PyLlama: a stable and versatile Python toolkit for the electromagnetic modelling of multilayered anisotropic media

Mélanie M. Bay, Silvia Vignolini, Kevin Vynck

Version of Record: https://www.sciencedirect.com/science/article/pii/S0010465521003684
Manuscript_756de8f2a8f5316054f85f1c2dc9fd64

PyLlama: a stable and versatile Python toolkit for the electromagnetic modeling of multilayered anisotropic media

Mélanie M. Bay[a], Silvia Vignolini[a], Kevin Vynck[b,*]

[a] *Department of Chemistry, University of Cambridge, Lensfield Road, Cambridge, CB2 1EW, UK*
[b] *Laboratoire Photonique Numérique et Nanosciences (LP2N), CNRS, IOGS, Univ. Bordeaux, 33400 Talence, France*

Abstract

PyLlama is a handy Python toolkit to compute the electromagnetic reflection and transmission properties of arbitrary multilayered linear media, including the case of anisotropy. Relying on a 4×4-matrix formalism, PyLlama implements not only the transfer matrix method, that is the most popular choice in existing codes, but also the scattering matrix method, which is numerically stable in all situations (e.g., thick, highly birefringent cholesteric structures at grazing incident angles). PyLlama is also designed to suit the practical needs by allowing the user to create, edit and assemble layers or multilayered domains with great ease. In this article, we present the electromagnetic theory underlying the transfer matrix and scattering matrix methods and outline the architecture and main features of PyLlama. Finally, we validate the code by comparison with available analytical solutions and demonstrate its versatility and numerical stability by modelling cholesteric media of varying complexity. A detailed documentation and tutorial are provided in a separate user manual. Applications of PyLlama range from the design of optical components to the modelling of polaritonic effects in polar crystals, to the study of structurally coloured materials in the living world.

Keywords: multilayers; anisotropic optical materials; optical modelling;

*Corresponding author.
E-mail address: kevin.vynck@univ-lyon1.fr; *Current address:* Institute of Light and Matter (ILM), CNRS, University Lyon 1, 69100 Villeurbanne, France

photonic crystals; cholesterics; surface phonon polaritons

PROGRAM SUMMARY

Program Title: PyLlama – Python Toolkit for the Electromagnetic Modelling of Multilayered Anisotropic Media
Developers repository link: `https://github.com/VignoliniLab/PyLlama`
Licensing provisions: GPLv3
Programming language: Python
Supplementary material: User guide and tutorials at `https://pyllama.readthedocs.io/`
Nature of problem: Computation of the optical reflection and transmission coefficients of arbitrary multilayered linear media, composed of an arbitrary number of layers, possibly mixing isotropic and anisotropic, absorbing and non-absorbing materials, for linearly or circularly polarized light.
Solution method: Implementation of both the transfer matrix method (faster) and the scattering matrix method (more robust) relying on a 4×4 matrix formalism.
Additional comments including Restrictions and Unusual features: Integration of a physical model to handle cholesteric structures, blueprint for the integration of user-created custom systems, hassle-free export of spectra for non-programmers even for complex and/or custom systems. External routines include: Numpy [1], Scipy [2], as well as Sympy [3] (optional).

References

[1] Numpy, `https://numpy.org/`

[2] Scipy, `https://www.scipy.org/`

[3] Sympy, `https://www.sympy.org/`

1. Introduction

Multilayered media made of anisotropic materials are widespread in nature [1] and in nanotechnologies, especially for optoelectronic [2], optical and photonic applications [3, 4]. For instance, chiral nematic (or cholesteric) structures consist in birefringent units arranged into periodic helicoidal architectures, which can selectively reflect circularly-polarised light in a specific wavelength range [5, 6, 7, 8, 9, 10]. In living organisms, these structures are responsible for the vibrant blue of *Pollia condensata* fruits [11] and the

shiny green of *Scarabeid* beetles [12]. They can also be fabricated through self-assembly mechanisms from bio-compatible materials, such as cellulose nanocrystals [13], cellulose derivatives [14] or their composites [15, 16], with applications in photonic pigments, color printing and optical sensors [13, 17, 18, 19, 16]. In optical engineering, multilayered anisotropic media have been used to realize linear polarizers, waveplates and birefrigent filters for laser and telecommunication technologies, imaging systems, displays or gas sensing [3, 20]. More recently, they opened new perspectives for subdiffraction wave focusing [21, 22] thanks to surface phonon polaritons [23] and low-loss surface wave guiding [24, 25].

Analytical solutions for the optical properties of anisotropic multilayers are only available in specific situations (e.g., for periodic cholesterics, at normal incidence [6, 26] or at oblique incidence with certain restrictions [7, 27]). A rigorous and general theoretical framework was established with the seminal contributions of Billard [28], Teitler and Henvis [29], and Berreman [30], who expressed the electromagnetic problem as a 4×4 matrix ordinary differential equation. 4×4 matrix formalisms experienced many developments over the years [31, 32, 33, 34, 35, 36, 37, 38], including faster algorithms [39] and a correct treatment of degeneracies causing singularities [40, 41]. Besides commercial solutions, such as the powerful program WVASE [42] dedicated to ellipsometry analysis [43], freely-available and open-access codes [44, 45] generally exploit the elegant transfer matrix method [46] to propagate the solution from layer to layer, as originally proposed by Berreman [30]. Unfortunately, such an approach can become numerically unstable for large systems in presence of evanescent modes due to the coexistence of exponentially decaying and growing waves (in the forward propagating direction) in the multilayer. This shortcoming was addressed by Ko and Sambles [33], who suggested to use a scattering matrix to treat all evanescent modes as forward or backward decaying waves. Scattering matrix formalisms have gained popularity in the framework of the rigorous coupled wave analysis (RCWA) for periodically-corrugated multilayered media [47, 48, 49, 50, 51] to avoid numerical instabilities. Some freely-available RCWA codes [52, 53, 54] can be used for uncorrugated multilayered anisotropic media, yet with an unnecessary complexity. The scattering matrix method has been implemented recently for the study of cholesterics but at normal incidence only [55].

Here, we present a freely-available and open-source Python toolkit named

"PyLlama" [1] that implements both the transfer matrix and the scattering matrix methods to compute the reflection and transmission properties of arbitrary (linear and uncorrugated) multilayered media in all possible situations. While the transfer matrix method proves being faster, the scattering matrix method ensures numerical stability in all cases. PyLlama can thus deal with systems composed of an arbitrary number of layers of any thicknesses, mixing isotropic and anisotropic, absorbing and non-absorbing materials, studied at all possible incident angles and wavelengths for both linear and circular polarizations. Surface waves may be excited by modelling the Otto-Kretschmann configurations [56, 41].

In addition to robustness, we designed the code to be convenient to use and included tools that are particularly suitable for the modelling of cholesteric liquid crystals. First, layers are treated as building blocks. The optical calculations occur internally at the layer level and the user can write their own routines, in form of new classes, to pile up layers into stacks according the parameters of their choice without having to handle optical parameters. Second, the code is organised in modules. Periodic structures can be easily combined into master structures with sub-periodicities, and different pre-defined structures can be stacked onto each other, in order to enable the modelling of complex situations, such as beetle cuticles made of a combination of cholesterics and absorbing layers [12, 57] or stacks of cellulose nanocrystal cholesteric structures separated by a nematic layer [58]. Third, once the user has defined which structures they need to model, a level of automation has been added to allow for the calculation of spectra and the export in MATLAB and/or Pickles with only a few lines of code. Lastly, more specific to cholesterics, we incorporated a physical model [10] that describes the cholesteric helicoid in various situations (choice of handedness, tilt, vertical compression of tilted helicoids). In particular, this enables the optical modelling (at normal and oblique incidences) of distorted cholesterics for which the distortion impacts the angular response, polarisation selectivity, and overall reflection spectrum of the structure [59, 10, 55, 60, 16].

PyLlama addresses a need from the scientific community for a simple, robust and flexible program, with many possibilities for future developments and improvements thanks to its object-oriented organisation and detailed documentation.

[1] Python toolkit for multilayered anisotropic media

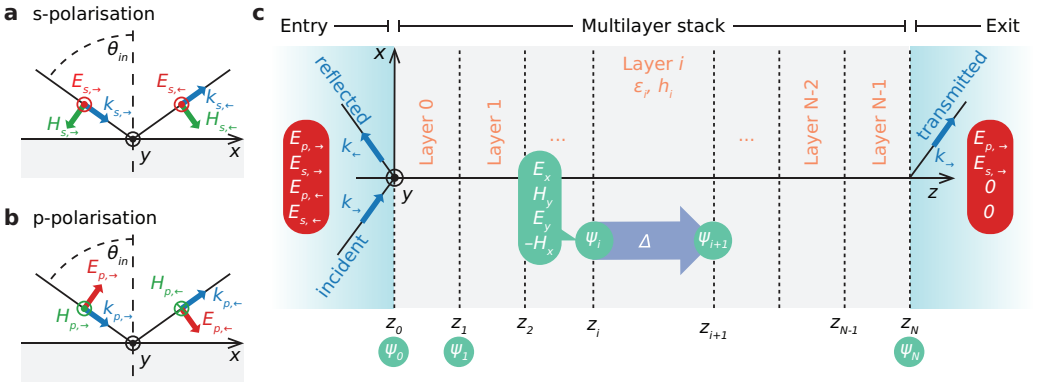

Figure 1: a, b) Schematic of an incident and a reflected s-polarised waves (a) or p-polarised waves (b) upon the stack. The plane of incidence is the (x, z) plane. The wavevectors k_{i} and k_{r} form an angle θ with the z-axis. c) Schematic of the multilayer stack bounded by the entry and exit half-spaces. This schematic clarifies the notations used in the main text.

The following sections are organized as follows. Section 2 describes the electromagnetic theory underlying the 4×4 matrix formalism, as well as the transfer matrix and scattering matrix methods to propagate the solution throughout the multilayer stack. Section 3 outlines the architecture and main features of PyLlama. Several practical examples of optical computations and comparisons with analytical solutions are given to illustrate the method and show its versatility. The performance and stability of the transfer matrix and scattering matrix methods are compared in Section 4 on the example of a cholesteric structure. Finally, we provide concluding remarks in Section 5. The Appendices contain details about the numerical analysis of the so-called partial waves in optical layers, a description of the discrete model for cholesterics, and all the material and wave parameters used to construct the figures.

2. Theory

2.1. Description of the problem

We consider a multilayer stack composed of N layers perpendicular to the z axis and translationally-invariant in the (xy) plane, see Fig. 1. The layers are indexed from 0 to $N-1$ to match Python's indexing convention. Each layer i is bounded by two interfaces at z_i and z_{i+1}, with an arbitrary thickness h_i, and described by an arbitrary permittivity tensor $\boldsymbol{\epsilon}_i$, which

may contain complex values to describe absorbing materials. The multilayer stack is enclosed between two semi-infinite isotropic media called "entry" and "exit" with refractive indices n_{entry} and n_{exit}, respectively.

The multilayer stack is illuminated from the entry half-space by a planewave at frequency ω propagating in the (xz)-plane at an angle θ_{in} from the surface normal z. The wavevector $\mathbf{k}$ of the incident planewave is thus given by

$$\mathbf{k} = \begin{bmatrix} k_x \\ k_y \\ k_z \end{bmatrix} = k_0 \begin{bmatrix} K_x \\ K_y \\ K_z \end{bmatrix}, \tag{1}$$

with $k_0 = \frac{2\pi}{\lambda}$. Here, $K_x = n_{\text{entry}} \sin(\theta_{\text{in}})$ and $K_y = 0$ are constant throughout the multilayer stack.

In the isotropic half-spaces and layers, planewaves can be decomposed into s and p polarisations, see Fig. 1(a,b). The s-polarised wave has its electric field perpendicular (s from the German *senkrecht*, perpendicular) to the plane of incidence (xz), along y, and the p-polarised wave has its electric field in the plane of incidence (xz). Of particular relevance for helicoidal structures, the waves can also be described in a circular polarisation basis denoted as R and L for the right and left-handed polarisations, respectively. Our objective here is to compute the complex reflection and transmission coefficients of the multilayer stack, defined as r_{kj} and t_{kj}, where j indicates the polarisation of the incident wave, either in the linear or the circular polarisation basis, and k indicates the polarisation of the reflected or transmitted wave, respectively. Note that for multilayers containing anisotropic layers, the cross-terms in the linear polarisation basis (r_{ps}, r_{sp}, t_{ps} and t_{sp}) can be non-zero. The reflectance and the transmittance can then be calculated respectively as $R_{kj} = |r_{kj}|^2$ and $T_{kj} = |t_{kj}|^2 \cos(\theta_{\text{in}})/\cos(\theta_{\text{out}})$ [61], where $\theta_{\text{out}} = \text{asin}\left[n_{\text{entry}} \sin(\theta_{\text{in}})/n_{\text{exit}}\right]$ is the angle of the planewave transmitted in the exit half-space.

2.2. 4×4 *matrix formalism*

Maxwell's equations set relationships between the 3 components of the electric field and the 3 components of the magnetic field of a lightwave in a

given medium, which, in a matrix form, read

$$\begin{bmatrix} 0 & 0 & 0 & 0 & -\frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ 0 & 0 & 0 & \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial x} \\ 0 & 0 & 0 & -\frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial z} & -\frac{\partial}{\partial y} & 0 & 0 & 0 \\ -\frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} & 0 & 0 & 0 \\ \frac{\partial}{\partial y} & -\frac{\partial}{\partial x} & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} E_x \\ E_y \\ E_z \\ H_x \\ H_y \\ H_z \end{bmatrix} = \frac{1}{c}\frac{\partial}{\partial t} \begin{bmatrix} D_x \\ D_y \\ D_z \\ B_x \\ B_y \\ B_z \end{bmatrix}. \tag{2}$$

Introducing the permittivity $\boldsymbol{\epsilon}$, permeability $\boldsymbol{\mu}$ and optical activity $\boldsymbol{\rho}$ and $\boldsymbol{\rho}'$ tensors of the material to relate the $\mathbf{D}$ and $\mathbf{B}$ fields to the $\mathbf{E}$ and $\mathbf{H}$ fields, applying translational-invariance in the x and y directions and considering harmonic fields (the $\exp(-i\omega t)$ convention is used hereafter), Eq. (2) can be rewritten as

$$\begin{bmatrix} 0 & 0 & 0 & 0 & -\frac{\partial}{\partial z} & 0 \\ 0 & 0 & 0 & \frac{\partial}{\partial z} & 0 & -iK_x \\ 0 & 0 & 0 & 0 & iK_x & 0 \\ 0 & \frac{\partial}{\partial z} & 0 & 0 & 0 & 0 \\ -\frac{\partial}{\partial z} & 0 & iK_x & 0 & 0 & 0 \\ 0 & -iK_x & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} E_x \\ E_y \\ E_z \\ H_x \\ H_y \\ H_z \end{bmatrix} = -\frac{i\omega}{c} \begin{bmatrix} \epsilon_{xx} & \epsilon_{xy} & \epsilon_{xz} & \rho_{xx} & \rho_{xy} & \rho_{xz} \\ \epsilon_{yx} & \epsilon_{yy} & \epsilon_{yz} & \rho_{yx} & \rho_{yy} & \rho_{yz} \\ \epsilon_{zx} & \epsilon_{zy} & \epsilon_{zz} & \rho_{zx} & \rho_{zy} & \rho_{zz} \\ \rho'_{xx} & \rho'_{xy} & \rho'_{xz} & \mu_{xx} & \mu_{xy} & \mu_{xz} \\ \rho'_{yx} & \rho'_{yy} & \rho'_{yz} & \mu_{yx} & \mu_{yy} & \mu_{yz} \\ \rho'_{zx} & \rho'_{zy} & \rho'_{zz} & \mu_{zx} & \mu_{zy} & \mu_{zz} \end{bmatrix} \begin{bmatrix} E_x \\ E_y \\ E_z \\ H_x \\ H_y \\ H_z \end{bmatrix}. \tag{3}$$

Expressing the normal (z) components as a function of the tangential (x, y) components, taking out the third and six rows and rearranging the matrices finally leads to the famous 4×4 matrix ordinary differential equation [28, 29, 30]

$$\frac{\partial \boldsymbol{\Psi}(z)}{\partial z} = \frac{i\omega}{c}\boldsymbol{\Delta}(z)\boldsymbol{\Psi}(z), \tag{4}$$

where $\boldsymbol{\Psi} = [E_x, H_y, E_y, -H_x]^{\mathrm{T}}$ is a vector describing the field at position z.[2]

The elements of the matrix $\boldsymbol{\Delta}$ depend on the wavevector constant component K_x and on the material permittivity, permeability and optical activity. When the material is non-magnetic and non-optically active, the matrix $\boldsymbol{\Delta}(z)$

[2]The field can indifferently be represented by the vector $\boldsymbol{\Psi} = [E_x, H_y, E_y, -H_x]^{\mathrm{T}}$ as in Refs. [30, 35, 39, 41, 55], or by $\boldsymbol{\Psi} = [E_x, E_y, H_x, H_y]^{\mathrm{T}}$ as in Ref. [44].

explicitly reads

$$\boldsymbol{\Delta}(z) = \begin{bmatrix} -K_x\frac{\epsilon_{zx}}{\epsilon_{zz}} & 1-\frac{K_x^2}{\epsilon_{zz}} & -K_x\frac{\epsilon_{zy}}{\epsilon_{zz}} & \frac{K_x}{\epsilon_{zz}} \\ \epsilon_{xx}-\frac{\epsilon_{xz}\epsilon_{zx}}{\epsilon_{zz}} & -K_x\frac{\epsilon_{xz}}{\epsilon_{zz}} & \epsilon_{xy}-\frac{\epsilon_{xz}\epsilon_{zy}}{\epsilon_{zz}} & \frac{\epsilon_{xz}}{\epsilon_{zz}} \\ 0 & 0 & 0 & 1 \\ \epsilon_{yx}-\frac{\epsilon_{yz}\epsilon_{zx}}{\epsilon_{zz}} & -K_x\frac{\epsilon_{yz}}{\epsilon_{zz}} & \epsilon_{yy}-K_x^2-\frac{\epsilon_{yz}\epsilon_{zy}}{\epsilon_{zz}} & \frac{\epsilon_{yz}}{\epsilon_{zz}} \end{bmatrix}, \tag{5}$$

where the z-dependence is in the permittivity tensor $\boldsymbol{\epsilon}$.

2.3. Propagation in a homogeneous layer

Let us now consider a homogeneous layer i located between two interfaces at positions z_i and z_{i+1}, see Fig. 1(c). The corresponding 4×4 matrix $\boldsymbol{\Delta}_i$ is then constant throughout the layer thickness and Eq. (4) directly leads to

$$\boldsymbol{\Psi}(z_{i+1}) = \exp\left(ik_0 h_i \boldsymbol{\Delta}_i\right)\boldsymbol{\Psi}(z_i) = \mathbf{R}_i\boldsymbol{\Psi}(z_i) \tag{6}$$

where $h_i = z_{i+1} - z_i$ is the layer thickness and $\mathbf{R}_i$ describes the propagation from z_i to z_{i+1}.

Discarding the layer label i for the sake of legibility, there are different ways to calculate the propagator $\mathbf{R}$:

1. Expanding $\mathbf{R}$ as an infinite sum, as $\exp\left(ik_0h\boldsymbol{\Delta}\right) = 1 + ik_0h\boldsymbol{\Delta} - \frac{1}{2}k_0^2h^2\boldsymbol{\Delta}^2 + ...$ as suggested by Berreman [30], or via Pad's decomposition as in Ref. [44]. This is directly implemented in the matrix exponential function `expm` in Pythons package Scipy (version 1.4.1) and in MATLAB (version R2019b).
2. Factorizing $\boldsymbol{\Delta} = \mathbf{P}\mathbf{Q}_{\boldsymbol{\Delta}}\mathbf{P}^{-1}$ where $\mathbf{Q}_{\boldsymbol{\Delta}}$ is a diagonal matrix containing the eigenvalues and $\mathbf{P}$ the eigenvectors of $\boldsymbol{\Delta}$. This leads to an eigendecomposition of $\mathbf{R}$ as $\mathbf{R} = \mathbf{P}\mathbf{Q}\mathbf{P}^{-1}$, with identical eigenvectors and eigenvalues straightforwardly related those of $\boldsymbol{\Delta}$, see below. This was also suggested by Berreman [30] and is implemented in Passler and Paarmann's code [41, 45].
3. Using an algorithm based on Sylvester's formulas applied to biaxial crystals, as proposed by Palto *et al.* [39], which is expected to converge faster and requires less calculation steps than Pad's decomposition.

In the second approach, which we will follow here, the fields in the layer are decomposed into four *partial waves* [61]. The eigendecomposition has a

clear physical meaning, as sketched in Fig. 2a. The matrix $\mathbf{P}$ contains the eigenvectors $\mathbf{p}_j$ of $\mathbf{\Delta}$ as

$$\mathbf{P} = \begin{bmatrix} \mathbf{p}_0 & \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 \end{bmatrix}, \tag{7}$$

where $\mathbf{p}_j = [E_{j,x}, H_{j,y}, E_{j,y}, -H_{j,x}]^{\mathrm{T}}$ in $\mathbf{P}$ thus represent the partial wave field components in the homogeneous layer. The diagonal matrix $\mathbf{Q} = \exp(ik_0 h \mathbf{Q}_{\mathbf{\Delta}})$ depends on the corresponding eigenvalues q_j of $\mathbf{\Delta}$ as

$$\mathbf{Q} = \begin{bmatrix} e^{ik_0 h q_0} & 0 & 0 & 0 \\ 0 & e^{ik_0 h q_1} & 0 & 0 \\ 0 & 0 & e^{ik_0 h q_2} & 0 \\ 0 & 0 & 0 & e^{ik_0 h q_3} \end{bmatrix}. \tag{8}$$

It thus describes the coherent propagation of the partial waves in the layer, where the eigenvalues q_j represent the z-components of the corresponding wavevectors $K_{j,z}$.

The four partial waves can generally be represented as a pair of waves travelling forward (towards $+z$, subscript $\rightarrow$) and a pair of partial waves that travel backward (towards $-z$, subscript $\leftarrow$). The two partial waves within each pair can then be identified according to their polarisation. In PyLlama, the partial waves are sorted as follows

- $j = 0$: forward direction ($\rightarrow$), mostly polarised along the x axis,
- $j = 1$: forward direction ($\rightarrow$), mostly polarised along the y axis,
- $j = 2$: backward direction ($\leftarrow$), mostly polarised along the x axis,
- $j = 3$: backward direction ($\leftarrow$), mostly polarised along the y axis

but this is only one possibility [3]. Details about the numerical analysis of the partial waves is provided in Appendix A.

Let us emphasize that the sorting is quite irrelevant for obtaining the transfer and scattering matrices of the multilayer stack (when one does not wish to extract transfer and scattering matrices for subsets of the system), the only constraint being that the intermediate matrices used in the calculation be invertible. The sorting allows avoiding such situations.

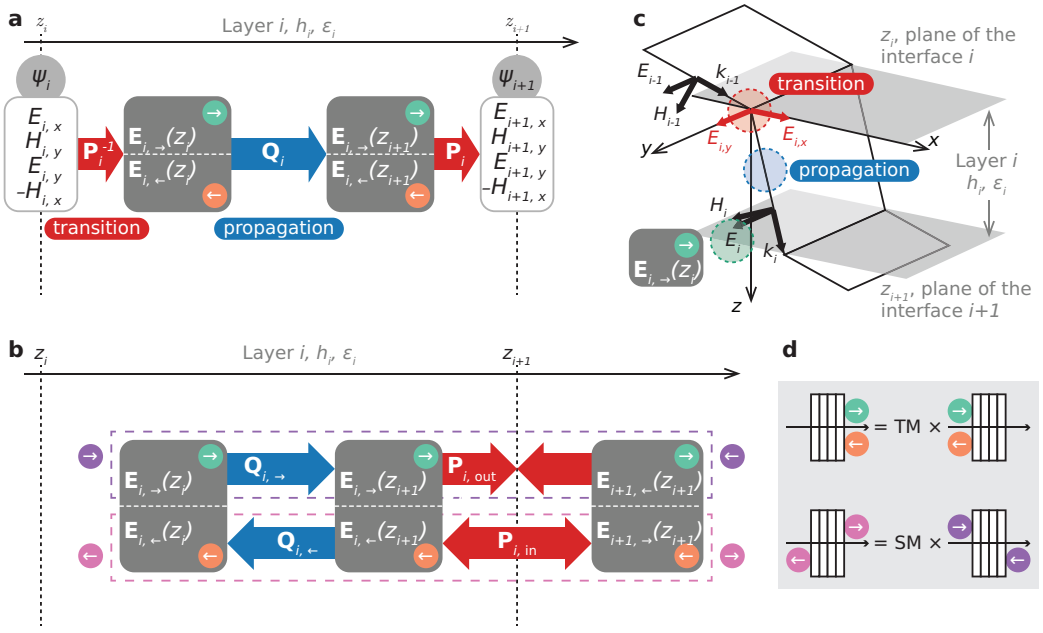

Figure 2: a, b) Schematic representations of the variables used to describe the light interaction with the layer i a) with the transfer matrix method and b) with the scattering matrix method. Both methods are based on the phenomena of transition and propagation, which are schematised with wide red or blue arrows labelled with the corresponding matrices used in the equations. c) 3D representation of a layer and its interfaces. Transition occurs at the interface between two layers and propagation occurs within the thickness of the layer. d) Schematic representation of the transfer matrix method and the scattering matrix method.

2.4. Transfer matrix method

The computation of $\mathbf{R}_i$ for layer i allows us to relate, via Eq. (6), the tangential components of the electric and magnetic fields at the interface z_i to those at the interface z_{i+1}.

Imposing the continuity of the field tangential components at each interface naturally leads to a generalization of the propagator for a multilayer stack of N layers ($N+1$ interfaces) as [30, 46]

$$\mathbf{\Psi}_N = \mathbf{R}\mathbf{\Psi}_0, \tag{9}$$

with

$$\mathbf{R} = \overset{\curvearrowleft}{\prod_{i=0}^{N-1}} \mathbf{R}_i = \overset{\curvearrowleft}{\prod_{i=0}^{N-1}} \mathbf{P}_i \mathbf{Q}_i \mathbf{P}_i^{-1}, \tag{10}$$

and $\overset{\curvearrowleft}{\prod}_{i=0}^{N-1}$ is a left-side matrix product.[4]

It is usually more convenient to express the electromagnetic field in terms of s and p polarisations in the entry and exit isotropic half-spaces. For a planewave in an isotropic medium, the s- and p-polarised components of the electric and magnetic fields are related to the tangential components of the fields as [61]

$$\begin{bmatrix} E_x \\ H_y \\ E_y \\ -H_x \end{bmatrix} = \mathbf{L} \begin{bmatrix} E_{p,\rightarrow} \\ E_{s,\rightarrow} \\ E_{p,\leftarrow} \\ E_{s,\leftarrow} \end{bmatrix}, \tag{11}$$

with

$$\mathbf{L} = \begin{bmatrix} \cos(\theta) & 0 & \cos(\theta) & 0 \\ n & 0 & -n & 0 \\ 0 & 1 & 0 & 1 \\ 0 & n\cos(\theta) & 0 & -n\cos(\theta) \end{bmatrix}. \tag{12}$$

[3] In Ref. [31], the partial waves are sorted according to their polarisation first and their direction second (which would correspond to the order 0, 2, 1, 3, here).

[4] We define here the matrix $\mathbf{Q}_i$ similarly to Berreman [30], but its inverse ($\mathbf{Q}'_i = \mathbf{Q}_i^{-1}$) is sometimes used in the literature [41, 44]; in this case, the left-side product in Eq. (9) should be replaced by a right-side product $\overset{\curvearrowright}{\prod}_{i=0}^{N-1} \mathbf{P}_i \mathbf{Q}'_i \mathbf{P}_i^{-1}$, leading to a propagator that is the inverse of the one considered here.

Here, $n = \{n_{\text{entry}}, n_{\text{exit}}\}$ and $\theta = \{\theta_{\text{in}}, \theta_{\text{out}}\}$ for the entry and exit media, respectively. Combining the corresponding matrices, $\mathbf{L}_{\text{entry}}$ and $\mathbf{L}_{\text{exit}}$, at both ends of Eq. (9), we can now define the *transfer matrix* $\mathbf{T}$ of the multilayer stack relating the ingoing and outgoing fields at the first interface z_0 to the ingoing and outgoing fields at the last interface z_N,

$$\begin{bmatrix} E_{p,\rightarrow}(z_N) \\ E_{s,\rightarrow}(z_N) \\ E_{p,\leftarrow}(z_N) \\ E_{s,\leftarrow}(z_N) \end{bmatrix} = \mathbf{T} \begin{bmatrix} E_{p,\rightarrow}(z_0) \\ E_{s,\rightarrow}(z_0) \\ E_{p,\leftarrow}(z_0) \\ E_{s,\leftarrow}(z_0) \end{bmatrix}. \tag{13}$$

with

$$\mathbf{T} = \mathbf{L}_{\text{exit}}^{-1} \mathbf{R} \mathbf{L}_{\text{entry}}. \tag{14}$$

The complex reflection and transmission coefficients are finally obtained as

$$\begin{aligned} r_{\text{pp}} =& \frac{T_{30}T_{23} - T_{20}T_{33}}{T_{22}T_{33} - T_{32}T_{23}} \\ r_{\text{ps}} =& \frac{T_{31}T_{23} - T_{21}T_{33}}{T_{22}T_{33} - T_{32}T_{23}} \\ r_{\text{sp}} =& \frac{T_{20}T_{32} - T_{30}T_{22}}{T_{22}T_{33} - T_{32}T_{23}} \\ r_{\text{ss}} =& \frac{T_{21}T_{32} - T_{31}T_{22}}{T_{22}T_{33} - T_{32}T_{23}} \\ t_{\text{pp}} =& T_{00} + T_{02}r_{\text{pp}} + T_{13}r_{\text{sp}} \\ t_{\text{ps}} =& T_{01} + T_{02}r_{\text{ps}} + T_{03}r_{\text{ss}} \\ t_{\text{sp}} =& T_{10} + T_{12}r_{\text{pp}} + T_{03}r_{\text{sp}} \\ t_{\text{ss}} =& T_{11} + T_{12}r_{\text{ps}} + T_{13}r_{\text{ss}} \end{aligned} \tag{15}$$

These coefficients can conveniently be assembled into a Jones matrix and expressed on the circular polarisation basis from [61]

$$\begin{bmatrix} r_{\text{RR}} & r_{\text{RL}} \\ r_{\text{LR}} & r_{\text{LL}} \end{bmatrix} = \left(\begin{bmatrix} 1 & 1 \\ i & -i \end{bmatrix} \right)^{-1} \begin{bmatrix} r_{\text{pp}} & r_{\text{ps}} \\ r_{\text{sp}} & r_{\text{ss}} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -i & i \end{bmatrix} \tag{16}$$

$$\begin{bmatrix} t_{\text{RR}} & t_{\text{RL}} \\ t_{\text{LR}} & t_{\text{LL}} \end{bmatrix} = \left(\begin{bmatrix} 1 & 1 \\ -i & i \end{bmatrix} \right)^{-1} \begin{bmatrix} t_{\text{pp}} & t_{\text{ps}} \\ t_{\text{sp}} & t_{\text{ss}} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -i & i \end{bmatrix} \tag{17}$$

2.5. Scattering matrix method

The scattering matrix method was proposed as a solution to the numerical instabilities encountered by the transfer matrix method in certain situations, such as cholesteric liquid crystals [33]. The scattering matrix relates the waves coming in the multilayer stack from both sides to the waves coming out from it, in contrast with the transfer matrix which relates the waves at the first interface to the waves at the last interface, see Eq. (13).

Following the transfer matrix formalism above, the transition from the set of partial waves in layer i to those in layer $i+1$ can simply be expressed as

$$\mathbf{P}_{i+1}\begin{bmatrix}\mathbf{E}_{i+1,\rightarrow}(z_{i+1})\\ \mathbf{E}_{i+1,\leftarrow}(z_{i+1})\end{bmatrix} = \mathbf{P}_i\mathbf{Q}_i\begin{bmatrix}\mathbf{E}_{i,\rightarrow}(z_i)\\ \mathbf{E}_{i,\leftarrow}(z_i)\end{bmatrix} \tag{18}$$

Knowing the travel direction and the z coordinate at which the electric fields are evaluated enables us to determine whether they are ingoing or outgoing waves, as schematised on Fig. 2b. Equation (18) can be rearranged with linear operations to dispatch the forward- and backward-propagating waves into ingoing and outgoing waves, leading to

$$\begin{aligned}\mathbf{P}_{i,\text{out}}\begin{bmatrix}\mathbf{E}_{i,\rightarrow}(z_{i+1})\\ \mathbf{E}_{i+1,\leftarrow}(z_{i+1})\end{bmatrix} &= \mathbf{P}_{i,\text{in}}\begin{bmatrix}\mathbf{E}_{i+1,\rightarrow}(z_{i+1})\\ \mathbf{E}_{i,\leftarrow}(z_{i+1})\end{bmatrix},\\ \mathbf{E}_{i,\rightarrow}(z_{i+1}) &= \mathbf{Q}_{i,\rightarrow}\mathbf{E}_{i,\rightarrow}(z_i),\\ \mathbf{E}_{i,\leftarrow}(z_{i+1}) &= \mathbf{Q}_{i,\leftarrow}\mathbf{E}_{i,\leftarrow}(z_i),\end{aligned} \tag{19}$$

where the matrices $\mathbf{P}_{i,\text{out}}$, $\mathbf{P}_{i,\text{in}}$, $\mathbf{Q}_{i,\rightarrow}$ and $\mathbf{Q}_{i,\leftarrow}$ are defined as

$$\begin{aligned}\mathbf{P}_{i,\text{out}} &= \begin{bmatrix}\mathbf{p}_{i,0} & \mathbf{p}_{i,1} & -\mathbf{p}_{i+1,2} & -\mathbf{p}_{i+1,3}\end{bmatrix},\\ \mathbf{P}_{i,\text{in}} &= \begin{bmatrix}\mathbf{p}_{i+1,0} & \mathbf{p}_{i+1,1} & -\mathbf{p}_{i,2} & -\mathbf{p}_{i,3}\end{bmatrix},\\ \mathbf{Q}_{i,\rightarrow} &= \begin{bmatrix}e^{ik_0hq_{i,0}} & 0 & 0 & 0\\ 0 & e^{ik_0hq_{1,0}} & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix},\\ \mathbf{Q}_{i,\leftarrow} &= \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & e^{ik_0hq_{i,2}} & 0\\ 0 & 0 & 0 & e^{ik_0hq_{i,3}}\end{bmatrix}.\end{aligned} \tag{20}$$

$\mathbf{E}_{i,\rightarrow}(z_{i+1})$ and $\mathbf{E}_{i,\leftarrow}(z_{i+1})$ can then be eliminated from Eq. (19),

$$\mathbf{P}_{i,\text{out}}\mathbf{Q}_{i,\rightarrow}\begin{bmatrix}\mathbf{E}_{i,\rightarrow}(z_i)\\ \mathbf{E}_{i+1,\leftarrow}(z_{i+1})\end{bmatrix} = \mathbf{P}_{i,\text{in}}\mathbf{Q}_{i,\leftarrow}\begin{bmatrix}\mathbf{E}_{i+1,\rightarrow}(z_{i+1})\\ \mathbf{E}_{i,\leftarrow}(z_i)\end{bmatrix}. \tag{21}$$

This eventually leads to the definition of the *scattering matrix* $\mathbf{S}_{i,i+1}$ between interfaces i and $i+1$, as

$$\begin{bmatrix} \mathbf{E}_{i+1,\rightarrow}(z_{i+1}) \\ \mathbf{E}_{i,\leftarrow}(z_i) \end{bmatrix} = \mathbf{S}_{i,i+1} \begin{bmatrix} \mathbf{E}_{i,\rightarrow}(z_i) \\ \mathbf{E}_{i+1,\leftarrow}(z_{i+1}) \end{bmatrix} \tag{22}$$

with

$$\mathbf{S}_{i,i+1} = \mathbf{Q}_{i,\leftarrow}^{-1} \mathbf{P}_{i,\mathrm{in}}^{-1} \mathbf{P}_{i,\mathrm{out}} \mathbf{Q}_{i,\rightarrow}. \tag{23}$$

Evidently, the combination of scattering matrices to describe multilayer stacks cannot be as straightforward as matrix multiplications in the case of the transfer matrix method. For example, the construction of $\mathbf{S}_{i,i+2}$ would require removing $\mathbf{E}_{i+1,\rightarrow}$ and $\mathbf{E}_{i+1,\leftarrow}$. As above, this can be done by linear operations.

Rewriting the scattering matrices in terms of their 2×2 quadrants,

$$\begin{aligned} \mathbf{S}_{i,i+1} &= \begin{bmatrix} \mathbf{S}_{00}^{(1)} & \mathbf{S}_{01}^{(1)} \\ \mathbf{S}_{10}^{(1)} & \mathbf{S}_{11}^{(1)} \end{bmatrix}, \\ \mathbf{S}_{i+1,i+2} &= \begin{bmatrix} \mathbf{S}_{00}^{(2)} & \mathbf{S}_{01}^{(2)} \\ \mathbf{S}_{10}^{(2)} & \mathbf{S}_{11}^{(2)} \end{bmatrix}, \\ \mathbf{S}_{i,i+2} &= \begin{bmatrix} \mathbf{S}_{00}^{(0)} & \mathbf{S}_{01}^{(0)} \\ \mathbf{S}_{10}^{(0)} & \mathbf{S}_{11}^{(0)} \end{bmatrix}, \end{aligned} \tag{24}$$

eventually leads to the following expressions for the quadrants of $\mathbf{S}_{i,i+2}$,

$$\begin{aligned} \mathbf{S}_{00}^{(0)} &= \mathbf{S}_{00}^{(2)} \mathbf{C}^{-1} \mathbf{S}_{00}^{(1)}, \\ \mathbf{S}_{01}^{(0)} &= \mathbf{S}_{01}^{(2)} + \mathbf{S}_{00}^{(2)} \mathbf{C}^{-1} \mathbf{S}_{01}^{(1)} \mathbf{S}_{11}^{(2)}, \\ \mathbf{S}_{10}^{(0)} &= \mathbf{S}_{10}^{(1)} + \mathbf{S}_{11}^{(1)} \mathbf{S}_{10}^{(2)} \mathbf{C}^{-1} \mathbf{S}_{00}^{(1)}, \\ \mathbf{S}_{11}^{(0)} &= \mathbf{S}_{11}^{(1)} \mathbf{S}_{11}^{(2)} + \mathbf{S}_{11}^{(1)} \mathbf{S}_{10}^{(2)} \mathbf{C}^{-1} \mathbf{S}_{01}^{(1)} \mathbf{S}_{11}^{(2)}, \end{aligned} \tag{25}$$

with

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \mathbf{S}_{01}^{(1)} \mathbf{S}_{10}^{(2)} \tag{26}$$

Carrying this combination for all layers of the stack, including the isotropic entry and exit media (with analytical formulas for their eigenvectors and with their propagation matrix set to identity), and expressing the forward- and backward-propagating electric fields in terms of incident, reflected and

transmitted waves eventually leads to the total scattering matrix of the multilayered stack,

$$\begin{bmatrix} E_{p,\rightarrow}(z_N) \\ E_{s,\rightarrow}(z_N) \\ E_{p,\leftarrow}(z_0) \\ E_{s,\leftarrow}(z_0) \end{bmatrix} = \mathbf{S} \begin{bmatrix} E_{p,\rightarrow}(z_0) \\ E_{s,\rightarrow}(z_0) \\ E_{p,\leftarrow}(z_N) \\ E_{s,\leftarrow}(z_N), \end{bmatrix}. \tag{27}$$

The reflection and transmission coefficients can be then obtained straightforwardly from the scattering matrix elements,

$$\begin{aligned} r_{\mathrm{pp}} &= S_{20} \\ r_{\mathrm{ps}} &= S_{21} \\ r_{\mathrm{sp}} &= S_{30} \\ r_{\mathrm{ss}} &= S_{31} \\ t_{\mathrm{pp}} &= S_{00} \\ t_{\mathrm{ps}} &= S_{01} \\ t_{\mathrm{sp}} &= S_{10} \\ t_{\mathrm{ss}} &= S_{11} \end{aligned} \tag{28}$$

The reflection and transmission coefficients in the circular polarisation basis can finally be obtained from Eqs. (16) and (17), respectively.

3. Implementation

3.1. Code architecture

PyLlama is practically structured to separate the implementation of the theory from the user, and at the same time to allow the user to access details about the optical calculations on demand. As schematised on Fig. 3, the code is organised into:

- Classes that carry the optical calculations: the class `Structure` represents the multilayer stack and lists the layers it consists in; the class `Layer` represents one layer and its partial waves, its child class `HalfSpace` being the equivalent for the entry and exit semi-infinite media; and the class `Wave` represents optical waves.

- Classes that contain routines to construct `Structures` from appropriate parameters. They are children of a master class `Model` and solely serve the purpose of building `Structures` in a way that is easier to manipulate for the user than the class `Structure` itself. A few models are already implemented in PyLlama.

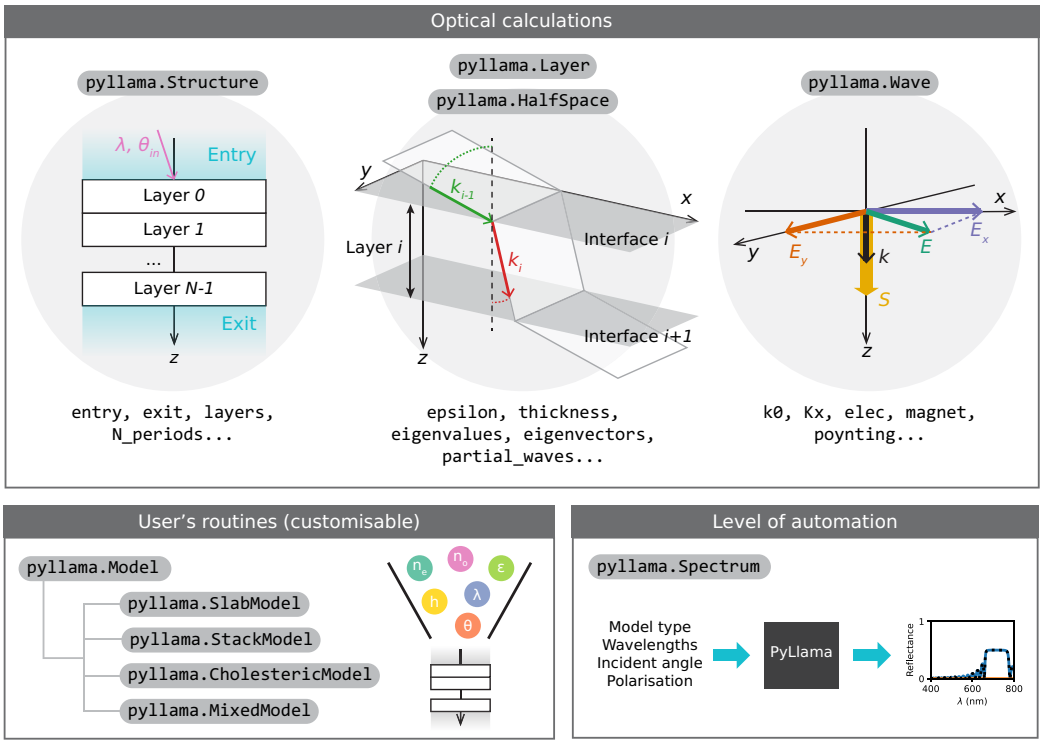

Figure 3: Schematic of the code organisation. The optical calculations are carried out in the classes `Structure`, `Layer` with its subclass `HalfSpace` and `Wave`. The children classes of `Model` generate `Structures` from chosen parameters according to the user's need and are more convenient to handle. Some models are predefined (a slab with `SlabModel`, a periodic stack with `StackModel`, a cholesteric with `CholestericModel` and a combination of several models with `MixedModels`) and the user may write their own additional routines. Lastly, the class `Spectrum` give an additional level of automation and allows the user to use the code as a black box, when and if needed.

- A class `Spectrum` that allows for a further level of automation for common operations such as calculating a full spectrum (in linear or circular polarisation) and exporting the calculated data. This class enables the use of the code as a black-box – though any detail on the optics remains accessible.

3.2. Optical calculations

Each partial wave in PyLlama is represented by the class `Wave` that contains fields for the electric field (`Wave.elec`), the magnetic field (`Wave.magnet`), the Poynting vector (`Wave.poynting`) as well as the constant component of the wavevector $K_x = q_j$ (`Wave.Kx`).

The class `Layer` contains fields for the layer permittivity ϵ (`Layer.eps`), its thickness h (`Layer.thickness`), and the wavevector constant projection on the x axis $k_x = K_x k_0$ (`Layer.Kx` and `Layer.k0`). With these parameters, the 4×4 matrix $\mathbf{\Delta}$ (`Layer.D`) is calculated with the function `Layer.build_D()` from the other fields of `Layer`. The class `Layer` also implements the calculation of the partial waves described in Section 2.3 as a list of four `Wave` instances stored in a field `Layer.partial_waves`. The entry and exit isotropic media are represented by the class `HalfSpace`, which is a children of `Layer` with the only differences residing in the calculation of the eigenvectors (calculated analytically) and eigenvalues (set to 1 for no propagation phenomena).

The eigenvalues and eigenvectors are calculated from the matrix $\mathbf{\Delta}$ (`Layer.D`) with Pythons numerical package Numpy or Pythons symbolic package Sympy. The eigenvalues are unique but the eigenvectors are defined up to a normalisation factor that depends on the algorithm (for example, Numpy normalises each eigenvector so that its module is 1). The eigenvectors calculated with one or another numerical method might therefore differ by a factor -1 or i, which will inhibit their sorting in the way that is described in Appendix A. This does not prevent the use of the transfer or scattering matrix methods to go from one layer to the next. However, the eigenvectors of the entry and exit `HalfSpaces` need to be set in a specific order to allow for the decomposition of the incident, reflected and transmitted electric fields into their appropriate components. In general, if the eigenvectors are calculated analytically (formulas may be found in the literature for specific anisotropic media [61, 39, 34, 55, 35]), the analytical formulas enable to sort them as forward and backward pairs with distinct polarisations (as p and s in the isotropic entry and exit semi-infinite spaces) and this would allow the user to

access intermediate transfer or scattering matrices for specific layers in the stack.

One key point emerging from the 4×4 matrix formalism is the possibility to treat all layers of the stack independently: once the matrix $\mathbf{\Delta}$ has been calculated for each layer (and the eigenvalues, eigenvectors and partial waves that arise from it) as instances of the `Layer` class, the `Layers` can be treated as building blocks to build a stack. In PyLlama, we represent the multilayer stack as a `Structure`, which contains a list of `Layers` (field `Structure.layers`) as well as the entry and exit half-spaces (fields `Structure.entry` and `Structure.exit`). Once all `Layers` of the `Structure` have been characterised, they can be combined together with the half-spaces, either with the transfer matrix method or with the scattering matrix method, to obtain the reflection and transmission coefficients.

The class `Structure` contains several functions that implement the optical calculations:

- `Structure.build_transfer_matrix()` calculates the transfer matrix of the stack (entry half-space, consecutive layers, and exit half-space)
- `Structure._build_scattering_matrix_to_next(this_layer, next_layer)` calculates the scattering matrix for the layer `this_layer` and the interface between `this_layer` and the following layer `next_layer`
- `Structure._combine_scattering_matrices(S_ab, S_bc)` combines the scattering matrices $\mathbf{S}_{a,b}$ `S_ab` and $\mathbf{S}_{b,c}$ `S_bc` into the scattering matrix $\mathbf{S}_{a,c}$
- `Structure.build_scattering_matrix()` calculates the scattering matrix for the complete multilayer stack
- `Structure.get_fresnel(method=<"SM"|"TM">)` calculate the reflection and transmission coefficients, which take the form of a Jones matrix. The user may choose the underlying matrix method with the argument `method`. The reflection coefficients can be converted to circular polarisation with the function `Structure.fresnel_to_fresnel_circ`.
- `Structure.get_refl_trans(method=<"SM"|"TM"|"EM">, circ=<True|False>)` calculates the reflectance and transmittance directly. The argument `circ` enables to choose between linear and circular polarisation bases.

The transfer matrix can also be calculated with the direct exponential of $\mathbf{\Delta}$ (without the diagonalisation) through the choice `"EM"`.

The user manual contains tutorials and a detailed documentation containing all implemented classes and functions as well as their parameters and returns.

3.3. Periodic stacks

Many interesting multilayer stacks are periodic (such as Bragg stacks and cholesterics). Their optical properties can of course be modelled by building a periodic structure "by hand" (e.g., a periodic list of permittivities) but they can also be characterised by their repeating unit to speed up the calculations: transfer matrices and scattering matrices can therefore be calculated for a repeating unit only, and combined together in a second step.

The basic situation is straightforward for the transfer matrix method. As shown in Eq. (9), the propagator $\mathbf{R}$ of a stack of N layers (excluding the entry and exit isotropic half-spaces) is equal to the product of the propagators of all layers, $\mathbf{R} = \overset{\curvearrowleft N-1}{\prod_{i=0}} \mathbf{R}_i$. When the N layers of the stack correspond to N_{per} repeating units made of n layers each, the propagator of the repeating unit is $\mathbf{R}_{\text{per}} = \overset{\curvearrowleft N-1}{\prod_{i=0}} \mathbf{R}_i$ such that the transfer matrix of the N_{per} repeating units is $\mathbf{R} = \overset{\curvearrowleft N_{\text{per}}-1}{\prod_{i=0}} \mathbf{R}_{\text{per}}$.

To speed up the computation time, the decomposition of the stack is different than multiplying one matrix per repeating unit. The total number of repeating units N_{per} is decomposed in powers of two: $N_{\text{per}} = \sum 2^k$ for the appropriate list of k_ps (the list of k_ps associated with N_{per} is the binary form of N_{per}: we obtain it by converting N_{per} to binary). We calculate the transfer matrix for sub-units that contain 2^m repeated units $\mathbf{R}_{\text{per}}^m$ for m between 0 and the higher power of two in the decomposition, and store them in a database. Then, we select the matrices that we need to build up the full system of N_{per} repeated units.

For the scattering method, Eq. (25) shows how to combine $\mathbf{S}_{i,i+1}$ (the scattering matrix of the layer i and its next interface with the layer $i+1$) with $\mathbf{S}_{i+1,i+2}$ (the scattering matrix of the layer $i+1$ and its next interface with the layer $i+2$) into the scattering matrix $\mathbf{S}_{i,i+2}$, which encompasses the layer i, the layer $i+1$, and the interface from the layer $i+1$ to the layer $i+2$. For a periodic stack, the repeating unit starts at the layer 0

and finishes at the interface between the layer $N-1$ layer and the layer 0 of the next repeating unit. However, the last period of the stack does not end by an interface between the layer $N-1$ and the layer 0, but with an interface between the layer $N-1$ and the isotropic half-space. Therefore, the combination of the repeated motive is only done $N_{\text{per}}-1$ times (with the decomposition of $N_{\text{per}}-1$ into powers of two) for the $N_{\text{per}}-1$ first periods, and the last period is added afterwards.

3.4. Useful routines and customisation

In principle, a direct implementation of the theory behind the transfer matrix or the scattering matrix methods is sufficient to obtain the reflectance and transmittance for a multilayer stack once the layers have been properly constructed. However, the construction of the layers from scratch can become tedious. We designed the class `Model` and its children as user-friendly routines that build `Structures` made of appropriate `Layers` and `HalfSpaces`. `Model` is a parent class in which we set features that are common to all models and we wrote a few children classes, `SlabModel`, `StackModel`, `StackOpticalThicknessModel` and `CholestericModel`. The user can therefore write a custom layer-building block of code, without having to re-write any of the underlying theory, with suitable parameters that are relevant to the physical stack being modelled.

A `Model` essentially represents an empty multilayer stack, consisting in two semi-infinite media separated by no layer. The idea behind the code architecture is that the children classes of `Model` contain additional parameters used to build a `Structure` that is relevant to the stack, for example:

- `SlabModel` represents a slab of homogeneous material and its parameters are its permittivity (optionally calculated from an inputted permittivity and a rotation angle (in radians) `rotangle_rad` around an arbitrary axis `rotaxis`) `eps`, its thickness (in nanometers) `thickness`.

- `StackModel` represents an arbitrary multilayer stack and its parameters are a list of permittivities `eps_list` and a list of thicknesses `thickness_list` corresponding to its layers, as well as a number of periods `N_per` that could in principle be set to one if the lists of permittivities and thicknesses contain an item for each layer.

- `StackOpticalThicknessModel` represents a stack of isotropic layers which all have the same optical thickness and its parameters are a

list of refractive indices `n_list` and the total thickness of the stack `total_thickness`.

- `CholestericModel` represents a cholesteric material made of birefringent units and its parameters are a `Cholesteric` object representing the (potentially distorted) helicoidal architecture `cholesteric`, the extraordinary and ordinary refractive indices of the birefringent units `n_e` and `n_o` and the number of repeating units where one unit is `cholesteric`. Our `CholestericModel` class interacts with our `Cholesteric` library: a `Cholesteric` instance is a parameter to build a `CholestericModel`.

Each children class of `Model` contains a redefinition of the parent function `Model.build_structure()` that overrides it and creates `Layers` in a routine that uses the model parameters. The user manual contains tutorials explaining how to create custom children classes appropriately.

3.5. Practical examples

Figure 4 shows examples of results obtained with our different models and validated with available analytical formulas. Figure 4a and b display the case of an isotropic Bragg stack. The system consists in a periodic unit of two layers that are repeated 10 times, including the entry and exit half-spaces that are the same as the first and last layers (with no thickness). The reflection spectra of the isotropic Bragg stack calculated with our code and the model `StackModel` agrees with Yeh's matrix method [61] for s and p-polarisations. Here we display spectra for an angle of incidence $\theta_{\text{in}} = 60°$.

Figure 4c and d display the case of an isotropic non-homogeneous slab analysed by Yeh [61]. The slab has a thickness L in the order of hundreds of nanometers, a refractive index n_0 at $z = 0$ and a refractive index z_s at $z = L$. The refractive index profile across the slab is continuous with an exponential profile defined by:

$$n(z) = n_0 \left(\frac{n_s}{n_0} \right)^{z/L} \tag{29}$$

Yeh gives analytical formulas for the reflection spectrum and also shows analytically that a series of N discrete layers with appropriate thicknesses and refractive indices can be used to model the non-homogeneous slabs, with a precision that increases when the resolution N increases. We used our model

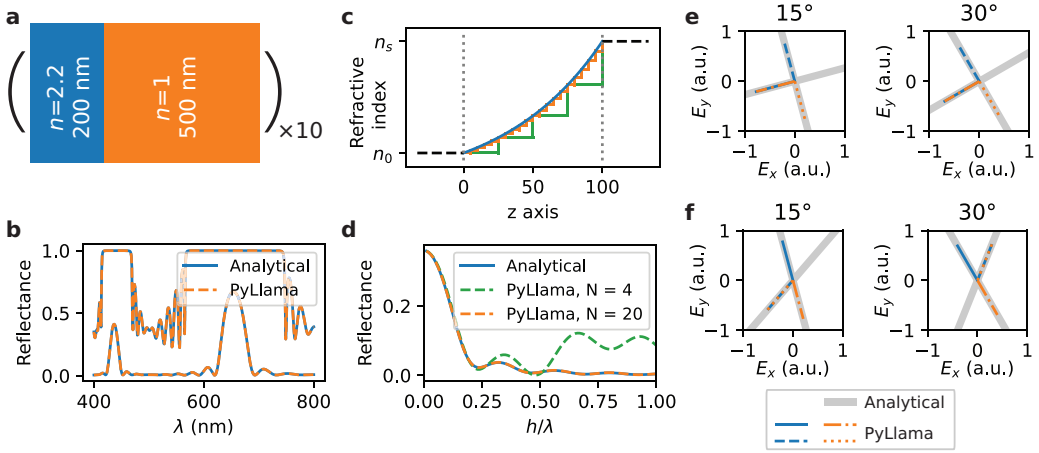

Figure 4: Comparison of PyLlama's results with other methods in the case of isotropic layers and uniaxial crystals. a, b) Isotropic Bragg stack. a) Schematic of the situation: the system consists in a periodic pattern of two layers that are repeated 10 times (the entry and exit half-spaces are the same as the first and last layers). b) Reflection spectrum of the isotropic Bragg stack calculated with PyLlama and with Yeh's matrix method [61] for an angle of incidence $\theta_{\text{in}} = 60°$. c, d) Isotropic non-homogeneous slab with exponentially varying refractive index. c) Refractive index profile: index varying continuously (blue), discretely with 20 layers of the same thickness (orange) and discretely with 4 layers of the same thickness (green). d) Reflectance calculated with PyLlama in the discrete scenarios and with Yeh's analytical formulas in the continuous scenario [61]. e, f) Uniaxial crystals extracted from a cholesteric structure with a rotation angle around the z axis $\phi_i = 15°$ and 30°. The direction of the electric field of the four partial waves calculated numerically with PyLlama matches these calculated analytically [35]. The angle of incidence is e) 0° and f) 60°.

`StackModel` and input a list of N permittivities (calculated from the N refractive indices) and a list of N thicknesses, all identical. We could also have used `StackModel` and calculated the individual thicknesses outside, it was more convenient to create a dedicated model. If we were to work often with non-homogeneous slabs, we could also have imagined a model `NonHomogeneousSlabModel` taking a function $n(z)$ (`lambda z : n0 * (ns / n0) ** (z / L)` in our case) as a parameter and automatically handling the splitting in layers of the same thickness. This illustrate the main idea behind the creation of models: the user can write up the classes (children of `Model`) that they wish in order to model structures by handling the parameters that the user think are convenient.

Figure 4e shows the direction of the eigenvectors computed for a single layer of nematic crystal (taken here as an individual `Layer` in a `Structure` generated by a `CholestericModel`) and compares it with analytical predictions [35]. The agreement is quantitative. It is interesting to note that the electric field follows the cholesteric rotating director at normal incidence, as predicted by De Vries [6] and confirmed here.

Cholesterics are of special concern in the present work. For modelling purposes, these continuous non-homogeneous materials are assimilated to discrete stacks of slices where each slice i (at depth s_i on the helical axis) possesses a director n_i and rotation angle ϕ_i, leading to a permittivity tensor $\epsilon(\phi_i)$. A more general discussion on cholesterics is provided in Appendix B.

The `Cholesteric` library allows the user to manipulate (distort, compress) the structure in a physical way, and then export the slices directors to build up the corresponding multilayer stack from which to calculate the reflectance. Details about the physical models derivation and how it handles vertical compression may be found in Ref. [10]. This enables us to easily model different configurations of practical interest. Figure 5 showcases a few configurations that our `Cholesteric` library can generate and that we can optically model: a right-handed cholesteric, a left-handed cholesteric, oblique incidence of light upon a straight or tilted cholesteric, a vertically compressed cholesteric, and a distorted cholesteric. To account for the tilt of a helicoid, we set the z-axis along the helical axis and we calculate the effective angle of incidence upon the stack with the absolute angle of incidence and the tilt of the `Cholesteric`.

As we can see from the cases represented here, the handedness, the angle of incidence and the distortion strongly impact the peak wavelength, the peak shape and the polarisation selectivity (reflectance from unpolarised light can

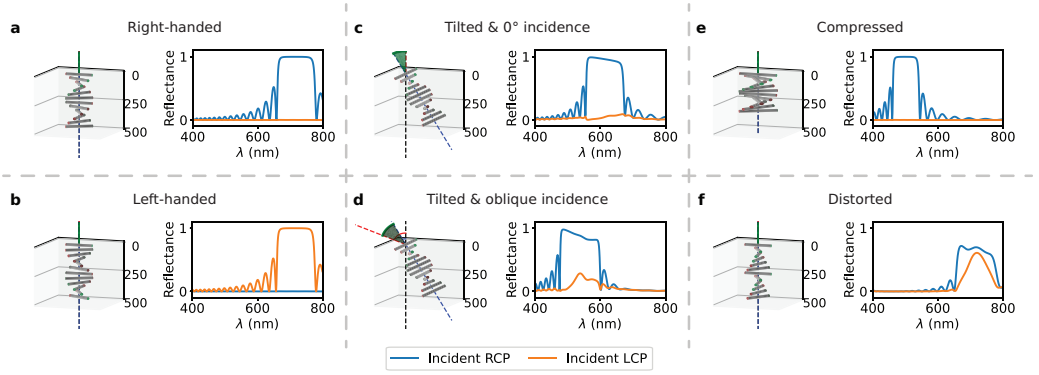

Figure 5: Collection of cholesteric-specific situations implemented in PyLlama (right-handed cholesteric, left-handed cholesteric, tilt, oblique incidence, vertical compression, distortion combining a tilt and a vertical compression), and their reflection spectra spectra calculated numerically. The user can easily mix-and-match these situations. Both the 3D representations of the cholesterics and the optical spectra are automatically computed through PyLlama.

even be higher than 50% in some cases). Each time, a `Cholesteric` object describes the physical architecture and the required parameters are extracted to build up an optical model upon it and enables to calculate reflection spectra in the circular-polarisation basis.

We use this example to illustrate results obtained with PyLlama and to demonstrate how flexible it is to handle custom structures. The user may add supplementary features to our `Cholesteric` library, or pair an external library to PyLlama in a similar way to model custom stacks.

3.6. Mixing multiple stacks

Throughout our code, `Layers` are treated as independent units and can be added to a `Structure` provided that the wavevector x-component stays the same. Similarly, `Structures` can be stacked together to model the behaviour of complex samples such as understanding the self-assembly of dry cellulose nanocrystal films that creates distorted domains [62], explaining the polarization independence of hydroxypropyl cellulose films distorted by cross-linking [60] or tuning the distortion to obtain samples with a particular polarization selectivity [55] or gaining insight on mechanical deformations imposed on cellulose nanocrystal elastomers [16]. Combining models into a master model allows to take advantage of sub-periodicities within each model and to keep the computation time fast.

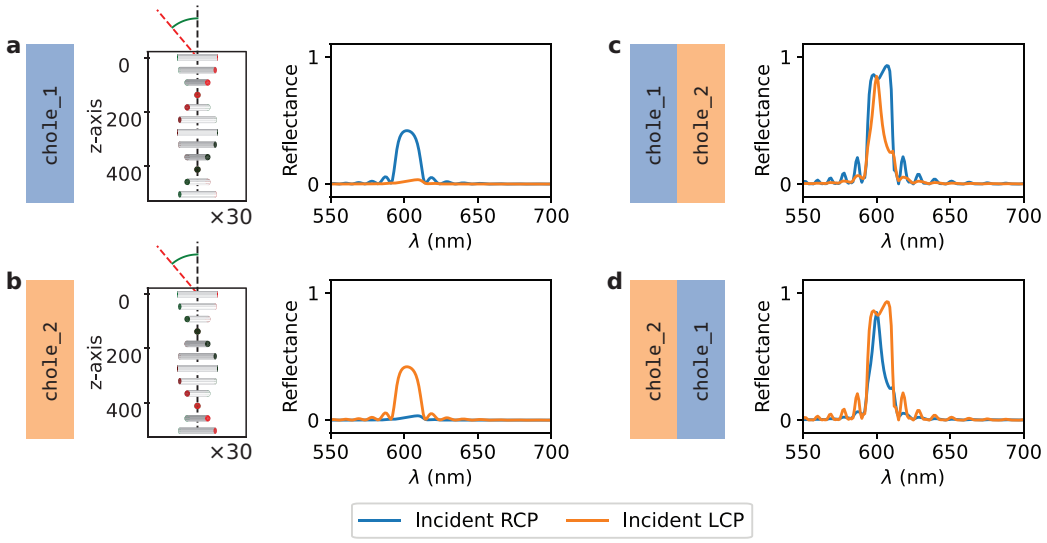

Figure 6: a, b) Schematic representation and reflection spectra under 45° incidence of a) a right-handed and b) a left-handed cholesteric with the same pitch (500 nm) and with the same number of pitches (30). c, d) Reflection spectra for the stacks of c) the right-handed cholesteric on top of the left-handed cholesteric and d) the left-handed cholesteric on top of the right-handed cholesteric.

The class `MixedModel`, child of `Model`, takes as additional parameter a list of models `models_list` to be combined (in the order of the list). They are screened in order to add only the models that are compatible with the first one from the list (in terms of the conservation of the wavevectors tangential component). The function `MixedModel.get_refl()` overrides its parent function `Model.get_refl()` and calls for `get_refl_TM_multiple_structures` (respectively, `SM`), which builds up transfer or scattering matrices for each sub-model and combines them with a cautious handling the interfaces between consecutive models for the scattering matrix method.

An example is given on Fig. 6 where two periodic `CholestericModels` of opposite handedness and same pitch are stacked upon each other in a different order. The resulting spectrum for the stacked cholesterics under oblique incidence show a difference depending on which cholesteric is on top of the other.

3.7. A level of automation

The classes `Structure`, `Layer`, `HalfSpace` and `Wave` contain every function that is necessary to calculate the reflectance of a mutlilayer stack at

a specific wavelength, and the class `Model` and its children enable to build `Structures` easily from appropriate parameters.

In order to enable the use of our code without requiring extensive programming skills and/or manipulation of the theoretical concepts to calculate reflection spectra for a given multilayer stack, we embedded our models inside the class `Spectrum` that provides a level of automation to the user to calculate the reflectance and transmittance for a range of wavelengths (to get a full spectrum) and export the data for further analysis in Python or in MATLAB, while the optics calculations are occurring in the background. The user who decides to write custom models can integrate them to the class `Spectrum` too.

`Spectrum` contains a list of wavelengths inputted by the user, a dictionary whose keys correspond to the model parameters and whose values correspond to the values assigned to these parameters inputted by the user, and an empty dictionary which will contain the calculated reflection spectra. The function `Spectrum.calculate_refl_trans()` enables to calculate reflection spectra in the linear polarisation basis or in the circular polarisation basis with a choice of the matrix method with the parameter, and with a display of the calculation progress with the parameter `talk=<False|True>`. Results for (potentially both) polarisation bases are then added to `Spectrum.data`. The function `Spectrum.export()` enables to export the content of `Spectrum.data` for further processing in Python or in MATLAB. Its argument is a filename `path_out` which extension should be `.pck` for an export with Pickles and further use in Python or `.mat` for an export in MATLAB-compatible format.

When pairing `Model` and its children with the class `Spectrum`, spectra from complex structures are straightforward to obtain even without much experience in programming. The custom `Models` created by the user can also be with the class `Spectrum`. The user manual contains tutorials explaining how to incorporate custom children classes appropriately.

4. Comparison between the transfer matrix and scattering matrix methods

The calculations behind the transfer matrix and the scattering matrix methods have been detailed in Section 2.4 and 2.5, and show that the combination of subsequent `Layers` with the transfer matrix method simply consist in matrix multiplications, Eq. (9) while their combination with the scattering

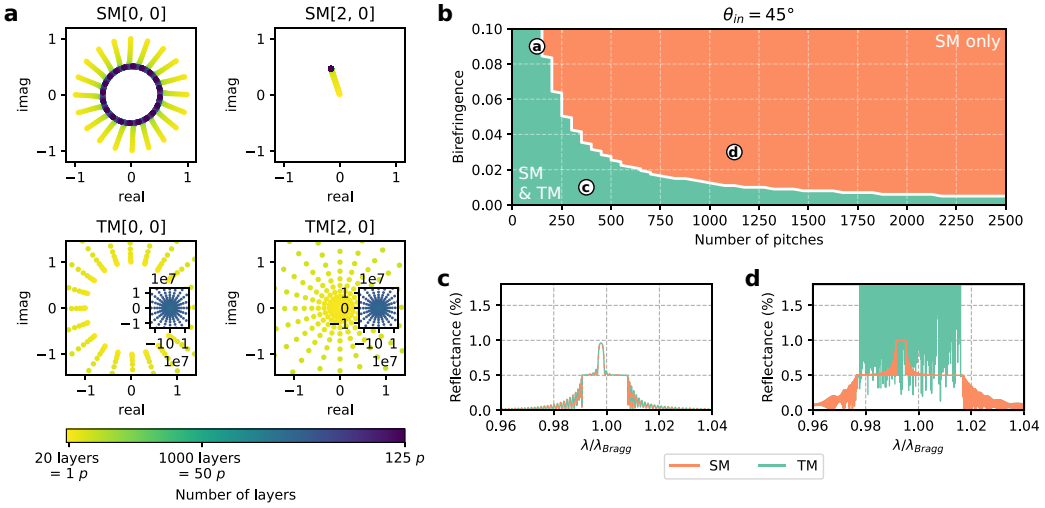

Figure 7: Comparison between the scattering matrix (SM) and transfer matrix (TM) methods a) Evolution of two matrix elements of indices $(0, 0)$ and $(2, 0)$ in the complex plane. The size of the system increases from one layer to a full pitch made of 20 layers to 125 pitches made of 125×20 layers. b) The colormap displays the combinations that cannot be calculated with the TM method. The configurations a), b) and c) are pinpointed on the colormap. c, d) The TM and SM methods give identical results at low birefringence and for thin systems (c) while the TM method becomes numerically unstable at high birefringence and for thick systems (d). The pitch and the Bragg wavelength λ_{Bragg} are given in Appendix B: the pitch p is the periodicity and the Bragg wavelength equals $np\cos(\theta_{in})$, where n is the average refractive index of the cholesteric.

matrix method requires to unpack matrix elements, to combine them externally and to reassemble them into a new scattering matrix, Eq. (25). The scattering matrix method is consequently slower than the transfer matrix method, but it is also more robust and stable.

Figure 7a shows the evolution of selected (representative) matrix elements from the transfer and scattering matrices when the number of layers in a cholesteric structure increases (Appendix Appendix B provides details about the structure). They are plotted in the complex plane. As one can see, the values obtained with the transfer matrix method are continuously growing, while the values obtained with the scattering matrix method are converging. In fact, at each calculation step, the values of the scattering matrix represent transmission and reflection coefficients, Eq. (28), and their modulus is therefore always under 1, while with the transfer matrix method, reflection coefficients are calculated as ratios between given matrix elements,

Eq. (15).

Usually, reflection spectra calculated from both methods lead to the same result (Figure 7c: a cholesteric structure with a birefringence $n_e - n_o = 0.015$ and 375 pitches). However, when the number of layers becomes too large or the birefringence becomes too high, the transfer matrix method fails dramatically (Figure 7d: a cholesteric structure with $n_e - n_o = 0.035$ and 1125 pitches) while the scattering matrix remains stable. Figure 7b displays which combinations of birefringences and number of pitches are not compatible with the transfer matrix method.

The transfer matrix method remains useful as it is faster than the scattering matrix method: computing a spectrum on the visible range (400 to 800 nm with a resolution of 1 nm) corresponding to the cholesteric structure on Figure 7c takes on average 6.5s with the transfer matrix method against 14.0s with the scattering matrix method (averaged over 10 runs with HP EliteBoox x360 1030 G2, with a standard deviation under 0.1).

5. Concluding remarks

We have presented a stable and versatile Python toolkit for the electromagnetic modelling of anisotropic multilayer stacks. PyLlama relies on known concepts and numerical techniques, but it answers a need from the scientific community for a simple, freely-available and open-source program for the purpose. It is accessible to use without extended programming experience even for complex multilayer stacks (distorted cholesterics, stacks containing periodic sub-structures...), and it is also easy to customise without having to re-write any optical calculation.

The strength of PyLlama is undoubtedly its capability to deal with arbitrary multilayers even in extreme situations (large number of layers, high birefringence, grazing angles of incidence) thanks to the implementation of the scattering matrix method, and to switch seamlessly to the transfer matrix method for faster computation speed, when suitable. PyLlama thus allows performing a side-by-side comparison between the transfer matrix and the scattering matrix methods.

Many additional functionalities could be added in the near future without considerable efforts, like the possibility to incorporate magneto-optic effects [37], and to compute spatial field distribution in the stack [41] and Bloch modes for periodic media [63]. Classes dedicated to the acquisition of datasets other than reflection and transmission spectra could be created

quite easily, such as for ellipsometry measurements [43] or critical angle measurements [64], which are useful tools to characterise samples. The necessary calculations are already carried out in the core of PyLlama, which paves the way to the implementation of a dedicated framework similar to our class `Spectrum`.

The electromagnetic modelling of space-time-modulated media – a rapidly growing topic in wave physics [65] – would be an exciting future development of PyLlama. The transfer matrix method has recently been generalized to consider time-varying systems [66] by writing the fields as a sum of harmonics of the modulation frequency and setting up time-varying boundary conditions. This results in matrices of larger sizes depending on the number of harmonics considered. The implementation of this idea for anisotropic materials may be of significant help to explore the multitude of exotic phenomena in such systems.

It is our hope that the scientific community will adopt PyLlama and contribute to its evolution.

6. Acknowledgements

This work was supported by ERC grant ERC-2014-STG H2020 639088 and Philip Leverhulme Prize (PLP-2019-271) for S.V. and M. M. B. The authors are indebted to Jean-Paul Hugonin (Laboratoire Charles Fabry, Palaiseau, France) for his critical reading of the manuscript, for verifying our numerical simulations with the RCWA software RETICOLO [52], and more generaly for very insightful discussions. The authors thank Bruno Frka-Petesic (Department of Chemistry, University of Cambridge, UK) for useful discussions about the optical properties of cholesteric structures. M.M.B. thanks Sing-Teng Chua (Department of Chemistry, University of Cambridge, UK) for proof-testing the user manual, and Mathieu Brèthes for his help regarding the documentation. K.V. thanks Akhlesh Lakhtakia (Penn State University, USA) for pointing out the seminal contribution of Jean Billard to the 4×4 matrix formalism [28].

Appendix A. Analysis of the partial waves

The four partial waves in a layer can generally be separated into a forward ($+z$) propagating pair and a backward ($-z$) propagating pair, where each

pair consists in a partial wave rather polarised along the x-axis and a partial wave rather polarised along the y-axis [31].

The main scope of the partial waves analysis is to identify the exponentially decaying and growing waves and sort them accordingly, which is important for the scattering matrix method. This can be done easily from the imaginary component of the eigenvalues (or wavevector) $q_j \equiv K_{j,z}$.

Here, we chose to rely instead on the Poynting vector along the propagation direction z, $\mathcal{S}_{j,z}$ (we discard here the layer label i), defined as

$$\boldsymbol{\mathcal{S}_j} = \begin{bmatrix} \mathcal{S}_{j,x} \\ \mathcal{S}_{j,y} \\ \mathcal{S}_{j,z} \end{bmatrix} = \begin{bmatrix} E_{j,y}H_{j,z} - E_{j,z}H_{j,y} \\ E_{j,z}H_{j,x} - E_{j,x}H_{j,z} \\ E_{j,x}H_{j,y} - E_{j,y}H_{j,x} \end{bmatrix} \tag{A.1}$$

with $E_{j,z}$ and $H_{j,z}$ expressed in function of $E_{j,x}$, $E_{j,y}$, $H_{j,x}$ and $H_{j,y}$,

$$\begin{aligned} E_{j,z} &= -\frac{\epsilon_{zx}}{\epsilon_{zz}}E_{j,x} - \frac{\epsilon_{zy}}{\epsilon_{zz}}E_{j,y} - \frac{K_{j,x}}{\epsilon_{zz}}H_{j,y} \\ H_{j,z} &= K_{j,x}E_{j,y} \end{aligned} \tag{A.2}$$

The direction of the partial wave is then given by the sign of the real part of $\mathcal{S}_{j,z}$, when real, and by the sign of the imaginary part of $\mathcal{S}_{j,z}$, when complex. For the sorting of the exponentially decaying and growing waves, we have observed that this procedure was fully equivalent to that with $K_{j,z}$ [5].

An inspection of the two waves within each pair can reveal the partial wave polarisation. If the material is anisotropic (without the crystal axes along the (x, y) axes), each partial wave $\mathbf{p}_j$ is analysed by calculating the ratio $C(\mathbf{p}_j)$:

$$C(\mathbf{p}_j) = \frac{|\mathcal{S}_{j,x}|^2}{|\mathcal{S}_{j,x}|^2 + |\mathcal{S}_{j,y}|^2} \tag{A.3}$$

and comparing it to its pair. If $C(\mathbf{p}_1) > C(\mathbf{p}_2)$, $\mathbf{p}_1$ describes a wave rather polarised along the x axis and $\mathbf{p}_2$ describes a wave rather polarised along the y axis. If the material is isotropic or anisotropic with the crystal axes aligned with the laboratory axes, this corresponds to p- and s-polarised waves, respectively. According to Ref. [41], in these cases, the two partial waves within

[5] The sorting with $K_{j,z}$ is implemented and commented in the PyLlama code for the interested user.

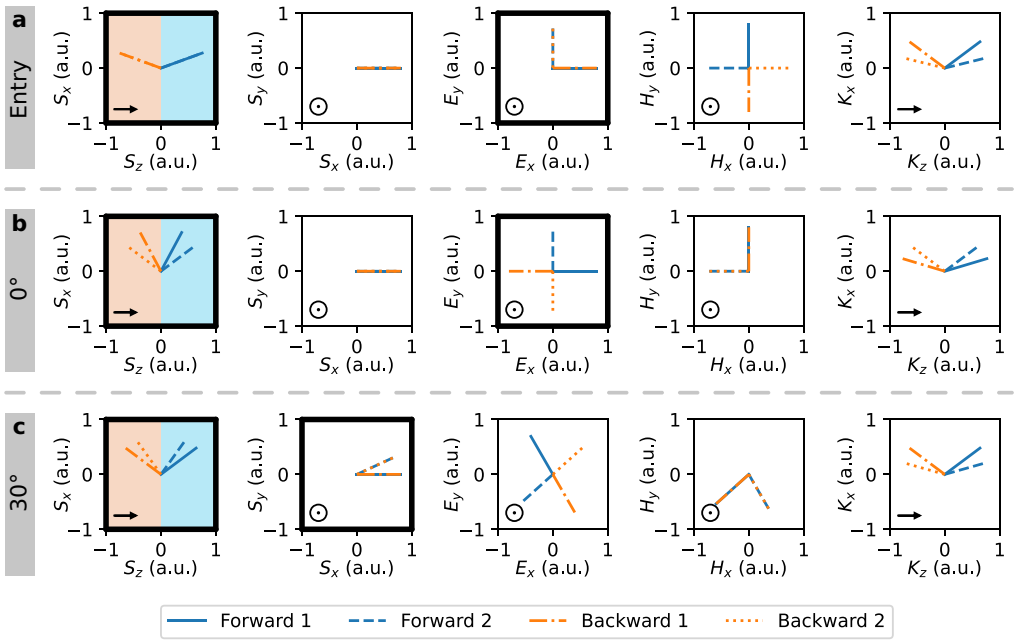

Figure A.8: Schematic of the analysis of the partial waves (eigenvalues $K_{j,z}$ and eigenvectors $[E_{j,x}, H_{j,y}, E_{j,y}, -H_{j,x}]$). a. An isotropic entry half-space. b. A uniaxial crystal (anisotropic) whose principal axes are oriented along the laboratory axes x and y. c. A uniaxial crystal whose principal axes are oriented at an arbitrary angle from the laboratory axes; the rotation angle of the crystal is 30° here.

each pair can be analysed with their electric field $\mathbf{E}_j$ instead of their Poynting vector $\boldsymbol{\mathcal{S}}_j$:

$$C_{\text{iso}}(\mathbf{p}_j) = \frac{|E_{j,x}|^2}{|E_{j,x}|^2 + |E_{j,y}|^2} \tag{A.4}$$

Figure A.8 illustrates the analysis and sorting of the partial waves in three particular cases (all encountered when dealing with a cholesteric structure): an isotropic layer, a uniaxial crystal (anisotropic) whose principal axes are oriented along the laboratory axes x and y, and a uniaxial crystal whose axes are not aligned with the x and y axes. First, the analysis of the direction of the Poynting vector $S_{j,z}$ (left column) enables to sort between forward (plotted in blue) and backward (plotted in orange) partial waves. Note that this is equivalent to sorting according to the direction of the wavevector $K_{j,z}$ (fifth column). Second, the direction of the electric field (third column) or of the Poynting vector (second column) along the x or y axes enables to

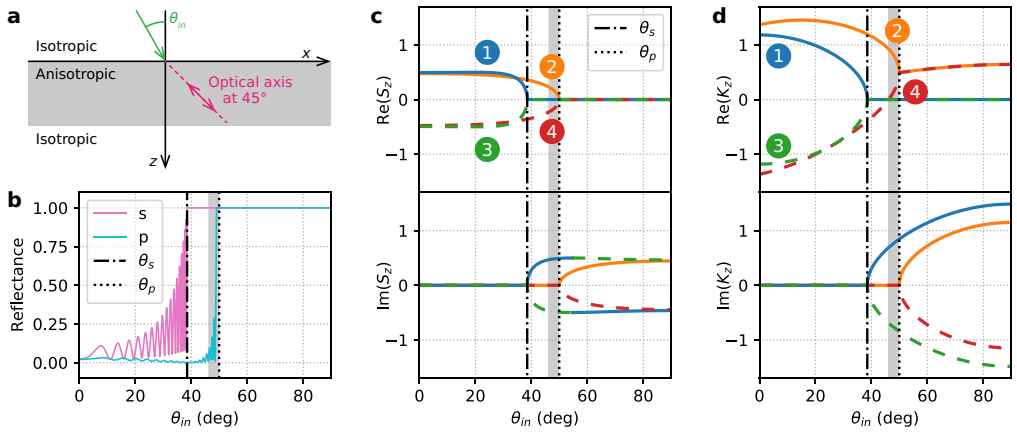

Figure A.9: Illustration of the necessity to sort forward- and backward-propagating partial waves according to the direction of the Poynting vector. a) Schematic of the situation. b) Reflectance plots for p and s polarisations and associated critical angles of total reflection according to Ref. [64] (vertical dashed lines). c, d) Below the critical angle for p-polarised waves (gray shaded area), the Poynting vector (describing the direction of the energy flux) and the wavevector (describing the direction of the wave front) have opposite signs.

identify the partial waves individually (plotted with different dashes). In each plot, the positive direction of the z axis is displayed in the bottom-left corner for clarity. Each segment has been normalised to the same length. The elements that are effectively used to sort the partial waves have a thick black border and the others are displayed to provide a more complete picture of the situation.

The wavevector $K_{j,z}$ and the Poynting vector $\mathcal{S}_{j,z}$ offer different information on the partial waves, indicating respectively the directions of the wave and of the energy flux. This is illustrated in Fig. A.9, where we consider a slab of uniaxial crystal whose director lays in the (xz) plane (see Figure A.9a). In such situations, p- and s-polarised incident waves can have different angles of total reflection (see Figure A.9b). Rivière provides analytical formulas for these critical angles [64], reported here as vertical dashed lines. In particular, when the optical axis of the crystal lays at 45° in the (xz)-plane, the critical angle of total reflection for p-polarisation is 49.9°. Interestingly, at slightly smaller angles (gray shaded area on Fig. A.9c-d), the partial wave $j = 4$ exhibits real wavevector and Poynting vector of opposite signs, $\mathrm{Re}[K_z] > 0$ and $\mathrm{Re}[\mathcal{S}_z] < 0$. This has no effect on the predictions of the matrix methods

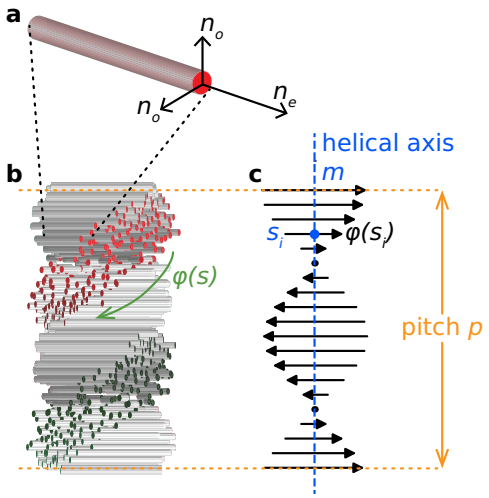

Figure B.10: Schematic representation of a cholesteric structure in the (xz) plane. The cholesteric structure repeats itself every half-pitch because the directors oriented towards 0° and 180° are equivalent; the red and green side of the elongated units and the tips of the arrows are a guide for the eye. a) Refractive indices of a birefringent unit. b) The materials periodic twisted structure. c) A physical model consists in planes whose directors rotate around the helical axis m.

because the imaginary part of the corresponding eigenvalue is zero.

Appendix B. Discrete model for cholesterics

A cholesteric structure is an assembly of elongated birefringent units organised in a stack whose planes twist continuously around a helical axis at a periodicity called the pitch. At a given depth s along the helical axis m, the elongated units point on average towards a preferred direction $n(s)$, which corresponds to a rotation angle $\phi(s)$ around the helical axis [6, 10], even if the direction of individual units may differ. Schematics of the birefringent units is displayed on Figure B.10a (single unit) and b (assembly of units) while Figure B.10c shows the directions of the rotating planes.

Optically, a cholesteric structure corresponds to a permittivity tensor $\epsilon(\phi(s))$ rotating at an angle $\phi(s)$ around the helical axis [5, 7], where $\epsilon(0)$ corresponds to a birefringent unit aligned with the plane of incidence (xz): $\epsilon_0 = \mathrm{diag}(n_e^2, n_o^2, n_o^2)$ where n_e and n_o are the refractive indices of a birefringent unit. The optical response of the structure therefore directly relates to the periodic modulation of directors around the helical axis.

Under normal incidence of light, cholesterics reflect perfectly circularly polarised light of the handedness that matches the rotation of the helicoid (contrary to a mirror that swaps the polarisation) and in a narrow wavelength region. The central wavelength λ_B of the reflection peak obeys Braggs law and is proportional to the cholesterics pitch p and its average refractive index n_{av} such that $\lambda_B = n_{av}p$ [6]. Under oblique incidence of light θ_{in}, the reflection peak shifts such that $\lambda_{Bragg} = n_{av}p\cos(\theta_{in})$ and the structure is less polarisation-selective than at normal incidence [6, 7].

Cholesteric structures that are distorted also reflect both polarisations [55, 10]. Recently, Frka-Petesic *et al.* developed a physical model to physically represent cholesteric structures, as well as the perturbation it encounters upon external constraints that affect the periodic directors and modify the optical response [10].

The optics of cholesterics is not trivial and no analytical formula can be found in the literature to predict the reflectance of a distorted cholesteric, for instance.

For modelling purposes, these continuous non-homogeneous materials are assimilated to discrete stacks of slices where each slice i (at depth s_i on the helical axis) possesses a director n_i and rotation angle ϕ_i, leading to a permittivity tensor $\epsilon(\phi_i)$, see Fig. B.10c.

Appendix C. Parameters for the figures

Appendix C.1. Figure 4 (validation against analytical methods)

For Fig. 4b, the parameters used to construct the Bragg stack (with `StackModel`) are

Parameter	**Variable**	**Value**
Permittivity of the 1st layer	`eps_list[0]`	$\begin{bmatrix} 2.2^2 & 0 & 0 \\ 0 & 2.2^2 & 0 \\ 0 & 0 & 2.2^2 \end{bmatrix}$
Permittivity of the 2nd layer	`eps_list[1]`	$\begin{bmatrix} 1.0 & 0 & 0 \\ 0 & 1.0 & 0 \\ 0 & 0 & 1.0 \end{bmatrix}$
Thickness of the 1st layer (nm)	`thickness_nm_list[0]`	200
Thickness of the 2nd layer (nm)	`thickness_nm_list[1]`	500
Number of periods	`N_per`	10
Index of entry medium	`n_entry`	1.0
Index of exit medium	`n_exit`	2.2
Angle of incidence (rad)	`theta_in_rad`	$\pi/3$
Wavelength (nm)	`wl_nm`	400 to 800

For Fig. 4d, the refractive indices and thicknesses used to construct the Bragg stack (with `StackModel`) are calculated from Eq. (29) with the number of periods indicated on Fig. 4c and $n_0 = 1$ and $n_s = 4$.

The partial waves shown on Fig. 4e and f are extracted from a cholesteric structure. The parameters used to construct the cholesteric object (with `Cholesteric`) are

Parameter	**Variable**	**Value**
Tilt (rad)	`tilt_rad`	0
Pitch (nm)	`pitch360`	500
Handedness	`handedness`	`1` (right)
Slices per pitch	`resolution`	360

The parameters used to construct the cholesteric model (with `CholestericModel`) are

Parameter	Variable	Value
Number of periods	`N_per`	1
Extraordinary index (along x)	`n_e`	1.4505
Ordinary index (along y and z)	`n_o`	1.4155
Index of entry medium	`n_entry`	1.433
Index of exit medium	`n_exit`	1.433
Angle of incidence (rad)	`theta_in_rad`	0 (e) and $\pi/3$ (f)
Wavelength (nm)	`wl_nm`	λ_{Bragg} = e) 716.5 f) 358.25

The partial waves are extracted from the 15th anisotropic slice and b) the 30th anisotropic slice that have a rotation angle of 15° and 30° around the z axis, respectively.

Appendix C.2. Figure 5 (various examples of cholesterics)

For Fig. 5a (right-handed), the parameters used to construct the cholesteric object (with `Cholesteric`) are

Parameter	Variable	Value
Tilt (rad)	`tilt_rad`	0
Pitch (nm)	`pitch360`	500
Handedness	`handedness`	right (`1`)
Slices per pitch	`resolution`	40
Compression ratio	`alpha`	1
Arbitrary distortion	`d`	none (`1`)

The parameters used to construct the cholesteric model (with `CholestericModel`) are

Parameter	Variable	Value
Number of periods	`N_per`	10
Extraordinary index (along x)	`n_e`	1.533
Ordinary index (along y and z)	`n_o`	1.333
Index of entry medium	`n_entry`	1.433
Index of exit medium	`n_exit`	1.433
Angle of incidence (rad)	`theta_in_rad`	0
Wavelength (nm)	`wl_nm`	400 to 800

For Fig. 5b (left-handed), the parameters used to construct the cholesteric object (with `Cholesteric`) are the same as Fig. 5a except

Parameter	Variable	Value
Handedness	`handedness`	left (`-1`)

The parameters used to construct the cholesteric model (with `CholestericModel`) are the same as Figure 5a.

For Fig. 5c (tilted and normal incidence), the parameters used to construct the cholesteric object (with `Cholesteric`) are the same as Fig. 5a except:

Parameter	Variable	Value
Tilt (rad)	`tilt_rad`	$\pi/6$

The parameters used to construct the cholesteric model (with `CholestericModel`) are the same as Figure 5a.

For Fig. 5d (tilted and oblique incidence), the parameters used to construct the cholesteric object (with `Cholesteric`) are the same as Fig. 5a except

Parameter	Variable	Value
Tilt (rad)	`tilt_rad`	$\pi/6$

The parameters used to construct the cholesteric model (with `CholestericModel`) are the same as Fig. 5a except:

Parameter	Variable	Value
Angle of incidence (rad)	`theta_in_rad`	$70\pi/180$

For Fig. 5e (compressed), the parameters used to construct the cholesteric object (with `Cholesteric`) are the same as Fig. 5a except:

Parameter	Variable	Value
Compression ratio	`alpha`	0.7

The parameters used to construct the cholesteric model (with `CholestericModel`) are the same as Fig. 5a.

For Fig. 5f (distorted), the parameters used to construct the cholesteric object (with `Cholesteric`) are the same as Fig. 5a except:

Parameter	Variable	Value
Arbitrary distortion	`d`	3

The parameters used to construct the cholesteric model (with `CholestericModel`) are the same as Figure 5a.

Appendix C.3. Figure 6 (`MixedModel` example)

The parameters used to construct the cholesteric objects `chole_1` and `chole_2` (with `Cholesteric`) are

Parameter	**Variable**	**Value**
Tilt (rad)	`tilt_rad`	0
Pitch (nm)	`pitch360`	550
Handedness	`handedness`	right (`1`) and left (`-1`)
Slices per pitch	`resolution`	40

The parameters used to construct the corresponding cholesteric models (with `CholestericModel`) are

Parameter	**Variable**	**Value**
Number of periods	`N_per`	30
Extraordinary index (along x)	`n_e`	1.443
Ordinary index (along y and z)	`n_o`	1.423
Index of entry medium	`n_entry`	1.433
Index of exit medium	`n_exit`	1.433
Angle of incidence (rad)	`theta_in_rad`	$40\pi/180$
Wavelength (nm)	`wl_nm`	400 to 800

Appendix C.4. Figure 7 (performances of the SM and TM methods)

The matrix elements shown on Fig. 7a are calculated from a cholesteric structure. The spectra shown on Fig. 7c and d are calculated from a cholesteric structure with similar parameters. The colormap shown on Fig. 7b is made by further varying the birefringence and the number of periods of this cholesteric.

The parameters used to construct the cholesteric object (with `Cholesteric`) are:

Parameter	**Variable**	**Value**
Tilt (rad)	`tilt_rad`	0
Pitch (nm)	`pitch360`	500
Handedness	`handedness`	right (`1`)
Slices per pitch	`resolution`	a) 20, b), c), d) 40

The parameters used to construct the cholesteric model (with `pyllama.CholestericModel`) are:

Parameter	Variable	Value
Number of periods	`N_per`	See figure
Extraordinary index (along x)	`n_e`	See figure
Ordinary index (along y and z)	`n_o`	See figure
Average index		1.433
Birefringence		Variable
Index of entry medium	`n_entry`	1.433
Index of exit medium	`n_exit`	1.433
Angle of incidence (rad)	`theta_in_rad`	a) 0, b, c, d) $\pi/4$
Wavelength (nm)	`wl_nm`	a) $\lambda_{Bragg} = 716.5$

Appendix C.5. Figure A.8 (sorting of the partial waves, examples)

The partial waves are extracted from a cholesteric structure. The parameters used to construct the cholesteric object (with `Cholesteric`) are:

Parameter	Variable	Value
Tilt (rad)	`tilt_rad`	0
Pitch (nm)	`pitch360`	1000
Handedness	`handedness`	right (`1`)
Slices per pitch	`resolution`	360

The parameters used to construct the cholesteric model (with `CholesericModel`) are:

Parameter	Variable	Value
Number of periods	`N_per`	1
Extraordinary index (along x)	`n_e`	3
Ordinary index (along y and z)	`n_o`	1.2
Index of entry medium	`n_entry`	2.1
Index of exit medium	`n_exit`	2.1
Angle of incidence (rad)	`theta_in_rad`	$\pi/9$
Wavelength (nm)	`wl_nm`	$\lambda_{Bragg} \approx 1346$

The partial waves are extracted from a) the entry isotropic half-space, b) the first anisotropic slice and c) the 30th anisotropic slice that has a rotation angle of 30° around the z axis.

Appendix C.6. Figure A.9 (sorting of the partial waves with the wavevector or the Poynting vector)

The parameters used to construct the slab (with `SlabModel`) are:

Parameter	**Variable**	**Value**
Permittivity	`eps`	$\begin{bmatrix} 1.682^2 & 0 & 0 \\ 0 & 1.183^2 & 0 \\ 0 & 0 & 1.183^2 \end{bmatrix}$
Thickness (nm)	`thickness_nm`	4000
Rotation angle (rad)	`rotangle_rad`	$\pi/8$
Rotation axis	`rotaxis`	y
Index of entry medium	`n_entry`	1.9
Index of exit medium	`n_exit`	1.433
Angle of incidence (rad)	`theta_in_rad`	0 to $\pi/4$
Wavelength (nm)	`wl_nm`	500

Figure A.9b represents the partial waves extracted for an incident angle of 49.9°.

References

[1] S. Kinoshita, Structural colors in the realm of nature - ISBN:9789812707833, World Scientific Publishing Company, 2008. doi:10.1016/B978-008043924-2/50055-9.
URL `http://books.google.com/books?hl=en&lr=&id=aqj4Y4N9BZoC&oi=fnd&pg=PA1&dq=S`

[2] T. D. Steiner, Semiconductor nanostructures for optoelectronic applications, Artech House, 2004.

[3] M. W. McCall, I. J. Hodgkinson, Q. Wu, Birefringent Thin Films and Polarizing Elements, World Scientific Publishing Company, 2014.
URL `https://www.worldscientific.com/worldscibooks/10.1142/p962`

[4] T. G. Mackay, A. Lakhtakia, Electromagnetic anisotropy and bianisotropy: a field guide, World Scientific, 2019.

[5] C. W. Oseen, The theory of liquid crystals, Transactions of the Faraday Society 29 (140) (1933) 883. doi:10.1039/tf9332900883.
URL `http://xlink.rsc.org/?DOI=tf9332900883`

[6] H. de Vries, Rotatory power and other optical properties of certain liquid crystals, Acta Crystallographica 4 (3) (1951) 219–226. doi:10.1107/S0365110X51000751.
URL http://scripts.iucr.org/cgi-bin/paper?S0365110X51000751

[7] V. Belyakov, V. Dmitrienko, V. Orlov, Optics of cholesteric liquid crystals, Soviet Physics Uspekhi 261 (February 1979) (1979) 63–88. doi:10.1070/PU1979v022n02ABEH005417.
URL http://iopscience.iop.org/0038-5670/22/2/R01

[8] R. S. Werbowyj, D. G. Gray, Optical Properties of (Hydroxypropyl)cellulose Liquid Crystals. Cholesteric Pitch and Polymer Concentration, Macromolecules 17 (8) (1984) 1512–1520. doi:10.1021/ma00138a016.
URL http://pubs.acs.org/doi/abs/10.1021/ma00138a016

[9] P. Sixou, J. M. Gilli, A. Ten Bosch, F. Fried, P. Maïssa, L. Varichon, M. H. Godinho, Cholesteric mesophases, Physica Scripta 1991 (T35) (1991) 47–52. doi:10.1088/0031-8949/1991/T35/010.

[10] B. Frka-Petesic, G. Kamita, G. Guidetti, S. Vignolini, The angular optical response of cellulose nanocrystal films explained by the distortion of the arrested suspension upon drying, Physical Review Materials 3 (4) (2019) 45601. doi:10.1103/PhysRevMaterials.3.045601.
URL https://doi.org/10.1103/PhysRevMaterials.3.045601

[11] S. Vignolini, P. J. Rudall, A. V. Rowland, A. Reed, E. Moyroud, R. B. Faden, J. J. Baumberg, B. J. Glover, U. Steiner, Pointillist structural color in Pollia fruit, Proceedings of the National Academy of Sciences 109 (39) (2012) 15712–15715. doi:10.1073/pnas.1210105109.
URL http://www.pnas.org/cgi/doi/10.1073/pnas.1210105109

[12] I. Carter, K. Weir, M. McCall, A. Parker, Variation in the circularly polarized light reflection of lomaptera (scarabaeidae) beetles, Journal of the Royal Society Interface 13 (120) (2016) 20160015.

[13] R. M. Parker, B. Frka-Petesic, G. Guidetti, G. Kamita, G. Consani, C. Abell, S. Vignolini, Hierarchical Self-Assembly of Cellulose Nanocrystals in a Confined Geometry, ACS Nano 10 (9) (2016) 8443–8449. doi:10.1021/acsnano.6b03355.

[14] D. G. Gray, Chiral nematic ordering of polysaccharides, Carbohydrate Polymers 25 (4) (1994) 277–284. doi:10.1016/0144-8617(94)90053-1.

[15] D. V. Saraiva, R. Chagas, B. M. de Abreu, C. N. Gouveia, P. E. Silva, M. H. Godinho, S. N. Fernandes, Flexible and structural coloured composite films from cellulose nanocrystals/hydroxypropyl cellulose lyotropic suspensions, Crystals 10 (2) (2020). doi:10.3390/cryst10020122.

[16] C. E. Boott, A. Tran, W. Y. Hamad, M. J. MacLachlan, Cellulose Nanocrystal Elastomers with Reversible Visible Color, Angewandte Chemie - International Edition 59 (1) (2020) 226–231. doi:10.1002/anie.201911468.

[17] R. Bardet, F. Roussel, S. Coindeau, N. Belgacem, J. Bras, Engineered pigments based on iridescent cellulose nanocrystal films, Carbohydrate Polymers 122 (2015) 367–375. doi:10.1016/j.carbpol.2014.10.020.
URL http://dx.doi.org/10.1016/j.carbpol.2014.10.020

[18] H. Yi, S. H. Lee, H. Ko, D. Lee, W. G. Bae, T. il Kim, D. S. Hwang, H. E. Jeong, Ultra-Adaptable and Wearable Photonic Skin Based on a Shape-Memory, Responsive Cellulose Derivative, Advanced Functional Materials 1902720 (2019) 1–12. doi:10.1002/adfm.201902720.

[19] T. H. Zhao, R. M. Parker, C. A. Williams, K. T. Lim, B. Frka-Petesic, S. Vignolini, Printing of Responsive Photonic Cellulose Nanocrystal Microfilm Arrays, Advanced Functional Materials 29 (21) (2019). doi:10.1002/adfm.201804531.

[20] P. Yeh, C. Gu, Optics of Liquid Crystal Displays, John Wiley & Sons, Inc., 2009.
URL https://www.wiley.com/en-gb/Optics+of+Liquid+Crystal+Displays,+2nd+Editio
p-9780470181768

[21] S. Dai, Q. Ma, T. Andersen, A. S. McLeod, Z. Fei, M. K. Liu, M. Wagner, K. Watanabe, T. Taniguchi, M. Thiemens, F. Keilmann, P. Jarillo-Herrero, M. M. Fogler, D. N. Basov, Subdiffractional focusing and guiding of polaritonic rays in a natural hyperbolic material, Nature Communications 6 (2015) 1–7. doi:10.1038/ncomms7963.

[22] P. Li, M. Lewin, A. V. Kretinin, J. D. Caldwell, K. S. Novoselov, T. Taniguchi, K. Watanabe, F. Gaussmann, T. Taubner, Hyperbolic phonon-polaritons in boron nitride for near-field optical imaging and focusing, Nature Communications 6 (May) (2015) 1–9. doi:10.1038/ncomms8507.

[23] S. Dai, Z. Fei, Q. Ma, A. Rodin, M. Wagner, A. McLeod, M. Liu, W. Gannett, W. Regan, K. Watanabe, et al., Tunable phonon polaritons in atomically thin van der waals crystals of boron nitride, Science 343 (6175) (2014) 1125–1129.

[24] D. P. Pulsifer, M. Faryad, A. Lakhtakia, Observation of the Dyakonov-Tamm wave, Physical Review Letters 111 (24) (2013) 1–5. arXiv:1311.4414, doi:10.1103/PhysRevLett.111.243902.

[25] T. G. Mackay, C. Zhou, A. Lakhtakia, Dyakonov–voigt surface waves, Proceedings of the Royal Society A 475 (2228) (2019) 20190317.

[26] S. Chandrasekhar, K. N. Rao, Optical rotatory power of liquid crystals, Acta Crystallographica Section A 24 (4) (1968) 445–451. doi:10.1107/S0567739468000902.

[27] J. L. Fergason, Cholesteric Structure-1 Optical Properties, Molecular Crystals 1 (2) (1966) 293–307. doi:10.1080/15421406608083274.
URL http://www.tandfonline.com/doi/abs/10.1080/15421406608083274

[28] J. Billard, Contribution à l'étude de la propagation des ondes électromagnétiques planes dans certains milieux matériels, Ph.D. thesis, Centre National de la Recherche Scientifique (1967).

[29] S. Teitler, B. W. Henvis, Refraction in Stratified, Anisotropic Media, Journal of the Optical Society of America 60 (6) (1970) 830. doi:10.1364/josa.60.000830.

[30] D. W. Berreman, Optics in Stratified and Anisotropic Media: 4x4-Matrix Formulation, Journal of the Optical Society of America 62 (4) (1972). doi:10.1364/JOSA.62.000502.

[31] P. Yeh, Electromagnetic propagation in birefringent layered media, Journal of the Optical Society of America 69 (5) (1979) 742. doi:10.1364/JOSA.69.000742.

URL https://www.osapublishing.org/abstract.cfm?URI=josa-69-5-742

[32] P. J. Lin-Chung, S. Teitler, 4 x 4 Matrix formalisms for optics in stratified anisotropic media, Journal of the Optical Society of America A 1 (7) (1984) 703. doi:10.1364/josaa.1.000703.

[33] D. Y. K. Ko, J. R. Sambles, Scattering matrix method for propagation of radiation in stratified media: attenuated total reflection studies of liquid crystals, Journal of the Optical Society of America A 5 (11) (1988) 1863. doi:10.1364/josaa.5.001863.

[34] K. Eidner, Light propagation in stratified anisotropic media: orthogonality and symmetry properties of the 4x4 matrix formalisms, Journal of the Optical Society of America A 6 (11) (1989) 1657. doi:10.1364/josaa.6.001657.

[35] C. Oldano, Electromagnetic-wave propagation in anisotropic stratified media, Physical Review A 40 (10) (1989) 6014–6020.

[36] M. Schubert, Polarization-dependent optical parameters of arbitrarily anisotropic homogeneous layered systems, Physical Review B - Condensed Matter and Materials Physics 53 (8) (1996) 4265–4274. doi:10.1103/PhysRevB.53.4265.

[37] M. Schubert, T. E. Tiwald, J. A. Woollam, Explicit solutions for the optical properties of arbitrary magneto-optic materials in generalized ellipsometry, Applied optics 38 (1) (1999) 177–187.

[38] S. Stallinga, Berreman 4x4 matrix method for reflective liquid crystal displays, Journal of Applied Physics 85 (6) (1999). doi:10.1103/PhysRevA.66.060303.

[39] S. P. Palto, An algorithm for solving the optical problem for stratified anisotropic media, Journal of Experimental and Theoretical Physics 92 (4) (2001) 552–560. doi:10.1134/1.1371338.

[40] W. Xu, L. Wood, T. Golding, Optical degeneracies in anisotropic layered media: Treatment of singularities in a 4x4 matrix formalism, Physical Review B - Condensed Matter and Materials Physics 61 (3) (2000) 1740–1743. doi:10.1103/PhysRevB.61.1740.

[41] N. C. Passler, A. Paarmann, Generalized 4x4 matrix formalism for light propagation in anisotropic stratified media: study of surface phonon polaritons in polar dielectric heterostructures, Journal of the Optical Society of America B 34 (10) (2017) 2128. arXiv:1707.00462, doi:10.1364/JOSAB.34.002128.
URL https://www.osapublishing.org/abstract.cfm?URI=josab-34-10-2128

[42] WVASE by J. A. Woollam.
URL https://www.jawoollam.com/ellipsometry-software/wvase

[43] M. Schubert, Infrared ellipsometry on semiconductor layer structures: phonons, plasmons, and polaritons, Vol. 209, Springer Science & Business Media, 2004.

[44] O. Castany, Berreman4x4: Python implementation of berreman's 4x4 matrix method (2016).
URL https://github.com/Berreman4x4/Berreman4x4

[45] N. C. Passler, A. Paarmann, Generalized 4 x 4 Matrix algorithm for light propagation in anisotropic stratified media (MATLAB files) (2017).
URL https://doi.org/10.5281/zenodo.847015

[46] T. G. Mackay, A. Lakhtakia, The Transfer-Matrix Method in Electromagnetics and Optics, Vol. 1, Morgan & Claypool Publishers, 2020. doi:10.2200/s00993ed1v01y202002ema001.

[47] N. Cotter, T. Preist, J. Sambles, Scattering-matrix approach to multilayer diffraction, JOSA A 12 (5) (1995) 1097–1103.

[48] L. Li, Formulation and comparison of two recursive matrix algorithms for modeling layered diffraction gratings, JOSA A 13 (5) (1996) 1024–1035.

[49] D. Whittaker, I. Culshaw, Scattering-matrix treatment of patterned multilayer photonic structures, Physical Review B 60 (4) (1999) 2610.

[50] S. G. Tikhodeev, A. Yablonskii, E. Muljarov, N. A. Gippius, T. Ishihara, Quasiguided modes and optical properties of photonic crystal slabs, Physical Review B 66 (4) (2002) 045102.

[51] M. Liscidini, D. Gerace, L. C. Andreani, J. Sipe, Scattering-matrix analysis of periodically patterned multilayers with asymmetric unit cells and birefringent media, Physical review B 77 (3) (2008) 035324.

[52] J.-P. Hugonin, P. Lalanne, RETICOLO software for grating analysis (v9) (2021). doi:10.5281/zenodo.4419063.

[53] V. Liu, S. Fan, S4: A free electromagnetic solver for layered periodic structures, Computer Physics Communications 183 (10) (2012) 2233–2244.

[54] T. A. Germer, pySCATMECH: A Python interface to the SCATMECH library (2021).
URL `https://pages.nist.gov/pySCATMECH/`

[55] A. J. J. Kragt, D. C. Hoekstra, S. Stallinga, D. J. Broer, A. P. H. J. Schenning, 3D Helix Engineering in Chiral Photonic Materials, Advanced Materials 1903120 (2019) 1903120. doi:10.1002/adma.201903120.
URL `http://doi.wiley.com/10.1002/adma.201903120`

[56] J. Sambles, G. Bradbery, F. Yang, Optical excitation of surface plasmons: an introduction, Contemporary physics 32 (3) (1991) 173–183.

[57] A. Mendoza-Galván, E. Munoz-Pineda, K. Järrendahl, H. Arwin, Pitch profile across the cuticle of the scarab beetle Cotinis mutabilis determined by analysis of Mueller matrix measurements, Royal Society Open Science 5 (12) (2018) 0–9. doi:10.1098/rsos.181096.

[58] S. N. Fernandes, P. L. Almeida, N. Monge, L. E. Aguirre, D. Reis, C. L. de Oliveira, A. M. Neto, P. Pieranski, M. H. Godinho, Mind the Microgap in Iridescent Cellulose Nanocrystal Films, Advanced Materials 29 (2) (2017) 1–7. doi:10.1002/adma.201603560.

[59] R. Dreher, Reflection properties of distorted cholesteric liquid crystals, Solid State Communications 12 (6) (1973) 519–522.

[60] C. L. C. Chan, M. M. Bay, G. Jacucci, R. Vadrucci, C. A. Williams, G. T. van de Kerkhof, R. M. Parker, K. Vynck, B. Frka-Petesic, S. Vignolini, Visual Appearance of Chiral Nematic Cellulose-Based Photonic Films: Angular and Polarization Independent Color Response with a Twist, Advanced Materials 31 (52) (2019). doi:10.1002/adma.201905151.

[61] P. Yeh, Optical waves in layered media, 1st Edition, Wiley-Interscience, 1988.

[62] B. Frka-Petesic, J. A. Kelly, G. Jacucci, G. Guidetti, G. Kamita, N. P. Crossette, W. Y. Hamad, M. J. MacLachlan, S. Vignolini, Retrieving the Coassembly Pathway of Composite Cellulose Nanocrystal Photonic Films from their Angular Optical Response, Advanced Materials 1906889 (2020). doi:10.1002/adma.201906889.

[63] Q. Cao, P. Lalanne, J.-P. Hugonin, Stable and efficient bloch-mode computational method for one-dimensional grating waveguides, JOSA A 19 (2) (2002) 335–338.

[64] D. Rivière, Y. Levy, C. Imbert, Determination of liquid crystal refractive indices from critical angle measurements, Optics Communications 25 (2) (1978) 206–210. doi:10.1016/0030-4018(78)90308-5.

[65] E. Galiffi, R. Tirole, S. Yin, H. Li, S. Vezzoli, P. A. Huidobro, M. G. Silveirinha, R. Sapienza, A. Alù, J. Pendry, Photonics of time-varying media, arXiv preprint arXiv:2111.08640 (2021).

[66] J. Li, X. Zhu, C. Shen, X. Peng, S. A. Cummer, Transfer matrix method for the analysis of space-time-modulated media and systems, Physical Review B 100 (14) (2019) 144311.