INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids **26**: 677–696 (1998)

PARALLEL COMPUTATION OF ATMOSPHERIC POLLUTANT DISPERSION UNDER UNSTABLY STRATIFIED ATMOSPHERE

C.H. LIU AND D.Y.C. LEUNG*

Department of Mechanical Engineering, 7/F, Haking Wong Building, The University of Hong Kong, Pokfulam Road, Hong Kong

SUMMARY

Pollutant dispersion under unstably stratified atmosphere was investigated numerically using the finite element method. The effects of atmospheric stability on plume trajectory were studied using a three-dimensional second-order closure dispersion model. The numerical model was implemented using domain decomposition method and carried out using a parallel computer. The computation accelerates significantly and the size of computation can be largely increased as a result of the parallelism. A passive contaminant point source was placed at the middle of the convective boundary layer to simulate the atmospheric dispersion. The requirement of the input of dispersion coefficients in k-theory and Gaussian models was replaced with direct input of turbulence flow data. It was found that the present numerical model can predict several non-Gaussian plume behaviours and the computed results agreed well with findings from experimental observations. © 1998 John Wiley & Sons, Ltd.

KEY WORDS: second-order closure model; pollutant dispersion; finite element method; unstably stratified atmosphere; parallel computation

1. INTRODUCTION

Atmospheric dispersion of contaminants in convective boundary layer is a current issue in air pollution studies because of the important consequences it may bring to our environment. Experience indicates that it is difficult to accurately predict the transport phenomenon of contaminants in the atmosphere, which is a function of the source conditions, meteorological factors and geographical locations. The Gaussian plume model is so far the most commonly used dispersion model in the world. However, laboratory experiments [1,2] found that several plume behaviours, such as the descent and rise of plume trajectory in an unstably stratified atmosphere, could not be accurately described by Gaussian and k-theory dispersion models. Also, these models use empirically determined turbulent dispersion coefficients to account for the effects of wind fluctuations, which, however, are site specific and thus introduce error to the modelling. The use of a higher order closure model can overcome the above weaknesses and opens up new avenues in this research area.

Second-order closure model is the one which has distinct capabilities in dispersion modelling. Lewellen [3] derived a second-order closure dispersion model from the exact Reynolds

* Correspondence to: Department of Mechanical Engineering, 7/F, Haking Wong Building, The University of Hong Kong, Pokfulam Road, Hong Kong. Tel.: +852 28597911; Fax: +852 28585415; E-mail: ycleung@hkucc.hku.hk

CCC 0271–2091/98/060677–20$17.50
© 1998 John Wiley & Sons, Ltd.
Received July 1996
Revised May 1997

stress equation, which accurately compares the spreading rate of pollutant with experimental observation of Pasquill and Smith [4] under neutrally stratified conditions. Sykes *et al.* [5] presented a system of transport equations for the second-order correlations of passive scalar fluxes and demonstrated its ability to predict dispersion characteristics in a wind tunnel experiment. Lamb [6] developed a Lagrangian diffusion model and obtained several non-Gaussian features. The results are also comparable to laboratory experiments [7–9]. Recently, Liu and Leung [10] computed the crosswind integrated pollutant concentration for both the horizontal and vertical plane in an unstably stratified atmospheric boundary layer, which compared well with experimental results.

The above results show the principal strength of the second-order closure model in pollutant dispersion modelling over the Gaussian approach. Furthermore, the direct use of turbulent flow data instead of empirical eddy diffusivities or dispersion coefficients makes it superior to the k-theory and Gaussian models in studying the effect of background turbulence on plume dispersion. It can provide a general framework within which complex flow and dispersion effects can be considered simultaneously.

It is recognized from literature that the major shortcoming of second- and higher-order closure models is the solving of more non-linear, coupled partial differential equations than k-theory and Gaussian dispersion models. In three-dimensional study, four equations are needed for a second-order closure model while only one is needed for a k-theory dispersion model. Therefore, the implementation of a second-order closure model into computation code is both memory and computation time intensive. Its usage is severely hampered by sequential computing. However, with the exploitation of parallel or concurrent computers, the large computation load for solving the complicated equations can be vastly improved.

The principle of parallel or concurrent computing is the use of many processors working cooperatively on a single problem. This is one of the technological advancements that influences large scale scientific and engineering computation. The use of parallel computation is one of the solutions for solving the above two barriers of implementation of second-order closure model.

In this paper, the methodology and procedures in parallelizing a second-order closure dispersion model will be discussed.

2. MATHEMATICAL MODEL

The mathematical model used in the present study is based on the following homogeneous equations as described elsewhere [3,11].

Transport equation for mean scalar concentration c:

$$\frac{\partial c}{\partial t} + U\frac{\partial c}{\partial x} = -\frac{\partial}{\partial y}\overline{v'c'} - \frac{\partial}{\partial z}\overline{w'c'}. \tag{1}$$

A second-order closure scheme is adopted to close the above equation, which involves the following second moment transport equations for the fluxes $\overline{v'c'}$ and $\overline{w'c'}$:

$$\frac{\partial}{\partial t}\overline{v'c'} + U\frac{\partial}{\partial x}\overline{v'c'} = -\overline{v'^2}\frac{\partial c}{\partial y} + a\frac{\partial}{\partial y}\left(q\Lambda\frac{\partial}{\partial y}\overline{v'c'}\right) + a\frac{\partial}{\partial z}\left(q\Lambda\frac{\partial}{\partial z}\overline{v'c'}\right) - \frac{Aq}{\Lambda}\overline{v'c'}, \tag{2}$$

© 1998 John Wiley & Sons, Ltd.

$$\frac{\partial}{\partial t}\overline{w'c'} + U\frac{\partial}{\partial x}\overline{w'c'} = -\overline{w'^2}\frac{\partial c}{\partial z} + \frac{g}{T_o}\overline{c'\theta'} + a\frac{\partial}{\partial y}\left(q\Lambda\frac{\partial}{\partial y}\overline{w'c'}\right) + a\frac{\partial}{\partial z}\left(q\Lambda\frac{\partial}{\partial z}\overline{w'c'}\right) - \frac{Aq}{\Lambda}\overline{w'c'}. \tag{3}$$

Covariance of concentration and temperature $\overline{c'\theta'}$ is required to close the system of transport equations as follows:

$$\frac{\partial}{\partial t}\overline{c'\theta'} + U\frac{\partial}{\partial x}\overline{c'\theta'} = -\overline{w'c'}\frac{\partial T}{\partial z} - \overline{w'\theta'}\frac{\partial c}{\partial z} + a\frac{\partial}{\partial z}\left(q\Lambda\frac{\partial}{\partial z}\overline{c'\theta'}\right) - 2bs\frac{q}{\Lambda}\overline{c'\theta'}. \tag{4}$$

3. COMPUTATION DOMAIN AND BOUNDARY CONDITIONS

The computation domain is shown in Figure 1. To economize the computer resources, the pollutant dispersion problem is assumed to be symmetrical about the centreline plane of the pollutant source, i.e. $y = 0$.

Gaussian distribution for c was used as the left-most boundary condition:

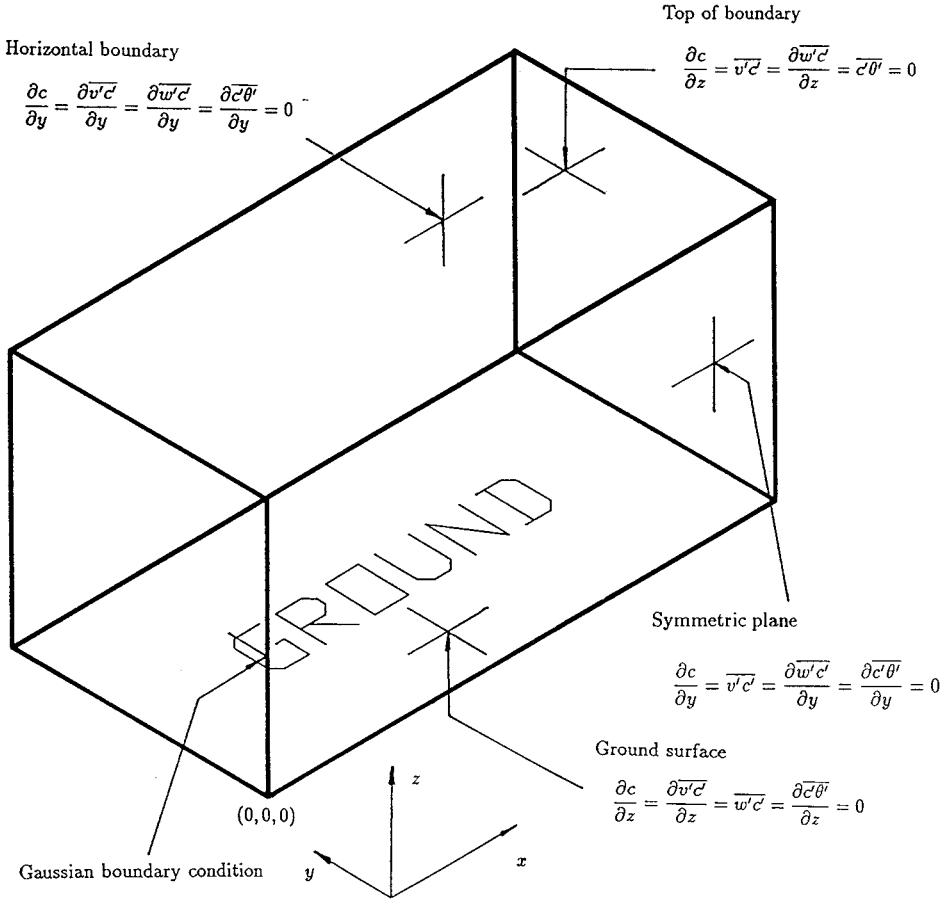

Figure 1. Computation domain and boundary conditions.

© 1998 John Wiley & Sons, Ltd.

$$c(\bar{x}, y, z) = \frac{Q}{2\pi\sigma_y\sigma_z U_s} \exp\left[-\frac{1}{2}\left(\frac{y}{\sigma_y}\right)^2\right]\left\{\exp\left[-\frac{1}{2}\left(\frac{z-z_s}{\sigma_z}\right)^2\right] + \exp\left[-\frac{1}{2}\left(\frac{z+z_s}{\sigma_z}\right)^2\right]\right\}, \quad (5)$$

where $\bar{x}$ is the downwind distance from the source at which the boundary conditions applied. Since σ_y and σ_z are functions of x and would be zero at $x = 0$, there is a singularity at $(0, 0, z_s)$, i.e. the point source centre where $c = \infty$. To avoid the singularity, $\bar{x}$ is assigned the value 0.05 m, such that the effects of second moment can be studied nearer to the source, while the computation accuracy will not be affected by the singularity near the pollutant point source. It should be noted that Equation (5) has accounted for the reflection of pollutant from the ground surface.

The horizontal and vertical dispersion coefficients $\sigma_y(\bar{x})$ and $\sigma_z(\bar{x})$ at $\bar{x}$ are obtained by the following expressions, suggested by Pai and Tsang [12]:

$$\sigma_y(\bar{x}) = \overline{v'^2}^{1/2} \frac{\bar{x}}{U_s}, \quad (6)$$

$$\sigma_z(\bar{x}) = \overline{w'^2}^{1/2} \frac{\bar{x}}{U_s}. \quad (7)$$

In the left-most boundary (i.e. Y–Z plane at $x = \bar{x}$), the boundary conditions for $\overline{v'c'}$ and $\overline{w'c'}$ are described by Equations (8) and (9), respectively,

$$\overline{v'c'} = -k\frac{\partial c}{\partial y}, \quad (8)$$

$$\overline{w'c'} = -k\frac{\partial c}{\partial z}, \quad (9)$$

where the eddy diffusivity k is given as:

$$k = 3 \times 0.1 \times \Lambda\sqrt{E}. \quad (10)$$

In calculating the turbulent length scale Λ, the following expression, suggested by Sun and Chang [13] was used:

$$\Lambda = 0.25\left\{1.8z_i\left[1 - \exp\left(\frac{-4z}{z_i}\right) - 0.0003\exp\left(\frac{8z}{z_i}\right)\right]\right\}. \quad (11)$$

Output from the Gaussian plume model is a solution to the transportation equation of mean scalar concentration c in Equation (1), by assuming the fluxes to be proportional to the gradient of mean scalar concentration c. As the effect of temperature fluctuation has not been considered in the Gaussian model, the covariance of concentration and temperature $\overline{c'\theta'}$ is set to zero at $x = \bar{x}$. In the right-most computation domain, natural boundary conditions were used.

At the top of the model, which is equal to the height of the convective boundary layer and at the surface boundary, the following boundary conditions suggested by Enger [14] were used:

At the top boundary

$$\frac{\partial c}{\partial z} = \frac{\partial\overline{w'c'}}{\partial z} = \overline{v'c'} = \overline{c'\theta'} = 0. \quad (12)$$

At the surface boundary

$$\frac{\partial c}{\partial z} = \overline{w'c'} = \frac{\partial\overline{v'c'}}{\partial z} = \frac{\partial\overline{c'\theta'}}{\partial z} = 0. \quad (13)$$

© 1998 John Wiley & Sons, Ltd.

Along the source centreline, the following boundary condition can be used due to symmetry:

$$\overline{v'c'} = \frac{\partial c}{\partial y} = \frac{\partial \overline{w'c'}}{\partial y} = \frac{\partial \overline{c'\theta'}}{\partial y} = 0. \tag{14}$$

In the horizontal direction, the width of the computation domain is sufficient and the following gradients can be set to zero:

$$\frac{\partial c}{\partial y} = \frac{\partial \overline{v'c'}}{\partial y} = \frac{\partial \overline{w'c'}}{\partial y} = \frac{\partial \overline{c'\theta'}}{\partial y} = 0. \tag{15}$$

The present model would be validated using the experimental observations of Willis and Deardorff [1] and Willis [2], therefore, the same computation sizes were used, i.e. 1.31, 0.6 and 0.3 m along the x-, y- and z-direction, respectively. After several tests against convergence, the number of grid points used is 31, 40 and 25 in the x-, y- and z-direction respectively, making a total of 31 000 points in the whole computation domain. The space of grid points along the three axes is distributed evenly.

4. NUMERICAL METHOD

The above set of equations was solved using a linear basis function of finite element scheme. Eight-node brick element and linear basis interpolation functions were used. The local element co-ordinates (ξ_i, η_i, ζ_i) of the eight-node points were assigned as:

$$(\xi_1, \eta_1, \zeta_1) = (-1, -1, -1)$$
$$(\xi_2, \eta_2, \zeta_2) = (1, -1, -1)$$
$$(\xi_3, \eta_3, \zeta_3) = (1, 1, -1)$$
$$(\xi_4, \eta_4, \zeta_4) = (-1, 1, -1)$$
$$(\xi_5, \eta_5, \zeta_5) = (-1, -1, 1)$$
$$(\xi_6, \eta_6, \zeta_6) = (1, -1, 1)$$
$$(\xi_7, \eta_7, \zeta_7) = (1, 1, 1)$$
$$(\xi_8, \eta_8, \zeta_8) = (-1, 1, 1)$$

It is assumed that the variable ϕ (i.e. c, $\overline{v'c'}$, $\overline{w'c'}$ and $\overline{c'\theta'}$) can be approximated by a linear basis function as follows:

$$\phi = \sum_{i=1}^{n} N_i \phi_i, \tag{16}$$

where $n = 8$ for an eight-node brick element. In matrix notation,

$$\phi = [N]\{\phi\}, \tag{17}$$

where

$$[N] = [N_1 \;\; N_2 \;\; N_3 \;\; N_4 \;\; N_5 \;\; N_6 \;\; N_7 \;\; N_8], \tag{18}$$

and

$$\{\phi\} = \{\phi_1 \;\; \phi_2 \;\; \phi_3 \;\; \phi_4 \;\; \phi_5 \;\; \phi_6 \;\; \phi_7 \;\; \phi_8\}^T. \tag{19}$$

The linear basis interpolation function N_i at each node point i is defined as:

© 1998 John Wiley & Sons, Ltd.

$$N_i = \frac{1}{8}(1+\xi_i\xi)(1+\eta_i\eta)(1+\zeta_i\zeta). \tag{20}$$

Galerkin finite element method was applied to Equations (1)–(4) with the Neumann boundary conditions (Equations (12)–(15)), to obtain the following spatial discretized finite element equations:

$$\int_v \left([N]^T[N]\left\{\frac{\partial c}{\partial t}\right\}\right)\mathrm{d}v + \int_v \left([N]^T U\frac{\partial [N]}{\partial x}\{c\} + [N]^T\frac{\partial [N]}{\partial y}\{\overline{v'c'}\} + [N]^T\frac{\partial [N]}{\partial z}\{\overline{w'c'}\}\right)\mathrm{d}v = 0, \tag{21}$$

$$\int_v \left([N]^T[N]\left\{\frac{\partial \overline{v'c'}}{\partial t}\right\}\right)\mathrm{d}v + \int_v \left([N]^T U\frac{\partial [N]}{\partial x}\{\overline{v'c'}\} + [N]^T\overline{v'^2}\frac{\partial [N]}{\partial y}\{c\} + \frac{\partial [N]^T}{\partial y}aq\Lambda\frac{\partial [N]}{\partial y}\{\overline{v'c}\} + \frac{\partial [N]^T}{\partial z}aq\Lambda\frac{\partial [N]}{\partial z}\{\overline{v'c'}\} + [N]^T\frac{Aq}{\Lambda}[N]\{\overline{v'c'}\}\right)\mathrm{d}v = 0, \tag{22}$$

$$\int_v \left([N]^T[N]\left\{\frac{\partial \overline{w'c'}}{\partial t}\right\}\right)\mathrm{d}v + \int_v \left([N]^T U\frac{\partial [N]}{\partial x}\{\overline{w'c'}\} + [N]^T\overline{w'^2}\frac{\partial [N]}{\partial z}\{c\} - [N]^T\frac{g}{T_o}[N]\{\overline{c'\theta'}\} + \frac{\partial [N]^T}{\partial y}aq\Lambda\frac{\partial [N]}{\partial y}\{\overline{w'c'}\} + \frac{\partial [N]^T}{\partial z}aq\Lambda\frac{\partial [N]}{\partial z}\{\overline{w'c'}\} + [N]^T\frac{Aq}{\Lambda}[N]\{\overline{w'c'}\}\right)\mathrm{d}v = 0, \tag{23}$$

$$\int_v \left([N]^T[N]\left\{\frac{\partial \overline{c'\theta'}}{\partial t}\right\}\right)\mathrm{d}v + \int_v \left([N]^T U\frac{\partial [N]}{\partial x}\{\overline{c'\theta'}\} + [N]^T\frac{\partial T}{\partial z}[N]\{\overline{w'c'}\} + [N]^T\overline{w'\theta'}\frac{\partial [N]}{\partial z}\{c\} + \frac{\partial [N]^T}{\partial z}aq\Lambda\frac{\partial [N]}{\partial z}\{\overline{c'\theta'}\} + [N]^T 2bs\frac{q}{\Lambda}[N]\{\overline{c'\theta'}\}\right)\mathrm{d}v = 0. \tag{24}$$

The above four weak Galerkin forms in Equations (21)–(24) lead to the following standard semi-discretized finite element equations:

$$[K]\{\phi\} + [M]\left\{\frac{\partial \phi}{\partial t}\right\} = \{0\}, \tag{25}$$

where

$$[M] = \begin{bmatrix} M_{11} & & & \\ & M_{22} & & \\ & & M_{33} & \\ & & & M_{44} \end{bmatrix}, \tag{26}$$

$$[K] = \begin{bmatrix} K_{11} & K_{12} & K_{13} & \\ K_{21} & K_{22} & & \\ K_{31} & & K_{33} & K_{34} \\ K_{41} & & K_{43} & K_{44} \end{bmatrix}, \tag{27}$$

© 1998 John Wiley & Sons, Ltd.

$$\{\phi\} = \begin{Bmatrix} c \\ \overline{v'c'} \\ \overline{w'c'} \\ \overline{c'\theta'} \end{Bmatrix}, \tag{28}$$

and

$$[M_{11}] = [M_{22}] = [M_{33}] = [M_{44}] = \int_v [N]^T[N]\,\mathrm{d}v, \tag{29}$$

$$[K_{11}] = \int_v \left(U[N]^T \frac{\partial [N]}{\partial x} \right) \mathrm{d}v, \tag{30}$$

$$[K_{12}] = \int_v \left([N]^T \frac{\partial [N]}{\partial y} \right) \mathrm{d}v, \tag{31}$$

$$[K_{13}] = \int_v \left([N]^T \frac{\partial [N]}{\partial z} \right) \mathrm{d}v, \tag{32}$$

$$[K_{21}] = \int_v \left(\overline{v'^2} [N]^T \frac{\partial [N]}{\partial y} \right) \mathrm{d}v, \tag{33}$$

$$[K_{22}] = \int_v \left(U[N]^T \frac{\partial [N]}{\partial x} + aq\Lambda \frac{\partial [N]^T}{\partial y} \frac{\partial [N]}{\partial y} + aq\Lambda \frac{\partial [N]^T}{\partial z} \frac{\partial [N]}{\partial z} + \frac{Aq}{\Lambda} [N]^T[N] \right) \mathrm{d}v, \tag{34}$$

$$[K_{31}] = \int_v \left(\overline{w'^2} [N]^T \frac{\partial [N]}{\partial z} \right) \mathrm{d}v, \tag{35}$$

$$[K_{33}] = \int_v \left(U[N] \frac{\partial [N]}{\partial x} + aq\Lambda \frac{\partial [N]^T}{\partial y} \frac{\partial [N]}{\partial y} + aq\Lambda \frac{\partial [N]^T}{\partial z} \frac{\partial [N]}{\partial z} + \frac{Aq}{\Lambda} [N]^T[N] \right) \mathrm{d}v, \tag{36}$$

$$[K_{34}] = \int_v \left(-\frac{g}{T_o} [N]^T[N] \right) \mathrm{d}v, \tag{37}$$

$$[K_{41}] = \int_v \left(\overline{w'\theta'} [N]^T \frac{\partial [N]}{\partial z} \right) \mathrm{d}v, \tag{38}$$

$$[K_{43}] = \int_v \left(\frac{\partial T}{\partial z} [N]^T[N] \right) \mathrm{d}v, \tag{39}$$

$$[K_{44}] = \int_v \left(U[N]^T \frac{\partial [N]}{\partial x} + aq\Lambda \frac{\partial [N]^T}{\partial z} \frac{\partial [N]}{\partial z} + 2bs \frac{q}{\Lambda} [N]^T[N] \right) \mathrm{d}v. \tag{40}$$

To facilitate the numerical integration, the isoparametric finite element method is used in the calculation of stiffness and mass matrices. Assuming that the global co-ordinates (x, y, z) can be expressed by the interpolation function N_i and the global node point co-ordinates (x_i, y_i, z_i) as:

$$x = [N]\{x\}, \tag{41}$$

$$y = [N]\{y\}, \tag{42}$$

$$z = [N]\{z\}, \tag{43}$$

where

$$\{x\} = \{x_1 \;\; x_2 \;\; x_3 \;\; x_4 \;\; x_5 \;\; x_6 \;\; x_7 \;\; x_8\}, \tag{44}$$

© 1998 John Wiley & Sons, Ltd.

$$\{y\} = \{y_1 \;\; y_2 \;\; y_3 \;\; y_4 \;\; y_5 \;\; y_6 \;\; y_7 \;\; y_8\}, \tag{45}$$

$$\{z\} = \{z_1 \;\; z_2 \;\; z_3 \;\; z_4 \;\; z_5 \;\; z_6 \;\; z_7 \;\; z_8\}. \tag{46}$$

By differentiating Equation (20) with respect to ξ, η and ζ, respectively, the following system of equations expressed in matrix form is obtained:

$$\begin{Bmatrix} \dfrac{\partial N_i}{\partial \xi} \\ \dfrac{\partial N_i}{\partial \eta} \\ \dfrac{\partial N_i}{\partial \zeta} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} & \dfrac{\partial z}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} & \dfrac{\partial z}{\partial \eta} \\ \dfrac{\partial x}{\partial \zeta} & \dfrac{\partial y}{\partial \zeta} & \dfrac{\partial z}{\partial \zeta} \end{bmatrix} \begin{Bmatrix} \dfrac{\partial N_i}{\partial x} \\ \dfrac{\partial N_i}{\partial y} \\ \dfrac{\partial N_i}{\partial z} \end{Bmatrix}. \tag{47}$$

The following matrix is obtained by substituting Equations (41)–(43) into (47):

$$\begin{Bmatrix} \dfrac{\partial N_i}{\partial \xi} \\ \dfrac{\partial N_i}{\partial \eta} \\ \dfrac{\partial N_i}{\partial \zeta} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial [N]}{\partial \xi}\{x\} & \dfrac{\partial [N]}{\partial \xi}\{y\} & \dfrac{\partial [N]}{\partial \xi}\{z\} \\ \dfrac{\partial [N]}{\partial \eta}\{x\} & \dfrac{\partial [N]}{\partial \eta}\{y\} & \dfrac{\partial [N]}{\partial \eta}\{z\} \\ \dfrac{\partial [N]}{\partial \zeta}\{x\} & \dfrac{\partial [N]}{\partial \zeta}\{y\} & \dfrac{\partial [N]}{\partial \zeta}\{z\} \end{bmatrix} \begin{Bmatrix} \dfrac{\partial N_i}{\partial x} \\ \dfrac{\partial N_i}{\partial y} \\ \dfrac{\partial N_i}{\partial z} \end{Bmatrix}. \tag{48}$$

Let the 3×3 matrix in Equation (48) be J, then the derivative of N_i with respect to x, y and z can be expressed as follows:

$$\begin{Bmatrix} \dfrac{\partial N_i}{\partial x} \\ \dfrac{\partial N_i}{\partial y} \\ \dfrac{\partial N_i}{\partial z} \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} \dfrac{\partial N_i}{\partial \xi} \\ \dfrac{\partial N_i}{\partial \eta} \\ \dfrac{\partial N_i}{\partial \zeta} \end{Bmatrix}. \tag{49}$$

The derivative of N_i with respect to (x, y, z) is then expressed in terms of local co-ordinates (ξ, η, ζ). This is done to simplify the calculation of stiffness and mass matrices from analytical calculation to numerical integration, which is then carried out by $3 \times 3 \times 3$ Gaussian Quadrature approximation [15].

For greater numerical stability in temporal discretization, the Crank–Nicolson scheme [16] is used to integrate the semi-discretized Equation (25) in time:

$$[K]\left\{\frac{\phi_i^{n+1} + \phi_i^n}{2}\right\} + [M]\left\{\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t}\right\} = 0. \tag{50}$$

Equation (50) is then rearranged to obtain the following equation:

$$\left[\frac{M}{\Delta t} + \frac{K}{2}\right]\{\phi_i^{n+1}\} = \left[\frac{M}{\Delta t} - \frac{K}{2}\right]\{\phi_i^n\}. \tag{51}$$

After applying the initial and boundary conditions, the system of equations is then solved by successive overrelaxation method [17], i.e.

© 1998 John Wiley & Sons, Ltd.

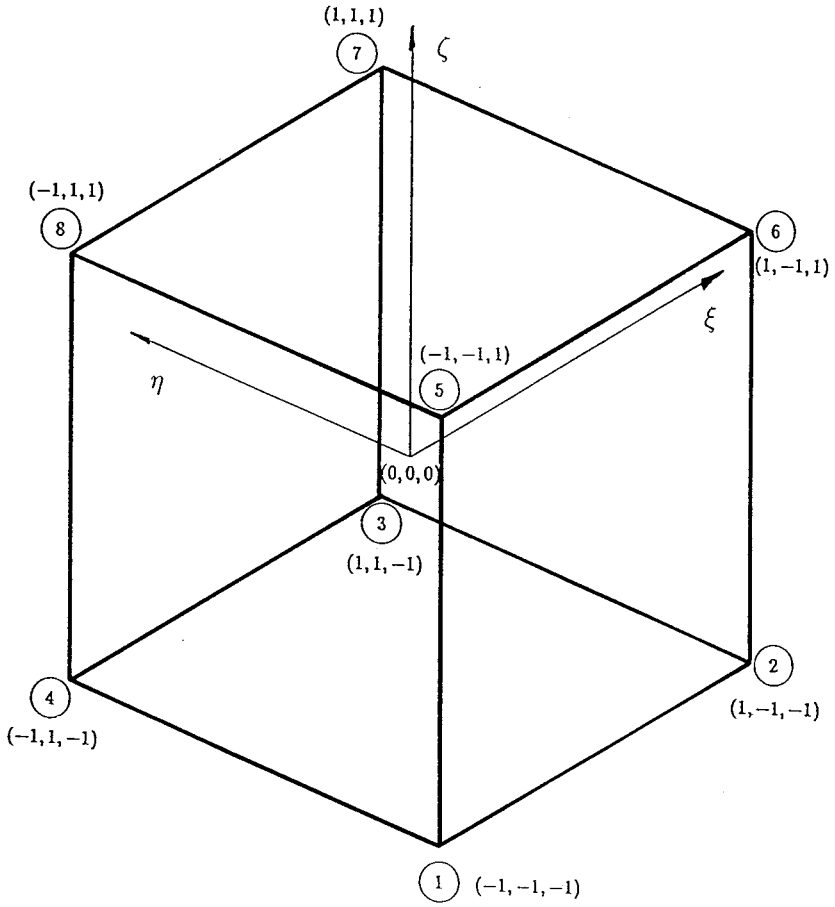

Figure 2. Local co-ordinates of an element.

$$\phi_i^k = \omega\phi_i^k + (1-\omega)\phi_i^{k-1}, \tag{52}$$

until the residue of Equation (51) is sufficiently small. Then, ϕ_i^n, the node value of node point i at time n can be obtained.

5. SPARSE MATRIX STORAGE

With the use of the finite element method to solve the system of partial differential equations discussed above, a large sparse system of linear algebraic equations (i.e. $[M]$ and $[K]$) will be obtained. A sparse matrix algorithm is an algorithm that performs a matrix computation in such a way as to take advantage of the zero/non-zero structure of the matrices involved [18]. With the advantages of sparsity—not explicitly storing or manipulating some or all of the zero elements—the size of problem and the efficiency of computation can be greatly increased.

For storage of the sparse matrix used in the present numerical model, compressed row storage method [19], the most general method to store sparse matrix, was adopted. It makes no assumption about the sparsity structure of the matrix, but puts the subsequent non-zeros of the matrix row in contiguous memory location.

© 1998 John Wiley & Sons, Ltd.

With the use of compressed row storage method, three vectors (one-dimensional array) were used for the storage of an $n \times n$ sparse matrix A. The first vector is a number of non-zero (nnz) entities of the matrix A. The memory ratio of two more integer vectors, one for row indices and one for the storage of the first element in each column, is as follows:

$$\frac{\text{compressed row storage method}}{\text{original sparse matrix in full form}} = \frac{nnz(\text{float/double}) + nnz(\text{integer}) + n(\text{integer})}{n \times n(\text{float/double})}. \quad (53)$$

The structure of the sparse matrix was determined by the connectivity of the node points in the computation domain as shown in Figures 2 and 3. If point i is connected to point j, the entity a_{ij} in the full matrix is treated as non-zero and put into the contiguous memory. The connectivity of the node points can be determined by an element and global node number table [20]. This scheme is simple and can predict the matrix structure from the non-zero structure of the problem instead of non-zero structure of the matrix. In other words, the structure of the sparse matrix was determined before constructing the global mass and stiffness matrices. So the memory allocation of the sparse matrix is in the form of compressed row storage instead of original sparse matrix in full form. The large memory requirement for original sparse matrix in full form can thus be avoided and larger scale computation can be achieved. By iterating the roots using the successive overrelaxation method, the calculation was also accelerated, since only non-zero elements of the sparse matrix were considered.

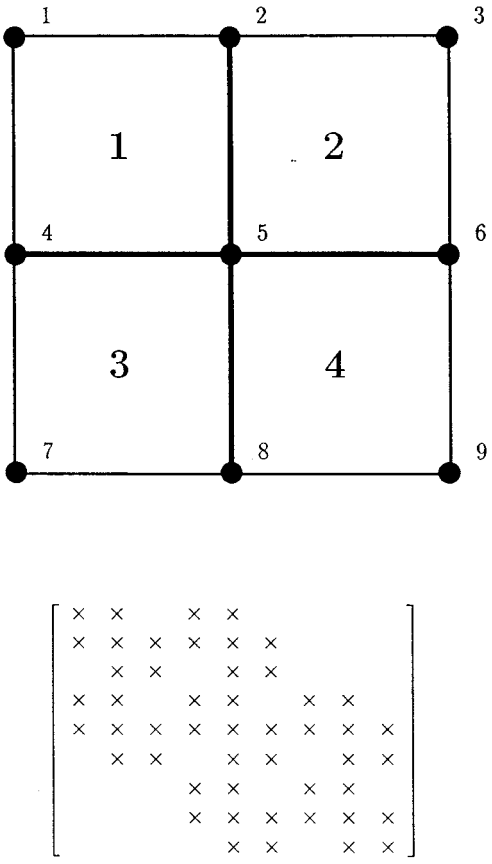

Figure 3. Sparse matrix structure determination by connectivity.

© 1998 John Wiley & Sons, Ltd.

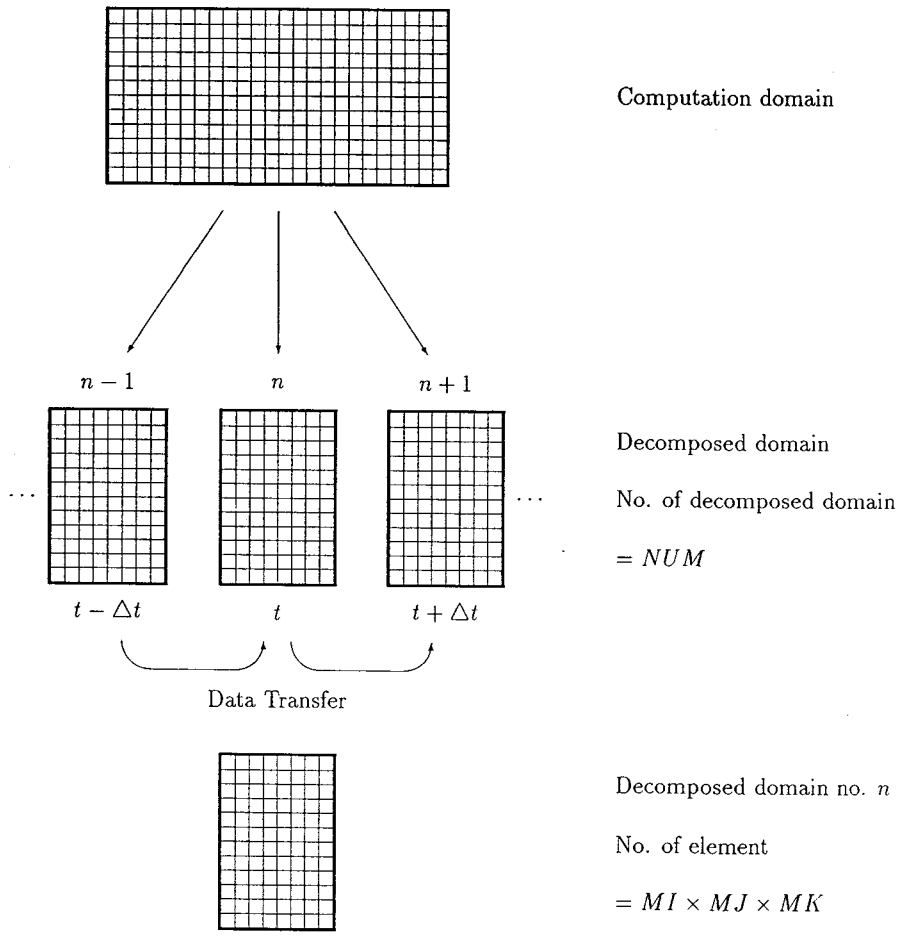

Figure 4. Decomposed domain.

6. SOLUTION ALGORITHM

The domain decomposition method, suggested by Akay and Beskik [21], was adopted for the present numerical model. The case with non-overlapping interfaces as shown in Figure 4 is studied. The original computation domain is divided into a number of subdomains or solution blocks along the x-axis. Each subdomain is then further divided into $MI \times MJ \times MK$ number of element along the x-, y- and z-directions, respectively and then solved using the finite element method discussed previously. After completion of the iteration of roots at each time step, a subdomain uses the solution of adjacent subdomain as boundary condition for the next iteration. Since the boundary condition in the x-direction remains natural in the right-most side (streamwise direction), the direction of solution passes from left to right along the x-axis.

The present calculation was carried out using an IBM 9076 Scalable POWERparallel System SP2. Sixteen nodes (processors), numbered from 0 to 15, were available for the present calculation. In the present arrangement one of the nodes is assigned as master (node no. 0), responsible for input and allocation of relevant data between other nodes. The remaining nodes (node nos. 1–15) acted as working nodes and are responsible for the calculation.

© 1998 John Wiley & Sons, Ltd.

The whole computation domain was divided into 15 subdomains, each working node is responsible for the calculation of one subdomain. After the construction of mass and stiffness matrices (i.e. spatial discretization), transient calculation starts (i.e. temporal discretization). The master node transfers the left-most boundary condition to node no. 1 at the start of time step t, then node no. 1 carries out the calculation of its subdomain for time t.

After node no. 1 completes its calculation at time step t, its right-most results (c, $v'c'$, $w'c'$, $c'\theta'$) will be sent to its adjacent subdomain in the right-hand side, i.e. node no. 2 (subdomain no. 2), thus, node no. 2 obtains its left-most boundary condition at time step t. Node no. 2 then carries out its calculation at time step t. When node no. 1 finishes the transfer of the right-most results to node no. 2, it receives left-most boundary conditions from master and then carries out its calculation at time step at $t+\Delta t$. Other nodes also carry out the calculation and data transfer similar to node nos. 1 and 2, as summarized in Figure 4.

The above parallelism can be explored using a parallel computer. Once the boundary conditions from adjacent subdomain are passed, each subdomain system can be solved separately without communication until the next time step. This method reduces the portion of computation time in message passing or information transfer between nodes. Implementation of the above domain decomposition method on a parallel computer could lead to a considerable gain in time. Furthermore, the size of problems can be increased as each processor only stores the unknowns of its awarded subdomain. The subdomain is chosen to equalize the computation load and computer resources.

7. RESULTS AND DISCUSSION

Results of the present numerical model are compared with the experimental observations of Willis and Deardorff [1] and Willis [2] for the release of pollutant at mid-level and ground level. These experiments were conducted in an unstably stratified convective water channel, which was used to study the effect of pollutant dispersion in a convective atmospheric boundary layer. The numerical and experimental results would also be compared with the result of another numerical model by Sykes *et al.* [11] which obtained the pollutant concentration by first determining the dispersion coefficients and assuming Gaussian distribution for the pollutant concentration. No such assumption was made in the present study for the pollutant concentration and the mathematical model described above was solved directly. Thus, more realistic non-Gaussian plume behaviours can be studied, particularly the dispersion in the vertical direction.

The plume centroid, $(\bar{y}, \bar{z})$, can be determined by

$$\bar{y} = \frac{\langle yc \rangle}{\langle c \rangle} \tag{54}$$

$$\bar{z} = \frac{\langle zc \rangle}{\langle c \rangle} \tag{55}$$

where the angular brackets represent area integrals in the cross-stream plane, i.e.

$$\langle \psi \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi \, \mathrm{d}y \, \mathrm{d}z, \tag{56}$$

and the dispersion coefficients σ_y and σ_z can be obtained by

$$\sigma_y^2 = \frac{\langle (y-\bar{y})^2 c \rangle}{\langle c \rangle}, \tag{57}$$

© 1998 John Wiley & Sons, Ltd.

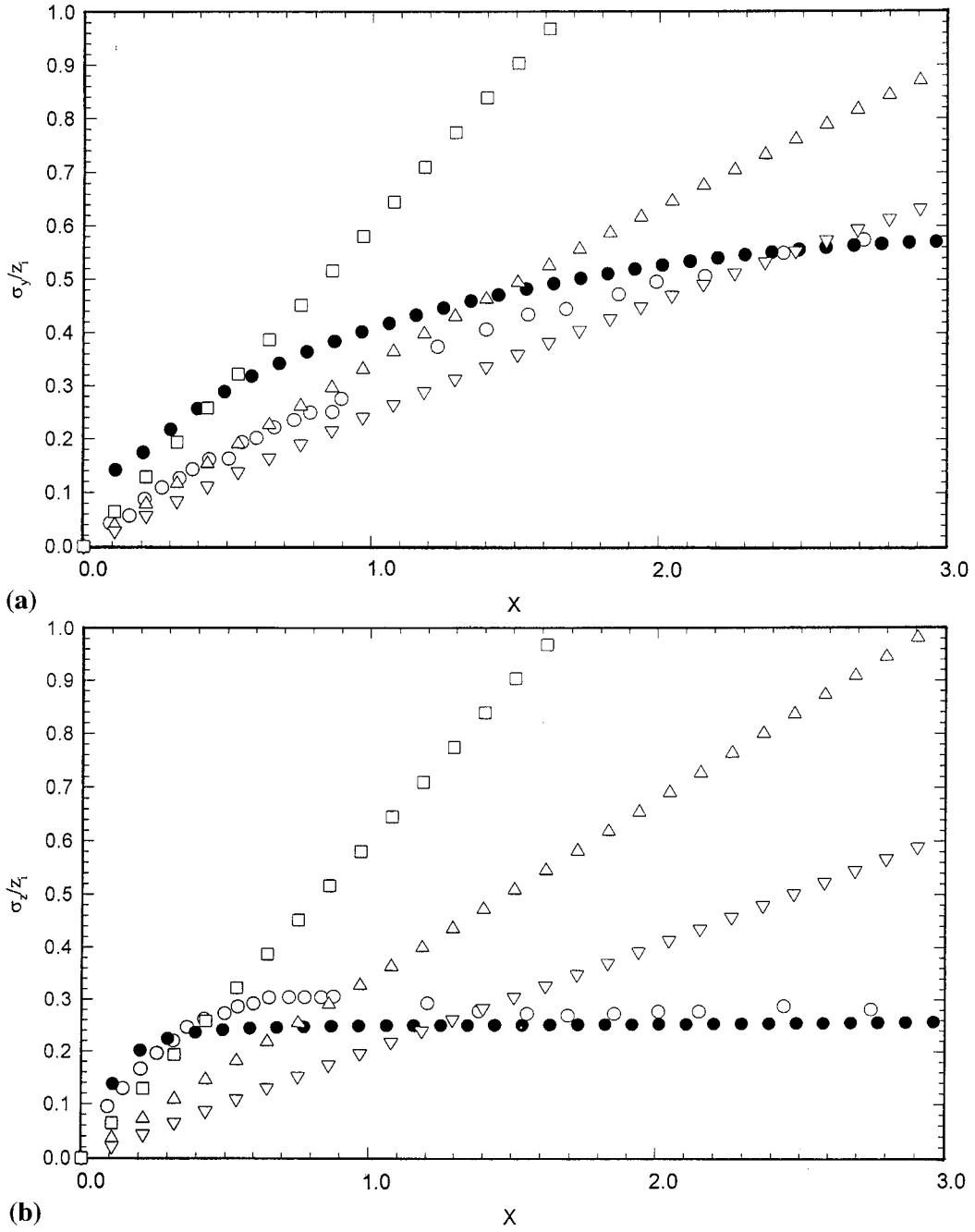

Figure 5. Dispersion coefficients for mid-level pollutant release. (a) σ_y; (b) σ_z. ● Present result; ○ experimental results of Willis and Deardorff [1] and Willis [2]; △ Briggs [22] Pasquill–Gifford stability class A; ▽ Briggs [22] Pasquill–Gifford stability class B; □ Hanna [23]. Dispersion coefficients for ground-level pollutant release. (c) σ_y; (d) σ_z. ● Present result; ○ experimental results of Willis and Deardorff [1] and Willis [2]; △ Briggs [22] Pasquill–Gifford stability class A; ▽ Briggs [22] Pasquill–Gifford stability class B; □ Hanna [23].

© 1998 John Wiley & Sons, Ltd.

$$\sigma_z^2 = \frac{\langle (z-\bar{z})^2 c \rangle}{\langle c \rangle}. \tag{58}$$

The model predictions of σ_y and σ_z as functions of downstream distance X from the source are compared with the experimental observation in Figure 5(a) and Figure 5(b) (for the cases

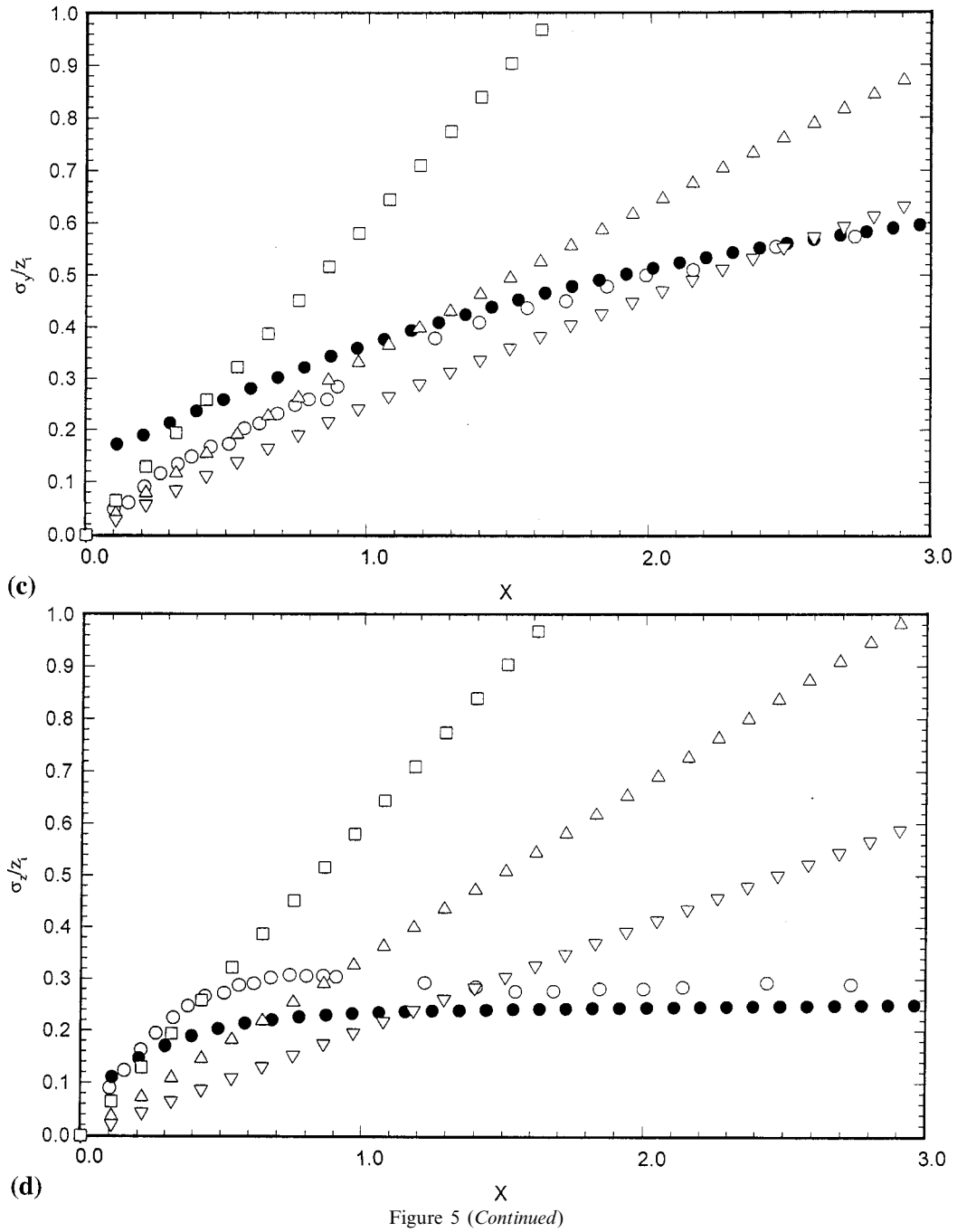

Figure 5 (*Continued*)

© 1998 John Wiley & Sons, Ltd.

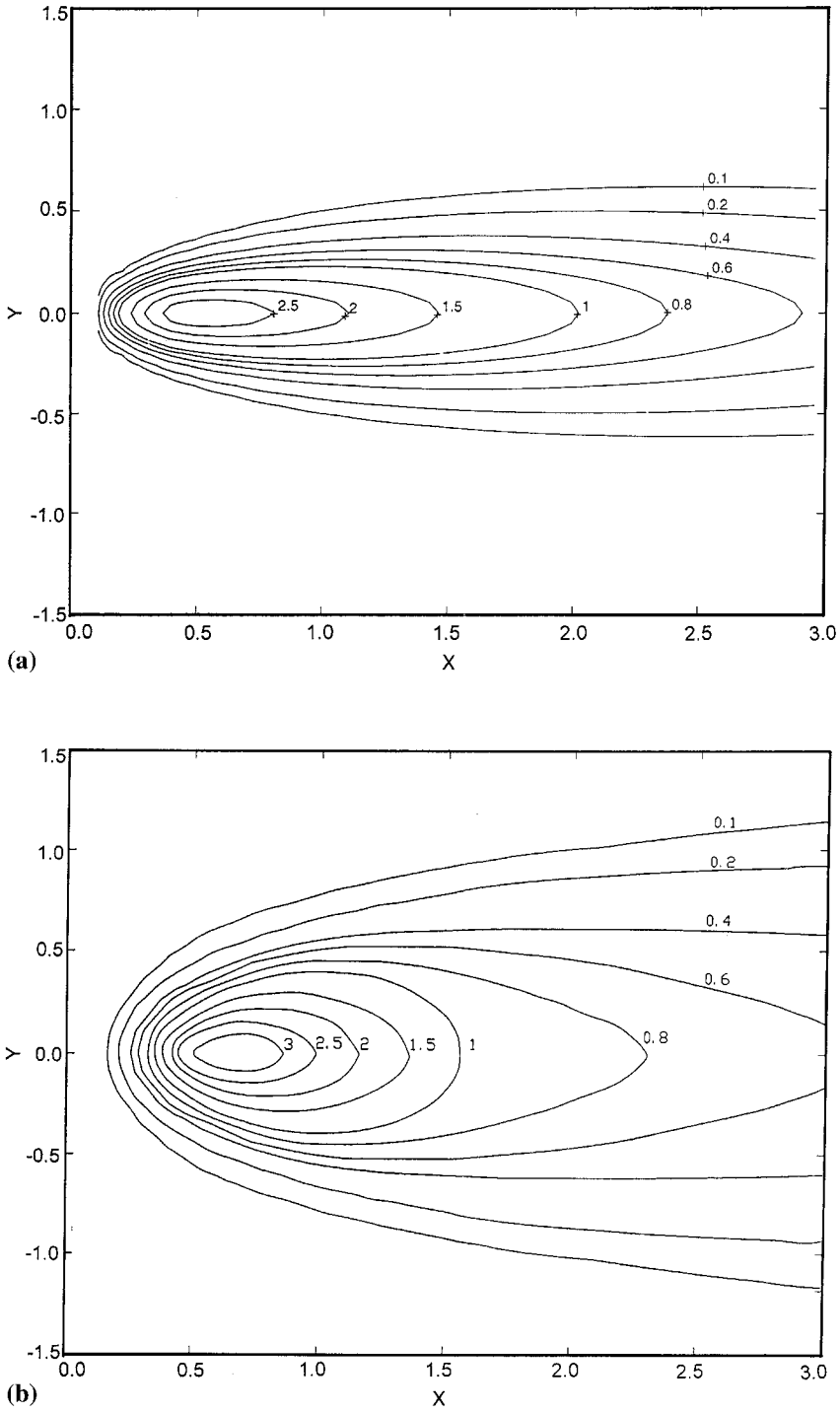

Figure 6. Ground surface non-dimensional pollutant contour plot. (a) present result; (b) Willis and Deardorff [2]; (c) Sykes *et al.* [11].

© 1998 John Wiley & Sons, Ltd.

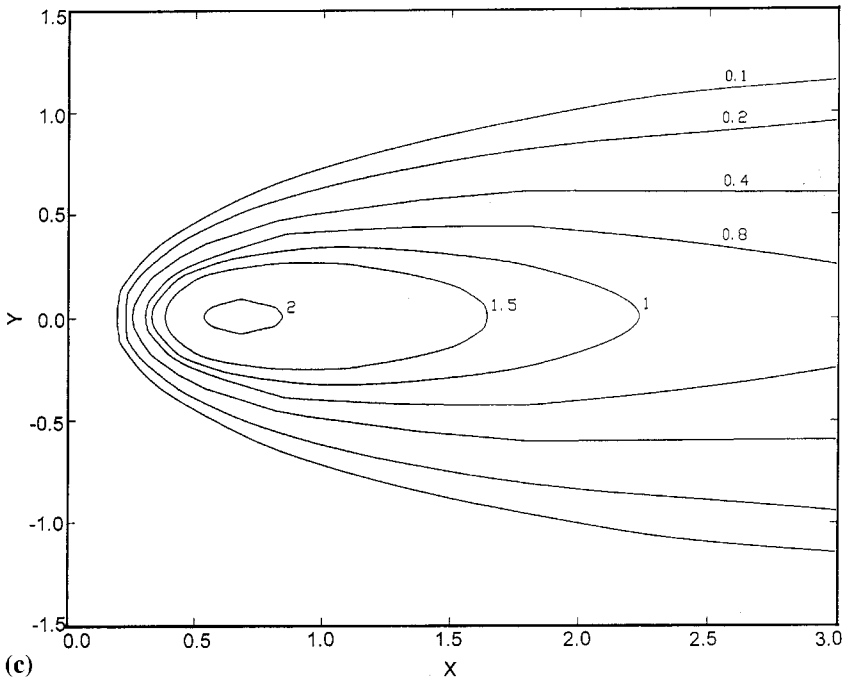

Figure 6 (*Continued*)

of mid-level release i.e. $z = 0.5z_i$) and in Figure 5(c) and Figure 5(d) (for the cases of ground level release i.e. $z = 0.067z_i$). Dispersion coefficients suggested by Briggs [22] and Hanna [23] were also shown in the figure for comparison.

For the comparison of horizontal dispersion coefficient σ_y, the agreement between experimental observation and numerical model is quite good, except for the early part of the flow ($X < 1$), where the numerical model overpredicted the values. This is mainly due to the uncertainty of the horizontal crosswind velocity variance $\overline{v'^2}$. Willis and Deardorff [24] only presented the averaged value of horizontal velocity variance, i.e. $(\overline{u'^2} + \overline{v'^2})/2$, which is used as the horizontal crosswind velocity variance $\overline{v'^2}$ in the present calculation. Hanna [23] overpredicted σ_y values, except those very close to the pollutant source. A better agreement with experimental values for Pasquill–Gifford (P–G) stability classes A and B [22] can be observed. The P–G stability classes A and B represent unstable atmospheric conditions and are therefore selected for the present comparison. It can be observed that Briggs [22] overpredicted σ_y for greater X ($\approx 30\%$ for stability class A at $X = 2.7$). The difference increases with increasing streamwise distance X for stability class A. As a whole, the present numerical model retains close approximation to the experimental observation throughout the measurement range.

For the comparison of vertical dispersion coefficient σ_z, it can be observed that the present model slightly underpredicted σ_z but the general profile agreed well with experimental observation throughout the measurement range. The σ_z values obtained by Hanna [23] are comparable with experimental observation for $X < 0.6$. Briggs [22] underpredicted the experimental σ_z values for $X < 1$ and overpredicted the values for other streamwise distances.

© 1998 John Wiley & Sons, Ltd.

Figure 6(a) shows the contour plot of ground level pollutant concentration C for the release of passive material in the middle of the mixing layer. The result is compared with the experimental observation of Willis [2] (Figure 6(b)), and the numerical results of Sykes *et al.* [11] (Figure 6(c)). It can be observed that both numerical models underpredicted the maximum surface impact ($\approx 17\%$ for the present numerical model and $\approx 33\%$ for Sykes *et al.* [11]). In addition, the non-Gaussian descent of the plume cannot be simulated in Sykes *et al.* [11], because the model is simplified by assuming Gaussian shape for pollutant concentration, resulting in underprediction of the ground level impact.

In the present numerical model the agreement between ground level impact and experimental observation is better than that of Skyes *et al.* [11], indicating that the present model can simulate more closely the descent of the plume to the surface. However, as the effect of temperature and stability have not been considered in adopting the boundary conditions, the descent of the plume predicted by the present model is smaller than that of the experimental observation, causing the underprediction. Although this can be improved by reducing the distance between the source and the left-most computation domain, this will also greatly increase the computation load and is therefore not favorable.

Figure 7 shows the concentration contour of X–Z plane along $Y = 0$. The descent of the plume trajectory and the impact on the ground which is discussed above can be observed. The plume reached the ground at $X \approx 0.5$, resulting in the maximum ground concentration observed in the ground level pollutant contour of Figure 6(a). It can also be observed that C increases for $X > 2$ and $Y > 0.8$, showing that the pollutant rises up into the mixed layer. This agrees with the plume behaviour described in Willis [2].

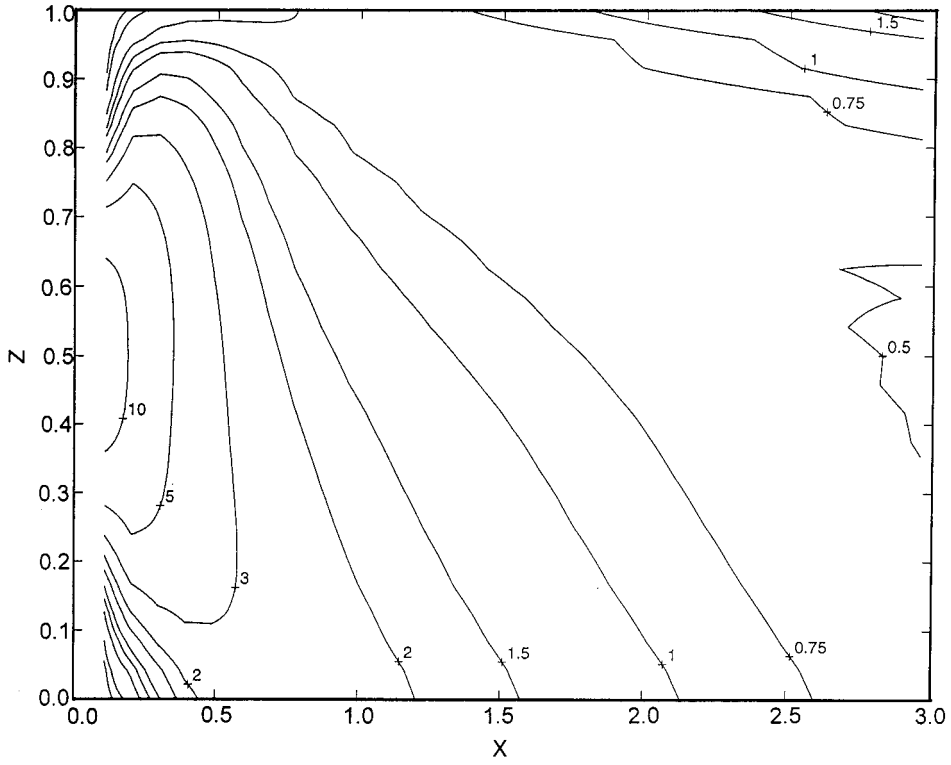

Figure 7. Non-dimensional pollutant contour plot of the present model; X–Z plane at $Y = 0$.

© 1998 John Wiley & Sons, Ltd.

8. CONCLUSION

It is demonstrated that the present second-order closure model can be used to simulate the transportation of plume under a convective boundary layer. The results of the present model compared well with recent experimental observations. Several non-Gaussian plume behaviours, such as the descent and rise of plume trajectory can be modelled by the present model. The second-order closure model involves four simultaneous partial differential equations in the three-dimensional calculation, which normally requires long computation time and large memory in traditional serial code. Parallelization of the present model accelerates the computation time and greatly increases the size of memory. Thus, more sophisticated dispersion modelling can be carried out by the second-order closure dispersion model using parallel computation.

ACKNOWLEDGMENTS

The authors wish to acknowledge the Hong Kong Research Grant Council for supporting this project.

APPENDIX A. NOMENCLATURE

a	turbulent closure constant = 0.3 (from Reference [12])
A	turbulent closure constant = 0.75 (from Reference [12])
b	turbulent closure constant = 0.125 (from Reference [12])
c	pollutant concentration
C	dimensionless concentration = $(U_m z_i^2 c)/Q$
$\overline{c'\theta'}$	covariance of concentration and temperature
E	turbulent kinetic energy
g	gravity acceleration
k	eddy diffusivities
$[K] = K_{ij}$	stiffness matrix
$[J]$	Jacobian matrix
$[M] = M_{ij}$	mass matrix
n	number of node points in a element, number of rows or column of a matrix
nnz	number of non-zero elements
$[N] = N_i$	interpolation function
q	turbulent velocity
Q	pollutant source emission rate
s	turbulent closure constant = 1.8 (from Reference [12])
t	time
Δt	time interval
T_o	potential temperature
U	mean wind speed along streamwise direction
U_m	mean wind speed in convective boundary layer
U_s	wind speed at source height
$\overline{v'^c}$, $\overline{w'^2}$	crosswind speed variances in lateral and vertical direction, respectively

© 1998 John Wiley & Sons, Ltd.

$\overline{v'c'}$, $\overline{w'c'}$	concentration fluxes
$\overline{w'\theta'}$	vertical heat flux
w_*	convective velocity scale
$\bar{x}$	streamwise distance from the point source where boundary condition apply
$\{x\} = x_i$	x co-ordinate of global node point in an element
X	dimensionless streamwise distance $= (w_* x)/(z_i U_m)$
x, y, z	Cartesian co-ordinate in streamwise, lateral and vertical direction, respectively
$\{y\} = y_i$	y co-ordinate of global node point in an element
Y	dimensionless lateral distance $= y/z_i$
$(\bar{y}, \bar{z})$	plume centroid along streamwise direction
$\{z\} = z_i$	z co-ordinate of global node point in an element
z_i	convective boundary layer thickness
z_s	pollutant emission height
Z	dimensionless vertical height $= z/z_i$
$[\phi] = \phi_i$	node value
ϕ_i^n	node value of node point i at time n
Λ	turbulent length scale
σ_y, σ_z	pollutant dispersion coefficients in lateral and vertical direction, respectively
ω	successive overrelaxation factor
(ξ_i, η_i, ζ_i)	local co-ordinate of node point i in an element

REFERENCES

1. G.E. Willis and J.W. Deardorff, 'A laboratory model of diffusion into convective planetary boundary layer', *Quart. J. R. Met. Soc.*, **102**, 427–445 (1976).
2. G.E. Willis, 'Laboratory modelling of dispersion in the convectively mixed layer', *Fifth Symp. On Turbulence Diffusion and Air Pollution, Amer. Meteor. Soc.*, pp. 155–156, 1980.
3. W.S. Lewellen, 'Use of invariant modelling', in W. Frost and T.H. Moulden (eds.), *Handbook of Turbulence*, Plenum Press, New York, 1977.
4. F. Pasquill and F.B. Smith, *Atmospheric Diffusion*, 2nd edn., Halstead Press, Wiley, New York, 1974.
5. R.I. Sykes, W.S. Lewellen and S.F. Parker, 'A turbulent-transport model for concentration fluctuations and flux', *J. Fluid Mech.*, **139**, 193–218 (1984).
6. R.G. Lamb, 'A numerical simulation of dispersion from an elevated point source in the convective boundary layer', *Atmos. Environ.*, **12**, 1297–1304 (1978).
7. J.W. Deardorff and G.E. Willis, 'A parameterization of diffusion into the mixed layer', *J. Appl. Meteor*, **14**, 1451–1458 (1975).
8. G.E. Willis and J.W. Deardorff, 'A laboratory study of dispersion from an elevated source within a modeled convective planetary layer', *Atmos. Environ.*, **12**, 1305–1311 (1978).
9. G.E. Willis and J.W. Deardorff, 'A laboratory study of dispersion from a source in the middle of the convective mixed layer', *Atmos. Environ.*, **15**, 109–117 (1981).
10. C.H. Liu and D.Y. C. Leung, 'Numerical study of atmospheric dispersion under unstably stratified atmosphere', to appear in *J. Wind Engg. Ind. Aero.*
11. R.I. Sykes, W.S. Lewellen and S.F. Parker, 'A Gaussian plume model of atmospheric dispersion based on second-order closure', *J. Climate Appl. Meteorol.*, **25**, 322–331 (1986).
12. P. Pai and T.H. Tsang, 'Finite element solution to turbulent diffusion in a convective boundary layer', *Int. J. Numer. Method Fluids*, **12**, 179–195 (1991).
13. W.Y. Sun and C.Z. Chang, 'Diffusion model for a convective layer. Part I: Numerical simulation of convective boundary layer', *J. Climate Appl. Meteorol.*, **25**, 1445–1453 (1986).
14. L. Enger, 'Simulation of dispersion in moderately complex terrain—Part b. The higher order closure dispersion model', *Atmos. Environ.*, **24A**, 2447–2455 (1990).
15. A.H. Stroud, *Gaussian Quadrature Formulas*, Prentice-Hall, Englewood Cliffs, NJ, 1966.
16. C.A.J. Fletcher, *Computational Techniques for Fluid Dynamics*, Springer, Berlin, 1991.
17. K.E. Atkinson, *An Introduction to Numerical Analysis*, 2nd edn., Wiley, New York, 1989.
18. J.R. Gilbert, 'Predicting structure in sparse matrix computation', *SIAM J. Matrix Anal. Appl.*, **15**, 62–79 (1994).
19. R. Barrett, H. Berry, T.F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine and H. Van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, 2nd edn., SIAM, Philadelphia, PA, 1994.

© 1998 John Wiley & Sons, Ltd.

20. T.J. Chung, *Finite Element Analysis in Fluid Dynamics*, McGraw-Hill, New York, 1978.
21. H.U. Akay and A. Beskik, 'A parallel algorithm for compressible flows through rotor-stator combinations', in D.S. Horst (ed.), *Parallel Computational Fluid Dynamics*, MIT Press, Cambridge, MA, 1992.
22. G.A. Briggs, 'Diffusion estimation for small emissions', in *Environmental Research Laboratories*, *Air Resources Atmosphere Turbulence and Diffusion Laboratory*, 1973 Annual Report of the USAEC ATDL-106, National oceanic and atmospheric administration, 1973.
23. S.R. Hanna, 'Applications in air pollution modeling', in F.T.M. Nieuwstadt and H. Van Dop (eds.), *Atmospheric Turbulence and Air Pollution Modelling*, Reidel, Dordrecht, 1981.
24. G.E. Willis and J.W. Deardorff, 'A laboratory model for the unstable planetary boundary layer', *J. Atmos. Sci.*, **31**, 1297–1307 (1974).

© 1998 John Wiley & Sons, Ltd.