Solar performances Optimization Code for optical behavior of multilayers stack in Python: SolPOC

Antoine Grosjean[1], Pauline Bennet[2], Thalita Drumond[1], Amine Mahammou[3,4], Denis Langevin[2], Audrey Soum-Glaude[5], Antoine Moreau[2]

[1] EPF Ecole d'ingénieur-e-s, 21 boulevard Berthelot, 34000 Montpellier, France
[2] Université Clermont Auvergne, Clermont Auvergne INP, CNRS, Institut Pascal, F-63000 Clermont-Ferrand, France
[3] PROMES-CNRS, UPR 8521, Rambla de la Thermodynamique, Tecnosud, 66100 Perpignan, France
[4] Université de Perpignan Via Domitia, 52 Av. Paul Alduy, 66820 Perpignan, France
[5] PROMES-CNRS, UPR 8521, 7 rue du Four Solaire, 66120 Font-Romeu-Odeillo-Via, France

Abstract:

SolPOC, Solar Performances Optimization Code, is a simple and fast code running under Python 3.9. This code is designed to solve Maxwell's equations in a multilayered thin film structures used in solar energy management systems. The code is specifically designed for research in coatings, thin film deposition, and materials research for solar energy application, like coatings for solar thermal, Concentrated Solar Power (CSP), photovoltaic (PV), low-e coating, building or even glasses. The code uses a stable method to quickly calculate reflectivity, transmissivity, and absorptivity from a stack of thin films over a full solar spectrum. SolPOC comes with several optimization methods, a multiprocessing pool, and a comprehensive database of refractive indices of real materials. The code has already produced major scientific advances in research on coatings for solar thermal systems and can be easily used in other domain, as coatings for building, optical, human vision etc. In the end, the code is simple to use for no-coder users thanks to main script and automatically save important results. The whole project is free and available on GitHub with complete documentations and tutorials.

Keywords: SolPOC, open-source, thin layers stack, multilayers coatings, global optimization, modeling, solar energy, soler thermal energy, CSP, photovoltaic, optics

1. OVERVIEW

1.1. Introduction

Solar technologies rely on the collection of the solar irradiation to generate electricity with photovoltaic cells (PV) or generate heat by heating a fluid in Solar Thermal Systems (STS) for domestic, residential, and industrial purposes. Also, free collection of solar energy is often beneficial in building through windows, for direct heat consumption reduction. The variability of use for solar energy is as vast than electricity or heat needs. In some specific cases, if solar irradiance is concentrated using mirrors, this heat can in turn be used to produce electricity via a turbine in Concentrated Solar Power (CSP).

All these technologies call for optically efficient components with complex and sometimes conflicting optical behaviors. In particular, the solar receiver should be highly absorbing in the solar range (0.28 – 4 µm) to harvest as much solar radiation as possible, but also lowly emissive in the infrared range (1 – 50 µm) to limit radiative thermal losses [1,2]. This spectral selectivity can be achieved using multilayered coating architectures, associating lowly emissive (e.g. metals) and highly absorptive materials (e.g. dielectric/ metal/dielectric multilayers or metal-ceramic composites), that need to be optically designed and optimized in terms of layer thicknesses and compositions, to guarantee their high optical performance. A judicious preselection of materials is also paramount, as the solar receivers should also be resistant to harsh operating conditions such as high temperatures, high solar irradiation, oxidant and erosive atmospheres and high thermomechanical stress for long durations, while remaining optically efficient. Moreover, new technologies like hybridization of photovoltaic (PV) and concentrated solar thermal (CST) technologies need specific coating with many thin layers [3,4]. In need a solution for PV/CST hybridization is the “PV mirror” configuration, where PV cells are installed on concentrators to produce electricity, and thermal absorbers are placed at their focus to produce heat (that can be used as such or converted into electricity). Such “PV mirror” require advanced coatings, with many thin layers [4–7]. Other solar applications like antireflective coating for PV, radiative cooling, coating, self-cleaning coatings, dielectric mirrors, or selective coating require thin layers stacks with advanced material and a growing number of thin layers.

In summary, the collect of solar energy requires highly quality surface properties. Theses latter are provided by thin layers stacks, where the thin layers numbers, the materials used, the substrate, the geometry or the deposition technique are subject to research. As solar technology advances, coatings are becoming more sophisticated, with co-functionalities and a becoming an increasingly crucial topic. Given the numbers of challenges involved, the solar research community requires a readily accessible software solution for i) descript the optical properties of large numbers of thin layers stack in the solar domain ii) include data from new materials iii) optimize these stacks to accommodate a wide range of thin-layer combinations and functionalities. This ideal software should be free, highly versatile, easily tunable and written in a widely recognized programming language, meeting the diverse needs of the solar energy research community.

1.2. Modeling and optimization of thin layers stack

Modelization and optimization a thin layers stack is showing interest for an inverse problem as different optical structures can show similar optical behavior. With the innovation of computers, it is now possible to solve these problems using numerical optimization methods, by defining a cost function linked to the desired optical response. This function is then minimized or maximized using algorithms. The first computer program to automate optical filter design was published by Dobrowolski and Lowe, since 1978 [8]. Though the years, different codes and optimisation methods have been proposed, sometimes specific to multilayered optical stack, as the Needle by Tikhonravov et al [9]. Global optimization methods, such as Simmulated Annealing, Particules Swarn Optimization (PSO), different evolutionnary algorithms or more recently supervided treaning are frenquently used [3,10–12].

Table 1 : Softwares for optical thin film coating

Software	Commercial statut	Open source	Versatile & Tunable	GUI	Designed for solar use	Ref
CODE/SCOUT	Commercial	No	No	Yes	No	[13]
Essential MacLeod	Commercial	No	No	Yes	No	[14]
FilmStar	Commercial	No	No	Yes	No	[15]
OpenFilters	Free	Yes	No	Yes	No	[16,17]
OpTaliX	Commercial	No	No	Yes	No	[18]
OptiLayer	Commercial	No	No	Yes	No	[19]
PhaseCODE	Commercial	No	No	Yes	No	[20]
PyMoosh	Free	Yes	Yes	No	No	[21–23]
TFCalc	Commercial	No	No	Yes	No	[24]
TMM-Fast	Free	Yes	Yes	No	No	[25,26]
RP-Coating V4	Commercial	No	No	Yes	No	[27]

Table 1 show a review of the existing softwares for optical thin film coating. Ideally, the solar community need a free software designed for solar use, ready to use and open source for provide the adaptability necessary for the large panel of our solar applications. Although many commercial programs (CODE/SCOUT, Essential MacLeod, FilmStar, OptiLayer and other) show excellent quality, numerous research institutions have to create their proprietary in-house software. The main reason is commercial software are not open source: it's not possible to added new functionality in the code despite they are more friendly to use thank to their Graphic User Interface (GUI). For fill this lack software in-house code are developed to offers the advantage of swift adjustments in response to emerging issues and can be tailored to precisely match the research's unique requirements, whereas commercial programs inherently offer less flexibility for customization. Recently different codes for optical thin film coating are shared freely with the optical and photonics community, as PyMoosh (2023), TMM-Fast (2022) or OpenFilters (2008). They provide key functions, but for solar community need to transform these raw materials into a functional software as adding multiprocessing tools, refractive index data and cost functions (also known as merit functions or objective functions) specific to solar energy systems.

Working in the solar energy coatings domain since 2014 we have, as many, developed our own in-house software thin layers stacks, named SolPOC. The major contributions of this code, compared to existing ones are to work across a wide spectral range (280 nm – 30 µm) and to be ready to use for no-expert people, bringing together the advantages of commercial and open-source code. This makes SolPOC code particularly relevant for research or educational in solar applications. In this paper, we have chosen to make our code SolPOC freely available to anyone, hoping our code can help the community to easily study, model and optimize thin layers stack for solar energy.

1.3. About SolPOC

Solar Performances Optimization Code (SolPOC) is a free, tunable, versatile, and fast software tool designed for research in the field of solar energy coating. The code operates within the Python 3 programming environment.

The current version of the code have the following key features:

1. Quicker and stable calculation of reflectivity, transmissivity, and absorptivity of thin layers stack using a vectorized (using NumPy package) Abélès formalism method [22].
2. Working with a full solar spectral range including infrared (e.g : 280 nm to 30 µm) [2]
3. Use refractive index data of real materials found in peer-reviewed papers [28].
4. Evaluate thin layers stack's solar properties.
5. Use Effective Medium Approximation methods (EMA) to model the optical behavior of material mixtures (dielectric mixtures, metal-dielectric, porous materials) [29].
6. Optimize stack optical performances according to a large panel of cost functions, including cost functions for solar energy systems, building and solar thermal uses.
7. Propose 6 different optimization methods based on evolutionary algorithms, such as PSO or Differential Evolution.
8. Highly quality parallel code, allow us to be working with multiprocessing.
9. Automatically results output (.txt files and .png images) to a folder and propose a simplified user interface, bringing together useful variables in a few lines of code.

2. SOLPOC

SolPOc has been designed for serve as a readily accessible solution in the field of coatings for solar energy and by keep in mind the fundamental principles of Open Scienc. The choice of Python as the programming language plays a pivotal role: Python packages can be readily located and installed through the PyPI repository. Python is also a very popular language, frequently lauded for its capacity to enabling seamless interaction with other programming languages and with a good capacity to decrease execution time. Furthermore, platforms like GitHub, which also function as social networks to some extent, coupled with the GNU General Public License, enable us to enhance code accessibility and reusability to the maximum extent possible.

SolPOC has been thoughtfully crafted to be user-friendly, especially for researchers, experimenters working with coatings and thin layers, as well as non-programming users. The software's architectural design includes a simplified user interface, condensing essential variables into concise lines of code and emulating a graphical user interface. This approach empowers non-expert users to quickly grasp the software's functionality and employ it effectively. Even if Python is an object-oriented language, we intentionally strongly minimized their utilization to facilitate the code comprehension. In addition, we have provided extensive documentation and different tutorials in Jupyter NoteBook.

2.1. Materials Data base

SolPOC provide large database of refractive indices for all types of materials, particularly those suitable for solar energy applications. The actual database includes more of 150 different materials, including metal, dielectric, conductive oxides, or semiconductors. This selection derived from a critical review of the scientific literature (refractiveindex.info web site) and technical catalogs (e.g., technical catalog from the glass industry) made by the authors [28,30]. Similar data set is used in commercial software [19]. We have preselected the most relevant data as refractive index measured on thin films rather than bulk materials, real measurements rather than modeling and studies with numerous measurement points to minimize reliance on interpolation and extrapolation made by the code. Also, most studies have been selected because they cover a large spectral domain (solar range 280 – 2500 nm, and often the IR range) necessary for the different calculation as the solar properties or the radiative losses calculation [31,32]. The complete database is available in folder "Materials", where each material is descripted by text files. We have choice text files to make the addition of new materials very easy and the User Guide descript how to include new materials with text files, making the database easily tunable.

Composite layers, such as cermets (mixture of dielectrics and metal) or porous materials (such as mixture of air and dielectric) are often used in coatings design, and especially in coatings for solar energy [33,34]. The latter provide interesting optical properties, like low refractive index (e.g porous SiO2) or solar absorptance behavior (e.g W-Al2O3) [34–37]. To calculate the spectral complexe refractive index of these composite materials, Effective Medium Approximation (EMA) methods have been proposed by Bruggeman since 1935 [29]. The most famous one are Bruggeman, Maxwell-Garnett, Yoldas mixing rules, or Landau-Lifshitz-Looyenga theories [29,38,39]. Bruggeman theory was selected early for the code, as already discussed in several papers [34,40–42].

2.2. Optical properties calculation

The key function in our code is to evaluate the optical behavior of a multilayered structure by solving Maxwell's equations with the best ratio between accuracy and rapidity. Since the first Lord Rayleigh's contributions to predict the reflectance of a multilayers stack, different calculation methods have been proposed to provide for scientists and then industrials the required tools. If the most common formalism is based on the Transfer Matrix Method (TMM), other formalisms are available with their own advantages and inconvenient such as the Scattering Matrix, the Abélès formalism (with is different than TMM), the Admittance method or more recently an adaptation called the Direchlet-to-Neunmann maps. A complete deep-review and comparison of these different formalisms has been provided recently by D. Langevin et al. [23].

Based on their work, we have chosen the Abélès formalism for SolPOC as the best compromise between time and stability instead of TMM [33]. Moreover, similarly as A.Luce et al with TMM-Fast, the Abélès formalism allow us to use the package NumPy library witch strongly reduce the calculation time per CPU [25]. We have looking for the highest efficiency for the NumPy implementation by optimizing the code structure. The characteristic matrices M calculation is voluntary only 3D dimension, to be of shape $[2, 2 \cdot L, \lambda]$ where L the number of thin layers and λ the wavelengths for avoid RAM use abuse. Finally, the code users must know that Abélès formalism involved a thin layer's number limit around 150 layers to avoid instability. More details are present in the User Guid and in the literature [23,33]. Figure 1 illustrate how the stack optical properties are calculated from refractive index using Abélès formalism.

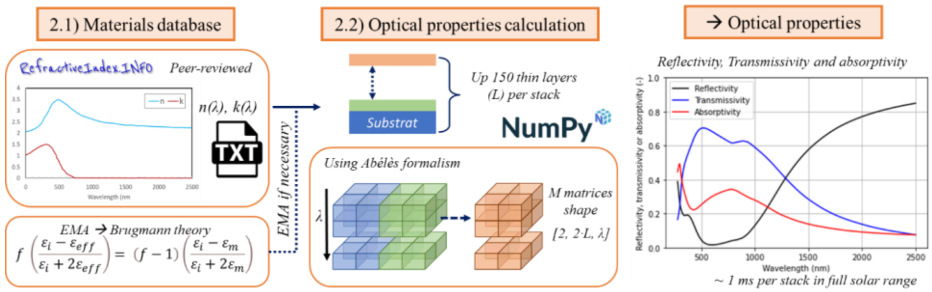

Figure 1 : SolPOC use refractive index from peer reviewed studies, added, if necessary, with EMA theory for created a thin layer stack. The optical properties are calculated using a Abélès formalism, using NumPy package for reduce time calculation.

2.3. Optimization process

The main aim of the Solar Performances Optimization code (SolPOC) is to design highly effective thin layer stack for solar energy systems. For this purpose, the code can optimize the layers thicknesses and with their compositions in the case of composites layers (cermet, porous or other) or by assuming than $dn/dy = 0$.

2.3.1. Cost function

The solar performance, as example solar reflectance for solar mirror, heliothermal effiency for thermal selective absorber, solar absorptance from a PV cell or visible solar transmittance for vacuum tube are solar optical properties described and quantify the thin layers stack used in solar energy [34–36]. In the actual version, 12 different functions, all described in the User Guide, are present in SolPOC for design classic coatings used in PV, CSP/CST or building. The latters are named cost function, as they can be used during an optimization process for evaluate a thin layer's stack. All function must return a value include between 0 and 1.

As a code designed for solar performance optimization, the ASTM G173-03 references solar spectra (also known as the AM 1.5 solar spectra) is include with the code [32,37]. The three different solar spectra: i) the Direct and Circumsolar (DC), ii) the Global Tilt (GT) and iii) the extraterrestrial is present as text files in and can be easy modified by the users. If necessary new solar spectra can be computed from SMARTS or Py-SMARTS and added in the code [38–40]. Different value, as example the normalized relative spectral distribution for the calculation of visible solar transmittance, are also include [35]. Our code is versatile and can easily incorporate different spectral distribution as example, but not limited, the Solar Material Protection Factor (SMPF), the Solar Skin Protection Factor or Color Rendering Factor (CRF) [41]. Normalized spectral responsivity for exemplary PV cells are also present as well than thermal absorber from our previous studies [36,42].

2.3.2. Optimization algorithms

For the stack optimization, according to a cost function/solar performances, a total of 6 different optimization methods are currently available in SolPOC and describe in the User Guide. We have implemented one in house Genetic Algorithm (GA) method, *Differential Evolution* (DE), *Particles Swarm Optimization* (PSO), and simulated annealing method and the *(1+1)-ES* algorithme [43,44]. For comparaison we have also included the optimisation algorithm named *strangle* used in the previous versions of the software on Scilab (named COPS), already published in our previous work [34,36,45]. Despite it's simplicity, it shows good results with simple stack structure. The Needle method, introduced by Tikhonravov et Trubetskov, often used in the multilayer community is not yet present in the code [9,46]. The objective of a global optimization algorithm is to explore large parameter space in order to find, if possible, the global optimum of a function. The Needle technique, by adding layers with the iteration steps, constantly extend the size of the parameter space. At each step of the process, a local optimum is found, then a new layer is added and the search for a local optimum is repeated. The nature of the inverse design method provided by Needle therefore largely differ from the global optimization methods cited above. In consequence, the implementation of a Needle method in SolPOC is let for future work. *Particle Swarm Optimization* have been used since 1995 for thin layer coating and show correct performances [12]. *Bennet et al* have show in their work that the gradient free method *Differential Evolution* can reach hight quality results, even superior, than Needle [47,48].

SolPOC provides six optimization methods, but in reality there are many more of them. Researchers of the optimization community often compare how these methods perform, although they usually test them on problems unrelated to solar energy [49–53]. For instance, there's a theorem called the "No Free Lunch Theorem in Optimization," which has been around since 1997. It basically says that an optimization method that works well for one specific problem may not work as well for another [54].The current version of *Differential Evolution* presented works well for optimizing thin coatings, as proven by our numerous simulations conducted with the code and by specific benchmarks for multilayered photonic structures[48]. Users are however encouraged to test different optimization methods for their specific problems, for instance by using benchmark platforms such as Nevergrad [55–57].

2.3.3. Regularity as optimality indicator

The evolutionary algorithms utilized in SolPOC are inherently non-deterministic. Two runs of the same algorithm may not converge to the same optimum, resulting in two different stacks of thin layers with distinct thicknesses. Initial efforts focus on ensuring convergence in each run, tracked automatically by the code through the cost function's optimization process. However, this alone does not guarantee the global optimum and may still lead to local optima.To address this, it is crucial to run the same optimization multiple times, each with different starting points, and then compare the solutions obtained. This practice allows for a comprehensive examination of the algorithm's reliability and facilitates comparisons between different algorithms or settings. "Consistency curves" are generated to estimate the confidence in an algorithm or make comparisons. For each run, the cost function's value at the end of the optimization is examined, and the consistency across multiple launches is assessed. Ideally, the cost function values are close or identical, indicating that the same optimum is consistently identified despite different starting

Commenté [AG1]: Les algorithmes évolutionnaires utilisés dans SolPOC sont intrinsèquement non déterministes. Deux lancements du même algorithme peuvent ne pas conduire à un même optimum, c'est à dire dans notre cas à deux empilements de couches mince avec des épaisseurs différents. Il faut dans un premier temps s'assurer que la convergence soit atteinte à chaque lancement. Pour cela le code traces automatiquement la valeur de la fonction de coût durant le processus d'optimisation. Mais cela ne garantit pas que l'optimum identifié soit global et évite bien les optimums locaux. Pour s'en assurer il est nécessaire de lancers plusieurs fois une même optimisation avec plusieurs points de départ différents puis de comparer les solutions obtenues entre elles. Pour effectuer facilement cette étude il est possible de représenter des "consistency curves" pour estimer la confiance que l'on a dans un algorithme ou pour effectuer des comparaisons entre différents algorithme ou différents paramétrages. Pour chaque lancement, on regarde la valeur de la fonction de coût obtenu à la fin de l'optimisation, puit on étudie si toutes les valeurs sont similaire pour plusieurs lancements. Idéalement les valeurs des fonctions de coût sont proches voir identique : cela signifie que malgré des points de départ différents le même optimum est systématiquement identifié. Si des valeurs différentes sont obtenue (malgré que chaque optimisation à convergée) cela signifie que l'algorithme identifie des minima locaux, et que l'on peut difficilement avoir confiance dans la meilleure solution obtenue. Il est alors nécessaire de relancer plusieurs fois l'algorithme d'optimisation pour identifier une meilleure solution.

Bien que cette démarche semble coûteuse en temps de calcul, nous recommandons de la mettre en place systématiquement. Notre expérience montre que les fonctions coût solaire sont riche en optimum locaux, notamment pour le solaire thermique. Pour aider la recherche SolPOC trace automatiquement la consistency curve avec les courbes de convergences tout en étant spécifiquement conçus via le multiprocessing pour avoir plusieurs lancements independent simultanément. Mettre en place cette démarche est donc transparent pour l'utilisateur et avec un faible impact sur le temps de calcul total.

points. If varied values are obtained (even with each optimization converging), it suggests that the algorithm identifies local minima, raising concerns about the confidence in the best solution. In such cases, restarting the optimization algorithm multiple times becomes necessary to identify a more reliable solution.

Despite the potential increase in calculation time, we strongly recommend systematically implementing this approach. Our experience indicates that solar cost functions often involve rich landscapes of local optima, especially in solar thermal applications. To streamline this process, SolPOC automatically generates the consistency curve alongside convergence curves. This is achieved using multiprocessing to execute several independent launches concurrently, ensuring transparency for the user and minimizing the impact on the overall calculation time.

2.3.4. Multiprocessing speed-up : Amdalh's law

SolPOC is engineered for fast calculation, using the multiprocessing library in Python. This strategic implementation significantly reduces calculation times by leverages the capabilities of modern processor architectures, effectively distinguishing our software from existing solutions. To illustrate the performance, tests were conducted using a simple cost function (6-period Bragg mirror over the full solar spectrum) on a server equipped with 2 Intel Xeon Gold 5220r CPU, for a total of 48 cores as show in Figure 2. The results demonstrated the high quality of the code aligning with Amdahl's Law, with an estimated parallelizable portion of the code reaching (noted *p*) ≈ 97.7 % [58]. This estimation was calculated through a fitting process (R2: 0.9969). This speedup is coupled with the benefit of NumPy package used for writing the Abélès formalism, also us to strongly decrease the calculation time.

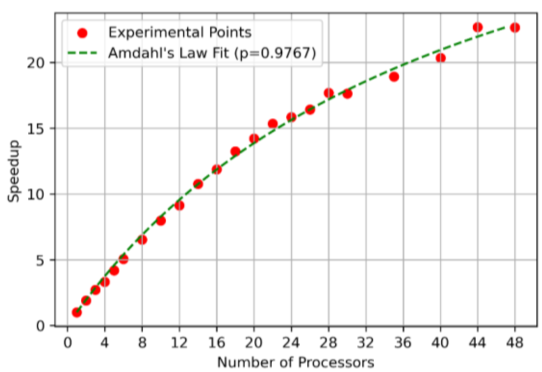

Figure 2 : Amdawl's law for SolPOC (v1.0.0) on 2 Intel Xeon Gold 5220r. If 1 run on 1 CPU takes approximately 36s of calculation time (example), 48 parallelized runs on 48 CPUs take only 77s.

2.4. Quality control

The initial version of SolPOC (named COPS) was developed using Scilab during the Ph.D thesis of the main author, successfully defended on March 7th, 2018 at PROMES CNRS (Perpignan, 66, France) [59]. Between 2018 and 2023, the code remained in active use by the author and the PROMES – CNRS laboratory in France. Its effectiveness and user-friendly interface contributed to its widespread adoption within local research teams. The code played a pivotal role in numerus scientific publications concerning antireflective coating, selective coating, and dielectric mirror [34,36,39,45] and two book chapters [2,60]. During the same period SolPOC served as a valuable tool in various Ph.D theses conducted at the PROMES CNRS laboratory [61,62]. Based on the positive feedback and the code evident utility, the decision in January 2023 to migrate the code to Python, introduce new functionalities, and release it as open-source software. This led to the current version of SolPOC (v1.0.0).

2.5. Overview

Figure 3 provides an overview of the optimization process in SolPOC, complementing the calculation of optical properties for a thin film stack as depicted in Figure 1. Utilizing the optical properties (reflectivity, transmissivity, and absorptivity) of a stack of thin films, the code uses one of the cost function presents in the code, such as solar reflectance in example. If necessary, solar spectra or other properties such as the human eye sensitivity or PV cells spectral efficiency in are directly embedded. The code then optimizes the stack using one of six available optimization methods. The operation is repeated multiple times, facilitated by a parallelizable code, enabling the identification of an optimized stack and to ensure confidence in the optimization results.

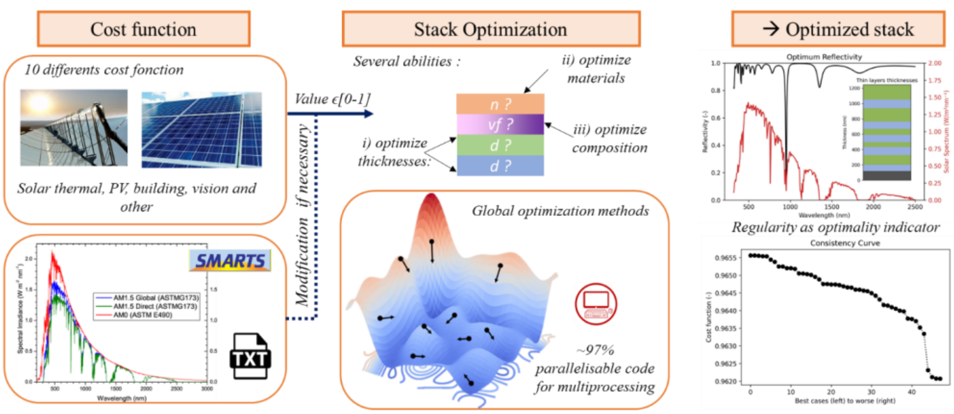

Figure 3: SolPOC offers a wide range of cost functions suitable for various applications including solar energy, buildings, vision and more. These cost functions can be optimized using various global optimization methods. Thank to multiprocessing, Consistency Curve are easy to manage for ensure the optimization quality.

3. EXAMPLE OF USE

We propose a series of optimization findings obtained with SolPOC and their corresponding interpretations. We optimize the various thin film coating applied to a paracolic trough collector, used in CSP. The aim of this collector is to concentrate the entire solar spectrum (280-2500 nm) through the utilization of solar mirror showing high solar reflectance. This concentrated solar energy is directed onto a vacuum thermal absorber, facilitating the production of high-temperature heat (350°C) for electricity generation. The collector configuration encompasses three critical components: i) a reflective coating for the solar mirror, ii) an anti-reflective coating on the vacuum tube with high solar transmittance and iii) a spectrally selective coating into the thermal absorber which exhibits high heliothermal efficiency (high solar absorptance with low radiative thermal losses). Figure 4 illustrates the diverse coatings employed in the system their associated desired optical properties.

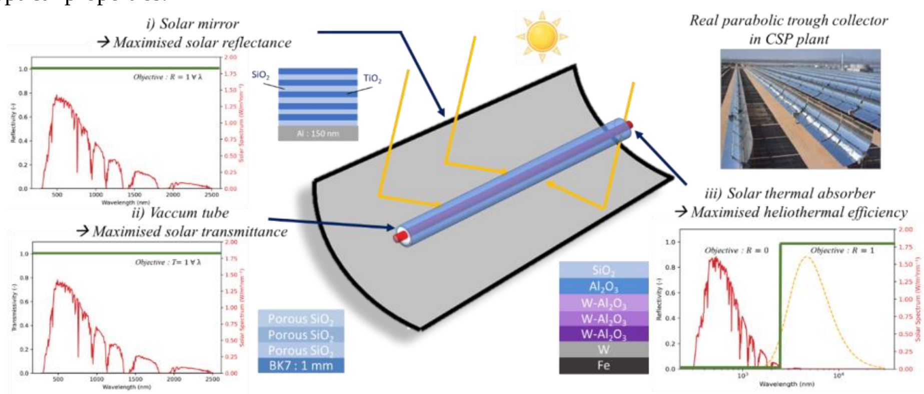

Figure 4 : Illustration of SolPOC capabilities for optimized all coatings utilized in a solar collector for in CSP plants: i) reflective coating, ii) antireflective coating and iii) spectrally selective coating.

To demonstrate the robust performance of our code, we deliberately selected sophisticated coatings characterized by a substantial number of thin layers. These intricate stacks have been meticulously chosen to serve as representative models for the cutting-edge solar coatings that are currently being developed within laboratory settings. Industrial solar mirrors are constructed using a single thin layer of reflective metal, often silver. In our demonstration, we propose enhancing the aluminum reflectance by adding 5 bilayers of SiO_2/TiO_2 on a top of an aluminum layer [34]. Regarding antireflective coatings, we have transitioned from using a single porous SiO_2 layer to a more effective three-layered design, but much more difficult to optimized [45]. Finally, the selective coating plays a crucial role in achieving efficient solar-to-thermal conversion. These coatings typically consist of three layers, including a metallic layer, a cermet layer, and dielectric layers applied to a metallic substrate. An example of conventional design is $Fe/W/W\text{-}Al_2O_3/Al_2O_3$ stack [36]. In our show case, we present a more complex version featuring three cermet layers and two antireflective coatings, further enhancing the heliothermal efficiency (*rH*).

The first column of Table 2 provides an overview of typical performances of industrial solar components, derived from System Advisor Model (SAM) by the NREL[63]. The second column give the theoretical results obtained with SolPOC, showing that the using SolPOC yielding high-quality results with superior solar performance compared to existing products. This underscores the software capability to manage more intricate layer stacks than those currently existing. Despite the substantial number of thin layers involved, the software can effectively handle them and deliver optimal solutions for all selected scenarios. Comprehensive result files are accessible on GitHub (docs/examples), facilitating straightforward result replication. Other examples not related to solar energy (Bragg mirror, antireflective coating for human eye) are also available.

Table 2 : Industrial values from SAM Software [63] *and theoretical example to illustrate optimization capabilities.*

	Industrial solar components		Optimization examples	
Surface	Number of thin layers	Typical value	Optimization parameters	SolPOC results
Solar mirror	1 metallic layer	R_S: 0.935	11 layers thickness	R_S: 0.966
Vacuum tube	1 porous layer	T_S: 0.964	3 layers thickness + 3 porosity rates	T_S: 0.994
Thermal absorber	3 layers with 1 cermet	A_S: 0.963 E(300°C): 0.08 rH(300°C): 0.953	6 layers thicknesses + 3 cermet inclusion rates	A_S: 0.975 E(300°C): 0.074 rH(300°C: 0.966

The optical properties of each case are present in Figure 5. For the solar mirror Al/[SiO2/TiO2]$_5$, SolPOC successfully optimizes thin-film thicknesses to align the reflectance curve with the solar spectrum (ASTM G173-03 DC, in red in the figure), achieving a theoretically expected solar reflectance (R_S) of 96.6%. Additionally, the code optimizes the thickness and porosity of three porous anti-reflective thin films (BK7/[porous SiO_2], ensuring near-perfect transmissivity across the entire solar spectrum and resulting in a solar transmittance (T_S) of 99.6%. In the case of the six-layered selective coating for the thermal absorber, SolPOC effectively optimizes the six thicknesses and three compositions of W-Al_2O_3 cermet to achieve a reflectance characteristic typical of thermal absorbers—low in the solar domain and high in the infrared domain. It for insure high solar absorptance (A_S) and low thermal emittance (E), due to infrared emissivity of a black body (in orange in the figure). This optimization leads to an impressive heliothermal efficiency (rH) value of 96.6%, tailored for an absorber temperature of 300°C.

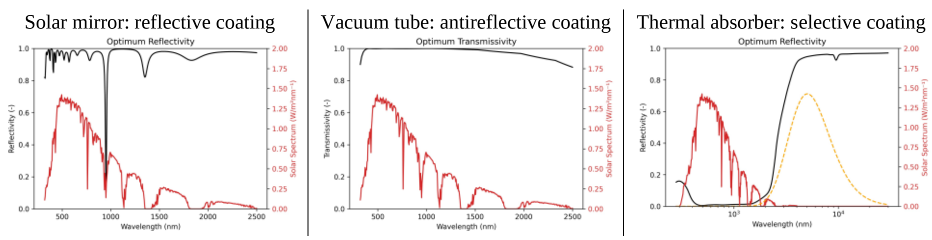

Figure 5 : Reflectivity or transmissivity curve for each case, with the optimized thin layers stack.

Figure 6 illustrates the optimization quality for each case, represented with consistency curve. Each consistency curve represents, for each of the three problems, the value of the cost function (i.e., the result) from 48 distinct optimizations that have converged. For ease of reading, the values are sorted in descending order, with the highest results (we aim to maximize here) to the left of each figure. The criterion here is to look for the presence of a plateau of extrema, indicating that the algorithm has repeatedly found a high-performance or even identical solution. In the case of the solar mirror (Figure 6, left) a small plateau of values at 96.55% can be observed on the left: the global optimum is reached 5 times out of 48. The rest of the curve slopes upwards, indicating that each optimization has produced a different result despite they are converged. This suggests that the problem is rich in local optima, making optimization challenging and indicative of a complex problem. In these instances, it is worthwhile to investigate the parameters of the optimization process, increase the budget, or benchmark optimization algorithm. In the case of the vacuum tube (Figure 6, middle), the problem is solved and optimization is highly qualitative. Each optimization converges to the same optimum, instilling great confidence in the validity of the result. This flat consistency curve is ideal, representing the kind of curve we would seek systematically. For the thermal absorber (Figure 6, right), the optimization produces a similar result (to 0.00025) 44 times over 48. The slightly increasing trend indicates micro-adjustments to the variables, such as thin-film thickness calculations below 0.1 nm. The presence of a few non-optimal values on the right-hand side of the graph indicates some local optima, underscoring the importance of running an optimization problem several times.

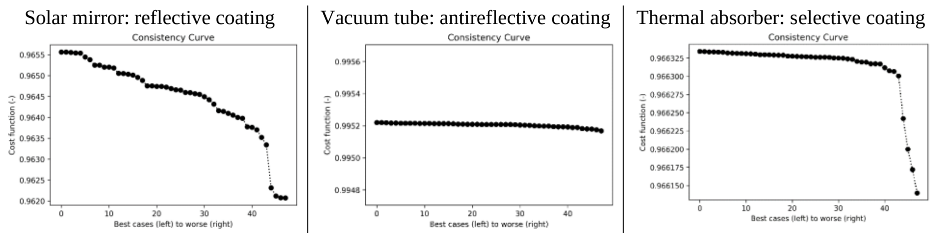

Figure 6 : Consistency Curve for different optimization process, using DE optimization method with an adequate budget

3.1. Availability

The actual version (v1.0.0) run under Python 3. which make it available for every system (Windows, Linux, MacOS) with can handle Python. The code automatically saves the most important results on texts files and PNG pictures: local writing rights are required. In addition, SolPOC is relevant on multicore computer thank to this ability to handle multiprocessing. Be sure that all modules require by the code are properly installed. The GitHub project include different tutorials, a User Guide with different documents and Jupyter NoteBook.

Software location: Github
Name: SolPOC
Identifier: https://github.com/mrgrosjeanantoine/SOLPOC
Licence : GNU General Public License v3.0
DOI :
Date published:

4. REUSE POTENTIAL

SolPOC can be used in all scientific domains where light incoming one or several thin layer(s) stacks deposited on a substrate. This software is very relevant for research, development, and education in solar energy, including solar thermal, photovoltaic or glasses [39,60]. SolPOC has already made a substantial contribution to coatings research for solar systems, such as solar mirror, antireflective coatings, and selective coatings designed for solar thermal applications [34,36,45]. Given that software employs a rapid and stable method to calculate reflectivity, transmissivity, and absorptivity within thin layers stack, we are confident that its utility extends to a wide spectrum of disciplines, including but not limited to:

- advanced reflective coatings, using metallic and/or dielectric layers.
- antireflective coatings for human eye vision, PV cells or solar thermal applications.
- coatings for optical instruments, such as Bragg mirrors.
- radiative cooling coatings
- low-e coatings and solar control glass for building application.
- Selective or absorbent coatings for solar thermal applications.

We are assured that SolPOC will continue to be asset to the solar community and can be readily adapted and applied to other communities in the future.

5. DISCUSSION

The aim differences between SolPOC and open-source code as PyMoosh or TMM-Fast is than SolPOC have been specially designed for the solar energy community. Even if PyMoosh (Update of Moosh code on Python) is a strong optics and photonics code for research and educational purposes with advanced functionality, it is not including a function for quickly evaluate the optical response over a full solar domain [23]. TMM-Fast code has been proposed in 2022, where SolPOC was already in use in our teams [25]. TMM-Fast, use, as its name suggest, the Transfer Matrix Method while we prefer to use the Abélès formalism [23]. Secondly for solar energy use and depending on the computer, an in-house benchmark shows us than SolPOC calculation time is similar or quicker than TMM-Fast.

6. CONCLUSION

The deployment of renewable energies such as photovoltaics or solar thermal systems requires innovative and highly efficiency surface coatings to efficiently harness and convert the abundant energy from the sun. Industrial and academic need a free, easily tunable, and efficient code for modeled and optimized thin layers stack for solar energy. After several years of development and internal use in PROMES-CNRS laboratories, we propose to make SolPOC (Solar Performances Optimization Code) freely available to the community. SolPOC is a Python code specifically crafted for studying and designing optical surface coatings for energy applications. The code relies on proven optical theories with different evolutionary optimization algorithms and cost function specially designed for solar energy utilization. We want to emphasize the critical importance of open science, advocating for the transparent sharing of codes and methods as no similar code exist. This approach stands as the most effective means to surmount the inevitable challenges encountered in the development of advanced coatings for solar energy systems.

Commenté [AG2]: Le déploiement des énergies renouvellable comme le photovoltaique ou le solaire thermique nécessite des traitements de surface innovant et adaptée pour collecter et convertir l'énergie gratuite de soleil. L'industrie et la recherche ont besoin d'un outil de modélisation et d'optimization gratuity, facilement adaptable au nouveau besoin et concus pour les usages spécifique de l'énergie solaire. Apres plusieurs années de developpement et d'utilisation interne nous proposer de mettre gratuitement à disposition de la communauté SolPOC (Solar Performances Optimization Code), un code Python spécifiquement concus pour étudier et concevoir des traitements de surface optique pour des applications énérgétique. Le code s'appuie sur des théorie optique éprouvé et des méthodes d'optimisation évolutionaire ayant fait leurs prevue pour l'étude, la modélisation et l'optimisation d'empillements de couches mince.

7. AUTHORS INFORMATION

Antoine Grosjean, main author. Wrote the code and the documentation, tested the code. Supervised the whole project since 2016.

Pauline Bennet, beta tested and quality control the code and the documentation, major contribution to optimization method (*Differential Evolution*)

Antoine Moreau, major contribution to *Differential Evolution*.

Thalita Drumond, implement the multiprocessing, control the code quality, writing documentation and GitHub repository

Amine Mahammou, beta tested the code and contribute to the documentation.

Denis Langevin, quality control and contributed to the optical theory.

Audrey Soum-Glaude, contributed to the optical theory and quality control the materials database. Supervised code users at PROMES-CNRS laboratory since 2016.

All the authors contributed to the writing of the article.

8. REFERENCES

[1] C. Atkinson, C.L. Sansom, H.J. Almond, C.P. Shaw, Coatings for concentrating solar systems - A review, Renewable and Sustainable Energy Reviews. 45 (2015) 113–122. https://doi.org/10.1016/j.rser.2015.01.015.

[2] Flamant Gilles, Matériaux pour le solaire à concentration, in: Le Solaire à Concentration, ISTE, 2021.

[3] H. Wankerl, M.L. Stern, A. Mahdavi, C. Eichler, E.W. Lang, Parameterized reinforcement learning for optical system optimization, J Phys D Appl Phys. 54 (2021) 305104. https://doi.org/10.1088/1361-6463/abfddb.

[4] K. Fisher, Z. (Jason) Yu, R. Striling, Z. Holman, PVMirrors: Hybrid PV/CSP collectors that enable lower LCOEs, AIP Conf Proc. 1850 (2017) 020004. https://doi.org/10.1063/1.4984328.

[5] Z.J. Yu, K.C. Fisher, Z.C. Holman, Evaluation of spectrum-splitting dichroic mirrors for PV mirror tandem solar cells, in: 2015 IEEE 42nd Photovoltaic Specialist Conference, PVSC 2015, Institute of Electrical and Electronics Engineers Inc., 2015. https://doi.org/10.1109/PVSC.2015.7355761.

[6] G. Wang, X.F. Cheng, P. Hu, Z.S. Chen, Y. Liu, L. Jia, Theoretical analysis of spectral selective transmission coatings for solar energy PV system, Int J Thermophys. 34 (2013) 2322–2333. https://doi.org/10.1007/s10765-011-1143-3.

[7] G. Wang, Y. Yao, J. Lin, Z. Chen, P. Hu, Design and thermodynamic analysis of a novel solar CPV and thermal combined system utilizing spectral beam splitter, Renew Energy. 155 (2020) 1091–1102. https://doi.org/10.1016/j.renene.2020.04.024.

[8] J.A. Dobrowolski, D. Lowe, Optical thin film synthesis program based on the use of Fourier transforms, Appl. Opt. 17 (1978) 3039–3050. https://doi.org/10.1364/AO.17.003039.

[9] A. Tikhonravov, M. Trubetskov, Development of the needle optimization technique and new features of “OptiLayer” design software, 1994. http://spiedl.org/terms.

[10] R. Morf, R.E. Kunz, Dielectric Filter Optimization By Simulated Thermal Annealing, in: K.H. Guenther, H.K. Pulker (Eds.), Thin Film Technologies III, SPIE, 1989: pp. 211 – 219. https://doi.org/10.1117/12.950040.

[11] J.-M. Yang, C.-Y. Kao, An evolutionary algorithm for the synthesis of multilayer coatings at oblique light incidence, Journal of Lightwave Technology. 19 (2001) 559–570. https://doi.org/10.1109/50.920855.

[12] J.C.C. Mak, C. Sideris, J. Jeong, A. Hajimiri, J.K.S. Poon, Binary particle swarm optimized 2x2 power splitters in a standard foundry silicon photonic platform, Opt. Lett. 41 (2016) 3868–3871. https://doi.org/10.1364/OL.41.003868.

[13] Wolfgang Theiss, CODE/SCOUT, (2023). https://www.wtheiss.com/ (accessed September 13, 2023).
[14] Thin Film Center, Essential Macleod, (2023). https://www.thinfilmcenter.com/essential.php (accessed September 13, 2023).
[15] FTG Software Associates, FilmStar DESIGN, (2023). https://www.ftgsoftware.com/design.htm (accessed September 13, 2023).
[16] S. Larouche, L. Martinu, OpenFilters: open-source software for the design, optimization, and synthesis of optical filters., Appl Opt. 47 13 (2008) C219-30. https://api.semanticscholar.org/CorpusID:35706163.
[17] Stéphane Larouche, Ludvik Martinu, OpenFilters, (2008). https://openfilters.software.informer.com/1.1/ (accessed September 13, 2023).
[18] Optenso, OpTaliX, (2023). https://www.optenso.com/ (accessed September 15, 2023).
[19] O. Com, OptiLayer Thin Film Software, 2022.
[20] GalebOptics, Phase CODE, (2023). https://galeboptics.com/phasecode.html (accessed September 14, 2023).
[21] J. Defrance, C. Lemaître, R. Ajib, J. Benedicto, E. Mallet, R. Pollès, J.-P. Plumey, M. Mihailovic, E. Centeno, C. Ciracì, D.R. Smith, A. Moreau, Moosh: A Numerical Swiss Army Knife for the Optics of Multilayers in Octave/Matlab, J Open Res Softw. 4 (2016) 13. https://doi.org/10.5334/jors.100.
[22] Moreau Antoine, Bennet Pauline, Langevin Denis, Wiecha Peter, PyMoosh, (2023). https://github.com/AnMoreau/PyMoosh (accessed September 12, 2023).
[23] D. Langevin, P. Bennet, A. Khaireh-Walieh, P. Wiecha, O. Teytaud, A. Moreau, PyMoosh : a comprehensive numerical toolkit for computing the optical properties of multilayered structures, (2023). http://arxiv.org/abs/2309.00654.
[24] Hulinks, TFCalc, (2023). https://www.hulinks.co.jp/en/tfcalc-e (accessed September 13, 2023).
[25] A. Luce, A. Mahdavi, F. Marquardt, H. Wankerl, TMM-Fast, a transfer matrix computation package for multilayer thin-film optimization: tutorial, Journal of the Optical Society of America A. 39 (2022) 1007. https://doi.org/10.1364/josaa.450928.
[26] Alexander Luce, TMM Fast, (2023). https://github.com/MLResearchAtOSRAM/tmm_fast (accessed September 13, 2023).
[27] RP Photonics, RP-Coating V4, (2013). https://www.rp-photonics.com/rp_coating.html (accessed September 14, 2023).
[28] M. N. Polyanskiy, Refractive index database, (2023). https://refractiveindex.info (accessed September 11, 2023).
[29] D.A.G. Bruggeman, Berechnung verschiedener physikalischer Konstanten von heterogenen Substanzen. I. Dielektrizitätskonstanten und Leitfähigkeiten der Mischkörper aus isotropen Substanzen, Ann Phys. 416 (1935) 636–664. https://doi.org/https://doi.org/10.1002/andp.19354160705.
[30] Schott Optical Glass Datasheets, 2023. obtained from http://www.schott.com (accessed September 11, 2023).
[31] A. Soum-Glaude, I. Bousquet, M. Bichotte, S. Quoizola, L. Thomas, G. Flamant, Optical Characterization and Modeling of Coatings Intended as High Temperature Solar Selective Absorbers, Energy Procedia. 49 (2014) 530–537. https://doi.org/10.1016/J.EGYPRO.2014.03.057.

[32] Aránzazu Fernández-García, Florian Sutter, Marco Montecchi, Fabienne Sallaberry (CENER), Anna Heimsath (Fraunhofer ISE), Carlos Heras, Estelle Le Baron, Audrey Soum-Glaude, Guidelines Parameters and Methode to Evaluate the Reflectance Properties OF Materials for Concentrating Solar Power Technology Under Laboratory Conditions, Official Reflectance Guideline Version 3.1 April 2020, 2020.
[33] F. Abelès, La théorie générale des couches minces, Journal de Physique et Le Radium. 11 (1950) 307–309. https://doi.org/10.1051/jphysrad:01950001107030700.
[34] A. Grosjean, A. Soum-Glaude, L. Thomas, Replacing silver by aluminum in solar mirrors by improving solar reflectance with dielectric top layers, Sustainable Materials and Technologies. 29 (2021) e00307. https://doi.org/10.1016/J.SUSMAT.2021.E00307.
[35] B.P. Jelle, Solar radiation glazing factors for window panes, glass structures and electrochromic windows in buildings—Measurement and calculation, Solar Energy Materials and Solar Cells. 116 (2013) 291–323. https://doi.org/10.1016/J.SOLMAT.2013.04.032.
[36] A. Grosjean, A. Soum-Glaude, L. Thomas, Influence of operating conditions on the optical optimization of solar selective absorber coatings, Solar Energy Materials and Solar Cells. 230 (2021) 111280. https://doi.org/10.1016/J.SOLMAT.2021.111280.
[37] D.R. Myers, K. Emery, C. Gueymard, Proposed reference spectral irradiance standards to improve concentrating photovoltaic system design and performance evaluation, in: Conference Record of the Twenty-Ninth IEEE Photovoltaic Specialists Conference, 2002., 2002: pp. 923–926. https://doi.org/10.1109/PVSC.2002.1190731.
[38] C.A. Gueymard, The SMARTS spectral irradiance model after 25 years: New developments and validation of reference spectra, Solar Energy. 187 (2019) 233–253. https://doi.org/10.1016/J.SOLENER.2019.05.048.
[39] A. Grosjean, E. Le Baron, Longtime solar performance estimations of low-E glass depending on local atmospheric conditions, Solar Energy Materials and Solar Cells. 240 (2022) 111730. https://doi.org/10.1016/J.SOLMAT.2022.111730.
[40] Silvana Ayala, Juan Russo, Py-SMARTS, (2020). https://github.com/NREL/pySMARTS (accessed September 19, 2023).
[41] B.P. Jelle, A. Gustavsen, T.N. Nilsen, T. Jacobsen, Solar material protection factor (SMPF) and solar skin protection factor (SSPF) for window panes and other glass structures in buildings, Solar Energy Materials and Solar Cells. 91 (2007) 342–354. https://doi.org/10.1016/j.solmat.2006.10.017.
[42] S. Winter, D. Friedrich, Effects of the New Standard IEC 60904-3:2008 on the Calibration Results of Common Solar Cell Types, (2009). https://doi.org/10.4229/24thEUPVSEC2009-4AV.3.67.
[43] J. Kennedy, R. Eberhart, Particle Swarm Optimization, in: page 1942--1948. Proceedings of the IEEE International Conference on Neural Networks (Ed.), 1995. https://doi.org/http://dx.doi.org/10.1109/ICNN.1995.488968.
[44] R. Storn, K. Price, Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization. 11 (1997) 341–359. https://doi.org/10.1023/A:1008202821328.
[45] A. Grosjean, A. Soum-Glaude, P. Neveu, L. Thomas, Comprehensive simulation and optimization of porous SiO2 antireflective coating to improve glass solar transmittance for solar energy applications, Solar Energy Materials and Solar Cells. 182 (2018) 166–177. https://doi.org/10.1016/J.SOLMAT.2018.03.040.

[46] A. V Tikhonravov, M.K. Trubetskov, G.W. DeBell, Optical coating design approaches based on the needle optimization technique, Appl. Opt. 46 (2007) 704–710. https://doi.org/10.1364/AO.46.000704.
[47] M.A. Barry, V. Berthier, B.D. Wilts, M.C. Cambourieux, P. Bennet, R. Pollès, O. Teytaud, E. Centeno, N. Biais, A. Moreau, Evolutionary algorithms converge towards evolved biological photonic structures, Sci Rep. 10 (2020). https://doi.org/10.1038/s41598-020-68719-3.
[48] P. Bennet, Optimisation numérique des structures photoniques, 2022. http://www.theses.fr/2022UCFAC052/document.
[49] N. Hansen, Y. Akimoto, P. Baudis, CMA-ES/pycma on Github, (2019). https://doi.org/10.5281/zenodo.2559634.
[50] FacebookResearch, Ax - adaptive experimentation, (2020). https://ax.dev/ (accessed September 22, 2023).
[51] J. Bergstra, B. Komer, C. Eliasmith, D. Yamins, D.D. Cox, Hyperopt: a Python library for model selection and hyperparameter optimization, Comput Sci Discov. 8 (2015) 14008. https://doi.org/10.1088/1749-4699/8/1/014008.
[52] F. Hutter, H. Hoos, K. Leyton-Brown, Lecture Notes in Computer Science, in: LION, Springer, 2011: pp. 507–523.
[53] J. Rapin, O. Teytaud, Nevergrad - A gradient-free optimization platform, (2018). https://github.com/facebookresearch/nevergrad (accessed September 22, 2023).
[54] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Transactions on Evolutionary Computation. 1 (1997) 67–82. https://doi.org/10.1109/4235.585893.
[55] P. Bennet, D. Langevin, C. Essoual, A. Khaireh-Walieh, O. Teytaud, P. Wiecha, A. Moreau, An illustrated tutorial on global optimization in nanophotonics, (2023). http://arxiv.org/abs/2309.09760.
[56] J. Rapin, P. Bennet, E. Centeno, D. Haziza, A. Moreau, O. Teytaud, Open Source Evolutionary Structured Optimization, in: Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion, Association for Computing Machinery, New York, NY, USA, 2020: pp. 1599–1607. https://doi.org/10.1145/3377929.3398091.
[57] P. Bennet, C. Doerr, A. Moreau, J. Rapin, F. Teytaud, O. Teytaud, Nevergrad: black-box optimization platform, ACM SIGEVOlution. 14 (2021) 8–15. https://doi.org/10.1145/3460310.3460312ï.
[58] D.P. Rodgers, Improvements in Multiprocessor Sgstem Design, 1985.
[59] A. Grosjean, Etude, modélisation et optimisation de surfaces fonctionnelles pour les collecteurs solaires thermiques à concentration, 2018. http://www.theses.fr/2018PERP0002/document.
[60] D. Ngoue, A. Grosjean, L. Di Giacomo, S. Quoizola, A. Soum-Glaude, L. Thomas, Y. Lalau, R. Reoyo-Prats, B. Claudet, O. Faugeroux, C. Leray, A. Toutant, J.-Y. Peroy, A. Ferrière, G. Olalde, 3 - Ceramics for concentrated solar power (CSP): From thermophysical properties to solar absorbers, in: O. Guillon (Ed.), Advanced Ceramics for Energy Conversion and Storage, Elsevier, 2020: pp. 89–127. https://doi.org/https://doi.org/10.1016/B978-0-08-102726-4.00003-X.
[61] L. Di Giacomo, PACVD/PVD de multicouches sélectives pour la conversion thermodynamique de l’énergie solaire, 2018. https://theses.hal.science/tel-03859282.

[62] D. Ngoue, (Nano)composites en revêtement déposés par technologie plasma pour la conversion de l’énergie solaire, 2021. http://www.theses.fr/2021PERP0011/document.
[63] National Renewable Energy Laboratory, System Advisor Model Version 2022.11.29 (SAM 2022.11.21), (2022). https://https://sam.nrel.gov (accessed June 26, 2023).