

# <sup>1</sup> cosmo-numba: B-modes and COSEBIs computations accelerated by Numba

<sup>3</sup> Axel Guinot  <sup>1</sup> and Rachel Mandelbaum 

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Published: unpublished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

## <sup>6</sup> Summary

<sup>7</sup> 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 systematic uncertainties. It is in this context that we present here a simple python-based software package to help in the computation of E-/B-mode decomposition, which can be used for systematic checks or science analysis. As we demonstrate, our implementation has both the high precision required and speed to perform this kind of analysis while avoiding a scenario wherein either numerical precision or computational time is a significant limiting factor.

## <sup>15</sup> Statement of need

<sup>16</sup> 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).  
<sup>17</sup> The COSEBIs rely on very high precision computation requiring more than 80 decimal places.  
<sup>18</sup> 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.

<sup>19</sup> 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.

## <sup>31</sup> Testing setup

<sup>32</sup> 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 Library<sup>1</sup> ([Chisari et al., 2019](#)). The cosmological parameters are taken from Aghanim et al. (2020).  
<sup>33</sup> For tests that involved covariance we are using the Stage-IV Legacy Survey of Space and Time (LSST) Year 10 as a reference. The characteristics are taken from the LSST Dark

<sup>1</sup> <https://github.com/LSSTDESC/CCL>

<sup>37</sup> Energy Science Collaboration (DESC) Science Requirements Document (SRD) ([The LSST](#)  
<sup>38</sup> [Dark Energy Science Collaboration et al., 2021](#)).

## <sup>39</sup> COSEBIs

<sup>40</sup> The COSEBIs are defined as:

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

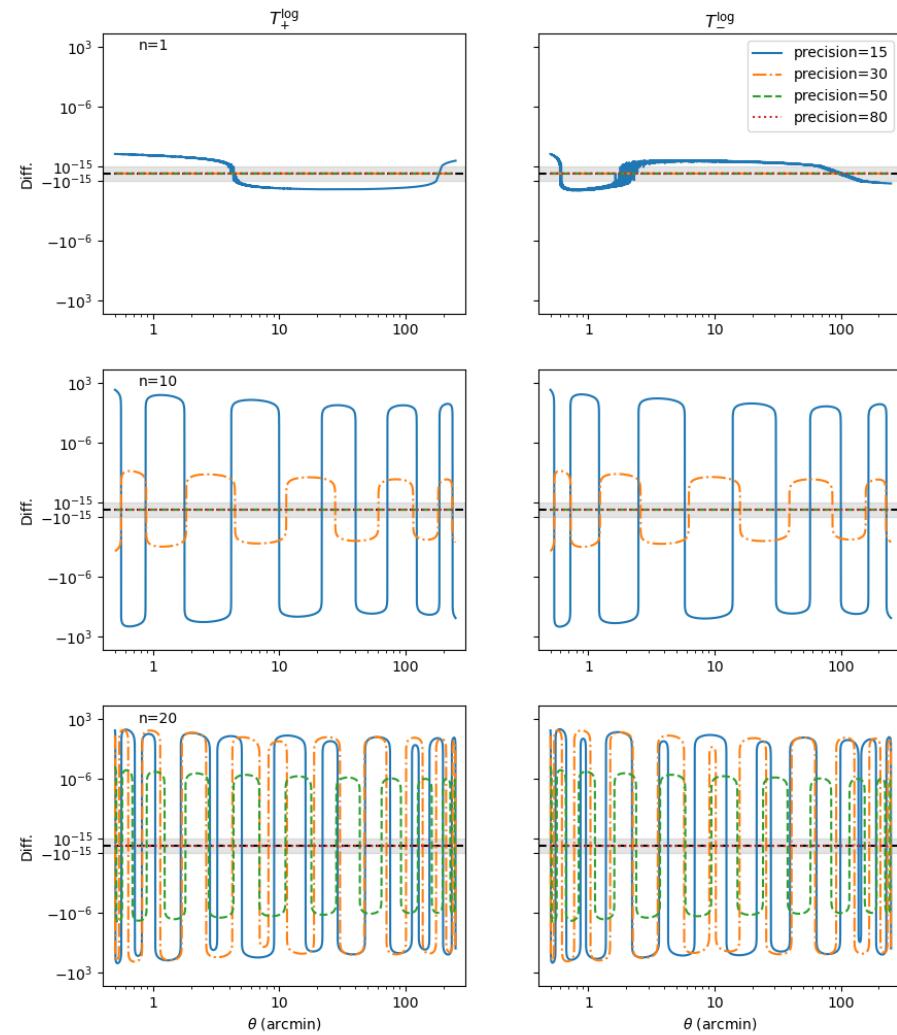
$$B_n = \frac{1}{2} \int_0^\infty d\theta \theta [T_{n,+}(\theta) \xi_+(\theta) - T_{n,-}(\theta) \xi_-(\theta)]; \quad (2)$$

<sup>42</sup> where  $\xi_\pm(\theta)$  are the shear correlation functions, and  $T_{n,\pm}$  are the weight functions for the  
<sup>43</sup> COSEBI mode  $n$ . The complexity is in the computation of the weight functions. Cosmo-numba  
<sup>44</sup> carries out the computation of the weight functions in a logarithmic scale defined by:

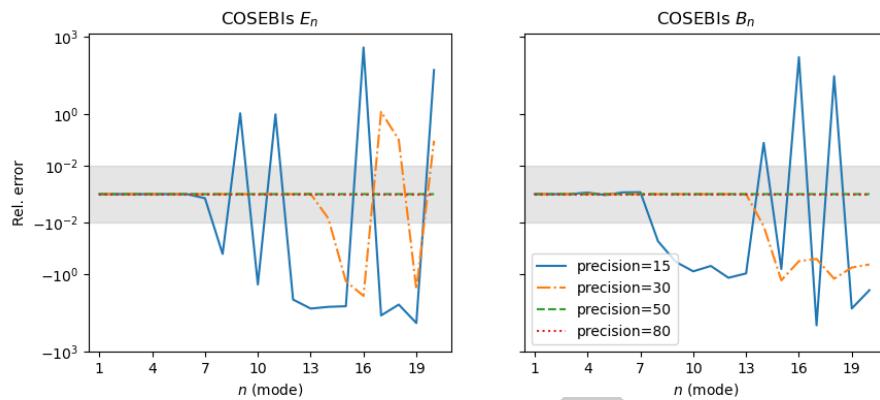
$$T_{n,+}^{\log}(\theta) = t_{n,+}^{\log}(z) = N_n \sum_{j=0}^{n+1} \bar{c}_{nj} z^j; \quad (3)$$

<sup>45</sup> where  $z = \log(\theta/\theta_{\min})$ ,  $N_n$  is the normalization for the mode  $n$ , and  $\bar{c}_{nj}$  are defined iteratively  
<sup>46</sup> from Bessel functions (we refer the readers to P. Schneider et al. ([2010](#)) for more details).

<sup>47</sup> We have validating our implementation against the original version in Mathematica from P.  
<sup>48</sup> Schneider et al. ([2010](#)). In [Figure 1](#) we show the impact of the precision going from 15 decimal  
<sup>49</sup> places, which corresponds to the precision one could achieve using float64, up to 80 decimal  
<sup>50</sup> places, the precision used in the original Mathematica implementation. We can see that classic  
<sup>51</sup> float64 precision would not be sufficient, and with a precision of 80 our code exactly recovers  
<sup>52</sup> the results from the original implementation. Given that the precision comes at very little  
<sup>53</sup> computational cost, we default to the original implementation using high precision. The impact  
<sup>54</sup> of the precision propagated to the COSEBIs is shown in [Figure 2](#). We can see that using a  
<sup>55</sup> lower precision than the default setting can incur a 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, with the shaded region representing the linear scale in the range  $[-10^{-15}, 10^{-15}]$ .



**Figure 2:** This figure shows the difference in the COSEBIs E- and B-modes 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, with the shaded region representing the linear scale in the range  $[-1, 1]$  percent.

<sup>56</sup> COSEBIs can also be defined from the power spectrum as:

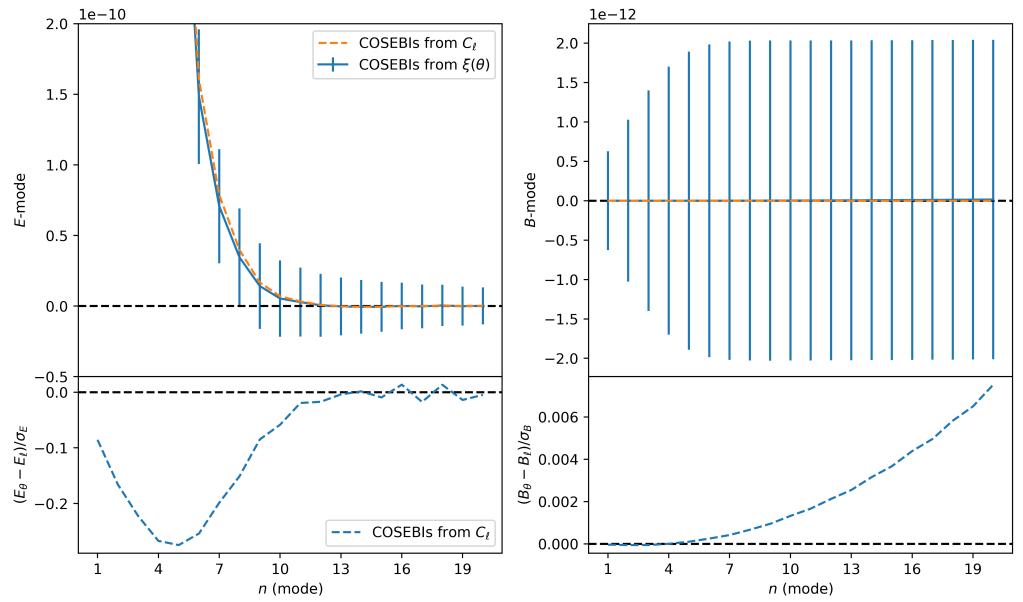
$$E_n = \int_0^\infty \frac{d\ell \ell}{2\pi} P_E(\ell) W_n(\ell); \quad (4)$$

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

<sup>57</sup> <sup>58</sup> where  $P_{E/B}(\ell)$  is the power spectrum of E- and B-modes and  $W_n(\ell)$  are the filter functions  
<sup>59</sup> which can be computed from  $T_{n,+}$  as:

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

<sup>60</sup> with  $J_0(\ell\theta)$  the 0-th order Bessel function. The [Equation 6](#) is a Hankel transform of order 0. It  
<sup>61</sup> can be computed using the FFTLog algorithm presented in [Hamilton \(2000\)](#) implemented here  
<sup>62</sup> in Numba. [Figure 3](#) shows the comparison between the COSEBIs computed from  $\xi_{\pm}(\theta)$  and  
<sup>63</sup> from  $C_{E/B}(\ell)$ . We can see that the COSEBI E- & B-modes agree very well, with at most  $0.3\sigma$   
<sup>64</sup> difference with respect to the LSST Y10 covariance. We consider that using either approach  
<sup>65</sup> would not impact the scientific interpretation and both could be used for consistency checks.



**Figure 3:** Comparison of the COSEBIs E- and B-mode computed from  $\xi_{\pm}(\theta)$  and  $C_{E/B}(\ell)$ . The *upper* panel shows the COSEBIs E-/B-modes while the *bottom* panel shows the difference with respect to the statistical uncertainty based on the LSST Y10 covariance.

## 66 Pure-Mode Correlation Functions

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

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

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

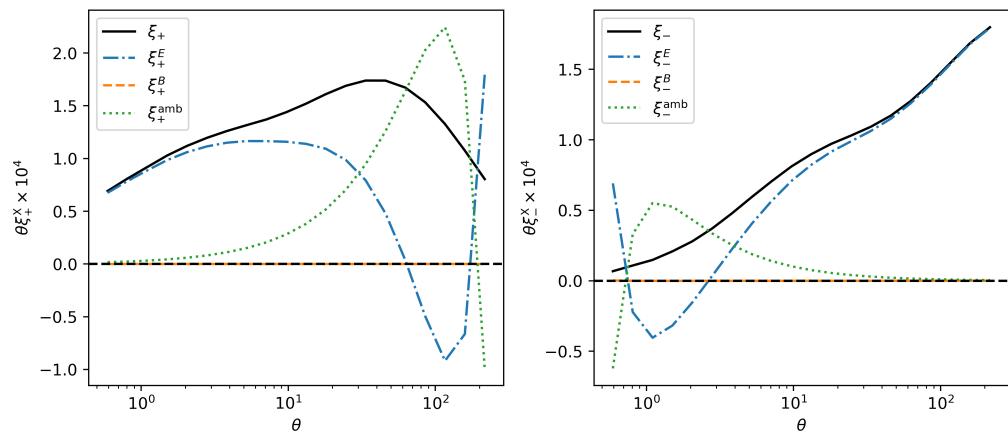
$$\xi_-^E(\vartheta) = \frac{1}{2} \left[ \xi_+(\vartheta) + \xi_-(\vartheta) + \int_{\vartheta_{\min}}^{\vartheta} \frac{d\theta \theta}{\vartheta^2} \xi_+(\theta) \left( 4 - \frac{12\theta^2}{\vartheta^2} \right) \right] - \frac{1}{2} [V_+(\vartheta) + V_-(\vartheta)], \quad (9)$$

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

71 where  $\xi_{\pm}(\theta)$  correspond to the shear-shear correlation function. The functions  $S_{\pm}(\theta)$  and  $V_{\pm}(\theta)$   
 72 are themselves defined by integrals and we refer the reader to Peter Schneider et al. (2022)  
 73 for more details about their definition. By contrast with the computation of the COSEBIs,  
 74 these integrals are more stable and straightforward to compute but still require some level  
 75 of precision. This is why we are using the qags method from QUADPACK<sup>2</sup> (Piessens et al.,  
 76 2012) with a 5th order spline interpolation. In addition, as one can see from the equations

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

77 above, the implementation requires a loop over a range of  $\vartheta$  values. This is why having a fast  
 78 implementation will be required if one wants to use those correlation functions in cosmological  
 79 inference. In Figure 4 we show the decomposition of the shear-shear correlation function into  
 80 the E-/B-modes correlation functions and ambiguous mode.



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

## 81 Acknowledgements

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

## 84 References

- 85 Aghanim, N., Akrami, Y., Ashdown, M., Aumont, J., Baccigalupi, C., Ballardini, M., Banday,  
 86 A. J., Barreiro, R. B., Bartolo, N., Basak, S., Battye, R., Benabed, K., Bernard, J.-P.,  
 87 Bersanelli, M., Bielewicz, P., Bock, J. J., Bond, J. R., Borrill, J., Bouchet, F. R., ...  
 88 Zonca, A. (2020). Planck 2018 results: VI. Cosmological parameters. *Astronomy & Astrophysics*, 641, A6. <https://doi.org/10.1051/0004-6361/201833910>
- 90 Chisari, N. E., Alonso, D., Krause, E., Leonard, C. D., Bull, P., Neveu, J., Villarreal, A., Singh,  
 91 S., McClintock, T., Ellison, J., Du, Z., Zuntz, J., Mead, A., Joudaki, S., Lorenz, C. S.,  
 92 Tröster, T., Sanchez, J., Lanusse, F., Ishak, M., ... Wagoner, E. L. (2019). Core cosmology  
 93 library: Precision cosmological predictions for LSST. *The Astrophysical Journal Supplement Series*, 242(1), 2. <https://doi.org/10.3847/1538-4365/ab1658>
- 95 Hamilton, A. J. S. (2000). Uncorrelated modes of the non-linear power spectrum. *Monthly  
 96 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.<br/>
    97 1365-8711.2000.03071.x)
- 98 Piessens, R., Doncker-Kapenga, E. de, Überhuber, C. W., & Kahaner, D. K. (2012). *Quadrpack:  
 99 A subroutine package for automatic integration* (Vol. 1). Springer Science & Business  
 100 Media.
- 101 Schneider, Peter, Asgari, M., Jozani, Y. N., Dvornik, A., Giblin, B., Harnois-Déraps, J.,  
 102 Heymans, C., Hildebrandt, H., Hoekstra, H., Kuijken, K., Shan, H., Tröster, T., &  
 103 Wright, A. H. (2022). Pure-mode correlation functions for cosmic shear and application to  
 104 KiDS-1000. *Astronomy & Astrophysics*, 664, A77. [https://doi.org/10.1051/0004-6361/202142479](https://doi.org/10.1051/0004-<br/>
    105 6361/202142479)

<sup>106</sup> Schneider, P., Eifler, T., & Krause, E. (2010). COSEBIs: Extracting the full e-/b-mode  
<sup>107</sup> information from cosmic shear correlation functions. *Astronomy and Astrophysics*, 520,  
<sup>108</sup> A116. <https://doi.org/10.1051/0004-6361/201014235>

<sup>109</sup> The LSST Dark Energy Science Collaboration, Mandelbaum, R., Eifler, T., Hložek, R., Collett,  
<sup>110</sup> T., Gawiser, E., Scolnic, D., Alonso, D., Awan, H., Biswas, R., Blazek, J., Burchat, P.,  
<sup>111</sup> Chisari, N. E., Dell'Antonio, I., Digel, S., Frieman, J., Goldstein, D. A., Hook, I., Ivezić,  
<sup>112</sup> Ž., ... Troxel, M. A. (2021). *The LSST dark energy science collaboration (DESC) science*  
<sup>113</sup> *requirements document*. <https://arxiv.org/abs/1809.01669>

DRAFT