Journal of
open research software

Defrance, J et al 2016 Moosh: A Numerical Swiss Army Knife for the Optics of Multilayers in Octave/Matlab. *Journal of Open Research Software,* 4: e13, DOI: http://dx.doi.org/10.5334/jors.100

SOFTWARE METAPAPER

Moosh: A Numerical Swiss Army Knife for the Optics of Multilayers in Octave/Matlab

Josselin Defrance[1,2], Caroline Lemaître[1,2], Rabih Ajib[1,2], Jessica Benedicto[1,2], Emilien Mallet[1,2], Rémi Pollès[1,2], Jean-Pierre Plumey[1,2], Martine Mihailovic[1,2], Emmanuel Centeno[1,2], Cristian Ciracì[3], David R. Smith[4] and Antoine Moreau[1,2,4]

[1] Université Clermont Auvergne, Université Blaise Pascal, Institut Pascal, BP 10448, F-63000 Clermont-Ferrand, France
joss.defrance@gmail.com

[2] CNRS, UMR 6602, IP, F-63178 Aubière, France

[3] Instituto Italiano di Tecnologia (IIT), Center for Biomolecular Nanotechnologies, Via Barsanti, I-73010 Arnesano, Italy

[4] Center for Metamaterials and Integrated Plasmonics, Duke University, Durham, North Carolina 27708, USA

Corresponding author: Josselin Defrance

The aim of Moosh is to provide a complete set of tools to compute all the optical properties of any multilayered structure: reflection, transmission, absorption spectra, as well as gaussian beam propagation or guided modes. It can be seen as a semi-analytic (making it light and fast) solver for Maxwell's equations in multilayers. It is written in Octave/Matlab, available on Github and based on scattering matrices, making it perfectly stable. This software is meant to be extremely easy to (re)use, and could prove useful in many research areas like photovoltaics, plasmonics and nanophotonics, as well as for educational purposes for the large number of physical phenomena it can illustrate.

Keywords: optics; photonics; plasmonics; photovoltaics; multilayers
Funding statement: This work has been supported by the French National Research Agency, "Physics of Gap-Plasmons" project number ANR-13-JS10-0003.

(1) Overview

Introduction

Multilayered structures have been extensively studied since the beginning of optics. Such structures actually include optical filters, anti-reflective coatings, basic photovoltaic cells, waveguides of many kinds, prism couplers (for the excitation of surface plasmons, for instance) and so on. The invariance of these structures by any translation in two directions makes the solving of Maxwell's equations particularly easy. From the very dawn of electromagnetic optics, analytical solutions have been exhibited for the simplest of these systems and have inspired quantum physics. These solutions are still taught nowadays and help the students form their physical intuition in many areas of optics and quantum physics. With time and the advent of computers, systematic ways for solving the equations provided by Maxwell's theory have been proposed like the transfer matrix formalism [1, 2]. Many programs, that are easily available, are based on these principles, like OpenFilters [3]. Many more codes, much more specialized, are written in the secrecy of labs to compute advanced optical properties of these systems as their physics became more complex. In particular, the advent of metamaterials [4], with totally unforeseen phenomena as negative refraction or perfect lensing [5] have opened new perspectives even for the simplest multi-layered structures. Over the years, the authors of the present paper have built a whole set of tools to assess very peculiar optical properties [6] in multilayers with sometimes very exotic responses (like negative magnetic permeability and negative permittivity [7], or metals with spatial dispersion [8]). These tools have been used to study extensively refraction and focusing by metamaterials [9, 10, 11, 15], photovoltaic devices [16], and exotic coupling like in light wheels [12, 13] and guided modes like gap-plasmons [14, 8] or the ones of semi-conductor lasers. We have now chosen to make these numerical tools freely available to any teacher or researcher, hoping they will help the community study the huge number of phenomenon they allow to simulate.

Implementation and architecture

The implementation of Moosh has actually much to do with a real army knife. A geometrical and an electromagnetic description of the structure are made in a central file *structure.m* in such a way that it is very simple to describe even very complex multilayers. Many programs are then available that are able to compute various optical properties of the structure described in *structure.m.* All these programs are shortly described in **Table 1** and the structure of the whole code is shown **Figure 1**.

Using Moosh thus means editing the *structure.m* file, then editing the program file (the part that should be modified is clearly identified and commented) and choosing the right parameters, then running the main program.

Supporting functions

Moosh is a matlab directory environment that contains the *structure.m* file, main codes that can be directly run and whose name begins with a capital letter, and functions that are called by other programs and that a standard user should never have to edit but that can be used as a library.

The directory *sources* should contain the following files:

Outline of the theory and link to Moosh's structure

In a local, homogeneous and isotropic media, the electric and magnetic fields obey Maxwell's equations that can be written

$$\nabla .\, D = \rho$$
$$\nabla .\, B = 0$$
$$\nabla \times H = j + \partial_t D$$
$$\nabla \times E = -\partial_t B$$

where $D = \epsilon_0 \epsilon\ E$ and $B = \mu_0 \mu H$, ϵ and μ being respectively the relative permittivity and permeability of the medium. From now we consider several media separated by interfaces that are all perpendicular to the *z* axis of the frame we are considering. We moreover assume that no field has any dependency on the *y* variable and that the time dependency is in $e^{-i\omega t}$(with $\omega = \frac{2\pi c}{\lambda}, \lambda$ being the wavelength in vacuum and *c* the speed of light). Under

File name	Calling Sequence	Description
structure.m		Describes the whole structure parameters including permittivity and permeability of every materials, how they are stacked and the thickness of every layer.
Beam.m	*Beam*	Simulates the propagation of a gaussian beam in the structure and maps the resulting field.
Green.m	*Green*	Computes the electromagnetic wave emitted by a source located inside one of the layers.
coefficient.m	*coefficient (theta,lambda)*	Computes the reflection and transmission coefficient of the structure.
absorption.m	*absorption (theta,lambda)*	Computes the percentage of the incident energy that is absorbed inside each layer, along the with reflection and transmission coefficients when the structure is illuminated with a plane wave.
Angular.m	*Angular*	Angular uses *absorption* to compute the absorption in each layer, the reflection and transmission coefficients as a function of the incidence angle of light, for a plane wave.
Spectrum.m	*Spectrum*	Spectrum uses *absorption* to compute the absorption in each layer, the reflection and transmission coefficients as a function of the wavelength of light in vacuum, for a plane wave.
dispersion.m	*dispersion (kx,lambda)*	A function that vanishes whenever a guided mode of the structure has *kx* as a propagation constant for a given *lambda.* The dispersion relation can thus be written as *dispersion(kx,lambda)=0*
descent.m	*descent (z0,step,stop)*	Steepest descent in the *kx* complex plane, starting at *z0* with an initial step *step* and stopping when the absolute value returned by *dispersion* is smaller than *stop.*
Map.m	*Map*	This function maps the response of the dispersion function in the complex plan. This function allows visualizing the position of guided modes in the complex plane.
Guidedmodes.m	*Guidedmodes*	This function uses *descent* to found zeros of the dispersion function to find modes of the structure.
Profile.m	*Profile(kx,lambda)*	Computes the field profile of a guided mode characterized by its propagation constant *kx* (computed using *descent*).
extsqrt.m	*extsqrt(z)*	Square root with a different cut from what is used by default in computers – intented for external layers when searching for guided modes.
Photo.m	*Photo*	Computes the theoretical short-circuit current, shows the absorption spectrum.
cascade.m	*cascade(S,T)*	Cascades two scattering matrices into one single scattering matrix

Table 1: Short description of the files contained in the Moosh directory.

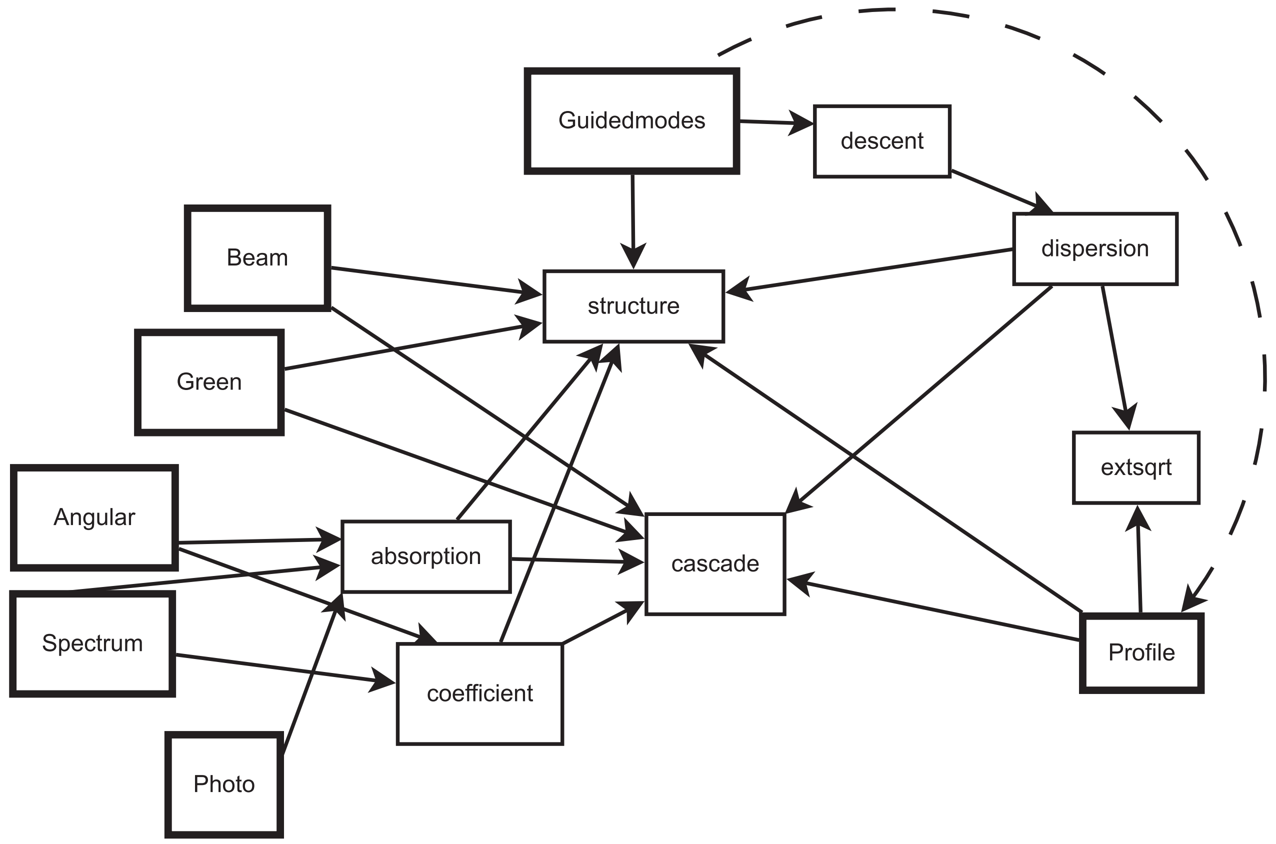

Figure 1: Diagram of Moosh indicating the main programs (bold border) and the functions that are called. Guidedmodes needs to be run before using Profile.

these assumptions the whole problem can actually be separated into two totally independent problems that correspond to actual polarizations of the incoming light, called *s* and *p*. For the s polarization, the only component of the electric field is E_y while for the p polarization H_y is the only component of the magnetic field. They both satisfy the same Helmholtz equation

$$\Delta\psi+\varepsilon\mu\omega^2\psi=0$$

so that both can be written under the same form inside the *j*-th layer (extending from z_j to z_{j+1}):

$$\psi_j(x,z,t)=(A_j^+e^{-ik_{z,j}(z-z_j)}+B_j^+e^{ik_{z,j}(z-z_j)})\cdot e^{i(k_x.x-\omega t)}$$

where $k_x^2+k_z^2=\varepsilon\mu k_0^2$ with $k_0=\frac{2\pi}{\lambda}$.

Maxwell's equations must be satisfied even in the sense of distributions, which leads to the fact that at each interface, two quantities have to remain continuous, that is E_y and $\frac{1}{\mu}\partial_z E_y$ (and not just the derivative of E_y) for *s*-polarization and H_y and $\frac{1}{\epsilon}\partial_z H_y$ for the *p*-polarization. Taking these conditions into account allows considering even the case of negative ϵ and μ [4,5].

This leads to relations between the different A_j^+ and B_j^+. Solving the whole system using classical methods proves to be generally unstable (especially when evanescent waves with an imaginary k_z are involved). Instead, we use scattering matrices [17], well known in quantum mechanics for one-dimensional problems or in the framework of transmission lines. In optics, they have been shown to be a perfectly stable method for almost any kind of problem, provided the determination of the complex square root is chosen for the inner layers so that the imaginary part of k_z is always positive [6]. We define A_j^- and B_j^- by $A_j^+=e^{ik_{z,j}h_i}A_j^-$ and $B_j^-=e^{ik_{z,j}h_i}B_j^+$, where h_j is the thickness of layer *j*. This can be written

$$\begin{bmatrix}A_j^+\\B_j^-\end{bmatrix}=\begin{bmatrix}0 & e^{ik_{z,j}h_i}\\ e^{i.k_{z,j}h_i} & 0\end{bmatrix}\begin{bmatrix}B_j^+\\A_j^-\end{bmatrix}$$

which defines a layer scattering matrix. When now considering an interface between medium *j* and medium *j* + 1, the continuity conditions lead to

$$\begin{bmatrix}A_j^-\\B_{j+1}^+\end{bmatrix}=\frac{1}{g_{j+1}+g_j}\begin{bmatrix}g_j-g_{j+1} & 2.g_{j+1}\\ g_j & g_{j+1}-g_j\end{bmatrix}\cdot\begin{bmatrix}B_j^+\\A_{j+1}^-\end{bmatrix}$$

where $g_j=\frac{k_{z,j}}{\mu_j}$ in *s*-polarization and $g_j=\frac{k_{z,j}}{\epsilon_j}$ in *p*-polarization. This defines an interface scattering matrix. All these scattering matrices can be combined together using a cascading formula [14] that is used in the function *cascade.m*. The scattering matrix corresponding to the whole multilayer simply contains all the reflection and transmission coefficients of the structure.

This method is at the very core of Moosh. The function *coefficient.m* computes the reflection and transmission coefficients of the whole structure. It is used by *Spectrum.m* and *Angular.m* to illustrate how they change when the wavelength resp. the incidence angle is made to vary.

Once the scattering matrix for the whole structure is known, the coefficients $A_j^\pm$ and $B_j^\pm$ can be computed for each layer and then the field E_y or H_y, depending on the polarization, can be computed in each layer. The component of the Poynting vector along the *z* axis can then be computed at each interface. This vector represents the amount of energy that is crossing the interface. In *s*-polarization, it is given at the top of each layer by

$$P_z = \Re\left[\frac{k^*_{z,j}}{2.\mu.\mu_0.\omega}.(A_j^+ + B_j^+)(A_j^{+^*} - B_j^{+^*})\right]$$

and in *p*-polarization by

$$P_z = \Re\left[\frac{k_{z,j}}{2.\in.\in_0.\omega}.(A_j^+ + B_j^+)(A_j^{+^*} - B_j^{+^*})\right].$$

Making the difference between the Poynting flow at the top and at the bottom of each layer gives access to the quantity of light that has been absorbed by any layer. The function *absorption.m* computes this absorption and normalizes it by the total amount of energy with which the structure was illuminated, thus returning the percentage of the incoming energy absorbed by each layer.

The study of photovoltaic devices requires specific tools, as the absorption is not sufficient to evaluate the actual efficiency of a device. The spectral density of photons is required, *i.e.* the number of photons received per wavelength unit. The program *Photo.m*, when the active layer (converting the photons into current) has been specified, computes the theoretical short-circuit current assuming an illumination by an am1.5 solar spectrum and a quantum efficiency (the conversion of photons into electron-hole pairs) of 1.

In order to simulate the propagation of a beam inside the structure, the incident beam can be decomposed on the plane wave basis using a Fourier transform. A computation of the field inside each layer is done for each plane wave, and the results are multiplied by the amplitude of each plane wave in the decomposition of the incident beam and summed. This is what the program *Beam.m* does. All the parameters of the incident beam including the angle of incidence, the width and its position with respect to the computation window have to be specified.

Multilayers are very often used as waveguides in optics. The best way to understand the propagation of light in such structures in a direction parallel to the interfaces, is to consider it as the propagation of different guided modes, whose profile remains unchanged when they propagate. It is thus particularly important to be able to find these modes and to compute their propagation constant (giving access to their phase and group velocity, propagation length) and their profile. These modes satisfy a dispersion relation, that can be written $f(k_x, \lambda) = 0$ with a complex propagation constant k_x and a real wavelength λ. The function *dispersion(kx,lambda)* computes *f* using a modified scattering matrix formalism. Finding the modes finally reduces to finding the zeros of *f*, which can be done using a steepest descent method for $|f|$ in the k_x complex plane. This is the role of *descent.m*. The program *Guidedmodes.m* tries to find all the modes that are quite close to the real axis (which makes them relevant) by performing numerous calls to descent and returning all the different propagation constants (and displaying the effective index k_x/k_0) corresponding to actual guided modes in the variable *guided_modes*. The quantity $\frac{1}{|f|}$ can be mapped using *Map.m*, helping to better understand the results provided by *Guidedmodes.m*. Note that changing the square root determination for the external layers, which can be done by modifying *extsqrt.m*, will change the cuts appearing on the map. This can be useful to reveal or hide poles of $\frac{1}{|f|}$. Finally, the function *Profile.m* computes the profile of the mode characterized by the propagation constant it has been provided with and that can be found with *Guidedmodes.m*.

The electromagnetic field emitted by a punctual source can be decomposed on the plane wave in the same manner as for a Gaussian beam, the only fundamental difference being that the source is placed inside the structure. The program *Green.m* computes the field resulting from placing a punctual source inside any layer of the structure.

Moosh comes with material parameters stored in a directory *data*. This directory contains functions that, provided with a wavelength in vacuum in nanometers, return the value of the relative permittivity for a given material.

Metals, in the above framework, are considered as homogeneous dielectrics characterized by a complex permittivity that can be typically given by a Drude-Lorentz model. But this local description is not fully accurate in some situations when particularly large wavevectors are considered (as is quite common in plasmonics). It is then possible to describe the response of metals using a hydrodynamic model [8]. A semi-analytical method for solving Maxwell's equation in that case can be devised [18]. We have implemented this method and included it in Moosh in a separate directory *nonlocal* with an identical structure to that of *sources*, so that the nonlocal version of Moosh can be used very easily.

Quality control

Moosh has been widely tested over the years. Its results have been extensively compared to completely analytical solutions as can be found in textbooks (for anti-reflective coatings, Bragg mirrors, waveguides), and compared with all kinds of numerical tools that exist including Fourier Modal Methods or finite elements (COMSOL). The stability of the method itself has been assessed by comparing it to transfer matrix methods and Dirichlet-to-Neuman maps and has been improved by carefully choosing the determination of the complex square root inside the layers [6]. Using Moosh, we have been able to replicate results that have been published by other teams in many areas of optics including metamaterials, nanophotonics, plasmonics and photovoltaics.

(2) Availability

Operating system

Moosh needs Octave or Matlab to be installed, which makes it available on Windows (XP SP3 and higher with cygwin), Linux, MacOS X and even Android.

Programming language

Octave 3.8 and higher, Matlab 7 and higher.

List of contributors

Fabien Krayzel and Rémi Pollès wrote the very first version of Beam.

Software location

Code repository

Name: Github

Identifier: https://github.com/AnMoreau/Moosh

Licence: GPL 2.0

Date published: October 5, 2015.

(3) Reuse potential

In order to illustrate the potential of Moosh, a few examples are provided with the code. These include (i) the computation of the absorption in a photovoltaic structure including lossy transparent conducting electrodes, (ii) the reflection corresponding to the excitation of a surface plasmon resonance using a prism coupler and the computation of the corresponding field, as well as the guided modes of the structure (one being actually a leaky mode) (iii) the calculation of the field emitted by a punctual source placed in a contra-directional coupler based on a negative group velocity waveguide, illustrating the light wheel phenomenon in a metamaterial framework. A few output examples are shown **Figure 2**. The functions provided with Moosh are easy to reuse with optimization techniques, for instance. We are confident that Moosh will be useful to the optics community in the future [19, 20].

Finally, we underline that the *s* polarization is actually equivalent to a 1D quantum problem for the wave function of a single particle, if the *x* coordinate is considered as being the time. Moosh can thus be used to simulate the behaviour of the wavefunction in a piecewise potential.

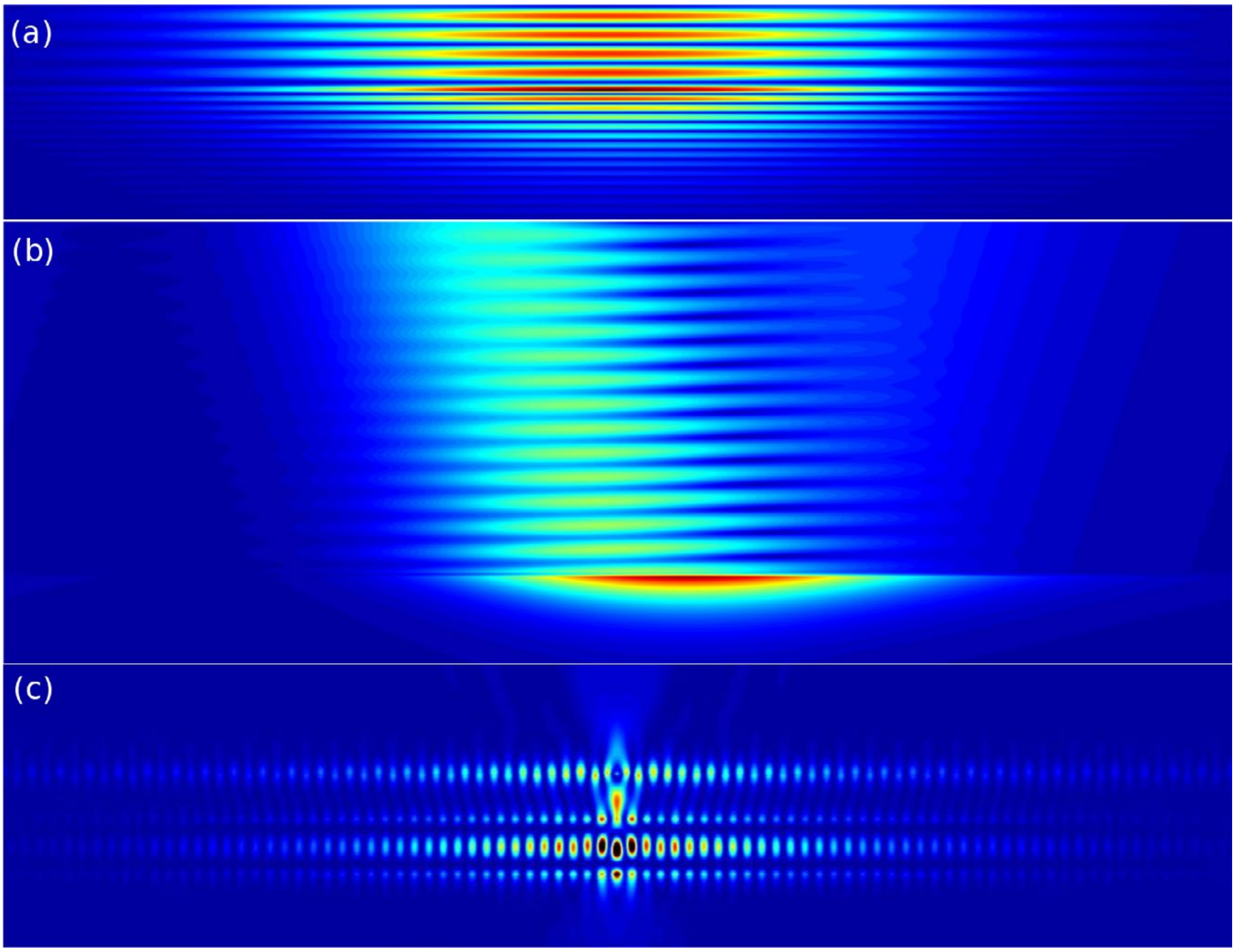

Figure 2: Output examples. (a) Reflection by a Bragg mirror, in which the light penetrates before being totally reflected (b) Excitation of a surface plasmon resonance (c) Excitation of a light wheel by a source placed in the dielectric waveguide.

Competing Interests

The authors declare that they have no competing interests.

Authors Information

Josselin Defrance, wrote part of the code, tested the code.

Caroline Lemaître, tested the code, wrote the documentation

Rabih Ajib, tested the code.

Jessica Benedicto, wrote part of the code, tested the code, contributed to the theory.

Emilien Mallet, supervised students, tested the code.

Rémi Pollès, wrote part of the code, tested the code, contributed to the theory, supervised students.

Jean-Pierre Plumey, contributed to the theory.

Martine Mihailovic, contributed to the theory, tested the code and supervised students.

Emmanuel Centeno, contributed to the theory, tested the code, and supervised students.

Cristian Ciracì, contributed to the theory and tested the code.

David R. Smith, contributed to the theory.

Antoine Moreau, supervised the whole project, contributed to all the aspects.

All the authors contributed to the writing of the article.

References

1. **Abélès, F** 1948 Sur la propagation des ondes electromagnetiques dans les milieux stratifies. *Annales de physique*, 3: 504–520.
2. **Abelès, F** 1950 La théorie générale des couches minces. *J. Phys. Radium*, 11(7): 307–309. DOI: http://dx.doi.org/10.1051/jphysrad:01950001107030700
3. **Larouche, S** and **Martinu, L** 2008 Openfilters: open-source software for the design, optimization, and synthesis of optical filters. *Appl. Opt.*, 47(13): C219–C230. DOI: http://dx.doi.org/10.1364/AO.47.00C219
4. **Smith, D R, Padilla, W J, Vier, D, Nemat-Nasser, S C** and **Schultz, S** 2000 Composite medium with simultaneously negative permeability and permittivity. *Phys. Rev. Lett.*, 84(18): 4184. DOI: http://dx.doi.org/10.1103/PhysRevLett.84.4184
5. **Pendry, J B** 2000 Negative refraction makes a perfect lens. *Phys. Rev. Lett.*, 85(18): 3966. DOI: http://dx.doi.org/10.1103/PhysRevLett.85.3966
6. **Krayzel, F, Polles, R, Moreau, A, Mihailovic, M** and **Granet, G** 2010 Simulation and analysis of exotic non-specular phenomena. *Journal of the European Optical Society-Rapid Publications*, 5: 10025. DOI: http://dx.doi.org/10.2971/jeos.2010.10025
7. **Moreau, A** and **Felbacq, D** 2008 Leaky modes of a left-handed slab. *Journal of the European Optical Society-Rapid Publications*, 3: 08032. DOI: http://dx.doi.org/10.2971/jeos.2008.08032
8. **Moreau, A, Ciracì, C** and **Smith, D R** 2013 Impact of nonlocal response on metallodielectric multilayers and optical patch antennas. *Phys. Rev. B*, 87(4): 045401. DOI: http://dx.doi.org/10.1103/PhysRevB.87.045401
9. **Pollès, R, Centeno, E, Arlandis, J** and **Moreau, A** 2011 Self-collimation and focusing effects in zero-average index metamaterials. *Opt. Expr.*, 19(7): 6149–6154. DOI: http://dx.doi.org/10.1364/OE.19.006149
10. **Benedicto, J, Pollès, R, Moreau, A** and **Centeno, E** 2011 Large negative lateral shifts due to negative refraction. *Opt. Lett.*, 36(13): 2539–2541. DOI: http://dx.doi.org/10.1364/OL.36.002539
11. **Bénédicto, J, Centeno, E** and **Moreau, A** 2012 Lens equation for flat lenses made with hyperbolic metamaterials. *Opt. Lett.*, 37(22): 4786–4788. DOI: http://dx.doi.org/10.1364/OL.37.004786
12. **Tichit, P-H, Moreau, A** and **Granet, G** 2007 Localization of light in a lamellar structure with left-handed medium: the light wheel. *Opt. Expr.*, 15(23): 14961–14966. DOI: http://dx.doi.org/10.1364/OE.15.014961
13. **Polles, R, Moreau, A** and **Granet, G** 2010 Light wheel buildup using a backward surface mode. *Opt. Lett.*, 35(19): 3237–3239. DOI: http://dx.doi.org/10.1364/OL.35.003237
14. **Moreau, A, Ciracì, C, Mock, J J, Hill, R T, Wang, Q, Wiley, B J, Chilkoti, A** and **Smith, D R** 2012 Controlled-reflectance surfaces with film-coupled colloidal nanoantennas. *Nature*, 492(7427): 86–89. DOI: http://dx.doi.org/10.1038/nature11615
15. **Bénédicto, J, Centeno, E, Pollès, R** and **Moreau, A** 2013 Ultimate resolution of indefinite metamaterial flat lenses. *Phys. Rev. B*, 88: 245138. DOI: http://dx.doi.org/10.1103/PhysRevB.88.245138
16. **Moreau, A, Smaali, R, Centeno, E** and **Seassal, C** 2012 Optically optimal wavelength-scale patterned ito/zno composite coatings for thin film solar cells. *J. Appl. Phys.*, 111(8): 083102–083102. DOI: http://dx.doi.org/10.1063/1.3703670
17. **Orfanidis, S J** 2002 *Electromagnetic waves and antennas'.* New Brunswick, NJ: Rutgers University. Available at http://eceweb1.rutgers.edu/~orfanidi/ewa/orfanidis-ewa-book.pdf.
18. **Benedicto, J, Pollès, R, Ciracì, C, Centeno, E, Smith, D R** and **Moreau, A** 2015 Numerical tool to take nonlocal effects into account in metallo-dielectric multilayers. *J. Opt. Soc. Am. A*, 32(8): 1581–1588. DOI: http://dx.doi.org/10.1364/JOSAA.32.001581
19. **Smalley, J S, Vallini, F, Shahin, S, Kanté, B** and **Fainman, Y** 2015 Gain-enhanced high-k transmission through metal-semiconductor hyperbolic metamaterials. *Opt. Mat. Express*, 5: 2300–2312. DOI: http://dx.doi.org/10.1364/OME.5.002300
20. **Riley, C T, Smalley, J S, Post, K W, Basov, D N, Fainman, Y, Wang, D, Liu, Z** and **Sirbuly, D J** 2015 High-Quality, Ultraconformal Aluminum-Doped Zinc Oxide Nanoplasmonic and Hyperbolic Metamaterials. *Small*. DOI: http://dx.doi.org/10.1002/smll.201501797

How to cite this article: Defrance, J, Lemaître, C, Ajib, R, Benedicto, J, Mallet, E, Pollès, R, Plumey, J-P, Mihailovic, M, Centeno, E, Ciracì, C, Smith, D R and Moreau, A 2016 Moosh: A Numerical Swiss Army Knife for the Optics of Multilayers in Octave/Matlab. *Journal of Open Research Software,* 4: e13, DOI: http://dx.doi.org/10.5334/jors.100

Submitted: 09 October 2015 **Accepted:** 09 March 2016 **Published:** 22 April 2016

]u[*Journal of Open Research Software* is a peer-reviewed open access journal published by Ubiquity Press

OPEN ACCESS