

# Offline coupling of segregated multi-physical simulations with consistent boundary conditions and source terms based on scattered data

Stefan Schoder

TU Graz, Aeroakustik und Vibroakustik, IGTE, Inffeldgasse 18, A-8010 Graz, AT, Email: stefan.schoder@tugraz.at

## Abstract

This article presents the openCFS submodule scattered data reader for coupling multi-physical simulations performed in different simulation programs. For instance, by considering a forward-coupling of a surface vibration simulation (mechanical system) to an acoustic propagation simulation using time-dependent acoustic absorbing material as a noise mitigation measure. The nearest-neighbor search of the target and source points from the interpolation is performed using the FLANN or the CGAL library. In doing so, the coupled field (e.g., surface velocity) is interpolated from a source representation consisting of field values physically stored and organized in a file directory to a target representation being the quadrature points in the case of the finite element method. A test case of the functionality is presented in the "testsuite" module of the openCFS software called "Abc2dcsvt". This scattered data reader module was successfully applied in numerous studies on flow-induced sound generation. Within this short article, the functionality, and usability of this module are described.

## Implementation of the scattered data reader

Within this contribution, we concentrate on the openCFS [13] module *scattered data reader*. The scattered data reader has several interpolation possibilities and datatype dependencies, ranging from spatially varying fields (so-called "csv") to spatially and temporally varying fields (so-called "csvt" type of scattered data reader). This csvt is a general implementation of field interpolation inside the finite element method core of openCFS and will be discussed in the following.

## File structure and data organization

Each point of the scattered data points is described by its coordinates (x,z,y), they are stored in the file name given in the user definition. A common (offline) approach to organize the input data is to use separate files. The organization of the files is done in a master file, pointing towards several subfiles. This master file contains information about the file path of the point location file and the time step value (physical time), and the path to the corresponding data file. It lists the timestamps that correspond to the data in the subsequent files. The file with the point locations contains the coordinates of the data points or locations (x,y,z). Each row typically represents a single point, and the columns correspond to the spatial dimensions in the Cartesian coordinate system (x,y,z). For each time step mentioned in the master file, there will be a separate data file. These files contain the values measured at each corresponding point loca-

tion. Each data file is organized as a table, where the rows represent the points, and the columns represent different data attributes or variables. The number of points in the geometry and data file must coincide.

## Nearest-neighbor search algorithm

As performed in [14], the CGAL [19] or the FLANN [4] library may be used to conduct the k-nearest-neighbor search regarding a certain distance measure (norm). As default setting, an isotropic hypersphere-search (distances based on the Euclidean norm) is conducted.

## Interpolation

An automatic conduction of the interpolation or approximation techniques to estimate the values at non-provided points is the purpose of this scattered-data reader. The interpolation of the data is based on Shepard's method

$$f(\mathbf{y}_k) = \frac{\mathbf{f}_i w_{ik}}{\delta_{ij} w_{ik}}$$

using the linear combination<sup>1</sup> of the data values  $\mathbf{f}_i$  and weights  $w_{ik}$  associated to a source point  $\mathbf{x}_i$

$$w_{ik} = \frac{1}{d(\mathbf{x}_i, \mathbf{x}_k)^p},$$

and a target point  $\mathbf{x}_k$ , the weighting exponent  $p$  and with the distance measure  $d()$ , e.g., the Euclidean distances. The interpolation methods can be easily extended to more sophisticated methods such as kriging, radial basis functions [7, 17], FEM interpolation based on triangulation [14].

## Connection to the finite element method

The finite element method (FEM) simulation discretizes the domain into finite elements, each with a quadrature point set (e.g., Gauss, Gauss-Lobatto, ...). These quadrature points serve as integration points for the numerical integration of the governing equations within each finite element. However, the original data might not align perfectly with the quadrature points, and the scattered data reader is used to provide the data here. Nearest-neighbor interpolation is a straightforward and computationally efficient method used to estimate the values at the quadrature points based on the data from the scattered points. The process can be summarized in the following steps: Find the nearest neighboring scattered data points for each quadrature point. This can be achieved by calculating the distances between the quadrature point and all the scattered points and selecting the k closest ones (FLANN or CGAL). Assign weights to the neighboring scattered data points based on their distance to

<sup>1</sup>Einstein's convention of summation is used.

the quadrature point and the interpolation method used. Generally speaking, the closer a scattered point is to the quadrature point, the higher its weight in the interpolation. Perform a weighted interpolation using a linear combination of the assigned weights and the data values of the neighboring scattered-data points. Repeat this process for all finite elements and quadrature points in the finite element mesh. After the scattered data reader completes the nearest-neighbor interpolation, the FEM simulation can proceed with numerical integration, solving the governing equations. The scattered data reader with nearest neighbor interpolation provides a handy tool for processing irregularly distributed data to boundary conditions or source terms of multi-physical simulations using openCFS.

## Usage

Within the simulation program, it can be used to interpolate field densities as source terms. In FEM simulations, the domain is discretized into finite elements, each containing a set of quadrature points. Nearest neighbor interpolation transfers data from scattered points within the domain to the quadrature points within the finite elements.

Similar to using the data for the source terms, it can be used to prescribe inhomogeneous boundaries. In FEM simulations, boundaries are discretized using quadrature points to compute boundary integrals. Nearest neighbor interpolation is employed to obtain values at the quadrature points on the boundary from the scattered data points.

In both variants, the nearest neighbor interpolation is applied to transfer data from scattered points to quadrature points within the domain or on the boundary. This allows the FEM simulation to work with the interpolated values and perform computations effectively.

## Setup

As illustrated in the "Abc2dcsvt" testsuite example of openCFS [13], first register a scattered data reader by the id and the file-name

```
...
<scatteredData>
  <csvt fileName="./csvt/data.descrip" id="myCSVt">
    <stepValues col="0"/>
    <stepFiles col="1"/>
    <coordinates>
      <comp dof="x" col="0"/>
      <comp dof="y" col="1"/>
<!-- <comp dof="z" col="2"/>-->
    </coordinates>
    <quantity name="scatter" id="acouPot" knnLib="Flann">
      <comp col="0"/>
      <!-- <comp dof="y" col="1"/>-->
      <!-- <comp dof="z" col="2"/>-->
    </quantity>
  </csvt>
</scatteredData>
...
```

The "stepValues" parameter "col" defines the line of the file path to the file containing the source point locations. With "stepFiles" and "col" one defines the line of the master-file where each physical time is stored (comma-

separated) followed by the file path to the data files in the same line. For each time step a new line is added to the file. The structure of the source point location file is defined within the xml-tag "coordinates". With the xml-tag "quantity", a scattered field is defined by the "id" and the nearest-neighbor search library "Flann"/"Cgal". The "id" is used later for using the "scatteredData" for boundaries or source terms using the same "id" as "quantityId" (see the second xml snippet).

In multiphysics simulations, different physical processes often interact with each other at the boundaries. These interactions may involve gradients, fluxes, or other boundary conditions that vary spatially and temporally [8]. The scattered data reader can efficiently organize and interpolate the inhomogeneous boundary terms to the corresponding boundary elements, allowing for accurate representation and simulation of complex boundary conditions. Then, one uses the scattered data for arbitrary boundary conditions within an openCFS simulation file.

```
...
<normalVelocity name="dombnd">
  <scatteredData quantityId="acouPot"/>
</normalVelocity>
...
```

The same can be done for any field. In this case, the openCFS community recommends using openCFS-Data [14].

## Application and Summary

This article presents a scattered data reader for connecting segregated multi-physical simulations. For instance, a hybrid aeroacoustic workflow can be realized efficiently [10, 5, 18, 20, 6, 2]. The methods' strength is leveraged, as the prescription of inhomogeneous boundary conditions for calculating the Helmholtz decomposition [16, 15, 9, 12] is possible. As performed recently, automotive OEMs use many software packages; the method supports an easy-to-integrate and easy-to-maintain interface [1, 21, 3].

In Multiphysics simulations, it is often computationally efficient to solve individual physical processes separately (segregated) and then exchange information iteratively until convergence is achieved. The scattered data reader's capability to handle irregularly distributed data allows each physical process to be simulated independently with its specific boundary and source data, ensuring accurate coupling between the domains. Consider a multiphysics simulation involving fluid-structure interaction (FSI), where the fluid and structural domains are simulated independently and exchange information at the interface in each iteration. The scattered data reader can facilitate this information exchange by handling the irregularly distributed data at the interface, ensuring consistent and accurate coupling between the fluid and structural simulations.

In this article, the scattered data reader module of openCFS was presented. The method can use CGAL or FLANN libraries for the geometrical operations. In

doing so, the reader provides an easy-to-integrate and easy-to-maintain interface for segregated multi-physical simulations using openCFS. In conclusion, the scattered data reader's ability to organize and interpolate inhomogeneous boundary terms and source values is paramount in multiphysics simulations. Its capability to handle irregularly distributed data points allows for an accurate representation of complex boundary conditions and sources, enabling researchers and engineers to perform segregated simulations of coupled physical phenomena efficiently and accurately. Finally, an automated method for predicting outliers in the offline connection may be valuable to consider [11]. The combination with the automated outlier detection makes the scattered data reader a valuable and robust tool in various scientific and engineering domains where multiphysics simulations are essential for gaining insights and making informed decisions.

## References

- [1] C. Freidhager, P. Maurerlehner, K. Roppert, A. Wurzinger, A. Hauser, M. Heimisch, S. Schoder, and M. Kaltenbacher. Simulationen von strömungsakustik in rotierenden bauteilen zur entwicklung von antriebskonzepten der autos der zukunft. *e & i Elektrotechnik und Informationstechnik*, 138(3):212–218, 2021.
- [2] M. Lenarcic, M. Eichhorn, S. Schoder, and C. Bauer. Numerical investigation of a high head francis turbine under steady operating conditions using foam-extend. In *Journal of Physics: Conference Series*, volume 579, page 012008. IOP Publishing, 2015.
- [3] P. Maurerlehner, S. Schoder, J. Tieber, C. Freidhager, H. Steiner, G. Brenn, K.-H. Schäfer, A. Ennemoser, and M. Kaltenbacher. Aeroacoustic formulations for confined flows based on incompressible flow data. *Acta Acustica*, 6:45, 2022.
- [4] M. Muja and D. G. Lowe. *FLANN: Fast library for approximate nearest neighbor*. M. Muja, 1.8.4 edition, 2013.
- [5] S. Schoder. cpcw–perturbed convective wave equation based on compressible flows. *arXiv preprint arXiv:2209.11410*, 2022.
- [6] S. Schoder and F. Czwielong. Dataset fan-01: Revisiting the eaa benchmark for a low-pressure axial fan. *arXiv preprint arXiv:2211.12014*, 2022.
- [7] S. Schoder, C. Junger, K. Roppert, and M. Kaltenbacher. Radial basis function interpolation for computational aeroacoustics. In *AIAA AVIATION 2020 FORUM*, page 2511, 2020.
- [8] S. Schoder and M. Kaltenbacher. Hybrid aeroacoustic computations: State of art and new achievements. *Journal of Theoretical and Computational Acoustics*, 27(04):1950020, 2019.
- [9] S. Schoder, M. Kaltenbacher, and K. Roppert. Helmholtz's decomposition applied to aeroacoustics. In *25th AIAA/CEAS Aeroacoustics Conference*, 2019-2561.
- [10] S. Schoder, M. Kaltenbacher, É. Spieser, H. Vincent, C. Bogey, and C. Bailly. Aeroacoustic wave equation based on pierce's operator applied to the sound generated by a mixing layer. In *28th AIAA/CEAS Aeroacoustics 2022 Conference*, page 2896, 2022.
- [11] S. Schoder, F. Kraxberger, S. Falk, A. Wurzinger, K. Roppert, S. Kniesburges, M. Döllinger, and M. Kaltenbacher. Error detection and filtering of incompressible flow simulations for aeroacoustic predictions of human voice. *The Journal of the Acoustical Society of America*, 152(3):1425–1436, 2022.
- [12] S. Schoder, E. Museljic, F. Kraxberger, and A. Wurzinger. Post-processing subsonic flows using physics-informed neural networks. In *2023 AIAA AVIATION Forum*, 2022.
- [13] S. Schoder and K. Roppert. opencfs: Open source finite element software for coupled field simulation–part acoustics. *arXiv preprint arXiv:2207.04443*, 2022.
- [14] S. Schoder and K. Roppert. opencfs-data: Data pre-post-processing tool for opencfs–aeroacoustics source filters. *arXiv preprint arXiv:2302.03637*, 2023.
- [15] S. Schoder, K. Roppert, and M. Kaltenbacher. Helmholtz's decomposition for compressible flows and its application to computational aeroacoustics. *SN Partial Differ. Equ. Appl.*, pages 1–20, 2020.
- [16] S. Schoder, K. Roppert, and M. Kaltenbacher. Post-processing of direct aeroacoustic simulations using helmholtz decomposition. *AIAA Journal*, pages 1–9, 2020.
- [17] S. Schoder, K. Roppert, M. Weitz, C. Junger, and M. Kaltenbacher. Aeroacoustic source term computation based on radial basis functions. *International Journal for Numerical Methods in Engineering*, 121(9):2051–2067, 2020.
- [18] S. Schoder, A. Wurzinger, C. Junger, M. Weitz, C. Freidhager, K. Roppert, and M. Kaltenbacher. Application limits of conservative source interpolation methods using a low mach number hybrid aeroacoustic workflow. *Journal of Theoretical and Computational Acoustics*, 29(01):2050032, 2021.
- [19] The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 4.10 edition, 2017.
- [20] L. Tieghi, S. Becker, A. Corsini, G. Delibra, S. Schoder, and F. Czwielong. Machine-learning clustering methods applied to detection of noise sources in low-speed axial fan. *Journal of Engineering for Gas Turbines and Power*, 145(3):031020, 2023.
- [21] M. Weitz, S. Schoder, and M. Kaltenbacher. Numerical investigation of the resonance behavior

of flow-excited helmholtz resonators. *PAMM*,  
19(1):e201900033, 2019.