

¹ cosmo-numba: B-modes and COSEBIs computations accelerated by Numba

³ Axel Guinot  ¹ and Rachel Mandelbaum  ¹

⁴ 1 Department of Physics, McWilliams Center for Cosmology and Astrophysics, Carnegie Mellon
⁵ 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Published: unpublished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](#))

⁶ Summary

⁷ Weak gravitational lensing is a widely used probe in cosmological analysis. It allows astrophysicists to understand the contend 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.

¹⁴ 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.

²⁵ 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.

³⁰ Testing setup

³¹ In the following two sections we will need fiducial shear-shear correlation functions, $\xi_{\pm}(\theta)$,
³² and power spectrum, $P_{E/B}(\ell)$. They have been computed using the Core Cosmology Library¹
³³ ([Chisari et al., 2019](#)) developed by the Dark Energy Science Collaboration. The cosmological
³⁴ parameters are taken from Aghanim et al. ([2020](#)). For tests that involved covariance we are
³⁵ using the Stage-IV Legacy Survey of Space and Time (LSST) Year 10 as a reference. The
³⁶ characteristics are taken from the Science Requirements Document (SRD) ([Collaboration et al., 2021](#)).

¹ <https://github.com/LSSTDESC/CCL>

38 COSEBIs

39 The COSEBIs are defined as:

$$40 E_n = \frac{1}{2} \int_0^\infty d\theta \theta [T_{n,+}(\theta) \xi_+(\theta) + T_{n,-}(\theta) \xi_-(\theta)], \quad (1)$$

$$41 B_n = \frac{1}{2} \int_0^\infty d\theta \theta [T_{n,+}(\theta) \xi_+(\theta) - T_{n,-}(\theta) \xi_-(\theta)]; \quad (2)$$

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

$$43 T_{n,+}^{\log}(\theta) = t_{n,+}^{\log}(z) = N_n \sum_{j=0}^{n+1} \bar{c}_{nj} z^j; \quad (3)$$

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

46 We have validating our implementation against the original version in Mathematica from P.
 47 Schneider et al. (2010). In Figure 1 we show the impact of the precision going from 15 decimal
 48 places, which corresponds to the precision one could achieve using float64, up to 80 decimal
 49 places, the precision used in the original Mathematica implementation. We can see that classic
 50 float64 precision would not be sufficient, and with a precision of 80 our code exactly recovers
 51 the results from the original implementation. **Given that the precision comes at very little**
 52 **computational cost, we default to the original implementation using high precision. The**
 53 **impact of the precision propagated to the COSEBIs is shown in Figure 2. We can see that in**
 54 **the lower setting, it can lead to several percent error.**

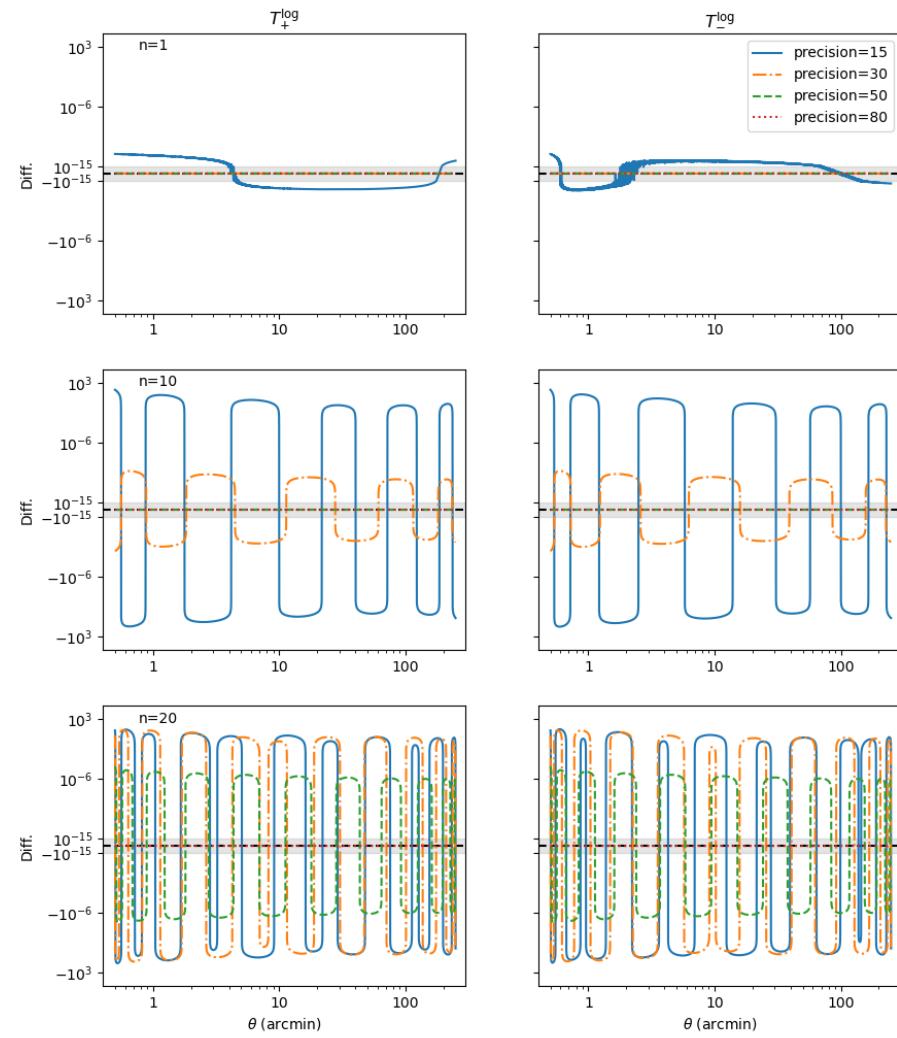


Figure 1: In this figure we show the impact of the precision in the computation of the weight functions T_{\pm}^{\log} . For comparison, a precision of 15 corresponds to what would be achieved using numpy float64. The difference is computed with respect to the original Mathematica implementation presented in P. Schneider et al. (2010). The figure uses symlog, the shaded region represent the linear scale, strarting at 10^{-15} representing the float64 precision for reference.

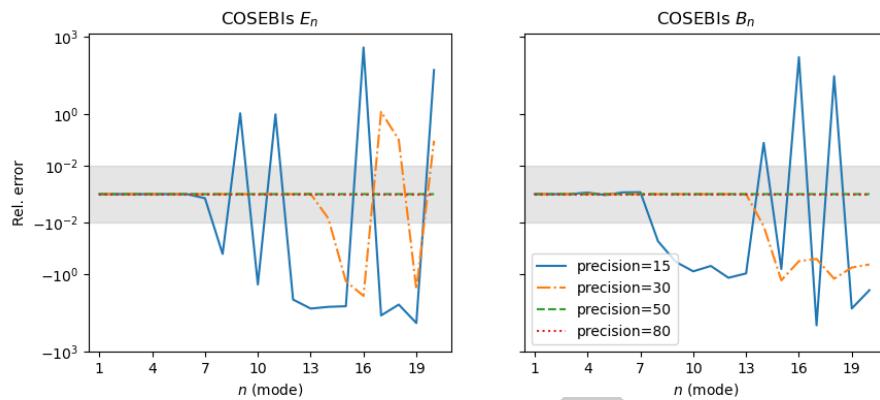


Figure 2: This figure shows the difference on the COSEBIs E- and B-mode relative to the original Mathematica implementation. We see that using only 15 decimal places would lead to several percent error making an implementation based on numpy float64 not suitable. The figure uses symlog, the shaded region represent the linear scale, starting at 1% for reference.

55 COSEBIs can also be defined from the power spectrum as:

$$56 \quad E_n = \int_0^\infty \frac{d\ell \ell}{2\pi} P_E(\ell) W_n(\ell); \quad (4)$$

$$57 \quad B_n = \int_0^\infty \frac{d\ell \ell}{2\pi} P_B(\ell) W_n(\ell); \quad (5)$$

58 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:

$$59 \quad W_n(\ell) = \int_{\theta_{\min}}^{\theta_{\max}} d\theta \theta T_{n,+}(\theta) J_0(\ell\theta); \quad (6)$$

60 with $J_0(\ell\theta)$ the 0-th order Bessel function. The Equation 6 is a Hankel transform of order 0.
61 It can be computed using the FFTLog algorithm presented in Hamilton (2000) implemented
62 here in Numba. Figure 3 shows the comparison between the COSEBI E- & B-modes agree very well, with at most
63 0.3 σ difference with respect to the LSST Y10 covariance.

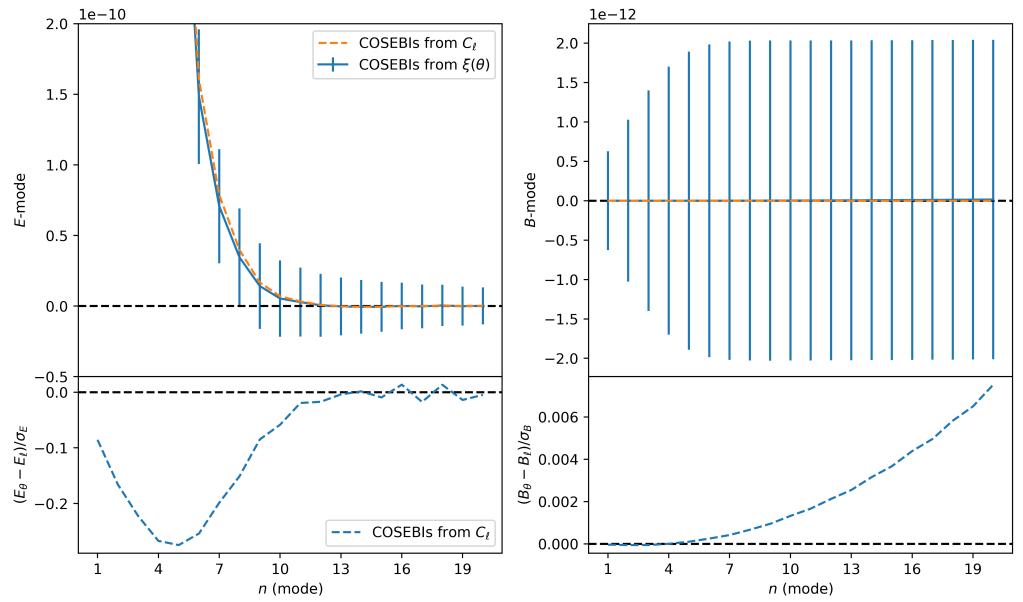


Figure 3: Comparison of the COSEBIs E- and B-mode computed from $\xi_{\pm}(\theta)$ and $C_{E/B}(\ell)$.

64 Pure-Mode Correlation Functions

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

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

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

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

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

69 where $\xi_{\pm}(\theta)$ correspond to the shear-shear correlation function. The functions $S_{\pm}(\theta)$ and $V_{\pm}(\theta)$
 70 are themselves defined by integrals and we refer the reader to Peter Schneider et al. (2022)
 71 for more details about their definition. By contrast with the computation of the COSEBIs,
 72 these integrals are more stable and straightforward to compute but still require some level of
 73 precision. This is why we are using the qags method from the QUADPACK² (Piessens et al.,
 74 2012) with a 5-th order spline interpolation. In addition, as one can see from the equations
 75 above, the implementation requires a loop over a range of ϑ values. This is why having a fast
 76 implementation will be required if one want to use those correlation functions in cosmological
 77 inference.

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

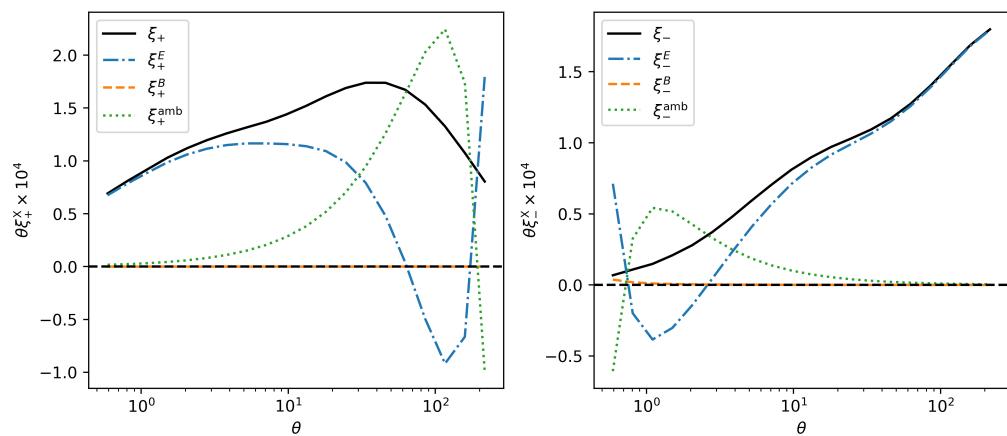


Figure 4: This figure shows the decomposition of the shear-shear correlation functions in E- and B-modes (and ambiguous mode).

78 Acknowledgements

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

81 References

- 82 Aghanim, N., Akrami, Y., Ashdown, M., Aumont, J., Baccigalupi, C., Ballardini, M., Banday,
83 A. J., Barreiro, R. B., Bartolo, N., Basak, S., Battye, R., Benabed, K., Bernard, J.-P.,
84 Bersanelli, M., Bielewicz, P., Bock, J. J., Bond, J. R., Borrill, J., Bouchet, F. R., ...
85 Zonca, A. (2020). Planck 2018 results: VI. Cosmological parameters. *Astronomy & Astrophysics*, 641, A6. <https://doi.org/10.1051/0004-6361/201833910>
- 87 Chisari, N. E., Alonso, D., Krause, E., Leonard, C. D., Bull, P., Neveu, J., Villarreal, A., Singh,
88 S., McClintock, T., Ellison, J., Du, Z., Zuntz, J., Mead, A., Joudaki, S., Lorenz, C. S.,
89 Tröster, T., Sanchez, J., Lanusse, F., Ishak, M., ... Wagoner, E. L. (2019). Core cosmology
90 library: Precision cosmological predictions for LSST. *The Astrophysical Journal Supplement Series*, 242(1), 2. <https://doi.org/10.3847/1538-4365/ab1658>
- 92 Collaboration, T. L. D. E. S., Mandelbaum, R., Eifler, T., Hložek, R., Collett, T., Gawiser,
93 E., Scolnic, D., Alonso, D., Awan, H., Biswas, R., Blazek, J., Burchat, P., Chisari, N. E.,
94 Dell'Antonio, I., Digel, S., Frieman, J., Goldstein, D. A., Hook, I., Ivezić, Ž., ... Troxel,
95 M. A. (2021). *The LSST dark energy science collaboration (DESC) science requirements*
96 document. <https://arxiv.org/abs/1809.01669>
- 97 Hamilton, A. J. S. (2000). Uncorrelated modes of the non-linear power spectrum. *Monthly
98 Notices of the Royal Astronomical Society*, 312(2), 257–284. [https://doi.org/10.1046/j.1365-8711.2000.03071.x](https://doi.org/10.1046/j.
99 1365-8711.2000.03071.x)
- 100 Piessens, R., Doncker-Kapenga, E. de, Überhuber, C. W., & Kahaner, D. K. (2012). *Quadpack:
101 A subroutine package for automatic integration* (Vol. 1). Springer Science & Business
102 Media.
- 103 Schneider, Peter, Asgari, M., Jozani, Y. N., Dvornik, A., Giblin, B., Harnois-Déraps, J.,
104 Heymans, C., Hildebrandt, H., Hoekstra, H., Kuijken, K., Shan, H., Tröster, T., &
105 Wright, A. H. (2022). Pure-mode correlation functions for cosmic shear and application to
106 KiDS-1000. *Astronomy & Astrophysics*, 664, A77. [https://doi.org/10.1051/0004-6361/202142479](https://doi.org/10.1051/0004-
107 6361/202142479)

¹⁰⁸ Schneider, P., Eifler, T., & Krause, E. (2010). COSEBIs: Extracting the full e-/b-mode
¹⁰⁹ information from cosmic shear correlation functions. *Astronomy and Astrophysics*, 520,
¹¹⁰ A116. <https://doi.org/10.1051/0004-6361/201014235>

DRAFT