ELSEVIER

Applied Numerical Mathematics 38 (2001) 187–222

APPLIED NUMERICAL MATHEMATICS

www.elsevier.nl/locate/apnum

# Long-time numerical computation of wave-type solutions driven by moving sources ☆

V.S. Ryaben'kii [a,b], S.V. Tsynkov [b,c,*], V.I. Turchaninov [a]

[a] *Keldysh Institute for Applied Mathematics, Russian Ac. Sci., 4 Miusskaya Sq., Moscow 125047, Russia*
[b] *ICASE, MS 132C, NASA Langley Research Center, Hampton, VA 23681-2199, USA*
[c] *School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv 69978, Israel*

**Abstract**

We propose a methodology for calculating the solution of an initial-value problem for the three-dimensional wave equation over arbitrarily long time intervals. The solution is driven by moving sources that are compactly supported in space for any particular moment of time; the extent of the support is assumed bounded for all times. By a simple change of variables the aforementioned formulation obviously translates into the problem of propagation of waves across a medium in motion, which has multiple applications in unsteady aerodynamics, advective acoustics, and other areas.

The algorithm constructed in the paper can employ any appropriate (i.e., consistent and stable) explicit finite-difference scheme for the wave equation. This scheme is used as a core computational technique and modified so that to allow for a non-deteriorating calculation of the solution for as long as necessary. Provided that the original underlying scheme converges in some sense, i.e., in suitable norms with a particular rate, we prove the grid convergence of the new algorithm in the same sense uniformly in time on arbitrarily long intervals. Thus, the new algorithm obviously does not accumulate error in the course of time; besides, it requires only a fixed non-growing amount of computer resources (memory and processor time) per one time step; these amounts are linear with respect to the grid dimension and thus optimal. The algorithm is inherently three-dimensional; it relies on the presence of lacunae in the solutions of the wave equation in odd-dimension spaces.

The methodology presented in the paper is, in fact, a building block for constructing the nonlocal highly accurate unsteady artificial boundary conditions to be used for the numerical simulation of waves propagating with finite speed over unbounded domains. 

☆ This work was supported by the National Aeronautics and Space Administration under NASA Contract No. NAS1-97046, and by Director's Discretionary Fund, while the first and second authors were in residence at ICASE, NASA Langley Research Center, Hampton, VA, USA.

* Corresponding author. Current address: Department of Mathematics, North Carolina State University, Box 8205, Raleigh, NC 27695, USA. Phone: (1-919)515-1877, Fax: (1-919)515-3798, URL: www.math.ncsu.edu/∼stsynkov.

*E-mail address:* tsynkov@math.ncsu.edu (S.V. Tsynkov).

0168-9274/01/$ – see front matter 
PII: S0168-9274(01)00038-1

*Keywords:* Wave equation; Lacunae; Finite-difference approximation; Explicit numerical integration; Arbitrarily long time intervals; Non-accumulation of error; Temporally uniform grid convergence; Fixed expenses per time step

## 1. Introduction

### *1.1. Formulation of the problem*

We will be solving numerically a Cauchy (initial-value) problem for the three-dimensional wave equation, $\boldsymbol{x} = (x_1, x_2, x_3)$:

$$\frac{\partial^2\varphi}{\partial t^2} - c^2\left(\frac{\partial^2\varphi}{\partial x_1^2} + \frac{\partial^2\varphi}{\partial x_2^2} + \frac{\partial^2\varphi}{\partial x_3^2}\right) = f(\boldsymbol{x}, t), \quad t \geqslant 0, \tag{1}$$

$$\varphi|_{t=0} = \left.\frac{\partial\varphi}{\partial t}\right|_{t=0} = 0. \tag{2}$$

The quantity $c$ in the differential equation (1) denotes the speed of wave propagation, e.g., the speed of sound. We will be interested in calculating the solution $\varphi = \varphi(\boldsymbol{x}, t)$ of problem (1), (2) only for those values of the spatial argument $\boldsymbol{x}$ that belong to the ball $S = S(t)$ of fixed diameter $d$ centered at a varying location $\boldsymbol{x_0}(t) = \int_0^t \boldsymbol{u}_0(t')\,\mathrm{d}t'$:

$$S(t) = \{\boldsymbol{x} \mid |\boldsymbol{x} - \boldsymbol{x}_0(t)| \leqslant d/2\}. \tag{3}$$

Here $\boldsymbol{u}_0 = \boldsymbol{u}_0(t)$ is the velocity of the center of $S(t)$, which we assume a given smooth function of time. We also assume that the motion of $S(t)$ is "subsonic":

$$|\boldsymbol{u}_0| \leqslant k < c. \tag{4}$$

A particular case of stationary source obviously corresponds to $\boldsymbol{u}_0(t) = \boldsymbol{0}$ and consequently, $\boldsymbol{x}_0(t) = \boldsymbol{0}$. We also note that any domain of fixed diameter $d$ can, in fact, be considered. The spherical shape of $S(t)$ is chosen for simplicity, and the discretization in subsequent sections is independent on this shape.

Regarding the right-hand side $f(\boldsymbol{x}, t)$ of Eq. (1), we always consider it a sufficiently smooth function with respect to all its arguments on $\mathbb{R}^3 \times (-\infty, +\infty)$ and also

$$\operatorname{supp} f(\boldsymbol{x}, t) = \{(x_1, x_2, x_3, t) \mid \boldsymbol{x} \in S(t),\ t > 0\}. \tag{5}$$

(This, in particular, implies that $f(\boldsymbol{x}, t)$ and a number of its derivatives $\partial f/\partial t$, $\partial^2 f/\partial t^2, \ldots$, turn into zero for $t = 0$.) We emphasize that although we assume that the sources are concentrated on $S(t)$, and the solution is also computed only on $S(t)$, this does not mean that the domain, on which the solution is computed, is limited to the region where the sources may be non-zero. This only means that both domains are bounded, and that the domain of the right-hand side is contained inside the domain of the solution $S(t)$ (but does not necessarily have to fill all of it).

In other words, we study the radiation of waves by a source, which is compactly supported in space for all times. The solution is of interest for us also on a compact domain, which fully contains this source and follows its motion if there is motion. This is a simplified model for many interesting physical phenomena that are more complex in their nature. A particular example related to calculation of the acoustic field around a maneuvering aircraft is provided in Appendix. The foregoing simplified model also appears very

useful when constructing the unsteady artificial boundary conditions (ABCs) for a variety of problems. A brief discussion on the subject of ABCs can be found in the Appendix as well.

Assume now that we need to solve the Cauchy problem (1), (2) on $\mathbb{R}^3$ for the time interval $0 \leqslant t \leqslant T_{\text{final}}$. A classical estimate for the solution (see, e.g., [2,12]) reads [1]

$$\|\varphi(\boldsymbol{x},t)\|_C \leqslant \frac{T_{\text{final}}^2}{2}\|f(\boldsymbol{x},t)\|_C, \tag{6}$$

and provided that $f \in C^2[t \geqslant 0]$ we also have $\varphi \in C^2[t \geqslant 0]$. Let us now remember that the right-hand side $f(\boldsymbol{x},t)$ is sufficiently smooth with respect to all its arguments. Then, we can differentiate equation (1) as long as this right-hand side permits; in so doing we arrive at

$$\frac{\partial^2}{\partial t^2}\left(\frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}}\frac{\partial^{\beta}}{\partial t^{\beta}}\varphi(\boldsymbol{x},t)\right) - c^2\Delta\left(\frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}}\frac{\partial^{\beta}}{\partial t^{\beta}}\varphi(\boldsymbol{x},t)\right) = \frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}}\frac{\partial^{\beta}}{\partial t^{\beta}}f(\boldsymbol{x},t), \tag{7}$$

where $\boldsymbol{\alpha} = (\alpha_1,\alpha_2,\alpha_3)$ is a multiindex, $|\boldsymbol{\alpha}| = \alpha_1+\alpha_2+\alpha_3$, $\partial \boldsymbol{x}^{\boldsymbol{\alpha}} \equiv \partial x_1^{\alpha_1}\partial x_2^{\alpha_2}\partial x_3^{\alpha_3}$, and $\beta$ is a regular "scalar" index. Again, Eq. (7) is valid in the classical sense for all those and only those specific values of $\boldsymbol{\alpha}$ and $\beta$, for which the right-hand side of (7) belongs to $C^2[t \geqslant 0]$. This guarantees that the solution is also in $C^2[t \geqslant 0]$ and thus the left-hand side of the equation exists. Clearly, Eq. (7) is the wave equation for the derivative $(\partial^{|\boldsymbol{\alpha}|}/\partial \boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})\varphi(\boldsymbol{x},t)$ of the original solution $\varphi(\boldsymbol{x},t)$, and the solution of (7) is driven by the corresponding derivative $(\partial^{|\boldsymbol{\alpha}|}/\partial \boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})f(\boldsymbol{x},t)$ of the original right-hand side $f(\boldsymbol{x},t)$. Regarding initial conditions for the derivative $(\partial^{|\boldsymbol{\alpha}|}/\partial \boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})\varphi(\boldsymbol{x},t)$, if we additionally assume (see above) that for a particular $\beta$ we have $(\partial^j/\partial t^j)f(\boldsymbol{x},t)|_{t=0} = 0$, $j = 0,1,\ldots,\beta-1$ (for the purpose of constructing the numerical algorithms, this restriction can, in fact, be alleviated, see Section 5), then we immediately see that $(\partial^{|\boldsymbol{\alpha}|}/\partial \boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})\varphi(\boldsymbol{x},t)$ will satisfy the same homogeneous initial conditions (2). Therefore, similarly to estimate (6), we have

$$\left\|\frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}}\frac{\partial^{\beta}}{\partial t^{\beta}}\varphi(\boldsymbol{x},t)\right\|_C \leqslant \frac{T_{\text{final}}^2}{2}\left\|\frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}}\frac{\partial^{\beta}}{\partial t^{\beta}}f(\boldsymbol{x},t)\right\|_C. \tag{8}$$

### *1.2. Core numerical technique*

In this paper, we construct an efficient numerical algorithm for solving the foregoing Cauchy problem (1), (2). This algorithm can employ any convergent finite-difference scheme for the wave equation as a core computational technique. The original computational procedure of this core scheme is modified in a special way so that to guarantee the grid convergence of the solution (with the same original rate) uniformly in time for arbitrarily long intervals.

Let us introduce a uniform Cartesian grid with the size $h$ in all spatial directions, and time step $\tau$ (grids of other types can be considered as well):

$$(x_{1m_1}, x_{2m_2}, x_{3m_3}, t_{m_4}) = (m_1h, m_2h, m_3h, m_4\tau), \quad m_1,m_2,m_3 = 0,\pm1,\pm2,\ldots,\ m_4 = 0,1,2,\ldots.$$

In every grid node $\boldsymbol{m} \equiv (m_1,m_2,m_3,m_4)$, Eq. (1) is replaced by the finite-difference equation

$$\sum_{\boldsymbol{n}\in N_{\boldsymbol{m}}} a_{\boldsymbol{mn}}\varphi_{\boldsymbol{n}} = f_{\boldsymbol{m}}, \tag{9}$$

[1] The norm $\|\cdot\|_C$ in Eq. (6) is a conventional maximum (supremum) norm, which coincides with the $L_\infty$ norm for continuous functions.

where $N_{\boldsymbol{m}}$ is the scheme's stencil that pertains to the node $\boldsymbol{m}$, $f_{\boldsymbol{m}}$ is a discrete approximation of the right-hand side $f(\boldsymbol{x},t)$, which, for simplicity, is always considered a node value:

$$f_{\boldsymbol{m}} = f(\boldsymbol{x},t)|_{(\boldsymbol{x},t)=(m_1h,m_2h,m_3h,m_4\tau)},$$

$a_{\boldsymbol{mn}}$ are the coefficients of the scheme, and $\varphi_{\boldsymbol{n}}$ is the discrete solution, $\varphi_{\boldsymbol{n}} \equiv \varphi_{n_1,n_2,n_3,n_4}$, which is defined on the same grid: $n_1,n_2,n_3 = 0,\pm1,\pm2,\ldots,\ n_4 = 0,1,2,3,\ldots$. Regarding the stencil $N_{\boldsymbol{m}}$ we will always assume that for a given $\boldsymbol{m} = (m_1,m_2,m_3,m_4)$ it is a collection of grid nodes located on time levels $m_4 - J_1,\ m_4 - J_1 + 1,\ldots,m_4,\ldots,m_4 + J_2 - 1, m_4 + J_2$:

$$N_{\boldsymbol{m}} = \bigcup_{j=-J_1,\ldots,J_2} N'_{m_1,m_2,m_3,m_4+j}, \tag{10}$$

where $N'_{m_1,m_2,m_3,m_4+j}$ are all nodes of the stencil that belong to the particular time level $m_4 + j$ only. Altogether, the stencil $N_{\boldsymbol{m}}$ given by (10) contains $J_1 + J_2 + 1 \equiv J + 1$ time levels. Regarding the coefficients $a_{\boldsymbol{mn}}$ we will additionally require that for a given "central" node $\boldsymbol{m} = (m_1,m_2,m_3,m_4)$ only one coefficient on the uppermost time level of the stencil be equal to one and all others be equal to zero (i.e., that there be only one node on the uppermost time level of the stencil):

$$a_{\boldsymbol{mn}} = \begin{cases} 1, & \text{for } (n_1,n_2,n_3) = (m_1,m_2,m_3),\ n_4 = m_4 + J_2, \\ 0, & \text{for } (n_1,n_2,n_3) \neq (m_1,m_2,m_3),\ n_4 = m_4 + J_2. \end{cases}$$

In other words, we require that the scheme (9) be explicit and normalized.

As the scheme (9) is a $(J+1)$-level explicit scheme, starting up the computation requires that the first $J$ time levels on the grid be initialized. We symbolically write it as follows:

$$\varphi_{\boldsymbol{n}}|_{n_4=-J_1}^{n_4=J_2-1} = \varphi_{\boldsymbol{n}}^0\ (=0). \tag{11}$$

In so doing, the first actually computed time level will be $j = J_2$. As we will show later (in the end of the consistency discussion below), *the data* $\varphi_{\boldsymbol{n}}^0$ *in* (11) *can always be chosen homogeneous,* $\varphi_{\boldsymbol{n}}^0 = 0$. Eqs. (11) obviously approximate initial conditions (2). For the schemes that employ finite differences of sufficiently high order in time, Eqs. (11) also represent the additional initial conditions required by the scheme only and not by the original continuous formulation.

As a simple example, we introduce a standard central-difference scheme that approximates problem (1), (2) on smooth solutions with the second order of accuracy. The stencil of this scheme consists of the following nine grid nodes ($J_1 = J_2 = 1$):

$$N_{\boldsymbol{m}} = \begin{cases} \{(m_1h,m_2h,m_3h,[m_4+1]\tau)\} \equiv N'_{m_1,m_2,m_3,m_4+1}, & j=1, \\ \left.\begin{array}{l} (m_1h,m_2h,m_3h,m_4\tau), \\ ([m_1\pm1]h,m_2h,m_3h,m_4\tau), \\ (m_1h,[m_2\pm2]h,m_3h,m_4\tau), \\ (m_1h,m_2h,[m_3\pm1]h,m_4\tau) \end{array}\right\} \equiv N'_{m_1,m_2,m_3,m_4}, & j=0, \\ \{(m_1h,m_2h,m_3h,[m_4-1]\tau)\} \equiv N'_{m_1,m_2,m_3,m_4-1}, & j=-1. \end{cases} \tag{12}$$

The coefficients $a_{mn}$ of the discrete operator and the values $f_m$ of the discrete right-hand side in (9) are defined as follows:

$$a_{mn} = \begin{cases} 1, & \text{if } \boldsymbol{n} = (m_1, m_2, m_3, m_4 + 1),\ \boldsymbol{n} = (m_1, m_2, m_3, m_4 - 1), \\ -r^2, & \text{if } \boldsymbol{n} = (m_1 \pm 1, m_2, m_3, m_4), \\ -r^2, & \text{if } \boldsymbol{n} = (m_1, m_2 \pm 1, m_3, m_4), \\ -r^2, & \text{if } \boldsymbol{n} = (m_1, m_2, m_3 \pm 1, m_4), \\ -2 + 6r^2, & \text{if } \boldsymbol{n} = (m_1, m_2, m_3, m_4), \end{cases} \tag{13}$$

$$f_m = \tau^2 f(m_1 h, m_2 h, m_3 h, m_4 \tau), \tag{14}$$

where $r$ is the Courant number

$$r = \frac{\tau}{h} \leqslant \frac{1}{c\sqrt{3}}. \tag{15}$$

Estimate (15) follows from the standard stability considerations of von Neumann type. Initial conditions (2) are replaced by the conditions

$$\varphi_{n_1,n_2,n_3,n_4} = 0, \quad n_4 = -1, 0. \tag{16}$$

Again, the scheme (12), (13), (14), (16), was introduced as an example only. As has been mentioned, any scheme that possesses the properties of stability and consistency on smooth solutions (see below), including high-order schemes, can be used for building the algorithm of the type described hereafter.

*Consistency.* We require that the finite-difference scheme that we use for calculating the approximate solution to problem (1), (2) possess the standard properties necessary for computation. Namely, we first require that the finite-difference equation (9) be consistent, i.e., approximate the differential equation (1) on smooth solutions $\varphi(\boldsymbol{x}, t)$:

$$\sup_{\boldsymbol{m}} \left| \sum_{\boldsymbol{n} \in N_m} a_{mn} \varphi(n_1 h, n_2 h, n_3 h, n_4 \tau) - f_m \right| \equiv \left\| \sum_{\boldsymbol{n} \in N_m} a_{mn} \varphi(n_1 h, n_2 h, n_3 h, n_4 \tau) - f_m \right\|_C^{(h)} \to 0,$$
$$h, \tau \to 0.$$

More precisely, we build the scheme (9) so that

$$\left\| \sum_{\boldsymbol{n} \in N_m} a_{mn} \varphi(n_1 h, n_2 h, n_3 h, n_4 \tau) - f_m \right\|_C^{(h)} \leqslant \sum_{\boldsymbol{\alpha}, \beta} \left\| \frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}} \frac{\partial^{\beta}}{\partial t^{\beta}} \varphi(\boldsymbol{x}, t) \right\|_C h^{|\boldsymbol{\alpha}|} \tau^{\beta}, \tag{17}$$

where estimate (17) is obtained using standard Taylor expansion technique for calculating the truncation error in finite-difference approximations. The set of indexes $\boldsymbol{\alpha}$ and $\beta$ in the sum on the right-hand side of inequality (17) corresponds to a particular collection of finite differences in the discrete operator of (9). *We emphasize that the solution $\varphi(\boldsymbol{x}, t)$ is always assumed sufficiently smooth so that to guarantee the estimate of type* (17) *for the particular choice of the scheme.* For example, for the central difference scheme (9), (12), (13), (14), (16) we have

$$\left\| \sum_{\boldsymbol{n} \in N_m} a_{mn} \varphi(n_1 h, n_2 h, n_3 h, n_4 \tau) - f_m \right\|_C^{(h)} \leqslant \sum_{i=1}^{3} \left\| \frac{\partial^4 \varphi}{\partial x_i^4} \right\|_C h^2 + \left\| \frac{\partial^4 \varphi}{\partial t^4} \right\|_C \tau^2. \tag{18}$$

Inequality (18) means that the aforementioned central-difference scheme approximates differential equation (1) with the second order of accuracy on the solutions $\varphi(\boldsymbol{x},t)$ with bounded fourth derivatives.

As has been just mentioned, consistency (17) is guaranteed for sufficiently smooth solutions $\varphi(\boldsymbol{x},t)$. In the end of Section 1.1 we have shown that *the requirement of smoothness for $\varphi(\boldsymbol{x},t)$ translates into a similar requirement for the right-hand side $f(\boldsymbol{x},t)$*. Using estimate (8) and assuming for generality that $r=\tau/h^{\eta}=\text{const}$ ($\eta=1$ in formula (15)), we rewrite the consistency inequality (17) as follows:

$$\left\|\sum_{\boldsymbol{n}\in N_{\boldsymbol{m}}} a_{\boldsymbol{mn}}\varphi(n_1h,n_2h,n_3h,n_4\tau)-f_{\boldsymbol{m}}\right\|_C^{(h)} \leqslant \frac{T_{\text{final}}^2}{2}\sum_{\boldsymbol{\alpha},\beta}\left\|\frac{\partial^{|\boldsymbol{\alpha}|}}{\partial\boldsymbol{x}^{\boldsymbol{\alpha}}}\frac{\partial^{\beta}}{\partial t^{\beta}}f(\boldsymbol{x},t)\right\|_C h^{|\boldsymbol{\alpha}|}\tau^{\beta}$$
$$\leqslant K_1(T_{\text{final}},f)h^{\min(|\boldsymbol{\alpha}|+\eta\beta)}. \tag{19}$$

Inequality (19) holds provided that all $(\partial^{|\boldsymbol{\alpha}|}/\partial\boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})f(\boldsymbol{x},t)$ required on the right-hand side are twice differentiable; $K_1$ is a constant that depends, generally speaking, on $T_{\text{final}}$ and $f$, but not on $h$. In particular, for the foregoing central-difference scheme we can rewrite (18) as ($r=\tau/h=\text{const}$):

$$\left\|\sum_{\boldsymbol{n}\in N_{\boldsymbol{m}}} a_{\boldsymbol{mn}}\varphi(n_1h,n_2h,n_3h,n_4\tau)-f_{\boldsymbol{m}}\right\|_C^{(h)} \leqslant \frac{T_{\text{final}}^2}{2}\left[\sum_{i=1}^{3}\left\|\frac{\partial^4 f}{\partial x_i^4}\right\|_C h^2+\left\|\frac{\partial^4 f}{\partial t^4}\right\|_C\tau^2\right]$$
$$\leqslant K_1(T_{\text{final}},f)h^2 \tag{20}$$

provided that $\partial^4 f/\partial x_l^4$, $l=1,2,3$, and $\partial^4 f/\partial t^4$ are $C^2$-smooth; $K_1$ in (20) is, again, a constant that may depend on $T_{\text{final}}$ and $f$, but not on the grid size $h$.

Let us reiterate that when referring to all those derivatives $(\partial^{|\boldsymbol{\alpha}|}/\partial\boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})f(\boldsymbol{x},t)$ that are required on the right-hand side of inequality (19), we actually mean that these derivatives, in turn, have to guarantee via estimate (8) the existence and boundedness of the corresponding terms in the truncation error expansion. Thus, the finite-difference equation (9) built of the stencil (10) determines the specific collection of indexes $\{(\boldsymbol{\alpha},\beta)\}$ required on the right-hand side of Eq. (19) in the exact same way as it determines a similar collection on the right-hand side of Eq. (17).

In particular, the maximal formal accuracy in time that a $J+1$-level stencil (10) can provide for the derivative $\partial^2\varphi/\partial t^2$ is $\mathrm{O}(\tau^J)$,[2] and the truncation error expansion in this case will start with $\partial^{J+2}\varphi/\partial t^{J+2}$. The requirement of boundedness for the latter derivative accordingly translates into the requirement of $\partial^{J+2}f/\partial t^{J+2}$ being $C^2$-smooth for $t\geqslant 0$. In terms of Section 1.1, we see that $\beta=J+2$, and to guarantee estimate (8) we require that $(\partial^j f/\partial t^j)|_{t=0}=0$ for $j=0,1,\ldots,\beta-1$ ($\beta-1=J+1$). Then, we can use the original differential equation (1) and initial conditions (2) and conclude that in this case $(\partial^j\varphi/\partial t^j)|_{t=0}=0$ for at least $j=0,1,\ldots,J+3$. *Consequently, we can always set $\varphi_n^0=0$ on the right-hand side of* (11).

*Stability.* We also require that the finite-difference scheme (9) be stable:

$$\|\varphi_{\boldsymbol{n}}\|_C\leqslant K_2\|f_{\boldsymbol{m}}\|_C,\qquad n_4\tau\leqslant T_{\text{final}}. \tag{21}$$

The constant $K_2$ in inequality (21) may depend on $T_{\text{final}}$, $K_2=K_2(T_{\text{final}})$, but cannot depend on the grid size. For the central-difference scheme (9), (12), (13), (14), (16) stability is guaranteed by (15).

[2] For simplicity, we assume here a "straightforward" finite-difference approximation. Replacing temporal derivatives in the error expansion with spatial ones via the PDE itself would widen the stencil in space. Compact approximations require extra smoothness of the right-hand side.

*Convergence.* Two inequalities (19) and (21) together imply convergence of the discrete solution $\varphi_{\boldsymbol{n}} \equiv \varphi_{n_1,n_2,n_3,n_4}$ to the continuous solution $\varphi(\boldsymbol{x},t)$ of problem (1), (2) on $\mathbb{R}^3 \times [0,\, T_{\text{final}}]$ as the grid size decreases:

$$\left\|\varphi(n_1h, n_2h, n_3h, n_4\tau) - \varphi_{n_1,n_2,n_3,n_4}\right\|_C^{(h)} \leqslant Kh^{\min(|\boldsymbol{\alpha}|+\eta\beta)}, \quad n_4\tau \leqslant T_{\text{final}}, \tag{22}$$

where $K = K(T_{\text{final}},\, f) = K_1 \cdot K_2$ is a constant that depends, generally speaking, on $T_{\text{final}}$ and $f$. The rate of convergence guaranteed by (22) is $\mathrm{O}(h^{\min(|\boldsymbol{\alpha}|+\eta\beta)})$; for the aforementioned central-difference scheme this rate is $\mathrm{O}(h^2)$.

### *1.3. Typical complications*

Speaking formally, estimate (22) allows one to use finite-difference scheme (9) for approximately calculating the solution $\varphi(\boldsymbol{x},t)$ of problem (1), (2) on arbitrarily long time intervals $0 \leqslant t \leqslant T_{\text{final}}$. *There are, however, two most substantial obstacles.* First, when calculating the solution using finite-difference equation (9) the number of grid nodes involved in the computation on each time level increases approximately as $(d/h + n_4)^3$ with the number of level $n_4$. Consequently, when $n_4 \approx T_{\text{final}}/\tau$ and, for example, $\tau = rh$ ($r = \text{const}$ is the Courant number), the number of nodes involved is of the same order of magnitude as $(T_{\text{final}}/h)^3$. (If $\tau = rh^\eta$, where $\eta > 1$, then the latter estimate will be even less favorable.) Therefore, the associated storage and CPU time requirements grow rapidly as the final time $T_{\text{final}}$ increases.

Second, estimate (22) guarantees grid convergence of the solution for any given time interval $[0,\, T_{\text{final}}]$ but this convergence is obviously not uniform in time (for a larger $T_{\text{final}}$, $K$ of (22) may also be larger). Indeed, besides the formal dependency of $K_1$ on $T_{\text{final}}$, see (19), the second component of $K$ from (22), i.e., the stability constant $K_2$, see (21), may also depend on $T_{\text{final}}$ (actually, grow with $T_{\text{final}}$). In other words, although for any initially prescribed $T_{\text{final}}$ we can achieve a desired accuracy by taking sufficiently small $h$, see (22), for a larger $T_{\text{final}}$ we may need to take a smaller $h$ ahead of time to achieve the same level of accuracy; and the dependency of this $h$ on $T_{\text{final}}$ may be strong and, in fact, prohibitive.

When calculating the solution $\varphi_{\boldsymbol{n}}$ using equation (9) on a fixed grid for long times, the aforementioned phenomenon translates into the accumulation of error by the algorithm; this accumulated error is going to eventually destroy the solution. From the standpoint of practical computing, the source of the error may, for example, be interpreted as either numerical dissipation ("drainage" of the energy from the system that takes place for dissipative schemes) or dispersion (frequency-dependent phase shift on the grid, which is unavoidable in multi-dimensional cases) or both. At any rate, this error is going to prevent us from accurately computing the solution on long time intervals using standard methodology (9).

Hereafter, we propose a technique for improving the standard scheme (9); this technique takes advantage of some special properties of the solution to problem (1), (2). The modified scheme guarantees that the error will not accumulate as the number $n_4$ of the time level increases. Moreover, both the memory and CPU time required for advancing each time step will remain bounded independently of $n_4$ (and $T_{\text{final}}$) once the grid sizes $h$ and $\tau$ are fixed.

The number of arithmetic operations required for advancing one time step using the new algorithm is $\mathrm{O}(N)$, where $N$ is the number of grid nodes in space (i.e., on one time level) inside a sphere of fixed diameter $d$; clearly, $N = \mathrm{O}(h^{-3})$. This number does not depend on $t$, i.e., does not increase with $n_4$ because unlike in the original scheme (9) the computational domain in the new algorithm will not need to expand in space as the time elapses. Obviously, the number $\mathrm{O}(N)$ is optimal (linear with respect to

the grid dimension) and cannot be improved by choosing any other algorithm. The required memory (number of words) in the new algorithm is of the order $O(N)$ as well.

The methodology that we propose for improving the original scheme (9) so that one can calculate the solution $\varphi(\boldsymbol{x}, t)$, $\boldsymbol{x} \in S(t)$, of problem (1), (2) on arbitrarily long time intervals, relies on a particular property of solutions to the three-dimensional wave equation (1), namely *the property of having lacunae*. Alternatively, this property is known as the Huygens' principle [12], or presence of aft fronts of the waves, in odd-dimension spaces. The lacunae-based technique is built here for calculating the solutions driven by moving sources and as such can be considered an extension of the technique developed previously in [9] for the case of stationary sources. In the future, the long-term lacunae-based integration methodology will be used to build global artificial boundary conditions (ABCs) for the numerical simulation of waves propagating with finite speed over infinite domains. The latter framework includes, in particular, the problems of both ambient and advective acoustics, as well as those of electromagnetic diffraction and scattering. The issue of ABCs is briefly touched upon in Appendix. In detail, the unsteady ABCs' methodology that we have mentioned will be described in the forthcoming publication [10]. A general survey of different ABCs' methodologies available in the literature can be found in the paper by Tsynkov [11].

The rest of the current paper is organized as follows. In Section 2, we describe the phenomenon of lacunae in the solutions of the three-dimensional wave equation. In Section 3, we show how one can make use of lacunae and modify any appropriate finite-difference scheme for the wave equation so that to allow for a non-deteriorating numerical integration of Eq. (1) over arbitrarily long time intervals. Provided that the original scheme is convergent in a particular sense, we demonstrate the same type of convergence of the modified algorithm uniformly in time. In Section 4 we provide some numerical results that corroborate the theoretical design properties of the lacunae-based algorithm. Finally, in Section 5 we briefly discuss possible generalizations of the new methodology.

## 2. Lacunae of the wave equation

### 2.1. Definition of lacunae

We return for the moment to the general continuous formulation of the Cauchy problem (1), (2) for the three-dimensional inhomogeneous wave equation with zero initial data. In this section, we do not make any specific assumptions regarding the right-hand side $f(\boldsymbol{x}, t)$ (like compact support) and simply suppose that it is a sufficiently smooth function with respect to all its arguments and that $f(\boldsymbol{x}, t) = 0$ for $t \leqslant 0$.

For every $(\boldsymbol{x}, t)$, the solution $\varphi = \varphi(\boldsymbol{x}, t)$ of problem (1), (2) can be written in the form of the Kirchhoff integral:

$$\varphi(\boldsymbol{x}, t) = \frac{1}{4\pi c^2} \iiint\limits_{\varrho \leqslant ct} \frac{f(\boldsymbol{\xi}, t - \varrho/c)}{\varrho} \, \mathrm{d}\boldsymbol{\xi}, \tag{23}$$

where $\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3)$, $\varrho = |\boldsymbol{x} - \boldsymbol{\xi}| = \sqrt{(x_1 - \xi_1)^2 + (x_2 - \xi_2)^2 + (x_3 - \xi_3)^2}$, and $\mathrm{d}\boldsymbol{\xi} = \mathrm{d}\xi_1 \, \mathrm{d}\xi_2 \, \mathrm{d}\xi_3$. The integration in (23) is performed over the ball of radius $ct$ centered at $\boldsymbol{x}$ in the space $\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3)$.

Formula (23), in fact, implies that the solution $\varphi(\boldsymbol{x},t)$ at the point $(\boldsymbol{x},t)$ depends only on the values of $f(\boldsymbol{\xi},\theta)$ on the surface of the characteristic cone (its lower portion) with the vertex $(\boldsymbol{x},t)$:

$$(x_1-\xi_1)^2+(x_2-\xi_2)^2+(x_3-\xi_3)^2=c^2(t-\theta)^2,\quad \theta<t, \tag{24a}$$

and does not depend on $f(\boldsymbol{\xi},\theta)$ when $(\boldsymbol{\xi},\theta)$ belongs to the interior of the cone (24a). In other words, changing the values of $f(\boldsymbol{\xi},\theta)$ in the interior of the cone (24a) will not affect the solution $\varphi(\boldsymbol{x},t)$ at the point $(\boldsymbol{x},t)$. To emphasize this circumstance, we will call the domain

$$(x_1-\xi_1)^2+(x_2-\xi_2)^2+(x_3-\xi_3)^2<c^2(t-\theta)^2,\quad \theta<t, \tag{24b}$$

i.e., the interior of the characteristic cone (24a), *the lacuna of the right-hand side* of Eq. (1) with respect to the point $(\boldsymbol{x},t)$. The presence of the lacuna (24b) of the right-hand side implies that *the solution* $\varphi(\boldsymbol{x},t)$ *of* (1), (2) *will also have a lacuna* $D^+(Q)$ with respect to the domain $Q$ of the right-hand side. Indeed, consider a $\delta$-source for equation (1) concentrated at the point $(\boldsymbol{\xi},\theta)$ of the space-time: $\delta(\boldsymbol{\xi},\theta)$. At any moment of time $t>\theta$, the solution of problem (1), (2) driven by this source will be concentrated on the surface of the sphere of radius $c(t-\theta)$ centered at $\boldsymbol{\xi}$ in the space $\boldsymbol{x}=(x_1,x_2,x_3)$. Inside this sphere, the solution will be identically zero: $\varphi(\boldsymbol{x},t)\equiv 0$ for $\varrho\equiv|\boldsymbol{x}-\boldsymbol{\xi}|<c(t-\theta)$. Therefore, let us now interpret the surface (24a) as the upper portion of the characteristic cone of equation (1) in the space-time $(\boldsymbol{x},t)$ with the vertex $(\boldsymbol{\xi},\theta)$. Then, the solution of (1), (2) driven by $\delta(\boldsymbol{\xi},\theta)$ is zero in the interior of the cone (24a), i.e., on the domain (24b) that we now denote by $D^+(\boldsymbol{\xi},\theta)$,

$$D^+(\boldsymbol{\xi},\theta)=\big\{(\boldsymbol{x},t)\mid|\boldsymbol{x}-\boldsymbol{\xi}|<c(t-\theta),\ t>\theta\big\}, \tag{24c}$$

and call *the lacuna of the fundamental solution of the wave equation.* (Note, this fundamental solution is actually a single layer on the spherical surface $|\boldsymbol{x}-\boldsymbol{\xi}|=c(t-\theta)$, $t>\theta$.) If we consider a general source $f(\boldsymbol{\xi},\theta)$ rather than the $\delta$-source $\delta(\boldsymbol{\xi},\theta)$, then for every particular $(\boldsymbol{\xi},\theta)$ the solution of (1), (2) inside the lacuna $D^+(\boldsymbol{\xi},\theta)$ given by (24c) does not depend on the value of $f(\boldsymbol{\xi},\theta)$ at this point $(\boldsymbol{\xi},\theta)$. By the superposition principle, the solution of (1), (2) with a general source $f$ will be concentrated on the set given by the union of all spheres $|\boldsymbol{x}-\boldsymbol{\xi}|=c(t-\theta)$, $t>\theta$, when the vertex $(\boldsymbol{\xi},\theta)$ of the cone (24a) sweeps the support of the right-hand side $f(\boldsymbol{\xi},\theta)$. Accordingly, the intersection of all $D^+(\boldsymbol{\xi},\theta)$ of (24c) for all $(\boldsymbol{\xi},\theta)\in Q$ will be called *the lacuna of the solution* $\varphi(\boldsymbol{x},t)$ *with respect to the domain* $Q$:

$$D^+(Q)=\bigcap_{(\boldsymbol{\xi},\theta)\in Q} D^+(\boldsymbol{\xi},\theta). \tag{24d}$$

Clearly, the solution $\varphi(\boldsymbol{x},t)$ of (1), (2) is zero on $D^+(Q)$ of (24d),

$$\varphi(\boldsymbol{x},t)\equiv 0\quad \text{for } (\boldsymbol{x},t)\in D^+(Q) \tag{25}$$

if

$$\operatorname{supp} f\subseteq Q. \tag{26}$$

Alternatively, one can say that changing the values of $f(\boldsymbol{\xi},\theta)$ on the domain $Q$ is not going to affect the solution $\varphi(\boldsymbol{x},t)$ of (1), (2) in the points of the lacuna $D^+(Q)$ given by (24d). In other words, we see that the waves governed by the three-dimensional wave equation (1) have *aft fronts.* If the source is compactly supported in both space and time, then at any given location $\boldsymbol{x}$ in space the solution $\varphi(\boldsymbol{x},t)$ becomes identically zero after a finite interval of time. This finite time interval is the time from the moment of source inception till the moment when the point $\boldsymbol{x}$ falls into the lacuna $D^+(Q)$ given by (24d), or in other

words, till all the waves generated by the source have passed through $\boldsymbol{x}$ and accordingly, the solution there has become zero again.

If the domain $Q$ is defined as follows (see Section 1.1)

$$Q = \{(\boldsymbol{x}, t) \mid \boldsymbol{x} \in S(t),\ t_0 < t < t_1\}, \tag{27}$$

then condition (26) implies that the solution $\varphi(\boldsymbol{x}, t)$ of (1), (2) satisfies the identities

$$\varphi(\boldsymbol{x}, t) \equiv 0, \quad \text{for } t \leqslant t_0 \tag{28a}$$

and

$$\varphi(\boldsymbol{x}, t) \equiv 0, \quad \text{for } \boldsymbol{x} \in S(t),\ t \geqslant t_2 \equiv t_0 + \frac{d + (t_1 - t_0)(c + k)}{c - k}. \tag{28b}$$

The first identity, (28a), is obvious, it takes place because the initial data of the Cauchy problem are homogeneous, see (2). The second identity, (28b) holds in virtue of (26) because the region of the space-time $(\boldsymbol{x}, t)$ defined as $\{\boldsymbol{x} \in S(t),\ t \geqslant t_2\}$, see (28b), is is completely contained inside the lacuna $D^+(Q)$ of (24d). *In other words, as long as* (4) *holds the time interval* $(d + (t_1 - t_0)(c + k))/(c - k)$ *is sufficient for all the waves generated by the sources inside* $S(t)$ *during* $t_0 \leqslant t \leqslant t_1$ *to completely leave the moving domain* $S(t)$. To see that (28b) is indeed correct one only has to realize that the sources concentrated on the ball $S(t_0)$ of diameter $d$ at the moment $t = t_0$, can be anywhere inside the sphere of diameter $d + 2k(t_1 - t_0)$ at the moment $t = t_1$ (the larger sphere is centered at the same location as $S(t_0)$), see Fig. 1. At $t = t_1$ the sources inside $S(t)$ cease to operate, and as the waves travel faster than the domain $S(t)$, $c > k$, all the waves generated prior to $t = t_1$ will eventually leave $S(t)$. This will happen at the moment $t = t_2$, when the wave emitted by the "leftmost" possible location of the source at $t = t_1$ pass the "rightmost" possible location of $S(t)$, which is schematically depicted in Fig. 1. A simple calculation yields the expression for $t_2$ given by formula (28b).

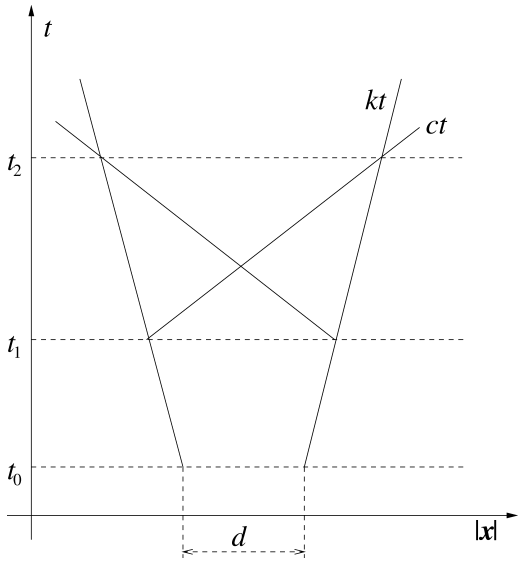


Fig. 1. Waves generated by a compactly supported source leaving the domain.

For the case of stationary sources, $k = 0$, the inequality of (28b) reduces to the obvious estimate $t \geqslant t_1 + d/c$, see [9]. Let us also note that the estimate for $t$ given in (28b) is, in fact, conservative, it does not use any assumptions regarding the character of the source motion except that its maximal speed is $k < c$. If, however, we make an additional assumption regarding the motion of the sphere, e.g., that it moves with a constant speed $k$ in some prescribed direction, then the estimate of (28b) can be improved and instead we obtain

$$\varphi(\boldsymbol{x}, t) \equiv 0, \quad \text{for } \boldsymbol{x} \in S(t),\ t \geqslant t_1 + \frac{d}{c-k}. \tag{28c}$$

For a stationary source, $k = 0$, (28c) again reduces to $t \geqslant t_1 + d/c$ of [9].

### 2.2. Decomposition of the source function in time

Let us now introduce the following partition of unity. Define the function

$$\Theta(t) = \begin{cases} \frac{1}{2} + P\left(t - \frac{1}{2}\right), & 0 \leqslant t \leqslant 1, \\ \frac{1}{2} + P\left(\frac{1}{2} - t\right), & -1 \leqslant t \leqslant 0, \\ 0, & |t| > 1, \end{cases} \tag{29}$$

where $P(t)$ is a continuously differentiable odd function: $(\mathrm{d}^j P(t)/\mathrm{d}t^j) \in C$, $j = 1, 2, \ldots, \beta + 2$, $P(-t) = -P(t)$, defined for $t \in [-\frac{1}{2}, \frac{1}{2}]$ and such that $P(\frac{1}{2}) = -\frac{1}{2}$, $(\mathrm{d}^j P(t)/\mathrm{d}t^j)|_{t=1/2} = 0$, $j = 1, 2, \ldots, \beta + 2$, where $\beta + 2$ is the maximal smoothness of the right-hand side $f(\boldsymbol{x}, t)$ with respect to time $t$ required by the consistency estimate (19). In particular, $P(t)$ can be a polynomial of the type $P(t) = \sum_j c_j t^{2j+1}$ with the coefficients $c_j$ chosen so that to satisfy the required level of smoothness, as well as the aforementioned conditions for $P(t)$ and its derivatives at $t = \frac{1}{2}$.

In so doing, $\Theta(t)$ given by (29) is an even function of $t$ with $\beta + 2$ continuous derivatives for all $t \in (-\infty, +\infty)$, and also $\Theta(t)$ is compactly supported, $\Theta(t) \equiv 0$ for $|t| \geqslant 1$, i.e.,

$$\operatorname{supp} \Theta(t) = [-1, 1].$$

Specify now a parameter $T$ and introduce the functions

$$\Theta_j(t, T) = \Theta\left(\frac{t - jT}{T}\right), \quad j = 0, 1, 2, \ldots .$$

Clearly,

$$\operatorname{supp} \Theta_j(t, T) = \big[(j-1)T, (j+1)T\big], \quad j = 0, 1, 2, \ldots .$$

Moreover, for any $T > 0$ because of the foregoing properties of $P(t)$ we have

$$\sum_{j=0}^{\infty} \Theta_j(t, T) \equiv 1, \quad t \geqslant 0. \tag{30}$$

The representation of a function, which is identically equal to 1, in the form (30) is a partition of unity. Notice that for every given $t$ no more than two terms on the left-hand side of the identity (30) may differ from zero.

We now represent the right-hand side $f(\boldsymbol{x},t)$ of Eq. (1) in the form

$$f(\boldsymbol{x},t)=f(\boldsymbol{x},t)\sum_{j=0}^{\infty}\Theta_j(t,T)=\sum_{j=0}^{\infty}\Theta_j(t,T)f(\boldsymbol{x},t)=\sum_{j=0}^{\infty}f_j(\boldsymbol{x},t,T), \tag{31}$$

where $f_j(\boldsymbol{x},t,T)=\Theta_j(t,T)f(\boldsymbol{x},t)$. Clearly,

$$\operatorname{supp} f_j(\boldsymbol{x},t,T)=Q_j(T), \tag{32}$$

$$Q_j(T)=\big\{(\boldsymbol{x},t)\mid \boldsymbol{x}\in S(t),\ (j-1)T\leqslant t\leqslant (j+1)T\big\}. \tag{33}$$

For $j=0$ we, in fact, will have $jT\leqslant t\leqslant (j+1)T$ in (33), because $f(\boldsymbol{x},t)=0$ for $t\leqslant 0$, see (5). Consider now the following sequence of problems:

$$\begin{aligned}&\frac{\partial^2\varphi_j}{\partial t^2}-c^2\left(\frac{\partial^2\varphi_j}{\partial x_1^2}+\frac{\partial^2\varphi_j}{\partial x_2^2}+\frac{\partial^2\varphi_j}{\partial x_3^2}\right)=f_j(\boldsymbol{x},t,T),\\ &\varphi_j|_{t=(j-1)T}=\left.\frac{\partial\varphi_j}{\partial t}\right|_{t=(j-1)T}=0,\quad j=0,1,2,\ldots.\end{aligned} \tag{34}$$

Note, for $j=0$, initial conditions in (34) are specified at $t=0$ rather than $t=-T$. For $j>0$ it is clear, in fact, that these homogeneous initial conditions can be specified at any moment of time before $(j-1)T$ because the right-hand side $f_j(\boldsymbol{x},t,T)$ kicks in only at $t=(j-1)T$ and thus we can consider $\varphi_j(\boldsymbol{x},t,T)=0$ for $t\leqslant (j-1)T$. Because of the linearity of problem (1), (2) and representation of $f(\boldsymbol{x},t)$ in the form of sum (31), the solution $\varphi(\boldsymbol{x},t)$ of problem (1), (2) can also be represented as a similar sum

$$\varphi(\boldsymbol{x},t)=\sum_{j=0}^{\infty}\varphi_j(\boldsymbol{x},t,T), \tag{35}$$

where $\varphi_j(\boldsymbol{x},t,T)$ is the solution of problem (34) for a specific $j$.

Let us now show that for $\boldsymbol{x}\in S(t)$ and any fixed $t>0$ there are only a few values of $j$ for which $\varphi_j(\boldsymbol{x},t,T)\neq 0$. First, we apply identities (28a) and (28b) which hold under conditions (26), (27) to the solution $\varphi_j(\boldsymbol{x},t,T)$ of problem (34). In so doing, instead of (26), (27) we use (32), (33). Then, instead of (28a) and (28b) we obtain the following two identities

$$\varphi_j(\boldsymbol{x},t,T)\equiv 0,\quad \text{for } t\leqslant (j-1)T \tag{36a}$$

and

$$\varphi_j(\boldsymbol{x},t,T)\equiv 0,\quad \text{for } \boldsymbol{x}\in S(t),\ t\geqslant (j-1)T+\frac{d+2T(c+k)}{c-k}. \tag{36b}$$

Identities (36) imply that for any given $t$ and $T$ the solution $\varphi_j(\boldsymbol{x},t,T)$ may differ from zero for $\boldsymbol{x}\in S(t)$ only if the following two inequalities hold simultaneously

$$(j-1)T<t, \tag{37a}$$

$$t<(j-1)T+\frac{d+2T(c+k)}{c-k}. \tag{37b}$$

A fixed prescribed $t=\widetilde{t}$ can meet both conditions (37) if and only if the index $j$ satisfies the inequalities

$$1+\frac{\widetilde{t}}{T}-\frac{d+2T(c+k)}{(c-k)T}<j<1+\frac{\widetilde{t}}{T}. \tag{38}$$

Therefore, there is only a finite number of values $j$ for which $\varphi_j(\boldsymbol{x}, t, T)$ differs from zero for $\boldsymbol{x} \in S(t)$ and a given $t$. If $k = 0$ and $T > d/c$ (and also $\widetilde{t}$ is sufficiently large) then there is no more than three such values of $j$. If $T \to +0$ or $k \to c$ then the number of indexes $j$ that satisfy (38) increases with no bound.

Henceforth, we will be using representation (35) for the solution $\varphi(\boldsymbol{x}, t)$ of problem (1), (2). We note that the term $\varphi_j(\boldsymbol{x}, t, T)$ in formula (35) is of interest for us only till the moment

$$\overline{t}_j = (j-1)T + \frac{d + 2T(c+k)}{c-k}, \tag{39}$$

as starting from this moment the component $\varphi_j(\boldsymbol{x}, t, T)$ turns into zero inside the computational domain $S(t)$ because of (36b), see also Fig. 1, and therefore no longer contributes into the sum (35). The moment $\overline{t}_j$ given by (39) is actually calculated as $t_2$ of (28b) by assuming that $t_0 = (j-1)T$ and $t_1 - t_0 = 2T$. In other words, the portion $\varphi_j(\boldsymbol{x}, t, T)$ of the overall solution $\varphi(\boldsymbol{x}, t)$ is present on the domain $S(t)$ only during a finite fixed interval of time:

$$T_{\text{interval}} = t_2 - t_0 \equiv \overline{t}_j - (j-1)T = \frac{d + 2T(c+k)}{c-k}, \tag{40}$$

which starts at $t_0 = (j-1)T$ and ends at $t_2 = \overline{t}_j$, see (39). *It is very important that the length of this interval* $T_{\text{interval}}$, *see* (40), *does not depend on* $j$. This allows us to conclude that similarly to estimate (8), we can obtain the following estimate for $\boldsymbol{x} \in S(t)$ and $(j-1)T \leqslant t \leqslant \overline{t}_j$ for all appropriate $\boldsymbol{\alpha}$ and $\beta$ and for all $j = 0, 1, 2, \ldots$:

$$\left\| \frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}} \frac{\partial^{\beta}}{\partial t^{\beta}} \varphi_j(\boldsymbol{x}, t, T) \right\|_C \leqslant \frac{T_{\text{interval}}^2}{2} \left\| \frac{\partial^{|\boldsymbol{\alpha}|}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}} \frac{\partial^{\beta}}{\partial t^{\beta}} f_j(\boldsymbol{x}, t, T) \right\|_C. \tag{41}$$

*We emphasize that unlike in* (8) *the multiplier* $T_{\text{interval}}^2/2$ *in inequality* (41) *does not depend on* $T_{\text{final}}$. Let us also note that the sphere $S(t)$ of diameter $d$ centered at $\boldsymbol{x}_0(t) \equiv (x_1^0(t), x_2^0(t), x_3^0(t))$ represents at the time moment $t = \overline{t}_j$ given by (39) *the aft front* of the propagation of $\varphi_j(\boldsymbol{x}, t, T)$ over the unperturbed zero background. In many cases the spherical surface $S(t)$ may, in fact, be a conservative estimate for the actual location of the aft front; but at any rate, $S(t)$ is always inside the aft front.

*Numerical algorithm proposed in Section* 3 *below is based on the idea that when calculating the solution* $\varphi(\boldsymbol{x}, t)$ *of* (1), (2), *for every* $t$ *we actually need to calculate only a few terms* $\varphi_j(\boldsymbol{x}, t, T)$ *in the sum* (35) *that differ from zero for* $\boldsymbol{x} \in S(t)$. *Each of these terms will drop out of the solution for* $t > \overline{t}_j$, *see formula* (39), *which essentially means that even when the total elapsed time is large, all calculations will still be performed only on a fixed predetermined time interval of length* $\sim T_{\text{interval}}$.

However, prior to actually describing the numerical algorithm, let us introduce an important new element of the formulation.

### 2.3. Periodization in space

Specify some $z > 0$ and consider the following problem that is periodic with the period $z$ in every coordinate direction $x_l$, $l = 1, 2, 3$:

$$\begin{aligned} &\frac{\partial^2 \psi_j}{\partial t^2} - c^2 \left( \frac{\partial^2 \psi_j}{\partial x_1^2} + \frac{\partial^2 \psi_j}{\partial x_2^2} + \frac{\partial^2 \psi_j}{\partial x_3^2} \right) = f_j(\boldsymbol{x}, t, T, z), \\ &\psi_j(\boldsymbol{x}, t, T, z) = 0, \quad t \leqslant (j-1)T, \end{aligned} \tag{42a}$$

$$\begin{aligned}
&\psi_j(x_1 + s_1 z, x_2 + s_2 z, x_3 + s_3 z, t, T, z) = \psi_j(\boldsymbol{x}, t, T, z),\\
&f_j(x_1 + s_1 z, x_2 + s_2 z, x_3 + s_3 z, t, T, z) = f_j(\boldsymbol{x}, t, T, z),\\
&\quad s_1,\ s_2,\ s_3 = 0, \pm 1, \pm 2, \ldots,\\
&f_j(\boldsymbol{x}, t, T, z) \equiv f_j(\boldsymbol{x}, t, T) \quad \text{if } |x_i| < z/2,\ i = 1, 2, 3.
\end{aligned} \tag{42b}$$

**Theorem 1.** *The solution* $\psi_j(\boldsymbol{x}, t, T, z)$ *of problem* (42) *coincides with the solution* $\varphi_j(\boldsymbol{x}, t, T)$ *of problem* (34)*,*

$$\varphi_j(\boldsymbol{x}, t, T) = \psi_j(\boldsymbol{x}, t, T, z), \tag{43}$$

*on the domain*

$$\boldsymbol{x} \in S(t), \quad (j-1)T \leqslant t < (j-1)T + \frac{z-d}{c+k}. \tag{44}$$

**Proof.** Let us first note that as long as $|\boldsymbol{u}_0| = |\mathrm{d}\boldsymbol{x}_0(t)/\mathrm{d}t| \leqslant k < c$ (see (4)), where $\boldsymbol{x}_0 = \boldsymbol{x}_0(t)$ may be any prescribed law of motion for the center of the sphere $S(t)$, the right-hand side $f_j(\boldsymbol{x}, t, T, z)$ of (42a), which is periodic in all three coordinate directions $x_1$, $x_2$, and $x_3$, may differ from zero only on the union of the balls $S_s(t)$, $\boldsymbol{s} = (s_1, s_2, s_3)$:

$$\begin{aligned}
&S_s(t) = \left\{ (x_1, x_2, x_3) \,\Big|\, \sum_{l=1}^{3} (x_l - s_l z)^2 \leqslant \left[ \frac{d}{2} + k(t - (j-1)T) \right]^2 \right\},\\
&\quad t \geqslant (j-1)T,\ s_1, s_2, s_3 = 0, \pm 1, \pm 2, \ldots
\end{aligned} \tag{45}$$

This actually follows from the fact that the sphere $S(t)$ for $t \geqslant (j-1)T$ completely belongs to the ball $S_{\boldsymbol{0}}(t)$, see (45). Moreover, it is easy to see that the lower portion of the characteristic cone (24a):

$$(x_1 - \xi_1)^2 + (x_2 - \xi_2)^2 + (x_3 - \xi_3)^2 = c^2(t - \theta)^2, \quad \theta < t,$$

with the vertex in an arbitrary point $(\boldsymbol{x}, t) \in S_{\boldsymbol{0}}(t)$ intersects none of the spherical domains $S_s(t)$ for $|\boldsymbol{s}|^2 \equiv s_1^2 + s_2^2 + s_3^2 \neq 0$ (i.e., none of the other balls (45)) on the time interval $(j-1)T \leqslant \theta \leqslant t$ if only

$$t < (j-1)T + \frac{z-d}{c+k}. \tag{46}$$

This argument actually becomes clear from geometric considerations, see Fig. 2. In this figure we schematically show the trajectories of $S_s(t)$ by straight lines $|\boldsymbol{x}| = \pm kt$ and the surface of the characteristic cone—by straight lines $|\boldsymbol{x}| = \pm ct$.

Then, provided that (46) holds, the Kirchhoff formula (23) implies that the value of the solution $\psi_j(\boldsymbol{x}, t, T, z)$ in the vertex $(\boldsymbol{x}, t)$ of the characteristic cone (24a) will not depend on the values of the right-hand side $f_j(\boldsymbol{\xi}, \theta, T, z)$ of Eq. (42a) on the domains $S_s(t)$ for $|\boldsymbol{s}| \neq 0$. In particular, the value $\psi_j(\boldsymbol{x}, t, T, z)$ will not change if the right-hand sides $f_j(\boldsymbol{\xi}, \theta, T, z)$ on all $S_s(t)$, $\boldsymbol{s} \neq \boldsymbol{0}$, were replaced by the identical zero for all $\theta \leqslant t$, where $t$ satisfies inequality (46). On the other hand, this replacement makes the right-hand side of (42a) coincide with the non-periodic right-hand side of equation (35), which has the solution $\varphi_j(\boldsymbol{x}, t, T)$. Thus, $\psi_j(\boldsymbol{x}, t, T, z) = \varphi_j(\boldsymbol{x}, t, T)$ for all those $(\boldsymbol{x}, t)$, for which $t$ satisfies (46) and $\boldsymbol{x}$ belongs to $S_{\boldsymbol{0}}(t)$. At the same time, it has been mentioned that $S(t) \subset S_{\boldsymbol{0}}(t)$ for any $t \geqslant (j-1)T$. This proves the theorem. □

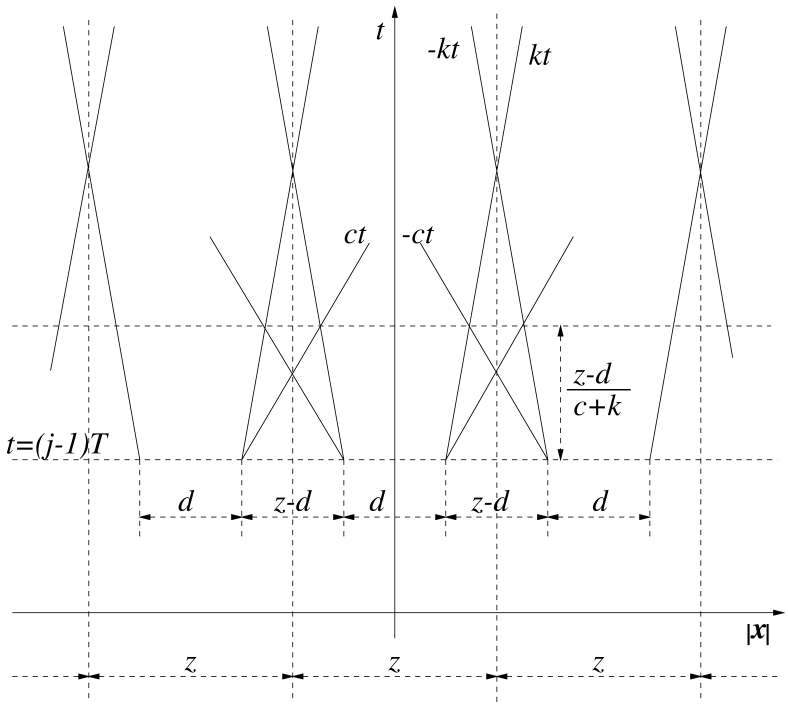


Fig. 2. Depiction of the periodic setup.

## 3. Finite-difference algorithm

In this section, we construct a non-deteriorating algorithm for the approximate calculation of the solution to problem (1), (2) on arbitrarily long time intervals using finite-difference equation (9). In fact, we are going to present three slightly different versions of this algorithm. All three versions will guarantee that the error will not accumulate with time (uniform grid convergence), and computer expenses per time step (both CPU time-wise and storage-wise) will not increase, i.e., will remain fixed and bounded throughout the entire computation. All three versions will also have the same non-improvable computational complexity, i.e., asymptotic order of the number of required arithmetic operations and amount of memory with respect to the grid size. However, the algorithms will differ from one another by the actual computer resources required (while still having the same asymptotic order), as well as by certain convenience features from the standpoints of both theoretical analysis and practical computing.

Hereafter, we assume that $T > 0$, $h > 0$, $\tau > 0$, and $z > 0$ are chosen so that $T/\tau$ and $z/h$ are positive integers, and $\tau/h^{\eta} = r = \text{const}$.

### *3.1. Basic non-deteriorating algorithm*

This version of the algorithm provides for the most convenient model to establish the fundamental desired property—*uniform grid convergence on arbitrarily long time intervals.* It is based on the

representation (35) of the solution $\varphi(\boldsymbol{x}, t)$ to problem (1), (2) on the ball $S(t)$. Let us fix some arbitrary integer $l \geqslant 1$ and consider $t$ from the interval

$$(l-1)T < t \leqslant lT. \tag{47}$$

For these $t$, formula (35) can be rewritten as follows:

$$\varphi(\boldsymbol{x}, t) = \varphi_{l-p}(\boldsymbol{x}, t, T) + \varphi_{l-p+1}(\boldsymbol{x}, t, T) + \cdots + \varphi_l(\boldsymbol{x}, t, T), \tag{48}$$

where $\varphi_j(\boldsymbol{x}, t, T)$, $j = l-p, l-p+1, \ldots, l$, are solutions of the corresponding problems (34). The positive integer number $p$ is chosen from the inequalities (47) and (38), i.e., so that for a given $t = \widetilde{t}$ that satisfies (47), the sum (48) contain only those $\varphi_j(\boldsymbol{x}, t, T)$ that differ from zero on $S(t)$—the corresponding $j$'s satisfy (38). Note, if for some small $l$ one or more indexes $l-p$, $l-p+1, \ldots$ appear negative, then we simply consider the sum

$$\varphi(\boldsymbol{x}, t) = \varphi_0(\boldsymbol{x}, t, T) + \cdots + \varphi_l(\boldsymbol{x}, t, T) \tag{49}$$

instead of (48). If, on the other hand, for a large $l$ instead of the smallest possible $p$ that satisfies the foregoing constraints (the constraints that follow from (47) and (38)), one takes, e.g., $p+1$, then an additional term $\varphi_{l-p-1}(\boldsymbol{x}, t, T)$ will simply appear in the sum (48). This term, however, will turn into zero for $\boldsymbol{x} \in S(t)$ and $t$ satisfying (47) and consequently, the work required for computing this term will be superfluous.

Assume, for definiteness, that $p$ is chosen according to the formula:

$$p = \left[\frac{d + 2T(c+k)}{(c-k)T}\right] + 1 = \left[\frac{T_{\text{interval}}}{T}\right] + 1, \tag{50}$$

where $[\cdot]$ denotes the integer part (see (38) and (40)). We will be calculating the solution $\varphi(\boldsymbol{x}, t)$ of problem (1), (2) on the grid $\boldsymbol{n} = (n_1, n_2, n_3, n_4)$ for

$$(n_1 h, n_2 h, n_3 h) \in S(n_4\tau), \quad (l-1)T < n_4\tau \leqslant lT, \tag{51}$$

i.e., inside the sphere $S(t)$ for $t$ satisfying (47). According to (48) and (50), *the exact values* of this solution on the grid are given by

$$\begin{aligned}\varphi(n_1 h, n_2 h, n_3 h, n_4\tau) = {} & \varphi_{l-p}(n_1 h, n_2 h, n_3 h, n_4\tau, T) + \varphi_{l-p+1}(n_1 h, n_2 h, n_3 h, n_4\tau, T) + \cdots \\ & + \varphi_l(n_1 h, n_2 h, n_3 h, n_4\tau, T).\end{aligned} \tag{52}$$

Instead of the exact values (52) the first, or basic, version of the non-deteriorating algorithm that we are discussing generates *approximate values* of the solution $\varphi(\boldsymbol{x}, t)$ on the grid according to the formula:

$$\varphi(n_1 h, n_2 h, n_3 h, n_4\tau) \approx \varphi_{\boldsymbol{n}}^{(I)} \equiv \varphi_{\boldsymbol{n}}(l-p, T) + \varphi_{\boldsymbol{n}}(l-p+1, T) + \cdots + \varphi_{\boldsymbol{n}}(l, T). \tag{53}$$

Each term $\varphi_{\boldsymbol{n}}(j, T)$, $j = l-p, l-p+1, \ldots, l$, on the right-hand side of relation (53) solves the following finite-difference counterpart to problem (34):

$$\begin{aligned}&\sum_{\boldsymbol{n} \in N_{\boldsymbol{m}}} a_{\boldsymbol{m}\boldsymbol{n}} \varphi_{\boldsymbol{n}}(j, T) = f_{\boldsymbol{m}}(j, T), \qquad j = l-p, l-p+1, \ldots, l, \\ &\varphi_{\boldsymbol{n}}(j, T)\big|_{n_4=(j-1)T/\tau - J_1}^{n_4=(j-1)T/\tau + J_2 - 1} = 0, \qquad \varphi_{\boldsymbol{n}}(0, T)\big|_{n_4=-J_1}^{n_4=J_2-1} = 0.\end{aligned} \tag{54}$$

Of course, in (54) if $j > 0$ we can formally consider $\varphi_{\boldsymbol{n}}(j, T) = 0$ for all $n_4 \leqslant (j-1)T/\tau + J_2 - 1$. The right-hand side $f_{\boldsymbol{m}}(j, T)$ of (54) is given by the expression

$$f_{\boldsymbol{m}}(j, T) = f_j(\boldsymbol{x}, t, T)|_{(\boldsymbol{x}, t) = (m_1 h, m_2 h, m_3 h, m_4\tau)} = \Theta_j(t, T) f(\boldsymbol{x}, t)|_{(\boldsymbol{x}, t) = (m_1 h, m_2 h, m_3 h, m_4\tau)}$$

(see (31)–(33)). Obviously,

$$f_{\boldsymbol{m}}(j,T)=0 \quad \text{if } m_4\tau \leqslant (j-1)T. \tag{55}$$

**Theorem 2.** *The error due to the replacement of the true solution $\varphi(\boldsymbol{x},t)$ of problem* (1), (2) *given by the exact formula* (52) *for $\boldsymbol{x}\in S(t)$, $(l-1)T<t\leqslant lT$, by the difference solution $\varphi_{\boldsymbol{n}}^{(I)}\equiv\sum_j\varphi_{\boldsymbol{n}}(j,T)$ obtained with the basic algorithm according to the approximate formula* (53), *satisfies*

$$\left\|\varphi(n_1h,\,n_2h,\,n_3h,\,n_4\tau)-\varphi_{\boldsymbol{n}}^{(I)}\right\|_C^{(h)}\leqslant(p+1)\widetilde{K}h^{\min(|\boldsymbol{\alpha}|+\eta\beta)}, \tag{56}$$

*where $n_1$, $n_2$, $n_3$, and $n_4$ satisfy* (51), *the power of $h$ in* (56) *is the same as in* (22), *the constant $\widetilde{K}$ depends neither on the grid size $h$ nor on $l$, and $p+1$ is the number of terms in the sum* (53).

Briefly, the idea of the proof is that because of the presence of lacunae, the exact solution is represented in the form (48), where the sum contains only a finite non-increasing number of terms. Each of these terms needs to be calculated only on a predetermined time interval of fixed non-growing length, which guarantees that the constants in both consistency and stability estimates will not depend on the total time $T_{\text{final}}$. This, in turn, immediately provides the temporally uniform convergence.

**Proof.** The sum (52) represents the exact solution to problem (1), (2) on the grid for $\boldsymbol{x}\in S(t)$, $(l-1)T<t\leqslant lT$, because of the presence of lacunae. Each term $\varphi_j(\boldsymbol{x},t,T)$, $j=l-p,l-p+1,\ldots,l$, in this sum needs to be considered only on the corresponding time interval $(j-1)T\leqslant t\leqslant lT$. The length of the largest of these intervals ($j=l-p$) does not exceed $(p+1)T\leqslant T_{\text{interval}}+2T\equiv\mu T_{\text{interval}}$ ($\mu=\text{const}$), see Eq. (50), and, what is most important, does not depend on $l$. Consequently, for each of these $\varphi_j(\boldsymbol{x},t,T)$, $j=l-p,l-p+1,\ldots,l$, estimate (41) holds (with $T_{\text{interval}}$ replaced by $\mu T_{\text{interval}}$). The continuous solution $\varphi_j(\boldsymbol{x},t,T)$ of problem (34) for a particular $j$ is approximated by the finite-difference solution $\varphi_{\boldsymbol{n}}(j,t)$ of problem (54) for the same $j$. Accordingly, the discrete problem (54) also needs to be solved only for the same time interval $(j-1)T\leqslant t\leqslant lT$ of the maximal length $\mu T_{\text{interval}}$ that does not depend on $l$. Therefore, the consistency result (19) for a particular $j$ and $(j-1)T\leqslant m_4\tau\leqslant lT$ takes the form

$$\begin{aligned}\left\|\sum_{\boldsymbol{n}\in N_{\boldsymbol{m}}}a_{\boldsymbol{mn}}\varphi_j(n_1h,n_2h,n_3h,n_4\tau,T)-f_{\boldsymbol{m}}(j,T)\right\|_C^{(h)}&\leqslant\frac{\mu^2T_{\text{interval}}^2}{2}\sum_{\boldsymbol{\alpha},\beta}\left\|\frac{\partial^{|\boldsymbol{\alpha}|}}{\partial\boldsymbol{x}^{\boldsymbol{\alpha}}}\frac{\partial^{\beta}}{\partial t^{\beta}}f_j(\boldsymbol{x},t,T)\right\|_C h^{|\boldsymbol{\alpha}|}\tau^{\beta}\\&\leqslant K_1^{(j)}(T_{\text{interval}},f_j)h^{\min(|\boldsymbol{\alpha}|+\eta\beta)}.\end{aligned} \tag{57}$$

Regarding the constants $K_1^{(j)}\equiv K_1^{(j)}(T_{\text{interval}},f_j)$, they, as always, do not depend on the grid size $h$. Moreover, we can, in fact, claim that $\forall j=l-p,l-p+1,\ldots,l$: $K_1^{(j)}\leqslant\widetilde{K}_1\equiv\widetilde{K}_1(T_{\text{interval}},f)$. This is easy to see provided that the smoothness properties of $f(\boldsymbol{x},t)$ remain uniformly "good" in time; or in other words, $(\partial^{|\boldsymbol{\alpha}|}/\partial\boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})f(\boldsymbol{x},t)$ remain continuous and bounded for all $t\geqslant0$. In this case, all $(\partial^{|\boldsymbol{\alpha}|}/\partial\boldsymbol{x}^{\boldsymbol{\alpha}})(\partial^{\beta}/\partial t^{\beta})f_j(\boldsymbol{x},t,T)$, see (57) will also remain continuous and uniformly bounded because the function $\Theta(t)$ of (29) that helps us build the partition of unity (30) and the partition of the right-hand side (31) was specially chosen sufficiently smooth so that not to decrease the extent of smoothness of the original $f(\boldsymbol{x},t)$.

Stability estimate (21) for the discrete scheme (54) can be rewritten as

$$\left\|\varphi_{\boldsymbol{n}}(j,T)\right\|_C\leqslant K_2^{(j)}\left\|f_{\boldsymbol{m}}(j,T)\right\|_C,\quad (j-1)T\leqslant n_4\tau\leqslant lT, \tag{58}$$

where the constant $K_2^{(j)}$ depends on the actual length of time interval $(j-1)T \leqslant t \leqslant lT$, but as none of these intervals exceeds $\mu T_{\text{interval}}$ we can again say that $\forall j = l-p, l-p+1, \ldots, l$: $K_2^{(j)} \leqslant \widetilde{K}_2 \equiv \widetilde{K}_2(T_{\text{interval}})$. The constant $\widetilde{K}_2$ obviously does not depend on $l$ and neither does it depend on the grid size $h$.

A standard argument then yields that for each term $\varphi_{\boldsymbol{n}}(j,T)$, $j = l-p, l-p+1, \ldots, l$, on the right-hand side of formula (53) the convergence estimate of type (22) will hold:

$$\begin{aligned}\left\|\varphi_j(n_1h, n_2h, n_3h, n_4\tau) - \varphi_{\boldsymbol{n}}(j,T)\right\|_C^{(h)} &\leqslant K_1^{(j)} K_2^{(j)} h^{\min(|\boldsymbol{\alpha}|+\eta\beta)} \\ &\leqslant \widetilde{K}_1 \widetilde{K}_2 h^{\min(|\boldsymbol{\alpha}|+\eta\beta)} \equiv \widetilde{K} h^{\min(|\boldsymbol{\alpha}|+\eta\beta)}. \end{aligned} \tag{59}$$

The constant $\widetilde{K}$ in (59) obviously depends neither on the actual elapsed time $t = lT$ nor on the grid sizes. Remembering now that there are $p+1$ terms altogether in the sum (53), we immediately arrive at the estimate (56) and thus prove the theorem. □

*We emphasize that Theorem* 2 *implies temporally uniform grid convergence of the discrete solution given by the basic algorithm to the original continuous solution of the domain $S(t)$ on arbitrarily long time intervals.* Indeed, as opposed to the original convergence estimate (22), where the constant $K$ depends on the final time $T_{\text{final}}$, and actually grows with the increase of $T_{\text{final}}$, the constant $\widetilde{K}$ of (56) depends on $T_{\text{interval}}$ and remains fixed for arbitrarily large times $T_{\text{final}}$ (again, as long as the smoothness properties of $f(\boldsymbol{x},t)$ remain uniformly "good" with respect to $t$).

Let us now estimate the computer resources required by the proposed basic algorithm. Clearly, both the operations count and the amount of memory (i.e., number of words) needed for advancing one time step when calculating each term $\varphi_{\boldsymbol{n}}(j,T)$ of (53) are of the same asymptotic order $\mathrm{O}(h^{-3})$ with respect to the grid size $h$. The number of terms $p+1$ does not change (i.e., does not grow) when the grid is refined as long as $T$ is fixed. Therefore, neither does the overall number of arithmetic operations, as well as the amount of memory, required when calculating the solution by means of formula (53)—both quantities remain of the order $\mathrm{O}(h^{-3})$. The number of grid nodes that belong to the sphere $S(t)$ for a fixed $t = n_4\tau$ is also of the order $\mathrm{O}(h^{-3})$; therefore, the foregoing algorithm appears asymptotically non-improvable—*linear with respect to the grid dimension*—as long as one uses scheme (9).

We also note that as we compute each of the terms $\varphi_{\boldsymbol{n}}(j,T)$, $j = l-p, l-p+1, \ldots, l$, on a finite interval of time, this computation also requires only a finite domain in space because the perturbations propagate with finite speed $c$; beyond this finite region across which the perturbation can propagate, the solution is equal to zero (in the exterior of the union of all characteristic cones with the vertexes sweeping the support of the right-hand side). Thus, the spatial domain, which originally was infinite, can, in fact, be truncated. Out of the several terms that need to be computed according to (53), the first one, $\varphi_{\boldsymbol{n}}(l-p,\,T)$, is the most expensive numerically, its calculation by an explicit scheme up to the time level $t = n_4\tau = lT$ requires the widest grid domain of the size approximately

$$\begin{aligned} d + 2(p+1)Tc \approx d + 2(T_{\text{interval}} + 2T)c &= d + 2\left(\frac{d}{c-k} + \frac{4cT}{c-k}\right)c \\ &= \frac{1}{c-k}\left((3c-k)d + 8c^2T\right). \end{aligned} \tag{60}$$

Dividing (60) by $h$ and taking the third power of the result, we will obtain a quantity of the order $\mathrm{O}(h^{-3})$ (as long as $T$ is fixed while the grid is refined). This quantity gives an estimate of what will be the actual amount of memory needed for advancing one time step using the first, basic, version of the algorithm.

Finally, let us mention that when discussing the long-time integration we can consider a formulation that slightly differs from (47). Considering $t$ from the interval (47) means, in fact, that $l$ can be arbitrarily large and we calculate the solution on the interval of a fixed length $T$, which can be placed arbitrarily far away from the initial data. Alternatively, one may be interested in knowing the overall temporal evolution of the solution, i.e., in calculating the solution on an arbitrarily long time interval, say from 0 to some large $T_{\text{final}}$. From the standpoint of building a non-deteriorating lacunae-based algorithm, this formulation is not much different from the one analyzed previously. For every time interval (47), $T \ll T_{\text{final}}$, i.e., every $l$, the solution will still be computed using formula (53). To advance further in time, we then need to replace $l$ by $l+1$ in formula (47). This will simply imply dropping the first term $\varphi_{\boldsymbol{n}}(l-p,T)$ on the right-hand side of formula (53) and adding the new last term $\varphi_{\boldsymbol{n}}(l+1,T)$. In so doing, each term $\varphi_{\boldsymbol{n}}(l-q,T)$, $q=0,1,\ldots,p-1$, for the previous interval $l$ becomes $\varphi_{\boldsymbol{n}}(l+1-(q+1),T)$ for the new interval $l+1$. Of course, for the new interval there is no need to calculate this term from the very beginning by solving the corresponding problem (54) starting from $j=l-q$; the calculation of each term of (53) that is not dropped when going from $l$ to $l+1$ (i.e., every term except the first one) is rather continued from the previous interval using the same explicit scheme. A further modification of the algorithm, which is described in Section 3.3 below, in fact uses the aforementioned interpretation of the long-term integration.

### 3.2. Modification based on periodization

As has been mentioned in Section 3.1, the actual size of the computational domain that we need for using the basic non-deteriorating algorithm is given by the estimate (60). In this section, we modify the basic algorithm so that to make use of the periodic formulation introduced in Section 2.3 and thus reduce the size of the computational domain and consequently, the required computer resources.

In the periodic version of the algorithm, instead of formula (48) we use the following representation of the solution $\varphi(\boldsymbol{x},t)$ for $\boldsymbol{x}\in S(t)$, $(l-1)T<t\leqslant lT$:

$$\varphi(\boldsymbol{x},t)=\psi_{l-p}(\boldsymbol{x},t,T,z)+\psi_{l-p+1}(\boldsymbol{x},t,T,z)+\cdots+\psi_l(\boldsymbol{x},t,T,z), \tag{61}$$

here $\psi_j(\boldsymbol{x},t,T,z)$, $j=l-p,l-p+1,\ldots,l$, are solutions of the corresponding problems (42). For small $l$ we, similarly to (49), may need to use representation

$$\varphi(\boldsymbol{x},t)=\psi_0(\boldsymbol{x},t,T,z)+\cdots+\psi_l(\boldsymbol{x},t,T,z) \tag{62}$$

instead of (61). When using either (61) or (62), the period $z$ obviously has to be chosen so that for every function $\varphi_j(\boldsymbol{x},t,T)$, $j=l-p,l-p+1,\ldots,l$, the equality

$$\varphi_j(\boldsymbol{x},t,T)=\psi_j(\boldsymbol{x},t,T,z)$$

hold on the entire time interval

$$(j-1)T<t<(j-1)T+\frac{d+2T(c+k)}{c-k}\equiv(j-1)T+T_{\text{interval}}, \tag{63}$$

on which according to formulae (37) the function $\varphi_j(\boldsymbol{x},t,T)$ may differ from zero for $\boldsymbol{x}\in S(t)$ (see also (40)). In other words, we require that the time interval (63) be contained inside the time interval (44), for which according to Theorem 1 the periodic and non-periodic solutions coincide:

$$(j-1)T+\frac{d+2T(c+k)}{c-k}\leqslant(j-1)T+\frac{z-d}{c+k}. \tag{64}$$

The latter inequality yields the following lower bound for the period $z$:

$$z \geqslant \frac{1}{c-k}\left(2cd + 2(c+k)^2 T\right). \tag{65}$$

*Condition* (65) *essentially guarantees that all the waves generated by the sources inside* $S(t)$ *that operate on the time interval* $2T$ *will leave the domain of interest* $S(t)$ *before the waves generated by all other sources from the periodic structure can enter this domain.*

To actually build the periodic algorithm, we replace the differential equation and initial condition of (42a) by the finite-difference equation and discrete initial condition, respectively:

$$\begin{aligned} &\sum_{\boldsymbol{n}\in N_{\boldsymbol{m}}} a_{\boldsymbol{mn}}\psi_{\boldsymbol{n}}(j,T,z) = f_{\boldsymbol{m}}(j,T,z), \quad j = l-p, l-p+1,\ldots,l, \\ &\psi_{\boldsymbol{n}}(j,T,z)\big|_{n_4=(j-1)T/\tau-J_1}^{n_4=(j-1)T/\tau+J_2-1} = 0, \qquad \psi_{\boldsymbol{n}}(0,T,z)\big|_{n_4=-J_1}^{n_4=J_2-1} = 0, \end{aligned} \tag{66a}$$

where the right-hand side $f_{\boldsymbol{m}}(j,T,z)$ is a $z$-periodic grid function with node values

$$f_{\boldsymbol{m}}(j,T,z) = f_j(\boldsymbol{x},t,T,z)|_{(\boldsymbol{x},t)=(m_1h,m_2h,m_3h,m_4\tau)}.$$

Again, in (66a) if $j > 0$ we can formally consider $\psi_{\boldsymbol{n}}(j,T,z) = 0$ for all $n_4 \leqslant (j-1)T/\tau + J_2 - 1$. The periodicity boundary conditions (42b) are replaced by their discrete counterparts in every coordinate direction $x_1$, $x_2$, and $x_3$ (the ratio of the period $z$ and grid size $h$, $z/h \equiv b$, is assumed positive integer):

$$\begin{aligned} &\psi_{\boldsymbol{n}}(j,T,z) = \psi_{\boldsymbol{n}'}(j,T,z), \qquad \boldsymbol{n} = (n_1,n_2,n_3,n_4), \quad \boldsymbol{n}' = (n_1+s_1b, n_2+s_2b, n_3+s_3b, n_4), \\ &\qquad\qquad s_1,\ s_2,\ s_3 = 0, \pm1, \pm2, \ldots . \end{aligned} \tag{66b}$$

The approximation to the solution $\varphi(\boldsymbol{x},t)$ for $\boldsymbol{x} \in S(t)$, $(l-1)T < t \leqslant lT$, in the periodic algorithm is obtained as an approximation to the sum (61) rather than (48), in do doing instead of (53) we obtain:

$$\varphi(n_1h, n_2h, n_3h,\ n_4\tau) \approx \varphi_{\boldsymbol{n}}^{(II)} \equiv \psi_{\boldsymbol{n}}(l-p,T,z) + \psi_{\boldsymbol{n}}(l-p+1,T,z) + \cdots + \psi_{\boldsymbol{n}}(l,T,z). \tag{67}$$

Each term $\psi_{\boldsymbol{n}}(j,T,z)$, $j = l-p, l-p+1,\ldots,l$, on the right-hand side of relation (67) solves the corresponding problem (66).

**Proposition 1.** *The uniform grid convergence result guaranteed by Theorem* 2 *for the basic non-deteriorating algorithm of Section* 3.1 *will hold for the periodic version of the algorithm as well, once we replace* $\varphi_{\boldsymbol{n}}^{(I)}$ *of* (53) *by* $\varphi_{\boldsymbol{n}}^{(II)}$ *of* (67) *in inequality* (56).

**Proof.** Proposition 1 is, in fact, clear because we have chosen the period $z$ according to (65) so that the periodic and non-periodic solutions coincide on $S(t)$, $(j-1)T < t < (j-1)T + T_{\text{interval}}$ for all $j = l-p, l-p+1,\ldots,l$. Therefore, estimate (56) for $\varphi_{\boldsymbol{n}}^{(II)}$ of (67) can be obtained by exactly repeating all steps of the proof of Theorem 2. □

As we did previously for the first algorithm, let us now consider the transition from $l$ to $l+1$ in formula (47) in the framework of the periodic algorithm. Assume we are interested in calculating the overall temporal evolution of the solution from $t = 0$ till an arbitrarily large $t = T_{\text{final}}$. Over this period of time, the domain $S(t)$ that was centered at $\boldsymbol{x}_0(0) = (x_1^0(0), x_2^0(0), x_3^0(0))$ at the moment $t = 0$ can travel arbitrarily far in space from its initial location, in fact, as far as $kT_{\text{final}}$:

$$\boldsymbol{x}_0(t) = \left(x_1^0(t), x_2^0(t), x_3^0(t)\right) = \boldsymbol{x}_0(0) + \int_0^t \boldsymbol{u}_0(\theta)\,\mathrm{d}\theta, \tag{68a}$$

$$\left|\boldsymbol{x}_0(t)\right| \leqslant kT_{\text{final}}, \quad 0 \leqslant t \leqslant T_{\text{final}}. \tag{68b}$$

In the $z$-periodic setting, all functions are defined for $|x_i| \leqslant z/2,\ i = 1, 2, 3$, and the edges $x_i = \pm z/2,\ i = 1, 2, 3$, are identified with one another. Accordingly, instead of the motion described by relation (68a), we consider the motion of $S(t)$ on a three-dimensional toroidal surface. Instead of (68a) we will then have

$$\boldsymbol{x}_0(t) = \left(x_1^0(t), x_2^0(t), x_3^0(t)\right) = \left\{\frac{1}{z}\left(\boldsymbol{x}_0(0) + \int_0^t \boldsymbol{u_0}(\boldsymbol{\theta})\,\mathrm{d}\theta\right)\right\} z - \frac{z}{2}, \tag{69a}$$

where $\{\cdot\}$ in (69a) denotes the fractional part. Also, conforming to the periodicity conditions, inequality (68b) transforms into

$$\left|\boldsymbol{x}_0(t)\right| \leqslant \frac{z}{2}, \quad 0 \leqslant t \leqslant T_{\text{final}}. \tag{69b}$$

The computational procedure does not change much. We calculate separately every term on the right-hand side of (67). When we go from $l$ to $l+1$, we stop calculating $\psi_{\boldsymbol{n}}(l-p, T, z)$ and add the new term $\psi_{\boldsymbol{n}}(l+1, T, z)$. This allows us to run the computation arbitrarily long with no error accumulation and no growth of computer expenses per time step. In so doing, of course, the center $\boldsymbol{x}_0(t)$ of $S(t)$, as well as the entire domain of interest $S(t)$ itself, can be located anywhere within the period, i.e., in the cube $\{|x_i| \leqslant z/2,\ i = 1, 2, 3\}$, and not necessarily close to its middle. This, however, does not affect the solution calculated inside $S(t)$ because according to condition (65), no waves from outside can enter the domain $S(t)$ before the waves generated inside $S(t)$ by the sources that operate during the interval $2T$ (say, $j = l - p$) leave it. Then, as soon as these waves have left, this entire portion of the solution is taken out by dropping the term $\psi_{\boldsymbol{n}}(l-p, T, z)$ in the sum (67). In other words, both the waves generated inside $S(t)$ that have already left it and the waves generated outside $S(t)$ by the other sources of the periodic structure that operate during the same time interval are eliminated. As has been pointed out, this does not change anything inside $S(t)$ as the waves have already left, but it prevents the waves generated outside from propagating further in.

The periodic algorithm of this section appears somewhat more efficient than the basic algorithm of Section 3.1 because the actual domain size and consequently, the number of grid nodes involved in the computation, is smaller for the periodic algorithm; this follows from the comparison of (65) against (60), $k < c$. The aforementioned better efficiency implies, of course, the better actual constants, whereas the asymptotic order with respect to $h$ for both the operations count and amount of memory required by the periodic algorithm remains at the same unimprovable level of $\mathrm{O}(h^{-3})$, which is also pertinent to the basic algorithm of Section 3.1. The number of terms $p+1$ in formulae (67), similarly to (53), does not depend on the grid size $h$ as long as $T$ remains fixed.

### *3.3. Modification based on the continuous time marching and discrete Fourier transform*

The third version of the algorithm uses the exact same representation of the solution as the previously described periodic algorithm does, see formula (61). However, the computations in the third version are organized in a substantially different way. Here, instead of calculating separately each term on the right-hand side of (67) we rather use a "one-sweep" time-marching approach and when it comes to the transition from $l$ to $l+1$ in (47), the term $\psi_{\boldsymbol{n}}(l-p, T, z)$ on the right-hand side of (67) is taken out by the explicit subtraction.

Introduce a new grid function

$$\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)=\sum_{j=l-p}^{\infty}\psi_{\boldsymbol{n}}(j,\,T,\,z), \tag{70}$$

where $\psi_{\boldsymbol{n}}(j,\,T,\,z)$ is the solution of problem (66) for a particular $j$. Here $p$ is the same integer number as in formula (61). For small $l$, the summation in (70) may need to start from $j=0$ rather than $j=l-p<0$. Notice that $\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)$ is a function of the argument $\boldsymbol{n}$, but it also depends on the discrete parameters $l$ and $p$. This means that for a given grid node $\boldsymbol{n}=(n_1,n_2,n_3,n_4)$ we may basically consider different values of $\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)$ that correspond to different $l$ (not necessarily such that $(l-1)T<n_4\tau\leqslant lT$). From the formulation of problem (66) for every $j$, it is clear that for any fixed $\boldsymbol{n}=(n_1,n_2,n_3,n_4)$ no more than several first terms of the series (70) may differ from zero.

In the particular case when $(l-1)T<n_4\tau\leqslant lT$ there is no more than $p+1$ non-zero terms in the series (70). In this case, the sum (70) coincides, in fact, with the approximation of the solution $\varphi(\boldsymbol{x},t)$ given by formula (67) for such $\boldsymbol{n}$ that satisfy $(n_1h,n_2h,n_3h)\in S(n_4\tau)$, $(l-1)T<n_4\tau\leqslant lT$ (see (51)). Thus, for this specific combination of $\boldsymbol{n}$, $l$, and $p$, the computation of $\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)$ by formula (70) can be interpreted as the computation of the approximate solution by formula (67). In this section, the approximate solution to problem (1), (2) will therefore be calculated according to formula (70).

Substituting expression (70) into the left-hand side of the finite-difference equation (9), we obtain

$$\sum_{\boldsymbol{n}\in N_{\boldsymbol{m}}}a_{\boldsymbol{mn}}\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)=f_{\boldsymbol{m}}(l-p,\,T,\,z)+f_{\boldsymbol{m}}(l-p+1,\,T,\,z)+\cdots+f_{\boldsymbol{m}}(l,\,T,\,z)+\cdots. \tag{71}$$

According to the definition of $\Theta(t)$, see (29), and formula (31), as well as the definition of $\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)$, see (70), and definitions of $\psi_{\boldsymbol{n}}(j,\,T,\,z)$ and $f_{\boldsymbol{m}}(j,\,T,\,z)$, see (66), the following equality holds:

$$f_{\boldsymbol{m}}\equiv f(m_1h,m_2h,m_3h,m_4\tau)=\sum_{j=l-p}^{\infty}f_{\boldsymbol{m}}(j,\,T,\,z) \tag{72}$$

for all those $\boldsymbol{m}$, for which

$$m_4\tau\geqslant(l-p)T. \tag{73}$$

(Again, for small $l$ we may need to replace $l-p$ by 0 on the right-hand side of both (72) and (73).) Eqs. (71) and (72) together imply that the function $\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)$ satisfies the finite-difference equation

$$\sum_{\boldsymbol{n}\in N_{\boldsymbol{m}}}a_{\boldsymbol{mn}}\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)=f_{\boldsymbol{m}}(l,\,p,\,T,\,z) \tag{74}$$

for those $\boldsymbol{m}$, for which inequality (73) holds. Besides, according to (11) we consider

$$\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)=0\quad\text{for } n_4\leqslant J_2-1. \tag{75}$$

Eqs. (75) will provide for the initial conditions when calculating $\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)$ consecutively starting with the small $l$: $l=1,2,\ldots$.

Let us now recall that the stencil $N_{\boldsymbol{m}}$, see (10), of the finite-difference equation (9) contains $J+1$ time levels ($J=2$ for the second order scheme (12), (13), (14), (16)). The values of the solution on the uppermost time level of the stencil are calculated with the help of the explicit formula (9) using the values on the $J$ preceding levels that have been obtained previously. We start the computation of $\Psi_{\boldsymbol{n}}(l,\,p,\,T,\,z)$ for $(l-1)T<n_4\tau\leqslant lT$ with the homogeneous initial data (75) and using the explicit scheme (74)

in the periodic setting of Section 2.3 (see also Section 3.2) advance one time step after another for $n_4 = J_2, J_2+1, \ldots$. This way we first calculate $\Psi_{\boldsymbol{n}}(l, p, T, z)$ for $l = 1$ while $0 < n_4\tau \leqslant T$, then $\Psi_{\boldsymbol{n}}(l, p, T, z)$ for $l = 2$ while $T < n_4\tau \leqslant 2T$, etc. *We emphasize that these computations are performed by the conventional explicit marching in time* (74) *with the periodicity boundary conditions in space.*

When we reach sufficiently far in time, namely, when $l - p$ becomes positive ($l - p > 0$), the standard marching algorithm for calculating $\Psi_{\boldsymbol{n}}(l, p, T, z)$ will require a special augmentation. This augmentation has to account for the fact that summation in formula (70) for large $l$ starts with $l - p$ rather than 0. Indeed, given a particular $l$ and $(l-1)T < n_4\tau \leqslant lT$, the sum (70) contains $p+1$ terms for $j = l-p, l-p+1, \ldots, l$. When $l$ increases by one, $l \mapsto l+1$, it will contain $p+1$ terms for $j = l-p+1, l-p+2, \ldots, l+1$. Therefore, $\forall l$ when switching from $\Psi_{\boldsymbol{n}}(l, p, T, z)$, which is used for $(l-1)T < n_4\tau \leqslant lT$, to $\Psi_{\boldsymbol{n}}(l+1, p, T, z)$, which is used for $lT < n_4\tau \leqslant (l+1)T$, the continuing straightforward time-marching should be augmented by dropping the first non-zero term $\psi_{\boldsymbol{n}}(l-p, T, z)$ from the sum (70).

Assume that for some sufficiently large $l$ we already know the approximate solution, i.e., the values of $\Psi_{\boldsymbol{n}}(l, p, T, z)$, on all time levels

$$t = n_4\tau: \quad n_4 = \frac{(l-1)T}{\tau} + 1,\ \frac{(l-1)T}{\tau} + 2, \ldots, \frac{lT}{\tau}. \tag{76}$$

We formally switch from $l$ to $l+1$ when for the uppermost time level of the stencil $N_{\boldsymbol{m}}$ we have $n_4 = lT/\tau + 1$. The switching consists of subtracting the term $\psi_{\boldsymbol{n}}(l-p, T, z)$ from the sum and thus completely disregarding it for the future computation. (Of course, when subtracting $\psi_{\boldsymbol{n}}(l-p, T, z)$, we take into account the structure of the stencil (10) of finite-difference scheme (9) or (74), see below.) Then, we continue marching in time and calculate the approximate solution $\Psi_{\boldsymbol{n}}(l+1, p, T, z)$ on the subsequent time levels

$$t = n_4\tau: \quad n_4 = \frac{lT}{\tau} + 1,\ \frac{lT}{\tau} + 2, \ldots, \frac{(l+1)T}{\tau}. \tag{77}$$

Having completed the time-marching computation of $\Psi_{\boldsymbol{n}}(l+1, p, T, z)$ for the levels (77), we again arrive at a situation when another term, this time $\psi_{\boldsymbol{n}}(l-p+1, T, z)$, needs to be taken out from the sum. *Thus, the procedure cyclically repeats itself and can be continued for as long as required.*

The aforementioned subtraction of $\psi_{\boldsymbol{n}}(l-p, T, z)$ is done according to the definition of $\Psi_{\boldsymbol{n}}(l, p, T, z)$, see (70), that implies

$$\Psi_{\boldsymbol{n}}(l+1, p, T, z) = \Psi_{\boldsymbol{n}}(l, p, T, z) - \psi_{\boldsymbol{n}}(l-p, T, z). \tag{78}$$

Equality (78) holds for all $\boldsymbol{n}$; we will use it for the last $J$ time levels of the grid (76):

$$t = n_4\tau: \quad n_4 = \frac{lT}{\tau} - J + 1,\ \frac{lT}{\tau} - J + 2, \ldots, \frac{lT}{\tau}. \tag{79}$$

As the first term $\Psi_{\boldsymbol{n}}(l, p, T, z)$ on the right-hand side of formula (78) is assumed known on the grid (76), it is also known on its sub-grid (79). *The second term $\psi_{\boldsymbol{n}}(l-p, T, z)$ on the right-hand side of formula* (78) *needs to be subtracted from $\Psi_{\boldsymbol{n}}(l, p, T, z)$ on the $J$ time levels given by* (79); *then, because the stencil $N_{\boldsymbol{m}}$ of* (10) *has $J+1$ levels altogether, starting from $n_4 = lT/\tau + 1$ and for all $n_4 > lT/\tau + 1$ the contribution of $\psi_{\boldsymbol{n}}(l-p, T, z)$ will no longer be present in the solution.* The term $\psi_{\boldsymbol{n}}(l-p, T, z)$ in (78) is the solution of problem (66) for $j = l-p$. The right-hand side $f_{\boldsymbol{m}}(l-p, T, z)$ of the latter problem may differ from zero only for those $\boldsymbol{m} = (m_1, m_2, m_3, m_4)$ that satisfy

$$(l-p-1)T < m_4\tau < (l-p+1)T, \tag{80}$$

and the initial data for calculating $\psi_{\boldsymbol{n}}(l-p,T,z)$ are homogeneous: $\psi_{\boldsymbol{n}}(j-p,T,z)=0$ for $n_4=(j-p-1)T/\tau-J_1\leqslant n_4\leqslant(j-p-1)T/\tau+J_2-1$, see (66). Let us also note that the parameters of the problem are obviously chosen so that by the time we subtract $\psi_{\boldsymbol{n}}(l-p,T,z)$ it is already a solution to the homogeneous equation or in other words, the interval of time (80), during which the right-hand side $f_{\boldsymbol{m}}(l-p,T,z)$ that drives $\psi_{\boldsymbol{n}}(l-p,T,z)$ may differ from zero, is located sufficiently far behind on the time axis with respect to the moment of subtraction $lT/\tau$.

Clearly, by the time we need to implement the subtraction (78), the contribution $\psi_{\boldsymbol{n}}(l-p,T,z)$ will have already been calculated once, however not separately but rather as a part of $\Psi_{\boldsymbol{n}}(l,p,T,z)$, see (70). To actually perform the subtraction (78) on the grid (79), we need to compute $\psi_{\boldsymbol{n}}(l-p,T,z)$ separately once more, and this computation will be split into two stages. First, we will calculate the solution $\psi_{\boldsymbol{n}}(l-p,T,z)$ with the explicit finite-difference scheme (66a) step after step in time for the levels

$$t=n_4\tau\colon\quad n_4=\frac{(l-p-1)T}{\tau}+J_2,\ldots,\frac{(l-p+1)T}{\tau}+J_2. \tag{81}$$

Clearly, the uppermost time level of the stencil $N_{\boldsymbol{m}}$ of (10) will reach the last time level given in (81), i.e., $n_4=(l-p+1)T/\tau+J_2$, at exactly the same moment of time when the center of the stencil will be at $m_4=(l-p+1)T/\tau$ and therefore, according to (80) at this very moment the finite-difference equation (66a) for $\psi_{\boldsymbol{n}}(l-p,T,z)$ becomes homogeneous. The computation of $\psi_{\boldsymbol{n}}(l-p,T,z)$ on the levels (81) takes

$$\nu\frac{2T}{\tau}\left(\frac{z}{h}\right)^3 \tag{82}$$

arithmetic operations, where $\nu$ is the number of operations required for advancing the solution one step in time in one grid node. (In other words, $\nu$ is the number of arithmetic operations "on the stencil" of the scheme.)

The values of the solution $\psi_{\boldsymbol{n}}(l-p,T,z)$ on the last $J$ levels of (81), i.e., $n_4=(l-p+1)T/\tau-J_1+1,\ldots,(l-p+1)T/\tau+J_2$, will be used as the initial data for calculating this solution $\psi_{\boldsymbol{n}}(l-p,T,z)$ on the time levels (79), on which it needs to be subtracted. As the right-hand side $f_{\boldsymbol{m}}(l-p,T,z)$ is equal to zero, $f_{\boldsymbol{m}}(l-p,T,z)=0$, for $m_4\geqslant(l-p+1)T/\tau$, see (80), an efficient way to calculate the solution $\psi_{\boldsymbol{n}}(l-p,T,z)$ for the grid (79) will be through representing it in the form of a discrete Fourier series while the initial data for $\psi_{\boldsymbol{n}}(l-p,T,z)$ on the last $J$ levels of (81) are known. The finite Fourier expansion is built with respect to the following system of grid basis functions $\boldsymbol{e}^{\overline{k}}_{\overline{n}}$:

$$\boldsymbol{e}^{\overline{k}}_{\overline{n}}=\exp\bigl(\mathrm{i}\langle\overline{n},\overline{k}\rangle\bigr),\qquad \mathrm{i}=\sqrt{-1},\quad \overline{n}=(n_1,n_2,n_3),\quad \overline{k}=(k_1,k_2,k_3),$$
$$k_1,k_2,k_3=0,1,2,\ldots,\frac{z}{h}-1,\quad \bigl\langle\overline{n},\overline{k}\bigr\rangle=n_1k_1+n_2k_2+n_3k_3.$$

Hereafter we assume that $z/h$ is a power of 2 so that the fast Fourier transform (FFT) can be used for for calculating the coefficients of the discrete Fourier series of a given grid function, as well as for restoring the node values of the grid function from its Fourier representation. Each of the foregoing transformations obviously requires

$$\mathrm{O}\left(\left(\frac{z}{h}\right)^3\ln\frac{z}{h}\right) \tag{83}$$

arithmetic operations.

Having introduced the discrete Fourier transform on the grid, we organize the second stage of the computation of $\psi_{\boldsymbol{n}}(l-p,T,z)$ on the time levels (79) as follows. First, we Fourier transform the data on the last $J$ levels of grid (81), which takes $J$ times the number of operation (83). Then, we advance each Fourier component independently to the levels $n_4 = lT/\tau - J,\dots,lT/\tau$, see (79), using the explicit formulae that are easily obtained from the Fourier representation of the finite-difference operator of (74); essentially, this reduces to multiplication by the appropriate powers of the corresponding amplification factors and obviously takes $\mathrm{O}((z/h)^3)$ operations. Finally, we need to restore the node values of $\psi_{\boldsymbol{n}}(l-p,T,z)$ on the grid (79) using the inverse FFT, which again takes $J$ times the number of operations (83). The overall computational cost of this second stage will then be

$$\mathrm{O}\left(J\left(\frac{z}{h}\right)^3\ln\frac{z}{h}\right)+\mathrm{O}\left(\left(\frac{z}{h}\right)^3\right)+\mathrm{O}\left(J\left(\frac{z}{h}\right)^3\ln\frac{z}{h}\right)=\mathrm{O}\left(\left(\frac{z}{h}\right)^3\ln\frac{z}{h}\right) \tag{84}$$

arithmetic operations because $J$ is fixed and does not depend on the grid size. Consequently, the total operations count for calculating $\psi_{\boldsymbol{n}}(l-s,T,z)$ on the grid (79) for subtraction, i.e., calculating the second term in the recurrence formula (78), consists of the expenses for the first (82) and second (84) stages and adds up to

$$\nu\frac{2T}{\tau}\left(\frac{z}{h}\right)^3+\mathrm{O}\left(\left(\frac{z}{h}\right)^3\ln\frac{z}{h}\right) \tag{85}$$

arithmetic operations. Recurrence formula (78) is used for subtracting the contribution of the entire "chunk" of $2T/\tau$ time levels (80). Therefore, if one recalculates the associated expense (85) proportionally per time step, it obviously reduces to

$$\nu\left(\frac{z}{h}\right)^3+\mathrm{O}\left(\frac{\tau}{2T}\left(\frac{z}{h}\right)^3\ln\frac{z}{h}\right)=\mathrm{O}\left(\left(\frac{z}{h}\right)^3\right). \tag{86}$$

The calculation of the first term in the recurrence formula (78) also requires $\mathrm{O}((z/h)^3)$ arithmetic operations per time step as this is done simply using the explicit scheme (74).

Summarizing, we conclude that the overall computational cost of the third version of the algorithm is $\mathrm{O}((z/h)^3)$ arithmetic operations per one time step. It is also easy to see that the required amount of memory (i.e., number of words) is of the order $\mathrm{O}((z/h)^3)$ as well.

**Proposition 2.** *The uniform grid convergence result guaranteed by Theorem* 2 *for the basic non-deteriorating algorithm of Section* 3.1 *will hold for the modified version described in this section as well.*

**Proof.** As the non-deteriorating algorithm of the current section essentially reproduces the calculations by formula (67) for the periodized version of Section 3.2 with the difference only in the computational procedure, Proposition 2 immediately follows from Proposition 1, see Section 3.2. □

Moreover, we also note that the interpretation of spatial periodicity given by formulae (69), and subsequent comments for the periodic algorithm of Section 3.2, apply to the current version with no changes.

*We emphasize that the variant of the non-deteriorating algorithm presented in this section is designed so that to still guarantee the grid convergence uniformly in time, see Theorem* 2, *but on the other hand*

*maximally resemble the standard time-marching computational procedure pertinent to an explicit scheme with the periodicity boundary conditions in space. Indeed, the only difference between the procedure that we have just described and typical unaltered time marching is that here we need to cyclically perform the subtraction* (78); *in so doing, the complexity of the modified procedure with respect to the grid dimension remains asymptotically the same as for the standard marching algorithm, see* (86).

## 4. Numerical demonstrations

To actually demonstrate that the lacunae-based algorithm is an appropriate procedure that does deliver according to its theoretical design properties, we present some numerical results for the wave equation. For our simulations, we assume axial symmetry and employ the $(r, z)$ cylindrical coordinates so that to account for the three-dimensional effects using two-dimensional geometry. Accordingly, Eq. (1) becomes:

$$\frac{\partial^2\varphi}{\partial t^2} - c^2\left(\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\varphi}{\partial r}\right) + \frac{\partial^2\varphi}{\partial z^2}\right) = f(r, z, t), \quad t \geqslant 0. \tag{87}$$

The solution $\varphi$ of Eq. (87), as well as the source function $f$, are functions of $r$, $z$, and $t$. The initial conditions for Eq. (87) remain homogeneous as before, see (2).

We introduce the rectangular auxiliary domain $[0, R] \times [-Z/2, Z/2]$ of variables $(r, z)$. The boundary conditions are periodic with the period $Z$ in the $z$ direction, and zero Dirichlet at $r = R$:

$$\varphi(r, z \pm Z, t) = \varphi(r, z, t), \qquad \varphi(R, z, t) = 0. \tag{88}$$

Note, in Sections 2.3 and 3.2 we have used lower-case "$z$" to denote the period. In the current section we use capital "$Z$" for this purpose so that not to confuse it with the conventional notation "$z$" for the axial coordinate.

The mathematical formulation of the problem obviously requires no boundary conditions at $r = 0$. However, for the purpose of subsequently building a discrete scheme (see below) we notice that the natural assumption of $\varphi(r, z, t)$ being a bounded smooth function, along with the axial symmetry, immediately imply that $(\partial\varphi/\partial r)|_{r=0} = 0$. Consequently, the Taylor expansion for $\varphi$ near $r = 0$ yields:

$$\varphi(r, \cdot) = \varphi(0, \cdot) + \frac{1}{2}\left.\frac{\partial^2\varphi}{\partial r^2}\right|_{r=0} \cdot r^2 + \mathrm{O}(r^3),$$

which means that

$$\frac{\partial\varphi}{\partial r} = \left.\frac{\partial^2\varphi}{\partial r^2}\right|_{r=0} \cdot r + \mathrm{O}(r^2).$$

Substituting the latter expression into (87) and considering the limit $r \to 0$, we obtain that on the $z$-axis, i.e., at $r = 0$, Eq. (87) reduces to:

$$\frac{\partial^2\varphi}{\partial t^2} - c^2\left(2\frac{\partial^2\varphi}{\partial r^2} + \frac{\partial^2\varphi}{\partial z^2}\right) = f(r, z, t), \quad r = 0,\ t \geqslant 0. \tag{89}$$

To assess the quality of our numerical method we need to build a reference exact solution of problem (87), (2). This solution is obtained using the Lorentz' transform:

$$\begin{aligned}\theta &= \frac{1}{\sqrt{1-k^2/c^2}}\cdot t - \frac{k/c}{\sqrt{1-k^2/c^2}}\cdot\frac{z}{c},\\ \zeta &= -\frac{k/c}{\sqrt{1-k^2/c^2}}\cdot ct + \frac{1}{\sqrt{1-k^2/c^2}}\cdot z.\end{aligned} \tag{90}$$

Transformation (90) introduces the new coordinate system $(r,\zeta,\theta)$. The origin of this new coordinate system moves with the speed $k$ along the $z$-axis of the original coordinate system. In other words, at every given $t$ it is positioned at $z=kt$ in the original frame of reference. In implementing transformation (90) we will always need to assume that $k<c$, as has also been suggested in Section 1.1, Eq. (4).

The key property of the Lorentz' transform (90) is that it does not change the form of the wave equation (1) (and consequently, (87) and (89)), see, e.g., [1]. As such, let us introduce an arbitrary function of time $\chi=\chi(t)$, $\chi(t)=0$ for $t\leqslant 0$, so that it also be smooth on the entire $\mathbb{R}$. If we also define $\rho^2=r^2+\zeta^2$, then

$$\psi(r,\zeta,\theta)=\frac{\chi(\theta-\rho/c)}{\rho} \tag{91a}$$

becomes a solution to the wave equation in the new coordinates $(r,\zeta,\theta)$. Solution (91a) is driven by a point $\delta$-type source, which is located at the origin $\{r=0,\ \zeta=0\}$ and modulated in time by the function $\chi(\theta)$. As $\chi'(0)=0$, this solution also satisfies the homogeneous initial conditions. Consequently, the function

$$\psi(r,z,t)=\frac{\chi(\theta(z,t)-\rho(r,z,t)/c)}{\rho(r,z,t)} \tag{91b}$$

obtained by substituting (90) into (91a) is a solution of Eq. (87) with the right-hand side $f(r,z,t)=\chi(t)\cdot\delta(r,z-kt)$. In other words, $\psi(r,z,t)$ of (91b) is a solution to the wave equation excited by a $\delta$-source that moves straightforwardly and uniformly and is modulated in time by a given smooth function $\chi$. Solution (91b) also satisfies homogeneous initial conditions (2). From the standpoint of physics, solution (91b) can be characterized as radiation of spherical waves by a moving point source.

Solution (91b) is obviously singular. To use it for testing the numerical algorithm we need to remove the singularity. For that, we first introduce the actual domain $S(t)$ as a ball of diameter $d$ with its center at the origin of the new coordinate system: $\{r=0,\ z=kt\}$. As such, this spherical domain moves uniformly along the $z$-axis, which obviously helps us keep the axial symmetry intact. Let us also define $\widetilde{r}^2=r^2+(z-kt)^2$ and introduce the function $Q=Q(\widetilde{r})$, $r\geqslant 0$, such that $Q(0)=0$, $Q(\widetilde{r})\equiv 1$ for $\widetilde{r}\geqslant\kappa d/2$, where $\kappa<1$, and also

$$\frac{\mathrm{d}^m Q(0)}{\mathrm{d}t^m}=\frac{\mathrm{d}^m Q(\kappa d/2)}{\mathrm{d}t^m}=0\quad\text{for } m=1,2,\ldots$$

till at least $m=4$. Then, it is easy to see that the function $\varphi(r,z,t)=\psi(r,z,t)\cdot Q(\widetilde{r})$ is regular everywhere. Moreover, it is easy to verify by direct differentiation that the function $f(r,z,t)\stackrel{\text{def}}{=}\Box\varphi(r,z,t)$, where $\Box$ denotes the wave operator, i.e., the left-hand side of Eq. (87), is also regular (continuous and bounded) everywhere. We will use $f(r,z,t)$ defined this way as the source function for Eq. (87). Clearly, $f(r,z,t)$ may, generally speaking, differ from zero only on the ball of a smaller diameter $\kappa d$ concentric with $S(t)$. Everywhere else, i.e., for $\widetilde{r}>\kappa d/2$, $f(r,z,t)=0$.

Clearly, the solution of problem (87), (2) driven by this $f(r,z,t)$ is the foregoing

$$\varphi(r,z,t) = \psi(r,z,t)\cdot Q(\tilde{r}). \tag{92}$$

This function satisfies the non-homogeneous wave equation with the right-hand side $f(r,z,t)$ on a smaller ball of diameter $\kappa d$ concentric with $S(t)$. Everywhere else it is a solution to the homogeneous wave equation because it coincides with $\psi(r,z,t)$ of (91b). Consequently, the solution $\varphi(r,z,t)$ of (92) can be interpreted as the radiation of waves by a compactly supported moving source $f(r,z,t)$. Numerically, we will be reproducing solution $\varphi(r,z,t)$ given by (92) on the domain $S(t)$ using finite-difference methods.

We employ three different explicit central-difference schemes in our simulations. In all three cases we construct a uniform rectangular grid on the plane $(r,z)$: $r_i = ih_r$, $i = 0,1,\ldots,N_r$, $h_r = R/N_r$, and $z_j = jh_z$, $j = 0,\pm1,\ldots,\pm N_z$, $h_z = Z/2N_z$. The discrete time levels are $t_n = n\tau$, $n = 0,1,\ldots$. For the cell-centered second-order scheme, we keep the values of the unknown function $\varphi$ at the grid nodes in the $z$ direction and at mid-points in the $r$ direction:

$$\frac{\varphi^{n+1}_{i+1/2,j} - 2\varphi^{n}_{i+1/2,j} + \varphi^{n-1}_{i+1/2,j}}{\tau^2} - c^2\left(\frac{1}{r_{i+1/2}}\frac{1}{h_r}\left[r_{i+1}\frac{\varphi^{n}_{i+3/2,j} - \varphi^{n}_{i+1/2,j}}{h_r} - r_i\frac{\varphi^{n}_{i+1/2,j} - \varphi^{n}_{i-1/2,j}}{h_r}\right] + \frac{\varphi^{n}_{i+1/2,j+1} - 2\varphi^{n}_{i+1/2,j} + \varphi^{n}_{i+1/2,j-1}}{h_z^2}\right) = f^{n}_{i+1/2,j}. \tag{93a}$$

Eqs. (93a) hold for all $i > 0$. As in this case we do not have the unknown function defined on the axis of symmetry, and the closest values that correspond to $i = 0$ are half-grid-size away: $\varphi^{n}_{1/2,j}$, then the scheme for $i = 0$ is obtained by simply assuming that $(\varphi^{n}_{i+1/2,j} - \varphi^{n}_{i-1/2,j})/h_r|_{i=0} = 0$, which can be interpreted as a second-order approximation of the natural condition $(\partial\varphi/\partial r)|_{r=0} = 0$. This immediately yields:

$$\frac{\varphi^{n+1}_{1/2,j} - 2\varphi^{n}_{1/2,j} + \varphi^{n-1}_{1/2,j}}{\tau^2} - c^2\left(\frac{1}{r_{1/2}}\frac{1}{h_r}r_1\frac{\varphi^{n}_{3/2,j} - \varphi^{n}_{1/2,j}}{h_r} + \frac{\varphi^{n}_{1/2,j+1} - 2\varphi^{n}_{1/2,j} + \varphi^{n}_{1/2,j-1}}{h_z^2}\right) = f^{n}_{1/2,j}. \tag{93b}$$

For the node-centered second-order scheme, $\varphi$ is taken at the actual grid nodes, and for $i > 0$ we have:

$$\frac{\varphi^{n+1}_{i,j} - 2\varphi^{n}_{i,j} + \varphi^{n-1}_{i,j}}{\tau^2} - c^2\left(\frac{1}{r_i}\frac{1}{h_r}\left[r_{i+1/2}\frac{\varphi^{n}_{i+1,j} - \varphi^{n}_{i,j}}{h_r} - r_{i-1/2}\frac{\varphi^{n}_{i,j} - \varphi^{n}_{i-1,j}}{h_r}\right] + \frac{\varphi^{n}_{i,j+1} - 2\varphi^{n}_{i,j} + \varphi^{n}_{i,j-1}}{h_z^2}\right) = f^{n}_{i,j}. \tag{94a}$$

To obtain the scheme on the axis of symmetry $i = 0$ in this case, we need to approximate equation (89). For the $\partial^2/\partial r^2$ derivative in this equation we can first formally write $(\partial^2\varphi/\partial r^2)|_{r=0} \approx (\varphi_{1,j} - 2\varphi_{0,j} + \varphi_{-1,j})/h_r^2$. This expression obviously reduces to $(\partial^2\varphi/\partial r^2)|_{r=0} \approx 2(\varphi_{1,j} - \varphi_{0,j})/h_r^2$ because of the symmetry: $\varphi_{-1,j} = \varphi_{1,j}$, and we consequently obtain:

$$\frac{\varphi^{n+1}_{i,j} - 2\varphi^{n}_{i,j} + \varphi^{n-1}_{i,j}}{\tau^2} - c^2\left(4\frac{\varphi^{n}_{1,j} - \varphi^{n}_{0,j}}{h_r^2} + \frac{\varphi^{n}_{0,j+1} - 2\varphi^{n}_{0,j} + \varphi^{n}_{0,j-1}}{h_z^2}\right) = f^{n}_{0,j}. \tag{94b}$$

The last scheme is a node-centered fourth-order scheme. More precisely, it approximates spatial derivatives with the accuracy $O(h_r^4 + h_z^4)$ and temporal derivative with the accuracy $O(\tau^2)$. For $i > 1$ we have:

$$\frac{\varphi_{i,j}^{n+1}-2\varphi_{i,j}^{n}+\varphi_{i,j}^{n-1}}{\tau^2}-c^2\Bigg(\frac{4}{3}\frac{1}{r_i}\frac{1}{h_r}\left[r_{i+1/2}\frac{\varphi_{i+1,j}^{n}-\varphi_{i,j}^{n}}{h_r}-r_{i-1/2}\frac{\varphi_{i,j}^{n}-\varphi_{i-1,j}^{n}}{h_r}\right]$$
$$-\frac{1}{3}\frac{1}{r_i}\frac{1}{2h_r}\left[r_{i+1}\frac{\varphi_{i+2,j}^{n}-\varphi_{i,j}^{n}}{2h_r}-r_{i-1}\frac{\varphi_{i,j}^{n}-\varphi_{i-2,j}^{n}}{2h_r}\right]$$
$$+\frac{-\varphi_{i,j+2}^{n}+16\varphi_{i,j+1}^{n}-30\varphi_{i,j}^{n}+16\varphi_{i,j-1}^{n}-\varphi_{i,j-2}^{n}}{12h_z^2}\Bigg)=f_{i,j}^{n}. \tag{95a}$$

For $i=1$ we have $r_{i-1}=r_0=0$ and consequently:

$$\frac{\varphi_{1,j}^{n+1}-2\varphi_{1,j}^{n}+\varphi_{1,j}^{n-1}}{\tau^2}-c^2\Bigg(\frac{4}{3}\frac{1}{r_1}\frac{1}{h_r}\left[r_{3/2}\frac{\varphi_{2,j}^{n}-\varphi_{1,j}^{n}}{h_r}-r_{1/2}\frac{\varphi_{1,j}^{n}-\varphi_{0,j}^{n}}{h_r}\right]-\frac{1}{3}\frac{1}{r_1}\frac{1}{2h_r}\left[r_2\frac{\varphi_{3,j}^{n}-\varphi_{1,j}^{n}}{2h_r}\right]$$
$$+\frac{-\varphi_{1,j+2}^{n}+16\varphi_{1,j+1}^{n}-30\varphi_{1,j}^{n}+16\varphi_{1,j-1}^{n}-\varphi_{1,j-2}^{n}}{12h_z^2}\Bigg)=f_{1,j}^{n}. \tag{95b}$$

Finally, for $i=0$ we again have to approximate Eq. (89). Using symmetry like in the previous case, we arrive at:

$$\frac{\varphi_{0,j}^{n+1}-2\varphi_{0,j}^{n}+\varphi_{0,j}^{n-1}}{\tau^2}$$
$$-c^2\Bigg(2\frac{-2\varphi_{2,j}^{n}+32\varphi_{1,j}^{n}-30\varphi_{0,j}^{n}}{12h_r^2}+\frac{-\varphi_{0,j+2}^{n}+16\varphi_{0,j+1}^{n}-30\varphi_{0,j}^{n}+16\varphi_{0,j-1}^{n}-\varphi_{0,j-2}^{n}}{12h_z^2}\Bigg)=f_{0,j}^{n}. \tag{95c}$$

For all three schemes, (93), (94), and (95), setting the discrete boundary conditions (88) on the outer boundary of the auxiliary domain $[0,R]\times[-Z/2,Z/2]$ is straightforward. An extra boundary condition is needed for the fourth-order approximation, and we simply set $\varphi_{N_r-1,j}^{n}=0$ in addition to $\varphi_{N_r,j}^{n}=0$. Regarding the time step $\tau$, all three schemes are explicit and as such, there is a Courant-type stability constraint.

As has been mentioned, the purpose of presenting the results of numerical computations that follow is to corroborate the theoretical design properties of the lacunae-based algorithm, i.e., to show the temporally uniform grid convergence on long time intervals. For that, we conduct a grid refinement study, i.e., approximate the exact solution (92) on a sequence of successively more fine grids. In so doing, the time step $\tau$ for the two second-order schemes (93) and (94) is always reduced with the same rate as the corresponding spatial sizes $h_r$ and $h_z$; and the time step $\tau$ for the fourth-order scheme (95) is reduced twice as fast (i.e., by a factor of four every time $h_r$ and $h_z$ are reduced by a factor of two) so that to demonstrate the fourth-order overall convergence in the end. The computations in each case were run till the dimensionless time $t=200\cdot d/c$, i.e., for 200 times the time interval required for a wave to cross the domain $S(t)$. This certainly qualifies as "long term" from the standpoint of any conceivable application.

The actual parameters that we have used for our simulations are the following: $R=Z=\pi$, $d=1.8$, $c=1$, $k=0.1$, $\kappa=0.8$. The spatial grid is composed of square cells: $N_r=N_z$ and consequently, $h_r=h_z=h$. The actual grid dimensions $N_r\times 2N_z$ are: $64\times 128$, $128\times 256$, and $256\times 512$. The temporal partition size $2T$, see (32) and (33), is found from (64) once the other parameters have

been set. The function $\Theta(t)$, as well as $Q(\tilde{r})$, on the intervals of their variation from 0 to 1 are built as polynomials of degree 9 (with only odd powers included), as suggested in the beginning of Section 2.2. This guarantees four continuous derivatives in transition to the constant (either 0 or 1). The function $\chi(t)$ is defined as follows: $\chi(t) = (1 + \frac{1}{4}\sin t)R(1 - t/(2\pi))$, where $R(t)$ is, again, a polynomial of degree 9 that decays smoothly from 1 to 0 on the interval $[1, 0]$ and thus provides four continuous derivatives, see Section 2.2. Finally, the actual algorithm used for computations is the version described in Section 3.3. In the radial direction $r$, instead of the conventional Fourier transform we use expansion with respect to the corresponding discrete eigenfunctions calculated numerically.

In Fig. 3 we show error profiles (more precisely, the logarithm of the relative error on the domain $S(t)$ in the maximum norm as it depends on the dimensionless time) on all three grids for both second-order schemes (93) and (94). In Fig. 4, similar curves are shown for the fourth-order scheme (95). From these figures we conclude that indeed no error is accumulated in the course of computations because all error profiles are flat throughout the entire $200 \cdot d/c$ time interval. Thus, the solution does not deteriorate as time elapses. Fig. 3 also shows that every time the grid is refined by a factor of two the error drops by approximately a factor of four, which indicates the second-order convergence. Similarly, Fig. 4 shows that every time the grid is refined by a factor of two the error drops by approximately a factor of sixteen, which is an indication of the fourth-order convergence. Consequently, we can conclude that numerical experiments fully corroborate the theoretical design properties of the algorithm.

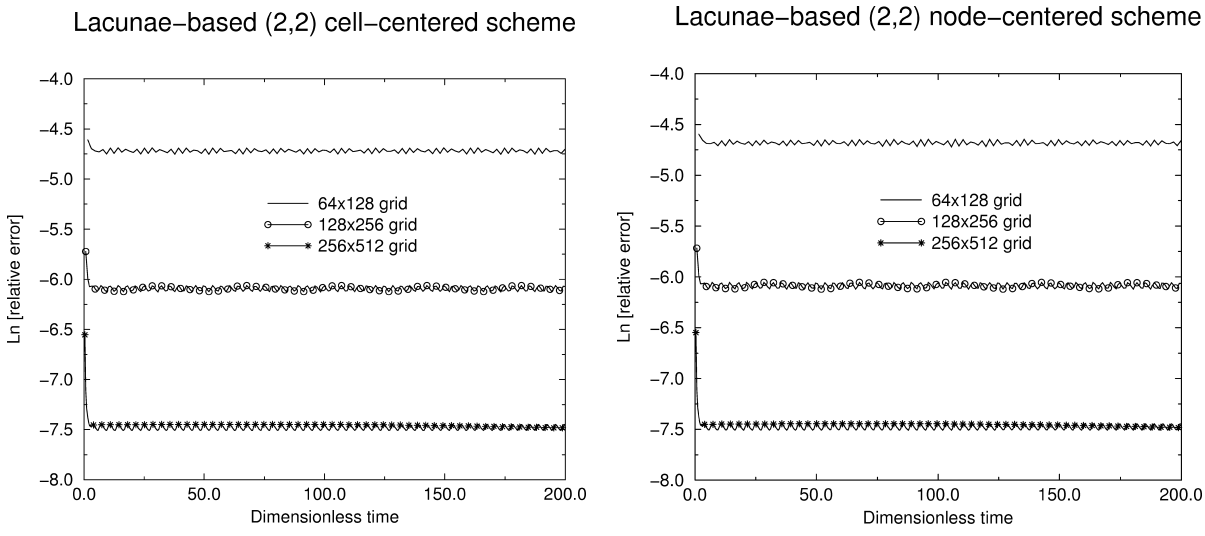


Fig. 3. Grid convergence study for the long-term integration of the wave equation. (a) The second-order scheme (93). (b) The second-order scheme (94).

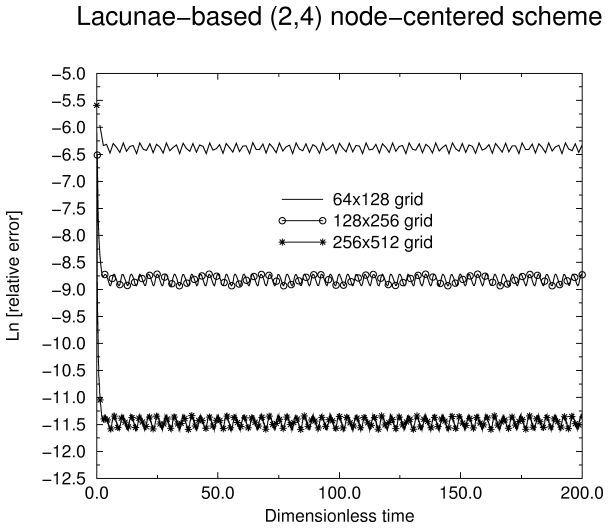


Fig. 4. Same as Fig. 3 for the fourth-order scheme (95).

## 5. Possible generalizations

First, let us note that the assumption of homogeneity of the initial data (2) can be alleviated and replaced by a weaker requirement

$$\varphi(\boldsymbol{x},t)|_{t=0}=\varphi_0(\boldsymbol{x}),\qquad \left.\frac{\partial\varphi(\boldsymbol{x},t)}{\partial t}\right|_{t=0}=\varphi_1(\boldsymbol{x}),$$

where $\varphi_0(\boldsymbol{x})$ and $\varphi_1(\boldsymbol{x})$ are sufficiently smooth functions that turn into zero outside the domain $S(t)|_{t=0}=S(0)$.

Further, as has been mentioned the requirement of smoothness for $f(\boldsymbol{x},t)$ throughout the entire space-time $(\boldsymbol{x},t)$ along with considering $f(\boldsymbol{x},t)\neq 0$ only for $t>0$ actually implies that $f(\boldsymbol{x},t)$ and its first several derivatives with respect to $t$ have to vanish as $t\to+0$. This condition can also be alleviated by requiring that the function $f(\boldsymbol{x},t)$, supp $f(\boldsymbol{x},t)\subset S(t)$, be smooth only for $t\geqslant 0$ rather than on the entire space-time $\mathbb{R}^3\times(-\infty,\infty)$. The resulting Cauchy problem, which appears somewhat more complex, can actually be reduced to problem (1), (2) if one represents the solution to the new problem as a sum of two functions:

$$\varphi_(\boldsymbol{x},t)=\overline{\varphi}(\boldsymbol{x},t)+\widetilde{\varphi}(\boldsymbol{x},t).$$

The function $\overline{\varphi}(\boldsymbol{x},t)$ will be a solution to the Cauchy problem with the given non-homogeneous initial data and the right-hand side $F(\boldsymbol{x},t)=\Theta(t)f(\boldsymbol{x},t)$ that turns into zero for $t\geqslant 1$. The function $\widetilde{\varphi}(\boldsymbol{x},t)$ will be a solution to the problem

$$\begin{aligned}&\frac{\partial^2\widetilde{\varphi}}{\partial t^2}-c^2\left(\frac{\partial^2\widetilde{\varphi}}{\partial x_1^2}+\frac{\partial^2\widetilde{\varphi}}{\partial x_2^2}+\frac{\partial^2\widetilde{\varphi}}{\partial x_3^2}\right)=f(\boldsymbol{x},t)-F(\boldsymbol{x},t),\quad t\geqslant 0,\\ &\widetilde{\varphi}|_{t=0}=\left.\frac{\partial\widetilde{\varphi}}{\partial t}\right|_{t=0}=0.\end{aligned}\tag{96}$$

Problem (96) is obviously of the type (1), (2). The problem for $\overline{\varphi}(\boldsymbol{x}, t)$ needs to be solved only till some $t = t_0$, after which $\overline{\varphi}(\boldsymbol{x}, t) \equiv 0$ when $\boldsymbol{x} \in S(t)$ because of the presence of lacunae in the solutions of the three-dimensional wave equation.

## Appendix

There are multiple problems in mechanics and physics that involve a wave propagation process of some kind as at least one key element. Therefore, an efficient numerical method for computation of waves on long time intervals with uniform error bounds, such as the one described in the paper, is important for various applications. As a specific example, let us consider the Euler equations of motion of inviscid compressible fluid written with respect to some inertial frame of reference $(\boldsymbol{x}, t)$:

$$\begin{aligned} &\frac{\partial \rho}{\partial t} + \nabla(\rho \boldsymbol{v}) = 0, \\ &\rho \frac{\partial \boldsymbol{v}}{\partial t} + \rho(\boldsymbol{v}\nabla)\boldsymbol{v} + \nabla p = \boldsymbol{0}, \\ &\rho \frac{\partial \varepsilon}{\partial t} + \rho \boldsymbol{v} \nabla \varepsilon + p \nabla \boldsymbol{v} = 0, \end{aligned} \tag{A.1}$$

where $\rho$ is the density, $p$—pressure, $\varepsilon$—internal energy, and $\boldsymbol{v}$ is the velocity vector; the system of equations (A.1) should also be supplemented by the equation of state. Let the new frame of reference $(\boldsymbol{x}', t')$ move with respect to the old one with the given velocity $\boldsymbol{u}_0 = \boldsymbol{u}_0(t)$ so that the new coordinates be connected with the old ones via the Galileo transform:

$$\boldsymbol{x}' = \boldsymbol{x} - \int_0^t \boldsymbol{u}_0(\theta)\, \mathrm{d}\theta, \qquad t' = t. \tag{A.2}$$

Note, for simplicity we assume only the translational motion of coordinate frames with respect to one another; but rotational motion can also be taken into account. Denoting the velocity vector in the new coordinates by $\boldsymbol{u}$, we obviously have $\boldsymbol{v} = \boldsymbol{u}_0 + \boldsymbol{u}$. Then, applying the transformation (A.2), we can rewrite Eqs. (A.1) in the new coordinates:

$$\begin{aligned} &\frac{\partial \rho}{\partial t'} + \nabla'(\rho \boldsymbol{u}) = 0, \\ &\rho \frac{\partial \boldsymbol{u}}{\partial t'} + \rho(\boldsymbol{u}\nabla')\boldsymbol{u} + \nabla' p = -\rho \frac{\partial \boldsymbol{u_0}}{\partial t'}, \\ &\rho \frac{\partial \varepsilon}{\partial t'} + \rho \boldsymbol{u} \nabla' \varepsilon + p \nabla' \boldsymbol{u} = 0. \end{aligned} \tag{A.3}$$

The nonhomogeneous term on the right-hand side of the momentum equation in (A.3) accounts for the acceleration of the coordinate system because the new frame of reference is, generally speaking, no longer inertial (unless $\boldsymbol{u}_0 = \text{const}$).

In practical terms, the transformation (A.2) may correspond, for example, to a maneuvering aircraft, where $\boldsymbol{u}_0(t)$ is its full velocity at a given moment $t$. Eqs. (A.3) will then correspond to the description of the flow around the aircraft in the frame of reference that is connected with this aircraft. The term $-\rho(\partial \boldsymbol{u_0}/\partial t')$ represents forces due to the translational acceleration of the coordinate frame. As has been

mentioned, Coriolis-type accelerations can also be accounted for, but this is not essential for the current general purposes.

Let us now consider stationary fluid in the original frame of reference:

$$\rho = \overline{\rho}, \qquad p = \overline{p}, \qquad \varepsilon = \overline{\varepsilon}, \qquad \boldsymbol{v} = \boldsymbol{0}. \tag{A.4}$$

In the new coordinate system the stationary solution (A.4) obviously becomes

$$\rho = \overline{\rho}, \qquad p = \overline{p}, \qquad \varepsilon = \overline{\varepsilon}, \qquad \boldsymbol{u} = -\boldsymbol{u}_0. \tag{A.5}$$

Linearizing Eqs. (A.3) against the solution (A.5), which is constant across the space for any particular time, we obtain

$$\begin{aligned} &\frac{\partial p}{\partial t'} - \boldsymbol{u}_0 \nabla' p + \nabla' \boldsymbol{u} = 0, \\ &\frac{\partial \boldsymbol{u}}{\partial t'} - (\boldsymbol{u}_0 \nabla') \boldsymbol{u} + \nabla' p = \boldsymbol{0}. \end{aligned} \tag{A.6}$$

The quantities $p$ and $\boldsymbol{u}$ in (A.6) denote the perturbations with respect to the corresponding background values given in (A.5). To actually perform the linearization, we need to assume that the fluid motion is adiabatic and that the foregoing perturbations are small; then, retaining only the first-order terms with respect to the perturbations, we arrive at Eqs. (A.6). Linear equations (A.6) are natural to use for describing the fluid flow in the far field, i.e., far away from the source of perturbations, which would be the aircraft in the example mentioned previously. We emphasize that the external acceleration term in Eqs. (A.6) has dropped, and Eqs. (A.6) actually look exactly like the equations of advective acoustics with the background velocity $\boldsymbol{u}_0$ that may depend on time. This means that *the process of propagation of small perturbations from an accelerating source in the far field is quasi-stationary,* at every given moment of time $t$ these perturbations behave like the perturbations from the source with the instantaneous speed $\boldsymbol{u}_0(t)$.

As has been mentioned, one can use linear equations (A.6) far away from the source of perturbations. Let us assume that there is a sufficiently large ball $S'$ of diameter $d$ centered at the origin $\boldsymbol{x}' = \boldsymbol{0}$ that fully contains the source of perturbations (aircraft) and such that linear homogeneous equations (A.6) hold outside $S'$. Inside $S'$ the flow needs to be described by a more complex model, e.g., the original Euler equations supplemented by the proper boundary conditions on the solid surface, or even more sophisticated. *However, for the purpose of describing the far field, this complex interior model can be simply replaced by the right-hand sides to Eqs.* (A.6) *that would be concentrated inside* $S'$. Formally, this is very easy to do. Assuming that we know the flow solution throughout the space (if there is an immersed configuration like aircraft contained in $S'$, we can use any smooth extension inside it), we simply substitute it into (A.6) and thus generate inhomogeneities inside $S'$:

$$\begin{aligned} &\frac{\partial p}{\partial t'} - \boldsymbol{u}_0 \nabla' p + \nabla' \boldsymbol{u} = g_1(\boldsymbol{x}', t'), \\ &\frac{\partial \boldsymbol{u}}{\partial t'} - (\boldsymbol{u}_0 \nabla') \boldsymbol{u} + \nabla' p = \boldsymbol{g}_2(\boldsymbol{x}', t'). \end{aligned} \tag{A.7}$$

This approach, of course, cannot (and need not to) be actually applied in practice, as it requires an a priori knowledge of the solution. For the current description, however, this is not essential. *Our emphasis here is rather on giving an example of motivation for the development of a non-deteriorating algorithm for long-term numerical computation when the source terms are assumed known.*

On one hand, the latter problem is important and interesting by itself from the standpoints of both theory and applications, for example, when flow perturbations are generated not by an immersed configuration (e.g., aircraft) but rather when the linearized flow equations are driven by known quadrupole-type sources that are often used for describing turbulence-generated noise. On the other hand, as shown by Ryaben'kii in [6,8], a universal construction of *unsteady artificial boundary conditions* (ABCs) for the near-field computations in fact reduces to the generation of appropriate right-hand sides to the far-field equations, and then solving the resulting linear nonhomogeneous problem throughout the space-time. We note that the term "ABCs" is generally used for external boundary conditions that serve as a closure to the truncated problem for the purpose of solving it numerically once the original problem is formulated on an unbounded domain, see [11]. The aforementioned appropriate right-hand sides that are built in [6,8] turn out to be concentrated inside $S'$ next to the external artificial boundary $\partial S'$. Thus, the numerical solution methodology that we refer to above is needed for constructing the ABCs as well.

The work by Ryaben'kii [6,8] treats the problem of unsteady ABCs in the finite-difference framework and *provides a general recipe for constructing such boundary conditions using the concept of difference potentials* (see [3–5,7]). The ABCs of [6,8] can be applied to different problems in various fields of scientific computing. In this paper we do not discuss the issue of ABCs per se. We, however, reiterate that an integral part of any ABCs of type [6,8] is a linear solver for nonhomogeneous equations with known right-hand sides. Thus, to implement the general methodology of [6,8] in a specific important case of long-term computations, we need to be able to numerically integrate linear equations with known right-hand sides over long time intervals. *The corresponding technique for wave radiation solutions is the focus of the analysis in this paper.*

Returning to the previous example with the linearized Euler equations, we simply assume the functions $g_1(\boldsymbol{x}', t')$ and $\boldsymbol{g}_2(\boldsymbol{x}', t')$, see (A.7), to be known, $\operatorname{supp} g_1(\boldsymbol{x}', t') \subset S' \times [0, +\infty)$ and $\operatorname{supp} \boldsymbol{g}_2(\boldsymbol{x}', t') \subset S' \times [0, +\infty)$; $S' = \{\boldsymbol{x}' \mid |\boldsymbol{x}'| \leqslant d/2\}$. In Eqs. (A.7), let us now change the independent variables back, i.e., from $(\boldsymbol{x}', t')$ to $(\boldsymbol{x}, t)$, see (A.2):

$$\begin{aligned} &\frac{\partial p}{\partial t} + \nabla \boldsymbol{v} = g_1\left(\boldsymbol{x} + \int_0^t \boldsymbol{u}_0(\theta)\,\mathrm{d}\theta, t\right) \equiv f_1(\boldsymbol{x}, t), \\ &\frac{\partial \boldsymbol{v}}{\partial t} + \nabla p = \boldsymbol{g}_2\left(\boldsymbol{x} + \int_0^t \boldsymbol{u}_0(\theta)\,\mathrm{d}\theta, t\right) \equiv \boldsymbol{f}_2(\boldsymbol{x}, t), \end{aligned} \tag{A.8}$$

where $p$ and $\boldsymbol{v}$ in (A.8) denote the perturbations with respect to the corresponding background values given in (A.4). Clearly,

$$\begin{aligned} &\operatorname{supp} f_1(\boldsymbol{x}, t) \subset \big\{(\boldsymbol{x}, t) \mid \boldsymbol{x} \in S(t),\ t > 0\big\}, \\ &\operatorname{supp} \boldsymbol{f}_2(\boldsymbol{x}, t) \subset \big\{(\boldsymbol{x}, t) \mid \boldsymbol{x} \in S(t),\ t > 0\big\}, \end{aligned} \tag{A.9}$$

where $S(t)$ is a ball of variables $\boldsymbol{x}$ defined by relation (3). Thus, Eqs. (A.8) can be interpreted as the equations of ambient acoustics driven by moving sources, at every given moment of time the sources are concentrated in the ball $S(t)$, see (3), of diameter $d$ centered at $\boldsymbol{x}_0(t)$.

To further simplify our considerations, let us assume that the velocity field $\boldsymbol{v}$ has a potential $\varphi = \varphi(\boldsymbol{x}, t)$ so that $\boldsymbol{v} = \nabla\varphi$. Then, we also have to assume that $\exists\psi = \psi(\boldsymbol{x}, t)$: $\boldsymbol{f}_2(\boldsymbol{x}, t) = \nabla\psi$; obviously $\operatorname{supp}\psi(\boldsymbol{x}, t) \subset \{(\boldsymbol{x}, t) \mid \boldsymbol{x} \in S(t),\ t > 0\}$ (see (A.9)). In so doing Eqs. (A.8) transform into:

$$\begin{aligned} &\frac{\partial p}{\partial t} + \Delta\varphi = f_1(\boldsymbol{x}, t), \\ &\nabla\left(\frac{\partial\varphi}{\partial t} + p - \psi\right) = 0. \end{aligned} \tag{A.10}$$

Clearly, the second equation of (A.10) implies that the expression in brackets is a function of time $t$ only. Using the existing uncertainty in the definition of $\varphi$ ($\boldsymbol{v} = \nabla\varphi$ and therefore an arbitrary function of time can be added to $\varphi$), we can always assume that

$$\frac{\partial\varphi}{\partial t} + p - \psi = 0. \tag{A.11}$$

Outside $S(t)$ Eq. (A.11) obviously translates into the standard definition of the potential $p = -\partial\varphi/\partial t$. Differentiating (A.11) in time and substituting the expression for $\partial p/\partial t$ into the first equation of (A.10) we recover the nonhomogeneous wave equation for $\varphi(\boldsymbol{x}, t)$:

$$-\frac{\partial^2\varphi}{\partial t^2} + \Delta\varphi = f_1(\boldsymbol{x}, t) + \frac{\partial\psi}{\partial t}(\boldsymbol{x}, t) \equiv f(\boldsymbol{x}, t). \tag{A.12}$$

Eq. (A.12) is the wave equation driven by a moving source: $\operatorname{supp} f(\boldsymbol{x}, t) \subset \{(\boldsymbol{x}, t) \mid \boldsymbol{x} \in S(t),\ t > 0\}$. It is practically the same as equation (1) except that in (A.12) the (dimensionless) speed of sound is equal to one.

To conclude, we should mention that besides acoustics (linearized Euler's equations), the lacunae-based algorithms can be built and likely prove useful for electromagnetics (Maxwell's equations) and elastodynamics (Lame's equations).

## References


[1] P. Garabedian, Partial Differential Equations, Chelsea, New York, 1986.
[2] F. John, Partial Differential Equations, Springer, New York, 1978.
[3] S.G. Mikhlin, N.F. Morozov, M.V. Paukshto, The Integral Equations of the Theory of Elasticity, B.G. Teubner, Stuttgart, 1995.
[4] V.S. Ryaben'kii, Boundary equations with projections, Russian Math. Surv. 40 (1985) 147–183.
[5] V.S. Ryaben'kii, Difference Potentials Method for Some Problems of Continuous Media Mechanics, Nauka, Moscow, 1987 (in Russian).
[6] V.S. Ryaben'kii, Exact transfer of difference boundary conditions, Functional Anal. Appl. 24 (3) (1990) 251–253.
[7] V.S. Ryaben'kii, Difference potentials method and its applications, Math. Nachr. 177 (1996) 251–264.
[8] V.S. Ryaben'kii, Nonreflecting time-dependent boundary conditions on artificial boundaries of varying location and shape, Appl. Numer. Math. 33 (2000) 481–492.
[9] V.S. Ryaben'kii, V.I. Turchaninov, S.V. Tsynkov, The use of lacunae of the three-dimensional wave equation for calculating the solution on long time intervals, Math. Model. 11 (12) (1999) 113–127 (in Russian).
[10] V.S. Ryaben'kii, S.V. Tsynkov, V.I. Turchaninov, Global discrete artificial boundary conditions for time-dependent wave propagation, to appear as ICASE Report, NASA Langley Research Center, Hampton, VA; also submitted to J. Comput. Phys.

[11] S.V. Tsynkov, Numerical solution of problems on unbounded domains. A review, Appl. Numer. Math. 27 (1998) 465–532.
[12] V.S. Vladimirov, Equations of Mathematical Physics, Dekker, New York, 1971.