Acta
Technica
Jaurinensis

Vol. 8, No. 1, pp. 47–62, 2015
DOI: 10.14513/actatechjaur.v8.n1.348
Available online at acta.sze.hu

On Wavelet Based Modeling of the Nitrogen Oxides Emission and Concentration due to Road Traffic in Urban Environment

Sz. Nagy

**Széchenyi István University, Department of Telecommunications,
Egyetem tér 1., H-9026 Győr, Hungary
e-mail: nagysz@sze.hu**

Abstract: Differential equations can be solved wavelet-based by representing the continuous functions by their wavelet expansion coefficients and thus the corresponding differential equations are transformed to matrix equations. The wavelet basis functions are organized into resolution levels of different frequency terms at different locations, and the main advantage of the wavelet expansion representation is that the wavelet based differential equation solving methods can be adaptive, it is possible to refine the solution locally, if the precision is not sufficient at some regions.

In case of the nitrogen oxides convection-advection equation, the urban environment should be taken as special material parameter in the differential equation's operator, and the matrix elements of the differential operator has to be calculated in a non-continuous environment, and the obstacles are placed so, that they are not at the boundaries of the support of the wavelets.

Keywords: nitrogen oxides, wavelets, matrix elements, convection-advection, differential equation

1. Introduction

Multiresolution analysis (MRA) or the wavelet analysis (the wavelet theory's basics are precisely summarized e.g., in [1, 2]) is a widely used tool of data and image processing, and it also has increasing share within differential equation solving methods. However, in modeling pollutant's convection, advection and dispersion they are only sporadically used [3] even though they have the natural ability to use different scales simultaneously.

The passengers and other participants of the urban traffic are exposed to the pollutants exhausted by the vehicles, and several of these pollutants are toxic, carcinogenic, or at least maleficent for living beings. The pollutants in the air are monitored regularly, but usually on only a few points per city and few times per day. Of course, these data are also valuable but higher spatial and temporal resolution would be more desirable, especially because of epidemiological reasons, like monitoring the effect of the pollutant concentration on respiratory organ, or cardiovascular problems [4, 5].

The combination of the measured data with computational modeling is a recent trend in approximating the concentration of the pollutants [8, 9] – like one of the most dangerous group, the nitrogen oxides –, and the used models can be grouped into two subgroups, the large-scale models tend to apply grids with grid distance of tens of kilometers [6], while the small-scale ones can not treat larger scale tendencies [7]. Crooks and Isakov use wavelets to combine the two scales [3] based on the scalability of the wavelets: that the resolution level can be different at different locations, if the solution function contains higher frequency terms in one place and restricted to lower frequencies at other places, like the urban and rural environment. This property is also the main reason of their applications in image processing and compressing, like in the JPEG2000 standard [10], or the NASA's ICER [11]. However, none of the models in [3] calculate on wavelet basis, just the combination scheme is wavelet based.

This article shortly introduces the wavelet analysis in the next section, then derives the advection-diffusion equation's discretized form, and as the last step, gives calculation rules for determining the discretized form in urban environment. The goal is not to solve the diffusion equation in wavelet basis, it can be found in many applications, like [12–14], but to give a solution for modeling the urban environment – mainly the position of the buildings and roads, the various wind speed, – in formulating the discretized matrix equations and calculating the matrix elements. For a proper modeling of the obstacles locally different diffusion coefficients would be necessary, and these coefficients can be well approximated as step functions of the position. The matrix elements generated from step functions are calculated in this contribution.

2. On wavelet analysis

Wavelets, the basis functions of MRA, are localized functions, they arise as dilations and translations of one common mother wavelet, moreover they can be used as simple building elements in expansion of continuous functions and they can describe the different features of different scales and positions well. Although wavelet-based methods, similarly to the finite elements solving schemes, are also members of the Galerkin-type solver family [15], the discretization technique in this case is more flexible, it can be systematically adapted to the problem during the solution itself, no previous knowledge or guess is necessary about the possible solution and its detailed and roughly representable regions. Wavelets are the basis functions of the system, they form an infinite series of refining resolution levels, each doubling the frequency resolution.

The effectiveness of the MRA technique lies in the fact, that in higher resolution levels most of the wavelets are unnecessary for sufficiently precise resolution, most of the expansion coefficients are zero in these levels and thus they should not be stored and should not be included into the calculations [13, 16–21]. Calculating these coefficients and discarding them would be the first approximation, however, it is a waste of time and capacity if a method can be derived that predicts, which coefficients are the most important ones, and which are almost zero. In order to ensure the adaptivity and economy, a clear, quick, easily calculable prediction algorithm is necessary to decide which wavelets of the next resolution level are needed and which can be neglected, and such algorithms exist both for elliptical differential equations with sources [22] and for eigenvalue type differential equations [23]. In these applications, starting from an already calculated resolution level, the authors produce an economic prediction algorithm for determining whether a given wavelet in a given position is necessary for the sufficiently precise next level solution or it can be neglected. The method can be used for estimating the error of the given resolution solution.

2.1. Multiresolution analysis

In multiresolution analysis (MRA) or wavelet analysis the studied Hilbert space is decomposed into infinite resolution level subspaces, each one embedded to the higher resolution level subspaces, $V_m \subset V_{m+1}$ $\{V_m,\ m \in \mathbb{Z}\}$. The basic resolution level V_0 is spanned by a single function, a so called *mother scaling function* $\phi(x)$, all the basis functions in the subspace are shifted versions of $\phi(x)$ on an equidistant grid as $\phi_\ell(x) = \phi(x - \ell)$,$\ell \in \mathbb{Z}$.

The higher resolution level mother scaling functions are generated from the basic resolution level one as $\phi(2^m x)$, and the basis functions are $\phi_{m,\ell}(x) = \phi(2^m x - \ell)$, which means that the grid is shrunk, as well. A very important property of the scaling functions is that the finer resolution level subspace V_{m+1} contains the lower resolution level subspaces, like V_m, thus all the functions in the rougher resolution level subspace – as the mother scaling function itself – can be expanded at the finer resolution subspace. This expression of the mother scaling function as a linear combination of the next level basis functions

$$\phi(x) = 2^{1/2} \sum_{i=0}^{N_s} h_i \phi(2x - i), \tag{1}$$

is called the *refinement equation*, and it is one of the basic equations of wavelet analysis. The expansion coefficients h_i determine the scaling function's shape, with the normalization condition $\sum_{i=0}^{N_s} h_i = 1$, and the number N_s gives the support of the mother scaling function, i.e., the support is $[0,N_s)$. In most of the cases, the larger support means smoother scaling functions, e.g., in the Daubechies scaling function family, the first one has $N_s = 1$ and it is a step function, the second one with $N_s = 3$ is everywhere continuous, but not everywhere differentiable, the third one is differentiable ($N_s = 5$), the fifth is twice differentiable ($N_s = 9$), etc.

A function, like the concentration of the nitrogen oxides in the air can be expressed – or usually just approximated – at resolution level M as

$$C^{[M]}(x) = \sum_{\ell \in \Omega_M} c_{M\ell}\, \phi_{M\ell}(x), \tag{2}$$

where $c_{M\ell}$ are the expansion coefficients, i.e. the discretized version of the concentration, and all the indices, variables can be in one two or three dimensions – in higher dimensions, a scaling function can be e.g., $\phi_{m_1,m_2,\ell_1,\ell_2}(x_1,x_2) = \phi_{m_1,\ell_1}(x_1) \cdot \phi_{m_2,\ell_2}(x_2)$, but other, non diadic product higher dimensions scaling functions exist, too, though very rarely used. The domain Ω_M where the indices run is connected to the support of the function $C(x)$ and the scaling functions and the number of its elements approximately doubles at each resolution level increment.

2.2. Wavelets and details

Wavelets are basis functions of the subspaces given as a difference between two neighboring resolution level scaling function subsets,

$$V_{m+1} = V_m \oplus W_m \tag{3}$$

These subsets, the so called *detail spaces*, are also generated from one single *mother wavelet* $\psi(x)$, similarly to the scaling functions, as $\psi_{m,\ell}(x) = \psi(2^m x - \ell)$.

The wavelets can also be expanded at the higher resolution level subspaces, like

$$\psi(x) = 2^{1/2} \sum_{i=0}^{N_s} (-1)^i h^*_{-i+1} \phi(2x - i), \tag{4}$$

with the same coefficients h_i as in (1). Here, the sign * means complex conjugation. As a result of introducing the wavelets, a function $C(x)$ can approximated at resolution level M not only as (2), but also as

$$C^{[M]}(x) = \sum_{\ell \in \Omega_0} c_{0\ell}\, \phi_{0\ell}(x) + \sum_{m=0}^{M-1} \sum_{\ell \in \Omega_m} d_{m\ell}\, \psi_{m\ell}(x). \tag{5}$$

Here Ω_m contains all the wavelets of resolution level m that overlap with the support of the function $C(x)$.

Theoretically the number of coefficients to be treated in (2) and in (5) are the same, but most of the coefficients $d_{m,\ell}$ are very small and can be neglected. Smooth functions can be approximated very precisely in low resolution levels, only those parts of the functions need higher resolution, where the function varies rapidly, or has derivative discontinuities.

Of course, the sets of coefficients $c_{M,\ell}$ and $c_{0,\ell}$ with $d_{m,\ell}$ can be transformed into each other using equations (1), (4) and their inverse.

3. The advection diffusion equation for nitrogen oxides

Nitrogen oxides are usually from high temperature combustion, in case of urban environment, the traffic and the combustion engines are the main source of the nitrogen monoxide, NO. As NO is a free radical, having one unpaired electron, it is easily oxidized in the air to nitrogen dioxide, NO_2, that forms the infamous brownish dome above larger cities, and it can be further oxidized to acids, or other compounds depending on the other pollutants.

Air quality modeling, especially the modeling of the nitrogen oxides concentration, where several gases transform to one another can be carried out using weather conditions – like the moisture and the wind speed –, average vehicle count, etc. for describing these phenomena convection–advection equation in 2D along the coordinate x [24, 25]

$$\left(\frac{\partial}{\partial t} - \mu\frac{\partial^2}{\partial x^2} + u\frac{\partial}{\partial x} - \sigma\right) C(x,y,t) = F(x,y,t), \tag{6}$$

is a good approximation. Here μ is the diffusion coefficient from Fick's law, u is the velocity of the wind in the direction of x, σ is the constant that covers the concentration changes due to chemical reactions, and $F(x,y,t)$ describes the sources of the pollutants.

It is possible to use the three dimensional version of the above equations, with proper height and 3D wind and drift velocities, but as a first step, for demonstration, Eq. (6) is sufficient.

3.1. Discretization of the equation

As a first step, Eq. (6) is discretized in time [22], i.e., it is approximated with a Crank-Nicholson finite difference equation. This scheme can be used in wavelet-based partial differential equation solvers, if not an eigensolution, but a time variation is needed, and there is a set of initial conditions. The bundary conditions are usually given in Dirichlet scheme, but in case of drifts – like the pollutants in the wind – the Neumann boundary conditions can be useful as well. The wavelet based solutions usually work well periodic boundary conditions.

Then both the known source and the unknown concentration should be expanded at the basic resolution level. As the basic scale can be chosen arbitrarily, without the loss of generality, we can select 0 as the basic level, thus

$$C^{[0]}(x) = \sum_{\ell\in\Omega_0} c_{0\ell}^{[0]}\,\phi_{0\ell}(x), \tag{7}$$

$$F^{[0]}(x) = \sum_{\ell\in\Omega_0} f_{0\ell}^{[0]}\,\phi_{0\ell}(x). \tag{8}$$

The notation $c_{0\ell}^{[0]}$ and $f_{0\ell}^{[0]}$ can be introduced for the vector of the expansion coefficients of the concentration and the source, respectively. The discretization step goes as follows. If in the finite time differences counterpart of Eq. (6) the above formulae are applied and the equation is multiplied by an arbitrary scaling function, or by an arbitrary linear combination of the scaling function, it should remain valid, according to the weak formulation of the discretized equation. According to the previous statement, if the inner product $\langle\cdot,\cdot\rangle$ is introduced, and the differential operator is denoted by $\hat{D}$, the time-discretized differential equation can be written as

$$\langle \phi_{0,k},\, \hat{D}c^{[0]}\rangle = \langle \phi_{0,k},f^{[0]}\rangle \tag{9}$$

for all basis functions $\phi_{0,k}$ of the subspace V_0 – of course, only those scaling functions are to be taken into account that overlap with the solution, i.e., the elements of Ω_0. The result is a matrix equation

$$D^{[0]}_{k,\ell}c^{[0]}_{\ell} = f^{[0]}_{k} \tag{10}$$

with the source vector

$$f^{[0]}_{k} = \langle \phi_{0,k},\, f^{[0]}\rangle = \int \phi^{*}_{0,k}(x)f^{[0]}(x)dx \tag{11}$$

and the stiffness matrix

$$D^{[0]}_{k,\ell} = \langle \phi_{0,k},\, \hat{D}\phi_{0,\ell}\rangle = \int \phi^{*}_{0,k}(x)\cdot \hat{D}\phi_{0,\ell}(x)dx. \tag{12}$$

If higher resolution terms are also necessary for the precision, not only scaling function matrix elements, but wavelet matrix elements are also necessary. The expansion of the concentration and the source function at maximum resolution level M is given as

$$\begin{aligned} C^{[M]}(x) &= \sum_{\ell\in\Omega_0} c^{[M]}_{0\ell}\;\phi_{0\ell}(x) + \sum_{m=0}^{M-1}\sum_{\ell\in\Omega_m} d^{[M]}_{m\ell}\;\psi_{m\ell}(x), & (13)\\ F^{[M]}(x) &= \sum_{\ell\in\Omega_0} f^{[M]}_{0\ell}\;\phi_{0\ell}(x) + \sum_{m=0}^{M-1}\sum_{\ell\in\Omega_m} g^{[M]}_{m\ell}\;\psi_{m\ell}(x), & (14)\end{aligned}$$

resulting in the Mth level matrix equation

$$\begin{pmatrix} \langle\phi_{0,k},\,\hat{D}\phi_{0,\ell}\rangle & \langle\phi_{0,k},\,\hat{D}\psi_{n,\ell}\rangle \\ \langle\psi_{m,k},\,\hat{D}\phi_{0,\ell}\rangle & \langle\psi_{m,k},\,\hat{D}\psi_{n,\ell}\rangle \end{pmatrix}\cdot\begin{pmatrix} c^{[M]}_{0,\ell} \\ d^{[M]}_{n,\ell}\end{pmatrix} = \begin{pmatrix} f^{[M]}_{0,k} \\ g^{[M]}_{m,k}\end{pmatrix}. \tag{15}$$

Here the notation

$$D^{[M]}_{\mu,\nu} = \langle \xi_{m,k,t},\, \hat{D}\xi_{n,\ell,s}\rangle \tag{16}$$

can also be introduced with the three-element indices $\mu = \{m,k,t\}$ and $\nu = \{n,\ell,s\}$, where t and s denotes the type of the basis function ξ which can be either wavelet, or scaling function.

3.2. Matrix elements of resolution level $M = 0$

The above matrix elements $D^{[0]}_{k,\ell}$, and $D^{[M]}_{\mu,\nu}$ can be calculated using the refinement equation (1) and the wavelet's expansion equation (4). No numerical integration is necessary, if the operator $\hat{D}$ contains only the following types of terms: differentiations according to the space variable – as long as the scaling functions can be differentiated as many times as necessary –, multiplications with any positive power of the space variable, or any combination of the two [23, 26].

Substituting the refinement equation into (12), straightforward, but lengthy calculations lead to an eigenvalue equation in case of $\hat{D}$ being either a derivative operator or a product with a power of the space variable. As an example, for the first derivative's matrix element the refinement equation for the differentiated scaling function is necessary, which differs from (1) only by a factor of 2, i.e.,

$$\frac{\partial}{\partial x}\phi(x) = 2 \cdot 2^{1/2} \sum_{i=0}^{N_s} h_i \frac{\partial}{\partial x}\phi(2x - i). \tag{17}$$

After substituting (1) and (4) into the formula

$$\langle \phi_{0,0}, \frac{\partial}{\partial x}\phi_{0,\ell}\rangle = \int (\phi(x))^* \frac{\partial}{\partial x}\phi(x - \ell)dx, \tag{18}$$

changing the integral variable then the summation index, the matrix element turns into

$$\langle \phi_{0,0}, \frac{\partial}{\partial x}\phi_{0,\ell}\rangle = 2\sum_{k=0}^{N_s} \sum_{i=2\ell-k}^{N_s+2\ell-k} h_k^* h_{i+k-2\ell}\langle \phi_{0,0}, \frac{\partial}{\partial x}\phi_{0,k}\rangle, \tag{19}$$

which is clearly an eigenvalue equation for the eigenvalue 0.5 of the matrix

$${}^1M_{\ell k} = \sum_i h_i^* h_{k+i-2\ell}. \tag{20}$$

For the second derivative the eigenvectors corresponding to eigenvalue 0.25, for the third derivative, the eigenvectors corresponding to eigenvalue 0.125, etc. are giving the matrix elements, if they exist.

Note, that the matrix element (18) depends only on the difference of the indices, i.e.,

$$\langle \phi_{0,j}, \frac{\partial}{\partial x}\phi_{0,j+\ell}\rangle = \langle \phi_{0,0}, \frac{\partial}{\partial x}\phi_{0,\ell}\rangle. \tag{21}$$

For the operators that contain products with x^p, similar considerations lead to a set of iterative matrix equations, where the product with x^p depends on the product with

$x^{p-1}, x^{p-2}, \dots$ [23]

$$\langle \phi_{0,0}, x^p \cdot \phi_{0,\ell} \rangle = 2^{-p} \sum_{q=0}^{p} \binom{p}{q} \sum_{k=0}^{N_s} \sum_{i=2\ell-k}^{N_s+2\ell-k} k^q \cdot h_k^* h_{i+k-2\ell} \langle \phi_{0,0}, x^q \cdot \phi_{0,i} \rangle. \tag{22}$$

For the 0th order polynomial, as well as for the 0th derivative $\langle \phi_{0,0}, \phi_{0,\ell} \rangle = \delta_{0\ell}$ is valid. Here δ_{ab} is the usual Kronecker delta distribution: it is 1 if $a = b$ and 0 in all other cases.

In the usual advection-diffusion equation no polynomials of the spatial coordinates are present, so the discretization of Eq. (6)

$$\left(T - \mu \langle \phi_{0,k}, \frac{\partial^2}{\partial x^2} \phi_{0,\ell} \rangle + u \langle \phi_{0,k}, \frac{\partial}{\partial x} \phi_{0,\ell} \rangle - \sigma \langle \phi_{0,k}, \phi_{0,\ell} \rangle \right) c_\ell^{[0]} = f_k^{[0]}, \tag{23}$$

where T summarizes the finite difference terms arising from the time discretization, depending on the method of deriving the finite differences, the number of previous time steps, the length of the time steps and the initial conditions.

3.3. Matrix elements for higher resolutions and wavelets

In case of (16) the matrix elements can be calculated from those of $D_{k,\ell}^{[0]}$ using (1), (17) and (4) – and its derivative counterparts – respectively [23,26]. As a first step, if one of the resolution indices are non-zero in (18) or (21) the refinement equation results in

$$\langle \phi_{00}, \frac{\partial}{\partial x} \phi_{m\ell} \rangle = 2 \sum_{k=0}^{N_s} h_k^* \langle \phi_{00}, \frac{\partial}{\partial x} \phi_{m-1\ \ell-2^{m-1}k} \rangle. \tag{24}$$

If the other index is also larger than 0, then the smaller of the two resolution levels can be compensated by refinement equations, and the problem is led back to (24) as

$$\langle \phi_{nk}, \frac{\partial}{\partial x} \phi_{m\ell} \rangle = \begin{cases} 2^n \langle \phi_{00}, \frac{\partial}{\partial x} \phi_{m-n\ \ell-2^{m-n}k} \rangle, & \text{if} \quad m > n, \\ 2^m \langle \phi_{00}, \frac{\partial}{\partial x} \phi_{n-m\ k-2^{n-m}\ell} \rangle, & \text{if} \quad n > m. \end{cases} \tag{25}$$

Using the scaling function expansion of the wavelet's derivative – which is very similar to (4), just a factor of 2 is introduced –, straightforwardly results in the wavelet matrix elements

$$\langle \phi_{nk}, \frac{\partial}{\partial x} \psi_{m\ell} \rangle = \sum_{i=0}^{N_s} h_{-i+1}^* \langle \phi_{nk}, \frac{\partial}{\partial x} \phi_{m+1\ i+2\ell} \rangle, \tag{26}$$

$$\langle \psi_{nk}, \frac{\partial}{\partial x} \phi_{m\ell} \rangle = \sum_{j=0}^{N_s} h_{-j+1} \langle \phi_{n+1\ j+2k}, \frac{\partial}{\partial x} \phi_{m\ell} \rangle, \tag{27}$$

$$\langle \psi_{nk}, \frac{\partial}{\partial x} \psi_{m\ell} \rangle = \sum_{i=0}^{N_s} \sum_{j=0}^{N_s} h_{-j+1} h_{-i+1}^* \langle \phi_{n+1\ j+2k}, \frac{\partial}{\partial x} \phi_{m+1\ i+2\ell} \rangle. \tag{28}$$

3.4. Matrix elements for discontinuous functions

In wavelet-based calculations the borders and the discontinuities cause always a problem, as the wavelets have finite support and can usually approximate continuous functions. Having a step-function in the system, like in case of the urban traffic the buildings, streets, and other obstacles necessitates numerical integration, which takes long time and gives not very high precision. For example, Table 1 contains the matrix element of a simple rectangular function

$$R_k(x) = \begin{cases} 1 & \text{if} \quad x \in [k,k+1), \\ 0 & \text{if} \quad x \in (-\infty,k) \cup [k+1,\infty). \end{cases} \tag{29}$$

for 2^{10}, 2^{15}, 2^{20} and 2^{23} points in the interval. The last row contains also the duration of the calculations, as an average of more runs on a desktop computer with dual core 2.2 GHz processor and 4 GB RAM, in Matlab.

Only the matrix elements ${}^kY_\ell = \langle \phi_{0,0}, R_k \phi_{0,\ell} \rangle$ are calculated, with $k \in \{0,1,\ldots,N_s - 1\}$ as these rectangular functions overlap with the support of the scaling function $\phi_{0,0}$. Also those matrix elements are zero, where the two basis functions do not overlap – i.e., where $|\ell| > N_s - 1$ is valid –, and where the rectangular function R_k does not overlap with $\phi_{0,\ell}$ – i.e., where $\ell < k - N_s + 2$ or $\ell > k$. All the other matrix elements can be derived by simple shifts, as $\langle \phi_{0,i}, R_{k+i} \phi_{0,\ell+i} \rangle = \langle \phi_{0,0}, R_k \phi_{0,\ell} \rangle$.

As the sum of the above matrix elements give the $\langle \phi_{0,0}, \phi_{0,\ell} \rangle$ integrals, the following sum rules have to be fulfilled

$$\begin{aligned}
\langle \phi_{0,0}, \phi_{0,-N_s+1} \rangle &= {}^0Y_{-N_s+1}, \\
\langle \phi_{0,0}, \phi_{0,-N_s+2} \rangle &= {}^0Y_{-N_s+2} + {}^1Y_{-N_s+2}, \\
\langle \phi_{0,0}, \phi_{0,-N_s+3} \rangle &= {}^0Y_{-N_s+3} + {}^1Y_{-N_s+3} + {}^2Y_{-N_s+3}, \\
&\vdots \\
\langle \phi_{0,0}, \phi_{0,0} \rangle &= {}^0Y_0 + {}^1Y_0 + {}^2Y_0 + \ldots + {}^{N_s-2}Y_0 + {}^{N_s-1}Y_0, \\
\langle \phi_{0,0}, \phi_{0,1} \rangle &= {}^1Y_1 + {}^2Y_1 + \ldots + {}^{N_s-2}Y_1 + {}^{N_s-1}Y_1, \\
&\vdots \\
\langle \phi_{0,0}, \phi_{0,N_s-2} \rangle &= {}^{N_s-2}Y_{N_s-2} + {}^{N_s-1}Y_{N_s-2}, \\
\langle \phi_{0,0}, \phi_{0,N_s-1} \rangle &= {}^{N_s-1}Y_{N_s-1},
\end{aligned} \tag{30}$$

Note, that all of the values on the left hand side are zero, except for $\langle \phi_{0,0}, \phi_{0,0} \rangle$, which is 1.

It can be seen from Table 1, that the precision increases slowly and the duration of the calculation increases very rapidly. Alternatively to the numerical integration I suggest

Table 1. Matrix elements ${}^{k}Y_{\ell}$ with their calculation time t of the rectangular functions $R_k(x)$ as a result of numerical integration with various precisions ε. Daubechies-6 scaling functions are used with $N_s = 5$.

ε	2^{10}	2^{15}	2^{20}	2^{23}
${}^{0}Y_{-4}$	−0.000000000015124	−0.000000000000001	0	0
${}^{0}Y_{-3}$	−0.000443983002980	−0.000441411389722	−0.000441330880502	−0.000441328607930
${}^{0}Y_{-2}$	−0.018852903241268	−0.018795005543038	−0.018793194141192	−0.018793143011087
${}^{0}Y_{-1}$	0.123570297966252	0.123335808619037	0.123328472377250	0.123328265298497
${}^{0}Y_{0}$	0.495840194948971	0.496622550356504	0.496647009103327	0.496647699480092
${}^{1}Y_{-3}$	0.000443982821493	0.000441411389712	0.000441330880502	0.000441328607930
${}^{1}Y_{-2}$	0.018988089017918	0.018929419189328	0.018927583639190	0.018927531827427
${}^{1}Y_{-1}$	−0.117900092490012	−0.117682953517957	−0.117676160594197	−0.117675968851946
${}^{1}Y_{0}$	0.471683782870471	0.470971706656950	0.470949448367932	0.470948820104503
${}^{1}Y_{1}$	0.123570297966252	0.123335808619037	0.123328472377250	0.123328265298497
${}^{2}Y_{-2}$	−0.000135184385099	−0.000134413646213	−0.000134389497999	−0.000134388816340
${}^{2}Y_{-1}$	−0.005702899144805	−0.005685354773212	−0.005684805492510	−0.005684789987707
${}^{2}Y_{0}$	0.031131664896956	0.031065665678285	0.031063599371681	0.031063541045095
${}^{2}Y_{1}$	−0.117900092490012	−0.117682953517957	−0.117676160594197	−0.117675968851946
${}^{2}Y_{2}$	−0.018852903241268	−0.018795005543038	−0.018793194141192	−0.018793143011087
${}^{3}Y_{-1}$	0.000032689977738	0.000032499671929	0.000032493709459	0.000032493541149
${}^{3}Y_{0}$	0.001342875637914	0.001338599123643	0.001338465237171	0.001338461457897
${}^{3}Y_{1}$	−0.005702899144805	−0.005685354773212	−0.005684805492510	−0.005684789987707
${}^{3}Y_{2}$	0.018988089017918	0.018929419189328	0.018927583639190	0.018927531827427
${}^{3}Y_{3}$	−0.000443983002980	−0.000441411389722	−0.000441330880502	−0.000441328607930
${}^{4}Y_{0}$	0.000001486637458	0.000001478184885	0.000001477919879	0.000001477912398
${}^{4}Y_{1}$	0.000032689977738	0.000032499671929	0.000032493709459	0.000032493541149
${}^{4}Y_{2}$	−0.000135184385099	−0.000134413646213	−0.000134389497999	−0.000134388816340
${}^{4}Y_{3}$	0.000443982821493	0.000441411389712	0.000441330880502	0.000441328607930
${}^{4}Y_{4}$	−0.000000000015124	−0.000000000000001	0	0
t (s)	0.1038	2.278	73.59	2717

another calculation method similar to the ones presented in the previous two subsections. If the matrix element is calculated according to its definition

$$ {}^kY_\ell = \langle \phi_{0,0},\, R_k(x)\phi_{0,\ell}\rangle = \int_k^{k^1} \phi^*(x)\phi(x-\ell)dx \tag{31}$$

and the refinement equation (1) is substituted into both $\phi^*_{0,0}$ and $\phi_{0,\ell}$ after changing of the integration variable one arrives at

$$ {}^kY_\ell = 2\sum_{i_1=0}^{N_s}\sum_{i_2=1}^{N_s} h^*_{i_1}h_{i_2}\left({}^{2\ell+i_2-i_1}Y_{2k-i_1} +^{2\ell+i_2-i_1} Y_{2k-i_1+1}\right). \tag{32}$$

The above equation also leads on an eigenvalue equation of the matrix

$$ M^{[0]}_{k,\ell} = \begin{pmatrix} B_1 & B_0 & 0 & 0 & \dots & 0 \\ B_3 & B_2 & B_1 & B_0 & \dots & 0 \\ B_5 & B_4 & B_3 & B_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & B_{N_s} & B_{N_s-1} \end{pmatrix}, \tag{33}$$

with the blocks $B_z =$

$$ \begin{pmatrix} \begin{matrix} h_{N_s}h^*_z + \\ h_{N_s}h^*_{z-1} \end{matrix} & h_{N_s-1}h^*_{z-1} & 0 & 0 & 0 & \dots & 0 \\ \begin{matrix} h_{N_s-2}h^*_z + \\ h_{N_s-3}h^*_{z-1} \end{matrix} & \begin{matrix} h_{N_s-1}h^*_z + \\ h_{N_s-2}h^*_{z-1} \end{matrix} & \begin{matrix} h_{N_s}h^*_z + \\ h_{N_s-1}h^*_{z-1} \end{matrix} & h_{N_s}h^*_{z-1} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & h_0h^*_z & \begin{matrix} h_1h^*_z + \\ h_0h^*_{z-1} \end{matrix} & \begin{matrix} h_2h^*_z + \\ h_1h^*_{z-1} \end{matrix} & \begin{matrix} h_2h^*_z + \\ h_1h^*_{z-1} \end{matrix} \\ 0 & \dots & 0 & 0 & 0 & h_0h^*_z & \begin{matrix} h_1h^*_z + \\ h_0h^*_{z-1} \end{matrix} \end{pmatrix}. \tag{34}$$

Matrix $M^{[0]}_{k,\ell}$ has an eigenvector corresponding to eigenvalue 1, and this eigenvector can be normalized due to (30), as $\langle\phi_{0,0},\,\phi_{0,i}\rangle = \delta_{i0}$. Numerical checks were carried out for various scaling functions. In case of Daubeches-6 basis set ($N_s = 5$), the normalization condition is

$$ 1 = \langle\phi_{0,0},\,\phi_{0,0}\rangle =^0 Y_0 +^1 Y_0 +^2 Y_0 +^3 Y_0 +^4 Y_0, \tag{35}$$

and the resulting matrix elements are listed in Table 2 The differences between the quantities ${}^kY_\ell$ numerically integrated and calculated with the eigenvalue method (32) are plotted

Table 2. Matrix elements ${}^{k}Y_{\ell}$ with their calculation time t of the rectangular functions $R_k(x)$ as a result of the eigenvalue-based calculation. Daubechies-6 scaling functions are used with $N_s = 5$.

${}^{0}Y_{-4}$	-0.000000000000000
${}^{0}Y_{-3}$	-0.000441328283277
${}^{0}Y_{-2}$	-0.018793135706778
${}^{0}Y_{-1}$	0.123328235715790
${}^{0}Y_{0}$	0.496647798105380
${}^{1}Y_{-3}$	0.000441328283277
${}^{1}Y_{-2}$	0.018927524425738
${}^{1}Y_{-1}$	-0.117675941460163
${}^{1}Y_{0}$	0.470948730352582
${}^{1}Y_{1}$	0.123328235715790
${}^{2}Y_{-2}$	-0.000134388718960
${}^{2}Y_{-1}$	-0.005684787772732
${}^{2}Y_{0}$	0.031063532712708
${}^{2}Y_{1}$	-0.117675941460163
${}^{2}Y_{2}$	-0.018793135706778
${}^{3}Y_{-1}$	0.000032493517105
${}^{3}Y_{0}$	0.001338460918000
${}^{3}Y_{1}$	-0.005684787772732
${}^{3}Y_{2}$	0.018927524425738
${}^{3}Y_{3}$	-0.000441328283277
${}^{4}Y_{0}$	0.000001477911329
${}^{4}Y_{1}$	0.000032493517105
${}^{4}Y_{2}$	-0.000134388718960
${}^{4}Y_{3}$	0.000441328283277
${}^{4}Y_{4}$	-0.000000000000000
t (s)	0.002132

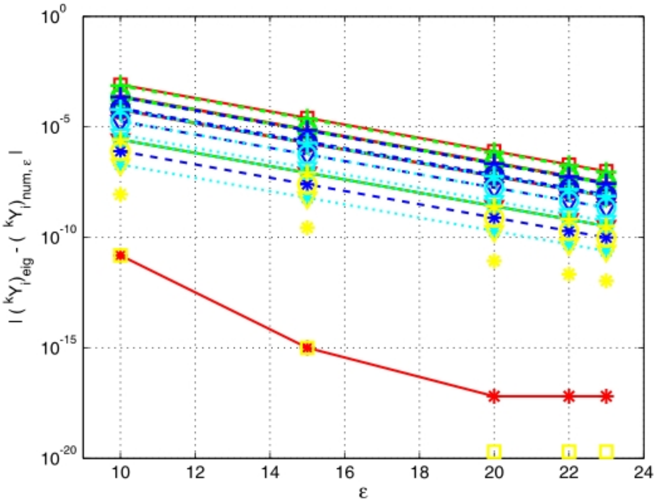

Figure 1. Difference of the numerically integrated values ${}^kY_\ell$ from their counterparts calculated with an eigenvalue equation of matrix $M^{[0]}_{k,\ell}$ as a function of the grid points in a unit interval 2^ε. Red color with continuous line means $k = 0$, green with dash-dot line mean $k = 1$, blue with dashed line $k = 2$, cyan with dotted line $k = 3$ and the color yellow and markers without line means $k = 4$. The markers $*$, $\bigtriangledown$, $\circ$, $+$, and $\square$ stand for the indices $\ell = k - 4$, $k - 3$, $k - 2$, $k - 2$ and k respectively.

in Figure 1. Also the calculation time is shown on Figure 2, where an approximate power law behavior can be seen as expected for the numerical integrations.

In case of derivatives, similarly to the continuous case described in (19), each differentiation introduces only a factor of 2 to the matrix $M^{[0]}_{k,\ell}$, thus the eigenvector corresponding to the eigenvalues 0.5, 0.25, etc. should be found for the matrix elements with the first, second, etc. derivatives respectively.

The transformation to higher resolution levels goes similarly to (24–28), except, that with each use of the refinement equation, the number of the elements $\langle \phi_{m,i}, R_k(x)\phi_{n,\ell}\rangle$ doubles, so for higher resolution levels an exponentially increasing number of lower resolution level integrals should be taken into account.

4. Summary

In wavelet-based modeling the concentration changes of nitrogen oxides and other air pollutants in urban environment discontinuities are arising around the obstacles that the

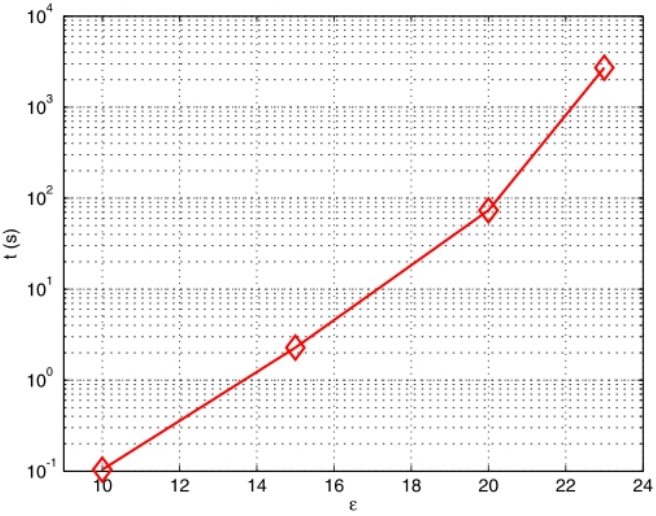

Figure 2. Computation time of the numerical integrations as a function of of the grid points in a unit interval 2^ε.

gases can not penetrate. The discretization of these discontinuities can be carried out not only by numerical integration, but also by a much quicker and more economic calculation – derived in this paper – based on the eigenvalue equation of a matrix generated from the coefficients of the refinement equation, the basic equation of the wavelet analysis. The size of the matrix is N_s^2, with N_s being the length of the support of the wavelets.

The calculated matrix elements are system independent, they depend only on the type of the wavelets used, thus in later calculations they can be loaded from a database, but for calculating this database as precisely as possible, the method developed in this article, based on the results of [23, 26], is necessary.

Acknowledgement

The publishing of this paper was supported by Hungary's New Széchenyi Plan, co-financed by the European Social Fund, grant ID. TÁMOP-4.2.2.C-11/1/KONV-2012-0012 "Smarter Transport" Infocommuncations support of cooperative transport systems.

References

[1] Daubechies I: Ten Lectures on Wavelets. CBMS-NSF regional conference series in applied mathematics 61, SIAM, Philadelphia, 1992.

[2] Chui CK: An Introduction to Wavelets. Academic Press, San Diego, 1992.

[3] Crooks J, Isakov V: A wavelet-based approach to blending observations with deterministic computer models to resolve the intraurban air pollution field. J. Air & Waste Management Assoc., Vol. 63, p. 1369-1385, 2013. DOI:10.1080/10962247.2012.758061

[4] Lobdell DT, Isakov V, Baxter L, Touma JS, Smuts MB, Özkaynak H: Feasibility of assessing public health impacts of air pollution reduction programs on a local scale: New Haven case study. Environ. Health Perspect., Vol. 119, pp. 487-493, 2011. DOI:10.1289/ehp.1002636

[5] Metzger K, Klein M, Flanders W, Peel J, Mulholland J, Langberg J, Tolbert P: Ambient air pollution and cardiac arrhythmias in patients with implantable defibrillators. Epidemiology Vol. 18, pp. 585-592, 2007. DOI:10.1097/EDE.0b013e318124ff0e

[6] Appel K, Bhave P, Gilliland A, Sarwar G, Roselle S: Evaluation of the community multiscale air quality (CMAQ) model version 4.5: Sensitivities impacting model performance; part II-particulate matter. Atmos. Environ. Vol. 42, pp. 6057-6066, 2008. DOI:10.1016/j.atmosenv.2008.03.036

[7] Cimorelli A, Perry S, Venkatram A, Weil J, Paine R, Wilson R, Lee R, Peters W, Brode WR: AERMOD: A dispersion model for industrial source applications. Part I: General model formulation and boundary layer characterization. J. Appl. Meteorol. Vol. 44, pp. 682-693, 2004. DOI:10.1175/JAM2227.1

[8] Gelfand A, Sahu S: Combining Monitoring Data and Computer Model Output in Assessing Environmental Exposure. Oxford University Press, Oxford 2009.

[9] Zidek J, Le N, Liu Z: Combining data and simulated data for spacetime fields: Application to ozone. Environ. Ecol. Stat., Vol. 19, pp. 37-56, 2012. DOI:10.1007/s10651-011-0172-1

[10] Christopoulos Ch, Skodras A, Ebrahimi T: The JPEG2000 Still Image Coding System: An Overview. IEEE Trans. Consumer Electronics, Vol. 46, pp. 1103-1127, 2000. DOI:10.1109/30.920468

[11] Kiely A, Klimesh M: The ICER Progressive Wavelet Image Compressor. IPN Progress Report 42-155, November 15, 2003.

[12] Urban K: Wavelet Methods for Elliptic Partial Differential Equations. Oxford University Press, Oxford, 2009.

[13] Dahmen W: Wavelets and Multiscale Methods for Operator Equations. Acta Numerica Vol. 6, p. 55, 1997. DOI:10.1017/s0962492900002713

[14] Nasif H, Omori R, Suzuki A, Naguib M, Nagy M: Wavelet-Based Algorithms for Solving Neutron Diffusion Equations. J. Nuclear Sci. Techn., Vol. 38, pp. 161-173, 2001. DOI:10.1080/18811248.2001.9715018

[15] Galerkin BG: On electrical circuits for the approximate solution of the Laplace equation (in Russian). Vestnik Inzh. Vol. 1, p. 897, 1915.

[16] Nagy Sz, Pipek J: Multiresolution Analysis of Density Operators, Electron Density, and Energy Functionals. Int. J. Quantum Chem., Vol. 84, p. 523, 2001. DOI:10.1002/qua.1406

[17] Pipek J, Nagy Sz: Refinement trajectory and determination of eigenstates by a wavelet based adaptive method. J. Chem. Phys. Vol. 125, 174107, 2006. DOI:10.1063/1.2363368

[18] Goedecker S, Ivanov O: Linear scaling solution of the Coulomb problem using wavelets. Solid State Commun., Vol. 105, p. 665-669, 1998.

[19] Yang Sh, Ni G, Ho SL, Machado JM, Rahman MA, Wong HC: Wavelet-Galerkin Method for Computations of Electromagnetic Fields – Computation of Connection Coefficients. IEEE Trans. Magn., Vol. 36, p. 644, 2000. DOI:10.1109/20.877532

[20] Rickard Y: An efficient wavelet-based solution of electromagnetic field problems. Appl. Numer. Math., Vol. 58, p. 472, 2008. DOI:10.1016/j.apnum.2007.01.020

[21] Nagy Sz, Pipek J: A wavelet-based adaptive method for determining eigenstates of electronic systems. Theor. Chim. Acta, Vol. 125, p. 471-479, 2010. DOI:10.1007/s00214-009-0653-6

[22] Dahmen W: Wavelet Methods for PDEs – Some Recent Developments. J. Comput. Appl. Math., Vol. 128, p. 123, 2001. DOI:10.1016/S0377-0427(00)00511-2

[23] Pipek J, Nagy Sz: An Economic Prediction of Refinement Coefficients in Wavelet-Based Adaptive Methods for Electron Structure Calculations. J. Comput. Chem., Vol. 34, p. 460-465, 2013. DOI:10.1002/jcc.23154

[24] Hillel R, Joseph A: Environmental fluid mechanics. Marcel Dekker Inc., New York Basel, 2001. DOI:10.1201/9780203908495

[25] Dang Q, Matthias E: Adequate Numerical Solution of Air Pollution Problems by positive Difference Schemes on unbounded Domains. Math. Computer Modelling, Vol. 44, pp. 834-856, 2006. DOI:10.1016/j.mcm.2006.02.016

[26] Dahmen W, Micchelli CA: Using the refinement equation for evaluating integrals of wavelets. SIAM J. Numer. Anal., Vol. 30, p. 507, 1993. DOI:10.1137/0730024