UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE - DEI

MASTER'S THESIS IN CONTROL SYSTEMS ENGINEERING

AUTOMOTIVE GRANULAR SYNTHESIZER (AGS): DESIGN AND EVALUATION OF A SOUND PROCESSING ALGORITHM FOR ELECTRIC VEHICLES

SUPERVISOR
PROF. SERGIO CANAZZA TARGON
UNIVERSITÀ DEGLI STUDI DI PADOVA

MASTER CANDIDATE
EDOARDO DI PIETRANTONIO

CO-SUPERVISORS
DOTT. ALESSANDRO FIORDELMONDO
UNIVERSITY OF UDINE

DOTT. ANNA ZUCCANTE
UNIVERSITÀ DEGLI STUDI DI PADOVA

STUDENT ID
2079448

ACADEMIC YEAR

2023-2024

"Elevata, vorrei agguantarla"
— Lory Del Santo, The Lady

Contents

Listing of figures

Listing of tables

Listing of acronyms

AGS Automotive Granular Synthesizer

EV Electric Vehicle

IECV Internal Combustion Engine Vehicle

EVS Electric Vehicle Sound

QV Quiet Vehicle

CSC Centro di Sonologia Computazionale

GSA Granular Synthesis Algorithm

CAN Controller Area Network

RPM Revolutions Per Minute

SPL Sound Pressure Level

dd Detection distance

tta Time to arrival

ttr Time to react

GUI Graphical User Interface

EGUI Experiment Graphical User Interface

VE Virtual Environment

SC Source Car

MDfS Minimum Distance from Source

UN United Nations

p.d.f. Probability density function

w.r.t. with respect to

1 Introduction

Electric Vehicle (EV)s sales are increasing all over the world, thus safety related problems are emerging. EVs and Quiet Vehicle (QV)s in general represent a danger due to their low-noise emission, especially for blind or unaware pedestrians. The necessity of studying car-pedestrian interaction, defining safety parameters and developing EV-specific sounds is emerging.

In this context this thesis, born from the partnership between Centro di Centro di Sonologia Computazionale (CSC) and FIAMM, aims to explore new horizons in quiet vehicles sound design by developing an innovative synthesizer, specifically crafted to the needs of various vehicle models.

Our first goal is to develop the Automotive Granular Synthesizer (AGS), an innovative tool capable of exploring the endless resources of granular synthesis to produce electric vehicle sounds. The AGS will be designed on top of realistic real-time Simulink car model which will be connected to a sound generating granular synthesizer.

The second goal is to provide a full environment to validate and test Electric Vehicle Sound (EVS)s using the AGS. This thesis presents the development of a benchmark for EVSs and an evaluation procedure, which will be shown by performing an experiment to analyse some AGS sounds. This environment will combine many important features, some of them usually ignored by previous research papers. Adaptive traffic loudness, detailed weather conditions and effective safety parameters are among the most significant aspects.

The sound design part will not be addressed in this thesis, but it was fundamental for the completion of this work. Sound design process is peculiarly described in Giorgio Povegliano master thesis [1].

1.1 CSC

Figure 1.1: CSC logo

The CSC[1] is an internationally known institute for innovative audio production and restoration techniques. It is not only the place where engineering, art and psychology meet, but a anti-disciplinary laboratory with its own frameworks and methods which aim to inclusion and dialogue.

In 1959 Giovanni De Biasi started the research on music technology in the University of Padova. In 1959 he realized the phonoelectric organ which was able to reproduce a pipe organ using electronic devices.

Computer music made its debut in 1972. In this decade big efforts were devoted to the development of innovative software for sound synthesis, interactive synthesis and scores encoding. Finally in 1979, professor De Biasi founded CSC [2]

In the '80 CSC research focused on live electronics. Giuseppe Di Giugno, in cooperation with IRCAM, La Biennale di Venezia and CSC, built the 4i sound processor; which was used in *"Prometeo, Tragedia dell'ascolto"* by Luigi Nono (1984). As home computing signed a revolution in the music industry, CSC changed its focus. New digital signal algorithm and synthesis techniques were developed. Among all papers, we highlight the study of Giovanni De Poli and Aldo Piccialli on granular synthesis which led to the Pitch Synchronous Granular synthesis algorithm [2]

In recent years CSC has proved to be a reference point for audio reservation and valorization. New research areas such as Acoustics for Security and Multimodal Interaction for Learning, Well-Being and Inclusion made their appearance [3]

[1] `http://csc.dei.unipd.it/`

1.2 Fiamm

FIAMM Energy Technology is a multinational that operates in 60 countries, with manufacturing facilities, sales and technical offices, and a vast network of importers and distributors. FIAMM was founded in Italy in 1942 when engineer Giulio Dolcetta took over ELETTRA, a manufacturer of naval products from the Pellizzari Arzignano Group, and transformed it into FIAMM (Fabbrica Italiana Accumulatori Motocarri Montecchio). Nowadays the production includes starting batteries for automobiles, industrial batteries for backup power in critical applications, and horns for a wide range of vehicles, including automotive, marine, and emergency vehicles such as ambulances and police cars. In recent years, with the rapid adoption of electric vehicles and the introduction of new safety regulations, FIAMM has significantly invested in the development of advanced acoustic warning systems specifically designed for electric and hybrid vehicles.

2
Background

This chapter aims to recall the necessity of sound design for EVs and the main challenges which have been faced until now in this research field.

2.1 EVs market

More and more EVs have been sold in the last years and this trend is not stopping. As we can evince from IEA [4] report "in Europe, new electric car registrations reached nearly 3.2 million in 2023, increasing by almost 20% relative to 2022". China and USA also have a positive trend in sales, while South American countries are promoting low-emission road transport (see "Green Mobility and Innovation Programme" in Brazil [4]). Increasing battery capacity and affordability of EVs should definitely lead to an even stronger presence of EVs in our daily lives; it is likely that the trend presented in fig 2.1 will be confirmed in the next few years.

Before the advent of EVs we were used to loud Internal Combustion Engine Vehicle (IECV)s and their typical engine noise which makes them distinguishable. It's no mistery that EVs are much quieter than ICEVs; this problem was even tackled by the Quiet Road Transport Vehicles (QRTV) United Nations working group which stated that "vehicles propelled in whole or in part by electric means, present a danger to pedestrians". Different solutions to solve the problem have been proposed, but the emanation of additional sounds from EVs is probably considered the most effective: this is what we intend as "EV sound" (EVS).

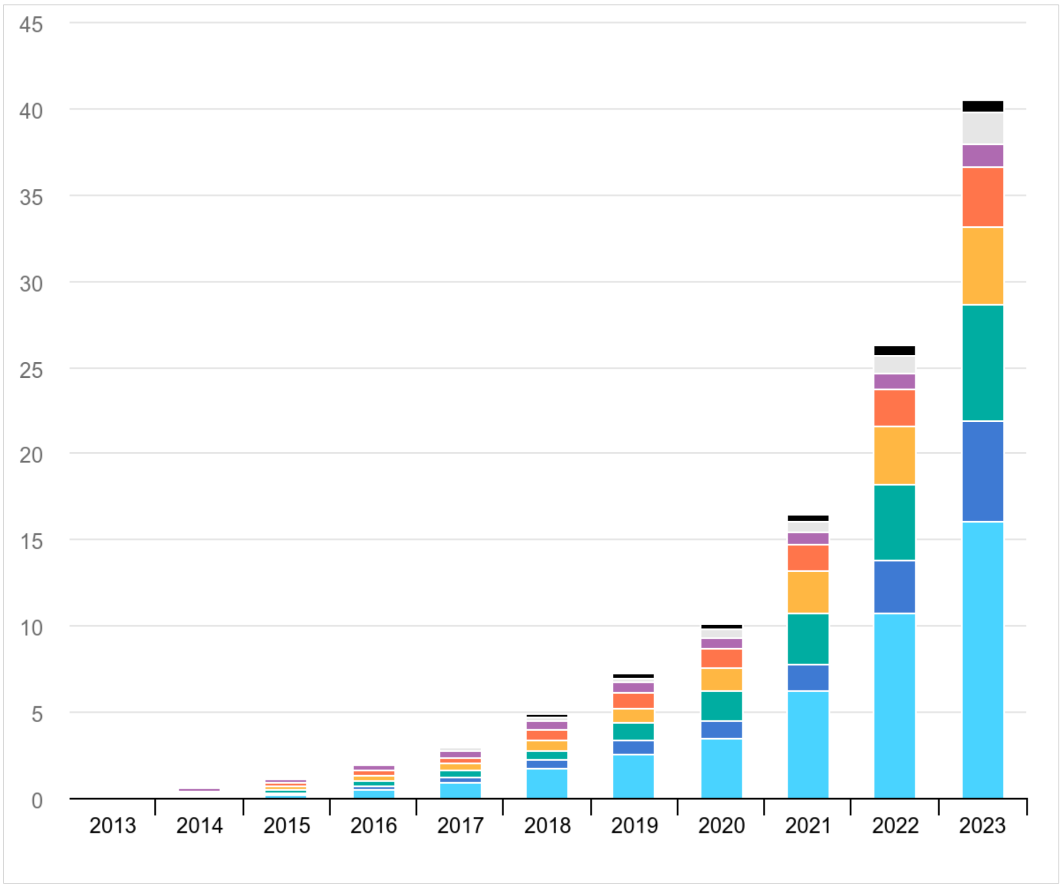


Figure 2.1: Global electric car stock - on y-axis millions of cars, different colors represents different global regions [4]

2.2 State of Art

In order to develop an effective EVS many studies were conducted.

L. Garay-Vega et al. [5] highlighted the critical conditions where EVs and ICEVs differ. They stated that "the overall sound levels for a vehicle approaching at a low speed (6 mph) are clearly lower for the EVs tested than for the ICE vehicles tested (2 to 8 dB(A))". The same study proves that tested EVs and ICEVs present "smaller differences at 10 mph and no significant difference after 20 mph".

Kerber and Fastl [6] designed an experiment dealing with the detection of the sounds of different cars in a typical urban background noise. This paper, focused on IECVs, provides a good method to analyse EVSs. This experiment shows how masking effects of the background acts on the approaching car sound.

Menzel et al [7] by testing different combination of sources and background noises proved that not every sound is suitable as EVS, implying that EVS design is a necessary process.

Few years later, Etienne Parizet et al [8] confirmed the previous results stating that some

”warning sounds do not improve performance with respect to the situation without warning”, plus they discovered that ”the temporal irregularity of warning sounds is a key factor”.

Those studies are evaluating the sound mainly considering their safety purpose and evaluating the effectiveness using detection tests.

The sound design process was also faced from a psychoacoustic perspective.

Starting from Wogalter and al [9] lexical suggestions and recommandations about sounds in QVs, Nyeste and al [10] conducted a preference study for sounds that might provide acceptable auditory cue. The psychoacoustic aspect is foundamental to promote good impressions on vehicle brand and to contribute positively to the soundscapes.

For what concerns electric vehicle sound synthesis, many different techniques have been studied to produce EVS, here we show the most significant ones.

Subtractive synthesis

This is arguably the most intuitive method among all. In subtractive synthesis sound is obtained by filtering complex waveforms. The use of this technique was proposed by June et al [11], but proved to be less effective than wavetable synthesis.

Wavetable synthesis

Currently multiple companies are still oriented towards wavetable synthesis in order to generate an EVS. Wavetable synthesis is based on the periodic reproduction of multiple arbitrary waveform; different effect such as mixing different sources and filtering could be added to obtain interesting sounds. The reason why this technique is popular is probably the ease of understanding, mapping and tuning the parameters. Furthermore it’s a very suitable method to reproduce IECVs sound, as proved in June B. et al [11].

Additive synthesis

Additive synthesis aims to produce a desired sound by summing multiple sine waves. It is known that it is possible to recreate any repetitive waveform by combining simpler waveforms or by specifying the frequency and amplitude of a series of sine waves.

The study conducted by Petiot et al [12] exploited the additive principle to design a genetic algorithm that is able to generate the most suitable sound starting from a group of sinusoids.

Granular

Granular synthesis produces sounds by positioning sound particles in a frequency-time plane. Sound texture and timbre depend on different particles, disposition in time or frequency, particle envelopes and other factors. In this context, granular synthesis has been adopted only in recent years.

Lazaro et al [13] proved that granular synthesis were able to understand the relationship between the parameters of granular synthesis and the subjective feelings of the listeners.

Giulio De Giorgi [14] developed a granular synthesis algorithm able to reproduce an ICEV, showing us the flexibility of this method.

3

Granular Synthesis Algorithm

3.1 Granular synthesis brief history

In 1959 that Xenakis modified a tape recorder to implement one of the first granular synthesizers. This process involved splicing magnetic tape into tiny segments, rearranging the segments, and taping the new string of segments together. The first composition to use this technique is called Analogique A et B composed in 1959.

Granular synthesis was first suggested as a computer music technique for producing complex sounds by Iannis Xenakis (1971) and Curtis Roads (1978). Barry Truax [15] implemented the technique with real-time synthesis in 1986 and incorporated it within an interactive compositional environment, the PODX system, see fig 3.1.

The advent of digital technology in the 1980s and 1990s further expanded the capabilities of granular synthesis. Digital synthesizers and software-based tools enabled more precise control over granular parameters, leading to more sophisticated and versatile sound design. Software like Max/MSP and SuperCollider provided composers and sound designers with platforms to experiment with granular synthesis in real-time.

Why granular synthesis for EVs?

Although granular synthesis can be considered a tricky synthesis technique it has lots of advantages:

- Its complex structure allows to create original texture in sounds just by changing few parameters (grain content, grain envelope, ...), as we will see in the next chapter. This characteristic results very useful when dealing with dynamic sounds that need to change over time (as a function of some external parameter).

Figure 3.1: PODX system
source by: `https://www.sfu.ca/~truax/pod.html`

- Granular synthesis has lots of parameters, which offer great customisation opportunities when producing a brand-dedicated sound.
- It requires far less memory space for waveform storage than EV-oriented wavetable synthesis, which usually requests multiple waveforms saved in memory.
- It can easily be tuned to produce periodic, *quasi*-periodic and *non*-periodic sounds.

3.2 Granular Synthesis overview

Before diving into the algorithm, we want to recall the basics of granular synthesis.

Granular synthesis is a method that considers a sound as the sum of multiple (even overlapping) grains; each grain can be considered as a *quantum* using the terminology of Dennis Gabor who firstly introduced the idea [16]. The features of the grains and their temporal location determine the sound timbre.

Summing up from Roads' *Microsound* [17] we can extract the basics of granular synthesis.

A grain is characterized by two main parameters, the envelope and the content, which refer respectively to the amplitude shape of the grain and the actual sound contribution. The envelope should be designed to prevent the glitches that would be caused by possible phase discontinuities between the grains. Typically, a grain is few milliseconds long; however, for certain applications, such as automotive sound design, this duration can be adjusted to fall outside this range.

Historically, in musical applications we encounter two main approaches:

- The first one is based on the use of sampled sounds to construct grains
- The second one is based on the use of abstract, entirely synthetic grains.

Disposition of grains in the time-frequency plane is also a key aspect of granular synthesis, we usually use the term *density*: which measures the amount of grain in a given area of the time-frequency plane. Inside [17] we can distinguish:

- Synchronous synthesis: where grains are periodically distributed in the time-frequency plane. Grains are activated in a synchronized or rhythmic manner. See example in fig 3.2

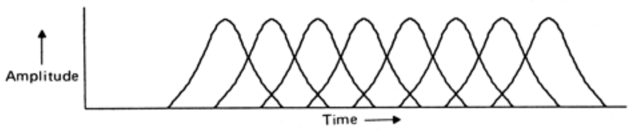


Figure 3.2: Synchronous granular synthesis [18]

- Asynchronous synthesis: where grains are irregularly distributed in the time-frequency plane. Grains are activated independently and at irregular intervals resulting in a more fluid and unpredictable sound texture, making it particularly well-suited for applications requiring dynamic and evolving audio or simulating unpredictable audio phenomena, such as the rumbling of an engine or environmental sounds. See example in fig 3.3

Figure 3.3: Asynchronous granular synthesis [18]

3.3 AGS sound generating algorithm

In this section we will explain in depth the sound generating algorithm of our synthesizer, starting from a general overview and then discussing the details.

The structure of the system is composed by 3 threads which work independently and simultaneously, each thread is then filtered. The audio coming out of each thread is summed up, adjusted in gain and fed into the speakers, see fig 3.4.

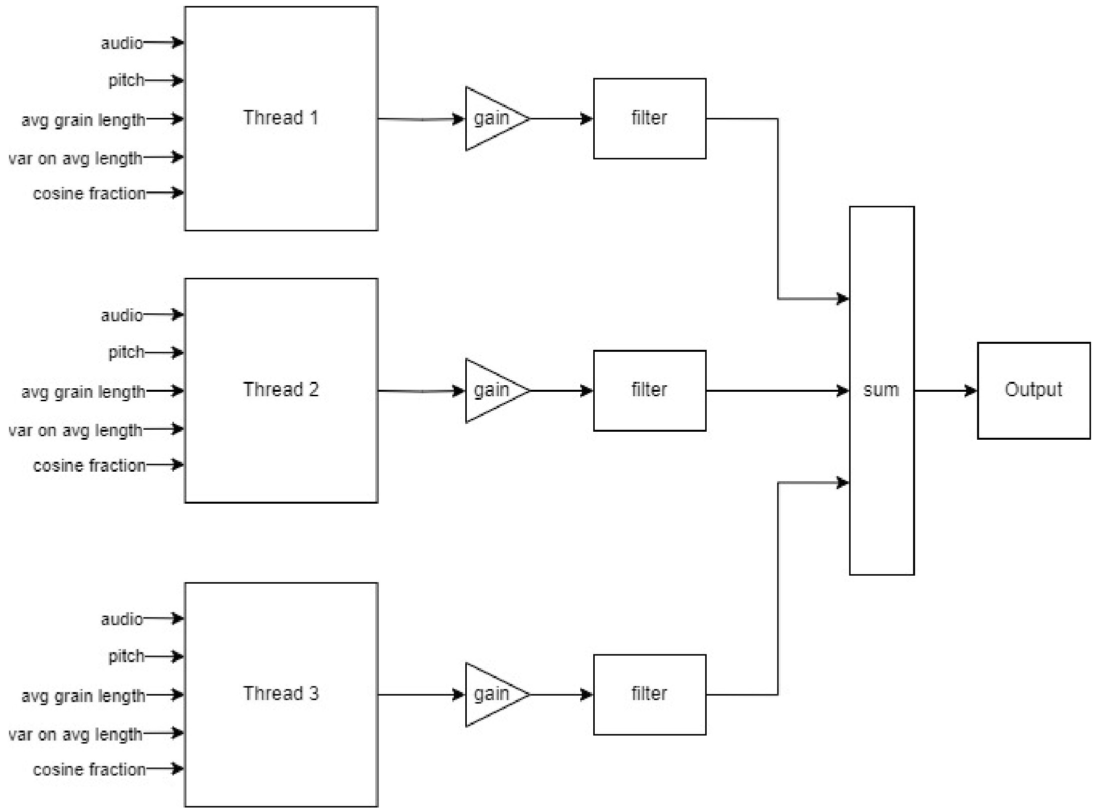


Figure 3.4: Threads block scheme

The filters adopted are generic Lowpass, Bandpass and Highpass filters, their cutoff and bandwidth is tunable as well as their order.

3.3.1 Thread

The purpose of the thread is to process a given audio file extracting grains and reorder them to form an original sound

Each thread has its own audio buffer and 8 sub-threads. Each sub-thread is responsible to randomly extract and process a grain. The output of the 8 sub-threads is then summed up and it represents the output of the thread. The internal thread structure is in fig 3.5, while

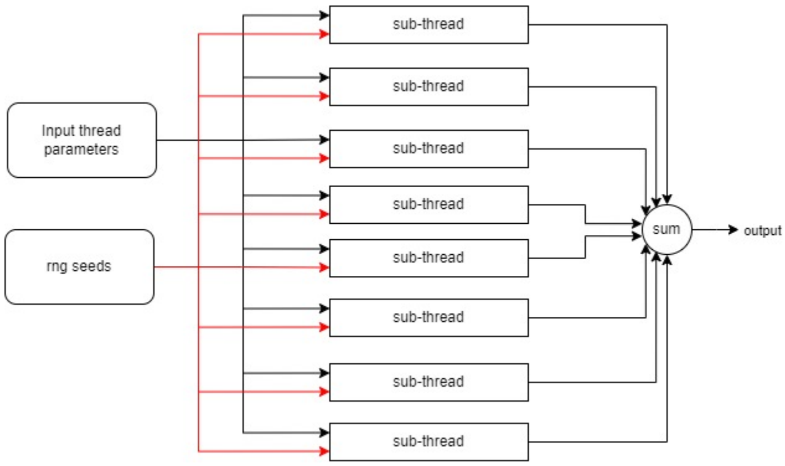


Figure 3.5: internal thread structure

The internal structure of the sub-thread is presented in fig 3.6, while in fig 3.7 we have a visual example of 4 sub-threads operating simultaneously. To better understand the workflow of the sub-thread we found to be useful a brief snippet of pseudocode 3.1. In the following sections we will describe each block of the internal sub-thread structure, see fig 3.6.

Algorithm 3.1 Sub-thread pseudocode

while $SimulationIsRunning = true$
  if $GrainFinishedFlag = true$
    Extract new grain and cut it
    Pitch shift the grain
    Window the pitch shifted grain
  else
    if $GrainRemainingSize \leq 1024$
      Store remaining samples for next iteration
    else
      Output a 1024 samples chunk
      $GrainRemainingSize \leftarrow GrainRemainingSize - 1024$
    end if
  end if
end while

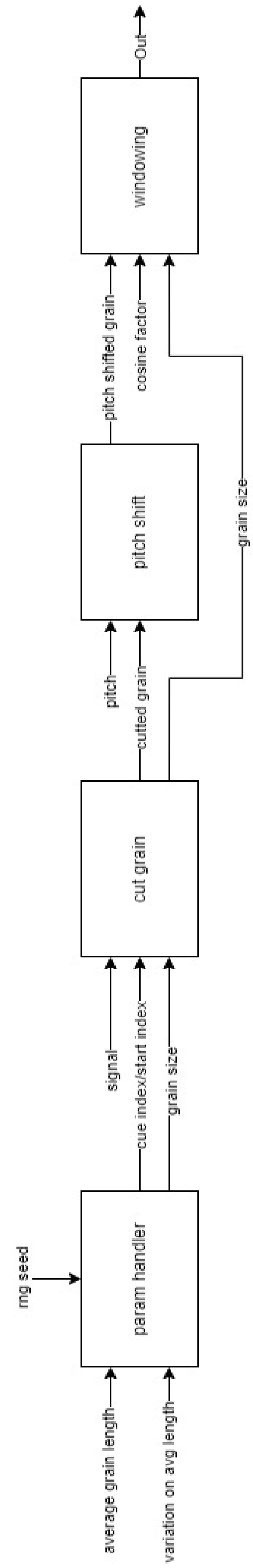


Figure 3.6: sub-thread structure

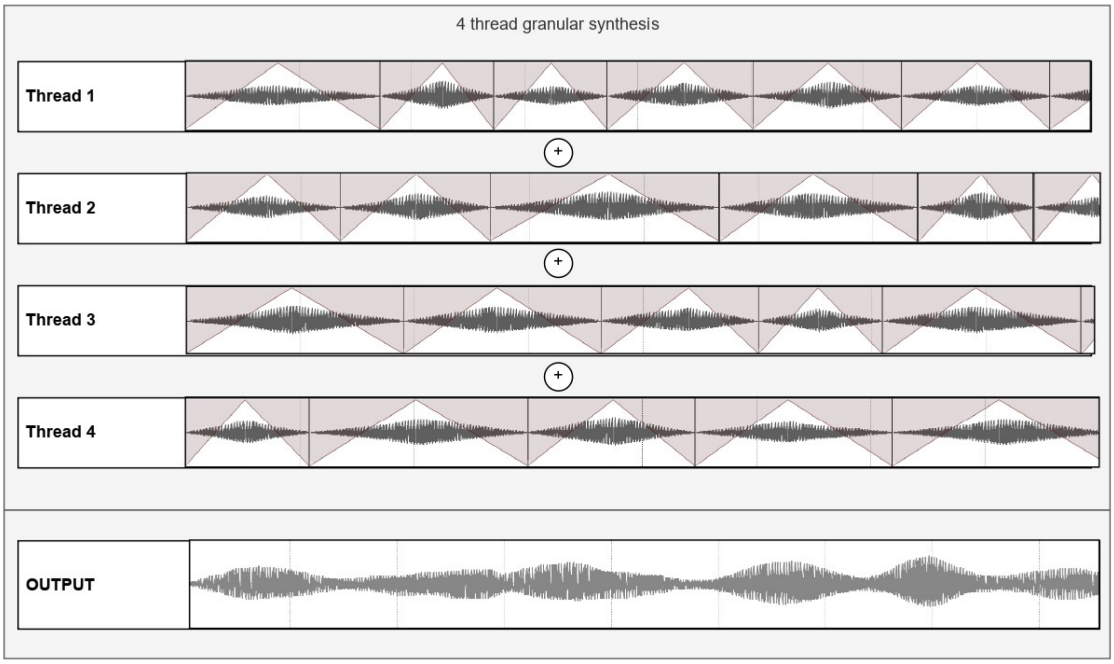


Figure 3.7: Output of 4 Threads summed to obtain the final output of the algorithm

Parameter handler

The grain extraction in based on a random process, which receives as input two parameter:

- AVG_{gl} is the average desired length of the grain
- VAR_{gl} is the variation in milliseconds allowed from AVG_{gl}

AVG_{gl} and VAR_{gl} From those parameters we are able to compute minimum and maximum possible grain length (MIN_{gl} and MAX_{gl}). Then, the algorithm generates two random numbers indicating the starting point CUE_{idx} and the grain duration GRS

- CUE_{idx}, which is the starting point of the grain, is selected from uniform random variable which ranges between $[1, audio\_length - MAX_{gl}]$
- GRS, which is the grain size, is selected from uniform random variable which ranges between $[MIN_{gl}, MAX_{gl}]$

Cut grain

This block has the simple role to cut the grain given CUE_{idx} and GRS from previous block. This block may seem unnecessary, but it was added to simplify the development in Simulink.

Pitch shift

This block is responsible to modify the pitch of the extracted grain, it was not developed autonomously, but it is a MatLab block [1]; this implementation exploit doppler effect [2]. Be careful that this block comes before the windowing block, in fact pitch shifting before windowing will return an extremely different result than the opposite.

Windowing

The windowing block receives the pitch shifted grain and applies a window function to it. The window function of choice is the cosine window, see fig 3.8. This function depends on a parameter called cosine factor *alpha*. Varying the α affects the attack/decay duration of the window, ranging from a rectangular window ($\alpha = 0$) to a Hann window ($\alpha = 1$). Note that, each thread has its own independent Cosine Factor.

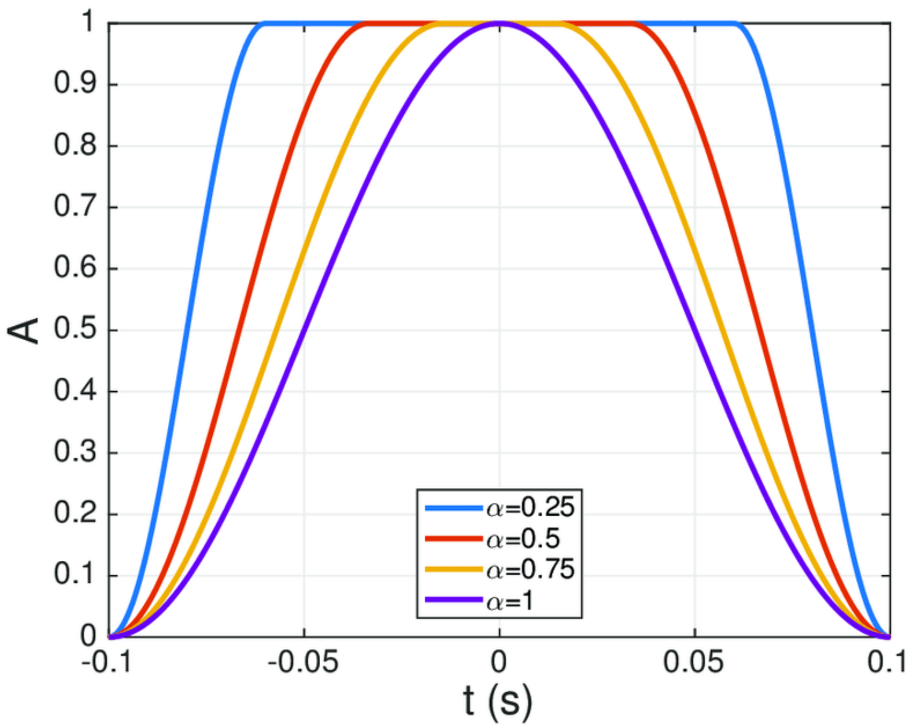


Figure 3.8: Cosine Window

[1] https : / / it . mathworks . com / help / simulink / supportpkg / raspberrypi _ ref / shift-the-audio-signal-pitch-using-matlab-function-block-in-simulink.html

[2] https://msp.ucsd.edu/techniques/latest/book-html/node115.html

Windowing block is also responsible for dividing the grain in chunks of 1024 samples to feed into the output. When the current grain is finished, a flag is raised. The flag propagates to previous blocks to announce that a new grain is needed.

3.3.2 Design choices

Now that the main architecture of the algorithm is explained we can justify some design choices.

First of all we can include AGS algorithm inside the category of *asynchronous synthesis*. We may be confused by the fact that each sub-thread produces a sequence of grains with *quasi* equal length, thus resulting in a *quasi-synchronous synthesis*. But, the presence of 8 overlapping sub-thread for each thread cancels synchronous synthesis effect. Nevertheless if parameters are tuned wisely some tremolo/jittering-like effects can be obtained.

Choosing 3 threads and 8 sub-threads per thread was dictated by the need of maintaining the computational effort low while having a sufficient number of *voices* to create a satisfying sound.

3.3.3 Implementation choices

As requested by FIAMM we developed the whole project using MatLab/Simulink Even though these applications are arguably not the best environment to deal with real-time audio, they were chosen beacuse they are very easy to understand for any developer thanks to their blocks-based structure. The main problems we encountered were varying-size variable and computational effort. The first problem was easier to face since it only required an in-depth study of Simulink features. While the second one required much more effort in deciding which were the optimal tools to let the Simulink model run effortlessly even without big computational power. The main bottlenecks of this implementations were 3:

- Pitch shift algorithm: implementing an efficient algorithm ourselves would have been a waste of time. Using Matlab developed scripts gave us the chance to speed up our code and focus on our main goals.
- The windowing block, which cannot be enhanced since is composed atomic or irreducible functions. Nevertheless the implementation of this block in low-level languages would be a big step up for our application.
- The repetition of identical blocks. This was our main concern since it was also slowing down the whole development process. The use of Simulink Reference Subsytems and

Masking tools provided an efficient way to run the code slightly faster, but most of all to avoid repetition error given by copy and paste actions.

4 Integration with car model

In this chapter we will discuss the integration between the previously described algorithm and the Simulink car model and finally giving the whole picture of the AGS and its Graphical User Interface (GUI). It is now necessary to explain the car model.

4.1 Simulink car model

The development of the full car model was conducted and described by Hascoskan Miray in his master thesis [19]. The given model simulates the dynamic behaviour of the car using Controller Area Network (CAN) bus simulation, which is a very common communication protocol used in vehicles of all types. Simulating the full communication system of a car allows us to test our sound generating algorithm directly with realistic benchmark.

In the simulation, a CAN bus network is modeled as the central communication path between the various electronic components of the vehicle. Simulation of CAN network operations replicates the data transmission and protocol behavior of a real CAN network, allowing analysis of network performance, data integrity, and reliability of communication between electronic control units.

Relying on a CAN bus network allows the testers to take into account real world delays and complications caused by this famous protocol. In particular, we've been asked to simulate a 100 ms delay caused by traffic and our model proved to be flexible and realistic enough for this request.

Furthermore CAN-based simulations CAN be easily merged together because CAN is an ID-based protocol that easily detects conflicts. This last feature opens up new possibilities and future integrations for our simulator.

The outputs of the model are extremely simple, since we only have three variables:

- Speed: ranging in interval [0, 200] mph
- Revolutions Per Minute (RPM): ranging in interval [0, 9000]
- Throttle: ranging in interval [0, 1], indicating how much pressure is applied to the pedal

The full model is composed by 2 main subsystems, which are connected in a feedback loop. This blocks are called:

- *Sensors, Actuators and Control*
- *Vehicle dynamics*

Sensors, Control and Actuators

This block is responsible for sensing variable coming out of *Vehicle dynamics* and throttle/brake commands and providing a control action to feed back into *Vehicle dynamics*.

Sensors are extremely simple systems that have to discretise the continuous incoming signals, before passing them to the Control section.

The original model had only one controlling strategy: cruise control, which takes advantage of a PID architecture to keep the desired speed constant. Even though the cruise control was effective, it was not always suitable for our purposes. This control method is not able to accelerate gradually, but uses maximum throttle pressure to reach desired speed. We wanted instead to control the model similarly to a normal automatic shift car, just by pressing throttle and brake. So, we improved the model by adding a manual control option that allows the user to "drive" using only pedals. This method proved to be very useful during test since we could test various acceleration rates and their effect on final sound.

Moving on, each actuator has a dynamic response characterized by a transfer function, simulating real-world actuation delays and inertia. For example, a throttle command does not immediately set the motor to the desired speed; instead, the response is governed by the transfer function, which in your model is represented by a first-order lag $\frac{1}{0.1s+1}$. This takes into account the physical delay in the motor's response to input changes.

Vehicle dynamics

The *Vehicle dynamics* block contains a state space model of a 4 wheeled car. It receives as control parameters throttle/brake actuators output. It depends on various fixed variables, the main ones are:

- Mass of the Chassis.
- Mass of the Wheels.
- Wheel Radius.
- Gear Ratio.

In this context, the *Gear Ratio* represents the ratio of the number of rotations of the motor (electric motor) to the number of rotations of the wheels.

To simulate real-world conditions, the model incorporates aerodynamic drag force and empirical formulas that account for the interaction between the tire tread and the road surface.

4.2 Merging car model and granular synthesis algorithm

The goal of the integration process is to obtain a synthesizer controlled only by driving inputs (throttle/brake). We showed that car model is able to reliably simulate a car and now we have to map output coming out of car model into meaningful inputs in the granular synthesis algorithm. For ease of implementation we focus our attention only on few car output variables: speed, RPM and throttle. So now before continuing we have to make some sound design observations.

Everyone is familiar with the sound of IECV vehicles and we CAN start from that assumption to make our first steps. In IECVs we know that the higher the RPM, the higher the pitch of the sound, so we want to mantain this behaviour in our model. We CAN also state that also speed affects signifiCANtly the pitch and the texture of the IECVs sound, then we want speed to play an important role in our AGS. Finally throttle pressure, which may seem superfluous, it's important: the loudness of a car suddenly accelerating is usually higher; we CAN easily obtain this feature.

In fig 4.1 we CAN see the block scheme that handles the mapping process; we will now discuss in depth every block

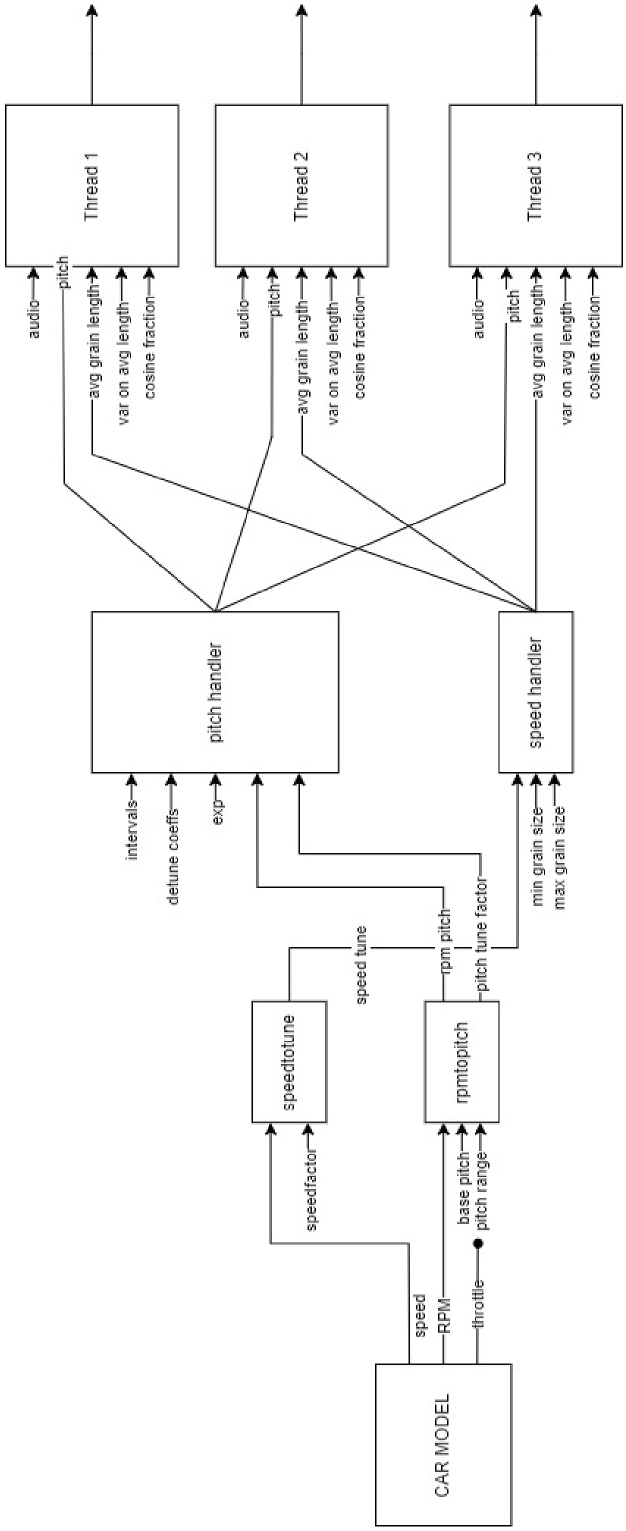


Figure 4.1: Block scheme of integration mapping

Rpm to pitch

This block takes as inputs:

- base pitch: which represents the minimum pitch available
- pitch range: which represents the maximum pitch span
- car RPM

These parameters need to compute:

- intermediate pitch: $basepitch + (rpm/maxRPM) * pitchrange$
- pitch tune factor: $rpm/maxRPM$

This block is a preprocessing block needed for the Pitch handler block

Speed to tune

This block takes as inputs:

- car speed
- speed factor: a variable taking values in range [0, 1], 1 is default value

These parameters need to compute:

- speed tune factor: $speedfactor * speed/maxSpeed$;

This block is a preprocessing block needed for the Speed handler block

Speed handler block

The role of this block is to compute the average grain length of the grains. This block takes as inputs:

- speed tune factor: a variable taking values in range [0, 1], 1 is default value
- max grain length Mgl: specified by the user in milliseconds
- min grain length mgl: specified by the user in milliseconds

To compute the average grain length (agl) we follow these steps:

1. convert Mgl and mgl from ms to samples
2. $agl = Mgl - speed\_tune\_factor * (Mgl - mgl)$

Basically the average grain length is mapped to the speed: the faster the car the shortest the grain and viceversa. We stated at the beginning of this section that speed must affect the overall sound and this block explains how.

Pitch handler block

This block is the most important and it regulates the desired pitch to feed into the pitch shifting algorithm; it takes as inputs:

- intervals *ints*: distance from base pitch expressed in semitones
- detuning coefficients *det*: specify how much detuning from specified intervals occours as the car increases RPM
- pitch evolving coefficinents *pvc*: indicate how base pitch increases over RPM (linear, quadratic or cubic function are available)
- base pitch *bp*: which is the *intermediate pitch* coming out of *RPM TO PITCH* block.
- pitch tune factor *ptf*: coming out of *RPM TO PITCH* block.

Note that each thread has its own interval, detuning coefficient and pitch evolving coefficient, while base pitch and pitch tune factor are the same for every thread. The final pitch it's easily computable from variables above:

$$pitch = bp^{pvc} + ints + det * ptf \tag{4.1}$$

Now, we CAN make some observation on the behaviour of this block:

- intervals are immutable offsets and CAN be used to achieve specific combinations around the base pitch (for example triads)
- detuning coefficients are useful to create oddly and unexpected but pleasuring effects as the car increases RPM. As the car increases RPM intervals CAN stretch as sometimes happens even in IECVs sound.
- pitch evolving coefficients are dictating the evolution of the base pitch of each thread, their modification CAN be very disrupting on the final result.

4.3 AGS GUI

As we have seen in the previous chapter the car model merged with the granular synthesis algorithm is complex, an accessible and intuitive GUI was needed to help the user. The AGS GUI was deleoped using MatLab App Designer Tool which is capable of creating an interface to control therefore a Simulink model in real-time. The GUI is composed by three panels we will explain next

4.3.1 Control panel

The control panel is not always accessible, it pops up when starting the simulation and closes when it stops, see fig 4.2. This synthesizer is designed to be directly controlled by the imaginary driver, so the control is over the throttle/brake or the speed. Let's start describing the interface:

- Switch Control Type: allows you to choose between Cruise Control mode or Manual Control: in Cruise Control mode you have control over the desired speed you want the vehicle to reach, while in Manual Control mode you have control over the virtual pedals.

- Speed Slider: active only in Cruise Control mode, allows you to choose the desired speed.

- Throttle and Brake Sliders: active only in Manual Control mode, allow you to press the imaginary pedals, a value of 1 means that the pedal is completely pressed.

- Speed Display: shows the current car speed in mph

- RPM Display: shows the current engine RPM

4.3.2 Synthesis panel

This section is dedicated to explaining the various tunable parameters for using the AGS. If you haven't already, please refer to Chapter 3 and 4 to understand how our implementation of granular synthesis works, refer to Section 5.1

As evident from the numbers 1, 2, 3, and 4 in Figure 4.3, the control knobs for each thread are organized in columns. Let's begin by explaining the function of each knob:

1. Interval: enables you to adjust the pitch of the thread from base pitch. The intervals are measured in semitones and span two octaves, ranging from one below to one above the original pitch of the sound.

2. Detune: allows you to select the amount of dynamic pitch shifting in semitones for each thread. The detuning reaches its maximum value when the car is at its highest RPM. For example, if Detune 1 is set to 7 semitones as the car accelerates to reach its maximum speed and the engine reaches its maximum RPM, the pitch will then shift upward by +7 semitones.

3. Filter: these 3 blocks allow to control the parameters of the filters (cutoff frequency, bandwidth and filter type)

4. Voice Volume: this knob represents the volume of each thread.

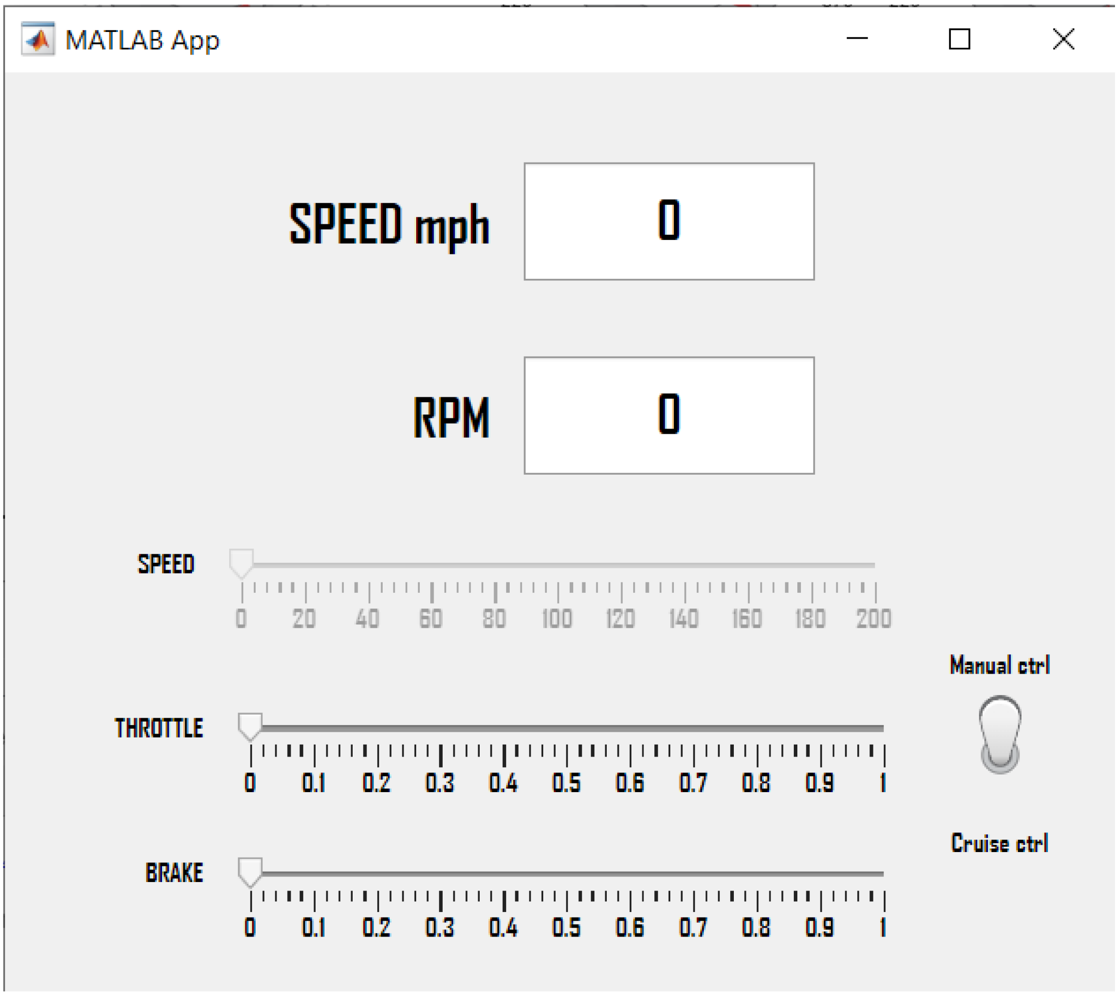


Figure 4.2: Control Panel

We now explain the control knobs on the right, represented in Figure 4.3 by the numbers 5, 6, and 7. These parameters regulate the general settings of all the threads. In other words, by adjusting these parameters, it is possible to alter the settings of every thread simultaneously:

5. Master Volume: allows you to set the gain of all the threads at the final stage.
6. Base Pitch: simultaneously regulates the initial pitch of all threads. It's important to note that the effect of this parameter is combined with that of Interval. When both are used, the final pitch shift in semitones for each thread will be the sum of the two parameters.
7. Pitch Range: simultaneously regulates the dynamic pitch shift of all threads, functioning similarly to the Detune knobs but applying to all threads at once. Like with Base Pitch and Interval, the effect of this parameter is combined with that of Detune, so the amount of pitch shift reached at the highest rpm and speed will be the sum of the two parameters (for each thread).

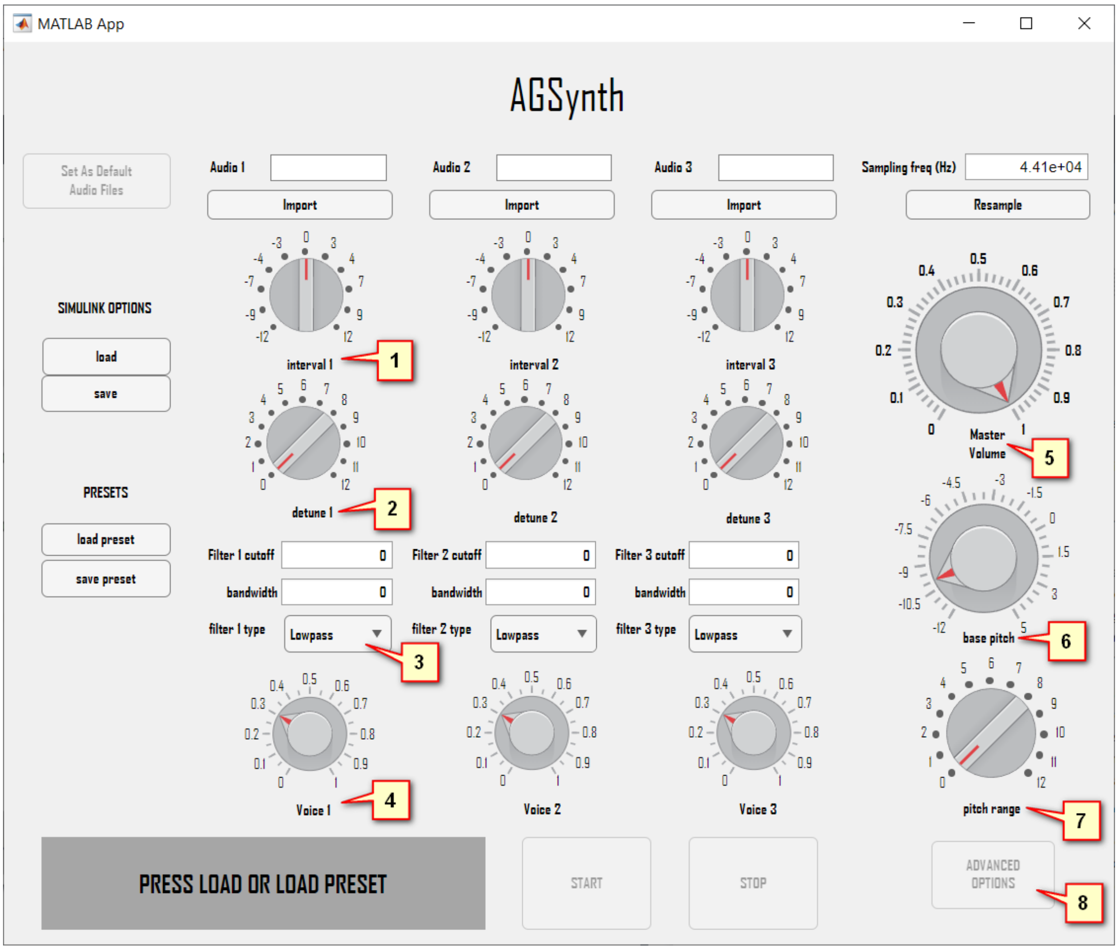


Figure 4.3: General Panel

8. Advanced Options: by pushing this button a new screen will appear, allowing to adjust some additional parameters that we will now present.

In the Advanced Options tab, see fig 4.4, you CAN find all the parameters related to the Granular Synthesizer, as well as a control for pitch dynamics. It's important to note that these parameters are closely linked to the behavior of the car, its speed, and the engine's rpm:

1. Min Grain Size: represents the average size of the grain when the car reaches its maximum speed.
2. Max Grain Size: Represents the average size of the grain when the car is stopped, i.e., when the engine is at its lowest speed

   In the range between the maximum reachable speed and the lowest speed, an average grain value is computed. In other words, as the car accelerates, the average grain size changes smoothly from the maximum (selected using knob 2) value to the minimum value (knob 1). The opposite occurs when the car brakes to come to a full stop; the grain size varies from the minimum value to the maximum value, recall section 3.3.1

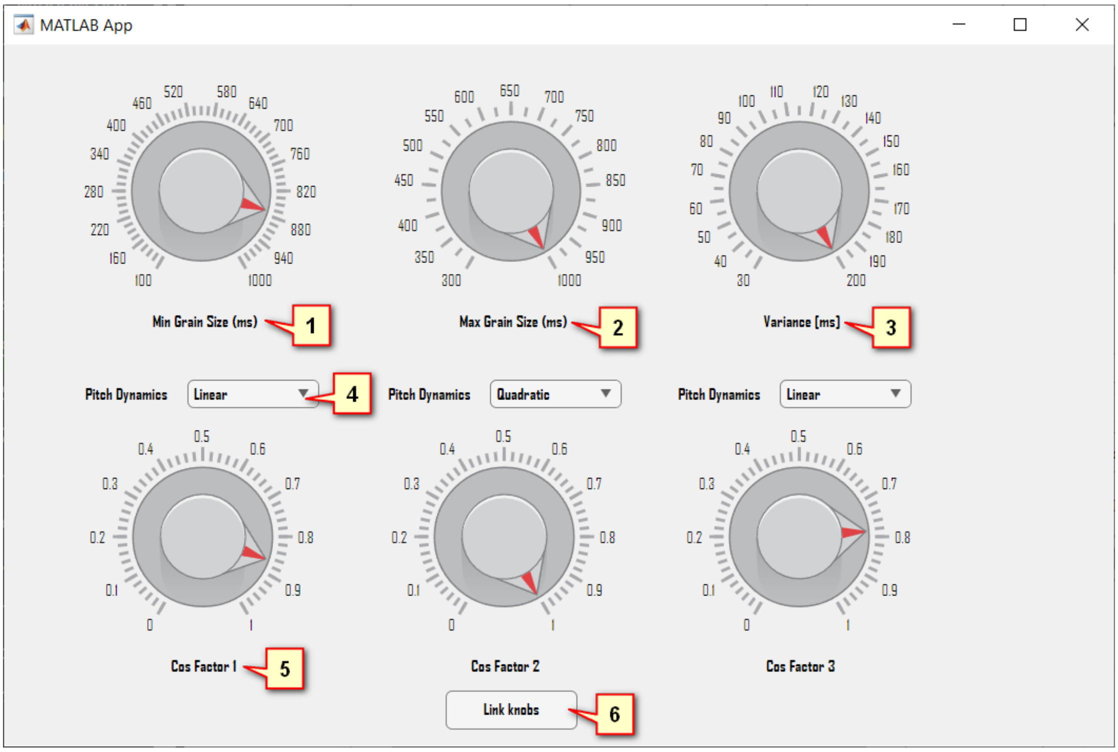


Figure 4.4: Advanced Options

3. Variance: represents the variation in the average grain size at each iteration. As explained earlier, to ensure the algorithm's effectiveness, a degree of randomization is necessary. This knob allows the user to select the maximum difference between the average duration of the grain and the duration randomly extracted by the algorithm. Ultimately, the final duration of the grain will fall within the range of Average Grain Size $\pm$ Variance.

4. Pitch Dynamics: this setting controls the exponent of the pitch dynamic. Selecting 'Linear' provides the default result, maintaining a steady pitch progression and reaching the final pitch set in Pitch Range. Choosing 'Quadratic' and 'Cubic' allows the pitch to rise faster and reach higher levels, resulting in a more pronounced pitch variation over RPM.

5. Cosine Factor: allows you to adjust the attack of each grain. As explained in section 3.3.1.

6. Link knobs: by pressing this button, the user CAN link the Cosine Factor of threads 2 and 3 to the Cosine Factor of thread 1. This allows the user to adjust the attacks of each thread simultaneously using the first knob on the left.

5
Sound design and evaluation

In this chapter we will recall the basics of sound design for EVs and determine various evaluation criteria. We will answer to the apparently simple questions: how should the sound be design? Various approaches to sound design are possible:

- Design for safety purposes
- Design from psychoacoustic perspective
- Branding oriented design

Since vehicle branding is out of our thesis interest, we are focusing on the first two points.

5.1 Psychoacoustic oriented sound design

My colleague, Giorgio Povegliano, focused his attention on psychoacoustic parameters of sound and their application to EVS design: we are recalling some of his conclusions [1].

The auditory signals of an EV starts from its sound design process and can influence the perception of the consumers. In IECVs the sound was merely dependant on mechanical features and was strongly constrained by the nature of these vehicles. On the other side quite vehicles offer a giant opportunity to convey a message to the customers through innovative sounds.

Psychoacoustics is the field that examines the behavioral effects of sound stimulation [20] and studies how factors like frequency, intensity, and duration of sound shape our auditory experiences. This subject offers valuable insights into how we interpret and react to various sounds and provides metrics to evaluate designers ideas with respect to the desired customers reaction [21].

The main challenge in psychoacoustic is to define which objective parameters in sound design are responsible for certain feeling and why. This is an hard task since each person has an individual perception that can be influenced by personal experiences and cultural differences [22]. To capture its subjective nature, sound quality in the automotive field is described using semantic descriptors, which are adjectives that characterize the sound, such as *powerful*, *sporty*, *refined* and *harsh*. Different studies have been conducted using this terminology and the result is that sound quality in vehicles can be understood through two main dimensions: *Power* and *Comfort* [22]. By analysing prior research in his work Giorgio found seven key acoustic features: *Roughness*, *Linearity*, *Engine Firing Order*, *Sound Pressure Level of Low Engine Orders*, *Loudness Level*, *Sharpness*, *Impulsiveness*.

5.1.1 Mapping sound characteristics to AGS

Input Audio

This audio serves as the foundational material from which all subsequent sound manipulations will be developed. The input audio fundamentally influences the final sound output, as it is the core element that the granular synthesis algorithm will process. Follows a list of some key factor for input audio choice:

- Source Quality: high-quality recordings with minimal noise and distortion provide a better foundation for sound design; audios with higher sampling frequencies are preferable. However, it's important to avoid resampling when possible.

- Sound Characteristics: The spectrum of the input audio is the most critical factor in shaping the final output of the algorithm. For instance, if the goal is to produce a sound perceived as aggressive, input audio with high loudness, rich high-frequency content, and a wide dynamic range (including sharp transients or small impulses) is preferred. In contrast, to generate a sound perceived as relaxing, input audio with lower loudness level, limited sharpness, and a more restricted dynamic range is preferred.

- Flexibility: a sound that can be easily adjusted in terms of pitch, duration, and filtering will provide more creative possibilities.

Pitch

The degree of pitch shifting during acceleration significantly influences users' perceptions of speed. Larger pitch intervals can create the impression of faster acceleration and higher speeds. Additionally, pitch adjustments can alter the harmonic content of the sound during acceleration, enriching the higher frequencies and contributing to a more powerful and dynamic auditory sensation. Conversely, by limiting the extent of pitch shift, the sound can maintain its original texture and timbre, providing a more consistent auditory experience.

Grain Duration

Grain duration influences the continuity, texture, and dynamic range of the synthesized sound. Short grain length tend to create a more *staccato* or glitchy effect, that can introduce impulsive elements and create powerful sounds. In contrast, longer grain length makes the audio output more refined, increase the *Comfort* feeling.

Furthermore, longer grains enable the preservation of more detailed aspects of the original sound, maintaining a closer resemblance to the source material. In contrast, shorter grains can lead to a more fragmented outcome that may deviate significantly from the original input audio.

Grain Envelope

By altering the shape of the window envelope, the frequency content, texture, and timbre of the final audio can be adjusted. Shorter attack times produce transients, which manifest as discontinuities in the final audio. Such transients enhance the impulsiveness of the final sound, creating a sense of power. Conversely, longer attack times smooths the transition between successive grains, reducing or eliminating transients and resulting in a more relaxing sound.

Filtering

As we have seen in chapter 3 filters are particularly useful as they are applied at the end of each processing chain; allowing Through filters, users can fine-tune the spectral characteristics of the output, such as attenuating unwanted frequencies. For instance, an highpass filter can be used to remove higher frequencies, reducing the sharpness of an overly aggressive sound. Conversely, a lowpass filter can be employed to cut lower frequencies, reducing roughness to obtain a more refined sound.

5.2 Safety oriented sound design

Safety oriented sound design has the final purpose to maximise the audibility of EVs from the pedestrian point of view; thus, minimizing the road accidents caused by pedestrians or drivers negligence or distraction.

Safety distance for collision avoidance

As stated in Kerber and Fastl [6] who reported Green studies [23] collision avoidance is strictly correlated to driver attention and reaction time. For concentrated road users the time between the perception and reaction can be as short as 0.7 seconds. People who are distracted need about twice the time (1.5 seconds). In slow driving car scenario we can compute the minimum distance for collision avoidance for different car speeds and reaction times: in fig 5.1 and table 5.1 results are reported considering a braking deceleration of $8m/s^2$, figure is recreated from [6]. These data will be extremely helpful to evaluate the effectiveness of the sound during simulations. Let me point out how minimum distance for collision avoidance increases almost linearly: at $50km/h$ fast reaction time can guarantee approximately 10 more meters to brake.

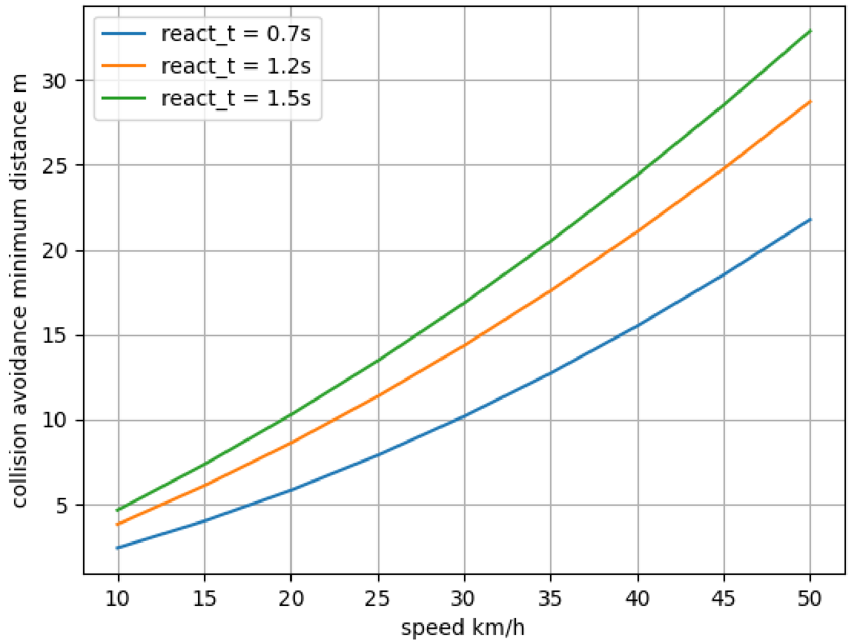


Figure 5.1: minimum distance for collision avoidance, assuming $8m/s^2$ of deceleration when braking- recreated from [6]

All this studies were conducted considering dry asphalt. We have to stress that in wet conditions braking process it's much more complex since it is dependent on many other factors,

	speed [km/h]								
reaction time [s]	10	15	20	25	30	35	40	45	50
0.7	2.4	4.0	5.8	7.9	10.2	12.7	15.5	18.5	21.8
1.2	3.8	6.1	8.6	11.3	14.3	17.6	21.0	24.8	28.7
1.5	4.6	7.3	10.2	13.4	16.8	20.5	24.4	28.5	32.9

Table 5.1: minimum distance for collision avoidance. On horizontal axis speed in km/h, on vertical axis driver reaction time

starting from drainage and tyres performance. Considering low car speeds and good drainage of water we will use the same braking distances as in dry road case.

Following this framework we are now defining some commonly used variables:

- *Detection distance (dd)*: distance at which the car is detected by pedestrian.
- *Time to arrival (tta)*: time needed for the car to arrive at pedestrian location maintaining its speed constant.
- *Time to react (ttr)*: time left for the pedestrian to avoid collision, it is counted from detection time.

Loudness

In EVS design loudness is usually measured in dBA rather than dB; dBA takes into account the varying sensitivity of the human ear to different frequencies of sound using an A-weighting filter, see fig 5.2.

When considering loudness we have to take into consideration two factors: safety and noise pollution. The ideal result would be to create the most quiet sound possible while keeping high safety standards for pedestrians. Many regulations have been adopted by different countries, in our thesis we will follow United Nations (UN) regulations written in [24]; this document is very complex and detailed, we will pick only essential directives for our experiment.

According to UN regulations loudness measuring points are placed at a 2 meters distance from the side of the car. Measurements are conducted both in standstill and low speed rolling conditions - usually 10, 20 km/h. In every case the measured loudness must be below 75 dBA: this will represent our loudness constraint.

Furthermore from preliminary experiments (similar to chapter 6), we derived some qualitative conclusions:

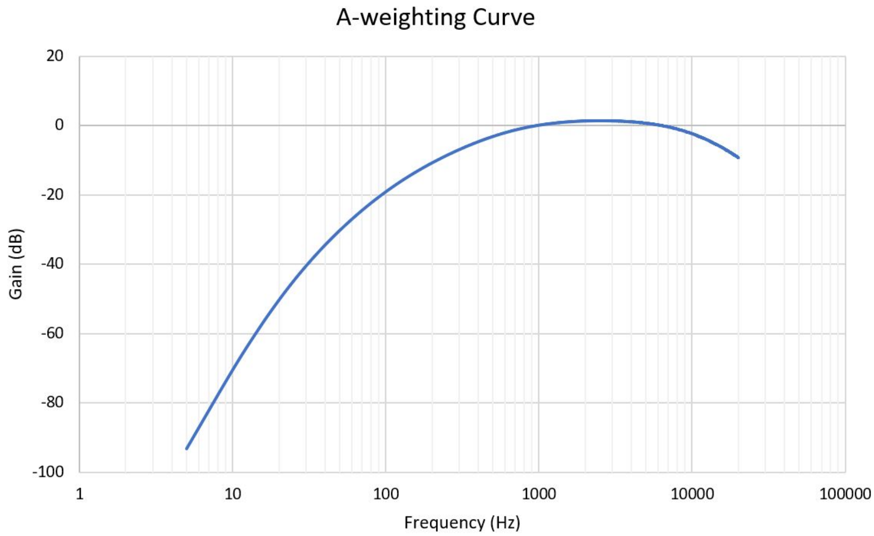


Figure 5.2: A-Weighting filter

- For speed ranging from 0 to 20 km/h pedestrians are able to detect the car within the safety distance range even using low loudness sounds.
- For speed around 30 km/h pedestrians are able to detect the car within the safety distance range only if we slightly increase SNR with respect to the background noise.
- For speed around 40/50 km/h pedestrians are able to detect the car within the safety distance range only if we significantly increase SNR with respect to the background noise.

In simple terms, if we keep the loudness constant for every speed the car will be heard anyway by the pedestrian, but not within the safety distance range. This leads us to a speed-wise loudness design, which is a rudimentary adaptive approach to car loudness control.

Additional safety parameters

In Misdariis N. et al [25] paper some interesting features were introduced:

- spectral flatness (SFM): a measure of the noisiness of a spectrum; it is computed by the ratio of the geometric mean to arithmetic mean of the energy spectrum value in each considered frequency band. Typically, SFM is close to zero for tonal sounds and close to one for noisy signals.

$$\text{SFM} = \frac{\left(\prod_{k=1}^{K} a(k)\right)^{1/K}}{\frac{1}{K} \cdot \sum_{k=1}^{K} a(k)}, \text{where } a(k) = \text{ amplitude in frequency band } k$$

- modulation rate (m): a measure of the signal modulation, computed by the ratio between difference and sum of the extreme values of the energy envelop.

$$m = \frac{E_{\max} - E_{\min}}{E_{\max} + E_{\min}}, \text{ where } E = \text{ energy envelop}$$

Those parameters themselves do not provide a safety measure, but are helpful to categorize sounds on a bidimensional plane. In fig 5.3 we can see how sounds with different lexical descriptor cover different zones in the plane. Filling the gaps could be a good sound design technique to avoid overlapping with other types of sounds.

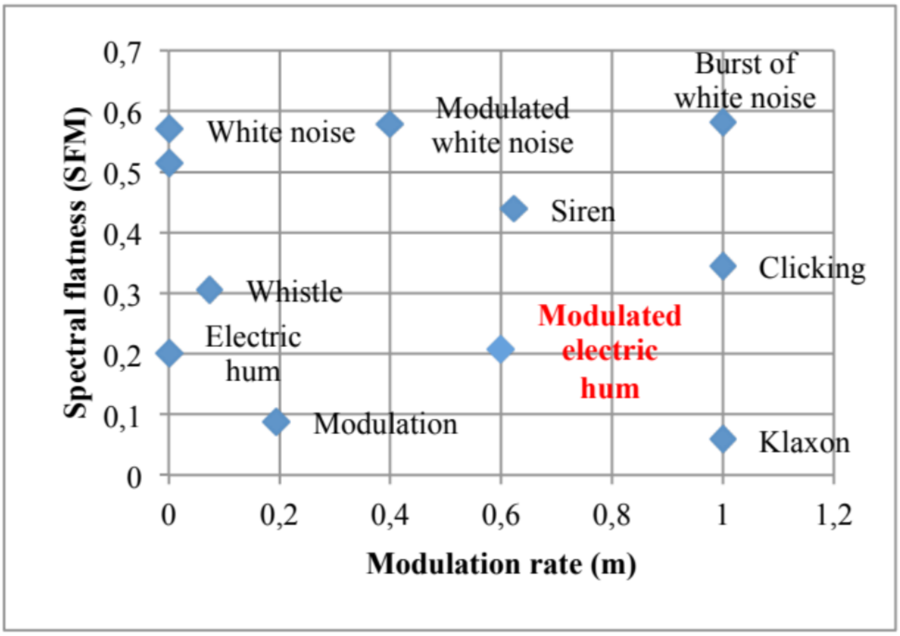


Figure 5.3: sfm-modulation rate plane [25]

5.3 Evaluation environment

Sound design engineers cannot underestimate the importance of testing environment; in fact, sound design is often a trail and error process. We identify 3 main types of environment.

Real world environment

Real world scenarios are usually on-road. Typically, a *test vehicle* emitting the sound is driven along a path and an audience has to detect the incoming car. The surrounding environment has to be accurately selected because it will deeply affect the experiment results. This evaluation method ensure maximum realism because all real world variables are considered. On the other hand there are many disadvantages:

- Environment specific experiment: changing scenario requires the study of a new experiment
- Error-prone approach: real world scenario adds many variables in experiment setup
- Expensive setup: the cost of audio reproduction system is very high
- Difficult to replicate: environmental variables are constantly changing and are usually not directly controllable

Laboratory environment

Laboratory environment experiments try to simulate real world scenarios in reverb controlled (or even anechoic) rooms. Here it's easier for the pedestrian to maintain full attention on the incoming vehicle. Background noises and other stimuli have to be replicated as reliably as possible. This method provides a good compromise between real world and virtual environments: simulation flexibility and repeatability represent its strengths. Expensive setup and lack of context are the main downsides.

Virtual environment

A Virtual Environment (VE) can be defined as a display system that creates an illusion of being in another physical place. VE provides the means to fully simulate an immersive scenario for the pedestrian, where the researcher has control over all the variables. VEs ensure better context and even more flexibility than laboratory environment. Experiment setup could be less expensive than previous cases, but setup time and VE development require much more effort than the first two methods.

6
Validation experiment

Until now we described the AGS and its functionalities, then we briefly introduced some sound design and evaluation techniques. Now we want to validate our AGS through an experiment. The main challenge presented in this chapter is the design and setup of an experiment environment to test an EVS.

For our tests we excluded real-world scenarios because of their poor flexibility, instead we chose to conduct the experiment in a Virtual environment

6.1 Previous related work

In August 2007, Kerber and Fastl [6] conducted an experiment designed for measuring detection of the sounds of different IECVs in a typical urban background noise. This experiment used only two static background noises and different car recordings. They defined a function to establish detection distance needed for collision avoidance; we followed a similar model in section 5.2. This experiment shows how critical is the masking effects of background on incoming car sound with respect to different speeds. Additionally they measured masked threshold for approaching vehicles and they developed a simple algorithm to predict vehicle distance at perception.

In 2011 Menzel and Yamauchi [7] improved Kerber experiment with new the input device and layout of the test procedure. ”In one experiment, subjects were asked to adjust the level of the warning sounds so that they are clearly audible” while, ”in a second experiment the goal was to adjust the level so that the warning sounds are just audible”. They highlighted that recommending one fixed level could be problematic. Some sound proved to be properly audible only in certain background conditions.

In June 2013, Misdariis et al [25] designed a preliminary study for EV sound design. The aim of the core experiment of this study was to measure the reaction time associated to the detection of the 10 equalized warnings in each of the 3 backgrounds. For this purpose a pseudo-realistic passing-by scenario was built, based on Kerber and Fastl work. They detected "a strong influence of the type of his three warning sounds as well as his four background noises on the level difference between "just audible" and "clearly audible" thresholds", confirming Menzel conclusions.

In February 2016, Sneha Singh published her thesis [26] which includes a literature review of both regulations on EV sound design and validations techniques. In particular it analyses performance variables such as: detection distance and time to arrival.

From the previously reported papers we want to highlight the main weaknesses in order to lead us to new improvements:

- the simulator software is either poorly described or extremely basic
- few background noises were considered and they are all loudness static
- different weather conditions are not taken into account
- spectrograms of tested sounds are not reported or analyzed, thus correlation between source features and preception are not explored
- detection distance is the only important safety parameter cited and used in the analysis
- ANOVA analysis is often very general and poorly informative

This experiment is based on an improved version of Kerber and Fastl [6] and Menzel [7] experiments. Background noise analysis uses mostly previous research by Cai et al [27] and Misdariis et al [25].

6.2 Simulator

When testing EVS many academic publications [7],[6], [25] use extremely basic simulators that are simplifying real-world fenomena and ignoring possible scenarios. Usually reflections is neglected and air absorption is poorly approximated. In our work we want to exploit better tools to obtain more solid results.

Our simulator is based on a modified version of the *pyroadacoustics*, developed by Stefano Damiano and Toon van Waterschoot [28]. *Pyroadacoustics* is an open-source library that enables to simulate the sound propagation in a road scenario. This simulator offers the possibility to run a simulation considering:

- One sound source moving along an arbitrary path
- Multiple background noise
- A set of microphones arbitrarily placed in the scene

Simulator features *Pyroadacoustics* offers multiple key features that add realism to the simulation:

- Advanced air absorption model implemented with FIR filters; it depends on 3 parameters temperature, pressure and relative humidity.
- Doppler effect which is not negligible when modeling an approaching sound source. This effect is responsible for hearing sounds with higher pitch when approaching towards us.
- Road reflection which is one of the main components in every road scenario; implemented with FIRs filters as well. This feature is dependent on the construction material.
- Microphone directional patterns can be modified (for example: cardioid, omnidirectional, ...).

Audio recording

The desired output of the simulator is an n-dimensional audio file (stereo is considered 2-dimensional audio file). The naive approach would be to run a simulation with n cardioid microphones oriented as desired; for stereo audio it would be 2 microphones in opposite direction, see fig 6.1. Unfortunately this method proved to be highly inefficient, too slow to collect a full dataset.

An improvement was necessary to speed up the simulation, so we developed the following procedure:

1. run the test using one omnidirectional microphone fig 6.2

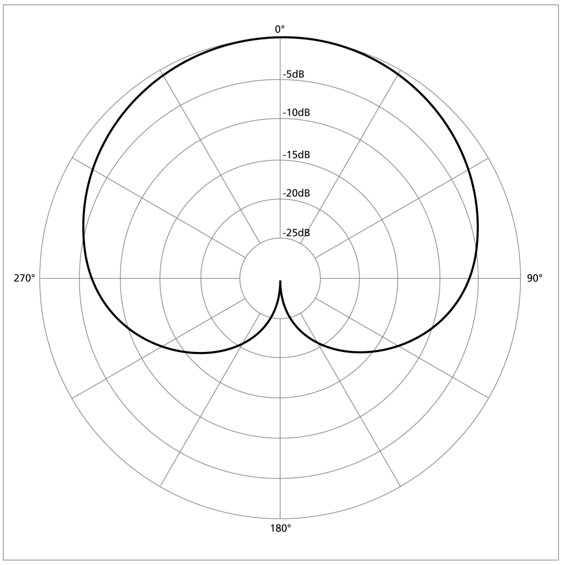


Figure 6.1: Cardioid microphone function

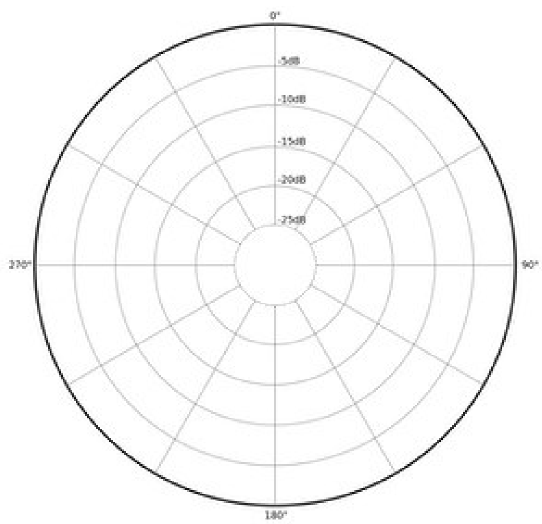


Figure 6.2: Omnidirectional microphone function

2. save the position and orientation of the source with respect to (w.r.t.) the microphone

3. using these information to post-process the output as if it was recorded by multiple cardioid-like microphones.

This simple algorithm allows us to run the simulation once and adapt the audio to many microphones configuration.

Audio storage and reproduction

Dataset creation and listening simulations are all implemented in python language. Dataset is saved in .npy format, which basically is a numpy *ndarray* containing all possible audio samples given by different scenarios.

For this experience we developed a custom GUI (Experiment Graphical User Interface (EGUI)) using Tkinter. The EGUI was designed to completely automate the experiment, allowing the particpant to take only few actions. For each samples of the test the user has to press the start button to reproduce the sample; it plays until the listener presses the stop button; detection time and other variables are computed right away. Then the user is asked to indicate the direction of the incoming vehicle. Data are then saved in .csv format inside the results dataset.

6.3 Experiment

6.3.1 Experimental procedure

Our listening test took place in a silent room. The goal is to measure detection distance - *dd* - of incoming vehicles having different sounds and speeds in multiple weather conditions. The listener has to hear the audio samples obtained from simulation and press a button when as soon as they hear an incoming car. Car are approaching at unspecified times. *dd* is easily computable knowing the car speed and time of detection after audio sample start. After detection, the listener has to specify the direction of the incoming car. When computing the *dd* we assume that the listener has the reaction time of 0.56 seconds.

As presented in Misdariis et al [25] learning effect given by repetitions and temporal patterns is highly influential on final result. To mitigate its effect we have to randomize as much as possible the patterns of simulations. Before starting the experiment, listeners are allowed to hear all EVSs before the experiment starts and to take a couple of tries.

A total of 24 people (14 males, 10 females, with age ranging from 18 to 60 - mean age is 36.5) took part to the experiment . All participants have no documented auditory impairment. Auditory tests were performed in a silent room with a background noise of 40 dB and using a pair of closed-back headphones (Audiotechnica ATH-M20x).

Stimuli

We decided to test 3 EVSs in different urban scenarios where Source Car (SC) is approaching at 3 different speeds - 20, 30, 50 km/h. We assume the SC to be a point in space and we define the Minimum Distance from Source (MDfS) as the minimum distance from car to listener during the experiment; in fig 6.3 we can see an example where the car is moving from left to right. Given a lane length of 3 meters, the car is considered to pass at 2 meters from the listener. The car can follow different straight line path: for example left-to-right, right-to-left, front-to-back, back-to-front. Two weather conditions (dry, rain) and two background noises (busy urban traffic and speech-like noise) are available. Every scenario is composed by:

- One EVS out of 3
- One weather condition out of 2
- One background noise (see 6.3.1 for details) out of 2
- One car speed out of 3

The total number of combinations is 36, the listener has to hear them all once, in random order.

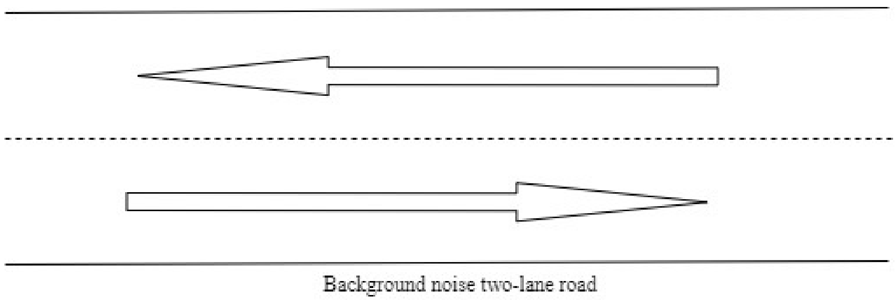
Background traffic noise is considered to be at 7.5 meters distance from the listener. This distance was chosen because typically the sounds of the approaching vehicles relies in DIN ISO 362 [29]. Chattering noise is considered to be all around the listener.

Selected EV Sounds

We have to test 3 EVSs in every scenario, as described above. We decided to use 3 different EVSs in order to highlight how their features are affecting on detectability:

1. Source A, fig 6.4: it could be described as a *futuristic* sound. It presents a loud frequency bands between 100 and 300 Hz and logarithmically decreasing spectrum until 2048 Hz. We have two frequency spikes at 800Hz and 3800 Hz.
2. Source B, fig 6.5: it could be described as a *pulsing* sound. Its spectrogram is extremely well distributed, we can clearly see harmonics: the fundamental harmonic is around 60 Hz. This sound presents periodic behaviour and some slightly beating effect.
3. Source C, fig 6.6: it could be described as a *noisy* sound. This sound is composed by a background noise component and a limited section of characterising frequency, between 128 and 1100 Hz.

In section 5.2 we reported some important regulations and results necessary for deciding loudness: following those guidelines we will adopt a speed-wise approach:



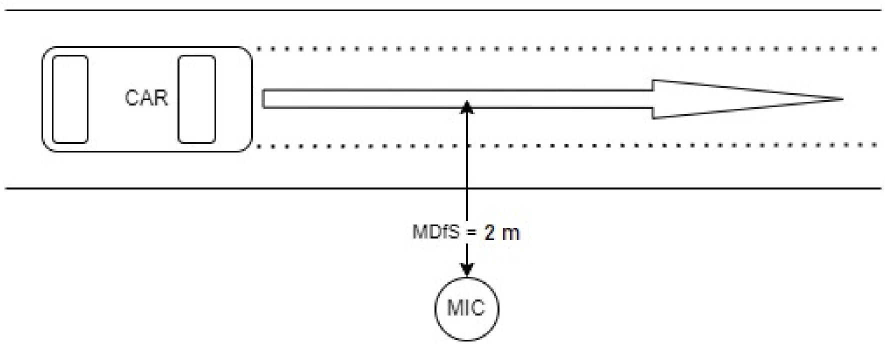


Figure 6.3: Experiment when car moving from left to right

- At 20 km/h loudness is 57 dbA
- At 30 km/h loudness is 62 dbA
- At 50 km/h loudness is 71 dbA

Loudness measurement microphone is considered at 2 meters distance from source.

It's very important to highlight that the sound produced by the SC tyres rolling on the road is not included. This choice was justified by the fact that in this experiment we want to analyse only the detectability properties of the synthesized sounds. In fact according to Vega et al [5] we know that after 20 mph tyre rolling sound tends to cover car emitted sounds.

Background noise

Background noise loudness is a key factor to obtain a realistic simulation and is the main obstacle for EVS designers. Many different choices can be made when chosing a background noise:

- What type of noise should be picked (white noise, traffic noise, chattering noise)?
- Where sould the noise come from when implementing a VE?

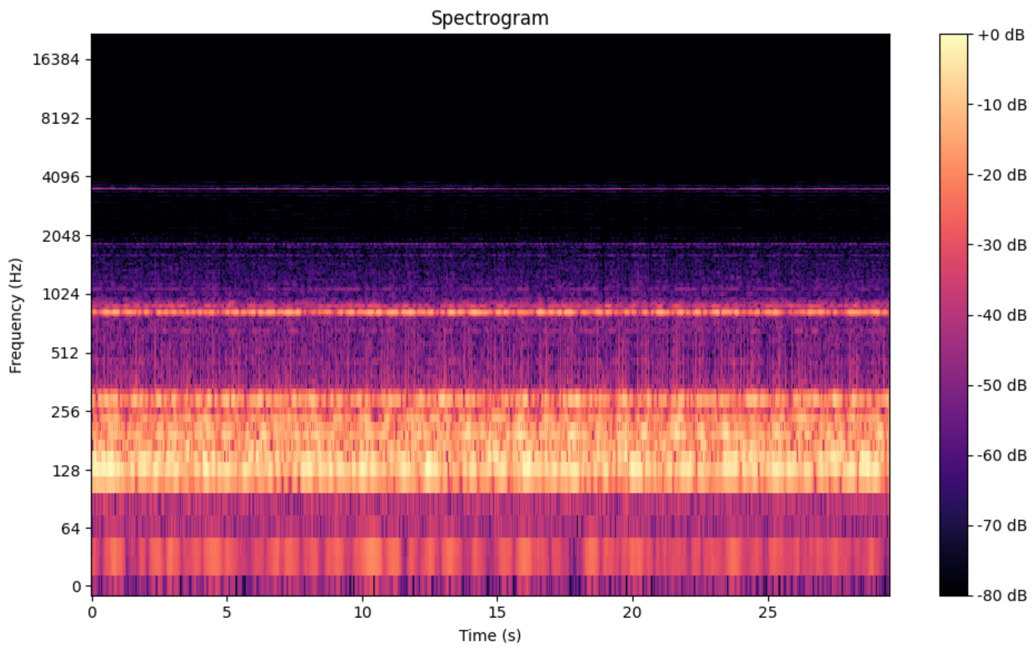


Figure 6.4: Source A spectrogram

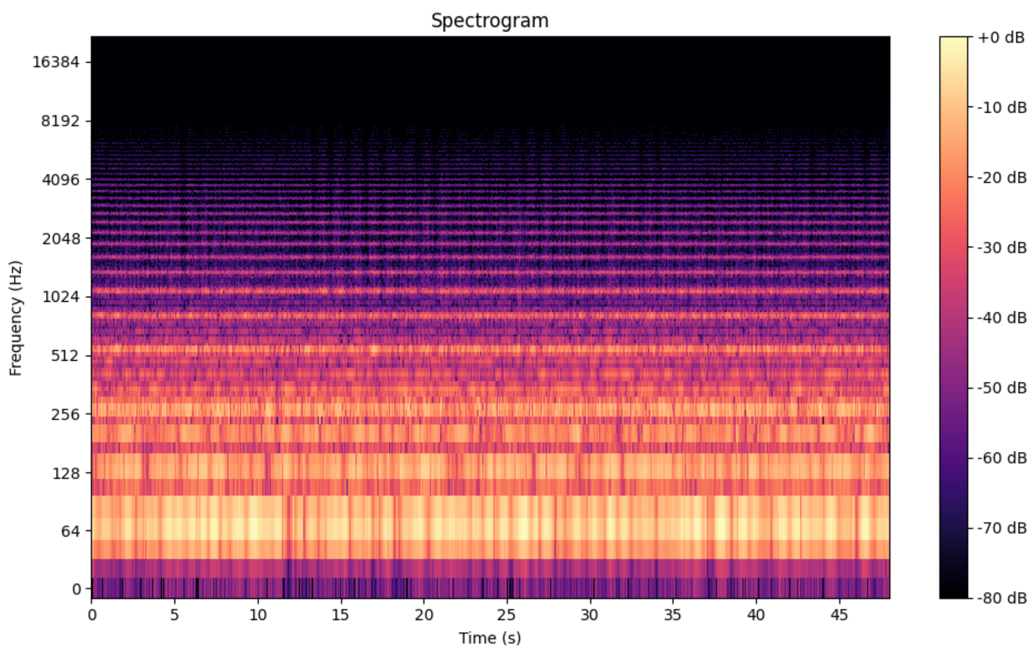


Figure 6.5: Source B spectrogram

- How loud should it be?

Many previous studies do not answer these questions and provide little information about background noise. In our work we want to propose a model that is clearly described and easily

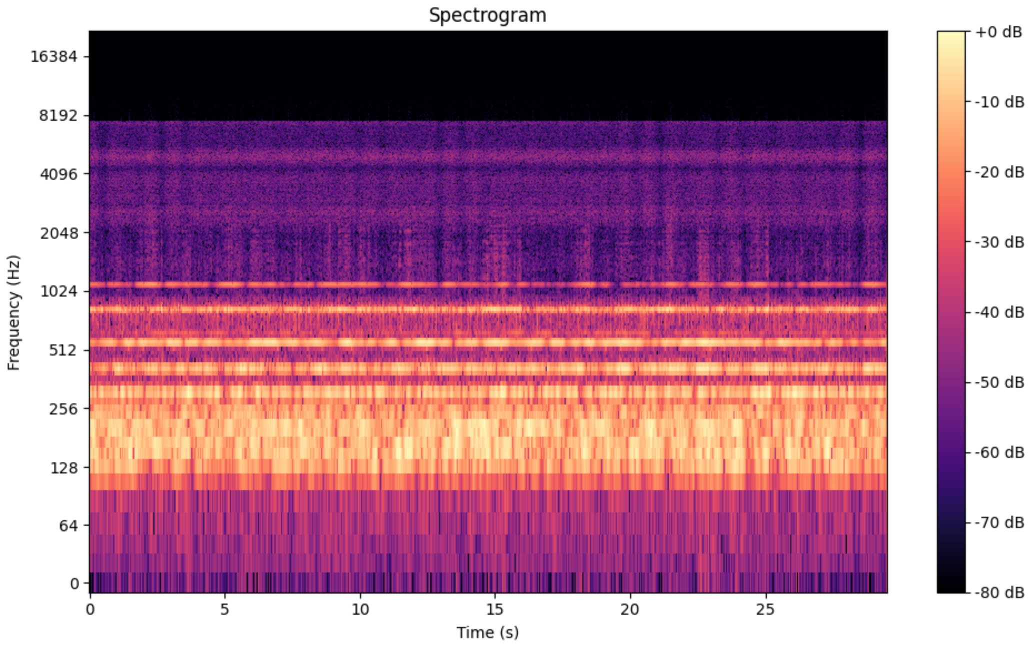


Figure 6.6: Source C spectrogram

reusable.

Noise type

We considered two scenarios:

- urban traffic noise without human and animal generated sounds
- chattering noise, sidewalk noise with many people walking and talking

The background recording were selected so that they are homogeneous in loudness and without abrupt additional sounds (sirens, screaming, etc). Both the presented scenarios are analyzed in wet and dry condition, as we will see in the next section rain has a key role in urban traffic noise. Wet noise will include the tickling sound of rain on concrete at low Sound Pressure Level (SPL) - the SNR between the source maximum loudness and the rain it's 15 dBA.

Finally we report that since the simulator offers the possibility to tune temperature, pressure and relative humidity, we set them as follows:

- Dry condition: $T = 20°, p = 1$ atm, $h = 50\%$
- Wet condition: $T = 20°, p = 1$ atm, $h = 100\%$

In real world background noise is generated by multiple sources (in our case multiple vehicles) at different distances from the listeners. The sound coming from those sources could be directly perceived and or reflected by many surfaces before reaching the listener. Since we are not able to reproduce complex scenarios we will model our background noise source as a distant road from the listener, see fig 6.3.

To establish the loudness level of the background noise we rely on scientific papers rather than a specific in loco measurement. Kerber and Fastl [6] used a speech-like noise recorded at Munich, Marienplatz at an overall A-weighted level of 62 dBA. Menzel, Yamauchi et al [7] used four environmental background sounds in Fukuoka, Japan: a two-lane busy street in down town 65.5 dbA, a two-lane road in a residential area 67.5 dbA, six-lane heavy traffic 73 dbA and a narrow road in a shopping area 61 dbA (dbA levels are approximated). In Misdariis et al [25] three samples recorded in Paris were selected, categorized as *residential* 65.1 dbA, *busy* 71.4 dbA and *shopping* 56.1 dbA.

Cai et al [27] paper instead measured the differences in A-weighted loudness levels of traffic noise between dry and wet conditions. They stated that, the mean difference in the sound pressure level between the wet and dry asphalt roads for light, middle-size, and heavy vehicles are 10.09 dBA, 5.56 dBA, and 4.26 dBA, respectively. This values are highly dependant on asphalt type, water absorption and water amount, it is plausible that that they may vary as reported by some other papers.

Finally, we want to recall that speed is the main parameter to determine traffic noise SPL, faster traffic flow is louder. Different models have been proposed: FHWA Traffic Noise Model [30] presents a nearly linear correlation, while in JTG B03-2006 [31] model loudness increases logarithmically with respect to speed.

In our experiments we will use a FHWA based noise level prediction. To predict the noise we will assume the following conditions:

- busy road: multiple cars passing each minute
- light vehicle traffic: medium size and heavy duty vehicles are excluded
- speed dependant simulation
- dry weather

Resulting noise levels for 20, 30 and 50 km/h are 58.0, 62.9 and 69.2 dbA respectively.

To obtain wet traffic noise we start from the dry noise and we perform a boost of the high frequencies (especially around 2.5 kHz) and a reduction of low frequencies as reported in

Cai, see fig 6.7 and 6.8. We will use these figures to equalize correctly our wet traffic noise. According to various papers wet traffic noise is louder and various models have been proposed: for consistency reasons we will rely on Cai study and we will adjust SPL as follows:

- For 20 km/h we want a final increase of 7 dbA, to reach a total of 65.0 dbA
- For 30 km/h we want a final increase of 5 dbA, to reach a total of 67.9 dbA
- For 50 km/h we want a final increase of 3 dbA, to reach a total of 72.2 dbA

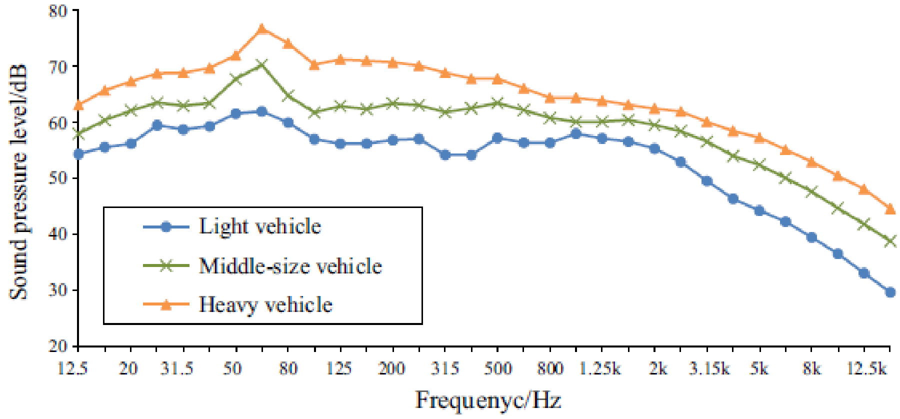


Figure 6.7: Noise spectrum for light, middle-size and heavy vehicles on a dry asphalt road, image from [27]

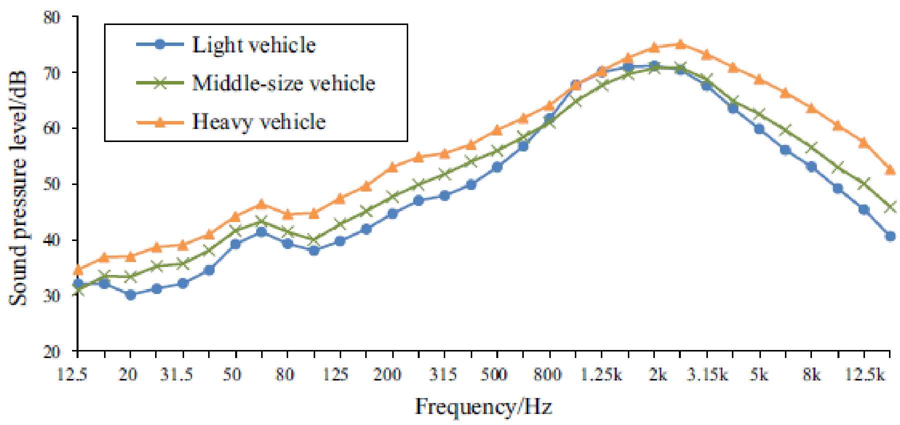


Figure 6.8: Noise spectrum for light, middle-size and heavy vehicles on a wet asphalt road, image from [27]

CHATTERING NOISE

The adopted chattering noise will be the same used by Misdariis et al [25], which was labeled as *shopping area*. Originally it was played at 56.1 dbA, but we will increase the SPL to 61 dbA, to achieve comparable results to Kerber and Fastl [6] and Menzel, Yamauchi et al [7] who both used speech-like noise.

6.3.2 Validation procedure

To present the result we will follow a precise validation procedure:

1. Remove data outliers given by distractions or misunderstandings of the listeners.
2. Plot the *dd* compared to the safe distance for each case of the simulation.
3. Analysis of the numerical results.
4. Recap and comparison of different tested sounds, draw final conclusion.

Removing data outliers is a critical procedure because we have to detect when the user is committing some mistake. A pre-test stage for our experiment was required to evaluate the most common mistakes of the listeners. We concluded that many people were detecting car way before they were audible. We assumed this behaviour was caused by suggestion. To solve this problem we allowed following listeners to hear some samples and take a couple of free tries before starting, same procedure reported in 6.3.1. After that, results were more consistent especially for the first audio simulations because participants were more confident.

Nevertheless testers were committing few distraction errors, for example pressing the stop button too late due to loss of concentration. For this reason we monitored each participants recording when they committed errors and successively removing those samples from data.

We collected 732 samples and 65 of them resulted not valid, for a total of 667 valid samples.

6.3.3 Results

Data analysis was performed using python and seaborn library for visualization. Data plots are grouped for source type.

Detection distance

Our first set of plot are depicting *dd* with respect to speed, weather and background type. Two lines for every speed represent the safety distance needed for focused (Green) and distracted driver (Red) to stop the car. Source A, B and C plots are respectively in figure 6.9. To have an estimation of a probability density function described by the samples we are exploiting seaborn catplot capabilities, figures are printed in the next sections.

For ease of notation we will denote 3 regions in figures 6.9, 6.11, 6.13:

- *Safe region*: above the red line, pedestrian is safe even if the driver is distracted
- *Intermediate region*: between red and green line, pedestrian safety depends on drivers attention
- *Danger region*: below the green line, driver has no time to stop in case pedestrian steps on the road.

Before commenting results for each source and condition we want to consider the general picture. Our ideal goal would be to have all detection points above the red line, but this is not realistic - also proved by [6]. Having all samples above the green line would be a satisfying outcome. Moreover, density of samples is another important aspect; high density of samples means that most of the listeners hear the sound at the same distance, while sparse samples indicates that the sound is perceived differently.

We clearly see that different sources have very different graphs, but they all follow a similar pattern: *dd* tends to increase with speed. Furthermore, detection within safe range increasingly gets worse with increasing speed.

Another interesting trend is that *dd*s for 50 km/h car speed are more sparse. This is justified by the increasing incidence of participant reaction times w.r.t. speed; we can easily induce that half a second difference in reaction time it's more critical when the incoming car is faster because it travels more meters than a slower car.

SOURCE A

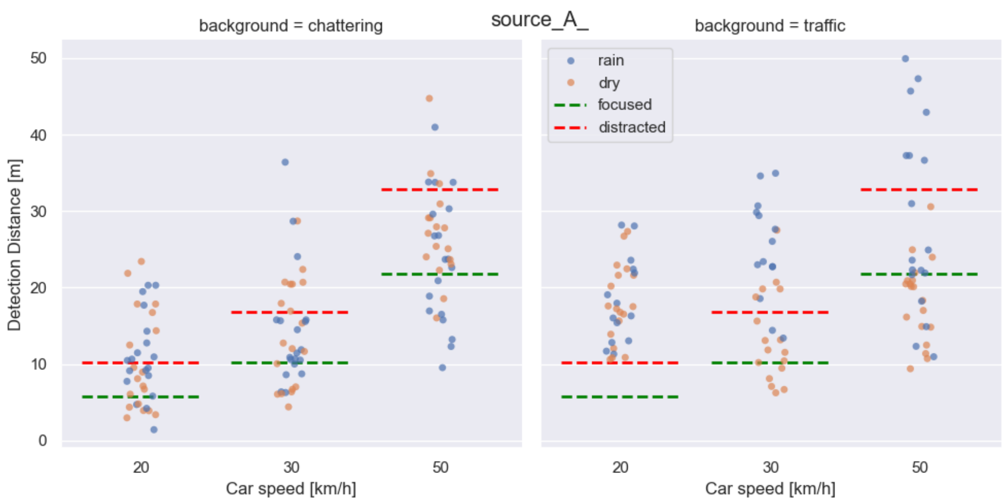


Figure 6.9: Source A - detection distance results

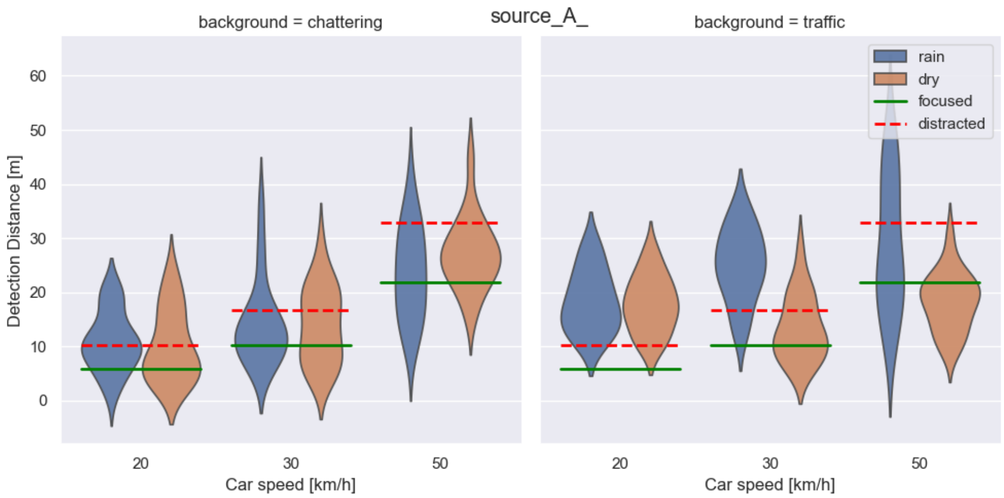


Figure 6.10: Source A - detection distance p.d.f.

In 6.9 single experiment samples are plotted, in 6.10 the estimated p.d.f. are shown. Under chattering background noise *dd* points are dense and centered around the *Intermediate region*, even for 50 km/h speed. There are no main differences between dry and wet condition.

Under traffic background noise *dd* points are distributed differently:

- for 20 km/h points are dense and mostly in the *Safe region*
- for 30 km/h points are sparse, but few of them in the *Danger region*.
- for 50 km/h points are sparse and mainly in the *Danger region* for dry condition.

Rain and dry samples are similarly distributed only for 20 km/h. Overall rainy samples are more detectable in traffic noise.

The loud frequency band between 100 and 300 Hz and the absence of frequencies above 2048 Hz, proved to give no particular advantages w.r.t. weather conditions. Furthermore, in high speed traffic scenarios presents very sparse distribution of samples, proving that this sound isn't suitable for intense urban noise.

SOURCE B

In 6.11 single experiment samples are plotted, in 6.12 the estimated p.d.f. are shown. Under chattering background noise points are extremely dense and centered on the red line - inbetween the *Intermediate* and *Safe regions*; very few samples are in the *Danger region*. Weather conditions are not influencing samples distributions

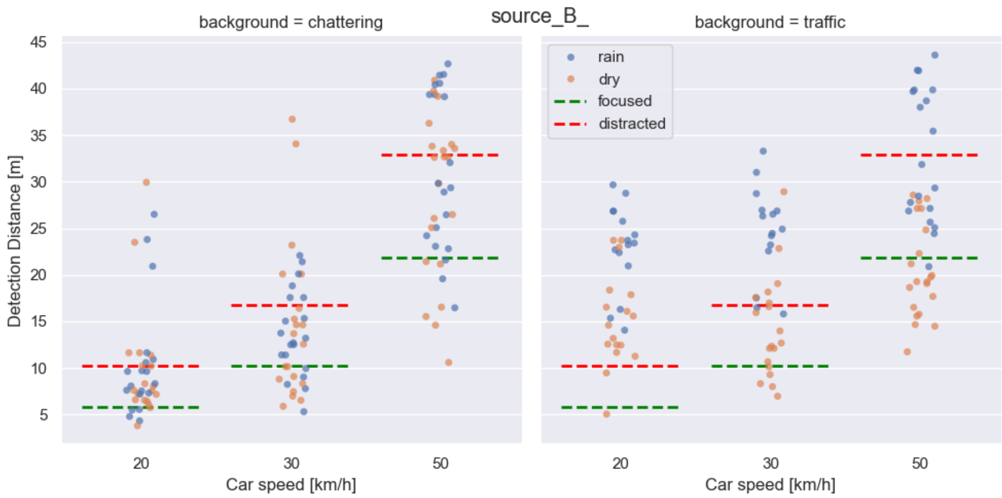


Figure 6.11: Source B - detection distance results

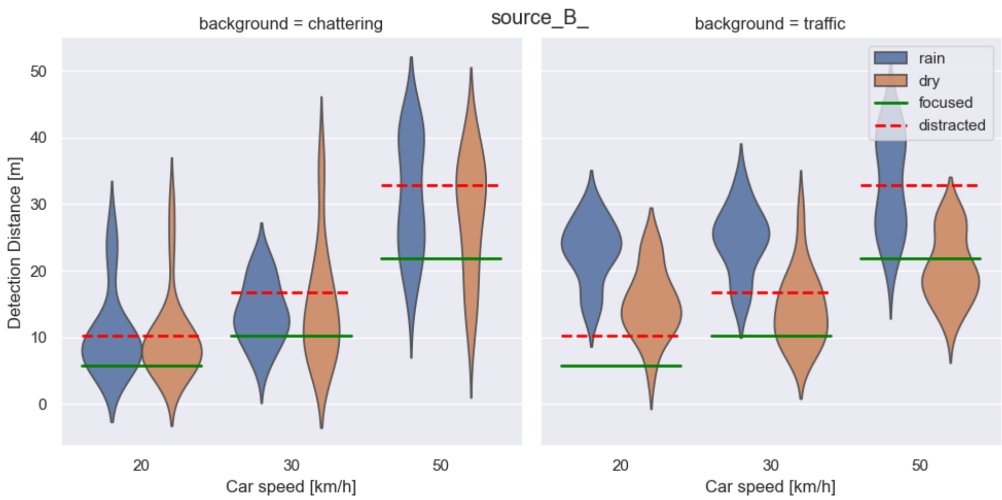


Figure 6.12: Source B - detection distance p.d.f.

Under traffic background noise samples are distributed differently w.r.t. weather conditions. Wet scenarios samples are dense and mostly in the *Safe region*, while dry samples are still dense but centered around *dd* of 17 meters. This means that dry p.d.f. is acceptable only for 20 and 30 km/h.

The presence of loud frequency band centered around 60 Hz and of an evenly distributed harmonics all over the spectrum proved to be extremely beneficial for detection in rainy traffic environment. In fact the lack of low and mid frequencies in the spectrum of a wet asphalt road sound - see fig 6.8 - allows the source B low frequency to stand out. Traffic dry scenario is still critical since peaks of noise around 60 Hz are masking the sound of the incoming car, leaving no time to react for high speeds.

SOURCE C

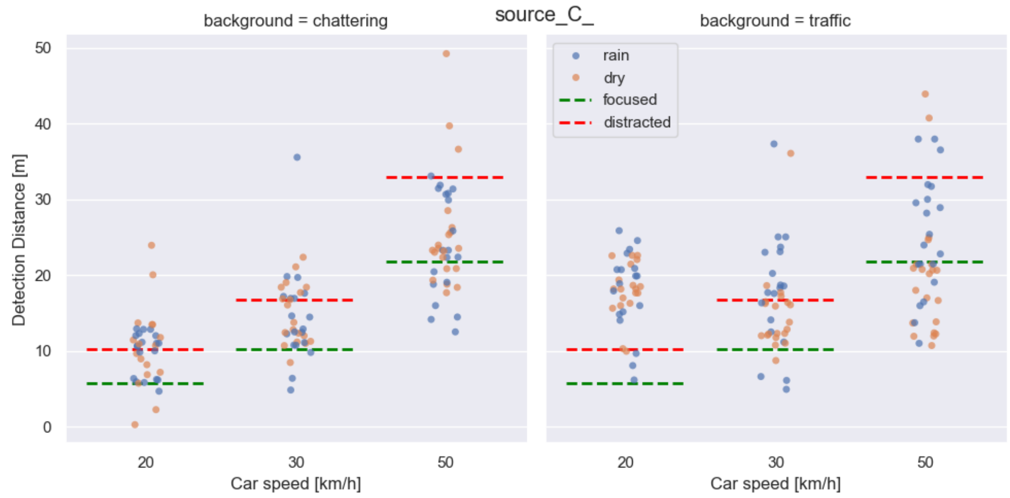


Figure 6.13: Source C - detection distance results

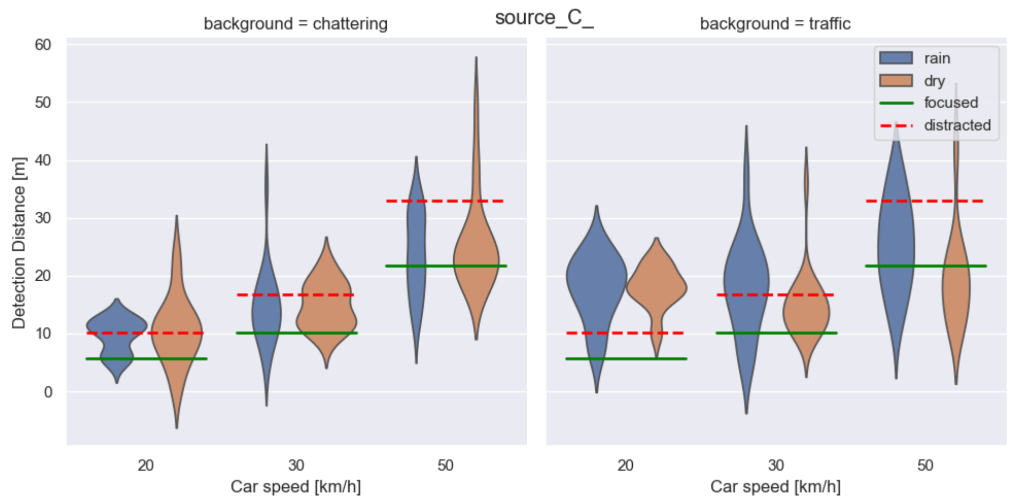


Figure 6.14: Source C - detection distance p.d.f.

In 6.13 single experiment samples are plotted, in 6.14 the estimated p.d.f. are shown. Source C presents extremely similar results to source B when adopting a chattering background noise. Although, when placed in a traffic environment it returns better results. P.d.f for wet and dry conditions at 20 and 30 km/h are very similar, on the contrary we saw that for source B it was not the case. For 50 km/h, dry condition seems to be detrimental for detectability. Overall we state that weather condition is less influential than in source A and B.

The loud frequency band between 128 and 1100 Hz and the noisy texture of the sounds are very distinguishable in every context except when traffic is becoming very loud. Since fast rolling traffic in dry conditions has a wide spectrum, noisy textures are masked and the loud frequency band alone is not powerful enough to make the sound perceivable.

Time to arrival

Time to arrival - *tta* - is the time needed for the car to arrive in front of the pedestrian location while maintaining its speed constant. This measure is very important since it defines how much time we do have to perform life-saving actions. Usually *tta* decreases as the incoming car speed increases, hence the ideal goal would be to have high *tta* even for high speeds.

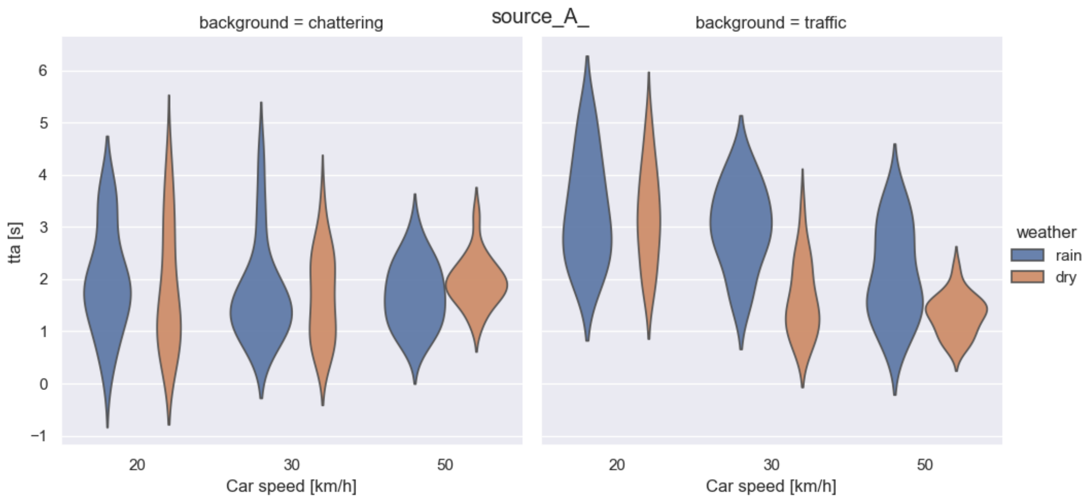


Figure 6.15: Source A - time-to-arrival p.d.f.

From *tta* analysis we expect extremely similar distributions to detection distance Observing fig 6.15, fig 6.16 and fig 6.17 and comparing them to previous plots we notice that our prediction is correct: distributions have the same shape but now the are centered differently. In the last section we noticed how under chattering noise all sources behaved similarly, so it was difficult to compare them; instead using *tta* we can extract more visual informations. We would like that *tta* to have a low variance distributions and to increase with speed. Looking at source C 6.17, we clearly see that neither of these conditions are satisfied. With the same approach we notice that source A 6.15 has high variance distributions for 20 and 30 km/h. Finally, considering source B 6.16, we see that *tta* for each speed has a low-variance Gaussian-like distribution. We conclude that source B is the most suitable for chattering noise scenarios.

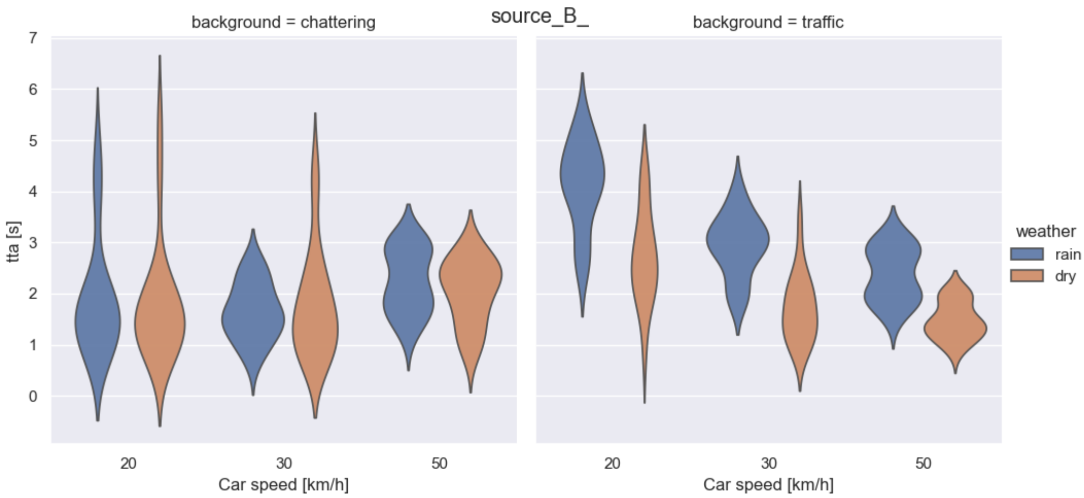


Figure 6.16: Source B - time-to-arrival p.d.f.

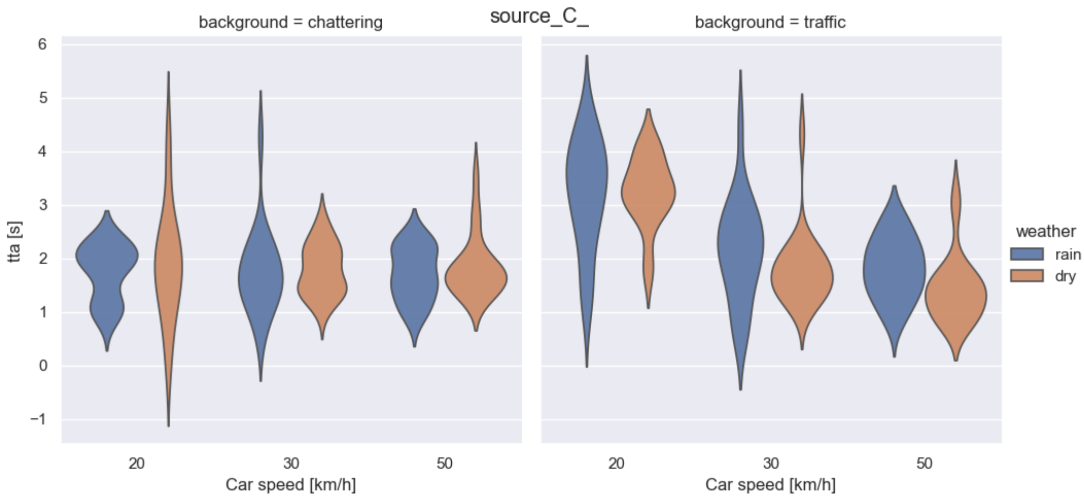


Figure 6.17: Source C - time-to-arrival p.d.f.

ANOVA analysis

Analysis of variance (ANOVA) is a statistical test used to assess the difference between the means of more than two groups. Analysts use the ANOVA test to determine the influence of independent variables on the dependent variable in a regression study. In our analysis the dependent variable will be *dd*.

Let's recall that ANOVA tests assume that the data is normally distributed and that variance levels in each group are roughly equal. To verify that distributions are aren't diverging from normality we ran a shapiro test for each possible combination of parameters, for a total of 36. Only in few cases we reported divergence from normality:

- source A - chattering - dry - 20 km/h where $pvalue = 0.008$

- source B - chattering - dry - 20 and 30 km/h, respectively $pvalue = 0.008$ and $pvalue = 0.015$

- source B - chattering - rain - 20 km/h where $pvalue = 0.002$

- source C - traffic - dry - 50 km/h where $pvalue = 0.006$

- source C - traffic - rain - 50 km/h where $pvalue = 0.001$

If your dependent variable is not normally distributed, you may be increasing your chance of a false positive result. ANOVA is not very sensitive to moderate deviations from normality; simulation studies, using a variety of non-normal distributions, have shown that the false positive rate is not affected very much by this violation of the assumption. Furthermore we observe that non-normality in our case is very likely to be generated by lack of samples; collecting more of them would probably solve the problem of non-divergence.

Since above mentioned distribution show slightly variations from normality and most of the distribution could assumed to be normal, we can proceed with our ANOVA analysis.

Sex and Age

Firstly, we perform two one-way ANOVA test using as independent variables respectively, Age and Sex of participants. In the first case we obtained an $F - value = 2.4444$ with $p = 0.1111$, while in the second case an $F - value = 0.666$ with $p = 0.5555$. We can easily concluded that there exists no correlation between *dd* and Sex/Age. See tab 6.1.

	F-value	P-value	Correlation
Sex	2.4	0.1	No
Age	0.6	0.5	No

Table 6.1: Sex and Age - ANOVA

Weather condition

To analyse the dependence between weather conditions and *dd* we firstly fixed one background. It resulted that for chattering background $F - value = 0.300982$ with $p = 0.583703$ while for traffic background $F - value = 54.53713$ with $p < 0.001$. ANOVA analysis confirms our previous observations: weather condition is highly influential in traffic environment, on the contrary it basically makes no difference in the chattering scenarios. See tab 6.2.

	F-value	P-value	Weather Correlation
chattering	0.3	0.58	No
traffic	54.5	< 0.001	Yes

Table 6.2: Weather Condition - ANOVA

Background

Using the background type as independent variable we fix one weather condition. In dry scenario we get $F - value = 0.019697$ with $p = 0.888486$, while in wet scenario $F - value = 45.906378$ with $p < 0.001$. This means that background type is highly influential only in case of no rain. See tab 6.3.

	F-value	P-value	Background Correlation
rain	0.02	0.88	No
dry	45.9	< 0.001	Yes

Table 6.3: Background noise - ANOVA

6.3.4 Source

We want to analyse how much *dd* depends on source type. To do so, we fix a background noise and we perform ANOVA test, using source type as independent variable. Fixing chattering noise we obtain $F - value = 0.310437$ with $p = 0.733363$, while fixing traffic noise $F - value = 3.200853$ with $p < 0.05$. We can conclude that there is a dependence only for traffic background. This is confirmed by the plots: with chattering noise the distribution of samples are similar for every source type, while in traffic environment clear differences among source type are showing up. See tab 6.4.

	F-value	P-value	Source Correlation
chattering	0.31	0.73	No
traffic	3.2	< 0.05	Yes

Table 6.4: Source - ANOVA

6.3.5 Conclusions

From this comprehensive experiment we can draw important conclusions on EVS design.

First of all, it's important to notice that no previous study has combined different sound, noises and weather conditions, defining a framework to evaluate EVSs. We will now discuss how improving the experiment with respect to previous papers [7], [6], [25] allowed us to contribute to the research.

In particular the Kerber et al [6] experiment used few static background noises, we recall that in our work we modeled a speed dependant loudness traffic background which increases the realism of this experience. We adapted our sources loudness equally w.r.t. background noise and we noticed that different sources with the same loudness gave extremely different results. We proved even using speed adaptive source loudness in a realistic traffic environment might not be enough to obtain a clearly detectable sound, expanding Menzel [7] and Kerber conclusions [6]. This demonstrates that tuning source loudness and modeling the source spectrum are two complementary and necessary design techniques. This result is important since few papers have tried to combine safety aspects of EVS with a comprehensive analysis of sound and background spectrum and loudness level.

By comparing results for two different weather conditions we highlighted how drastically different they can be, drawing attention on how important this comparison is when design the sound. Starting from Cai et al [27] paper we produced a realistic background traffic for both dry and wet conditions, then we proved that in dry conditions is more difficult to detect sounds with poor energy spectrum for high frequencies. On the contrary we tested that sounds with high energy low-mid frequencies are particularly suitable for rainy environments. This observation lead us to a more spectrum-based approach in EV sound design, providing a possible framework to address the problems presented in [7] and [25].

Finally we presented a new evaluation method that relies on time-to-arrival as metric instead of detection distance. We saw that, even though some detection distance plots were poorly informative, by comparing time-to-arrival probability density functions we could chose the most suitable source for chattering noise scenarios.

7
Conclusion

At the end of this dissertation we want to briefly recap the content, recalling how this document contributes to the EV sound design problem.

In the first part we explained why silent EVs represent a danger for pedestrians, considering their increasing sales number. To address this challenge we designed and implemented the AGS, providing an innovative tool which exploits the capabilities of granular synthesis to create EVSs.

Starting from previous research and regulations ([6], [7], [25], [24]), we provided a detailed set of metrics to guide the EV sound design engineer; in particular we focused on safety oriented sound design.

The last part of this thesis was dedicated to design an experiment to validate EVSs. By analysing [6] [7] [25] [26], we observed that those experiment where lacking of a complex simulation environment and a deep study of background noises and weather conditions, moreover they only use detection distance as reference metric. Furthermore article [25] faced the problem of comparing different EVSs but it never mentions the correlations between source and background noise spectrum. Finally we observed that all articles presented only a partial ANOVA analysis when trying to explain any kind of correlations between source and noise. To solve those issues we designed a comprehensive experiment and a validation procedure that provides a framework to test EVSs and with the help of 24 participants we tested it. It turned out that:

- Tuning source loudness and modeling the source spectrum are two complementary and necessary design techniques.
- Is fundamental to study urban traffic at different loudness level since results will show different trends w.r.t. static loudness noise.

- In presence of traffic urban noise, in dry conditions is more difficult to detect sounds with poor energy spectrum for high frequencies.
- Analysing the scenarios with different weather conditions is essential because pedestrian perception can drastically change.
- Sometimes detection distance as a metric is poorly informative, so time-to-arrival probability density functions are more suitable to choose the best source sound.

Future work

EV sound design still has to make big steps forward and starting from the experience gained with this thesis we can set the path.

In our work we limited our analysis to few source and environmental scenarios; increasing the number of weather conditions, background noises and sources types could bring to new conclusions. Furthermore, since we only considered a simple pass by scenario where the pedestrian is standing still, implementing new pedestrian-car types of interaction and compare them to our results would be interesting.

Moreover, designing a comprehensive study using both safety related metrics and psychoacoustic ones and trying to find correlations between them, could lead to groundbreaking results. In this work the analysis of the sources spectrograms and their correlations w.r.t. to final results were very concise, we hope that in the future some more advanced studies (hopefully integrating updated noise emission regulations) will be conducted.

Finally we want to stress that our simulation does not replace real world test, which are much more complex due to the presence of many uncontrollable parameters. Real world environment trying to emulate our experimental conditions would not only be necessary to test the EVS itself but also to validate our experimental procedure and see if it provides a good approximation of a real environment.

References

[1] G. Povegliano, *Automotive Granular Synthesizer (AGS) for Electric Vehicles: Development and Affective Impact Analysis*. University of Padova, CSC, 2024.

[2] S. Canazza and G. D. Poli, "Four Decades of Music Research, Creation, and Education at Padua's Centro di Sonologia Computazionale," *Computer Music Journal*, vol. 43, no. 4, pp. 58–80, 12 2019. [Online]. Available: https://doi.org/10.1162/comj_a_00537

[3] S. Canazza, G. De Poli, and A. Vidolin, "Gesture, music and computer: The centro di sonologia computazionale at padova university, a 50-year history," *Sensors*, vol. 22, no. 9, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/9/3465

[4] IEA, "Global ev outlook 2024, iea," 2024. [Online]. Available: https://www.iea.org/reports/global-ev-outlook-2024

[5] Garay-Vega, Hastings *et al.*, "Quieter cars and the safety of blind pedestrians: Phase 1." 2010.

[6] S. Kerber and H. Fastl, "Perceptibility of approaching vehicles in urban background noise," in *Proc. of 36th Intern. Congress on Noise Control Engineering INTER-NOISE 2007, Istanbul, Turkey*, 2007.

[7] Y. Menzel *et al.*, "Psychoacoustic experiments on feasible sound levels of possible warning signals for quiet vehicles," 2011.

[8] Parizet, Robart *et al.*, "Detectability and annoyance of warning sounds for electric vehicles," *Proceedings of Meetings on Acoustics*, vol. 19, 06 2013.

[9] M. Wogalter, R. Ornan, R. Lim, and R. Chipley, "On the risk of quiet vehicles to pedestrians and drivers," 2001.

[10] P. Nyeste and M. Wogalter, "On adding sound to quiet vehicles," 2008.

[11] B. June and K. Jangyoung, "Engine sound design for electric vehicle through wavetable software synthesizer," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 22 p ±1644, p. 1639, 2018.

[12] J.-F. Petiot, K. Legeay, and M. Lagrange, "Optimization of the sound of electric vehicles according to unpleasantness and detectability," *Proceedings of the Design Society: International Conference on Engineering Design*, vol. 1, no. 1, p. 3949–3958, 2019.

[13] M. J. Lazaro, S. Kim, M. Choi, K. Kim, D. Park, S. Moon, and M. Yun, "Design and evaluation of electric vehicle sound using granular synthesis," *Journal of the Audio Engineering Society*, vol. 70, pp. 294–304, 05 2022.

[14] G. D. Giorgi, *Design and prototyping of an Advanced Acoustic Vehicle Alerting System for Electric Vehicles*. Massimiliano Curti - Politecnico di Torino: Rel. Marco Carlo Masoero, 2021.

[15] B. Truax, "Real-time granular synthesis with a digital signal processor," *Computer Music Journal*, vol. 12, no. 2, pp. 14–26, 1988. [Online]. Available: http://www.jstor.org/stable/3679938

[16] G. D., "Acoustical quanta and the theory of hearing." *Nature*, pp. 159, 591–594, 1947.

[17] R. C., *Microsound*, 2004.

[18] S. F. University. [Online]. Available: https://www.sfu.ca/~truax/gran.html#:~:text=Granular%20synthesis%20was%20first%20suggested,range%20of%2010%2D30%20ms.

[19] M. Hascoskan, *CAN Simulation of Electric Vehicles for designing sound associated with sustainable mobility*. University of Padova, CSC, 2024.

[20] W. A. Yost, "Overview: Psychoacoustics," in *Human Psychophysics*, ser. Springer Handbook of Auditory Research, W. A. Yost, A. N. Popper, and R. R. Fay, Eds. New York, NY: Springer, 1993, vol. 3, pp. 1–12.

[21] R. H. Lyon, "Product sound quality - from perception to design," *Sound and Vibration*, vol. 37, pp. 18–22, 3 2003.

[22] P. Jennings, G. Dunne, R. Williams, and S. Giudice, "Tools and techniques for understanding the fundamentals of automotive sound quality," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 1, pp. 1–16, 10 2010.

[23] M. Green, "How long does it take to stop?methodological analysis of driver perception-brake times," *Transportation human factors*, vol. 2, no. 3, pp. 195–216, 2000.

[24] UN, "Concerning the adoption of harmonized technical united nations regulations for wheeled vehicles, equipment and parts which can be fitted and/or be used on wheeled vehicles and the conditions for reciprocal recognition of approvals granted on the basis of these united nations regulations," *Regulation No. 138*, vol. Addendum 137.

[25] N. Misdariis, A. Gruson, and P. Susini, "Detectability study of warning signals in urban background noises: A first step for designing the sound of electric vehicles," *Proceedings of Meetings on Acoustics*, vol. 19, no. 1, p. 040032, 05 2013. [Online]. Available: https://doi.org/10.1121/1.4799454

[26] S. Singh, *Understanding and Improving Methods for Exterior Sound Quality Evaluation of Hybrid and Electric Vehicles*. University of Warwick, 2016.

[27] M. Cai, S. Zhong, H. Wang, Y. Chen, and W. Zeng, "Study of the traffic noise source intensity emission model and the frequency characteristics for a wet asphalt road," *Applied Acoustics*, vol. 123, pp. 55–63, 2017.

[28] S. Damiano and T. van Waterschoot, "Pyroadacoustics: a road acoustics simulator based on variable length delay lines," in *Proc. 25th Int. Conf. Digital Audio Effects (DAFx20in22)*, 2022, pp. 216–223.

[29] I. 362-1:2022, "Acoustics — engineering method for measurement of noise emitted by accelerating road vehicles — part 1: M and n categorie."

[30] C. W. Menge, C. F. Rossano, G. S. Anderson, C. J. Bajdek *et al.*, "Fhwa traffic noise model, version 1.0 technical manual," United States. Federal Highway Administration. Office of Environment, Tech. Rep., 1998.

[31] C. Ministry of Transportation, "Jtg b03-2006," 2006. [Online]. Available: https://www.chinesestandard.net/PDF/English.aspx/JTGB03-2006

Acknowledgments

Arriviamo dunque al paragrafo di più difficile stesura. Nel continuo turbinio della vita da studente fatta di esami, progetti, incontri, feste è difficile fermarsi per fare un bilancio, se non quando ci viene imposto. Dunque, quì ed ora scrivo le mie riflessioni.

Una persona al tavolo circondata da familiari ed amici è come se si guardasse allo specchio, e così ora mi trovo. Sono debitore a tutti perché ognuno mi ha offerto qualcosa che io conservo e ho reso parte di me. Non so se sono stato o sarò in grado di sdebitarmi, però intanto offro a tutti un paio di calici.

Nelle Marche c'è un detto, "le cerque non fa li melaranci" (ovvero "le querce non fanno le arance") che ci ricorda simpaticamente che la radice della nostra esistenza è sempre la famiglia; sono molto contento che questo detto sia vero perché sono fiero di essere parte della mia famiglia. Ringrazio voi per avermi sostenuto durante tutti i miei studi con i vostri sacrifici. Ancora di più ringrazio i miei genitori che mi hanno trasmesso i valori più preziosi e una buona dose di ironia. Questa soddisfazione la devo principalmente a loro che mi hanno saputo indirizzare fin da quando ero piccolo e che mi hanno dato tutto il supporto per arrivare fin quì. Grazie.

In questi 24 anni ho avuto la fortuna di aver sempre affianco i miei amici a Macerata dalle giornate passate al campo dei pini a scappare dal guardiano a 15 anni, ai ritrovi estivi concessi dalle pause universitarie. Per poi ritrovare inaspettatamente l'amico di una vita dopo 23 anni come coinquilino. Ne abbiamo fatta di strada! Un saggio proverbio non a caso recita "se trovi un amico nuovo non dimenticar l'antico". Grazie anche a voi.

E poi sono partito. A Padova ho trovato la mia seconda casa. Riordinare in testa tutti i bei ricordi è difficile. È facile dimenticarsi di un evento, ma è quasi impossibile scordarsi come qualcuno ti ha fatto sentire e in particolare voi, i miei amici. Quelli che mi hanno accolto subito come fossi un fratello, che mi hanno insegnato a vivere serenamente, che mi hanno trascinato in esperienze uniche, che mi sanno prendere in giro, che sanno ravvivare anche una qualsiasi giornata di studio, che mi hanno spalleggiato in manifestazioni, concerti e biblioteche. Gli amici ti ricordano chi sei e io voglio essere un po' come tutti voi.

Infine per citare chi di ringraziamenti se ne intende "Vi ringrazio e vi auguro cento anni di felicità, sempre in buona salute e come il vostro cuore desidera!".