rapporti tecnici INGV

DISGAS-2.0: A model for passive DISpersion of GAS

332

Istituto Nazionale di
Geofisica e Vulcanologia

rapporti tecnici INGV

DISGAS-2.0: A MODEL FOR PASSIVE DISPERSION OF GAS[*]

Antonio Costa[1], Giovanni Macedonio[2]

[1]**INGV** (Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Bologna)
[2]**INGV** (Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Napoli - Osservatorio Vesuviano)

[*]Code is available from server at http://datasim.ov.ingv.it/models/disgas.html

332

Index

Abstract

The code *DISGAS* (model for passive DISpersion of GASes) is a Eulerian model for passive dispersion of diluted gas and fine dust particles. Turbulent diffusion is based on the K-theory and the wind field can be evaluated assuming either a uniform wind profile based on the Similarity Theory or using a terrain-following mass-consistent wind model. *DISGAS* can be used to forecast concentration of gas (or dust) over complex terrains. The inputs to the model are topography, wind observations from meteorological stations, atmospheric stability information, and gas flow rate from a discrete number of point sources. *DISGAS* is written in fortran 90. Here we describe the model and an application example.

Introduction

Several volcanic and non-volcanic sources in central Italy and worldwide emit a large amount of gas, such as carbon dioxide (CO_2) and other components, through focused or diffuse degassing, producing consistent gas plumes that are dispersed in the lower part of the atmosphere [e.g., Chiodini et al., 2005, 2007, Costa et al., 2005].

The cloud dispersion of a gas denser than air released from a given source, is governed by the gravity and by the effects of lateral eddies which increase the mixing with air around the edges of the plume, decreasing its density. In the initial phase the negative buoyancy controls the gas dispersion and the cloud follows the ground (gravitational phase). During this phase, the dispersion of heavy gas is markedly different from passive gas dispersion (and even from positively buoyant gas). When density contrast is not important, gas dispersion is basically governed by wind and atmospheric turbulence (passive dispersion).

From a theoretical point of view, dispersion of natural gases can be fully studied by solving the complete equations for mass, momentum, and energy transport. However, in actual practice, different simplified approaches able to describe only specific phases or aspects of gas transport are commonly used.

The most general and computationally most expensive models are given by three-dimensional CFD models based on the transport theory of mass, momentum, energy, and species [e.g., Macedonio and Costa, 2002; Cortis and Oldenburg, 2009; Cerminara et al., 2015]. This approach is able to simulate dispersion of both heavy and light gases accounting for topographic effects, variation of atmospheric conditions and wind direction, etc.

For a gas denser than air the shallow layer approach, which uses depth-averaged variables to describe the flow behaviour, is commonly used [e.g., Hankin and Britter, 1999; Costa et al., 2008; Folch et al., 2009]. Models based on this approach are applied to describe gravity driven flows of dense gas over complex topography [Costa et al., 2008; Chiodini et al., 2010].

For dispersion of a diluted gas, which is passively driven by wind advection and atmospheric turbulence, simpler advection-diffusion equations can be solved [e.g. Prabha and Mursch-Radlgruber, 1999a], like in the model we describe here.

DISGAS is based on an explicit solver of the advection-diffusion equation which uses a second-order Lax-Wendroff scheme [e.g., Ewing and Wang, 2001] and can be coupled with a wind profile described by the similarity theory [e.g., De Bruin et al., 2000] or with a mass-consistent wind model [Douglas and Kessler, 1990].

The generalization of the advection-diffusion equation including a sedimentation term, allows the model to be applied to study even the dispersion of a diluted mixture of very fine particles into the atmospheric surface layer. For instance, with such a generalization the model can describe the dispersion of small fibres like asbestos or small powders like pollen.

Here, firstly we summarize the theory and equations on which the model is based, then we illustrate the numerical method used to solve the equations, and, in the Appendix A, we describe the program setup, folder structure, input/output files, and how to run the code. Simulation results obtained for the dispersal of CO_2 emitted from the Solfatara crater in the Campi Flegrei, in surroundings areas of the he city of Naples, Italy, is also presented as an example of application [Granieri et al., 2013]. The format and the description of input and output files is reported in the Appendix B and Appendix C.

1. Physical model

A full description of gas dispersion into the atmosphere requires the solution of transport equations for mass, momentum, energy, and species [see e.g. Macedonio and Costa, 2002]. This implies a relatively high computational effort and often is practically impossible to study gas dispersion over complex large domains, especially in the surface layer where a wide range of length scales are involved.

In order to reduce the computational time it is necessary introducing some simplifying assumptions such as that of incompressible fluid. For dilute gases the passive dispersion approach can be used. In this case the main objective is to solve the advection-diffusion equation for the gas concentration $C = c + c'$:

$$\frac{\partial c}{\partial t} + \frac{\partial}{\partial x}(uc) + \frac{\partial}{\partial y}(vc) + \frac{\partial}{\partial z}(wc) + \frac{\partial}{\partial x}\left(\overline{u'c'}\right) + \frac{\partial}{\partial y}\left(\overline{v'c'}\right) + \frac{\partial}{\partial z}\left(\overline{w'c'}\right) = Q \tag{1}$$

where $V = (u + u', v + v', w + w')$ is the wind field and Q is the source term, i.e. the gas mass flux from the ground. Plain terms in lower case letters represent the average part, terms with the symbol prime ' their turbulent fluctuations and the overbar defines the filtering operation on the product of the fluctuation terms [e.g. Sagaut, 2006].

Hence, as in other studies [e.g. Prabha and Mursch-Radlgruber, 1999a,b; Costa et al., 2005], we do not solve the complete set of the coupled equations for mass, momentum, energy, and concentration, but we solve eq. (1) and approximate the wind field as given by a uniform profile (for relatively small domains) or estimate it using a diagnostic wind model which produces a zero three-dimensional divergence velocity field consistent with the measured values, avoiding artificial generation or loss of gas. Turbulent terms are parameterized according with the K-theory as described later.

The passive dispersion approach can be used when the gas is diluted enough, and the validity of such an approximation can be assessed by estimating the Richardson number [e.g., Britter and McQuaid, 1988; Cortis and Oldenburg, 2009; Costa et al., 2013]:

$$Ri = \frac{1}{V^2}\frac{(g'q)^{2/3}}{R^{2/3}} \tag{2}$$

where $g' = g(\rho_g - \rho_a)/\rho_a$ is the reduced gravity acceleration (being g the gravity acceleration, ρ_a and ρ_g the air and the gas density respectively), q is the volumetric flow rate, R the plume size (e.g. plume radius), and V is the wind velocity at the reference altitude (i.e., 10 m). For $Ri < 0.25$ transport is substantially passive whereas for $Ri > 1$ is mainly density driven [Cortis and Oldenburg, 2009; Costa et al., 2013].

1.1 Wind field

For evaluating the wind field *DISGAS* has two options:

i) The simplest choice (named SIMILARITY) consists of assuming a horizontal wind profile calculated in accord to the Monin-Obukhov similarity theory [e.g., Monin and Yaglom, 1979; Byun 1990]:

$$U(z) = \frac{u_*}{k}\left[\ln\left(\frac{z}{z_0}\right) - \phi_m\left(\frac{z}{L}\right)\right] \tag{3}$$

where $k = 0.4$ is the von Karman constant, z_0 is the roughness length, L denotes the Monin-Obukhov length, u_* the friction velocity, and ϕ_m the atmospheric stability function for the momentum (e.g., Dyer, 1974; Jacobson, 1999). The friction velocity can be calculated as:

$$\frac{u_*}{k} = \frac{U(z_1)}{\ln\left(\frac{z_1}{z_0}\right) - \phi_m\left(\frac{z_1}{L}\right) + \phi_m\left(\frac{z_0}{L}\right)} \tag{4}$$

where z_1 is the height at which the wind velocity $U(z_1)$ is estimated. Roughness can be assumed uniform or given as a matrix covering the computational domain. Using the SIMILARITY option implies the wind can be not mass-consistent (e.g. $\nabla \cdot V$ can be locally not null). It is adequate for flat small domains but is not suitable for large domain and complex terrains.

ii) Another choice (named `diagno`) consists of a wind field estimated by using a wind model based on the Diagnostic Wind Model (DWM) that was developed by the United States Environmental Protection Agency (EPA). This model generates the wind $\boldsymbol{V}=(u, v, w)$ in a terrain-following coordinate system.

DIAGNO needs topography data, average wind on the computational domain, and atmospheric stability information within the scale of the domain (i.e. the temperature gradient dT/dz).

In a first step the domain-mean wind is adjusted for the kinematic effects of terrain (lifting and acceleration of the airflow over terrain obstacles), thermodynamically generated slope flows, and blocking effects.

In a second step, wind observations, when available, are added to the first step field, and an objective analysis scheme is used to produce a new gridded field (u, v, w). The scheme is designed so that the observations are used to define the wind field within a user-specified radius of influence while the first step (u, v, w) field is used in sub-regions in which observations are unavailable.

Finally, a divergence-minimization procedure in a terrain-following coordinate is iteratively applied until the inequality:

$$\nabla \cdot \boldsymbol{V} \leq \boldsymbol{\varepsilon} \tag{5}$$

is satisfied ($\boldsymbol{\varepsilon}$ is an arbitrarily user-defined small number).

The final product of DWM is an approximately null-divergence wind field consistent with the observations (for further information see Douglas and Kessler, [1990]). The approximation of a null-divergence wind field is generally applicable up to a kilometre above ground level [Dutton and Fichtl, 1969] therefore is suitable to our study, since we deal dispersion in the surface layer, i.e. the lowest part of the Planetary Boundary Layer (PBL). Since terrain-following coordinate are used such an approximation still holds even if ground level varies significantly within the computational domain. *DISGAS* model interpolates the wind field into its computational grid.

1.2 Turbulence parameterization

Turbulent fluxes are given by the product of the fluctuation terms, and in agreement with the "K-theory", they are expressed in terms of gradients of average concentration:

$$\overline{u'c'} = -K_x \frac{\partial \overline{c}}{\partial x}; \quad \overline{v'c'} = -K_y \frac{\partial \overline{c}}{\partial y}; \quad \overline{w'c'} = -K_z \frac{\partial \overline{c}}{\partial z} \tag{6}$$

where the turbulent diffusivity tensor is assumed to be diagonal and horizontally isotropic, i.e. $K = (K_x, K_y, K_z)$.

Inside the atmospheric surface layer, the Monin-Obukhov similarity theory allows the vertical turbulent diffusivity K_z to be estimated in terms of friction velocity u_* and the Monin-Obukhov length L:

$$K_z = \frac{kzu_*}{\phi_h} \tag{7}$$

where z is the distance from the ground and ϕ_h is the atmospheric stability function for temperature [e.g., Jacobson,1999]:

$$\phi_h = \begin{cases} Pr_t + \beta_h \frac{z}{L} & \text{for} \quad \frac{z}{L} > 0 \ \text{(stable condition)} \\ Pr_t \left(1 - \gamma_h \frac{z}{L}\right)^{-1/2} & \text{for} \quad \frac{z}{L} > 0 \ \text{(unstable condition)} \\ Pr_t & \text{for} \quad \frac{z}{L} = 0 \ \text{(neutral condition)} \end{cases} \tag{8}$$

with the turbulent Prandtl number $Pr_t = 0.95$, $\beta_h = 7.8$, and $\gamma_h = 11.6$. From a practical point of view, L is often a quantity difficult to estimate directly (although modern sonic wind anemometers can furnish directly the value of L). Instead, it is operationally easier to evaluate this length using the "Bulk Richardson number" method [Golder, 1972]:

$$Ri_b = \frac{g}{T} \frac{\partial \theta}{\partial z} \frac{z_m^2}{u_m^2} \tag{9}$$

where θ and T are the potential and the absolute air temperature, z_m the geometric mean thickness of the considered layer and u_m is the logarithmically interpolated wind speed at z_m. The Richardson number Ri is a measure of the dynamics on the buoyancy effects: we have atmospheric instability when $Ri_b < Ri_{crit}$ and stability for $Ri_b > Ri_{crit}$ ($Ri_{crit} \simeq 0.2$). The term z/L can be calculated analytically in terms of Ri_b by using the relationship [Byun, 1990]:

$$\frac{z}{L} = Ri_b \frac{\phi_m^2}{\phi_h} \tag{10}$$

As explained in the Appendix B, Monin-Obukhov length L can be directly furnished by the user (if the option SONIC is used for the wind file) or estimated from air temperature difference following the "Bulk Richardson number" method (if the option CUP is used for the wind file).

Concerning the horizontal eddy diffusivity, K_z and K_H can be assumed either approximated as constant or considered variable. In the latter case, following the large eddy approach [e.g., Smagorinsky, 1963; Pielke et al., 1992; Byun and Schere, 2006], the horizontal eddy diffusivity $K_x = K_y = K_H$ is considered isotropic and estimated as [Byun and Schere, 2006]:

$$K_H = Pr_t \alpha^2 \Delta x \Delta y \sqrt{\left(\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)^2} \tag{11}$$

were $\alpha \simeq 0.28$ is a dimensionless constant empirically determined, and Δx and Δy the horizontal grid spacing. A lower limit of $K_H = 0.075(\Delta x \Delta y)^{4/3}$ m^2/s is imposed [Pielke et al., 1992].

As we mentioned above, here we neglect the off-diagonal horizontal and vertical turbulent diffusion components [e.g. Toon et al., 1988, Jacobson et al., 1996; Park and Kim, 1999]. For a more general treatment that includes the off-diagonal diffusion components see Byun and Chin [1999].

2. Numerical solution

Considering the assumptions introduced above and a null-divergence wind field, in a terrain-following coordinate system [Douglas and Kessler, 1990] consisting of

$$X = x, \ \ Y = y, \ \ Z = z - h(x,y) \tag{12}$$

being $h(x,y)$ the height of the topography), eq. (1) for the mean scaled concentration C_* (i.e. consistent with transformation 12) may be re-written in a generalized form as [e.g., Toon et al., 1988; Jacobson et al., 1996, Costa et al., 2006]:

$$\frac{\partial C_*}{\partial t} + U \frac{\partial C_*}{\partial X} + V \frac{\partial C_*}{\partial Y} + W \frac{\partial C_*}{\partial Z} = \frac{\partial}{\partial X}\left(K_H \frac{\partial C_*}{\partial X}\right) + \frac{\partial}{\partial Y}\left(K_H \frac{\partial C_*}{\partial Y}\right) + \frac{\partial}{\partial Z}\left(K_Z \frac{\partial C_*}{\partial Z}\right) + Q_* \tag{13}$$

here (U, V, W) denote the scaled wind speeds, K_H and K_z are the diagonal scaled diffusion coefficients and Q_* the source term in the new coordinates. In the case of the simple transformations (12) the Jacobian is equal to the unit, $J=\partial z/\partial Z=1$, therefore $c(x,y,z) = C_*(X,Y,Z)$ and $Q=Q_*$. In the case of particle dispersion, here W represents the sum of the vertical wind component and particle settling velocity V_s.

The advective terms are discretised according to the second-order Lax-Wendroff scheme [e.g., Ewing and Wang, 2001]. For instance, the advective term in the X-direction yields:

$$-\Delta t \left(U \frac{\partial C_*}{\partial X}\right)_{i,j,k}^n = -\frac{Cr}{2}\left(C_{*,i+1,j,k}^n - C_{*,i-1,j,k}^n\right) + \frac{Cr^2}{2}\left(C_{*,i+1,j,k}^n - 2C_{*,i,j,k}^n + C_{*,i-1,j,k}^n\right) \tag{14}$$

where $Cr = U\Delta t/\Delta X$ represents the Courant number (Δx is the grid spacing and Δt the computational time step). In order to reduce the numerical over-and under-shooting that commonly affects high-order schemes near discontinuities one can use slope limiter methods. In fact, these methods preserve the monotonicity of the solution while the accuracy remains higher than the first order upwind methods [Sweeby, 1984; Wang and Hutter, 2001; Ewing and Wang, 2001]. For instance, applying the minmod slope-limiter method and

considering the general case with non-uniform velocity, equation (14) becomes:

$$-\Delta t \left(U \frac{\partial C_*}{\partial X} \right)^n_{i,j,k} = -\frac{\Delta t}{\Delta X} \left(U^n_{i,j,k} C^n_{*,i,j,k} - U^n_{i-1,j,k} C^n_{*,i-1,j,k} \right) - $$
$$\frac{\Delta X}{2} \left[Cr^n_{i,j,k} \left(1 - Cr^n_{i,j,k} \right) \sigma^n_{i,j,k} - Cr^n_{i-1,j,k} \left(1 - Cr^n_{i-1,j,k} \right) \sigma^n_{i-1,j,k} \right] \tag{15}$$

where $Cr^n_{i,j,k} = U^n_{i,j,k} \Delta t / \Delta X$, and:

$$\sigma^n_{i,j,k} = \text{minmod} \left\{ \frac{C^n_{*,i+1,j,k} - C^n_{*,i,j,k}}{\Delta X}, \frac{C^n_{*,i,j,k} - C^n_{*,i-1,j,k}}{\Delta X} \right\} \tag{16}$$

with minmod(*a*,*b*)=1/2(sgn(*a*)+sgn(*b*))min(|*a*|,|*b*|) [Ewing and Wang, 2001]. The introduction of alternative limiter methods is straightforward.

The diffusion terms are evaluated using a central difference scheme for the general case of a not uniform turbulent diffusivity tensor, e.g. in the *X*−direction using:

$$\frac{\partial}{\partial X} \left(K_H \frac{\partial C_*}{\partial X} \right)^n_{i,j,k} = \frac{1}{2\Delta X^2} \left[\left(K^n_{H,i,j,k} - K^n_{H,i-1,j,k} \right) C^n_{*,i-1,j,k} - \left(K^n_{H,i+1,j,k} + 2K^n_{H,i,j,k} + K^n_{H,i-1,j,k} \right) C^n_{*,i,j,k} + \right.$$
$$\left. \left(K^n_{H,i+1,j,k} - K^n_{H,i,j,k} \right) C^n_{*,i+1,j,k} \right] \tag{17}$$

The stability of the numerical scheme is ensured by using a time step Δt lower than the critical. As established by Hindmarsh et al. [1984], an explicit scheme for the multidimensional advection diffusion equation is numerically stable if the condition:

$$\Delta t \leq \frac{1}{\left[2 \left(\frac{K_H}{\Delta X^2} + \frac{K_H}{\Delta Y^2} + \frac{K_Z}{\Delta Z^2} \right) + \frac{U}{\Delta X} + \frac{V}{\Delta Y} + \frac{W}{\Delta Z} \right]} \tag{18}$$

Concerning the *Z*−direction, the numerical scheme was generalized to a non-uniform vertical grid.

3. Application example

DISGAS code was applied to simulate gas dispersion from the Solfatara volcano, Campi Flegrei, Italy [Granieri et al., 2013]. Solfatara releases more than 1000 t d^{-1} of CO_2 through soil diffuse degassing from a relatively small area (0.5-1 km^2) [Chiodini et al., 2001; Granieri et al., 2013]. Temperature of the gases emitted from Solfatara area is relatively high: fumaroles temperature is between 96 °C and 162 °C [Chiodini et al., 2001] and, flux weighted temperature of diffusing soil is 66 °C [Costa et al., 2005]. This implies a density decrease due to temperature that almost balances the increase due to the greater molecular weight directly at the source (M_{CO2}/M_{air} = 44/29 while T_{CO2}/T_{air} ∼ 400/300). Moreover low wind conditions are rarely recorded at Solfatara (daily average wind intensity $U > 1$ m/s is about 75% of the total and the cases with $U < 0.1$ m/s 0.05% of the total). For these reasons, the passive dispersion assumption, described in Section 2, is appropriate [Costa et al., 2005; Granieri et al., 2013].

Simulation results of gas dispersal were first compared with observations, then the model was used to investigate space and time evolution of the volcanic CO_2 plume under different meteorological conditions in order to assess its potential hazard.

Using DISGAS, Granieri et al., [2013] showed that the urban area of Naples is mantled by the volcanic plume of the Solfatara when atmospheric circulatory patterns are dominated by the locally frequent marine winds. In these conditions the air CO_2 content increases above the normal air CO_2 content, with differences varying from more than one thousand ppm (above the normal atmospheric background) in proximity of the Solfatara crater to a few ten ppm at a few kilometres from the source. Although these values indicate that there is no risk to the population, the elevated CO_2 concentration contributes to the deterioration of air quality in Naples, especially on days with atmospheric stagnant conditions.

An emission rate five times higher than the current one may pose a hazard even for the human health. In Figure 1 we reported some simulation results of Granieri et al., [2013]. Simulations of typical CO_2 concentrations at 1.5 m of height (typical breathing level for humans) suggest that CO_2 concentration reaches

higher values during nocturnal hours (from 2:00 to 7:00), when atmosphere is very stable. The level goes down up to a few hundred ppm at a few km from the source. During daytime, with unstable conditions and more intense wind circulation, gas is diluted and typical concentration values range from a few hundreds ppm near the source up to a few tens ppm (above the normal atmospheric background) at 3-4 km from the source (for further details see Granieri et al., [2013]).
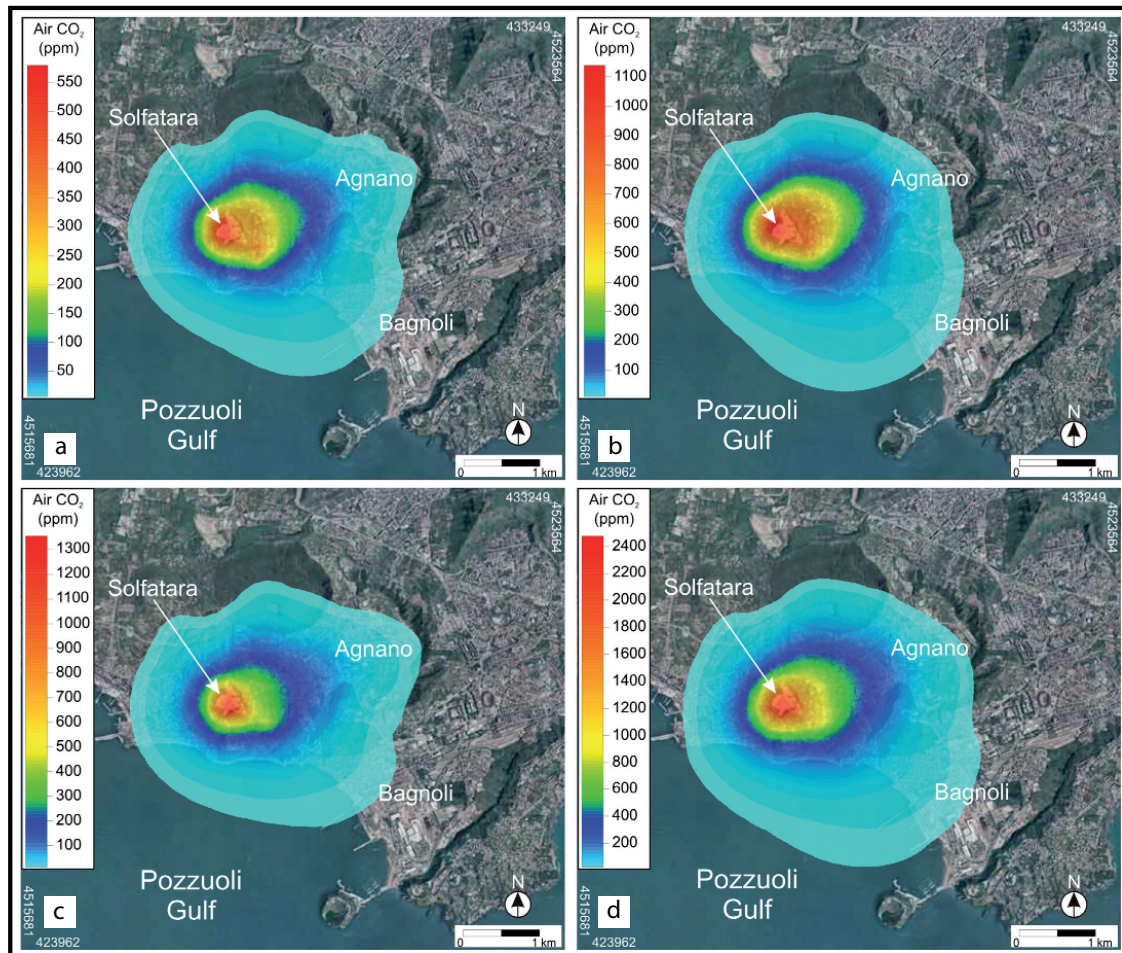

Figure 1. Average CO_2 concentration maps at 1.5 m above ground level assuming the mean CO_2 emission rate of 1067 ton/d (figures a and b on the top) and the maximum CO_2 emission rate measured in the 1998-2008 period of 1800 ton/d (figures c and d on the bottom), over a whole diurnal cycle (figures a and c) and over nocturnal periods (figures b and d). Modified after Granieri et al. [2013].

References

Arastoopour H., Lin S., Weil S., (1982). *Analysis of vertical pneumatic conveying of solids using multiphase flow models*. AIChE J., 28, 467-473.

Britter R., McQuaid J., (1988). *Workbook on the dispersion of dense gases*. Tech. rep., HSE Contract Research Report No. 17/1988, Trinity Road Bootle, Merseyside L20 3QY, UK.

Byun D., (1990). *On the analytical solutions of flux-profile relationships for the atmospheric surface layer*. J. Appl. Met., 29, 652-657.

Byun D., Chin J., (1999). *Science Algorithms of the EPA Model-3 Community Multiscale Air Quality (CMAQ) Modeling System*. ed. Byun D.W. and Chin J.K.S. Volume III, EPA-600/R-99-030, Washington, DC.

Byun D., Schere K., (2006). *Review of the governing equations, computational algorithms, and other components of the Models-3 Community Multiscale Air Quality (CMAQ) modeling system*. Applied Mechanics Reviews, 59, 51-77.

Cerminara M., Esposti Ongaro T., Berselli L.C., (2015). *ASHEE: a compressible, Equilibrium-Eulerian model for volcanic ash plumes*. Geosci. Model Dev. Discuss., 8, 8895-8979. doi:10.5194/gmdd-8-8895-2015.

Chiodini G., Baldini A., Barberi F., Carapezza M., Cardellini C., Frondini F., Granieri D., Ranaldi M., (2007). *Carbon dioxide degassing at Latera caldera (Italy): Evidence of geothermal reservoir and evaluation of its potential energy*. J. Geophys. Res., 112 (B12204).

Chiodini G., Frondini F., Cardellini C., Granieri D., Marini L., Ventura G., (2001). *CO_2 degassing and energy release at Solfatara volcano, Campi Flegrei, Italy*. J. Geophys. Res., 106, 16213-16221.

Chiodini G., Granieri D., Avino R., Caliro S., Costa A., Minopoli C., Vilardo G., (2010). *Non-volcanic CO_2 Earth degassing: The case of Mefite di Ansanto (Southern Apennines), Italy*. Geophys. Res. Lett., 37 (L11303).

Chiodini G., Granieri D., Avino R., Caliro S., Costa A., Werner C., (2005). *Carbon dioxide diffuse degassing: implications on the energetic state of a volcanic hydrothermal systems*. J. Geophys. Res. 110 (B08204).

Cortis A., Oldenburg C., (2009). *Short-range atmospheric dispersion of carbon dioxide*. Boundary-Layer Meteorology, 133 (1), 17-34.

Costa A., Macedonio G., Folch A., (2006). *A three-dimensional Eulerian model for transport and deposition of volcanic ashes*. Earth Planet. Sci. Lett., 241, 3/4, 634-647.

Costa A., Chiodini G., Granieri D., Folch A., Hankin R., Caliro S., Avino R., Cardellini C., (2008). *A shallow layer model for heavy gas dispersion from natural sources: application and hazard assessment at Caldara di Manziana, Italy*. Geochem. Geophys. Geosyst., 9 (Q03002).

Costa A., Folch A., Macedonio G., (2013). *Density-driven transport in the umbrella region of explosive eruptions: effects on tephra dispersion models*. Geophys. Res. Lett., 40, 1-5.

Costa A., Macedonio G., Chiodini G., (2005). *Numerical model of gas dispersion emitted from volcanic sources*. Annals of Geophysics, 48, 805-815.

Douglas S., Kessler R., (1990). *User's Manual for the Diagnostic Wind Model*. ed. Carr L. Volume III, EPA-450/4-90-007C, San Rafael, CA.

Dutton J., Fichtl G., (1969). *Approximate equations of motion for gases and liquids*. J. Atm. Sci. 26, 241-254.

Dyer A., (1974). *A review of flux-profile relationships*. Bound.-Lay. Meteorol., 7, 363-372.

Ewing R., Wang H., (2001). *A summary of numerical methods for time-dependent advection-dominated partial differential equations*. J. Comput. Appl. Math., 128, 423-445.

Folch A., Costa A., Hankin R., (2009). *Twodee-2: A shallow layer model for dense gas dispersion on complex topography*. Comput. Geosci., 35 (3), 667-674.

Ganser G., (1993). *A rational approach to drag prediction of spherical and non-spherical particles*. Powder Technol., 77, 143-152.

Golder D., (1972). *Relations among stability parameters in the surface layer*. Bound.-Lay. Meteorol., 3, 47-58.

Granieri D., Costa A., Macedonio G., Chiodini G., Bisson M., (2013). *Carbon dioxide in the city of Naples: contribution and effects of the volcanic source*. J. Volcanol. Geotherm. Res., 260, 52-62.

Hankin R., Britter R., (1999). *TWODEE: the Health and Safety Laboratory's shallow layer model for heavy gas dispersion. Part 1. Mathematical basis and physical assumptions*. J. Hazard. Mater., A66, 211-226.

Hindmarsh A., Gresho P., Griffiths D., (1984). *The stability of explicit Euler integration for certain finite difference approximations of the multidimensional advection-diffusion equation*. Int. J. Numer. Methods Fluids, 4, 853-897.

Jacobson M., (1999). *Fundamentals of atmospheric modelling*. 1st Edition. Cambridge University Press, New York.

Jacobson M., Lu R., Turco R., Toon O., (1996). *Development and application of a new air pollution modeling system. Part I: gas-phase simulations*. Atmos. Environ., 30, 1939-1963.

Macedonio G., Costa A., (2002). *Finite Element modeling of gas dispersion in the atmosphere*. In: Buccianti A., Marini L., Ottonello G., Vaselli O. (Eds.) *Proceedings of the Arezzo Seminar in Fluids Geochemistry*. Pacini Editore, Ospedaletto (Pisa) - Italy, pp. 147-159.

Monin A., Yaglom A., (1979). *Statistical Fluid Mechanics: Mechanics of Turbulence*. The MIT Press, USA,

volumes 1 and 2.

Park S., Kim C., (1999). *A numerical model for the simulation of SO$_2$ concentrations in the Kyongin region, Korea*. Atmos. Environ., 33, 3119-3132.

Pielke R., Cotton W., Walko R., Tremback C., Nicholls M., Moran M., Wesley D., Lee T., Copeland J., (1992). *A comprehensive meteorological modeling system-RAMS*. Meteor. Atmos. Phys., 49, 69-91.

Prabha T., Mursch-Radlgruber E., (1999a). *Investigation of air pollution distribution in Linz: case studies to evaluate a K-type diffusion model coupled with a mass-consistent wind model*. Atmos. Environ., 33, 4067-4080.

Prabha T., Mursch-Radlgruber E., (1999b). *Modeling of diffusion in a wide Alpine Valley*. Theor. Appl. Climatol., 64, 93-103.

Sagaut P., (2006). *Large Eddy Simulation for Incompressible Flows: an Introduction*. Springer Science & Business Media, Germany, pp 558.

Smagorinsky J., (1963). *General circulation experiments with the primitive equations, part I: the basic experiment*. Mon. Weather Rev., 91 (3), 99-164.

Sweeby P., (1984). *High resolution schemes using flux limiter for hyperbolic conservation laws*. SIAM J. Numer. Anal., 21 (5), 995-1011.

Toon O., Turco R., Westphal D., Malone R., Liu M., 1988. *A multidimensional model for aerosols: Description of computer analogs*. J. Atm. Sci., 45, 2123-2143.

Wang Y., Hutter K., (2001). *Comparisons of numerical methods with respect to convectively dominated problems*. Int. J. Numer. Methods Fluids, 37, 721-745.

Wilson L., Huang T., (1979). *The influence of shape on the atmospheric settling velocity of volcanic ash particles*. Earth Planet. Sci. Lett., 44, 311-324.

Appendices

Appendix A. Program setup

Here we will describe how to install *DISGAS* model and the code structure.

Installation

• On a Unix/Linux/Mac OS X Operating System:

1. Uncompress and untar file `disgas-version.tar.gz` with the command:

```
 tar xzvf disgas-version.tar.gz
```

This will generate directory `disgas-version` and the directory sub-tree in accord to the folder structure described in the Figure A.1.

2. Configure, compile, and install *DISGAS* and the utilities (`presfc, diagno, grd2lw`, and `gasview`). Enter the directory `disgas-version`, then issue the commands:

```
configure
make install
```

After the command "`make install`", the generated executable files `diagno, presfc, disgas, grd2lw`, and `gasview` will be installed in the directory `$HOME/bin` (the directory bin under the home directory will be automatically created if needed). It is important that the directory `$HOME/bin` is in the PATH of your shell. After compilation you may issue the command make clean to remove unneeded files.
Please note that the utility `gaswiev` is compiled and installed only if you have the pgplot library installed on your computer (see: http://www.astro.caltech.edu/~tjp/pgplot/)

Folder structure

Figure A.1 shows the folder structure of *DISGAS*. When untarred, disgas-version.tar.gz generates a directory tree as shown in Figure A.1. The directory `disgas-version/src` contains the source code of *DISGAS* in fortran-90. The source codes of the provided utilities are in the folder `disgas-version/utils`, each in the corresponding subdirectories (e.g., `diagno, grd2lw` and `gasview`).
The directory `disgas-version/runs` contains two subdirectories "example1" and "example2" with examples of applications of *DISGAS* that can be used to test the installation.
To simplify the management of the files, the example1 and example2 directories contain directory `infiles` with the input files, and an initially empty directory `outfiles` where the output files will be created and stored. The latter directory can be cleaned by issuing the command "`make clean`" inside the "example1" or "example2" directory.

Finally, the doc directory contains the manual of *DISGAS* and of `gasview` in pdf format.

Program run

Both `diagno` and `disgas` can be launched typing:

```
diagno
disgas problemname.inp
```

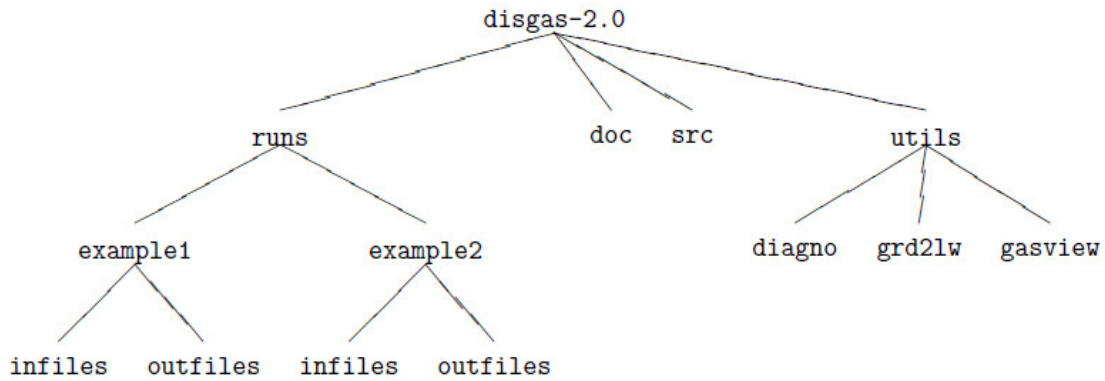where `problemname.inp` is the name of the control input file (see Appendix B).

Figure A.1. Directory tree of `disgas-version`.

The file `diagno.inp` denotes the *DIAGNO* input file. Beside `diagno.inp` for running `diagno`, it is necessary to furnish, in the same directory, a file surface.dat containing data from wind stations and a topography file `topography.grd`.

The file surface.dat can be generated using the utility `presfc` that needs the input file `presfc.dat` (for further descriptions see Douglas and Kessler, 1990).
The *DIAGNO* computational domain has to be equal or larger than the *DISGAS* computational domain.

NOTE: To create a new run simply create a new folder in the directory runs (with subfolders `infiles` and `outfiles`), copy the input files and the problemname.inp.

Appendix B. The disgas input and output files

DISGAS needs of the following input files:

• File: `problemname.inp` - Control file that defines a run. Mandatory.

• File: `topography.grd` - Regional ground elevation file. Optional.

• File: `source.dat` - Source term (gas fluxes) file. Mandatory.

• File: `winds.dat` - Meteorological data file. Optional.

• File: `diagno.out` - `diagno` meteorological data file. Optional.

NOTE: File names used in this manual are given just for illustrative purposes.

Names and paths of input files are absolutely free and can be defined by the user in the control file `problemname.inp`.

The control file `problemname.inp`

The *DISGAS* control file is passed to the program as a call argument. This file is made up with a set of blocks that define all the computational and physical parameters needed by the dispersion model (Table B.1 shows an example of control file). Parameters within a block are listed one per record, in arbitrary order, and can optionally be followed by one or more blank spaces and a comment. A detailed description of each record is given below. Real numbers can be expressed following the fortran notation (e.g.: `12e7` = 12×10^7).

BLOCK `TIME`

• `YEAR`: Initial year.

• `MONTH`: Initial month (1-12).

• `DAY`: Initial day (1-31).

• `HOUR`: Initial hour (0-23).

• `MINUTE`: Initial minute (0-59).

• `SIMULATION_INTERVAL_(SEC)`: Simulation time duration (in seconds).

• `RESTART_RUN`: Select whether the simulation loads the restart file (NO or YES).

• `RESET_TIME`: In case of start from the *restart* file, select whether the internal variable time is reset to zero or continues from last dump.

NOTE: The parameters `YEAR, MONTH, DAY, HOUR`, and `MINUTE` are used to check the consistence with both the restart and the meteorological files.

BLOCK `GRID`

• `NX`: Number of grid cells along the *x*-direction.

• `NY`: Number of grid cells along the *y*-direction.

• `NZ`: Number of layers along the *z*-direction.

• `Z_LAYERS_(M)`: List of the quotes of the layers in the *z*-direction. They must be exactly `NZ`.

• `DX_(M)`: Grid spacing along the *x*-direction (in meters).

• `DY_(M)`: Grid spacing along the *y*-direction (in meters).

• `X_ORIGIN_(UTM_M)`: *x*-coordinate of the grid bottom left corner (UTM coordinates in meters).

• `Y_ORIGIN_(UTM_M)`: *y*-coordinate of the grid bottom left corner (UTM coordinates in meters).

BLOCK `PROPERTIES`

• `DISPERSION_TYPE`: Type of material dispersed. Possibilities are `GAS` (no settling velocity) or `PARTICLES` (settling velocity is calculated in accord to the particle properties specified below).

• `PARTICLE_DIAMETER`: Diameter of particles in meters (only used if `DISPERSION_TYPE = PARTICLES`).

• `PARTICLE_DENSITY`: Density of particles in kg/m^3 (only used if `DISPERSION_TYPE = PARTICLES`).

• `PARTICLE_SHAPE_PARAMETER`: Particle shape parameter to be used to calculated particle settling velocity, i.e. particle sphericity or particle aspect ratio. In the case of Ganser [1993] model it corresponds to particle sphericity (only used if `DISPERSION_TYPE = PARTICLES`).

• `PARTICLE_MODEL_VSET`: Parameterization used for calculating particle settling velocity. Possibilities are: `ARASTOOPOUR` for the Arastoopour et al. [1982] model (spherical particles), `GANSER` for Ganser [1993] model (irregular particles). Only used if `DISPERSION_TYPE = PARTICLES`.

BLOCK `TOPOGRAPHY`

• `EXTRACT_TOPOGRAPHY_FROM_FILE`: Flag indicating whether the topography of the computational domain is extracted from a ground elevation file or not. Possibilities are `YES/NO`. If `YES`, topography is extracted from the file defined in the FILES BLOCK. If `NO`, a flat terrain is assumed with slopes in the x and y directions defined by the parameters `X_SLOPE_(DEG)` and `Y_SLOPE_(DEG)`.

• `Z_ORIGIN_(M)`: Quote of the bottom-left corner of the topography.
This is used to generate a flat topography when the flag `EXTRACT_TOPOGRAPHY_FROM_FILE` is set to `NO`.

• `X_SLOPE_(DEG):` Slope of the flat topography in the *x* direction (in degrees). This parameter is used to generate a flat topography when the flag `EXTRACT_TOPOGRAPHY_FROM_FILE` is set to `NO`.

• `Y_SLOPE_(DEG):` Slope of the flat topography in the *y* direction (in degrees). This parameter is used to generate a flat topography when the flag `EXTRACT_TOPOGRAPHY_FROM_FILE` is set to `NO`.

BLOCK `METEO`

• `WIND_MODEL`: Flag indicating the wind model.
Possibilities are `SIMILARITY` or `DIAGNO`. If `SIMILARITY` is set, the wind values are read from a file defined in the BLOCK `FILES` containing the time-dependent wind observations from a station, and the vertical profile is evaluated according to the similarity theory and interpolated according the roughness matrix. If `DIAGNO` option is used, meteorological data (variable in both space and time) are read from a *DIAGNO* output file defined in the block `FILES`.

Note that in the later case the program *DIAGNO* must necessarily run before the program *DISGAS* and parameters read from the file `winds.dat` are used as input for *DIAGNO* (see Sec. 1.1).

• `HORIZONTAL_TURB_MODEL`: Flag indicating the model for the horizontal diffusion coefficients. This may be equal to `CONSTANT` for constant horizontal diffusion coefficients or `SMAGORINSKY` for horizontal diffusion coefficient from the Smagorinsky parameterization (see eq. (11)).

• `VERTICAL_TURB_MODEL`: Flag indicating the model for the vertical diffusion coefficient. This may be set to 0 (zero) for constant vertical diffusion coefficient or 1 (one) for vertical diffusion coefficient from Atmospheric Boundary Level (ABL) model (see eq. (7)).

• `ROUGHNESS_MODEL`: Flag indicating the soil roughness length model. Possibilities are `UNIFORM` (set with the parameter `ROUGHNESS_LENGTH`) or `MATRIX` for non-uniform roughness length, provided with a file (see BLOCK `FILES`).

• `ROUGHNESS_LENGTH`: Terrain roughness height z_0 (in m). The surface roughness height is related to the vertical wind profile. Values range from about 10^{-5} m over an iced surface, 0.005 m over naked soil, 0.05 m over soil covered by tall grass, or up to 1 m or more over forest or urban areas.

• `DIFF_COEFF_HORIZONTAL`: Horizontal diffusion coefficient in m^2/s, used if `HORIZONTAL_TURB_MODEL` is equal to `CONSTANT`.

• `DIFF_COEFF_VERTICAL`: Vertical diffusion coefficient in m^2/s. (used if `VERTICAL_TURB_MODEL` is equal to `CONSTANT`).

BLOCK `FILES`

• `TOPOGRAPHY_FILE_PATH`: Name (including relative or absolute path) of the topography file. See Appendix B. for file format details. This record is read only when `EXTRACT_TOPOGRAPHY_FROM_FILE` = `YES`.

• `ROUGHNESS_FILE_PATH`: Name (including relative or absolute path) of the file containing the roughness. See Sec. Appendix B for file format details. This record is read only when `ROUGHNESS_MODEL` = `MATRIX`.

• `RESTART_FILE_PATH`: Name (including relative or absolute path) of the restart file. This file is automatically generated and overwritten by *DISGAS* at every print time. It may be later used to restart the simulation from the last dump. See Appendix B.

• `SOURCE_FILE_PATH`: Name (including relative or absolute path) of the source file. See Appendix B for file format details.

• `WIND_FILE_PATH`: Name (including relative or absolute path) of the wind data file. See section Appendix B for file format details.

• `DIAGNO_FILE_PATH`: Name (including relative or absolute path) of the *DIAGNO* output file. This is a binary unformatted file created by *DIAGNO*. Only used when `WIND_MODEL` = `DIAGNO`.

• `OUTPUT_DIRECTORY`: Name (including relative or absolute path) of the folder where *DISGAS* output files are written.

BLOCK `OUTPUT`

• `OUTPUT_GRD_TYPE`: Format for the GRD file. Possibilities are ASCII or BINARY (in the current version ASCII only).

• OUTPUT_INTERVAL_(SEC): Time interval to output results (in s).

• OUTPUT_U_VELOCITY: Flag indicating whether the wind velocity in the x direction (in m/s) has to be printed or not. Possibilities are YES or NO. If YES, a GRD-format file for each height is dumped in the folder defined by the OUTPUT_DIRECTORY recorded every OUTPUT_INTERVAL_(SEC) seconds.

• OUTPUT_V_VELOCITY: Flag indicating whether the wind velocity in the y direction (in m/s) has to be printed or not. Possibilities are YES or NO. If YES, a GRD-format file for each height is dumped in the folder defined by the OUTPUT_DIRECTORY recorded every OUTPUT_INTERVAL_(SEC) seconds.

• OUTPUT_CONCENTRATION: Flag indicating whether the gas concentration (in ppm) has to be printed or not. Possibilities are YES or NO. If YES, a GRD-format file for each height is dumped in the folder defined by the OUTPUT_DIRECTORY record every OUTPUT_INTERVAL_(SEC) seconds.

• LOG_VERBOSITY_LEVEL: Flag indicating verbosity level of the log file. Possibilities are 0, for low verbosity log file, and 1, for high verbosity log file.

Table B.1. Sample of the input control file `problemname.inp`.

DISGAS INPUT FILE

```
TIME
YEAR      = 2007
MONTH     = 2
DAY       = 24
HOUR      = 12
MINUTE = 00
SIMULATION_INTERVAL_(SEC) = 120
RESTART_RUN      = NO
RESET_TIME      = NO

GRID
NX        = 40
NY        = 32
NZ        = 11
Z_LAYERS_(M) = 0. 1. 4. 8. 12. 16. 24. 32. 40. 50. 60.
DX_(M) = 50.
DY_(M) = 50.
X_ORIGIN_(UTM_M)= 427000
Y_ORIGIN_(UTM_M) = 4520000
PROPERTIES
DISPERSION_TYPE = PARTICLES      (GAS/PARTICLES)
PARTICLE_DIAMETER = 2.0e-6       (PARTICLE DIAMETER IN M)
PARTICLE_DENSITY = 2610.         (PARTICLE DENSITY KG/M3)
PARTICLE_SHAPE_PARAMETER=0.779   (PARTICLE SPHERICITY)
PARTICLE_MODEL_VSET = GANSER     (ARASTOOPOUR/GANSER/WILSON_HUANG)
```

Table B.1 (continues)

```
TOPOGRAPHY

Z_ORIGIN_(M) = 0.0
X_SLOPE_(DEG) = 0.0
Y_SLOPE_(DEG) = 0.0
METEO
WIND_MODEL                  = DIAGNO          (SIMILARITY/DIAGNO)
HORIZONTAL_TURB_MODEL       = SMAGORINSKY      (CONSTANT/SMAGORINSKY)
VERTICAL_TURB_MODEL         = SIMILARITY       (CONSTANT/SIMILARITY)
ROUGHNESS_MODEL             = UNIFORM          (UNIFORM/MATRIX)
ROUGHNESS_LENGTH            = 0.18             (For ROUGHNESS_MODEL=UNIFORM)
DIFF_COEFF_HORIZONTAL       = 10.              (For turbulence model=CONSTANT)
DIFF_COEFF_VERTICAL         = 10.              (For turbulence model=CONSTANT)

FILES                       (Relative or absolute file paths)
TOPOGRAPHY_FILE_PATH        = ./infiles/topography.grd
ROUGHNESS_FILE_PATH         = ./infiles/roughness.grd
RESTART_FILE_PATH           = ./infiles/restart.dat
SOURCE_FILE_PATH            = ./infiles/source.dat
WIND_FILE_PATH              = ./infiles/winds.dat
DIAGNO_FILE_PATH            = ./infiles/diagno.out
OUTPUT_DIRECTORY            = ./outfiles/
```

(continues)

```
OUTPUT

OUTPUT_GRD_TYPE            =       ASCII     (ASCII/BINARY)
OUTPUT_INTERVAL_(SEC) = 3600                 Seconds between prints
OUTPUT_U_VELOCITY = NO                       (YES/NO)
OUTPUT_V_VELOCITY = NO                       (YES/NO)
OUTPUT_W_VELOCITY = NO                       (YES/NO)
OUTPUT_GROUND_LOAD= YES                      (YES/NO)
OUTPUT_CONCENTRATION = YES                   (YES/NO)
LOG_VERBOSITY_LEVEL = 0                      (0:low verbosity/1:high)
```

The topography file `topography.grd`

The topography file specifies ground elevation at a regional scale (i.e. in a region typically equal or 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 (grid spacing) can be different. The only necessary requirement is that the computational domain must lay within the bounds of the region where topography is specified. *DISGAS* reads the topography file and automatically interpolates elevations onto the nodes of the computational grid. In this version of *DISGAS*, the topography must be stored in the ASCII GRD format. The structure of a GRD format file is described in Appendix C (see also Table C.1).

The source file `source.dat`

The source file specifies the gas fluxes (mass flow rate expressed in kg/s) from different point sources. *DISGAS* reads this file and automatically locates the source onto the nearest node of the computational domain. In case multiple records are assigned the same node, the mass fluxes for that node are summed.
Each concentrated point source is listed in the source file with the following record structure:

• `X_s` : x-coordinate (UTM in m) of the source.

• `Y_s` : y-coordinate (UTM in m) of the source.

• `Z_s` : z-coordinate (in m) of the source, height from ground level.

• `PHI_s`: Flux associated to the point source, expressed in kg/s.

There is no limit to the number of point sources; point sources located outside the computational domain are automatically rejected.

The roughness length file `roughness.grd`

This file specifies the roughness-length in a region typically equal or larger than the computational domain. The roughness length (in m) must be specified on a structured grid using arbitrary (but constant) grid spacing (e.g., 5 m, 10 m, 100 m, etc.). Discretization along *x*- and *y*-directions (grid spacing) can be different. The only necessary requirement is that the computational domain must lay within the bounds of the region where roughness length is specified.

DISGAS reads this file and automatically interpolates the values onto the nodes of the computational grid. In this version of *DISGAS*, the roughness length must be stored in the ASCII GRD format. The structure of the roughness length file is the same as the `topography.grd` file, and is described in Appendix C (see also Table C.1).

The wind data file `winds.dat`

The wind data file contains meteorological data at different time slices. If the record `WIND_MODEL`, in the control input file, is `UNIFORM`, *DISGAS* reads this wind file and estimates the Atmospheric Surface Layer parameters. Otherwise, if the record `WIND_MODEL` is `DIAGNO`, the wind field is instead read by the *DIAGNO* output file (file `diagno.out`). The wind file format is described in Table B.2 (if code=`CUP`) and Table B.3 (if code = `SONIC`) and the meaning of the used symbols is the following:

• `X_UTM, Y_UTM` : UTM coordinates of the wind station (specified in the first record of the file).

• `Z_ref` : Elevation above ground of the wind station (specified in the first record of the file).

• `iyr` : Measurement year.

• `imo` : Measurement month (1-12).

• `idy` : Measurement day (1-31).

• `ihr` : Measurement hour (0-23).

• `imi` : Measurement minute (0-59).

• `code` : Flag that indicates the type of anemometer. Possibilities are `CUP` or `SONIC`. When option `SONIC` is used the Monin-Obukhov length `L` is directly read from the file, whereas for `CUP` option is estimated from the air temperature difference.

• `t_1` : Time slice starting time (in s after initial time). Initial time for measurements is assumed to be at day `idy`, hour `ihr`, and minute `imi`.

• `t_2` : Time slice ending time (in s after initial time). Initial time for measurements is assumed to be at day `idy`, hour `ihr`, and minute `imi`. A data time slice spans during the interval [`t_1`, `t_2`].

• `wx` : x-component of wind speed (in m/s).

• `wy` : y-component of wind speed (in m/s).

• `T_z0` : Temperature (in °C) at the ground level.

• `T_zref` : Temperature (in °C), measured at Z ref.

• `p` : Atmospheric pressure (in hPa).

• `ustar` : Friction velocity (in m/s).

• `L` : Monin-Obukhov length (in m).

<u>NOTE</u>: The records `iyr` to `imi` are used just to check consistency of meteorological data files with the input control file. Initial time must coincide in both cases.

Table B.2. Format of the wind file `winds.dat` with `code=CUP`.

X_UTM	Y_UTM	Zref				
iyr	imo	idy	ihr	imi	code	# code=CUP
t_1	t_2	wx	wy	T_z0	T_zref	p
...	...	...	...	...	...	...
t_(nt-1) t_nt		wx	wy	T_z0	T_zref	p

Table B.3. Format of the wind file `winds.dat` with `code=SONIC`.

X_UTM	Y_UTM	Zref				
iyr	imo	idy	ihr	imi	code	# code=SONIC
t_1	t_2	wx	wy	T_zref	ustar	L
...	...	...	...	...	...	...
t_(nt-1)						
t_nt		Wx	wy	T_zref	ustar	L

The file `diagno.out`

The 3-D time-dependent wind field produced by the utility *DIAGNO* is stored in the unformatted file `diagno.out`. The format of this file is defined by subroutine `outfil.f` under the directory `utils/diagno`. This subroutine was modified with respect to the original contained in the utility `diagno` written by Douglas and Kessler (1990). The format of file `diagno.out` is not described here since it cannot be edited.

The DISGAS output files

At each user-specified time (see `OUTPUT_INTERVAL_(SEC)` record in the control file) *DISGAS* generates 2D matrices written in GRD-format for the gas concentration, each corresponding to a different Z-layer (e.g. `c_00i_00000j.grd` where i denotes the layer and j the time slice). Moreover, `disgas` can produce GRD files containing the x and y components of the wind velocity at each Z-layer (e.g. `u_00i_00000j.grd` where i denotes the layer and j the time slice).

The output files are stored in the directory defined by the `OUTPUT_DIRECTORY` record of the *DISGAS* control file. Files in GRD format can be read directly by several plotting programs like the commercial software *Surfer*[c]. Alternatively, the user may also generate its own plots using functions from several free packages (e.g. *pgplot*) like the *Gasview* utility freely available with the package.

The `problemname.log` *file*

The name of this file is passed with the program call argument. It contains information concerning the run (summary of input data, run time error messages, CPU time, etc.).

The restart file `restart.dat`

The restart file is automatically produced by *DISGAS* and is overwritten at each print step. This file contains all the information needed to restart the simulation starting from the time of last dump. This may be useful for long runs that need to be interrupted and then restart, or for modifying some parameters (e.g., the wind field or the source file) during runtime. The restart file is written in binary and cannot be directly edited. The format of the restart file is defined in the subroutines `wrirst.f90` (which generates the restart file) and `setvar.f90` (which reads the restart file), both located in the directory `src`.

Appendix C. Format of the standard GRD files

The structure of a GRD format file is described in Table C1 and the meaning of the used symbols is the following:

• `NX`: Number of grid points along x-direction.

• `NY`: Number of grid points along y-direction.

• `XO`: x-coordinate (UTM in m) of the grid bottom left corner.

• `XF`: x-coordinate (UTM in m) of the grid top right corner point.

• `YO`: y-coordinate (UTM in m) of the grid bottom left corner point.

• `YF`: y-coordinate (UTM in m) of the grid top right corner point.

• `VAL`: Value at each grid point. It consists of an array of NX× NY values stored starting from the bottom-left corner and moving towards right then up towards the top of the domain.

Table C.1. Format of a GRD file.

```
NX   NY

XO   XF

YO   YF

MIN(VAL)    MAX(VAL)

VAL(i,1)  ... ... i=1:NX

...  ...

VAL(i,j)  ... ... i=1:NX

...  ...  ...

VAL(i,NY) ... ... i=1:NX
```

http://www.ingv.it