INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **54**:1215–1238
Published online 21 February 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1438

Particle transport method for convection problems with reaction and diffusion

O. Shipilova[1,*,†], H. Haario[1] and A. Smolianski[2]

[1]*Lappeenranta University of Technology, P.O. Box 20, FIN-53851, Lappeenranta, Finland*
[2]*Institute of Mathematics, Zurich University, Winterthurerstr. 190, CH-8057 Zurich, Switzerland*

SUMMARY

The paper is devoted to the further development of the particle transport method for the convection problems with diffusion and reaction. Here, the particle transport method for a convection–reaction problem is combined with an Eulerian finite-element method for diffusion in the framework of the operator-splitting approach. The technique possesses a special spatial adaptivity to resolve solution singularities possible due to convection and reaction terms. A monotone projection technique is used to transfer the solution of the convection–reaction subproblem from a moving set of particles onto a fixed grid to initialize the diffusion subproblem. The proposed approach exhibits good mass conservation and works with structured and unstructured meshes.

The performance of the presented algorithm is tested on one- and two-dimensional benchmark problems. The numerical results confirm that the method demonstrates good accuracy for the convection-dominated as well as for convection–diffusion problems. Copyright © 2007 John Wiley & Sons, Ltd.

Received 2 May 2006; Accepted 4 December 2006

KEY WORDS: convection–reaction–diffusion; particles; adaptivity; projection; mass conservation

1. INTRODUCTION

Convection–diffusion–reaction (CDR) problems to be considered in this paper arise in many natural phenomena and industrial applications. The special feature of this kind of problems is an ability to switch between parabolic and hyperbolic types with respect to the domination of the diffusion or convection terms. The presence of discontinuities or high gradients, further here

*Correspondence to: O. Shipilova, Lappeenranta University of Technology, P.O. Box 20, FIN-53851, Lappeenranta, Finland.
†E-mail: olga.shipilova@lut.fi

Contract/grant sponsor: The Finnish Technology Development Agency; contract/grant number: T14 (MASI)

singularities, of the exact solution for the convection-dominated problems makes their numerical resolution difficult, often resulting in either artificial oscillations or excessive numerical diffusion of the solution. Over the last three decades, numerous techniques have been created in the framework of Eulerian, Lagrangian and mixed Eulerian–Lagrangian concepts for the transient diffusion problems, see, e.g. [1–5].

In the Eulerian methods, the solution moves with respect to a fixed grid, while in the Lagrangian methods a grid changes with the solution. In general, the fixed-grid methods are able to produce accurate numerical solutions with high quality of discontinuities resolution for the convection-dominated as well as for diffusion-dominated problems; however, the connection between time and space discretization may result in strict conditions on time and space step. The Lagrangian methods, which are not confined by the stepping restrictions, may be a better choice to treat transient problems; unfortunately, the moving mesh underlaying the computations may be subjected to excessive distortion. Due to these reasons, a Lagrangian approach is often combined with the Eulerian one to achieve a good accuracy for the solution of convection–diffusion problems, see, for example, the Eulerian–Lagrangian localized adjoint method (ELLAM) [6–10], the Lagrange–Galerkin method [11, 12], the free-Lagrangian and semi-Lagrangian methods [13–15], and the characteristics finite-element (FE) method [16, 17]. Here, an additional fixed grid is used to avoid severe distortion of the evolving grid. Other mixed Eulerian–Lagrangian approaches for the CDR problems are based on splitting of the diffusion and convection operators and resolving each of them by the most suitable solver. Thus, for example, in the class of particles-in-cell methods, which originally came from the computational plasma physics ([18, 19] and references in it), the reaction–diffusion is approximated by an Eulerian finite difference method, while for the convection a Lagrangian (characteristics based) one is used, see, for example [20].

In this work we use the idea of operator-splitting to continue the development of the particle transport method (PTM), first presented in [21], for the case of a general CDR problem. An FE method is used here for the diffusion subproblem while all difficulties connected with the hyperbolic operator on the convection part are resolved with PTM. The spatial adaptivity is used in the framework of meshless concept, see, e.g. [22, 23]. The solution of the convection–reaction subproblem is carried out by a moving system of particles, that allows us to avoid the big computational expenses due to full reconstruction of the mesh. The implementation of the operator-splitting means a presence of the projection between two grids, in our case between moving particles and fixed mesh, that might be overly diffusive and, moreover, it might not save mass (see, for instance, the discussions in [24–26]). Here, we use a special, fast and monotone projection technique exhibiting a good mass conservation that has been presented in the framework of the PTM in [21]. The main differences with respect to the particle-in-cell approach is the spatial adaptivity in the regions of discontinuities and the projection methods.

The paper is organized as follows. Section 2 is devoted to the formulation of the problem and operator-splitting technique used in the proposed method. The description of the numerical scheme is given in Section 3. Results of numerical experiments can be found in Section 4. In [21] pure convection and one-dimensional diffusion are presented. Here we extend the work to the two-dimensional CDR cases for such problems as layered reaction, the convection–reaction of a Gaussian pulse, the mixing of hot and cold fronts and convection–diffusion of the Gaussian pulse. The first two convection–reaction tests have been chosen to demonstrate the performance of the introduced spatial adaptivity technique and to conduct the error analysis of the method. In the case

DOI: 10.1002/fld

of convection–diffusion of a Gaussian hill, the results are compared with the ones obtained by fractional step finite difference methods [10]. We conclude with the problem of chromatographic separation to show how the proposed technique works on a more general case of the system of CDR equations in comparison with a pure Eulerian method, the method of lines (MOL). Some conclusions are drawn in Section 5.

2. MATHEMATICAL MODEL AND OPERATOR SPLITTING

We consider the following CDR system.

Governing equation:

$$\frac{\partial u}{\partial t} + \mathbf{v} \cdot \nabla u = D\Delta u + Re(u, \mathbf{x}, t) \quad \text{in } \Omega \times (0, T) \tag{1}$$

where $u = u(\mathbf{x}, t)$ and the space variable $\mathbf{x}$ belongs to the domain $\Omega \subset \mathbb{R}^d$, $d = 1, 2$, the time variable t varies in the interval $[0, T]$, $\mathbf{v} = \mathbf{v}(\mathbf{x}, t)$ is the given solenoidal, i.e. divergence free, velocity field on $\Omega \times [0, T]$, D is the known constant diffusion coefficient. The reaction term takes the form $Re(u, \mathbf{x}, t) = K(\mathbf{x}, t)r(u)$, where K may be a (stepwise) discontinuous function of $\mathbf{x}$.

Initial conditions:

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}) \quad \text{in } \Omega \tag{2}$$

Boundary conditions:

$$u(\mathbf{x}, t) = u_{\text{in}}(\mathbf{x}, t) \quad \text{on } \Gamma_{\text{in}} \times (0, T) \tag{3}$$

$$\alpha u(\mathbf{x}, t) + \beta \frac{\partial u}{\partial n}(\mathbf{x}, t) = u_{\text{out}}(\mathbf{x}, t) \quad \text{on } \Gamma_{\text{out}} \times (0, T) \tag{4}$$

here, α and β are nonnegative constants, Γ_{in} and Γ_{out} are the portions of the domain boundary $\partial\Omega$ ($\Gamma_{\text{in}} \cup \Gamma_{\text{out}} = \partial\Omega$), where, respectively, inflow and outflow boundary conditions are defined.

The peculiarity of the described problem is that governing equation (1) might turn its type from parabolic to hyperbolic when, respectively, the diffusion or convection term dominates. If the diffusion coefficient D is very small, the degeneration of the differential operator gives rise to the contribution of (1) with boundary conditions (3) and (4). Moreover, for a small D the approximation of the solution, which may change rapidly, becomes a hard numerical problem. In this case, the domination of the convective term might result in the spurious oscillations or smearing. Thus, it is worth performing the operator splitting to separate the convection and diffusion parts.

To perform the operator splitting, described in [27], the time interval $[0, T]$ is divided into M subintervals $[t_{m-1}, t_m]$, $m = 1, 2, \ldots, M$, where $t_0 = 0$ and $t_M = T$. Denote by $u^{(m)}$ the approximation to $u(\mathbf{x}, t_m)$ ($u^{(0)} \equiv u_0$ is the given initial condition). Then on every time interval $[t_{m-1}, t_m]$ problem (1)–(4) is approximated by a sequence of two subproblems.

DOI: 10.1002/fld

(*) *Convection–reaction subproblem*:

$$\begin{aligned} \frac{\partial u}{\partial t} + (\mathbf{v}\cdot\nabla)u = Re(u,\mathbf{x},t) \quad & \text{in } \Omega\times[t_{m-1},t_m] \\ u(\mathbf{x},t) = u^{(m-1)}(\mathbf{x}) \quad & \text{in } \Omega\times\{t=t_{m-1}\} \\ u(\mathbf{x},t) = u_{\text{in}}(\mathbf{x},t) \quad & \text{on } \Gamma_{\text{in}}\times(t_{m-1},t_m) \end{aligned} \tag{5}$$

The solution of the convection–reaction subproblem, $u^{(*)}$, defines an initial condition for the diffusion subproblem.

(**) *Diffusion subproblem*:

$$\begin{aligned} \frac{\partial u}{\partial t} - D\Delta u = 0 \quad & \text{in } \Omega\times[t_{m-1},t_m] \\ u(\mathbf{x},t) = u^{(*)}(\mathbf{x}) \quad & \text{in } \Omega\times\{t=t_{m-1}\} \\ u(\mathbf{x},t) = u_{\text{in}}(\mathbf{x},t) \quad & \text{on } \Gamma_{\text{in}}\times(t_{m-1},t_m) \\ \alpha u(\mathbf{x},t) + \beta\frac{\partial u}{\partial n}(\mathbf{x},t) = u_{\text{out}}(\mathbf{x},t) \quad & \text{on } \Gamma_{\text{out}}\times(t_{m-1},t_m) \end{aligned} \tag{6}$$

It gives the solution of CDR problem (1)–(4) on the mth time step, $u^{(m)}$. If the time step $\Delta t = t_m - t_{m-1}$ is a constant, the error of the splitting is known to be $O(\Delta t)$, see for details [27].

This kind of splitting will allow us to use an FE method for the diffusion subproblem, while the transport part will be resolved with the PTM [21].

3. NUMERICAL TECHNIQUE

This section is devoted to the description of the numerical technique based on PTM in the framework of operator splitting (*)–(**) for a CDR problem.

In the algorithm we use a combination of a stationary mesh in the domain, further called as *the grid*, and a moving system of particles, further called as *the particles*. The grid is used for solving the diffusion subproblem and an adaptive construction of the particle set. The convection of the solution is carried out by the particles as well as integration of the reaction term. The shift between the grid and the particles is implemented by linear interpolation for each grid node within the projection procedure.

Thus, dividing the time integration by given time steps and supposing that initially we have a fixed grid, the numerical technique to solve the CDR problem on the mth time step consists of four main stages: (i) creation of the adaptive particle set in accordance with the solution's approximation at the previous time step, $u^{(m-1)}$; (ii) solving of convection–reaction subproblem (*) on the time interval (t_{m-1}, t_m) by the PTM on the appropriate particle set; (iii) projection of the intermediate solution $u^{(*)}$ from the particle set onto the grid; and (iv) solving of diffusion subproblem (**) on the grid by a standard Eulerian method for diffusion problems, for instance, the Galerkin FE-discretization in space and some implicit discretization in time. These steps will be explained in more detail below.

DOI: 10.1002/fld

3.1. Adaptive particle set

The particle set is used for the solution of convection–reaction subproblem (*) by means of PTM, where special care is taken for the region of steep fronts or discontinuities. To locate the adaptive set of particles the grid nodes are used. In general, the creation of the particle system for the convection–reaction problem is based on two main ingredients: what is *a priori* known about the singularities and how the solution may evolve during the integration. First of all, the singularities due to initial and inflow values have to be taken into account. Additionally, the reaction may cause regions of discontinuities or high gradients to appear during the computation, due to, e.g. different reaction rates in the reactive domain or fast reactions between various components in a reactive flow. So we should adapt the distribution of particles at the initial stage and near the inflow part of the boundary, take into account what is known about the discontinuities of the reaction rate function $K = K(\mathbf{x}, t)$, as well as monitor the solution.

The adaptivity procedures may be based on various indicator function values, further called here as *signal values*. For singularities due to discontinuities or sharp fronts, the absolute value of the gradient of the solution is employed as the signal. In addition, the examples presented below show how areas with a *smooth* solution also require some attention to achieve reasonable accuracy when using linear interpolation for the projection. Here, the absolute value of the second derivatives of the solution is used as the signal for adaptivity. More details on the construction of the sharp and smooth signal values can be found in [21].

The heuristics for creating particles on the basis of the signal values may, naturally, be implemented in various ways. We shortly describe here the algorithm for the adaptivity introduced in [21].

Let G_i be the signal value of the solution on the ith grid element and denote by G_{mean} and $G_{\max}$ the mean and the maximum of the signal value on the grid. If N_{current} gives the number of particles on the ith element so far, we add

$$N_{\text{add}} = \min\{N_{\max} - N_{\text{current}}, \varphi(G_i)\}$$

new particles on the ith element, where $\varphi \equiv 0$ on $[0, G_{\text{mean}})$ monotonically increases within $[G_{\text{mean}}, G_{\max}]$ and $\varphi(G_{\max}) = N_{\max}$. Here $N_{\max}$ is a user-defined number between, e.g. 5 and 10, which gives the maximum number of particles allowed for one grid element. In our numerical tests, a linear function on $[G_{\text{mean}}, G_{\max}]$ has always been the choice for φ. In this work we confine ourselves to one- and two-dimensional cases, where an element of the grid is, respectively, an interval and a triangle, but the implementation in three-dimensional case and in other types of grids is absolutely analogous.

Considering convection–reaction subproblem (*) on the mth time step, we should implement the adaptivity both at the initial stage of the time step, based on grid-solution $u^{(m-1)}$ and reaction rate K, and during the computation on the region. Thus, after the initial adaptivity on the mth step, the particle set will contain (some of) the grid points and points added by the prior knowledge of the subproblem. Since we are typically interested in sharp fronts due to the reaction, the gradient-based signal function G is used. The function's value at new particles is interpolated by the projection procedure to be discussed below on the $m = 2, \ldots, M$ time steps; at the initial, $m = 1$, time step, the value is given by the initial condition function.

The adaptivity during the computation supplements new particles in the region of their lack, developed due to convective transport of the particle system. The nodes and elements of the grid

DOI: 10.1002/fld

in this region form the basis for creation of new inflow particles: the nodes lying on the inflow boundary Γ_{in} become new particles and receive the value of the boundary condition function u_{in}, the elements are adapted by the algorithm discussed above. The inflow adaptivity has to take into account the possible singularities developing due to different reaction rates in this particle-free region. Thus, the elements that contain boundary nodes and, moreover, lie on the discontinuity lines of the reaction rate function require special care; more particles, e.g. $N_{\max}$, should be generated on them. These elements, called *inflow edges* in the following, can be found and marked *a priori*, i.e. on the initial step, $t=0$, by using the information from the set

$$S=\{\mathbf{x}\in\Omega|K \text{ is discontinuous at } \mathbf{x}\}$$

Thus, in the beginning of the simulation an integer array Q is constructed by the rule

$$Q(i)=\begin{cases}1 & \text{if the } i\text{th element is connected with } \Gamma_{\text{in}}\\ 2 & \text{if the } i\text{th element is an inflow edge}\\ 0 & \text{otherwise}\end{cases}$$

Then the inflow adaptivity on the mth time step uses the information from Q as follows.

- If $Q(i)=1$, add $\varphi(G_i)$ new particles on the ith element;
- If $Q(i)=2$, add $N_{\max}$ new particles;
- If $Q(i)=0$, add $N_{\text{add}}=\min\{N_{\max}-N_{\text{current}},\varphi(G_i)\}$,

where the parameters $N_{\max}$, N_{current} and $\varphi(G_i)$ are determined by the algorithm for adaptivity.

The values u of the solution function associated with new particles, introduced by the inflow adaptivity, are also interpolated with the projection procedure to be discussed below. As an alternative, they might also be computed by the reaction term using the corresponding time shift, determined by the entry moments of the particles.

3.2. Convection–reaction by PTM

In the framework of the PTM, we consider the convection–reaction problem as a pair of ordinary differential equations. Using the Lagrangian notation of the full derivative Equation (5) can be rewritten in the form

$$\frac{\mathrm{d}u}{\mathrm{d}t}=Re(u,\mathbf{x},t) \tag{7}$$

This equation means that the reaction applies to each fluid particle (infinitely small fluid volume) individually. Hence, it must be integrated along the particles trajectories, defined by

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}=\mathbf{v}(\mathbf{x},t) \tag{8}$$

If one considers (7) along characteristics this equation appears as a (non-linear) ODE of the first order. The initial condition is given by u_0, u_{in} provides influx for the fluid particles coming into Ω through Γ_{in}. Note that in the absence of the inflow boundary conditions, Equations (7) and (8) may be solved simply as an ODE system by any suitable ODE solver.

DOI: 10.1002/fld

Thus on each time step, all the particles are assigned the values of function u, computed on the previous time step, at the respective locations. To find the new particle positions, one has to solve (8) on the time interval $[t_{m-1}, t_m]$, $m = 1, 2, \ldots,$ with the initial condition at the moment t_{m-1}

$$\mathbf{X}(t_{m-1}) = \mathbf{X}^0 \tag{9}$$

where $\mathbf{X}^0$ is the vector of coordinates of particles, created at the initial stage of the mth time step.

Once the new position of the particles, $\mathbf{X}(t_m)$, is found at the time instant t_m, it is necessary to determine novel values of the function u, which is subjected to reaction. Hence, one has to solve (7) on the time interval $[t_{m-1}, t_m]$, $m = 1, 2, \ldots,$ with the initial condition at the moment t_{m-1}

$$u(\mathbf{X}, t_{m-1}) = u^{(m-1)} \tag{10}$$

where $u^{(m-1)}$ is the value of the function u computed on the previous time step ($u^{(0)} = u_0$).

Systems (8), (9) and (7), (10) may be solved by any ODE solver, e.g. in case of a stiff reaction system by stiff solvers. In the numerical tests, we have always used the Runge–Kutta solver for both systems. The order of the method accuracy is varied from the third to the fifth one in our tests.

3.3. Projection

The projection procedure of the proposed technique consists of two main ingredients: a simple linear interpolation—as an alternative to the 'radial basis functions' approximation typical for meshless methods (see [28]) or L_2-projection typical for the mesh-based methods (see [24])—and the global discrete coordinate system (GDCS). This combination yields a computationally cheap technique, which exhibits good mass conservation, to transfer data from the particle system onto the grid, as demonstrated by numerical tests described below. The interpolation is performed sequentially for each grid node on the minimal triangle, formed by particles closest to the node. Creating additional global coordinates allows us to optimize the procedure of searching of a minimal triangle and to achieve real inter-independence of the particles, as far as there is no need for an ordered storage of them. Since for each grid node the neighbouring particles are known by GDCS in advance, the complexity of the projection procedure is $O(n)$, where n is the number of grid nodes. The detailed discussion of the projection technique can be found in [21].

3.4. Diffusion by FEM

Finite-element methods are widely used in computational fluid dynamics, see, for example, [29–31]. Here, we shall present only basic steps of the FE approximation.

Supposing that the computational domain Ω is presented in the form of triangular mesh Ω_h, a solution of a diffusion equation in the framework of the FE discretization is introduced in the following form:

$$u(\mathrm{x}, t) = \sum \widehat{u}_i(t)\varphi_i(\mathrm{x}) \tag{11}$$

where $\widehat{u}_i(t)$ is a nodal or element value of the approximated solution to be found at time instant t, $\varphi_i(\mathrm{x})$ is a shape function.

Substitution of (11) into (6), multiplication by a test function ϕ_j and integration of the obtained equality over the domain Ω gives the weak formulation of the diffusion problem, which in matrix

DOI: 10.1002/fld

notation is formulated as

$$\mathbf{M}\frac{\mathrm{d}\widehat{u}}{\mathrm{d}t} + \mathbf{K}\widehat{u} = \mathbf{F} \tag{12}$$

If the test functions equal the shape functions, $\phi_i = \varphi_i$ (Galerkin weighting), then

$$M_{ij} = \int_\Omega \phi_j \phi_i \, \mathrm{d}\Omega$$

$$K_{ij} = \int_\Omega \nabla\phi_j \cdot (D\nabla\phi_i) \, \mathrm{d}\Omega + \int_{\Gamma_{\text{out}}} \alpha\phi_j \phi_i \, \mathrm{d}\Gamma$$

$$F_j = \int_{\Gamma_{\text{out}}} u_{\text{out}}\phi_j \, \mathrm{d}\Gamma$$

The solution of the diffusion problem is defined by the solution of ODE (12) with the appropriate initial condition at the initial time moment. This problem can be evaluated by either an explicit or an implicit solver; in our numerical experiments the set of MATLAB solvers for ordinary differential equations is used.

The semidiscretization presented by (12) can be also achieved by means of the finite differences. In our numerical tests the MOL with finite difference space discretization is used for a one-dimensional problem of chromatographic separation.

4. NUMERICAL EXPERIMENTS

In this section we consider five test problems: layered reaction, mixing of hot and cold fronts, convection–reaction and convection–diffusion of the Gaussian hill, and the chromatographic separation problem as an example of a full CDR system. The examples will allow us to evaluate the proposed method, especially with respect to artificial oscillations and preservation of the total mass.

The relative mass conservation error is measured as follows:

$$\varepsilon_{\text{m}} = \frac{|M_{\text{ex}} - M_{\text{ap}}|}{M_{\text{ex}}}$$

where the exact total mass, M_{ex},

$$\int_\Omega u^{\text{ex}}(\mathbf{x}, t) \, \mathrm{d}\mathbf{x}$$

is known in the examples, and the total approximate mass, M_{ap}, is computed by

$$\int_\Omega u^{\text{ap}}(\mathbf{x}, t) \, \mathrm{d}\mathbf{x} = \sum_{T_i \subset \Omega_h} \int_{T_i} u^{\text{ap}}(\mathbf{x}, t) \, \mathrm{d}\mathbf{x}$$

where $u^{\text{ex}}(\mathbf{x}, t)$ and $u^{\text{ap}}(\mathbf{x}, t)$ are, respectively, the exact and approximated solutions and T_i is an element of the grid Ω_h, characterized by the grid size h of the discretization.

DOI: 10.1002/fld

In the case of a smooth solution function (see the Gaussian pulse's case), we also monitor the error in the max, l_∞, norm and in the L_2-norm

$$\varepsilon_\infty = \max_{i=1,n} |u_i^{\text{ap}} - u_i^{\text{ex}}|$$

$$\varepsilon_2 = \sqrt{\sum_{i,j=1}^{n} h^2 (u_i^{\text{ap}} - u_i^{\text{ex}})^2}$$

where u_i^{ex} and u_i^{ap} are, respectively, the exact and approximated solutions in node i, n is the number of grid nodes, and h is the characteristic size of the grid.

An unstructured Delaunay triangulation to be used in two-dimensional numerical tests is generated by the PDE Toolbox of the software package MATLAB, while a uniform distribution of points is implemented in the one-dimensional case presented.

4.1. Experiment 1: convection–reaction problem

The first set of numerical experiments is devoted to the extension of the particle transport method on the convection problems with reaction. Here, we shall consider the problem of layered reaction to demonstrate the technique to resolve the singularities developed due to different reaction rates in the domain as well as the inflow boundary conditions. In addition, a problem with a smooth solution, Gaussian pulse, is solved to analyse the error of the approximation obtained by PTM; for this test the mass conservation error and the error in l_∞ are checked.

4.1.1. Layered reaction. The problem of layered reaction models, for example, flow of immiscible fluids, where each fluid has its own rate of reaction, and the system is described as follows:

$$u_t + (\mathbf{v} \cdot \nabla)u = -Ku \quad \text{in } \Omega \times [0, 2] \tag{13}$$

$$u(\mathbf{x}, 0) = u_0 = 0 \quad \text{in } \Omega \tag{14}$$

$$u(\mathbf{x}, t) = u_{\text{in}} = 1.0 \quad \text{in } \Gamma_{\text{in}} \times (0, 2) \tag{15}$$

where the space variable $\mathbf{x} = (x_1, x_2)$ belongs to the domain $\Omega = [0, 2.1] \times [0, 1.0]$, t varies in the time interval $[0, 2]$, and $\Gamma_{\text{in}} = \{\mathbf{x} | x_1 = 0.0\}$ is an inflow portion of the domain boundary. The convective velocity field is given as

$$\mathbf{v} = \{4x_2(1 - x_2), 0\}$$

Here, the reaction rate function K is a piecewise constant function in Ω (see also the sketch in Figure 1)

$$K(\mathbf{x}) = \begin{cases} 0.6 & \text{if } x_2 \in \left[0, \frac{1}{3}\right) \\ 1.6 & \text{if } x_2 \in \left[\frac{1}{3}, \frac{2}{3}\right] \\ 0.6 & \text{if } x_2 \in \left(\frac{2}{3}, 1\right] \end{cases}$$

DOI: 10.1002/fld

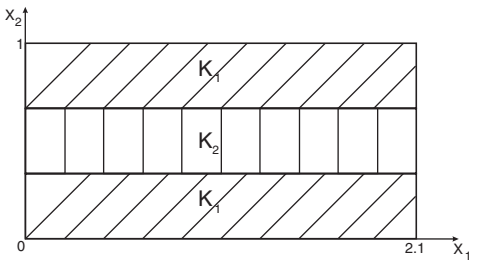

Figure 1. Layered reaction. Reaction rate function.

The PTM time step is chosen as $\Delta t = 0.1$. The movement of the particles and reaction are evaluated by the third-order Runge–Kutta method. The starting system of particles is a set of 1234 points non-uniformly distributed on Ω. An unstructured mesh with 10 146 nodes is used for the projection. Since the zero-initial condition is chosen and a free region develops near Γ_{in} due to convective transport of the particles, the adaptivity procedure concerns the inflow part only. Here, the discontinuous set of the reaction term is not empty,

$$S = \{\mathbf{x} \in \Omega | x_2 = \tfrac{1}{3} \text{ or } x_2 = \tfrac{2}{3}\}$$

thus, the inflow adaptivity should treat inflow edges in a special way, i.e. add more particles on them. The results of the sharp front inflow adaptivity after five time steps are presented in Figure 2. It can be seen that the addition of extra points on the inflow edges, see Figure 2(c) and (e), allows us to resolve the solution in the vicinities of high gradient without artificial smearing or oscillations. In this example, the interpolated values are used for the new inflow particles.

The particle distribution and the solution of the problem after 10, 15 and 20 time steps are presented in Figures 3 and 4. The inflow adaptivity based on u_{in} and the set S is implemented after each time step to avoid the artificial oscillations of the solution near the inflow portion of the boundary.

4.1.2. Convection–reaction of the Gaussian pulse. We consider the following problem:

$$u_t + (\mathbf{v} \cdot \nabla)u = -Ku \quad \text{in } \Omega \times [0, 2\pi] \tag{16}$$

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}) = \exp\left(-\frac{(x_1 - a)^2 + (x_2 - b)^2}{2\sigma^2}\right) \quad \text{on } \Omega \tag{17}$$

where $\mathbf{x} = (x_1, x_2) \in \Omega = [0, 1.0] \times [0, 1.1]$, $t \in [0, 2\pi]$. The initial condition function u_0 is a Gaussian hill centred at the point $(a, b) = (0.35, 0.5)$ and with standard deviation $\sigma = 0.06$, see Figure 5. The convective velocity field is given as

$$\mathbf{v} = \{x_2 - 0.5, -x_1 + 0.5\}$$

that is the rotation of the Gaussian pulse around the point $(0.5, 0.5)$. Reaction rate K is a function of t

$$K(t) = 3\cos(2t)$$

DOI: 10.1002/fld

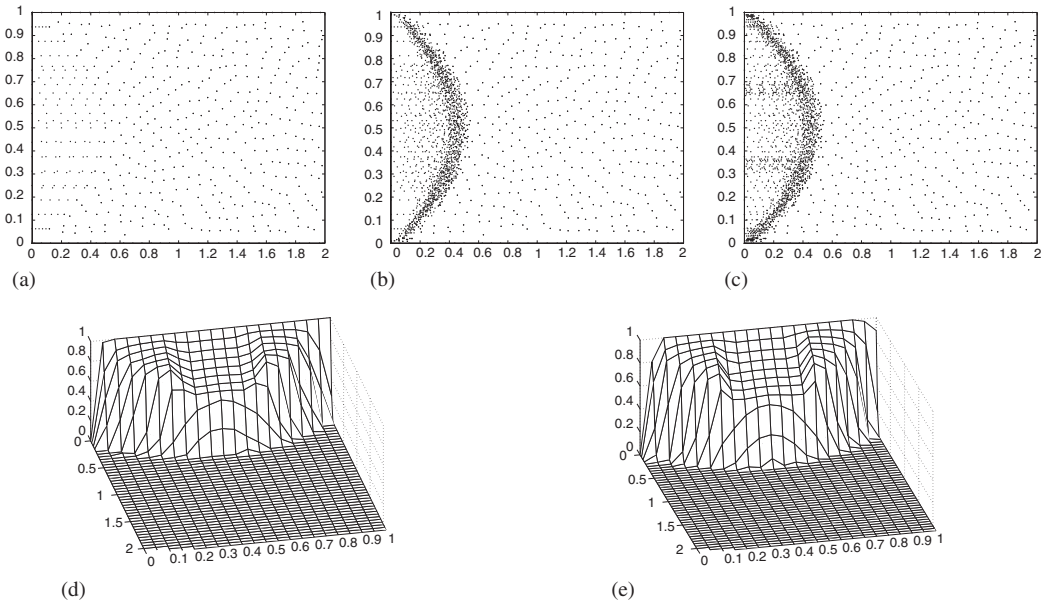

Figure 2. Layered reaction. Inflow adaptivity: (a) not adaptive supplement of new particles; (b) inflow adaptivity without taking into account the inflow edges; (c) inflow adaptivity based on the set S; (d) solution after five time steps without the inflow adaptivity; and (e) solution after five time steps with the inflow adaptivity.

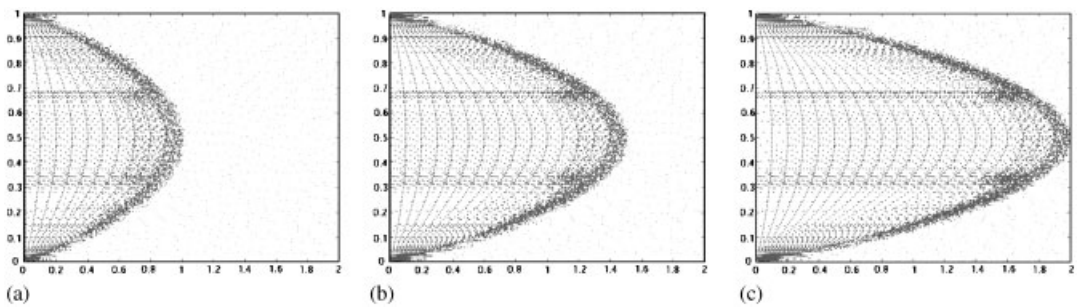

Figure 3. Layered reaction. Particle distribution: (a) $t = 1.0$; (b) $t = 1.5$; and (c) $t = 2.0$.

The simulation time step is chosen as $\Delta t = \pi/10$. The initial adaptivity with $N_{\max} = 4$ consists of both the smooth and sharp front procedures; their effect on the particles set can be seen in Figure 6. The movement of the particles and integration of the reaction term are accomplished by the fifth-order Runge–Kutta method. Figure 7 demonstrates the solution of the problem for an unstructured projection grid with 4381 nodes and an initial non-uniform particle set with 2228 points.

The error analysis was done with respect to the total number of nodes and particles. The relative error in l_∞-norm, ε_∞, and relative mass conservation error, ε_m, were checked. The results, presented in Table I and Figure 8, indicate that the particle transport method achieves a second-order accuracy

DOI: 10.1002/fld

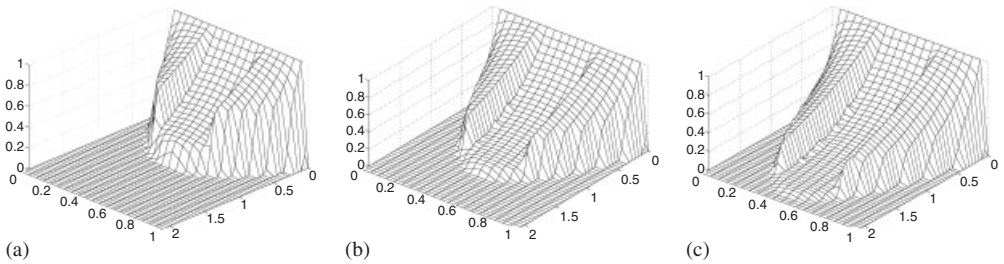

Figure 4. Layered reaction. Solution: (a) $t = 1.0$; (b) $t = 1.5$; and (c) $t = 2.0$.

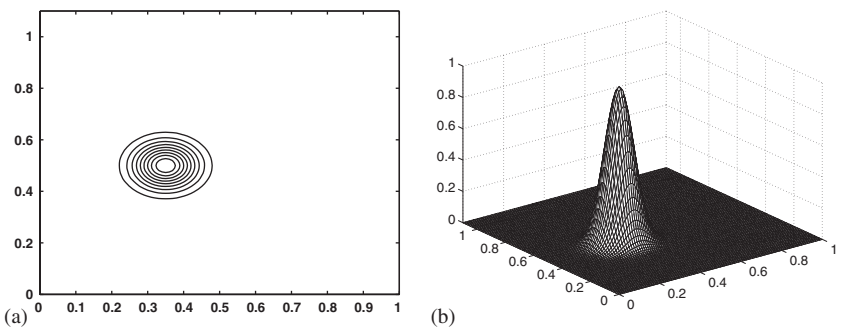

Figure 5. Convection–reaction of the Gaussian pulse: initial condition.

for mass conservation and in l_∞-norm with respect to $h = 1/\sqrt{n + N}$, where n is the number of nodes and N is the number of particles.

4.2. Experiment 2: convection–diffusion problem

The performance of the particle transport method in the framework of the operator-splitting technique for the convection–diffusion problems will be demonstrated in this subsection on the problem of the mixing of hot and cold fronts and convection–diffusion of the Gaussian pulse.

4.2.1. Mixing of hot and cold fronts. We consider a problem, suggested in [32], of mixing of hot and cold fronts on the domain $\Omega = \{x_1 \in [0, 8]\} \times \{x_2 \in [0, 8]\}$. Here, the initial temperature profile is a tangential function with the maximum value 0.964 on the edge $x_2 = 0$ and the minimum value -0.964 on $x_2 = 8$, see Figure 9,

$$u_0(\mathbf{x}) = -\tanh\left(\frac{x_2 - 4}{2}\right)$$

During the simulation the steep front between the minimum and maximum temperatures is twisted by the rotation velocity field

$$\mathbf{v} = \left(-\frac{T_v(x_2 - 4)}{r}, \frac{T_v(x_1 - 4)}{r}\right)$$

DOI: 10.1002/fld

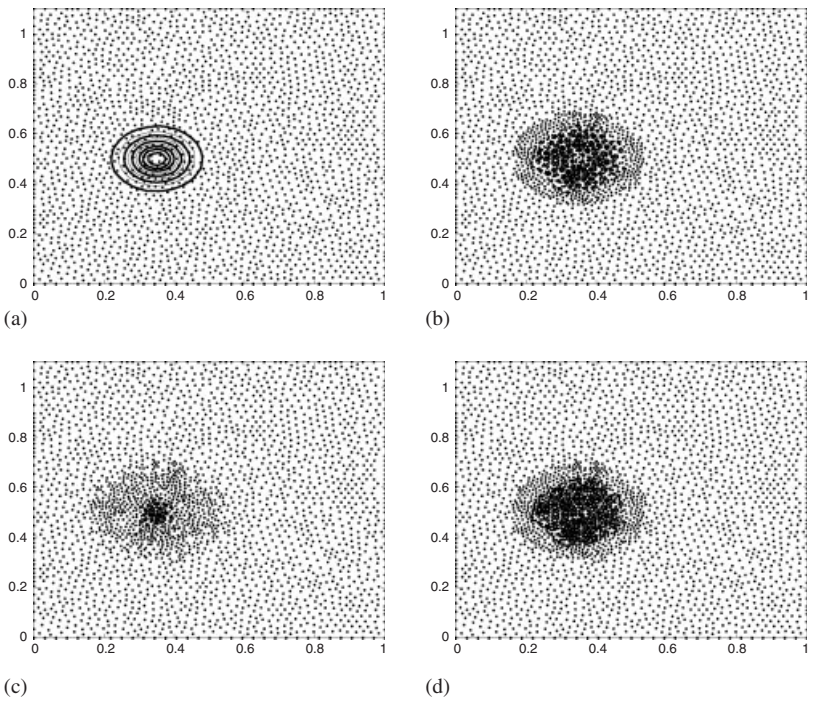

Figure 6. Convection–reaction of the Gaussian pulse. Initial adaptivity: (a) initial distribution and the contour plot of the initial condition; (b) sharp front adaptivity; (c) smooth adaptivity; and (d) 'sharp plus smooth' adaptivity.

Here, r is the distance between a point (x_1, x_2) and the rotation origin $(4, 4)$, the coefficient T_v is equal to the ratio $v_t/v_{t_{max}}$, where $v_t = \mathrm{sech}^2(r)\tanh(r)$ is a tangential velocity at (x_1, x_2) and $v_{t_{max}}$ is the maximum tangential velocity on Ω. We shall evaluate the problem with two values of diffusion coefficient, namely $D = 10^{-5}$ and 10^{-2}.

The time step of the computation is $\Delta t = 0.5$, $t \in [0, 10]$. The grid and the initial set of particles are non-uniform and have the same amount of points, namely 7105. At each time step, the particles are generated from the grid and adapted by the steep-front adaptivity with $N_{max} = 1$. The movement of the particles is approximated by the fifth-order Runge–Kutta method. The diffusion subproblem is resolved by the finite-element solver within MATLAB.

The distribution of the particles during the simulation for the case of small diffusion (convection–dominated case) is presented in Figure 10; it can be seen that new points are introduced in the vicinity of the developing vortex that helps to catch narrow structures around the rotation origin. In Figure 11 the numerical solutions obtained by PTM at $t = 0.5$, 5 and 10 are presented. Here, the adaptivity allows us to reach high resolution of the narrow steep fronts around the rotation origin on the relatively sparse grid.

The solution of the problem with a larger diffusion coefficient, $D = 10^{-2}$, is presented in Figure 12. Here, the fronts of different temperature are mixed both by the velocity field and by the dispersion process; thus the steep fronts diffuse in the middle of the domain.

DOI: 10.1002/fld

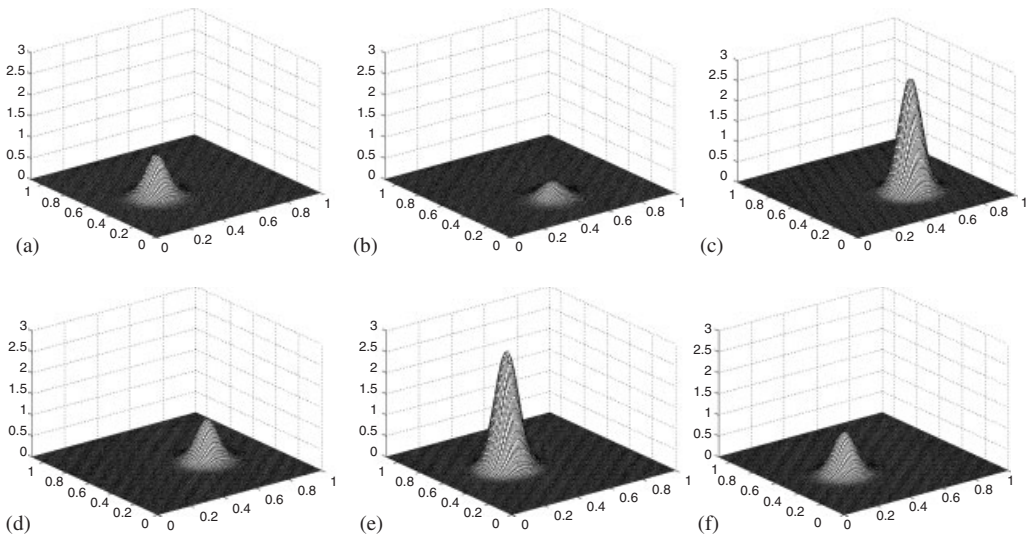

Figure 7. Convection–reaction of the Gaussian pulse. Solution: (a) $t = 0.0$; (b) $t = 1.5708$; (c) $t = 2.5133$; (d) $t = 3.1416$; (e) $t = 5.6549$; and (f) $t = 6.2832$.

Table I. Convection–reaction of the Gaussian pulse. Error analysis with respect to the total number of the grid nodes and particles. One full rotation, $\Delta t = \pi/10$.

# Particles	# Nodes	Sum #	ε_m	ε_∞
458	2228	2686	5.4e−03	2.37e−02
2843	4381	7224	2.2e−03	1.80e−02
3275	4381	7656	2.1e−03	9.50e−03
4076	8688	12 766	1.8e−03	1.01e−03
8688	8688	17 376	8.0e−04	6.70e−04

4.2.2. Convection–diffusion of the Gaussian pulse. For the sake of error analysis of the method we use the specification of a convection–diffusion problem and results, presented by Dong Liang *et al.* [10], for the fractional step central difference method (FS-CDM) and the fractional step upstream difference method (FS-USDM). The technique of fractional step is an efficient approach to solve multidimensional problems by means of splitting space dimensions onto the separate subproblems. The finite difference methods (USDM and CDM) are the classical solvers for convection–diffusion and diffusion-dominated problems. However, the domination of the convection term might lead to difficulties for these methods to approximate an accurate solution.

Here, the convection–diffusion of the Gaussian pulse with the standard deviation $\sigma = 0.1$ and centred at the point $(a, b) = (0.5, 0.5)$ is simulated on the domain $\Omega = [0, 2] \times [0, 2]$. The hill is convected by the constant velocity field $\mathbf{v} = (2, 2)$. The diffusion coefficient D is chosen as 10^{-3} and 10^{-1}. The Dirichlet inflow and Newmann outflow boundary conditions, respectively, on $\Gamma_{in} = [0; 2] \times \{x_2 = 0\} \cup \{x_1 = 0\} \times [0; 2]$ and $\Gamma_{out} = [0; 2] \times \{x_2 = 2\} \cup \{x_1 = 2\} \times [0; 2]$ are set

DOI: 10.1002/fld

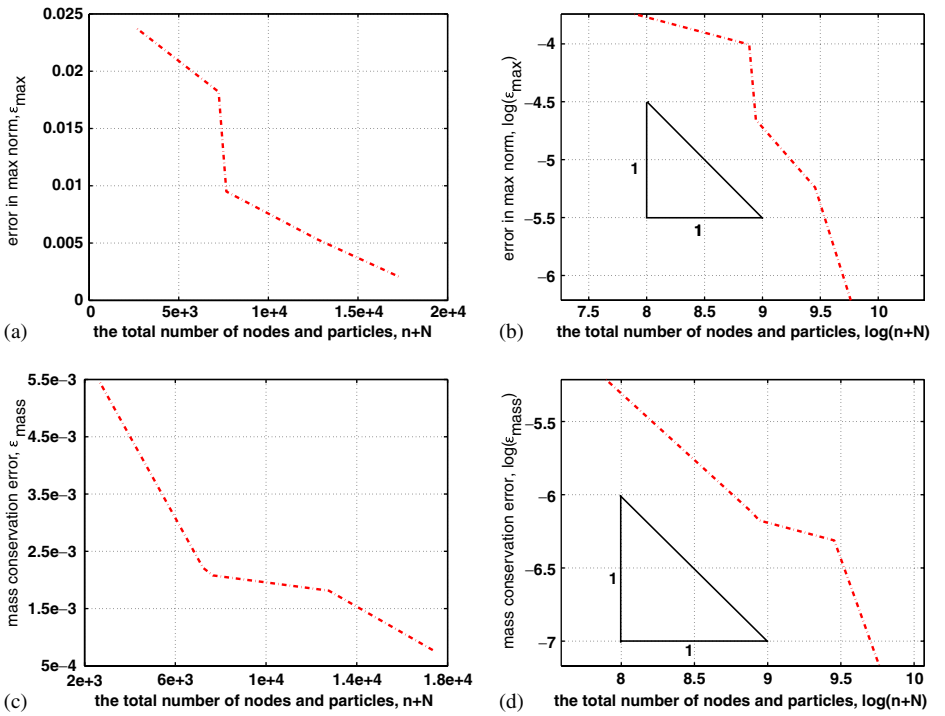

Figure 8. Convection–reaction of the Gaussian pulse, error analysis with respect to the total number of nodes and particles: (a) error in l_∞-norm, ε_∞; (b) error in l_∞-norm, ε_∞, logarithmic scale; (c) relative mass conservation error, ε_m; and (d) relative mass conservation error, ε_m, logarithmic scale.

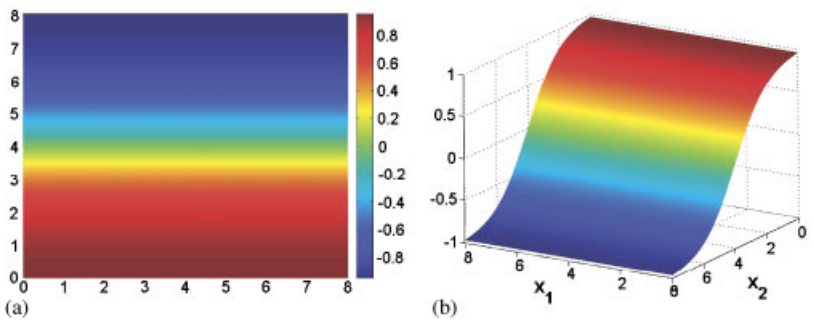

Figure 9. Mixing of hot and cold fronts. Initial condition.

from the exact solution of the problem

$$u(\mathbf{x}, t) = \frac{\sigma^2}{s(t)} \exp\left(-\frac{(x_1 - a - v_1 t)^2 + (x_2 - b - v_2 t)^2}{2s(t)^2}\right)$$

where $s(t) = \sqrt{(\sigma^2 + 2Dt)}$.

DOI: 10.1002/fld

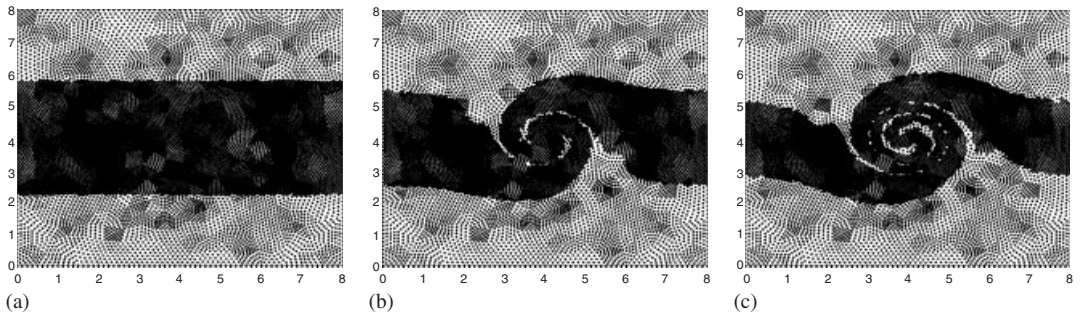

Figure 10. Mixing of hot and cold fronts. Particle distribution, $D = 10^{-5}$: (a) $t = 0.5$; (b) $t = 5$; and (c) $t = 10$.

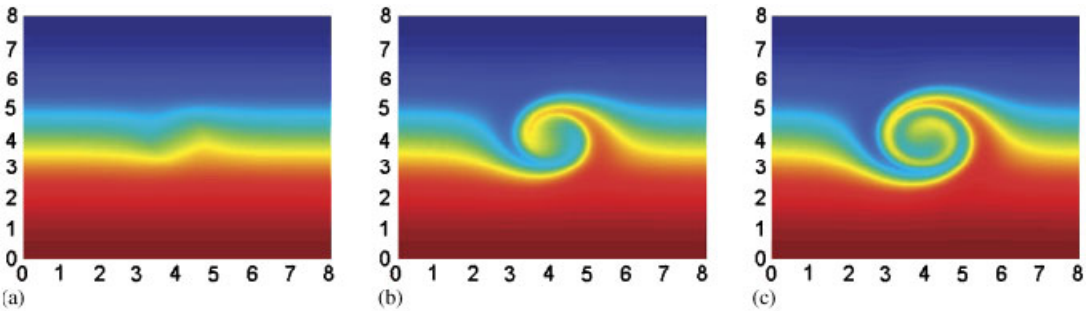

Figure 11. Mixing of hot and cold fronts. Solution obtained by PTM, $D = 10^{-5}$: (a) $t = 0.5$; (b) $t = 5$; and (c) $t = 10$.

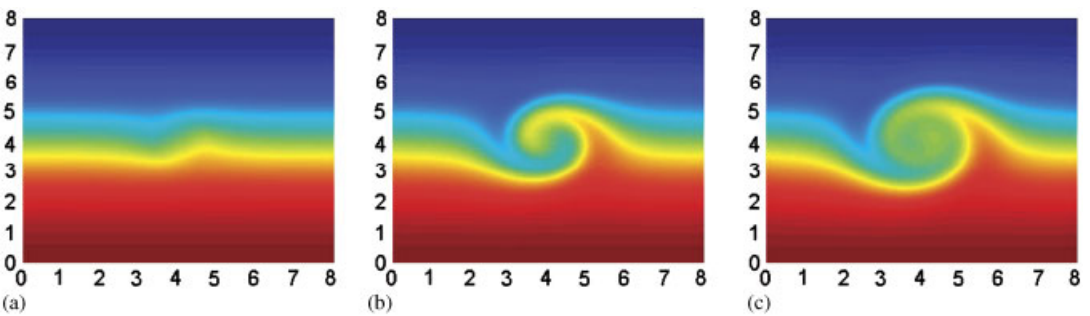

Figure 12. Mixing of hot and cold fronts. Solution obtained by PTM, $D = 10^{-2}$: (a) $t = 0.5$; (b) $t = 5$; and (c) $t = 10$.

The simulation is continued till the time $T = 0.6$ with the time step $\Delta t = 0.1$. To accomplish the comparison of the results, the number of nodes, n, of the computational grid for PTM was approximately equal to the number of degrees of freedom, $1/h^2$, used for FS-CDM and FS-USDM, i.e. the number of discretization points in x_1- and x_2-dimensions, where h is the step

DOI: 10.1002/fld

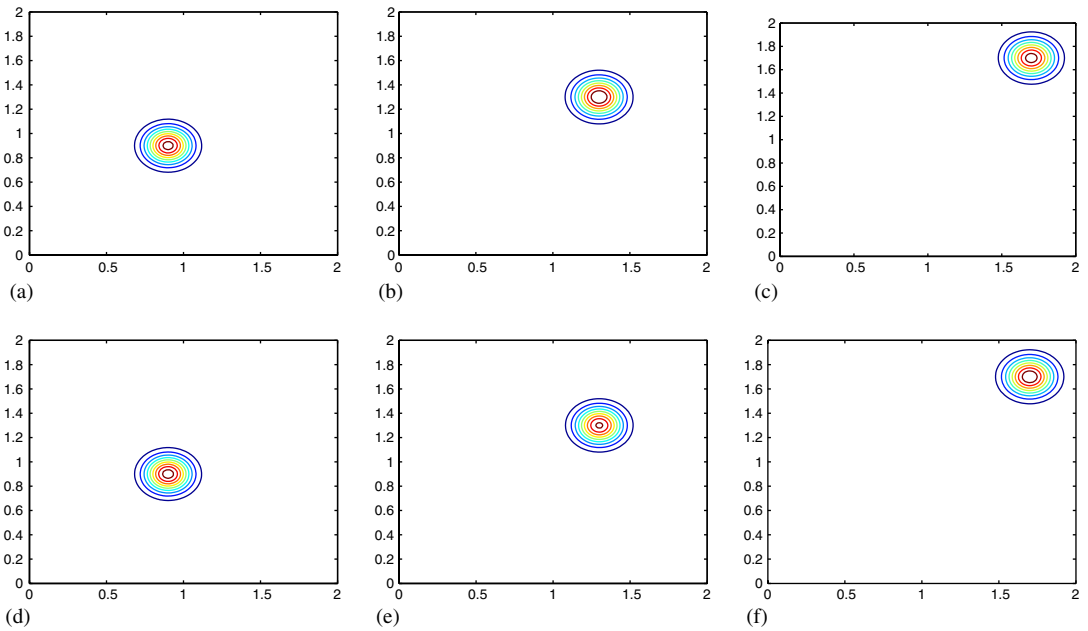

Figure 13. Convection–diffusion of the Gaussian pulse, $D = 10^{-3}$. Numerical and exact solution on the grid with 10 145 nodes, $\Delta t = 0.1$: (a) PTM, $t = 0.2$; (b) PTM, $t = 0.4$; (c) PTM, $t = 0.6$; (d) exact, $t = 0.2$; (e) exact, $t = 0.4$; and (f) exact, $t = 0.6$.

Table II. Convection–diffusion of the Gaussian pulse, $D = 10^{-3}$: error analysis.

		PTM			FS-CDM	FS-USDM
error	t	$n = 665$	$n = 2577$	$n = 10\,145$	$h = 1/100$	$h = 1/100$
ε_∞	0.2	1.0109e−01	3.5716e−02	7.9107e−03	8.5363e−01	8.5826e−01
	0.4	2.3885e−01	7.2807e−02	2.1919e−02	8.7061e−01	8.7301e−01
	0.6	3.0581e−01	1.1488e−01	3.4336e−02	8.5529e−01	8.5693e−01
ε_2	0.2	6.5643e−03	1.8941e−04	4.8726e−04	1.4700e−01	1.4813e−01
	0.4	1.7847e−02	5.0664e−03	1.3835e−03	1.5561e−01	1.5621e−01
	0.6	2.1245e−02	6.9166e−03	1.9332e−03	1.5580e−01	1.5625e−01

of discretization. Thus, the grid in PTM numbering 10 145 nodes corresponds to the fractional dimensions step $h = \frac{1}{100}$; this grid is used as a basis for the construction of the adaptive particles system with $N_{\max} = 1$ at each time step.

The numerical solution for the case of $D = 10^{-3}$ obtained by PTM on the 10 145-node grid is shown in Figure 13 as well as the exact solution on the same grid; it can be seen that the approximation evaluated by the proposed technique demonstrates a good accordance with the reference solution. The error magnitudes in l_∞- and L_2-norms, presented in Table II in

DOI: 10.1002/fld

Table III. Convection–diffusion of the Gaussian pulse, $D = 10^{-1}$: error analysis.

		PTM			FS-CDM	FS-USDM
error	t	$n = 665$	$n = 2577$	$n = 10\,145$	$h = 1/100$	$h = 1/100$
ε_∞	0.2	1.1739e−01	1.2746e−01	1.3185e−01	1.2670e−01	1.2857e−01
	0.4	2.7657e−02	3.0650e−02	3.1401e−02	7.2594e−02	7.3658e−02
	0.6	1.9055e−02	2.0748e−02	2.1275e−02	5.0645e−02	5.1397e−02
ε_2	0.2	1.4338e−02	1.5203e−02	1.5521e−02	4.2802e−02	4.3498e−02
	0.4	5.3882e−03	5.6843e−03	5.7882e−03	3.2558e−02	3.3158e−02
	0.6	2.5127e−03	2.5366e−03	2.5141e−03	2.4578e−02	2.5028e−02

comparison with those of FS-CDM and FS-USDM, indicate that for the convection-dominated case, $D = 10^{-3}$, PTM achieves better results already on the grid with 665 nodes (in terms of FS-CDM and FS-USDM it corresponds to $h \approx \frac{3}{100}$). For a larger diffusion coefficient, $D = 10^{-1}$, see Table III, all three methods demonstrate nearly the same values of errors; however, PTM needs only 2577 ($h \approx \frac{2}{100}$) points, while FS-CDM and FS-USDM work with twice many points. In addition, it is worth noticing that PTM demonstrates nearly the same results in the error for both convection-dominated and convection–diffusion problems.

4.3. Experiment 3: chromatographic separation

Here, we simulate chromatographic separation to consider a system of CDR equations. Two mixed substances are injected with appropriate solvents into a column, which can be described as a one-dimensional domain due to dominance of its length. The time of the injection is usually very short in comparison to the whole simulation period; thus in the beginning (the left end) of the domain the mixture forms a pulse that moves and gradually separates with time, due to different absorption properties of the components.

The chromatographic separation can be described by a number of numerical models (see, for instance, [33–35]); in this work we consider the following CDR system:

$$\begin{aligned}
\frac{\partial C_1}{\partial t} &= -v\frac{\partial C_1}{\partial x} + D\frac{\partial^2 C_1}{\partial x^2} - p_1(C_1^e - Q_1) \quad \text{on } \Omega \times [0, 400] \\
\frac{\partial C_2}{\partial t} &= -v\frac{\partial C_2}{\partial x} + D\frac{\partial^2 C_2}{\partial x^2} - p_2(C_2^e - Q_2) \quad \text{on } \Omega \times [0, 400] \\
\frac{\partial Q_1}{\partial t} &= p_3(C_1^e - Q_1) \quad \text{on } \Omega \times [0, 400] \\
\frac{\partial Q_2}{\partial t} &= p_4(C_2^e - Q_2) \quad \text{on } \Omega \times [0, 400]
\end{aligned} \tag{18}$$

Here C_i, $i = 1, 2$, are the concentrations of substances in a mixture to be separated, Q_i, $i = 1, 2$, are the absorbed concentrations, v is the convective velocity of the flow in the computational domain $\Omega = [0, 50]$, D is the diffusivity coefficient, and p_i, $i = \overline{1, 4}$, are the absorption parameters

DOI: 10.1002/fld

of the components. The term C_i^e is an equilibrium isotherm of the ith component given by $C_i^e = e_i \max\{C_i, 0\}^\alpha$, α defines the profiles formed by separated substances; we shall consider the case $\alpha = 0.7$. The parameters of the problem are chosen as $v = 0.6$, $p_1 = 1.302$, $p_2 = 1.0016$, $p_3 = 1.3$ and $p_4 = 1$, $e_1 = 2.4337$ and $e_2 = 0.9230$.

The initial conditions for the ith component, $i = 1, 2$, are

$$\begin{aligned} C_i(x, 0) &= 0 \quad \text{on } \Omega \\ Q_i(x, 0) &= 0 \quad \text{on } \Omega \end{aligned} \tag{19}$$

and boundary conditions for the ith component, $i = 1, 2$, are

$$\begin{aligned} C_i(0, t) &= \begin{cases} 0.03 & \text{if } t \leqslant 100 \\ 0 & \text{otherwise} \end{cases} \\ \frac{\partial C_i(100, t)}{\partial x} &= 0 \quad \text{for } t \in [0, 400] \end{aligned} \tag{20}$$

For the numerical simulation the time step is $\Delta t = 0.5$, the grid is the set of 200 points uniformly distributed on Ω. On each mth time step, $[t_{m-1}, t_m]$, $m = \overline{1700}$, system (18)–(20) is divided into two subproblems in the framework of the proposed method: convection–absorption and diffusion.

At each time step particle systems are generated from the grid for all components, i.e. for the concentration functions C_1, Q_1, C_2 and Q_2, to resolve convection and absorption parts of the problem. Each set is adapted with the steep front adaptivity procedure on the basis of the grid-value function of the respective concentration, while $N_{\max} = 10$. The movement of the particles and the reaction of function values carried out by them are integrated by the fourth-order Runge–Kutta method. To accomplish the next step—diffusion—the concentrations of the components C_1 and C_2 are projected from the particles onto the grid, where the MOL is applied. For the sake of comparison, whole CDR system (18)–(20) was also solved by only MOL with the same problem and discretization parameters.

In [21] it was demonstrated how in the case of the convection domination PTM resolves the discontinuities defined by the initial conditions with comparison to MOL. Here, the sharp fronts develop during the simulation due to the inflow conditions and the absorption term. Figure 14 presents the solutions of the problem at $t = 100$, 200 and 300 with the diffusion coefficient $D = 10^{-5}$. The solution obtained by the combination of PTM and MOL in the framework of the operator splitting is not contaminated by any artificial oscillations as it can be seen in Figure 14(a)–(c). The method of lines, see Figure 14(d)–(f), demonstrates the generation of non-physical negative concentration of C_1 in the case of small diffusion coefficient. The concentration of the injected substances with respect to time is shown in Figure 15. Here, for the case of $D = 10^{-5}$, the solution obtained by PTM, Figure 15(a), achieves the equilibrium between time 100 and 300 and then the concentration of the second component ($C_2 + Q_2$) decreases while the first component ($C_1 + Q_1$) still saves the mass; this means that the part of the separated mixture containing C_1 and Q_1 is still in the separation column while the part with C_2 and Q_2 reaches its right end. The concentration equilibrium approximated by MOL, Figure 15(b), demonstrates the artificial increasing of the mass for the first component while the mass of the second one is decreasing. To achieve the high-quality resolution of the steep fronts for the considered problem, MOL needs at least three times more grid nodes. To avoid the artificial growth of the mass in this case, a non-negativity constraint can also be applied to the function value within the MOL

DOI: 10.1002/fld

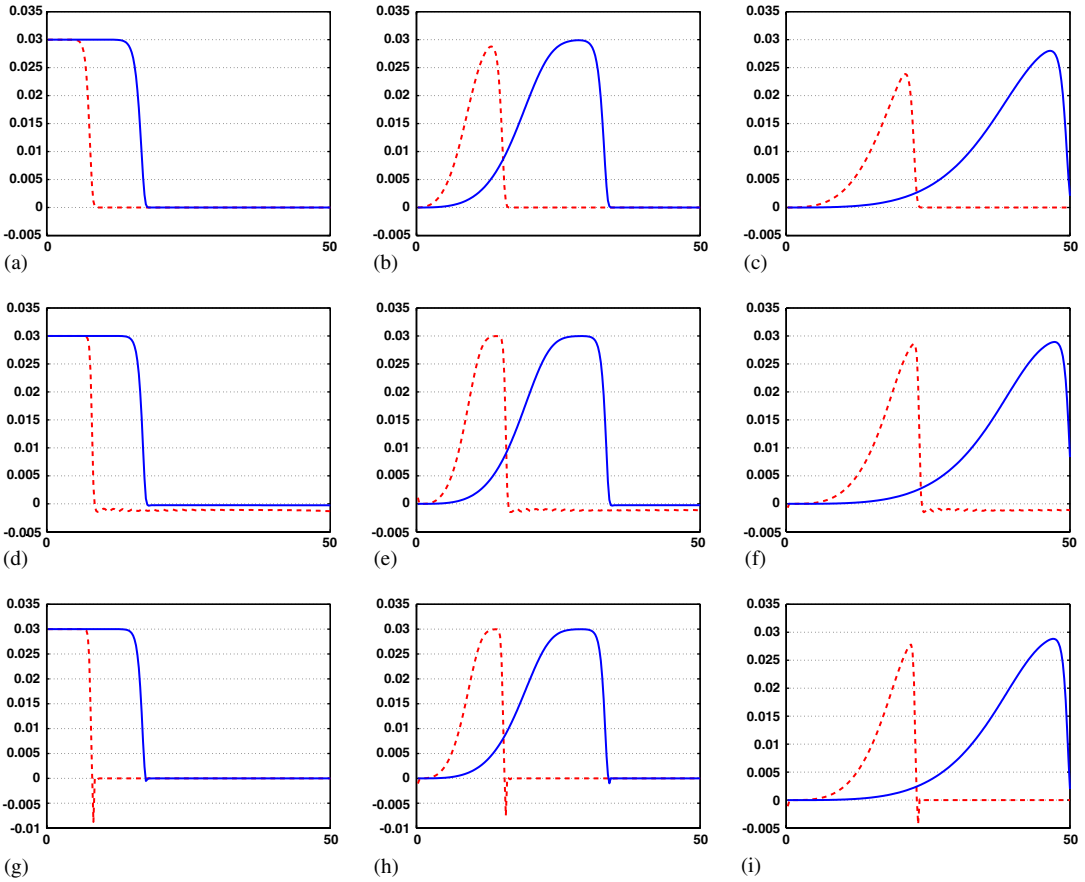

Figure 14. Chromatographic separation problem, 200—node grid: the dashed line—C_1 concentration; the solid line—C_2 concentration: (a) PTM + MOL, $t = 100$; (b) PTM + MOL, $t = 200$; (c) PTM + MOL, $t = 300$; (d) MOL, $t = 100$; (e) MOL, $t = 200$; (f) MOL, $t = 300$; (g) MOLs, $t = 100$; (h) MOLs, $t = 200$; and (i) MOLs, $t = 300$.

algorithm. The results of the improved MOLs are demonstrated in Figures 14(g)–(i) and 15(c); though concentrations reach the equilibrium in the time interval [100, 300], spurious oscillations in the vicinity of steep fronts of C_1 and C_2 are now generated. It is worth noticing that in the framework of PTM no special restrictions on the solution value is applied and the prevention of oscillations is reached only by the implementation of the spatial adaptivity. This procedure allows us to avoid the numerical oscillation even for a very sparse grid ($\sim$ 50 nodes); however, in this case the numerical diffusion due to linear interpolation within the projection increases. For the relatively large diffusion, $D = 10^{-1}$, both methods show good quality of approximation on the same grids.

In addition, the execution time of the particle transport method in comparison to the method of lines was checked on the zero-diffusion case that gives the results almost identical to the ones for $D = 10^{-5}$. Here the simulation interval is [0, 250] with $\Delta t = 0.5$. We vary the number of nodes for

DOI: 10.1002/fld

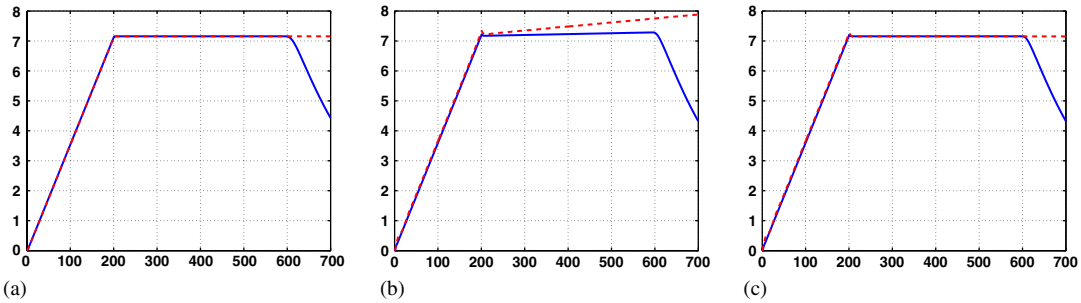

Figure 15. Chromatographic separation problem, the concentration equilibrium: the dashed line—$C_1 + Q_1$ concentration; the solid line—$C_2 + Q_2$ concentration: (a) PTM + MOL; (b) MOL; and (c) MOLs.

Table IV. Chromatographic separation problem. Computational time (s) for PTM and MOL with respect to the size of the computational grid, $\Delta t = 0.5$, $t \in [0, 250]$.

# Nodes	40	100	200	300	400	500
PTM	29.63	29.94	30.66	31.06	32.83	33.09
MOLs	20.11	29.8746	63.672	78.77	108.39	151.32

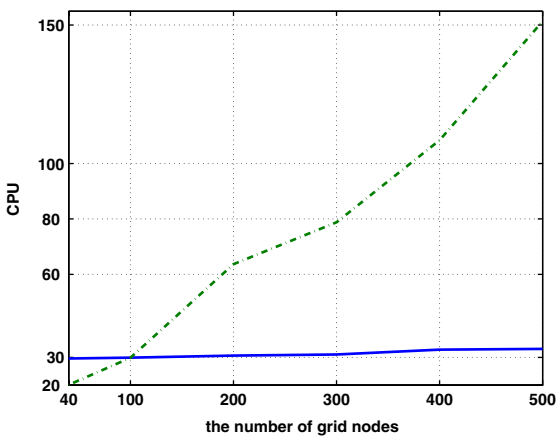

Figure 16. Computational time for chromatographic separation problem, $\Delta t = 0.5$, $t \in [0, 250]$: the solid line—PTM; the dashed line—MOL.

the computational grid within MOL that corresponds to the fixed grid used to save the information on Q_1 and Q_2 within PTM. The convection of C_1 and C_2 in PTM, as well as the absorption, is evaluated on the adapted particle set that initially consists of 40 points; the functions Q_1 and Q_2 are projected from the fixed grid onto the particle system and back at each time step. Table IV and Figure 16 show that the CPU time of PTM changes from 29 to 33 s, while the CPU time of MOL grows from 20 to 151 s. The small variation in the execution time of PTM is explained by the fact

DOI: 10.1002/fld

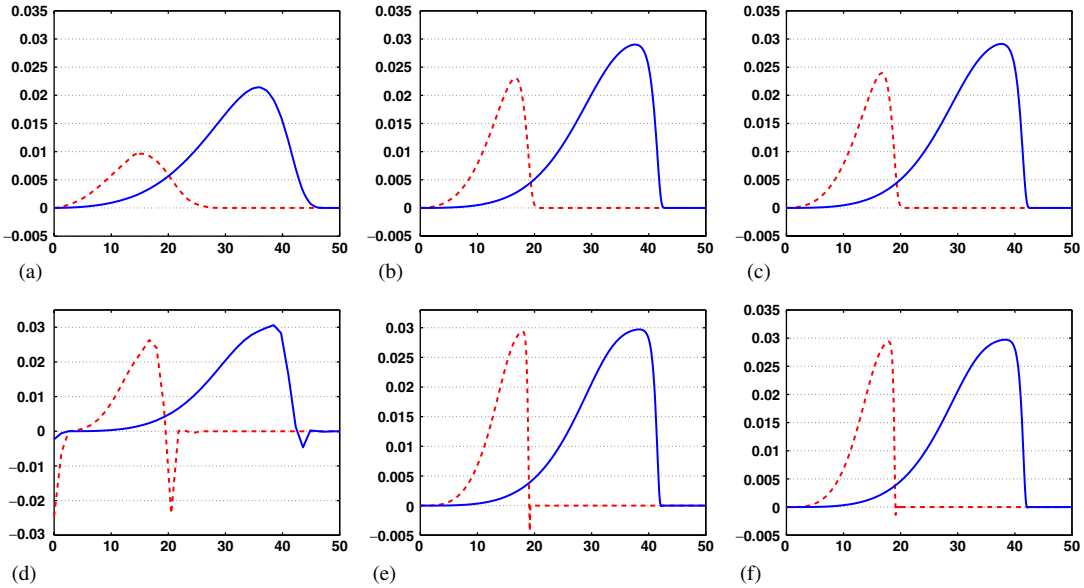

Figure 17. Chromatographic separation problem, solution at $t = 250$ on different grids: the dashed line—C_1 concentration; the solid line—C_2 concentration: (a) PTM, 40 nodes; (b) PTM, 400 nodes; (c) PTM, 500 nodes; (d) MOLs, 40 nodes; (e) MOLs, 400 nodes; and (f) MOLs, 500 nodes.

that the evaluation of the ODE systems (the most time-consuming procedure) is only performed on the adapted particle set, the size of which is unchanged. The increase of the number of grid nodes used as a storage for the functions Q_1 and Q_2 only means the growth of the CPU time for the projection procedure that has an optimal complexity and takes about 20% of the whole computational time. It is worth noting that although for the smallest size of the grid (40 nodes) MOL works faster, large oscillations are generated in the vicinities of the steep front, while PTM approximation is free of artificial oscillations, see Figure 17(a) and (d). However, PTM smooths excessively the numerical solution on the coarse mesh. The increase of the number of grid nodes allows us to reduce the numerical diffusion in PTM and spurious oscillations in MOL, but it leads to the drastic change in MOL execution time. On the 400-node grid, where both methods give already close results (further increasing gives almost the same solutions, see Figures 17(b), (c), (e) and (f)), MOL is three times slower than PTM. Thus, we can conclude that for convection dominated cases when the diffusion term can be dropped out, the particle transport method can be successfully applied as an accurate and fast scheme. That is important, for instance, for problems of parameter estimations, where a large number of the simulations are usually required.

5. CONCLUSIONS

In the paper, an explicit Eulerian–Lagrangian technique, the particle transport method (PTM), was extended to the case of a convection–diffusion–reaction problem. In the framework of the operator splitting the convection–reaction and diffusion subproblems are solved separately, where for the

DOI: 10.1002/fld

first one the PTM is used and the second is resolved by the mesh-based scheme, namely finite-element method or the method of lines. The projection between a moving system of particles of convection–reaction solver and the fixed grid of the diffusion one is accomplished by a procedure based on linear interpolation. The possible singularities due to convection and reaction term are treated with a high quality due to the spatial adaptivity within PTM. Such a combination allows us to obtain a monotone method that exhibits a good mass conservation and accuracy on unstructured grids; the numerical comparison shows that the proposed scheme can demonstrate better results than some of the Eulerian methods based on structured grids.

Our current research is focused on combining results and solving such a problem as a gas bubble dissolution in a viscous flow that presents a combination of incompressible Navier–Stokes and CDR equations. Here the proposed technique can be used for the mass-transfer part of problem as well as for the resolution of the movement of the interface between gas and liquid fractions, described by convection of a level set function. We also suppose that the method can be very efficient for the simulation of the atmospheric front movement and of the chemical crystallization.

ACKNOWLEDGEMENTS

The authors thank the anonymous referee for pointing out the particle-in-cell method to us.

REFERENCES

1. Kröner D. *Numerical Schemes for Conservation Laws*. Wiley/Teubner: New York, Stuttgart, 1966.
2. Finlayson BA. *Numerical Methods for Problems with Moving Fronts*. Ravenna Park Publishing: Seattle, 1992.
3. LeVeque RJ. *Numerical Methods for Conservation Laws*. Birkhäuser: Basel, 1992.
4. Löhner R. *Applied CFD Technique*: *An Introduction Based on FEM*. Wiley: New York, 2002.
5. Donea J, Huerta A. *Finite Element Methods for Flow Problems*. Wiley: Chichester, 2003.
6. Celia MA, Russell TF, Herrera I, Ewing RE. An Eulerian–Lagrangian localized adjoint method for the advection–diffusion equation. *Advances in Water Resources* 1990; **13**:187–206.
7. Healy RW, Russell TF. A finite-volume Eulerian–Lagrangian localized adjoint method for solution of the advection–dispersion equation. *Water Resources Research* 1993; **29**:2399–2414.
8. Dahle HK, Ewing RE, Russell TF. Eulerian–Lagrangian localized adjoint methods for a nonlinear convection–diffusion equation. *Computer Methods in Applied Mechanics and Engineering* 1995; **122**:223–250.
9. Wang H, Ewing RE, Qin G, Lyons SL, Al-Lawatia M, Man S. A family of Eulerian–Lagrangian localized adjoint methods for multi-dimensional advection–reaction equations. *Journal of Computational Physics* 1999; **152**:120–163.
10. Dong L, Chuanbin D, Hong W. A factional step ELLAM approach to high-dimensional convection–diffusion problems with forward particle tracking. *Journal of Computational Physics* 2006; doi:10.1016/j.jcp2006.06.022.
11. Morton KW, Priestley A, Suli E. Stability of the Lagrange–Galerkin method with non-exact integration. *Mathematical Modelling and Numerical Analysis* 1998; **22**:625–653.
12. Priestley A. Exact projection and the Lagrange–Galerkin method: a realistic alternative to quadrature. *Journal of Computational Physics* 1994; **112**:547–571.
13. Crowley WP. FLAG: a Free-Lagrangian method for numerical simulating hydrodynamics flows in two dimensions. *Lecture Notes in Physics*, vol. 8. Springer: New York, 1971.
14. Oliviera A, Fortunato AB. Toward an oscillation-free, mass-conservative, Eulerian–Lagrangian transport method. *Journal of Computational Physics* 2002; **183**:142–164.
15. Iske A, Käser M. Conservative semi-Lagrangian advection on adaptive unstructured meshes. *Numerical Methods for Partial Differential Equations* 2004; **20**:388–411.
16. Varoglu E, Finn WDL. Finite elements incorporating characteristics for one-dimensional diffusion–convection equation. *Journal of Computational Physics* 1980; **34**:371–379.
17. Douglas Jr J, Russell TS. Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures. *SIAM Journal on Numerical Analysis* 1982; **19**:871–875.

DOI: 10.1002/fld

18. Birdsall CK, Langdon AB. *Plasma Physics via Computer Simulation.* McGraw-Hill: New York, 1985.
19. Jacobs GB, Hesthaven JS. High-order nodal discontinuous Galerkin particle-in-cell method on Unstructured Grids. *Journal of Computational Physics* 2006; **214**:96–121.
20. O'Ruorke PJ, Brackbill JU, Larrouturou B. On particle-grid interpolation and calculating chemistry in particle-in-cell methods. *Journal of Computational Physics* 1993; **109**:37–52.
21. Smolianski A, Shipilova O, Haario H. A fast high-resolution algorithm for linear convection problems: particle transport method. *International Journal for Numerical Methods in Engineering*, doi: 10.1002/nme.1899, in press.
22. Breitkopf P, Huerta A. *Meshfree and Particle Based Approaches in Computational Mechanics*. ISTE Publishing Company, 2004.
23. Li S, Liu WK. *Meshfree Particle Methods*. Springer, 2004.
24. Hansbo P. A free-Lagrangian finite element method using space-time elements. *Computer Methods in Applied Mechanics and Engineering* 2000; **188**:347–361.
25. de Sampaio PAB, Lyra PRM, Morgan K, Weatherill NP. Petrov–Galerkin solutions of the impressible Navier–Stokes equations in primitive variable with adaptive remeshing. *Computer Methods in Applied Mechanics and Engineering* 1993; **106**:143–178.
26. Löhner R. Robust vectorized search algorithms for interpolation on unstructured grid. *Journal of Computational Physics* 1996; **118**:380–387.
27. Marchuk GI. Spliting and alternating direction methods. *Handbook of Numerical Analysis I*. North-Holland: Amsterdam, 1990; 197–462.
28. Behrens J, Iske A. Grid-free adaptive semi-Lagrangian advection using radial basis functions. *Computers and Mathematics with Applications* 2002; **43**:319–327.
29. Zienkiewicz OC, Morgan K. *Finite Elements and Approximation.* Wiley: New York, 1983.
30. Reddy JN. *An Introduction to the Finite Element Method.* McGraw-Hill: New York, 1984.
31. Zienkiewicz OC, Taylor RL. *The Finite Element Method*, *Volume 3*: *Fluid Dynamics.* Butterworth Heinemann: Oxford, 2000.
32. Tamamidis P, Assanis DN. Evaluation for various high-order-accuracy schemes with and without flux limiters. *International Journal for Numerical Methods in Fluids* 1993; **16**:931–948.
33. Lode F, Houmard M, Migliorini C, Mazzotti M, Morbidelli M. Continuous reactive chromatography. *Chemical Engineering Science* 2001; **56**:269–291.
34. Klatt K-U, Hanisch F, Dfinnebier G, Engell S. Model-based optimization and control of chromatographic processes. *Computers and Chemical Engineering* 2000; **24**:1119–1126.
35. Teoh HK, Turner M, Titchener-Hooker N, Sorensen E. Experimental verification and optimisation of a detailed dynamic high performance liquid chromatography column model. *Computers and Chemical Engineering* 2001; **25**:893–903.

DOI: 10.1002/fld