

cosmo-numba: B-modes and COSEBIs computations accelerated by Numba

Axel Guinot ¹ and Rachel Mandelbaum ¹

¹ Department of Physics, McWilliams Center for Cosmology and Astrophysics, Carnegie Mellon University, Pittsburgh, PA 15213, USA

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Published: unpublished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

Weak gravitational lensing is a widely used probe in cosmological analysis. It allows astrophysicists to understand the content and evolution of the Universe. We are entering an era where we are not limited by the data volume but by systematics. It is in this context that we are presenting here a simple python based software package to help in the computation of E-/B-mode decomposition which can be use for systematic checks or science analysis. As we demonstrate after, our implementation has both the high precision required and speed to perform this kind of analysis.

Statement of need

The E-/B-mode composition for cosmic shear poses a significant computational challenge given the need for high precision (required to integrate oscillatory functions over a large integration range and achieve accurate results) and speed. Cosmo-numba meets this need, facilitating the computation of E-/B-modes decomposition using two methods. One of them is the Complete Orthogonal Sets of E-/B-mode Integrals (COSEBIs) as presented in P. Schneider et al. (2010). The COSEBIs rely on very high precision computation requiring more than 80 decimal places. P. Schneider et al. (2010) propose an implementation using mathematica. cosmo-numba uses a combination of sympy and mpmath to reach the required precision. This python version enables an easier integration within cosmological inference pipelines, which are commonly python-based, and facilitates the null tests.

This software package also enables the computation of the pure-mode correlation functions presented in Peter Schneider et al. (2022). Those integrals are less numerically challenging than the COSEBIs, but having a fast computation is necessary for computing the covariance matrix. One can also use those correlation functions for cosmological inference, in which case the large number of calls to the likelihood function will also require a fast implementation.

Testing setup

In the following two sections we will need fiducial shear-shear correlation functions, $\xi_{\pm}(\theta)$, and power spectrum, $P_{E/B}(\ell)$. They have been computed using the Core Cosmology Library¹ (Chisari et al., 2019) developed by the Dark Energy Science Collaboration.

COSEBIs

The COSEBIs are defined as:

¹<https://github.com/LSSTDESC/CCL>

$$E_n = \frac{1}{2} \int_0^\infty d\theta \theta [T_{n,+}(\theta) \xi_+(\theta) + T_{n,-}(\theta) \xi_-(\theta)], \quad (1)$$

$$B_n = \frac{1}{2} \int_0^\infty d\theta \theta [T_{n,+}(\theta) \xi_+(\theta) - T_{n,-}(\theta) \xi_-(\theta)]; \quad (2)$$

where $\xi_\pm(\theta)$ are the shear correlation functions, and $T_{n,\pm}$ are the weight functions for the COSEBI mode n . The complexity is in the computation of the weight functions. Cosmo-numba carries out the computation of the weight functions in a logarithmic scale defined by:

$$T_{n,+}^{\log}(\theta) = t_{n,+}^{\log}(z) = N_n \sum_{j=0}^{n+1} \bar{c}_{nj} z^j; \quad (3)$$

where $z = \log(\theta/\theta_{\min})$, N_n is the normalization for the mode n , and \bar{c}_{jn} are defined iteratively from Bessel functions (we refer the readers to P. Schneider et al. (2010) for more details).

We have validated our implementation against the original version in Mathematica from P. Schneider et al. (2010). In Figure 1 we show the impact of the precision going from 15 decimal places, which corresponds to the precision one could achieve using float64, up to 80 decimal places, the precision used in the original Mathematica implementation. We can see that classic float64 precision would not be sufficient, and with a precision of 80 our code exactly recovers the results from the original implementation. Similarly, the impact on the COSEBIs is shown in Figure 2.

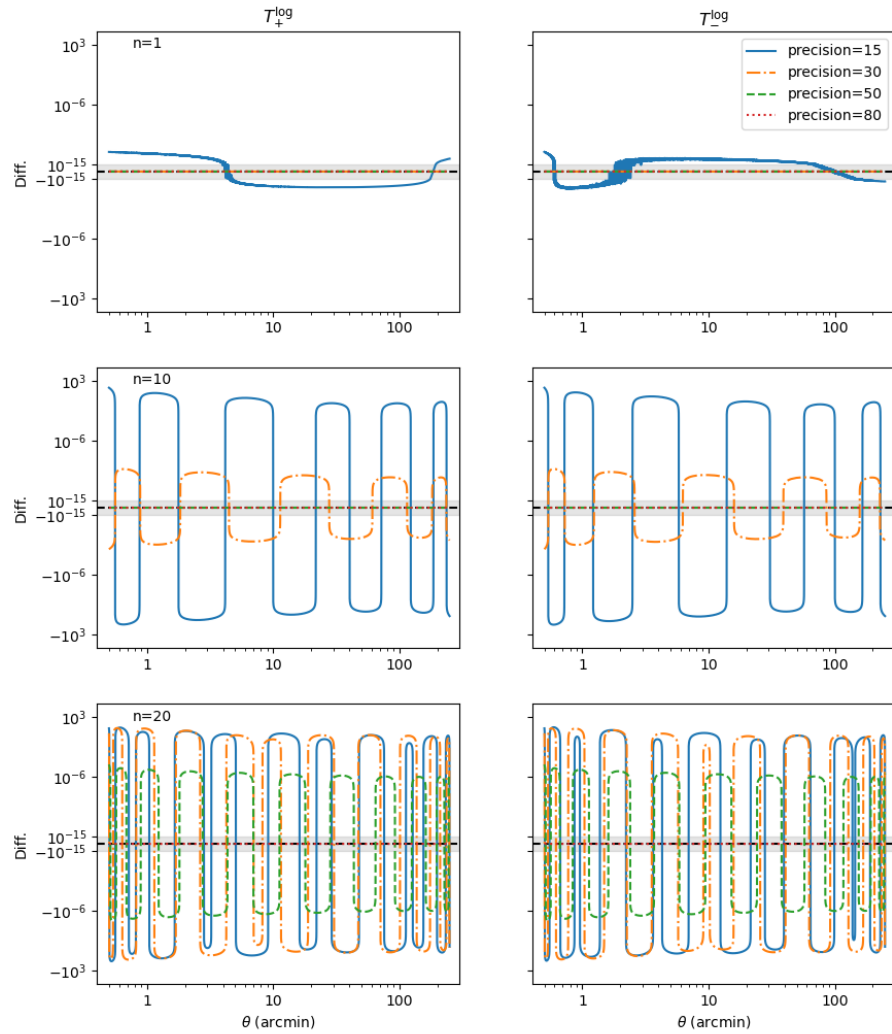


Figure 1: In this figure we show the impact of the precision in the computation of the weight functions T_{\pm}^{\log} . For comparison, a precision of 15 corresponds to what would be achieved using numpy float64. The difference is computed with respect to the original Mathematica implementation presented in P. Schneider et al. (2010). The figure uses symlog, the shaded region represent the linear scale.

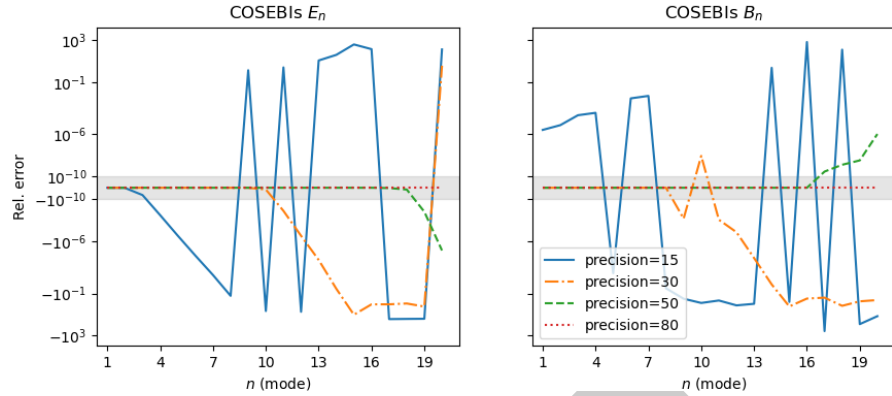


Figure 2: This figure shows the difference on the COSEBIs E- and B-mode relative to the original Mathematica implementation. We see that using only 15 decimal places would lead to several percent error making an implementation based on numpy float64 not suitable. The figure uses symlog, the shaded region represent the linear scale.

COSEBIs can also be defined from the power spectrum as:

$$E_n = \int_0^\infty \frac{d\ell}{2\pi} \ell P_E(\ell) W_n(\ell); \quad (4)$$

$$B_n = \int_0^\infty \frac{d\ell}{2\pi} \ell P_B(\ell) W_n(\ell); \quad (5)$$

where $P_{E/B}(\ell)$ is the power spectrum of E- and B-modes and $W_n(\ell)$ are the filter functions which can be computed from $T_{n,+}$ as:

$$W_n(\ell) = \int_{\theta_{\min}}^{\theta_{\max}} d\theta \theta T_{n,+}(\theta) J_0(\ell\theta); \quad (6)$$

with $J_0(\ell\theta)$ the 0-th order Bessel function. The Equation 6 is a Hankel transform of order 0. It can be computed using the FFTLog algorithm presented in Hamilton (2000) implemented here in Numba. Figure 3 shows the comparison between the COSEBIs computed from $\xi_{\pm}(\theta)$ and from $C_{E/B}(\ell)$. We can see that the COSEBI E-modes agree very well but the B-modes are more stable when computed from the $C(\ell)$ space.

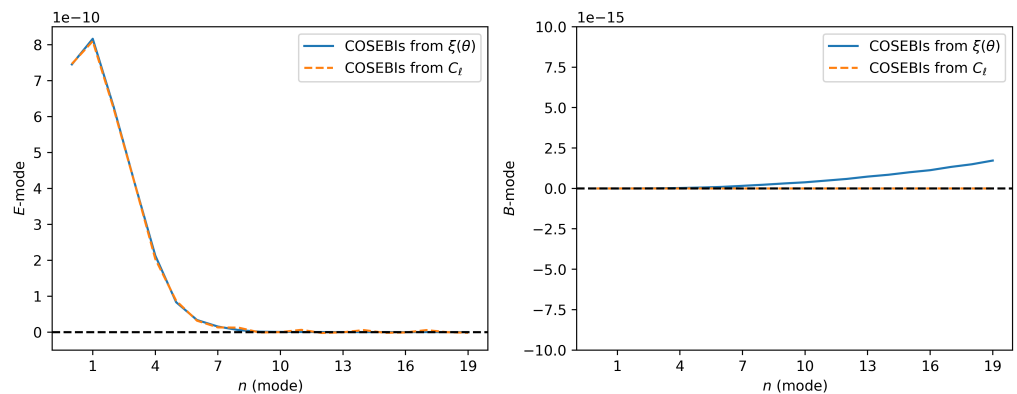


Figure 3: Comparison of the COSEBIs E- and B-mode computed from $\xi_{\pm}(\theta)$ and $C_{E/B}(\ell)$.

Pure-Mode Correlation Functions

In this section we describe the computation of the pure-mode correlation functions as defined in Peter Schneider et al. (2022). There are defined as follow:

$$\xi_+^E(\vartheta) = \frac{1}{2} \left[\xi_+(\vartheta) + \xi_-(\vartheta) + \int_{\vartheta}^{\vartheta_{\max}} \frac{d\theta}{\theta} \xi_-(\theta) \left(4 - \frac{12\vartheta^2}{\theta^2} \right) \right] - \frac{1}{2} [S_+(\vartheta) + S_-(\vartheta)], \quad (7)$$

$$\xi_+^B(\vartheta) = \frac{1}{2} \left[\xi_+(\vartheta) - \xi_-(\vartheta) - \int_{\vartheta}^{\vartheta_{\max}} \frac{d\theta}{\theta} \xi_-(\theta) \left(4 - \frac{12\vartheta^2}{\theta^2} \right) \right] - \frac{1}{2} [S_+(\vartheta) - S_-(\vartheta)], \quad (8)$$

$$\xi_-^E(\vartheta) = \frac{1}{2} \left[\xi_+(\vartheta) + \xi_-(\vartheta) + \int_{\vartheta_{\min}}^{\vartheta} \frac{d\theta}{\theta^2} \xi_+(\theta) \left(4 - \frac{12\theta^2}{\vartheta^2} \right) \right] - \frac{1}{2} [V_+(\vartheta) + V_-(\vartheta)], \quad (9)$$

$$\xi_-^B(\vartheta) = \frac{1}{2} \left[\xi_+(\vartheta) - \xi_-(\vartheta) + \int_{\vartheta_{\min}}^{\vartheta} \frac{d\theta}{\theta^2} \xi_+(\theta) \left(4 - \frac{12\theta^2}{\vartheta^2} \right) \right] - \frac{1}{2} [V_+(\vartheta) - V_-(\vartheta)]; \quad (10)$$

where $\xi_{\pm}(\theta)$ correspond to the shear-shear correlation function. The functions $S_{\pm}(\theta)$ and $V_{\pm}(\theta)$ are themselves defined by integrals and we refer the reader to Peter Schneider et al. (2022) for more details about their definition. By contrast with the computation of the COSEBIs, these integrals are more stable and straightforward to compute but still require some level of precision. This is why we are using the qags from the QUADPACK² (Piessens et al., 2012) method with a 5-th order spline interpolation. In addition, as one can see from the equations above, the implementation requires a loop over a range of ϑ values. This is why having a fast implementation will be required if one want to use those correlation functions in cosmological inference.

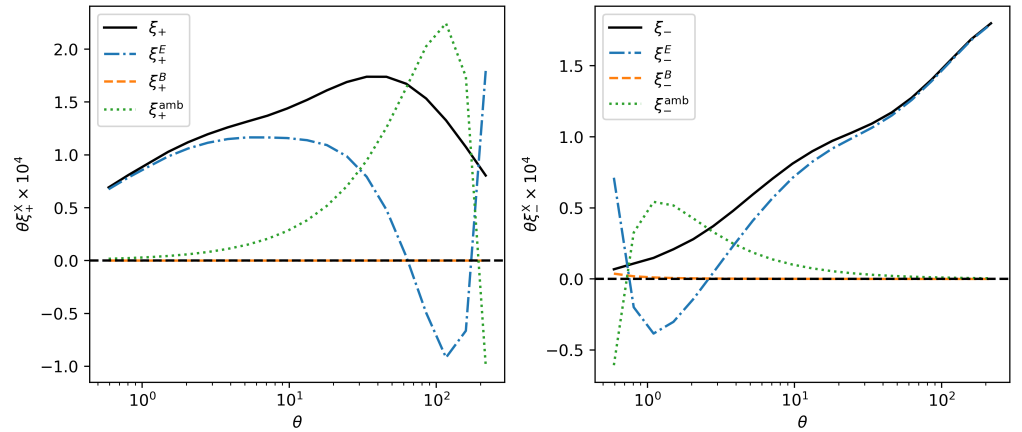


Figure 4: This figure shows the decomposition of the shear-shear correlation functions in E- and B-modes (and ambiguous mode).

Acknowledgements

The authors acknowledge the support of a grant from the Simons Foundation (Simons Investigator in Astrophysics, Award ID 620789).

²We use C version of the library wrapped to python using Numba: <https://github.com/Nicholaswogan/NumbaQuadpack>

References

- Chisari, N. E., Alonso, D., Krause, E., Leonard, C. D., Bull, P., Neveu, J., Villarreal, A., Singh, S., McClintock, T., Ellison, J., Du, Z., Zuntz, J., Mead, A., Joudaki, S., Lorenz, C. S., Tröster, T., Sanchez, J., Lanusse, F., Ishak, M., ... Wagoner, E. L. (2019). Core cosmology library: Precision cosmological predictions for LSST. *The Astrophysical Journal Supplement Series*, 242(1), 2. <https://doi.org/10.3847/1538-4365/ab1658>
- Hamilton, A. J. S. (2000). Uncorrelated modes of the non-linear power spectrum. *Monthly Notices of the Royal Astronomical Society*, 312(2), 257–284. <https://doi.org/10.1046/j.1365-8711.2000.03071.x>
- Piessens, R., Doncker-Kapenga, E. de, Überhuber, C. W., & Kahaner, D. K. (2012). *Quadpack: A subroutine package for automatic integration* (Vol. 1). Springer Science & Business Media.
- Schneider, Peter, Asgari, M., Jozani, Y. N., Dvornik, A., Giblin, B., Harnois-Déraps, J., Heymans, C., Hildebrandt, H., Hoekstra, H., Kuijken, K., Shan, H., Tröster, T., & Wright, A. H. (2022). Pure-mode correlation functions for cosmic shear and application to KiDS-1000. *Astronomy & Astrophysics*, 664, A77. <https://doi.org/10.1051/0004-6361/202142479>
- Schneider, P., Eifler, T., & Krause, E. (2010). COSEBIs: Extracting the full e-/b-mode information from cosmic shear correlation functions. *Astronomy and Astrophysics*, 520, A116. <https://doi.org/10.1051/0004-6361/201014235>