# AeroAcoustics.jl: A Julia package for aeroacoustics

Oliver Lylloff © [1]¶

1 Technical University of Denmark (DTU), Dept. of Wind and Energy Systems, Section for Airfoil and Rotor Design (ARD) ¶ Corresponding author

Summary

Aeroacoustics is an interdisciplinary research field situated between aerodynamics and acoustics. It focuses on understanding the sound generated by the motion of air around solid bodies and plays an important role in research areas such as automotive, aerospace, and wind energy sciences (Allen et al., 2002). However, the airflow responsible for sound generation typically generates unwanted background noise, which masks the true acoustic emission of the test item from measurement equipment. Therefore, isolating the sound emitted by the test item becomes essential to obtain reliable evaluations, which is pivotal in the development of noise-reducing technologies for cars, airplanes, and wind turbines. One common solution strategy involves the use of *acoustic images* computed using microphone arrays. The acoustic images help visualize and quantify the spatial distribution of sound across the test item, and enables precise assessment of acoustic emissions without contamination from background noise (Glegg & Devenport, 2017).

`AeroAcoustics.jl` is a Julia (Bezanson et al., 2017) package for working with microphone arrays and acoustic imaging techniques. Methods for computing acoustic images often share the same codebase and utility functions, making them suitable candidates for a modular software package. It is the intention, that `AeroAcoustics.jl` can serve as a foundation for benchmarking existing methods and developing new ones within the aeroacoustics community. In addition to that, `AeroAcoustics.jl` is also a good candidate for teaching and entry-level programmers due to its simple API.

Statement of need

The `AeroAcoustics.jl` package provides a collection of commonly used acoustic imaging techniques, developed by the aeroacoustic community, such as conventional beamforming (Johnson & Dudgeon, 1993), DAMAS (Brooks & Humphreys, 2006), and CleanSC (Sijtsma, 2007). The `AeroAcoustics.jl` package also includes newer techniques found in the literature, e.g., fista (O. Lylloff et al., 2015) and mean mutual-coherence weights (Amaral et al., 2018).

Only one other open-source software package exists with similar features; acoular (Sarradj & Herold, 2017), which is written in Python. Acoular has a broader scope than `AeroAcoustics.jl`, including the generation of simulated data, and analysis in both the time-domain and frequency-domain. A few repositories can be found online with code written in Matlab, but with limiting packaging functionality and few features; these do not constitute proper open-source software packages [1].

A benchmark case study was conducted in 2017, where different research groups with different software used the same datasets and the same algorithms to compare results (Bahr et al., 2017; Sarradj et al., 2017). One conclusion was that, although using seemingly the same algorithm written in different software codes, the results differed. The study did not compare the codes

---

[1] An overview can be found here: `beamforming-tools`

because most of the participants did not use publicly shared codes. This is a testament to the need for open-source software, and `AeroAcoustics.jl` provides yet another option for the community to adopt open-source practises.

`AeroAcoustics.jl` is the fast, entry-level open source software package, that fills the gap between the existing codebases available online. It has already been used in publications (Fischer et al., 2022, 2023; Plaza et al., 2022), a PhD thesis (O. A. Lylloff, 2020), for research and commercial activities at the Poul la Cour Wind Tunnel, and also in teaching 46600 Aeroacoustics.

Example

To use the methods included in `AeroAcoustics.jl`, an `Environment` struct must be defined. This holds essential information about the computational domain, microphone array geometry and measurement data. The `Environment` struct is used by most functions in the `AeroAcoustics.jl` package, which makes it easy to add new methods and algorithms because the required input is typically stored in `Environment`. Below is an example of this:

```
E = Environment(
    z0=z0,
    micgeom=micgeom,
    CSM=CSM,
    flim=(3000,4000),
    Nx=21,
    Ny=21,
    xlim=(-0.5,0.5),
    ylim=(-0.5,0.5),
    multi_thread=true
)
steeringvectors!(E)
b = beamforming(E)
```

where
- `z0` is the distance from the microphone array to the acoustic imaging plane,
- `micgeom` is the coordinates to each microphone in the microphone array,
- `CSM` is the cross-spectral matrix (measurement data),
- `flim` are the frequency limits of interest,
- `Nx` and `Ny` is the desired number of grid points in the x- and y-direction of the acoustic image
- `xlim` and `ylim` is the desired extent of the acoustic image
- `multi_thread` can enable or disable multi-threading in appropriate functions across the `AeroAcoustics.jl` package.

In this example, `steeringvectors!(E)` mutates the argument and adds steeringvectors to the `Environment`, which is required to compute acoustic images. The function is aware of the defined options in `Environment` and will only calculate for the appropriate domains, e.g., frequency limits, grid points, etc., reducing the computational load. Finally, the acoustic image is computed using `b = beamforming(E)`.

An example from a wind tunnel measurement is shown in Figure 1. Between the green walls, which are made of Kevlar and transparent to sound, an airflow encloses a section of an airfoil (shown in black), and acoustic measurements are conducted with a microphone array (blue dots) just on the other side of the green wall. The resulting acoustic image is overlayed the airfoil for visualization, using `Makie.jl` (Danisch & Krumbiegel, 2021), and shows, that the dominant part of the acoustic energy arises from the trailing edge of the airfoil (as expected from theory).
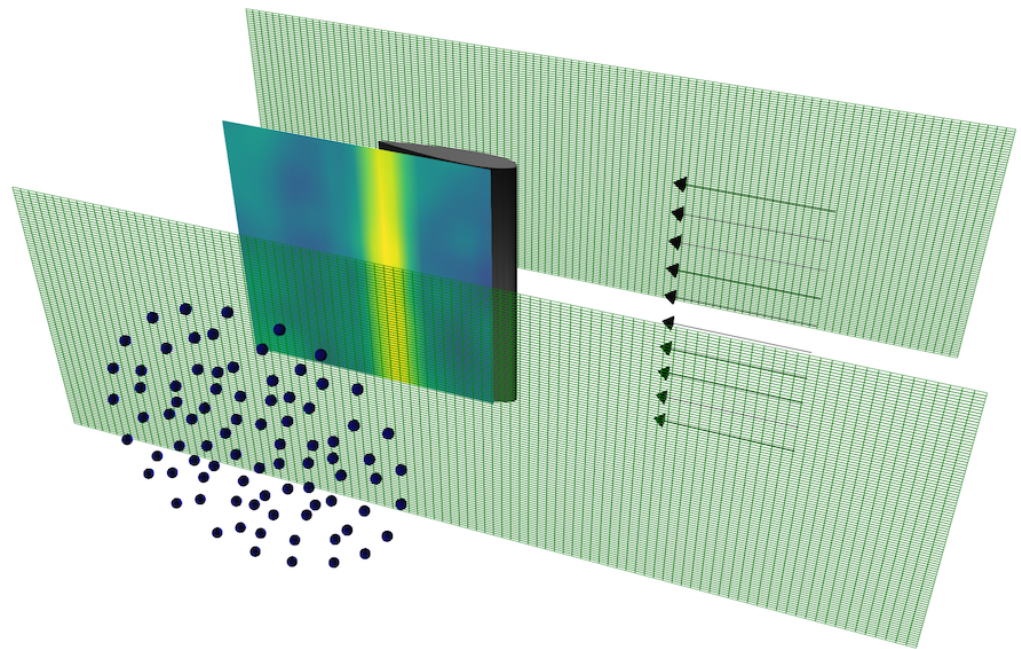
Figure 1: Example of beamforming in a wind tunnel

References

Allen, C. S., Blake, W. K., Dougherty, R. P., Lynch, D., Soderman, P. T., & Underbrink, J. R. (2002). Aeroacoustic measurements. In T. J. Mueller (Ed.), *Aeroacoustic Measurements*. Springer Science & Business Media. https://doi.org/10.1007/978-3-662-05058-3

Amaral, F. R. do, Himeno, F. H. T., Carmo Pagani, C. do, & Medeiros, M. A. F. de. (2018). Slat noise from an MD30P30N airfoil at extreme angles of attack. *AIAA Journal*, *56*, 964–978. https://doi.org/10.2514/1.J056113

Bahr, C. J., Humphreys, W. M., Ernst, D., Ahlefeldt, T., Spehr, C., Pereira, A., Leclère, Q., Picard, C., Porteous, R., Moreau, D. J., Fischer, J., & Doolan, C. J. (2017). A comparison of microphone phased array methods applied to the study of airframe noise in wind tunnel testing. *23rd AIAA/CEAS Aeroacoustics Conference, 2017*, 1–19. https://doi.org/10.2514/6.2017-3718

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Rev.*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Brooks, T. F., & Humphreys, W. M. (2006). A deconvolution approach for the mapping of acoustic sources (DAMAS) determined from phased microphone arrays. *Journal of Sound and Vibration*, *294*, 856–879. https://doi.org/10.1016/j.jsv.2005.12.046

Danisch, S., & Krumbiegel, J. (2021). Makie.jl: Flexible high-performance data visualization for Julia. *Journal of Open Source Software*, *6*(65), 3349. https://doi.org/10.21105/joss.03349

Fischer, A., Bak, C., Lylloff, O., Olsen, A. S., Mikkelsen, R. F., Ildvedsen, S. B., Beckerlee, J. S., Borgoltz, A., Kuester, M., & Intaratep, N. (2022). Cross validation of the aerodynamic and acoustic measurements in two kevlar-walled wind tunnels. *Journal of Physics: Conference Series*, *2265*, 022103. https://doi.org/10.1088/1742-6596/2265/2/022103

Fischer, A., Lylloff, O., Grande, E. F., & Bak, C. (2023). Analytical corrections for acoustic transmission in kevlar-walled wind tunnels. *AIAA Journal*, *61*, 3119–3133. https://doi.org/

10.2514/1.J062313

Glegg, S., & Devenport, W. (2017). *Aeroacoustics of low mach number flows: Fundamentals, analysis, and measurement* (pp. 1–537). Academic Press, Elsevier. ISBN: 9780128097939

Johnson, D. H. D. H., & Dudgeon, D. E. (1993). Array signal processing: Concepts and techniques. In *Book* (p. 533). Englewood Cliffs: PTR Prentice Hall. ISBN: 9780130485137

Lylloff, O. A. (2020). *Aeroacoustic wind tunnel tests* [PhD thesis, DTU Wind Energy]. https://doi.org/10.11581/dtu:00000102

Lylloff, O., Fernández-Grande, E., Agerkvist, F., Hald, J., Tiana Roig, E., & Andersen, M. S. (2015). Improving the efficiency of deconvolution algorithms for sound source localization. *The Journal of the Acoustical Society of America*, *138*(1), 172–180. https://doi.org/10.1121/1.4922516

Plaza, G. V. i, Fischer, A., Lylloff, O., Bak, C., Olsen, A. S., Bertagnolio, F., Luesutthiviboon, S., Pereira, L. T. L., Ragni, D., Avallone, F., Suryadi, A., & Herr, M. (2022, June). Benchmarking of the NACA 63 3 -018 trailing-edge noise in a broad reynolds number range as part of the IEA task 39. *28th AIAA/CEAS Aeroacoustics 2022 Conference*. https://doi.org/10.2514/6.2022-2981

Sarradj, E., & Herold, G. (2017). A Python framework for microphone array data processing. *Applied Acoustics*, *116*, 50–58. https://doi.org/10.1016/j.apacoust.2016.09.015

Sarradj, E., Herold, G., Sijtsma, P., Martinez, R. M., Malgoezar, A., Snellen, M., Geyer, T. F., Bahr, C. J., Porteous, R., Moreau, D. J., & Doolan, C. J. (2017, June). A microphone array method benchmarking exercise using synthesized input data. *23rd AIAA/CEAS Aeroacoustics Conference, 2017*. https://doi.org/10.2514/6.2017-3719

Sijtsma, P. (2007). CLEAN based on spatial source coherence. *International Journal of Aeroacoustics*, *6*, 357–374. https://doi.org/10.1260/147547207783359459