



Long range FDTD over undulating terrain

Bram de Greve, Timothy Van Renterghem, Dick Botteldooren

Ghent University, Department of Information Technology, Sint-PietersNieuwstraat 41, B-9000 Ghent, Belgium,
{[bram.degrev](mailto:bram.degrev@intec.ugent.be),[timothy.van.renterghem](mailto:timothy.van.renterghem@intec.ugent.be),[dick.botteldooren](mailto:dick.botteldooren@intec.ugent.be)}@intec.ugent.be,

The Finite-Difference Time-Domain (FDTD) method has shown to be an excellent tool for solving the inhomogeneous moving medium sound propagation equations outdoors. Due to its nature, it can easily take care of complex wind and temperature profiles, typically found in outdoor situations. In its conventional form, the FDTD uses a uniform Cartesian grid. For propagation over an undulating terrain these grids need to be refined beyond the acoustic limit to allow accurate description of the ground surface. This increases memory requirements and cpu-time considerably. In this paper, we will discuss how these computational requirements can be relaxed to allow these simulations to be run on lower end machines.

The technique that is proposed contains two steps. The first part consists in introducing an optimized non-Cartesian FDTD scheme which uses a structured grid that follows the undulations of the terrain. It is shown that for the problem at hand, this general approach can be simplified keeping the increase in cpu-time per grid cell low. It will be shown that the approach allows us to keep the number of grid cells per wavelength low, while avoiding spurious reflections and diffractions. Thus, both memory requirements and computation time are considerably lowered.

The proposed approach has the additional advantage that it allows more easily to implement a grid co-moving with essentially unidirectional waves. Columns of the Cartesian grid are activated based on a pressure threshold. This is a simple but effective approach to limit the computing resources to the active areas, while retaining an accurate result.

1 Introduction

Detailed numerical modeling of outdoor sound propagation has gained interest recently both from an environmental and from a military point of view. The Finite-Difference Time-Domain (FDTD) technique has shown to be interesting to simulate sound propagation in an inhomogeneous moving medium [1, 2, 3]. It turns out to be considerably more efficient if a structured cartesian grid can be used. Although such a grid allows to discretise most buildings, noise barriers, etc. very well, it is inefficient to model propagation over an undulating terrain. In this paper, we propose a structured non-cartesian grid FDTD as a possible solution. This approach has the additional advantage that it allows to use a moving grid-window. This keeps memory usage within limits even for studying very long range propagation. The model is compared to a grid-refined cartesian approach and a Generalized Terrain PE (GTPE) simulation [4, 5].

2 Inhomogeneous moving medium sound propagation equations

The following set of linearized equations is commonly used for time-domain simulations outdoors. Sound propagation is described in an inhomogeneous moving medium. The main effects of wind flow outdoors (convection, refraction and scattering) and refraction by tem-

perature gradients are accounted for in detail. Atmospheric absorption is not included.

$$\frac{\partial p}{\partial t} + \mathbf{v}_0 \cdot \nabla p + c^2 \rho_0 \nabla \cdot \mathbf{v} = 0 \quad (1a)$$

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}_0 \cdot \nabla) \cdot \mathbf{v} + (\mathbf{v} \cdot \nabla) \mathbf{v}_0 + \frac{1}{\rho_0} \nabla p = 0 \quad (1b)$$

In the previous equations, \mathbf{v} is the particle velocity, \mathbf{v}_0 is the background flow velocity, p is the acoustic pressure, ρ_0 is the ambient mass density, c is the adiabatic speed of sound and t is time. The integration of these equations is performed with the Finite-Difference Time-Domain (FDTD) method. A staggered spatial grid is used in combination with the prediction-step staggered-in-time (PSIT) approach [2]. The background flow velocities that will be used in this paper are sufficiently small to perform stable and accurate calculations with the PSIT scheme. The following notations are used:

$$p_{i,j}^l = p(i\Delta x, j\Delta y, l\Delta t) \quad (2a)$$

$$v_{x,i+\frac{1}{2},j}^{l+\frac{1}{2}} = v_x((i+\frac{1}{2})\Delta x, j\Delta y, (l+\frac{1}{2})\Delta t) \quad (2b)$$

$$v_{y,i,j+\frac{1}{2}}^{l+\frac{1}{2}} = v_y(i\Delta x, (j+\frac{1}{2})\Delta y, (l+\frac{1}{2})\Delta t) \quad (2c)$$

The discretised pressures are defined at the integer spaces and time indices, the velocity components are put in between the latter. The staggered-in-time approach allows for in-place computation, which is computational efficient. The acoustic velocities and pressures are updated in an alternating *leapfrog* manner.

3 Curvilinear Coordinates

The grid resolution required for FDTD simulations depends on the wavelength of interest and the detail required to describe the structure. To model the smooth surface of an undulating terrain accurately, a grid resolution beyond the acoustic limit of ten cells per wavelength is required.

To reduce the required grid resolution, a structured grid that follows the undulations of the terrain can be applied. The equations must therefore be rewritten in a curvilinear coordinate system [6, 7]. It will be shown that for the problem at hand, this general approach can be simplified to minimize the computational overhead.

For the purpose of this paper, a non-orthonormal grid is constructed that is skewed to follow the undulations $h(x)$ of the terrain. $\mathbf{1}_x$ and $\mathbf{1}_y$ are the base vectors of the underlying orthonormal reference grid, u_x and u_y are the coordinates in the curvilinear grid:

$$\mathbf{r}(u_x, u_y) = x(u_x, u_y) \mathbf{1}_x + y(u_x, u_y) \mathbf{1}_y \quad (3)$$

$$x(u_x, u_y) = u_x \quad (4a)$$

$$y(u_x, u_y) = u_y + h(u_x) \quad (4b)$$

To this non-orthogonal coordinate system, a covariant base can be defined so that a vector field such as the particle velocity field \mathbf{v} can be split in covariant components:

$$\mathbf{A}_x = \frac{\partial \mathbf{r}}{\partial u_x} = \mathbf{1}_x + \frac{dh}{du_x} \mathbf{1}_y \quad (5a)$$

$$\mathbf{A}_y = \frac{\partial \mathbf{r}}{\partial u_y} = \mathbf{1}_y \quad (5b)$$

$$\mathbf{v} = v_x \mathbf{A}_x + v_y \mathbf{A}_y \quad (6)$$

\mathbf{a}^\perp is defined as the result of rotating a 90° counterclockwise:

$$\mathbf{a}^\perp = -(\mathbf{a} \cdot \mathbf{1}_y) \mathbf{1}_x + (\mathbf{a} \cdot \mathbf{1}_x) \mathbf{1}_y \quad (7)$$

This allows to write a property that expresses that the area A of the unit cell is constant:

$$A = \mathbf{A}_x^\perp \cdot \mathbf{A}_y = \left(-\frac{dh}{du_x} \mathbf{1}_x + \mathbf{1}_y \right) \cdot \mathbf{1}_y = 1 \quad (8)$$

A contravariant base is also associated to the curvilinear grid, and a vector \mathbf{v} can be split in its contravariant components as well:

$$\mathbf{A}^x = \frac{-\mathbf{A}_x^\perp}{A} = \mathbf{1}_x \quad (9a)$$

$$\mathbf{A}^y = \frac{\mathbf{A}_y^\perp}{A} = -\frac{dh}{du_x} \mathbf{1}_x + \mathbf{1}_y \quad (9b)$$

$$\mathbf{v} = v^x \mathbf{A}^x + v^y \mathbf{A}^y \quad (10)$$

A property of the contravariant base is that it can be used to obtain the covariant components of a vector, and vice versa:

$$v_x = \mathbf{v} \cdot \mathbf{A}^x \quad v_y = \mathbf{v} \cdot \mathbf{A}^y \quad (11a)$$

$$v^x = \mathbf{v} \cdot \mathbf{A}_x \quad v^y = \mathbf{v} \cdot \mathbf{A}_y \quad (11b)$$

From this, a metric tensor g to convert contra- to covariant coordinates can be defined as:

$$g^{\alpha\beta} = \mathbf{A}^\alpha \cdot \mathbf{A}^\beta \quad (12)$$

$$\begin{bmatrix} g^{xx} & g^{xy} \\ g^{yx} & g^{yy} \end{bmatrix} = \begin{bmatrix} 1 & -\frac{dh}{du_x} \\ -\frac{dh}{du_x} & 1 + \left(\frac{dh}{du_x}\right)^2 \end{bmatrix} \quad (13)$$

It is interesting to note that the co- and contravariant bases and metric tensor are all independent of u_y and t . This allows for an efficient implementation because a single value can be calculated, stored and used per column.

Differential operators in curvilinear coordinates are expanded as:

$$(\nabla\phi)^\alpha = \nabla\phi \cdot \mathbf{A}_\alpha = \frac{\partial\phi}{\partial u_\alpha} \quad (14a)$$

$$\begin{aligned} \nabla \cdot \mathbf{v} &= \frac{1}{A} \left[\frac{\partial}{\partial u_x} (Av_x) + \frac{\partial}{\partial u_y} (Av_y) \right] \\ &= \frac{\partial v_x}{\partial u_x} + \frac{\partial v_y}{\partial u_y} \end{aligned} \quad (14b)$$

$$\begin{aligned} \mathbf{v} \cdot \nabla\phi &= (v_x \mathbf{A}_x + v_y \mathbf{A}_y) \cdot \left(\frac{\partial\phi}{\partial u_x} \mathbf{A}^x + \frac{\partial\phi}{\partial u_y} \mathbf{A}^y \right) \\ &= v_x \frac{\partial\phi}{\partial u_x} + v_y \frac{\partial\phi}{\partial u_y} \end{aligned} \quad (14c)$$

4 Terrain FDTD

To implement FDTD over an undulating terrain, the curvilinear differential operators (14) must be applied to the sound propagation equations (1).

4.1 Pressure equation

The pressure equation (1a) in curvilinear coordinates is expanded to:

$$\begin{aligned} p_{i,j}^l &= p_{i,j}^{l-1} - \Delta t c^2 \rho_0 \nabla \cdot \mathbf{v}_{i,j}^{l-\frac{1}{2}} \\ &\quad - \Delta t \mathbf{v}_{0,i,j} \cdot \nabla p_{p,i,j}^{l-\frac{1}{2}} \end{aligned} \quad (15)$$

with

$$\begin{aligned} \nabla \cdot \mathbf{v}_{i,j}^{l-\frac{1}{2}} &= \frac{v_{x,i+\frac{1}{2},j}^{l-\frac{1}{2}} - v_{x,i-\frac{1}{2},j}^{l-\frac{1}{2}}}{\Delta x} \\ &+ \frac{v_{y,i,j+\frac{1}{2}}^{l-\frac{1}{2}} - v_{y,i,j-\frac{1}{2}}^{l-\frac{1}{2}}}{\Delta y} \end{aligned} \quad (16a)$$

$$\begin{aligned} \mathbf{v}_{0,i,j} \cdot \nabla p_{p,i,j}^{l-\frac{1}{2}} &= \frac{v_{0,x,i-\frac{1}{2},j} + v_{0,x,i+\frac{1}{2},j}}{2} \cdot \\ &\frac{p_{p,i+1,j}^{l-\frac{1}{2}} - p_{p,i-1,j}^{l-\frac{1}{2}}}{2\Delta u_x} \\ &+ \frac{v_{0,y,i,j-\frac{1}{2}} + v_{0,y,i,j+\frac{1}{2}}}{2} \cdot \\ &\frac{p_{p,i,j+1}^{l-\frac{1}{2}} - p_{p,i,j-1}^{l-\frac{1}{2}}}{2\Delta u_y} \end{aligned} \quad (16b)$$

$$p_{p,i,j}^{l-\frac{1}{2}} = p_{i,j}^l - \frac{\Delta t}{2} c^2 \rho_0 \nabla \cdot \mathbf{v}_{i,j}^{l-\frac{1}{2}} \quad (16c)$$

where v_x , v_y , $v_{0,x}$ and $v_{0,y}$ are the covariant velocity components.

4.2 Contravariant velocity equations

From (14a) it follows that the velocity equations must be written using the contravariant velocity components. For the x component of velocity we obtain:

$$\begin{aligned} v_{i+\frac{1}{2},j}^{x,l+\frac{1}{2}} &= v_{i+\frac{1}{2},j}^{x,l-\frac{1}{2}} - \frac{\Delta t}{\rho_0} \frac{\partial}{\partial u_x} p_{i+\frac{1}{2},j}^l \\ &- \mathbf{v}_{0,i+\frac{1}{2},j} \cdot \nabla v_{p,i+\frac{1}{2},j}^{x,l} \\ &- \mathbf{v}_{\mathbf{P},i+\frac{1}{2},j} \cdot \nabla v_{0,i+\frac{1}{2},j}^{x,l} \end{aligned} \quad (17)$$

with

$$\frac{\partial}{\partial u_x} p_{i+\frac{1}{2},j}^l = \frac{p_{i+1,j}^l - p_{i,j}^l}{\Delta u_x} \quad (18a)$$

$$\begin{aligned} \mathbf{v}_{0,i+\frac{1}{2},j} \cdot \nabla v_{p,i+\frac{1}{2},j}^{x,l} &= v_{0,x,i+\frac{1}{2},j} \frac{v_{p,i+\frac{3}{2},j}^{x,l} - v_{p,i-\frac{1}{2},j}^{x,l}}{2\Delta u_x} \\ &+ \frac{\sum_{\mu=0}^1 \sum_{\nu=0}^1 v_{0,y,i+\mu,j+\nu-\frac{1}{2}}}{4} \cdot \\ &\frac{v_{p,i+\frac{1}{2},j+1}^{x,l} - v_{p,i-\frac{1}{2},j+1}^{x,l}}{2\Delta u_y} \end{aligned} \quad (18b)$$

$$\begin{aligned} \mathbf{v}_{\mathbf{P},i+\frac{1}{2},j} \cdot \nabla v_{0,i+\frac{1}{2},j}^{x,l} &= v_{p,x,i+\frac{1}{2},j} \frac{v_{0,i+\frac{3}{2},j} - v_{0,i-\frac{1}{2},j}}{2\Delta u_x} \\ &+ \frac{\sum_{\mu=0}^1 \sum_{\nu=0}^1 v_{p,y,i+\mu,j+\nu-\frac{1}{2}}}{4} \cdot \\ &\frac{v_{0,i+\frac{1}{2},j+1} - v_{0,i-\frac{1}{2},j+1}}{2\Delta u_y} \end{aligned} \quad (18c)$$

$$v_{p,i+\frac{1}{2},j}^{x,l} = v_{i+\frac{1}{2},j}^{x,l-\frac{1}{2}} - \frac{\Delta t}{2\rho_0} \frac{\partial}{\partial u_x} p_{i+\frac{1}{2},j}^l \quad (18d)$$

where v^x , v^y , v_p^x and v_p^y are contravariant velocity components.

4.3 Velocity transformation

The pressure equation (15) uses the covariant velocity components, while the velocity equation (17) obtains contravariant components. To *close the loop*, a final transformation from contra- to covariant components is needed:

$$v_{x,i+\frac{1}{2},j}^{l+\frac{1}{2}} = g_{i+\frac{1}{2}}^{xx} v_{i+\frac{1}{2},j}^{x,l+\frac{1}{2}} + g_{i+\frac{1}{2}}^{xy} v_{i+\frac{1}{2},j}^{y,l+\frac{1}{2}} \quad (19a)$$

$$v_{y,i,j+\frac{1}{2}}^{l+\frac{1}{2}} = g_i^{yx} v_{i,j+\frac{1}{2}}^{x,l+\frac{1}{2}} + g_i^{yy} v_{i,j+\frac{1}{2}}^{y,l+\frac{1}{2}} \quad (19b)$$

$v_{i+\frac{1}{2},j}^{y,l+\frac{1}{2}}$ and $v_{i,j+\frac{1}{2}}^{x,l+\frac{1}{2}}$ are not in the grid as such, and must be linearly interpolated from the four closest values:

$$v_{i+\frac{1}{2},j}^{y,l+\frac{1}{2}} = \frac{\sum_{\mu=0}^1 \sum_{\nu=0}^1 v_{i+\mu,j+\nu-\frac{1}{2}}^{y,l+\frac{1}{2}}}{4} \quad (20a)$$

$$v_{i,j+\frac{1}{2}}^{x,l+\frac{1}{2}} = \frac{\sum_{\mu=0}^1 \sum_{\nu=0}^1 v_{i+\mu-\frac{1}{2},j+\nu}^{x,l+\frac{1}{2}}}{4} \quad (20b)$$

4.4 Boundary condition

The whole purpose of the terrain FDTD is to model the undulations of the terrain more accurately. It is obviously necessary to implement an accurate terrain boundary condition. To model a boundary with a locally reacting surface impedance, the normal component of velocity has to be known on the boundary. The normal vector \mathbf{n} for a ground boundary condition is proportional to the contravariant base vector \mathbf{A}^y :

$$\mathbf{n} = \frac{\mathbf{A}^y}{\|\mathbf{A}^y\|} \quad (21)$$

Using (11a), the normal component of velocity is found to be directly proportional to the covariant velocity component:

$$v_n = \mathbf{v} \cdot \mathbf{n} = \frac{v_x}{\|\mathbf{A}^y\|} \quad (22)$$

It follows from this that a surface impedance boundary condition cannot be directly implemented using equations (15), (17) and (19). A workaround using interpolations can be carried out, but a more sustainable solution is to transform the velocity equations to work on the covariant components of velocity instead.

4.5 Covariant velocity equations

To transform the equation of the x component of velocity, (17) is substituted in (19):

$$\begin{aligned}
 v_{x,i+\frac{1}{2},j}^{l+\frac{1}{2}} &= v_{x,i+\frac{1}{2},j}^{l-\frac{1}{2}} \\
 &- \frac{\Delta t}{\rho_0} g_{i+\frac{1}{2},j}^{xx} \frac{\partial}{\partial u_x} p_{i+\frac{1}{2},j}^l \\
 &- \frac{\Delta t}{\rho_0} g_{i+\frac{1}{2},j}^{xy} \frac{\partial}{\partial u_y} p_{i+\frac{1}{2},j}^l \\
 &- \mathbf{v}_{0,i+\frac{1}{2},j} \cdot \nabla v_{p,x,i+\frac{1}{2},j}^l \\
 &- \mathbf{v}_{\mathbf{p},i+\frac{1}{2},j} \cdot \nabla v_{0,x,i+\frac{1}{2},j}^l
 \end{aligned} \tag{23}$$

with

$$\begin{aligned}
 \frac{\partial}{\partial u_y} p_{i+\frac{1}{2},j}^l &= \frac{p_{i,j+1}^l - p_{i,j}^l}{2\Delta u_x} \\
 &+ \frac{p_{i+1,j+1}^l - p_{i+1,j}^l}{2\Delta u_x}
 \end{aligned} \tag{24a}$$

$$\begin{aligned}
 v_{p,x,i+\frac{1}{2},j}^l &= v_{x,i+\frac{1}{2},j}^{l-\frac{1}{2}} \\
 &- \frac{\Delta t}{2\rho_0} g_{i+\frac{1}{2},j}^{xx} \frac{\partial}{\partial u_x} p_{i+\frac{1}{2},j}^l \\
 &- \frac{\Delta t}{2\rho_0} g_{i+\frac{1}{2},j}^{xy} \frac{\partial}{\partial u_y} p_{i+\frac{1}{2},j}^l
 \end{aligned} \tag{24b}$$

This necessary velocity transformation is now embedded in equation (23) which replaces both (17) and (19) in the FDTD scheme.

5 Automatic grid window

The terrain FDTD has the additional advantage that it is more easy to implement a moving grid window. Although it is possible to use such a window with a regular cartesian grid, the terrain FDTD has the benefit that the window entirely moves through the propagating medium. No tweaks to adjust the column length or the cell equation have to be applied.

When simulating propagation of pulse-like source signals, only a fragment of the grid and computation time is usefully spend on propagating the pulse. A lot of computational resources is wasted on computing zeros. It is tempting to limit the grid to only surround the pulse so that no resources are wasted. In long range simulations, this is particularly easy because the pulse will behave much like a planar wave propagating horizontally, after it has traveled some distance. A moving horizontal window can be applied to surround the pulse more tightly.

Many implementations of moving grid windows use a fixed window width that moves at constant speed across the grid. The advantage is an easy implementation, even

when dynamic memory management is lacking. However, the disadvantages of this approach is that it doesn't allow for reflections, and the pulse width may exceed the window width after some time.

In languages with dynamic memory management, an automatic window that automatically allocate grid columns based on a pressure threshold p_{min} can be used. When a pressure value in a grid column exceeds p_{min} , n additional columns are activated on each side, to allow the pressure field to *leak* into the new columns. A column is deallocated when no pressure value is above the threshold in the neighbourhood of $2n + 1$ columns.

The benefit of an automatic grid window is that it will allocate as much columns as necessary to capture the pulse. More importantly, it also is capable of capturing reflections of the pulse on various objects. The window can automatically be split two separate windows moving in opposite directions. To separate windows will also merge automatically if window collision occurs. This is all transparent to the algorithm.

6 Results

6.1 Comparison to refined cartesian FDTD

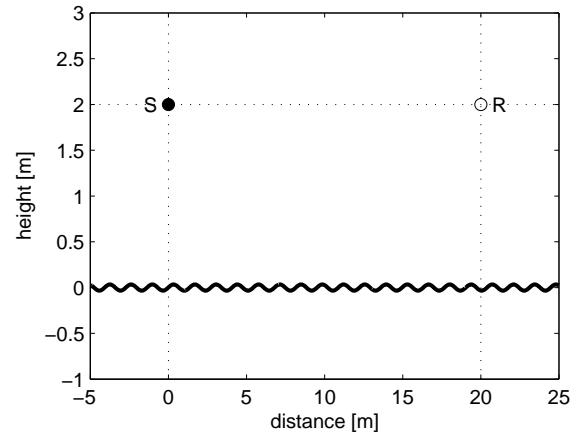


Figure 1: setup of simulation for comparison of GTPE and TFDTD

The terrain FDTD (TFDTD) is tested against a regular cartesian FDTD with orthogonal grid. The source S and receiver R are put at distance of $W = 20$ m from each other, and at a height of $H = 2$ m above an undulating surface, as in Figure 1. A sinusoidally shaped terrain is used with wavelength $\lambda_g = 0.34$ m and amplitude one tenth of the wavelength:

$$h(x) = \frac{\lambda_g}{10} \sin \frac{2\pi x}{\lambda_g} \tag{25}$$

The cellsize of the TFDTD is $\Delta x_1 = 0.02$ m. CFDTD1 is a cartesian FDTD with the same cellsize of TFDTD. CFDTD2 and CFDTD4 are cartesian FDTDs too with a two and four times refined grid respectively ($\Delta x_2 = 0.01$ m, $\Delta x_4 = 0.005$ m). TFDTD has the same memory requirements as CFDTD1 and is only slightly slower. CFDTD2 and CFDTD4 consume respectively 4 and 16 more memory and are about 8 and 64 times slower.

Since the terrain is sinusoidally shaped, Bragg reflection will occur. The wave number of the terrain undulations is $k_g = \frac{2\pi}{\lambda_g}$. Additional Bragg reflection orders will cut-in at frequencies f_n :

$$f_n = \frac{c}{2\pi} \frac{nk_g}{2} = 500n \text{ Hz} \quad (26)$$

Figure 2 compares simulation results. At low frequencies, the ground interference dip is clearly observed. As the cartesian grid is refined, CFDTD and TFDTD converge to each other, but convergence is slow. The cut-in at 500 Hz is well described by all numerical models. This is not surprising since it solely depends on the periodicity of the ground surface and not its fine structure.

6.2 Comparison to generalized terrain PE method

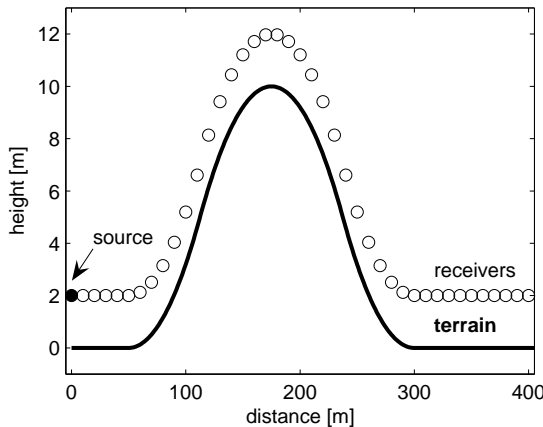


Figure 3: setup of simulation for comparison between GTPE and TFDTD.

For comparison with generalized terrain PE, the same setup is used as in [5], page 79. Figure 3 shows a hill with a height of $H = 10$ m and a length of $W = 250$ m, starting from $x_0 = 50$ m. It is composed of three circular arcs with radius R . The central arc is positioned between $x_0 + \frac{W}{4}$ and $x_0 + \frac{3W}{4}$. The source receivers are positioned every 10 m at an altitude of 2 m above the ground, which is modeled as a perfectly reflecting surface.

$$R = \frac{H}{4} + \frac{W^2}{16H} \quad (27)$$

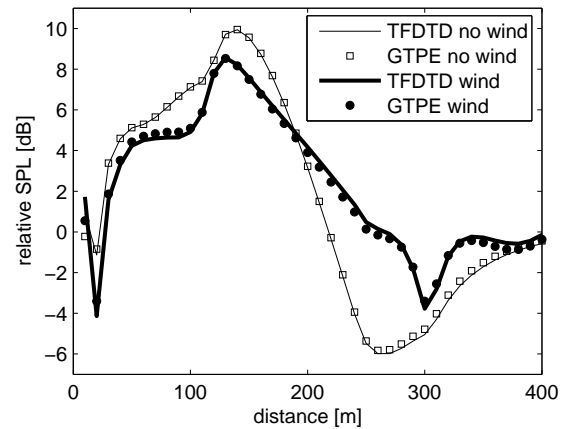


Figure 4: comparison of sound pressure level relative to free field between GTPE and TFDTD. $f = 300$ Hz, $b = 1$ m/s, $z_0 = 0.1$ m

Two situations are considered. The first has a non-moving medium. In the second, a logarithmic wind profile in function of the altitude above the ground is applied, with $b = 1$ m/s and $z_0 = 0.1$ m.

$$v_0(x, y) = b \ln \left(\frac{y - h(x)}{z_0} + 1 \right) \mathbf{1}_x \quad (28)$$

Figure 4 shows for both situations the relative sound pressure level of frequency $f = 300$ Hz, computed with terrain FDTD (TFDTD) and generalized terrain PE (GTPE). The agreement between the TFDTD and GTPE method is very good. Though, it must be stressed that the example is chosen for GTPE to work well. From this comparison however, we can get confidence that due to its nature TFDTD may be also useable in more complex situations, though this is to be confirmed with further experiments.

7 Conclusion

It is shown how the inhomogeneous moving medium sound propagation equations can be implemented in Finite-Difference Time-Domain (FDTD) using a structured skewed grid to follow the undulations of a terrain $h(x)$. The velocity equations are transformed to the covariant base to make it easier to implement surface impedance boundary conditions. An automatic grid window can reduce the computational resource for pulse-like source signals while retaining reflections.

The terrain FDTD (TFDTD) is compared to both a refined cartesian FDTD (CFDTD) and a generalized terrain PE (GTPE) method. As CFDTD is refined, it slowly converge to TFDTD which is as computational complex as CFDTD1. The agreement between TFDTD and GTPE method is very good.

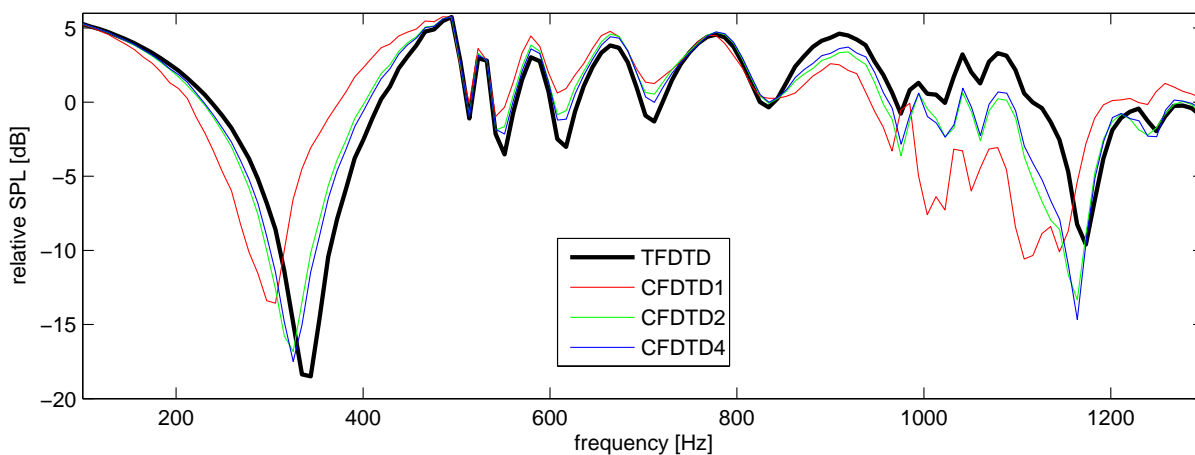


Figure 2: comparison of sound pressure level relative to free field between TFDTD and different refined CFDTDs. TFDTD and CFDTD1: $\Delta x = 0.02$ m, CFDTD2: $\Delta x = 0.01$ m, CFDTD4: $\Delta x = 0.005$ m.

References

- [1] R. Blumrich and D. Heimann, 'A linearized Eulerian sound propagation model for studies of complex meteorological effects', *J. Acoust. Soc. Am.*, Vol. 112. No. 2. pp. 446–455 (2002)
- [2] T. Van Renterghem, 'The finite-difference time-domain method for the simulation of sound propagation in a moving medium', *Phd thesis*, Universiteit Gent, Belgium (2003)
- [3] E.V. Ostashev, D.K. Wilson, L.B. Liu, D.F. Aldridge, N.P. Symons, D. Marlin, 'Equations for finite-difference, time-domain simulation of sound propagation in moving inhomogeneous media and numerical implementation', *J. Acoust. Soc. Am.*, Vol. 117. No. 2. pp. 503–517 (2005)
- [4] R.A. Sack and M. West, 'A parabolic equation for sound propagation in two dimensions over any smooth terrain profile: the Generalized Terrain Parabolic Equation (GT-PE)', *Appl. Acoust.*, Vol. 45. pp. 113–129 (1995)
- [5] E.M. Salomons, '*Computational atmospheric acoustics*', Kluwer Academic Publishers, Dordrecht, ISBN 0-7923-7161-5 (2001)
- [6] M. Fusco, 'FDTD algorithm in curvilinear coordinates', *IEEE Trans. Antennas & Propagat.*, Vol. 38. No. 1. pp. 76–89 (1990)
- [7] R. Holland, 'Finite-Difference solutions of Maxwell's equations in generalised nonorthogonal coordinates', *IEEE Trans. Nucl. Sci.*, Vol. 30. No. 6. pp. 4589–4591 (1983)



ForumAcusticum
BUDAPEST



29 Aug – 2 Sep, 2005
Budapest, Hungary

PROCEEDINGS



www.fa2005.org



Copyright ©2005
OPAKFI Tudományos Egyesület

ISBN 963 8241 68 3

All rights reserved. No part of this CD-ROM may be reproduced or transmitted in any form or by any means without the written permission from OPAKFI.