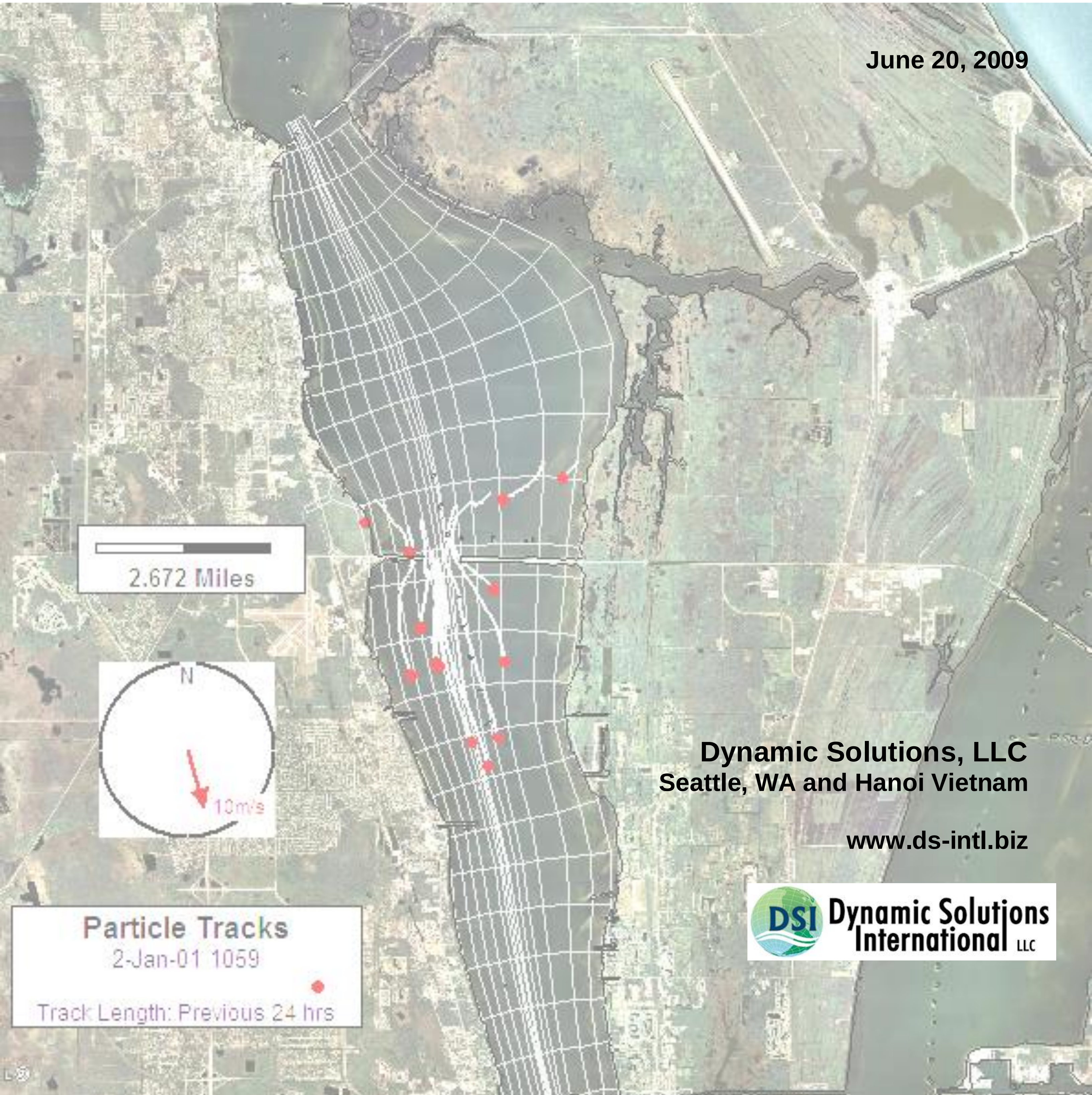
Implementation of a
Lagrangian Particle Tracking Sub-Model
for the
Environmental Fluid Dynamics Code
(Draft)



June 20, 2009

Dynamic Solutions, LLC
Seattle, WA and Hanoi Vietnam

www.ds-intl.biz

DSI Dynamic Solutions International LLC

Table of Contents

List of Tables

List of Figures

1 Introduction

The Environmental Fluid Dynamics Code (EFDC) is a general-purpose modeling package for simulating three dimensional (3-D) flow, transport, and biogeochemical processes in surface water systems including: rivers, lakes, estuaries, reservoirs, wetlands, and near-shore to shelf-scale coastal regions (Hamrick, 1992a, 1992b and 1996). The EFDC model was originally developed at the Virginia Institute of Marine Science for estuarine and coastal applications and is public domain software. The US EPA has continued to support its development and now EFDC is part of a family of models recommended by EPA for TMDL development. In addition to hydrodynamic and salinity and temperature transport simulation capabilities, EFDC is capable of simulating cohesive and non-cohesive sediment transport, near-field and far-field discharge dilution from multiple sources, the transport and fate of toxic contaminants in the water and sediment phases, and the dissolved oxygen/nutrient process (i.e. eutrophication). Special enhancements to the hydrodynamics of the code, including vegetation resistance, drying and wetting, hydraulic structure representation, wave current boundary layer interaction, and wave-induced currents, allow refined modeling of wetland and marsh systems, controlled-flow systems, and near-shore wave-induced currents and sediment transport.

Dynamic Solutions - International LLC (DSI), has developed a version (EFDC_DSI) of the code that streamlines the modeling process and links to DSI's pre- and post-processing tool EFDC_Explorer (Craig, 2008). The EFDC_DSI code is open source and is periodically synchronized with the EPA GVC version of EFDC to provide the most up-to-date version.

The enhancements to the EFDC_DSI/EFDC_Explorer modeling system were:

EFDC_DSI Modifications

- Added Lagrangian Particle Transport (LPT) sub-model.
- Added the option to the LPT to fix the depth of a particle to a user specified depth.
- Added the random walk option to the LPT.
- Removed all older/previous versions of particle tracking from the code.

EFDC_Explorer Modifications

- Pre-processing: Added particle seeding and LPT control options.
- Post-processing:
    - Added 2D plan view plotting and animation of LPT's.
    - Added the ability to export one or more particle tracks to ASCII files for linkage to 3rd party applications.

2 Mathematical Model

The advection-diffusion equation for mass transport in a three dimensional curvilinear orthogonal coordinate system is:

$$\frac{\partial c}{\partial t} + div(\vec{V}c) = \frac{\partial}{\partial x}\left(D_H \frac{\partial c}{\partial x}\right) + \frac{\partial}{\partial y}\left(D_H \frac{\partial c}{\partial y}\right) + \frac{\partial}{\partial z}\left(D_V \frac{\partial c}{\partial z}\right) \quad (1)$$

where *t* is time, (*x,y,z*) Lagrangian coordinates of a particle, c is concentration, $\vec{V} = (u, v, w)$ velocity of fluid flow, and D_H, D_V are the horizontal and vertical diffusion coefficients, respectively.

The differential equations for the Lagrangian movement of particles is consistent with Eq.(1) and are as follows:

$$dx = dx_{drift} + dx_{ran} = \left(u + \frac{\partial D_H}{\partial x}\right)dt + \sqrt{2D_H dt}(2p-1) \quad (2)$$

$$dy = dy_{drift} + dy_{ran} = \left(v + \frac{\partial D_H}{\partial y}\right)dt + \sqrt{2D_H dt}(2p-1) \quad (3)$$

$$dz = dz_{drift} + dz_{ran} = \left(w + \frac{\partial D_V}{\partial z}\right)dt + \sqrt{2D_V dt}(2p-1) \quad (4)$$

In which *dt* is the time step and *p* is a random number from a uniformly distributed random variable generator having mean of 0.5. When transformed using the $2p - 1$ the random component has a mean of zero and a range from -1 to 1. The transformed random value allows the diffusion term to move particles +/- about the advected position. Eq's 2-4 follow the 3D random walk approach used by Dunsbergen et al. (1993).

In order to determine the Lagrangian trajectory of the particle, the equations (2)-(4) were incorporated into EFDC model. The numerical solution was separately divided into the advective transport and random components as described above. This approach allows the user to enable (i.e. turn on random walk) or disable (advective transport only) the random components for either the horizontal and/or the vertical directions.

3 Module development in EFDC

The Fortran90 module, DRIFTER.F90, has been developed and merged into EFDC to solve the Equations 2-4. The module contains the following subroutines and functions:

- DRIFTERC: Solves the Eqs.(2)-(4) using one of three numerical options
- DRIFTER.INP: Reading the input parameters
- READSTR: Reading the comment lines
- CONTAINER: Determining the cell containing the drifter
- AREACAL: Calculating the area of polygons
- DRIFVELCAL : Interpolating the velocity components at the previous location of the drifter
- RANDCAL: Calculating the random movement
- EDGEMOVE: Dealing with the drifter hitting the land bounadry or internal walls
- INSIDECELL: Determining if the drifter is inside a cell
- DRIFTERWDEP: Interpolating the bathymetry and total water depth at the location of drifter
- DRIFTERLAYER: Determining the layer containing the drifter

Three options are available for the solution of the differential equations 2-4. They are explicit Euler, predictor-corrector Euler and forth order Runge-Kutta. Their discretization for the equations are as follows:

- Explicit Euler method: This method is very simple with the approximation of O(Δt):

$$x_{n+1} = x_n + \Delta t u(t_n, x_n, y_n, z_n) \tag{5}$$
$$y_{n+1} = y_n + \Delta t v(t_n, x_n, y_n, z_n) \tag{6}$$
$$z_{n+1} = z_n + \Delta t w(t_n, x_n, y_n, z_n) \tag{7}$$

- Predictor-corrector Euler method: This method has the advantage of explicit and implicit features with the approximation of O(Δt2).

$$x_{n+1} = x_n + 0.5\Delta t\left(u(t_n, x_n, y_n, z_n) + u\left(t_{n+1}, x_{n+1}^p, y_{n+1}^p, z_{n+1}^p\right)\right) \tag{8}$$
$$y_{n+1} = y_n + 0.5\Delta t\left(v(t_n, x_n, y_n, z_n) + v\left(t_{n+1}, x_{n+1}^p, y_{n+1}^p, z_{n+1}^p\right)\right) \tag{9}$$
$$z_{n+1} = z_n + 0.5\Delta t\left(w(t_n, x_n, y_n, z_n) + w\left(t_{n+1}, x_{n+1}^p, y_{n+1}^p, z_{n+1}^p\right)\right) \tag{10}$$

where $\left(x_{n+1}^{p}, y_{n+1}^{p}, z_{n+1}^{p}\right)$ are calculated by Eqs. (4)-(6)

- Runge-Kutta 4 method: This method has the approximation of O(Δt4) and has been shown in the testing for this project that it is best option of the three solution techniques provided.

$$x_{n+1} = x_n + \frac{1}{6}\left(\Delta x_1 + 2\Delta x_2 + 2\Delta x_3 + \Delta x_4\right) \tag{11}$$

$$y_{n+1} = y_n + \frac{1}{6}\left(\Delta y_1 + 2\Delta y_2 + 2\Delta y_3 + \Delta y_4\right) \tag{12}$$

$$z_{n+1} = z_n + \frac{1}{6}\left(\Delta z_1 + 2\Delta z_2 + 2\Delta z_3 + \Delta z_4\right) \tag{13}$$

in which

$$\begin{aligned}
\Delta x_1 &= \Delta t u\left(t_n, x_n, y_n, z_n\right) \\
\Delta x_2 &= \Delta t u\left(t_n + 0.5\Delta t, x_n + 0.5\Delta x_1, y_n + 0.5\Delta y_1, z_n + 0.5\Delta z_1\right) \\
\Delta x_3 &= \Delta t u\left(t_n + 0.5\Delta t, x_n + 0.5\Delta x_2, y_n + 0.5\Delta y_2, z_n + 0.5\Delta z_2\right) \\
\Delta x_4 &= \Delta t u\left(t_n + \Delta t, x_n + \Delta x_3, y_n + \Delta y_3, z_n + \Delta z_3\right) \\
\Delta y_1 &= \Delta t v\left(t_n, x_n, y_n, z_n\right) \\
\Delta y_2 &= \Delta t v\left(t_n + 0.5\Delta t, x_n + 0.5\Delta x_1, y_n + 0.5\Delta y_1, z_n + 0.5\Delta z_1\right) \\
\Delta y_3 &= \Delta t v\left(t_n + 0.5\Delta t, x_n + 0.5\Delta x_2, y_n + 0.5\Delta y_2, z_n + 0.5\Delta z_2\right) \\
\Delta y_4 &= \Delta t v\left(t_n + \Delta t, x_n + \Delta x_3, y_n + \Delta y_3, z_n + \Delta z_3\right) \\
\Delta z_1 &= \Delta t w\left(t_n, x_n, y_n, z_n\right) \\
\Delta z_2 &= \Delta t w\left(t_n + 0.5\Delta t, x_n + 0.5\Delta x_1, y_n + 0.5\Delta y_1, z_n + 0.5\Delta z_1\right) \\
\Delta z_3 &= \Delta t w\left(t_n + 0.5\Delta t, x_n + 0.5\Delta x_2, y_n + 0.5\Delta y_2, z_n + 0.5\Delta z_2\right) \\
\Delta z_4 &= \Delta t w\left(t_n + \Delta t, x_n + \Delta x_3, y_n + \Delta y_3, z_n + \Delta z_3\right)
\end{aligned}$$

4 Input and Output Files

EFDC_Explorer provides the graphic user interface to the required input and output files. The new and updated input files for the LPT module are:

EFDC.INP

The option that activates the particle tracking computations is located in the EFDC.INP card group 67. The variable ISPD contains the flag

```
C67 ISPD NPD NPDRT NWPD ISLRPD ILRPD1 ILRPD2 JLRPD1 JLRPD2 MLRPDRT IPLRPD
       N   0     0    0      0      0      0      0      0       0      0
```

in which

ISPD = 0: Particle tracking disabled
2: Explicit Euler
3: Predictor-corrector Euler
4: Runge-Kutta 4

All other fields on card group 67 are ignored. They have been left in for backwards compatibility with other versions of EFDC. Card group 68 is ignored but retained for backwards compatibility also.

CORNERS.INP

This file is created by EFDC_EXPLORER (EE) which contains the coordinates of four vertices of the polygons belonging to the computational grid.

DRIFTER.INP

This is the main input file of the drifter module and contains the data on the computational time, frequency for output, number of drifters and their initial locations as well as the other necessary parameters. It is created by EE and a sample file is shown as below:

```
*** SJWMD LPT Test
* DRIFTER.INP File for EFDC
* Comment Lines Starts With '*'
* Using C67 ISPD in EFDC.INP
* ISPD:     0 No tracking calculation
*           2 Explicit Euler
*           3 Pre-Corrector Euler
*           4 Runge-Kutta 4
* ZOPT:     Lagrangian Particle Tracking Option in the Z direction:
*           0 Depths are Fixed at the Initial Seeding Depth
*           1 Fully 3D Lagrangian Neutrally Buoyant Particles
* PRAN:     Random Walk Option to add a random movement
*           0 No random component
*           1 Random Walk, Horizontal ONLY
```

```
*           2 Random Walk, Vertical   ONLY
*           3 Random Walk, 3D Random  FULL
* DIFOP     Option For Random Walk Diffusivity
*           0 Use AH(L,K) and AV(L,K) from EFDC computations (Horizontal diffusion
*                                     should be turned on in EFDC)
*           1 Use HORDIF  VERDIF from this file
* HORDIF    Horizontal Diffusivity (m^2/s)
* VERDIF    Vertical Diffusivity   (m^2/s)
* DEPOP:    Option for specifying initial vertical position in input file
*           0 Elevations are specified
*           1 Depths are specified
*      ZOPT PRAN DIFOP   HORDIF    VERDIF      DEPOP
        0      0          0           0.01           0.001         1
**********************************************************************
*   Julian Time to BEGIN and END Particle Tracking Computations
*                                   OUTPUT
*   BEGIN[day]       END[day]      FREQ[min]
             0             1             1
**********************************************************************
* Number Of Drifters: NPD
         5
**********************************************************************
* Initial Coordinates Of Drifters
*       XLA         YLA         ZLA/DLA
        35.000       135.000         1.000
        45.000       135.000         1.000
        55.000       135.000         1.000
        65.000       135.000         1.000
        75.000       135.000         1.000
```

DRIFTER.OUT

DRIFTER.OUT contains the LPT model output in binary form. This file contains the coordinates (*x*,*y*,*z*) of the each particle with time. The coordinate system used is the same as used by the model. The units are in meters. This file is used by EE to present the particle tracks results.

5 Model Tests Cases

In order to validate the particle tracking model, a number of tests were carried out to test various model options and conditions. EE was used to set up the model runs and produced the graphics included in this section.

5.1 Test 1: 1D Open Channel

The computational domain is a straight channel with the length of 20000 m and the width of 10 m. The grid is a rectangular mesh of 200x1 with the grid size $\Delta x = 100$ m. The bed slope of the channel is 0.0001 resulting in model bathymetry as shown in Figure 1 (shown with a N-S exaggeration of 20). The initial and boundary conditions for hydrodynamics of the channel were:

$$h(t,x) = 2 \text{ m}, \quad \forall x > 0, t = 0 \tag{14}$$

$$Q(t,x) = 10 \ m^3/s, \quad \forall t > 0, x = 0 \tag{15}$$

$$\eta(t,x) = 5.2 \ m, \quad \forall t > 0, x = 20000 \text{ m} \tag{16}$$

The initial coordinates of 5 drifters were seeded with the same location of 500.0, 5.091, 6.950. The LPT options and parameters are shown in Figure 2.

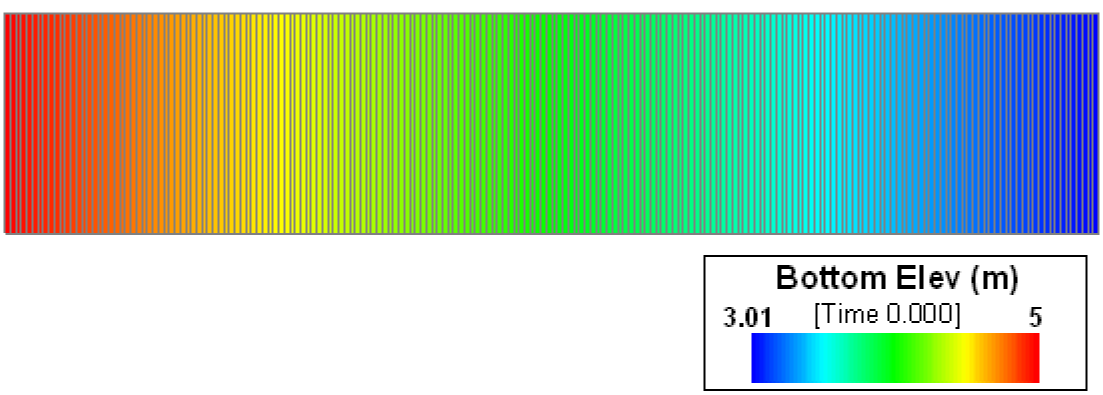


Figure 1 Test 1: Plan view of grid and bathymetry.

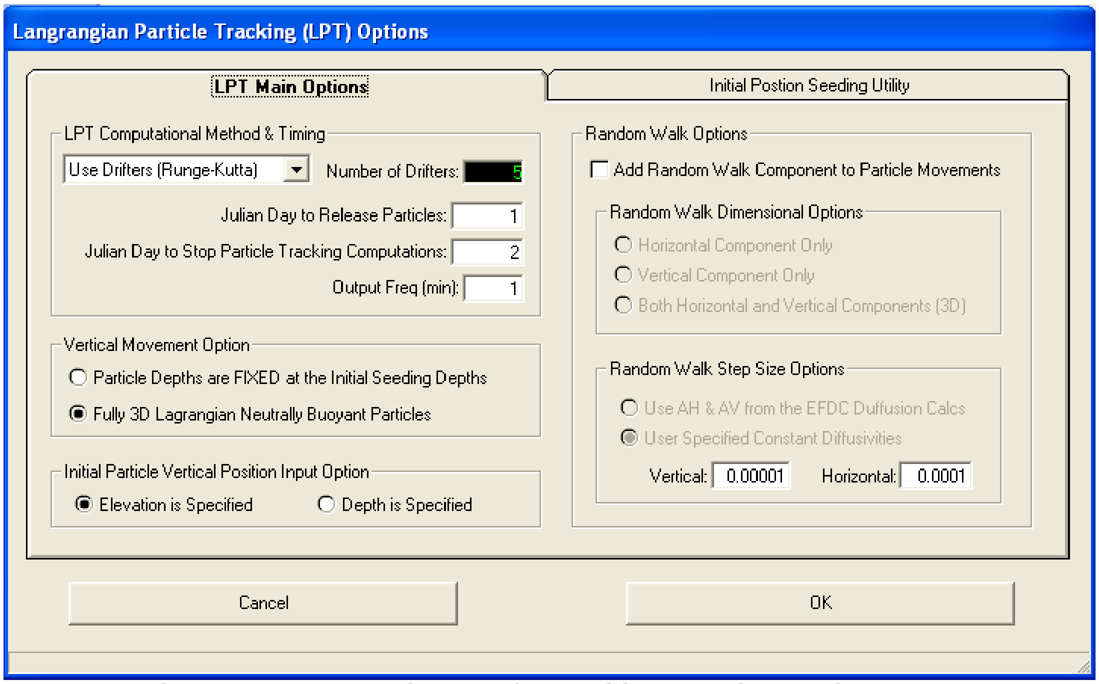


Figure 2 Test 1: Setting up of LPT without random walk.

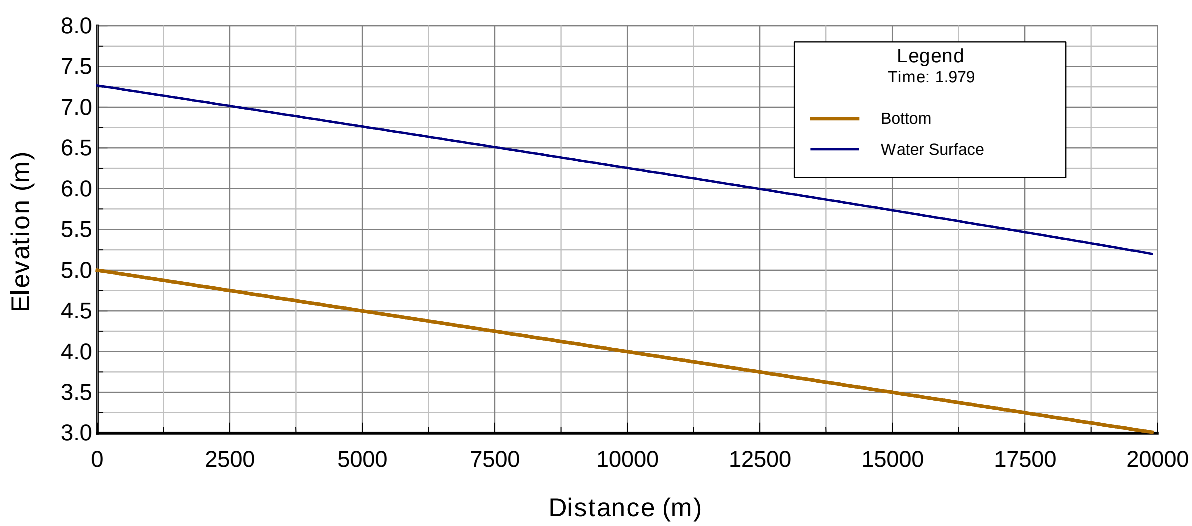


Figure 3 Water surface and bottom elevations at t=1.979 days.

Figure 3 provides a longitudinal profile of the model, showing the water surface (blue line) and bottom (brown line) elevations. The objective of the model setup was to provide, essentially, a uniform (i.e. unchanging along the river) and steady state flow field. This allows a simple computational approach to check the movement of the particles.

Table 1 Test 1 particle velocity results.

Time (days)	X (m)	Y (m)	Particle Vel (m/s)
1.0000	500.0	5.091	-
1.0070	766.4	5.091	0.441
1.0556	2619.7	5.091	0.441
1.0625	2884.5	5.091	0.444
1.0764	3414.2	5.091	0.441
1.0834	3679.1	5.091	0.438
1.0903	3944.0	5.091	0.444
1.1250	5269.1	5.091	0.442
1.1597	6595.1	5.091	0.442
1.1945	7922.3	5.091	0.441
1.2292	9250.8	5.091	0.443
1.2639	10581.1	5.091	0.444
1.2986	11912.6	5.091	0.444
1.3333	13247.9	5.091	0.445
1.3681	14586.5	5.091	0.445
1.4028	15929.5	5.091	0.448
1.4722	18633.7	5.091	0.451
1.4792	18905.8	5.091	0.450

Table 1 provides a summary of the drifter positions with time and the resulting particle velocities. It can be seen that the particle velocities were nearly steady during this period. The travel distance of the drifter at t=1.479 (over 11.5 hours) can be evaluated by the simple analytical calculation with the average velocity (computed from the EFDC velocity field) of 0.445 *m/s* as follows:

$$L = 0.445(1.479 - 1)86400 + 500 = 18924.28\,m,$$

The computed distance by the EFDC_DSI particle tracking was 18905.8 *m*, producing a relative error of -0.098%. Figure 4 shows the trajectories of the 5 drifters. All 5 particles move only in a straight line and all move together.

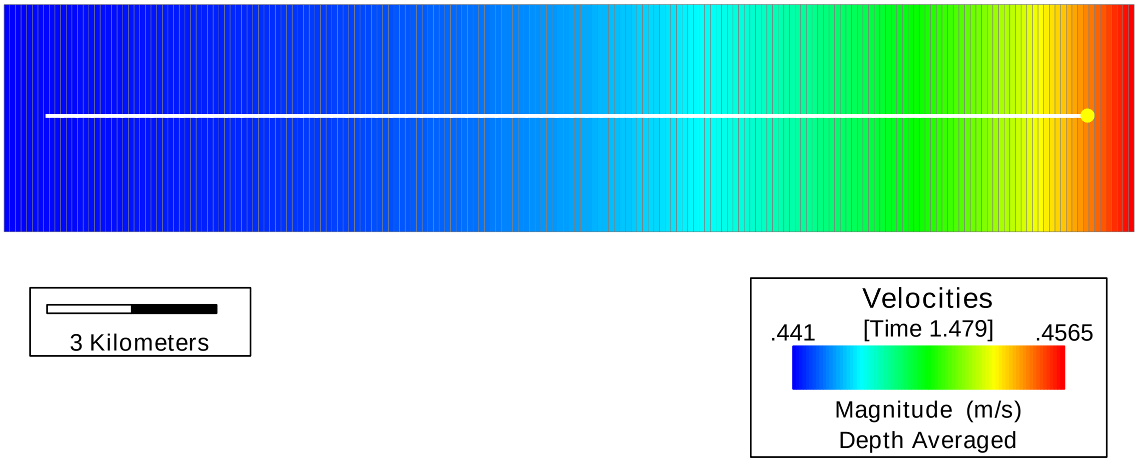


Figure 4 Trajectories of the five drifters without random walk.

Random walk was then enabled for this test case. The random walk options and diffusion coefficients are shown in Figure 5. Figure 6 shows the results for this case for the 5 particles.

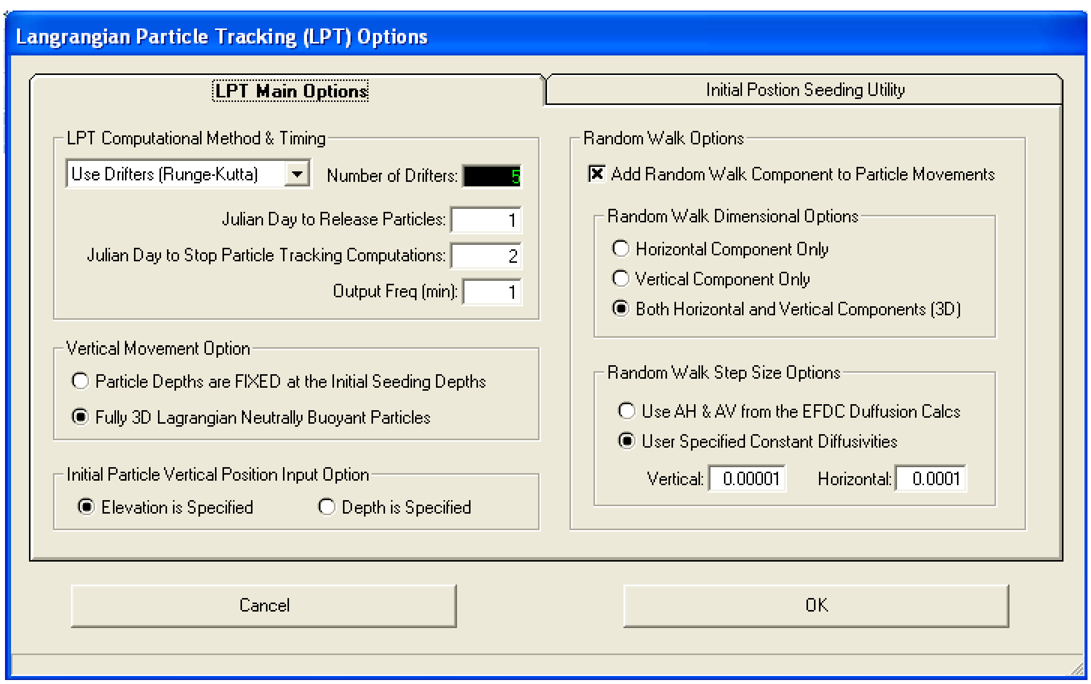


Figure 5 Test 1: Setup of LPT with random walk.

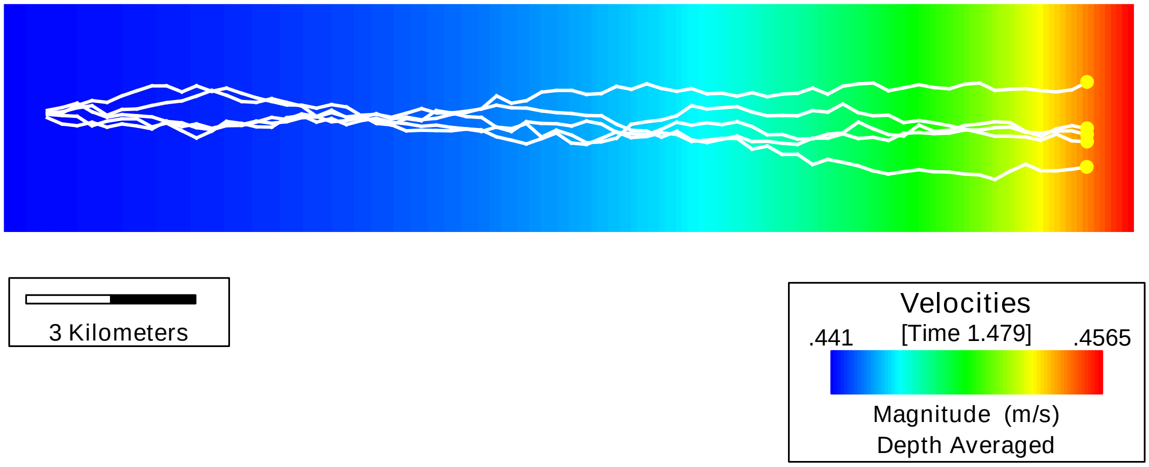


Figure 6 Trajectories of the five drifters with random walk enabled.

5.2 Test 2: Rectangle Bay

The computational domain for this test case is a rectangle with the dimensions as shown in Figure 7. The mouth of the bay is located to the west and is 200 m wide. The bay is represented by a regular grid with $\Delta x = \Delta y = 50\,m$. The vertical direction is divided into 5 equal sigma layers.

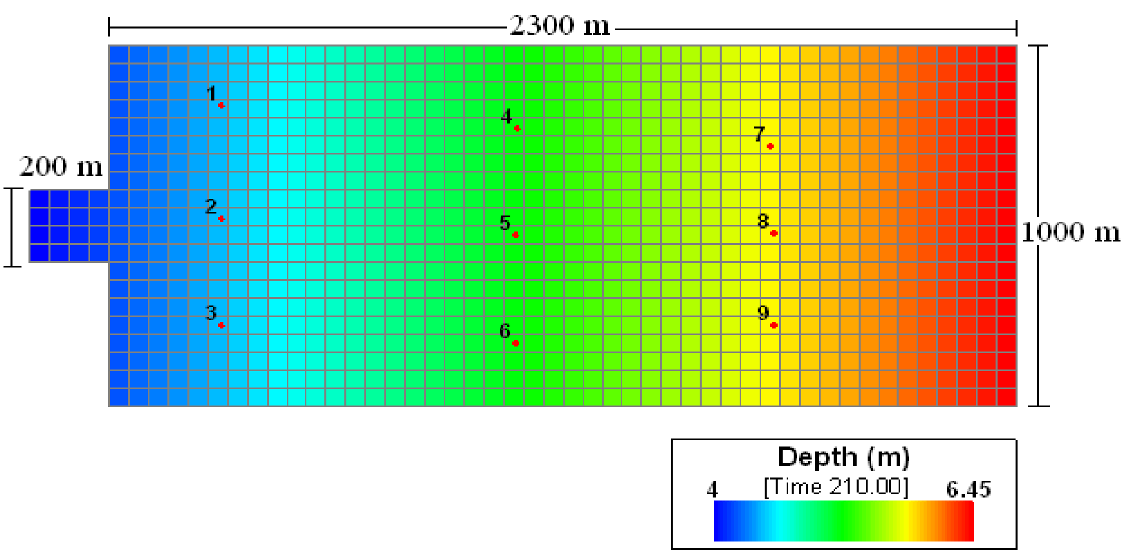


Figure 7 Test 2: Computational domain and the initial locations of 9 drifters

The bed was configured with a constant adverse slope of 0.001 (i.e. deepens away from the inlet). It is assumed that the water level at the open boundary can be represented by the harmonic function:

$$\eta(x,y,t) = ACos\left(\frac{2\pi}{T}t\right) \tag{17}$$

in which $A = 0.5\,m$, $T = 43200\,s$.

The purpose of this case was to test the solution without random movement of drifters with full 3D movement. The initial locations of 9 drifters are shown in Figure 7. The initial positions, in the coordinate system used by the model, are stored in DRIFTER.INP file.

From Figure 8 it can be seen that the initial vertical position of the drifters are specified as depth. The time to release the drifters was t=211 (Julian day) and the time to finish tracking was 215 days.

The particle trajectories for the 9 drifters over the four day period are presented in Figure 9. It can be observed that the velocity field includes vortices due to flow convergence and divergence around the inlet and the flow reversals due to the direction of tidal flow. Drifters 1, 3, 4 and 6 quickly moved towards the entrance. The drifters 2 and 5 initially moved further into the bay. Later in the simulation drifter 2 circulated back to the entrance where it exited the domain. The drifters 7-9 are located relatively far from the entrance and are under the influence of the east end eddy.

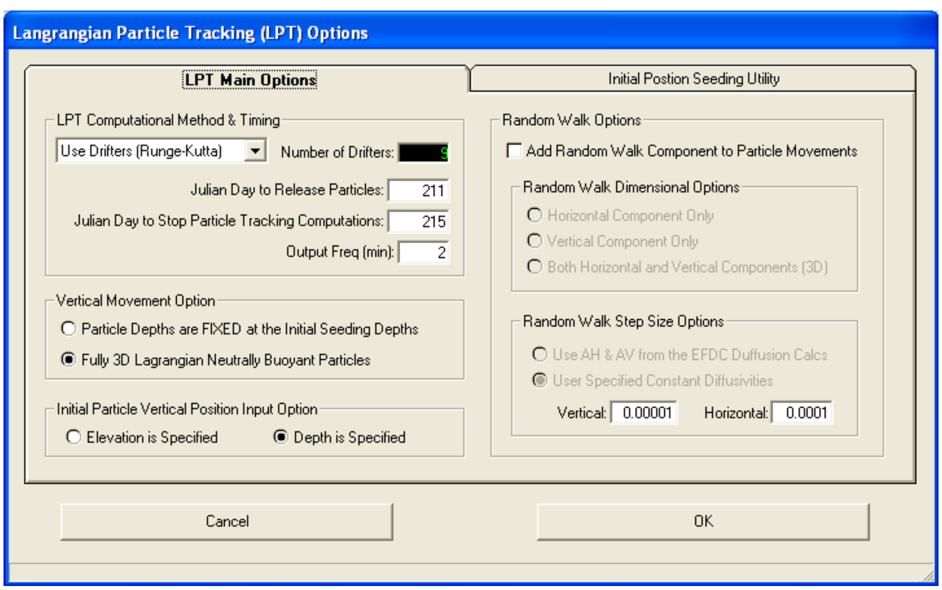


Figure 8 Test 2: Setting up of LPT without random walk.

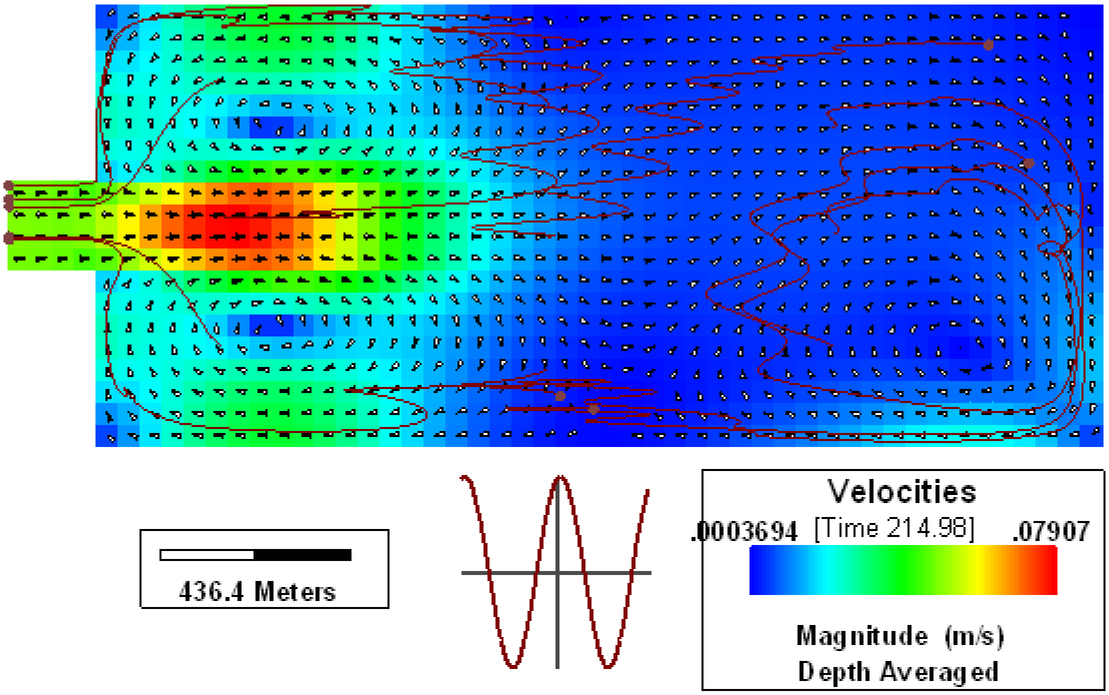


Figure 9 Velocity field and trajectories of 9 drifters over 4 days.

5.3 Test 3: Harbor_U

Harbor_U consists of a rectangular domain with flat bottom, an open boundary to the east and a flow boundary along the southwest edge. U component masks were inserted into the model to test the functionality of the Lagrangian Particle Tracks computations when masks are used. Figure 10 shows the model domain, grid and location of the U component masks.

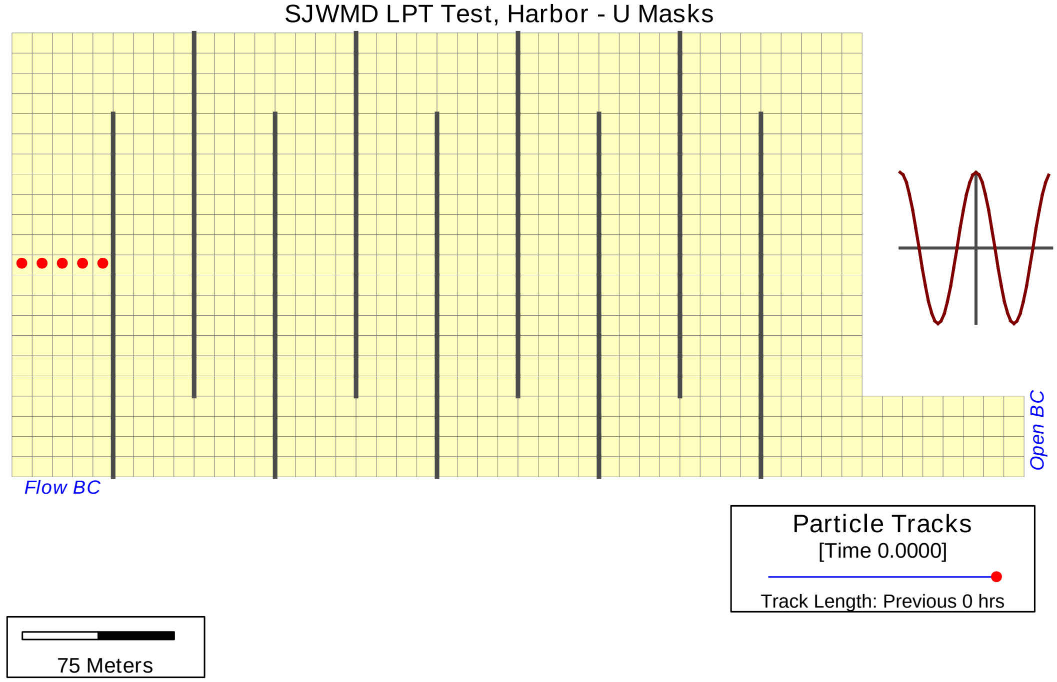


Figure 10 Test 3: Harbor_U grid showing the masks, boundaries and initial particle locations.

The initial and boundary conditions for discharge and water levels are:

$$h(t,x,y) = 5.874\,m,\ \forall x,y > 0, t = 0 \tag{18}$$

$$q(t,x,y) = 1\ m^3/s,\ \forall t > 0 \tag{19}$$

$$\eta(x,y,t) = ACos\left(\frac{2\pi}{T}t\right) \tag{20}$$

in which $A = 0.914\,m$, $T = 43200\,s$.

The model represents the vertical component as a depth averaged system with 1 sigma layer. The depths of the 5 drifters are initialized at specified depths. Two cases were considered; with the random components and without. The horizontal diffusion coefficient was assigned as 0.01 m^2/s for the with random walk case.

The results of the simulations are presented in Figures 11-12 showing the trajectories of the 5 drifters over one day. Figure 11 shows the particle tracks colored by elevation. Even though there was no vertical component, the tidal range results in changing particle elevations.

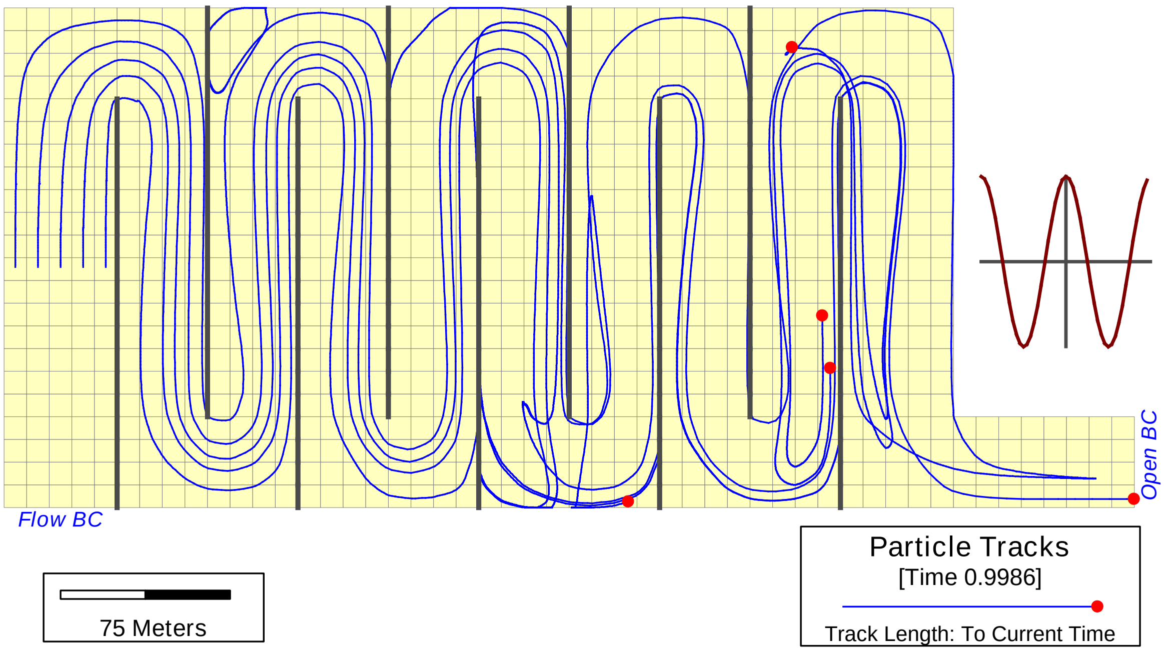


Figure 11 Harbor_U: Trajectories of 5 drifters over 1 day (no random walk).

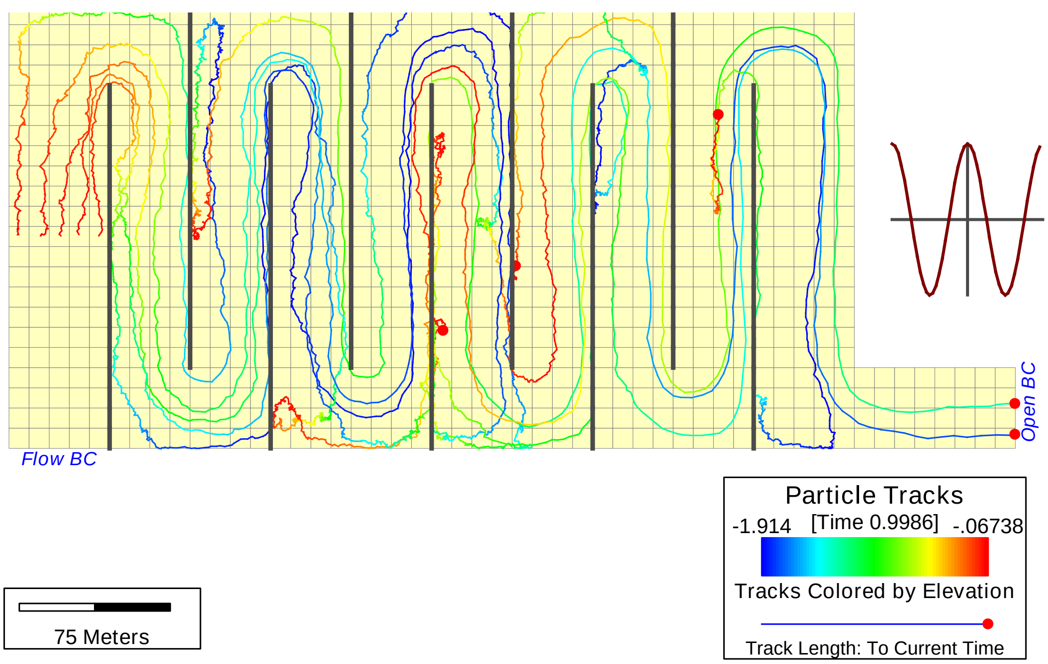


Figure 12 Harbor_U: Trajectories of 5 drifters over 1 day (random walk).

5.4 Test 4: Harbor_V

The computational domain and the hydrodynamic conditions were the same as was used with the Harbor_U test case. However, the masks tested for this case were for the V component. Fifteen drifters were initially seeded into the model domain at the locations shown in Figure 13. The particles were configured to move freely in full 3D. This model was tested using two cases; random walk disabled and with random movements enabled. The test case with random walk enabled used horizontal and vertical diffusion coefficients of 0.01 and 0.001 m^2/s respectively.

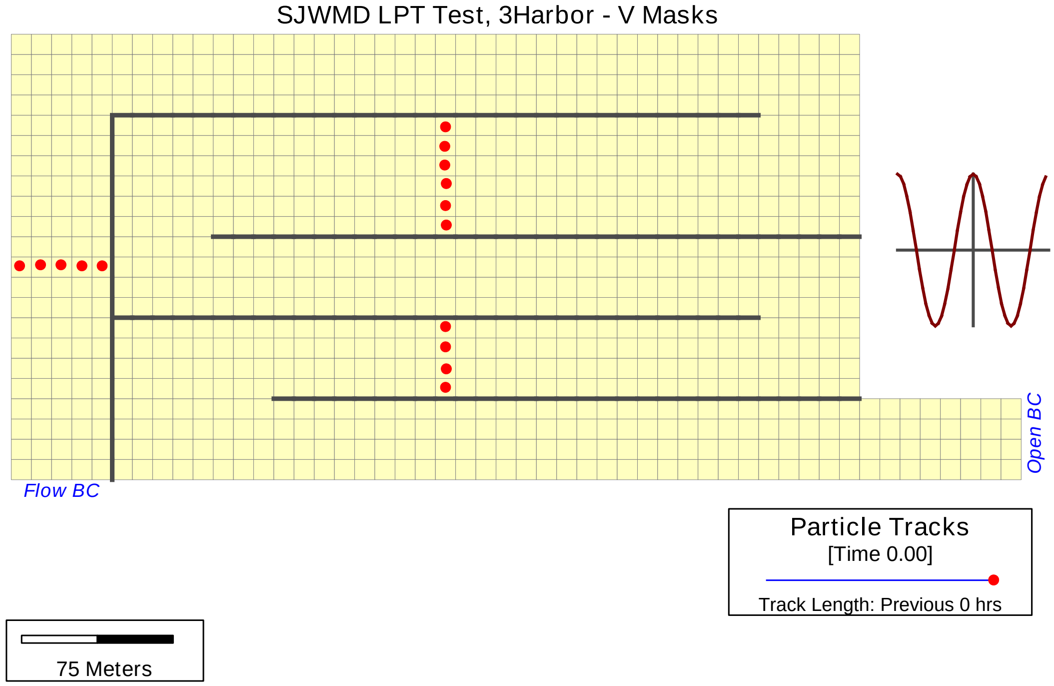


Figure 13 Test 4: Harbor_V grid showing the masks and the initial locations of the 15 drifters.

Figures 14 and 15 present the drifter trajectories for the entire model run for the two test cases. It can be seen that the drifters moved in the model domain as expected. The V masks controlled the particle movements with the particles influenced by the open boundary tidal signal and the constant inflows. similarly to the U mask test as above. However, It should be noted that in V mask test, the diffusion coefficients are very large, so the effect of random walk is quite clear.

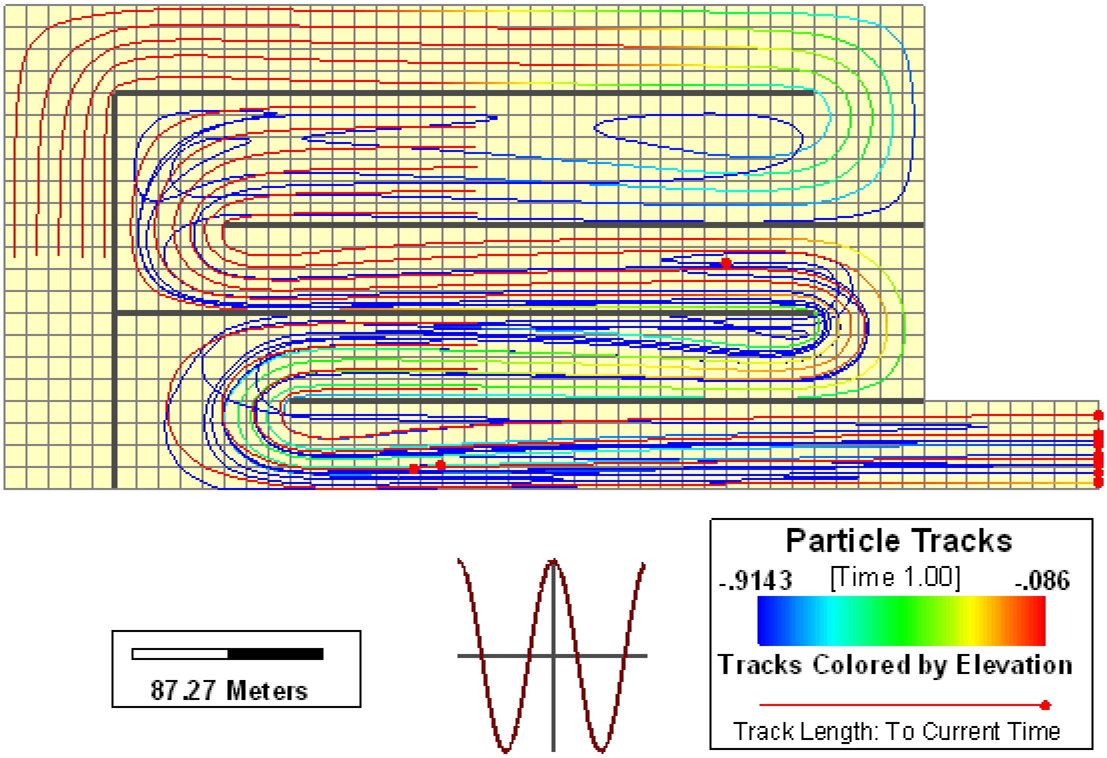


Figure 14 Harbor_V: Trajectories of the 15 drifters over 1 day (no random walk).

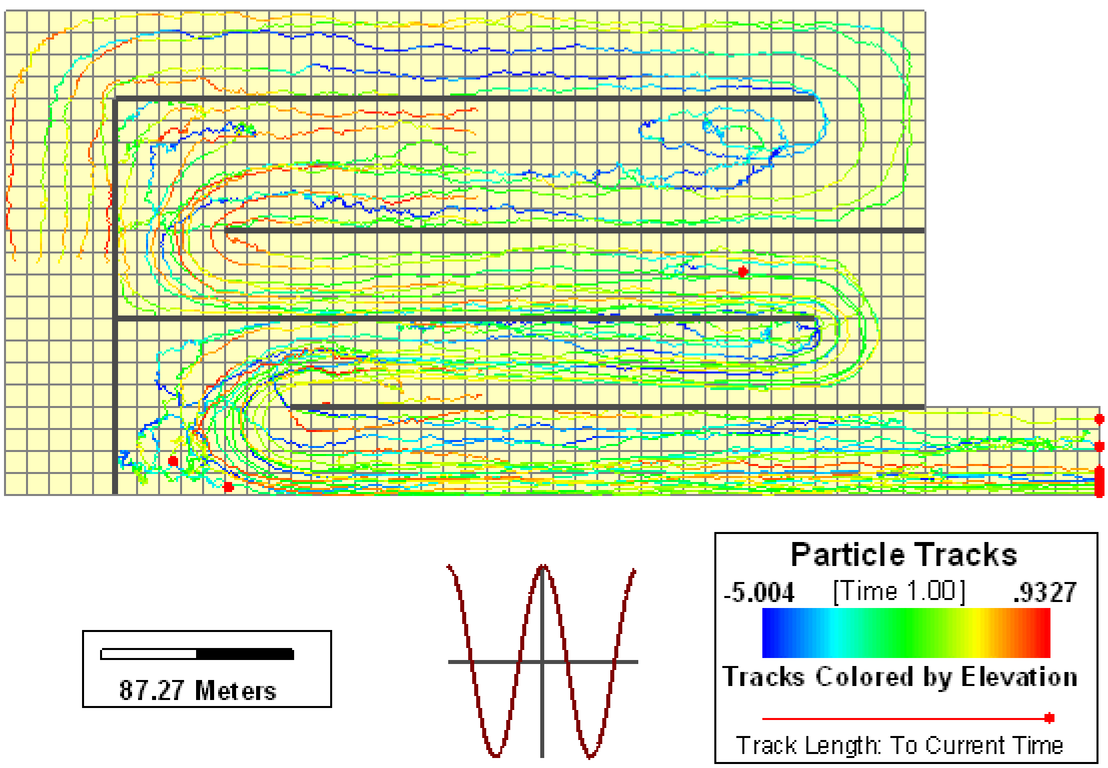


Figure 15 Harbor_V: Trajectories of the 15 drifters over 1 day (random walk).

5.5 *Test 5: Caloosahatchee Estuary*

The purpose of this test case was to test the application of the LPT sub-model with a multi layer curvilinear grid with realistic bathymetry. The model used for this effort was the model of the Caloosahatchee Estuary. This model was a 3D water quality model developed by DSI for the Florida Department of Environmental Protection (FDEP) (DSI, 2008). The model had 733 cells with four vertical layers. The initial locations of 34 drifters are shown in Figure 16. In this case the full 3D and random movement of drifters are simulated.

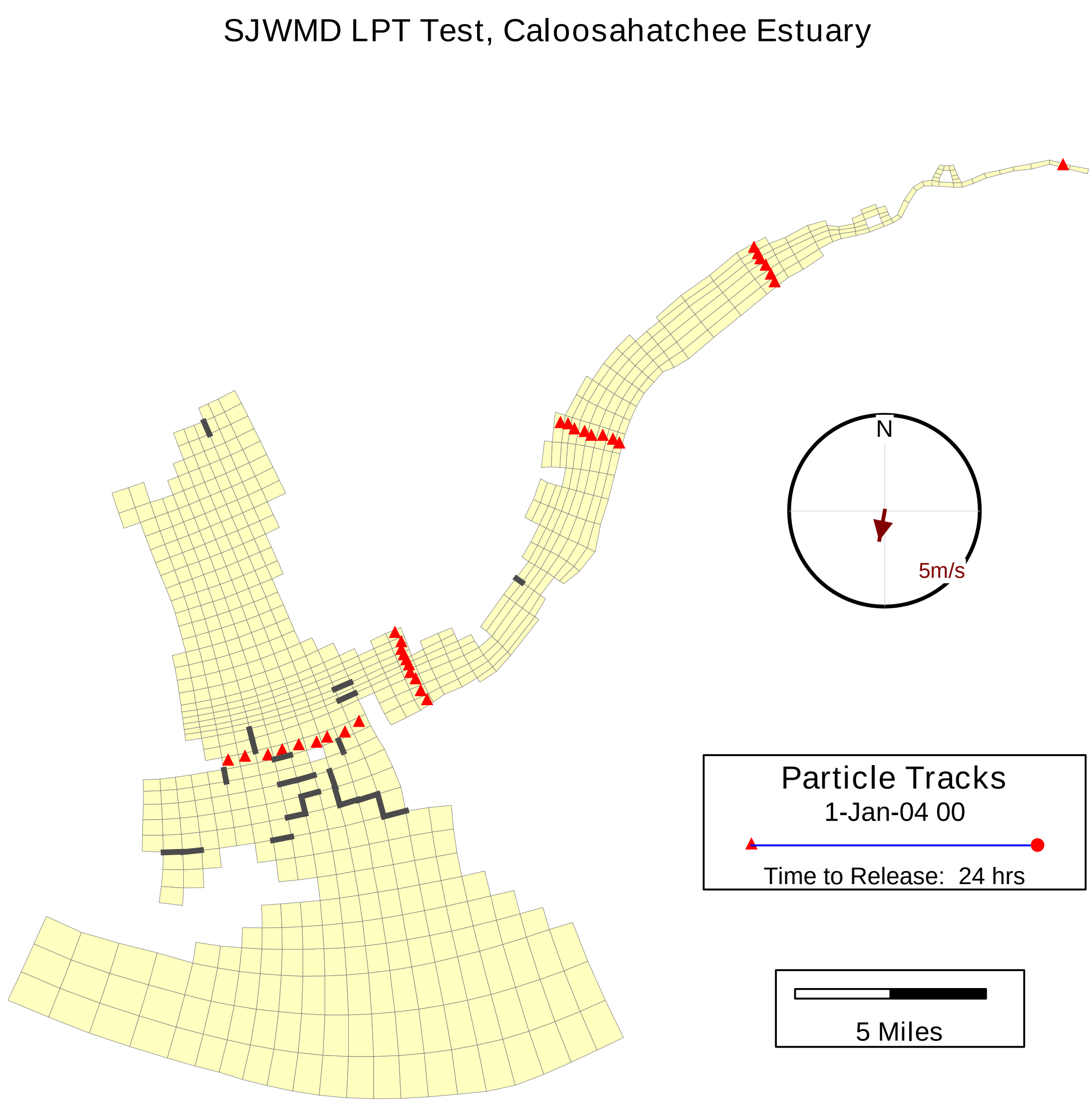


Figure 16 Test 5: Caloosahatchee grid showing the masks and the initial locations of the 34 drifters.

The trajectories of 34 drifters over the previous 24 hours are presented in Figure 17. The particles accurately followed the curvilinear grid and velocities.

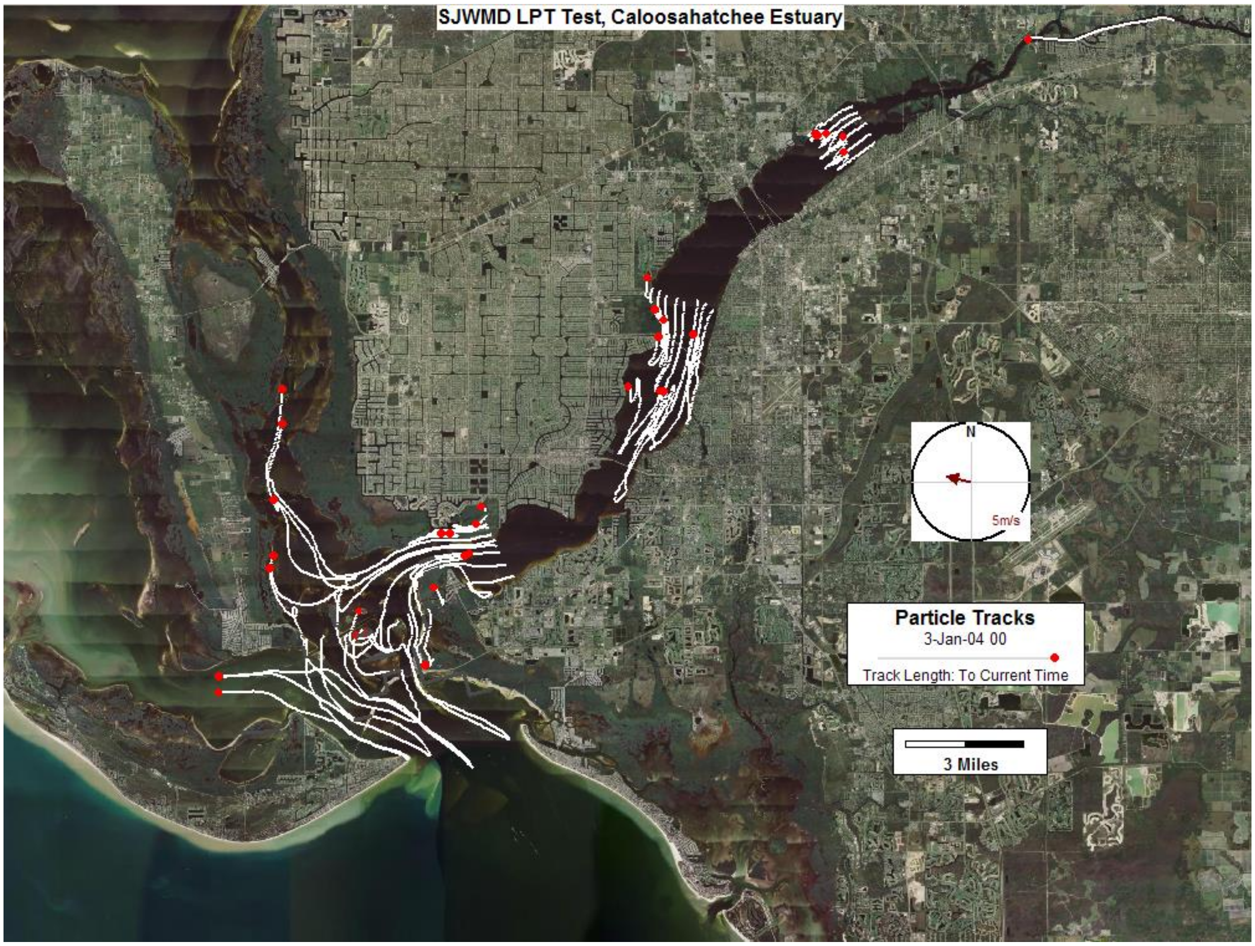


Figure 17 Caloosahatchee: Particle trajectories over the previous 24 hours.

6 Conclusions

The following summarize the conclusions of the project:

- The EFDC_DSI Fortran code has been updated to incorporate a Lagrangian Particle Drifter Fortran 90 module.
- The old, now obsolete, particle tracking code in EFDC has been removed.
- EFDC_Explorer (EE) has been updated to incorporate the pre- and post-processing of LPT's. The EE pre-processing provides full control for initial particle seeding, LPT computational option selection and plotting. The EE post-processing provides for a range of display options for the tracks, animations to the screen and or AVI files, and the ability to export any or all of the particle tracks to ASCII files.
- The LPT model has been validated through the comparison with the analytical calculation using a quasi-steady state uniform flow case with a relative error of -0.098%.
- A range of test cases were performed to evaluate the performance of the LPT sub-model using various EFDC options. A review of the LPT results from these test cases indicate that the code changes to EFDC_DSI and EFDC_Explorer worked well.
- The LPT sub-model has been implemented with the following major options:
    - Particles are free to move in full 3D,
    - Particles can be fixed at a user specified depth, and
    - A random walk component can be added to either of the two options above.
- Three numerical methods were tested to solve the advective term of the LPT sub-model. It was determined that the Runga-Kutta 4 method was preferred due to its higher level of numerical accuracy. It was determined that the computational burden of the Runge-Kutta 4 method was not significant within the overall model run rimes.

7 REFERENCES

Craig, Paul M. 2008. *User's Manual for EFDC_Explorer: A Pre/Post Processor for the Environmental Fluid Dynamics Code*. Dynamic Solutions-International, LLC, Knoxville, TN.

Dunsbergen, D.W. and Stalling, G.S. 1993, *The Combination of a Random Walk Method and a Hydrodynamic Model for the Simulation of Dispersion of Dissolved Matter In Water*, Transactions on Ecology and the Environment vol 2, WIT Press, www.witpress.com, ISSN 1743-3541.

DSI & CDM, 2008, *Caloosahatchee River Basin TMDL Model Support: Section 5 Site Specific Model Selection, Preparation, Verification, Calibration, and Validation*, FDEP, Tallahassee FL.

Hamrick, J.M., 1992a: *A Three Dimensional Environmental Fluid Dynamics Computer Code: Theoretical and Computational Aspects*. The College of William and Mary, Virginia Institute of Marine Science. Special Report 317, 63 pp.

Hamrick, J.M., 1992b: *Estuarine environmental impact assessment using a three dimensional circulation and transport model. Estuarine and Coastal Modeling*, Proceedings of the 2nd International Conference, M. L. Spaulding et al., Eds., American Society of Civil Engineers, New York, 292 303.

Hamrick, J.M., 1996: *A User's Manual for the Environmental Fluid Dynamics Computer Code (EFDC)*. The College of William and Mary, Virginia Institute of Marine Science, Special Report 331, 234 pp.

Hamrick, J.M., and T.S. Wu, 1997: *Computational design and optimization of the EFDC/HEM3D surface water hydrodynamic and eutrophication models. Chapter 16 In: Next Generation Environmental Models and Computational Methods*, G. Delic and M.F. Wheeler, Eds., Society of Industrial and Applied Mathematics, Philadelphia.

Hoffman, J.D. (2001), *Numerical Methods for Engineers and Scientists*, McGraw-Hill International Edition.

Valocchi, A. J. and Quinodoz, H. A. M. (1989). *Application of the random walk method to simulate the transport of kinetically adsorbing solutes*, Proc. of the Symposium held during the Third IAHS Scientific Assembly, Baltimore, MD, May 1989, IAHS Publ. no. 185.

Tetra Tech (2007), *The Environmental Fluid Dynamics Code, User Manual, US EPA Version 1.01*, Fairfax, VA.