Ocean Science Journal, Vol. 40 No. 4, 201-208(2005)

Article

Ocean Science Journal
http://osj.kordi.re.kr

A Second-Order Particle Tracking Method

Seok Lee*, Heung-Jae Lie, Kyu-Min Song, and Chong-Jeanne Lim
Ocean Circulation and Climate Research Division, KORDI, Ansan P.O. Box 29, Seoul 425-600, Korea

Received 22 November 2005; Revised 5 December 2005; Accepted 15 December 2005

Abstract – An accurate particle tracking method for a finite difference method model is developed using a constant acceleration method. Being assumed constant temporal and spatial gradients, the new method permits temporal-spatial variability of particle velocity. Test results in a solid rotating flow show that the new method has second-order accuracy. The performance of the new method is compared with that of other methods; the first-order Euler forward method, and the second-order Euler predictor-corrector method. The new method is the most efficient method among the three. It is more accurate and efficient than the other two.

Keywords – particle tracking, accuracy, efficiency, constant acceleration method, FDM model

1. Introduction

Particle tracking methods are practically used to track a drifting motion in a numerical model. It is applicable to determine Lagrangian trajectories or residual currents in tidal models (Foreman *et al.* 1992; Cheng and Casulli 1982). Coupled with a random-walk motion, the particle method is an efficient tool for a transport-dispersion study in coastal areas (Dimou and Adams 1993; Al-Rabeh and Gunay 1992; Suh 1998). Random walk particle tracking methods have been popularly applied to the simulation of groundwater flow and pollution (Hassan *et al.* 2001; Abulaban and Nieber 2000). It is also a practical application to predict a drifting and suspending object trajectory or an oil spill dispersion (Bennett and Clites 1987). Usually, particle displacement is calculated by linear- or bilinear- interpolated constant velocity in most cases. Through this method, employing constant tracking velocity, the calculation accuracy can be controlled by the tracking time interval (Bensabat *et al.* 2000). In the case of the Euler forward method, the first order accuracy, the tracking velocity is determined at the starting position. It is simple and easy, but there can be significant calculation errors when there is a strong variability in the current field. In the Euler predictor-corrector method, the second order method, the particle velocity is determined at the center between the starting and finishing position calculated by the Euler forward method. In the iterative method, particle velocity is determined by conversed velocity from iteration of the Euler predictor-corrector steps. In the fourth order, the Runge-Kutta method, the particle velocity is evaluated four times in terms of the tracking step. This method is very accurate but highly time consuming.

In this study, we propose a new second-order method. In this method, the particle displacement is determined by the initial velocity at the starting position as well as the constant acceleration. This method is similar to that of Bennett and Clites (1987), where the spatial gradient is considered constant. Bennett and Clites (1987) applied the particle tracking to predict oil spill dispersion accompanied by a steady current field. The new method is applicable to the unsteady current field. We develop the algorithm for the finite difference method model and analyze the accuracy and efficiency of the method in the artificial field.

2. Methods

Assume that we have obtained the non-uniform unsteady

*Corresponding author. E-mail: lees@kordi.re.kr

velocity field calculated using a numerical model. In a 2-dimensional x-y plane, the velocity is situated on a spatial grid point with a discrete temporal interval of simulation time step. The particle velocity for position R at time t can be determined by linear interpolation in time (Appendix A).

$$V = u(R(t), t)\vec{i} + v(R(t), t)\vec{j} \tag{1}$$

where, $R = x\vec{i} + y\vec{j}$, $\vec{i}$ and $\vec{j}$ are unit vectors, u and v are velocity components of x and y direction, respectively.

The particle's displacement during one numerical simulation time step ($T_1 \le t \le T_2$) can be expressed as the integration of particle velocity.

$$R_2 = R_1 + \int_{T_1}^{T_2} V(R(t), t)dt \tag{2}$$

where, R_1 and R_2 are the particle's position vectors at time T_1 and T_1, respectively.

One simulation time step can be divided into several tracking time increments to reduce the calculation error (Bensabat *et al.* 2000).

$$R_2 = R_1 + \sum_{j=1}^{N} \Delta r_j = R_1 + \sum_{j=1}^{N} \int_{t_{j-1}}^{t_j} V(R, t)dt \tag{3}$$

where, $t_i = i\Delta t + T_1$, $i=0,...,N$, $\Delta t = (T_2 - T_1)/N$, $\Delta r (= \Delta x\vec{i} + \Delta x\vec{j})$ is the displacement of particles during an increment, N is the tracking increment number.

Equation (3) is nonlinear when particle velocity is unsteady and non-uniform (Bensabat *et al.* 2000). Linearization is required to solve the Equation (3). Generally, a mean tracking velocity is assumed in many methods. Particle velocity is considered constant in each time increment:

$$\Delta r_j = \bar{V}_j \Delta t \tag{4}$$

In the Euler forward method, the mean tracking velocity is resumed as velocity at the location of the starting point (r_{start}).

$$\bar{V}_j = V(r_{start}, t_{j-1/2}) = const \tag{5}$$

where, $t_{j-1/2}$ denotes mid time of the increment.

However, in the Euler forward method, first order accuracy has resulted in relatively large calculation errors when there are significant spatial-temporal variations in the velocity field. In the Euler predictor-corrector method, second order accuracy, mean tracking velocity is resumed as average velocity at the starting point and trial end point calculated by the Euler forward method (predictor step). The new end position of the particle is predicted using the corrected velocity (corrector step).

$$\bar{V}_j = \{V(r_{start}, t_{j-1/2}) + V(r_{end}, t_{j-1/2})\}/2 = const \tag{6}$$

where, r_{end} is the trial end position calculated at the predictor step.

In this study, we assume a constant acceleration to solve the equation (3). When a particle moves with a constant acceleration, the displacement during a tracking time increment can be determined by initial velocity and constant acceleration as follows:

$$\Delta r_j = V(r_{start}, t_{j-1})\Delta t + \frac{1}{2} a \Delta t^2 \tag{7}$$

where, $V(r_{start}, t_{j-1})$ is the initial velocity at the starting position and time, a is the constant acceleration vector in the tracking time increment.

The accuracy of the method is closely correlated to defining the acceleration. The acceleration, the time derivative of velocity, can be decomposed into local and convective accelerations on the 2-dimensional x-y plane as follows:

$$a = \frac{\partial V}{\partial t} + \bar{u}\frac{\partial V}{\partial x} + \bar{v}\frac{\partial V}{\partial y} \tag{8}$$

Three derivative terms in Equation (8) are determined as linear velocity gradients (Appendix B.). We assume that three gradient terms are constant in a tracking time increment. The unknown mean advective velocities ($\bar{u}$, $\bar{v}$) are considered as the displacement per unit time. Then Equation (8) can be rewritten as follows:

$$a = a^x\vec{i} + a^y\vec{j} \tag{9}$$

$$a^x = u_t + \frac{\Delta x}{\Delta t}u_x + \frac{\Delta y}{\Delta t}u_y \tag{10}$$

$$a^y = v_t + \frac{\Delta x}{\Delta t} v_x + \frac{\Delta y}{\Delta t} v_y \qquad (11)$$

Here, superscript indicates the vector component and the subscript indicates the derivative.

With Equations (10) and (11) Equation (7) can be reformed as linear equations as follows:

$$\Delta x = u_0 \Delta t + \frac{1}{2}\left(u_t + \frac{\Delta x}{\Delta t} u_x + \frac{\Delta y}{\Delta t} u_y\right)\Delta t^2 \qquad (12)$$

$$\Delta y = v_0 \Delta t + \frac{1}{2}\left(v_t + \frac{\Delta x}{\Delta t} v_x + \frac{\Delta y}{\Delta t} v_y\right)\Delta t^2 \qquad (13)$$

where, u_0 and v_0 are initial velocities of x and y direction, respectively.

Without time gradient terms Equations (12) and (13) are similar with the second-order method of Bennett and Clites (1987). They derived the formulas from Taylor's series expansion and are applied on a steady state condition. For unsteady conditions, the time gradient terms can work as a correction of time variations during a time increment. Linear Equations (12) and (13) and simultaneous equations of Δx and Δy can be rewritten as follows:

$$\left(1 - \frac{\Delta t}{2} u_x\right)\Delta x - \frac{\Delta t}{2} u_y \; \Delta y = \left(u_0 + \frac{\Delta t}{2} u_t\right)\Delta t \qquad (14)$$

$$-\frac{\Delta t}{2} v_x \; \Delta x + \left(1 - \frac{\Delta t}{2} v_y\right) \; \Delta y = \left(v_0 + \frac{\Delta t}{2} v_t\right)\Delta t \qquad (15)$$

Equations (14) and (15) can be solved using a simple matrix inversion algorithm.

3. Test Results

For the accuracy test, we use an artificial flow field in the manner of a C-grid system. The velocity components of x and y are set on the left face and the lower face of a rectangular cell, respectively. In the 40×40 grids domain, the flow rotates around the center of the domain (Fig. 1).

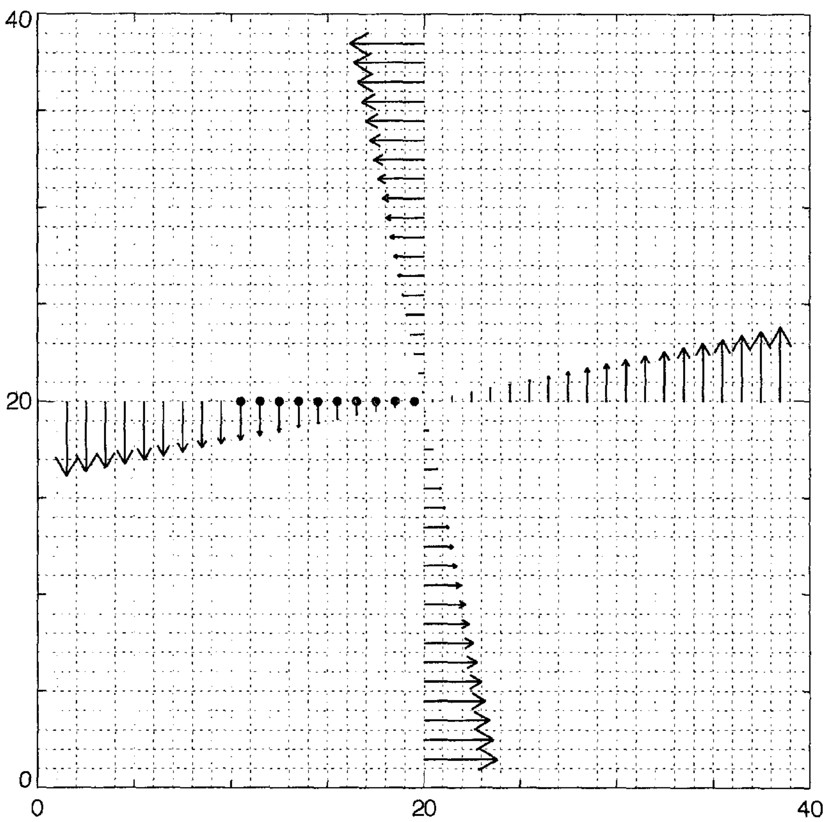


Fig. 1. Domain and flow field for comparison of the three particle tracking methods. The flow rotates around the center of the domain with a constant angular velocity of 1 cycle per 30 seconds. Initially, 10 particles are released on the dark solid circles.

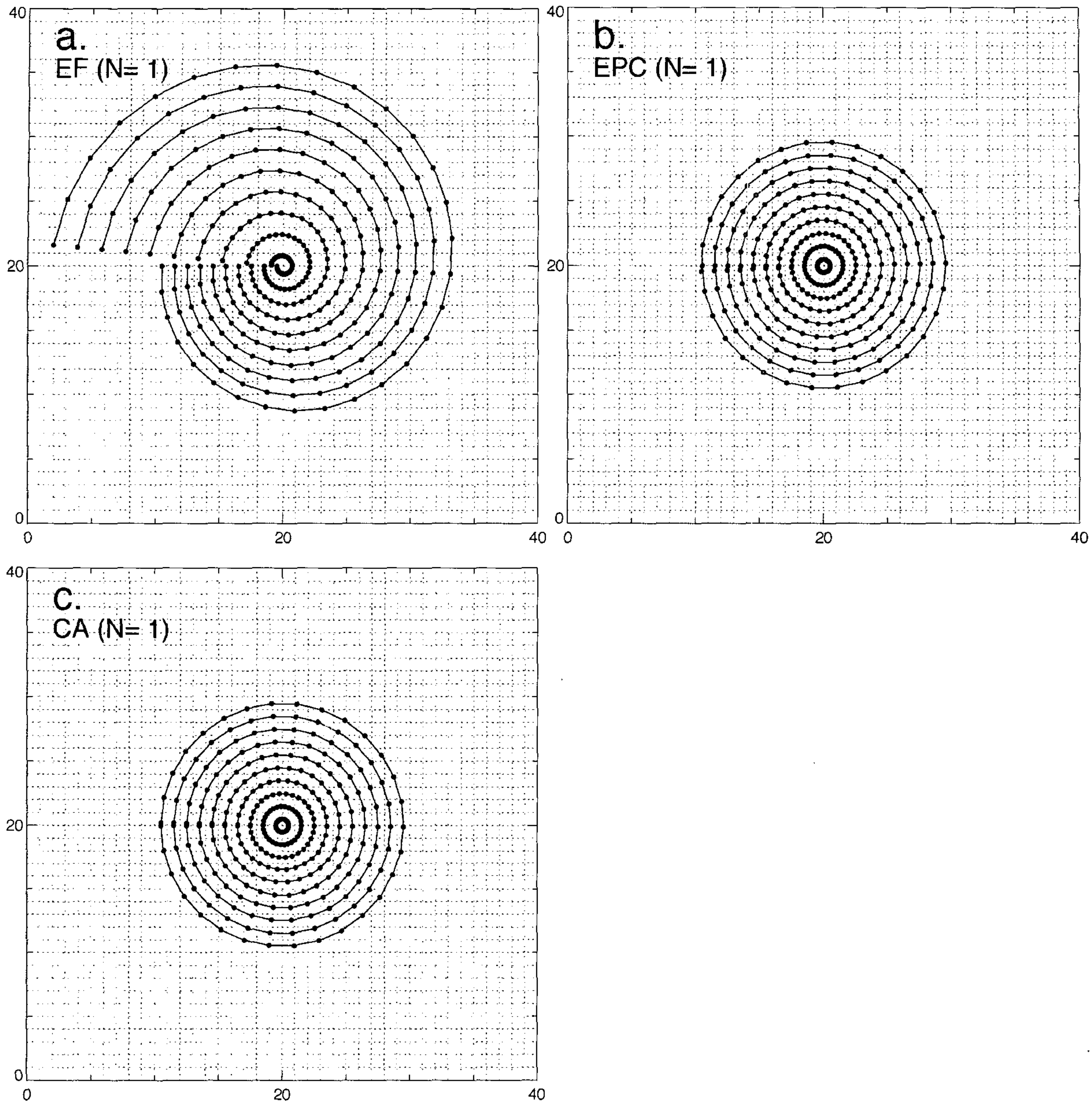


Fig. 2. Trajectories of 10 particles during 1 rotation cycle calculated by three particle tracking methods; (a) The EF method, (b) EPC and (c) CA. The tracking increment number is 1 (N=1).

The grid interval of all cells is 1m and the constant angular velocity of the flow is 1 cycle per 30 seconds. This means that if there is no error, a particle should return to the initial position 30 second later. We compare the accuracy of three particle-tracking methods: the Euler forward method (EF), Euler predictor-corrector method (EPC) and constant acceleration method (CA). For the test simulation, we release 10 particles on initial positions marked in Fig. 1. The simulation time step is fixed to 1 second and the interval for the tracking time increment (Δt) is controlled by the tracking increment number (N). Figure 2 shows the trajectories of 10 particles estimated by the three methods during a 30-second period, when the tracking increment number is 1. End positions of all trajectories do not exactly coincide with the initial positions. The calculation error for each method can be evaluated by the mismatch-distance between the initial and end positions. The EF method has the largest error error (Figure 2a) and

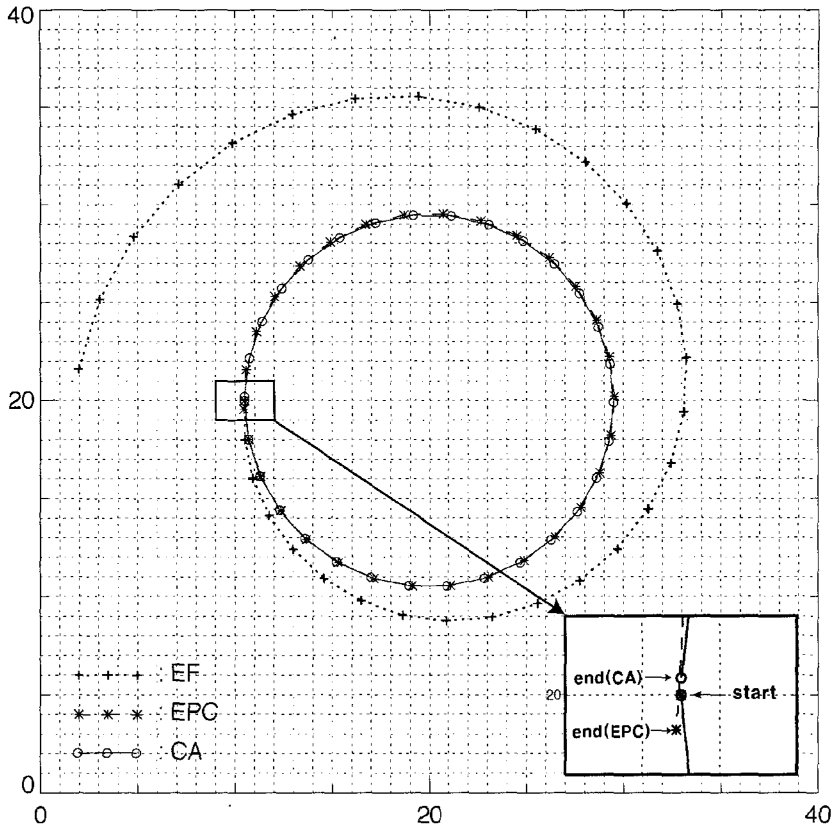


Fig. 3. Overlapped trajectories calculated by three particle tracking method.

CA method has the smallest (Figure 2c). The error for the EPC method is similar to that of the CA method (Figure 2b). In all these methods, the error of the outer trajectory is larger than that of the inner trajectory because the error is proportional to the particle velocity. Figure 3 shows the overlap of the largest trajectories in the three methods. The trajectory of the EF method shows outward drifting. On the contrary, the trajectories of the CA and the EPC methods show almost constant radii. The trajectory from the EPC method slightly leads exact solution. However, the trajectory of the CA method is slightly delayed. Differences in the error tendencies were pointed out by Bennett and Clites (1987). The first-order EF method is accompanied by amplitude and phase errors. However, the "second-order trapezoidal method" in their study (similar to the CA method in this study) is accompanied only by a phase error. Figure 4 shows trajectories of the 10 particles as in Figure 2, except that the tracking increment number is 3. All results in Figure 4 show less significant errors than those in Figure 2. The end positions of CA and EPC methods are almost the initial positions. This means that the calculation error is inversely related to the particle tracking time interval. For a quantitative comparison, the normalized calculation error is defined by the distance rate between the total length of the trajectory and the mismatch distance:

$$Er = \frac{\sqrt{(x_{int} - x_{end})^2 + (y_{int} - y_{end})^2}}{L} \quad (16)$$

where, L is the total length of a trajectory, (x_{int}, y_{int}) and (x_{end}, y_{end}) are the initial and end positions, respectively.

The normalized error defined by Equation (16) is independent on the particle velocity, but only dependent on the method and tracking increment number. Figure 5 shows the variation of normalized errors according to the tracking increment number. When the tracking increment number is 1, the error involving the EF method is about 14 % and those of the EPC and the CA methods are

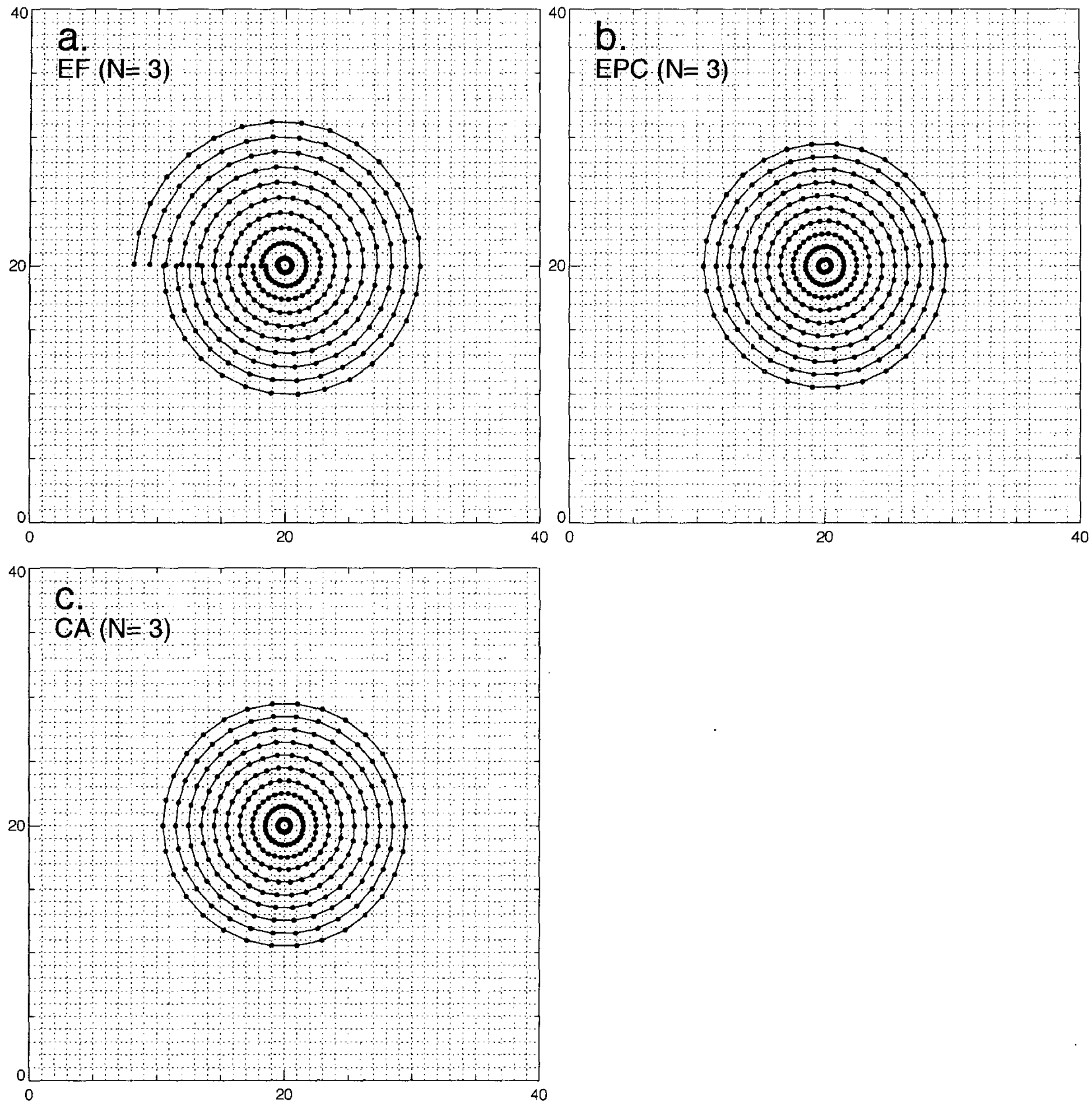


Fig. 4. Trajectories of 10 particles during 1 rotation cycle calculated by three particle tracking methods; (a) The EF method, (b) EPC and (c) CA. The tracking increment number is 3 (N=3).

about 0.70 % and 0.35 %, respectively. In these three methods, errors rapidly decrease as the tracking increment number increases.

Figure 6 shows the accuracy of the three methods. If a method is m^{th}-order accuracy, error must be proportional to the m^{th}-power of the tracking time interval (Δt);

$$Er = \alpha \Delta t^m = \alpha\left(\frac{\Delta T}{N}\right)^m \quad (17)$$

where, α is a constant number, ΔT is a constant simulation time interval, N is a tracking increment number.

We take a log for both sides of the Equation (17);

$$\log_{10}(Er) = \log_{10}(\alpha \Delta T^m) - m\log_{10}(N) \quad (18)$$

The slope m in the log scale denotes the accuracy of a method. In Figure 6, the slope of the EF method is about -1. This means the EF method is the first-order

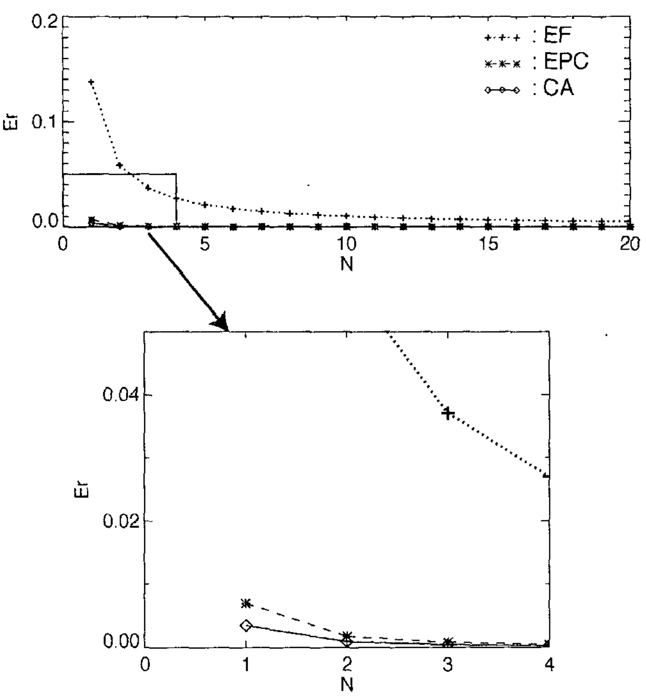


Fig. 5. Variation of normalized errors according to the tracking increment number.

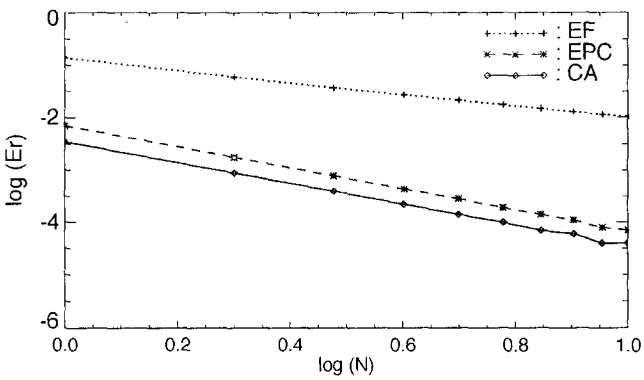


Fig. 6. Comparison of normalized errors. The horizontal and vertical axes are in log scale.

accuracy. On the other hand, the EPC and the CA methods are the second-order accuracy and their slopes arrive at about -2. Table 1 shows the summary of comparison results.

4. Conclusions and Discussions

The CA method is the most accurate particle tracking method among the three methods and the EF method is the most inaccurate. The computation time for the CA method is about 114% of that for the EF method based on our study. The computation time-cost of the CA method is relatively smaller then that of the EPC method (about 171% of that of the EF method) though the algorithm of the EPC method is simpler than that of the CA method. For the practical application, the particle tracking method requires extra time-consuming steps for the determination of the grid number, the verification of landward over-shooting, for adding random-walk motion, etc. Usually, the computation time for those steps is longer than the computation time for particle tracking itself. When the extra steps are reduced for specific applications, the EPC method can be faster than the CA method. If EF methods are applied to the calculation of the Lagrangian residual in a tidal model, it can cause significant errors. The EF method requires much more time than the CA (or EPC) method to maintain similar accuracy. Therefore, we can conclude that the CA method is the most efficient method among the three for practical applications. When one gives much more weight to accuracy, the fourth-order Runge-Kutta method can be used although it needs much more time. For the test of the CA method, we assume the flow field has been derived from a finite difference method model and has introduced simple interpolation algorithms of velocity gradients from the flow field. The CA method is also applicable to a finite element method model, when one has determined appropriate velocity gradients from a nodal velocity field.

Table 1. Summary of comparison results.

Method	$Er(\times 10^{-5})$				Accuracy (m)	Computation time rate
	N=1	N=3	N=5	N=10		
EF	135795	3702	2125	1028	-1.08	100 %
EPC	696	77	28	7	-2.00	171 %
CA	345	39	14	4	-2.00	114 %

Acknowledgement

We are grateful to two reviewers for their valuable comments and suggestions for improving this manuscript. This study is supported by the Ministry of Maritime Affairs and Fisheries (MOMAF) under the KORDI contract PM320-03.

References

Abulaban, A. and J.L. Nieber. 2000. Modeling the effects of nonlinear equilibrium sorption on the transport of solute plumes in saturated heterogeneous porous media. *Adv. Water Res.*, **23**, 893-905.

Al-Rabeh, A.H. and N. Gunay. 1992. On the application of a particle dispersion model. *Coastal Eng.*, **17**, 195-210.

Bennett, J.R. and A.H. Clites. 1987. Accuracy of trajectory calculation in a finite-difference circulation model. *J. Comp. Phys.*, **68**, 272-282.

Bensabat, J., Q. Zhou, and J. Bear. 2000. An adaptive pathline-based particle tracking algorithm for the Eulerian-Lagrangian method. *Adv. Water Res.*, **23**, 383-397.

Cheng, R.T. and V. Casulli. 1982. On Lagrangian residual currents with applications in South San Francisco Bay, California. *Water Resour. Res.*, **18**(6), 1652-1662.

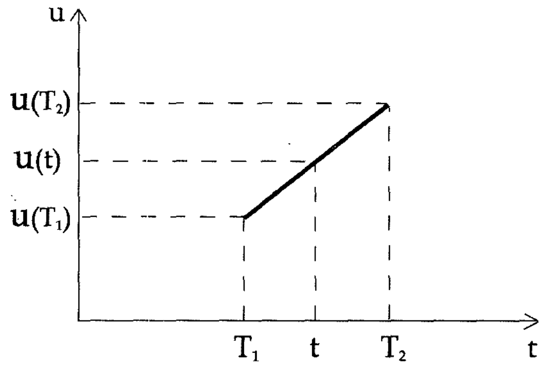
Dimou, K.N. and E.E. Adams. 1995. A random-walk, particle tracking model for well-mixed estuaries and coastal waters. *Estuar. Coast. Shelf Sci.*, **37**, 99-110.

Foreman, M.G.G., A.M. Baptista, and R.A. Walters. 1992. Tidal model studies of particle trajectories around a shallow coastal bank. *Atmos. Ocean*, **30**(1), 43-69.

Hassan, A.E., R. Andricevic, and V. Cvetkovic. 2001. Computational issues in the determination of solute discharge moments and implications for comparison to analytical solutions. *Adv. Water Res.*, **24**, 607-619.

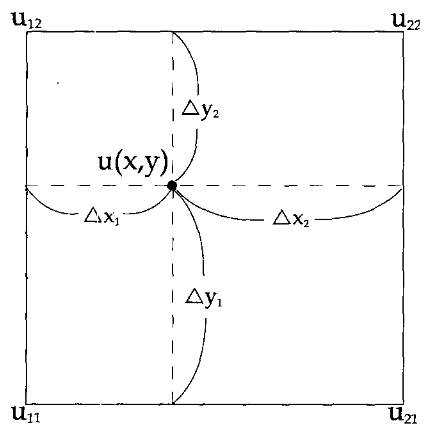
Suh, S.W. 1998. Thermal dispersion analysis using semi-active particle tracking in near field combined with two-dimensional Eulerian-Lagrangian far field model. *J. Kor. Soc. Coast. Ocean Eng.*, **10**(2) 73-82. (In Korean)

Appendix A. Linear interpolation scheme in time.



$$u(t) = \frac{u(T_1)\cdot(T_2-t)+u(T_2)\cdot(t-T_1)}{T_2-T_1}$$

Appendix B. Bi-linear interpolation schemes in space for the Equation (8).



$$u(x,y) = \frac{(u_{11}\Delta x_2\Delta y_2 + u_{21}\Delta x_1\Delta y_2 + u_{12}\Delta x_2\Delta y_1 + u_{22}\Delta x_1\Delta y_1)}{\Delta x\cdot\Delta y}$$

$$u_x(x,y) = \frac{(u_{21}-u_{11})\Delta y_2 + (u_{22}-u_{12})\Delta y_1}{\Delta x\cdot\Delta y}$$

$$u_y(x,y) = \frac{(u_{12}-u_{11})\Delta x_2 + (u_{22}-u_{21})\Delta x_1}{\Delta x\cdot\Delta y}$$

Where, $\Delta x = \Delta x_1 + \Delta x_2$, $\Delta y = \Delta y_1 + \Delta y_2$.