

## Computer Programs in Physics

## AFIS - A simulation framework for detection of aerosol fluorescence with integrating spheres

Julian Soltau <sup>a,b,\*</sup>, Arne Walter <sup>a</sup>, Frank Duschek <sup>a</sup>, Thomas Dekorsy <sup>c</sup><sup>a</sup> DLR, Institute of Technical Physics, Langer Grund, 74239, Hardthausen, Germany<sup>b</sup> AES Technology GmbH, Brauweg 40, 37073, Göttingen, Germany<sup>c</sup> DLR, Institute of Technical Physics, Pfaffenwaldring 38–40, 70569, Stuttgart, Germany

## ARTICLE INFO

Editor: Dr. J Ballantyne

## Keywords:

Aerosols  
Fluorescence  
Integrating sphere  
Monte Carlo  
Ray-Tracing  
Python

## ABSTRACT

We present a new simulation framework for the detection of aerosol fluorescence with integration spheres. Utilizing a Monte Carlo based ray-tracing approach, aerosol fluorescence within integrating sphere setups is simulated from photon generation through laser excitation over interactions with the setup components to losses and finally detection. Through modular design, the position and number of openings, sensors, etc. can be freely configured. Therefore, potential experimental setups can be evaluated with regard to overall performance, bottlenecks can be identified and the impact of different component parameters determined.

## PROGRAM SUMMARY

*Program Title:* AFIS - Aerosol Fluorescence in Integrating Spheres*CPC Library link to program files:* <https://doi.org/10.17632/nj9dg3tr6d.1>*Licensing provisions:* BSD 3-clause*Programming language:* Python

*Nature of problem:* Measuring (bio-)aerosol fluorescence is a complex task, especially for thin aerosols. In order to evaluate new experimental setups utilizing an integrating sphere, simulation data is essential to assess which system configurations yield promising results. Therefore, a simulation environment capable of calculating the different interactions within the setup is necessary, ideally providing a high level of customizability for the simulated setups.

*Solution method:* The AFIS simulation framework utilizes a ray-tracing approach based on a classical Monte Carlo description of the involved processes. Through batch-wise processing and penalization the computational efficiency is increased.

## 1. Introduction

Aerosols play an important role in the daily life of humans, affecting them in an array of different ways like pollution, public health or climate change [1]. With the COVID-19 pandemic demonstrating the possible devastating effects of a pathogen transmitted via aerosols. Therefore, the development of reliable methods to measure bioaerosols is essential. But due to the inherent characteristics of aerosols, such as the small particle size and the low combined volume of the particles, which are not uniformly spread within the air, measuring aerosols is a complicated problem. Amongst others, optical methods are employed for the measurement of bioaerosols within the air, especially fluorescence [2]. While optical methods are generally limited by their signal strength, optical instruments can be used to amplify these signals. One such component is an integrating sphere [3], which is an established

tool for photometry or radiometry [4]. An integrating sphere is a hollow sphere with a highly reflective inner coating. A measurement inside provides the benefit of almost full radial collection of signal photons, compared to other measurement setups that can only cover a fraction of the spatial angles. This comes at the cost of losing any temporal information from the signal photons and possible absorption losses at the sphere's coating over the course of multiple reflections. Integrating spheres are used for fluorescence measurements to for example determine the quality of fruits and vegetables [5] or identify different types of human tissue [6]. These applications are well understood, with computational models and their experimental validation described in the literature [7,8]. All these applications have one thing in common: They use solid or bottled liquid samples. While it is possible to collect aerosols with an air sampler [9] or an impinger [10] and analyse the filter or the sampling liquid, a continuous monitoring needs either a measurement

\* Corresponding author.

E-mail addresses: [julian.soltau@dlr.de](mailto:julian.soltau@dlr.de) (J. Soltau), [arne.walter@dlr.de](mailto:arne.walter@dlr.de) (A. Walter), [frank.duschek@dlr.de](mailto:frank.duschek@dlr.de) (F. Duschek), [thomas.dekorsy@dlr.de](mailto:thomas.dekorsy@dlr.de) (T. Dekorsy).<https://doi.org/10.1016/j.cpc.2025.110008>

Received 29 July 2025; Received in revised form 8 December 2025; Accepted 16 December 2025

Available online 18 December 2025

0010-4655/© 2026 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

performed directly on the gaseous sample or a continuous extraction and measurement loop. In both cases changes to the experimental setup are required, compared to the fixed probe setups used for solid samples.

In order to optimize such setups, modelling is required. This is necessary due to the combination of processes like fluorescence [11], elastic scattering [12], diffuse reflection [13] and re-absorption, which feature varying interaction probabilities and directional profiles. This leads to a convolution of effects depending on propagation direction, polarization and wavelength of the photons. Adding the influence of the components' placement on the sphere's surface on top of this makes it very difficult to determine optimal geometries. But these are required as the sphere's properties need to be defined in advance and cannot be changed later without replacing the entire sphere. Therefore, a previous assessment of different geometries is necessary. Due to the variety of design parameters and their various ranges, this leads to a large number of possible geometries, which need to be assessed. Thus, a tool is required that has the capability to quickly provide a reliable analysis of the characteristics for a given geometry. Furthermore, it needs to allow for a quick changes to the geometry. A calculation in closed form would be ideal, but due to the heavily stochastic nature of the involved processes it is not feasible. Therefore, computational models are favoured, in particular Monte Carlo approaches.

Monte Carlo methods are a well-established method for ray-tracing applications. Through their random sampling of stochastic distributions, they converge towards solutions with an excellent accuracy over increasing sample sizes. This accuracy in combination with their straightforward implementation and their capabilities for parallel computation make them a popular choice for various applications that require the modelling of photon behaviour [14–19]. A Monte Carlo model for the optimization of integrating sphere setups used for aerosol fluorescence measurements needs to describe both the aerosol properties and interaction as well as the integrating sphere. To the authors' knowledge such a model is not available in the literature. In the case of aerosols, previous work exists for scattering interactions [20,21], also with different applications for atmospheric aerosols [22,23], but not for aerosol fluorescence. Fluorescence on the other hand is well modelled for the analysis of tissue samples [24,25], medical imaging [26,27] or elementary particle analyses [28,29], but not for aerosols. Lastly, detailed models exist for investigations of the optical properties of a sample [7,30], but these only apply to solid or liquid samples. Similarly, commercial ray-tracing software, such as FRED [31], is focused on the simulation of interactions with solid objects. Therefore, the AFIS simulation framework was newly developed to combine all of these aspects into a single tool that can be used to design corresponding experimental setups. The simulation framework provides the possibility to study the interaction of aerosols and photon within integrating spheres. Through the use of the established Monte Carlo approach, photon distributions for custom defined geometries can be simulated. This provides researchers with a tool to model the fluorescence of aerosols in integrating spheres for the first time, allowing them to quickly analyse different geometries and gain insights into the most important design parameters for their laboratory setups, by providing information on the viability of individual experimental setups as well as the option perform an in-depth optimization.

## 2. Experimental background

Fig. 1 shows the basic form of an experimental setup, which consists of an integrating sphere with two openings for an aerosol stream and one opening for a sensor used to measure the fluorescence signal. The aerosol stream is guided to the openings on opposite sides of the sphere and creates an interaction region between the openings. This interaction region is illuminated by a laser that is directed through the same openings to excite the aerosol particles, which leads to the emission of fluorescence photons. These photons then propagate inside the sphere. During this propagation they are reflected at the sphere's wall and po-

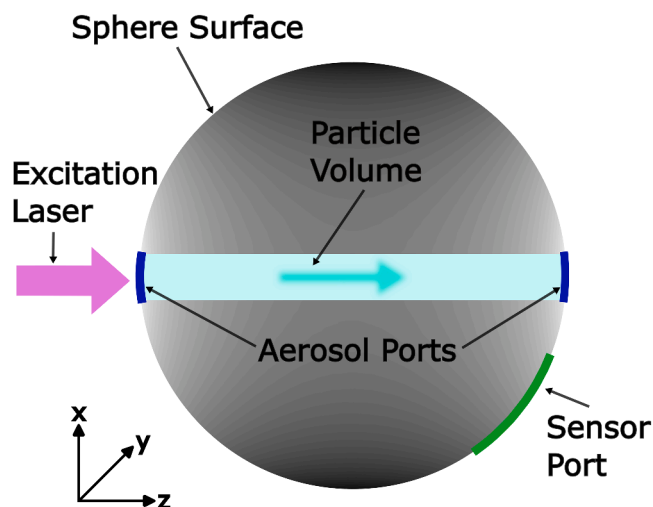


Fig. 1. The basic form of the experimental setup. An integrating sphere with two openings for the aerosol (blue), a laser used to excite the particles in the aerosol stream and a sensor (green) placed at an opening of the sphere's surface. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

tentially cross the interaction region again leading two secondary interactions. The propagation continues until they either exit the sphere through one of the openings, are absorbed or measured by the detector.

## 3. Implementation and architecture

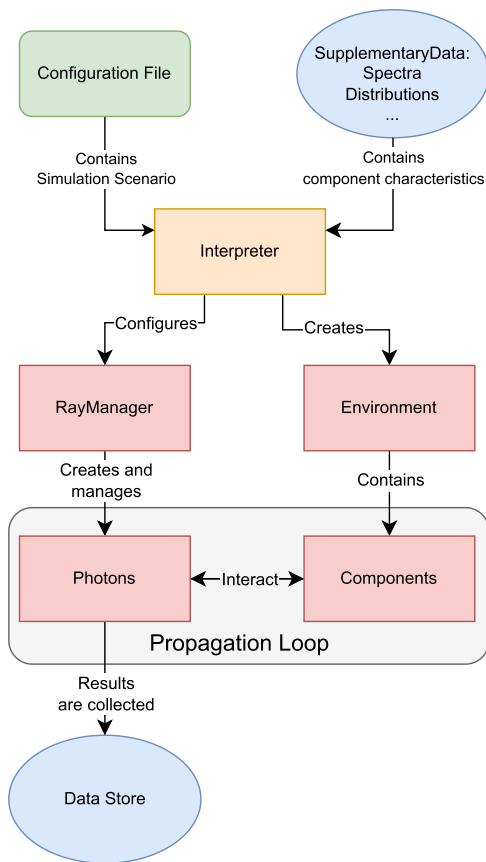
The purpose of the AFIS framework is to provide the necessary insights into the behaviour of the photons required to evaluate possible experimental setups. Therefore, the implementation follows an object-oriented approach, which allows for a modular design of the simulated environments. Thus, making it easy to modify components of the environment.

The complete simulation framework was implemented in Python. Both the dynamic typing and the support for multi-type variables were valuable in the design of the classes and provide means for handling of characteristics such as spectral data. Furthermore, it makes the code generally easy to read. Additionally, existing packages are used for the implementation of the physical processes involved in this simulation.

For the interactions between photons and geometric components a classical Monte Carlo approach was chosen, matching their statistical nature. Additionally, the framework provides processing options for photons originating from elastic scattering and fluorescence events, due to their different initial directional spread. This is necessary for the evaluation of an experimental setup, as the elastic scattering is expected to be the main contribution for background signals and has therefore to be considered for calculations of the signal-to-noise ratio.

### 3.1. Flow diagram

The full procedure of one simulation run is shown in Fig. 2. The interpreter is provided with a configuration file that contains all the settings for the run as well as the specifications for the components that need to be fetched from the supplementary data. Supplementary data contains information such as spectra or size distributions. Both the configuration file and the supplementary data are in JSON format. Utilizing this information, the components of the environment are created before the RayManager is instantiated. The RayManager is the handler for all photons and controls their creation, propagation and termination. After the simulation has reached the set number of photons the data is written to a compressed JSON data store.



**Fig. 2.** The structure of a simulation run within the framework. The simulation scenario is defined through a configuration file that the interpreter uses to setup the simulation. Based on the supplementary data, the environment is built and then utilized for the propagation loop of the photons. The results are collected in a data store, which is output at the end of the simulation.

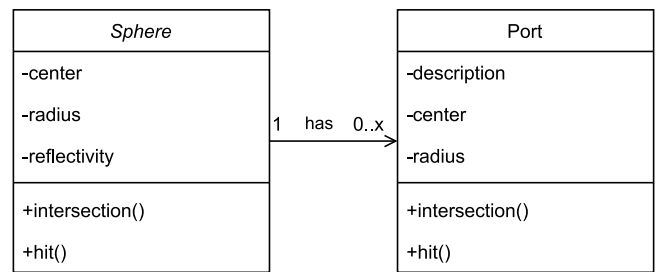
### 3.2. Environment

The environment is the collection of all components within a run of the simulation. This includes the integration sphere, its openings (ports) and the particle volume. Generally, all components have properties used to configure their characteristics as well as an “intersection”- and a “hit”-method. The “intersection”-method is used to calculate the intersection points of a propagating photon and a component, while the “hit”-method is used to compute the interaction between them. The implementation of the methods is individual for all components and will be described in the following sections.

As the whole setup is based on the integrating sphere, it is also used to define the coordinate system. Cartesian coordinates are used with the origin placed at the centre of the sphere and the z-axis aligned with the particle volume. The particle volume is a cylinder along the z-axis, which represents the volume where the aerosol particles are illuminated. At both ends of the particle volume the sphere has ports used to guide the aerosol stream through the sphere. The final component are the sensor ports, which represent the detectors in an experiment.

#### 3.2.1. Integrating sphere

With the sphere being centred around the origin, the only physical parameters needed are the diameter of the sphere and the coating of its interior, which defines the wavelength-dependent reflectivity of its wall. Additionally, it contains a list of all ports that are placed on its surface, as shown in Fig. 3. The calculation of the intersection points is done via a quadratic equation as the propagation vector of a photon within a sphere always has two intersections points with the sphere and therefore



**Fig. 3.** The class diagrams of the sphere and the ports, containing their attributes and methods. As the ports are a physically part of the sphere, they are directly associated to the sphere.

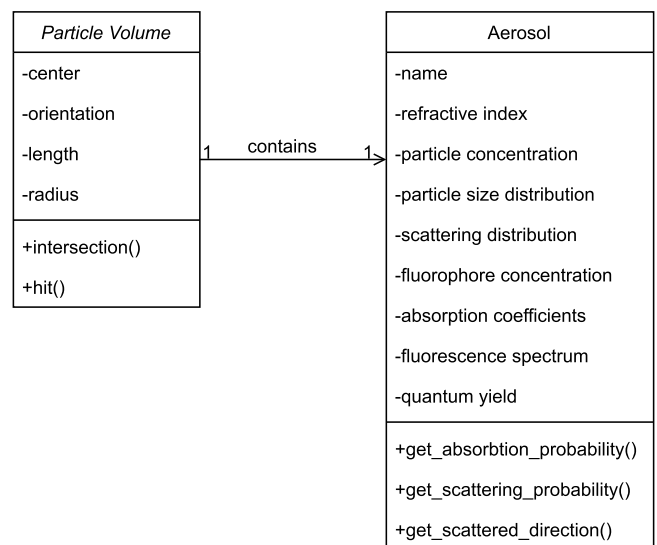
yields two solutions. Of these two solutions the one in the propagation direction is picked. For the interaction an evaluation between absorption and reflection has to be performed as a first step, with the probabilities based on the reflection efficiency at the photon’s wavelength defined through the coating’s characteristics. In the case of a reflection the new direction is sampled from a cosine distribution, due to the Lambertian scattering on the sphere’s surface.

#### 3.2.2. Particle volume

The particle volume covers the space of the aerosol and describes its interactions with the photons. Its position and size are defined by the aerosol ports. Additionally, it carries all properties of the aerosol, as shown in Fig. 4. The calculation of the intersections between the particle volume and the photons is also calculated via a quadratic equation, but in contrast to the sphere, the existence of two solutions is not guaranteed. Therefore, additional cases with less solutions have to be considered and limitations to solutions within the sphere are implemented. Furthermore, the distance travelled within the volume is calculated as a necessary parameter for the probability calculations of interactions within the volume. While the particle volume contains the method for the interactions, its execution requires methods from the aerosol class.

$$P(d) = \sigma[\lambda] \cdot \rho \cdot d \tag{1}$$

The aerosol class contains the properties of the aerosol, like the particle and fluorophore concentration as well as the distribution of particle



**Fig. 4.** The class diagrams of the particle volume and the aerosol, containing their attributes and methods. While the particle volume mostly defines the space containing the aerosol, the detailed data of the aerosol is carried by the aerosol class.

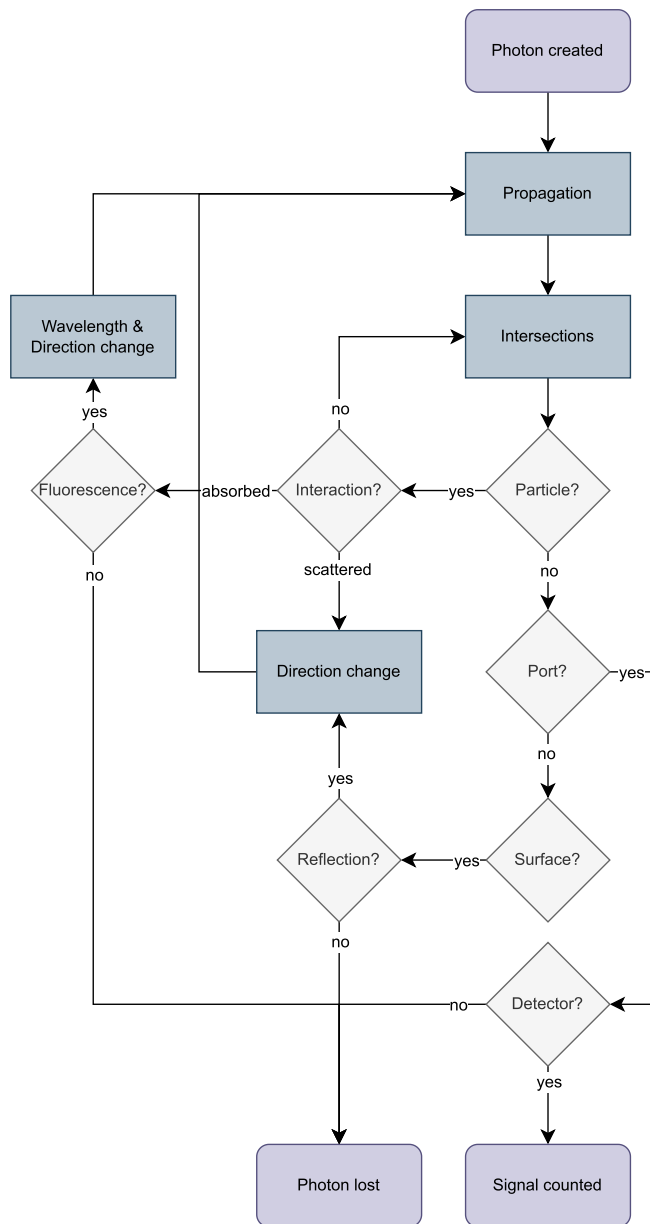


Fig. 5. The propagation loop of the photons. After their creation they are continuously propagating on a straight line until the path has an intersection with a component of the environment. The interaction with the components can result in the photon to continuing its propagation on the same path or may change the path. The propagation ends when a photon interacts with a port or is absorbed by a different component.

sizes, absorption coefficients, etc. It also provides the methods to calculate the interaction probabilities. These are calculated according to Eq. 1, with the distance  $d$  that the photon travels within the volume, the wavelength-dependent interaction parameter  $\sigma_i$  and the respective particle concentration  $\rho_i$  along the path. While  $d$  is obtained from the intersections of the photon with the particle volume, the two other parameters are either directly obtained from the aerosols properties or calculated. The calculated interaction probabilities are then used by the particle volume to determine if photons interact with the aerosol. The different interactions for absorption, fluorescence and Mie scattering can then be evaluated. All calculations regarding Mie scattering utilize the implementation of the *miepython* package [32]. For the implementation of both absorption and fluorescence, the properties of the aerosol particles are utilized. Starting with the wavelength-dependent absorp-

tion probability, an absorption event is assessed. If an absorption occurred, it is followed by an evaluation of a possible fluorescence event through the particle's fluorescence quantum yield. The wavelength of fluorescence photons is then determined based on the fluorescence spectrum, supplied through the supplementary data, of the aerosol particles.

### 3.2.3. Ports

All openings on the sphere are classified as ports. These can be the openings used to guide the aerosol through the sphere, sensors placed on the spheres surface or any other opening. Their class diagram is also contained in Fig. 3. They are defined by their centre and the radius of the opening and have an additional description to distinguish between different ports. As the intersection between a port and a photon corresponds to a vector-plane intersection, the intersection point is calculated through solving the plane equation and limiting solutions to the area of the port. If such a solution exists, the photon is considered to have hit the port. In this case the photon is tagged with the port and moved to the data store.

### 3.2.4. Light source

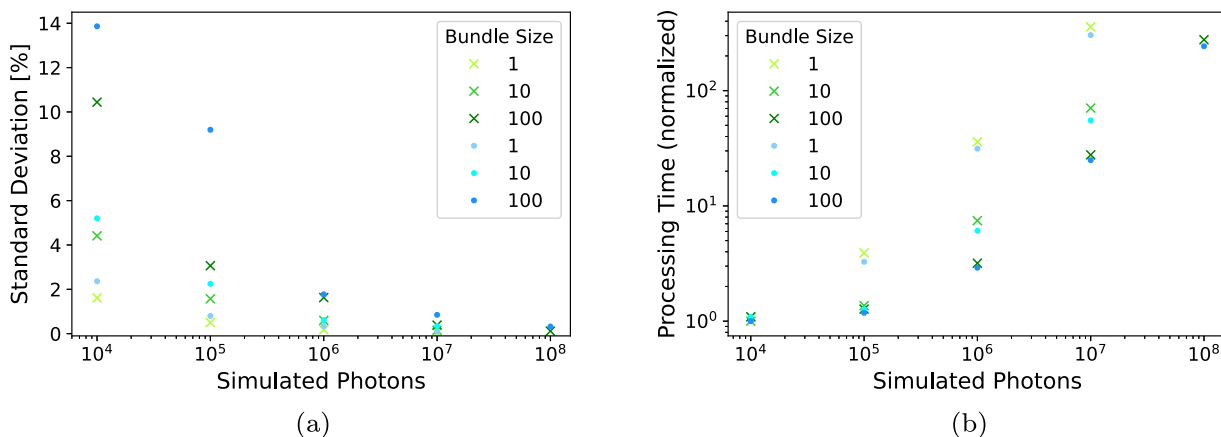
While not directly included in the integrating sphere setup, the illumination also needs to be defined. The light source specifies the wavelength, luminous power and repetition rate used for the illumination. The illumination in its current state was implemented with LED light sources in mind and is thus not polarized. Furthermore, integrating spheres have proven to be depolarizing, reducing the polarization of light reflected within [33]. Thus, making polarization effects less impactful. In order to switch to a polarized light source, the calculations in the Mie code would have to be exchanged for their polarized counterparts, also provided by *miepython*, and the directional profile of the fluorescence emission has to be changed.

### 3.3. Photon propagation

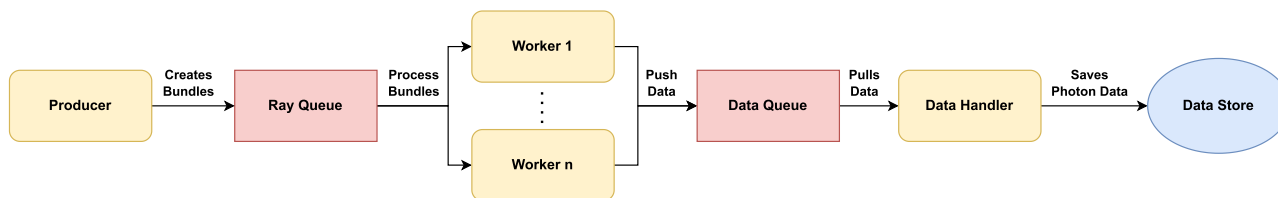
Utilizing the previously established environment, photons follow their propagation loop, shown in Fig. 5, and access the methods provided by the components of the environment. The propagation loop starts after the photon's creation and terminates when the photon leaves the sphere or is absorbed by one of the components. During the propagation the photon moves along a straight line until it has an intersection point with a component of the environment. The intersections between photons path and the components of the geometry are checked and resulting interactions are calculated in the following order: Interaction volume, ports and then the sphere. This is done according to the logical order in which a photon would intersect with them, if all of them are aligned on its path. Depending on the involved component the resulting interactions are calculated. Currently, this includes reflections at the walls of the sphere, Mie scattering, absorption and fluorescence by aerosol particles as well as the detection by a sensor. As long as the interaction does not end its propagation, the photon continues this loop.

### 3.4. Bundling of photons

Instead of only propagating photons individually, AFIS offers the option to propagate them as batches. This means that the calculations are performed for a group of multiple photons rather than single ones. Within one bundle all photons share a common position and direction, while each photon has an individual wavelength drawn from the respective spectrum. This requires only one calculation for the movement of all photons within the bundle instead of calculating the movement for every single photon and therefore lowers the overall number of calculations. For the interaction between the photons and other elements, individual calculations are performed if these processes depend on the individual wavelengths. Should an interaction cause a photon to behave differently than the other photons in the bundle, e.g. it is scattered by the aerosol, while the others are not, it is removed from the bundle



**Fig. 6.** Visualisation of the impact of splitting the simulated photons into photon bundles. Fluorescence (green) and scattered (blue) photons are considered separately as their different directional components have a relevant impact on both aspects. The case of  $10^8$  photons was only simulated for a bundle size of 100 photons. (a) The standard deviation of the registered photons on a component over a set of 10 simulation runs for different bundle and sample size combinations. (b) The processing time for different bundle and sample size combinations normalized to the fastest processing time. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



**Fig. 7.** Schematic description of the data flow through the different sub-processes (yellow) and data pipelines (red) within the RayManager. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

and propagated individually afterwards. This option allows for a more thorough investigation of photon paths as using bundles reduces the probability of a path being ended by a single event. Furthermore, at the cost of a reduced spatial granularity of the photon paths, the sample size can be increased without increasing computation time in the same manner.

To illustrate this, multiple simulation runs with a simple geometry containing only one sensor with the same diameter as the aerosol port were performed for different sample sizes and bundle sizes. The sample size refers to the total number of simulated photons within a run, while the bundle size defines how many photons are contained within one bundle. For each combination of the sample and bundle size, the simulation was run ten times and the number of photons measured at the detector is calculated. Afterwards, the standard deviation of the measured photons over the ten runs is calculated. This is done separately for scattered and fluorescence photons, as they behave differently, based on their initial directions. This is shown in Fig. 6a. Additionally, in Fig. 6b the processing time for the different combinations is shown, normalized to the smallest processing time, which is identical for all settings with a sample size of 1000 photons.

As these plots show, the use of photon bundles can increase the sample size and therefore also reduce the statistical fluctuations, without increasing the computation time at the same scale as the sample size. Based on this a trade-off between accuracy and computation time can be estimated, which provides the possibility to choose a combination of simulated photons and their bundle size depending on the desired accuracy of the simulation and the available computational resources. This allows researchers to get a first impression within minutes, as a single setup can be evaluated in less than two minutes with an error below 2%. In case of a more detailed analysis the error can be reduced to 1%, with the required time increasing to 75 minutes. Thus the simulation framework provides the possibility to perform a quick assessment over a wide set of geometries followed by a detailed analysis of the most

promising candidates. All with the computational power of a normal office laptop. Through the use of more computational power these times can be substantially increased.

### 3.5. Parallel processing

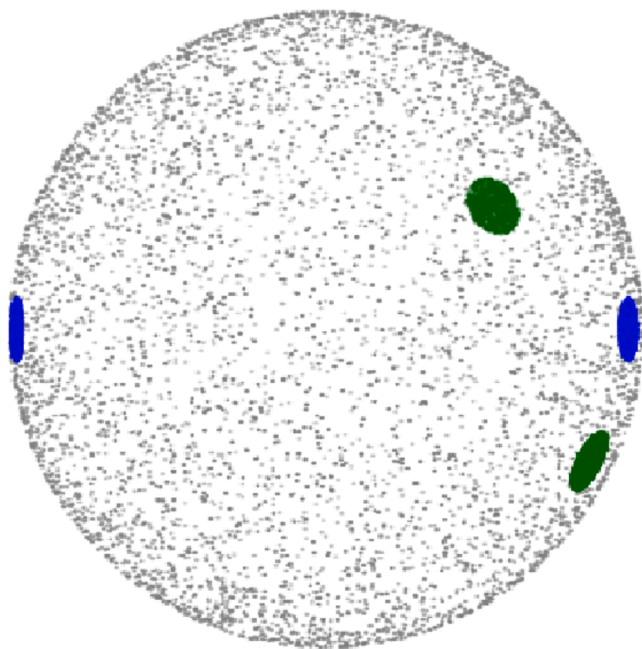
Another step to decrease the processing time, is the implementation of parallel processing for the propagation loop of the photons. As the individual photons or bundles are independent of each other, they can also be processed independently. Therefore, the photons can be submitted to a pool of worker processes, which process their assigned photons without need for synchronisation. All of these processes are controlled by an instance of the RayManager class, which utilizes different sub-processes, as visualized in Fig. 7. The first stage is the producer that is tasked to populate the ray queue with ray bundles until the defined number of simulated photons is reached. These are then pulled from the queue by the worker processes that calculate their propagation within the geometry. Each worker has its own sub-process, to avoid being limited by Python's global interpreter lock. After a worker finishes the calculation of one set of photons it draws a new one from the queue. All photons, whose calculation are finished, are placed in the data queue. There they are fetched by the data handler, who sorts and stores them in the data store.

## 4. Application examples

The data generated from the AFIS simulation framework, provides insights into different aspects of the simulated geometry. Three examples of obtainable information are shown in the following.

### 4.1. Photon distribution

Arguably the most important characteristic when evaluating an experimental setup is the expected signal, as this the key metric for its



**Fig. 8.** Visualization of the photon distribution from one simulation run. The black dots correspond to photons absorbed by the sphere and the green areas are composed of the photons counted by the sensors, while the blue areas (entrance to the left, exit to the right) are the photons leaving the sphere through an aerosol port. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

feasibility. On this basis different setups can be compared to see which one provides the best signal. But for a general understanding the distribution of photons over all components is of interest as well, as this allows to quantify where signal losses originate from and thus allows to minimize losses. Based on the data provided from the framework these values can easily be obtained. Table 1 shows the distribution of the photons in an exemplary run of the simulation with an environment that has two opposite sensors, each being 45° from the aerosol exit. As seen in

**Table 1**

Distribution of the photons over the components in an exemplary simulation run. The area is the fraction of the surface area covered by the component, while the fluorescence/scattered photons show the fraction of corresponding photons absorbed/registered by that component.

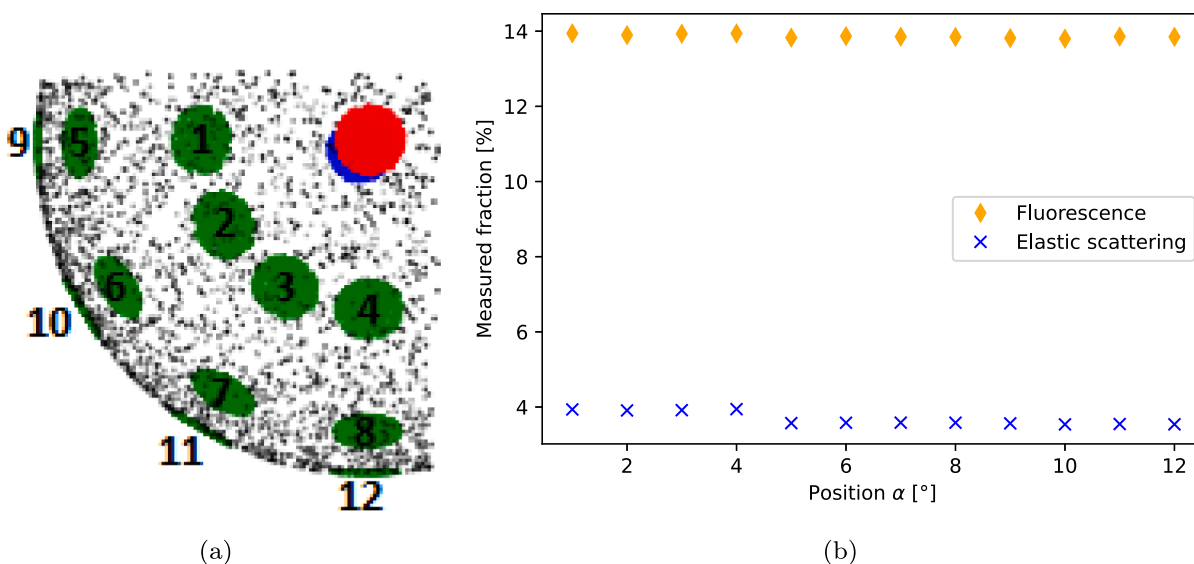
Component	Area	Fluorescence photons	Scattered photons
Sphere	99 %	(47.9 ± 0.1) %	(67 ± 0.1) %
Particle Volume	-	(0 ± 0.1) %	(0 ± 0.1) %
Entrance	0.25 %	(14 ± 0.1) %	(6.9 ± 0.1) %
Exit	0.25 %	(13.9 ± 0.1) %	(19.1 ± 0.1) %
Sensor 1	0.25 %	(12.1 ± 0.1) %	(3.5 ± 0.1) %
Sensor 2	0.25 %	(12.1 ± 0.1) %	(3.5 ± 0.1) %

the table, aerosol and sensor ports have the same size but register different fractions of the fluorescence and scattered photons. This is due to proximity effects and the directional profile of Mie scattering. Based on this data a setup can be quickly evaluated regarding expected signal and background levels.

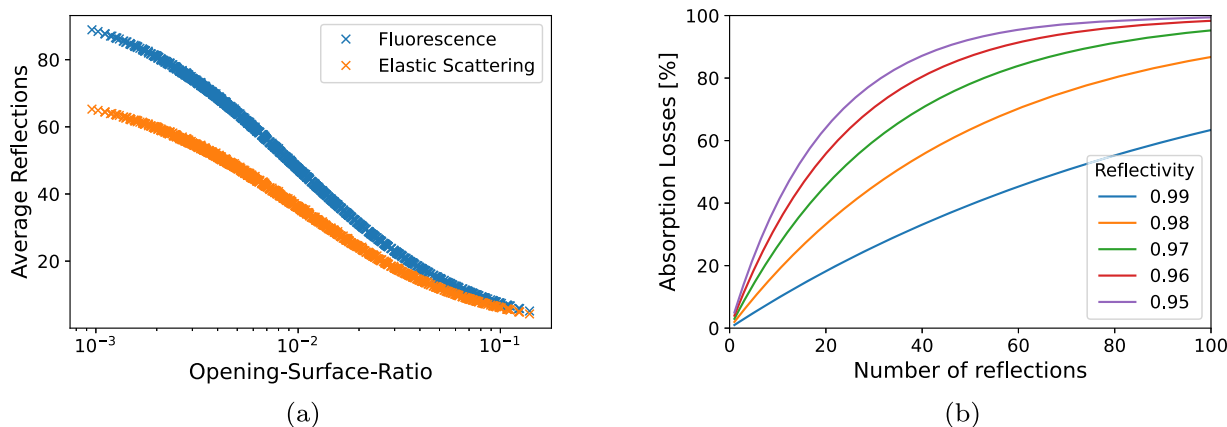
But the data obtained from the simulation can also be used to provide more in-depth insights into the photon distribution. Due to the data store containing data on all individual photons from the simulation run, a detailed display of the photon distribution is possible, as shown in Fig. 8. This plot shows the final position of all photons from the simulation run and shows where photons were absorbed or registered by a port. Thereby, an analysis of spatial effects is possible beyond the numerical distribution.

#### 4.2. Sensor placement

With the capabilities of calculating the distribution of the photons over the different components, one of the most important questions is the effect of the sensor placement on to the number of detected photons. Fig. 9 shows a comparison of different sensor positions. For this comparison a set of simple geometries with a single sensor was used. The geometries were identical for all parameters except the sensor position, which was varied in steps of 30°. As one would expect, the sensors with the same angle along the z-axis show no difference beyond statistical fluctuations due to the symmetry of the processes involved. For the fluorescence photons the different angles along the t-axis also show no



**Fig. 9.** Comparison of the detected signal at different sensor positions. (a) The different sensor positions. The positions are placed in 30° increments, both along the z-axis and in the plane perpendicular to it. The sensors are visualized in green, while the aerosol entrance is blue and the aerosol exit is red. (b) The detected fraction of photons both for fluorescence and elastic scattering at each of the sensor positions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



**Fig. 10.** Combining the average reflections of a setup with the reflectivity of the coating the average absorption losses at the sphere's surface can be estimated. (a) The average amount of times a photon is reflected inside the sphere based on the ratio between openings and the sphere's surface. The underlying data set covers a variation of sphere sizes, port sizes and port numbers while using the same sphere coating. (b) The relation between the number of reflections and the absorption losses based on the reflectivity, for different reflectivities.

relevant differences. In the case of the scattered photons on the other hand a notable difference can be seen. While this difference is small in absolute numbers, the relative difference is significant. This is in line with the scattering characteristics of Mie scattering.

#### 4.3. Average reflections

The average number of times a photon is reflected inside the sphere is a value that can provide insights into the absorption losses, but is difficult to calculate due to the involved dependencies of wavelengths and directions. As there is no coating with a 100% reflectivity, an increase of reflections always corresponds to an increase of absorption losses. Fig. 10a shows the average reflections based on the ratio between openings in the sphere and the sphere's total surface area obtained from a large set of simulation runs with the same sphere coating. As an example Fig. 10b shows the relationship between the number of reflections and the absorption losses for different reflectivities. Together they highlight the impact absorption losses can have and the usefulness of the average number of reflections to determine them.

#### 5. Summary

In this paper we introduced the AFIS simulation framework aimed at investigating the behaviour of photons resulting from aerosol fluorescence within integrating spheres. In its current form it provides insights into the behaviour of photons originating from either fluorescence or Mie scattering caused by the aerosol's particles. To the authors' knowledge this is the first simulation framework providing these kind of insights. Through the framework, the distribution of the photons within the integrating sphere can be calculated. Therefore, the expected signal for an experimental setup can be estimated. With this data produced by the simulation framework, both quick assessments and in-depth analysis can be performed for the design of integrating sphere setups. This makes it a powerful tool for the analysis of experimental setups and allows to study the influence specific components have on to the expected signal levels. Through three application examples an impression of the offered possibilities was shown, such as a detailed analysis of the optimal sensor position, which is a key aspect for designing an integrating sphere setup. Furthermore, the AFIS code is not limited to the given examples. Due to the modular design, changes to the simulated geometry can be performed without much effort and different parameters for components and aerosols can be tested. Furthermore, additional effects or components can be added to the code if necessary and are easy to integrate into the existing framework. This allows researchers to tailor it towards their research questions. Thus, making it a practical new tool for researches to

design experimental setups used in the investigation of aerosol particles through integrating spheres.

#### CRedit authorship contribution statement

**Julian Soltau:** Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Conceptualization; **Arne Walter:** Writing – review & editing, Methodology, Conceptualization; **Frank Duschek:** Writing – review & editing, Supervision, Project administration, Conceptualization; **Thomas Dekorsy:** Writing – review & editing, Supervision, Funding acquisition.

#### Data availability

Data will be made available on request.

#### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### References

- [1] J.H. Seinfeld, S.N. Pandis, *Atmospheric chemistry and physics: From air pollution to climate change*, Wiley-Blackwell, 3 edition, 2016.
- [2] Z. Huang, X. Yu, Q. Liu, T. Maki, K. Alam, Y. Wang, F. Xue, S. Tang, P. Du, Q. Dong, D. Wang, J. Huang, Bioaerosols in the atmosphere: A comprehensive review on detection methods, concentration and influencing factors, *Sci. Total Environ.* 912 (2024) 168818. <https://doi.org/10.1016/j.scitotenv.2023.168818>
- [3] J.A. Jacquez, H.F. Kuppenheim, Theory of the Integrating Sphere, *J. Opt. Soc. Am.* 45 (6) (1955) 460–470. <https://doi.org/10.1364/JOSA.45.000460>
- [4] L. Hanssen, Integrating-sphere system and method for absolute measurement of transmittance, reflectance, and absorbance of specular samples, *Appl. Opt.* 40 (19) (2001) 3196–3204. <https://doi.org/10.1364/AO.40.003196>
- [5] R. Lu, R. Van Beers, W. Saeys, C. Li, H. Cen, Measurement of optical properties of fruits and vegetables: a review, *Postharvest Biol. Technol.* 159 (2020) 111003. <https://doi.org/10.1016/j.postharvbio.2019.111003>
- [6] A.u. Rehman, I. Ahmad, S.A. Qureshi, Biomedical Applications of Integrating Sphere: A Review, *Photod. Photodyn. Ther.* 31 (2020) 101712. <https://doi.org/10.1016/j.pdpdt.2020.101712>
- [7] F. Foschum, F. Bergmann, A. Kienle, Precise determination of the optical properties of turbid media using an optimized integrating sphere and advanced Monte Carlo simulations. Part 1: theory, *Appl. Opt.* 59 (10) (2020) 3203–3215. <https://doi.org/10.1364/AO.386011>
- [8] F. Bergmann, F. Foschum, R. Zuber, A. Kienle, Precise determination of the optical properties of turbid media using an optimized integrating sphere and advanced Monte Carlo simulations. Part 2: experiments, *Appl. Opt.* 59 (10) (2020) 3216–3226. <https://doi.org/10.1364/AO.385939>
- [9] E. Montilla, S. Mogo, V. Cachorro, A. de Frutos, An integrating sphere spectral system to measure continuous spectra of aerosol absorption coefficient, *J. Aerosol Sci.* 42 (3) (2011) 204–212. <https://doi.org/10.1016/j.jaerosci.2011.01.003>

- [10] T. šantl Temkiv, P. Amato, U. Gosewinkel, R. Thyraug, A. Charton, B. Chicot, K. Finster, G. Bratbak, J. Löndahl, High-Flow-Rate Impinger for the Study of Concentration, Viability, Metabolic Activity, and Ice-Nucleation Activity of Airborne Bacteria, *Environ. Sci. Technol.* 51 (19) (2017) 11224–11234. <https://doi.org/10.1021/acs.est.7b01480>
- [11] J.R. Lakowicz (Ed.), *Principles of Fluorescence Spectroscopy*, Springer US, Boston, MA, Boston, MA, 2006. <https://doi.org/10.1007/978-0-387-46312-4>
- [12] C.F. Bohren, D.R. Huffman, *Absorption and scattering of light by small particles*, A Wiley-Interscience publication, Wiley-CH Verlag GmbH & Co KGaA, New York, New York, 2004.
- [13] J.-H. Lambert, *Photometria sive de mensura et gradibus luminis, colorum et umbrae, Sumpibus viduae Eberhardi Klett, typis Christophori Petri Detleffsen*, 1760. <https://doi.org/10.3931/e-rara-9488>
- [14] Q. Fang, Mesh-based Monte Carlo method using fast ray-tracing in Plücker coordinates, *Biomed. Opt. Express* 1 (1) (2010) 165–175. <https://doi.org/10.1364/BOE.1.000165>
- [15] H.W. Jensen, N.J. Christensen, Photon maps in bidirectional Monte Carlo ray tracing of complex objects, *Comput. Graph.* 19 (2) (1995) 215–224. [https://doi.org/10.1016/0097-8493\(94\)00145-O](https://doi.org/10.1016/0097-8493(94)00145-O)
- [16] T.-M. Li, M. Aittala, F. Durand, J. Lehtinen, Differentiable Monte Carlo ray tracing through edge sampling, *ACM Trans. Graph.* 37 (6) (2018) 222:1–222:11. <https://doi.org/10.1145/3272127.3275109>
- [17] E.R. Freniere, G.G. Gregory, R.A. Hassler, Edge diffraction in Monte Carlo ray tracing, in: *Optical Design and Analysis Software*, 3780, SPIE, 1999, pp. 151–157. <https://doi.org/10.1117/12.363773>
- [18] Z. Liu, K. Wang, X. Luo, S. Liu, Precise optical modeling of blue light-emitting diodes by Monte Carlo ray-tracing, *Opt. Express* 18 (9) (2010) 9398–9412. <https://doi.org/10.1364/OE.18.009398>
- [19] M.I. Disney, P. Lewis, P.R.J. North, Monte Carlo ray tracing in optical canopy reflectance modelling, *Remote Sens. Rev.* (2000). <https://doi.org/10.1080/02757250009532389>
- [20] A. Macke, M.I. Mishchenko, K. Muinonen, B.E. Carlson, Scattering of light by large nonspherical particles: ray-tracing approximation versus T-matrix method, *Opt. Lett.* 20 (19) (1995) 1934–1936. <https://doi.org/10.1364/OL.20.001934>
- [21] Y.M. Govaerts, M.M. Verstraete, Raytran: a Monte Carlo ray-tracing model to compute light scattering in three-Dimensional heterogeneous media, *IEEE Trans. Geosci. Remote Sens.* 36 (2) (1998) 493–505. <https://doi.org/10.1109/36.662732>
- [22] G.N. Plass, G.W. Kattawar, Monte Carlo Calculations of Light Scattering from Clouds, *Appl. Opt.* 7 (3) (1968) 415–419. <https://doi.org/10.1364/AO.7.000415>
- [23] G.W. Kattawar, G.N. Plass, J.A. Guinn, Monte Carlo Calculations of the Polarization of Radiation in the Earth's Atmosphere-Ocean System, *J. Phys. Oceanogr.* 3 (4) (1973) 353–372. [https://doi.org/10.1175/1520-0485\(1973\)003<0353:MCCOTP>2.0.CO;2](https://doi.org/10.1175/1520-0485(1973)003<0353:MCCOTP>2.0.CO;2)
- [24] J. Swartling, A. Pifferi, A.M.K. Enejder, S. Andersson-Engels, Accelerated Monte Carlo models to simulate fluorescence spectra from layered tissues, *JOSA A* 20 (4) (2003) 714–727. <https://doi.org/10.1364/JOSAA.20.000714>
- [25] Q. Liu, C. Zhu, N. Ramanujam, Experimental validation of Monte Carlo modeling of fluorescence in tissues in the UV-visible spectrum, *J. Biomed. Opt.* 8 (2) (2003) 223–236. <https://doi.org/10.1117/1.1559057>
- [26] J. Chen, X. Intes, Comparison of Monte Carlo methods for fluorescence molecular tomography—computational efficiency, *Med. Phys.* 38 (10) (2011) 5788–5798. <https://doi.org/10.1118/1.3641827>
- [27] V. Cuplov, I. Buvat, F. Pain, S. Jan, Extension of the GATE Monte-Carlo simulation package to model bioluminescence and fluorescence imaging, *J. Biomed. Opt.* 19 (2) (2014) 026004. <https://doi.org/10.1117/1.JBO.19.2.026004>
- [28] X. Llovet, E. Valovirta, E. Heikinheimo, Monte Carlo Simulation of Secondary Fluorescence in Small Particles and at Phase Boundaries, *Microchim. Acta* 132 (2) (2000) 205–212. <https://doi.org/10.1007/s006040050013>
- [29] M.D. Roberts, The role of atmospheric multiple scattering in the transmission of fluorescence light from extensive air showers, *J. Phys. G: Nucl. Part. Phys.* 31 (11) (2005) 1291. <https://doi.org/10.1088/0954-3899/31/11/012>
- [30] J. Jelken, T. Brall, P. Gelbing, F. Foschum, A. Kienle, Characterization of the Optical Properties of Photoluminescent Turbid Media Using an Integrating Sphere and Monte Carlo Simulations, *Materials* 17 (24) (2024) 6072. <https://doi.org/10.3390/ma17246072>
- [31] R.G. Irvin, Photon Engineering LLC: FRED optical engineering software by photon engineering: advancing the state of the art of optical computations, in: C. Chair (Ed.), *SPIE Exhibition Product Demonstrations, OP20EX*, International Society for Optics and Photonics, SPIE, 2020, p. OP20EX0V. <https://doi.org/10.1117/12.2580746>
- [32] S. Prah, miepython: Pure python implementation of Mie scattering, 2023, <https://doi.org/10.5281/zenodo.8218010>
- [33] S.C. McClain, C.L. Bartlett, J.L. Pezzaniti, R.A. Chipman, Depolarization measurements of an integrating sphere, *Appl. Opt.* 34 (1) (1995) 152–154. <https://doi.org/10.1364/AO.34.000152>