TWODEE-2: A Shallow Layer Model for Dense Gas Dispersion on Complex Topography *

A. Folch a, A. Costa a R.K.S. Hankin b

aI.N.G.V., Napoli- Osservatorio Vesuviano, Italy.
bNational Oceanography Centre, Southampton SO14 3ZH, UK

Abstract

TWODEE-2 is a Fortran 90 code based on a previous code (TWODEE). It is designed to solve the shallow water equations for fluid depth, depth-averaged horizontal velocities and depth-averaged fluid density. The shallow layer approach used by TWODEE-2 is a compromise between the complexity of CFD models and the simpler integral models. It can be used for forecasting gas dispersion near the ground and/or for hazard assessment over complex terrains. The inputs to the model are topography, terrain roughness, wind measurements from meteorological stations and gas flow rate from the ground sources. Optionally the model can be coupled with the output of a meteorological processor which generates a zero-divergence wind field incorporating terrain effects. Model outputs are gas concentration, depth-averaged velocity, averaged cloud thickness and dose. The model can be a useful tool for gas hazard assessment by evaluating where and when lethal concentrations for humans and animals can be reached.

Key words: Dense gas transport, Fortran code, Gas hazard, Computational model

1 Introduction

Many volcanic and non-volcanic areas in Italy emit huge amounts of gas into the atmosphere. One of the most frequent gases discharged from both volcanic (e.g., Solfatara Volcano) and non-volcanic sources (e.g., central Italy vents) is...

* Code is available from server at http://www.iamg.org/CGEditor/index.htm
* Corresponding author. Present address: A. Folch, INGV Naples - Osservatorio Vesuviano, Via Diocleziano, 328 - 80124 Napoli, Italy.

Email addresses: afolch@ov.ingv.it (A. Folch), costa@ov.ingv.it (A. Costa), r.hankin@noc.soton.ac.uk (R.K.S. Hankin).
carbon dioxide (CO$_2$) which has a molecular weight greater than air. Under 
stable atmospheric conditions and/or in presence of topographic depressions, 
CO$_2$ concentration can reach high values resulting in lethal effects to humans 
or animals. In fact, several episodes of this phenomenon were recorded at 
different areas in central Italy (Rogie et al., 2000) and worldwide. The most 
tragic example was the 1986 degassing of Lake Nyos, Cameroon, when a dense 
cloud of carbon dioxide hugging the ground suffocated more than 1700 people 
in one night (Clarke, 2001).

The cloud dispersion of gases denser than air released from natural sources 
is governed by gravity and by the effects of lateral eddies which decrease the 
cloud density through the incorporation of surrounding air. In the initial phase 
the negative buoyancy controls the gas dispersion and the cloud follows the 
ground (gravitational phase). In contrast, when the density contrast becomes 
less important, gas dispersion is mainly governed by wind and atmospheric 
turbulence (passive dispersion phase) (e.g., Costa et al., 2005). In principle, 
gas dispersion can be studied by fully solving the transport equations for 
mass, momentum and energy, but because the demanding computational re-
quirements, different simplified models which describe only specific processes 
are commonly used. Approaches range from the simplest analytical Gaussian 
models to the more complex Computational Fluid Dynamics (CFD) models 
e.g., Macedonio and Costa, 2002). A compromise between the complexity of 
CFD models and the simpler integral models is given by the shallow layer 
approach which uses depth-averaged variables to describe gravity-driven flows 
of dense gas over complex topography (Hankin and Britter, 1999a; Venetsanos 
et al., 2003).

Here we describe **TWODEE-2**, a FORTRAN 90 code that solves a time-dependent 
model for dispersion of a heavy gas based on the shallow layer approach. 
TWODEE-2 is derived from the optimization and improvement of a previous 
FORTRAN 77 code (**TWODEE**) developed by Hankin and Britter (1999a,b,c). 
In the manuscript, firstly we present the model equations and overview the 
resolution algorithm. Secondly we describe the code structure and the I/O 
files. Finally we show an application example.

2 Heavy Gas Transport Model

2.1 Model variables

**TWODEE-2** is based on depth-averaged equations obtained by integrating con-
servation equations over the fluid depth. Such an approach is able to describe 
the cloud in terms of four variables: cloud depth, two depth-averaged hori-
zontal velocities, and depth-averaged cloud density as functions of time and position. Since real clouds do not have a definite upper surface it is necessary to define cloud depth in terms of the vertical concentration distribution. In fact, we point out that the actual vertical concentration profile is not uniform as for fluids usually described by shallow water equations, but characterised by an exponential decay (Hankin and Britter, 1999a). In twodee-2, $h$ is that height below which some fraction $\alpha$ of the buoyancy is located:

$$\int_{z=0}^{h} (\rho(z) - \rho_a) dz \equiv \alpha \int_{z=0}^{\infty} (\rho(z) - \rho_a) dz$$

(1)

where $\rho$ and $\rho_a$ are, respectively, the cloud and the ambient fluid (air) densities, $z$ is the vertical coordinate, and $\alpha$ is a constant. The choice $\alpha = 0.90 - 0.95$ is normally adopted. Depth-averaged density $\overline{\rho}$ and velocity components $(\overline{u}, \overline{v})$ are defined as:

$$h(\overline{\rho} - \rho_a) \equiv \int_{z=0}^{\infty} (\rho(z) - \rho_a) dz$$

(2)

$$h(\overline{\rho} - \rho_a)\overline{u} \equiv \int_{z=0}^{\infty} (\rho(z) - \rho_a) u(z) dz$$

(3)

$$h(\overline{\rho} - \rho_a)\overline{v} \equiv \int_{z=0}^{\infty} (\rho(z) - \rho_a) v(z) dz$$

(4)

The vertical distribution of density can be calculated from its depth-averaged value as (Hankin and Britter, 1999c):

$$\rho(z) = \rho_a + \frac{2}{S_1(\overline{\rho} - \rho_a)} \exp \left( - \frac{2}{S_1} \frac{z}{h} \right)$$

(5)

where $S_1$ is a shape parameter. The vertical concentration distribution $c$ (in ppm) results:

$$c(z) = c_b + (10^6 - c_b) \times \frac{\rho(z) - \rho_a}{\rho_g - \rho_a}$$

(6)

where $c_b$ is the background concentration (in ppm). Another useful quantity output by the model is the dose $D$, a temporal integrated variable defined as:

$$D(t, z) = \int_{0}^{t} [c(z)]^n dt$$

(7)
2.2 Model equations

Assuming an homogeneous fluid and a hydrostatic pressure distribution, the shallow water equations for flows having a non-uniform vertical profile are given by (Hankin and Britter, 1999a):

\[
\frac{\partial h}{\partial t} + \frac{\partial h\bar{u}}{\partial x} + \frac{\partial h\bar{v}}{\partial y} = u_{\text{entr}} + u_{\text{sou}}
\]  

(8)

\[
\frac{\partial h(\bar{p} - \rho_a)}{\partial t} + \frac{\partial h(\bar{p} - \rho_a)\bar{u}}{\partial x} + \frac{\partial h(\bar{p} - \rho_a)\bar{v}}{\partial y} = u_{\text{entr}}\rho_a + u_{\text{sou}}\rho_g
\]  

(9)

\[
\frac{\partial h\bar{u}}{\partial t} + \frac{\partial h\bar{u}^2}{\partial x} + \frac{\partial h\bar{u}\bar{v}}{\partial y} + \frac{1}{2}S_1 \frac{\partial g(\bar{p} - \rho_a)h^2}{\partial x} + S_1 g(\bar{p} - \rho_a)h \frac{\partial e}{\partial x} + \frac{1}{2} \rho C_D |\bar{u}| V_x + k\rho_a \left[ \frac{\partial}{\partial t} + u_a \frac{\partial}{\partial x} + v_a \frac{\partial}{\partial y} \right] \left[ h(\bar{u} - u_a) \right] = u_{\text{entr}}\rho_a u_a
\]  

(10)

\[
\frac{\partial h\bar{v}}{\partial t} + \frac{\partial h\bar{v}^2}{\partial y} + \frac{\partial h\bar{u}\bar{v}}{\partial x} + \frac{1}{2}S_1 \frac{\partial g(\bar{p} - \rho_a)h^2}{\partial y} + S_1 g(\bar{p} - \rho_a)h \frac{\partial e}{\partial y} + \frac{1}{2} \rho C_D |\bar{u}| V_y + k\rho_a \left[ \frac{\partial}{\partial t} + u_a \frac{\partial}{\partial x} + v_a \frac{\partial}{\partial y} \right] \left[ h(\bar{v} - v_a) \right] = u_{\text{entr}}\rho_a v_a
\]  

(11)

where the meaning of all variables is given in Table 1. The equations above express, respectively, the balances of volume, mass and components of momentum. Terms in the left hand sides of the momentum equations (10) and (11) include the temporal variation, the effect of convection, the pressure gradient for the case of hydrostatic but non-uniform density profile, the effect of ground slope, the surface shear stress (proportional to the drag coefficient \(C_D\)), the force per unit area exerted by turbulent shear stress and, finally, the leading edge terms that account for interaction among dense and ambient fluid. The latter gives a correction to the shallow water equations which give erroneous results for the leading edge as they assume that the pressure distribution is hydrostatic. This term is proportional to the semi-empirical coefficient \(k\), evaluated in terms of the front Froude number \(Fr\):

\[
k = \frac{2}{(S_1 Fr^2)}
\]  

(12)
The drag coefficient and the turbulent shear stress force are estimated as:

\[ C_D = \max \left( \min \left( \frac{2u_*^2}{u_a^2}; 10^{-2} \right); 10^{-4} \right) \]  

(13)

\[ \mathbf{V} = \zeta h \overline{\mathbf{v}} \nabla (h|\mathbf{u}| \nabla \mathbf{u}) \]  

(14)

where \( \zeta \) is a small constant of proportionality. On the other hand, the air entrainment velocity \( u_{\text{entr}} \) is calculated using empirical formulations (Hankin and Britter, 1999a):

\[ u_{\text{entr}} = \frac{K}{1 + bRi} \sqrt{u_*^2 + (\alpha_2 w_*)^2 + \frac{1}{2} C_D \alpha_3^2 |\mathbf{u}|^2 + \alpha_7^2 |\mathbf{u} - \mathbf{u}_a|^2} \]  

(15)

where \( b, \alpha_2, \alpha_3, \) and \( \alpha_7 \) are empirical constants. Eq. (15) reflects the top entrainment of ambient fluid through the incorporation of eddies into the body of the cloud (edge entrainment is not currently considered). TWODEE-2 is based on the numerical solution of the governing equations (8) to (11) together with the closure relationships (12) to (15) using the algorithm described in Hankin and Britter (1999b).

2.3 Wind model

Concerning definition of the wind field TWODEE-2 admits two options, uniform wind or spatially variable wind which allows to incorporate terrain effects. For the first option, when the wind is considered horizontally uniform, meteorological data at a height \( z = z_{\text{ref}} \) are directly read from a wind data file, commonly provided by a ground-based station. For the second option, when the wind is spatially variable, data at height \( z = z_{\text{ref}} \) are provided by the program DIAGNO. The program DIAGNO, also included in the TWODEE-2 package, is a meteorological processor based on the Diagnostic Wind Model (Douglas and Kessler, 1990). The DWM model generates a quasi steady-state gridded wind field from input data ("observations") at a point of the domain. The model adjusts the domain-scale mean wind for terrain effects and then performs a divergence minimization to ensure mass conservation. In both cases, uniform or spatially variable wind field, TWODEE-2 reads temperature and wind velocity at \( z = z_{\text{ref}} \) as inputs and uses these data to calculate the vertical wind profile according to the Monin-Obukhov Similarity Theory (e.g., Jacobson, 1999):

\[ U_{a}(z) = \frac{u_*}{K} \left[ \ln \left( \frac{z}{z_0} \right) - \psi_m \left( \frac{z}{L} \right) + \psi_m \left( \frac{z_0}{L} \right) \right] \]  

(16)
\( L \) and \( u_* \) are estimated using the non-iterative method of Louis (1979) based on the bulk Richardson number \( R_{ib} \):

\[
R_{ib} \approx \frac{g [\theta(z_{ref}) - \theta(z_0)] (z_{ref} - z_0)}{\theta(z_0) [u^2_a(z_{ref}) + v^2_a(z_{ref})]} \tag{17}
\]

The stability function \( \psi_m \) is evaluated from the Businger-Dyer relationship (e.g., Dyer, 1974) for unstable situations, i.e., \( z/L < 0 \), or in accord to Van Ulden and Holstag (1985) for stable situations i.e., \( z/L > 0 \), (for the details see, e.g., Jacobson, 1999).

### 2.4 Solving algorithm

`TWODEE-2` is based on the numerical solution of the governing equations (8) to (11) together with the relations (12) to (15) and a set of boundary conditions. `TWODEE-2` imposes different boundary conditions for outgoing and incoming fluxes: for outgoing flux, zero derivative conditions, whereas for incoming flux, null concentrations at boundaries (i.e., \( h = 0, \bar{p} = \rho_a, \bar{u} = 0, \) and \( \bar{v} = 0 \)). The numerical method is based on the Flux Corrected Transport (FCT) scheme of Zalesak (1979). This scheme combines the low numerical diffusion of high order schemes with the absence of numerical oscillations typical of low order schemes. Fundamentally, FCT calculates the fluxes between adjacent elements using a weighted average of flux as computed by a low order scheme and a high order scheme. The weighting is done in such a manner as to use the high order scheme unless doing so would result in the creation of overshoots (that is, new extrema in the advected quantity) not predicted by the low order scheme. The stability of the numerical scheme is ensured by using a time step \( \Delta t \) equal than the critical:

\[
\Delta t = \frac{C_r \min(\Delta x, \Delta y)}{|u| + \sqrt{\frac{h_0(\bar{p} - \rho_a)}{\bar{p}}}} \tag{18}
\]

where \( C_r \) is the Courant number. For a detailed description of the solving algorithm see Hankin and Britter (1999b) and the references therein.

### 3 The `TWODEE-2` set up

#### 3.1 Installation and folder structure

- On a Unix/Linux/Mac X OS:
(1) Decompress and untar the file `twodee2.tar.gz`.

(2) Compile the program `diagno` . Enter the directory `twodee2/Programs/Diagno/Sources`, then issue the command “`make`”. You can edit the Makefile to select your favourite compiler.

(3) Compile the program `TWODEE-2` . Enter the directory `twodee2/Programs/Twodee2/Sources`, then issue the command “`make`”. You can edit the Makefile to select your favourite compiler.

- On a Windows OS decompress the file `twodee2.tar.gz` on your selected directory. The `twodee2.tar.gz` file already contains Windows executables for `TWODEE-2` and `diagno` programs.

Table 2 shows the `TWODEE-2` folder structure. The directory `twodee2/Programs` contains the programs `diagno` and `TWODEE-2` with the corresponding source files. The directory `twodee2/Runs` contains the runs, each within its own folder. An example run named “Example” is provided with the installation.

### 3.2 Program run

`TWODEE-2` and `diagno` programs can be launched typing, respectively,

- “`Twodee2.exe problemname.inp problemname.log`”
- “`Diagno.exe problemname.inp problemname.log`”

where `problemname.inp` and `problemname.log` are the names (including paths) of the control input file and of the log file. Both file names are passed as a program call argument. However, it is highly recommended to launch the programs using the script files included in the package. For instance, to run the problem “Example”:

- On a Windows OS enter the folder `twodee2/Runs/Example` and launch the script `Example.win`.
- On a Mac X/Unix/Linux OS enter the folder `twodee2/Runs/Example` and launch the script `Example.unix`.

**NOTE:** To create a new run simply create a new folder (with sub-folders `infiles` and `outfiles`), copy the input files and the script into it and modify the line of the script which defines the variable `problemname`.

### 4 The `TWODEE-2` input files

`TWODEE-2` needs the following input files:
• File `problemname.inp` - Control file that defines a TWODEE-2 (and a DIAGNO) run. Mandatory. This file is passed to the program(s) as a call argument and contains a set of blocks that define all the computational and physical parameters needed by the model(s).

• File `topography.dat` - Regional ground elevation file. Optional. The topography file specifies ground elevation $e$ at a regional scale (i.e., in a region typically larger than the computational domain). Topography must be specified on a structured grid using arbitrary (but constant) grid spacing (e.g., 5 m, 10 m, 100 m, etc). Discretizations along $x$- and $y$-directions can be different. The only necessary requirement is that the computational domain must lay within the bounds of the region where topography is specified. TWODEE-2 reads the topography file and automatically interpolates elevations onto the nodes of the computational grid.

• File `terrain.dat` - Regional terrain roughness length file. Mandatory. The terrain file specifies the roughness length $z_0$ at a regional scale. The file format is analogous to that of the regional topography file. However, these two files can cover different regions and/or have different spatial discretizations.

• File `restart.dat` - Restart (initial conditions) file. Optional. The restart file can be used to start a new run from the end of a previous simulation. It is automatically created each time TWODEE-2 prints results or at the end of a run. Any restart file previously created is destroyed whenever a new restart file is printed.

• File `source.dat` - Source term (dense gas fluxes) file. Mandatory. The source file specifies dense gas fluxes (in mass flow rate or mass flow rate per unit area) from different rectangular areas or point sources. TWODEE-2 reads this file and automatically calculates the upward source velocity $u_{sou}$ (see eqs. (8) and (9)) and interpolates the mass flow rate onto the nodes of the computational grid. Interpolation is done ensuring mass conservation. The advantage of this approach is that the source file becomes independent of the computational mesh (i.e., the source file is created only once and is the same regardless the location and/or the spatial resolution of the computational grid).

• File `winds.dat` - Meteorological data file. Mandatory. The wind data file contains meteorological data at different time slices. If the record `WIND_MODEL` in the control input file is `UNIFORM`, TWODEE-2 reads this file and estimates the Atmospheric Surface Layer parameters. Otherwise, if the record `WIND_MODEL` is `DIAGNO`, this file is instead read by DIAGNO which uses values as input and TWODEE-2 simply uses the DIAGNO output (file `diagno.res`).

• File `points.dat` - File that defines the coordinates of the tracked points (points where time evolution of concentration is output). Optional.

• File `boxes.dat` - File that defines the coordinates of the tracked boxes (areas where evolution of averaged concentration is output). Optional.

• File `diagno.res` - DIAGNO output file. Optional.
For a detailed description on the formats and contents of the files above see the file README_Files.pdf included in the TWODEE-2 package.

NOTE: File names are given just for illustrative purposes. Names and locations of TWODEE-2 I/O files are absolutely free and are defined by the user within the control file problemname.inp.

5 The TWODEE-2 output files

TWODEE-2 outputs the following files:

- File problemname.log. This file, passed as a program call argument, contains information concerning the run (summary of input data, run time error messages, CPU time, etc.). It also outputs some basic indicators of the cloud evolution every minute.
- Files of results in GRD-format. For each user-specified time TWODEE-2 can generate 2D contour-files written in GRD-format for the following variables: $h$, $\pi$, $\tau$, $\rho$, $c$, and $D$. These files can be read directly by several plotting programs like the commercial software Grapher©. Alternatively, the user may also generate its own plots using functions from several free packages (e.g., gnuplot).
- TWODEE-2 can also output CSV-format files with concentration at defined points and/or boxes every minute. The CSV (“comma separated variables”) is a free ASCII format in which variables are stored in columns separated by commas. It can be read by Excel© or by any text editor.

6 Application Example

TWODEE-2 was applied to simulate dense gas dispersion episode from natural sources in Central Italy. As an example, we report results for an experiment carried out in the area of Caldara di Manziana (CdM), Italy, a quaternary volcanic crater located 20 km NW of Rome (see Costa et al. (2007) for details). At CdM the emission of CO$_2$ occurs both as soil diffuse degassing and from two localized sources, a water pool and a vent (Figure 1). During an experiment carried out on 5 and 6 February 2007 an open path infrared CO$_2$/H$_2$O analyzer measured CO$_2$ concentration at 2.20 m above the ground for 15 hours, from 17:40 (local time, LT) of FEB 5th 2007 to 08:40 (LT) of FEB 6th 2007. During the same time span, a meteorological station placed at 3.45 m height acquired pressure, air temperature and wind speed. Both datasets were averaged every ten minutes. The computational domain (1 km$^2$) was discretized using a grid of 125x125 points with a nodal distance of 8m. Zero initial cloud depth and velocity were assumed. The averaged critical time step was around
0.3s, implying almost 200000 iterations to simulate the whole time interval (54000s). The CPU time required was less than 2 hours on a standard Pentium IV desktop PC.

A view from above of the gas plume at 2.2 m height, showing the temporal evolution at different time slices, is plotted on Figure 2. The cloud concentration reflects both wind intensity and direction. Wind blew SW during the first 2h, then oscillated between SE and NE and, finally, shifted towards SSE in the last 2h. Figure 3 compares concentrations measured by the CO$_2$/H$_2$O analyzer at the point having UTM coordinates (260252;4663873;2.20) with the simulated concentrations averaged over an area of 25m × 25m centered at the same point. Note how TWODEE-2 tracks the evolution of concentration despite the turbulent nature of the flow, which produces a highly oscillatory behaviour.

7 Discussion, validity and limitations

TWODEE-2 is a FORTRAN 90 code for the dispersion of heavy gases based on the shallow layer approach. This approach is valid in the limit \( H^*_s/L^*_s \ll 1 \), where \( H^*_s \) is the undisturbed fluid height and \( L^*_s \) the characteristic wave length scale in the flow direction (Ferrari and Saleri, 2004). The code is suitable for simulating the gravitational phase of gas dispersion when the negative buoyancy controls the transport and the cloud follows the ground. When the density contrast becomes less important (\( \Delta \rho < 0.001 \) kg/m$^3$ Mohan et al., 1995), a model based on the passive dispersion approximation is more suitable. TWODEE-2 removes several limitations of its parent FORTRAN 77 version:

- The computational grid is geo-referenced and can have an arbitrary number of nodes.
- Topography, terrain roughness, and dense gas fluxes for diffuse or punctual sources can be extracted from regional files.
- A non-uniform roughness height allows the user to simulate domains which contain zones with contrasted properties (e.g., a forest or a lake).
- The wind field is described through the similarity theory and can be both uniform and non-uniform. Time-dependent spatially-variable wind fields accounting for terrain influence can be generated by the program DIAGNO by furnishing a domain mean wind value.
- Has multiple output possibilities. For example, TWODEE-2 can output the height at which a certain concentration value is achieved or can track the evolution of concentration at discrete points or at given areas.
- Improves code performance. TWODEE-2 solves the same governing equations using the same algorithm that its parent software. However, the implementation of the code is more efficient, a fact that reduces the computational
times by a factor 2 to 5 depending on each particular run.

Potential applications of the model include CO$_2$ concentration distribution under a variety of atmospheric conditions and assessment of environmental hazards from potential leakage and seepage related to accumulation of large quantities of injected CO$_2$ in geologic sequestration sites (e.g., Oldenburg and Unger, 2003, 2004). The results obtained in test applications (Costa et al., 2007) show the high performance of the model when the input parameters are correctly furnished. However, the code still has some limitations to be corrected in a future. Improvements should include:

- The possibility to use non-uniform and/or non-structured computational meshes.
- The edge entrainment term to simulate the introduction of ambient fluid not only from the top of the cloud but also from the edges.
- To allow DIAGNO to read data from more than one meteorological station.
- To account for thermodynamic effects such as gas condensation by introducing an additional equation for gas enthalpy.

Acknowledgments

This work was supported by the Department of Civil Protection of Italy - INGV Reasearch Project V5, 2005-2007. We thank two anonymous reviewers for improving the quality of the manuscript.
Fig. 1. Map of CO$_2$ flux at CdM. The measured CO$_2$ soil fluxes range from $2.3 \times 10^{-8}$ kg m$^{-2}$ s$^{-1}$ to $2.8 \times 10^{-4}$ kg m$^{-2}$ s$^{-1}$, with an average value of $7.7 \times 10^{-6}$ kg m$^{-2}$ s$^{-1}$ or 160 ton d$^{-1}$. The contribution from the focused vents is around 0.17 kg s$^{-1}$ or 15 ton d$^{-1}$ (Rogie et al., 2000). Axes show UTM coordinates in m.
Fig. 2. Time evolution of concentration at 2.2 m height. Four different time slices at 2, 6, 10 and 14h are plotted for illustrative purposes. Axes show UTM coordinates in m.
Fig. 3. Comparison between measured (red dots) and simulated (black line) concentrations averaged every ten minutes. Point UTM coordinates are (260252,4663873).
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b$</td>
<td>Entrainment Velocity Empirical Coefficient. See eq. (15). Usually set to 0.11</td>
</tr>
<tr>
<td>$c$</td>
<td>Cloud Dense Gas Concentration in ppm</td>
</tr>
<tr>
<td>$c_b$</td>
<td>Dense Gas Background Concentration in ppm</td>
</tr>
<tr>
<td>$C_D$</td>
<td>Skin Friction (Drag) Coefficient. See eq. (13)</td>
</tr>
<tr>
<td>$C_r$</td>
<td>Courant Number. Usually set to 0.25</td>
</tr>
<tr>
<td>$D$</td>
<td>Dose. See eq. (7)</td>
</tr>
<tr>
<td>$e = e(x,y)$</td>
<td>Terrain Elevation</td>
</tr>
<tr>
<td>$F_r$</td>
<td>Front Froude Number. See eq. (12). Usually set to 1</td>
</tr>
<tr>
<td>$g$</td>
<td>Gravity acceleration</td>
</tr>
<tr>
<td>$g(\bar{\rho} - \rho_a)$</td>
<td>Cloud Buoyancy</td>
</tr>
<tr>
<td>$h$</td>
<td>Cloud Depth</td>
</tr>
<tr>
<td>$K$</td>
<td>Von Karman Constant. Usually set to 0.4</td>
</tr>
<tr>
<td>$L$</td>
<td>Monin-Obukhov Length</td>
</tr>
<tr>
<td>$n$</td>
<td>Toxicity Exponent. See eq. (7)</td>
</tr>
<tr>
<td>$P$</td>
<td>Pressure</td>
</tr>
<tr>
<td>$R_{ib}$</td>
<td>Bulk Richardson Number</td>
</tr>
<tr>
<td>$S_1$</td>
<td>Shape Parameter. See eq. (5). Usually set to 0.5</td>
</tr>
<tr>
<td>$t$</td>
<td>Time</td>
</tr>
<tr>
<td>$T$</td>
<td>Temperature</td>
</tr>
<tr>
<td>$\mathbf{u} = (\bar{u}, \bar{v})$</td>
<td>Depth-Averaged Velocity</td>
</tr>
<tr>
<td>$u_{entr}$</td>
<td>Air Entrainment Velocity. See eq. (15).</td>
</tr>
<tr>
<td>$u_{sou}$</td>
<td>Dense Gas Inflow Velocity</td>
</tr>
<tr>
<td>$u_*$</td>
<td>Friction Velocity</td>
</tr>
<tr>
<td>$\mathbf{u}_a = (u_a, v_a)$</td>
<td>Ambient Fluid Velocities along $(x,y)$ respectively</td>
</tr>
<tr>
<td>$U_a$</td>
<td>Ambient Fluid Velocity Modulus</td>
</tr>
<tr>
<td>$\mathbf{V} = (V_x, V_y)$</td>
<td>Turbulent Shear Stress Force (per unit area)</td>
</tr>
<tr>
<td>$w_*$</td>
<td>Atmospheric Convective Velocity</td>
</tr>
<tr>
<td>$(x,y)$</td>
<td>Horizontal Spatial Coordinates</td>
</tr>
<tr>
<td>$z$</td>
<td>Vertical Coordinate</td>
</tr>
<tr>
<td>$z_0 = z_0(x,y)$</td>
<td>Roughness Length</td>
</tr>
<tr>
<td>$z_{ref}$</td>
<td>Reference Height</td>
</tr>
</tbody>
</table>

continue →
### Table 2

<table>
<thead>
<tr>
<th>twodee2</th>
<th>Programs</th>
<th>Diagno</th>
<th>Sources</th>
<th>Source code files</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Diagno.win.exe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Windows executable</td>
</tr>
<tr>
<td></td>
<td>Twodee2</td>
<td>Sources</td>
<td></td>
<td>Twodee2.win.exe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Windows executable</td>
</tr>
<tr>
<td>Runs</td>
<td>Example</td>
<td>infiles</td>
<td></td>
<td>Example input files</td>
</tr>
<tr>
<td></td>
<td></td>
<td>outfiles</td>
<td></td>
<td>Example output files</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Example.unix</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Unix/Linux/Mac X script</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Example.win.bat</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Windows (DOS) script</td>
</tr>
<tr>
<td></td>
<td>README_Files.pdf</td>
<td></td>
<td></td>
<td>Description of I/O files</td>
</tr>
</tbody>
</table>
References


dimensional shallow layer model for dense gas dispersion including complex

Zalesak, S., 1979. Fully multidimensional flux-corrected method transport for