

## Cloud Computing Simulation Tools for Nanophotonics at nanoHUB.org

L. J. Prokopeva,<sup>1</sup> U. Guler,<sup>1</sup> X. Ni,<sup>1</sup> J. Fang,<sup>1</sup> R. Chandrasekar,<sup>1</sup> V. P. Drachev,<sup>1</sup> V. Shalaev,<sup>1</sup>  
Z. Liu,<sup>1,4</sup> N. Arnold,<sup>2</sup> T. Klar,<sup>2</sup> A. O. Korotkevitch,<sup>3</sup> Z. B. Wang,<sup>5</sup> B. Luk'yanchuk,<sup>6</sup>  
G. B. Adams III,<sup>7</sup> and A. V. Kildishev<sup>1</sup>

<sup>1</sup>Birck Nanotechnology Center, School of ECE, Purdue University, West Lafayette, IN 47907, USA  
kildishev@purdue.edu

<sup>2</sup>Institute of Applied Physics, Johannes Kepler University Linz, Austria

<sup>3</sup>Department of Mathematics and Statistics, University of New Mexico, USA

<sup>4</sup>Institute of High Performance Computing, A\*STAR, Singapore

<sup>5</sup>The Bangor University, Bangor, Gwynedd LL57 2DF, UK

<sup>6</sup>Data Storage Institute, A\*STAR, Singapore

<sup>7</sup>Network for Computational Nanotechnology, Purdue University, West Lafayette, IN 47907, USA

**Abstract:** Here we review our progress in developing simulation tools for nanophotonics, staged at [www.nanoHUB.org](http://www.nanoHUB.org). Publishing at nanoHUB is a fast and effective way to deliver a scientific application as a cloud computing service; we focus at several practical aspects of this process. Our on-line tools provide electromagnetic and multiphysics simulations of planar, circular and spherical multilayered nanophotonic devices. Updated and totally new tools utilize an original GUI paradigm developed in Matlab with fast GUI integration, power and utility of Matlab syntax and components, integration with the material database, two-parameter sweep and other useful features. In particular, we introduce two new tools – *PhotonicsPOS* (scattering solutions for a multi-shell spherical particle on a planar multi-layered substrate) and *PhotonicsGAIN-0D* (simulations of active media described by a four-level atomic system).

**Keywords:** Electromagnetics, Cloud Computing, Photonics, Plasmonics

### 1. Introduction

Graphical User Interface (GUI) development and provision with public/shared access for colleagues/collaborators/community to a final software application are two inevitable obligatory stages of every scientific code development. Those stages, however, are not always accomplished in research groups due to the significant time and technical effort that it may require. We solve these problems by (a) using Matlab to wrap the tool with a GUI, (b) developing a common GUI paradigm and components to be used in every one of our applications, and (c) using [www.nanoHUB.org](http://www.nanoHUB.org) for sharing and publishing a final tool. These choices allow for quick packaging and delivery times of our scientific programming codes. The first part of this paper discusses general aspects of scientific code packaging stages (a-c) pertinent to our simulation tool set for nanophotonics.

Nanophotonics is a science and art of the extreme control of light leading to many revolutionary optical devices, breaking the diffraction limit of conventional optics, and going beyond classical

electromagnetic theory, e.g. [1, 2]. This powerful control of light is achieved with the use of modern nanotechnology, capable of producing artificial nanostructured materials with tailored optical properties (metamaterials). Needless to say, effective and accurate software plays a huge role in (i) modeling the propagation of light in passive and active dispersive metamaterials and (ii) optimizing the designs of optical devices and material patches. It is also important to deliver the fully packaged simulation software to experimentalists and theorists in a very short time. Currently we have six online tools [3-8] at nanoHUB to simulate light propagation in planar and circular nanostructured optical materials and devices. Two other new simulation tools will appear soon at the same web portal: *PhotonicsPOS* – simulating scattering for a multi-shell spherical particle on a planar multi-layered substrate and *PhotonicsGAIN-0D* – numerical analyses of local response and amplification capability of active media described by a four-level atomic system. All eight tools are reviewed in the second part of the paper.

## 2. NanoHUB.org – a Perfect Way to Publish Nanotechnology-Related Software

### A. GUI Programming Tools

Selecting the right programming language and tool is of utmost importance – specifically, one that minimizes GUI development times while fulfilling all the custom requirements. There are several reasonable options for time-limited GUI development for scientific codes, listed in increasing order of flexibility: Rappature (specifically designed by nanoHUB.org for making scientific computing GUIs), Matlab (using GUIDE or plain scripting), and some C-based GUI API.

Rappature [9] is open source software available at [www.rappature.org](http://www.rappature.org) that is an extremely fast way to wrap simple scientific code with a quality GUI and stage it online at [www.nanohub.org](http://www.nanohub.org) with minimal additional coding. The developer needs only to provide a Rappature-specific XML description of parameters, and the GUI is created automatically. A rich collection of GUI components is complemented with built-in units support as well as recording/comparing results of previous simulations. The drawbacks are that (i) only simple GUIs can be managed and (ii) Rappature links to third party libraries (e.g. tcl, vtk), making it tedious to install in custom environments different from the nanoHUB Linux workspace. Our nanophotonics tools need nontrivial dynamical GUIs to instantly create nanostructures and access the material database, and cannot be managed by Rappature.

C-based GUI APIs does give flexibility and cross-platformity for the supported environments/OS with minor building/linking hassle. However, in practice, experience with the chosen programming tool is required and even then development time is considerable. This is a good option if GUI development could be outsourced from the research group and there are appropriate time scales for a final product.

Matlab is a good compromise between the development time and GUI flexibility with some tricks and effort needed to program the most dynamical GUI components. The final application can be easily compiled (in Windows, Linux, Mac OS) by using ‘*deploytool*’ in Matlab, and it gets all the power and utility of Matlab syntax and components, including matrix manipulation, numerical functions, and plotting components. Our previous tools were written with Qt and are now being updated with a new Matlab GUI paradigm and its new features (see below). Our simple tools [3, 4] are done using Rappature.

### B. Common Modules for Nanophotonics in Matlab

Below we list several important aspects and key graphical elements of our new GUI framework.

Access to material database. Our material database is a collection of optical properties for different materials, which includes tabulated experimental data and theoretical models (e.g., the Drude-Lorentz model, Debye model, critical points model, Sellmeier model). Interfaces were written for (1) indexing of newly added materials, (2) quick incremental search of materials in database, (3) permittivity calculation for theoretical models and permittivity interpolation for tabulated data. In Fig. 1, we show developed graphical component for instant incremental search in the database. Search options include (1) search area – within the entire label or within the chemical formula, (2) location of the searched pattern – anywhere within the search area or at the beginning, and (3) case sensitivity.

History and import/export of input files. All tools support XML import/export of all input parameters, so that parameter sharing and returning backs to a specific snapshot of the GUI are easy. The application

will also automatically backup the entire input panel each time the ‘solve’ button is pressed.

Saving previous simulation results in cache. The results of all simulations are stored in cache, and recalculation (possibly time consuming) of same structures is avoided. Thus our simulation tools provide a cloud-based storage of the simulation results and their visualization. Cached data can be kept privately or within a nanoHUB group. Another feature of our software is that all figures are undockable with a standard Matlab menu bar and toolbar for adjustments and export in image or raw data formats.

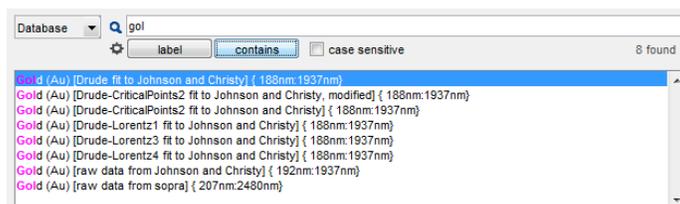


Figure 1. A graphical component for instant incremental search in Database.

Two-parameter sweep. Design optimization of nanophotonics devices usually requires a sweep over a range of some parameter(s). E.g., the previous Qt version of the SHA-2D GUI solver allowed only for spectral (wavelength) sweep. With the new GUI framework a user can set two arbitrary parametric vectors (denoted as p1 and p2, as can be seen in Fig. 2), and then assign p1 and p2 (or their Matlab-compatible mathematical expressions) to any physical parameter defining the model. Right clicking on any edit box brings the table of actual values in the range. This way we can analyze for example (1) spectral behavior, (2) angular dependency, (3) geometry variations of the structure, or any of those two at a time. Also this option can be used to set a custom dispersive dielectric function of a material (beyond those available in the material database). If both parameters are vectors, then a two-dimensional parametric map of results will be provided for analyses by the end of calculations. The default sweep regime is called ‘All-to-All’, for which parametric space is formed by all possible p1-p2 pairs taken within their ranges. The other regime is available by toggling the regime button to ‘One-to-One’, and it is used to set up a sequence of 1-1 pairs (number of elements in p1 and p2 must be equal). ‘One-to-one’ format provides flexible distribution of points in the parametric space, which may give a coarse parametric map in less time. This regime is suitable for real experimental and optimization techniques, which are usually formed as a sequence of parameter sets.

Two-listbox compact interface for structure creation. We developed a dynamic and compact two-listbox interface to easily create the geometries and dynamically visualize them (Fig. 2). The functional buttons on top of the listboxes include ‘create’, ‘remove’, ‘duplicate’, ‘edit’, ‘move-down’, and ‘move-up’ operations. The structure view optionally shows the currently selected segment and incident angle of the field. The colors of the structure view can be set according to user preferences.

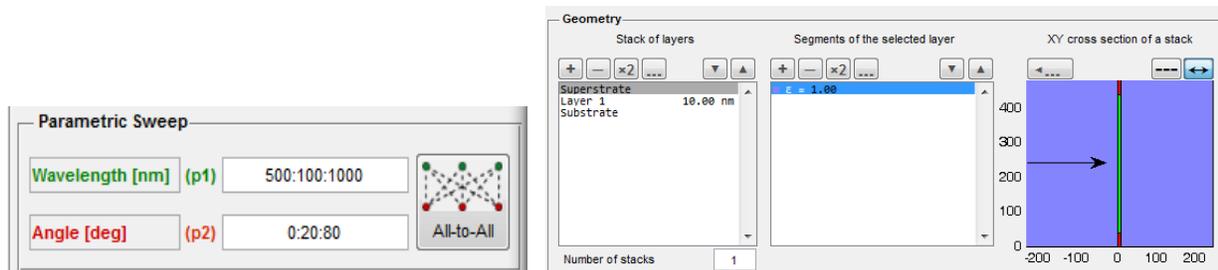


Figure 2. Parametric sweep set-up panel and two-panel interface example for PhotonicsSHA-2D.

### C. Sharing and Publishing the Final Application

For free sharing and open/restricted publishing of final software relating to nanotechnology we encourage using the professional web portal [www.nanohub.org](http://www.nanohub.org), created to gain the nanotechnology knowledge base from professionals all over the world. NanoHUB cloud service is a key resource for

research and education in the major areas of nanoelectronics, nanoelectromechanical systems, and their application to nano-biosystems. In particular, it provides quick staging of installation-free in-browser applications with all the benefits of cloud services. Other useful features on the nanoHUB portal include (1) simulation session sharing and discussion groups for effective collaboration and (2) application supporting documents, tutorials and courses available for uploading and viewing on-site. The nanoHUB project is software-wise and hardware-wise supported by the Network for Computational Nanotechnology (NCN), which is located at the Birck Nanotechnology Center at Purdue University.

### 3. Nanophotonic Tools

Our currently published nanoHUB tools (see [10] for details) already provide a set of free cloud computing services for simulation of light propagation in nanophotonics devices and metamaterials of different shapes. These tools allow for approximating and comparing the frequency dispersive optical properties of various materials (PhotonicsDB), simulations of (a) planar nanostructured lamellar photonic crystals (PhotonicsRT), (b) cascaded photonic or plasmonic 2D metamaterials (PhotonicsSHA-2D), and (c) cylindrical multilayered transformation optics lenses (PhotonicsCL, Hyperlens Layer Designer and Hyperlens Design Solver), see Fig. 3. The tools previously written with Qt are now being updated with a new GUI paradigm and will support all new features listed earlier. Moreover, PhotonicsSHA-2D is now being updated with a more accurate eigenvalue engine, see [11] for details.

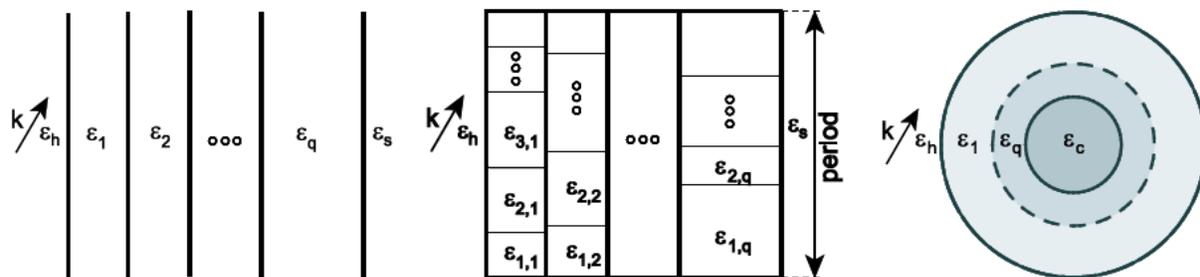


Figure 3. Nanostructures supported in PhotonicsRT, PhotonicsSHA-2D, and PhotonicsCL.

PhotonicsPOS (Particle on Substrate) tool is designed to provide scattering solutions for a multi-shell spherical particle located over a planar multi-layered substrate, one of the most exciting cases for a wide range of applications from nanotechnology to defense technologies. Scattering of light from spherical particles was studied widely on or over the last century since Gustav Mie provided the solutions for the problem [12]. Studies with particular approximations such as particle in quasistatic limit and substrate with high conductivity were published recently on the subject [13-16]. In 1986 Bobbert and Vlioger presented the analytical solution for the problem and worked out the results for a simplified case, specifically the far field scattered field [17]. In 2004, Wang and Luk'yanchuk presented results in the near field region where analytical solutions were used in order to obtain field distribution [18].

In this work, we present a tool based on the previous particle on a substrate studies with further modifications such as a multi-shell particle and a multi-layered substrate. Furthermore, several modifications were made to the standard Mie solution in order to account for the reflection of scattered spherical waves from the planar substrate as well as for certain application-related scenarios. The GUI of our tool has an input panel to set the geometry and incident field and an output panel providing parametric map for cross section efficiencies and a field map for the selected sweep parameters.

PhotonicsGAIN-0D (four-level gain media tool) provides time-domain simulations of 4-level atomic gain systems and post-processing analyses of the amplification process, see [19] for details. The development of the PhotonicsGAIN-0D solver has been motivated by the problem of achieving loss compensation in plasmonic elements of optical metamaterials by introducing organic dyes into a dielectric host of the metamaterial structure [20-22], in which case light amplification is based on the stimulated radiation emission from the dye-doped host, a phenomenon broadly used in laser science [23].

In this tool instead of modeling full-wave pulse propagation through a thick gain medium, we study the local response (at a fixed spatial point) of the gain medium to a given sequence of pump and probe pulses of external time-dependent electric field. We calculate the population kinetics on all energy levels (pumping process, inversion, and then relaxation) and the effective polarizability (negative imaginary component means amplification), Fig. 4. The tool helps to study, validate, and fit to experiment a simplified 4-level atomic model of a given gain medium. Real gain systems may involve a large number of energy levels, but many of them can be reduced to the 4-level system with similar essential features and semi-quantitative behavior. For this reason we hope that the tool is helpful for different applications of gain media beyond active metamaterials, e.g. in organic dye lasers, and organic dye spectroscopy.

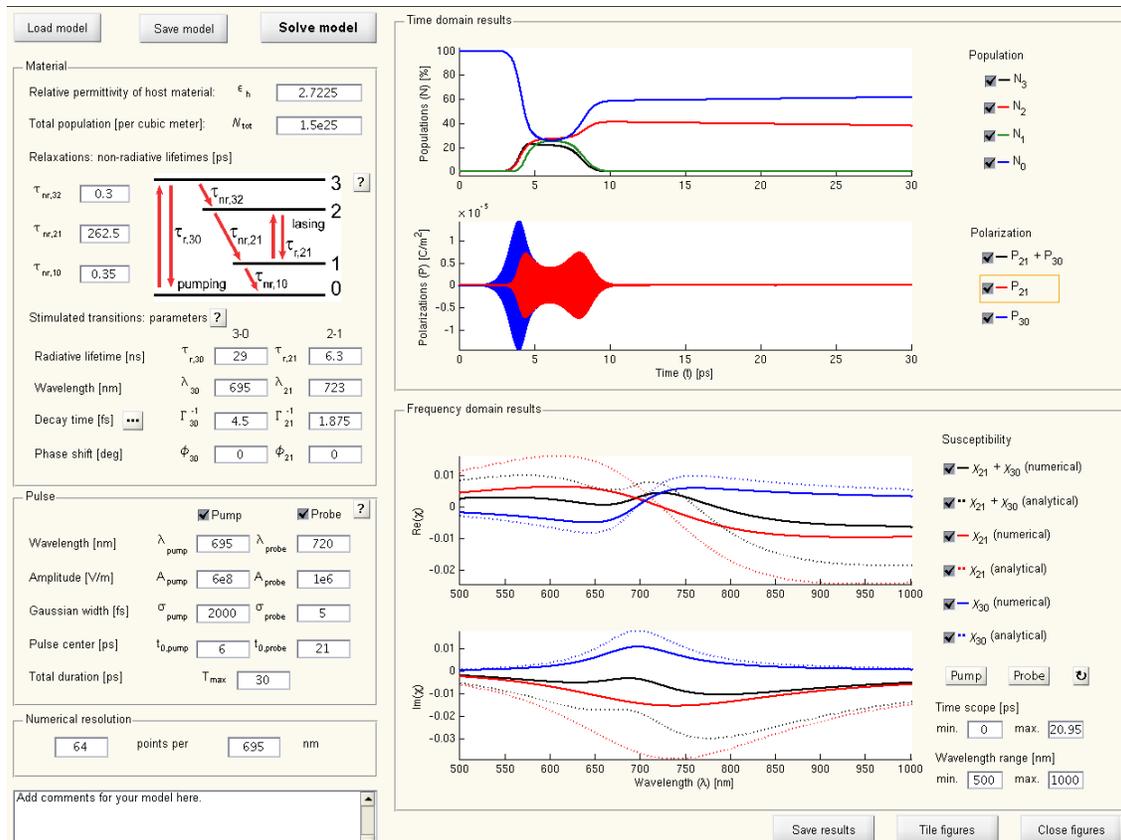


Figure 4. GUI of PhotonicsGAIN-0D – 4-level gain system response to pump-probe pulse sequence.

#### 4. Conclusions

In this paper, we have discussed general aspects of GUI development for scientific programming codes. We have reviewed our simulation tools for nanophotonics at nanoHUB.org and have presented two new online tools: PhotonicsPOS (scattering solutions for a multi-shell spherical particle on a planar multi-layered substrate) and PhotonicsGAIN-0D (simulations of active media described by a four-level atomic system).

Purdue authors gratefully cite support of different parts of this effort by ONR-MURI N000014-10-1-0942, ARO-MURI 50342-PH-MUR, W911NF-09-1-0539, AFOSR-MURI FA9550-12-1-0024, ARO 57981-PH (W911NF-11-1-0359). AVK and LJP were also partially supported by the AFRL Materials & Manufacturing Directorate Applied Metamaterials Program. NA and TAK acknowledge funding by ERC Starting Grant 257158 “ActiveNP”. LP was partially supported by RFFI grant 12-01-00844-a.

## References

- [1] A. V. Kildishev et al., "Materializing a Binary Hyperlens Design," *Appl. Phys. Lett.*, vol. 94, no. 7, pp. 0711021-0711023, 2009.
- [2] A. V. Kildishev et al., "Negative Refractive Index in Optics of Metal-Dielectric Composites," *J. Opt. Soc. Am. A*, vol. 23, no. 3, pp. 423-433, 2006.
- [3] X. Ni et al., "PhotonicsDB: Optical Constants," *NanoHUB.org*, <http://nanohub.org/resources/photonicscdb>, 2008.
- [4] S. Ishii et al., "PhotonicsRT: Wave Propagation in Multilayer Structures," *NanoHUB.org*, <http://nanohub.org/resources/photonicsrct>, 2008.
- [5] X. Ni et al., "PhotonicsSHA-2D: Modeling of Single-Period Multilayer Optical Gratings and Metamaterials," *NanoHUB.org*, <http://nanohub.org/resources/sha2d>, 2009.
- [6] M. Swanson et al., "Hyperlens Layer Designer," *NanoHUB.org*, <http://nanohub.org/resources/hypiedesigner>, 2008.
- [7] M. Swanson et al., "Hyperlens Design Solver," *NanoHUB.org*, <http://nanohub.org/resources/hypiesolver>, 2008.
- [8] X. Ni et al., "PhotonicsCL: Photonic Cylindrical Multilayer Lenses," *NanoHUB.org*, <http://nanohub.org/resources/photonicscl>, 2011.
- [9] M. McLennan and R. Kennell, "HUBzero: A Platform for Dissemination and Collaboration in Computational Science and Engineering," *Comput. Sci. Eng.*, vol. 12, no. 12, pp. 48-52, 2010.
- [10] X. Ni et al. "Cloud Computing for the Science of Light: Nanophotonics Tools at nanoHUB.org," *the 27th Annual Review of Progress in Applied Computational Electromagnetics*, pp. 63-68, 2011.
- [11] A. Korotkevich et al. "Fast Eigensolver for Optical Metasurfaces," *the 29th Annual Review of Progress in Applied Computational Electromagnetics*, 2013 (submitted).
- [12] G. Mie, "Beiträge zur optik trüber medien, speziell kolloidaler metallösungen," *Ann. Phys.*, vol. 330, no. 3, pp. 377-405 (1908).
- [13] G. Videen, "Light Scattering from a Sphere on or near a Surface," *J. Opt. Soc. Am. A*, vol. 8, no. 3, pp. 483-489, 1991.
- [14] G. Videen, "Light Scattering from a Sphere behind a Surface," *J. Opt. Soc. Am. A*, vol. 10, no. 1, pp. 110-117, 1993.
- [15] G. Videen et al., "Scattering from a Small Sphere near a Surface," *J. Opt. Soc. Am. A*, vol. 10, no. 118, pp. 118-126, 1993.
- [16] T. Wriedt and A. Docu, "Light Scattering from a Particle on or near a Surface," *Opt. Comm.*, vol. 152, pp. 376-384, 1998.
- [17] P. Bobbert and J. Vlioger, "Light Scattering by a Sphere on a Substrate," *Physica A*, vol. 137, pp. 209-242, 1986.
- [18] Z. B. Wang et al., "Angle Effect in Laser Nanopatterning with Particle-Mask," *J. Appl. Phys.*, vol. 96, no. 11, pp. 6845-6850, 2004.
- [19] N. Arnold et al., "Modeling the Local Response of Gain Media in Time-Domain," *the 29th Annual Review of Progress in Applied Computational Electromagnetics*, 2013 (submitted).
- [20] J. Trieschmann et al., "Experimental Retrieval of the Kinetic Parameters of a Dye in a Solid Film," *Opt. Express*, vol. 19, no 19, pp. 18253-18259, 2011.
- [21] S. Wuestner et al., "Gain and Plasmon Dynamics in Active Negative-Index Metamaterials," *Phil. Trans. R. Soc. A*, vol. 369, pp. 3525-3550, 2011.
- [22] T. A. Klar et al., "Negative-Index Metamaterials: Going Optical," *IEEE J. Sel. Top. Quantum Electron.*, vol.12, no. 6, pp. 1106-1115, 2006.
- [23] A. E. Siegman, *Lasers*, University Science Books, Sausalito, CA, 1986.