

# OpenFilters: open-source software for the design, optimization, and synthesis of optical filters

Stéphane Larouche\* and Ludvik Martinu

Regroupement québécois sur les matériaux de pointe (RQMP) and Department of Engineering Physics,  
École polytechnique de Montréal, P.O. Box 6079, Station Centre-Ville, Montréal, Québec H3C 3A7, Canada

\*Corresponding author: stephane.larouche@polymtl.ca

Received 18 September 2007; revised 19 October 2007; accepted 19 October 2007;  
posted 1 November 2007 (Doc. ID 87410); published 23 January 2008

The design of optical filters relies on powerful computer-assisted methods. Many of these methods are provided by commercial programs, but, in order to adapt and improve them, or to develop new methods, one needs to create his own software. To help people interested in such a process, we decided to release our in-house software, called OpenFilters, under the GNU General Public License, an open-source license. It is programmed in Python and C++, and the graphical user interface is implemented with wxPython. It allows creation of multilayer and graded-index filters and calculation of reflection, transmission, absorption, phase, group delay, group delay dispersion, color, ellipsometric variables, admittance diagram, circle diagram, electric field distribution, and generation of reflection, transmission, and ellipsometric monitoring curves. It also provides the refinement, needle, step, and Fourier transform methods. © 2008 Optical Society of America

*OCIS codes:* 310.0310, 310.1620, 310.4165, 310.5696, 310.6805.

## 1. Introduction

The conception of optical filters was revolutionized when, 50 years ago, Baumeister demonstrated the possibility to optimize optical filters using a computer program [1]. Before then, most optical filters consisted of quarter wave layers, and their design relied on graphical methods. Since then, a large variety of complex design procedures have been developed that rely on sophisticated numerical methods and the use of computers [2]. Nowadays, we have at our disposal very powerful synthesis methods, such as the needle method, that allow one to design optical filters for any physically sound problem.

Many commercial programs are available to design optical filters; the most important ones were reviewed in 2003 by Kruschwitz [3]. Despite the high quality of these commercial programs, many companies and research institutions continue to develop their own in-house software. In-house software can be rapidly modified when a new problem arises and adapted to fit the specific needs of the organization,

while commercial programs, by their nature, are less customizable.

At the Functional Coating and Surface Engineering Laboratory (FCSEL) we developed an expertise in the design and fabrication of graded-index and intermediate index optical filters and found that commercial programs offer only limited capabilities for the design of such filters. Also in recent years we developed new design methods, and we wanted to incorporate them in a complete design environment, including popular design methods and an easy-to-use graphical user interface. This brought us to the creation, from scratch, of a new design program, a process that took us many years.

In the hope of helping other groups with similar goals to avoid the same troubles, we decided to release this program, called OpenFilters, as open-source software under the GNU General Public License (GPL). It is freely available on our website, <http://www.polymtl.ca/larfis>. We believe that the availability of such open-source software is a significant advance for the optical coating community. It will reduce the barrier encountered by people entering the community by giving them the opportunity to start developing new features from a fully functional

program. We hope that this will stimulate the improvement of existing analysis and design methods as well as the creation of new ones.

Our aim is to present what OpenFilters offers so the reader can decide whether there is interest in using it to justify the choices we made during its creation, to show how to use it, to explain under what conditions it can be used or modified, and to describe its implementation so that one can determine if it can be adapted to fit his or her specific needs.

We first present the GPL, followed by a review of the methods used by OpenFilters to calculate the properties of optical filters, as well as the design, optimization, and synthesis methods it proposes. Contrary to our other publications about the design and synthesis methods we developed, we do not attempt to provide a complete development of the design methods; we present only the main equations used by OpenFilters and give references for interested readers to learn more. We then present a very short user guide and the structure of the implementation of OpenFilters. Finally, we outline some plans for the future development of OpenFilters and indicate how one can collaborate in its development.

## 2. GNU General Public License

The GPL [4] is an open source license, or free software license. It gives the user the freedom to access the source code of the program and to modify it to suit its needs. The user does not need to distribute the modifications made to OpenFilters. But, if one decides to do so, the derived work must also be released under the GPL. One should carefully read the GPL before distributing any work derived from OpenFilters.

The GPL does not apply to the results obtained with OpenFilters. It can be used to design filters for commercial applications, and the results can be published under different conditions. However, we would be grateful if you refer to this article or our website when publishing results obtained with OpenFilters.

## 3. What OpenFilters Offers: a Review of the Calculation, Design, Optimization, and Synthesis of Optical Filters

In this section we provide basic information to understand how OpenFilters designs, optimizes, synthesizes, and calculates the optical properties of interference filters.

### A. Calculation of the Optical Properties

As shown in Fig. 1, a classical optical filter consists of a stack of layers of thicknesses  $d_i$  and indices of refraction  $N_i$  between an incidence and an exit media of indices of refraction  $N_{\text{inc}}$  and  $N_{\text{ex}}$ , respectively.

OpenFilters calculates the filter properties using the characteristic matrix approach developed by Abelès [5] and presented in detail in most optical coating textbooks such as Macleod's [6]. In this exact approach, the  $i$ th layer is represented by

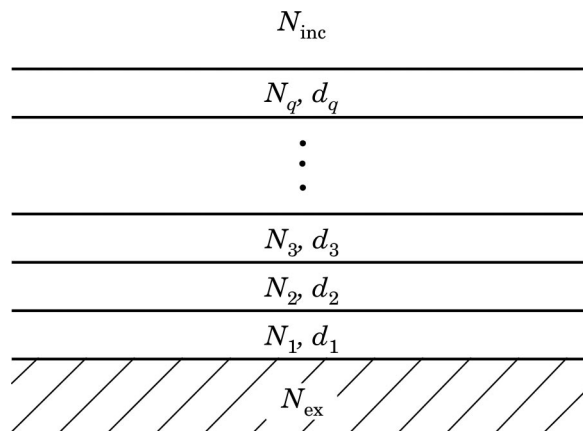


Fig. 1. Multilayer optical filter.

$$M_i = \begin{bmatrix} \cos \varphi_i & (i/\eta_i) \sin \varphi_i \\ i\eta_i \sin \varphi_i & \cos \varphi_i \end{bmatrix}, \quad (1)$$

where

$$\eta_i = \begin{cases} N_i \cos \theta_i & \text{s polarization} \\ N_i / \cos \theta_i & \text{p polarization} \end{cases} \quad (2)$$

is the pseudoindex of the layer,

$$\varphi_i = \frac{2\pi}{\lambda} N_i d_i \cos \theta_i \quad (3)$$

is the phase shift of the wave inside the layer,  $\theta_i$  is the angle of propagation in the layer, and  $\lambda$  is the wavelength of light in vacuum.

According to the Snell–Descartes' law,  $\alpha = N_i \sin \theta_i$  is constant [7]. To avoid computer intensive trigonometric function calculations, the pseudoindices and phase shift can be calculated using

$$\eta_i = \begin{cases} \sqrt{N_i^2 - \alpha^2} & \text{s polarization} \\ N_i^2 / \sqrt{N_i^2 - \alpha^2} & \text{p polarization} \end{cases}, \quad (4)$$

$$\varphi_i = \frac{2\pi}{\lambda} \sqrt{N_i^2 - \alpha^2} d_i \quad (5)$$

instead of Eqs. (2) and (3), respectively.

The characteristic matrix describing the multilayer is

$$M = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} = \prod_{i=q}^1 M_i, \quad (6)$$

where  $q$  is the number of layers, and the product is taken in reverse order since the matrices of upper layers must be multiplied on the left.

The amplitude reflection and transmission coefficients of the multilayer are

$$r = \frac{\eta_{\text{inc}} m_{11} - \eta_{\text{ex}} m_{22} + \eta_{\text{inc}} \eta_{\text{ex}} m_{12} - m_{21}}{\eta_{\text{inc}} m_{11} + \eta_{\text{ex}} m_{22} + \eta_{\text{inc}} \eta_{\text{ex}} m_{12} + m_{21}}, \quad (7)$$

$$t = \frac{2\eta_{\text{inc}}}{\eta_{\text{inc}} m_{11} + \eta_{\text{ex}} m_{22} + \eta_{\text{inc}} \eta_{\text{ex}} m_{12} + m_{21}}, \quad (8)$$

where  $\eta_{\text{inc}}$  and  $\eta_{\text{ex}}$  are the pseudoindices of the incidence and exit media, respectively. The reflectance and transmittance are

$$R = rr^* = |r|^2, \quad (9)$$

$$T = \frac{\text{Re } N_{\text{ex}}}{\text{Re } N_{\text{inc}}} tt^* = \frac{\text{Re } N_{\text{ex}}}{\text{Re } N_{\text{inc}}} |t|^2, \quad (10)$$

where  $\text{Re}$  is the real part and  $*$  denotes the complex conjugate. The absorptance is simply

$$A = 1 - R - T. \quad (11)$$

When dealing with a rapidly changing signal, such as a pulse train, one also needs to know the dispersive properties of the filter. The global phase shifts on reflection and transmission modulo  $2\pi$ ,  $\phi_r$ , and  $\phi_t$  are the angles made by  $r$  and  $t$  in the complex plane so that

$$\phi_r = \arg r = \arctan \frac{\text{Im } r}{\text{Re } r}, \quad (12)$$

$$\phi_t = \arg t = \arctan \frac{\text{Im } t}{\text{Re } t}, \quad (13)$$

where  $\text{Im}$  is the imaginary part and the signs of the numerator and the denominator must be evaluated separately.

By themselves, the phase shifts do not provide much information about the behavior of the filter. One is usually more interested in the delay introduced when a pulse is reflected or transmitted by the filter, the group delay (GD), or in the deformation of a finite bandwidth pulse caused by the variation of GD with the wavelength, the group delay dispersion (GDD). GD and GDD are obtained from the phase using

$$\text{GD} = -\frac{d\phi}{d\omega}, \quad (14)$$

$$\text{GDD} = -\frac{d^2\phi}{d\omega^2}, \quad (15)$$

where  $\omega = 2\pi c/\lambda$  is the angular frequency and  $c$  is the speed of light in vacuum. There are analytical formulas to calculate the GD and GDD, but it is impossible to calculate them analytically from the characteristic matrix of a multilayer. For that reason, OpenFilters cal-

culates GD and GDD numerically from the  $\phi_r$  or  $\phi_t$  spectrum.

Since ellipsometry is commonly used to characterize optical filters, OpenFilters can generate the ellipsometric spectrum of a filter. Following the convention most commonly used in the optical thin film community [8], the ellipsometric angles  $\Psi$  and  $\Delta$  are defined by

$$\rho = \frac{-r_p}{r_s} = \tan \Psi e^{i\Delta}, \quad (16)$$

where  $r_s$  and  $r_p$  are the amplitude reflection coefficients for s- and p-polarized light, respectively. Therefore,

$$\Psi = \arctan \frac{|r_p|}{|r_s|}, \quad \Delta = \arg(-r_p) - \arg(r_s). \quad (17)$$

There also are applications, such as color shifting security devices, where the color of a filter is of interest. To calculate it, one must multiply the spectrum of an illuminant by that of the filter and color matching functions representing the sensitivity of the eye. OpenFilters strictly follows the CIE 15: 2004 standard [9]. OpenFilters also offers tools that provide an intuitive understanding of some filters: the admittance diagram [6,10], the circle diagram [11], and the electric field distribution [12].

Finally, to prepare the filter deposition, OpenFilters can generate the reflection, transmission, or ellipsometric monitoring curves. To do so, it simply *grows* the filter by separating it in thin sublayers that are successively added and by calculating the properties after the addition of each of these sublayers.

## B. Consideration of the Backside

Up to this point we have considered the case of a semi-infinite exit medium. This is clearly unrealistic, and it is necessary to account for the effect of the substrate backside. Since the coherence length of light is usually shorter than the substrate thickness, this case must be treated differently from that of a layer. As shown in Fig. 2, one can simply add multiple reflections and transmissions occurring in the substrate. They follow geometric series whose sums are

$$R = R_F + \frac{T_F T_{FR} R_B \exp 4\beta_i}{1 - R_{FR} R_B \exp 4\beta_i}, \quad (18)$$

$$T = \frac{T_F T_B \exp 2\beta_i}{1 - R_{FR} R_B \exp 4\beta_i}, \quad (19)$$

where  $R_F$  and  $T_F$  are the reflectance and transmittance of the front side,  $R_{FR}$  and  $T_{FR}$  are the reflectance and transmittance of the front side in reverse direction,  $R_B$  and  $T_B$  are the reflectance and transmittance of the backside,

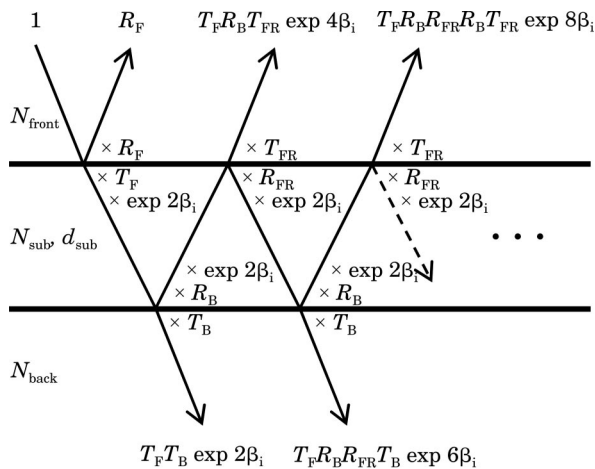


Fig. 2. Optical filter with backside.

$$\beta_i = \text{Im} \left( \frac{2\pi}{\lambda} \sqrt{N_{\text{sub}}^2 - \alpha^2 d_{\text{sub}}} \right) \quad (20)$$

is related to the absorption in the substrate, and  $N_{\text{sub}}$  is the index of refraction of the substrate.

The different reflection and transmission coefficients can be calculated from Eqs. (9) and (10) while considering appropriate incidence and exit media. It should also be noted that, since the characteristic matrices Eq. (6) are persymmetric, the product  $\prod_{i=1}^q M_i$  can be obtained from  $\prod_{i=q}^1 M_i$  by simply interchanging the diagonal elements.

The situation is more complicated for ellipsometry since the incoherent effect of the front side and backside phase shifts must be considered. In that case, OpenFilters uses the formalism of Yang and Abelson [13].

### C. Graded-Index Filters

The index of refraction of graded-index filters varies continuously throughout the thickness. It is possible to treat some simple cases analytically, but this approach is very limited. A more general approach consists in approximating the index gradient by a series of very thin sublayers [14]. The thickness of the sublayers must be small enough to correctly represent the index profile, but large enough to limit calculation time and avoid numerical errors.

The discretization of the index profile is usually accomplished by a series of sublayers with constant physical or optical thickness. This approach is simple since the software needs to calculate only the function generating the index profile at equally spaced points. However, when one wants to calculate the properties of a filter made of real materials with dispersive refractive index, it is necessary to calculate the dispersion of the index for every sublayer. To avoid these repeated calculations, OpenFilters discretizes the index profile using predefined index levels.

The discretization process is shown in Fig. 3. First, the index profile  $n(z)$  is calculated at a large number of points with a thickness interval chosen to provide

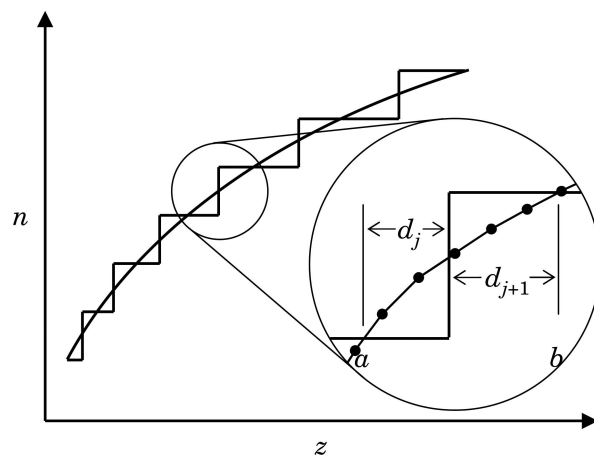


Fig. 3. Discretization of the refractive index profile of a graded-index filter (see text for details).

multiple points in each index interval. Then, for each index interval, linear interpolation is used to determine the points  $a$  and  $b$ , where the index profile crosses the predefined index levels  $n_j$  and  $n_{j+1}$ , defining a small section of the index profile. Finally, that section of the index profile is replaced by two sublayers of thicknesses  $d_j$  and  $d_{j+1}$  so that

$$d_j + d_{j+1} = b - a, \quad (21)$$

$$n_j d_j + n_{j+1} d_{j+1} = \int_a^b n(z) dz, \quad (22)$$

where the integral is evaluated using the trapezoidal rule. This guarantees that the discretized filter preserves both the physical thickness and the optical thickness of the original index profile.

### D. Optimization of Multilayer Filters

It is often difficult, or even impossible, to *manually* design a filter respecting all the specifications for a given application. However, it is possible to optimize a filter using refinement [1]. In this method, the designer provides a starting design whose properties are sufficiently close to the specifications. Then, an optimization algorithm is used to adjust the thickness of the layers and eventually their index of refraction.

In addition to the starting design, the designer must provide targets describing the specifications. The targets can be any property computable from the parameters of the filter; usually, these are the reflection, the transmission, phase properties, or the color defined for some polarization and angle of incidence. Depending on the application, it may be necessary to define a target at a single wavelength or over a range of wavelengths. In the latter case, the target must in fact be discretized and defined at a finite number of wavelengths. Using the targets, it is possible to define a merit function (MF), representing the quality of a design, that the refinement must minimize. OpenFilters uses

$$\chi^2 = \sum_{i=1}^m \left( \frac{B_i - \bar{B}_i}{\Delta B_i} \right)^2, \quad (23)$$

where  $m$  is the number of targets,  $B_i$  is a property of interest,  $\bar{B}_i$  is the target value for that property, and  $\Delta B_i$  is the tolerance for that property.

The optimization algorithm used by OpenFilters is the Levenberg–Marquardt algorithm [15]. The properties of a filter that are optimized, the thickness and the index of refraction of the layers, are represented by a vector  $\mathbf{a} = [a_1, a_2, \dots]$ . The Levenberg–Marquardt algorithm calculates the first derivatives of the merit function with regard to every parameter  $a_k$ ,

$$\frac{d\chi^2}{da_k} = 2 \sum_{i=1}^m \left[ \frac{B_i - \bar{B}_i}{(\Delta B_i)^2} \frac{dB_i}{da_k} \right], \quad (24)$$

and it uses these derivatives to determine the point where the MF is the lowest in a trust region centered around the present value of  $\mathbf{a}$ . Repeated improvements of the MF allow the Levenberg–Marquardt algorithm to find a local minimum.

Following van der Laan and Frankena [16], let us now show how to calculate  $dB_i/da_k$  in the case where  $B_i$  is the reflection or the transmission. According to Eqs. (9) and (10),

$$\frac{dR}{da_k} = 2r^* \frac{dr}{da_k}, \quad (25)$$

$$\frac{dT}{da_k} = 2 \frac{\text{Re}(N_s)}{\text{Re}(N_m)} t^* \frac{dt}{da_k}. \quad (26)$$

According to Verly [17], the derivative of the amplitude reflection coefficient  $r$  relative to the parameter  $a_k$  can be calculated using

$$\frac{dr}{da_k} = \text{Tr} \left( \frac{dM}{da_k} \psi_r \right), \quad (27)$$

where Tr is the trace and

$$\psi_r = \frac{t}{2N_{\text{inc}}} \begin{bmatrix} N_{\text{inc}}(1-r) & -(1+r) \\ N_{\text{inc}}N_{\text{ex}}(1-r) & -N_{\text{ex}}(1+r) \end{bmatrix}. \quad (28)$$

Using a similar formalism, the derivative of the amplitude transmission coefficient can be calculated using

$$\frac{dt}{da_k} = \text{Tr} \left( \frac{dM}{da_k} \psi_t \right), \quad (29)$$

where

$$\psi_t = -\frac{t^2}{2N_{\text{inc}}} \begin{bmatrix} N_{\text{inc}} & 1 \\ N_{\text{inc}}N_{\text{ex}} & N_{\text{inc}} \end{bmatrix}. \quad (30)$$

According to Eq. (6), the derivative of the matrix of the multilayer relative to a parameter  $a_k$  of the  $j$ th layer is

$$\frac{dM}{da_k} = \prod_{i=q}^{j+1} M_i \frac{dM_j}{da_k} \prod_{i=j-1}^1 M_i. \quad (31)$$

If parameter  $a_k$  is the thickness  $d_j$  of the layer, Eqs. (1) and (5) give

$$\frac{dM_j}{dd_j} = \frac{dM_j}{d\varphi_j} \frac{d\varphi_j}{dd_j}, \quad (32)$$

where

$$\frac{dM_j}{d\varphi_j} = \begin{bmatrix} -\sin \varphi_j & (i/\eta_j) \cos \varphi_j \\ i\eta_j \cos \varphi_j & -\sin \varphi_j \end{bmatrix}, \quad (33)$$

$$\frac{d\varphi_j}{dd_j} = \frac{2\pi}{\lambda} \sqrt{N_j^2 - \alpha^2}. \quad (34)$$

Similarly, if  $a_k$  is the index of refraction of the layer,

$$\frac{dM_j}{dN_j} = \frac{dM_j}{d\varphi_j} \frac{d\varphi_j}{dN_j} + \frac{dM_j}{d\eta_j} \frac{d\eta_j}{dN_j}, \quad (35)$$

where  $dM_j/d\varphi_j$  is given by Eq. (33),

$$\frac{dM_j}{d\eta_j} = \begin{bmatrix} 0 & -(i/\eta_j^2) \sin \varphi_j \\ i \sin \varphi_j & 0 \end{bmatrix}, \quad (36)$$

$$\frac{d\eta_j}{dN_j} = \begin{cases} \frac{N_j}{\sqrt{N_j^2 - \alpha^2}} & \text{s polarization} \\ \frac{N_j}{\sqrt{N_j^2 - \alpha^2}} \left( 2 - \frac{N_j^2}{N_j^2 - \alpha^2} \right) & \text{p polarization} \end{cases}, \quad (37)$$

$$\frac{d\varphi_j}{dN_j} = \frac{2\pi}{\lambda} \frac{N_j}{\sqrt{N_j^2 - \alpha^2}} d_j. \quad (38)$$

Note that the index of refraction of a layer is usually defined by its real part  $n_{j,0} = \text{Re}(N_{j,0})$  at a reference wavelength  $\lambda_0$ . In that case,

$$\frac{dM_j}{dn_{j,0}} = \frac{dM_j}{dN_j} \frac{dN_j}{dn_{j,0}}, \quad (39)$$

where  $dN_j/dn_{j,0}$  is determined from the dispersion of the refractive index of the material.

OpenFilters also offers the possibility to refine the index of refraction of a layer while keeping its optical thickness constant [18]. Details will be published in an upcoming article.

Finally, the effect of the backside must be considered. According to Eqs. (18) and (19), the derivatives



of the reflectance and transmittance by a parameter  $a_k$  of the front stack are

$$\frac{dR}{da_k} = \frac{dR_F}{da_k} + \frac{T_F T_{FR} R_B \exp 4\beta_i}{1 - R_{FR} R_B \exp 4\beta_i} \left( \frac{1}{T_F} \frac{dT_F}{da_k} + \frac{1}{T_R} \frac{dT_R}{da_k} + \frac{R_B \exp 4\beta_i}{1 - R_{FR} R_B \exp 4\beta_i} \frac{dR_{FR}}{da_k} \right), \quad (40)$$

$$\frac{dT}{da_k} = T \left( \frac{1}{T_F} \frac{dT_F}{da_k} + \frac{R_B \exp 4\beta_i}{1 - R_{FR} R_B \exp 4\beta_i} \frac{dR_{FR}}{da_k} \right). \quad (41)$$

## E. Synthesis of Multilayer Filters

Even after being refined, the filter does not always fully respect the specifications. Or it can be difficult for the designer to provide a starting design. In that case, one must resort to synthesis methods. The most commonly used synthesis method is the needle method. OpenFilters also offers the step method that we developed.

### 1. Needle Method

The needle method was first described by Tikhonravov in 1982 [19]. However, it only began to be widely used in the middle of the 1990s [20,21]. As shown in Fig. 4(a), it consists of adding thin layers at optimal positions in the filter and then adjusting their thickness by refinement. Needles are added until a satisfactory solution is found or the addition of more needles does not improve the filter.

The optimal position to add needles is determined by calculating the derivative of the MF with regard to the thickness of an infinitesimally thin layer as a function of the position where it is added. Where the derivative is negative, it is favorable to add a needle. Usually, one needle at a time is added at the position where the derivative of the MF is the most negative.

The addition of a needle in the  $j$ th layer represented by the matrix  $M_j$ , which has a thickness corresponding to a phase shift  $\varphi_j$ , creates two sublayers represented by matrices  $M_{j,2}$  and  $M_{j,1}$  with thicknesses corresponding to phases  $\varphi_{j,2}$  and  $\varphi_{j,1}$  such that

$\varphi_j = \varphi_{j,2} + \varphi_{j,1}$ . These two sublayers are separated by a needle represented by the matrix  $M_n$  and  $M_j = M_{j,2} M_n M_{j,1}$ . The effect of the addition of the needle of index  $N_n$  and infinitesimal thickness  $d_n$  is, therefore,

$$\left. \frac{dM_j}{dd_n} \right|_{d_n=0} = M_{j,2} \left. \frac{dM_n}{dd_n} \right|_{d_n=0} M_{j,1}. \quad (42)$$

To speed up the calculations, Eq. (42) can be rewritten as

$$\left. \frac{dM_j}{dd_n} \right|_{d_n=0} = \frac{1}{2} \left[ \left( \frac{\eta_j}{\eta_n} + \frac{\eta_n}{\eta_j} \right) \frac{dM_j}{d\varphi_j} + \left( \frac{\eta_j}{\eta_n} - \frac{\eta_n}{\eta_j} \right) \times \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \frac{dM_j(\Delta\varphi_j)}{d\Delta\varphi_j} \right] \frac{d\varphi_n}{dd_n}, \quad (43)$$

where  $\eta_n$  is the pseudoindex of the needle,  $\varphi_n = (2\pi/\lambda)\sqrt{N_n^2 - \alpha^2}d_n$  is the phase shift in the needle, and  $dM_j(\Delta\varphi_j)/d\Delta\varphi_j$  is identical to Eq. (33) in which  $\varphi_j$  is replaced by  $\Delta\varphi_j = \varphi_{j,1} - \varphi_{j,2}$ . Equation (43) is faster to compute than Eq. (42) because there is no matrix multiplication,  $dM_j/d\varphi_j$  and  $dM_j(\Delta\varphi_j)/d\Delta\varphi_j$  do not depend on the needle material, and  $dM_j/d\varphi_j$  does not depend on the position of the needle.

### 2. Step Method

When materials with a continuous range of index of refraction are available, Verly proposed to determine both the optimal position and the index of refraction of the added needle [17]. On our side, we proposed to add steps in the index profile instead of needles [22]. As shown in Fig. 4(b), the step method consists of adding infinitesimal steps in the index profile and then refining the index of refraction and the thickness of the separated layers.

Similar to the needle method, the optimal position to add steps is determined by calculating the derivative of the MF with regard to the addition of a step as a function of the position where the step is added.

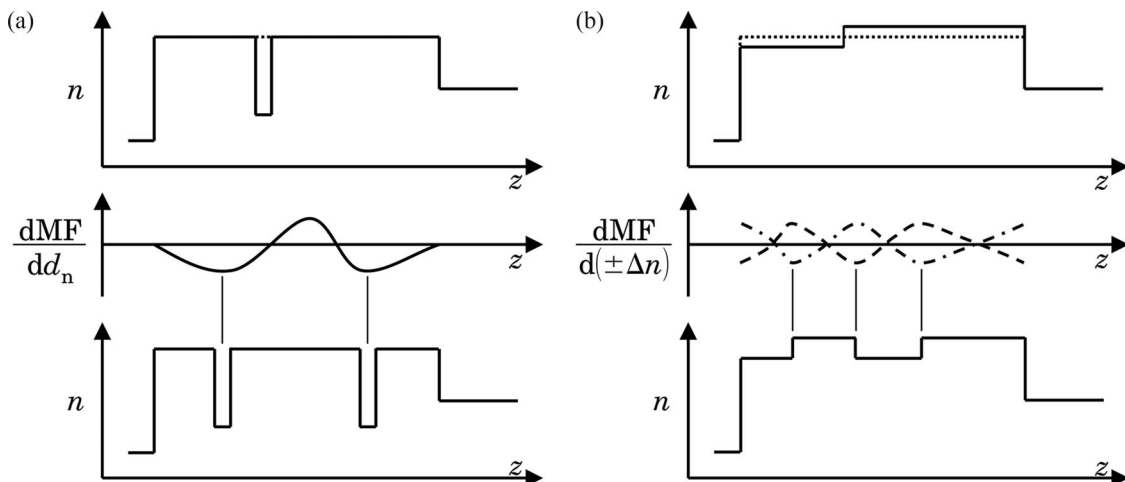


Fig. 4. Schematic representation of (a) the needle method and (b) the step method.

The  $j$ th layer is separated in two sublayers, and the refractive index of the first sublayer is increased while that of the second sublayer is decreased (or vice versa). If the index change is equally distributed on both sublayers, the effect of the addition of a step is

$$\left. \frac{dM_j}{d\Delta N_j} \right|_{\Delta N_j=0} = \frac{1}{2} \frac{dM_{j,2}}{dN_j} M_{j,1} - M_{j,2} \frac{1}{2} \frac{dM_{j,1}}{dN_j}. \quad (44)$$

To speed up the calculation, this equation can be rewritten as

$$\left. \frac{dM_j}{d\Delta N_j} \right|_{\Delta N_j=0} = \frac{1}{2} \left[ \frac{dM_j}{d\varphi_j} \frac{d\Delta\varphi_j}{dN_j} + \left( \frac{dM_j(\Delta\varphi_j)}{d\eta_j} + \frac{1}{\eta_j} (\cos \varphi_j - \cos \Delta\varphi_j) \right) \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \frac{d\eta_j}{dN_j} \right], \quad (45)$$

where  $dM_j(\Delta)/d\eta_j$  is identical to Eq. (36), where  $\varphi_j$  is replaced by  $\Delta\varphi_j$ .

If the index of refraction of the layer where the step is added is at a limit of the available index range, it is impossible to distribute the change of index equally, and the calculation must be slightly modified. Details about the step method will be published in an upcoming article.

#### F. Design of Graded-Index Filters

We have seen how to design, optimize, and synthesize multilayer filters. We now turn to the design of graded-index filters. We present the Fourier transform method and a direct application, the design of rugate filters, as well as quintic antireflective layers.

##### 1. Fourier Transform Method

The Fourier transform method for the design of optical filters was developed by Delano [23] and Sossi [24], and later popularized by Dobrowolski and Lowe [25]. It directly relates the desired reflection or transmission spectra with the index profile of the filter through the use of an amplitude and phase functions,  $Q$  and  $\Psi$ , respectively, called the  $Q$  function. The index depth profile  $n(x)$  is obtained by using the inverse Fourier transform [25]

$$\ln \frac{n(x)}{n_m} = \frac{2}{\pi} \int_0^\infty \frac{Q(\sigma)}{\sigma} \sin(\Psi(\sigma) - \sigma x) d\sigma, \quad (46)$$

where  $x$  is the double-centered optical thickness,  $\sigma = 2\pi/\lambda$  is the wavenumber,  $\lambda$  is the wavelength, and

$$n_m = \sqrt{n_{\min} n_{\max}} \quad (47)$$

is the geometric mean index of the filter. The double-centered optical thickness

$$x = 2 \int_0^z n(u) du \quad (48)$$

is twice the optical thickness measured from the center of the filter, where  $z$  is the physical thickness and  $u$  is an integration variable. Note that in real applications, the  $Q$  function is defined at a finite number of wavelengths in a limited range, and the integral of Eq. (46) is replaced by a sum.

Multiple  $Q$  functions have been proposed [23–32], and OpenFilters offers many of them. However, since the  $Q$  functions are only approximate, there will be discrepancies between the target and the obtained filter. This problem can be solved by iteratively correcting the  $Q$  function [33]. OpenFilters also considers the dispersion of the index of refraction in the Fourier transform method using a method we developed [34].

##### 2. Rugate Filters

A direct application of the Fourier transform method is the design of single- or multiple-wavelength reflectors called rugate filters [26]. It can easily be understood that the Fourier transform of a  $Q$  function defined for a single wavelength gives a sinusoidal index profile. Furthermore, a multiband rugate filter can easily be designed by multiplying multiple sinusoidal profiles:

$$\ln \frac{n(x)}{n_m} = \prod_i \bar{Q}_i \sin(2\pi x/\lambda_i + \varphi_i) w(x), \quad (49)$$

where  $\lambda_i$  are the wavelengths of the band,  $\varphi_i$  are phase shifts that allow one to center the index profile, and

$$\bar{Q}_i = \frac{1}{2} \ln \frac{n_m + \Delta n_i/2}{n_m - \Delta n_i/2} \quad (50)$$

are the reduced  $Q$  functions that control the amplitude  $\Delta n_i$  of the index profile.

Since the sinusoidal index profile cannot be infinite, it is necessary to multiply it by a finite envelope  $w(x)$ . The spectrum of the filter is the convolution of the Fourier transform of the sinus and that of the envelope. Therefore, if that envelope is simply a rectangular function, the bands are convoluted with a sinus cardinal, causing sidelobes. An envelope providing apodization is the Kaiser function,

$$w(x) = \frac{I_0(\beta \sqrt{1 - 4x^2})}{I_0(\beta)} \Pi(x), \quad (51)$$

where  $I_0$  is the modified Bessel function of the first kind and zeroth order,  $\Pi(x)$  is a rectangular function, and  $\beta$  is a parameter that controls the amount of apodization.

The bandwidth of a band,  $\Delta\lambda_i$ , is related to the amplitude of its index profile, while the reflecting power is related to the amplitude of the index profile and the number of periods. According to Bovard [30],

$$\Delta n = 2n_m \sin\left(\frac{\pi}{4} \frac{\Delta\lambda_i}{\lambda_i}\right), \quad (52)$$

$$x = \frac{Q}{\pi \bar{Q}_i |W_0| / \lambda_i}, \quad (53)$$

where  $W_0$  is the continuous component of the Fourier transform of  $w(x)$ .

OpenFilters' rugate module corrects for the dispersion of the material using the method that Poitras and we have developed [35]. It consists of (1) designing separate single-band rugate filters for each band (2), converting the index profiles to a common wavelength using the dispersion of the material, and (3) multiplying the index profiles.

### 3. Quintic Layers

Graded-index layers can also be used as antireflective layers. Indeed, when the index change is continuous and very slow, there is almost no reflection in a layer. One particularly effective antireflective layer to avoid reflection between two materials of indices  $n_1$  and  $n_2$  is the quintic index profile [36]

$$n(u) = n_1 + (n_2 - n_1)(10u^3 - 15u^4 + 6u^5), \quad (54)$$

where  $u = z/d$ ,  $z$  is the physical depth, and  $d$  is the total physical thickness of the filter.

OpenFilters also offers to scale the thickness optically ( $u = x/2OT$ ) and the index exponentially, giving

$$n(u) = n_1 \exp \left[ \ln \frac{1 + (n_2 - n_1)}{n_1} (10u^3 - 15u^4 + 6u^5) \right], \quad (55)$$

which we have found to give even better results.

### 4. Very Short User Guide

Now that we have shown what OpenFilters offers, we succinctly present how to use it. All the functions of OpenFilters are accessible through a graphical user interface (GUI), which is designed so that the vast majority of the information is presented in a single main window, as shown in Fig. 5.

OpenFilters comes with a bank of materials from Palik [37,38] or characterized in our laboratory. More materials can be added to fit the user's needs by use of the GUI. At this moment, OpenFilters offers three ways to enter the dispersion of the index of refraction: a single complex value for dispersionless materials, a table of complex values as a function of wavelength, or a Cauchy dispersion model with an Urbach absorption tail. The material dispersion formulas are saved in text files.

In OpenFilters, material mixtures are described using the real properties of the materials, instead of mixing laws such as the effective medium approximation. To do so, the user must enter the dispersion of the index of refraction for multiple mixtures, and

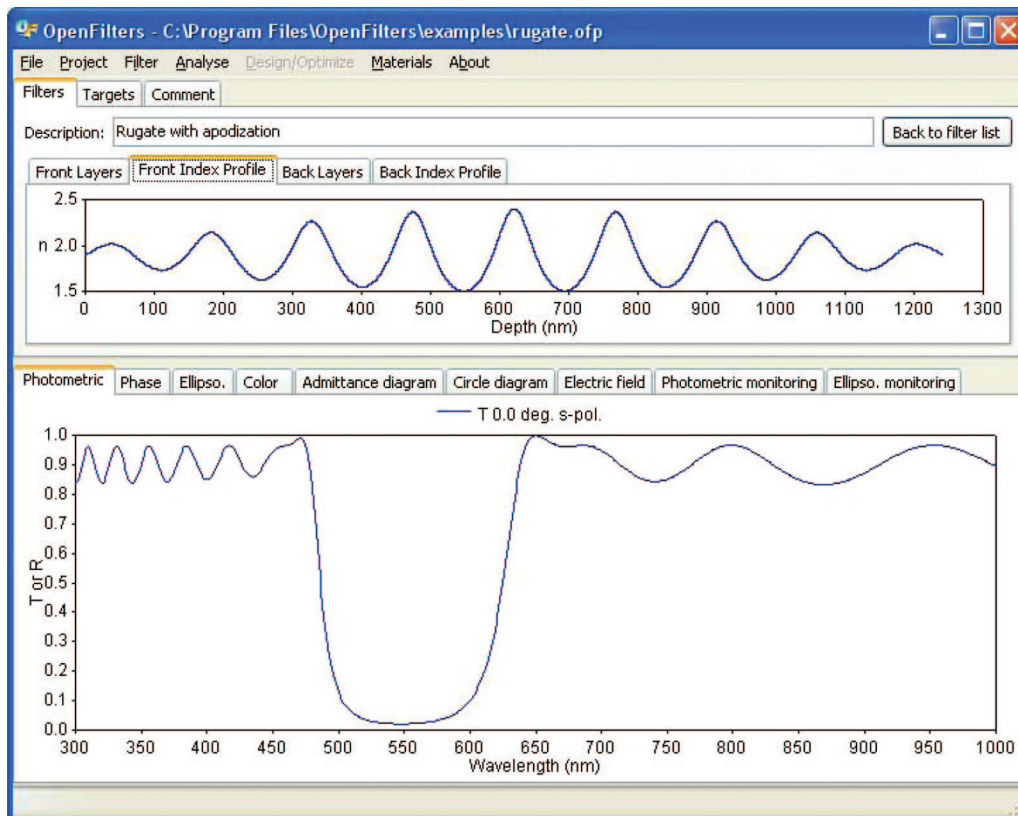


Fig. 5. (Color online) Main window of OpenFilters.



the index of refraction of intermediate mixtures is interpolated using a cubic spline. OpenFilters also provides a mechanism to account for the limited resolution of the index of refraction during the eventual deposition of graded-index filters. When the optical properties are entered, the user must assign a number to every mixture. Then, OpenFilters considers that only intermediate mixtures represented by whole numbers are possible to deposit.

The operations in OpenFilters are centered around projects. A project allows the user to design and compare multiple solutions to the same problem by regrouping a set of filters designed to meet the same set of targets. After the creation of a project, the user can add filters and targets to it and modify or remove them in the order wanted.

When the user adds a new filter to the project, a dialog is presented that makes it necessary for the user to specify the substrate material and thickness, the front and back media, the reference wavelength, the wavelength range, and the graded-index step spacing. The reference wavelength is the wavelength at which the index of refraction of mixtures is defined and is used when calculating the optical thickness of the layers. The step spacing is used in the discretization of the index profile of graded-index filters as explained in Subsection 3.C. The user may choose to discretize graded-index filters according to the *deposition* steps as described in the material file. All these properties may be changed later if necessary.

It is possible to add layers to the filter individually or through the use of a stack formula. When mixtures are used, in addition to the thickness of the layer, the user must specify its index of refraction. Layers can be added, removed, or modified at the will of the user.

OpenFilters comes with modules to design rugate and quintic filters. The rugate module offers the possibility to design single-band and multiband rugates and to specify the number of periods and the amplitude of the index profile either directly or from the bandwidth and the reflection power of the bands. It also offers the possibility to multiply the index profile by a Kaiser apodization envelope. In the quintic module, the starting and ending indices can be selected manually or automatically matched to the surrounding layers. The index can be linearly or exponentially scaled, while the thickness can be physically or optically scaled.

After a filter has been created, it is possible to calculate many of its properties: transmission, reflection, absorption, phase, GD, GDD, ellipsometric variables  $\Psi$  and  $\Delta$ , color, the circle and admittance diagrams, and the electric field distribution. The color is calculated in XYZ,  $xyY$ ,  $L^*a^*b^*$ , and  $L^*u^*v^*$  color spaces using the CIE1931 or CIE1964 colorimetric observers and one of the 38 illuminants defined by the Commission Internationale de l'Éclairage [9]. OpenFilters can also generate photometric or ellipsometric monitoring curves. When the user asks the software to calculate a property, a dialog is presented

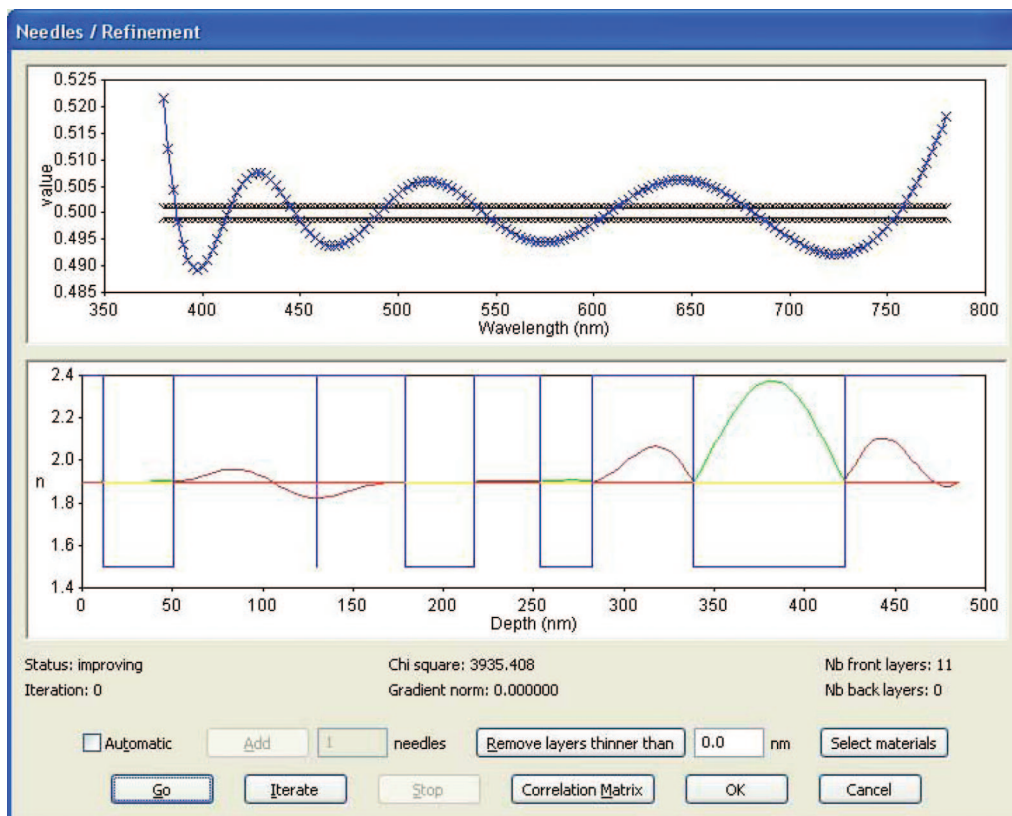


Fig. 6. (Color online) Needle method dialog.

for selection of the conditions at which to calculate the property, such as the angle of incidence and the polarization.

To design, optimize, or synthesize a filter, the user must define targets. OpenFilters offers single wavelength and spectral reflection and transmission targets, and color targets. In the case of spectral targets, the user must set a range of wavelengths and a resolution for the target, and then define the value of the target at some wavelengths. The values between those wavelengths are linearly interpolated. It is possible for the user to specify that the target is an equality or an inequality. For color targets, the user has the choice of the same color spaces, observers, and illuminants already mentioned.

Once targets are set, the user can optimize an existing filter using refinement, synthesize a filter using the needle or step methods, or design one using the Fourier transform method. When the user selects one of these methods, a dialog is presented that allows control of the design, optimization, or synthesis. For example, the needle method dialog, presented in Fig. 6, offers the possibility to select the materials of the needles, to refine the filter, to add a selectable number of needles, or to remove thin layers. The user can also select the automatic mode in which refinement, addition of needles, and removal of thin layers are automatically done until an acceptable solution is found or the addition of more needles is no longer helpful.

Finally, the project can be saved in a text file and reopened later. All the results given by OpenFilters can be exported to text files to allow the user to plot them with a software of his or her choice.

## 5. Under the Hood

If the user is interested in modifying OpenFilters to fit specific needs and to collaborate on its development, it is necessary to understand the structure of its implementation.

OpenFilters is written in Python [39] and C++ using object oriented programming. Major logical elements of the software (materials, projects, filters, targets, design/optimization/synthesis routines) are represented by classes programmed in Python. To speed up the calculations, most of the computationally intensive tasks are performed by two modules implemented in C++: the *abeles* and *moremath* modules. The GUI is programmed using the wxPython toolkit [40].

The *abeles* module is named after Florin Abeles who first presented the characteristic matrix approach to calculate the optical properties of optical coatings [5]. It implements classes to calculate the index of refraction of materials from their dispersion formulas, computes the characteristic matrices of layers, multiplies these matrices, calculates the properties of the filter, and evaluates the derivatives used in refinement, needle, and step methods.

The *moremath* module, named to differentiate it from the standard Python module *math*, implements the Levenberg–Marquardt algorithm routines

to solve linear equation systems using Gauss–Jordan elimination or QR decomposition, interpolation methods (mainly the cubic spline), and a few more tools. The Levenberg–Marquardt code is similar to MINPACK [41] but allows the use of inequality targets and the bounding of the fitted parameters (to avoid negative thicknesses and bound the index of refraction to acceptable values).

Each kind of material is represented by a different class that derives from the material base class. At this moment, there are six derived classes for simple materials and mixtures, and for the three dispersion models previously described. All these classes include a common set of methods to set the dispersion formula of the material and to obtain its index of refraction for a single wavelength or a spectrum. Many of the calculations are delegated to the *abeles* module. More dispersion models can be easily added by simply creating more derived classes without needing to modify other parts of the code.

Projects are instances of the project class. It is a very plain class that simply implements methods to add or remove filters and targets.

All offered kinds of target derive from the target base class. Derived classes are implemented for single wavelength and spectral reflection or transmission, and for color. They include methods to set the target and to interact with optimization methods. If new target classes are added, it will also be necessary to modify the optimization methods accordingly.

The optical filter class implements methods to add and remove homogeneous and graded-index layers, and to calculate the properties of a filter with the help of the *abeles* package. When a property is calculated at a given angle of incidence, the matrices for that angle of incidence are saved. Therefore, it is not necessary to recalculate them when determining another property at the same angle of incidence.

The classes implementing design, optimization, and synthesis methods derive from the optimization base class. The refinement and the Fourier transform method classes are directly derived from the optimization class. The refinement class is further derived to create the needle and step method classes. Optimization methods use the *abeles* and *moremath* modules intensively during the calculations.

All the files in which materials, illuminants, observers, or projects are saved are text files in an easily understandable format.

The GUI is clearly separated from the rest of the implementation to allow creation of another user interface if necessary. OpenFilters is somewhat multithreaded: computationally intensive tasks are done in a separate thread to preserve the GUI responsiveness, and it is possible to stop most of these operations. However, at this moment, these operations are done in a single thread.

Finally, *rugate* and *quintic* modules are self-contained. This implementation was selected to allow the creation of modules without having to modify any other part of the code. The modules are loaded when

OpenFilters starts, so it is possible to add modules without recompiling it.

## 6. Conclusions and Future Plans

We have developed OpenFilters, an open-source program for the design of optical filters. It offers new features such as the step method and the consideration of dispersion in the Fourier transform method as well as popular methods such as refinement and the needle method. It was recently used in the OIC 2007 design contest and provided a result comparable to the winning design [42].

We plan to continue the development of OpenFilters and would be pleased to find collaborators. Following the open-source philosophy, we therefore invite you not only to use OpenFilters but also to modify it to fit your needs and to collaborate on its development. If you send us the modifications, and we believe they are of interest to the community, we will be glad to include them in the distribution of OpenFilters.

In the near future we plan to include gradual evolution [43–45], design cleaning [45], and preproduction analysis [46–49] to OpenFilters. In the longer term, we would like to extend the OpenFilters project to develop a reverse-engineering software.

We would also like to build a large database of materials' refractive index. To help us, we invite you to send us the index of refraction of the materials that you measure as well as a description of the fabrication condition and measurement setup.

We acknowledge the financial support of the National Sciences and Engineering Research Council of Canada (NSERC), and we thank our colleagues of the FCSEL, particularly Marie-Maude de Denus Baillargeon and Bill Baloukas, who tested early versions of OpenFilters, for their feedback. Stéphane Larouche also thanks the NSERC for its scholarship.

A preliminary version of this work was presented at the Optical Society of America's Tenth Topical Meeting on Optical Interference Coatings held in Tucson, Arizona, USA, 3–8 June 2007 [50].

## References

1. P. Baumeister, "Design of multilayer filters by successive approximations," *J. Opt. Soc. Am.* **48**, 955–958 (1958).
2. J. A. Dobrowolski, "The impact of computers on the design and manufacture of optical multilayer coatings during the past 50 years," in *50th Annual Technical Conference Proceedings* (Society of Vacuum Coaters, 2008), pp. 289–301.
3. J. Kruschwitz, "Software tools speed optical thin-film design," *Laser Focus World* **39**, 153–157 (2003).
4. Free Software Foundation, "GNU general public license," <http://fsf.org/>.
5. F. Abelès, "Recherches sur la propagation des ondes électromagnétiques sinusoïdales dans les milieux stratifiés. Application aux couches minces," *Ann. Phys. (Paris)* **5**, 596–640, 706–782 (1950).
6. H. A. Macleod, *Thin-Film Optical Filters*, 3rd ed. (Institute of Physics Publishing, 2001).
7. Sh. A. Furman and A. V. Tikhonravov, *Basics of Optics of Multilayer Systems* (Éditions Frontières, 1992).
8. R. H. Muller, "Definitions and conventions in ellipsometry," *Surf. Sci.* **16**, 14–33 (1969).
9. CIE, *Colorimetry*, 3rd ed. CIE 15:2004 (Commission Internationale de l'Éclairage, 2004).
10. P. J. Leurgans, "The impedance concept in thin film optics," *J. Opt. Soc. Am.* **41**, 714–717 (1951).
11. J. H. Apfel, "Graphics in optical coating design," *Appl. Opt.* **11**, 1303–1312 (1972).
12. V. V. Veremei and I. M. Minkov, "Distribution of light intensity within a dielectric mirror," *Opt. Spectrosc. (USSR)* **33**, 1175–1178 (1972); translation, *Opt. Spectrosc.* **33**, 640–641 (1972).
13. Y. H. Yang and J. R. Abelson, "Spectroscopic ellipsometry of thin films on transparent substrates: a formalism for data interpretation," *J. Vac. Sci. Technol. A* **13**, 1145–1149 (1995).
14. P. Baumeister, "Simulation of a rugate filter via a stepped-index dielectric multilayer," *Appl. Opt.* **25**, 2644–2645 (1986).
15. J. J. Moré, "The Levenberg-Marquardt algorithm: implementation and theory," *Lect. Notes Math.* **630**, 105–116 (1978).
16. C. J. van der Laan and H. J. Frankena, "Fast computation method for derivatives of multilayer stack reflectance," *Appl. Opt.* **17**, 538–541 (1978).
17. P. G. Verly, "Modified needle method with simultaneous thickness and refractive-index refinement for the synthesis of inhomogeneous and multilayer optical thin films," *Appl. Opt.* **40**, 5718–5725 (2001).
18. S. Larouche and L. Martinu, "Optical filters with constant optical thickness and refined refractive indices," in *Optical Interference Coatings on CD-ROM* (Optical Society of America, 2007), p. TuD8.
19. A. V. Tikhonravov, "A method of synthesis of optical coverings which uses the necessary optimality conditions," *Vestn. Mosk. Univ. Fiz. Astronomiya* **37**, 91–93 (1982); translation, *Mosc. Univ. Phys. Bull.* **37**, 108–110 (1982).
20. A. V. Tikhonravov, M. K. Trubetskov, and G. W. DeBell, "Application of the needle optimization technique to the design of optical coatings," *Appl. Opt.* **35**, 5493–5508 (1996).
21. B. T. Sullivan and J. A. Dobrowolski, "Implementation of a numerical needle method for thin-film design," *Appl. Opt.* **35**, 5484–5492 (1996).
22. S. Larouche and L. Martinu, "A new step method for the synthesis of optical filters with arbitrary indices," in *49th Annual Technical Conference Proceedings* (Society of Vacuum Coaters, 2006), pp. 305–308.
23. E. Delano, "Fourier synthesis of multilayer filters," *J. Opt. Soc. Am.* **57**, 1529–1533 (1967).
24. L. Sossi, "A method for the synthesis of multilayer dielectric interference coatings," *Eesti NSV Teaduste Akadeemia Toimetised Füüsika, Matemaatika* **23**, 229–237 (1974). English translation available from the Translation Services of the Canada Institute for Scientific and Technical Information (CISTI).
25. J. A. Dobrowolski and D. Lowe, "Optical thin film synthesis program based on the use of Fourier transforms," *Appl. Opt.* **17**, 3039–3050 (1978).
26. B. G. Bovard, "Rugate filter theory: an overview," *Appl. Opt.* **32**, 5427–5442 (1993).
27. L. Sossi, "On the theory of the synthesis of multilayer dielectric light filters," *Eesti NSV Teaduste Akadeemia Toimetised Füüsika, Matemaatika* **25**, 171–176 (1976). English translation available from the Translation Services of the Canada Institute for Scientific and Technical Information (CISTI).
28. P. G. Verly, J. A. Dobrowolski, W. J. Wild, and R. L. Burton, "Synthesis of high rejection filters with the Fourier transform method," *Appl. Opt.* **28**, 2864–2875 (1989).
29. B. G. Bovard, "Derivation of a matrix describing a rugate dielectric thin film," *Appl. Opt.* **27**, 1998–2005 (1988).

30. B. G. Bovard, "Fourier transform technique applied to quarterwave optical coatings," *Appl. Opt.* **27**, 3062–3063 (1988).
31. R. Szipöcs and A. Köházi-Kis, "Theory and design of chirped dielectric laser mirrors," *Appl. Phys. B* **65**, 115–135 (1997).
32. H. Chang, S.-S. Lee, M. R. Chul, and S. Lim, "Inhomogeneous optical filter design with the use of a Riccati equation," *Micro-wave Opt. Technol. Lett.* **22**, 140–144 (1999).
33. P. G. Verly and J. A. Dobrowolski, "Iterative correction process for optical thin film synthesis with the Fourier transform method," *Appl. Opt.* **29**, 3672–3684 (1990).
34. S. Larouche and L. Martinu, "Dispersion implementation in optical filter design by the Fourier transform method using correction factors," *Appl. Opt.* **46**, 7436–7441 (2007).
35. D. Poitras, S. Larouche, and L. Martinu, "Design and plasma deposition of dispersion-corrected multiband rugate filters," *Appl. Opt.* **41**, 5249–5255 (2002).
36. W. H. Southwell, "Gradient-index antireflection coatings," *Opt. Lett.* **8**, 584–586 (1983).
37. E. D. Palik, ed., *Handbook of Optical Constants of Solids* (Academic, 1985).
38. E. D. Palik, ed., *Handbook of Optical Constants of Solids II* (Academic, 1991).
39. "Python," <http://www.python.org/>.
40. "wxpython," <http://wxpython.org/>.
41. J. Moré, B. Garbow, and K. Hillstom, "MINPACK," <http://www.netlib.org/minpack>.
42. M. Tilsch and K. Hendrix, "Optical Interference Coatings design contest 2007: triple bandpass and nonpolarizing beam splitter," *Appl. Opt.* **47**, C55–C69 (2008).
43. J. A. Dobrowolski, "Completely automatic synthesis of optical thin film systems," *Appl. Opt.* **4**, 937–946 (1965).
44. J. A. Dobrowolski, "Versatile computer program for absorbing optical thin film systems," *Appl. Opt.* **20**, 74–81 (1981).
45. A. V. Tikhonravov, M. K. Trubetskov, and G. W. DeBell, "Optical coating design approaches based on the needle optimization technique," *Appl. Opt.* **46**, 704–710 (2007).
46. P. W. Baumeister, "Methods of altering the characteristics of a multilayer stack," *J. Opt. Soc. Am.* **52**, 1149–1152 (1962).
47. W. P. Thoeni, "Deposition of optical coatings: process control and automation," *Thin Solid Films* **88**, 385–397 (1982).
48. H. Zorc, "Optimum multilayer design selection in relation to production errors," *Vacuum* **37**, 101–102 (1987).
49. A. V. Tikhonravov and M. K. Trubetskov, "Computational manufacturing as a bridge between design and production," *Appl. Opt.* **44**, 6877–6884 (2005).
50. S. Larouche and L. Martinu, "OpenFilters: an open source software for the design and optimization of optical coatings," in *Optical Interference Coatings on CD-ROM* (Optical Society of America, 2007), p. WB6.