

Optical Engineering

SPIDigitalLibrary.org/oe

Graphics processing unit–accelerated finite-difference time-domain method for characterization of photonic crystal fibers

Oleksiy Shulika
Igor Guryev
Natalia Gurieva
Igor Sukhoivanov
Jose Amparo Andrade Lucio

Graphics processing unit–accelerated finite-difference time-domain method for characterization of photonic crystal fibers

Oleksiy Shulika, Igor Guryev,* Natalia Gurieva, Igor Sukhoivanov, and Jose Amparo Andrade Lucio

University of Guanajuato, Engineering Division, Av. Universidad, S/N, Yuriria, Guanajuato, Mexico

Abstract. In the work, we have presented the technique based on the graphics processing unit accelerated finite-difference time-domain (FDTD) method for characterization of a single-mode photonic crystal fiber (PCF) with an arbitrary refractive index profile. In contrast to other numerical methods, the FDTD allows studying the mode propagation along the fiber. Particularly, we have focused attention on the method details that allowed us to reduce dramatically the computation time. It has been demonstrated that the accuracy of dispersion obtained by the FDTD method is comparable to the one provided by the finite elements method while possessing lower computation time. The method has been used to determine the fundamental mode cut-off of all-normal dispersion PCF and to find fiber losses beyond this wavelength. © The Authors. Published by SPIE under a Creative Commons Attribution 3.0 Unported License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.OE.52.12.126109](https://doi.org/10.1117/1.OE.52.12.126109)]

Keywords: finite-difference time-domain method; graphics processing unit computing; long-distance propagation.

Paper 131503P received Oct. 2, 2013; revised manuscript received Nov. 15, 2013; accepted for publication Nov. 22, 2013; published online Dec. 30, 2013.

1 Introduction

Today, the photonic crystal fiber (PCF) technologies allow experimenting with complex and exotic fiber profiles discovering new possibilities and applications such as fiber sensors,¹ lasers,² and even textiles.³ The properties of such fibers are well investigated. However, in specific problems such as supercontinuum generation⁴ and formation of the parabolic beam profile,⁵ the exact dispersion, losses and non-linearity profile of the fiber should be known. The method used to determine the dispersion depends on a fiber type. In case of large-mode PCF, such as LMA-20, even an analytical method works fine. However, as the fiber core size becomes smaller, numerical methods should be used. Moreover, in case of low-size core, the fundamental mode cut-off of the fiber may exist close to an excitation wavelength which may introduce losses to the fiber due to the mode leaking.

The dispersion can be computed with any numerical method capable of finding an eigen-frequency of the micro-structure. However, the plane-wave expansion (PWE) method^{6,7} can only treat linear materials with no material dispersion. The finite-element method (FEM), while providing more flexibility in the structure definition and high accuracy, appears to be quite slow and, moreover does not allow determining exactly the losses beyond the fundamental mode cut-off.

Therefore, in this work, we have focused our attention on the finite difference-time domain (FDTD) method.⁸ Being comparatively easy to implement, it provides a good basis for the hardware acceleration such as graphics processing unit (GPU) parallel computing.

The paper is organized as follows. We present the FDTD algorithm with unsplit-field perfectly-matched layer (PML) suitable for the fiber eigenfrequency computations. Then we

describe the procedure and present basic steps for the method acceleration by means of GPU. The dispersion computed with the FDTD method is compared with the one obtained by the FEM technique. Finally, we find the losses of the fiber beyond the fundamental mode cut-off by solving the long-distance propagation problem.

2 FDTD Method for PhC Fibers Characterization

The best way to find the dispersion of a PhC fiber is to compute its dispersion diagram [i.e., the eigenfrequencies versus the propagation constant $\omega(\beta)$]. In the most basic case, such as linear, dispersionless materials, this can be done by a numerous methods such as PWE, FEM, FDTD, etc. However, when long-term effects should be taken into account, the FDTD method becomes preferable.

2.1 FDTD Algorithm Formulation

The time-dependent field distribution within the fiber is found by solving the system of Maxwell's equations. In general, the FDTD technique is used to present the equations in the form of recursive formulas where the field distribution at the next time step is found from the one at the previous step.

Using the FDTD technique, the PhC fiber eigenfrequencies can be computed in at least two different ways. First, we can define the PhC-fiber as an infinite periodic structure with a defect representing the fiber core. This technique is similar to the one used in the PWE method. It requires a very basic FDTD implementation involving periodic boundary conditions. A delta-pulse is launched into a structure with further response analysis. However, it has many serious disadvantages. Particularly, with this method, it is only possible to treat purely periodic structures with no parametric variations that are now possessed by the most of the dispersion-managed fibers. Other technical difficulties occur when trying to analyze the time response and to extract the eigenfrequency.

*Address all correspondence to: Igor Guryev, E-mail: guryev@ieee.org

In this case, it is necessary to use the fast Fourier transform (FFT) or discrete Fourier transform (DFT). However, the precision of such methods rely on the computation time and if the fiber possesses two or more eigenfrequencies close to each other, the FFT or DFT will be unable to resolve them. This is not a big deal when investigating the effective mode index, but becomes a great problem when trying to compute the dispersion of the fiber. Therefore, this technique, while being very simple, appears to be very limited and cannot be used in an applied research.

The second technique consists of detailed representation of the fiber profile under investigation. To emulate the continuity of the computation region, it is surrounded by absorbing boundary conditions. In this case, we have used PML.^{8,9} The delta-pulse (or a predefined field pattern) is launched into a fiber core. After a certain time, only the propagation modes are left in the fiber, whereas the noise and leaky modes are effectively absorbed by the PML. The method

possesses the highest efficiency when investigating a single-mode PhC fiber. In this case, the frequency of the eigen-mode can be determined by direct measurement of the oscillation period, avoiding the DFT or FFT.

In our work, we have implemented the FDTD with the nonsplit field PML. To emulate the field propagation along z axis possessing the propagation constant β , the z components of the electric and magnetic fields are presented in the form of harmonic functions as follows:

$$\begin{aligned} E'(x, y, z) &= E(x, y) \cdot e^{i\beta \cdot z} \\ H'(x, y, z) &= H(x, y) \cdot e^{i\beta \cdot z} \end{aligned} \quad (1)$$

In this case, z derivatives are found analytically and the problem becomes two dimensional. The recursive equations for the electric components are then presented in the following form:

$$\begin{aligned} D_x|_{n,l}^{j+1} &= \left(\frac{2\epsilon\kappa_y - \sigma_y\Delta t}{2\epsilon\kappa_y + \sigma_y\Delta t} \right) D_x|_{n,l}^j + \left(\frac{2\epsilon\Delta t}{2\epsilon\kappa_y - \sigma_y\Delta t} \right) \times \left[\left(H_z|_{n+1/2,l+1/2}^{j+1/2} - H_z|_{n+1/2,l-1/2}^{j+1/2} \right) - i \cdot \beta' H_y|_{n+1/2,l+1/2}^{j+1/2} \right] / \Delta, \\ E_x|_{n,l}^{j+1} &= \left(\frac{2\epsilon\kappa_z - \sigma_z\Delta t}{2\epsilon\kappa_z + \sigma_z\Delta t} \right) E_x|_{n,l}^j + \left(\frac{1}{(2\epsilon\kappa_z + \sigma_z\Delta t)\epsilon} \right) \times \left[(2\epsilon\kappa_x + \sigma_x\Delta t) D_x|_{n,l}^{j+1} - (2\epsilon\kappa_x - \sigma_x\Delta t) D_x|_{n,l}^j \right], \end{aligned} \quad (2)$$

$$\begin{aligned} D_y|_{n,l}^{j+1} &= \left(\frac{2\epsilon\kappa_z - \sigma_z\Delta t}{2\epsilon\kappa_z + \sigma_z\Delta t} \right) D_y|_{n,l}^j + \left(\frac{2\epsilon\Delta t}{2\epsilon\kappa_z - \sigma_z\Delta t} \right) \times \left[i \cdot \beta' H_x|_{n+1/2,l+1/2}^{j+1/2} - \left(H_z|_{n+1/2,l+1/2}^{j+1/2} - H_z|_{n-1/2,l+1/2}^{j+1/2} \right) \right] / \Delta, \\ E_y|_{n,l}^{j+1} &= \left(\frac{2\epsilon\kappa_x - \sigma_x\Delta t}{2\epsilon\kappa_x + \sigma_x\Delta t} \right) E_y|_{n,l}^j + \left(\frac{1}{(2\epsilon\kappa_x + \sigma_x\Delta t)\epsilon} \right) \times \left[(2\epsilon\kappa_y + \sigma_y\Delta t) D_y|_{n,l}^{j+1} - (2\epsilon\kappa_y - \sigma_y\Delta t) D_y|_{n,l}^j \right], \end{aligned} \quad (3)$$

$$\begin{aligned} D_z|_{n,l}^{j+1} &= \left(\frac{2\epsilon\kappa_x - \sigma_x\Delta t}{2\epsilon\kappa_x + \sigma_x\Delta t} \right) D_z|_{n,l}^j + \left(\frac{2\epsilon\Delta t}{2\epsilon\kappa_x - \sigma_x\Delta t} \right) \times \left[\left(H_y|_{n+1/2,l+1/2}^{j+1/2} - H_y|_{n-1/2,l+1/2}^{j+1/2} \right) - \left(H_x|_{n+1/2,l+1/2}^{j+1/2} - H_x|_{n+1/2,l-1/2}^{j+1/2} \right) \right] / \Delta, \\ E_z|_{n,l}^{j+1} &= \left(\frac{2\epsilon\kappa_y - \sigma_y\Delta t}{2\epsilon\kappa_y + \sigma_y\Delta t} \right) E_z|_{n,l}^j + \left(\frac{1}{(2\epsilon\kappa_y + \sigma_y\Delta t)\epsilon} \right) \times \left[(2\epsilon\kappa_z + \sigma_z\Delta t) D_z|_{n,l}^{j+1} - (2\epsilon\kappa_z - \sigma_z\Delta t) D_z|_{n,l}^j \right]. \end{aligned} \quad (4)$$

For the magnetic field:

$$\begin{aligned} B_x|_{n+1/2,l+1/2}^{j+1/2} &= \left(\frac{2\epsilon\kappa_y - \sigma_y\Delta t}{2\epsilon\kappa_y + \sigma_y\Delta t} \right) B_x|_{n+1/2,l+1/2}^{j-1/2} - \left(\frac{2\epsilon\Delta t}{2\epsilon\kappa_y + \sigma_y\Delta t} \right) \times \left[(E_z|_{n,l+1}^j - E_z|_{n,l}^j) - i \cdot \beta' E_y|_{n,l}^j \right] / \Delta, \\ H_x|_{n+1/2,l+1/2}^{j+1/2} &= \left(\frac{2\epsilon\kappa_z - \sigma_z\Delta t}{2\epsilon\kappa_z + \sigma_z\Delta t} \right) H_x|_{n+1/2,l+1/2}^{j+1/2} + \left(\frac{1}{(2\epsilon\kappa_z + \sigma_z\Delta t)\mu} \right) \times \left[(2\epsilon\kappa_x + \sigma_x\Delta t) B_x|_{n+1/2,l+1/2}^{j+1/2} \right. \\ &\quad \left. - (2\epsilon\kappa_x - \sigma_x\Delta t) B_x|_{n+1/2,l+1/2}^{j-1/2} \right], \end{aligned} \quad (5)$$

$$\begin{aligned} B_y|_{n+1/2,l+1/2}^{j+1/2} &= \left(\frac{2\epsilon\kappa_z - \sigma_z\Delta t}{2\epsilon\kappa_z + \sigma_z\Delta t} \right) B_y|_{n+1/2,l+1/2}^{j-1/2} - \left(\frac{2\epsilon\Delta t}{2\epsilon\kappa_z + \sigma_z\Delta t} \right) \times \left[i \cdot \beta' E_x|_{n,l}^j - (E_z|_{n+1,l}^j - E_z|_{n,l}^j) \right] / \Delta, \\ H_y|_{n+1/2,l+1/2}^{j+1/2} &= \left(\frac{2\epsilon\kappa_x - \sigma_x\Delta t}{2\epsilon\kappa_x + \sigma_x\Delta t} \right) H_y|_{n+1/2,l+1/2}^{j+1/2} + \left(\frac{1}{(2\epsilon\kappa_x + \sigma_x\Delta t)\mu} \right) \times \left[(2\epsilon\kappa_y + \sigma_y\Delta t) B_y|_{n+1/2,l+1/2}^{j+1/2} \right. \\ &\quad \left. - (2\epsilon\kappa_y - \sigma_y\Delta t) B_y|_{n+1/2,l+1/2}^{j-1/2} \right], \end{aligned} \quad (6)$$

$$\begin{aligned}
B_z|_{n+1/2,l+1/2}^{j+1/2} &= \left(\frac{2\epsilon\kappa_x - \sigma_x\Delta t}{2\epsilon\kappa_x + \sigma_x\Delta t} \right) B_z|_{n+1/2,l+1/2}^{j-1/2} - \left(\frac{2\epsilon\Delta t}{2\epsilon\kappa_x + \sigma_x\Delta t} \right) \times \left[(E_y|_{n+1,l}^j - E_y|_{n,l}^j) - (E_x|_{n,l+1}^j - E_x|_{n,l}^j) \right] / \Delta, \\
H_z|_{n+1/2,l+1/2}^{j+1/2} &= \left(\frac{2\epsilon\kappa_y - \sigma_y\Delta t}{2\epsilon\kappa_y + \sigma_y\Delta t} \right) H_z|_{n+1/2,l+1/2}^{j+1/2} + \left(\frac{1}{(2\epsilon\kappa_y + \sigma_y\Delta t)\mu} \right) \times \left[(2\epsilon\kappa_z + \sigma_z\Delta t) B_z|_{n+1/2,l+1/2}^{j+1/2} \right. \\
&\quad \left. - (2\epsilon\kappa_z - \sigma_z\Delta t) B_z|_{n+1/2,l+1/2}^{j-1/2} \right].
\end{aligned} \tag{7}$$

Here $\beta' = \beta \cdot \Delta$, and the coefficient κ and the conductivity σ vary gradually within the PML area providing minimum back reflection from the computation region boundary.

After launching the field into a fiber, the propagation is computed by the leapfrog algorithm for a specific value of the propagation constant β . After a certain propagation time due to a PML, only the fundamental mode is left in the fiber while the leaky ones are absorbed. The response is then analyzed, and the mode frequency is found.

3 FDTD Method Acceleration with CUDA-Enabled GPU

One of the main problems of the FDTD method compared, for example, with FEM and especially PWE, is large computation time. In particular cases such as dispersion computation, it is possible to reduce the number of time-steps by applying the field teleportation and by avoiding the FFT when analyzing the spectrum. However, to achieve an acceptable dispersion accuracy it is still necessary to compute at least 200000 time steps. Other tasks such as long-distance propagation simulations require many more steps ($\approx 10^7$ per centimeter of a fiber). Being implemented directly in MATLAB, the FDTD technique appears to be very slow and computation of 30 points of the dispersion curve lasted for several days.

First, there have been attempts made to accelerate the method using parallel CPU computing. For this, we have run the computation at the supercomputer “Piritakua” of the University of Guanajuato. The supercomputer is equipped with 30 CPUs which allowed computing 30 times faster. However, this was not even close to the time COMSOL Multiphysics required to compute the dispersion with the same accuracy.

After this, there has been decided to take advantage of the GPU computing since “Piritakua” is equipped with three compute unified device architecture (CUDA)-enable graphic cards (Tesla M2070, Tesla M2050, and GeForce GTX580). Moreover, there are also three mobile workstations equipped with NVIDIA Quadro 1000 M and Quadro 3000 M.

Usually, the “leap-frog” algorithm⁸ is used to compute the field propagation by means of the recursive expressions 2 to 7. In the algorithm, the field distribution at the next time step is found from the distribution at the previous one. The only way to parallelize the computation in this case is within a single time step.

One of well-known bottleneck of the GPU computing is memory operations. Therefore, before creating any CUDA kernel, we have introduced several changes to optimize the code itself and to make it suitable for massive parallel computing, namely

- *We avoided complex numbers and used a single precision numbers for the field computation.* Since the field distribution is computed at each time step, the optimization of memory traffic here is of crucial importance. In the PCF model presented here, the real-to-imaginary conversion occurs only in the analytical expression for z -derivative as $i \cdot \beta'$ (e.g., $\frac{\partial E_x}{\partial z} \rightarrow i\beta E_x$). Therefore, if an initial field distribution would be real, we could avoid the complex numbers by changing $i\beta E_x \rightarrow \beta E_x$. This does not change the field distribution but allows halving the memory amount required for the field storage. Another point we can sacrifice to reduce the memory traffic is reduction of the numbers’ precision. Typically, MATLAB creates a double-precision variable when dealing with floating-points numbers. However, each variable occupies 8 bytes and, in case of the field distribution, such precision is basically unnecessary. Therefore, it would be wise to deal with a single-precision variable to reduce two times the arrays’ size. In the postprocessing, however, using the double precision is strongly recommended for operations such as the Fourier transform, interpolation, etc. Therefore, just by smart variables managing, it is possible to accelerate the GPU processing almost four time.
- *We avoided array indexing.* In most of the implementation of the FDTD method, the E- and H-field arrays possess different dimensions to emulate the middle-point field, such as $H_{n+1/2,l+1/2}$. This involves indexing while passing the matrices into a kernel. Such indexing appeared to significantly slow down the algorithm and, therefore, it has been decided to use the arrays of the same size leaving the indexing to the kernel.
- *We implemented the CUDA kernel in “C”.* Initially, the algorithm has been implemented using MATLAB because of its accelerated matrix operations and ability to use CUDA. However, when we implemented the CUDA kernel in “C” and used MATLAB functions to call it, we encountered a problem of the “parameters by reference,” i.e., MATLAB does not support user-defined pointers. When passing a matrix as a parameter to a function, it works pretty fast (as fast as passing the pointer in “C”) until we change the data in these matrices. In this case, it creates a copy of the matrix and returns the pointer to a calling function. This behavior resulted in a significant increase of the computation time since such copying appeared at any computation step. Therefore, it has been decided to implement all the time-iterations code in “C” while all the analyses were carried out in MATLAB. An initial field distribution, the coefficients, and the results are passed by files.

Table 1 Measured computation times for 100,000, 250,000 and 500,000 time steps.

	100,000	250,000	500,000
1 CPU	12,500 s	30200 s	61100 s
8 CPUs	1500 s	3750 s	7350 s
MATLAB-CUDA Quad 1000M	480 s	920 s	1600 s
C-CUDA Quad 1000M	200 s	422 s	828 s
C-CUDA Tesla M2070	50 s	105 s	208 s

This operation, being very slow in principle, does not affect significantly the total computation time since it is performed only twice for each value of β .

- *We used the texture memory.* Since we use the PML boundary conditions that involve time-constant but spatially distributed coefficients, it was reasonable to bind them to the texture memory which allowed almost two times acceleration of the algorithm.
- *We aligned the dimensions of the matrices with number of threads.* Since all the GPUs possess 32-thread parallelism, it is reasonable to align the size of the computation matrices with a multiple of 32. Particularly, in our work, we have taken the structures of 352×320 points.

All these details taken into account allowed us to achieve single-GPU acceleration of more than 200 times as compared to a single CPU. The computation times measured for a single eigenfrequency but for a different number of time steps are given in Table 1.

Apart from the great acceleration, from the table we can estimate the amount of time taken by non-CUDA related operations (particularly, file I/O). As it can be easily calculated, they take several seconds and do not introduce significant delay even at 100,000 points.

4 Numerical Results

The most important part of the method is the PML since it should effectively absorb the leaky modes. Therefore, we

have first analyzed the fiber response to the Gaussian pulse computed with and without the PML. As can be seen in Fig. 1, when the PML is absent, a lot of chaotic oscillations present in the response even after 200,000 steps. They are formed by the leaky modes that are multiply reflected from the computation area boundary. On the other hand, when the fiber is surrounded with PML, the chaotic oscillations vanish after about 10,000 steps which means that the leaky modes are effectively absorbed.

Another factor that can reduce the accuracy of the dispersion computation is the Fourier transform. Unfortunately, even at 500,000 time steps, the precision of the eigenfrequency is still very low and will cause a dispersion error of about 10^4 ps/nm km.

The problem has been overcome by assuming that the fiber is single mode and, therefore, possesses a single eigenfrequency. Taking this into account, we can measure the period directly from the response. Such a technique reduced dramatically the number of time steps. Now, 100,000 points is enough to estimate the dispersion with acceptable accuracy (of an order of 5 ps/nm km).

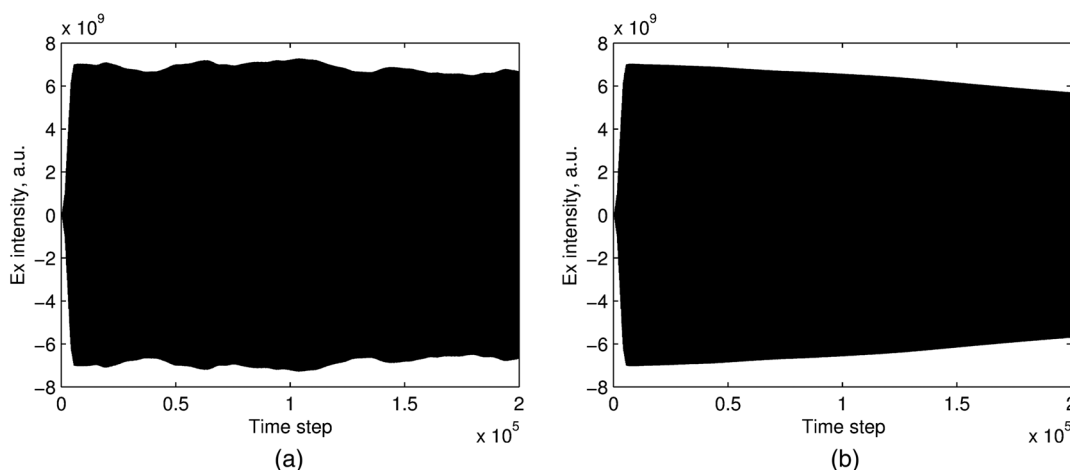
4.1 PCF Dispersion Computation

To verify the method, we have compared the dispersion with the one presented in Ref. 10. Namely, we have investigated the dispersion of the fused silica fiber with pitch $a = 2 \mu\text{m}$ and the radii varying from $r = 0.1a$ to $r = 0.2a$. The dispersion itself is calculated from the eigenfrequencies using the relationship:

$$D(\lambda) = -\frac{2\pi c}{\lambda^2} \frac{d^2\beta}{d\omega^2} = -\frac{\lambda}{c} \frac{d^2 n_{\text{eff}}}{d\lambda^2}. \quad (8)$$

The transverse-field component distribution and the dispersion are presented in Fig. 2. The field distribution after 100,000 time steps possesses good central symmetry and all the field is concentrated inside the core which allows expected correct dispersion computation.

The dispersion, on the other hand, shows a good agreement with the FEM results within the wavelengths $\lambda = 700 \dots 1700$ nm [see Fig. 2(c)]. An error does not exceed 10 ps/nm km, which is in good agreement with previous estimations. Since the error is a direct consequence of

**Fig. 1** The fiber temporal response computed without PML (a) and with PML (b).

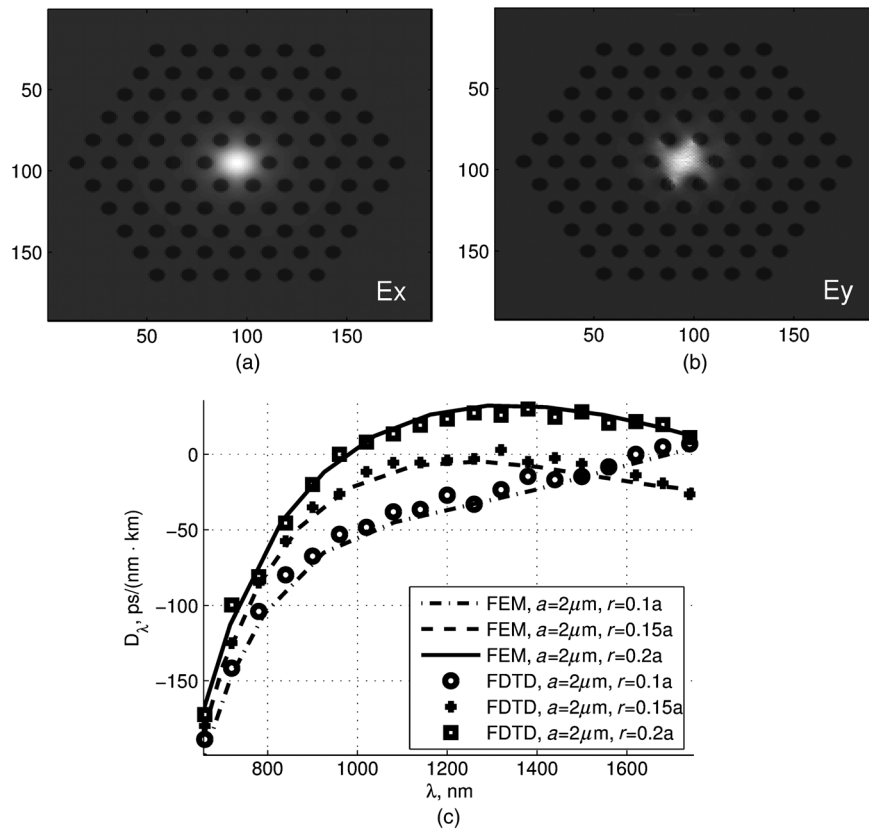


Fig. 2 The Ex (a) and Ey (b) field distribution inside the PCF core and the dispersion computed by FEM and by the FDTD method (c).

inaccurate estimation of the eigenfrequency due to the lack of points, the further error reduction can be made by performing a longer computation. In case of using the method in the optimization procedure, this may be critical, but for a single computation, the accuracy improves dramatically.

Although the accuracy of the FDTD results was lower than one of the FEMs, a computation with FEM takes about 11 h (at 6 CPUs) while FDTD solves the problem within 1 h (at a single GPU). Another advantage of the FDTD technique is the ability to take into account the dispersion and nonlinearity⁸ when the field propagation is computed, which can increase the range of applications and the precision of the method.

4.2 Long Distance Propagation in PCF

Among the most important applications of the accelerated FDTD method is a simulation of a mode propagation along the fiber. This is particularly important to determine the fundamental mode cut-off of the PCF and the losses below this limit.

As can be demonstrated in Ref. 5, the all-normal dispersion PCF can be effectively used for supercontinuum generation as well as for pulse reshaping and compression after about 10 cm of the fiber.

The method presented here can be used to simulate the propagation along the fiber and to verify its single-mode operation.

In Fig. 3(a), it is demonstrated that the electric field intensity in the center of the PCF core is computed at $\lambda = 940$ nm along 2 cm of fiber. The field magnitude does not seem to change. The large-period oscillations (of about 4-mm period)

appear due to the interference between almost degenerated orthogonal modes with frequency difference $<1\%$.

However, in case of the field propagation at $\lambda = 2000$ nm, the magnitude drops to zero after about 1 cm of propagation [see Fig. 3(b)].

From these results, we conclude that the fundamental mode cut-off of the fiber is somewhere between 1 and 2 μm . To find its exact value, we have computed losses by approximating the magnitude with an exponential

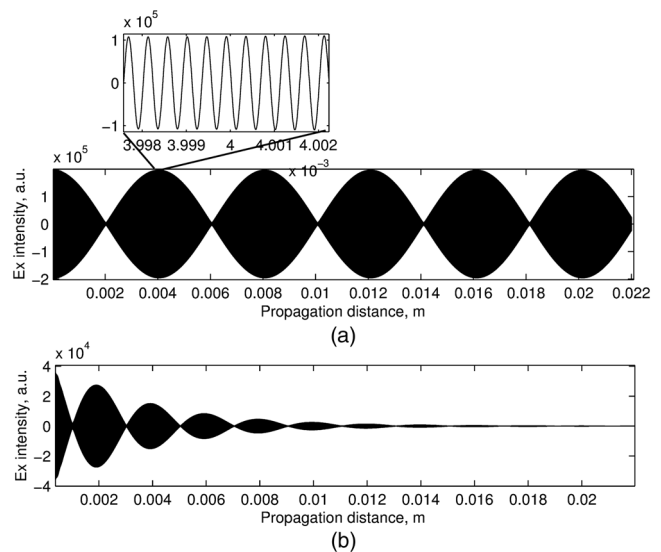


Fig. 3 Field propagation along the PCF at 940 nm (a) and at 2000 nm (b).

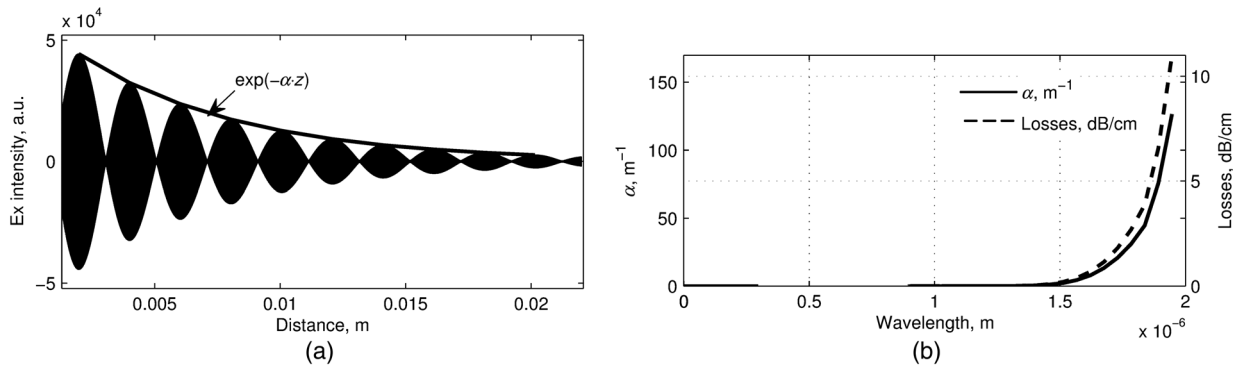


Fig. 4 Exponential approximation of the magnitude (a) and the PCF losses (b).

function as presented in Fig. 4(a). The loss curves are presented in Fig. 4(b). Analyzing the losses, the fundamental mode cut-off of the PCF appears at a wavelength 1020 nm where the nonmaterial losses become different from zero.

Therefore, computing the long-distance propagation by means of the accelerated FDTD method allowed determining exactly the fiber fundamental mode cut-off as well as fiber losses that should be taken into account in problems such as supercontinuum generation.

5 Conclusions

In the work, we have demonstrated the FDTD method that can be effectively used for dispersion investigation of an optical fiber possessing arbitrary refractive index profile. Due to CUDA-enabled GPU acceleration, the productivity of the method has been dramatically increased, and the computation time has been reduced 200 times as compared to a single CPU. Results analysis has demonstrated that the accuracy of the dispersion computation is compared with the FEM while taking less time. High performance of the method opens new possibilities for investigating the long-distance mode propagation which allows finding the fiber losses beyond the fundamental mode cut-off.

Although we have considered only single-mode fibers, the method is suitable for multimode fibers as well. The eigenfrequencies in this case are found by means of Fourier transform, which, of course, will require much longer computation time, but would allow treating for several modes.

Acknowledgments

This work has been supported by the project PROMEP/NPTC No.PROMEP/103.5/11/3908.

References

1. W. H. Ding and Y. Jiang, "All-solid photonic band gap fiber based distributed fiber optic pressure sensor," *Opt. Express* **20**(13), 14054–14063 (2012).
2. Y. Lu and X. Gu, "All-fiber passively q-switched fiber laser with a sm-doped fiber saturable absorber," *Opt. Express* **21**(2), 1997–2002 (2013).
3. B. Gauvreau et al., "Color-changing and color-tunable photonic bandgap fiber textiles," *Opt. Express* **16**(20), 15677–15693 (2008).
4. H. Hundertmark et al., "Supercontinuum generation with 200 ps laser pulses in an extruded sif fiber at 1560 nm," *Opt. Express* **11**(24), 3196–3201 (2003).
5. I. A. Sukhoivanov et al., "All-normal dispersion photonic crystal fiber for parabolic pulses and supercontinuum generation," *Proc. SPIE* **8847**, 88470H (2013).

6. I. Sukhoivanov and I. Guryev, *Photonic Crystals—Physics and Practical Modeling, (Optical Sciences)*, 1st ed., Springer, Berlin (2009).
7. K. Sakoda, *Optical Properties of Photonic Crystals*, Springer-Verlag, Berlin (2005).
8. A. Taflove and S. Hagness, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*, 3rd ed., Artech House, Norwood, Massachusetts (2005).
9. J. Berenger, "A perfectly matched layer for the absorption of electromagnetic waves," *J. Comput. Phys.* **114**, 185–200 (1994).
10. K. Saitoh and M. Koshiba, "Empirical relations for simple design of photonic," *Opt. Express* **13**, 267–274 (2005).

Oleksiy Shulika has received both BS and MS degrees in electronic engineering with highest honors from the Kharkiv National University of Radio Electronics, Kharkiv, Ukraine, in 2000 and 2001, respectively, and PhD degree in optics and laser physics from V. N. Karazin National University, Ukraine, in 2008. Currently, he is with University of Guanajuato, Mexico. His research interests are ultrafast optics, fiber optics, semiconductor optics, semiconductor nanostructures, and optical micro- and nanostructures.

Igor Guryev graduated in laser and optoelectronic techniques from the University of Radio Electronics in 2005. He received his PhD in optics and laser physics in 2010 from the V. N. Karazin Kharkiv National University. He is currently a titular professor and researcher in the Engineering Division of the University of Guanajuato, Mexico. His research interests are in theoretical and experimental investigation of optical microdevices on the basis of nonlinear photonic crystals.

Natalia Gurieva received her PhD degree in information technologies in 2010. Since 2011, she is the professor of the Department of Art and Management, Division of Engineering, University of Guanajuato, Mexico. Her research interests are in theoretical and experimental investigation in the field of computer technologies and systems in publishing and printing industry; colorimetric characterization of color input and output devices, particularly scanners, digital cameras, printers, displays, and printing equipment.

Igor Sukhoivanov received his PhD in radiophysics and quantum electronics in 1985 and degree of Dr. of Phys. Mat. Sc. in optics and laser physics in 2002 from V. N. Karazin National University, Ukraine. He is currently a professor titular B in the Engineering Division of the University of Guanajuato, Mexico. His research interests are the investigation of quantum well devices, optics of ultrashort pulses, and nonlinear photonic crystals.

José Amparo Andrade Lucio received his BS degree in 1993 in communications and electrical engineering and ME in electrical engineering in 1995, both from the Guanajuato University in Salamanca, México. He received his DSc degree in optical sciences in 1999 from the National Institute for Astrophysics, Optics and Electronics in Puebla, México. Since 1999, he has been with Guanajuato University as a full professor.