 CMC, vol.15, no.3, pp.173-198, 2010

Space-Time Adaptive Fup Multi-Resolution Approach for Boundary-Initial Value Problems

Hrvoje Gotovac[1], Vedrana Kozulić[2] and Blaž Gotovac[1]

Abstract: The space-time Adaptive Fup Collocation Method (AFCM) for solving boundary-initial value problems is presented. To solve the one-dimensional initial boundary value problem, we convert the problem into a two-dimensional boundary value problem. This quasi-boundary value problem is then solved simultaneously in the space-time domain with a collocation technique and by using atomic Fup basis functions. The proposed method is a generally meshless methodology because it requires only the addition of collocation points and basis functions over the domain, instead of the classical domain discretization and numerical integration. The grid is adapted progressively by setting the threshold as a direct measure of the solution accuracy at a given resolution level. At higher resolution levels, collocation points are only added in the space-time sub-domains where the solution correction is greater than the prescribed threshold. In contrast to the classical time-stepping schemes, in which globally accumulated errors can arise and which are not easily adapted to multiple time steps, the space-time AFCM covers all space and time multiple scales, while global error is strictly controlled in time by an *a priori* threshold.

Keywords: Fup basis functions, collocation, partial differential equations, adaptive grid, meshless method, prescribed accuracy of time integration.

1 Introduction

Classical solutions of time-dependent partial differential equations (PDEs) are reduced to time integration of ordinary differential equations (ODEs) with respect to spatial discretization and corresponding boundary conditions through the common method of lines (MOL) [Ascher and Petzold (1998); Hairer and Wanner (1996)]. Time integration of ODEs or time-dependent PDEs that require resolution at the

[1] University of Split, Faculty of Civil Engineering and Architecture, Split, Croatia
[2] Corresponding author: Vedrana Kozulić, University of Split, Faculty of Civil Engineering and Architecture, Split, Croatia, E-mail: vedrana.kozulic@gradst.hr

fastest time scales of the system can be very costly if the system exhibits multiple time scales of different magnitudes. If different time scales are localized to different components that correspond to localizations in space for a PDE, efficient time integration thus requires the use of different time steps for different components. If the solution is intermittent in both space and time, one adapts the spatial mesh to the solution at a fixed time and uses an adjustable time step to control local errors in time. This approach enforces use of the same time step for all spatial locations, which is clearly not optimal for problems that are simultaneously intermittent in both space and time. However, global errors accumulate in time, even as spatial errors are controlled by adaptive approximation. There is no guarantee that temporal truncation errors will not accumulate over time and eventually exceed the desired error tolerance. In fact, the common explicit and implicit time-marching schemes provide no control over global errors in time [Alam, Kevlahan, and Vasilyev (2006)].

Common numerical techniques, such as finite difference, finite element and finite volume methods, are often used to solve problems described by PDEs through the MOL. However, these conventional methods are associated with many numerical difficulties, such as the usage of classical numerical integration, derivative discontinuities or non-efficient adaptive procedures. Therefore, we focus in this paper on meshless methods that naturally eliminate the above-mentioned problems.

In recent years, a number of meshless methods have been developed for solving PDEs with the classical MOL approach. Among others, prominent meshless discretization techniques include the Meshless Local Petrov-Galerkin (MLPG) Method. Various MLPG methods were compared and shown to be promising contenders for the Finite Element Method [Atluri and Shen (2002)]. Remarkable successes of the MLPG method have been reported in solving the convection-diffusion problems [Lin and Atluri (2000)], elasto-static problems [Atluri, Han, and Rajendran (2004)], elasto-dynamic problems [Han and Atluri (2004)] and for atomistic/continuum simulation [Shen and Atluri (2005)]. The convergence, accuracy, numerical stability and computational efficiency of four various formulations for solution of boundary value problems where the meshless point interpolation method was employed based on various basis functions with using both a weak form and strong form (by collocation of the governing PDE) can be found in [Sladek, Sladek and Zhang (2006)].

Furthermore, meshless methods that are based on radial basis functions (RBFs) have increasingly attracted attention from researchers. The idea of using RBFs for solving PDEs was first proposed in [Kansa (1990)], where the collocation method was used to solve parabolic, hyperbolic and elliptic PDEs. The RBF-based numerical methods represent one of the key directions in meshless methods research for

fluids [Amaziane, Naji and Ouazar (2004)], solids [Mai-Duy, Khennane and Tran-Cong (2007)] and moving boundaries [La Rocca, Power, La Rocca and Morale (2005)]. Applications of RBFs to solution of the Navier-Stokes equations were reported in [Mai-Duy (2004)], the numerical simulation of two-phase flow in porous media in [Iske and Käser (2005)], and dealing with transport phenomena in [Šarler (2005)]. The heat transfer model was solved [Lorbiecka, Vertnik, Gjerkeš, Manojlovič, Senčič, Cesar and Šarler (2009)] by the meshless technique by using local collocation with radial basis functions. Cho, Golberg, Muleshkov and Li (2004) presented a meshless approach to numerically solving a class of second order time-dependent PDEs based on a combination of the method of particular solutions and the Trefftz method.

A number of papers have been published in the last several years that describe an adaptive strategy in RBF solutions of PDEs. Sarra (2005) developed and successfully applied an adaptive RBF method to the solution of nearly-singular and time-dependent Burger's and Advection equations in 1D. A dynamic adaptive scheme was proposed by Wu (2004) for time-dependent PDEs. Behrens, Iske, and Käser (2003) successfully applied an adaptive algorithm with local TPS-RBFs interpolation to linear evolutionary PDEs. The method uses a local interpolation to evaluate an error indicator and to detect regions where the approximation requires more accuracy. Many of the adaptive strategies mentioned above are driven by a front-tracking scheme that utilizes a posterior error indicator to detect regions that require refinement; see Lee, Im, Jung, Kim and Kim (2007) and Iske and Käser (2005).

The concept of wavelet analysis was introduced in applied mathematics in the late 1980s, and interest has grown recently in developing wavelet-based numerical algorithms for both uniform and adaptive node-distribution schemes for the solution of PDEs. Recently, multi-resolution wavelet analysis has been developed as a potentially adaptive approach to the construction of optimum adaptive node distribution in nearly-singular problems; see Cruz, Mendes, and Magalhaes (2001); Mehra and Kevlahan (2008) and Vasilyev and Kevlahan (2005). Libre, Emdadi, Kansa, Shekarchi, and Rahimian (2008) developed a modified adaptive wavelet scheme to solve nearly-singular potential PDEs. Recently, there have been many attempts to develop new adaptive procedures focused upon the use of, among others, adaptive wavelet collocation methods [Bertoluzza (1996); Bertoluzza and Naldi (1996); Cruz, Mendes, and Magalhaes (2001); Hesthaven and Jameson (1998); Holmstrom (1999); Vasilyev and Paolucci (1997); Alam, Kevlahan, and Vasilyev (2006)].

Aside from wavelets and splines, there is a relatively lesser-known class of atomic or R_{bf} basis functions (Rvachev's basis functions) [Rvačev and Rvačev (1971); Rvačev (1982)]. Atomic functions are classified in between classic polynomials and spline functions. However, in practice, their application as basis functions

is closer to those of splines or wavelets. In this paper we use Fup basis functions, which are one type of atomic basis functions, recent reviewed by Kolodiazhny and Rvačev (2007). Gotovac and Kozulić (1999) systemized the existing knowledge on atomic functions and presented the transformation of basis functions into a numerically-applicable form. The application of Fup basis functions has been demonstrated in signal processing [Kravchenko, Basarab, and Perez-Meana (2001)], initial value problems [Gotovac and Kozulić (2002)], and collocation methods for boundary value problems [Kozulić and Gotovac (2000); Gotovac, Andričevič, and Gotovac (2007)].

The main feature of the Fup basis functions, as well as wavelets, is the spatial multiresolution of signals and functions, resolving all the spatial frequencies and scales that are obtained by Fup collocation transform (FCT). Recently, the Adaptive Fup Collocation Method (AFCM), with an adaptive spatial algorithm but with a classical time-marching algorithm, was published [Gotovac, Andričevič, and Gotovac (2007); Kozulić, Gotovac, and Gotovac (2007)].

The second approach involves solving PDEs simultaneously in the space-time domain. This simultaneous approach controls global time integration errors, but involves time as an additional coordinate and thus increases the dimension and computational burden of the problem. There are adaptive space-time finite element methods [Cao and Demeler (2005)], finite difference methods [Wackers and Koren (2003)] and wavelet collocation methods [Alam, Kevlahan, and Vasilyev (2006)].

In this paper, we present a new space-time AFCM with resolution of all space and time multiple scales. Essentially, AFCM solves PDEs and corresponding boundary conditions in the same way as an FCT approximates a function by different resolution levels. All existing algorithms that use wavelets and splines [e.g., Vasilyev and Bowman (2000) and Wang, Keast, and Muir (2004)], as well as a recent form of the AFCM [Gotovac, Andričevič, and Gotovac (2007)], use localized basis functions only to obtain an efficient adaptive strategy, but the PDE itself is solved by classic time-marching on a non-uniform adaptive grid (including all levels). In this paper, we present a novel form of the AFCM, with Fup basis functions at each level, by using a collocation framework in the space-time domain. Each non-zero level solves only the residuals of the PDEs from all previous levels and gives particular solution corrections. Adaptive criteria add new collocation points in the next level only in the space-time zones where solution corrections are greater than the prescribed threshold. Thus, in the case of a moving steep front, one can track its position and increase the local resolution of the grid by adding higher-resolution basis functions in that region.

The rest of this paper is organized as follows. A short description of the main features of the Fup basis functions is presented in section 2. The procedure of

the Fup collocation transform is explained in section 3. Section 4 describes how to extend the Adaptive Fup Collocation Method for solving boundary-initial value problems. The efficiency of the proposed meshless method is illustrated by two numerical examples in section 5: the Burgers equation and an advection-dispersion problem. Finally, we give a summary and conclusions in section 6.

2 Fup basis functions

Atomic basis functions are infinitely-differentiable functions with compact support [Gotovac and Kozulić (1999), Rvačev and Rvačev (1971)]. Atomic functions $y(.)$ are defined as solutions of differential functional equations of the following type:

$$Ly(x) = \lambda \sum_{k=1}^{M} C_k y(ax - b_k) \tag{1}$$

where L is a linear differential operator with constant coefficients, λ is a nonzero scalar, C_k are coefficients of the linear combination, $a > 1$ is a parameter that defines the length of the compact support, and b_k are coefficients that determine displacements of the basis functions. Rvačev and Rvačev (1971), in their pioneering work, called these basis functions "atomic" because they span the vector spaces of all three fundamental functions in mathematics: algebraic, exponential and trigonometric polynomials. Also, atomic functions can be divided into an infinite number of smaller pieces that maintain all their properties, implying a so-called "atomic structure."

The simplest function, which is the most-studied of the atomic basis functions, is the *up(x)* function. The function *up(x)* is a smooth function with compact support over [-1,1], which is obtained as a solution of a differential functional equation

$$up'(x) = 2up(2x+1) - 2up(2x-1) \tag{2}$$

with the normalized condition $\int_{-\infty}^{\infty} up(x)dx = \int_{-1}^{1} up(x)dx = 1$. The function *up(x)* can be expressed as an inverse Fourier transform:

$$up(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{itx} \prod_{j=1}^{\infty} \left(\frac{\sin(t2^{-j})}{t2^{-j}} \right) dt. \tag{3}$$

Since Eq. (3) represents an exact but mathematically-intractable expression, Rvačev (1982) and Gotovac and Kozulić (1999) provided a numerically more-adequate expression for calculating the function *up(x)*:

$$up(x) = 1 - \sum_{k=1}^{\infty} (-1)^{1+p_1+\ldots+p_k} p_k \sum_{j=0}^{k} C_{jk}(x-0, p_1 \ldots p_k)^j \tag{4}$$

where coefficients C_{jk} are rational numbers determined according to the following expression:

$$C_{jk} = \frac{1}{j!} 2^{j(j+1)/2} up(-1+2^{-(k-j)}); \quad j = 0,1,...,k; \quad k = 1,2,...,\infty. \tag{5}$$

Calculation of the $up(-1+2^{-r})$; $r \in [0,\infty]$ in binary-rational points (Eq. (5)), as well as all details regarding the calculation of the function *up(x)* values, are provided in [Gotovac and Kozulić (1999)] and [Gotovac and Kozulić (2002)]. The argument $(x-0, p_1...p_k)$ in Eq. (4) is the difference between the real value of coordinate x and its binary form in k bits, where $p_1...p_k$ are digits, 0 or 1, of the binary representation of the x coordinate. Therefore, the accuracy of the x coordinate computation, and, thus the accuracy of the *up(x)* function at an arbitrary point, depends on machine accuracy.

From Eq. (2), it can be seen that the derivatives of the *up(x)* function can be calculated simply from the values of the function itself. The general expression for the derivative of the mth degree is

$$up^{(m)}(x) = 2^{C_{m+1}^2} \sum_{k=1}^{2^m} \delta_k up(2^m x + 2^m + 1 - 2k), \quad m \in N \tag{6}$$

where $C_{m+1}^2 = m(m+1)/2$ is the binomial coefficient and δ_k are the coefficients with value ± 1, according to the recursive formulas $\delta_{2k-1} = \delta_k$, $\delta_{2k} = -\delta_k$, $k \in N$, $\delta_1 = 1$. It can be observed that the derivatives consist of the *up(x)* function compressed to an interval of 2^{-m+1} length, with ordinates extended by the $2^{C_{m+1}^2}$ factor.

The $Fup_n(x)$ function satisfies the following differential-functional equation:

$$Fup_n'(x) = 2 \sum_{k=0}^{n+2} \left(C_{n+1}^k - C_{n+1}^{k-1} \right) Fup_n(x) \left(2x - 2^{-n-1}k + 2^{-n-2}(n+2) \right) \tag{7}$$

where n is the *Fup* order. Index n also denotes the highest degree of the polynomial that can be expressed exactly as a linear combination of $n+2$ $Fup_n(x)$ basis functions, uniformly displaced by a characteristic interval 2^{-n}.

For $n = 0, Fup_0(x) = up(x)$, since $Fup_n(x)$ and its derivatives can be calculated using a linear combination of displaced *up(x)* functions instead of using their Fourier transforms:

$$Fup_n(x) = \sum_{k=0}^{\infty} C_k(n) up\left(x - 1 - \frac{k}{2^n} + \frac{n+2}{2^{n+1}} \right) \tag{8}$$

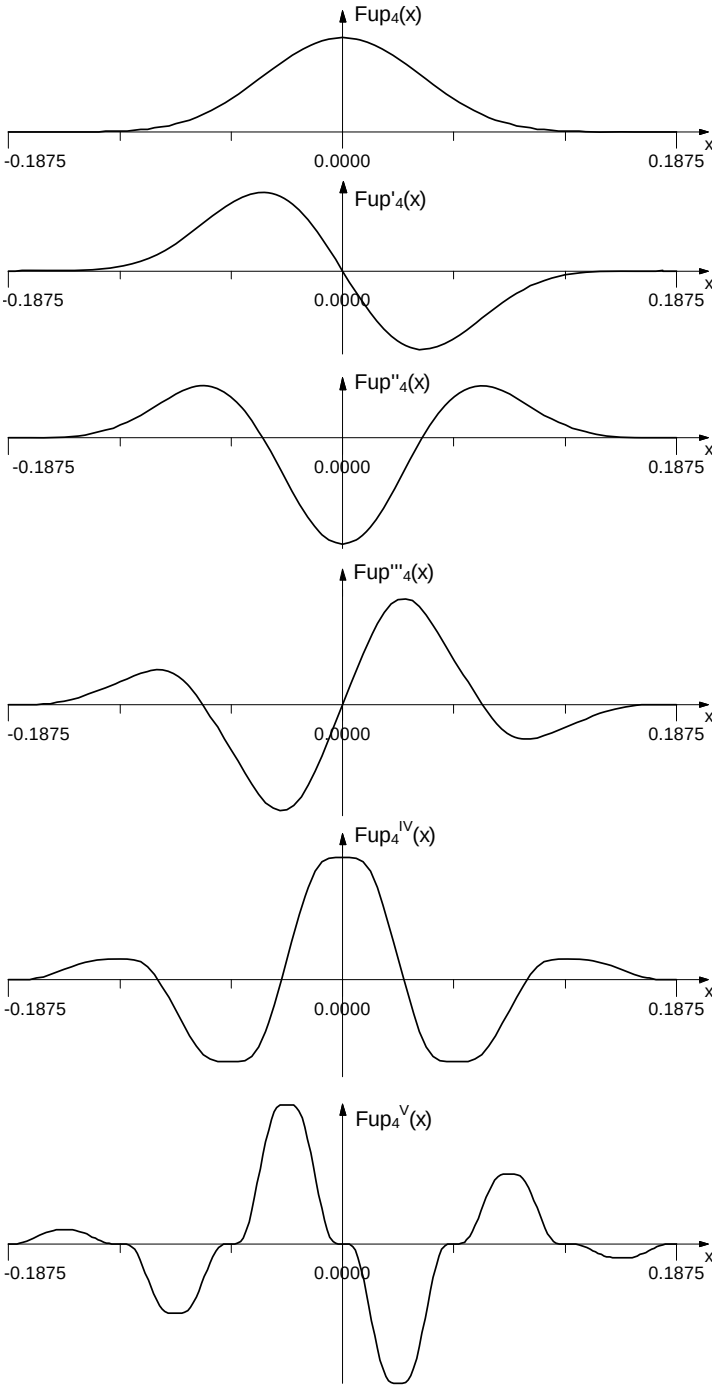

Figure 1: Function $Fup_4(x)$ and its first five derivatives

where $C_0(n) = 2^{C_{n+1}^2} = 2^{n(n+1)/2}$. In turn, $C_k(n) = C_0(n) \cdot C'_k(n)$, where a recursive formula is used for calculating auxiliary coefficients $C'_k(n)$:

$$C'_0(n) = 1, \text{ when k} = 0; \text{ i.e., when k} > 0$$
$$C'_k(n) = (-1)^k C_{n+1}^k - \sum_{j=1}^{\min\{k;2^{n+1}-1\}} C'_{k-j}(n) \cdot \delta_{j+1} \tag{9}$$

The $Fup_n(x)$ is defined over the compact support $[-(n+2)2^{-n-1}; (n+2)2^{-n-1}]$. Fig. 1 shows the $Fup_4(x)$ function and its first five derivatives, which are used in this paper.

Basis functions for numerical analyses of two-dimensional problems are obtained from the Cartesian product of the two one-dimensional *Fup* basis functions defined for each direction:

$$Fup_n(x,y) = Fup_n(x) \cdot Fup_n(y). \tag{10}$$

Calculations of all required derivatives of the function $Fup_n(x,y)$ can be written in an analogue form. Fig. 2 gives an axonometric presentations of basis function $Fup_4(x,y)$ and its partial derivatives.

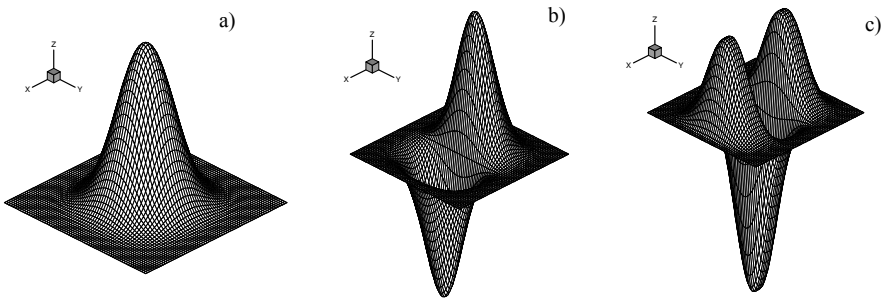

Figure 2: a) $Fup_4(x,y)$; b) $\frac{\partial Fup_4(x,y)}{\partial x}$; c) $\frac{\partial^2 Fup_4(x,y)}{\partial x^2}$

3 Fup collocation transform

The Fup collocation transform (FCT), developed by Gotovac, Andričevič, and Gotovac (2007), is an efficient numerical tool for describing various types of signals and functions using linear combinations of the Fup basis functions. It is a discrete transform, similar to the classical discrete Fourier transform. However, a natural

advantage of the Fup collocation transform is that it is based on the chosen basis functions, with compact support (Fig.1). In other words, the specific frequencies are associated with a particular spatial location, which is not possible in the classic Fourier transform due to the non-localized properties of classic trigonometric basis functions. The transform is obtained through a collocation procedure and is therefore called the Fup collocation transform (FCT). The high efficiency of the FCT lies in the transform property, which keeps only the significant Fup coefficients that accurately describe the chosen function. Other Fup coefficients present a residual between a true function and their Fup presentation, which must be less than the prescribed threshold ε. This threshold has a fundamental meaning for the FCT, because it represents the approximation accuracy or the FCT precision level.

The multi-resolution approximation of any $u(x)$ function can be expressed as a linear combination of *Fup* basis functions in the following way:

$$u^J(x) = \sum_{j=0}^{J} \sum_{k=-n/2}^{(2^{j_{\min}+j}+n/2)} d_k^j \varphi_k^j(x) \tag{11}$$

where j is the resolution level, from zero to a maximum level J, needed for the desired accuracy, defined by a threshold ε. The Fup order is n; j_{min} is the resolution at the zero level, d_k^j are the Fup coefficients, φ_k^j are the Fup basis functions, and k denotes the location index at the current level. Thus, the grid is divided into distinct resolution levels. The minimum and maximum levels are predefined by the user.

If we define the domain as $\Omega = [X_1, X_2]$, then the characteristic interval at each level is equal to the scale or distance between adjacent collocation points:

$$\Delta x_j = (X_2 - X_1)/2^{j_{\min}+j}. \tag{12}$$

To demonstrate the Fup collocation transform, let us consider the following test function:

$$f(x) = \frac{1}{2} \cdot \left[1 - \tanh\left(\frac{x-0.75}{0.02}\right)\right] + e^{(-64^2(x-1.5)^2)} \tag{13}$$

with a relatively high threshold of ε=0.1, which implies that the residual between the Fup approximation and the function (13) must be less than the prescribed threshold. Other parameters are $j_{min} = 3, X_1 = 0, X_2 = 2$, and $n = 4$.

Basis functions are characterized by vertices or peaks where they have maximum values (Fig. 1). All basis functions with vertices inside the domain are called internal basis functions. Other functions are external basis functions, and only their influence within the domain is considered. The best choices for locations of the collocation points are the vertices of the internal basis functions.

The main difficulty in transformations with localized basis functions is the special treatment of the boundaries. For a complete Fup approximation in each characteristic interval Δx_j, we need $n+2$ $Fup_n(x)$ basis function [Gotovac and Kozulić (1999)]. For all $n/2$ external basis functions at the left and right boundaries, the collocation points are located at the boundaries (at X_1 and X_2). The approximations for internal and external basis functions should satisfy the function values in corresponding collocation points and the first $n/2$ derivatives in boundary collocation points (at X_1 and X_2), respectively, [Gotovac, Andričevič, and Gotovac (2007); Kozulić, Gotovac, and Gotovac (2007)].

The location of each basis function is actually determined by the location of the vertex and is defined by $b_k^j = X_1 + k\Delta x_j$. Calculations of basis function values and their derivatives at a general characteristic interval Δx_j should be done in the following form with respect to a basic characteristic interval 2^{-n}:

$$\varphi_k^{j(m)}(x) = \frac{1}{\left(2^n \Delta x_j\right)^{(m)}} Fup_n^{(m)}\left(\frac{x - b_k^j}{2^n \Delta x_j}\right) \tag{14}$$

where m is the order of the derivative. The compact support of the basis function at each level has length $(n+2)\Delta x_j$.

Fig. 3 shows the adaptive multi-resolution Fup collocation transform for the chosen function in Eq. (13). Fig. 3a shows adaptive grids for all levels and internal basis functions for the zero and the first level. Each subsequent level includes twice as many internal basis functions with half as much support and scale (Eq. (12)). Obviously, there are two sensitive regions with sharp gradients that require denser distributions of collocation points than the rest of the domain. The key step of the adaptive numerical algorithm is the transfer from the current level to the next level. The residual between the true function and the previous-level approximation is checked. Points where residual is below the prescribed threshold are removed from the grid (Fig. 3c). For the first and then for each subsequent level, the collocation algorithm should only satisfy the residual between the true function and the previous-level approximation.

By using the previously described adaptive procedure, the Fup collocation transform $u^J(x)$ of any function $u(x)$ is expressed in the form of (11). The d_k^j unknown Fup coefficients are obtained by solving a system of linear collocation equations. The function values are satisfied in collocation points:

$$\sum_{k \in Z^j} d_k^j \varphi_k^j(x_p^j) = r^j(x_p^j), p \in Z^j : 0 \le p \le 2^{j_{\min}+j}; \quad j = 0, ..., J \tag{15}$$

where Z^j is the irregular grid containing only the significant collocation points and the *Fup* basis functions that were obtained using the adaptive procedure.

The boundary derivatives are satisfied in points X_1 and X_2:

$$\sum_{k\in Z^j} d_k^j \varphi_k^{j^{(m)}}(X_b) = r^{j^{(m)}}(X_b), \quad m = 1,...,n/2; \quad b = 1,2; \quad j = 0,....,J. \tag{16}$$

The residual vectors in Eqs. (15) and (16) have the following forms:

$$\begin{aligned} r^j(x_p^j) &= f(x_p^j), \quad p \in Z^j : 0 \le p \le 2^{j_{\min}+j}; \quad j = 0 \\ r^j(x_p^j) &= f(x_p^j) - u^{j-1}(x_p^j), \quad p \in Z^j : 0 \le p \le 2^{j_{\min}+j}; \quad j = 1,...,J \\ r^{j^{(m)}}(X_b) &= f^{(m)}(X_b), \; j = 0; \quad b = 1,2 \\ r^{j^{(m)}}(X_b) &= 0, \quad j = 1,...,J \end{aligned} \tag{17}$$

In the same way as FCT performs a multi-scale decomposition of any function $u(x)$, it is possible to solve differential equations with corresponding boundary conditions by using the same adaptive procedure. This should be done by applying the corresponding differential operators in internal collocation points, and boundary-initial conditions in boundary collocation points.

4 Adaptive Fup Collocation Method (AFCM) for boundary-initial value problem

Here, we extend the adaptive procedure presented in Section 3 for solving boundary-initial value problems.

Generally, one-dimensional boundary-initial value problems can be described by the following nonlinear time-dependent partial differential equation:

$$LIu(x,t) \equiv \frac{\partial u(x,t)}{\partial t} + \mathrm{K}Iu(x,t) = f(x,t), \quad x \in (X_1,X_2), \quad t \in (0,T) \tag{18}$$

with corresponding boundary and initial conditions:

$$LBu(X_b,t) = g_b(X_b,t), \quad b = 1,2, \quad t \in (0,T) \tag{19}$$

$$u(x,0) = u_0(x), \quad x \in (X_1,X_2), \quad t = 0 \tag{20}$$

where $u(x,t)$ is a solution that depends on one spatial variable x, LI is the partial differential operator, KI is an operator that consists of partial derivatives with respect to the space only, LB is a boundary differential operator, and f, g_b and u_0 are known functions. The operators LB can be specified to impose Dirichlet, Neumann or mixed boundary conditions.

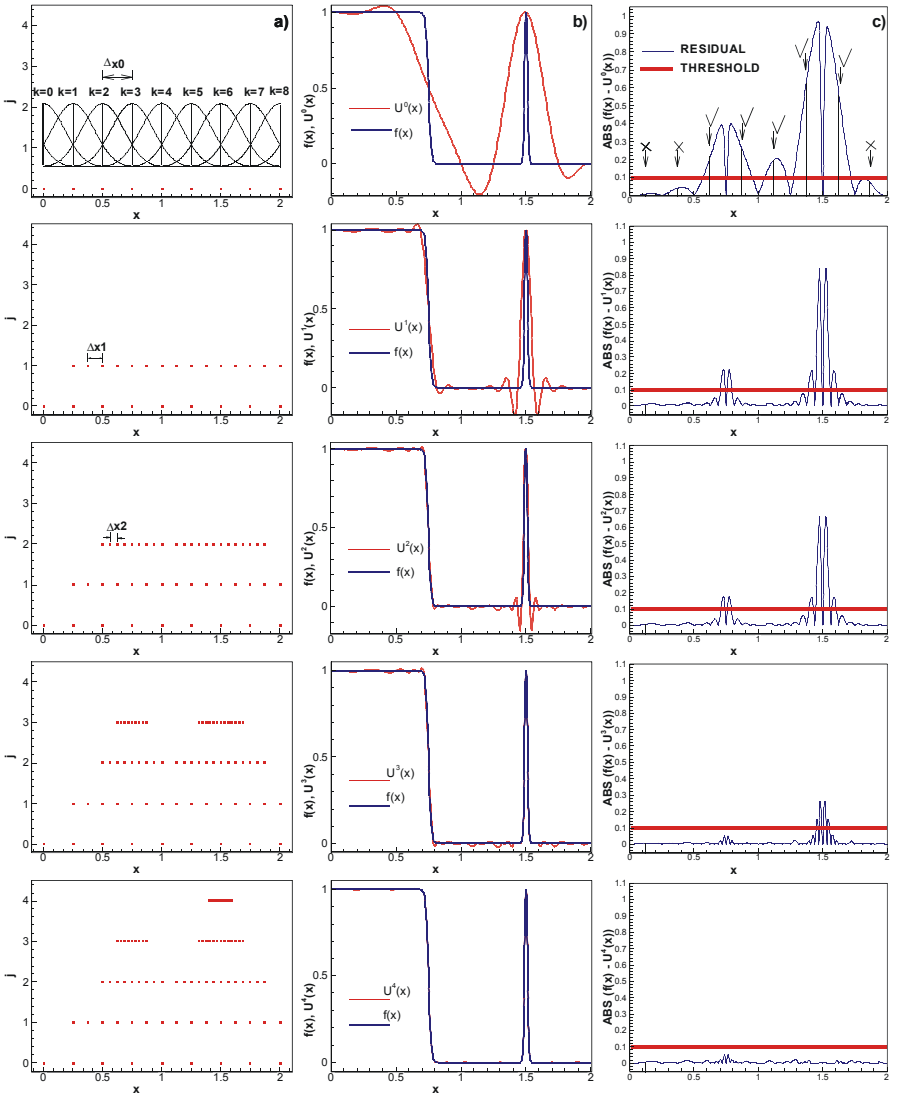

Figure 3: Multi-resolution approximation of the test function (13): a) adaptive grid development and internal basis functions, b) FCT approximation and function (13), c) *a priori* adaptive criteria for new collocation points based on the residuals between the test function (13) and its FCT approximations.

We can consider this problem as a boundary value problem in the space-time domain. Fup collocation discretization reduces the problem to a system of algebraic equations for domain collocation points of the form:

$$\sum_{k,l\in Z^j} d^j_{k,l} LI\varphi^j_{k,l}(x^j_p,t^j_q) = r^j(x^j_p,t^j_q): \quad 0 \le p \le 2^{j_{\min x}+j},\ 0 < q \le 2^{j_{\min t}+j}; \tag{21}$$

for the spatial boundary collocation points of the form:

$$\sum_{k,l\in Z^j} d^j_{k,l} LB\varphi^j_{k,l}(x^j_p,t^j_q) = r^j(x^j_p,t^j_q): \quad p = 0 \text{ or } p = 2^{j_{\min x}+j}, p,q \in Z^j; \tag{22}$$

and for initial conditions in corresponding temporal boundary collocation points of the form:

$$\sum_{k,l\in Z^j} d^j_{k,l} \varphi^j_{k,l}(x^j_p,t^j_q) = r^j(x^j_p,t^j_q): \quad q = 0,\ p,q \in Z^j. \tag{23}$$

In Eqs. (21)−(23), j takes on values from zero to the maximum level necessary for a desired accuracy, $d^j_{k,l}$ are *Fup* coefficients, $\varphi^j_{k,l}$ are *Fup* basis functions, k presents the index of collocation points at the current level for x-direction, l presents the index of collocation points at the current level for the t direction, $j_{\min x}$ and $j_{\min t}$ are numbers of collocation points at zero level in x and t directions, respectively, and r^j is the residual vector. The system (21)-(23) satisfies the differential flow equation in the internal collocation points (internal *Fup* coefficients) and boundary-initial conditions in the corresponding boundary collocation points (external *Fup* coefficients; see Fig. 4).

The residual vector in previous expressions has the following form:

$$\begin{aligned} r^j(x^j_p,t^j_q) &= f(x^j_p,t^j_q): \quad (0 \le p \le 2^{j_{\min x}+j},\ 0 < q \le 2^{j_{\min t}+j},\ p,q \in Z^j);\ j = 0 \\ r^j(x^j_p,t^j_q) &= f(x^j_p,t^j_q) - \sum_{i=0}^{j-1} \sum_{k,l\in Z^j} d^i_{k,l} LI\varphi^i_{k,l}(x^j_p,t^j_q): \\ &\qquad (0 \le p \le 2^{j_{\min x}+j},\ 0 < q \le 2^{j_{\min t}+j},\ p,q \in Z^j);\ j > 0 \end{aligned} \tag{24}$$

$$r^j(x^j_p,t^j_q) = g_b(x^j_p,t^j_q): \quad (p = 0 \text{ or } p = 2^{j_{\min x}+j},\ p,q \in Z^j;\ b = 1,2);\ j = 0$$

$$\begin{aligned} r^j(x^j_p,t^j_q) &= g_b(x^j_p,t^j_q) - \sum_{i=0}^{j-1} \sum_{k,l\in Z^j} d^i_{k,l} LB\varphi^i_{k,l}(x^j_p,t^j_q): \\ &\qquad (p = 0 \text{ or } p = 2^{j_{\min x}+j},\ p,q \in Z^j;\ b = 1,2);\ j > 0 \end{aligned} \tag{25}$$

$$r^j(x_p^j,t_q^j) = u_0(x_p^j,t_q^j): \quad (q=0,\ p,q\in Z^j);\ j=0$$

$$r^j(x_p^j,t_q^j) = u_0(x_p^j,t_q^j) - \sum_{i=0}^{j-1}\sum_{k,l\in Z^j} d_{k,l}^i \varphi_{k,l}^i(x_p^j,t_q^j): \quad (q=0,\ p,q\in Z^j);\ j>0 \tag{26}$$

Eqs. (21)-(26) define a boundary-initial value problem that can be solved uniquely by a simultaneous space-time adaptive procedure. The given differential equation is solved only at the zero level of resolution. Each non-zero level solves only a residual of the differential equation from the solution of all previous levels and gives a particular solution correction. Fup coefficients are associated with a specific resolution level and location in the space/time domain. Adaptive criteria add new collocation points at the next level only in the zones where the solution correction is greater than the prescribed threshold. Therefore, an adaptive criterion directly estimates the accuracy of the solution. With this approach, the Fup coefficients measure the local fluctuations of the solution simultaneously in space and time, and provide global control of both spatial and temporal errors.

If the partial differential equation is nonlinear, the algebraic problem is also nonlinear. Numerical solution of the problem is performed by use of the common Newton's damped method (see Gotovac, Andričevič, and Gotovac (2007)). Finally, it is worthwhile to note that AFCM can divide the time domain into many subdomains or time steps, such that final time solution in previous subdomain becomes the initial condition for the next subdomain. In this manner, time steps possess more common-sense comparisons with classic time integration, but in a much more efficient form.

4.1 Implementation of a general boundary conditions

The AFCM solves a one-dimensional boundary-initial value problem as a boundary value problem in the space-time domain. The treatment of general boundary conditions is a straightforward task, due to the collocation nature of the algorithm and the use of localized Fup basis functions.

Fig. 4 shows an example of a regular space-time dyadic grid for a given resolution level. This figure represents vertex locations of internal and external basis functions and the corresponding collocation points if the $Fup_4(x,y)$ functions are used. The complete Fup approximation needs two external basis functions at each boundary. For all external basis functions, the corresponding collocation points are located at the boundary. Thus, boundary collocation points are multiple collocation points where differential equation and boundary/initial conditions are satisfied, while additional possible equations should neglect higher partial derivatives which do not

affect solution accuracy near the boundary (Fig. 4).
For solving this problem, the following equations are applied:

- the PDE is satisfied for all internal collocation points:

$$d1) \quad LIu(x,t) = f(x,t).$$

- At boundary collocation points in the space direction, three equations are satisfied: *Left boundary* *Right boundary*

$$\begin{aligned} &l1) \quad LIu(X_1,t) = f(X_1,t) && r1) \quad LIu(X_2,t) = f(X_2,t) \\ &l2) \quad LBu(X_1,t) = g_1(X_1,t) && r2) \quad LBu(X_2,t) = g_2(X_2,t) \\ &l3) \quad \frac{\partial^4 u(X_1,t)}{\partial x^4} = 0 && r3) \quad \frac{\partial^4 u(X_2,t)}{\partial x^4} = 0 \end{aligned}$$

- For boundary collocation points at the initial time, the initial condition, the first and the second partial derivatives of the function $u(x,t)$ at the initial time are satisfied:

$$i1) \quad u(x,t_0) = u_0(x); \quad i2) \quad \frac{\partial u(x,t_0)}{\partial t} = u_1(x); \quad i3) \quad \frac{\partial^2 u(x,t_0)}{\partial t^2} = u_2(x)$$

where $u_1(x)$ and $u_2(x)$ are the first and the second derivatives, respectively, of the known function $u_0(x)$ with respect to x variable.

- For boundary collocation points at the final time, three equations are satisfied:

$$f1) \quad LIu(x,T) = f(x,T); \quad f2) \quad \frac{\partial^3 u(x,T)}{\partial t^3} = 0; \quad f3) \quad \frac{\partial^4 u(x,T)}{\partial t^4} = 0.$$

- For collocation points at corners, we need more equations. Numerical experiments [Kozulić and Gotovac (2000)] showed that it is convenient to neglect mixed partial derivatives of the solution function:

$$\begin{aligned} &c1) \quad \frac{\partial^6 u(x,t)}{\partial x^3 \partial t^3} = 0; \quad c2) \quad \frac{\partial^7 u(x,t)}{\partial x^4 \partial t^3} = 0; \\ &c3) \quad \frac{\partial^7 u(x,t)}{\partial x^3 \partial t^4} = 0; \quad c4) \quad \frac{\partial^8 u(x,t)}{\partial x^4 \partial t^4} = 0; \quad x = X_1, X_2; \quad t = t_0, T. \end{aligned}$$

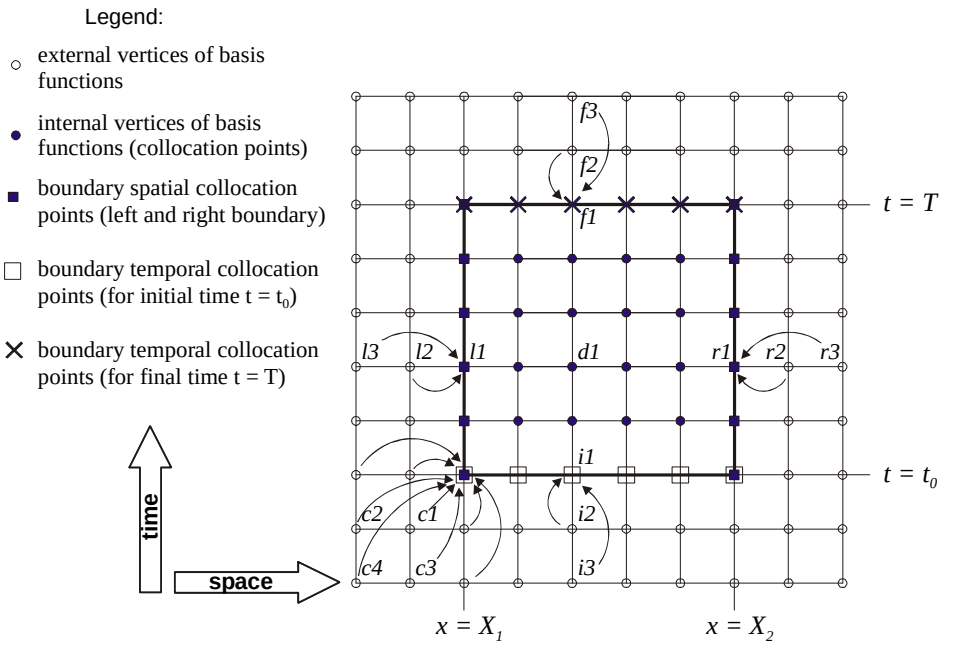

Figure 4: An example of a regular space-time dyadic grid

5 Numerical examples

5.1 Burgers' equation

The Burgers equation is an ideally-suited test problem for the numerical solution of partial differential equation (PDE), because it is nonlinear and has a known exact solution. In the past several years, there have been many studies on the numerical solutions of Burgers' equation, see Liu (2006), Liu (2009).

The Burgers equation has been of considerable physical interest because it is an appropriate simplification of the Navier-Stokes equations, and is also the governing equation for a number of one-dimensional flow systems. The problem is described by the following partial differential equation, initial and boundary conditions:

$$\frac{\partial u}{\partial t} = v\frac{\partial^2 u}{\partial x^2} - u\frac{\partial u}{\partial x}; \quad u(x,0) = -\sin(\pi x); \quad u(\pm 1,t) = 0 \tag{27}$$

where x and t are dimensionless space and time variables, respectively, and u is the dimensionless velocity. The computational domain and viscosity are defined by: $x \in [-1,1]$; $t \in [0, 1.5/\pi]$; $v = 10^{-2}/\pi$. The initial conditions are very simple and

monotonic. The Dirichlet boundary conditions are homogeneous. The analytical solution of the problem is known [Vasilyev and Paolucci (1997)]:

$$u^*(x,t) = -\frac{\int_{-\infty}^{\infty} \sin(\pi(x-\eta)) \exp\left(-\frac{\cos(\pi(x-\eta))}{2\pi\nu}\right) \exp\left(-\frac{\eta^2}{4\nu t}\right) d\eta}{\int_{-\infty}^{\infty} \exp\left(-\frac{\cos(\pi(x-\eta))}{2\pi\nu}\right) \exp\left(-\frac{\eta^2}{4\nu t}\right) d\eta} \tag{28}$$

Fig. 5 shows the time evolution of the analytic solution. The solution is characterized by a one-dimensional shock that is fixed in space, but rapidly increases in time. The shock is very narrow due to the small viscosity. The change from a uniformly smooth distribution to the shock structure is observed at the time $t = 1/\pi$.

Fig. 6 shows the numerical solution in the $x-t$ domain obtained with space-time AFCM using $Fup_4(x,t)$ basis functions. The results obtained at the time $T = 0.4$, which is sufficient for the smooth initial condition to become highly intermittent, are presented. In the solution, a fixed shock that becomes steeper as time increases is represented by lines of different velocities that join together at one point, $x = 0$, when time reaches its final value. The initial grid is determined by $j_{\min x}$= 8 and $j_{\min t}$= 2. Grid adaptation is performed in both space and time directions (Fig. 7) and represents all resolved scales and frequencies. We use here only one time subdomain or time step, while classical numerical time integration requires a few hundreds steps for the prescribed local temporal error [Vasilyev and Bowman (2000); Alam, Kevlahan, and Vasilyev (2006); Kozulić, Gotovac, and Gotovac (2007)]. The local time step is particularly interesting. The time scale is smaller where the spatial gradients are steeper. The results presented in Figures 6 and 7 were obtained with prescribed accuracy criterion $\varepsilon = 10^{-2}$.

Fig. 8 shows the dependence between the number of collocation points N selected by AFCM at time $T = 1.5/\pi$ and the L_{max} norm for different values of the threshold parameter $\varepsilon = 10^{-2}$, 10^{-3}, 10^{-4} and 10^{-5}.

$L_{\max}$ norm represents the maximum absolute error between the numerical (u) and analytic (u^*) solutions. For $\varepsilon = 10^{-2}$, the maximum number of grid points is 8941. A smaller threshold implies more collocation points and higher accuracy. Notice that absolute error is strictly less than the prescribed threshold. This clearly shows that the threshold defines the error indicator and the *a priori* adaptive criterion in space-time AFCM, as a direct measurement of the difference between the numerical and exact solutions.

5.2 Advection-dispersion problem

This problem describes the mixing of transport processes, for instance in porous media. For the mathematical model of the groundwater contamination problem,

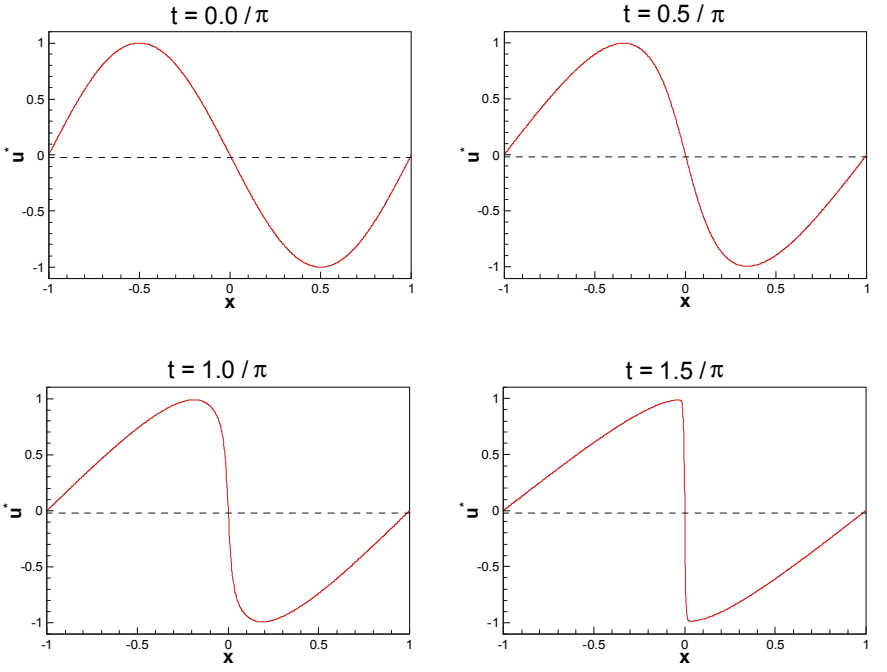

Figure 5: Exact solution of the Burgers equation at different values of t

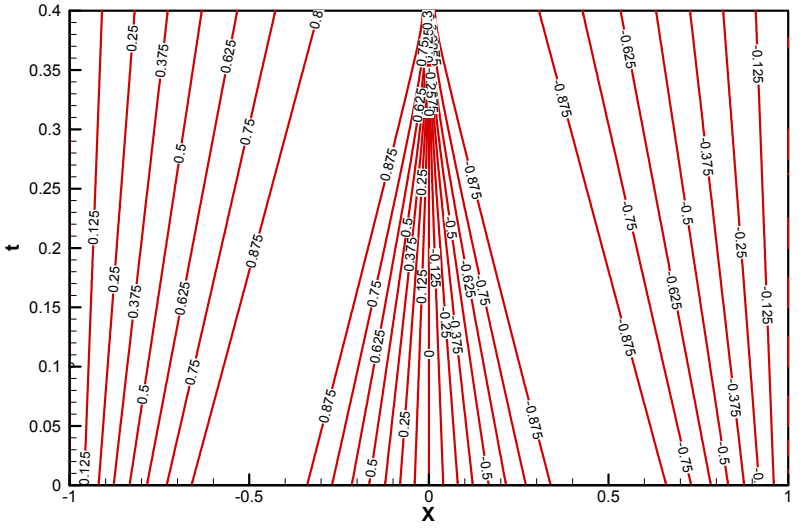

Figure 6: Numerical solution of the Burgers equation in the $x-t$ domain

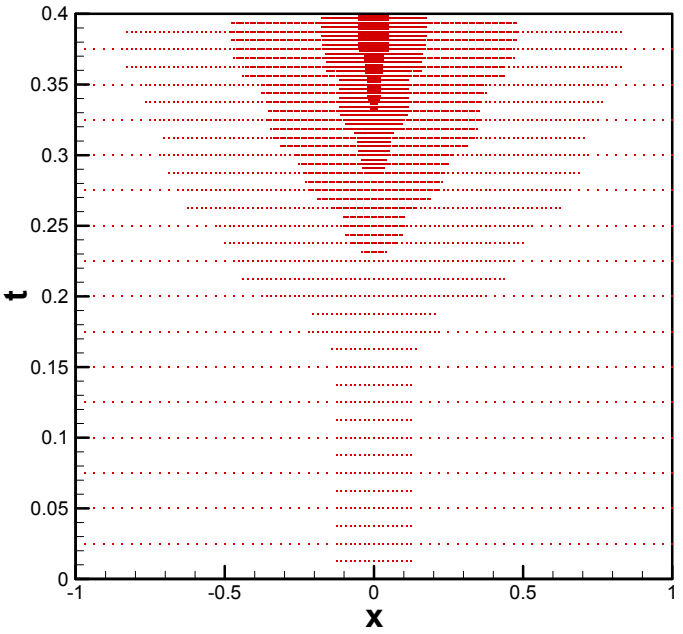

Figure 7: Adaptive space-time grid

many researchers use the backward in time advection-dispersion equation to govern the problem. Liu, Chang and Chang (2009) recently derived a closed-form solution of the second kind Fredholm integral equation by employing the quasi-boundary method and demonstrated very good results in the numerical computations of advection-dispersion equation.

Here, we consider a uniform and constant velocity field. The one-dimensional advection – dispersion process can be described by the following equation, initial and boundary conditions:

$$\frac{\partial C(x,t)}{\partial t} = D\frac{\partial^2 C(x,t)}{\partial x^2} - u\frac{\partial C(x,t)}{\partial x} \tag{29}$$

$$C(x,0) = 0 \tag{30}$$

$$C(0,t) = C_0; \quad \frac{\partial C(2,t)}{\partial x} = 0 \tag{31}$$

where C is the concentration (M/L^3), D is the dispersion coefficient and u is the transport velocity in the x direction. The domain, dispersion, velocity and threshold are defined by: $L = 2(m); \quad D = 3\cdot 10^{-8}\left(m^2/s\right); \quad u = 10^{-3}\left(m/s\right); \quad \varepsilon = 10^{-2}.$

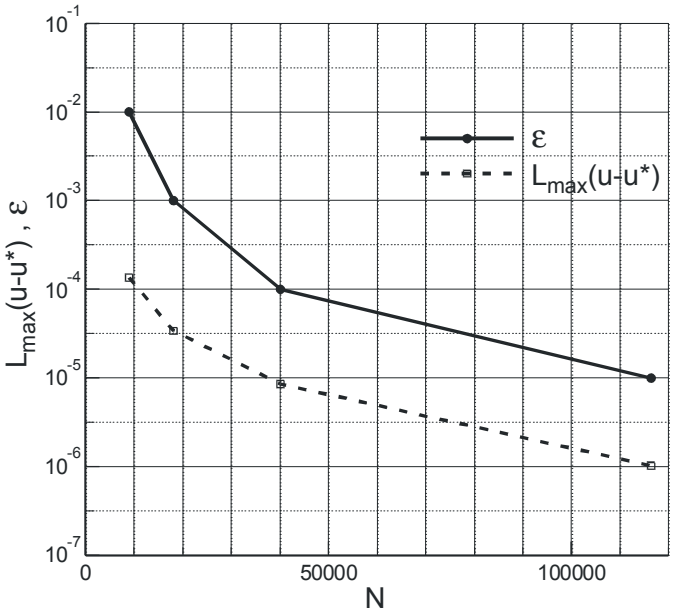

Figure 8: Dependence between maximum absolute error (dashed line) and the number of collocation points required by different choices of the threshold parameter (solid line)

The initial condition (Eq. (30)) shows that initially the domain was occupied by fresh water. The left boundary consists of the salt source (or some other denser fluid) that continuously flows into the domain. The right boundary specifies that there is no dispersion flux through the boundary.

Fig. 9 shows the numerical solution in the $x-t$ domain obtained with space-time AFCM using Fup_4 basis functions. It represents the change in solute concentration over the space over time, and this change occurs in a narrow transition zone. Fig. 10 shows an adaptive grid in the space-time domain. In initial stages of the process, a denser grid is needed due to very challenging initial conditions and the creation of a very sharp concentration front. The accumulation of grid points in the region of intermittency and the non-uniformity of the time step in space shows the efficiency of the method. Once again, we use only one time step, and the difference between presented numerical and analytical solutions (see Ogata and Banks (1961)) is strictly less than the prescribed threshold for t > 0.3. For early times, the numerical error is slightly higher than the threshold, due to discontinuity of initial conditions. However, this initial error does not propagate further over time, because

AFCM converts the boundary-initial value problem to a quasi-boundary problem controlling the global temporal/spatial error.

This example shows the ability of the method to handle moving fronts and to change the grid dynamically, following a front during the simulation.

6 Summary and conclusions

In this paper, we demonstrated an algorithm for solving boundary-initial value problems. The proposed method (AFCM) solves nonlinear PDEs simultaneously in the space and time domains. In numerical procedures, a collocation technique and infinitely differentiable Fup basis functions are used. The adaptive Fup multi-resolution approach is able to dynamically track the evolution of the solution and to allocate higher grid density as necessary. Therefore, the number of collocation points or basis functions needed to represent a solution is considerably optimized. Spatiotemporal discretization and grid adaptation depend on the character of the solution and accuracy criteria. Numerical results show accumulations of grid points in regions of intermittency and non-uniformities of time steps in space. The time domain can be split to the more subdomains, while solutions from previous subdomains serve as initial conditions for other subdomains.

Each nonzero level solves only residuals of the PDEs from all previous levels and gives a particular solution correction. Numerical experiments with known analytic solutions show that solution correction on some level is greater than the difference between the analytic solution and the overall solution up to that level. This means that an adaptive criterion directly estimates accuracy of the solution even if analytical solution is not known *a priori*. The strength of the space-time AFCM is that it controls global time integration errors, contrary to classical time-integration schemes, and enables an adaptive grid that directly shows all the time and space scales, with a near-minimal number of collocation points, necessary in order to obtain solution with prescribed accuracy.

However, it should be mentioned that the convergence of the numerical solution depends significantly on initial conditions and the types of the partial differential equations used. Initial conditions that describe abrupt fronts should be analyzed at a high level of resolution to be approximated accurately at a given threshold. Consequently, node spacing must be sufficiently refined to capture these discontinuous fronts. Generally speaking, partial differential equations that are predominantly hyperbolic are much more difficult to solve numerically those that are predominantly parabolic. Finally, the main weakness of the presented methodology is its introduction of an additional dimension, which is more severe for demanding 3-D spatial problems because they then become 4-D problems (x, y, z, t). However, use

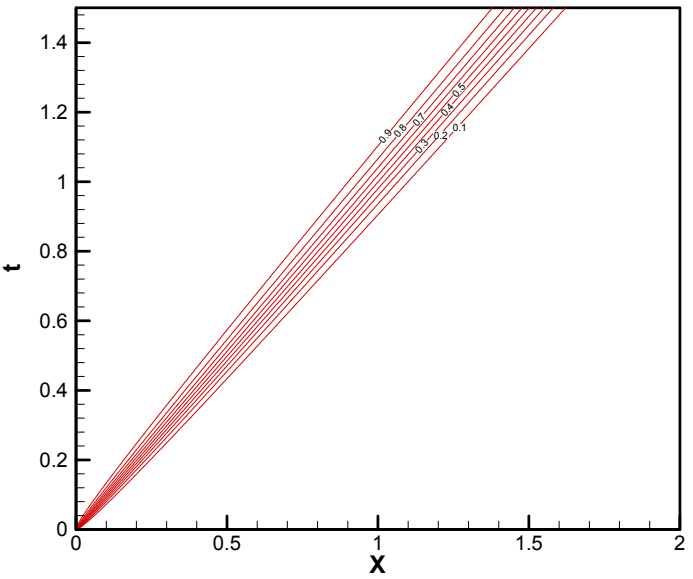

Figure 9: Numerical solution of the advection-dispersion equation

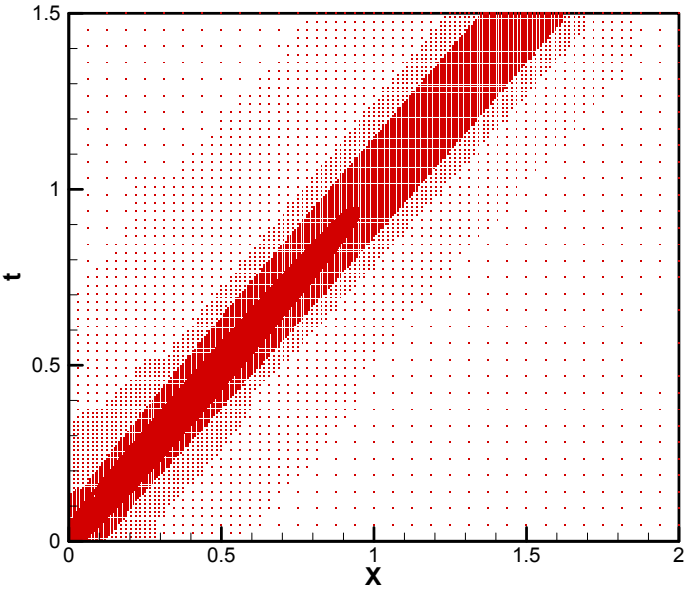

Figure 10: Adaptive grid in the space-time domain

of smaller time steps (which are still larger than in those used in classical temporal integration) and parallel computing can still preserve the efficiency of the proposed methodology.

References

Alam, J.M.; Kevlahan, N.K.-R.; Vasilyev, O.V. (2006): Simultaneous space-time adaptive wavelet solution of nonlinear parabolic differential equations. *J. Comput. Phys*., vol. 214, pp. 829-857.

Amaziane, B.; Naji, A.; Ouazar, D. (2004): Radial basis function and genetic algorithms for parameter identification to some groundwater flow problems. *CMC: Computers, Materials & Continua,* vol. 1, no. 2, pp. 117-128.

Ascher, U.M.; Petzold, L. (1998): *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Society for Industrial and Applied Mathematics, Berkeley.

Atluri, S. N.; Han, Z. D.; Rajendran, A. M. (2004): A new implementation of the meshless finite volume method, through the MLPG "mixed" approach. *CMES: Computer Modeling in Engineering & Sciences*, vol. 6, no. 6, pp. 491–514.

Atluri, S. N.; Shen, S. (2002): The meshless local Petrov-Galerkin (MLPG) method: A simple & lesscostly alternative to the finite element and boundary element methods. *CMES: Computer Modeling in Engineering & Sciences,* vol. 3, no. 1, pp. 11-52.

Behrens, J.; Iske, A.; Käser, M. (2003): Adaptive meshfree method of backward characteristics for nonlinear transport equations. in: *Meshfree Methods for Partial Differential Equations* (Bonn, 2001), in: *Lect. Notes Comput. Sci. Eng.*, vol. 26, Springer, Berlin, pp. 21–36.

Bertoluzza, S. (1996): Adaptive wavelet collocation method for the solution of Burgers equation. *Transport Theory and Statistical Physics*, vol. 25, pp. 339-352.

Bertoluzza, S.; Naldi, G. (1996): A wavelet collocation method for the numerical solution of PDEs. *Applied & Computational Harmonic Analysis*, vol. 3, pp. 1-9.

Cao, W.; Demeler, B. (2005): Modeling analytical ultracentrifugation experiments with an Adaptive Space-Time Finite Element Solution of the Lamm Equation. *Biophysical J.*, vol. 89, no. 3, pp. 1589-1602.

Cho, H.A.; Golberg, M.A.; Muleshkov, A.S.; Li, X. (2004): Trefftz methods for time dependent partial differential equations. *CMC: Computers, Materials & Continua,* vol. 1, no. 1, pp. 1-37.

Cruz, P.; Mendes, A.; Magalhaes, F.D. (2001): Using wavelets for solving PDEs: an adaptive collocation method. *Chem. Eng. Sci.*, vol. 56, pp. 3305-3309.

Gotovac, B.; Kozulić, V. (1999): On a selection of basis functions in numerical analyses of engineering problems. *Int. J. Engineering Modelling*, vol. 12, no. 1-4, pp. 25-41.

Gotovac, B.; Kozulić, V. (2002): Numerical solving the initial value problems by R_{bf} basis functions. *Struct. Eng. Mech.*, vol. 14, no. 3, pp. 263-285.

Gotovac, H.; Andričevič, R.; Gotovac, B. (2007): Multi–resolution adaptive modeling of groundwater flow and transport problems. *Adv. Water Resour.*, vol. 30, no. 5, pp. 1105-1126.

Hairer, E.; Wanner, G. (1996): *Solving Ordinary Differential Equations II*. Springer-Verlag, Berlin.

Han, Z. D.; Atluri, S. N. (2004): A Meshless Local Petrov-Galerkin (MLPG) approaches for solving 3-dimensional elasto-dynamics. *CMC: Computers, Materials & Continua*, vol. 1, no. 2, pp. 129–140.

Hesthaven, J.S.; Jameson, L.M. (1998): A wavelet Optimized Adaptive Multi–domain method. *J. Comput. Physics*, vol. 145, pp. 280-296.

Holmstrom, M. (1999): Solving hyperbolic PDEs using interpolation wavelets. *Scientific Computing*, vol. 21, pp. 405-420.

Iske, A.; Käser, M. (2005): Two-phase flow simulation by AMMoC, an adaptive meshfree method of characteristics. *CMES: Computer Modeling in Engineering & Sciences,* vol. 7, no. 2, pp. 133-148.

Kansa, E.J. (1990): Multiquadrics: A scattered data approximation scheme with applications to computational fluid-dynamics II: Solutions to parabolic, hyperbolic and elliptic partial differential equations. *Comput. Math. Appl.*, vol. 19, no. 8/9, pp. 147-161.

Kolodiazhny, V.M.; Rvačev, V.A. (2007): Atomic functions: generalization to the multivariable case and promising applications. *Cybern. Syst. Anal.*, vol. 43, no. 6, pp. 893-911.

Kozulić, V.; Gotovac, B. (2000): Numerical analysis of 2-D problems using Fup basis functions. *Int. J. Engineering Modelling*, vol. 13, no. 1-2, pp. 7-18.

Kozulić, V.; Gotovac, H.; Gotovac, B. (2007): An adaptive multi-resolution method for solving PDE' s. *CMC: Computers, Materials, and Continua*, vol. 6, no. 2, pp. 51-70.

Kravchenko, V.F.; Basarab, M.A.; Perez-Meana, H. (2001): Spectral properties of atomic functions used in digital signal processing. *J. Commun. Technol. Electron.*, vol. 46, pp. 494-511.

La Rocca, A.; Power, H.; La Rocca, V.; Morale, M. (2005): A meshless approach based upon radial basis function hermite collocation method for predicting

the cooling and the freezing times of foods. *CMC: Computers, Materials & Continua,* vol. 2, no. 4, pp. 239-250.

Lee, C.; Im, C.H.; Jung, H.K.; Kim, H.K.; Kim, D.W. (2007): A posteriori error estimation and adaptive node refinement for fast moving least square reproducing kernel (FMLSRK) method. *CMES: Computer Modeling in Engineering & Sciences*, vol. 20, no. 1, pp. 35-41.

Libre, N.A.; Emdadi, A.; Kansa, E.J.; Shekarchi, M.; Rahimian, M. (2008): A fast adaptive wavelet scheme in RBF collocation for nearly singular potential PDEs. *CMES: Computer Modeling in Engineering & Sciences*, vol.38, no.3, pp.263-284.

Lin, H.; Atluri, S. N. (2000): Meshless local Petrov-Galerkin (MPLG) method for convection-diffusion problems. *CMES: Computer Modeling in Engineering & Sciences,* vol. 1, no. 2, pp. 45-60.

Liu, C.-S. (2006): An efficient backward group preserving scheme for the backward in time Burgers equation. *CMES: Computer Modeling in Engineering & Sciences,* vol. 12, no. 1, pp. 55-65.

Liu, C.-S. (2009): A fictitious time integration method for the Burgers equation. *CMC: Computers, Materials & Continua,* vol. 9, no. 3, pp. 229-252.

Liu, C.-S.; Chang, C.-W.; Chang, J.-R. (2009): A quasi-boundary semi-analytical method for backward in time advection-dispersion equation. *CMC: Computers, Materials & Continua,* vol. 9, no. 2, pp. 111-135.

Lorbiecka, A.Z.; Vertnik, R.; Gjerkeš, H.; Manojlovic, G.; Sencic, B.; Cesar, J.; Šarler, B. (2009): Numerical modeling of grain structure in continuous casting of steel. *CMC: Computers, Materials & Continua,* vol. 8, no. 3, pp. 195-208.

Mai-Duy, N. (2004): Indirect RBFN method with scattered points for numerical solution of PDEs. *CMES: Computer Modeling in Engineering & Sciences*, vol. 6, no. 2, pp. 209-226.

Mai-Duy, N.; Khennane, A.; Tran-Cong, T. (2007): Computation of laminated composite plates using integrated radial basis function networks. *CMC: Computers, Materials & Continua,* vol. 5, no. 1, pp. 63-77.

Mehra, M.; Kevlahan, N.K.R. (2008): An adaptive wavelet collocation method for the solution of partial differential equations on the sphere. *J. Comput. Phys.*, vol. 227, pp. 5610–5632.

Ogata; Banks (1961): published in Fetter, C.W. (1994). *Applied Hydrogeology*, Third Edition. Prentice-Hall Inc., New Jersey, USA.

Rvačev, V.L. (1982): *Theory of R-functions and their application*, (in Russian). Naukova dumka, Kiev.

Rvačev, V.L.; Rvačev, V.A. (1971): On a finite function, (in Russian). *DAN URSR*,

Ser. A 6, pp. 705-707.

Sarra, S.A. (2005): Adaptive radial basis function methods for time dependent partial differential equations. *Appl. Numer. Math.*, vol. 54, no. 1, pp. 79–94.

Shen, S.P.; Atluri, S.N. (2005): A tangent stiffness MLPG method for atom/continuum multiscale simulation. *CMES: Computer Modeling in Engineering & Sciences*, vol. 7, no. 1, pp. 49–67.

Sladek, V.; Sladek, J.; Zhang, C. (2006): A comparative study of meshless approximations in local integral equation method. *CMC: Computers, Materials & Continua,* vol. 4, no. 3, pp. 177-188.

Šarler, B. (2005): A radial basis function collocation approach in computational fluid dynamics. *CMES: Computer Modeling in Engineering & Sciences,* vol. 7, no. 2, pp. 185-193.

Vasilyev, O.V.; Bowman, C. (2000): Second generation wavelet collocation method for the solution of partial differential equations. *J. Comput. Physics*, vol. 165, pp. 660-693.

Vasilyev, O.V.; Kevlahan, N.K.R. (2005): An adaptive multilevel wavelet collocation method for elliptic problems. *J. Comput. Phys.*, vol. 206, pp. 412–431.

Vasilyev, O.V.; Paolucci, S. (1997): A fast adaptive wavelet collocation algorithm for multidimensional PDEs. *J. Comput. Physics*, vol. 125, pp. 16-56.

Wackers, J.; Koren, B. (2003): A Simple and Efficient Space-Time Adaptive Grid Technique for Unsteady Compressible Flows. CWI Report in: *Modelling, Analysis and Simulation*, MAS-E0321, ISSN 1386-3703, pp. 1-11.

Wang, R.; Keast, P.; Muir, P. (2004): A high order global spatially adaptive collocation method for 1-D parabolic PDEs. *Appl. Numer. Math.*, vol. 50, pp. 239-260.

Wu, Z. (2004): Dynamically Knots setting in meshless method for solving time dependent propagation equation. *Comput. Meth. Appl. Mech. Eng.*, vol.193, no.12-14, pp. 1221-1229.