



# Atmospheric Ray Tracing: An efficient, open-source framework for finding eigenrays in a stratified, moving medium

Philipp Schäfer\* and Michael Vorländer

Institute for Hearing Technology and Acoustics, RWTH Aachen University, Kopernikusstraße 5, 52074 Aachen, Germany

Received 18 March 2021, Accepted 12 May 2021

**Abstract** – In this paper, an open-source framework for ray tracing in a stratified moving medium is introduced. This framework provides an efficient method to find eigenrays connecting a source with a receiver and is designed for the purpose of aircraft noise auralization. The method is tested with respect to accuracy and run-time in an aircraft flyover scenario and compared to a state of the art method. The investigation showed that this method provides eigenrays with preset accuracy for source positions most relevant for flyover scenarios and that it is significantly faster than the state of the art method. According to the performance analysis, the presented approach has great potential for integration into future real-time auralizations of aircraft noise.

**Keywords:** Ray Tracing, Stratified, Moving medium, Aircraft noise, Auralization, Open-source

## 1 Introduction

Aircraft are one of the major sound sources with respect to noise pollution [1]. Especially residents living close to airports can suffer from aircraft noise. Noise recommendations are typically based on time-averaged sound level metrics like  $L_{\text{den}}$  or  $L_{\text{night}}$  [1], which account for human perception just in a very basic manner. For example, it was shown that a reduced noise level does not necessarily lead to a reduction of noise annoyance [2]. Thus, development has been undertaken to enhance aircraft design using perception-based methods [2–4]. Through the concept of auralization, different aircraft designs can be evaluated and compared, e.g. by presenting synthesized aircraft noise in listening tests. The same approach can be used to communicate these design changes to residents [5]. In both cases, real-time auralization embedded in auditory-visual virtual reality technology can advantageously enable quasi-instantly changes to the sound propagation conditions. For example, the user could flip the wind direction or change between daytime and nighttime weather conditions. In contrast to precomputed scenarios, the respective parameters can be adjusted freely within their physical limits. However, for this purpose, the underlying models must be very efficient with regard to the computation time.

The process of auralization can be separated into representations for sound generation, propagation and reproduction [6]. For this purpose, a model to characterize the source, the sound transmission (or propagation) and the

receiver are required. Each model provides parameters which are then fed into the digital signal processing (DSP) elements of the actual auralization framework. The separation between these models is not always strict. However, if separated properly, an exchange of a single component is possible. The source model provides the “raw” time-domain sound signal with appropriate power emitted by a source which is used at the beginning of the processing chain. Additionally, the model should provide a directivity function representing the directional sound emission into the far field. The receiver model is the interface to the 3D audio reproduction which is based on the incident directions of sound waves at the receiver position. Typical reproduction methods are binaural synthesis using headphones, or Ambisonics and VBAP using loudspeakers. Finally, the sound propagation model represents the impact of the environment between source and receiver. This includes effects like spreading loss, medium attenuation and interaction with obstacles, e.g. specular reflections.

Many approaches exist for simulating the sound propagation in the atmosphere. Wave-based approaches usually provide relatively accurate results with uncertainties just due to uncertain input data. Typical examples are fast-field program (FFP), parabolic equation (PE) and the finite-difference time-domain (FDTD) method [7]. Due to their accuracy these methods are very suitable for providing benchmarks. For example, Kirby recently released a semi analytic framework for atmospheric sound propagation based on the finite element method [8]. However, the wave-based approaches are computationally slow and therefore less convenient for the purpose of auralization, especially

\*Corresponding author: [psc@akustik.rwth-aachen.de](mailto:psc@akustik.rwth-aachen.de)

if real-time processing is required. Additionally, these methods typically do not provide the incident direction of the sound waves at the receiver as required for the 3D sound reproduction, at least not without post-processing for spatial (directional) wave decomposition. Directional information, however, is directly available in the case of models based on the concept of geometrical acoustics. Using a high-frequency approximation by neglecting wave-based effects such as scattering and diffraction, sound waves can be reduced to sound paths or rays [7, 9]. This approach has a significant advantage in terms of computational complexity. Knowing the sound paths connecting a source and a receiver, the acoustic properties, such as spreading loss and medium attenuation, can be determined based on appropriate models. These, again form the basis for the actual audio processing.

Many state of the art auralization frameworks for aircraft noise use a sound propagation model based on geometrical acoustics [3, 5, 10, 11]. A typical assumption is that the sound paths between the aircraft and receiver are straight [3, 5, 11]. Such a model is very easy to implement as the sound paths for direct sound and the ground reflection can be found using deterministic algorithms like the image source method. Most importantly, it is extremely efficient; Sahai et al. [5] actually introduced a real-time capable system based on this approach.

In the atmosphere, sound propagates on curved paths due to refraction and advection caused by the inhomogeneity and movement of the medium [7]. As first observed in studies by Reynolds [12, 13], this can have an audible impact on the perceived sound. Using a more computationally complex method like ray tracing, these effects can be considered, while still allowing for reasonable computation times. For this purpose, Arntzen et al. [10] introduced an efficient method to find the eigenrays, the sound paths connecting a source with a receiver, in a stratified moving medium. However, their method is based on the effective sound speed approximation neglecting the effect of advection which can have a considerable influence on the sound paths [7]. Furthermore, details about the implementation are not revealed and consequently, the code is not open-source.

Thus, the present work introduces an open-source ray tracing framework for the purpose of aircraft noise auralization. This approach considers both effects, refraction and advection. It provides an efficient method similar to the one by Arntzen et al. to find eigenrays in a stratified moving medium. The objective of this study is to determine the performance of this algorithm in terms of accuracy and real-time capability in the context of aircraft auralization. For this purpose, a typical flyover scenario using an aircraft take-off trajectory is simulated. The algorithm is evaluating with respect to run-time and accuracy of the eigenrays and compared to the state of the art method.

## 2 Atmospheric Ray Tracing

The Atmospheric Ray Tracing (ART) framework is an open-source C++ tool to simulate sound propagation

through an inhomogeneous, moving atmosphere. It is designed for the purpose of auralizing aircraft noise, but can also be used for other applications. ART considers refraction caused by inhomogeneity of speed of sound and wind. In contrast to the effective sound speed approach, it also considers the wind component perpendicular to the propagation direction (advection). In the context of auralization, it is desired to find the rays connecting a source with a receiver, which are called *eigenrays* [7]. Hence, this framework provides an efficient method to determine eigenrays for the direct sound and the ground reflection in a 3D environment.

ART is part of the C++ library collection ITAGeometricalAcoustics.<sup>1</sup> It comes with a Matlab interface<sup>2</sup> that allows a fast and easy simulation of sound paths emitted by a source and finding eigenrays connecting a source and with a receiver. A working binary of this interface can be found in the [Supplementary Data](#).

The following sections provide a detailed discussion of the models and methods used in the framework. This includes the model for the atmosphere, the utilized ray equations, the ground reflection, as well as the *adaptive ray zooming* method, which is used to speed up the process of finding eigenrays. Finally, a brief overview on how to derive acoustic parameters for the auralization from these eigenrays is given.

### 2.1 Stratified atmosphere model

When simulating sound propagation in the atmosphere, obviously, the speed of sound  $c$  and wind (medium movement)  $\mathbf{v}$  have to be considered. The latter is a vector pointing in wind direction  $\mathbf{v}_{\text{dir}}$  while its length corresponds to the wind velocity  $v$ . Generally, both parameters vary throughout the medium. This inhomogeneity or more specifically, the gradients  $\nabla c$  and  $\nabla \mathbf{v}$ , lead to sound refraction [14]. Thus, in order to execute ray tracing in the atmosphere, a proper model for all of the above mentioned parameters is required.

An important assumption for the ray tracing process is that all atmospheric parameters are timeindependent and therefore typically time-averaged values are considered [7]. With respect to turbulence, the corresponding time scale is assumed to be small compared to the acoustic period and travel time. Hence, turbulence does not affect the resulting sound paths. It can be included during the auralization process using time-variant filters [15].

Furthermore, a common simplification for the atmosphere is the so-called *stratified medium* [7]. For this model, it is assumed that all atmospheric parameters depend on altitude  $z$  only. Additionally, the vertical component of the medium movement  $v_z$  is often neglected, since it is typically exceeded by its horizontal component  $\mathbf{v}_{\perp}$  by a factor of 10–100 [7]. This simplified model has multiple advantages.

First, there are analytic models available, which allow an easy implementation of the atmospheric properties. This is especially convenient regarding temperature and medium

<sup>1</sup> <https://git.rwth-aachen.de/ita/ITAGeometricalAcoustics>

<sup>2</sup> <https://git.rwth-aachen.de/ita/ITAGeometricalAcoustics/-/tree/master/apps/ARTMatlab>

movement, since also for their gradients an analytic formula can be found implicitly. It should also be noted, that based on the assumptions mentioned above the horizontal gradients are negligible ( $\nabla \rightarrow (0, 0, \frac{d}{dz})$ ). Another advantage of using analytic models is that they allow an easy comparison between different sound propagation tools. For the temperature, the vertical gradient depends on several factors, e.g. humidity and altitude. The International Standard Atmosphere (ISA) [16] provides a simplified model for the troposphere ( $z < 11$  km) assuming a constant gradient and dry air:

$$T(z) = T_0 - 0.0065 \frac{\text{K}}{\text{m}} \cdot z, \quad (1)$$

where  $T_0 = 288.15$  K is the temperature at mean sea level. For the medium movement, the logarithmic wind profile [17] is a suitable model within the atmospheric surface layer. According to this model, the wind velocity  $v = |\mathbf{v}|$  is zero below the so-called surface roughness  $z_0$ . Above, it increases logarithmically:

$$v(z) = \frac{v_*}{\mathcal{K}} \cdot \ln \left( \frac{z}{z_0} \right). \quad (2)$$

Here,  $\mathcal{K}$  denotes the so-called *Kármán constant*, which is approximately 0.40 [17], while  $v_*$  refers to the friction velocity.

Another advantage of the stratified medium, is that weather measurement data which is publicly available is typically altitude-dependent, e.g. from atmospheric soundings [18], and therefore is ready to be used. Furthermore, the assumptions underlying this model also can speed up the ray tracing algorithm as will be shown in Section 2.2.

In the ART framework, the stratified atmosphere is represented by a flexible class, which allows selection from a set of predefined profiles for temperature, static pressure, wind and relative humidity. In addition to the above mentioned ISA model, a constant profile can be used for temperature and static pressure. For the wind profile, the predefined profiles use a wind velocity which is either logarithmic (Eq. (2)), constant or zero (meaning no medium movement). For each of these profiles, the wind direction is altitude-independent. Also for the relative humidity, a constant value can be selected. Most of the above mentioned profiles, can be further parametrized by the user, e.g. by setting the wind direction and friction velocity of the logarithmic wind profile. Note, that the class representing the atmosphere is designed in a modular way, so that an extension of additional analytic profiles is possible.

For the speed of sound, the formula,

$$c(z) = \sqrt{\gamma R \cdot T(z)}, \quad (3)$$

is used as in Ref. [19]. Here,  $\gamma = 1.4$  is the adiabatic index and  $R = 287.058 \frac{\text{m}^2}{\text{s}^2}$  the molar gas constant for dry air. Generally, the speed of sound also depends on the specific humidity (see (1.1) in [7]). However, in order to reduce the required input data, this is neglected in its influence on the speed of sound. Additionally, this significantly reduces the complexity of the speed of sound gradient formula.

Nevertheless, the humidity is accounted for by calculating the medium attenuation, as shown later in Section 2.5.

In addition to using analytic formulas, the profiles can be imported as altitude-dependent data sets. Internally, these data are then represented by piecewise polynomials which are generated using cubic splines. This allows not only a fast and robust interpolation between the data points, but also evaluation of the gradients of temperature and wind vector based on the derivative of the respective polynomials with respect to  $z$ . The previously mentioned Matlab interface provides a routine to directly import data files based on atmospheric soundings provided on [18]. However, it is also possible to import data using Matlab vectors/matrices, which allows importing from other sources.

## 2.2 Ray equations

Generally, rays can be interpreted as a trajectory generated by the movement of a fixed point on a wavefront normal [20]. Thus, rays propagating through an inhomogeneous, moving medium like the atmosphere can be described using two vectors: a fix point on a wavefront  $\mathbf{r} = (x, y, z)$  and the corresponding wavefront normal  $\mathbf{n}$ . As the wavefront propagates, the wavefront normal changes due to refraction caused by the inhomogeneity of the medium on the one hand. On the other hand, the wind vector  $\mathbf{v} = \mathbf{v}(\mathbf{r})$  and the speed of sound  $c = c(\mathbf{r})$  change with the position of the wavefront. This leads to a system of ordinary differential equations (ODEs) in respect to time, one regarding the velocity of the fix point  $\frac{d}{dt}\mathbf{r}$  (also referred to as group velocity [7]) and one for the refraction  $\frac{d}{dt}\mathbf{n}$ .

An efficient formulation of these ray equations is given by Pierce [9], which is also used in this approach. Pierce, instead of using the wavefront normal directly, uses the so-called *wave slowness* or *slowness vector*,

$$\mathbf{s} = \frac{\mathbf{n}}{c + \mathbf{n} \cdot \mathbf{v}}. \quad (4)$$

While this vector points into the same direction as  $\mathbf{n}$ , its norm equals the reciprocal of the effective speed of sound, which is the reason for its name. The main advantage of this approach is, that for a stratified medium as described in Section 2.1, the  $x$ - and  $y$ -components of  $\mathbf{s}$  are constant along a ray [7].<sup>3</sup> Thus, the ODE for the refraction can be reduced from three dimensions<sup>4</sup> to one, which reduces the computational effort. Then, the equation for the  $z$ -component of  $\mathbf{s}$  becomes,

$$\frac{d}{dt}s_z = -\frac{\Omega}{c} \frac{d}{dz}c - s_{\perp} \cdot \frac{d}{dz}v_{\perp}, \quad (5)$$

with  $\Omega = 1 - \mathbf{v}_{\perp} \cdot \mathbf{s}_{\perp}$ . In the equation above, the index  $\perp$  refers to the horizontal components of the respective vectors. In contrast, the equation for the ray velocity [7, 9],

$$\frac{d}{dt}\mathbf{r} = c \cdot \mathbf{n} + \mathbf{v} = \frac{c^2}{\Omega} \mathbf{s} + \mathbf{v}, \quad (6)$$

<sup>3</sup> see (3.64) in [7] using  $\mathbf{s} = \mathbf{b}/c0$ .

<sup>4</sup> see (8-1.10a)/(8-1.10b) in [9] or (10.72)/(10.73) in [7] for a full set of equations.

is three-dimensional leading to a system of four equations in total.

Knowing the initial values for  $\mathbf{r}$  (e.g. source position) and  $\mathbf{n}$  or  $\mathbf{s}$  respectively (initial ray direction), Equations (5)–(6) can be solved using numerical methods like the Euler or Runge–Kutta (RK4) method to estimate the ray trajectory. Although the presented framework allows to use either of the mentioned methods, using the Euler method is not recommended, due to its lack of accuracy and stability [21]. Hence, all results presented in this paper, are based on the RK4 method. Solving these ODEs using a certain time step size  $\Delta t$ , leads to a time series of  $\mathbf{r}$  and  $\mathbf{s}$ . Since outside of the Atmospheric Ray Tracing framework, we are mainly interested in the direction of propagation, the wavefront normal  $\mathbf{n} = |\mathbf{s}|$  is stored instead of the slowness vector. This direction is required, e.g. when applying directivities for the source or doing 3D sound reproduction at the receiver.

It should be noted that certain analytic wind and temperature profiles permit analytic solutions for the ray equations (Eq. (3.79) in [7]). This is a very important reference for benchmarking ray tracing results. However, using such an analytic solution for a general sound propagation tool is inflexible, since for each combination of these profiles, a separate solution has to be implemented. More importantly, this approach does not work when using arbitrary profiles, e.g. from measurements.

### 2.3 Ground reflection

When simulating sound propagation in the context of aircraft noise, the ground is often modeled as a horizontal plane [3, 5, 7, 22]. Although this neglects the topography of the ground which is known to have a significant influence on sound propagation [14], it allows an efficient implementation of the ground reflection. It is especially reasonable when the source is located relatively far above the ground, e.g. aircraft. Additionally, this model is compatible with the assumption of a stratified medium, since using non-flat terrain would be contradictory to horizontal layering of the atmosphere.

In the ART framework, the ground is modeled using the  $xy$ -plane ( $z = 0$ ). Using this model together with a stratified medium, specular reflections can be implicitly carried out by tracing rays through the ground [7]. For this purpose, all atmospheric parameters are evaluated using the absolute altitude  $|z|$ . Since rays below the surface behave like mirrored versions of rays which were actually reflected, the sign  $\frac{d}{dz}s_z$  has to be flipped for  $z < 0$ . This is simply done by multiplying the well-known signum function to the right-hand side of Equation (5):

$$\frac{d}{dt}s_z = \text{sign}(z) \cdot \left( -\frac{\Omega}{c} \frac{d}{dz}c - \mathbf{s}_\perp \cdot \frac{d}{dz}\mathbf{v}_\perp \right). \quad (7)$$

Although not required for the ray tracing itself, a reflection is detected by observing the sign of  $z$ : a sign change between two consecutive time steps indicates a reflection. Then, the position of the reflection and the corresponding wavefront

normal (incident direction) can be determined doing a linear interpolation between those steps. This information is stored, since it is of interest in the context of auralization, e.g. when applying a reflection factor.

When using the ray tracing results outside of the ray tracing algorithm (e.g. for visualization of sound paths), it is usually desirable to mirror parts of the rays which are below the ground. For this purpose, the  $z$ -components of  $\mathbf{r}$  and  $\mathbf{n}$  can be corrected using,

$$z' = |z| \quad \text{and} \quad n'_z = \text{sign}(z) \cdot n_z. \quad (8)$$

### 2.4 Adaptive ray zooming

As explained before, for the purpose of auralization, it is desirable to find the eigenrays connecting a source (e.g. an aircraft) with a receiver. Using the assumption of a stratified atmosphere and a flat ground, there are typically two eigenrays in a flyover scenario: one for the direct sound and a second one for the ground reflection.<sup>5</sup> The aircraft is located far above the receiver in a huge distance and the emission directions of both eigenrays have a significant vertical component. Due to this distance, a high angular resolution for the emission angle of the rays is required to find these eigenrays within a certain accuracy. Using a brute force approach by emitting rays with a certain angular resolution, leads to a huge overhead of rays which are irrelevant for the auralization. To reduce this mismatch, Arntzen et al. [10] introduced an approach iteratively increasing the angular resolution by “zooming in on the closest ray and relaunching a new cluster of rays in a new direction around the last closest ray”. Unfortunately, no further details on the implementation were given.

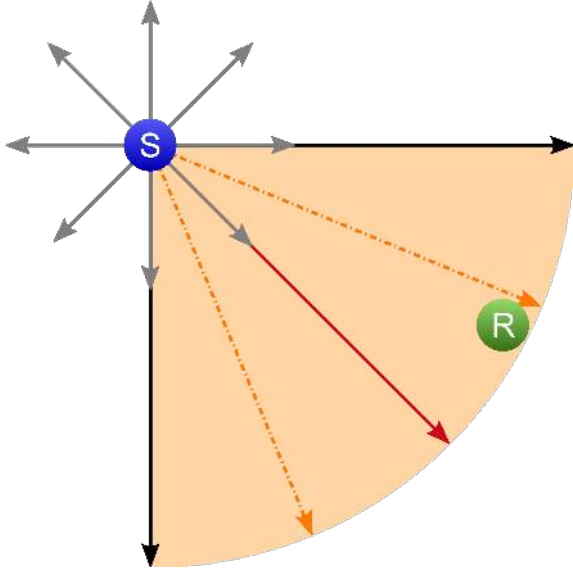
Based on this idea, a method called *adaptive ray zooming* is presented here. A rough overview of this method has already been introduced in [23]. However, it is presented here in more detail. It should be noted that in contrast to Arntzen et al., who use the effective sound speed approximation, the presented approach considers the effect of advection. Furthermore, the code of ART framework is open source and a ready-to-use binary is provided.

#### General approach

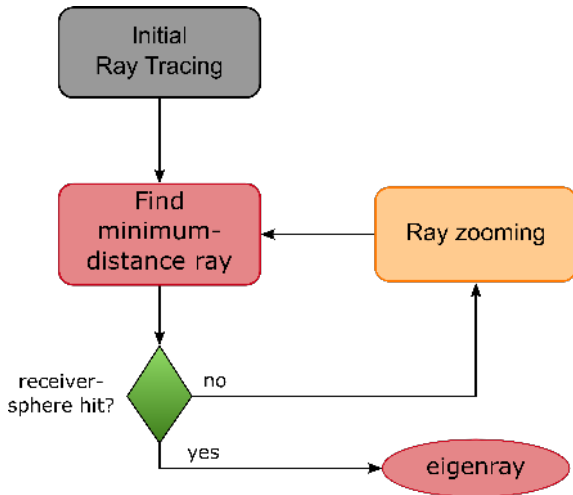
To give an introduction into the adaptive ray zooming method, an example of a two-dimensional scene is given in Figure 1. In an initial step, rays are emitted using a low angular resolution (gray rays). Then, the ray of minimum distance to the receiver is determined (red ray). During the actual ray zooming, additional rays (orange) are launched between this ray and its direct neighbors (highlighted area). In this way, the angular resolution is doubled but only in a particular direction. The process of finding the ray of minimum distance and ray zooming is repeated until the determined ray penetrates a receiver sphere as shown in Figure 2.

<sup>5</sup> Under certain conditions, the number of eigenrays might vary as being discussed later in the *limitations* section.





**Figure 1.** 2D example of the adaptive ray zooming approach. Initial rays are indicated in gray. The ray closest to the receiver is marked in red. Its neighbors, which determine the limits for the next ray zooming iteration, are highlighted in black. Additional rays are launched in between (orange) to double the resolution in a particular direction.

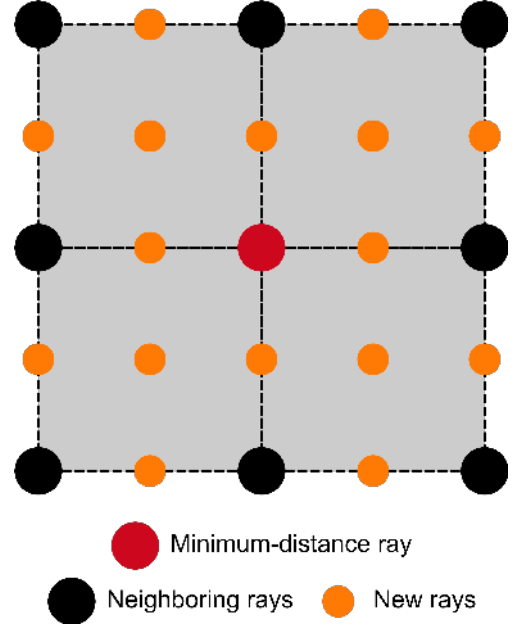


**Figure 2.** Block diagram of the adaptive ray zooming approach.

### Implementation in 3D

During the initial ray tracing the angular resolution is set to  $\Delta\theta = 30^\circ$  in elevation and  $\Delta\phi = 36^\circ$  in azimuth leading to 62 rays in total. For this step only, the time step size  $\Delta t$  for solving the ODEs is increased by a factor of 10 reducing the computational effort.

While tracing a ray, its distance to the receiver is tracked during each time step, while storing the ray parameters corresponding to the minimum distance. To further improve the efficiency, tracing is stopped as soon as the



**Figure 3.** Wavefront formed by rays considered during one iteration of the adaptive ray zooming approach as well as additionally launched rays.

distance increases.<sup>6</sup> In a post-processing step, the time and position of minimum distance is refined by linear interpolation between the stored time step and its neighbors. This is particularly important during the initial ray tracing where the temporal resolution is rather low. Finally, the ray of minimum distance is found comparing the ray-receiver distance between all considered rays.

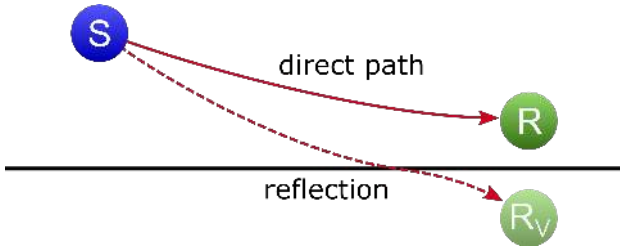
During the actual ray zooming, only the ray of minimum distance and its neighbors are considered. In the three-dimensional case, these are typically nine rays as shown in Figure 3. Following, 16 additional rays are launched to double the resolution in both, elevation and azimuth. Note, that in the first ray zooming iteration, also the 9 original rays have to be relaunched to obtain the desired temporal resolution which was reduced during the initial ray tracing.

A special case occurs, if the ray of minimum distance was emitted towards one of the poles. Then, the number of neighbors actually depends on the azimuth resolution  $\Delta\phi$ ,

$$N_{\text{neigh,pole}} = \left\lceil \frac{360^\circ}{\Delta\phi} \right\rceil, \quad (9)$$

instead of being fixed at 8. The number of additionally launched rays is actually three times as high. Consequently, not only is the number of rays to be calculated higher than usual, it also increases with angular resolution. This can lead to a drastic increase of emitted rays when the source is more or less directly above the receiver. To avoid this effect, the azimuth resolution is fixed if the zooming direction is one of the poles. In this way, a drastic increase of computation time is prevented.

<sup>6</sup> This is the default setting, however, it can be disabled so that each ray is traced for a user-defined propagation time.



**Figure 4.** Virtual receiver position considered to find eigenray of ground reflection.

In order to find an additional eigenray for the ground reflection, a second, virtual receiver position is considered. For this purpose, the receiver position is simply mirrored at the ground by flipping the sign of its  $z$ -component as shown in Figure 4. The approach is analogous to the well-known image source method. Thus, the initial ray tracing returns two rays of minimum distance, one for the direct sound path and the other for the ground reflection. Then, the adaptive ray zooming is carried out for each of these rays using the respective receiver position.

As discussed earlier, a receiver sphere is used as accuracy criterion. As soon as the distance between a ray and the receiver is below the user-defined receiver radius, the ray is considered as eigenray and the algorithm stops. However, there are additional abort criteria to avoid infinite loops: a maximum number of iterations and a minimum angular resolution. These are necessary, since there are some limiting cases where the presented method is unable to find eigenrays with the desired accuracy.

### Limitations

A typical limitation of ray tracing in an inhomogeneous medium is dealing with the so-called *shadow zone* (see Fig. 1.4 in [7]). This phenomenon happens when the source is located close to the ground, where the gradient of the wind vector is usually strong and therefore dominates the refraction. If the receiver resides upwind and the elevation looking from the source to the receiver is almost horizontal, the upward refraction can prevent rays from entering the shadow zone. Although the ray tracing solution suggests that the source is not audible for the receiver, sound energy can still enter the shadow zone due to turbulent scattering and diffraction. Thus, additional measures when auralizing scenes where a receiver resides within a shadow zone would be required. For example, Arntzen et al. propose an empirical correction method for the transmission loss inside such a zone [10].

Under the same conditions, but if a receiver resides downwind, the downward refraction caused by the wind gradient can lead to turning points of the rays. In other words, rays which were already reflected are bent back towards the ground and reflected again [7]. Then, also eigenrays of higher reflection order may exist which currently cannot be found using adaptive ray zooming.

Another special case of sound propagation in inhomogeneous media is ducting (also called channeling) [7, 9].

Under certain conditions, rays which are launched at a flat elevation angle can be refracted upwards and downwards. Then, in a stratified medium, rays are trapped within a duct extending between a lower and upper altitude limit. In the atmosphere, this can be caused e.g. by temperature inversion [22]. Since rays within this duct cross each other although not being reflected, the adaptive ray zooming method could make a false decision and therefore fail to find the direct eigenray. This could happen, if source and receiver reside inside a duct or are at least close to it. However, this effect has not been tested yet.

Finally, for rays arriving at grazing incidence, the model for the ground reflection can lead to erroneous results, since the assumption of a specular reflection is not valid [24].

All of the problematic cases described above mainly occur when rays are launched at angles close to horizontal. For the application of aircraft noise auralization, typically flyovers are considered. In such a scenario, the aircraft is usually at higher altitude, while the receiver resides close to the ground which is contradictory to the requirement above. Thus, it is reasonable to neglect these cases when searching for eigenrays. Although it might be possible to extend the adaptive ray zooming method to cover these cases, this would probably involve special model extensions with increased computation times. As this again disagrees with the goal of using the ART framework for real-time auralization of flyover cases, we postpone this to work in future.

### Advanced ray zooming

As shown in Figure 3, the nine rays considered for the ray zooming span a wavefront which can be divided into four quadrants. Knowing in which part receiver is located, the number of rays can be further reduced. In best case, only four rays have to be considered for the ray zooming and only five new rays have to be launched. In order to limit the launch elevation and azimuth, two independent checks are carried out respectively.

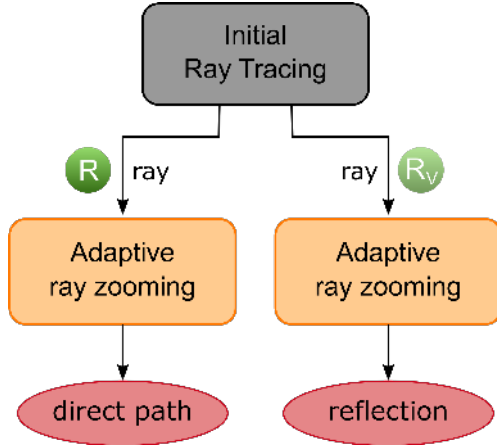
Figure 6 indicates how a decision for the elevation can be done. For this purpose the vectors from the ray of minimum distance and the rays “above” and “below” to the receiver position,  $\mathbf{d}_{\min}$ ,  $\mathbf{d}_{\text{top}}$  and  $\mathbf{d}_{\text{bottom}}$ , are considered. Since the receiver resides in the lower half of the wavefront, the vectors  $\mathbf{d}_{\min}$  and  $\mathbf{d}_{\text{top}}$  point in a similar direction while  $\mathbf{d}_{\min}$  and  $\mathbf{d}_{\text{bottom}}$  point in a rather opposite direction. Comparing the scalar products of the respective normalized vectors,

$$\chi_{\text{top}} = \frac{\mathbf{d}_{\min}}{|\mathbf{d}_{\min}|} \cdot \frac{\mathbf{d}_{\text{top}}}{|\mathbf{d}_{\text{top}}|}, \quad (10)$$

and,

$$\chi_{\text{bottom}} = \frac{\mathbf{d}_{\min}}{|\mathbf{d}_{\min}|} \cdot \frac{\mathbf{d}_{\text{bottom}}}{|\mathbf{d}_{\text{bottom}}|}, \quad (11)$$

the first is larger than the second. In an extreme case where all vectors are collinear,  $\chi_{\text{top}}$  and  $\chi_{\text{bottom}}$  become 1 and  $-1$  respectively. Thus, the launch elevation is



**Figure 5.** Block diagram of the overall approach to find eigenrays of direct path and ground reflection.

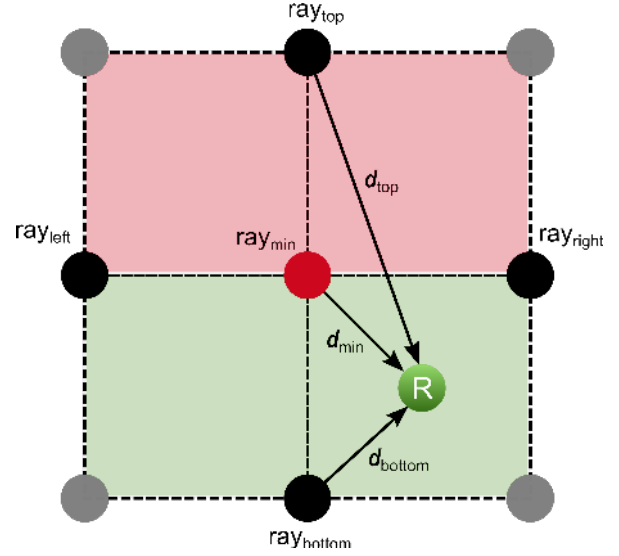
limited to the rays corresponding to the smaller scalar product. An analogous approach is used to make a decision for the azimuth angle. For this purpose, the rays “left” and “right” of the minimum-distance ray are considered.

It should be noted that this approach does not work if zooming towards a pole, since the wavefront cannot be split into quadrants. Also note, that a single false decision in the adaptive ray zooming approach causes the algorithm to fail finding an eigenray. Further reducing the considered directions using the advanced method, could lead to such a wrong decision and therefore decrease the accuracy. Thus, the results from both methods are compared later in Section 3.2.

## 2.5 Auralization parameters

When auralizing sound propagation through the atmosphere, multiple acoustic parameters have to be derived from the eigenrays. Although the presented paper focuses on the simulation of sound paths, a brief overview is given in this section. To be able to consider sound source directivities and to do 3D sound reproduction, the exit angle at the source and the incident angle at the receiver are required. Both parameters are implicitly given when solving the ray equations, since the wavefront normals at the source and receiver are known. When dealing with moving sources and receivers, the Doppler effect should be considered [25]. This effect can be modeled using a variable delay-line (VDL) which takes the propagation delay as input in order to virtually squeeze or stretch the acoustic signal along the time-axis [25]. Also this parameter is implicitly given, since the ray equations are solved using a time integration.

While propagating through the atmosphere, sound is attenuated by the medium. This effect is modeled using the frequency-dependent attenuation coefficient of ISO 9613-1 [26]. The overall attenuation is acquired by integrating this parameter along the ray path using left Riemann sums. Another frequency-dependent effect to be considered is amplitude and phase fluctuations caused by turbulent scattering [15, 27].



**Figure 6.** Vectors used to check whether receiver resides “above” or “below” the current ray of minimum distance. An analogous procedure can be done to check whether it is located “left” or “right”.

Finally, another important parameter is the spreading loss. Due to the presence of refraction, the well-known distance law for point sources does not hold here, since the spherical waveform is distorted. Instead, the Blokhintzev invariant [28] is used which gives a formulation for the energy conservation along a ray path. From this, a formula for the sound pressure  $p$  can be derived (Eq. (3.62) in [7]). Its primary statement is that the sound pressure  $p$  along a ray is proportional to the square root of a cross-sectional area of a ray tube  $A$ . Assuming that the density  $\rho$  is constant along a ray, the following proportionality is derived:

$$p^2 \sim \frac{c}{A(1 + \mathbf{n} \cdot \mathbf{v}/c)\sqrt{1 + 2\mathbf{n} \cdot \mathbf{v}/c + v^2/c^2}}. \quad (12)$$

Now, the spreading loss factor  $\alpha_{\text{spread}}$  corresponds to the proportion of the sound pressure at the receiver  $p_{\text{receiver}}$  to the reference sound pressure  $p_{\text{ref}}$  1 m away from the source:

$$\alpha_{\text{spread}} = \sqrt{\frac{p_{\text{receiver}}}{p_{\text{ref}}}} \sim \sqrt{\frac{A_{\text{ref}}}{A_{\text{receiver}}}}. \quad (13)$$

Here, it can be seen that it is anti-proportional to the square root of the cross-section area at the receiver  $A_{\text{receiver}}$ . In the presented framework, instead of using a ray tube, this area is calculated based on a set of adjacent rays as used during the adaptive ray zooming method (see Fig. 3). While the area at the receiver is calculated using an approximation of triangles, the reference area is calculated using a spherical approximation. Since the area at the receiver depends on the angular resolution of the rays, the user can define a maximum resolution used for its calculation. If this resolution is not yet reached when the eigenray is found, an additional ray tracing step is carried out using the specified resolution. During our tests, a resolution of  $\Delta\phi = \Delta\theta = 0.01^\circ$  showed reasonable results.

### 3 Results

#### 3.1 Validation of ray tracing model

To validate the sound paths calculated by the presented ray tracing model, it is compared to the Outdoor Sound Propagation Calculator tool by Wilson [29]. This tool uses ray equations as stated in [7] which are equivalent to the equations used here. Since the type of refraction strongly depends on the wind direction, three different settings are investigated: down wind, upwind and side wind. All other parameters for the atmosphere are shown in Table 1. A scenario with a source at 200 m altitude is chosen. For each wind direction, multiple rays are emitted in the positive  $x$ -direction using launch elevation angles between  $-90^\circ$  and  $+90^\circ$  and a  $10^\circ$  step size. The ray equations are solved using the Runge–Kutta method and a time step size of 0.1 s. Each ray is traced for 10 s.

The same settings as for the ART framework are applied to the framework by Wilson. While for the ART framework, the weather profiles are evaluated using the analytical formulas during the ray tracing process, Wilson uses discrete vectors for speed of sound and  $x$ - and  $y$ -component of the wind vector respectively. Thus, the weather profiles are evaluated using an altitude vector of 1 m resolution and fed to Wilson’s framework.

The resulting rays are shown in Figure 7. Generally, the rays are refracted as expected: downward reflection in downwind direction, upward refraction in upwind direction. For perpendicular wind, the effect of advection causes the rays to drift in positive  $y$ -direction (see Fig. 7d). Due to the inhomogeneity of the wind velocity, the  $xy$ -plane projections of some rays result in curved paths as mentioned in [7]. The ART results almost coincide with the rays of the Outdoor Sound Propagation Calculator. Nevertheless, there is a small difference in the paths. The maximum deviation found was 11 m which corresponds to just 0.32% of the ray’s path length (approx. 3.4 km). Assuming a spherical spreading loss, this again corresponds to a negligible deviation of just 0.03 dB. Generally, the deviation occurs when a ray gets close to the ground. This is probably caused by the different handling of weather profiles. For the logarithmic wind profile, the wind gradient is particular strong close to the ground (in the vicinity of  $z_0$ ). Here, a different sampling of the profile can easily lead to the mentioned deviations. Concluding, the results suggest, that the ART framework solves the ray equations with acceptable accuracy.

#### 3.2 Performance of adaptive ray zooming

Knowing that the ART framework traces rays with appropriate accuracy, the performance of the adaptive ray zooming method for finding eigenrays can be evaluated. For this purpose, the algorithm is executed to find eigenrays between a fixed receiver position and a source moving along an aircraft take-off trajectory. The trajectory is generated using the software MICADO [30] with the origin being the initial aircraft position. The aircraft then moves in positive  $x$ -direction while gaining altitude. Thus, the  $y$ -component

**Table 1.** Settings for atmospheric profiles used throughout this paper.

Temperature	
Profile	ISA
Wind	
Profile	Logarithmic
$v_*$	0.6 m/s
$z_0$	0.1 m

of the source position is always zero. The receiver is located at (6500, 0, 1.8) m. The scenario corresponds to the I.C.A.O. standard for aircraft flyover measurements [31]. For the atmosphere, the same temperature and wind velocity profiles as in the previous section are used (see Tab. 1). Now, the wind direction is chosen to  $\mathbf{v}_{\text{dir}} = \mathbf{v}/v = (-1, 0, 0)$ , so that the aircraft takes-off in upwind direction which is a typical scenario [32]. The settings used for the ODE solver and the adaptive ray zooming method are listed in Table 2. In order to investigate the influence of the advanced ray zooming compared to the basic method, all simulations are carried out with and without this option.

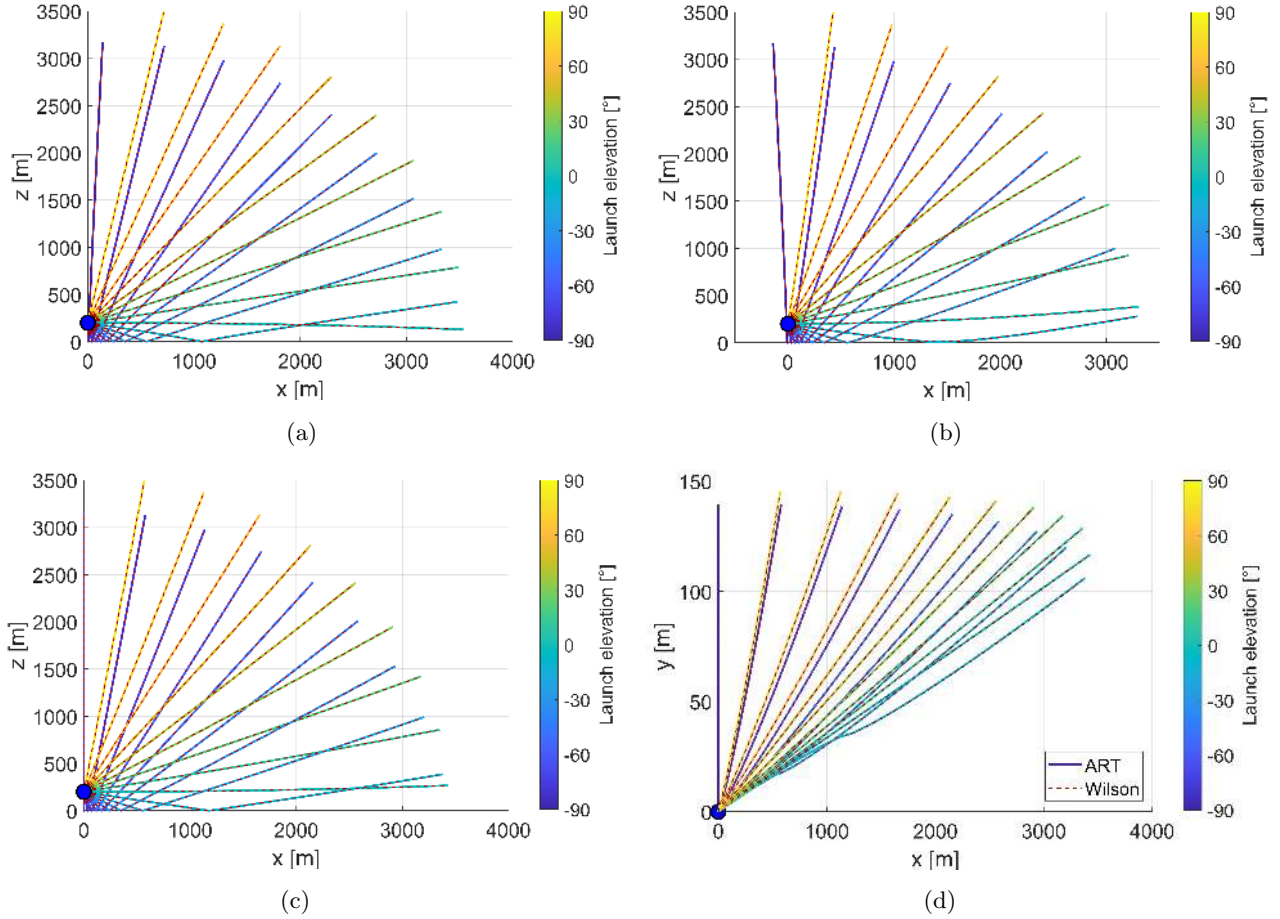
In the context of auralization two things are of interest: the accuracy of the eigenrays and the run-time of the algorithm. If the accuracy of the eigenrays is too low, the derived acoustic parameters used for the auralization do not properly represent the sound propagation through the atmosphere. More importantly, a high inaccuracy between two consecutive time frames can lead to audible artifacts during the auralization process. On the other hand, it is desired to find the eigenrays with low latency. A low run-time allows a fast preparation of offline auralization where the eigenrays are calculated beforehand. If the algorithm is fast enough, it might be possible to auralize aircraft noise based on ray tracing in real-time.

With respect to accuracy, the distance between the eigenrays and the receiver position, called ray-receiver distance in the following, is calculated. For the runtime evaluation, the adaptive ray zooming method is carried out 100 times for each source position. For this purpose, a typical work station computer with an Intel® Core™ i7-7700 CPU (3.6 GHz), 32 GB RAM and Windows 10 (64 bit) operating system is used. Although the ART framework allows calculation of multiple rays in parallel, all calculations are done sequentially. If the algorithm was to be used during a real-time auralization, it would run in a separate thread while the main thread would handle the audio processing. Thus, the run-time of a parallelized approach is not representative for real-time auralization.

In addition to the ray-receiver distance and the runtime, also visualizations of the eigenrays for three different source positions are shown in Figure 8. Note, that an animation of the full flyover is provided here.<sup>7</sup> It can be seen that during the first 60–70 s, the desired accuracy of 1 m cannot be reached. Here, due to strong upwind refraction, the receiver resides within the shadow zone (see Fig. 8b). While the

<sup>7</sup> <https://youtu.be/AHOG6LCeZp8>





**Figure 7.** Comparison of ray propagation using the ART model (solid lines) and the model by Wilson [29] (dashed lines) for three different wind directions. Note that the differences between solid and respective dashed lines are so small that they are not visible. (a) Downwind ( $\mathbf{v}_{\text{dir}} = 1, 0, 0$ ). (b) Upwind ( $\mathbf{v}_{\text{dir}} = -1, 0, 0$ ). (c) Sidewind ( $\mathbf{v}_{\text{dir}} = 0, 1, 0$ ), side view. (d) Sidewind ( $\mathbf{v}_{\text{dir}} = 0, 1, 0$ ), bird's-eye view.

**Table 2.** Simulation settings for ART framework used throughout this paper.

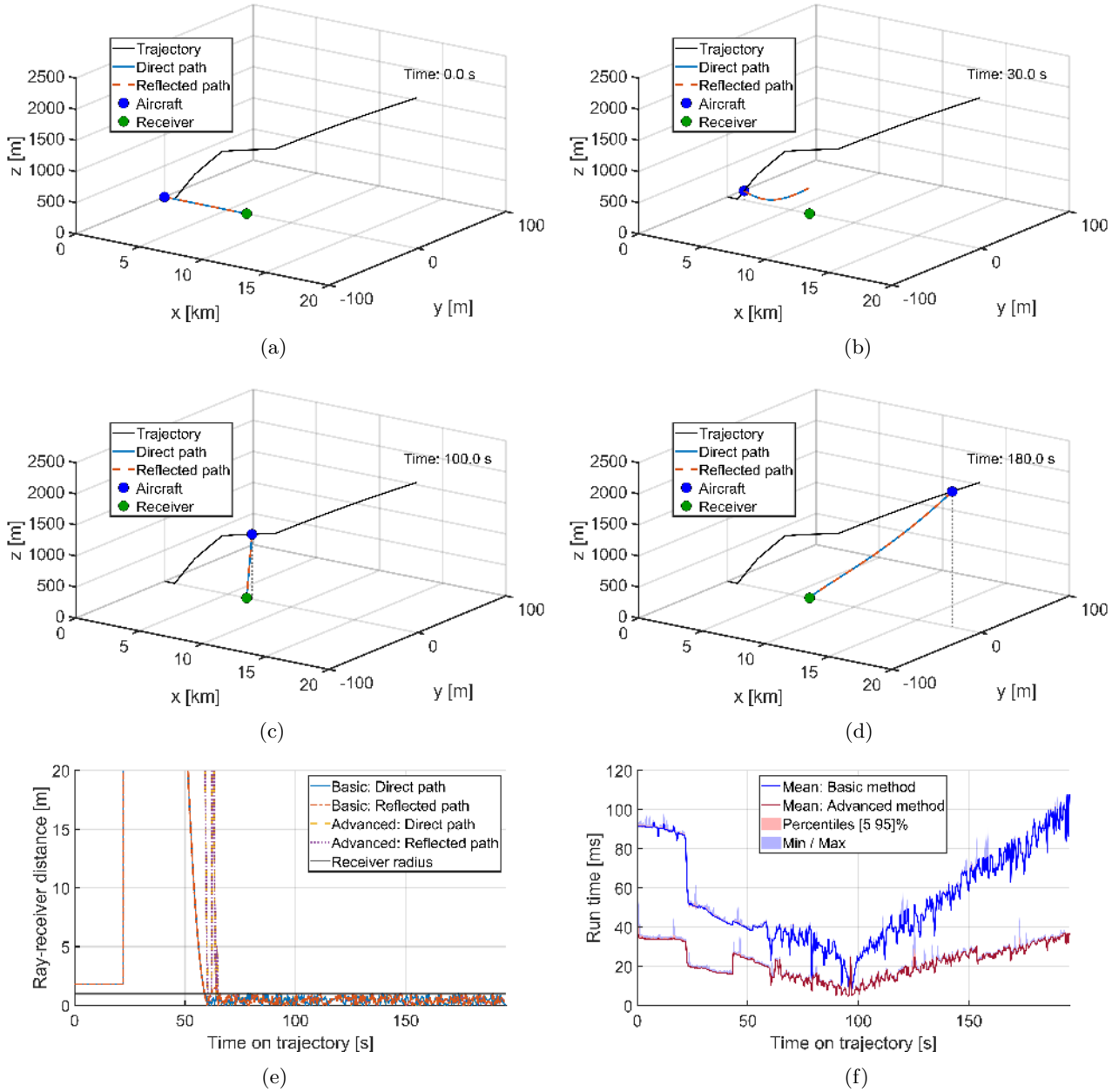
ODE solver	
Method	Runge–Kutta
Integration step size	0.1 s
Ray zooming	
Receiver radius	1 m
Angle for spreading loss	$0.01^\circ$
Maximum number of iterations	30
Minimum ray resolution	$1^\circ$

source is on the ground, the rays can still travel through the windless channel below the surface roughness  $z_0$  (see Fig. 8a). Once the aircraft takes off, the distance between the eigenrays and the receiver increases abruptly. As it gains altitude, the distance decreases over time until the receiver leaves the shadow zone again (see Fig. 8e). While this transition is rather smooth for the basic method, there are discontinuities for the advanced approach. Here, using the advanced method can lead to a wrong decision in an earlier iteration leading to a strong increase in the

ray-receiver distance. Once the receiver leaves the shadow zone, both approaches have the same accuracy.

Also the run-time of the algorithm is influenced by the shadow zone as shown in Figure 8f. Since the algorithm fails to find eigenrays with the defined accuracy, the maximum number of iterations for the ray zooming method is used. Hence, the trend of the runtime is much smoother than for the later points on the trajectory where the number of required iterations fluctuates. The influence of the shadow zone is particularly strong, while the source is still on the ground. Here, the upward refraction caused by the wind is most distinctive. All rays leaving the windless zone are refracted farther away from the receiver. Thus, they have to travel much farther until they reach the point of closest distance. This leads to an increased run-time, since the effort to compute a ray is directly dependent on the rays' propagation time. Once the aircraft takes off (after approx. 22 s), the average path lengths of the considered rays drop significantly and with it the run-time, especially for the basic method.

Regarding the general trend, two things can be observed. First, the run-time of the adaptive ray zooming method scales with the distance between source and receiver. Again,



**Figure 8.** Performance of adaptive ray zooming approach during aircraft flyover. (a) Eigenray results while aircraft is grounded. (b) Eigenray results during take-off phase. (c) Eigenray results while aircraft is above receiver. (d) Eigenray results while aircraft resides beyond receiver. (e) Distance between eigenrays and receiver using basic and advanced ray zooming method, and (f) C++ run-times of adaptive ray zooming method using the basic and the advanced approach.

this is due to the dependency of a ray's computation time on its path length. Secondly, there is a considerable speed-up using the advanced ray zooming method. For the basic method, the runtime ranges from 8 ms when the aircraft is close to the receiver to over 100 ms at when it is at the end of the trajectory. The corresponding source-receiver distances are 1 km and 13.5 km approximately. For the advanced method, the respective run-times are 5 ms and 38 ms which corresponds to speed-ups of about 1.6 and 2.8. An exception can be seen when the aircraft is nearly

directly above the receiver. Then, the zooming direction is toward the nadir of the aircraft. This increases the number of emitted rays and therefore leads to an increase in computation time. For the same reason, the advanced ray zooming method does not give a benefit here. Nevertheless, the advanced approach clearly outperforms the basic method in terms of run-time.

When auralizing based on these eigenray results, artifacts will occur due to the abrupt changes of ray-receiver distance while the receiver resides within the shadow zone.

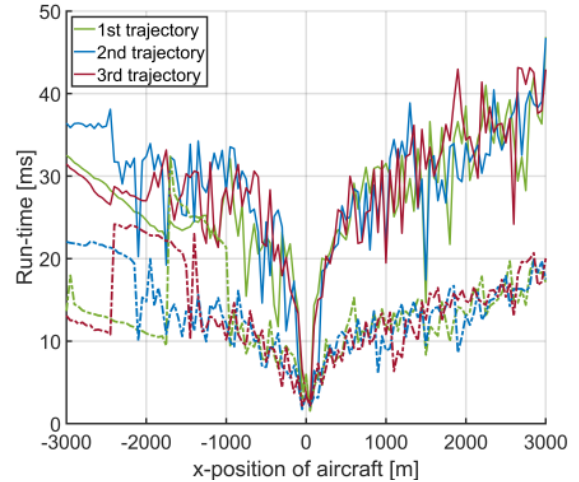
These artifacts will be more prominent using the advanced ray zooming method, as the discontinuities are more distinctive here. However, due to the general limitations of ray tracing, neither the basic nor the advanced method are able to properly reproduce the sound propagation while the receiver resides within the shadow zone. Thus, it is not advisable to use this approach in these cases. In a real scenario, however, the path between a grounded aircraft and the receiver are typically blocked by obstacles, such as buildings. Thus, the considered hemi-free-field model is not reasonable for more realistic cases anyway. In real-life cases with airports, the shadow zone should not be a problem. The next challenge is to integrate aircraft noise models and models of buildings and non-flat terrain. Furthermore, during an actual auralization of a flyover scenario, the aircraft positions closer to the receiver are of most interest, since the sound pressure level is maximal here. Limiting the aircraft trajectory not only avoids the problem with the shadow zone but also leads to a decrease of the maximum computation time for the eigenrays. For example, if only considering positions with a maximum distance of 4 km to the receiver, the run-time stays below 52 ms for the basic and 25 ms for the advanced approach.

### 3.3 Run-time comparison to method by Arntzen

In [22], Arntzen investigates the run-time of his method to find eigenrays based on a sequential CPU and a parallel GPU implementation. For this purpose, he uses three different aircraft trajectories (see Fig. 8.9 in [22]). The same trajectories are used in the present work, in order to compare the results. Since the receiver altitude is not specified, the receiver position is chosen to (0, 0, 1.8) m here. Also the weather conditions are not fully reproducible, especially since Arntzen uses the effective speed of sound. Thus, the parameters of Section 3.2 for weather and simulation settings (see Tabs. 1 and 2) are used here as well.

Again, each eigenray search is carried out 100 times. In Figure 9, the mean run-time is shown for the three trajectories using the basic and advanced ray zooming method, respectively. Also here the strong upward refraction caused by the wind leads to the receiver residing within the shadow zone for negative  $x$ -positions. This is most prominent for the first trajectory, where the receiver leaves this zone only after the aircraft reaches  $x \approx -1000$  m. Here, this leads to a strong increase of the run-times for the advanced method.

Now the results are compared to Figure 8.10 in [22]. Although Arntzen uses the effective sound speed approximation which simplifies the ray equations, the presented framework is at least 10 times faster than his CPU implementation. Neglecting the influence of the shadow zone, the speed-up factor even is 20 for most source positions along the trajectory using the advanced method. Considering the different clock rates of the utilized CPUs (2.4 GHz vs. 3.6 GHz), the speedup is still significant. Compared to Arntzen's GPU-based parallelized approach, the ART framework is at least as fast. For the advanced ray zooming method, the speed-up is still roughly double. When the aircraft nears the receiver, both, basic and advanced



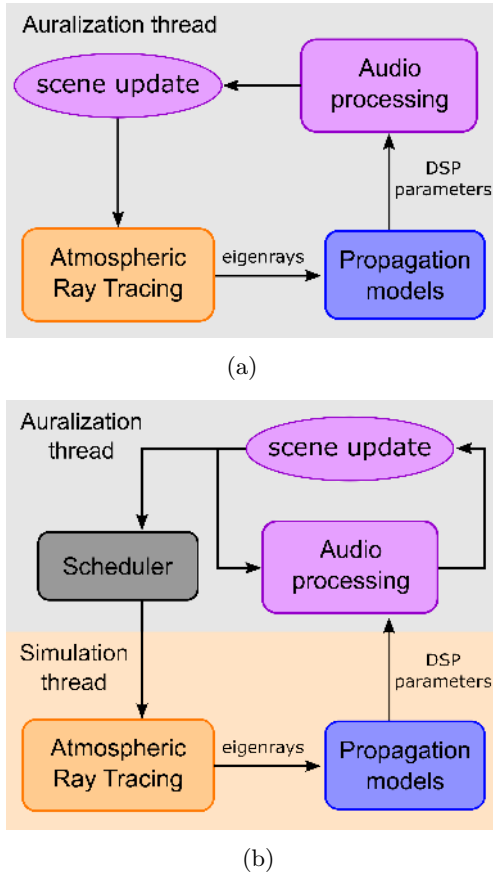
**Figure 9.** Mean run-times of the adaptive ray zooming method using the same aircraft trajectories as Arntzen (compare to Figure 8.10 in [22]). Solid lines show the results for the basic and dashed lines for the advanced approach.

method, are significantly faster than the approach by Arntzen. In addition to that, the CPU implementation has the advantage of working on computers without GPUs, which is the case for many laptops.

All in all, the results suggest that the presented framework outperforms the ray tracing approach by Arntzen in terms of run-time. This is especially the case regarding the sequential CPU implementation. However, Arntzen's method is able to deal with some cases not currently handled by the ART framework. For example, it is capable of finding more than two eigenrays in a downwind scenario. Furthermore, it provides an empirical solution to estimate the spreading loss for a receiver in the shadow zone. Nevertheless, also this solution comes with a limitation as it is formulated for a receiver directly on the ground and based on linear sound speed profiles.

### 3.4 Real-time performance and potential optimization

To estimate the real-time capability of the algorithm, the run-time is compared to blockwise audio processing as done during real-time auralization. There are two ways of integrating the simulation into an auralization process. As shown in Figure 10, the audio processing and the simulation can be handled sequentially or the simulation can be run in a separate thread. In both cases, the auralization thread is already considerably loaded by the actual audio processing, e.g. the interpolation of samples in the variable delay-line (VDL) for the Doppler shift or the convolution with head-related transfer functions (HRTFs) for binaural reproduction. Thus, for the sequential approach, the simulation run-time needs to be significantly lower than the time corresponding to the block length. Nevertheless, also when using a separate thread for the simulation, there is a computational overhead. First, the simulation requests have to be properly scheduled. Second, after a simulation is finished, the results have to be extracted in the main thread.



**Figure 10.** Simplified diagrams of integrating the Atmospheric Ray Tracing framework into an auralization process. (a) Sequential auralization approach. (b) Simulating in separate thread.

Assuming a rather large audio block length of 1024 samples, this corresponds to 23 ms using a sampling rate of 44.1 kHz. As discussed above, the eigenray search should be considerably faster than this block time. Even using the advanced ray zooming method, however, the run-times presented in Figures 8f and 9 are under this value only for source positions close to the receiver. This has been confirmed in a preliminary auralization attempt where both approaches, the sequential and parallel simulation, have been tested for real-time capability. Using the same setup as for the results in Figure 8, including test system, aircraft trajectory and receiver position, neither of the two approaches allowed for auralization in real-time. However, for the sequential approach, moving the receiver closer to the trajectory (position: [2000, -100, 1.8] m), a real-time auralization worked at least temporarily when the aircraft was closest to the receiver. Although the corresponding time window was only a few seconds, this indicates that the current run-time of the ray zooming method is close to a threshold allowing for real-time auralization.

Thus, with further improvements of the system, a real-time auralization of aircraft noise based on ray tracing should be possible. For instance, the process of finding eigenrays could be further accelerated using additional assumptions to achieve real-time capability. For example,

the emission angles of direct and reflected path are very similar for a receiver close to the ground which is typically the case when auralizing aircraft noise. In a dynamic scenario, the same holds for the emission angles of eigenrays between consecutive time steps. This could be used to limit the considered directions for the eigenray search and therefore reducing the number of calculated rays. A different but complementary approach would be to interpolate between the simulation results within the auralization framework as suggested in [22]. For this purpose, the simulation has to be carried out in a separate thread as the audio processing. The results are stored in a buffer. In each processing step, the data are then interpolated accordingly. This approach allows higher update rates for the audio processing than for the simulation. Thus, it could enable using even smaller block sizes than 1024. Particularly for modeling of the Doppler shift based on the propagation delay, this approach might be advantageous. Depending on the scene, the propagation delay can require higher update rates to avoid artifacts caused by abrupt time-shifts in the variable delay-line.

Note that the ART framework already has been successfully used to create offline auralizations of aircraft flyovers. A video of a binaural scene is publicly available.<sup>8</sup>

## 4 Conclusion

In this paper, an open source ART framework for fast simulation of sound propagation in a stratified atmosphere is introduced. This framework uses an efficient method, called adaptive ray zooming, for finding eigenrays connecting a source and receiver. Although the algorithm is designed for the purpose of aircraft noise auralization, it could be applied to other scenarios such as quick parametric simulation of local variations in day-to-day weather conditions in order to create monthly and yearly averages. The framework comes with a Matlab interface which allows an easy setup of simulations and is available as a binary.

The method was tested using an aircraft take-off trajectory for the source positions and a fixed receiver position close to the ground. Although there are limitations to the ray zooming method, e.g. when the receiver resides within the shadow zone, it works well for most source positions, especially those close to the receiver, which are most relevant for the evaluation of a flyover.

A run-time evaluation underlined the efficiency of the adaptive ray zooming method. Comparison to a similar method by Arntzen et al. indicated that it is even more efficient than this state of the art approach. Although the present approach uses more complex ray equations incorporating the effect of advection, the presented run-times are significantly lower for a sequential implementation.

The results also showed that the method is not yet fast enough to enable real-time auralizations. In future work, the process of finding eigenrays could be further accelerated using additional assumptions with respect to emission

<sup>8</sup> [https://youtu.be/Zn26naG\\_e24](https://youtu.be/Zn26naG_e24)



angles of direct and reflected paths or between consecutive time frames in a dynamic scene. Furthermore, the simulation results could be interpolated in the auralization framework. This would enable to use higher update rates for the audio processing than for the simulation. Using one or combining multiple of these approaches might lead to a real-time capable system.

A typical application for auralization of aircraft, is the psychoacoustic evaluation of the perceived noise using listening tests. For example, different aircraft designs can be compared by varying the source signal. Once a real-time auralization is possible, test subjects would be able to quasi-instantly change source parameters or weather conditions freely within the respective physical limits. This again enables implementation of novel listening test designs.

Finally, due to its fast run-time, the ART framework enables simulation of a large number of scenarios which can be used for statistical analysis. In this context, it would be interesting to investigate the influence of the weather conditions on the actual auralization result. For this purpose, measured data, e.g. from atmospheric soundings, could be used to provide a realistic variance of the weather parameters at a single location.

## Supplementary data

The supplementary material of this article is available at <https://acta-acustica.edpsciences.org/10.1051/aacus/2021018/olm>. This paper comes with supplementary data which allow reproduction of the data-based plots shown here. This data includes a binary of the ART framework using the Matlab interface ARTMatlab. Additionally, a copy of the utilized version of the ITA Toolbox [33] is provided. Running the respective simulations and creating the plots requires a working Matlab version. Here, all plots were created using Matlab 2019b. Additionally, an animation is provided to complement Figure 8. This animation shows the eigenrays for all considered aircraft positions along the trajectory.

## Acknowledgments

The authors would like to thank Vladimir Ostashev for the fruitful discussions about sound propagation and ray tracing in the atmosphere which helped to improve the content of this article substantially.

## References

1. World Health Organization (WHO): Environmental noise guidelines for the European Region, 2018.
2. S.A. Rizzi, A. Christian: A psychoacoustic evaluation of noise signatures from advanced civil transport aircraft, in 22nd AIAA/CEAS Aeroacoustics Conference, Lyon, France, American Institute of Aeronautics and Astronautics, 2016.
3. R. Pieren, L. Bertsch, D. Lauper, B. Schaffer: Improving future low-noise aircraft technologies using experimental perception- based evaluation of synthetic flyovers. *Science of The Total Environment* 692 (2019) 68–81.
4. C. Dreier, M. Vorländer: Psychoacoustic optimisation of aircraft noise – challenges and limits, in INTER-NOISE and NOISE-CON Congress and Conference Proceedings, Vol. 261, Institute of Noise Control Engineering, 2020, 2379–2386.
5. A. Sahai, F. Wefers, S. Pick, E. Stumpf, M. Vorländer, T. Kuhlen: Interactive simulation of aircraft noise in aural and visual virtual environments. *Applied Acoustics* 101 (2016) 24–38.
6. M. Vorländer: Auralization: Fundamentals of acoustics, modelling, simulation, algorithms and acoustic virtual reality. RWTH edition, 2nd ed. Springer International Publishing, Cham, 2020.
7. V.E. Ostashev, D.K. Wilson: Acoustics in moving inhomogeneous media, 2nd edn. CRC Press, 2016.
8. R. Kirby: Atmospheric sound propagation in a stratified moving media: Application of the semi analytic finite element method. *The Journal of the Acoustical Society of America* 148, 6 (2020) 3737–3750.
9. A.D. Pierce: Acoustics: An introduction to its physical principles and applications. Acoustical Society of America, Woodbury, NY, 1989.
10. M. Arntzen, S.A. Rizzi, H.G. Visser, D.G. Simons: Framework for simulating air craft flyover noise through nonstandard atmospheres. *Journal of Aircraft* 51, 3 (2014) 956–966.
11. S.A. Rizzi, A.R. Aumann, L.V. Lopes, C.L. Burley: Auralization of hybrid wing-body aircraft flyover noise from system noise predictions. *Journal of Aircraft* 516 (2014) 1914–1926.
12. O. Reynolds: On the refraction of sound by the atmosphere. *Proceedings of the Royal Society of London* 22 (1874) 531–548.
13. O. Reynolds: On the refraction of sound by the atmosphere. *Philosophical Transactions of the Royal Society of London* 166 (1876) 315–324.
14. J.E. Piercy, T.F.W. Embleton, L.C. Sutherland: Review of noise propagation in the atmosphere. *The Journal of the Acoustical Society of America* 61, 6 (1977) 1403–1418.
15. F. Rietdijk, J. Forssen, K. Heutschi: Generating sequences of acoustic scintillations. *Acta Acustica united with Acustica* 103, 2 (2017) 331–338.
16. International Organization for Standardization: ISO 2533:1975, Standard Atmosphere, 1975.
17. R.O. Weber: Remarks on the definition and estimation of friction velocity. *Boundary-Layer Meteorology* 93, 2 (1999) 197–209.
18. University of Wyoming, College of Engineering, Department of Atmospheric Science. Atmospheric Soundings, 2021. <http://weather.uwyo.edu/upperair/sounding.html>.
19. M.A. Garces, R.A. Hansen, K.G. Lindquist: Traveltimes for infrasonic waves propagating in a stratified atmosphere. *Geophysical Journal International* 135, 1 (1998) 255–263.
20. R.B. Lindsay: Mechanical Radiation. International Series in Pure and Applied Physics, Mc-Graw Hill, 1960.
21. W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling: Numerical Recipes in C: The art of scientific computing, 2nd ed. Cambridge University Press, 1992.
22. M. Arntzen: Aircraft noise calculation and synthesis in a non-standard atmosphere, PhD thesis. Delft University of Technology, 2014.
23. J. Mecking, J. Stienen, P. Schafer, M. Vorländer: Efficient simulation of sound paths in the atmosphere, in INTER-NOISE and NOISE-CON Congress and Conference Proceedings, Vol. 255. Institute of Noise Control Engineering, Hongkong, 2017, 3464–3469.

24. J.S. Suh, P.A. Nelson: Measurement of transient response of rooms and comparison with geometrical acoustic models. The Journal of the Acoustical Society of America 105, 4 (1999) 2304–2317.
25. H. Strauss: Implementing Doppler shifts for virtual auditory environments. Journal of the Audio Engineering Society 104 (1998) 1–4.
26. International Organization for Standardization: ISO 9613–1:1993, Acoustics – Attenuation of sound during propagation outdoors – Part 1: Calculation of the absorption of sound by the atmosphere, 1993.
27. V.E. Ostashev, D.K. Wilson: Statistical characterization of sound propagation over vertical and slanted paths in a turbulent atmosphere. Acta Acustica united with Acustica 104, 4 (2018) 571–585.
28. D.I. Blokhintzev: Acoustics of non-homogeneous moving media, NACA TM-1399, 1956.
29. D.K. Wilson: Outdoor sound propagation calculator, in V.E. Ostashev, D.K. Wilson (Eds.), Acoustics in moving inhomogeneous media, 2nd ed., Taylor & Francis, 2015.
30. K. Risse, E. Anton, T. Lammering, K. Franz, R. Hoernschemeyer: An integrated environment for preliminary aircraft design and optimization, in 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA, Honolulu, Hawaii. American Institute of Aeronautics and Astronautics, 2012.
31. ICAO: Environmental Technical Manual, Volume I – Procedures for the Noise Certification of Aircraft. Standard, International Civil Aviation Organization, 2018.
32. ICAO: Doc 4444, Procedures for Air Navigation Services – Air Traffic Management, 16th ed. International Civil Aviation Organization, 2016.
33. M. Berzborn, R. Bomhardt, J. Klein, J.-G. Richter, M. Vorländer: The ITA-Toolbox: An Open Source MATLAB Toolbox for acoustic measurements and signal processing, in 43th Annual German Congress on Acoustics. Kiel, Germany, 2017, 222–225.

[https://www.routledge.com/downloads/Y105698/Y105698\\_Web\\_Download.zip](https://www.routledge.com/downloads/Y105698/Y105698_Web_Download.zip)

**Cite this article as:** Schäfer P. & Vorländer M. 2021. Atmospheric Ray Tracing: An efficient, open-source framework for finding eigenrays in a stratified, moving medium. Acta Acustica, 5, 26.