INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2006; **51**:331–350
Published online 6 December 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1132

An efficient 3D particle transport model for use in stratified flow

J. W. Stijnen*,†, A. W. Heemink‡ and H. X. Lin§

Delft University of Technology, Delft Institute of Applied Mathematics, Mekelweg 4, 2600 GA, The Netherlands

SUMMARY

Random-walk models are a versatile tool for modelling dispersion of both passive and active tracers in turbulent flow. The physical and mathematical foundations of stochastic Lagrangian models of turbulent diffusion have become more and more solid over the years. An important aspect of these types of models that has not received much attention is the behaviour of the particles near boundaries. Often, a simple stochastic, numerical scheme is used. Because turbulent mixing in the vertical direction is much more complicated than in the two horizontal directions, it is in the vertical direction that a simple numerical scheme, such as the Euler scheme, may cause problems.

In this paper our main goal is the development of an efficient 3D particle transport model that can be used in stratified flow. For this type of situation the vertical direction is of special interest. First, a closer look is taken at some considerations that should be regarded when choosing a numerical scheme. Specifically schemes are investigated that can be used in the vertical direction, where the diffusion coefficient is varying in that direction. Experiments are performed regarding the accuracy of different numerical schemes in various situations. The behaviour of the particles near an impermeable layer interface is investigated. The stochastic Heun and Runge–Kutta schemes turn out to be very attractive for this type of model.

For the simulation of the transport of various physical quantities, such as salinity, heat, silt, oxygen, or bacteria, different types of models are available. In this case we will take a closer look at the modelling of the transport of pollutants from point sources (either instantaneous or continuous transport). For this purpose a 3D particle transport model has been developed that is especially suited for stratified situations such as can be found in estuaries. The main idea is to use a simple numerical scheme for the horizontal directions and a higher-order method for the vertical direction. The results play an important role in making specific choices for this type of particle transport model. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: particle transport model; higher order; numerical method; stratification

*Correspondence to: J. W. Stijnen, HKV consultants, Botter 11-29, 8203 AC, Lelystad, The Netherlands.
†E-mail: j.w.stijnen@hkv.nl
‡E-mail: A.W.Heemink@ewi.tudelft.nl
§E-mail: H.X.Lin@ewi.tudelft.nl

Contract/grant sponsor: Dutch National Institute for Marine and Coastal Management (RIKZ)

Received 3 August 2005
Revised 14 October 2005
Accepted 20 October 2005

1. INTRODUCTION

The behaviour of particles in turbulence has been studied for many years, i.e. in meteorology References [1–5], and hydrology [6–9]. Much research has been carried out to calculate the Lagrangian trajectories of these particles, ranging anywhere from highly idealized flow to situations as complex as the unstable convective boundary layer. The level of understanding in these types of models has greatly increased over the years. The behaviour of the ensemble average concentration of a passive tracer is quite well understood, and the amount of knowledge about concentration fluctuations from two-particle (or more) models has grown as well [1, 10–12]. But, however advanced these models are, limited attention is paid to the numerical part. Since these models are all part of the family of stochastic differential equations, and cannot be solved exactly due to their complexity and nonlinearity, they all require (in some form or another) a stochastic numerical scheme to simulate them.

There are basically two ways to look at the movement of a group of particles. The Eulerian way describes what happens at a fixed point (or region) in space. This allows the observation of phenomena at a specific location, and results in a deterministic advection–diffusion type partial differential equation (PDE). The Lagrangian method follows the particles through space at every time step, and allows the observation of fluctuations in the path of the particles. The Lagrangian approach results in a stochastic differential equation (SDE), describing the behaviour of the particles. The latter one is used many times in this article. Some basic properties, characteristics and background information of this approach are briefly discussed below.

In many cases, the Lagrangian alternative to a Eulerian partial differential equation is available. By interpreting the concentration $C(t,x)$ of particles as the probability $P(t,x)$ that a particle ends up at location x at time t (released at x_0 at time t_0), we can write the 1D advection–diffusion equation as

$$\frac{\partial P}{\partial t} + u\frac{\partial P}{\partial x} = D\left(\frac{\partial^2 P}{\partial x^2}\right) \tag{1}$$

$$P(t_0,x) = \delta_0(x - x_0) \tag{2}$$

where D is the diffusion coefficient, u represents a velocity, and δ_0 stands for the Dirac delta. This last equation is an example of a so-called Fokker–Planck equation (see Reference [13]). In more general terms, this equation describes the propagation of the transition probability density function (PDF) $P = P(t,x;t_0,x_0) = P(X_t = x|X_{t_0} = x_0)$ of the stochastic process X_t according to

$$\begin{aligned} \frac{\partial P}{\partial t} &= -\frac{\partial}{\partial x}(f(t,x)P) + \frac{1}{2}\frac{\partial^2}{\partial x^2}(g^2(t,x)P) \\ \lim_{t\downarrow t_0} P(t,x;t_0,x_0) &= \delta(x - x_0) \end{aligned} \tag{3}$$

We see that the Fokker–Planck equation is an advection–diffusion type differential equation that describes the evolution of the conditional probability density function. It is known that the Lagrangian model that corresponds to Equation (3) is the following Itô stochastic differential

equation (SDE):

$$\begin{aligned} \mathrm{d}X_t &= f(t,X_t)\,\mathrm{d}t + g(t,X_t)\,\mathrm{d}\beta_t \\ X_0 &= x_0 \end{aligned} \tag{4}$$

where f and g are arbitrary functions and $\mathrm{d}\beta_t$ stands for an incremental Brownian motion process. The solution of the Fokker–Planck equation is the conditional probability density function associated with the solution, X_t, of the stochastic differential equation (4). Note that the SDE (4) must be interpreted in the Itô sense if it is to be consistent with the Fokker–Planck equation (3). The partial differential equation (3) is completely consistent with the Lagrangian description of Equation (4). It gives a complete description of variations in time and space of the PDF. Details can be found in Reference [14], or Reference [5], for example.

As mentioned before, in order to solve these SDEs, we need to resort to numerical methods. Note that the term 'numerical method' in this and subsequent cases refers to the stochastic version of these schemes. For ease of use and simplicity the most favourable of these schemes is the Euler scheme. It is cheap, it is simple to implement, and it can give reasonable results.

One of the major drawbacks of the Euler scheme, however, is that it has very poor convergence behaviour [15]. This implicitly means that in order to get accurate results, a small time step is required, and therefore many calculations. Another drawback of the Euler method is that it has unwanted behaviour in the neighbourhood of boundaries. Because the scheme is so inaccurate, a time step that is too large will result in particles that unintentionally cross a boundary.

In a hydrodynamical model for example, this could mean that particles end up on land, pass through the water surface, or disappear into the bottom. This sort of behaviour is undesired, and should be kept to a minimum. Of course, when a particle moves across a boundary, we could simply move it back to its old position and draw a new random number. That, however, would change the nature of the model, because we would favour a certain direction that the particle should take over others. Instead, what is sometimes done, is that the time step is halved, and a new random number is drawn. The original time step is split in two, letting the particle travel two short time steps instead of a single big one. This process is repeated until the particle does not pass the boundary anymore. The result is that the particle bends along a certain boundary (see Figure 1), but never crosses it. Physically speaking, this means that the diffusion coefficient approaches zero the closer one gets to a closed boundary.

Consequently, this means that near boundaries, detailed calculations are done to keep all particles inside the model. If this needs to be done for many particles, the number of computations near boundaries might increase drastically. The situation sketched above occurs near physical boundaries, but for a vertical dispersion model in stratified flow this problem becomes also visible in other ways. They can have different layers, each with its own characteristics (salinity, density, temperature). When particles are released in one layer, they are not allowed to cross to another layer. The problem in such a case is similar to the one described above, but more subtle because there is no solid boundary, only an 'imaginary' one.

Luckily, a wide variety of stochastic schemes other than the Euler scheme is available, varying from fairly simple to extremely complex, that may improve the above-mentioned disadvantages of the Euler scheme. In the next section we will focus on various numerical approximations to SDEs, and look specifically at numerical schemes that can be used instead

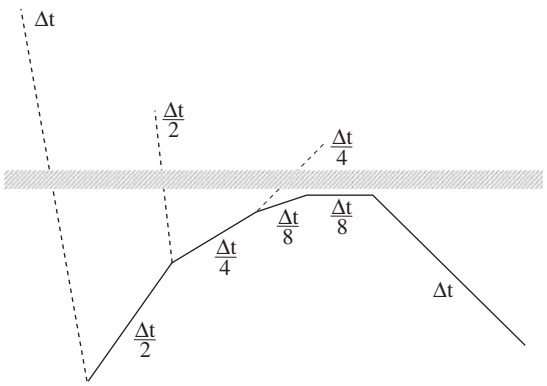

Figure 1. Example of the movement of a particle that bends along the coast.

of the Euler scheme in order to make more accurate, and more efficient computations. Later in this article (see Section 3), results from the next section will be taken into consideration in the development of a numerical scheme in the vertical direction of the 3D particle transport model.

2. BOUNDARY TREATMENT IN PARTICLE MODELS

In this paper we specifically focus on the behaviour of particles near (solid) boundaries. To this end we have investigated a variety of numerical methods for SDEs. These can roughly be categorized as follows:

- Explicit schemes, contained in this category are for example the one-step Euler or Milstein scheme, a predictor–corrector method such as Heun, or a multi-step method such as the Runge–Kutta scheme.
- Implicit schemes, such as the implicit Euler, and implicit Milstein schemes.
- Semi-implicit schemes, that lie somewhere in between the explicit and fully implicit methods. The trapezoidal scheme, for example, belongs to this category.
- Extrapolation methods make use of lower order methods such as the Euler or Milstein schemes to obtain a higher order approximation. An example of such a method is Richardson extrapolation.
- Stochastic Taylor expansions.

The Euler and Milstein scheme are both members of this last category as well. The development of higher-order schemes based on Taylor expansion require more and more information about the derivatives and are not discussed further. The differences between various numerical methods become clearer with the following example. It is inspired by the application in Section 1 of this paper. Results from the example are the motivation for the actual numerical scheme that was implemented in the 3D particle transport model, which will be discussed later in this paper (Section 3).

Let us take a closer look at a simplified situation. Consider the diffusion equation in a one-dimensional channel with infinite depth, but with spatial variations in the diffusion coefficient

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial x} D(x) \frac{\partial C}{\partial x} \qquad C(t_0, x) = \delta_0(x - 1) \tag{5}$$

where $C(t,x)$ is the concentration, $D(x)$ the diffusion coefficient, t stands for time, x for position, and δ_0 the Dirac delta. There is an SDE that reveals the same behaviour for the particles as this equation. The PDF, $P(t,x)$, of the particles describes the density of particles at a certain location, which is just the concentration $C(t,x)$ of the particles. With the connection through the PDF we therefore have consistency with the following random-walk model:

$$\mathrm{d}X_t \overset{\text{Itô}}{=} \frac{\mathrm{d}D}{\mathrm{d}x}\,\mathrm{d}t + \sqrt{2D}\,\mathrm{d}\beta_t \tag{6}$$

$$X_{t_0} = 1 \tag{7}$$

Notice the advective term in the particle model, even though this term is not (directly) present in the diffusion equation. The term is sometimes called the *correction term for spatial variations in diffusion*. In order to get a more specific SDE, we can choose a different diffusion coefficient, for instance one equal to $D(X_t) = X_t^2$ $(X_t \geqslant 0)$:

$$\mathrm{d}X_t \overset{\text{Itô}}{=} 2X_t\,\mathrm{d}t + \sqrt{2}X_t\,\mathrm{d}\beta_t \tag{8}$$

By choosing the initial position equal to $X_{t_0} = 1$, we see that the exact solution starts positive, and remains so

$$X_t = \mathrm{e}^{t+\sqrt{2}\beta_t} \tag{9}$$

As long as the initial condition is positive, the entire solution process is positive as well. Note that when $X_t \to 0$, the diffusion will go to zero as well. We can see the line $X_t = 0$ as an impermeable layer that the particles should not cross. When simulating sample paths from the exact solution (9), we can see that it is impossible for a particle to cross the layer. Simulating trajectories numerically, on the other hand, can lead to problems. Due to an unfavourable realization of the random numbers, in combination with a time step that is too large, a particle may cross the boundary.

Of course this behaviour is unwanted, especially in a three-dimensional particle model, where this may not only occur at the bottom (or surface) of the flow, but also in the middle of stratified flow, or in the neighbourhood of the physical boundaries of the model. A solution that is sometimes used to solve this was described in Section 1. When a particle crosses the boundary, it is moved back to its old position. Drawing a new random number would mean that we change the nature of the PDF, because we reject a certain choice for the random numbers. Instead, the time step is halved, and a new random number is drawn. This process is repeated, so that the particle bends along the boundary (see Figure 1), but never crosses it.

Since the size of the time step often becomes small if particles are near boundaries, the amount of computations increases. It would be advantageous in such cases to minimize the number of particles that crosses the boundaries in the first place. The use of a higher-order

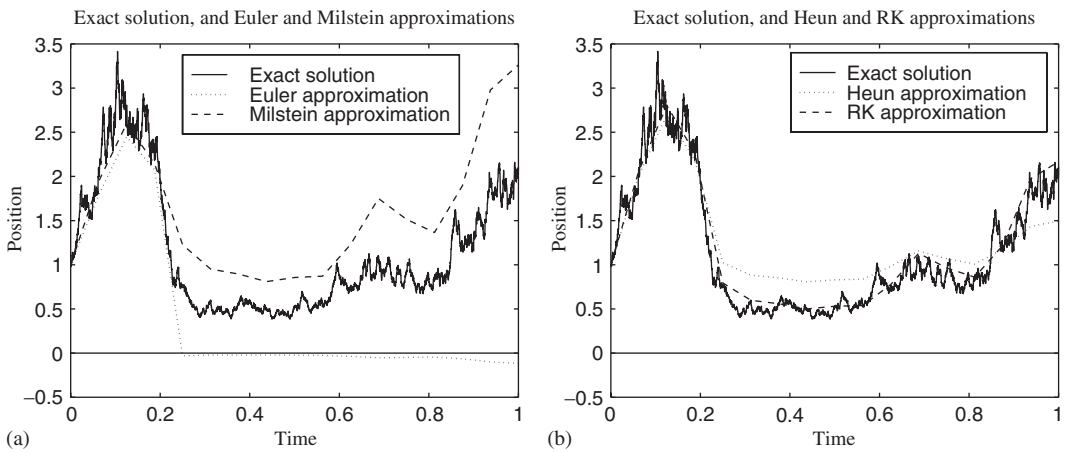

Figure 2. (a) An example of a single particle track in a one-dimensional channel with an impermeable layer located at $x = 0$. The spatially varying diffusion coefficient is equal to $D(X_t) = X_t^2$. The exact solution is generated using 2^{12} time steps, while both the Euler and Milstein schemes use only 2^7 time steps. (b) The same example, only with the Heun and Runge–Kutta schemes. The SDE was first transformed.

numerical scheme may perhaps not completely prevent this from happening, but it will at least reduce the number of times that it does happen.

To illustrate the effect of the numerical scheme, we approximate the solution of Equation (8) with several numerical schemes. The simplest scheme is the explicit Euler scheme

$$\begin{aligned} Y_{n+1} &= Y_n + 2Y_n\Delta t + \sqrt{2}Y_n\Delta\beta_n \\ Y_0 &= 1 \end{aligned} \tag{10}$$

where Δt is the discrete time step for the numerical calculation, and $\Delta\beta_n$ is the Brownian motion increment associated with this time step. In order to generate numerical solutions we divide the interval $[0, 1]$ in 2^{12} steps, and calculate a sample Brownian motion process for these fine time steps ($\mathrm{d}t$). We use those to calculate an 'exact' solution with the aid of Equation (9), which will be our reference solution. On a computer it is of course impossible to generate an exact solution, since this is only obtained for $\mathrm{d}t \downarrow 0$. We use the same set of random numbers to calculate the corresponding random increments that are needed on the coarser time steps ($\Delta t = 2^{-7}$) in the numerical scheme (10). This means that in the numerical schemes there are 2^5 of the small time steps $\mathrm{d}t$ between each evaluation at the coarser time steps Δt ($2^{12}/2^7 = 2^5$).

Some examples of possible realizations in the one-dimensional channel are shown in Figures 2 and 3. For clarity, results for only a single particle are plotted. In each approximation, as well as in the exact solution, the same set of random numbers is used. The results from the schemes are quite far apart, ranging from extremely poor (implicit Euler) to surprisingly good (Runge–Kutta). Notice how the particle using the implicit Euler scheme crosses the layer, while the one using the implicit Milstein scheme does not. The version of the trapezoidal scheme (a semi-implicit scheme with a 50–50% weighting factor between the

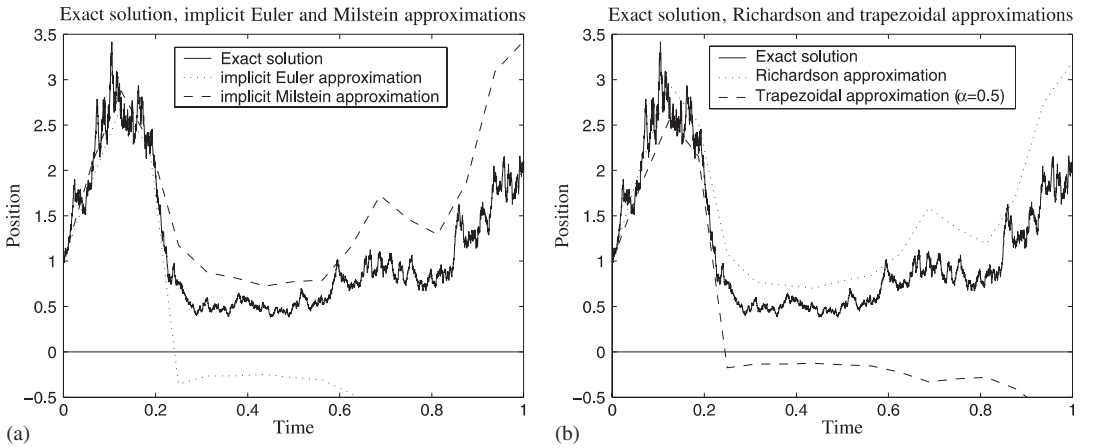

Figure 3. (a) An example of a single particle track in a one-dimensional channel with an impermeable layer located at $x = 0$. The spatially varying diffusion coefficient is equal to $D(X_t) = X_t^2$. The exact solution has been generated using 2^{12} time steps, while both the implicit Euler and Milstein schemes use only 2^7 time steps. (b) The same example, only with the Richardson extrapolation and a semi-implicit trapezoidal scheme ($\alpha = 0.5$).

explicit and fully implicit schemes) that was used, produces results somewhere in the middle of the results for the explicit and implicit Euler schemes. Of the schemes that are based on the Euler scheme, the Richardson scheme is the only one that performs significantly better than the Euler scheme itself.

Before using the Heun and Runge–Kutta scheme, we must first transform the SDE from Itô to Stratonovich (see for example Reference [16]). Otherwise the solutions that these two numerical schemes produce will not be the ones that we are interested in. The transformed SDE looks as follows:

$$\mathrm{d}X_t \overset{\text{Strat}}{=} X_t\,\mathrm{d}t + \sqrt{2}X_t\,\mathrm{d}\beta_t \tag{11}$$

$$X_{t_0} = 1 \tag{12}$$

This test case is a rather extreme situation: the numerical schemes try to approximate the solution on a very coarse grid, namely once every $2^7 = 128$ time steps. What it shows, is that the numerical scheme has a considerable impact on the movement of a particle.

It is clear that any improvement over the Euler scheme is welcome, because it is the only scheme (explicit, implicit, or a combination) where the particle actually crosses the boundary. The other schemes are bound to give more accurate results, faster convergence, and better behaviour near boundaries. Especially the Runge–Kutta scheme looks promising for this specific case. The results will improve when using smaller time steps, although the crossing of particles across the invisible boundary may never be avoided completely, due to unfavourable increments.

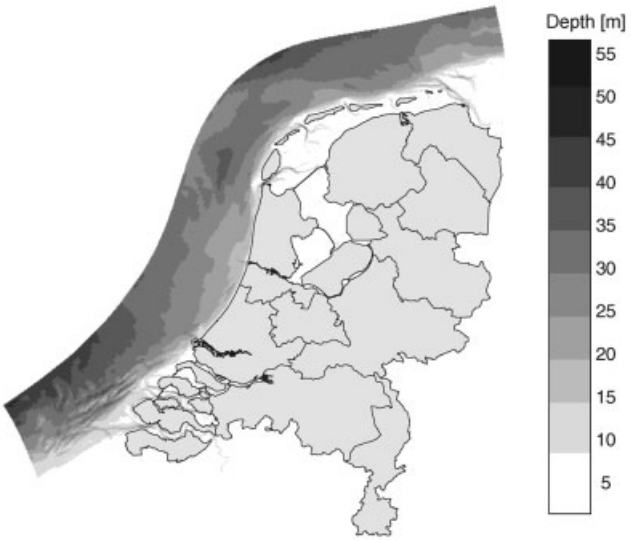

Figure 4. The area of interest is the band of shallow water in front of the Dutch coast.

3. A 3D PARTICLE MODEL FOR THE DUTCH COASTAL ZONE

The coastal zone of The Netherlands is very sensitive to disturbances in its ecosystem. Therefore, it is important to be able to take protective action in case of pollutant calamities. Think for example of a ship, or a factory leaking chemicals. The resulting transport of pollutants to a certain area may result in damage to, or even the extinction of marine life. A prediction model for such a situation would be a big help in determining what measures should be taken. Especially the Wadden Sea is an area in The Netherlands with a delicate balance between different ecological systems. The geographical area of the Dutch coastal zone is made up of a complex whole of rivers, estuaries, and tidal inlets that is connected to a shallow sea. The model area is a band perpendicular to the coast with a width of approximately 80 km. A picture of the area is shown in Figure 4. The depth of that part of the North Sea is generally not more than 45 m. Because of the many fresh-water rivers discharging into the North Sea, and the number of estuaries that are the result of this, these areas are of specific interest.

These estuaries are generally semi-enclosed bodies of water, connected to the sea, within which sea water is measurably diluted by fresh water. Interaction of two chemically and physically different water masses gives rise to complex sedimentological and biological processes and patterns. The most fundamental characteristic of an estuary is the interaction between salt and fresh water. This leads to a common classification by salinity structure relating to the degree of separation or mixing of the two water masses. Three different mixing regimes are identified: stratified, partially mixed and well-mixed or homogeneous. In this section we are primarily interested in the stratified case, where the fresh and salt water masses remain distinct. Typically, when a fresh water river discharges into a saline sea, a salt wedge is formed at the interface where the two bodies of water meet (see Figure 5).

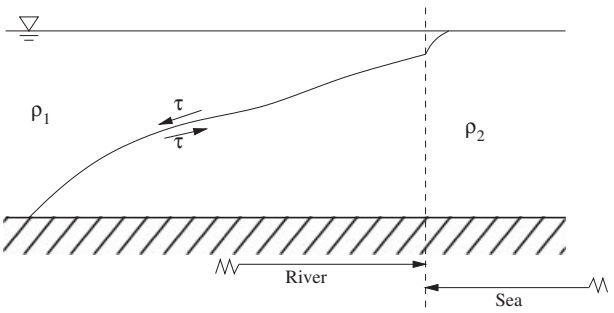

Figure 5. An example of a 'static' salt wedge in an estuary, where ρ_1 is the density of fresh water and ρ_2 is the density of salt water, τ is the tension along the wedge.

The more dense sea water meets the river water and intrudes along the bottom under the fresh water. The length of the intruding wedge is determined by the resulting equilibrium between on the one hand, the friction along the interface, and on the other hand, the pressure gradient resulting from the inclination of the interface. When this equilibrium is well established, the salt wedge will be in a stable position with the fresh water flowing seaward on the surface and spreading out in a thin surface layer at sea. A pollutant that is released in either one of those layers is likely to stay within that particular layer. Of course there will be transport at molecular level across the interface, but the bulk of the pollutant will stay in the original layer. Particle models should reflect this behaviour as well, unfortunately with a rudimentary numerical scheme such as the Euler scheme this is quite cumbersome and requires a very small time step.

A model based on a Lagrangian viewpoint was developed, i.e. the pollutant is followed through a certain region in space. The model that was used in this application is a three-dimensional, stochastic, particle transport model based on the advection–diffusion equation, by the name of SIMPAR (see References [17, 18]). Originally, the model used to be a 2DH random-walk model (i.e. depth averaged), developed by the Dutch National Institute for Marine and Coastal Management, the RIKZ. The model only included horizontal movement of the particles, but it was consistent with a depth-averaged advection–diffusion equation. The movement of the particles occurred in two steps: a drift term based mainly on advection, and a diffusion term, modelled as the stochastic component. Recently, the vertically averaged version was extended to include particle movement in all three directions. To investigate the behaviour of the particle movement in stratified flow, we take a closer look at the vertical direction.

A higher-order method was implemented specifically for the movement in the vertical direction to model the above-mentioned behaviour in estuaries more accurately. Especially in three-dimensional models, where the nature of the turbulent mixing in the vertical direction is much more important than that in the horizontal direction, the simplicity of the numerical scheme in vertical direction may cause problems.

The three-dimensional, incompressible shallow flow model TRIWAQ, also made available by the RIKZ, generates the flow data that is used as input by the stochastic transport model (for more details on the TRIWAQ model you are referred to Reference [19]). The TRIWAQ model calculates time-independent quantities such as the water depth in the model, as well

as time-dependent quantities such as velocities, and water levels. The model also contains a k–ε turbulence model that yields eddy viscosities. Other elements that are included are for example bottom friction, wind, and a transport model to calculate the salinity conditions. The model grid for these calculations is curvilinear and has more grid cells in areas of specific interest, usually near the coast. The results from these calculations are then used as prior knowledge for the Lagrangian particle transport model.

3.1. The model equations

The SIMPAR model is based on the following set of equations:

$$\frac{\partial C}{\partial t} + \frac{\partial}{\partial x}(uC) + \frac{\partial}{\partial y}(vC) + \frac{\partial}{\partial z}(wC) = \frac{\partial}{\partial x}\left(D_{\mathrm{H}}\frac{\partial C}{\partial x}\right) + \frac{\partial}{\partial y}\left(D_{\mathrm{H}}\frac{\partial C}{\partial y}\right) + \frac{\partial}{\partial z}\left(D_{\mathrm{V}}\frac{\partial C}{\partial z}\right) \tag{13}$$

with initial condition

$$C(t_0, x, y, z) = \delta_0(x_0 - x)\delta_0(y_0 - y)\delta_0(z_0 - z) \tag{14}$$

where

$$\begin{aligned} C &= \text{constituent concentration } (\mathrm{kg/m^3}) \\ u &= \text{layer-averaged velocity field in } x\text{-direction } (\mathrm{m/s}) \\ v &= \text{layer-averaged velocity field in } y\text{-direction } (\mathrm{m/s}) \\ w &= \text{layer-averaged velocity field in } z\text{-direction } (\mathrm{m/s}) \\ x &= x\text{-position } (\mathrm{m}) \\ y &= y\text{-position } (\mathrm{m}) \\ z &= z\text{-position } (\mathrm{m}) \\ D_{\mathrm{H}} &= \text{horizontal diffusion coefficient } (\mathrm{m^2/s}) \\ D_{\mathrm{V}} &= \text{vertical diffusion coefficient } (\mathrm{m^2/s}) \end{aligned} \tag{15}$$

Since the probability P that a particle arrives at location (x, y, z) at time t is equal to the concentration in that point, we may substitute $P(t, x, y, z) = C(t, x, y, z)$. The corresponding Fokker–Planck equation is then found to be

$$\frac{\partial P}{\partial t} = -\frac{\partial}{\partial x}\left[\left(u + \frac{\partial D_{\mathrm{H}}}{\partial x}\right)P\right] - \frac{\partial}{\partial y}\left[\left(v + \frac{\partial D_{\mathrm{H}}}{\partial y}\right)P\right] - \frac{\partial}{\partial z}\left[\left(w + \frac{\partial D_{\mathrm{V}}}{\partial z}\right)P\right] + \frac{1}{2}\frac{\partial^2}{\partial x^2}(2D_{\mathrm{H}}P) + \frac{1}{2}\frac{\partial^2}{\partial y^2}(2D_{\mathrm{H}}P) + \frac{1}{2}\frac{\partial^2}{\partial z^2}(2D_{\mathrm{V}}P) \tag{16}$$

$$P(t_0, x, y, z) = \delta_0(x_0 - x)\delta_0(y_0 - y)\delta_0(z_0 - z) \tag{17}$$

A large number of books, and articles has been written on the connection between random-walk models and the Fokker–Planck equation, and for more detailed information we refer to for example References [3, 5, 11, 13, 15].

The set of stochastic differential equations governing the motion of particles in three dimensions looks as follows (in a simplified notation):

$$\mathrm{d}X_t \overset{\text{Itô}}{=} f_1(t,X_t,Y_t,Z_t)\,\mathrm{d}t + G_{11}(t,X_t,Y_t,Z_t)\,\mathrm{d}\beta_{1,t} \tag{18}$$

$$\mathrm{d}Y_t \overset{\text{Itô}}{=} f_2(t,X_t,Y_t,Z_t)\,\mathrm{d}t + G_{22}(t,X_t,Y_t,Z_t)\,\mathrm{d}\beta_{2,t} \tag{19}$$

$$\mathrm{d}Z_t \overset{\text{Itô}}{=} f_3(t,X_t,Y_t,Z_t)\,\mathrm{d}t + G_{33}(t,X_t,Y_t,Z_t)\,\mathrm{d}\beta_{3,t} \tag{20}$$

$$X_{t_0} = x_0, \quad Y_{t_0} = y_0, \quad Z_{t_0} = z_0 \tag{21}$$

where $\mathrm{d}t$ stands for the time step. The terms in the (3×1) drift vector $f = (f_1, f_2, f_3)$ and the (3×3) diffusion matrix G can be found from the Fokker–Planck equation (see (17))

$$f_1 = u + \frac{\partial D_\mathrm{H}}{\partial x}, \quad f_2 = v + \frac{\partial D_\mathrm{H}}{\partial y}, \quad f_3 = w + \frac{\partial D_\mathrm{V}}{\partial z} \tag{22}$$

$$G_{11} = G_{22} = \sqrt{2D_\mathrm{H}} \tag{23}$$

$$G_{33} = \sqrt{2D_\mathrm{V}} \tag{24}$$

The (3×1) vector

$$\mathrm{d}\beta_t = (\mathrm{d}\beta_{1,t}, \mathrm{d}\beta_{2,t}, \mathrm{d}\beta_{3,t})^\mathrm{T} \tag{25}$$

consists of Brownian motion increments. The increments are Gaussian distributed, stochastically independent with zero mean and variance $\mathrm{d}t$

$$E\{\mathrm{d}\beta^j(t)\} = 0 \tag{26}$$

$$E\{\mathrm{d}\beta^i(t)\,\mathrm{d}\beta^j(s)\} = \delta_{ij}\delta(t-s)\sqrt{\mathrm{d}t\,\mathrm{d}s} \tag{27}$$

Because the grid on which the necessary flow information is generated is curved, a transformation is required to convert the transport equations from a rectangular coordinate system to a curvilinear one. The curvilinear transformation is only used in the horizontal direction. The vertical direction is not affected by this. The details of this transformation will not be discussed here, and for the resulting equations you are referred to References [19, 20]. The following transformation is introduced:

$$x = x(\xi,\eta) \tag{28}$$

$$y = y(\xi,\eta) \tag{29}$$

where (x, y) are the coordinates in the Cartesian, or physical plane, and (ξ, η) the ones in the curvilinear plane. The base vectors in the (ξ, η) plane are given by

$$\underline{e}_\xi = \left(\frac{1}{\sqrt{g_{\xi\xi}}} \frac{\partial x}{\partial \xi}, \frac{1}{\sqrt{g_{\xi\xi}}} \frac{\partial y}{\partial \xi} \right) \tag{30}$$

$$\underline{e}_\eta = \left(\frac{1}{\sqrt{g_{\eta\eta}}} \frac{\partial x}{\partial \eta}, \frac{1}{\sqrt{g_{\eta\eta}}} \frac{\partial y}{\partial \eta} \right) \tag{31}$$

with the Jacobian $\sqrt{g^*}$ of the orthogonal horizontal transformation given by

$$\sqrt{g^*} = \sqrt{g_{\xi\xi}} \sqrt{g_{\eta\eta}} \tag{32}$$

$$g_{\xi\xi} = \left(\frac{\partial x}{\partial \xi} \right)^2 + \left(\frac{\partial y}{\partial \xi} \right)^2 \tag{33}$$

$$g_{\eta\eta} = \left(\frac{\partial x}{\partial \eta} \right)^2 + \left(\frac{\partial y}{\partial \eta} \right)^2 \tag{34}$$

Because the transformation is orthogonal, we also have that

$$\underline{e}_\xi \cdot \underline{e}_\eta = 0 \quad \longleftrightarrow \quad \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} + \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} = 0 \tag{35}$$

It is customary to introduce a σ-transformation for the vertical direction in a 3D model (see for example References [19, 21, 22])

$$\sigma = \frac{z - \zeta(x, y, t)}{H(x, y, t)} \quad \text{or} \tag{36}$$

$$z = \sigma H(x(\xi, \eta), y(\xi, \eta), t) + \zeta(x(\xi, \eta), y(\xi, \eta), t) \tag{37}$$

with $\sigma \in [0, 1]$, ζ stands for the water level, and z stands for the depth, and H is the total water depth. For the notation we use a tilde above the drift vector and diffusion matrix to indicate that these are now transformed. Furthermore, a subscript ξ or η is used to indicate that a variable is taken in either ξ or η direction (u_ξ and u_η). Letters that are underlined indicate vectors.

In case of an orthogonal transformation in the horizontal and a σ-transformation in the vertical direction, the transformed drift consists now of three different parts

$$\underline{\tilde{f}} = (\underline{\tilde{f}})_{\mathrm{A}} + (\underline{\tilde{f}})_{\mathrm{D}} + (\underline{\tilde{f}})_{\mathrm{C}} \tag{38}$$

where $(\underline{\tilde{f}})_{\mathrm{A}}$ stands for the drift due to advection

$$(\tilde{f}_1)_{\mathrm{A}} = \frac{1}{\sqrt{g_{\xi\xi}}} u_\xi, \quad (\tilde{f}_2)_{\mathrm{A}} = \frac{1}{\sqrt{g_{\eta\eta}}} u_\eta, \quad (\tilde{f}_3)_{\mathrm{A}} = \frac{1}{H} \omega \tag{39}$$

$(\underline{\tilde{f}})_{\mathrm{D}}$ stands for the drift due to diffusion

$$(\tilde{f}_1)_{\mathrm{D}} = \frac{1}{g_{\xi\xi}}\frac{\partial D_{\mathrm{H}}}{\partial \xi} - \frac{1}{Hg_{\xi\xi}}\frac{\partial z}{\partial \xi}\frac{\partial D_{\mathrm{H}}}{\partial \sigma} \tag{40}$$

$$(\tilde{f}_2)_{\mathrm{D}} = \frac{1}{g_{\eta\eta}}\frac{\partial D_{\mathrm{H}}}{\partial \eta} - \frac{1}{Hg_{\eta\eta}}\frac{\partial z}{\partial \eta}\frac{\partial D_{\mathrm{H}}}{\partial \sigma} \tag{41}$$

$$(\tilde{f}_3)_{\mathrm{D}} = \frac{1}{H^2}\frac{\partial D_{\mathrm{V}}}{\partial \sigma} + \frac{1}{H^2}\left[\frac{1}{g_{\xi\xi}}\left(\frac{\partial z}{\partial \xi}\right)^2 + \frac{1}{g_{\eta\eta}}\left(\frac{\partial z}{\partial \eta}\right)^2\right]\frac{\partial D_{\mathrm{H}}}{\partial \sigma} - \frac{1}{H}\left[\frac{1}{g_{\xi\xi}}\frac{\partial z}{\partial \xi}\frac{\partial D_{\mathrm{H}}}{\partial \xi} + \frac{1}{g_{\eta\eta}}\frac{\partial z}{\partial \eta}\frac{\partial D_{\mathrm{H}}}{\partial \eta}\right] \tag{42}$$

and $(\underline{\tilde{f}})_{\mathrm{C}}$ stands for the drift due to curvature of the grid

$$(\tilde{f}_1)_{\mathrm{C}} = \frac{D_{\mathrm{H}}}{g_{\xi\xi}}\left[\frac{1}{\sqrt{g_{\eta\eta}}}\frac{\partial\sqrt{g_{\eta\eta}}}{\partial \xi} - \frac{1}{\sqrt{g_{\xi\xi}}}\frac{\partial\sqrt{g_{\xi\xi}}}{\partial \xi}\right] \tag{43}$$

$$(\tilde{f}_2)_{\mathrm{C}} = \frac{D_{\mathrm{H}}}{g_{\eta\eta}}\left[\frac{1}{\sqrt{g_{\xi\xi}}}\frac{\partial\sqrt{g_{\xi\xi}}}{\partial \eta} - \frac{1}{\sqrt{g_{\eta\eta}}}\frac{\partial\sqrt{g_{\eta\eta}}}{\partial \eta}\right] \tag{44}$$

$$(\tilde{f}_3)_{\mathrm{C}} = \frac{1}{H^2}\frac{1}{\sqrt{g_{\xi\xi}}}\frac{\partial z}{\partial \xi}\left[\frac{D_{\mathrm{H}}}{\sqrt{g_{\xi\xi}}}\frac{\partial H}{\partial \xi} - H\sqrt{g_{\xi\xi}}(\tilde{f}_1)_{\mathrm{C}}\right] + \frac{1}{H^2}\frac{1}{\sqrt{g_{\eta\eta}}}\frac{\partial z}{\partial \eta}\left[\frac{D_{\mathrm{H}}}{\sqrt{g_{\eta\eta}}}\frac{\partial H}{\partial \eta} - H\sqrt{g_{\eta\eta}}(\tilde{f}_2)_{\mathrm{C}}\right] - D_{\mathrm{H}}\left[\frac{1}{g_{\xi\xi}}\frac{\partial}{\partial \xi}\left(\frac{1}{H}\frac{\partial z}{\partial \xi}\right) + \frac{1}{g_{\eta\eta}}\frac{\partial}{\partial \eta}\left(\frac{1}{H}\frac{\partial z}{\partial \xi}\right)\right] \tag{45}$$

The expressions for the drift due to diffusion are greatly simplified if the diffusion is assumed to be constant in the horizontal direction

$$(\tilde{f}_1)_{\mathrm{D}} = 0 \tag{46}$$

$$(\tilde{f}_2)_{\mathrm{D}} = 0 \tag{47}$$

$$(\tilde{f}_3)_{\mathrm{D}} = \frac{1}{H^2}\frac{\partial D_{\mathrm{V}}}{\partial \sigma} \tag{48}$$

The other parts of the drift vector $\underline{f}$ do not change. Of course when the diffusion in the vertical direction is assumed to be constant as well, the entire drift due to diffusion $(\underline{\tilde{f}})_{\mathrm{D}}$ is

no longer present. For the transformation of the diffusion part of the equation we find

$$(\tilde{G}_{11}) = \sqrt{\frac{2D_H}{g_{\xi\xi}}}, \quad (\tilde{G}_{22}) = \sqrt{\frac{2D_H}{g_{\eta\eta}}}, \quad (\tilde{G}_{33}) = \frac{1}{H}\sqrt{2D_V} \tag{49}$$

Particle movement in the third dimension does not necessarily have to be calculated using the σ-transformation. To simplify things, it was decided not to use the σ-transformation in SIMPAR 3D. Calculations are done in absolute positions. This simplifies the curvature and diffusion terms in the vertical direction even more

$$(\tilde{f}_3)_A = w, \quad (\tilde{f}_3)_C = 0, \quad (\tilde{f}_3)_D = \frac{\partial D_V}{\partial z}, \quad (\tilde{G}_{33}) = \sqrt{2D_V} \tag{50}$$

because all derivatives of z to ξ and η are now zero ($z = z(x, y)$ and not $z = z(\xi, \eta)$). Note that this also means that the velocities in the vertical direction are no longer the vertical velocities related to the sigma-coordinates, but the real, physical velocities in that direction.

The development of a true three-dimensional higher-order numerical scheme would of course be ideal. One of the drawbacks of higher-order numerical methods in large models is usually that a lot more work needs to be done by the computer. Especially in three-dimensional models the effect of higher-order methods on the overall amount of calculation time is problematic. Because the horizontal directions are of less interest, it was decided that the motions of the particles in those directions are sufficiently modelled by a simple Euler scheme. Since the movement of the particles is independent in every direction, it is possible to split the movement of the particle in a horizontal and a vertical component. This in turn provides the possibility to use two different numerical schemes: one for each direction. For the particle movement in the vertical direction, the Euler scheme was therefore replaced by the more accurate Milstein or the stochastic Runge–Kutta numerical scheme. This resulted in a more efficient computation than a fully three-dimensional model, that still has the benefits of a higher-order scheme in the direction that we are most interested in.

Because of our interest in a higher-order scheme in vertical direction, it is necessary to transform the original Itô SDE into a Stratonovich one. An additional term in the drift, $(\tilde{f}_3)_S$, is required (only for the SDE in vertical direction)

$$\tilde{f}_3 = (\tilde{f}_3)_A + (\tilde{f}_3)_D + (\tilde{f}_3)_C + (\tilde{f}_3)_S \tag{51}$$

where

$$(\tilde{f}_3)_S = +\frac{1}{2} G_{33} \frac{\partial}{\partial z}(G_{33}) \tag{52}$$

$$= +\frac{1}{2}\sqrt{2D_V}\frac{\partial}{\partial z}(\sqrt{2D_V})$$

$$= +\sqrt{D_V}\frac{\partial}{\partial z}(\sqrt{D_V})$$

$$= +\frac{1}{2}\frac{\partial D_V}{\partial z} \tag{53}$$

Boundary conditions are available from the TRIWAQ model. The particles are not allowed to pass through the solid boundary of the coast and the bottom. In the case that this does happen, the method of halving the time step, as explained in the beginning of this paper (Section 1), is applied. It is slightly different when encountering some form of stratification, for example when salt water slides under fresh water from a river. The particles are not supposed to pass freely from one layer to another, because of the different densities in the different layers. Physically speaking, there will of course be exchange of particles from the fresh-water layer and the salt-water layer. However, since the bulk of the particles does not mix, similar behaviour should be expected from the particle model. If a particle moves outside the area of interest it is removed from the calculation.

3.2. An experiment in a simulated estuary

Because of the complexity of the model and the many factors that have impact on the calculation, it was decided to simulate the above-mentioned situation in estuaries in a more controlled environment (more details can also be found in Reference [23]). A simple geometry was used to test the effects of different numerical schemes. The objective was to create an example of a stratified situation where it is possible to identify the areas of different density, simply by looking at the profile of the vertical diffusion distribution. Obviously, the diffusion should be equal to zero, both near the surface and the bottom. Furthermore, the diffusion around the interface between fresh and salt water should be close to zero as well. Consider a rectangular channel, 60 km in length, 5 km in width, and 10 m in depth. Two physical layers are simulated to create a stratified system. The left side of the model contains a velocity boundary condition, while the right side of the model has a water level boundary. The top layer flows from left to right, with a velocity $v_1 = 0.6\,\mathrm{m/s}$ and a density $\rho_1 = 1020\,\mathrm{kg/m^3}$. The bottom layer flows from left to right as well, but with a velocity $v_2 = 0.3\,\mathrm{m/s}$, and a density $\rho_2 = 1036\,\mathrm{kg/m^3}$. This means that the fresh water on top slides across the salt water at the bottom. Depending on the amount of turbulence that is internally generated by the model, these two layers will either stay stratified, or become completely mixed.

In the vertical direction, 20 layers were used in the flow calculations of the shallow-water finite-difference model. This number was chosen to obtain enough detail in the vertical direction, especially near the interface between the fresh and salt water. Due to bottom friction enough turbulent energy is generated to obtain a parabolic diffusion profile in vertical direction in the bottom layer. A comparable parabolic diffusion profile is generated in the top layer by adjusting the effects of the wind (Figure 6).

Since the salt water is heavier than the fresh water, it requires a certain amount of energy before the two layers will mix. Enough turbulent energy needs to be generated before the salt water is 'lifted' across the interface to cause mixing. Too small an amount of friction or wind will cause the diffusion to stay uniformly distributed across a layer (flow stays stratified) instead of generating the desired parabolic profile (see Figure 5). Next, particles are released near the beginning of the channel, at a depth of 7.5 m. Since we are specifically interested in the behaviour of the particles in vertical direction, the results for the horizontal propagation of the particles are not included here. A vertical cross section is shown in Figure 7 for two of the tested numerical schemes. The figures show the results for 100 particles in the case of the Euler scheme and the stochastic version of the Runge–Kutta scheme.

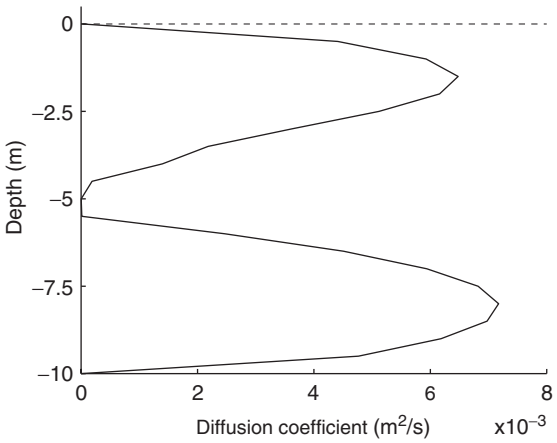

Figure 6. The vertical diffusion profile in a rectangular channel.

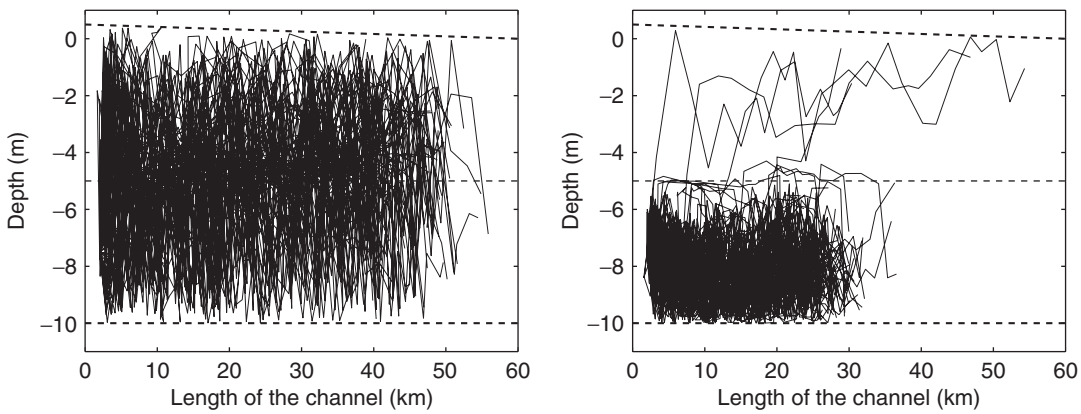

Figure 7. The release of 100 particles in stratified flow in a channel. A vertical cross section of the channel is shown. On the left side a numerical simulation with the Euler scheme, and on the right side the same simulation with the higher order stochastic Runge–Kutta scheme. The simulation time was equal to 36 h, with a time step of one hour.

Besides the fact that the choice of the numerical scheme in the vertical direction affects the particle behaviour in that direction, it also affects the particle behaviour in the *horizontal* direction. Because the top layer moves at twice the speed of the bottom layer, particles that arrive in this layer are transported much faster than they should. As is obvious from Figure 7, this introduces a large amount of artificial horizontal dispersion. In fact, in this specific scenario, the horizontal size of the cloud is nearly two times larger when the Euler scheme is used than that from the Runge–Kutta scheme.

The time step of one hour in these simulations was chosen based on the horizontal scales. The transport is based on the advection–diffusion equation, indicating that the large-scale motions in the flow are of interest. The diffusion coefficient in the coastal zone is $O(10^2)\,\mathrm{m}^2/\mathrm{s}$, the time step is equal to 1 h. A typical diffusion step in horizontal direction would be a random

number drawn from a standard normal distribution multiplied by a factor $\sqrt{2D\,dt}$, which would be around 850 m. The diffusion coefficient in the vertical direction is $O(10^{-6})\,\mathrm{m}^2/\mathrm{s}$, but due to the effects of shear this results in a turbulent diffusion coefficient in the order of $10^{-3}\,\mathrm{m}^2/\mathrm{s}$. This means that the random increments in the vertical direction with this time step are around 2.6 m. Since the depth of the channel is only 10 m, and the height of the two physical layers is roughly half of that, it is very likely that a particle will cross the interface between the two layers. This type of model contains two different time scales: one for the horizontal plane, where time steps of one hour give no problems, and another for the vertical direction where a time step of one hour does give problems.

3.3. A real-life 3D particle simulation in the Dutch coastal zone

After the promising results of the test case scenario, some more real-life simulations were performed, this time in a not so controlled environment: the Dutch coastal zone. The hydrographical situation of this area was calculated on a grid of 127×368 points. A realistic bottom profile was used.

A group of particles was released in the southernly part of the North Sea, approximately 30 km from the coast. These particles were then tracked for the next 4 days. Because the residual flow (to the North) is very small, a lengthy simulation would be required. In order to avoid this a heavy wind (a constant 10 m/s) coming from the Southwest was introduced in the simulations, which basically prevented the particles from moving South again due to the tide. That way, the particles move along most of the Dutch coastal area in just a few days of simulation. In Figure 8 some details of two specific particles can be found. Both the horizontal and the vertical paths are shown. It should be noted that the vertical scale looks rather rough, but this is exaggerated due to scaling. The particles actually travel well over 200 km, while the bottom topography varies between about 10 and 20 m in depth. The particles move across the entire depth because there is no stratification. The pictures of the bottom topography show some interesting physical properties: clearly the gully extending from the Nieuwe Waterweg into sea can be distinguished ($t = 9$). Another interesting feature is the sand bank in front the isle of Texel, where one of the two particles moves to after approximately 77 h.

For completeness a sample of the mean particle displacement for the 4-day simulation is shown in Figure 9. Also included is an indication of the size of the cloud. From these figures it is clear that due to the differences in the treatment of numerical procedures, the variations in horizontal transport are minimal.

A simulation with the newly implemented Runge–Kutta scheme takes approximately three times longer than a simulation with the Euler scheme. In the past the results of the model have been compared to the results from the Eulerian transport model TRIWAQ. The TRIWAQ model has been used to generate the velocity fields. Since both models are based on the same (advection–diffusion) equation, the results are comparable. It falls outside the scope of this paper to give a full comparison between the Lagrangian and the Eulerian approach.

4. CONCLUSIONS

In the first part of the article various stochastic numerical schemes have been compared to find the best suitable scheme in the main problem of interest: a fully three-dimensional particle

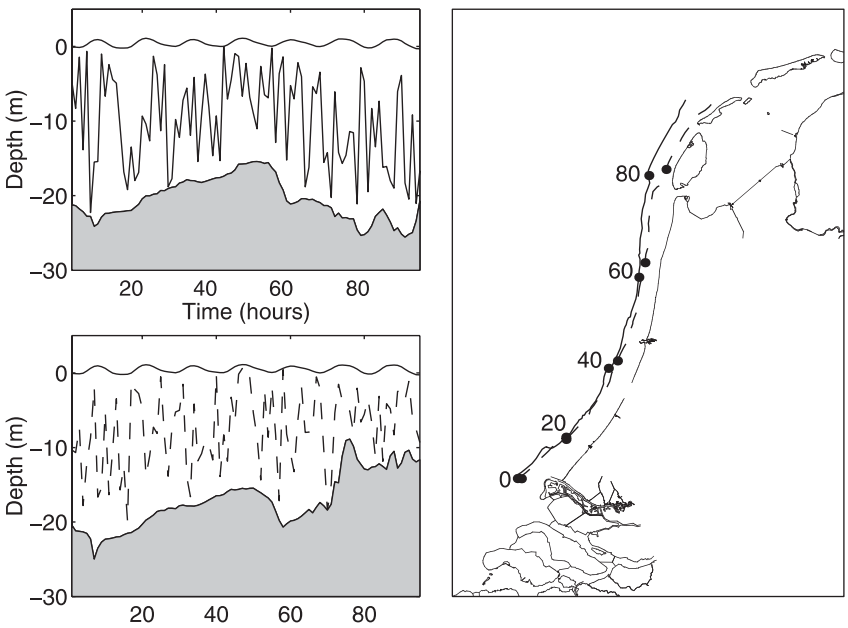

Figure 8. A sample of the path of two particles in the Dutch coastal zone. On the left two particles are followed in time and a cross section in the vertical direction is shown. On the right-hand side the horizontal movement of the same two particles is shown. The numbers in the right-hand side picture are the hours that have passed since the initial release. These times correspond to the times in the figures on the left-hand side.

transport model for use in stratified flow. Specific examples with a spatially varying diffusion coefficient have been investigated. Each experiment was performed by a number of different, easy-to-use numerical schemes: Euler, Milstein, Richardson, Heun, and Runge–Kutta, and various implicit ones. Some of these schemes are known to produce the Itô solution of a SDE (Euler, Milstein, and Richardson), while others produce the Stratonovich solution (Heun and Runge–Kutta). These last two schemes therefore involve a transformation so that they yield the desired Itô solution.

Even though some of these schemes only require a minor additional effort compared with the Euler scheme, the changes in terms of much more accurate results and faster convergence are rewarding. Multiple-step schemes such as the Heun, or Runge–Kutta scheme give good results. The set of experiments that was carried out is an indication that any improvement over the Euler scheme is welcome. Even though this is not trivial, and some care needs to be taken in the choice and application of the scheme, such a scheme does not necessarily need to be complicated or expensive. As advanced as the current Lagrangian particle transport models for dispersion in turbulent flow sometimes are, it is surprising to see how many of these models still use the Euler scheme, while minor adaptations may greatly improve the model's accuracy.

In the second part of the article more realistic scenarios were studied. At first a simplified case was evaluated: a rectangular channel in which stratification occurs. The simple geometry

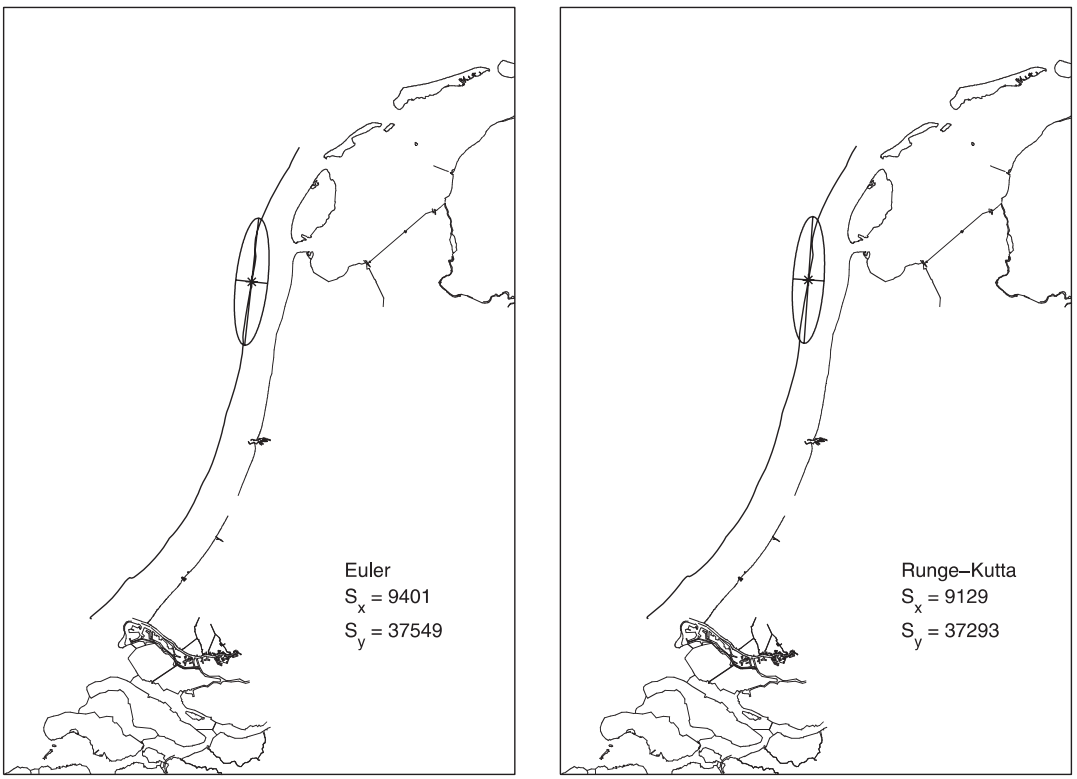

Figure 9. The mean displacement of a cloud of particles. On the left are the results with the Euler scheme, on the right the same results with the Runge–Kutta scheme. An example of the size of the cloud is shown after 3 days of simulation. The size of the short axis of the ellipse is given by S_x and the long axis by S_y (both in meters). The size of the ellipse is chosen such that 95% of all particles are within its circumference. Variations in patch size are minimal: less than 300 m in both directions after 3 days. Variation in the mean displacement is in the same order of magnitude: about 400 m after 3 days.

was chosen in order to be able to closely observe the vertical movement of the particles in a somewhat controllable environment. Several numerical schemes were implemented (Euler, Milstein, and Runge–Kutta) to see how each of these schemes would cope with the stratified conditions in the channel. Results from this scenario revealed that the choice of the numerical scheme has quite an impact on the overall result. Not only are the results in the vertical direction different, but the choice for the numerical scheme also affects the distribution of the particles in the *horizontal* direction. In the particular experiment described above (see Section 3.2), the size of the particle cloud in horizontal direction ended up being almost twice as large under certain conditions.

After the simple geometry, the method was applied to a more complicated geometry of the Dutch coastal zone. Unless a specific zone with stratified flow is included in the simulation, the results for different numerical schemes were similar (for the mean). Note that because of the large differences in scales between the horizontal and vertical directions, the choice of the

size of the time step has important implications. A time step that is suitable for horizontal movement might not be the most appropriate one for vertical movement and *vice versa*.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support of the Dutch National Institute for Marine and Coastal Management (RIKZ).

REFERENCES

1. Durbin PA. A stochastic model of two-particle dispersion and concentration fluctuations in homogeneous turbulence. *Journal of Fluid Mechanics* 1980; **100**(2):279–302.
2. Pope SB. PDF methods for turbulent reactive flows. *Progress in Energy and Combustion Science* 1985; **11**:119–192.
3. Thomson DJ. Criteria for the selection of stochastic models of particle trajectories in turbulent flows. *Journal of Fluid Mechanics* 1987; **180**:529–556.
4. Sawford BL. Recent developments in the Lagrangian stochastic theory of turbulent dispersion. *Boundary-Layer Meteorology* 1993; **62**:197–215.
5. Rodean HC. *Stochastic Lagrangian Models of Turbulent Diffusion*. Meteorological Monographs, vol. 26, No. 48. American Meteorological Society: 45 Beacon Street, Boston, MA 02108, 1996.
6. Taylor GI. Diffusion by continuous movements. *Proceedings of the London Mathematical Society*, *Series A* 1921; **20**:196–211.
7. Batchelor GK. Diffusion in a field of homogeneous turbulence II. The relative motion of particles. *Proceedings of the Cambridge Philosophical Society* 1952; **48**:345–362.
8. De Baas AF. Some properties of the Langevin model for dispersion. *Ph.D. Dissertation*, Delft University of Technology, 1988.
9. Borgas MS, Yeung PK. Conditional fluid-particle accelerations in turbulence. *Theoretical and Computational Fluid Dynamics* 1998; **11**:69–93.
10. Kaplan H, Dinar N. A three-dimensional stochastic model for concentration fluctuation statistics in isotropic homogeneous turbulence. *Journal of Computational Physics* 1988; **79**:317–335.
11. Thomson DJ. A stochastic model for the motion of particle pairs in isotropic high-Reynolds-number turbulence, and its application to the problem of concentration variance. *Journal of Fluid Mechanics* 1990; **210**:113–153.
12. Crone GC. Parallel Lagrangian models for turbulent transport and chemistry. *Ph.D. Dissertation*, University of Utrecht, 1997.
13. Jazwinsky AH. *Stochastic Processes and Filtering Theory*. Academic Press: New York, 1970.
14. Heemink AW. Stochastic modeling of dispersion in shallow water. *Stochastic Hydrology and Hydraulics* 1971; **4**:161–174.
15. Kloeden PE, Platen E. *Numerical Solution of Stochastic Differential Equations*. Springer: New York, 1995.
16. Rümelin W. Numerical treatment of stochastic differential equations. *SIAM Journal on Numerical Analysis* 1982; **19**(3):604–613.
17. Elorche M. Vooronderzoek particle-module in SIMONA. *Werkdocument RIKZ/OS-94.143x*, 1994 (in Dutch).
18. Lin HX, Heemink AW, Stijnen JW, Cosman A, Van Beek P. Parallellization of the particle model SIMPAR. In *Advances in Hydro-Science and Engineering*, Holz KP, Bechteler W, Wang SSY, Kawahara M (eds), vol. III. Center for Computational Hydroscience and Engineering, Carrier Hall, University of Mississippi, MS 38667, USA, 1998.
19. Zijlema M. TRIWAQ—three-dimensional incompressible shallow flow model. *Technical Documentation*, National Institute for Coastal and Marine Management, 1997.
20. Ten Cate E, WAQUA. *Technical Documentation. SIMONA-Report 98-01*, 1999.
21. Stelling GS. On the construction of computational methods for shallow water flow problems. *Ph.D. Dissertation*, Delft University of Technology, 1984.
22. Dunsbergen DW. Particle models for transport in three-dimensional shallow water flow. *Ph.D. Dissertation*, Delft University of Technology, 1994.
23. Stijnen JW, Heemink AW, Lin HX. Higher order numerical methods for pollutant transport. In *3rd International Symposium on Environmental Hydraulics*, Boyer D, Rankin R (eds). Tempe, Arizona State University, Arizona, U.S.A., 2001; 1–6.

