

HIGH PERFORMANCE 2D SIMULATIONS FOR THE PROBLEM OF OPTICAL BREAKDOWN

D.A. Fadeev¹

¹ Institute of Applied Physics of the Russian Academy of Sciences (IAP RAS), Nizhny Novgorod, Russia

Annotation

Methods of numerical simulation of two-dimensional short laser pulse nonlinear dynamics are discussed. In this work parallel processing methods for modern CPU (central processing units) architectures supporting non-uniform memory access are considered. The method of adaptive mesh subdivision is proposed to reduce non-uniform load of each CPU during processing of nonlinearity. The results of the tests performed on the Intel Nehalem based a workstation with eight cores are presented.

Keywords: computational electromagnetic methods (050.1755); ionization (260.3230); ultrafast phenomena (260.7120); optical breakdown; parallel processing; non-uniform memory access (NUMA); adaptive calculations

Citation: Fadeev DA. High performance 2D simulations for the problem of optical breakdown. *Computer Optics* 2016; 40(5): 654-658. DOI: 10.18287/2412-6179-2016-40-5-654-658.

Introduction

The tendency to enhance computation power by increasing the number of processing units instead of clock frequency has been clearly seen in the recent years. This trend encourages developers to revise well proven numerical methods for architectures with serial data processing. The most unique approaches are required for massive parallel systems like graphical processing units (GPU) which have good prospects for general purpose computing and already made a big impact in this area [1]. Nevertheless, classic parallel systems are still applicable since GPUs have programming specifics and comparatively small amount of available memory on each graphics accelerator. Thus, a GPU can be considered as a central processing unit (CPU) with local memory, while a large-scale multi-GPU system looks like a classic parallel system with GPUs instead of CPUs.

In this paper, we will discuss the numerical methods for the problem of nonlinear laser pulse dynamics in non-stationary media (e.g., plasma formed during optical breakdown). This problem includes a number of linear algebra methods: tridiagonal linear system solution, fast Fourier transform, and explicit Runge-Kutta method. All numerical implementations proposed in this paper were initially focused on classic parallel platforms supporting non-uniform memory access (i.e. each die has its own memory controller and physical memory, which results in enhanced bandwidth for local memory access and reduced bandwidth for concurrent access; see the full story in [2, 3]). Thus, the main goal was to maintain local data processing in every part of the algorithm. From the issues highlighted in the previous paragraph such an approach is quite flexible and could be easily adopted for the majority of modern systems (e.g., GPU workstations).

The next subsection introduces the reader to the physical problem under consideration, shows its specific features and describes its importance. In the second subsection a discrete model will be introduced and the algorithm typically used for such problems will be described. Section 2 concerns different features of algorithm implementation. The simulation results demonstrating real laser pulse dynamics during the breakdown of ambient air are presented in Section 3.

1. Algorithm

1.1. Background and mathematical model

Calculations of laser pulse dynamics are of great interest in modern physics [4–8]. This is due to rapid development of lasers providing pulses with extremely low duration and high peak intensity. The results of numerical simulations of nonlinear processes give unique data, which cannot be measured directly from experiments. From the high performance computer science point of view the most interesting simulations concern dynamics of laser pulse electric field in 3D space. In this paper we will discuss the Cauchy problem for the distribution of laser pulse electric field in space and time. The mathematical model is based on reduced wave equation with some relations for electric currents or polarization conditioned by media under consideration (ambient air in our case). In most researches performed for linear polarized pulses the model uses only one component of laser pulse electric field (scalar model). This is due to almost paraxial propagation of the pulse in the medium, so the ratio of the main component to other components remains high. In this article we will discuss circular polarization of a laser pulse assuming E to be complex, with real and imaginary parts representing orthogonal polarizations. In the majority of cases back scattering is low and is neglected, thus allowing to focus on the laser pulse mass center area (or the imaginary point propagating with the speed of light in vacuum) and consider only comparatively slow processes of nonlinear laser pulse shape dynamics in a new co-moving frame of reference. From the number of models, reduced according to the points highlighted above, the one most applicable for extremely short laser pulses is discussed below. The model considers laser pulse dynamics in coordinates x , y and $\tau = t - z/c$, where c is the speed of light in vacuum. The initial conditions are set at the cross-section $z = z_0$. The laser pulse dynamics is described by the following equation:

$$\frac{\partial^2 E}{\partial z \partial \tau} = \Delta_{\perp} E + N(E), \quad (1)$$

where N is some nonlinear function of field amplitude E . For the case of optical breakdown of air, it is given by the following set of equations:

$$\begin{aligned}
 N &= n_p E + N_K + N_R, \\
 \frac{\partial n_p}{\partial \tau} &= \frac{1}{|E|} \exp\left(-\frac{1}{|E|}\right), \\
 N_K &= |E|^2 E, \\
 N_R &= \int_{-\infty}^{\tau} \exp(\tau' - \tau) |E(\tau')|^2 E(\tau') d\tau',
 \end{aligned}
 \tag{2}$$

where $n_p E$ is the air plasma response calculated according to the simplified Keldysh formula, N_K is Kerr instant response and N_R is Raman delayed response. In this paper we will discuss the radial symmetric case, thus the Laplacian in (1) has the standard form in cylindrical coordinates [9]:

$$\Delta = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} \right).
 \tag{3}$$

The model (1–2) is quite modern compared to the standard methods [4, 5] based on scalar equations for envelope [6], where vacuum dispersion should be corrected via additional linear terms. Equation (1) has good prospects for processes with severe spectrum broadening, and is well proven for the problem of ultra-short strong laser pulse interaction with relativistic plasma [10, 11].

1.2. Background and mathematical model

The discrete model is based on an equidistant net for E along both directions of r and τ . The approximation for the Laplacian for vector \mathbf{x} from \mathbb{R}^N is chosen in the following form:

$$\begin{aligned}
 \{\Delta_N \mathbf{x}\}_i &= \\
 \frac{x_{i+1} + x_{i-1} - 2x_i}{dr^2} + \frac{2(x_{i+1} - x_i)}{dr^2(2i+1)}, \quad \forall i \in (0, N-1).
 \end{aligned}
 \tag{4}$$

For $i=0$ the same formula is used under the assumption $x_{-1} = x_0$. Such an approximation provides the Laplacian in the radial symmetric case identical to the four-point approximation of 2D Laplacian discretization at center point ($i=0$ corresponds to $r=0$). For calculation of the $(N-1)$ -th term one should assume $x_N = x_{N-1}$.

Considering the Laplacian in the form (4) with the above assumptions one can see that the constructed operator Δ is Hermitian in the following metrics:

$$\begin{aligned}
 (\mathbf{x}, \mathbf{y}) &= \sum_{i=0}^{N-1} x_i y_i f(i) dr, \\
 f(i) &= \begin{cases} 3/8, & i = 0, \\ i+1/2, & i \in (0, N-1]. \end{cases}
 \end{aligned}
 \tag{5}$$

Formula (4) gives one order lower approximation than the standard one [9], but provides Hermitian matrix for Laplacian operator, which leads to energy conservation for the evolution operator built according to the Crank-Nicholson scheme.

In order to solve eq. (1) the split-step Fourier method [12] was used which assumes the evolution step to be split into two sub-steps: linear and nonlinear ones. The linear part is then considered in Fourier space, where the

operator of partial derivative along the τ coordinate can be represented as simple as multiplication by $i\omega$ term:

$$\Omega \frac{\partial}{\partial z} \mathbf{e}_- = \Delta \mathbf{e}_-,
 \tag{6}$$

where $\mathbf{e}_- = \mathbf{F} \mathbf{e}$ is the Fourier image along τ coordinate of the $\{\mathbf{e}\}_i = E(r_{[i/Nr]}, \tau_{i \bmod Nr})$ vector from \mathbb{C}^{NrNr} space representing the electric field E set on the numerical net, $\mathbf{F} = \mathbf{I}_{Nr} \otimes \mathbf{F}_{Nr}$ is the full Fourier transformation matrix along τ in \mathbb{C}^{NrNr} space, \mathbf{I}_{Nr} is the identical square matrix of N_r dimension, \mathbf{F}_{Nr} is the Fourier transformation matrix in \mathbb{C}^{Nr} space, finally the operator $\Omega = \mathbf{I}_{Nr} \otimes \text{diag}(i\omega_0, \dots, i\omega_{Nr-1})$ is the operator of multiplication by $i\omega$. The above approximation of partial derivation assumes approximation of E values along τ coordinate via Fourier series, which is quite convenient for oscillating wave packets. The Laplace operator can be then written as $\Delta = \Delta_{Nr} \otimes \mathbf{I}_{Nr}$. The electric field vector \mathbf{e} can be propagated by dz along z according to eq. (6) using the Crank-Nicolson scheme [13]:

$$\mathbf{e}(z+dz) = \mathbf{U} \mathbf{e}(z) \stackrel{\text{def}}{=} \left(1 + \mathbf{L} \frac{dz}{2}\right) \left(1 - \mathbf{L} \frac{dz}{2}\right)^{-1} \mathbf{e}(z),
 \tag{7}$$

where \mathbf{U} is the evolution step operator, \mathbf{L} is the Hermitian (in terms of metrics (5)) linear operator having the following form $\mathbf{L} = \Omega^{-1} \Delta = \text{diag}(i\omega_0^{-1}, \dots, i\omega_{Nr-1}^{-1}) \otimes \Delta_{Nr}$.

The method (7) has some useful properties. First of all, (7) has a second order of approximation (error is equal to $O(dz^3)$), but more important is that \mathbf{U} is a unitary operator. Thus, in terms of metrics (5) the relation $|\mathbf{e}|^2 = \text{const}$ is met, which corresponds to conservation of $|\mathbf{e}|^2$ integral along τ and r with metric weight r . This property provides stability of the numerical scheme, even for very low frequencies ω .

The non-linear step is performed using the explicit predictor corrector method. The Eulerian approximation (prediction) \mathbf{e}_+ is calculated using Fourier transformation along τ for each r_i independently:

$$\mathbf{e}_+ = \mathbf{e}(z) + \mathbf{F}^{-1} \Omega^{-1} \mathbf{F} N(\mathbf{e}(z)) \circ \mathbf{e}(z) dz.
 \tag{8}$$

Here, N is the nonlinear function of \mathbf{e} vector producing a vector of the same dimension and according to eqns. (2) providing approximation of the non-linear part using the trapezoid method of integration. We are not writing explicit expression for $N(\mathbf{e})$ since it is evident, yet taking much space. The symbol \circ denotes component-wise multiplication.

The final approximation (correction) for $\mathbf{e}(z+dz)$ has the following form

$$\begin{aligned}
 \mathbf{e}(z+dz) &= \mathbf{e}(z) + (dz/2) \left(\mathbf{F}^{-1} \Omega^{-1} \mathbf{F} N(\mathbf{e}(z)) \circ \mathbf{e}(z) + \right. \\
 &\quad \left. + \mathbf{F}^{-1} \Omega^{-1} \mathbf{F} N(\mathbf{e}_+) \circ \mathbf{e}_+ \right)
 \end{aligned}
 \tag{9}$$

The evolution operator (9) gives the second order of approximation. For a circular polarization, vector $N(\mathbf{e})$ has only real components and appears to be smooth along the τ coordinate, thus the trapezoid method of integration provides fair approximation.

The proposed method was well proven for the problems of laser plasma interaction in strongly nonlinear regimes [10, 11].

2. High performance implementation of the algorithm

2.1. Linear part

Assuming fast Fourier transformation to be the most time wasting function, the elements of $\mathbf{e}(z)$ were organized in memory in the natural way providing the smallest distance between cells that have neighboring values along τ . This data order reduces the utmost unwanted data transaction between the nodes. But this brings some difficulties for realization of \mathbf{U} from equation (7). Further we will use additional two-index notation for \mathbf{e}_- components $\{\mathbf{e}_-\}_{ij} = \{\mathbf{e}_-\}_{iNr+j}$. Since (7) implies the solution of N_r three-diagonal linear equations with dimension N_r , the value $\{\mathbf{e}_-\}_{i+1,q}$ depends on all $\{\mathbf{e}_-\}_{j<i+1,q}$ values in the forward run of the Thomas [14] algorithm and on all $\{\mathbf{e}_-\}_{j>i+1,q}$ values in the backward run. In the case of a classic parallel system, the solution for each sub-system (corresponding to different ω_q) can be obtained by independent processing of data in parallel, like it is shown in the scheme in fig. 1. As the Thomas algorithm has a small arithmetic load, the performance depends mostly on memory bandwidth. This leads to a bottleneck condition when several processing units access the same data from shared memory (see fig. 1).

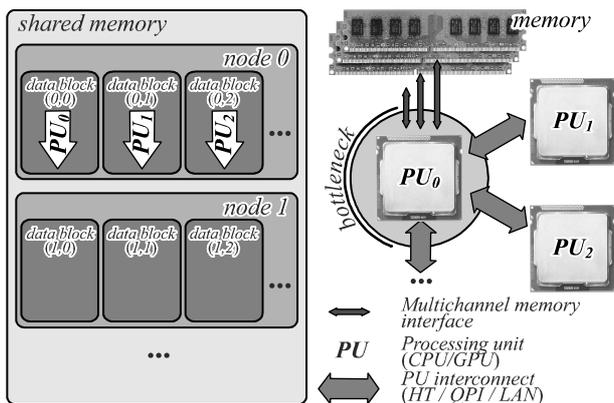


Fig.1. An illustration for inefficient method of data processing along columns in case of non-uniform memory access architecture. All processing units perform memory operations for data located in the physical memory space of the first processing unit leading to a bottleneck condition

Modern processors have integrated memory controller, thus the shared memory is non-uniform [2, 3]. This allows obtaining a bandwidth of independent memory operations equal to the bandwidth of each node (CPU die with its physical memory) multiplied by the number of nodes in the system. So, all operations should be performed within each node with a small amount of cross access (access of CPU from one node to the physical memory of another node).

Such an algorithm can be implemented based on the following idea. Let the number of nodes be N_n . At the first stage, all the nodes except the first one (node_0) are idle, while the first node executes the forward part of the

Thomas algorithm for ω_0 data column, i.e. for elements $(\{\mathbf{e}_-\}_{0,0}, \dots, \{\mathbf{e}_-\}_{Nr/Nn-1,0})$. At the second stage, the second node (node_1) continues forward integration according to the Thomas algorithm for elements of zero column, i.e. elements $(\{\mathbf{e}_-\}_{Nr/Nn,0}, \dots, \{\mathbf{e}_-\}_{2Nr/Nn-1,0})$, while the first node begins to operate with elements from the second column ω_1 , i.e. elements $(\{\mathbf{e}_-\}_{0,1}, \dots, \{\mathbf{e}_-\}_{Nr/Nn-1,1})$. At the third stage, the first node starts operating in the third column, the second one operates in the second column, and the third node continues with new elements from the first column. After N_n stages all the nodes operate simultaneously with delay along ω as is shown in fig. 2. Thus, each node operates only with its own physical memory.

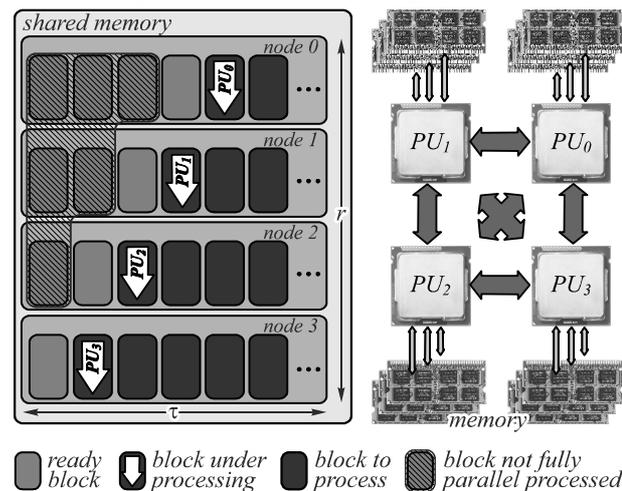


Fig. 2. An illustration for efficient method of data processing along columns in case of non-uniform memory access architecture. All processing units perform memory operations for data located only in their own physical memory space. Small portions of data are rarely transferred between nodes, thus making negligible impact on efficiency. The main time penalty is due to the delayed start of the first blocks (shaded in the scheme)

In the final implementation, blocks of data should be used instead of columns. It is conditioned by the CPU architecture designed to read and write an L1 cache line in one r/w cycle, so each step of integration in the Thomas algorithm should batch proceed for some next $\{\mathbf{e}_-\}_{i,j}$, $\{\mathbf{e}_-\}_{i,j+1}$, $\{\mathbf{e}_-\}_{i,j+2}, \dots$ values. The dimensions of the blocks should be $N_r/N_n \times (\text{size of L1 cache line})$. Hence, larger data dimensions should be used to obtain good performance.

2.2. Non-linear part

For minimizing the number of Fourier transformations the next scheme should be used. The non-linear part can be calculated according to (9 – 10) as follows:

$$\delta \mathbf{e}_+ = N(\mathbf{e}(z)) \circ \mathbf{e}(z),$$

$$\mathbf{e}_+ = \mathbf{e}(z) + \mathbf{F}^{-1} \mathbf{\Omega}^{-1} \mathbf{F} \delta \mathbf{e}_+ dz,$$

$$\mathbf{e}_-(z + dz) = \mathbf{e}_-(z) + (dz/2) \mathbf{\Omega}^{-1} \mathbf{F} (\delta \mathbf{e}_+ + N(\mathbf{e}_+) \circ \mathbf{e}_+),$$

$$\mathbf{e}(z) = \mathbf{F}^{-1} \mathbf{e}_-(z).$$

In the above scheme only four Fourier transformations are needed.

During the tests it was found that the computation time of the non-linear term N is strongly non-uniform along the r coordinate. The reason for that is the different time for calculation of the exponent in (2). Moreover, the computing load of (2) can be reduced by setting a threshold value for $|\{\mathbf{e}\}_{i,j}|$ under which the ionization rate in the right-hand side of eqns. (2)) can be assumed to be zero and the corresponding calculation of ionization rate can be omitted. This requires some algorithm to balance the sizes of \mathbf{e} sub-vectors located at each node. Here, the following algorithm implementation will be discussed. The time T_i spent by each node i on actual calculations was accumulated during some quite significant number of evolutionary steps, let us say 100. After that, new sizes for \mathbf{e} sub-vectors were calculated according to the following formulae

$$N_{ri}^+ = \frac{1}{4} \left(3N_{ri} + \frac{1}{C} \frac{N_{ri}}{T_i} \right), \quad C = \frac{1}{N_r} \sum_{i=0}^{N_r} \frac{N_{ri}}{T_i}. \quad (11)$$

Further, the sub-vectors located at each node were merged to a vector representing the field E on the whole mesh ($N_r \times N_\tau$) and then split according to (11) along the r coordinate – each new sub-vector of $N_{ri} \times N_\tau$ dimension is distributed to the i -th node.

The result of adaptive mesh splitting is shown in fig. 3. Fig. 3a presents the times (measured with $times()$ from `sys/times.h`) of actual computing (time spent in linear and nonlinear steps until non-blocking pthread mutex synchronization is reached) depending on the evolution step (the step in the plot is divided by 100 – the number of steps between mesh re-splitting procedure). In fig. 3b the same times are shown but N_{ri} are set to be constant – the mesh adapting is turned off. It is well seen that adapting the mesh strongly reduces T_{max} representing the time of each step, including synchronization time overheads. From fig. 3c it is well seen that sub-vector sizes vary appreciably. The number of steps before re-split procedure was invoked is conditioned by two factors: masking of time overheads due to merging of \mathbf{e} sub-vectors and memory reallocation and averaging of execution times T_i .

3. Sample simulation of optical breakdown and results

Numerical simulations were held for mesh dimensions $N_r=1024, N_\tau=512$. The entire the simulation took 34000 evolution steps. The testing platform was a dual processor workstation equipped with a pair of Intel Nehalem X5550 @ 2.67 Ghz, all memory channels were populated with PC3-8500 modules (total amount of memory was 96 Gb). The code was built with g++ 6.1.1 with O3 optimization and run on Linux 4.7.2 with NUMA support.

Fig. 4 presents three snapshots of simulated laser field amplitude. The initial pulse (Fig. 4a) was calculated by parabolic phase correction with conventional lens having focal length 1500 mm and translation by 1300 mm along the optical axis towards the focal point applied to the collimated 4.5 mJ pulse of Gaussian shape along r and τ of 35 fs FWHM duration and 10 mm FWHM diameter.

The translation was performed to skip the linear phase of laser pulse propagation.

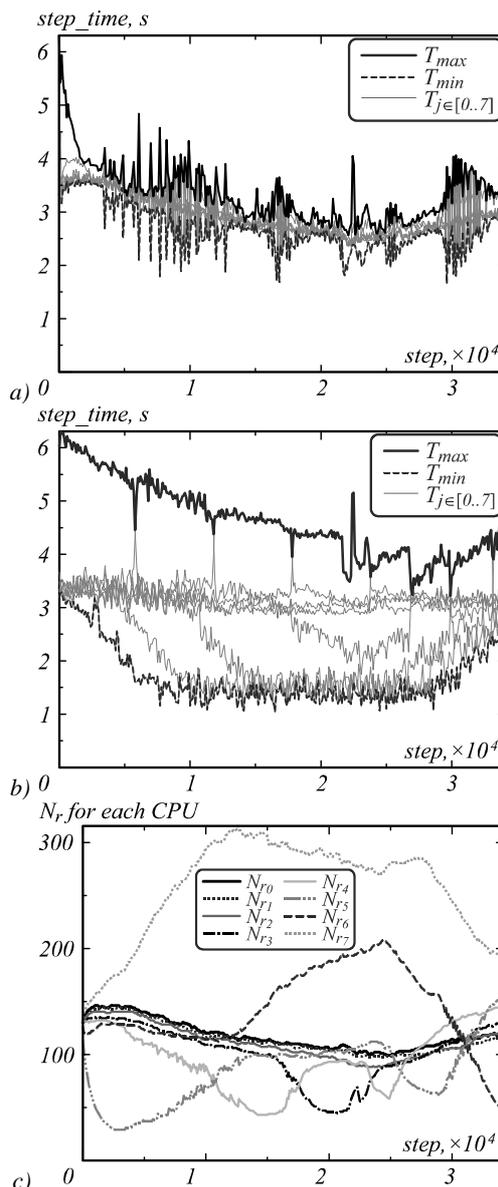


Fig. 3. Performance of adaptive mesh algorithm vs basic algorithm with fixed block sizes. Execution times of actual calculation for each node (T_i), T_{max} and T_{min} are maximal and minimal times for the case of adaptive mesh (a); same parameters for the case of fixed mesh (b); sub-vector size variation (c). One step on abscissa axis corresponds to 100 actual steps

In the second snapshot the process of ionization begins and plasma starts to refract the laser pulse creating structures also observed in [4]. In the third picture the result of the strong interaction of optical breakdown plasma and laser pulse is presented: the laser pulse is split into two, the last part of the laser pulse is refocused and creates the second impact of ionization.

Comparison of fig. 4 and fig. 3 shows the correlation between the calculation speed and the laser pulse shape. It is hard to predict the size of sub-vectors because of the complicated mechanism of calculating the ionization rate. Thus, adaptive mesh splitting allows pronounced performance improvement. The ratio of simulation times for adaptive and fixed methods was about 0.7.

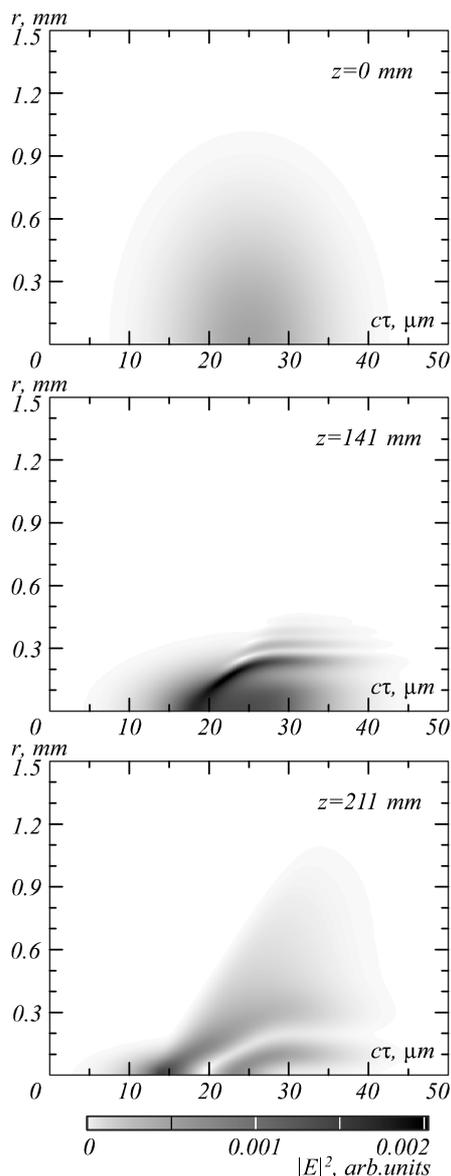


Fig. 4. An example of simulated optical breakdown. Snapshots of $|E|^2$ correspond to step = 0, step = 11300 and step = 17000 from left to right. The presented simulation example corresponds to the one in fig. 3

Conclusion

Some techniques of numerical simulation of the two-dimensional nonlinear Cauchy problem have been considered. It was shown that, despite the non-local nature of data processing in implicit algorithms, it is possible to develop an approach free from time overhead due to data transmission between the nodes of a distributed computational system.

The proposed method is applicable for abstract systems.

Authors' information

Daniil Aleksandrovich Fadeev (b. 1984 Nizhny Novgorod, Russia) graduated from Nizhny Novgorod State University of N.I. Lobachevski in 2008, faculty of Advanced School of General and Applied Physics. Since 2008. D.A. Fadeev is a junior researcher in Institute of Applied Physics RAS, Nizhny Novgorod, Russia. Scientific interests are computational physics and algorithms, new computational platforms, plasma physics, non-linear electrodynamics. E-mail: fadev.d.a@gmail.com and fadev@appl.sci-nnov.ru.

At the same time, the specific example of NUMA systems shows its advantages even in small-scale workstations. It is shown that in the case of merging and reallocation of memory the mesh adapting procedure is quite efficient. By increasing the number of steps between allocations this method may be used for large-scale systems with Ethernet/InfiniBand connection of nodes.

References

- [1] Burau H, Widera R, Hönig W, Juckeland G, Debus A, Kluge T, Schramm U, Cowan TE, Sauerbrey R, Bussmann M. PIC on GPU: A Fully Relativistic Particle-in-Cell Code for a GPU Cluster. *IEEE Trans Plasma Sci* 2010; 38(10): 2831-2839.
- [2] Non-uniform memory access. Source: (https://en.wikipedia.org/wiki/Non-uniform_memory_access)
- [3] Müller D. Memory and Thread Management on NUMA Systems. Diploma Thesis. Dresden; 2013.
- [4] Kandidov VP, Shlenov SA, Kosareva OG. Filamentation of high-power femtosecond laser radiation. *Quantum Electron* 2009; 39(3): 205-228. DOI: 10.1070/QE2009v039n03ABEH013916.
- [5] Schmitt-Sody A, Kurz HG, Bergé L, Skupin S, Polynkin P. Picosecond laser filamentation in air. *New J Phys* 2016; 18: 093005.
- [6] Brabec T, Krausz F. Nonlinear Optical Pulse Propagation in the Single-Cycle Regime. *Phys Rev Lett* 1997; 78: 3282. DOI: 10.1103/PhysRevLett.78.3282.
- [7] Dostovalov AV, Wolf AA, Babin SA, Dubov MV, Mezentsev VK. Numerical investigation of the effect of the temporal pulse shape on modification of fused silica by femtosecond pulses. *Quantum Electron* 2012; 42(9): 799-804. DOI: 10.1070/QE2012v042n09ABEH014960.
- [8] Chen Y, Theberge F, Kosareva O, Panov N, Kandidov VP, Chin SL. Evolution and termination of a femtosecond laser filament in air. *Optics Letters* 2007; 32(24): 3477-3479. DOI: 10.1364/OL.32.003477.
- [9] Britt S, Tsynkov S, Turkel E. A Compact Fourth Order Scheme for the Helmholtz Equation in Polar Coordinates. *J Sci Comput* 2010; 45(1): 26-47. DOI: 10.1007/s10915-010-9348-3.
- [10] Balakin AA, Litvak AG, Mironov VA, Skobelev SA. Compression of femtosecond petawatt laser pulses in a plasma under the conditions of wake-wave excitation. *Phys Rev A* 2013; 88(2): 023836. DOI: 10.1103/PhysRevA.88.023836.
- [11] Pipahl A, Anashkina EA, Toncian M, Toncian T, Skobelev SA, Bashinov AV, Gonoskov AA, Willi O, Kim AV. High-intensity few-cycle laser-pulse generation by the plasma-wakefield self-compression effect. *Phys Rev E* 2013; 87(3): 033104.
- [12] Bogomolov YaL, Yunakovskiy AD. Split-step Fourier Method for Nonlinear Schroedinger Equation. In: Proceedings of the International Conference "Days on diffraction", 2006. Saint-Petersburg: Universitas Petropolitana; 2006: 34-42.
- [13] Bellman R, Angel E. Dynamic programming and partial differential equations. London: Academic Press, Inc. (London) LTD; 1972. ISBN-13: 978-0120579501.
- [14] Thomas LH. Elliptic Problems in Linear Differential Equations over a Network. Watson Science Computer Laboratory Report. New York, NY: Columbia University; 1949.

Code of State Categories Scientific and Technical Information (in Russian – GRNTI): 29.31.27.

Received May 18, 2016. The final version – November 2, 2016.