GUIRAZR

A Graphical Interface of the RAZR MATLAB Toolbox

Oldenburg, Dept. of Medical Physics and Acoustics - Signal Processing
2017
Sébastien Lagarde

Introduction:

This program provide a demonstration of the RAZR MATLAB toolbox currently developed by Torben Wendt.

It's provide a graphical user interface which allow to control several parameters as reverberation, materials ,position of the sources, receivers, hrtf data..
It's possible to run the acoustic simulation in real time within specific conditions. See the section computation to get more information about these conditions.

Contacts:

Sébastien Lagarde: sebastien.lagarde@phelma.grenoble-inp.fr

Kai Siedenburg: kai.siedenburg@uni-oldenburg.de

Warning:

Many functionalities are still under work or may change. Feel free to improve it, or add some remarks on it. Contact me if you have report/correct some bug. See the RAZR documentation for help on RAZR behavior.

Get started:

- First download the following tools: razr, cipic database, portaudio, playrec, and install them. Troubleshooting with playrec install ? See the specific note.
- The main graphical window is the place you get the control to your room size, sources, receiver position and direction.
  - The black rectangular box is representing in 2D your acoustic room. You can drag it, resize it as well by a right click on it.
  - The black point is the receiver you can move it and set its direction easily by a right click on it and turning the direction line.
  - To add some sound sources click inside the room. A new windows will invite you to select one or several track.
  - For each track you will add, a box will ask you which channel you want to input into your room. The position of the sources are automatically set in a shape ellipse (depending of the click position in the room) around the position you clicked. As for the receiver you can move the sources by a click on it.

- You are now ready to start the simulation. Press play!
- During the simulation you can change all the parameters it will automatically update in the output audio result if you are in real time modes, manually if you are in sequential by pressing the load button

Change the room property and options:

There are several menu on the right to change the acoustic property of the room.

- Roomsize: Specify the size of the room by keyboard input instead of a mouse resize. The window scale change automatically if you change these parameters.
- Receiver direction: You can change the direction of the receiver in these box instead of using the direction line.
- Reverb: You can specify the shape of the t60 frequency dependences of your room.
- Materials: You can choose the materials of your room by checking the corresponding option and click on the menu
- HRTF: You can choose different Head Relative Transfer Function. Only supported for the CIPIC database for now.
- Parameters: Menu to specify others configuration, computation method:
    - Advanced: The options specify by GET_DEFAULT_OPTIONS.m will be in this new window.
    - Set manually positions (useful to set the elevation of the sources/receiver!!)
    - Set the output audio driver
    - Set the computation technics : sequential /parallel (soonly available) and playing modes: real time or full song computation
    - Load/Save a room configuration (see get_room_L in razr) -Load isn't working for now-
    - Automatic Rescale graphical options which move the instrument proportionally to the room rescale, movement. Set to true by default.

About computation:

- Global behavior:
    - For each instrument in your room a room impulse response is computed. Until there is no room parameters change, sources /receiver position move the room impulse response previously is kept. When a change occurs on a sources only the room impulse response related to this sources is computed again. If you change the receiver position or a room parameters, all the room impulse response have to be compute again.
    - The corresponding room impulse response is convolved with the input instrument, mixed with the others output and add in the audio output buffer.
- Modes:
    - Sequential mean for each instrument the room impulse response is computed sequentially.
    - Parallels: the idea is to compute in parallel the room impulse response. You can obtain a significant speed up in the program if you're choosing these options. Test on apply_rir as shown some downgrading of the performance in running on parallel. So this feature is not implemented for apply_rir. It's probably because matlab already compute convolution in parallel on your processor cores -if possible.
- Real Time & Latency:
    - In real time mode(Default playing mode) a timer periodically check the last changes in the GUI, and compute the audio result for the next audio frame.

  - In non-real-time mode for all the instrument the full length of the sounds is compute. You can still upload a room change on the resulting sound by pressing load button.
  - The latency mainly depends of your current RT60 and the number of instrument, the output audio may produce some glitch or stop when you're changing some room parameters in real time modes. It's depend of your hardware configuration. In non-real-time mode you may wait for long time to have the new configuration loaded. The dialog on the down right corner inform you about the current processing. By going into advanced parameters window you can modify the number of reflection or set the max length of the rir. It can help to speed up the process but you will lose the qualities.

Further Improvement:

Here is a list of some idea that could be integrated:

- Audio player graphical object.
- The possibility to integrate others HRTF database than only CIPIC.
- Include a 3D representation of the scene. See folder \graph and \test3d to get a basic idea how to implement it.
- The possibility to specify the room parameters by command line while using interface.
- Coupled room.
- Add an option to set manually the latency.
- Possibility to delete the instrument of the scene.
- View the room impulse/frequency response for each instrument.
- Possibility to use the absorption coefficient, temperature in parameter window.
- Record Output in realtime mode just record the parts of the song which has been already computed. Have a real time recording could be interesting.
- Handle others sampling rate than 44100 Hz, and have the possibility to use song with different sampling rate.
- Build efficient programming tool to associate quickly new razr parameters to graphic objects.
- Build a custom imroi class to handle all the draggable/resizable object with specific behavior.

Bugs:

As noted above all functionalities are not fully supported. The main issue to solve are presented in the section:

- Parameters some razr option may crash the simulation (e.g: verbose)
- Load room function
- Parameters window make crash the simulation while real time mode is running, to solve this. Don't use gcf (get current figure) to obtain the handle of the figure. Store all the data in 'UserData' matlab global variable.
- Song with different sampling rate may produce weird output or crash razr…
- If you have a problem with the sampling rate check out the selected audio driver, use the a driver create for input audio as input may produce a higher sample rated output. You can also use playrec('reset') and playrec('init',fs,driverID,-1) to reinitialize the driver choose manually

Playrec installation troubleshooting:

If you have some trouble to install playrec on windows you should have a look to the following points:

- As specified in playrec documentation to build the playrec mex file you must before have Windows SDK installed. If you have some trouble to install Windows SDK this patch might be useful: Windows SDK 7.1 patch, have a look also on Matlab Windows SDK topic. To ensure you have a Matlab C++ Compiler type in matlab command: *mex -setup*. Also ASIO and Directsound are not needed for our application so you can uncheck the playrec setup .
- Another solution is to use directly these pre-compiled mex files (built for my configuration) without installing playrec yourself. Don't forget to add them to matlab path.