Loss engineering expansion for MPB

This is the 2D version of the free MPB extension code for the computation of the bandstructure, group index and propagation loss of photonic crystal (PhC) slab waveguides. The effective period method is used to approximate the behaviour of the 3D structure, as outlined in reference [1]. Any 2D simulation of a 3D structure is always an approximation and therefore we recommend that this approach issued for larger parameter sweeps, with final results validated using either 3D simulations or experimental results.

It consists of several parts. First a control file for the free MIT Photonics-Bands (MBP) package, written in scheme, to calculate the electromagnetic modes and dispersion curve of the photonic crystal waveguide as well as calculating the out-of-plane and back-scattering coefficients as outlined in reference [2]. The additional files are for data analysis and consist of Linux shell and Matlab scripts. All files are contained in Loss_engineering_2D.zip.

All scripts can be downloaded under the free GPL license, and no installation is required. However, an installed version of MPB (available free from: HTTP://ab-initio.mit.edu/wiki/index.php/MPB) is required. Additionally, Matlab is required for the (optional) analysis tools.

When publishing results obtained with this software, please cite references [1,2]. Furthermore we suggest a statement similar to the exemplary one below in the text of your manuscript:
"Simulations were performed using the effective period method [1], with the propagation loss modeled according to reference [2]."

Features

Efficient 2D simulation (approx. 100 times faster than 3D)
Calculation of propagation loss for different line defect waveguides

1. Prebuild support for dispersion engineering, using a variety of designs [3]

- Simultaneous calculation of bandstructure, field distributions, group index curves and propagation loss possible.
- Free software under the GNU General Public License.

The calculation is time efficient and a 2D computation of the bandstructure, the group velocity curve, as well as the out-of-plane and back-scattering coefficients for a slow light waveguide on an Intel i-7 desktop (2.6GHz), using mpb-split 8, over 51k-points using 16x16x30 grid is finished in approximately 1 min and uses 100MB of RAM.

Contact info

For support using this script or suggestion for improving it please contact Sebastian Schulz (sschulz@uottawa.ca or sebastianandreasschulz@gmail.com) alternatively Sebastian Schulz also follows discussions on the mpb-discuss mailing list.

For general problems or questions regarding MPB, please consult the MPB Documentation (HTTP://ab-initio.mit.edu/wiki/index.php/MPB_manual), the mpb-discuss archives or the mpb-discuss mailing list.

References

[1] S.A.Schulz, A.H. K. Park, I. De Leon, J. Upham and R. W. Boyd "Beyond the effective index method: improved accuracy for 2D simulations of photonic crystal waveguides", Journal of Optics **17**, 075006 (2015).

[2] L.O'Faolain, S. A. Schulz, D. M. Beggs, T. P. White, M. Spasenović, L. Kuipers, F. Morichetti, A. Melloni, S. Mazoyer, J. P. Hugonin, P. Lalanne and T. F. Krauss "Loss engineered slow light waveguides", Optics Express **18,** 27627-27638 (2010).

[3] S.A.Schulz, L. O. Faolain, D. M. Beggs, T. P. White, A. Melloni and T. F. Krauss "Dispersion engineered slow light in photonic crystals: a comparison", Journal of Optics **12,** 104004 (2010).

How to run 2D photonic crystal slab waveguide calculations with mpb using the effective period approximation

Sebastian Schulz
sschulz@uottawa.ca, sebastianandreasschulz@uottawa.ca

Background

If you have not used MPB before, please read the MPB manual:
`http://web.mit.edu/meep_v1.0/mpb/doc/index.html` or
`http://ab-initio.mit.edu/wiki/index.php/MPB_manual`

We have prepared a script to make 2D simulations of Photonic crystal waveguides using the effective period method easy. The script, which is called `W1_2D_v1.ctl`, is implemented using the MIT Photonics Band software (MPB), with the propagation loss calculated according to reference [1]. The scripts are made available under the the free GPL license. If you use this script, please cite references [1] and [2] (and the reference for MPB, as stated on its website, link at the end of this document) and include a statement similar to the following one in your text:
"Simulations were performed in MPB using the effective period method [1], with the propagation loss modelled according to reference [2]."

MPB works by defining a unit cell and using periodic boundary conditions. It extends this unit cell infinitely in all directions. It then uses a plane-wave method for calculating the eigenfrequencies of the defined structure for any values of the Bloch wavevector **K** that you ask for.

In the case of a photonic crystal slab waveguide, it is convenient to define a rectangular unit cell as shown in figure 1a. The unit cell is indicated by the grey boxes, and both in-plane and out-of-plane directions are shown. As the only true direction of periodicity that we wish to consider is along the waveguide (the *x*-direction), we must make the unit cell large enough in the *y*-direction such that the fields confined to the waveguide don't overlap with the artificially created waveguides in neighbouring supercells.
The vertical behaviour is approximated using the effective period method and therefore the vertical slab height is set to infinity, resulting in a z-invariant structure.

Also, as we are only interested in the direction along the waveguide (the truly periodic direction, here along the *x*-axis), the we can confine our bandstructure calculations to this direction only – i.e. we take the wavevectors $\mathbf{K}=(K_x, 0, 0)$. By default the script considers values of K_x to be 0.3, 0.31, 0.32, …, 0.49, 0.5. Note that these are dimensionless values in units of $a/2\pi$, such that $K_x=0.5$ is at the bandedge (i.e. the edge of the Brillouin Zone). This can be altered using the parameters `Ks`, `Ke` and `Kinterp`. `Ks` defines the start point, `Ke` defines the end

point, and `Kinterp` how many values are interpolated between the start and end points. See later for how to pass command line arguments.

To run the script just type one of the commands:

```
mpb W1_2D_v1.ctl > yourfile.out
mpb-split 4 W1_2D_v1.ctl > yourfile.out
```

The first runs a single process of MPB, calculating at each K-value in turn. With normal parameters, each K-value should take around 10s to run, and use around 15MB of RAM. MPB supports multicore processors, so the second command takes advantage of this by splitting the K-values to be calculated across the different processors and aggregating the results at the end. Please not that our script does not currently work with MPI.

These will run all the default parameters (a silicon W1 membrane photonic crystal waveguide) and place the output in `yourfile.out`.

To reformat the output and extract your results, use the commands below:
To extract the bandstructure:
```
awk -f bsfiles_2D.awk yourfile.out > yourfile_bs.dat
```

To extract the group index (assuming you have calculated the group velocities):
```
awk -f ngfiles_2D.awk yourfile.out > yourfile_ng.dat
```

The files `bsfile_2D.awk`, `ngfiles_2D.awk` and `extract_coefficients.sh` can be found in the same location as `W1_2D_v1.ctl`. The file `bsfiles_2D.awk` searches for line with the phrase "tefreqs" and prints out the **K**-vectors (actually the value of K_x) and the frequencies (in dimensionless units of a/λ) of all the calculated bands. The file `ngfiles.awk` searches for the lines with "tevelocities" and prints out the **K**-vector, frequency and x-component of the group velocity of the band of interest (in usual cases b=23 in the file).

If you want to change the default parameters, you can do so at the command line. The format of the commands is (for example):

```
mpb r0=0.35 W1_2D_v1.ctl > yourfile.out
```

which alters the size of the circles from 0.3 (the default) to 0.35. As many of these arguments as you like can be passed, so

```
mpb r0=0.35 r1=0.29 r2=0.34 W1_2D_v1.ctl > yourfile.out
```

alters the sizes of the circles in the first row to 0.29, those in the second to 0.34 and the rest to 0.35. Note when using mpb-split that the number of processors should be the first parameter passed, e.g.

```
mpb-split 8 r0=0.35 r1=0.29 r2=0.34 W1_2D_v1.ctl > yourfile.out
```

The size and positions of the first three rows of holes can be altered by using the parameters `r1`, `r2`, `r3`, `s1`, `s2`, `s3`, `p1`, `p2`, and `p3`. The r-parameters adjust the radius of the holes (see fig. 1b). The s-parameters shift the hole positions away from the waveguide (see fig. 1c). The p-parameters shift the hole positions along the waveguide (see fig. 1d). These parameters are entered as fractions of the lattice constant (either `a1` or `a2` – see below). Also, the width of the waveguide can be altered by using the `w` parameter, which sets the width of the central waveguide to be `w*a*sqrt(3)`; see figure 1e. Thus, setting `w=1` gives a W1 waveguide, and other values give a so-called Ww waveguide. Note, however, that in this definition a W2 waveguide is NOT two rows of holes removed.

Usually all parameters can be entered as dimensionless ones, by parametrising them in some or other units of *a*, the lattice constant. However, in real-life situations, the thickness of the slab that the waveguides are fabricated in is fixed – i.e. it does not scale with the lattice constant, which is determined by your in-plane design. Therefore the lattice constant is entered as real values in nm using the parameters `a1nm`. Internally it is converted into the dimensionless value `a1` by dividing them by the value you enter for `a1nm`. The lattice can be "stretched" (or "squashed", see fig. 1f) along the direction of the waveguide, by setting the parameter `a2nm` as a different value to `a1nm`.
The vertical slab height is modelled by entering the correct values for the effective slab mode index (neff), the bulk material index (n_bulk) and the fraction of the field contained within the slab into the Matlab analysis script. These values can be obtained by solving for the fields in a slab waveguide, either through standard mode solver software or any other tool of your choice.

Tutorial TO BE ADJUSTED

For this tutorial we calculate the 2D data for Figure 2 from reference [1] (also a 2D approximation of Figure 4b from [2]).
To run the mpb-calculation run one of the following commands:

```
mpb a1nm=410.0 calculation-type=4 r0=0.270 s1=-0.127 W1_2D_v1.ctl > example.out
```

```
mpb-split X a1nm=410.0 calculation-type=4 r0=0.270 s1=-0.127 W1_2D_v1.ctl > example.out
```

where X is the number of processors that you want to use.

This will calculate the dispersion curve, group velocity curve and scattering coefficients for the structure. The waveguide is based on a W1 waveguide in a silicon-air membraned structure, with 410nm lattice period, ~220nm diameter holes and the first row of holes is shifted 52nm away from the defect.
While the 3D simulation of this is fairly time consuming (15-20 processor hours), the 2D version is much more efficient (8 processor minutes)
Once the calculation is finished, please move on to the data analysis tutorial.

Data Analysis

Please read the MPB data analysis tutorial first at:
http://ab-initio.mit.edu/wiki/index.php/MPB_Data_Analysis_Tutorial

The calculation described in the previous chapter will have produced two files. One called `epsilon.h5` and example.out. In this chapter we will only consider `example.out`; for help on how to use `epsilon.h5` refer to the h5utils manual at:
http://ab-initio.mit.edu/wiki/index.php/H5utils

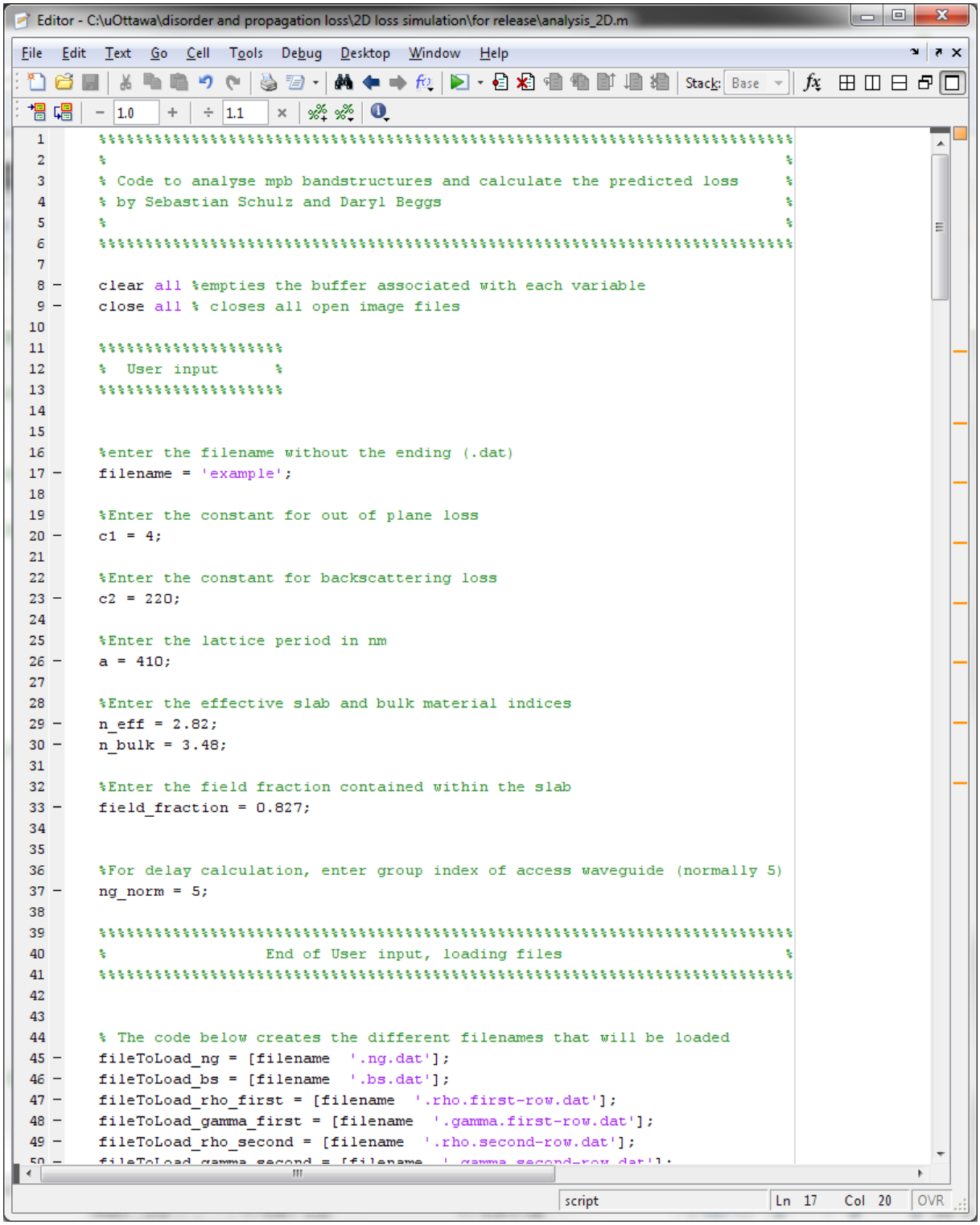
`example.out` contains the full results from our calculation, that means the bandstructure, group velocity and scattering coefficient data, alongside information about the structural parameters (period, hole radius etc.). We have provided several scripts that make it easy to reformat the output. First run the following two commands:

```
awk -f bsfiles_2D.awk example.out > example.bs.dat
awk -f ngfiles_2D.awk example.out > example.ng.dat
```

The first command searches for lines with the phrase "tefreqs" (indicating bandstructure data) and prints out the **K**-vectors (actually the value of K_x) and the frequencies (in dimensionless units of a/λ) of all the calculated bands.
The second command searches for the lines with "tevelocities" (indicating group velocity data) and prints out the **K**-vector, frequency and x-component of the group index ($n_g = 1/v_g$) of the band of interest (in this case b = 23 in the file).

To extract the scattering coefficients open the file `extract_data.sh`:
This shell script searches for lines with "integral-holes-list" and "gamma" (indicating out of plane scattering) or "rho" (indicating backscattering) and prints out the scattering coefficients for the first three rows of holes (subsequent rows have a negligible contribution to the propagation loss). If you want to run this script for an output file other than `example.out`, open the script in an editor and replace "`example.out`" with the corresponding filename. Then run:
`extract_data.sh`

We are now ready to predict the propagation loss according to equation 2 from reference [2].
For this we have provided an easy to use Matlab script, which uses the `*.dat` files from above as input.
Open and run the script: `analysis_2D.m` using Matlab You should see the same as below.




```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                       %
% Code to analyse mpb bandstructures and calculate the predicted loss   %
% by Sebastian Schulz and Daryl Beggs                                   %
%                                                                       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all %empties the buffer associated with each variable
close all % closes all open image files

%%%%%%%%%%%%%%%%%%%%
%  User input      %
%%%%%%%%%%%%%%%%%%%%


%enter the filename without the ending (.dat)
filename = 'example';

%Enter the constant for out of plane loss
c1 = 4;

%Enter the constant for backscattering loss
c2 = 220;

%Enter the lattice period in nm
a = 410;

%Enter the effective slab and bulk material indices
n_eff = 2.82;
n_bulk = 3.48;

%Enter the field fraction contained within the slab
field_fraction = 0.827;


%For delay calculation, enter group index of access waveguide (normally 5)
ng_norm = 5;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                  End of User input, loading files                     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% The code below creates the different filenames that will be loaded
fileToLoad_ng = [filename  '.ng.dat'];
fileToLoad_bs = [filename  '.bs.dat'];
fileToLoad_rho_first = [filename  '.rho.first-row.dat'];
fileToLoad_gamma_first = [filename  '.gamma.first-row.dat'];
fileToLoad_rho_second = [filename  '.rho.second-row.dat'];
```

The parameters c_1 and c_2 are explained in more detail in reference [2]. The default values are twice those from reference [2], since we made some small changes to the code since then. As these parameters are dependent on the fabrication quality of your device, the values that provide the best fit to your own experimental data might deviate from the default.

The parameters `n_bulk`, `n_eff` and `field_fraction` stand for the bulk slab material index, the effective modal index of the vertical PhC slab and the fraction of the vertical slab mode that is contained within the slab, respectively. These parameters are need to account for the percentage of light can can scatter from

defects (field_fraction) and for a renormalization of the operating wavelength (n_bulk and n_eff).
The script works as follows:

2. It reads all the input files generated in the previous steps and creates corresponding variables
3. Next it converts all the frequencies to wavelength in nm using the effective period method (therefore if you choose a different period or materials for your calculation you need to change the values (`a`, `n_bulk` or `n_eff` here).
4. The different contributions to gamma and rho are then added together
5. Equation 2 from [2] and the `field_fraction` (needs to be adapted for different slab layouts, either material or vertical dimensions) is used to calculate the propagation loss alpha (in dB/cm) for each k-point
6. The bandstructure, group index and propagation loss are plotted and saved to corresponding Matlab figure files.

The plot of propagation loss against group index is the theoretical line from fig. 4B in [2].
You will see that there are two plots of propagation loss against group index, with the loss expressed in dB/cm and dB/ns. As explained in [1] it can be very useful to express propagation loss in dB/ns. This value is calculated inside the script using the equation:

```
loss_time = 30*loss./(abs(ng)-ng_norm)
```

where `loss_time` is in dB/ns, `loss` is in dB/cm, `abs(ng)` is the modulus of the group index and `ng_norm` is defined at the beginning of `loss_anaysis.m`. The factor of 30 is the speed of light (3e8m/s = 30cm/ns). In this equation the time that gives the /ns for dB/ns is the extra delay caused by moving through the structure at `ng_norm` instead of ng. The default value of `ng_norm` is 4.5 as most silicon wires and photonic crystals have a group index of 4-5 in the fast light regime and therefore this calculates the loss per ns delay.
However if you want to know the loss relative to light propagating in an alternative medium enter the group index of the alternative medium as `ng_norm`.

For help with the parameters available for the mpb calculation please refer to the table and figure on the next pages.

Further Resources
mpb-discuss mailing list (http://ab-initio.mit.edu/cgi-bin/mailman/listinfo/mpb-discuss)
MIT MPB website (http://ab-initio.mit.edu/wiki/index.php/Main_Page)
(Please also include the citation for MPB
"Moulding the flow of light" John D. Joannopoulos, Steven G. Johnson, Joshua N. Winn, and Robert D. Meade. Princeton University Press 2008, http://ab-initio.mit.edu/book/

Reference

parameter			default	notes
a1nm	the lattice constant		400	in units of nm
a2nm	lattice constant along the waveguide		a1nm	in units of nm
calculation-type	chooses what to calculate 0: basic calculation of bandstr 1: bandstr + group velocities 2: bandstr + output fields profiles 3: bandstr + group velocities + output fields profiles 4: bandstr + group velocities + scattering coefficients 5: bandstr + group velocities + scattering coefficients + output field profiles		1	
index-slab	refractive index of the slab material		3.48	defaults to Si
index-holes	refractive index of the holes and cladding		1	defaults to air
Kinterp	the number of K-points *between* Ks and Ke to calculate		19	integer
Ks	the first K-point to calculate		0.3	in units of a/2π
Ke	the final K-point to calculate		0.5	in units of a/2π
numbands	the number of bands to calculate		30	integer
p1	shift of first row of holes along the waveguide		0	in units of a2
p2	shift of second row of holes along the waveguide		0	in units of a2
p3	shift of third row of holes along the waveguide		0	in units of a2
r0	background radius of holes		0.3	in units of a1
r1	radius of first row of holes		r0	in units of a1
r2	radius of second row of holes		r0	in units of a1
r3	radius of third row of holes		r0	in units of a1
res	resolution of grid points in the plane of the slab		16	per a1
s1	shift of first row of holes inwards		0	in units of a1
s2	shift of second row of holes inwards		0	in units of a1
s3	shift of third row of holes inwards		0	in units of a1
w	width of the waveguide (i.e. Ww) – NOTE: W2 is not two holes missing		1	in units of a*sqrt(3)
W1band	the band index number of interest		23	integer

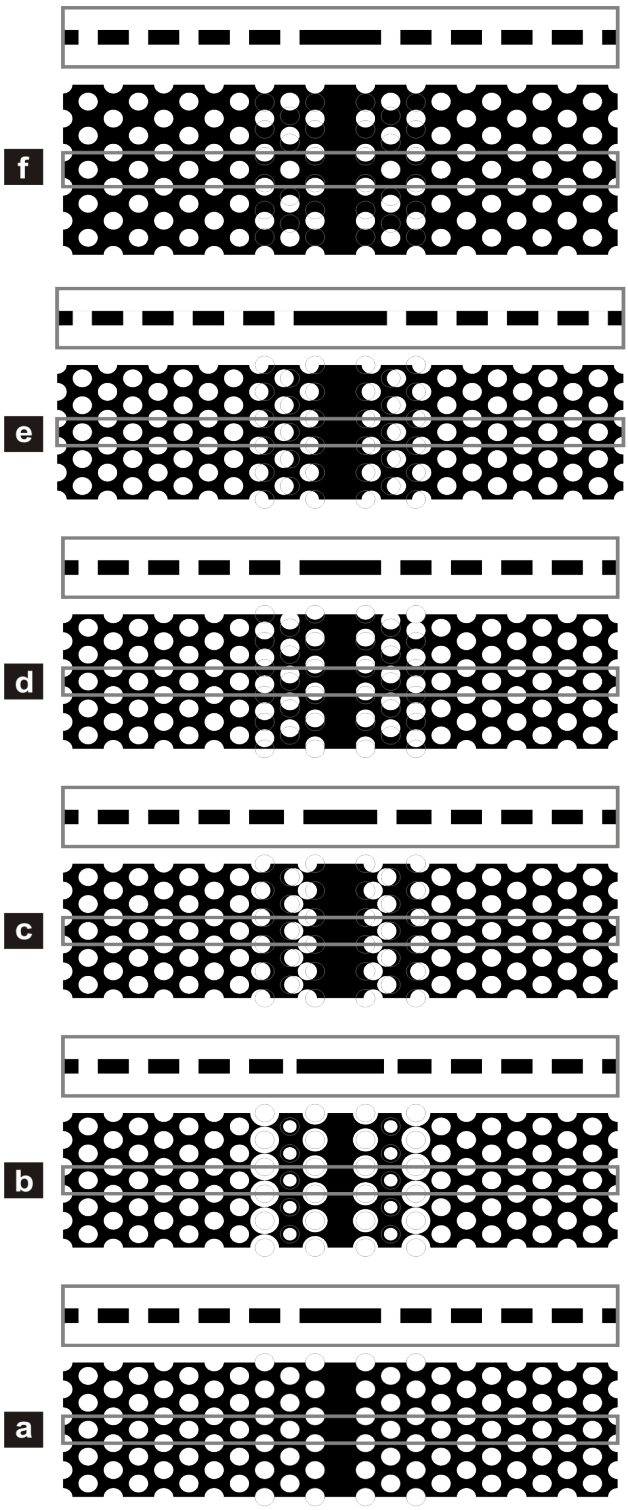


Figure 1

References

[1] S.A.Schulz, A.H. K. Park, I. De Leon, J. Upham and R. W. Boyd "Beyond the effective index method: improved accuracy for 2D simulations of photonic crystal waveguides", Journal of Optics **17**, 075006 (2015).

[2] L.O'Faolain, S. A. Schulz, D. M. Beggs, T. P. White, M. Spasenović, L. Kuipers, F. Morichetti, A. Melloni, S. Mazoyer, J. P. Hugonin, P. Lalanne and T. F. Krauss "Loss engineered slow light waveguides", Optics Express **18,** 27627-27638 (2010).

[3] S.A.Schulz, L. O. Faolain, D. M. Beggs, T. P. White, A. Melloni and T. F. Krauss “Dispersion engineered slow light in photonic crystals: a comparison”, Journal of Optics **12,** 104004 (2010).