

27.1 Sound source modelling

27.1.1 General formalism

Acoustic directivity and probability density of sound particles emission

From a numerical modelling point of view, the problem of sound emission from a point source is to attribute to the sound particles, at a given moment (which can be the origin of time), the exact position of the source, propagation directions in accordance with the directivity of the source. For example, in the case of an omnidirectional sound source, it is necessary to verify that the sound particles, once emitted, are uniformly distributed over a sphere centred on the source (see corresponding figure: *Elementary geometry for a sound source*). The first solution that comes to mind is to make a deterministic discretization of the source's radiation in spherical coordinates; the method would consist in considering emission angles θ and ϕ by constant steps. Nevertheless, such discretization of emission angles causes an artificial concentration of sound particles at the poles of the sphere symbolizing the source.

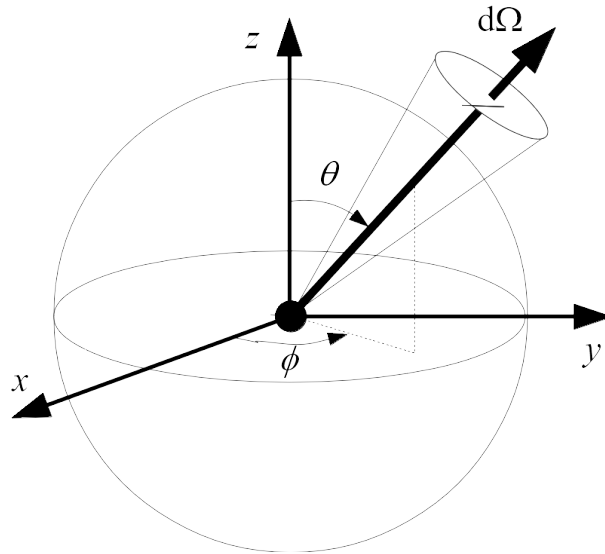


Fig. 1: Elementary geometry for a sound source

The direction of emission of a particle is defined in spherical coordinates by the angles ϕ (azimuth) between 0 and 2π , and θ between 0 and π (polar angle), or by the elementary solid angle $d\Omega$.

It is in fact necessary to check the respect of the number of particles emitted by elementary solid angle $d\Omega = d\phi \sin \theta d\theta$, checks the directivity of the source, either in a deterministic or random way (figure[geo_source]). In

our case, we chose a random draw of the initial directions of the particles. It is then necessary to define the probability densities $g(\phi)$ and $p(\theta)$, respectively for the angles $\phi \in [0, 2\pi]$ and $\theta \in [0, \pi]$, such as the probability density $Q(\theta, \phi)$ of emission in the direction (θ, ϕ) (or probability density $F(\Omega)$ of emission in the elementary solid angle Ω) either in accordance with the directivity of the source. In practice, $Q(\theta, \phi)$ is nothing more than the directivity of the source. By definition, these probability densities verify the following relationship:

$$\begin{aligned} \int F(\Omega) d\Omega &= \iint Q(\theta, \phi) \sin \theta d\phi d\theta \\ &= \int_0^{2\pi} g(\phi) d\phi \int_0^\pi p(\theta) \sin \theta d\theta \\ &= 1. \end{aligned}$$

From the point of view of sound power, this is equivalent to considering that the power W of the source is well distributed over the surface of a sphere centered on the source:

$$\iint W Q(\theta, \phi) \sin \theta d\phi d\theta = W.$$

If the densities $g(\phi)$ and $p(\theta)$ are independent, they check:

$$\int_0^{2\pi} g(\phi) d\phi = 1, \quad (27.1)$$

and

$$\int_0^\pi p(\theta) \sin \theta d\theta = 1. \quad (27.2)$$

Source power and elementary particle energy

Acoustically, for a period of time Δt , a power source W emits an amount of energy $E = W \times \Delta t$. From a particle concept point of view, each particle carries an elementary energy ε_0 . If the source emits N sound particles, the energy conservation between the two approaches therefore requires:

$$N \times \varepsilon_0 = W \times \Delta t,$$

then

$$\varepsilon_0 = \frac{W}{N} \times \Delta t.$$

27.1.2 Omnidirectionnal source

Physical description

An omnidirectional sound source is a source that radiates in all directions with the same amplitude. The directivity of the source is similar to a sphere. The directivity coefficient Q associated with the directivity of the source is equal to 1.

Modeling

In the case of an omnidirectional source, the densities $g(\phi)$ and $p(\theta)$ being uniform, the relationships (??) and (??) verify:

$$\int_0^{2\pi} g(\phi) d\phi = \int_0^{2\pi} A d\phi = 1$$

and

$$\int_0^\pi p(\theta) \sin \theta d\theta = \int_0^\pi B \sin \theta d\theta = 1,$$

where $A = 1/2\pi$ and $B = 1/2$ are two normalization constants. In practice, the method consists first of drawing an angle θ between 0 and 2π as follows:

$$\phi = 2\pi \times u \in [0, 2\pi]$$

where u is a random number between 0 and 1, and defined by a uniform distribution. If the angle θ is chosen in the same way (*i.e.* $\theta = \pi \times v$, v being a random number with a uniform distribution between 0 and 1), the distribution of the emission directions does not respect the elementary solid angle consistency, since the condition ([cond_p]) is not verified. In this case, the directions around the poles would be preferred. It is actually necessary to determine the angles θ which verify a distribution proportional to $\sin \theta$. In this simple case, the procedure consists in applying the inverse cumulative distribution function method. According to the relationship (??), the probability $f(\hat{\theta})$ of drawing an angle $\theta < \hat{\theta}$ is given by:

$$f(\hat{\theta}) = \frac{1}{2} \int_0^{\hat{\theta}} \sin \theta d\theta = \frac{1}{2} [\cos \hat{\theta} - 1].$$

Knowing that this distribution is between 0 and 1, the choice of the angle $\hat{\theta}$ is made by randomly drawing a number $v \in [0.1]$, following a uniform distribution, such as:

$$\hat{\theta} = \cos^{-1}(2v - 1) \in [0, \pi].$$

The propagation vectors $\mathbf{v}(v_x, v_y, v_z)$, of standard c (sound velocity), are then defined by the relationships:

$$\mathbf{v} = \begin{cases} v_x = c \cos \theta \sin \phi \\ v_y = c \sin \theta \sin \phi \\ v_z = c \cos \phi \end{cases}$$

Verification

The figures *Distribution of emission angles ϕ for an omnidirectional source* and *Distribution of emission angles θ for an omnidirectional source* shows an example of the distribution of angles θ and ϕ obtained according to this printing method with 10000 achievements. It is easy to see that the angle distribution checks the theoretical distribution. It is understood that the quality of the random draw depends on the method of generating random numbers, and that compliance with theoretical distributions increases with the number of draws.

27.1.3 Unidirectional source

Physical description

A unidirectional sound source is a source that radiates in a single direction of space, defined by the angles θ and ϕ , at the point of emission and in the absolute reference point. The directivity coefficient associated with the directivity of the source is then defined by:

$$Q(\theta, \phi) = \delta(\theta, \phi) = \delta(\theta) \times \delta(\phi),$$

where δ is the Dirac distribution function.

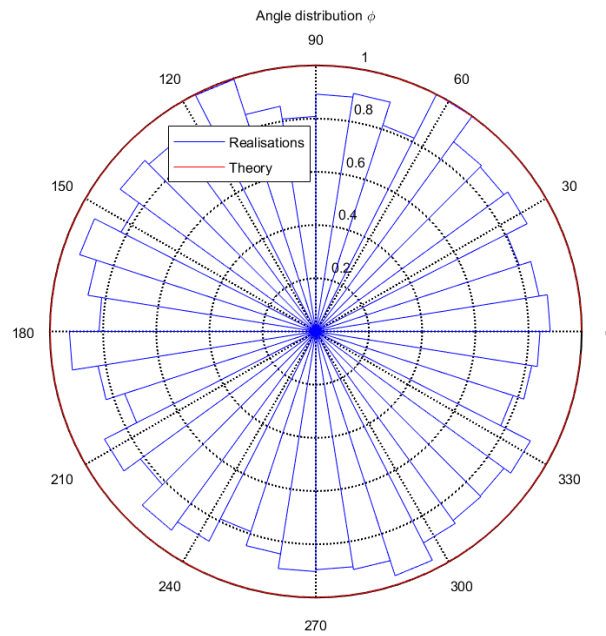


Fig. 2: Distribution of emission angles ϕ for an omnidirectional source

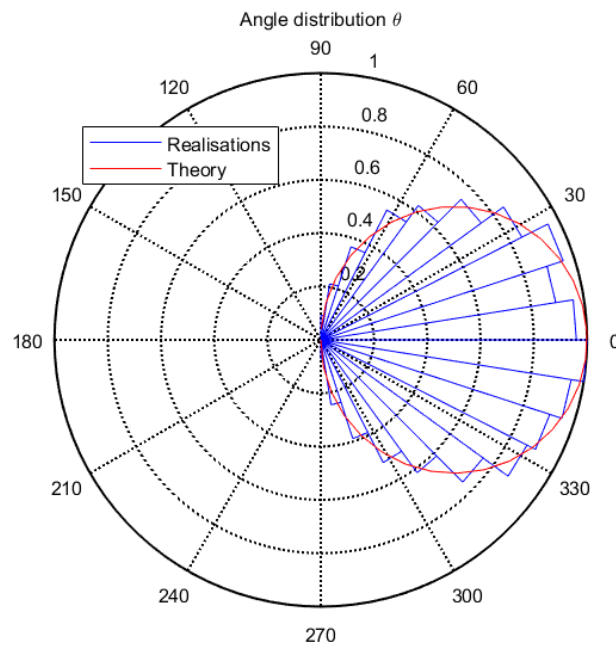


Fig. 3: Distribution of emission angles θ for an omnidirectional source

Modelling

The modelling of this type of source is entirely deterministic and therefore does not pose a problem. It is enough to define a vector $\mathbf{s}(s_x, s_y, s_z)$ in the absolute reference frame of the scene, such that:

$$\mathbf{s} = \begin{cases} s_x = s \cos \theta \sin \phi \\ s_y = s \sin \theta \sin \phi \\ s_z = s \cos \phi \end{cases}$$

27.1.4 Plane sources (XY, YZ and XZ)

Physical description

A ‘plane’ type sound source is a source that radiates in a plane. By default, we consider three sources XY, YZ and XZ defined by the three reference planes (xOy) , (yOz) and (xOz) respectively.

Modelling

The procedure consists in randomly determining the direction of propagation of a particle in a plane. Using the angle convention presented in figure *Elementary geometry for a sound source*, we will have:

$$\begin{cases} \text{XY plane: } \theta = 2\pi \times u & \text{and} & \phi = \pi/2 \\ \text{YZ plane: } \theta = 0 & \text{and} & \phi = 2\pi \times u \\ \text{XZ plane: } \theta = \pi/2 & \text{and} & \phi = 2\pi \times u \end{cases}$$

where u refers to a random number between 0 and 1. Thereafter, the propagation vector $\mathbf{v}(v_x, v_y, v_z)$, of standard c (sound velocity), is defined by the relationships:

$$\mathbf{v} = \begin{cases} v_x = c \cos \theta \sin \phi \\ v_y = c \sin \theta \sin \phi \\ v_z = c \cos \phi \end{cases}$$

27.2 Acoustic propagation modelling

27.2.1 Acoustic propagation

Physical description

In the absence of absorption and reflection on the walls of the domain or on objects, the decrease in sound intensity from an omnidirectional source is written:

$$I = \frac{Q}{4\pi r^2},$$

where r is the distance to the source, and Q is the directivity of the source in the direction of observation ($Q = 1$ for an omnidirectional source). This decrease reflects the phenomenon of geometric dispersion, which describes the “spreading” of a spherical wave as it propagates.

Modeling

Considering the method presented for an omnidirectional source, the geometric dispersion is automatically respected. Indeed, the proposed numerical method allows to obtain a uniform distribution of particles over an elementary solid angle. On a sphere, the particle distribution $n(r)$ (in m^2) is therefore equal to:

$$n(r) = \frac{N}{4\pi r^2},$$

where N is the number of particles. The particle distribution therefore verifies the same decrease as the intensity. It should be noted, however, that the further away the observation point is from the source, the more sound particles will be required.

Verification

The figure *Verification of the respect of the geometric dispersion with the SPPS code* shows the numerical results of the free field propagation (the free field is simulated by considering a long corridor with perfectly absorbent limits), for an omnidirectional sound source, without atmospheric absorption (number of particles $N = 20 \times 10^6$). The agreement is excellent.

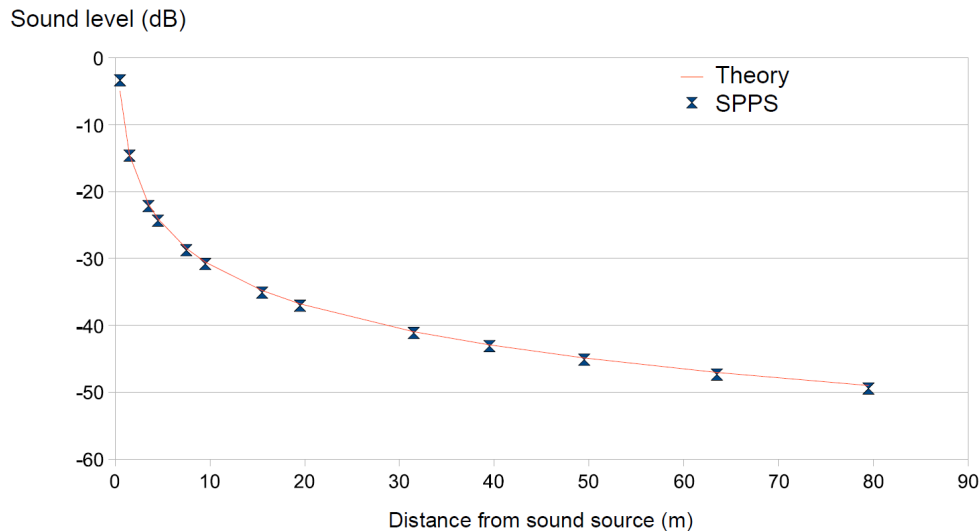


Fig. 4: Verification of the respect of the geometric dispersion with the SPPS code

The numerical simulations are compared with the theoretical decrease ($N = 20 \times 10^6$ particles). The marker presents the result of the simulation with the **SPPS** code. The solid line shows the theoretical geometric dispersion.

27.2.2 Atmospheric absorption

Physical description

During its propagation in air, a sound wave is partially attenuated by particular physical mechanisms (“classical” transmission processes, molecular absorption due to rotational relaxation, molecular absorption due to vibratory relaxation of oxygen and nitrogen) [BSPE84]. Thus, after a propagation distance r , the amplitude p_t of the sound pressure decreases according to the relationship (ISO 9613-1):

$$p_t = p_i \exp \left(-\frac{\ln(10)}{20} \alpha_{\text{air}} \times r \right)$$

where p_i is the initial pressure. Considering that the sound intensity is proportional to the square of the sound pressure,

$$I \propto p_t^2 \propto \exp\left(-2\frac{\ln(10)}{20}\alpha_{\text{air}} \times r\right)$$

and writing that the intensity I of the sound wave decreases with the relationship:

$$I = I_0 \exp -m r, \quad (27.3)$$

where I_0 is the initial intensity of the sound wave. Then, the atmospheric absorption coefficient m (in Np/m) can be expressed from the atmospheric absorption coefficient α_{air} (in dB/m), by the relationship:

$$m = \frac{\ln 10}{10} \alpha_{\text{air}}.$$

In **SPPS** code, the atmospheric absorption coefficient α_{air} is calculated according to *ISO 9613-1*, considering the centre frequency of each frequency band of calculation according to the reference standard (*cf.* paragraph 8.2.1 of *ISO 9613-1*). This approximation is considered valid if the product of the source-receiver distance (in km) by the square of the centre frequency (in kHz) does not exceed 6 km.kHz² for the third octave bands and 3 km.kHz² for the octave bands. However, the propagation distance must not exceed 6 km for third octave bands and 3 km for octave bands, regardless of the centre frequency considered.

Random modelling

By choosing the ‘random’ calculation mode, atmospheric absorption is taken into account as a probability of the sound particle disappearing during its displacement. The corresponding probability density can be defined from the relationship (??):

$$f(r) = \exp -m r.$$

This quantity expresses the probability that the particle will not be absorbed during its propagation distance r . The probability density $f(r)$ is well between 1 and 0 (see figure *Modelling of atmospheric absorption by a random process*):

- $f(0) = 1$, the probability is maximum, the particle cannot be absorbed if it does not move;
- $f(\infty) = 0$, the probability is zero since the particle cannot spread infinitely.

It is also easily verified that the probability of propagation is independent of the previous probability of propagation:

$$f\left(\sum_{n=1}^N r_i\right) = \prod_{i=1}^N f(r_n).$$

Taking atmospheric absorption into account is relatively simple. It is sufficient to consider a uniform random number ζ between 0 and 1, at each time step, and to compare this number to the probability density $f(d_0)$ corresponding to an elementary displacement $d_0 = c\Delta t$ on a time step Δt . If this number ζ is less than $f(d_0)$, there will be propagation. Otherwise, there will be atmospheric absorption, thus disappearing the particle. Even on a small number of particles, this method makes it possible to correctly take into account atmospheric absorption.

Energetic modelling

By choosing the ‘energetic’ calculation option, the energy of the particle is weighted throughout its movement, using the relationship (??).

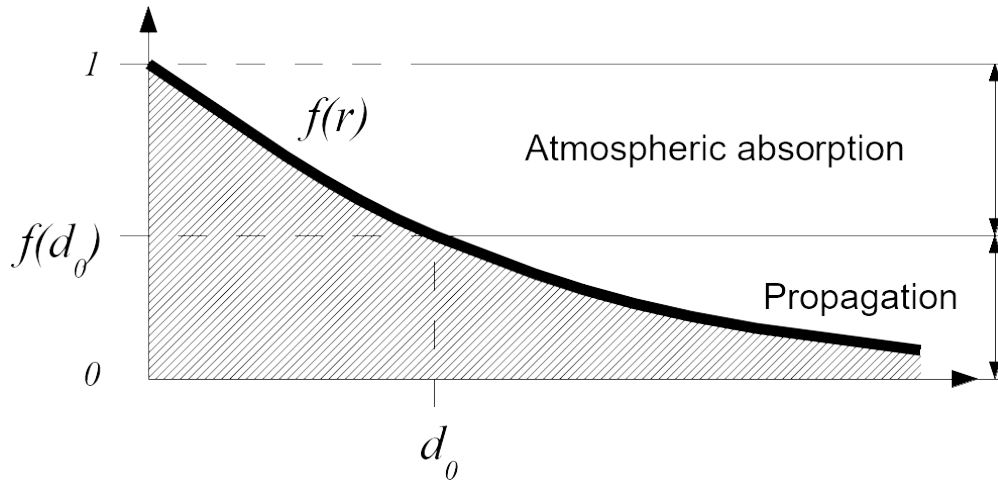


Fig. 5: Modelling of atmospheric absorption by a random process
The curve $f(r)$ separates the propagation domain from the atmospheric absorption domain.

Verification

As an illustration, the figure *Illustration of the modelling of atmospheric absorption in the SPPS code* shows the sound decrease calculated by the **SPPS** code at 10 kHz for classical atmospheric conditions ($T = 20$ Celsius, $H = 50$ %, $P = 101325$ Pa, or $m = 0.036 \text{ m}^{-1}$), for both types of modelling, compared to the theoretical decrease presented to the relationship (??); the simulation is identical to the one for the verification of the respect of the geometric dispersion.. As expected, energetic modelling gives a better result than random modelling, the average deviation from the theoretical curve being 0.17 dB and 0.41 dB respectively, the calculation times being similar.

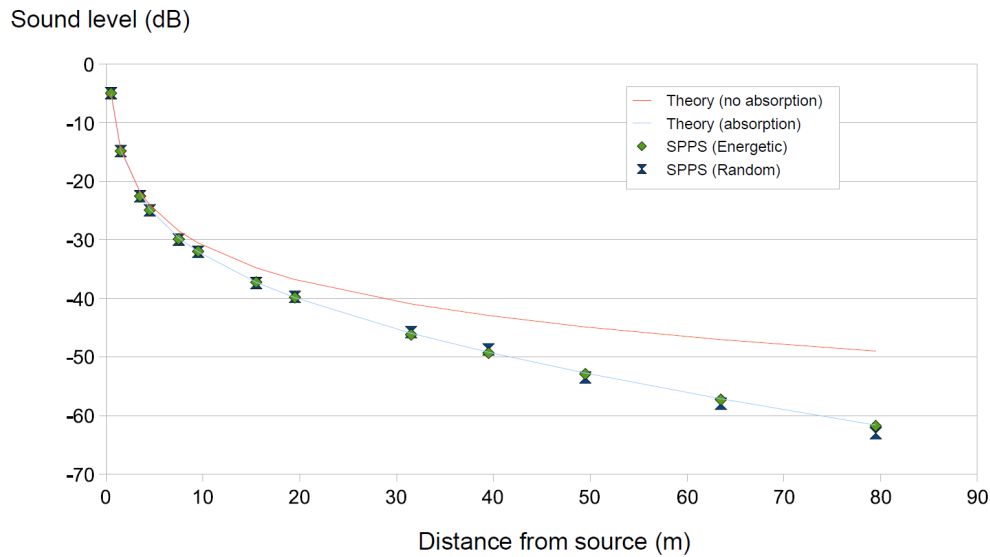


Fig. 6: Illustration of the modelling of atmospheric absorption in the SPPS code
Comparison with the theoretical decrease (with et without atmospheric absorption) Simulations performed with ($N = 20 \times 10^6$ particles) at 10 kHz, for conventional atmospheric conditions: $T = 20$ Celsius, $H = 50$ %, $P = 101325$ Pa, or $m = 0.036 \text{ m}^{-1}$.

27.2.3 Acoustic velocity profile

Physical description

In outdoor environments, acoustic propagation can be influenced by micrometeorological conditions governed by thermal (heat transfer) and aerodynamic (wind profiles) laws. The phenomena have a very strong interaction with the ground (topography, surface and subsoil temperature, hygrometry, crops, forests, obstacles, buildings, etc.). In addition, they evolve rapidly in time and space, making their analytical description and numerical modelling complex. The thermal and aerodynamic factors that influence propagation are as follows:

- Thermal factors: heat exchanges between the ground and the lower layer of the atmosphere lead to a variation in air temperature as a function of the height above the ground, and therefore to a variation in sound velocity.
- Aerodynamic factors: due to the roughness of the ground surface, wind speed is always higher at height than at ground level. In a given situation, the speed of sound in the presence of wind corresponds to the algebraic sum of the speed of sound in the absence of wind and the projection of the wind vector on the direction of propagation considered. This speed therefore varies according to the height above the ground.

By analogy with the laws of optics (see figure *Schematic representation of the acoustic refraction in the atmosphere in the presence of a vertical velocity profile*), the effect of atmospheric conditions on acoustic propagation can be described through the expression of the acoustic index n of the propagation medium. If placed in a vertical section, this index is assumed to vary with altitude z and with source-receptor distance r , such that:

$$n(r, z) = \frac{c(r, z)}{c_0} = \langle n(r, z) \rangle + \mu(r, z),$$

where c the reference one. Thus, two distinct phenomena can be distinguished that affect acoustic propagation, refraction and atmospheric turbulence. These phenomena are respectively related to the deterministic parts $\langle n \rangle$ and stochastic μ of the propagation medium index. In practice, these refraction and turbulence phenomena co-exist and interact, leading to complex propagation conditions, as well as a very wide dispersion of the sound levels encountered *in situ*, all of which are identical (topography, soil type, source-receptor geometry, etc.).

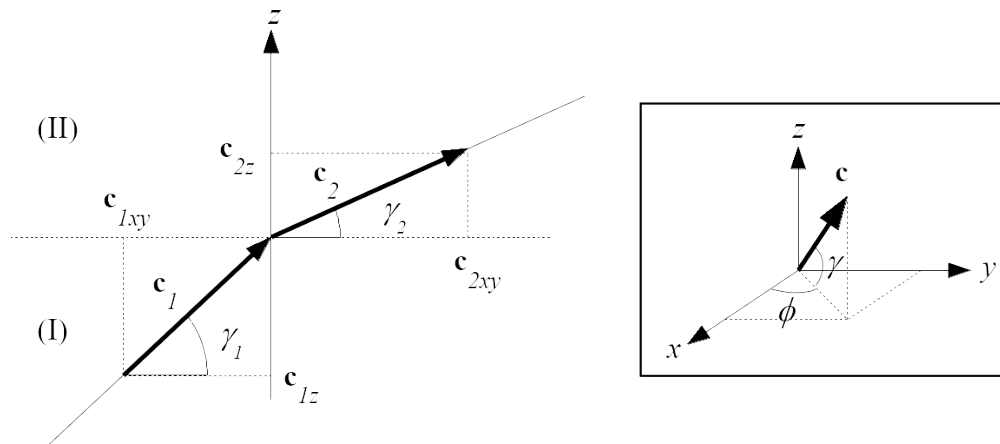


Fig. 7: Schematic representation of the acoustic refraction in the atmosphere in the presence of a vertical velocity profile

Acoustic velocity profile model

The average sound velocity profile thus depends on the average wind and temperature profiles. This velocity profile can be described analytically, depending on whether it follows a linear (“lin”), logarithmic (“log”), hybrid (“log-lin”) or other law. The “log” profiles thus have the advantage of translating the very strong vertical gradient of sound velocity in the immediate vicinity of the ground, but do not accurately reflect the more moderate evolution with altitude above a

certain height. On the other hand, the “lin” profile minimizes the effects in the vicinity of the ground and are therefore not representative of reality when placed at very low altitude. A good compromise therefore consists in using hybrid profiles of the type “log-lin” (valid especially for a so-called “stable” atmosphere), expressed through the parameters a_{\log} and b_{lin} which appear in the analytical expression of the vertical profile of the effective sound velocity:

$$\langle c(z) \rangle = c_0 + a_{\log} \times \ln \left(1 + \frac{z}{z_0} \right) + b_{\text{lin}} \times (z - z_0), \quad (27.4)$$

où z_0 is the roughness parameter, whose typical values range from 10^{-2} m for short grass to several meters in urban areas. The vertical gradient is then expressed by deriving according to the variable z :

$$\frac{\partial \langle c(z) \rangle}{\partial z} = \frac{a_{\log}}{z_0} + b_{\text{lin}}.$$

The main effect of propagation in a medium of variable speed is to bend the sound rays downwards or upwards depending on whether the vertical gradient of sound velocity is positive (*conditions (very) favorable* to propagation) or negative (*conditions (very) unfavorable* to propagation) respectively. The transient state between these two states represents *homogeneous* propagation conditions.

- **Homogeneous:** the speed c is the same at any point in the domain and equal to the reference speed c_0 , the latter being calculated as a function of temperature and humidity conditions by the formula:

$$c_0 = 343.2 \sqrt{\frac{T}{T_{\text{ref}}}},$$

where T is the temperature (K), and $T_{\text{ref}} = 293.15$ K the reference temperature (*ISO 9613-1*).

- **Very unfavorable :** $a_{\log} = -1$ and $b_{\text{lin}} = -0.12$;
- **Unfavorable :** $a_{\log} = -0.4$ and $b_{\text{lin}} = -0.04$;
- **Favorable:** $a_{\log} = +0.4$ and $b_{\text{lin}} = +0.04$;
- **Very favorable:** $a_{\log} = +1$ and $b_{\text{lin}} = +0.12$.

The curvature of the radius, at the boundary of zones (I) and (II), is obtained by applying the Huygens-Fresnel construction, resulting in the following Snell-Descartes law [Sal01] (see figure *Schematic representation of the acoustic refraction in the atmosphere in the presence of a vertical velocity profile*):

$$\frac{\cos \gamma_1}{c_1} = \frac{\cos \gamma_2}{c_2}, \quad (27.5)$$

where c_1 and c_2 are respectively the norms of the propagation vectors \mathbf{c}_1 and \mathbf{c}_2 , and where the angles γ_1 and γ_2 are defined with respect to the horizontal axis in the plane (xOy). By construction, the projection of the propagation direction in the plane (xOy), defined by the angle ϕ in spherical coordinates, is preserved.

Modelling

Whatever the method of calculation chosen, the speed is taken into account is identical. At each time step, the speed standard is calculated according to the chosen speed profile, based on the relationship (??). To determine the new direction of propagation, due to the change in speed, the relationship (??) must then be applied. Knowing the angle θ_1 of the initial propagation direction, the new propagation direction is defined by:

$$\cos \gamma_2 = \frac{c_2}{c_1} \cos \gamma_1 = \frac{c_2}{c_1} \frac{\sqrt{c_{1x}^2 + c_{1y}^2}}{c_1}. \quad (27.6)$$

The coordinates of the propagation vector are then obtained by:

$$\mathbf{c}_2 = \begin{cases} c_{2x} = c_2 \cos \gamma_2 \cos \phi \\ c_{2y} = c_2 \cos \gamma_2 \sin \phi \\ c_{2z} = c_2 \sin \gamma_2 \end{cases}$$

with

$$\sin \phi = \frac{c_{1y}}{\sqrt{c_{1x}^2 + c_{1y}^2}},$$

and

$$\cos \phi = \frac{c_{1x}}{\sqrt{c_{1x}^2 + c_{1y}^2}}.$$

From a numerical simulation point of view, the calculation of $\cos \gamma_2$ by the relationship (??) can give values higher than 1 which is obviously not physical. This case occurs when the curvature (*turning point*) of a radius is reversed. To avoid this problem and impose a change in curvature, the procedure consists in imposing the value from γ_2 to $1 - \epsilon$ (ϵ being a very small value) and changing the orientation of the component c_{2z} .

As an example, the figure [illustration_refraction] shows two illustrations of how acoustic refraction is taken into account using this method. This figure can be compared directly with the examples given in the reference [Sal01] (figures 4.5 and 4.6).

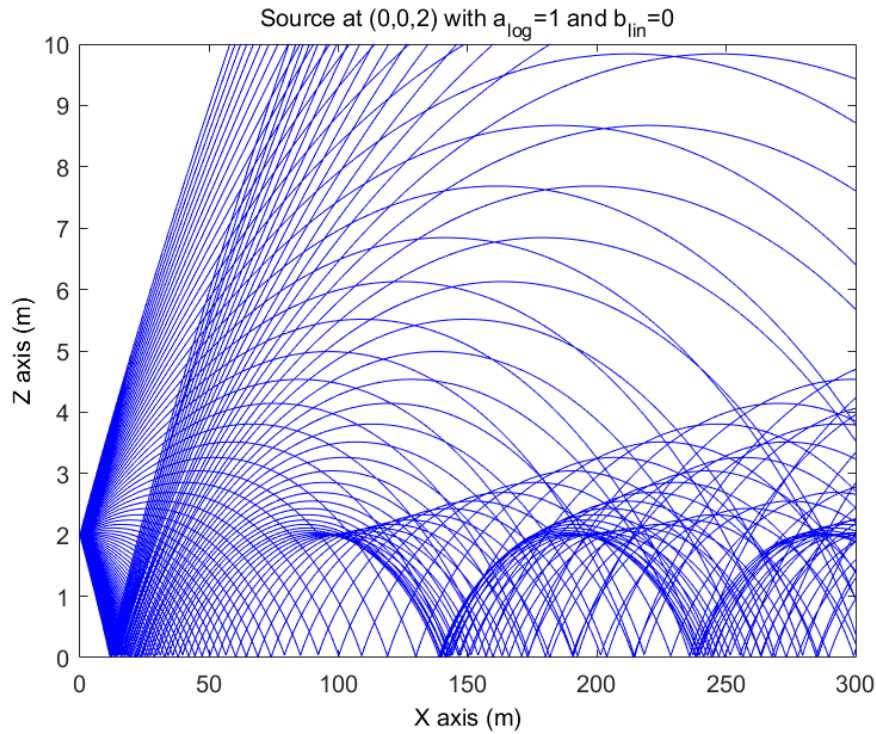


Fig. 8: Illustration of the modelling of acoustic refraction with SPPS in downward condition

27.2.4 Diffusion by fittings

The presence of a large number of objects on the path of a sound wave can lead to a diffusion process. This process can be simulated in a deterministic way, by modelling each object individually. When the number of objects becomes

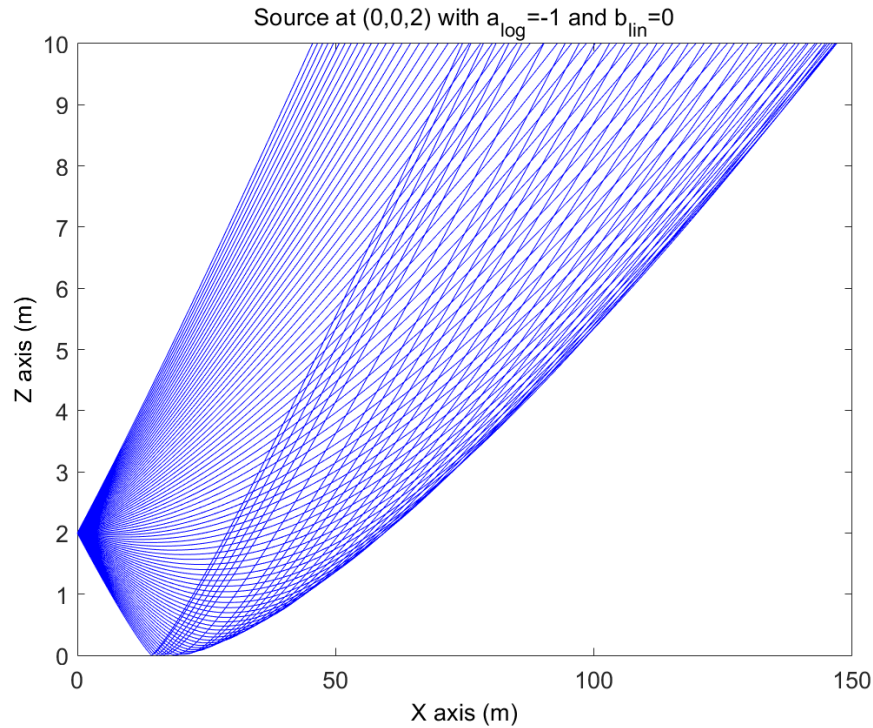


Fig. 9: Illustration of the modelling of acoustic refraction with SPSS in upward condition

large, and these objects are of similar sizes (example of an industrial hall with many machines (without acoustic emission) or similar boxes), it may be more interesting to statistically model this fitting zone.

Physical description

In order to take into account the diffusion and absorption of diffusing (or scattering) objects distributed in the propagation medium, we considered an approach similar to that of Ondet and Barbry [OB89], which itself is based on the work of many authors [Kut81][Aul85][Aul86][Lin82], among others. In this approach,

- the diffusing objects (or obstacles, or fittings, or scattering objects...) are considered as punctual. Sound particles are returned in all directions of space at each collision with a scattering object (except in the case of absorption). This assumption is generally valid when the wavelength is in the order of magnitude of the characteristic dimension of the obstacle;
- the scattering phenomenon follows a Poisson process, which means that the probability of collision of a sound particle with a scattering object follows a Poisson's law. The collision probabilities are independent of each other (the collision probability during time t and $t + dt$ is independent of collisions before time t);
- the objects in the fitting zone do not produce particles (*i.e.* these objects are not sound sources).

A sound wave propagating in the environment may come into contact with scattering objects, causing the wave to diffract simultaneously and, in part, to absorb it. By analogy, in the particle approach, a particle that comes into contact with a scattering object can either be absorbed or reflected in another direction of propagation (see figure: *Schematic representation of the acoustic diffusion by obstacles in a fitting zone*). At the macroscopic scale, *i.e.* considering all the sound particles simultaneously, a diffusion process occurs, characterized by:

- the absorption coefficient α_c of the scattering objects;

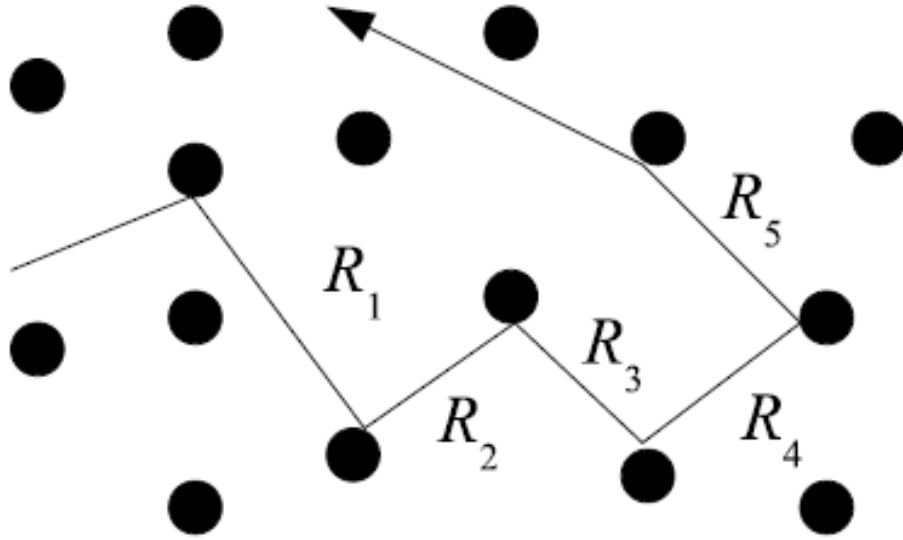


Fig. 10: Schematic representation of the acoustic diffusion by obstacles in a fitting zone

- the bulk density n_c of the propagation medium, defined by the number N_c of obstacles present in the volume V_c :

$$n_c = \frac{N_c}{V_c}.$$

- the average scattering section q_c , *i.e.* the average surface of the scattering object, seen by a particle in a given incident direction. In practice, this data is very difficult to obtain, if not impossible, since the scattering objects are of complex and different shapes. In this condition, it is common to assimilate the diffusing object to a sphere, having the same external surface s_c as the object. Whatever the angle of incidence, the visible cross-section of the sphere (mean scattering cross-section) will be equal to a quarter of the total surface area of the sphere, or:

$$q_c = \frac{s_c}{4}.$$

- the average diffraction section per unit volume ν_c , also called the diffusion frequency, by

$$\nu_c = n_c q_c,$$

– if all the scattering objects are identical, or

$$\nu_c = \frac{1}{V} \sum_{p=1}^{N_c} \frac{s_{c_p}}{4}$$

– if each object diffusing p is defined by its surface s_{c_p} . In practice, and in the rest of the document, the diffusing objects will be considered uniform in the same volume of diffusion. Nevertheless, within the same propagation volume, several separate diffusion volumes can be considered.

Since the scattering phenomenon follows a Poisson's law, the probability that a sound particle will collide with scattering objects after a time t_k is equal to:

$$W_k(c t_k) = \frac{(\nu_c c t_k)^k}{k!} \exp(-\nu_c c t_k),$$

where $c t_k$ is the distance covered during a time t_k , which can be expressed from the distance R_p between two collisions (see figure: *Schematic representation of the acoustic diffusion by obstacles in a fitting zone*):

$$c t_k = \sum_{p=1}^k R_p.$$

Since the collision probabilities are independent of each other [OB89], it is easy to show that the random variables R_i (noted R afterwards) follow the following probability density $f(R)$:

$$f(R) = \nu_c \exp(-\nu_c R). \quad (27.7)$$

The average free path λ_c (average distance between two collisions) is simply obtained by expressing the first moment of the above probability density, namely:

$$\lambda_c = \int_0^\infty R f(R) dR = \frac{1}{\nu_c}.$$

Modelling

By definition, the cumulative distribution function, associated with this probability density, is defined by the following relationship:

$$p(\hat{R}) = \int_0^{\hat{R}} f(R) dR = 1 - \exp(-\nu_c \hat{R}). \quad (27.8)$$

This cumulative distribution function simply expresses the probability that the particle will collide with a scattering object during a long path \hat{R} . This function is therefore null for $\hat{R} = 0$ and equal to 1 for $\hat{R} = \infty$. The numerical simulation of the diffusion process is performed by the inverse cumulative distribution function method, obtained by reversing the relationship (??), i.e.:

$$\hat{R} = -\frac{1}{\nu_c} \ln [1 - p(\hat{R})]. \quad (27.9)$$

The cumulative distribution function being between 0 and 1, it can be assimilated to a random variable ξ between 0 and 1. By drawing a succession of random variables ξ_i , we can determine a succession of paths of length \hat{R}_i that satisfies the distribution function (??) of our problem:

$$\hat{R}_n = -\frac{1}{\nu_c} \ln [1 - \xi_n].$$

An example of a random draw using the inverse cumulative distribution method is shown in Figure [fig:verification_diffusion_encrowding]. The comparison with the theoretical distribution shows an excellent behaviour of the method.

When a sound particle n enters a fitting zone, it is associated with a collision distance R_n with an object of the congestion by applying the relationship (??). As the sound particle propagates in the crowded area, a test is performed to determine if the cumulative distance d_n of the particle in the crowded area is less or more than R_n . If the distance traveled d_n is greater than the collision distance R_n the particle collides with an object. In the ‘Energetic’ mode, the energy of the sound particle is weighted by the average absorption coefficient α_c of the space requirement and continues its propagation in a random direction (uniform distribution). In the ‘Random’ mode, a new random draw on a uniform variable u allows to determine if the particle is absorbed by the diffusing object (same procedure than for considering absorption by a wall), or reflected in a uniform direction. After each collision, the cumulative propagation distance is reset to zero, and a new draw is made to determine the next collision distance. Whatever the calculation mode chosen, as long as the distance travelled by the particle is less than the collision distance, the particle continues its path without changing direction.

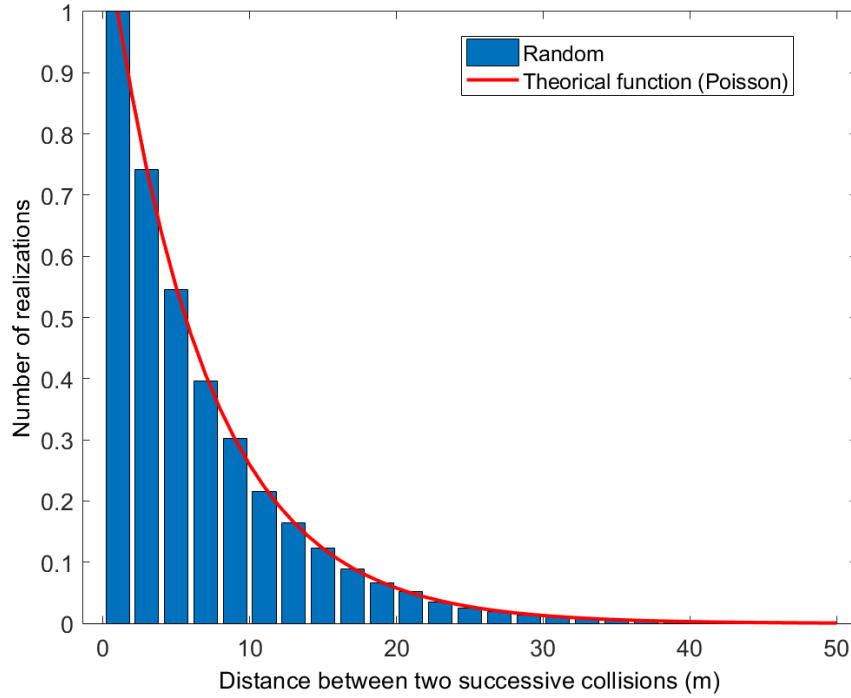


Fig. 11: Distribution of the \hat{R}_i paths obtained by the inverse cumulative distribution function method and comparison with the theoretical distribution

27.3 Wall modelling

27.3.1 Physical description

Absorption, dissipation and acoustic transmission

In contact with a wall (see figure *Illustration of the mechanisms of reflection, absorption, dissipation (internal loss) and acoustic transmission through a wall*), a sound wave will be partly reflected back into the domain for one part R , partly dissipated by transforming the acoustic energy into heat in the material for the other part β , the rest being transmitted through the material in the adjacent domain for the other part τ . The latter coefficient is defined as the transmission factor. If W_i is the power incident on a wall, then a part $W_r = R W_i$ will be reflected, a part $W_d = \beta W_i$ will be dissipated in the material, and a part $W_t = \tau W_i$ will be transmitted through the partition. By construction (there can be no creation of energy, nor more absorption than incident energy), the coefficients R , β and τ are between 0 and 1, so the energy balance of the wall is written:

$$R + \beta + \tau = 1.$$

It is usual to define the absorption coefficient α of the wall as the sum of the transmitted part τ and the dissipated part (internal loss) β , in the form $\alpha = \beta + \tau$, so that the above energy balance is written:

$$R = 1 - \alpha.$$

The absorption coefficient of a material can be measured using the standardised procedures *ISO 354* for the reverberation chamber method, as well as *ISO 10534-1* and *ISO 10534-2* for the impedance tube method. For the sound transmission coefficient, the reader may refer to the different parts of the standards for air transmission (ISO 140).

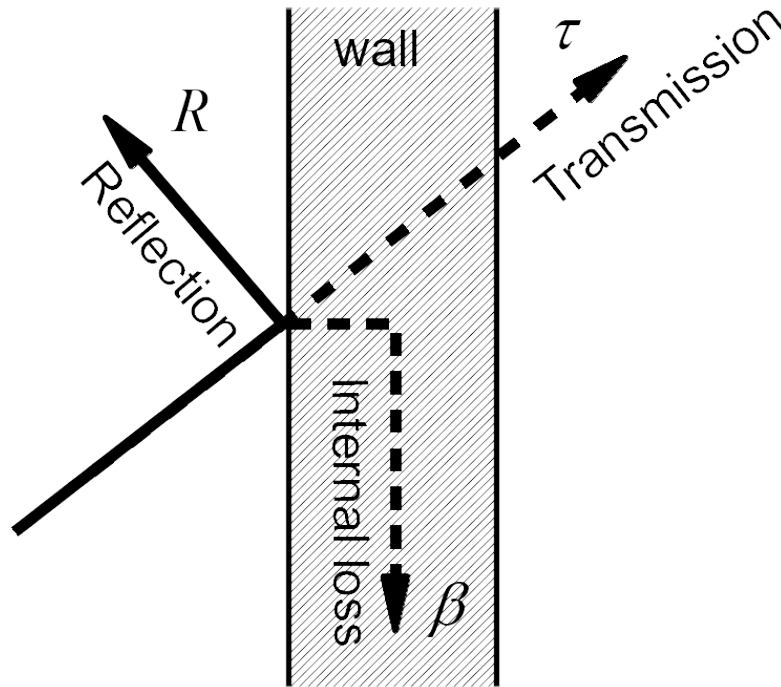


Fig. 12: Illustration of the mechanisms of reflection, absorption, dissipation (internal loss) and acoustic transmission through a wall

Acoustic diffusion

On the other hand, depending on the shape, size and distribution of the wall irregularities, the sound wave can be reflected simultaneously in the specular direction and in other directions. In room acoustics, it is common to consider that a fraction $1 - s$ of the sound energy will be reflected in the direction of specular reflection, while a fraction s of the energy will be reflected in the other directions of space, according to a law of reflection characterized by irregularities in the wall [EAS01]. In the latter case, we speak of *diffuse reflection*, where s is called scattering coefficient.

In room acoustics, numerous theoretical and experimental studies are currently underway to characterize or measure these laws of reflection [VM00][CDAntonio16]. However, the common practice is to use Lambert's law to describe a diffuse reflection. The value of the scattering coefficient s can be obtained by a standardized measurement procedure (ISO 17497-1).

Note: It is important to note that this scattering coefficient (ISO 17497-1), which defines the ratio between non-specular reflected energy on the total energy, is different from the diffusion coefficient δ (ISO 17497-2), which defines the ability of the surface to uniformly scatter in all direction. In some acoustic simulation software, the diffraction coefficient s may sometimes wrongly called the diffusion coefficient.

27.3.2 Acoustic reflection modelling

Random modelling

First, a sound particle colliding with a wall can either be absorbed by the wall (with a probability α), or reflected in a new direction of propagation (with a probability $R = 1 - \alpha$). In practice, the absorption/reflection choice is made by

randomly drawing a number u between 0 and 1, following a uniform distribution:

- If this number is less than $\alpha = 1 - R$ (at the considered point), the particle is absorbed and disappears from the propagation medium.
- If this number is greater than $\alpha = 1 - R$, the particle is reflected and continues to propagate in a new direction. In this last case, to determine the type of reflection (specular or diffuse), a new random draw v is performed between 0 and 1:
 - If this number is less than the value of $1 - s$ (*i.e.* $v < (1 - s)$) at the point considered, the particle is specularly reflected, in accordance with the well-known Snell-Descartes laws.
 - Otherwise (*i.e.* if $v > (1 - s)$), the reflection is diffuse. In the latter case, it is necessary to determine the direction of diffuse reflection.

Energetic modelling

When a particle collides with a wall, its energy ϵ is weighted by the reflection coefficient $R = 1 - \alpha$. The choice of specular or diffuse reflection, depending on the scattering coefficient, is identical to the ‘Random’ method: a random draw v is performed between 0 and 1. If this number is less than the value of $1 - s$, the particle is specularly reflected. Otherwise, the reflection is diffuse, in a direction to be determined. An entirely ‘Energetic’ treatment would be possible by duplicating the particle into two particles, the first being reflected in the specular direction and the second in the diffuse direction; however this last solution is not yet implemented in the **SPPS** code..

27.3.3 Modelling of the reflection laws

Formalism

Let us consider an incident sound particle on a wall, whose incident direction is defined by the spherical coordinates (θ, ϕ) (see figure *Elementary geometry of a reflection by a wall*). This particle has a probability of $P(\theta, \phi; \theta', \phi') \equiv P(\Omega, \Omega')$ being reflected in the elementary solid angle $d\Omega' = \sin \phi' d\phi' d\theta'$ [Joy74][Joy75][Joy78][LP03].

$P(\Omega, \Omega') d\Omega'$ actually represents the fraction of the incident sound intensity that is reflected in the solid angle $d\Omega'$. Let’s say $j(\theta, \phi)$ the incident flow of particles. The flow of reflected particles $j'(\theta', \phi')$ is expressed:

$$j'(\theta', \phi') \cos \phi' = \int P(\theta, \phi; \theta', \phi') j(\theta, \phi) \cos \phi d\Omega. \quad (27.10)$$

Defining the reflection law $R(\theta, \phi; \theta', \phi') \equiv R(\Omega, \Omega')$ on the following:

$$R(\theta, \phi; \theta', \phi') = \frac{P(\theta, \phi; \theta', \phi')}{\cos(\phi')},$$

the relation (??) can be written:

$$j'(\theta', \phi') = \int R(\theta, \phi; \theta', \phi') j(\theta, \phi) \cos \phi d\Omega.$$

It is important to emphasize the difference between the law of reflection R and the probability P . P refers to a probability of reflection of a sound particle by elementary solid angle, while R refers to the flow of energy reflected in a direction ϕ' , for an incident flow in the direction ϕ .

The first law of thermodynamics requires the conservation of the flow on the surface (in the absence of absorption). The law of reflection $R(\Omega, \Omega')$ must therefore check the following condition:

$$\int P(\Omega, \Omega') d\Omega' = \int R(\Omega, \Omega') \cos \phi' d\Omega' = 1,$$

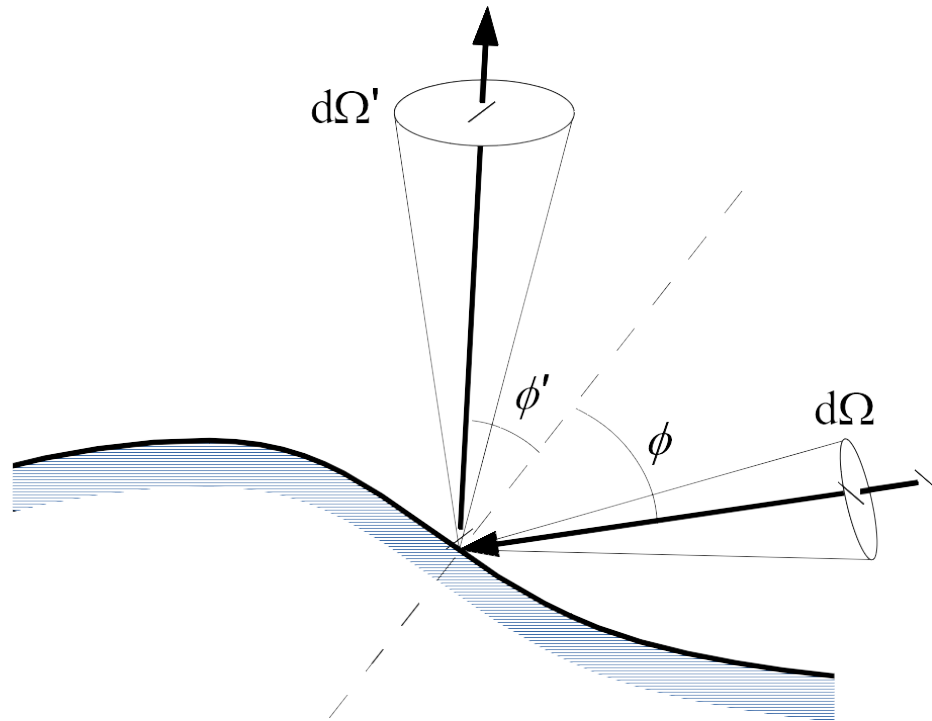


Fig. 13: Elementary geometry of a reflection by a wall

Variables ϕ and ϕ' refer respectively to the angles of incidence and reflection with respect to the normal at the wall. The angles θ and θ' in the tangent plane, directions of incidence and reflection, are not represented ($\phi, \phi' \in [0, \pi/2]$ and $\theta, \theta' \in [0, 2\pi]$).

or, in spherical coordinates,

$$\iint R(\theta, \phi; \theta', \phi') \cos \phi' \sin \phi' d\theta' d\phi' = 1.$$

The second law of thermodynamics imposes this time:

$$\int P(\Omega, \Omega') d\Omega = \int R(\Omega, \Omega') \cos \phi d\Omega = 1, \quad (27.11)$$

or, in spherical coordinates,

$$\iint R(\theta, \phi; \theta', \phi') \cos \phi \sin \phi d\phi d\theta = 1. \quad (27.12)$$

On the other hand, the principle of reciprocity requires that

$$R(\Omega, \Omega') = R(\Omega', \Omega). \quad (27.13)$$

Despite some recent studies, and apart from Lambert's law widely used to model diffuse reflections [Kut16], to our knowledge there are no other laws of reflection for real walls. In the **SPPS** code, however, we propose arbitrary modes of reflection, more or less real, presented in figure *Illustration of the reflection laws proposed in the SPPS code*.

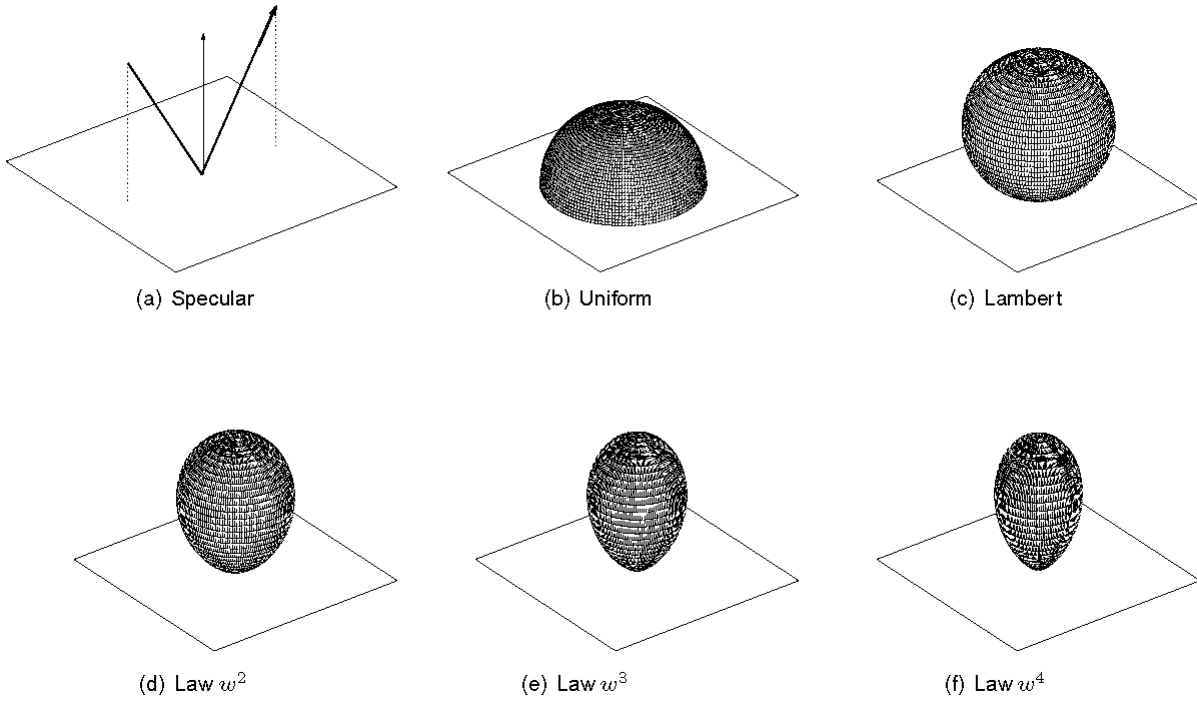


Fig. 14: Illustration of the reflection laws proposed in the **SPPS** code

Specular reflection

Physical description

The simplest mode of reflection is defined by specular reflection and can be written in three dimensions:

$$R(\theta, \phi; \theta', \phi') = 2\delta(\theta - \theta' \pm \pi) \delta(\sin^2 \phi - \sin^2 \phi'),$$

where the couples (θ, ϕ) and (θ', ϕ') refer respectively to the incident and reflected directions of the sound particles on the wall. Although the form of this expression is not conventional, this relationship verifies the laws of Snell-Descartes, as well as the conditions (??) to (??).

Modelling

From a numerical point of view, the simulation of this reflection law does not pose a major problem, since the angles of incidence of each particle on a wall are known. The modelling is identical for the ‘Random’ and ‘Energetic’ approaches.

Uniform reflection

Physical description

A uniform reflection law produces a distribution of reflection angles $P(\Omega')$ that is equiprobable. This should not be confused with Lambert’s law of reflection, for which uniformity is verified by the law of reflection. In the case of a uniform law, particles are reflected uniformly throughout the half space, regardless of the angle of incidence. Under these conditions, the reflection density after normalization is written:

$$P(\Omega')d\Omega' = P(\theta', \phi')d\Omega' = \left[\frac{1}{2\pi} d\theta' \right] [\sin \phi' d\phi'],$$

or

$$R(\theta', \phi') = \frac{1}{2\pi \cos \phi'},$$

with $\theta' \in [0, 2\pi]$ and $\phi' \in [0, \pi/2]$.

Modelling

From a numerical point of view, the determination of the angle of reflection is again obtained by the inverse cumulative distribution function method,

$$f(\hat{\phi}) = \int_0^{\hat{\phi}} \sin \phi' d\phi' = 1 - \cos \hat{\phi},$$

which gives

$$\hat{\phi} = \cos^{-1}(1 - u),$$

where u is a random number between 0 and 1.

Lambert reflection

Physical description

In optics, a perfectly diffuse surface is a surface that also appears bright regardless of the angle of observation, and regardless of the angle of incidence. In terms of acoustics, a perfectly diffuse surface will reflect the same energy in all directions regardless of the angle of incidence. From a mathematical point of view, this condition requires that

the law of reflection R be independent of the direction of reflection, and therefore of the direction of incidence (by reciprocity). After normalization, the law of reflection is written:

$$R(\theta, \phi; \theta', \phi') = \frac{1}{2\pi} \times 2.$$

The coefficient $1/2\pi$ is related to the normalization according to the angle θ' (uniform distribution between 0 and 2π). The second coefficient (factor 2) is related to the normalization according to the angle ϕ' . The probability of $P(\Omega')d\Omega'$ is therefore reduced to

$$\begin{aligned} P(\Omega') d\Omega' &= R \cos \phi' \sin \phi' d\theta' d\phi' \\ &= \left[\frac{1}{2\pi} d\theta' \right] [2 \cos \phi' \sin \phi' d\phi'], \end{aligned}$$

where the expression $\cos \phi'$ is related to Lambert's law. It is easy to check that R checks the conditions (??) to (??). It is important to note the difference between a random surface and an uniform surface. The first one is conditioned by an uniform random reflection law R , while the second is defined by a uniform probability P .

Modelling

From a numerical point of view, the random drawing of the reflection angles (in three dimensions) must be carried out in accordance with the distribution $P(\Omega')$, *i.e.* $P(\phi')$ in our case. Applying the inverse cumulative distribution function method, the probability $f(\hat{\phi})$ that a sound particle is reflected in an angle ϕ' between 0 and $\hat{\phi}$ is given by the following relationship:

$$f(\hat{\phi}) = 2 \int_0^{\hat{\phi}} \cos \phi' \sin \phi' d\phi' = \sin^2 \hat{\phi}.$$

Since this probability is between 0 and 1, the choice of angle $\hat{\phi}$ is made by randomly drawing a number $u \in [0.1]$, such that:

$$\hat{\phi} = \sin^{-1} \sqrt{u} = \cos^{-1} (1 - u)^{\frac{1}{2}}.$$

Warning: It is important to note that other authors [BMC90][Hod91][Lam96][XKaJ02] mention other relationships. The correct formulation depends on the angle convention. It is also possible that some of the relationships that are proposed may not be accurate.

Normal reflection w^n

Physical description

Let us consider a law of reflection independent of the incident direction and defined only by the angle of reflection ϕ' around the normal to the wall (θ' being uniform between 0 and 2π), of the form:

$$\begin{aligned} R(\Omega') &= R(\theta', \phi') \\ &= \frac{1}{2\pi} \times (n + 1) \cos^{n-1} \phi' \\ &= \frac{n + 1}{2\pi} w^{n-1}. \end{aligned}$$

where n is a positive integer. The quantity noted $w = \cos \phi'$ is none other than the projection of the direction of reflection on the normal at the wall. According to the notations in this document, the probability P will therefore be expressed:

$$\begin{aligned} P(\Omega') d\Omega' &= P(\phi') \sin \phi' d\theta' d\phi' \\ &= \left[\frac{1}{2\pi} d\theta' \right] \times [(n+1) \cos^n \phi' d\phi'] \\ &= \frac{n+1}{2\pi} w^n d\theta' d\phi' \end{aligned}$$

which justifies the name “law in w^n ”. This form of reflection is identical to the one introduced in (lepolles2003). It can be noted that this type of law is a generalized form of Lambert’s law ($n = 1$) and the uniform law ($n = 0$).

Modelling

From a numerical point of view, the random drawing of the angles of reflection is achieved by applying the inverse cumulative distribution function method. The probability $f(\hat{\Omega})$ (*i.e.* the probability $f(\hat{\phi})$ since the direction of reflection depends only on the angle with respect to normal) that a sound particle is reflected in an elementary solid angle between 0 and $\hat{\Omega}$ is then given by the relationship:

$$f(\hat{\phi}) = (n+1) \int_0^{\hat{\phi}} \cos^n \phi' \sin \phi' d\phi' = 1 - \cos^{n+1} \hat{\phi}.$$

Since this probability is between 0 and 1, the choice of angle $\hat{\phi}$ is made by randomly drawing a number $u \in [0.1]$, such that:

$$\hat{\phi} = \cos^{-1} (1 - u)^{\frac{1}{n+1}}.$$

27.3.4 Acoustic transmission modelling

Physical description

As mentioned above, the part of the power that is not reflected by the wall is either dissipated in the wall or transmitted. The acoustic transmission is then defined by the transmission factor τ , defined as the ratio of the transmitted power W_t by the wall to the incident power W_i . If there is no dissipation in the wall (*i.e.* everything is transmitted through the wall), then $\tau = \alpha$. Otherwise, $\tau < \alpha$. In practice, acoustic transmission is rather defined by the insertion loss (IL or R) of the wall, which is a function of the transmission factor through the following relationship:

$$R = 10 \log \left(\frac{W_i}{W_t} \right) = -10 \log \tau.$$

Random modelling

The modelling is similar to the one used for acoustic reflection. To determine if the sound particle is dissipated or transmitted by the partition, it is necessary to draw a number w between 0 and α :

- If this number is less than τ , the particle is transmitted and keeps its direction of propagation;
- Otherwise the particle is ‘dissipated’ and disappears from the propagation medium.

Energetic modelling

The energy modeling is performed by weighting the energy of the particle once transmitted by the partition, by the coefficient τ . In this case, there is no change of direction of propagation.

27.3.5 Verification of wall modelling

The figure *Verification of the procedure for modelling absorption (dissipation and transmission) and reflection (specular and diffuse) phenomena* illustrates the result of the reflection procedure (reflection=specular/diffuse, absorption=loss/transmission) in random mode, with the following acoustic parameters: scattering coefficient $s = 0.6$, absorption coefficient $\alpha = 0.8$, transmission coefficient $\tau = 10^{-R/10} = 0.1$ (attenuation index $R = 10$ dB). With 10000 realizations, the different phenomena are found (in terms of number of realizations) with the same proportions as the imposed acoustic parameters.

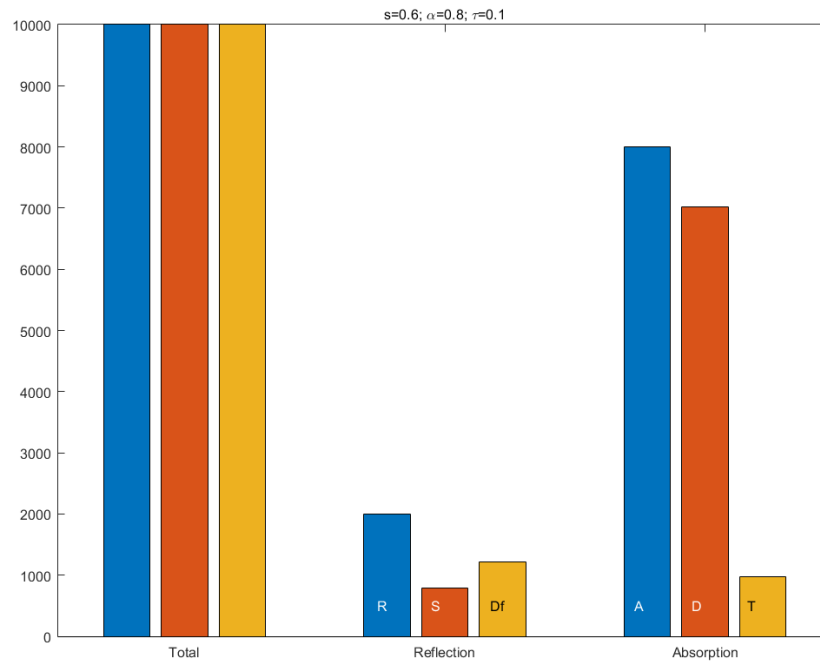


Fig. 15: Verification of the procedure for modelling absorption (dissipation and transmission) and reflection (specular and diffuse) phenomena

In the previous figure, only the random drawing process is modelled (note that this simulation was realized with a very simplification of the method, not the one implemented in the **SPPS** code, this last one being more performant).

- The first group 'Total' corresponds to the sum of the number of achievements (initial number of particles, reflection + absorption, specular + diffus + dissipation + transmission); these three quantities therefore have the exactly same value, which is the initial number of sound particles (*i.e.* 10000).
- The second group corresponds to the 'Reflection' process, divided into the total number of reflections (R, 2005 particles, *i.e.* an equivalent reflection coefficient of 0.2005, which is quite the same that the initial condition of the random process $R = 1 - \alpha = 0.2$), specular reflections (S, 788 particles) and diffusion (Df, 1217 particles, *i.e.* an equivalent scattering coefficient of $1217/2005=0.61$, which is quite the same that the initial condition of the random process $s = 0.6$)).
- The third group corresponds to the 'Absorption' process, divided into the total number of absorption (A, 7995 particles, *i.e.* an equivalent absorption coefficient of 0.7995, which is quite the same that the initial condition of the random process $\alpha = 0.8$)), dissipation (D, 7014 particles) and transmission (T, 981 particles, *i.e.* an equivalent transmission coefficient of 0.0981, which is quite the same that the initial condition of the random process $\tau = 0.1$)).

27.4 Calculation of sound levels at observation points

In the **SPPS** code, two types of receivers are considered:

- ‘Volume’ receivers model the so-called “classical” point receivers. Since the notion of point receiver is not applicable in the **SPPS** code, since in theory the probability that a sound particle will pass through a point receiver is zero, it is necessary to give a volume to the receiver point, to count the number of particles that have passed through it, and thus deduce the energy density at the observation point. For a point receiver, the **SPPS** code returns the sound pressure level, at each time step of the calculation and for each frequency band considered;
- ‘Surface’ receivers are surface elements (like a face element of the 3D scene) on which incident sound intensities are calculated, which then makes it possible to construct acoustic maps. For a surface receiver, the code **SPPS** returns the sound intensity, at each time step of the calculation and for each frequency band considered.

In addition, the calculation code also determines the overall sound pressure level in the 3D model, summing the contributions of each particle, at each time step and for each frequency band.

27.4.1 Calculation of the sound pressure level at a volume receiver

The amount of **energy** $E_{\text{rec}}^j(n)$ (in J or W.s) in the frequency band j received at a volume receiver, at the time step n (*i.e.* to time $n\Delta t$) is equal to the sum of the energies ε_i^j brought by each particle i in the frequency band j , crossing the receiver volume during the time step n (figure[principle_receiver_volumique]):

$$E_{\text{rec}}^j(n) = \sum_i^{N_0} \varepsilon_i^j = \sum_i^{N_0} \frac{W}{N} \epsilon_i^j \times \Delta t_i,$$

where N_0 is the total number of particles passing through the receiver volume and Δt_i is the presence time of the particle i in the receiver volume ($\Delta t_i < \Delta t$), and ϵ_i^j the weighting coefficient (between 0 and 1) associated with the particle i in the frequency band j . If the calculation mode is “random”, the coefficient ϵ_i^j is constant and equal to the unit (1). If the calculation mode is “energetic”, the coefficient ϵ_i^j translates the cumulative loss of energy in the frequency band j of the particle i throughout its path due to the physical phenomena encountered (absorption by walls and obstructions, atmospheric absorption, sound transmission...). The presence time Δt_i can also be expressed as a function of the length of the path of the particle i in the receiving volume, *i.e.* ℓ_i , such as $\Delta t_i = \ell_i/c$, c being the velocity of the particle at the observation point. Considering a “volume” receiver is defined by a spherical volume of radius r_{rec} (and volume V_{rec}), the **energy density** $w_{\text{rec}}^j(n)$ (J/m³) in the volume receiver, for the frequency band j , is given by:

$$w_{\text{rec}}^j(n) = \frac{E_{\text{rec}}(n)}{V_{\text{rec}}} = \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_0} \epsilon_i^j \frac{\ell_i}{c}. \quad (27.14)$$

Note: The definition of the receiver volume is a crucial element for the quality and representativeness of the results. It must be large enough to count sound particles as they propagate, but not too large to make the energy density calculated at the observation point representative.

The **sound intensity** $I_{\text{rec}}^j(n)$ (in W/m²) at the receiving point is given by:

$$I_{\text{rec}}^j(n) = c \times w_{\text{rec}}^j(n) = \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_0} \epsilon_i^j \ell_i.$$

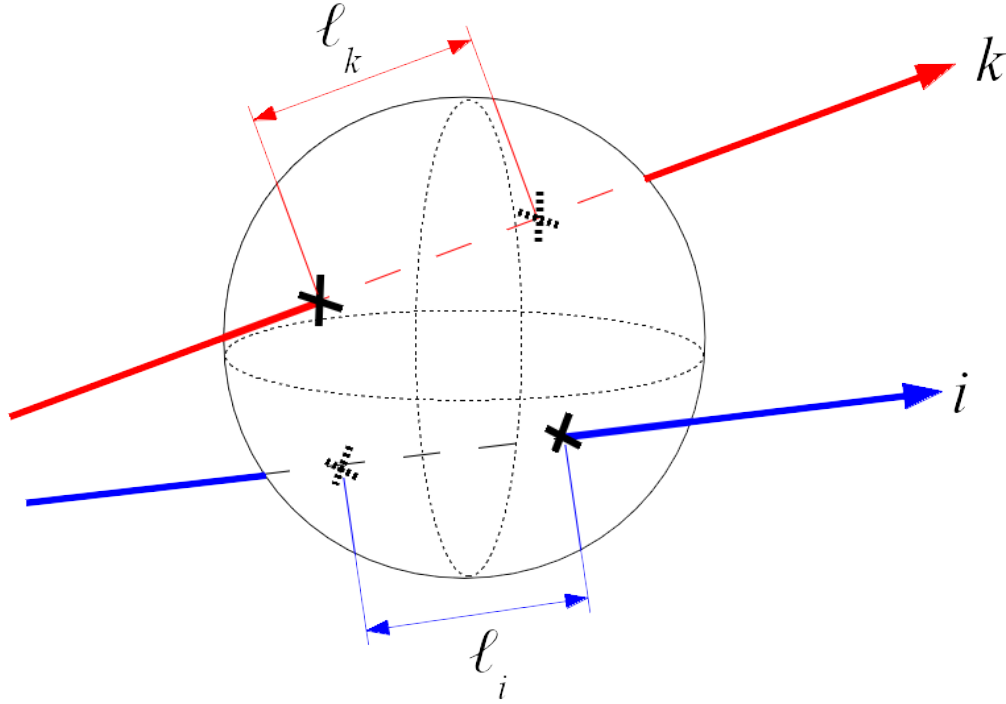


Fig. 16: Principle of the calculation of the sound pressure level for a volume receiver

The energy density in the volume receiver is calculated by summing the energy contributions of each particle passing through the receiver. The contribution of the particle i is calculated from the path ℓ_i of the particle in the volume receiver (??).

The **sound intensity level** $L_{I,rec}^j(n)$ and the **sound pressure level** $SPL_{rec}(n)$ (in dB) can then be derived from the sound intensity by the following relationships:

$$L_{I,rec}^j(n) = 10 \log \left(\frac{P_{2,rec}^j(n)}{\rho_0 c I_0} \right),$$

and

$$SPL_{rec}^j(n) = 10 \log \left(\frac{P_{2,rec}^j(n)}{p_0^2} \right),$$

where $I_0 = 10^{-12} \text{ W/m}^2$ and $p_0 = 20 \times 10^{-6} \text{ Pa}$ denote the reference sound intensity and sound pressure. Each of the previous quantities (energy, energy density, intensity and sound levels) are calculated for each frequency band.

27.4.2 Intensity vector at a volume receiver

The vector **sound intensity** $\mathbf{I}_{rec}^j(n)$ (in W/m^2 or $\text{J/m}^2 \cdot \text{s}$) at the receiving point, for the frequency band j , is defined as the sum of the energy densities carried by the velocity vector \mathbf{v}_i of the particles (standard c_i) passing through the receiving volume:

27.4.3 Calculation of the lateral sound pressure level at a volume receiver

For the calculation of certain acoustic parameters, such as those based on lateral energy (LF with a weighting in $|\cos \theta|$ and LFC with a weighting in $\cos^2 \theta$), it is necessary to consider a weighting of the sound intensity as a function of the

angle θ between the observation direction of the point receiver (in principle oriented towards the sound source) and the incident direction of the particles at the receiver. As a result, the **SPPS** code also returns the following two quantities, homogeneous to the square of the sound pressure (*i.e.* in Pa^2):

$$\begin{aligned} P2_{\text{rec}, \cos \theta}^j(n) &= \rho_0 c^2 \times w_{\text{rec}, \cos \theta}^j(n) \\ &= \rho_0 c \frac{W}{N} \frac{1}{V_{\text{rec}}} \times \sum_i^{N_0} \epsilon_i^j \ell_i |\cos \theta_i|, \end{aligned}$$

and

$$\begin{aligned} P2_{\text{rec}, \cos^2 \theta}^j(n) &= \rho_0 c^2 \times w_{\text{rec}, \cos^2 \theta}^j(n) \\ &= \rho_0 c \frac{W}{N} \frac{1}{V_{\text{rec}}} \times \sum_i^{N_0} \epsilon_i^j \ell_i \cos^2 \theta_i. \end{aligned}$$

where θ_i is the corresponding angle for the particle i .

27.4.4 Calculation of the sound intensity level on a surface receiver

The **power** W_{surf}^j (in W) received by a surface element of size ΔS of normal \mathbf{n} is equal to the sum of the energy provided by each particle i in the frequency band j per time unit Δt , at the time step n (*i.e.* at time $n\Delta t$), either:

$$W_{\text{surf}}^j(n) = \sum_i^{N_0} \frac{\epsilon_i^j}{\Delta t} \frac{\mathbf{v}_i}{c} \cdot \mathbf{n} = \frac{W}{N} \sum_i^{N_0} \epsilon_i \cos \theta_i,$$

where \mathbf{v}_i refers to the velocity vector (of norm c) of the particle i , θ_i the angle between normal \mathbf{n} of the surface and the direction of the particle, and where N_0 is the total number of sound particles colliding with the surface element ΔS .

The **sound intensity** $I_{\text{surf}}^j(n)$ (in W/m^2) received by the surface element ΔS at the time step n is equal to the power received divided by the surface, either:

$$I_{\text{surf}}^j(n) = \frac{W}{N} \frac{1}{\Delta S} \sum_i^{N_0} \epsilon_i^j \cos \theta_i.$$

The **sound level** $L_{\text{surf}}^j(n)$ (in dB) can then be calculated by the following relationship:

$$L_{\text{surf}}^j(n) = 10 \log \frac{I_{\text{surf}}^j(n)}{I_0},$$

where $I_0 = 10^{-12} \text{ W/m}^2$ is the reference intensity.

27.4.5 Calculation of the sound pressure level on a surface receiver

Equivalent to the calculation of the sound intensity level on a surface receiver, the quadratic sound pressure $P2_{\text{surf}}^j(n)$ (in Pa) received by the surface element ΔS at the time step n is equal to the product of the sound intensity (without weighting the angle of incidence on the surface) by the characteristic impedance of the air ρ_0 that is:

$$P2_{\text{surf}}^j(n) = \rho_0 \rho_0 c \frac{W}{N} \frac{1}{\Delta S} \sum_i^{N_0} \epsilon_i^j.$$

The **sound pressure level** $L_{\text{SPL}, \text{surf}}^j(n)$ (in dB) can then be calculated by the following relationship:

$$L_{\text{SPL}, \text{surf}}^j(n) = 10 \log \frac{P2_{\text{surf}}^j(n)}{p_0^2},$$

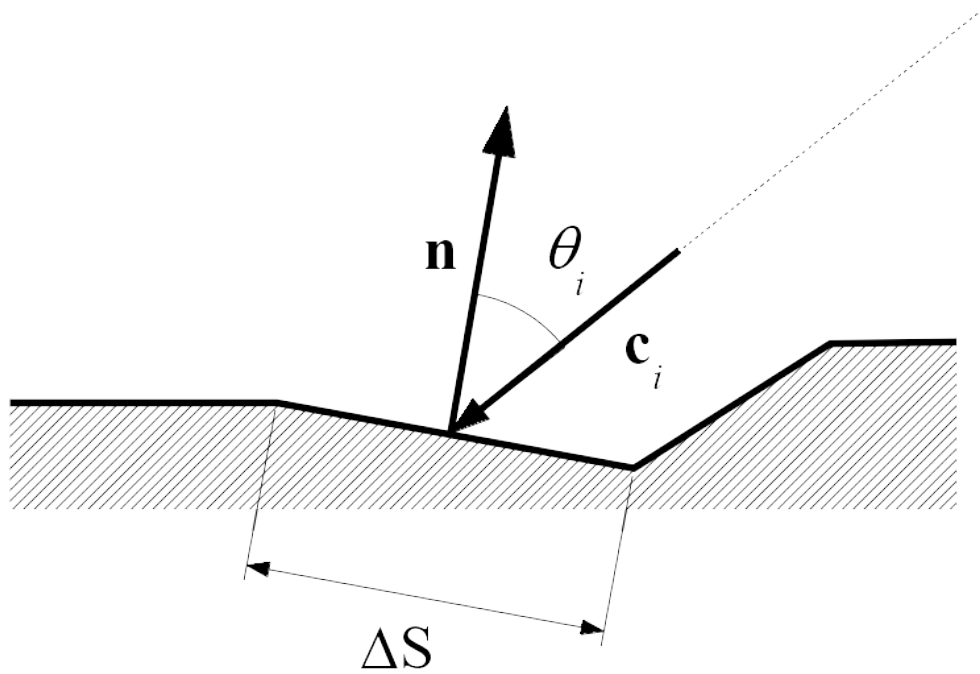


Fig. 17: Principle of the calculation of the sound intensity level for a surface receiver

where $p_0 = 20 \times 10^{-6}$ Pa is the reference sound pressure.

Warning: Note that the calculation of a sound pressure level for a surface is quite ambiguous. It should not be comparable to grid of punctual receivers on a plane.

27.4.6 Calculation of the overall sound pressure level in the model

The overall sound intensity $I_{\text{global}}^j(n)$ in the model, for the frequency band j , is calculated by summing the intensities carried by all the sound particles present N_n in the model at the time step n (i.e. at time $n\delta t$):

$$I_{\text{global}}^j(n) = \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_n} \epsilon_i^j \ell_i.$$

The **sound pressure level** $SPL_{\text{global}}^j(n)$ (in dB) can then be calculated by the following relationship:

$$SPL_{\text{global}}^j(n) = 10 \log \left(\frac{P_{\text{global}}^j(n)}{p_0^2} \right),$$

where $p_0 = 20 \times 10^{-6}$ Pa is the reference sound pressure.

PRESENTATION

TCR is a numerical code based on the Classical Theory of Reverberation.

- Visit the [official I-Simpa website](#) for more information about SPPS code.
- See the main [characteristics of the code](#) that is embedded in I-Simpa.

The simulation code TCR (from French “Théorie Classique de la Réverbération”) is a numerical application of the Classical Theory of Reverberation as proposed by Sabine. It allows to obtain an evaluation of the diffuse sound field in a single room on the basis of Sabine’s relations for the reverberation time and for the sound level of the reverberated field.

More information can be obtained in the following reference book: Kuttruff, Heinrich. Room Acoustics, CRC Press (Sixth Edition, 2016).

USING TCR WITHIN I-SIMPA

TCR is already embedded in I-Simpa. Follow the next instructions to run a calculation with the TCR code in I-Simpa.

Double left click on the TCR calculation code in I-Simpa to display all calculation parameters:

- *Meshing*, for the meshing configuration
- *Calculation parameters* for the calculation parameters

Right click on the selected calculation code to display all possible actions:

- Run calculation
- Job list

29.1 Meshing

Some calculation codes may require a meshing of the domain (SPPS for example). I-Simpa used the [TetGen meshing code](#) (A Quality Tetrahedral Mesh Generator and a 3D Delaunay Triangulator). See the documentation for more information.

Within I-Simpa, TetGen can be parametrized:

- **‘debugmode’** Check/uncheck for activate the debug mode of TetGen;

Warning: With the debug mode, the meshing is not realized.

- **‘minratio’** Defines the ratio between the radius of the sphere that would contain a mesh and the lenght of the mesh. This parameter could be useful in order to avoid long meshes.
- **‘userdefineparams’** Defines new parameters by replacing default parameters.
- **‘appendparams’** Add parameters to the dafault parameters.
- **‘preprocess’** Check/uncheck for trying to repair the 3D scene if possible.
- **‘maxvol’** Maximun accepted value of a mesh volume (m3). The value can be modified only if the ‘ismaxvol’ parameter is check.
- **‘ismaxvol’** Check/uncheck for using a given maximum value of the mesh volume ‘maxvol’.
- **‘constraintrecepteurss’** Maximun accepted value of the mesh surface (m2) for a surface receiver. The value can be modified only if the ‘isareaconstraint’ parameter is checked.
- **‘isareaconstraint’**. Check/uncheck for using a given value of the mesh surface ‘constraintrecepteurss’.

29.2 Calculation parameters

- **‘Active calculation of atmospheric absorption’** Enables calculation of atmospheric absorption.
- **‘Export surface receivers for each frequency band’** Check to export the surface receiver results, for each frequency band. If uncheck, only the global value is exported.

STANDARDS

30.1 Acoustics

ISO 9613-1 ISO 9613-1:1993, Acoustics – Attenuation of sound during propagation outdoors – Part 1: Calculation of the absorption of sound by the atmosphere http://www.iso.org/iso/catalogue_detail.htm?csnumber=17426

ISO 3382-1 ISO 3382-1:2009, Acoustics – Measurement of room acoustic parameters – Part 1: Performance spaces http://www.iso.org/iso/catalogue_detail.htm?csnumber=40979

NF S31-133 NF S31-133:2011, Acoustics – Outdoor noise - Calculation of sound levels <https://www.boutique.afnor.org/norme/nf-s31-133/acoustique-bruit-dans-l-environnement-calcul-de-niveaux-sonores/article/678966/fa169343>

ISO 354 ISO 354:2003, Acoustics – Measurement of sound absorption in a reverberation room <https://www.iso.org/standard/34545.html>

ISO 10534-1 ISO 10534-1:1996, Acoustics – Determination of sound absorption coefficient and impedance in impedance tubes – Part 1: Method using standing wave ratio <https://www.iso.org/standard/18603.html>

ISO 10534-2 ISO 10534-2:1998, Acoustics – Determination of sound absorption coefficient and impedance in impedance tubes – Part 2: Transfer-function method <https://www.iso.org/standard/22851.html>

ISO 17497-1 ISO 17497-1:2004, Acoustics – Sound-scattering properties of surfaces – Part 1: Measurement of the random-incidence scattering coefficient in a reverberation room <https://www.iso.org/standard/31397.html>

ISO 17497-2 ISO 17497-2:2012, Acoustics – Sound-scattering properties of surfaces – Part 2: Measurement of the directional diffusion coefficient in a free field <https://www.iso.org/standard/55293.html>

ISO 266 ISO 266:1997, Acoustics – Preferred frequencies <https://www.iso.org/standard/1350.html>

EN ISO 3382-1 ISO 3382-1:2009, Acoustics – Measurement of room acoustic parameters – Part 1: Performance spaces <https://www.iso.org/standard/40979.html>

30.2 Other

ISO 639-1 NF ISO 639-1:2003 – Codes for the representation of names of languages - Part 1 : Alpha-2 code <https://www.boutique.afnor.org/standard/nf-iso-639-1/codes-for-the-representation-of-names-of-languages-part-1-alpha-2-code/article/742951/fa104093>

ISO 639-2 NF ISO 639-2:1999 – Codes for the representation of names of languages - Part : alpha-3 code <https://www.boutique.afnor.org/standard/nf-iso-639-2/codes-for-the-representation-of-names-of-languages-part-alpha-3-code/article/727490/fa043864>

- ISO 639-3** NF ISO 639-3:2007 – Codes for the representation of names of languages - Part 3 : alpha-3 code for comprehensive coverage of languages <https://www.boutique.afnor.org/standard/nf-iso-639-3/codes-for-the-representation-of-names-of-languages-part-3-alpha-3-code-for-comprehensive-coverage-of-languages/article/752747/fa138697>
- ISO 639-4** NF ISO 639-4:2010 – Codes for the representation of names of languages - Part 4 : general principles of coding of the representation of names of languages and related entities, and application guidelines <https://www.boutique.afnor.org/standard/nf-iso-639-4/codes-for-the-representation-of-names-of-languages-part-4-general-principles-of-coding-of-the-representation-of-names/article/724011/fa138698>
- ISO 639-5** NF ISO 639-5:2008 – Codes for the representation of names of languages - Part 5 : Alpha-3 code for language families and groups <https://www.boutique.afnor.org/standard/nf-iso-639-5/codes-for-the-representation-of-names-of-languages-part-5-alpha-3-code-for-language-families-and-groups/article/645452/fa138699>
- ISO 639-5** NF ISO 639-6:2010 – Codes for the representation of names of languages - Part 6 : alpha-4 code for comprehensive coverage of language variants <https://www.boutique.afnor.org/standard/nf-iso-639-6/codes-for-the-representation-of-names-of-languages-part-6-alpha-4-code-for-comprehensive-coverage-of-language-variants/article/699300/fa148722>

GLOSSARY

absorption coefficient Absorption coefficient of an object or a material. The value of the absorption coefficient is a real number between 0. (no absorption, *i.e.* perfect reflection) and 1 (total absorption, *i.e.* no reflection).

scattering coefficient The scattering coefficient is used in room acoustics software to model the surface diffusivity, such as room wall. When this coefficient is set to 1, it means that surfaces are 100% diffuse (*i.e.* generate diffuse reflection only), while, when this coefficient is set to 0, sound reflection are perfectly specular.

mean free path Mean distance between two collisions (unit: meter). The mean free path is at the basis of the Classical Theory of Reverberation, and allow to estimate the reverberant (diffuse) field and several room acoustics indicators. It statistically describes the distance between two successive wall reflections in a room.

It can also be used to model the acoustic diffusion in a volume that contains many obstacles (fittings). In this case, it statistically describes the distance between two successive collisions with the fittings.

REFERENCES

INDICES AND TABLES

- `genindex`

BIBLIOGRAPHY

- [Aul85] N. Auletta. Contribution à L'étude de la Propagation Du Son Dans Les Locaux Industriels Encombrés – 1ère Partie (étude Théorique). *Revue d'Acoustique*, 75:477–487, 1985.
- [Aul86] N. Auletta. Contribution à L'étude de la Propagation Du Son Dans Des Locaux Industriels Encombrés – 2ème Partie. *Revue d'Acoustique*, 76:5–18, 1986.
- [BSPE84] H. E. Bass, L. C. Sutherland, J. Piercy, and L. Evans. Absorption of sound by the atmosphere. In *Physical acoustics: Principles and methods*, volume 17, pages 145–232. Academic Press, Inc., 1984. URL: <http://adsabs.harvard.edu/abs/1984papm...17..145B>.
- [BMC90] P.J. Burns, J.D. Maltby, and M.A. Christon. Large-scale surface to surface transport for photons and electrons via monte carlo. *Computing Systems in Engineering*, 1(1):75 – 99, 1990. URL: <http://www.sciencedirect.com/science/article/pii/095605219090049Q>, doi:[https://doi.org/10.1016/0956-0521\(90\)90049-Q](https://doi.org/10.1016/0956-0521(90)90049-Q).
- [CDAntonio16] Trevor Cox and Peter D'Antonio. *Acoustic Absorbers and Diffusers : Theory, Design and Application*. CRC Press, November 2016. ISBN 978-1-315-36921-1. URL: <https://www.taylorfrancis.com/books/9781315369211>, doi:10.1201/9781315369211.
- [EAS01] J. J. Embrechts, D. Archambeau, and G. B. Stan. Determination of the scattering coefficient of random rough diffusing surfaces for room acoustics applications. *Acta Acustica united with Acustica*, 87(4):482–494, 2001. URL: <https://www.ingentaconnect.com/content/dav/aaua/2001/00000087/00000004/art00007>.
- [Hod91] Murray Hodgson. Evidence of diffuse surface reflections in rooms. *The Journal of the Acoustical Society of America*, 89(2):765–771, 1991. URL: <https://doi.org/10.1121/1.1894636>, arXiv:<https://doi.org/10.1121/1.1894636>, doi:10.1121/1.1894636.
- [Joy74] W. B. Joyce. Classical-particle description of photons and phonons. *Phys. Rev. D*, 9:3234–3256, Jun 1974. URL: <https://link.aps.org/doi/10.1103/PhysRevD.9.3234>, doi:10.1103/PhysRevD.9.3234.
- [Joy75] W. B. Joyce. Sabine's reverberation time and ergodic auditoriums. *The Journal of the Acoustical Society of America*, 58(3):643–655, 1975. URL: <https://doi.org/10.1121/1.380711>, arXiv:<https://doi.org/10.1121/1.380711>, doi:10.1121/1.380711.
- [Joy78] W. B. Joyce. Exact effect of surface roughness on the reverberation time of a uniformly absorbing spherical enclosure. *The Journal of the Acoustical Society of America*, 64(5):1429–1436, 1978. URL: <https://doi.org/10.1121/1.382120>, arXiv:<https://doi.org/10.1121/1.382120>, doi:10.1121/1.382120.
- [Kut16] Heinrich Kuttruff. *Room Acoustics*. CRC Press, October 2016. ISBN 978-1-315-37215-0. URL: <https://www.taylorfrancis.com/books/9781315372150>, doi:10.1201/9781315372150.
- [Kut81] K. Heinrich Kuttruff. Sound decay in reverberation chambers with diffusing elements. *The Journal of the Acoustical Society of America*, 69(6):1716–1723, 1981. URL: <https://doi.org/10.1121/1.385951>, arXiv:<https://doi.org/10.1121/1.385951>, doi:10.1121/1.385951.

- [Lam96] Y. W. Lam. A comparison of three diffuse reflection modeling methods used in room acoustics computer models. *The Journal of the Acoustical Society of America*, 100(4):2181–2192, 1996. URL: <https://doi.org/10.1121/1.417927>, arXiv:<https://doi.org/10.1121/1.417927>, doi:10.1121/1.417927.
- [LP03] Thierry Le Pollès. *Modélisation des champs diffus en acoustique architecturale par la théorie des transports : application au milieu urbain*. PhD thesis, Université du Maine, France, 2003. URL: <http://www.theses.fr/2003LEMA1016>.
- [Lin82] E. A. Lindqvist. Sound attenuation in large factory spaces. *Acta Acustica united with Acustica*, 50(5):313–328, 1982. URL: <https://www.ingentaconnect.com/content/dav/aaua/1982/00000050/00000005/art00004>.
- [OB89] A. M. Ondet and J. L. Barbry. Modeling of sound propagation in fitted workshops using ray tracing. *The Journal of the Acoustical Society of America*, 85(2):787–796, 1989. URL: <https://doi.org/10.1121/1.397551>, arXiv:<https://doi.org/10.1121/1.397551>, doi:10.1121/1.397551.
- [Sal01] E. M. Salomons. *Computational Atmospheric Acoustics*. Springer Netherlands, 2001. ISBN 978-0-7923-7161-8. URL: <https://www.springer.com/gp/book/9780792371618>.
- [Sch65] M. R. Schroeder. New method of measuring reverberation time. *The Journal of the Acoustical Society of America*, 37(3):409–412, 1965. URL: <https://doi.org/10.1121/1.1909343>, arXiv:<https://doi.org/10.1121/1.1909343>, doi:10.1121/1.1909343.
- [VM00] Michael Vorländer and Eckard Mommertz. Definition and measurement of random-incidence scattering coefficients. *Applied Acoustics*, 60(2):187 – 199, 2000. URL: <http://www.sciencedirect.com/science/article/pii/S0003682X99000560>, doi:[https://doi.org/10.1016/S0003-682X\(99\)00056-0](https://doi.org/10.1016/S0003-682X(99)00056-0).
- [XKaJ02] Zeng Xiangyang, Chen Ke-an, and Sun Jincai. Development of a hybrid computer model for simulating the complicated virtual sound field in enclosures. *Applied Acoustics*, 63(5):481 – 491, 2002. URL: <http://www.sciencedirect.com/science/article/pii/S0003682X01000548>, doi:[https://doi.org/10.1016/S0003-682X\(01\)00054-8](https://doi.org/10.1016/S0003-682X(01)00054-8).

INDEX

A

absorption coefficient, [139](#)

E

EN ISO 3382-1, [137](#)

I

ISO 10534-1, [137](#)

ISO 10534-2, [137](#)

ISO 17497-1, [137](#)

ISO 17497-2, [137](#)

ISO 266, [137](#)

ISO 3382-1, [137](#)

ISO 354, [137](#)

ISO 9613-1, [137](#)

M

mean free path, [139](#)

N

NF S31-133, [137](#)

S

scattering coefficient, [139](#)