WPTherml: A Python Package for the Design of Materials for Harnessing Heat

James F. Varner, Noor Eldabagh, Derek Volta, Reem Eldabagh, Jonathan Foley

Submitted date: 28/06/2019 • Posted date: 28/06/2019
Licence: CC BY-NC-ND 4.0

Citation information: Varner, James F.; Eldabagh, Noor; Volta, Derek; Eldabagh, Reem; Foley, Jonathan (2019): WPTherml: A Python Package for the Design of Materials for Harnessing Heat. ChemRxiv. Preprint.

WPTherml is a Python package for the design of materials with tailored optical and thermal properties for the vast number of energy applications where control of absorption and emission of radiation, or conversion of heat to radiation or vice versa, is paramount. The optical properties are treated within classical electrodynamics via the Transfer Matrix Method which rigorously solve Maxwell’s equations for layered isotropic media. A flexible multilayer class connects rigorous electrodynamics properties to figures of merit for a variety of thermal applications, and facilitates extensions to other applications for greater reuse potential. WPTherml can be accessed at https://github.com/FoleyLab/wptherml.

File list (3)

| File Name                  | Size       | View       | Download |
|----------------------------|------------|------------|----------|
| 271-3307-1-RV.docx         | 0.93 MiB   | view on ChemRxiv | download file |
| 271-3307-1-RV.pdf          | 765.01 KiB | view on ChemRxiv | download file |
| WPtherml.png               | 836.81 KiB | view on ChemRxiv | download file |
Software paper for submission to the Journal of Open Research Software

To complete this template, please replace the blue text with your own. The paper has three main sections: (1) Overview; (2) Availability; (3) Reuse potential.

Please submit the completed paper to: editor.jors@ubiquitypress.com

(1) Overview

Title
WPTherml: A Python Package for the design of materials for harnessing heat.

Paper Authors
1. James F. Varner
2. Noor Eldabagh
3. Derek Volta
4. Reem Eldabagh
5. Jonathan J. Foley IV (corresponding author)

Paper Author Roles and Affiliations
1. Software development, planning, and testing, co-wrote the manuscript and documentation. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
2. Software development and testing. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
3. Software development and testing. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
4. Software development and testing. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
5. Project conception, software development, planning, and testing, co-wrote the manuscript and documentation. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470

Abstract
WPTherml is a Python package for the design of materials with tailored optical and thermal properties for the vast number of energy applications where control of absorption and emission of radiation, or conversion of heat to radiation or vice versa, is paramount. The optical properties are treated within classical electrodynamics via the Transfer Matrix Method which rigorously
solve Maxwell's equations for layered isotropic media. A flexible multilayer class connects rigorous electrodynamics properties to figures of merit for a variety of thermal applications, and facilitates extensions to other applications for greater reuse potential. WPTherml can be accessed at https://github.com/FoleyLab/wptherml.

Keywords
Nanophotonics; Solar Thermophotovoltaics; Passive Cooling; Radiative Cooling; Computational Electrodynamics; Concentrated Solar Power; Transfer Matrix Method

Introduction
In 2017 over sixty five percent of energy produced was not useable and released as waste heat [1]; consequently, there are a large number of opportunities to develop technologies which can mitigate waste heat associated with energy production, or which can be used to convert or reclaim waste heat for useful purposes. Nano-scale materials, including multi-layered nanostructures among others, with tailored optical and thermal emission properties are centrally important for many of these envisioned technologies, which include solar thermophotovoltaics (STPV) [2-4], radiative cooling devices [5], and highly efficient incandescent lighting [6] that minimize IR losses.

WPTherml (William Paterson University's tool for Thermal Energy and Radiation management with Multi-Layer nanostructures), is a computational engine for materials which can be leveraged for these and other technologies where the control of optical and/or thermal radiation properties is paramount. Specifically, WPTherml can be utilized to simulate optical properties (reflectivity, transmissivity, absorptivity, and emissivity) of nanostructures made from layered isotropic media; thermal radiation spectra of these nanostructures can be derived from these optical quantities in conjunction with Planck’s blackbody radiation law. With these spectral quantities in hand, figures of merit for the design of such multi-layer nanostructures for applications such as STPV, passive radiative cooling, concentrated solar power, incandescent lighting, and plasmonics can be computed. Thus, the WPTherml package serves as an engine for computing spectral quantities, and for relating these spectral quantities to relevant figures of merit. Central to the WPTherml package is a multilayer class; computation of spectral quantities and figures of merit are performed by methods of the multilayer class and are stored as attributes. Spectral quantities are computed within the framework of classical electrodynamics,
specifically utilizing the transfer matrix method [7], which is
implemented in a library called \textit{tmm}. The required inputs for the
transfer matrix method include the geometry (thickness) and
refractive index of each layer in the multi-layer nanostructure, so
the \textit{multilayer} class has an associated method for parsing user
input in an intuitive format to specify the geometry and material
composition of a multi-layer structure. Refractive index data for a
number of common materials is stored and accessed within a data
library called \textit{datalib}. This library also provides access to relevant
data for the computation of various figures of merit, for example,
spectral response functions of several popular photovoltaic
materials (see Figure 1 for an illustration of the spectral response
of InGaAsSb), the AM1.5 solar spectrum, the photopic luminosity
function, and the atmospheric transmissivity spectrum can all be
accessed using the functionality of \textit{datalib}.

The core of WPThermal is the transfer matrix method,
implemented in the \textit{tmm} library, which is a classical
electrodynamics approach that relates the incident electromagnetic
field upon a multi-layer structure to the reflected and transmitted
electromagnetic fields, thereby permitting computation of
reflection, transmission, and absorption of light by the multilayer:

\begin{equation}
\begin{pmatrix}
1 \\
r
\end{pmatrix} =
\begin{pmatrix}
M_{1,1} & M_{1,2} \\
M_{2,1} & M_{2,2}
\end{pmatrix}
\begin{pmatrix}
t \\
0
\end{pmatrix}, (1)
\end{equation}

where $M_{i,j}$ are the elements of the transfer matrix, $r$ is the
reflection amplitude,

\begin{equation}
\begin{aligned}
r &= \frac{M_{2,1}}{M_{1,1}}, (2)
\end{aligned}
\end{equation}

and $t$ is the transmission amplitude,

\begin{equation}
\begin{aligned}
t &= \frac{1}{M_{1,1}}, (3)
\end{aligned}
\end{equation}

The reflection and transmission amplitudes may be used to
compute the reflectivity, transmissivity, absorptivity, and emissivity
of the structure:

\begin{equation}
\begin{aligned}
R &= r^* r, (4) \\
T &= t^* t \frac{n_i \cos \theta_i}{n_i \cos \theta_L}, (5) \\
A &= \epsilon = 1 - R - T, (6)
\end{aligned}
\end{equation}
where $R$ denotes the reflectivity, $T$ the transmissivity, $A$ the absorptivity, and $\epsilon$ the emissivity. It should be noted that all quantities above depend on wavelength, and the explicit wavelength dependence of the elements of the transfer matrix have been defined in detail [8]. The core transfer matrix library has been utilized by one of the authors to model thermal emission of multi-layer planar nanostructures [4], and the optical properties of the 2D material known as Borophene [9].

In Equation (6), an important relation known as Kirchoff’s law [2-4] is invoked, which equates the absorption spectrum of a multi-layer structure to its emissivity spectrum. The emissivity spectrum plays a key role in shaping the thermal emission spectrum of a structure at a given temperature,

$$TE(\lambda, T_{ml}) = \rho(\lambda, T_{ml}) \epsilon(\lambda),$$

where $TE(\lambda, T)$ denotes the thermal emission, and

$$\rho(\lambda, T_{ml}) = \frac{2hc^2}{\lambda} \exp\left(\frac{hc}{\lambda k_B T_{ml}}\right) - 1.$$

is Planck’s blackbody law, where $T_{ml}$ denotes the temperature of the multi-layer structure, $\lambda$ denotes the wavelength of light, $h$ is Planck’s constant, $c$ is the speed of light, and $k_B$ is Boltzmann’s constant. In the following, we will illustrate the particular application of WPTherml to solar thermophotovoltaics (STPV), where one tailors the thermal emission spectrum of an object through its emissivity such that it is dramatically different from an ideal blackbody emitter, which has an emissivity of 1 for all wavelengths. Following that discussion, we will briefly summarize several other applications for which WPTherml has been specifically designed to address.

Example application to STPV

**Overview of application**

Management of thermal emission as an energy source, as is done in STPV, may have many beneficial impacts in multiple markets including but not limited to clean energy, military use, and the consumer market [3]. Here we illustrate the use of WPTherml to simulate the thermal emission of a multi-layer nanostructure optimized for performance as an STPV emitter; this structure is illustrated in Figure 1 (a). Briefly, the objective is to tune the thermal emission of a material so that it is well matched to the
spectral response of a photovoltaic (PV) material. In Figure 1 (b), we illustrate the thermal emission spectrum of the structure when it is heated to 1700 K and overlay it with the spectral response function of an InGaAsSb PV cell, which is commonly used in STPV applications [4], along with the blackbody spectrum at 1700 K. Importantly, the thermal emission of the structure illustrated in Figure 1(a) strongly departs from the blackbody spectrum, and has excellent overlap with the spectral response of InGaAsSb. To further illustrate the suitability of this structure for STPV applications, we plot the product of the thermal emission of this structure with the spectral response of InGaAsSb in Figure 1(c). It is in fact the ratio of the area under this curve, relative to the total area under the thermal emission curve, which plays a critical role in the overall conversion efficiency in a given STPV device (see [8] for more details). An example of code used to generate these figures in WPTherml is included below Figure 1.

**Example Code**

```python
from wptherml.wpml import multilayer
from matplotlib import pyplot as plt
from wptherml.datalib import datalib

structure = {
    'Temperature': 1700,
    'Material_List': ['Air', 'Al2O3', 'SiO2', 'TiO2', 'SiO2', 'Al2O3', 'W', 'Air'],
```

![Figure 1](image_url)

**Figure 1**: (a) Example of a multi-layer structure that exhibits selective thermal emission that is nearly optimal for the spectral response of InGaAsSb PVs. (b) The spectral response (in A/W) of InGaAsSb plotted against the thermal emission spectrum of the structure illustrated in (a) and the blackbody spectrum at 1700 K; these thermal emission spectra are scaled by an arbitrary number (what is the number in this case?) to aid visualization against the spectral response function. (c) Plot of the convolution of the spectral response function of InGaAsSb with the thermal emission of the structure in (a) plotted against the thermal emission itself; these two functions are the integrands in that give rise to the short circuit current and the incident power in a STPV system, respectively. See reference [8] for a more detailed discussion of these quantities and their relation to the STPV efficiency.
In the above code example, an instance of the `multilayer` class called `stpv_fig` is created following user input defined in the dictionary called `structure`. The physical structure that this instance models consists of several dielectric layers (aluminium oxide (Al$_2$O$_3$), titanium dioxide (TiO$_2$), and glass (SiO$_2$)), and an optically-thick tungsten (W) layer, which are defined in the ‘Material List’ key of the `structure` dictionary. Upon instantiation, the optical and thermal emission properties of this structure at 1700 K (specified by ‘Temperature’) are computed for 1000 wavelengths between 300 and 6000 nm (specified in ‘Lambda List’) and stored as attributes of `stpv_fig`. Additionally, because the key ‘STPV_EMIT’ is set to the value 1, several STPV figures of merit for this structure, including the spectral efficiency, power density, and the TPV efficiency, will be computed at instantiation and stored as attributes the `stpv_fig.spectral_efficiency_val`, `stpv_fig.power_density_val`, `stpv_fig.tpv_efficiency_val`, respectively. The mathematical details of these figures of merit are discussed in references [4,8].

After instantiation, a modification to one of the layers is made whereby the layer with index 1 (originally Al$_2$O$_3$) is changed to an
alloy of Al$_2$O$_3$ and W with a volume fraction of 75% W; note that in this example, we specify a static refractive index of 1.66+0i for the Al$_2$O$_3$ layer. The model used to define the wavelength-dependent refractive index of this alloy is known as the Maxwell-Garnett effective medium theory [4], specified by the argument “MaxwellGarnett” (an alternative model known as the Bruggeman model [4] can be invoked with the argument “Bruggeman”). Once this modification is made, the optical properties (reflectivity, transmissivity, emissivity) are recomputed by calling the fresnel() method, and the thermal emission spectrum is recomputed with these updated quantities by calling the thermal_emission() method.

Also illustrated in this example is how our data library datalib can be used to extract relevant data (either stored or computed) including the spectral response function of InGaAsSb (datalib.SR_InGaAsSb(array_of_wavelengths)) and the blackbody spectrum at a specific temperature (datalib.BB(array_of_wavelengths, temperature)). The remaining code in the example represents standard usage of the python library matplotlib, for which extensive documentation already exists [10]. A full list of methods and attributes associated with the multilayer class can be found in the WPTherml README documentation [11].

**Brief Summary of Additional Applications**

The control of optical and thermal emission properties of materials is critical for a number of applications, and here we briefly summarize some of these applications for which WPTherml provides unique capabilities.

Incandescent light bulbs with efficiencies approaching or exceeding that of LED-based lights can be realized by tailoring the thermal emission of filament materials [6]. In particular, the thermal emission should be high in the visible and low in the IR at standard operating temperatures (~2500 K). The key to this property is to tailor the emissivity of the filament material to have strong overlap with the photopic luminosity function, which describes the sensitivity of the human eye to visible light [6]. A figure of merit for incandescent bulbs, known as the luminous efficiency, is related to the convolution of the thermal emission and the photopic luminosity function [6,8], and this figure of merit can be computed using the method luminous_efficiency() of the multilayer class (see [11] for more details).

Daytime radiative cooling is an application where a material is used to passively cool a space through a net negative radiative
power flux from the material’s surface, thereby eliminating or mitigating the need for the use of air conditioning [5]. The key properties of these materials are that they are highly reflective in the solar spectrum and highly emissive in mid-IR wavelengths, specifically wavelengths where the atmosphere is highly transparent (~8-14 microns); Hence, these materials minimize the positive (warming) radiative power flux associated with solar absorption and absorption of thermal radiation from the atmosphere, and maximize the negative (cooling) radiative power flux associated with the material’s thermal emission at its operating temperature. These power fluxes are computed by the method `cooling_power()` in the `multilayer` class. An illustration of the emissivity of such a radiative cooling structure taken from reference [5] is shown in Figure 2, and the code used to make this structure can be found in a Jupyter notebook that ships with our package [12].

![Figure 2](image.png)

**Figure 2:** (a) Visible absorptivity/emissivity (blue line) of a multi-layered structure optimized for radiative cooling plotted alongside the AM1.5 solar spectrum, illustrating that this structure minimizes absorption of solar power. (b) IR emissivity (red line) of the multi-layered structure optimized for radiative cooling plotted alongside the transmissivity of the earth’s atmospheres, illustrating that this structure maximizes ambient thermal radiation and minimizes absorption of thermal radiation from the atmosphere. Further details of this structure can be found in reference [5], and code to reproduce these figures in WPTerml can be found in a Jupyter notebook that ships with our package [12].

The optical properties of thin metal films has been the subject of intensive investigation owing to their ability to support optical resonances known as surface plasmon polaritons [13-15], and
perfectly absorbing modes [16-18]. We have included the capability to analyze multi-layer structures for both surface plasmon polariton and perfectly absorbing modes through the find_spp() and find_pa() methods of the multilayer class, respectively. Mathematical details of these modes can be found in references [8, 13, 16].

Finally, to aid in the user’s intuition for the relation between optical properties, thermal emission, and appearance of a structure, we have implemented methods to render the color of a structure based on its reflectance (ambient_color()) or based on its thermal emission (thermal_color()). The details behind these methods are discussed in reference [8].

Description of alternative packages

There are several open-source software packages that implement the transfer matrix method to simulate the optical response (for example, reflectivity, transmissivity, absorptivity, emissivity) of multi-layer nanostructures; a partial list includes several python-based transfer matrix method packages [19-23], a cython implementation [24], and a Matlab implementation [25]. Related electrodynamics methods, specifically the S-matrix method, have also been implemented in open-source software packages [26-28]. The authors are currently unaware of any alternative software package that leverages such rigorous electrodynamics techniques for the use in thermal radiation applications; consequently, WPTherml provides a unique and useful capability for a large community of scientists and engineers who aim to design materials and technologies for harnessing heat.

Implementation and architecture

WPTherml is written in pure Python 3 with minimal dependencies including numpy, scipy, and matplotlib. WPTherml has been tested with Anaconda Python 3.6 and 3.7 running on Windows 10, Mac OS Sierra, Mac OS Mojave, and Fedora 22. WPTherml includes a multilayer class, several function libraries (tmm, numlib, stpvlib, coolinglib, colorlib, and lightlib), and a data library (datalib). The function libraries contain the functions that perform calculations to extract spectral information and figures of merit for the various applications supported by WPTherml as follows:
- tmm leverages the transfer matrix method to compute optical properties (reflectivity, transmissivity, absorptivity, emissivity) of multi-layered nanostructures.
- **numlib** provides a simple function for numerical quadrature, which is used to compute a variety of figures of merit.
- **stpvlid** uses optical properties and thermal emission to compute figures of merit relevant to solar thermophotovoltaic, concentrated solar, and standard photovoltaic applications.
- **coolinglib** uses optical properties and thermal emission to compute figures of merit relevant to radiative cooling.
- **colorlib** uses optical properties and thermal emission to compute color of structures, and to classify a spectrum as giving rise to one of the 7 colors in the ROYGBIV spectrum.
- **lightlib** uses optical properties and thermal emission to compute figures of merit relevant for incandescent light sources.
- **datalib** provides data to support various calculations, including:
  - Refractive index as a function of wavelength for an extensive number of common materials; this data is required by the transfer matrix method for computing optical properties.
  - Spectral Response functions for several common photovoltaic materials; this data is required for computing STPV and standard PV efficiencies.
  - AM1.5 spectral data; this data is required for traditional PV efficiency calculations, concentrated solar absorption calculations, and radiative cooling applications.
  - Atmospheric transmission spectrum; this data is required for radiative cooling applications.
  - Spectral response functions for the human eye; this data is required for rendering the colors of structures and for computing the luminous efficiency of incandescent light sources.

The `multilayer` class provides an intuitive framework for connecting user input to the functionality of WPTherml. The user specifies information about the materials, geometry (thickness), and calculation type to the multilayer class through a dictionary whose keys are descriptive of their meaning with respect to the class. The following code illustrates a simple example where a dictionary called `structure` is defined such that when passed to the multilayer class, it specifies the 3-layer structure illustrated in Figure 3 below:

```python
structure = {
    'Material_List': ['Air', 'SiO2', 'TiO2', 'Au', 'Air'],
}  
```

This needs a header, and also, remove the colon above, because Figure 3 is not immediately below that sentence. Also, make sure you describe what this block of code is in the text.
coated_{au\_film} = \text{multilayer}(\text{\texttt{structure}})

It is important to note the ordering convention used for the 'Material List' and 'Thickness List' keys, whose values are lists of strings and floats that specify the materials and thicknesses of each layer in the multi-layer structure, respectively. In particular, these lists are both ordered starting from the side upon which light is incident in a typical reflection experiment and ending in the material into which light would finally be transmitted in the same experiment. Also of note is that both reflected light and thermally emitted light emanate from the top side of the structure. All optical and thermal emission spectra will be computed in the range 400 – 800 nm, with resolution of \((800-400)/1000\) nm as specified by the value of 'Lambda List'. A full list of supported keys that can be parsed by the \texttt{multilayer} class is provided in the WPThermal README documentation [11].

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{three_layer_structure.png}
\caption{Schematic of the three-layer structure as defined by the 'structure' dictionary above, illustrating the convention for the direction of incoming, reflected, transmitted, and emitted light with respect to the order in which materials and thicknesses are specified by the user.}
\end{figure}

A schematic of the flow of data between the user, the \texttt{multilayer} class, and the data and function libraries is illustrated in Figure 4 below.
Quality control

Unit testing was performed for all function libraries, reproducing results from peer-reviewed literature on various thermal applications when applicable. The tmm library was unit tested against the existing python transfer matrix package developed by S. J. Byrnes [19,20], and a system test of WPTherml that is directly comparable to this package can be run by the user through the Jupyter notebook entitled ‘Validate_Fresnel’ that ships with our package [29]. This test also serves as a check for the stpvlb library. The user may also run a system test of WPTherml that checks the functionality of coolinglib through the Jupyter notebook entitled ‘Validate_Cooling’ that ships with our package [30].

(2) Availability

Operating system

WPTherml has been tested on MacOS Sierra, MacOS Mojave, Windows 10, and Fedora 22. Since WPTherml is written in Python, it should run on any system on which Python and the WPTherml dependencies run.
Programming language

WPTherml is written in Python 3 and has been tested with Anaconda Python (3.6 and 3.7).

Additional system requirements

The total size of the WPTherml package source code and supporting data stored in datalib is approximately 50 MB; typical execution should not pose any significant memory demands.

Dependencies

WPTherml depends on numpy, scipy, and matplotlib.
The code was tested with
- numpy version 1.14.3
- scipy version 1.1.0
- matplotlib version 2.2.2

List of contributors

James Varner; Software development, planning, and testing.
Noor Eldabagh; Software development and testing.
Derek Volta; Software development and testing.
Reem Eldabagh; Software development and testing.
Jonathan J. Foley IV; Project conception, software development, planning, and testing.

Software location:

Archives (e.g. institutional repository, general repository)
(required – please see instructions on journal website for depositing archive copy of software in a suitable repository)

Name: SourceForge
Persistent identifier: https://sourceforge.net/projects/wptherml
Licence: GPL 3.0
Publisher: Jonathan Foley
Version published: v1.03 beta
Date published: 03/22/2019

Code repository

Name: GitHub
Identifier: https://github.com/FoleyLab/wptherml
Licence: GPL 3.0
Date published: 03/04/2019
Emulation environment (if appropriate)

Name: The name of the emulation environment
Identifier: The identifier (or URI) used by the emulator
Licence: Open license under which the software is licensed here
Date published: dd/mm/yy

Language
Python 3

(3) Reuse potential
WPTherml was developed to support the simulation and design of a practically endless number of structures for multiple applications where optical and thermal emission properties are paramount. The package was also developed to be easily extensible to support new applications as the developers or users see fit. Several potential applications are illustrated in Jupyter notebooks that ship with the package, and the developers plan to continue to add and refine these examples over time. Users seeking support for additional features can communicate with us through the issues page of the GitHub repository [31], or may contact the corresponding author by email at foleyj10@wpunj.edu.

Acknowledgements
The authors wish to acknowledge support from the College of Science and Health and the Department of Chemistry at William Paterson University, specifically Dean Kenneth Wolf, Dean Venkat Sharma, Prof. Bhanu Chauhan, and Dr. Mukesh Sahni, for their vision and persistence in realizing the computational infrastructure that allowed the development of WPTherml. J. F. V. acknowledges support from the Louis Stokes Alliance for Minority Participation, and J. J. F. acknowledges support from the ART program at William Paterson University.

Funding statement

Competing interests
The authors declare that they have no competing interests.

References
[1] https://flowcharts.llnl.gov/commodities/energy
[2] Gupta, M. C.; Ungaro, C.; Foley IV, J. J.; Gray, S. K. Optical nanostructures design, fabrication, and applications for solar/thermal energy conversion. Solar Energy 165, 100-114 (2018)
[3] Zhou, Z.; Sakr, E.; Sun, Y.; Bermel, P. Solar thermophotovoltaics: reshaping the solar spectrum. Nanophotonics 5, 1-21 (2016)
[4] Jeon, N.; Hernandez, J. J.; Rosenmann, D.; Gray, S. K.; Martinson, A. B. F.; Foley IV, J. J. Pareto Optimal Spectrally Selective Emitters for Thermophotovoltaics via Weak Absorber Critical Coupling. Adv. Energy Mater. **8**, 1801035 (2018)

[5] Raman, A. P.; Anoma, M. A.; Zhu, L.; Rephaeli, E.; Fan, S. Passive radiative cooling below ambient air temperature under direct sunlight. Nature **515**, 540-544 (2014)

[6] Ilic, O.; Bermel, P.; Chen, G.; Joannopoulos, J. D.; Celanovic, I.; Soljačić, M. Tailoring High Temperature Radiation and the Resurrection of the Incandescent Source. Nat. Nanotechnol. **11**, 320-324 (2016)

[7] Yeh, P. Optical Waves in Layered Media (Wiley, New York) 1988

[8] https://github.com/FoleyLab/wptherml/blob/master/documentation/Eq
duations.pdf

[9] Adamska, L.; Sadasivam, S.; Foley IV, J. J.; Darancet, P.; Sharifzadeh, S. First-principles investigation of Borophene as a monolayer transparent conductor. J. Phys. Chem. C **122** 4037-4045 (2018)

[10] https://matplotlib.org/contents.html

[11] https://foleylab.github.io/wptherml/

[12] https://github.com/FoleyLab/wptherml/blob/master/Validate_Cooling.ipynb

[13] Raether, H. Surface Plasmons on Smooth and Rough Surfaces and on GRATINGS (Springer-Verlag, 1988)

[14] Maier, S. M. Plasmonics: Fundamentals and applications (Springer, 2007).

[15] Foley IV, J. J.; McMahon, J. M.; Schatz, G. C.; Harutyunyan, H.; Wiederrecht, G. P.; Gray, S. K. Inhomogeneous surface plasmon polaritons. ACS Photon. **1**, 739-745 (2014)

[16] Kats, M. A., Blachard, R., Genevet, P. & Capasso, F. Nanometre optical coatings based on strong interference effects in highly absorbing media. Nat. Mat. **12**, 20-24 (2013).

[17] Foley IV, J. J.; Harutyunyan, H.; Rosenmann, D.; Divan, R.; Wiederrecht, G. P.; Gray, S. K. When are Surface Plasmon Polaritons Excited in the Kretschmann-Raether Configuration? Sci. Rep. **5**, 9929 (2015)

[18] Foley IV, J. J.; Ungaro, C.; Sun, K.; Gupta, M. C.; Gray, S. K. Design of emitter structures based on resonant perfect absorption for thermophotovoltaic applications **23**, A1373-A1387 (2015)

[19] Byrnes, S. J. Multilayer Optical Calculations (2018)

https://arxiv.org/abs/1603.02720

[20] https://pypi.org/project/tmm/

[21] Yuffa, A. J.; Scales, J. A. Object-oriented electrodynamic S-matrix code with modern applications, J. Comp. Phys. **20**, 4823-4835 (2012)

[22] https://pypi.org/project/openTMM/

[23] https://kitchenknif.github.io/PyTMM/

[24] https://github.com/ovidiopr/tmmnlay
[25] https://www.mathworks.com/matlabcentral/fileexchange/47637-transmittance-and-reflectance-spectra-of-multilayered-dielectric-stack-using-transfer-matrix-method
[26] Liu, V.; Fan, S. S4: A free electromagnetic solver for layered periodic structures, *Comp. Phys. Comm.* 183, 2233-2244 (2012)
[27] https://web.stanford.edu/group/fan/S4/
[28] https://nanohub.org/resources/s4sim/about
[29] https://github.com/FoleyLab/wptherml/blob/master/Validate_Fresnel.ipynb
[30] https://github.com/FoleyLab/wptherml/blob/master/Validate_Cooling.ipynb
[31] https://github.com/FoleyLab/wptherml/issues

**Copyright Notice**
Authors who publish with this journal agree to the following terms:

Authors retain copyright and grant the journal right of first publication with the work simultaneously licensed under a [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/) that allows others to share the work with an acknowledgement of the work's authorship and initial publication in this journal.

Authors are able to enter into separate, additional contractual arrangements for the non-exclusive distribution of the journal's published version of the work (e.g., post it to an institutional repository or publish it in a book), with an acknowledgement of its initial publication in this journal.

By submitting this paper you agree to the terms of this Copyright Notice, which will apply to this submission if and when it is published by this journal.
Software paper for submission to the Journal of Open Research Software

To complete this template, please replace the blue text with your own. The paper has three main sections: (1) Overview; (2) Availability; (3) Reuse potential.

Please submit the completed paper to: editor.jors@ubiquitypress.com

(1) Overview

Title
WPTherml: A Python Package for the design of materials for harnessing heat.

Paper Authors
1. James F. Varner
2. Noor Eldabagh
3. Derek Volta
4. Reem Eldabagh
5. Jonathan J. Foley IV (corresponding author)

Paper Author Roles and Affiliations
1. Software development, planning, and testing, co-wrote the manuscript and documentation. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
2. Software development and testing. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
3. Software development and testing. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
4. Software development and testing. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470
5. Project conception, software development, planning, and testing, co-wrote the manuscript and documentation. Department of Chemistry, William Paterson University, 300 Pompton Road, Wayne, NJ 07470

Abstract
WPTherml is a Python package for the design of materials with tailored optical and thermal properties for the vast number of energy applications where control of absorption and emission of radiation, or conversion of heat to radiation or vice versa, is paramount. The optical properties are treated within classical electrodynamics via the Transfer Matrix Method which rigorously solve Maxwell's equations for layered isotropic media. A flexible multilayer class connects rigorous electrodynamics properties to figures of merit for a variety of thermal applications, and facilitates extensions to other applications for greater reuse potential. WPTherml can be accessed at https://github.com/FoleyLab/wptherml.
Keywords  
Nanophotonics; Solar Thermophotovoltaics; Passive Cooling; Radiative Cooling; Computational Electrodynamics; Concentrated Solar Power; Transfer Matrix Method

Introduction  
In 2017 over sixty five percent of energy produced was not useable and released as waste heat [1]; consequently, there are a large number of opportunities to develop technologies which can mitigate waste heat associated with energy production, or which can be used to convert or reclaim waste heat for useful purposes. Nano-scale materials, including multi-layered nanostructures among others, with tailored optical and thermal emission properties are centrally important for many of these envisioned technologies, which include solar thermophotovoltaics (STPV) [2-4], radiative cooling devices [5], and highly efficient incandescent lighting [6] that minimize IR losses.

WPTherml (William Paterson University's tool for Thermal Energy and Radiation management with Multi-Layer nanostructures), is a computational engine for materials which can be leveraged for these and other technologies where the control of optical and/or thermal radiation properties is paramount. Specifically, WPTherml can be utilized to simulate optical properties (reflectivity, transmissivity, absorptivity, and emissivity) of nanostructures made from layered isotropic media; thermal radiation spectra of these nanostructures can be derived from these optical quantities in conjunction with Planck’s blackbody radiation law. With these spectral quantities in hand, figures of merit for the design of such multi-layer nanostructures for applications such as STPV, passive radiative cooling, concentrated solar power, incandescent lighting, and plasmonics can be computed. Thus, the WPTherml package serves as an engine for computing spectral quantities, and for relating these spectral quantities to relevant figures of merit. Central to the WPTherml package is a multilayer class; computation of spectral quantities and figures of merit are performed by methods of the multilayer class and are stored as attributes. Spectral quantities are computed within the framework of classical electrodynamics, specifically utilizing the transfer matrix method [7], which is implemented in a library called tmm. The required inputs for the transfer matrix method include the geometry (thickness) and refractive index of each layer in the multi-layer nanostructure, so the multilayer class has an associated method for parsing user input in an intuitive format to specify the geometry and material composition of a multi-layer structure. Refractive index data for a number of common materials is stored and accessed within a data library called datalib. This library also provides access to relevant data for the computation of various figures of merit, for example, spectral response functions of several popular photovoltaic materials (see Figure 1 for an illustration of the spectral response of InGaAsSb), the AM1.5 solar spectrum, the photopic luminosity function, and the atmospheric transmissivity spectrum can all be accessed using the functionality of datalib.
The core of WPTherm is the transfer matrix method, implemented in the *tmm* library, which is a classical electrodynamics approach that relates the incident electromagnetic field upon a multi-layer structure to the reflected and transmitted electromagnetic fields, thereby permitting computation of reflection, transmission, and absorption of light by the multilayer:

\[
\begin{pmatrix}
1 \\
\mathbf{r}
\end{pmatrix} = \begin{pmatrix}
M_{1,1} & M_{1,2} \\
M_{2,1} & M_{2,2}
\end{pmatrix} \begin{pmatrix}
t \\
0
\end{pmatrix},
\]  
(1)

where \(M_{i,j}\) are the elements of the transfer matrix, \(r\) is the reflection amplitude,

\[r = \frac{M_{2,1}}{M_{1,1}} \]  
(2)

and \(t\) is the transmission amplitude,

\[t = \frac{1}{M_{1,1}} \]  
(3)

The reflection and transmission amplitudes may be used to compute the reflectivity, transmissivity, absorptivity, and emissivity of the structure:

\[R = r^*r, \]  
(4)

\[T = t^*t \frac{n_L \cos(\theta_L)}{n_1 \cos(\theta_1)} \]  
(5)

\[A \equiv \epsilon = 1 - R - T, \]  
(6)

where \(R\) denotes the reflectivity, \(T\) the transmissivity, \(A\) the absorptivity, and \(\epsilon\) the emissivity. It should be noted that all quantities above depend on wavelength, and the explicit wavelength dependence of the elements of the transfer matrix have been defined in detail [8]. The core transfer matrix library has been utilized by one of the authors to model thermal emission of multi-layer planar nanostructures [4], and the optical properties of the 2D material known as Borophene [9].

In Equation (6), an important relation known as Kirchoff’s law [2-4] is invoked, which equates the absorption spectrum of a multi-layer structure to its emissivity spectrum. The emissivity spectrum plays a key role in shaping the thermal emission spectrum of a structure at a given temperature,

\[TE(\lambda, T_{ml}) = \rho(\lambda, T_{ml})\epsilon(\lambda), \]  
(7)

where \(TE(\lambda,T)\) denotes the thermal emission, and
\[ \rho(\lambda, T_{ml}) = \frac{2hc^2}{\lambda} \frac{1}{\exp\left(\frac{hc}{\lambda k_B T_{ml}}\right) - 1} \]  

is Planck’s blackbody law, where \( T_{ml} \) denotes the temperature of the multilayer structure, \( \lambda \) denotes the wavelength of light, \( h \) is Planck’s constant, \( c \) is the speed of light, and \( k_B \) is Boltzmann’s constant. In the following, we will illustrate the particular application of WPTherm to solar thermophotovoltaics (STPV), where one tailors the thermal emission spectrum of an object through its emissivity such that it is dramatically different from an ideal blackbody emitter, which has an emissivity of 1 for all wavelengths. Following that discussion, we will briefly summarize several other applications for which WPTherm has been specifically designed to address.

Example application to STPV

**Overview of application**

Management of thermal emission as an energy source, as is done in STPV, may have many beneficial impacts in multiple markets including but not limited to clean energy, military use, and the consumer market [3]. Here we illustrate the use of WPTherm to simulate the thermal emission of a multilayer nanostructure optimized for performance as an STPV emitter; this structure is illustrated in Figure 1 (a). Briefly, the objective is to tune the thermal emission of a material so that it is well matched to the spectral response of a photovoltaic (PV) material. In Figure 1 (b), we illustrate the thermal emission spectrum of the structure when it is heated to 1700 K and overlay it with the spectral response function of an InGaAsSb PV cell, which is commonly used in STPV applications [4], along with the blackbody spectrum at 1700 K. Importantly, the thermal emission of the structure illustrated in Figure 1(a) strongly departs from the blackbody spectrum, and has excellent overlap with the spectral response of InGaAsSb. To further illustrate the suitability of this structure for STPV applications, we plot the product of the thermal emission of this structure with the spectral response of InGaAsSb in Figure 1(c). It is in fact the ratio of the area under this curve, relative to the total area under the thermal emission curve, which plays a critical role in the overall conversion efficiency in a given STPV device (see [8] for more details). An example of code used to generate these figures in WPTherm is included below Figure 1.
**Figure 1:** (a) Example of a multi-layer structure that exhibits selective thermal emission that is nearly optimal for the spectral response of InGaAsSb PVs. (b) The spectral response (in A/W) of InGaAsSb plotted against the thermal emission spectrum of the structure illustrated in (a) and the blackbody spectrum at 1700 K; these thermal emission spectra are scaled by an arbitrary number (what is the number in this case?) to aid visualization against the spectral response function. (c) Plot of the convolution of the spectral response function of InGaAsSb with the thermal emission of the structure in (a) plotted against the thermal emission itself; these two functions are the integrands in that give rise to the short circuit current and the incident power in a STPV system, respectively. See reference [8] for a more detailed discussion of these quantities and their relation to the STPV efficiency.

**Example Code**

```python
from wptherml.wpml import multilayer
from matplotlib import pyplot as plt
from wptherml.datelib import datalib

structure = {
    'Temperature': 1700,
    'Material_List': ['Air', 'Al2O3', 'SiO2', 'TiO2', 'SiO2', 'Al2O3', 'W', 'Air'],
    'Thickness_List': [0, 20e-9, 255e-9, 150e-9, 255e-9, 10e-9, 900e-9, 0],
    'Lambda_List': [300e-9, 6000e-9, 1000],
    'STPV_EMIT': 1
}

stpv_fig = multilayer(structure)

stpv_fig.layer_alloy(1, 0.75, 1.66+0j, 'W', 'MaxwellGarnett')

stpv_fig.fresnel()

stpv_fig.thermal_emission()

SE_InGaAsSb = datalib.SR_InGaAsSb(stpv_fig.lambda_array)

BB = datalib.BB(stpv_fig.lambda_array, stpv_fig.T_ml)

plt.xlabel('Wavelength (nm)', fontsize=14)
plt.xlim(300,4000)
plt.ylim(0,1.5)
```
In the above code example, an instance of the `multilayer` class called `stpv_fig` is created following user input defined in the dictionary called `structure`. The physical structure that this instance models consists of several dielectric layers (aluminium oxide (Al₂O₃), titanium dioxide (TiO₂), and glass (SiO₂)), and an optically-thick tungsten (W) layer, which are defined in the `Material_List` key of the `structure` dictionary. Upon instantiation, the optical and thermal emission properties of this structure at 1700 K (specified by `Temperature`) are computed for 1000 wavelengths between 300 and 6000 nm (specified in `Lambda_List`) and stored as attributes of `stpv_fig`. Additionally, because the key `STPV_EMIT` is set to the value 1, several STPV figures of merit for this structure, including the spectral efficiency, power density, and the TPV efficiency, will be computed at instantiation and stored as attributes the `stpv_fig.spectral_efficiency_val`, `stpv_fig.power_density_val`, and `stpv_fig.tpv_efficiency_val`, respectively. The mathematical details of these figures of merit are discussed in references [4,8].

After instantiation, a modification to one of the layers is made whereby the layer with index 1 (originally Al₂O₃) is changed to an alloy of Al₂O₃ and W with a volume fraction of 75% W; note that in this example, we specify a static refractive index of 1.66+i for the Al₂O₃ layer. The model used to define the wavelength-dependent refractive index of this alloy is known as the Maxwell-Garnett effective medium theory [4], specified by the argument “MaxwellGarnett” (an alternative model known as the Bruggeman model [4] can be invoked with the argument “Bruggeman”). Once this modification is made, the optical properties (reflectivity, transmissivity, emissivity) are recomputed by calling the `fresnel()` method, and the thermal emission spectrum is recomputed with these updated quantities by calling the `thermal_emission()` method.

Also illustrated in this example is how our data library `datalib` can be used to extract relevant data (either stored or computed) including the spectral response function of InGaAsSb (`datalib.SR_InGaAsSb(array_of_wavelengths)`) and the blackbody spectrum at a specific temperature (`datalib.BB(array_of_wavelengths, temperature)`). The remaining code in the example represents standard usage of the python library `matplotlib`, for which extensive documentation already exists [10]. A full list of methods and attributes associated with the `multilayer` class can be found in the WPTherml README documentation [11].
Brief Summary of Additional Applications

The control of optical and thermal emission properties of materials is critical for a number of applications, and here we briefly summarize some of these applications for which WPTherml provides unique capabilities.

Incandescent light bulbs with efficiencies approaching or exceeding that of LED-based lights can be realized by tailoring the thermal emission of filament materials [6]. In particular, the thermal emission should be high in the visible and low in the IR at standard operating temperatures (~2500 K). The key to this property is to tailor the emissivity of the filament material to have strong overlap with the photopic luminosity function, which describes the sensitivity of the human eye to visible light [6]. A figure of merit for incandescent bulbs, known as the luminous efficiency, is related to the convolution of the thermal emission and the photopic luminosity function [6,8], and this figure of merit can be computed using the method `luminous_efficiency()` of the `multilayer` class (see [11] for more details).

Daytime radiative cooling is an application where a material is used to passively cool a space through a net negative radiative power flux from the material’s surface, thereby eliminating or mitigating the need for the use of air conditioning [5]. The key properties of these materials are that they are highly reflective in the solar spectrum and highly emissive in mid-IR wavelengths, specifically wavelengths where the atmosphere is highly transparent (~8-14 microns); Hence, these materials minimize the positive (warming) radiative power flux associated with solar absorption and absorption of thermal radiation from the atmosphere, and maximize the negative (cooling) radiative power flux associated with the material’s thermal emission at its operating temperature. These power fluxes are computed by the method `cooling_power()` in the `multilayer` class. An illustration of the emissivity of such a radiative cooling structure taken from reference [5] is shown in Figure 2, and the code used to make this structure can be found in a Jupyter notebook that ships with our package [12].
Figure 2: (a) Visible absorptivity/emissivity (blue line) of a multi-layered structure optimized for radiative cooling plotted alongside the AM1.5 solar spectrum, illustrating that this structure minimizes absorption of solar power. (b) IR emissivity (red line) of the multi-layered structure optimized for radiative cooling plotted alongside the transmissivity of the earth’s atmospheres, illustrating that this structure maximizes ambient thermal radiation and minimizes absorption of thermal radiation from the atmosphere. Further details of this structure can be found in reference [5], and code to reproduce these figures in WPTherm can be found in a Jupyter notebook that ships with our package [12].

The optical properties of thin metal films has been the subject of intensive investigation owing to their ability to support optical resonances known as surface plasmon polaritons [13-15], and perfectly absorbing modes [16-18]. We have included the capability to analyze multi-layer structures for both surface plasmon polariton and perfectly absorbing modes through the find_spp() and find_pa() methods of the multilayer class, respectively. Mathematical details of these modes can be found in references [8, 13, 16].

Finally, to aid in the user’s intuition for the relation between optical properties, thermal emission, and appearance of a structure, we have implemented methods to render the color of a structure based on its reflectance (ambient_color()) or based on its thermal emission (thermal_color()). The details behind these methods are discussed in reference [8].

Description of alternative packages

There are several open-source software packages that implement the transfer matrix method to simulate the optical response (for example, reflectivity, transmissivity, absorptivity, emissivity) of multi-layer nanostructures; a partial list includes several python-based transfer matrix method packages [19-23], a cython implementation [24], and a Matlab implementation [25]. Related electrodynamics methods, specifically the S-matrix method, have also been implemented in open-source software packages [26-28]. The authors are currently unaware of any alternative software package that leverages such rigorous electrodynamics techniques for
the use in thermal radiation applications; consequently, WPTherml provides a unique and useful capability for a large community of scientists and engineers who aim to design materials and technologies for harnessing heat.

**Implementation and architecture**

WPTherml is written in pure Python 3 with minimal dependencies including numpy, scipy, and matplotlib. WPTherml has been tested with Anaconda Python 3.6 and 3.7 running on Windows 10, Mac OS Sierra, Mac OS Mojave, and Fedora 22. WPTherml includes a *multilayer* class, several function libraries (*tmm, numlib, stpvlib, coolinglib, colorlib, and lightlib*), and a data library (*datalib*). The function libraries contain the functions that perform calculations to extract spectral information and figures of merit for the various applications supported by WPTherml as follows:

- *tmm* leverages the transfer matrix method to compute optical properties (reflectivity, transmissivity, absorptivity, emissivity) of multi-layered nanostructures.
- *numlib* provides a simple function for numerical quadrature, which is used to compute a variety of figures of merit.
- *stpvlib* uses optical properties and thermal emission to compute figures of merit relevant to solar thermophotovoltaic, concentrated solar, and standard photovoltaic applications.
- *coolinglib* uses optical properties and thermal emission to compute figures of merit relevant to radiative cooling.
- *colorlib* uses optical properties and thermal emission to render color of structures, and to classify a spectrum as giving rise to one of the 7 colors in the ROYGBIV spectrum.
- *lightlib* uses optical properties and thermal emission to compute figures of merit relevant for incandescent light sources.
- *datalib* provides data to support various calculations, including:
  - Refractive index as a function of wavelength for an extensive number of common materials; this data is required by the transfer matrix method for computing optical properties.
  - Spectral Response functions for several common photovoltaic materials; this data is required for computing STPV and standard PV efficiencies.
  - AM1.5 spectral data; this data is required for traditional PV efficiency calculations, concentrated solar absorption calculations, and radiative cooling applications.
  - Atmospheric transmission spectrum; this data is required for radiative cooling applications.
  - Spectral response functions for the human eye; this data is required for rendering the colors of structures and for computing the luminous efficiency of incandescent light sources.

The *multilayer* class provides an intuitive framework for connecting user input to the functionality of WPTherml. The user specifies information
about the materials, geometry (thickness), and calculation type to the multilayer class through a dictionary whose keys are descriptive of their meaning with respect to the class. The following code illustrates a simple example where a dictionary called ‘structure’ is defined such that when passed to the multilayer class, it specifies the 3-layer structure illustrated in Figure 3 below:

This needs a header, and also, remove the colon above, because Figure 3 is not immediately below that sentence. Also, make sure you describe what this block of code is in the text.

```python
def multilayer(structure):
    coated_au_film = multilayer(structure)
```

It is important to note the ordering convention used for the ‘Material_List’ and ‘Thickness_List’ keys, whose values are lists of strings and floats that specify the materials and thicknesses of each layer in the multilayer structure, respectively. In particular, these lists are both ordered starting from the side upon which light is incident in a typical reflection experiment and ending in the material into which light would finally be transmitted in the same experiment. Also of note is that both reflected light and thermally emitted light emanate from the top side of structure. All optical and thermal emission spectra will be computed in the range 400 – 800 nm, with resolution of (800-400)/1000 nm as specified by the value of ‘Lambda_List’. A full list of supported keys that can be parsed by the multilayer class is provided in the WPTherml README documentation [11].

![Figure 3: Schematic of the three-layer structure as defined by the ‘structure’ dictionary above, illustrating the convention for the direction of incoming, reflected, transmitted, and emitted light with respect to the order in which materials and thicknesses are specified by the user.](image-url)
A schematic of the flow of data between the user, the multilayer class, and the data and function libraries is illustrated in Figure 4 below.

**Figure 4**: Schematic illustrating the flow of data between the user, the multilayer class, and the data and function libraries of WPTherml. More details can be found in the WPTherml README documentation [11].

**Quality control**

Unit testing was performed for all function libraries, reproducing results from peer-reviewed literature on various thermal applications when applicable. The tmm library was unit tested against the existing python transfer matrix package developed by S. J. Byrnes [19,20], and a system test of WPTherml that is directly comparable to this package can be run by the user through the Jupyter notebook entitled 'Validate_Fresnel' that ships with our package [29]. This test also serves as a check for the stpvlib library. The user may also run a system test of WPTherml that checks the functionality of coolinglib through the Jupyter notebook entitled 'Validate_Cooling' that ships with our package [30].

(2) Availability

**Operating system**

WPTherml has been tested on MacOS Sierra, MacOS Mojave, Windows 10, and Fedora 22. Since WPTherml is written in Python, it should run on any system on which Python and the WPTherml dependencies run.

**Programming language**
WPThermal is written in Python 3 and has been tested with Anaconda Python (3.6 and 3.7).

**Additional system requirements**

The total size of the WPThermal package source code and supporting data stored in datalib is approximately 50 MB; typical execution should not pose any significant memory demands.

**Dependencies**

WPThermal depends on numpy, scipy, and matplotliblib.

The code was tested with
- numpy version 1.14.3
- scipy version 1.1.0
- matplotlib version 2.2.2

**List of contributors**

James Varner; Software development, planning, and testing.
Noor Eldabagh; Software development and testing.
Derek Volta; Software development and testing.
Reem Eldabagh; Software development and testing.
Jonathan J. Foley IV; Project conception, software development, planning, and testing.

**Software location:**

*Archive* (e.g. institutional repository, general repository) (required – please see instructions on journal website for depositing archive copy of software in a suitable repository)

- **Name:** SourceForge
- **Persistent identifier:** https://sourceforge.net/projects/wptherml
- **Licence:** GPL 3.0
- **Publisher:** Jonathan Foley
- **Version published:** v1.03 beta
- **Date published:** 03/22/2019

**Code repository**

- **Name:** GitHub
- **Identifier:** https://github.com/FoleyLab/wptherml
- **Licence:** GPL 3.0
- **Date published:** 03/04/2019

**Emulation environment** (if appropriate)

- **Name:** The name of the emulation environment
- **Identifier:** The identifier (or URI) used by the emulator
**Licence:** Open license under which the software is licensed here

**Date published:** dd/mm/yy

**Language**

Python 3

**Reuse potential**

WPTherm was developed to support the simulation and design of a practically endless number of structures for multiple applications where optical and thermal emission properties are paramount. The package was also developed to be easily extensible to support new applications as the developers or users see fit. Several potential applications are illustrated in Jupyter notebooks that ship with the package, and the developers plan to continue to add and refine these examples over time. Users seeking support for additional features can communicate with us through the issues page of the GitHub repository [31], or may contact the corresponding author by email at foleyj10@wpunj.edu.

**Acknowledgements**

The authors wish to acknowledge support from the College of Science and Health and the Department of Chemistry at William Paterson University, specifically Dean Kenneth Wolf, Dean Venkat Sharma, Prof. Bhanu Chauhan, and Dr. Mukesh Sahni, for their vision and persistence in realizing the computational infrastructure that allowed the development of WPTherm. J. F. V. acknowledges support from the Louis Stokes Alliance for Minority Participation, and J. J. F. acknowledges support from the ART program at William Paterson University.

**Funding statement**

**Competing interests**

The authors declare that they have no competing interests.

**References**

[1] https://flowcharts.llnl.gov/commodities/energy

[2] Gupta, M. C.; Ungaro, C.; Foley IV, J. J.; Gray, S. K. Optical nanostructures design, fabrication, and applications for solar/thermal energy conversion. *Solar Energy* **165**, 100-114 (2018)

[3] Zhou, Z.; Sakr, E.; Sun, Y.; Bermel, P. Solar thermophotovoltaics: reshaping the solar spectrum. *Nanophotonics* **5**, 1-21 (2016)

[4] Jeon, N.; Hernandez, J. J.; Rosenmann, D.; Gray, S. K.; Martinson, A. B. F.; Foley IV, J. J. Pareto Optimal Spectrally Selective Emitters for Thermophotovoltaics via Weak Absorber Critical Coupling. *Adv. Energy Mater.* **8**, 1801035 (2018)

[5] Raman, A. P.; Anoma, M. A.; Zhu, L.; Rephaeli, E.; Fan, S. Passive radiative cooling below ambient air temperature under direct sunlight. *Nature* **515**, 540-544 (2014)
[6] Ilic, O.; Bermel, P.; Chen, G.; Joannopoulos, J. D.; Celanovic, I.; Soljačić, M. Tailoring High Temperature Radiation and the Resurrection of the Incandescent Source. Nat. Nanotechnol. 11, 320-324 (2016)
[7] Yeh, P. Optical Waves in Layered Media (Wiley, New York) 1988
[8] https://github.com/FoleyLab/wptherml/blob/master/documentation/Equations.pdf
[9] Adamska, L.; Sadasivam, S.; Foley IV, J. J.; Darancet, P.; Sharifzadeh, S. First-principles investigation of Borophene as a monolayer transparent conductor. J. Phys. Chem. C 122, 4037-4045 (2018)
[10] https://matplotlib.org/contents.html
[11] https://foleylab.github.io/wptherml/
[12] https://github.com/FoleyLab/wptherml/blob/master/Validate_Cooling.ipynb
[13] Raether, H. Surface Plasmons on Smooth and Rough Surfaces and on Gratings (Springer-Verlag., 1988)
[14] Maier, S. M. Plasmonics: Fundamentals and applications (Springer., 2007).
[15] Foley IV, J. J.; McMahon, J. M., Schatz, G. C.; Harutyunyan, H.; Wiederrecht, G. P.; Gray, S. K. Inhomogeneous surface plasmon polaritons. ACS Photon. 1, 739-745 (2014)
[16] Kats, M. A., Blachard, R., Genevet, P. & Capasso, F. Nanometre optical coatings based on strong interference effects in highly absorbing media. Nat. Mat. 12, 20–24 (2013).
[17] Foley IV, J. J.; Harutyunyan, H.; Rosenmann, D.; Divan, R.; Wiederrrecht, G. P.; Gray, S. K. When are Surface Plasmon Polaritons Excited in the Kretschmann-Raether Configuration? Sci. Rep. 5, 9929 (2015)
[18] Foley IV, J. J.; Ungaro, C.; Sun, K.; Gupta, M. C.; Gray, S. K. Design of emitter structures based on resonant perfect absorption for thermophotovoltaic applications 23, A1373-A1387 (2015)
[19] Byrnes, S. J. Multilayer Optical Calculations (2018) https://arxiv.org/abs/1603.02720
[20] https://pypi.org/project/tmm/
[21] Yuffa, A. J.; Scales, J. A. Object-oriented electrodynamic S-matrix code with modern applications, J. Comp. Phys. 20, 4823-4835 (2012)
[22] https://pypi.org/project/openTMM/
[23] https://kitchenknife.github.io/PyTMM/
[24] https://github.com/ovidior/tmmnlay
[25] https://www.mathworks.com/matlabcentral/fileexchange/47637-transmittance-and-reflectance-spectra-of-multilayered-dielectric-stack-using-transfer-matrix-method
[26] Liu, V.; Fan, S. S4: A free electromagnetic solver for layered periodic structures, Comp. Phys. Comm. 183, 2233-2244 (2012)
[27] https://web.stanford.edu/group/fan/S4/
[28] https://nanohub.org/resources/s4sim/about
[29] https://github.com/FoleyLab/wptherml/blob/master/Validate_Fresnel.ipynb
[30] https://github.com/FoleyLab/wptherml/blob/master/Validate_Cooling.ipynb
[31] https://github.com/FoleyLab/wptherml/issues

Copyright Notice
Authors who publish with this journal agree to the following terms:
Authors retain copyright and grant the journal right of first publication with the work simultaneously licensed under a Creative Commons Attribution License that allows others to share the work with an acknowledgement of the work's authorship and initial publication in this journal.

Authors are able to enter into separate, additional contractual arrangements for the non-exclusive distribution of the journal’s published version of the work (e.g., post it to an institutional repository or publish it in a book), with an acknowledgement of its initial publication in this journal.

By submitting this paper you agree to the terms of this Copyright Notice, which will apply to this submission if and when it is published by this journal.
