

Oblique-incidence, arbitrary-profile wave injection for electromagnetic simulations

F. Pérez and M. Grech

Laboratoire pour l'Utilisation des Lasers Intenses, CNRS, École Polytechnique, CEA, Université Paris-Saclay, Sorbonne Université, F-91128, Palaiseau Cedex, France

(Received 17 September 2018; published 29 March 2019)

In an electromagnetic code, a wave can be injected in the simulation domain by prescribing an oscillating field profile at the domain boundary. The process is straightforward when the field profile has a known analytical expression (typically, paraxial Gaussian beams). However, if the field profile is known at some other plane, but not at the boundary (typically, nonparaxial beams), some preprocessing is needed to calculate the field profile after propagation back to the boundary. We present a parallel numerical technique for this propagation between an arbitrary tilted plane and a given boundary of the simulation domain, implemented in the Maxwell-Vlasov particle-in-cell code SMILEI.

DOI: [10.1103/PhysRevE.99.033307](https://doi.org/10.1103/PhysRevE.99.033307)**I. INTRODUCTION**

Electromagnetic (EM) codes are popular tools in various fields of physics, from nonlinear photonics to laser-matter and laser-plasma physics. In such codes, an EM wave can be introduced in the simulation domain by two means. The EM wave can be imposed as an initial condition, i.e., prescribing the fields throughout the domain at the initial time of simulation, of course ensuring that these fields satisfy the Poisson and zero-magnetic-divergence equations. This approach requires a simulation domain large enough to contain the whole EM wave. Furthermore, the knowledge of the full spatial profile at a given time may be challenging to obtain. To remove these constraints, a second technique, which we consider in the present article, consists in imposing the EM wave as a time-varying boundary condition. This second approach only requires the boundary surface to be large enough, and the knowledge of the EM field spatiotemporal profile is only required at the boundary.

Although the present article is relevant to any EM code or Maxwell solver, we illustrate the proposed method with particle-in-cell (PIC) simulations [1] using the open-source PIC code SMILEI [2]. The PIC method simulates the self-consistent evolution of both the field and particle distribution of a plasma. It is widely used, from astrophysical studies [3,4] to ultra-intense laser-plasma interaction [5,6]. In SMILEI, Maxwell's equations are solved using the finite-difference-time-domain approach [7,8], and EM waves can be injected or absorbed using the *Silver-Müller* boundary conditions [9]. These conditions allow for EM wave injection by prescribing the transverse magnetic field profiles at a boundary of the simulation domain.

Planar waves and paraxial Gaussian beams have a direct analytical formulation of the EM field in the whole space. Specifying their field as a function of time at one given boundary is thus trivial. However, other profiles do not have an analytical representation, or at least not in the whole space. Often they are known in a given plane which does not correspond to a simulation box boundary. This is typical for experimental profiles or theoretical nonparaxial beams. In that

case, specifying the field profiles at a given boundary is more involved. The present article describes a method to facilitate this process.

A recent work by Thiele *et al.* [10] detailed a technique to preprocess an EM wave profile specified at a given plane (parallel to, but not at the boundary) to propagate it backward and obtain the field profiles at the boundary. This approach is largely based on the angular spectrum method (ASM) used in other domains, such as acoustics [11,12] and digital holography [13]. It consists in applying a propagation factor to the fields in the spatial frequency domain, thus relies heavily on Fourier transforms. The theory by Thiele *et al.* extends this principle to temporal frequencies, consequently allowing to prescribe a temporal profile to the EM wave. This is obviously a strong requirement, in PIC codes, for modeling ultra-short (femtosecond) laser pulses.

In the present article, we combine the work by Thiele *et al.* and that of Matsushima *et al.* [13] to preprocess ultra-short-pulse EM waves prescribed at an arbitrary, oblique plane inside the simulation domain. The paper is organized as follows. Section II summarizes the theory by Thiele *et al.*, which we complete by that of Matsushima *et al.* in Sec. III. The numerical technique deployed in SMILEI is presented in Sec. IV, and examples in two and three dimensions are given in Sec. V. Finally, our conclusions are given in Sec. VI.

II. PROPAGATION BETWEEN PARALLEL PLANES

The ASM theory can be summarized in a simple manner (note that the following discussions relate to three-dimensional simulations, but can be directly applied to two-dimensional simulations by discarding the z axis.). It is valid for any scalar field A satisfying a wave equation

$$c^2 \Delta A(x, y, z, t) = \partial_t^2 A(x, y, z, t), \quad (1)$$

where x, y, z , and t are the space and time coordinates, and c is the wave velocity. In our situation, the field A may be any component of the EM field (or of the EM vector potential), and c is the speed of light in vacuum, as the propagation

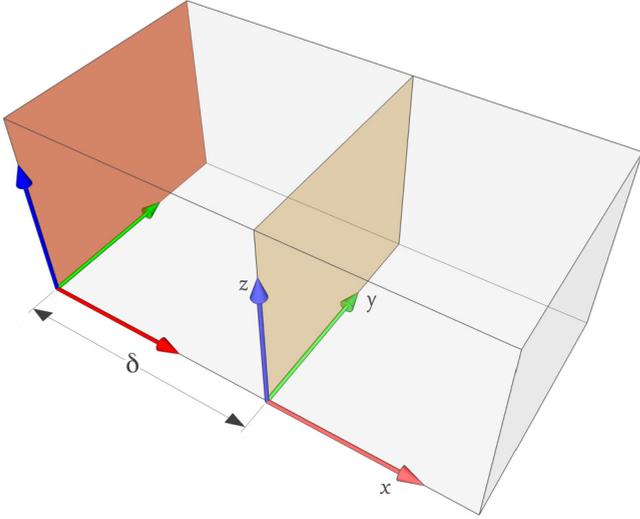


FIG. 1. Box model of a simulation where the EM wave is prescribed in a parallel plane at a distance δ from a boundary.

is only considered without plasma. We study specifically a propagation along the x axis, between two planes $x = -\delta$ and $x = 0$, as illustrated in Fig. 1.

The three-dimensional Fourier transform of Eq. (1) for the variables y, z , and t gives

$$(\partial_x^2 + k_x^2)\hat{A}(x, k_y, k_z, \omega) = 0, \quad (2)$$

where k_y, k_z , and ω are the respective conjugate variables, in the frequency domain, of y, z , and t ; and $k_x(k_y, k_z, \omega) \equiv \sqrt{\omega^2/c^2 - k_y^2 - k_z^2}$. Equation (2) has general solutions proportional to $\exp(-ik_x x)$ for waves propagating towards positive x . This means that, if the profile A is known at $x = 0$, the profile at $x = -\delta$ is obtained after multiplying \hat{A} by $\exp(ik_x \delta)$:

$$\hat{A}(-\delta, k_y, k_z, \omega) = \exp(ik_x \delta)\hat{A}(0, k_y, k_z, \omega). \quad (3)$$

Note that the function $k_x(k_y, k_z, \omega)$ assumes pure imaginary values where $k_y^2 + k_z^2 > \omega^2/c^2$. As those correspond to evanescent waves, they should not contribute to the propagation, and are simply removed from the calculation. In other terms, Eq. (3) should be replaced by

$$\hat{A}(-\delta, k_y, k_z, \omega) = P_\delta(k_y, k_z, \omega)\hat{A}(0, k_y, k_z, \omega), \quad (4)$$

where we introduced the propagation factor

$$P_\delta(k_y, k_z, \omega) = \begin{cases} \exp[ik_x(k_y, k_z, \omega)\delta] & \text{if } \frac{\omega^2}{c^2} > k_y^2 + k_z^2 \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

To recover the field profile $A(-\delta, y, z, t)$ in real space, a three-dimensional inverse Fourier transform would be sufficient. However, storing all values of the (y, z, t) profile might consume too much time and disk space. Instead, as suggested in Ref. [10], only a two-dimensional inverse Fourier transform on k_y and k_z may be carried out. This results in a $\tilde{A}(-\delta, y, z, \omega)$ profile, where ω still corresponds to the temporal Fourier modes. If necessary, only a few of these modes (the most intense ones) can be kept to ensure a reasonable disk-space usage. This number of temporal modes kept in memory must be specified by the user as there is no optimal choice appropriate

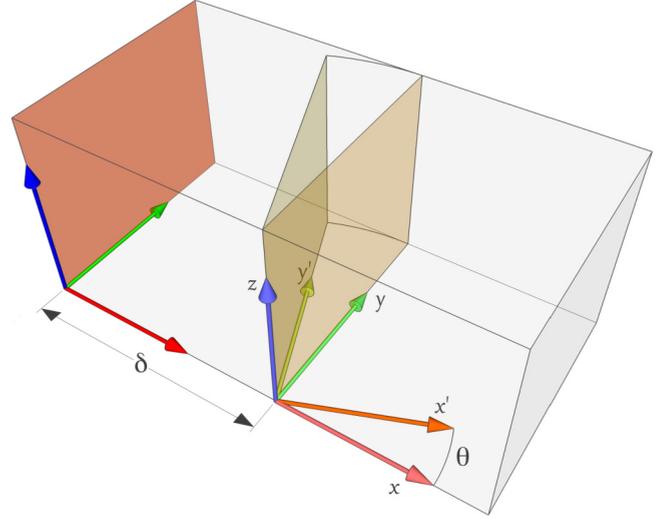


FIG. 2. Box model of a simulation where the EM wave is prescribed in an oblique plane.

for all cases. Note that this filtering is generally beneficial for an additional reason: it removes any spurious zero-frequency mode that may cause unphysical results.

In the end, the full $A(-\delta, y, z, t)$ profile is calculated during the actual PIC simulation, summing over the different ω according to

$$A(-\delta, y, z, t) = f(y, z, t) \times \sum_{\omega} \text{Re}(\tilde{A}(-\delta, y, z, \omega)e^{i\omega t}), \quad (6)$$

where ϕ is the complex argument of \tilde{A} and $f(y, z, t)$ is an additional profile, defined by the user. This optional profile f provides some extra control over the temporal reconstruction of the wave: as a finite number of temporal modes may be kept, the reconstructed wave is periodic, thus spurious repetitions of the EM pulse may occur later during the simulation. This custom function f may be used to remove those unwanted repetitions.

III. PROPAGATION BETWEEN TILTED PLANES

In the context of image reconstitution for digital holography, Matsushima *et al.* [13] proposed an extension of the ASM to handle the propagation between two nonparallel planes, as illustrated in Fig. 2. In this section, we summarize a version of that theory that follows the previous section's formulation.

The rotation of the wave is handled in the Fourier space, that is working directly on the field $\hat{A}(x, k_y, k_z)$. In this section, the argument ω is not explicitly written as it does not change the reasoning.

We consider, for simplicity, a rotation around the axis of k_z . Denoting rotated quantities with a “prime” symbol, the rotation of the wave vector by an angle θ reads

$$\begin{aligned} k'_x &= k_x \cos \theta + k_y \sin \theta, \\ k'_y &= -k_x \sin \theta + k_y \cos \theta, \\ k'_z &= k_z, \end{aligned} \quad (7)$$

where $k'_x \equiv \sqrt{\omega^2/c^2 - k_y'^2 - k_z'^2}$.

To obtain an expression of the rotated profile $\hat{A}'(x', k'_y, k'_z)$, let us first apply a propagator term $\exp(ik_x x)$ in the Fourier transform

$$\begin{aligned} A(x, y, z) &= \iint \hat{A}(x, k_y, k_z) \exp(ik_y y + ik_z z) dk_y dk_z \\ &= \iint \hat{A}(x=0, k_y, k_z) \exp(\vec{k} \cdot \vec{r}) dk_y dk_z, \end{aligned} \quad (8)$$

where $\vec{k} = (k_x, k_y, k_z)$ and $\vec{r} = (x, y, z)$. Note that this equation is also true in the rotated frame

$$A'(x', y', z') = \iint \hat{A}'(x'=0, k'_y, k'_z) \exp(\vec{k}' \cdot \vec{r}') dk'_y dk'_z. \quad (9)$$

Knowing that rotation preserves the scalar product $\vec{k} \cdot \vec{r}$, we may operate a change of variable $(k_y, k_z) \rightarrow (k'_y, k'_z)$ in Eq. (8)

$$A(x, y, z) = \iint a(k'_y, k'_z) \exp(\vec{k}' \cdot \vec{r}') \left| \cos \theta - \frac{k'_y}{k'_x} \sin \theta \right| dk'_y dk'_z, \quad (10)$$

where we introduced $a(k'_y, k'_z) = \hat{A}(x=0, k_y, k_z)$. As the original profile is equal to the rotated one in rotated coordinates, i.e., $A(x, y, z) = A'(x', y', z')$, identifying the integrands in Eqs. (9) and (10) results in

$$\hat{A}(x=0, k_y, k_z) = \hat{A}'(x'=0, k'_y, k'_z) \left| \cos \theta - \frac{k'_y}{k'_x} \sin \theta \right|^{-1}. \quad (11)$$

This last equation constitutes the calculation to be carried out to obtain the rotated wave at $x=0$. More precisely, the user prescribes $A'(x'=0, y', z')$ which is converted to $\hat{A}'(x'=0, k'_y, k'_z)$ using a double Fourier transform. Then Eq. (11) consists in interpolating values on the rotated Fourier space, and applying the factor to obtain $\hat{A}(x=0, k_y, k_z)$. After this rotation, one may apply the same propagation factor as in Eq. (5) to combine both a rotation and a translation. Follows an inverse Fourier transform, identical to the description of the previous section, to recover the profile in real space.

This technique applies to any scalar field, and by extension, to any component of the EM field. In SMILEI, the boundary conditions only require the knowledge of the magnetic field profiles transverse to the boundary. As an example, for the $x=0$ boundary, one needs to prescribe the components B_y and B_z , the other components being naturally obtained by solving Maxwell's equations. Hence, in the rotated frame, the user needs to define the profiles $B'_{y'}(x'=0, y', z')$ and $B'_{z'}(x'=0, y', z')$, i.e., the magnetic field projected to y' and z' only. The rotation and propagation back to the boundary $x=0$ leads to the fields $B_y(x=0, y, z)$ and $B_z(x=0, y, z)$, which lie in the $x=0$ plane, but remain the components projected to y' and z' . To recover the components along the simulation directions y and z , a simple projection is applied:

$$B_y = B_{y'} \cos \theta, \quad B_z = B_{z'}. \quad (12)$$

IV. NUMERICAL IMPLEMENTATION

The theory described in the previous sections presents a few numerical obstacles requiring careful treatment, especially to enable a parallel treatment on a large number of processors. Let us first summarize the general numerical

process. In the particular case of SMILEI, this involves arrays of complex numbers of initial size (N_y, N_z, N_t) corresponding, by default, to the number of cells N_y and N_z of the spatial mesh and the number of timesteps N_t required for the PIC simulation. The final array size is reduced to (N_y, N_z, M_t) , with $M_t < N_t$, should one keep only some of the temporal Fourier modes (as described in Sec. II). The process is outlined as follows:

- (1) The EM wave profile A' defined by the user is evaluated for all (y', z', t') coordinates.
- (2) Its Fourier transform \hat{A}' is computed along the three axes.
- (3) For each ω , the total spectral energy is computed, and only those ω with the highest magnitude (according to some user-defined criterion) are kept.
- (4) An interpolation method transforms \hat{A}' into \hat{A} corresponding to a rotation in frequency space, see Eq. (11).
- (5) The array \hat{A} is multiplied by the propagation factor of Eq. (5).
- (6) The inverse Fourier transform is computed along k_y and k_z only.
- (7) The resulting array is stored in a file. The file is read later by each process to reconstruct the wave at each timestep, according to Eq. (6).

Steps 2 and 6 make use of an implementation of the fast Fourier transform (FFT) algorithm. However, to enable multiprocessor computation, the arrays must be split among N_{proc} processors. There are several approaches to perform a parallel FFT (see Ref. [14] for a review). In SMILEI, we chose to split the initial array A' in equal parts along its first dimension, y (note that, before step 1, the array size is extended, if necessary, to a multiple of N_{proc}). As a consequence, each processor owns an array of size (n_y, N_z, N_t) , where $n_y = N_y/N_{\text{proc}}$. Step 2 begins by applying the FFT algorithm to the last two dimensions z and t . To transform the first dimension y , some data reorganization is necessary. We employ the message passing interface (MPI) protocol to communicate parts of the arrays between processors, so that each processor finally owns a portion of the global array corresponding to a slab along the second dimension z , of size (N_y, n_z, N_t) , where $n_z = N_z/N_{\text{proc}}$. This allows the computation of the FFT along y , thus providing the total Fourier transform \hat{A}' . Note that, in this whole process, the decomposition of the array between processors has changed.

In step 6, the calculation is almost the same, but in reverse order. First, the inverse FFT is computed along y . Then, the array decomposition between processors is reversed again, using MPI, so that they each own a portion of size (n_y, N_z, M_t) . Finally, the inverse FFT is computed over the axis z .

Step 4 requires some interpolation algorithm that maps $\hat{A}'(k'_y, k_z, \omega)$ to $\hat{A}(k_y, k_z, \omega)$. In computing terms, for each point at location (k_y, k_z, ω) of the resulting array \hat{A} , we determine the location (k'_y, k_z, ω) in the initial array \hat{A}' from where the value is copied, using Eq. (7). As this location may not fall exactly on an array point, we must interpolate between two consecutive k'_y . Importantly, the complex numbers given by a typical FFT of a laser profile present a rapidly varying argument, often close to, or faster than the grid resolution. Consequently, one must not interpolate linearly between two complex numbers to avoid cancellation of two numbers with

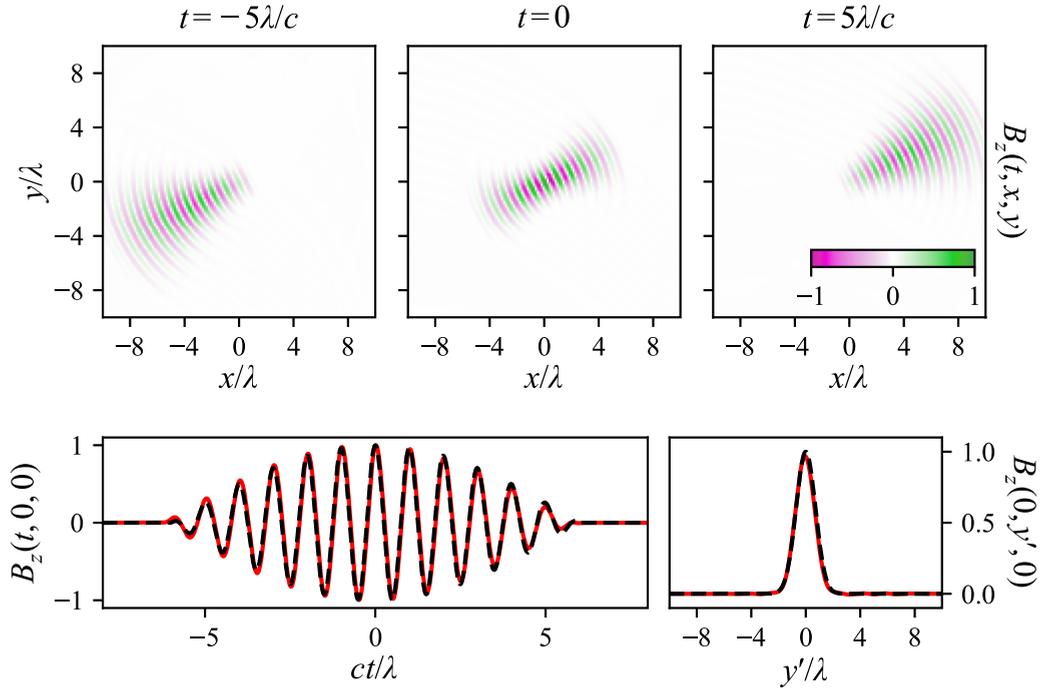


FIG. 3. Two-dimensional simulation of a tightly focused beam prescribed in a tilted plane. Top panels: Magnetic field B_z at three different times. Bottom panels: Comparison of the obtained (red lines) and requested (dashed black lines) magnetic field profiles, as a function of time (left) at $x = y = 0$, and as a function of y' (right) at $t = 0$.

opposite phases. Instead, in SMILEI, we considered that the magnitude and argument are both physically significant: the former represents the *weight* assigned to each mode, while the latter corresponds to a delay in space and/or time. This consideration supports a separate interpolation for the magnitude and the argument of these complex numbers, which is done in SMILEI. One additional precaution is necessary for the interpolation of the argument: as the global phase is supposed to vary smoothly across the array, we ensure that two consecutive arguments are always in the same order (e.g., the second larger than the first, adding 2π to the second when necessary).

V. EXAMPLES

The overall method described in the previous sections has been implemented in SMILEI in both two- and three-dimensional Cartesian geometries.

Let us first present a two-dimensional simulation of a tightly focused laser pulse with an angle of incidence of 25° , a wavelength λ and a linear polarization in the simulation plane (x, y) . The full simulation domain extends from $x = -16\lambda$ to 16λ and $y = -96\lambda$ to 96λ . To inject this laser we prescribed a B_z magnetic field profile along a line y' tilted by 25° with respect to the $x = -16\lambda$ boundary of the simulation domain. The intensity profile is Gaussian in space and has a \cos^2 shape in time, the focus being located in the middle of the simulation box ($x = 0, y = 0, y' = 0$) and the time $t = 0$ denoting the time at which the laser field is maximal

$$B_z(t, y') = a_0 \exp\left(-\frac{y'^2}{w^2}\right) \cos\left(\frac{\pi t}{2\tau}\right) \Pi\left(\frac{t}{\tau}\right) \cos(\omega_0 t), \tag{13}$$

with $\omega_0 = 2\pi c/\lambda$ the laser angular frequency, $\Pi(t) = 1$ for $-1 < t < 1$ and 0 otherwise, $a_0 = 1$ the field amplitude in arbitrary units, $w = \lambda$ the waist, and $\tau = 6\lambda/c$ the duration. The spatial and temporal resolutions were set to $\Delta x = \Delta y = \lambda/32$ and $c\Delta t = 0.95\Delta x/\sqrt{2}$, respectively.

Figure 3 presents the results obtained keeping only the 128 most intense temporal modes, and illustrate that a sufficient

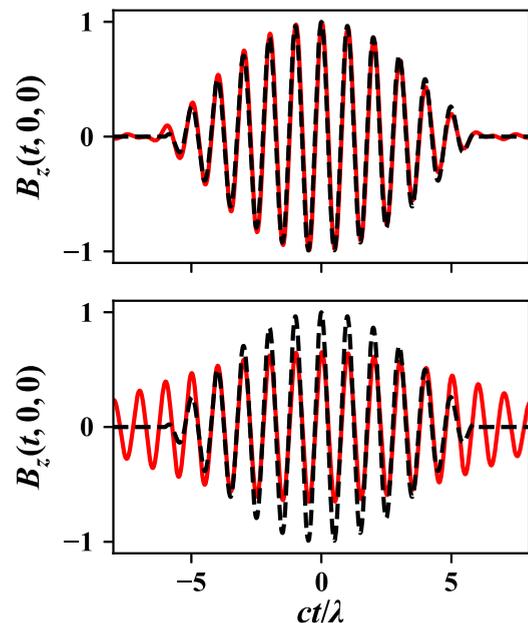


FIG. 4. Same simulations as in Fig. 3, keeping only 16 (top panel) and 4 modes (bottom panel).

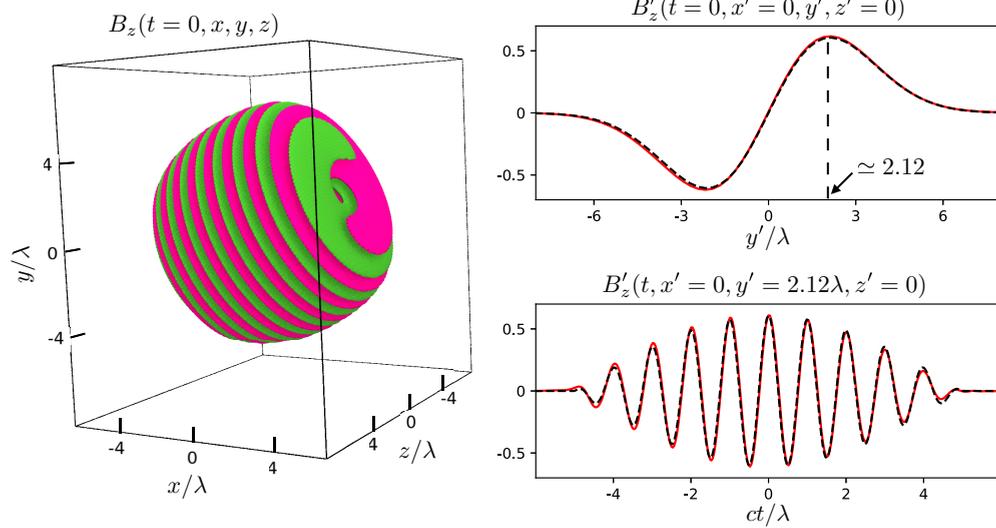


FIG. 5. Three-dimensional simulation of an oblique-incidence Laguerre-Gauss beam. Left: Three-dimensional rendering of magnetic field isocontours at $B_z = \pm a_0/10$. Top right: spatial profile in the tilted plane at $t = 0$. Bottom right: temporal profile at $x' = z' = 0$, $y' = 2.12\lambda$. Red and dashed black lines represent obtained and requested profiles, respectively.

precision for a short-pulse laser simulation is preserved. The top panels show both the spatial and temporal aspects of the laser pulse propagation. The bottom panels prove the excellent quantitative agreement between the simulated field evolution and the prescribed profiles. When less temporal modes are kept, the agreement decreases, as shown in Fig. 4 with 16 and 4 modes. We would like to stress, however, that the adequate number of modes to ensure a good description of the injected fields is strongly problem-dependent.

To illustrate our method in a three-dimensional simulation, we prescribe, in a (y', z) plane tilted by 25° , a Laguerre-Gauss beam (mode 1,0) with a \cos^2 temporal shape as

$$B_z(t, y', z) = a_0 \sqrt{2} \frac{r'}{w} \exp\left(-\frac{r'^2}{w^2}\right) \cos\left(\frac{\pi t}{2\tau}\right) \times \Pi\left(\frac{t}{\tau}\right) \cos(\omega_0 t - \phi'), \quad (14)$$

where $r' = \sqrt{y'^2 + z^2}$, $\phi' = \arctan(z/y')$, $w = 3\lambda$, and $\tau = 5\lambda/c$. The simulation box extends from $x = -8\lambda$ to 8λ , from $y = -24\lambda$ to 24λ , and from $z = -16\lambda$ to 16λ . The spatial and temporal resolutions are $\Delta x = \Delta y = \lambda/16$ and $c\Delta t = 0.95\Delta x/\sqrt{3}$, respectively. In this case, all the temporal modes were kept. Figure 5 illustrates the propagated laser pulse in a three-dimensional rendering of B_z isocontours. The right-

hand-side panels show the excellent quantitative agreement between the obtained and requested field profiles.

VI. CONCLUSION

In summary, the present article reviews and combines two related theories for the propagation of waves between two planes. They both extend the ASM, the first for temporal profiling, and the second for propagation between tilted planes. We combine both approaches and describe the numerical implementation in a parallel processing environment using the open-source PIC code SMILEI. This permits to define an arbitrary wave profile (amplitude and phase) at any oblique plane in the simulation box, instead of defining it at a boundary.

It will be applied to several situations relevant to high-intensity laser-plasma interaction: tightly focused, or spatially chirped laser pulses, and waves featuring orbital angular momentum.

ACKNOWLEDGMENTS

The authors thank Rachel Nuter for useful discussions and comparisons to her PIC code, and the SMILEI development team for technical support. This work was granted access to HPC resources from GENCI-TGCC (Grant No. 2017-x2016057678).

- [1] C. K. Birdsall and A. B. Langdon, *Plasma Physics Via Computer Simulation* (McGraw-Hill, New York, 1985).
- [2] J. Derouillat, A. Beck, F. Pérez, T. Vinci, M. Chiamello, A. Grassi, M. Flé, G. Bouchard, I. Plotnikov, N. Aunai, J. Dargent, C. Riconda, and M. Grech, *Comput. Phys. Comm.* **222**, 351 (2018).
- [3] A. Grassi, M. Grech, F. Amiranoff, F. Pegoraro, A. Macchi, and C. Riconda, *Phys. Rev. E* **95**, 023203 (2017).
- [4] S. F. Martins, R. A. Fonseca, L. O. Silva, and W. B. Mori, *Astrophys. J. Lett.* **695**, L189 (2009).
- [5] T. Tajima and J. M. Dawson, *Phys. Rev. Lett.* **43**, 267 (1979).
- [6] C. Thauy and F. Quéré, *J. Phys. B* **43**, 213001 (2010).
- [7] R. Nuter, M. Grech, P. Gonzalez de Alaiza Martinez, G. Bonnaud, and E. d'Humières, *Eur. Phys. J. D* **68**, 177 (2014).
- [8] A. Taflove, *Computation Electrodynamics: The Finite-Difference Time-Domain Method, 3rd Ed.* (Artech House, Norwood, MA, 2005).
- [9] H. Barucq and B. Hanouzet, *Asymptotic Anal.* **15**, 25 (1997).

- [10] I. Thiele, S. Skupin, and R. Nuter, *J. Comput. Phys.* **321**, 1110 (2016).
- [11] P. R. Stepanishen and K. C. Benjamin, *J. Acoust. Soc. Am.* **71**, 803 (1982).
- [12] G. T. Clement and K. Hynynen, *J. Acoust. Soc. Am.* **108**, 441 (2000).
- [13] K. Matsushima, H. Schimmel, and F. Wyrowski, *J. Opt. Soc. Am. A* **20**, 1755 (2003).
- [14] J. Richard, Implementation and Evaluation of 3D FFT Parallel Algorithms Based on Software Component Model, Master's thesis, LIP - ENS Lyon, Orleans-Tours, 2014.