


FluidSF: A Python package for calculating turbulent fluid statistics

Cassidy M. Wagner^{1*}, Ara Lee¹, and Brodie Pearson^{1*}

¹ Oregon State University  Corresponding author * These authors contributed equally.

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Open Journals](#) 

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

Fluid systems are everywhere, from small-scale engineering problems to planetary-and-larger-scale systems (atmosphere, ocean, galactic gas clouds). These systems are often turbulent, where motion is chaotic, unpredictable, and can only be characterized through statistical analyses. Spatial structure functions (SFs) are one such statistical analysis technique for turbulence, that require calculation of spatial differences in properties as a function of their separation distance. By combining and then averaging these spatial differences, various types of SF can be constructed to measure physical properties of fluid flow, such as heat and energy transfers, energy density, intermittency etc. However, calculating SFs is often a cumbersome and computationally-intensive task tailored to the specific format of a given fluid dataset.

Statement of need

FluidSF is a flexible Python package for calculating spatial structure functions (SFs) from general fluid datasets that describe spatial variations in one-, two-, or three-dimensions. The package can construct an array of traditional and modern SFs, and can be easily modified to calculate user-defined SFs that utilize general fluid properties, including scalars (e.g., vorticity, density) and/or vectors (e.g., velocity, magnetic field). The package also includes several tools to process data and diagnose useful properties. The flexibility of this package enables geophysical, astrophysical, and engineering applications such as; quantifying the energy cycles within Earth's ocean (Balwada et al., 2022; J. Pearson et al., 2019), Earth's atmosphere (Lindborg, 1999), and Jupiter's atmosphere (Young & Read, 2017), diagnosing the intermittency of magnetohydrodynamic plasma turbulence (Wan et al., 2016) and the scaling laws of idealized 3D turbulence (Iyer et al., 2020), or measuring the characteristics of ocean surface temperature (Schloesser et al., 2016) or the anisotropy of flow over rough beds (Coscarella et al., 2020).

Paragraph on package capabilities & limitations. Regularly-gridded data, Lat-lon gridded data, track/directional sampling, 1D-data, evenly-spaced, irregularly-spaced (what are limitations), binning, bootstrapping(?), local advection terms (B. C. Pearson et al., 2021), Bessel function examples(?) (Xie & Bühler, 2018) examples of time-averaging, SWOT application. What are limitations (can it take 2D data in a vector rather than array format? Can it calculate 2D or 3D maps of SF rather than just a function of $|r|$ magnitude?). Perhaps these don't need to be mentioned, or can be stated as future developments.

Spatial SFs are constructed by averaging the correlations between spatial differences of properties. For example, given an arbitrary scalar field (ϕ), we could calculate a structure function,

$$SF_{\phi\phi}(\mathbf{r}) = \overline{\delta\phi\delta\phi} = \overline{[\phi(\mathbf{x} + \mathbf{r}) - \phi(\mathbf{x})][\phi(\mathbf{x} + \mathbf{r}) - \phi(\mathbf{x})]}$$

where \mathbf{x} denotes the position of a data point, $\delta\phi$ denotes the spatial variation of ϕ , and the overline denotes an average over all positions (\mathbf{x}). Structure functions depend on the

40 separation vector (\mathbf{r}), and are often analyzed with an assumption of isotropic flow statistics
41 [$SF(\mathbf{r}) \rightarrow SF(r = |\mathbf{r}|)$]. There are many types of physically-useful structure functions. The
42 example above is a second-order scalar SF (i.e., it contains 2 δ terms of the scalar ϕ), but
43 additional physical insight can be gained from third- & higher-order SFs (3+ δ terms), SFs
44 that depend on other scalar or vector fields, and SFs that blend information from various
45 scalar/vector fields.

Some basic structure functions from a simulation of anisotropic 2D or QG turbulence: (a)
Snapshot of the vorticity field used for SF calculations, (b) various Zonally- and
meridionally-separated SFs, including third-order velocity, advective, and vorticity SFs from 2D
turbulence simulation, (c) Inter-scale fluxes inferred from structure functions and diagnosed
directly from the simulation

Figure 1: Some basic structure functions from a simulation of anisotropic 2D or QG turbulence: (a)
Snapshot of the vorticity field used for SF calculations, (b) various Zonally- and meridionally-separated
SFs, including third-order velocity, advective, and vorticity SFs from 2D turbulence simulation, (c)
Inter-scale fluxes inferred from structure functions and diagnosed directly from the simulation

Structure functions calculated from satellite observations of the ocean surface:

Figure 2: Structure functions calculated from satellite observations of the ocean surface:

Map of the 2D spatial variation of the (a) 3rd-order longitudinal and (b) advective velocity
SFs from the same data set as Figure 1:

Figure 3: Map of the 2D spatial variation of the (a) 3rd-order longitudinal and (b) advective velocity
SFs from the same data set as Figure 1:

46 FluidSF can utilize a variety of fluid data, including data sets with 1-, 2-, and 3-dimensional
47 spatial data, and from domains with periodic or non-periodic boundary conditions (Figure
48 1). In addition to regular Cartesian-gridded data (1D, 2D or 3D), the software also has
49 some support for non-uniform latitude-longitude grids (1D or 2D). Since FluidSF is written
50 in Python, any fluid data initialized and loaded as NumPy arrays can be used to calculate SFs.
51 To demonstrate the flexibility of input data, Figure 1 shows SFs calculated by FluidSF for a
52 simulation of quasi-geostrophic turbulence in a periodic domain using `GeophysicalFlows.jl`
53 (?), while Figure 2 shows SFs calculated from satellite observations of the ocean surface made
54 by the SWOT (Surface Waves and Ocean Topography) mission (Morrow et al., 2018).

55 FluidSF can also calculate a wide array of traditional and novel structure functions, including
56 $SF_{\phi\phi}$ (Eq. 1), second- and third-order SFs of longitudinal and transverse velocity, blended
57 velocity-scalar third-order SFs, and advective SFs of velocity, vorticity and scalars (B. C. Pearson
58 et al., 2021). FluidSF can calculate these SFs in specific directions (i.e., aligned with the
59 Cartesian co-ordinates), and for 2D data it can diagnose a map showing how the SFs vary with
60 the magnitude and orientation of the separation vector \mathbf{r} (Figure 3). FluidSF also includes tools
61 to make the calculation and processing of SFs easier, such as Therefore FluidSF also computes
62 advection for 2D and 3D data. It supports blended SFs, i.e. a combination of longitudinal and
63 transverse velocity SFs, whereas other software only supports purely longitudinal or transverse
64 SFs.

65 To explore spatial variations in SFs, FluidSF computes 2D polar maps of SFs that vary in
66 separation distance and separation direction.

67 Related Work

68 FluidSF uniquely contributes to the field through a combination of expanded data support,
69 the ability to diagnose a wide array of SF types, and tools for analyzing spatial variations

in SFs. There are a small number of open source software available that calculate structure functions. fastSF is a parallelized C++ code designed to compute structure functions from Cartesian grids of data (Sadhukhan et al., 2021). Fuchs et al. (2022) created an open source MATLAB toolkit that performs a variety of turbulence analysis, including structure functions. An complimentary and alternative method to structure functions for analyzing turbulence data is coarse-graining. FlowSieve is a primarily C++ package that uses coarse-graining to estimate ocean and atmospheric turbulence properties from Global Climate Model data (Storer & Aluie, 2023).

Acknowledgements

The development of FluidSF was supported by the National Science Foundation under Grant No. 2023721.

References

- Balwada, D., Xie, J.-H., Marino, R., & Feraco, F. (2022). Direct observational evidence of an oceanic dual kinetic energy cascade and its seasonality. *Science Advances*, 8(41), eabq2566.
- Coscarella, F., Penna, N., Servidio, S., & Gaudio, R. (2020). Turbulence anisotropy and intermittency in open-channel flows on rough beds. *Physics of Fluids*, 32(11).
- Fuchs, A., Kharche, S., Patil, A., Friedrich, J., Wächter, M., & Peinke, J. (2022). An open source package to perform basic and advanced statistical analysis of turbulence data and other complex systems. *Physics of Fluids*, 34(10), 101801. <https://doi.org/10.1063/5.0107974>
- Iyer, K. P., Sreenivasan, K. R., & Yeung, P. (2020). Scaling exponents saturate in three-dimensional isotropic turbulence. *Physical Review Fluids*, 5(5), 054605.
- Lindborg, E. (1999). Can the atmospheric kinetic energy spectrum be explained by two-dimensional turbulence? *Journal of Fluid Mechanics*, 388, 259–288.
- Morrow, R., Blurmstein, D., & Dibarboure, G. (2018). Fine-scale altimetry and the future SWOT mission. *New Frontiers in Operational Oceanography*, 191–226.
- Pearson, B. C., Pearson, J. L., & Fox-Kemper, B. (2021). Advective structure functions in anisotropic two-dimensional turbulence. *Journal of Fluid Mechanics*, 916, 49. <https://doi.org/10.1017/jfm.2021.247>
- Pearson, J., Fox-Kemper, B., Barkan, R., Choi, J., Bracco, A., & McWilliams, J. C. (2019). Impacts of convergence on structure functions from surface drifters in the gulf of mexico. *Journal of Physical Oceanography*, 49(3), 675–690.
- Sadhukhan, S., Bhattacharya, S., & Verma, M. K. (2021). fastSF: A parallel code for computing the structure functions of turbulence. *Journal of Open Source Software*, 6(57), 2185. <https://doi.org/10.21105/joss.02185>
- Schloesser, F., Cornillon, P., Donohue, K., Boussidi, B., & Iskin, E. (2016). Evaluation of thermosalinograph and VIIRS data for the characterization of near-surface temperature fields. *Journal of Atmospheric and Oceanic Technology*, 33(9), 1843–1858.
- Storer, B. A., & Aluie, H. (2023). FlowSieve: A Coarse-Graining Utility for Geophysical Flows on the Sphere. *Journal of Open Source Software*, 8(84), 4277. <https://doi.org/10.21105/joss.04277>
- Wan, M., Matthaeus, W., Roytershteyn, V., Parashar, T., Wu, P., & Karimabadi, H. (2016). Intermittency, coherent structures and dissipation in plasma turbulence. *Physics of Plasmas*,

114 23(4).

115 Xie, J.-H., & Bühler, O. (2018). Exact third-order structure functions for two-dimensional
116 turbulence. *Journal of Fluid Mechanics*, 851, 672–686. [https://doi.org/10.1017/jfm.2018.](https://doi.org/10.1017/jfm.2018.528)
117 528

118 Young, R. M., & Read, P. L. (2017). Forward and inverse kinetic energy cascades in jupiter's
119 turbulent weather layer. *Nature Physics*, 13(11), 1135–1140.

DRAFT