ORNL/TM-2022/2602

Initial overview, description, and assessment of the DELFIC atmospheric transport model

Matthew J. Krupcale

September 18, 2023

**Approved for public release.
Distribution is unlimited.**

OAK RIDGE
National Laboratory

ORNL IS MANAGED BY UT-BATTELLE LLC FOR THE US DEPARTMENT OF ENERGY

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL/TM-2022/2602

Nuclear Nonproliferation Division

Initial overview, description, and assessment of the DELFIC atmospheric transport model

Matthew J. Krupcale

September 18, 2023

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831-6283
managed by
UT-Battelle LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF LISTINGS

ABBREVIATIONS

ABL	atmospheric boundary layer
ASL	atmospheric surface layer
ATM	atmospheric transport model
CFD	computational fluid dynamics
DASA	Defense Atomic Support Agency
DELFIC	Defense Land Fallout Interpretive Code
DNA	Defense Nuclear Agency
DNS	direct numerical simulation
DPM	discrete parcel method
DTM	diffusive transport module
DTRA	Defense Threat Reduction Agency
FDM	finite difference method
GZ	ground zero
ICRM	initialization and cloud rise module
ODE	ordinary differential equation
OPM	output processor module
PDE	partial differential equation
PDF	probability density function
QBMM	quadrature-based moment method
RANS	Reynolds-averaged Navier-Stokes
RMS	root mean squared
TFM	two-fluid model
TKE	turbulent kinetic energy

ACKNOWLEDGMENTS

This work was funded primarily by the Countering Weapons of Mass Destruction (CWMD) Office within the Department of Homeland Security (DHS) under a National Technical Nuclear Forensics Center (NTNFC) postdoctoral fellowship and by the Department of Energy's National Nuclear Security Administration, Office of Defense Nuclear Nonproliferation Research and Development. I would especially like to thank Dr. Vincent J. Jodoin and Dr. Pablo Moresco for providing valuable feedback and mentorship.

ABSTRACT

We present an overview and description of the atmospheric transport model (ATM) component of the Defense Land Fallout Interpretive Code (DELFIC) and its revision history. In particular, we discuss the hybrid Eulerian–Lagrangian model of the DELFIC diffusive transport module (DTM), its assumptions and constraints, and assess its capability for modeling the transport, dispersion, and deposition of nuclear fallout debris. This hybrid model is an efficient transport model for the wide range of particle sizes found in nuclear fallout debris. Furthermore, we validate the 1979 version of DELFIC using the 1979 test case input and output presented in its documentation. Except for some rounding differences and exposure rate differences from the output processor module (OPM), the DTM output is in very good agreement with the 1979 results. Overall, the DELFIC DTM provides a reasonable and rapid approximation to the transport and deposition of local nuclear fallout debris.

1. INTRODUCTION AND BACKGROUND

In the mid 1960s, the Defense Atomic Support Agency (DASA)[*] supported the creation of a local nuclear fallout prediction code called DELFIC [2, 3]. It was intended for research in local nuclear fallout prediction as well as to serve as a standard to judge predictions by less-capable production codes. Thus, DELFIC attempts to include as much physics of the fallout transport and radioactivity modeling as possible, and it has undergone revisions in the 1970s and beyond to improve its accuracy and physics models. These models include mass and fallout cloud geometry initialization, cloud rise, atmospheric transport and deposition or settling, and radioactive transmutation and exposure. The focus of this work, however, is DELFIC's atmospheric transport, dispersion, and deposition models.

Given a nuclear fallout cloud mass and geometry of a specified particulate size distribution, the objective of the ATM is to simulate the atmospheric trajectory, dispersion, and ground deposition of the cloud particulates. The DELFIC ATM is a deterministic transport model that discretizes the cloud into independent parcels. Using the specified meteorological conditions, the parcels are transported and dispersed in the horizontal dimensions according to a Gaussian bivariate concentration profile with increasing variance. Thus, DELFIC falls within the class of Gaussian puff models, albeit with some additional capabilities tailored for nuclear fallout modeling. Additionally, since the cloud advection and diffusion are modeled using both Lagrangian and Eulerian methods, DELFIC is somewhat of a hybrid Eulerian–Lagrangian[†] ATM code. In particular, DELFIC shares aspects of both a Lagrangian discrete parcel method (DPM) [5] and an Eulerian advection-diffusion code for modeling the dispersed fallout cloud particles.

The horizontal and vertical dimensions are treated somewhat differently because advection dominates the horizontal motion, whereas turbulent diffusion plays a more significant role in the vertical dimension [8, 9]. When vertical diffusion is neglected, DELFIC uses the original, simple advective transport model [10] from 1967, which when combined with a simple gravitational settling model is called the "advection plus settling" mode [8, 9]. A more sophisticated transport model called the diffusive transport module (DTM) was incorporated into DELFIC in 1971 and accounts for anisotropic turbulence and vertical advection and diffusion. Both modes are technically capable of using spatially and temporally varying fields, although it is

[*] DASA preceded both the Defense Nuclear Agency (DNA) and what is now known as the Defense Threat Reduction Agency (DTRA).

[†] This is not to be confused with a gas–solid multiphase code in which, e.g., the dispersed solid phase cloud particle behavior is modeled using a Lagrangian method while the continuum carrier gas or fluid phase is modeled using Eulerian methods. Such codes are said to employ a "Eulerian–Lagrangian" (or "Lagrangian–Eulerian") method [4, 5, 6, 7]. Since DELFIC does not actually solve for the continuum carrier flow fields, it is not "Eulerian–Lagrangian" in that sense.

unclear how well such variations scale computationally in practice. The DTM was then simplified in the 1979 and later versions by removing the vertical diffusion solver and relying on the original Lagrangian advective transport model with gravitational settling and Gaussian dispersion in the horizontal dimension [2, 3].

Like many ATM codes, the meteorological data describing the continuum air fluid flow are treated as an input for the simulation. At a minimum, an ATM requires the fluid velocity $\mathbf{u}$, but typically it also requires the carrier fluid density ρ, pressure p, and temperature T or their means[*]. These meteorological data can be constructed using many potential methods ranging from simple atmospheric sounding to complete reanalysis data sets constructed from sophisticated weather models and data assimilation. Such weather models are often Eulerian computational fluid dynamics (CFD) solvers with empirical models and boundary conditions suitable for modeling the Earth's atmospheric meteorology. DELFIC then uses this meteorological data with one-way coupling—i.e., the meteorological fields drive the fallout and not vice versa—to model the transport and dispersion of fallout cloud particles within this fluid flow.

Section 2 discusses general particle dynamics and steady-state kinematic solutions that serve as the basis for the DELFIC "advection plus settling" transport described in Section 3.4. Both this Lagrangian particle transport and the Eulerian dispersion model based on the Gaussian puff model are discussed in detail in Section 3. The resulting deposited mass concentration on the ground of the airborne parcels is examined by comparing analytical deposition models to the DELFIC geometric approximation in Section 4. Then we present in Section 5 the 1979 version implementation of the DTM [3] as well as a description of its subroutines. The results of running the 1979 version of the code with the 1979 example test problem are then presented and compared with the original 1979 output in Section 6. Finally, Section 7 makes some conclusions about the DTM.

[*] In the case of a deterministic model, the mean value is equivalent to the variable itself.

2. PARTICLE DYNAMICS AND KINEMATICS

The Lagrangian discrete parcel method is based on the dynamics of discrete particles, the solution of which is approximately described by steady-state kinematics. Section 2.1 discusses the potential body, surface, and collision forces acting on rigid, spherical particles and their relevance for the transport of fallout debris particles. We then derive the steady-state kinematic solution of the discrete particles with various initial and boundary conditions and corresponding constraints in Section 2.2. This kinematic solution serves as the basis of the DELFIC "advection plus settling" transport described in Section 3.4.

2.1 DISCRETE PARTICLE DYNAMICS

In the Lagrangian description of the dispersed cloud particle phase, each particle or parcel is treated as a discrete particle. Thus, each physical particle in the dispersed phase modeled by the discrete particle or parcel has the same physical characteristics: mass, size, temperature, etc. In this case, the properties of each particle (or parcel) are characterized using a set of ordinary differential equations following the particle (or parcel) trajectory. In particular, the particle position (or center of mass) $\mathbf{x}_\mathrm{p}(t)$ satisfies the kinematic equation

$$\frac{\mathrm{d}\mathbf{x}_\mathrm{p}}{\mathrm{d}t} = \mathbf{v}_\mathrm{p}(t), \tag{1}$$

where $\mathbf{v}_\mathrm{p}(t)$ is the particle (or center of mass) velocity. The particle velocity then satisfies the momentum conservation equation written in the form of Newton's second law for particle momentum $\mathbf{p}_\mathrm{p}(t) = m_\mathrm{p}(t)\mathbf{v}_\mathrm{p}(t)$ [6, 7, 11]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(m_\mathrm{p}\mathbf{v}_\mathrm{p}\right) = \mathbf{F}_\mathrm{b} + \mathbf{F}_\mathrm{s} + \mathbf{F}_\mathrm{c}, \tag{2}$$

where $m_\mathrm{p}(t) = \rho_\mathrm{p}(t)V_\mathrm{p}(t)$ is the particle (or parcel) mass, and the right-hand side force terms are the body, surface, and collision forces.

If the particles (or parcels) undergo mass or energy transfer, then these must also be modeled [5, 6], and Section 2.1.1 discusses how to handle the dynamics of variable-mass particles. However, treating the particles as rigid spheres with diameter $d_\mathrm{p} = 2r_\mathrm{p}$, the mass m_p, density ρ_p, and volume V_p are constant in time [11], and the mass term can be pulled outside of the time derivative: $\mathrm{d}(m_\mathrm{p}\mathbf{v}_\mathrm{p})/\mathrm{d}t = m_\mathrm{p}\mathrm{d}\mathbf{v}_\mathrm{p}/\mathrm{d}t$. This may be a reasonable assumption if the fallout particles are sufficiently removed from their original physical formation process and are only undergoing physical transport processes. In DELFIC, debris particles which are formed during the cloud rise are then presumed to have constant mass in the ATM, allowing DELFIC to use Eq. 2 as demonstrated in Section 2.2.1. Sections 2.1.2 to 2.1.4 then discuss the potential body, surface, and collision forces acting on spherical particles and appearing on the right-hand side of Eq. 2.

2.1.1 Variable Mass

Equation 2 attempts to describe the particle dynamics in terms of particle momentum $\mathbf{p}_\mathrm{p}(t) = m_\mathrm{p}(t)\mathbf{v}_\mathrm{p}(t)$ with potentially variable mass $m_\mathrm{p}(t)$. However, Newton's second law Eq. 2 is technically only correct for constant-mass particles $m_\mathrm{p}(t) = m_\mathrm{p}$, for which the mass can be factored out of the particle dynamics equation. More formally, however, the correct momentum conservation equation for variable-mass systems is [12]

$$\begin{aligned}\frac{\mathrm{d}\mathbf{p}_\mathrm{p}}{\mathrm{d}t} = m_\mathrm{p}\frac{\mathrm{d}\mathbf{v}_\mathrm{p}}{\mathrm{d}t} + \mathbf{v}_\mathrm{p}\frac{\mathrm{d}m_\mathrm{p}}{\mathrm{d}t} &= \mathbf{F}_\mathrm{b} + \mathbf{F}_\mathrm{s} + \mathbf{F}_\mathrm{c} + \mathbf{v}'\frac{\mathrm{d}m_\mathrm{p}}{\mathrm{d}t} \\ \Rightarrow m_\mathrm{p}\frac{\mathrm{d}\mathbf{v}_\mathrm{p}}{\mathrm{d}t} &= \mathbf{F}_\mathrm{b} + \mathbf{F}_\mathrm{s} + \mathbf{F}_\mathrm{c} + \mathbf{v}_\mathrm{r}\frac{\mathrm{d}m_\mathrm{p}}{\mathrm{d}t},\end{aligned} \tag{3}$$

where $\mathbf{v}'$ is the initial velocity of the mass element $\mathrm{d}m_\mathrm{p}$ before capture or after emission, and $\mathbf{v}_\mathrm{r} = \mathbf{v}' - \mathbf{v}_\mathrm{p}$ is the velocity of the mass element $\mathrm{d}m_\mathrm{p}$ relative to m_p. Thus, only in frame of reference of the mass $\mathrm{d}m_\mathrm{p}$ (i.e., $\mathbf{v}' = 0$) is Eq. 2 correct for variable-mass systems. In the case that $\mathbf{v}_\mathrm{r} = 0 \Rightarrow \mathbf{v}' = \mathbf{v}_\mathrm{p}$, Eq. 3 reduces to the classical Newton's second law $\mathbf{F} = m_\mathrm{p}\mathbf{a}_\mathrm{p}$ despite the variable mass. This is the case for isotropic mass loss or gain in the frame of reference moving with the center of mass [13]. For simple, spherically symmetric fallout particles, this should be a reasonable assumption, but more generally, we can neglect the changes in particle mass when

$$v_\mathrm{r}\left|\frac{\mathrm{d}m_\mathrm{p}}{\mathrm{d}t}\right| \ll \left|\mathbf{F}_\mathrm{b} + \mathbf{F}_\mathrm{s} + \mathbf{F}_\mathrm{c}\right|. \tag{4}$$

Thus, if the fallout particles undergo anisotropic mass changes, the model must compare the relative magnitude of the external forces to the momentum change due to variable mass in order to use Eq. 2.

2.1.2 Body Forces

Typically, one only accounts for gravitational body forces [5, 7]:

$$\mathbf{F}_\mathrm{b} = \mathbf{F}_\mathrm{grav} = m_\mathrm{grav}\mathbf{g} = -m_\mathrm{grav}\nabla\Phi_\mathrm{grav}, \tag{5}$$

where

$$m_\mathrm{grav} = m_\mathrm{p} - \rho V_\mathrm{p} = \left(\rho_\mathrm{p} - \rho\right)V_\mathrm{p} = m_\mathrm{p}\left(1 - \frac{\rho}{\rho_\mathrm{p}}\right), \tag{6}$$

includes both gravitation and buoyancy forces by defining the effective gravitational mass in terms of carrier fluid density ρ, and we can define the gravitational potential as [14]

$$\Phi_\mathrm{grav} = -\mathbf{g}\cdot\mathbf{x}_\mathrm{p}, \tag{7}$$

with $\mathbf{g} \equiv -g\hat{\mathbf{r}}$ the gravitational acceleration vector pointing toward the Earth's polar center and $\mathbf{x}_\mathrm{p}$ the polar position of the particle. The buoyant force is effectively the result of including the pressure forces along the direction of gravity. That is, $\mathbf{F}_\mathrm{buoy} = F_\mathrm{buoy}\hat{\mathbf{r}} = -\int_{V_\mathrm{p}}\nabla\cdot p\delta_{ij}\mathrm{d}V\cdot\hat{\mathbf{r}}\hat{\mathbf{r}} = -\int_{\partial V_\mathrm{p}} p\delta_{ij}\cdot\hat{\mathbf{n}}\mathrm{d}S\cdot\hat{\mathbf{r}}\hat{\mathbf{r}}$ [5, 15], where p is the pressure. Technically, the buoyancy force is a surface force that could be included in Eq. 13 when using the point-force approach [15], but we include it here because the effect is to modify the gravitational body force. Then the surface forces in Section 2.1.3 when using the point-force approach should account for the remaining Cauchy stress tensor forces (i.e., due to the other pressure force directions and the deviatoric stress tensor*).

Other potential body forces that can be included where appropriate are electromagnetic, electrophoretic, thermophoretic, and fictitious forces due to a frame of reference with translational acceleration $\mathbf{a}(t)$ and rotation vector $\boldsymbol{\Omega}(t)$ (i.e., fictitious inertial, Coriolis, centrifugal, and Euler forces) [5, 16]:

$$\begin{aligned}
\mathbf{F}_\mathrm{em} &= q_\mathrm{p}\left(\mathbf{E} + \mathbf{v}_\mathrm{p}\times\mathbf{B}\right), && \text{(8a)}\\
\mathbf{F}_\mathrm{ep} &= \left(\frac{\pi}{6}\right)q_\mathrm{p}\mathbf{E}, && \text{(8b)}\\
\mathbf{F}_\mathrm{tp} &= -\frac{6\pi\mu\nu d_\mathrm{p}C_\mathrm{s}}{1 + 6C_\mathrm{m}\mathrm{Kn}_\mathrm{p}}\left(\frac{k/k_\mathrm{p} + 2C_\mathrm{t}\mathrm{Kn}_\mathrm{p}}{1 + 2k/k_\mathrm{p} + 4C_\mathrm{t}\mathrm{Kn}_\mathrm{p}}\right)\nabla\ln T, && \text{(8c)}\\
\mathbf{F}_\mathrm{translational} &= -m_\mathrm{p}\mathbf{a}, && \text{(8d)}\\
\mathbf{F}_\mathrm{Coriolis} &= -2m_\mathrm{p}\boldsymbol{\Omega}\times\mathbf{v}_\mathrm{p}, && \text{(8e)}\\
\mathbf{F}_\mathrm{centrifugal} &= -m_\mathrm{p}\boldsymbol{\Omega}\times\left(\boldsymbol{\Omega}\times\mathbf{x}_\mathrm{p}\right), && \text{(8f)}\\
\mathbf{F}_\mathrm{Euler} &= -m_\mathrm{p}\frac{\mathrm{d}\boldsymbol{\Omega}}{\mathrm{d}t}\times\mathbf{x}_\mathrm{p}, && \text{(8g)}
\end{aligned}$$

*See, e.g., Maxey [15] Eqs. 26 and 27.

where q_p is the particle charge, μ and $\nu = \mu/\rho$ are the carrier fluid dynamic* and kinematic viscosities, respectively; $C_\mathrm{s} = 1.17$ is the thermal slip coefficient; $C_\mathrm{m} = 1.14$ is the momentum exchange coefficient; $C_\mathrm{t} = 2.18$ is the thermal exchange coefficient; $\mathrm{Kn}_\mathrm{p} = \lambda_\mathrm{p}/d_\mathrm{p}$ is the Knudsen number for spherical particles with mean free path λ_p; and k and k_p are the thermal conductivities for the carrier and dispersed phases, respectively.

Assuming that particles are nearly neutrally charged and/or electromagnetic fields in the atmosphere are negligible, Lorentz and electrophoretic forces can be ignored. This is probably a reasonable assumption for explosions at altitudes lower than $\sim 10\,\mathrm{mi}$, where the air density is relatively high, because the ionization and electron charge density will be very spatially limited and temporally limited to at most $\sim 4\,\mathrm{min}$ [17]. Fallout debris that reaches a height of $\sim 35\,\mathrm{mi}$ may result in additional ionization of the D-region due to beta particles, however [17]. Phoretic forces such as the electrophoretic and thermophoretic forces are caused by large field gradients and are typically significant only for submicron particles. Since the electric field and temperature gradients should be relatively weak in the atmosphere removed from the early times around the explosion, we may also neglect both the electrophoretic and thermophoretic forces.

Among the fictitious forces, the Coriolis and centrifugal forces are sufficient for a fluid element relative to the Earth's surface because the Earth's angular speed is nearly constant, and the relative translational motion can be neglected [14]. Furthermore, the centrifugal force can also be written in terms of a potential gradient:

$$\mathbf{F}_\mathrm{centrifugal} = -m_\mathrm{p}\boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{x}_\mathrm{p}) = -m_\mathrm{p}\nabla\Phi_\mathrm{centrifugal}, \tag{10}$$

with [14]

$$\Phi_\mathrm{centrifugal} = -\frac{1}{2}\left|\boldsymbol{\Omega} \times \mathbf{x}_\mathrm{p}\right|^2. \tag{11}$$

Thus, the centrifugal and gravitational forces may be combined into an effective gravitational potential known as the geopotential

$$\Phi_\mathrm{geo} = \Phi_\mathrm{grav} + \Phi_\mathrm{centrifugal} = -\mathbf{g}\cdot\mathbf{x}_\mathrm{p} - \frac{1}{2}\left|\boldsymbol{\Omega} \times \mathbf{x}_\mathrm{p}\right|^2. \tag{12}$$

Typically, meteorological data sets provide the geopotential, which thus represents the effective gravity on Earth due to true gravity and the centrifugal force, resulting in an effective gravity perpendicular to the level surfaces of the approximately oblate spheroid of the Earth. Thus, although the centrifugal force is much smaller than the gravitational force [18, 19], it is typically incorporated with the gravitational potential and does not appear explicitly in the body force terms.

On the other hand, while the centrifugal force is much larger than the Coriolis force [18, 19], the Coriolis force should not be neglected for motions over a duration on the order of the Earth's period of rotation, $L/v_\mathrm{p} \gtrsim \Omega^{-1} = 24\,\mathrm{h}/2\pi \approx 1.4 \times 10^4\,\mathrm{s}$ [18, 20]. Using Figure 20, typically sized† particles $d_\mathrm{p} \sim 100\,\mathrm{\mu m}$ have a terminal velocity $v_\mathrm{t} \sim 1\,\mathrm{m\,s^{-1}}$, whereas Glasstone [17] Figure 9.96 shows a cloud centroid height of $h_\mathrm{cloud} \sim 30 \times 10^3\,\mathrm{ft} = 9\,\mathrm{km}$ or $h_\mathrm{cloud} \sim 50 \times 10^3\,\mathrm{ft} = 15\,\mathrm{km}$ for a $10^2\,\mathrm{kt}$ or $10^3\,\mathrm{kt}$ yield explosion,

*The dynamic viscosity in the atmosphere can be calculated as a function of temperature using the following expression derived from kinetic theory [1]:

$$\mu = \frac{\beta T^{3/2}}{T + S}, \tag{9}$$

where $\beta = 1.458 \times 10^{-6}\,\mathrm{Pa\cdot s/K^{1/2}}$, and $S = 110.4\,\mathrm{K}$.

†The default particle size distribution used by DELFIC is a log-normal distribution with median particle diameter [2]

$$d_{\mathrm{p},50} = \begin{cases} 0.407\,\mathrm{\mu m} & \text{(near) surface} \\ 0.15\,\mathrm{\mu m} & \text{air burst} \end{cases},$$

respectively. Supposing that the vertical wind velocity $u_{\mathrm{s},z}$ is of similar or lesser magnitude than the terminal velocity v_t, then $v_\mathrm{p} \sim v_\mathrm{t}$. Thus, typically sized particles may have a fall time scale on the order of $10\,\mathrm{km}/1\,\mathrm{m\,s^{-1}} = 10^4\,\mathrm{s} \sim \Omega^{-1} \approx 1.4 \times 10^4\,\mathrm{s}$, and Coriolis forces may affect the trajectory for typically sized and smaller particles (i.e., $d_\mathrm{p} \lesssim 10\,\mathrm{\mu m}$) capable of staying aloft and traveling large distances. On the other hand, it may be reasonable to neglect the Coriolis force for large particles (i.e., $d_\mathrm{p} \gtrsim 100\,\mathrm{\mu m}$) that fall out fairly quickly (i.e., $v_\mathrm{t} \sim 10\,\mathrm{m\,s^{-1}}$) and do not travel particularly far.

2.1.3 Surface Forces

There are two approaches for describing the surface forces: the point-force approach and the resolved surface approach [7]. In the point-force approach, the flow around the particles is unresolved, and it uses a linear combination of empirical or analytical surface-averaged forces. The computational cost of the resolved-surface approach is high; therefore, this approach is only suitable for a very small number of particles and/or when their shape is complex [7]. Thus, the point-force approach is more often used for practical applications with many small particles.

2.1.3.1 Point-force approach

The point-force approach does not resolve the detailed flow and surface forces around the particles and instead uses a linear combination of surface-averaged forces. Thus, to use the point-force approach, the particle diameters d_p must be smaller than the relevant length scales L in the carrier fluid flow [6, 21]. For turbulent flows, this length scale is the Kolmogorov length scale $L \sim \eta_\mathrm{K} \equiv (\nu^3/\varepsilon)^{1/4}$ [6, 21, 22], which as discussed in Section 3.2 is on the order of $\eta_\mathrm{K} \sim 1\,\mathrm{mm}$ in the atmosphere. Thus, most fallout particles should be sufficiently small to be modeled using the point-force approach in the atmosphere.

In the point-force approach, the surface force term is further decomposed into inertial, steady-state drag, virtual (or added) mass, Basset history, and lift (both Saffman and Magnus) forces [6, 7, 15]:

$$\mathbf{F}_\mathrm{s} = \mathbf{F}_\mathrm{inertial} + \mathbf{F}_\mathrm{drag} + \mathbf{F}_\mathrm{vm} + \mathbf{F}_\mathrm{B} + \mathbf{F}_\mathrm{S} + \mathbf{F}_\mathrm{M}, \tag{13}$$

resulting in the volume-median diameter (used to distribute the fallout cloud mass)

$$d_{\mathrm{p},V,50} = \begin{cases} 130\,\mathrm{\mu m} & \text{(near) surface} \\ 0.63\,\mathrm{\mu m} & \text{air burst} \end{cases}.$$

Thus, the typical surface burst particle diameter is $d_\mathrm{p} \sim d_{\mathrm{p},V,50} \sim 100\,\mathrm{\mu m}$.

with [5, 6, 7, 15, 23]

$$\mathbf{F}_{\mathrm{inertial}} = \rho V_{\mathrm{p}} \left[\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)}, \tag{14a}$$

$$\mathbf{F}_{\mathrm{drag}} = -3\pi\mu d_{\mathrm{p}} \left[\mathbf{v}_{\mathrm{p}} - \mathbf{u}_{\mathrm{s}} - \frac{d_{\mathrm{p}}^2}{24}\left(\nabla^2\mathbf{u}\right)_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)}\right], \tag{14b}$$

$$\mathbf{F}_{\mathrm{vm}} = -\frac{\rho V_{\mathrm{p}} C_{\mathrm{vm}}}{2} \left[\frac{\mathrm{d}\mathbf{v}_{\mathrm{p}}(t)}{\mathrm{d}t} - \left[\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} - \frac{d_{\mathrm{p}}^2}{40}\frac{\mathrm{d}}{\mathrm{d}t}\left(\nabla^2\mathbf{u}\right)_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)}\right], \tag{14c}$$

$$\begin{aligned}\mathbf{F}_{\mathrm{B}} = -\frac{3}{2}C_{\mathrm{B}} d_{\mathrm{p}}^2\sqrt{\pi\mu\rho}\Bigg\{\int_0^t \left(t-t'\right)^{-1/2}\left[\frac{\mathrm{d}\mathbf{v}_{\mathrm{p}}(t')}{\mathrm{d}t'} - \left[\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t')} - \frac{d_{\mathrm{p}}^2}{24}\frac{\mathrm{d}}{\mathrm{d}t'}\left(\nabla^2\mathbf{u}\right)_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t')}\right]\mathrm{d}t' \\ + t^{-1/2}\left[\mathbf{v}_{\mathrm{p}}(0) - \mathbf{u}_{\mathrm{s}}(0)\right]\Bigg\},\end{aligned} \tag{14d}$$

$$\mathbf{F}_{\mathrm{S}} = \rho C_{\mathrm{L}}\left(\mathbf{v}_{\mathrm{p}} - \mathbf{u}_{\mathrm{s}}\right)\times\boldsymbol{\omega}_{\mathrm{s}}, \tag{14e}$$

$$\mathbf{F}_{\mathrm{M}} = -\frac{\pi}{8}d_{\mathrm{p}}^3\rho\left(\mathbf{v}_{\mathrm{p}} - \mathbf{u}_{\mathrm{s}}\right)\times\left(\boldsymbol{\Omega}_{\mathrm{p}} - \frac{1}{2}\boldsymbol{\omega}_{\mathrm{s}}\right), \tag{14f}$$

where C_{vm}, C_{B}, and C_{L} are the virtual mass, Basset, and (Saffman) lift, coefficients, respectively,

$$\mathbf{u}_{\mathrm{s}}(t) \equiv \mathbf{u}\left[\mathbf{x}_{\mathrm{p}}(t), t\right] \tag{15}$$

$$\left(\nabla^2\mathbf{u}\right)_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} \equiv \left[\nabla^2\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} \tag{16}$$

$$\boldsymbol{\omega}_{\mathrm{s}}(t) \equiv \left[\nabla\times\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} \tag{17}$$

are the undisturbed fluid velocity $\mathbf{u}(\mathbf{x},t)$, velocity Laplacian $\nabla^2\mathbf{u}$, and vorticity $\boldsymbol{\omega}$, respectively, observed at the particle center of mass $\mathbf{x}_{\mathrm{p}}(t)$, and $\boldsymbol{\Omega}_{\mathrm{p}}(t)$ is the angular velocity of the particle obtained from the particle angular momentum equation [6, 24]. Note that Eqs. 14 contain two different accelerations: the first is the usual particle acceleration $\mathrm{d}\mathbf{v}_{\mathrm{p}}(t)/\mathrm{d}t$, and the second is the fluid acceleration observed at the particle center of mass $[\mathrm{D}\mathbf{u}/\mathrm{D}t]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)}$, with the material or substantial derivative defined (as usual) as the time derivative following the fluid element

$$\frac{\mathrm{D}}{\mathrm{D}t} \equiv \frac{\partial}{\partial t} + \mathbf{u}(\mathbf{x},t)\cdot\nabla = \frac{\partial}{\partial t} + u_i(\mathbf{x},t)\frac{\partial}{\partial x_i}. \tag{18}$$

This latter acceleration has involved some uncertainty [15, 23] about use of the substantial derivative Eq. 18,

$$\begin{aligned}\left[\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} &\equiv \left[\frac{\partial}{\partial t}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} + u_i\left[\mathbf{x}_{\mathrm{p}}(t), t\right]\left[\frac{\partial}{\partial x_i}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} \\ &= \left[\frac{\partial}{\partial t}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} + u_{\mathrm{s},i}(t)\left[\frac{\partial}{\partial x_i}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)},\end{aligned} \tag{19}$$

or the observed fluid velocity time derivative,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}_{\mathrm{s}}(t) &= \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}\left[\mathbf{x}_{\mathrm{p}}(t), t\right] \\ &= \left[\frac{\partial}{\partial t}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} + \frac{\mathrm{d}x_{\mathrm{p},i}(t)}{\mathrm{d}t}\left[\frac{\partial}{\partial x_i}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} \\ &= \left[\frac{\partial}{\partial t}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)} + v_{\mathrm{p},i}(t)\left[\frac{\partial}{\partial x_i}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_{\mathrm{p}}(t)}.\end{aligned} \tag{20}$$

Equation 19 is sometimes also denoted as $[\mathrm{D}\mathbf{u}/\mathrm{D}t]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)} = \mathrm{D}\mathbf{u}_\mathrm{s}/\mathrm{D}t$ [23], which unfortunately also creates some confusion with respect to Eq. 20 for $\mathrm{d}\mathbf{u}_\mathrm{s}/\mathrm{d}t$. Physically, Eq. 19 represents the fluid acceleration seen by an observer moving with the fluid velocity $\mathbf{u}$ at the particle center of mass $\mathbf{x}_\mathrm{p}(t)$, whereas Eq. 20 represents the fluid acceleration seen by an observer traveling with the particle [15,23]. It is the former acceleration that is correct [23] and physically more realistic [15]. Nonetheless, it is common for authors to mistakenly use $\mathrm{d}\mathbf{u}_\mathrm{s}/\mathrm{d}t$, especially because for low Reynolds numbers they are approximately the same [15].

Inertial force

The inertial force term is simply the result of forces on the particle by the fluid due to the undisturbed fluid flow. It can be found quite generally and without specific assumptions about low Reynolds number [15] simply by integrating the Cauchy stress tensor forces as suggested in Section 2.1.2. Assuming the particle is sufficiently small, the Cauchy stress tensor (or divergence) can be treated as a constant over the particle surface (or volume). This leads directly to Eq. 14a with the Cauchy stress tensor written in terms of the Navier-Stokes equation undisturbed acceleration term at the particle center of mass.

Steady-state drag

The steady-state drag term in Eq. 14b includes the Faxen force, which is a second-order correction to the Stokes drag for the curvature of the velocity profile in the conveying flow field [5]. In most applications of dispersed multiphase flows, for which $r_\mathrm{p}/L \ll 1$, the Faxen term is small enough to be neglected in comparison to other effects such as Reynolds number effects [5,6]. Thus, Eq. 14b is typically written in one of several forms [5,6]:

$$\begin{aligned}
\mathbf{F}_\mathrm{drag} &= -\frac{1}{2}\rho C_\mathrm{D} A_\mathrm{D,p} \left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right| \left(\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right) && \text{(21a)}\\
&= -3\pi\mu d_\mathrm{p} f_\mathrm{drag} \left(\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right) && \text{(21b)}\\
&= -\frac{m_\mathrm{p}}{\tau_\mathrm{drag}} \left(\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right), && \text{(21c)}
\end{aligned}$$

where C_D is the drag coefficient, $A_\mathrm{D,p} = \pi r_\mathrm{p}^2 = \pi d_\mathrm{p}^2/4$ is the representative area of the droplet or the projected area of the particle or droplet in the direction of the relative velocity [5],

$$f_\mathrm{drag} = \frac{\rho C_\mathrm{D} d_\mathrm{p} \left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right|}{24\mu} = \frac{C_\mathrm{D}\mathrm{Re}_\mathrm{r}}{24} \tag{22}$$

is the drag factor for the ratio of the drag coefficient to the Stokes drag [5,6,7],

$$\mathrm{Re}_\mathrm{r} = \frac{\rho d_\mathrm{p} \left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right|}{\mu} = \frac{d_\mathrm{p} \left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right|}{\nu} \tag{23}$$

is the particle, dispersed phase, or relative Reynolds number [5,6], and

$$\tau_\mathrm{drag} = \frac{\tau_\mathrm{Stokes}}{f_\mathrm{drag}} \tag{24}$$

is the particle-to-fluid drag relaxation or response time written in terms of the Stokes timescale [6,11]

$$\tau_\mathrm{Stokes} = \frac{m_\mathrm{p}}{3\pi\mu d_\mathrm{p}} = \frac{\rho_\mathrm{p} d_\mathrm{p}^2}{18\mu} = \frac{\rho_\mathrm{p} d_\mathrm{p}^2}{18\rho\nu}. \tag{25}$$

For solid particles of irregular (i.e., nonspherical) shape, several corrections to the drag factor f_drag exist relying on the volume- or surface-area- equivalent sphere diameters $d_{\mathrm{p},V} = (6V_\mathrm{p}/\pi)^{1/3}$ or

$d_{\mathrm{p},A} = (4A_\mathrm{p}/\pi)^{1/2}$ as well as possibly a particle surface sphericity $c = \pi d_{\mathrm{p},A}/P_\mathrm{p}$, where P_p is the particle's projected perimeter [5,6]. In all cases, the drag of irregular shapes is up to 50% higher than the drag of a sphere of the same volume-equivalent diameter [6]. Figure 1 plots the Stokes timescale as a function of (spherical) particle diameter d_p for various particle densities ρ_p. The Stokes time scale is less than $\tau_\mathrm{Stokes} \lesssim 10\,\mathrm{s}$ for particles up to $d_\mathrm{p} \lesssim 1\,\mathrm{mm}$ and decreases rapidly as the square of d_p decreases.

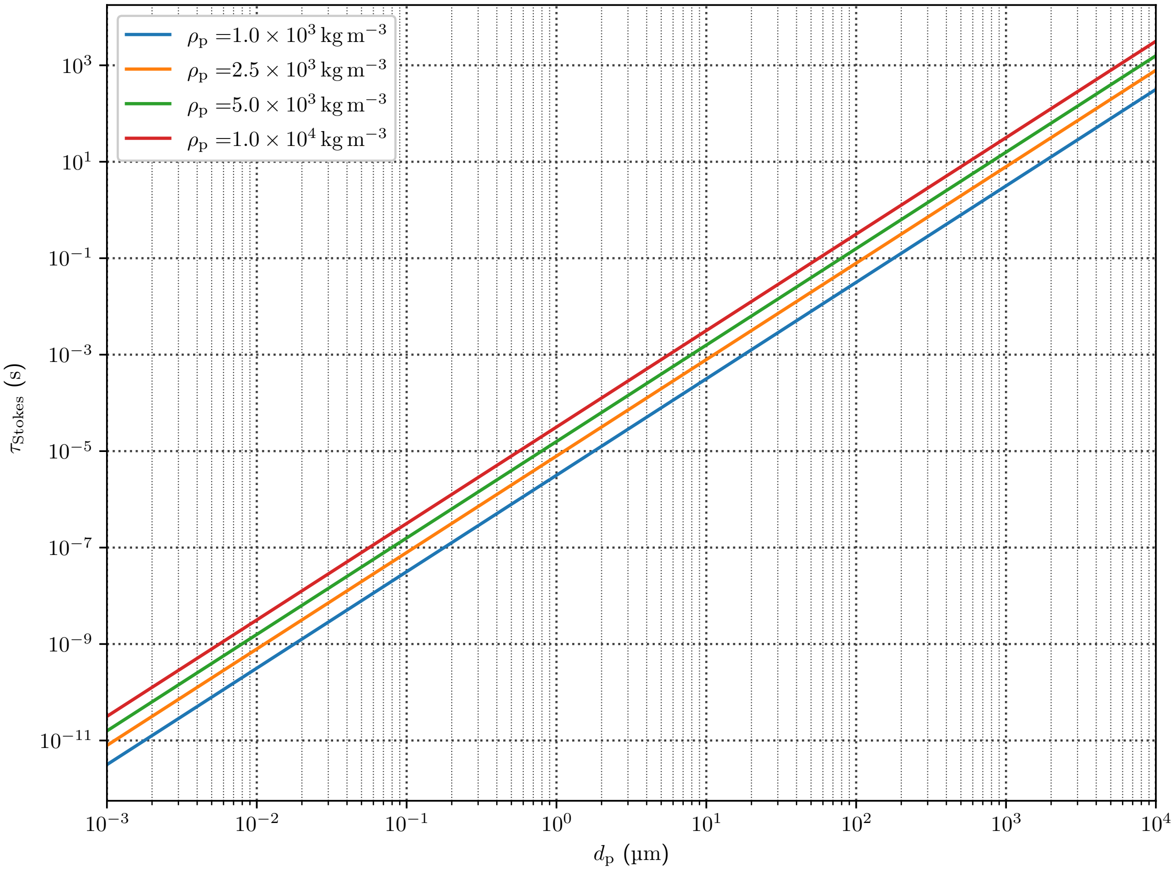

Figure 1. Stokes time scale τ_Stokes for solid spherical particles of mass density $\rho_\mathrm{p} \in \{1, 2.5, 5, 10\} \times 10^3\,\mathrm{kg\,m^{-3}}$ and varying diameter $10^{-3}\,\mathrm{\mu m} \le d_\mathrm{p} \le 10^4\,\mathrm{\mu m}$ in air at the altitude $z = 1\,\mathrm{km}$ using the US Standard Atmosphere [1]. The dynamic viscosity of air $\mu = 1.757\,85 \times 10^{-5}\,\mathrm{Pa\,s}$ is calculated using Eq. 9 with $T = 281.651\,\mathrm{K}$.

In situations the particle density is much larger than the fluid density $\rho/\rho_\mathrm{p} \ll 1$—such as for solid particles in air, for which $\rho/\rho_\mathrm{p} \sim 10^{-3}$—it is common to neglect the fluid inertia, virtual (added) mass, and Basset history force terms [5,6,21]. Even though these forces can at times be significant in such cases, they are typically an order of magnitude smaller than the steady-state drag force and have second-order effects on the particle dispersion [6]. Similarly, the Saffman and Magnus lift forces play an appreciable role only when there is significant fluid vorticity or particle rotation. Furthermore, among Eqs. 14, only the steady-state drag Eq. 14b does not explicitly include a factor ρ such that when divided by $m_\mathrm{p} = \rho_\mathrm{p} V_\mathrm{p}$ will yield the small factor $\rho/\rho_\mathrm{p} \sim 10^{-3}$ for solid particles in air. Thus, it is common in practice for these cases to employ a point-force surface force model based only on the steady-state drag:

$$\mathbf{F}_\mathrm{s} \approx \mathbf{F}_\mathrm{drag} = -3\pi\mu d_\mathrm{p} f_\mathrm{drag}\left(\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right), \quad \rho/\rho_\mathrm{p} \ll 1. \tag{26}$$

For solid fallout particles in the air, this is a reasonable assumption, but for gaseous fallout particles such as xenon, this is no longer necessarily the case. Nevertheless, transport codes often approximate the surface forces by the drag force alone.

Virtual, added, or apparent mass force

The virtual, added, or apparent mass force accounts for a drag due to the difference in acceleration between particles and the surrounding fluid [7]. The particle acceleration creates a corresponding acceleration of the fluid, which is at the expense of work done by the particle [5, 7]. For spherical particles, the virtual mass coefficient is taken as unity $C_\mathrm{vm} = 1$, but in general it can depend on dispersed phase volume fraction α_p and the particle Reynolds number Re_r [5, 7]. This force can be dominant when there is a significant difference between the density of the continuous and dispersed phases, such as for solid particles in a fluid [7]. As in the case of the Faxen second-order correction term appearing in the drag force Eq. 14b, the analogous correction term in Eq. 14c is usually neglected [5]:

$$\mathbf{F}_\mathrm{vm} = -\frac{\rho V_\mathrm{p} C_\mathrm{vm}}{2}\left(\frac{\mathrm{d}\mathbf{v}_\mathrm{p}(t)}{\mathrm{d}t} - \left[\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)}\right). \tag{27}$$

In addition to the drag correction due to finite Reynolds number for particles in general and non-Stokes flows, the particle relaxation or response time scale also has a mass factor due to the added (virtual) mass force [25, 26, 27]:

$$\tau_\mathrm{p} = \left(1 + \frac{\rho}{2\rho_\mathrm{p}}\right)\tau_\mathrm{drag} = \left(1 + \frac{\rho}{2\rho_\mathrm{p}}\right)\frac{\tau_\mathrm{Stokes}}{f_\mathrm{drag}} = \frac{\left[2(\rho_\mathrm{p}/\rho) + 1\right]}{36}\frac{d_\mathrm{p}^2}{\nu}\frac{1}{f_\mathrm{drag}}. \tag{28}$$

When $\rho_\mathrm{p} \gg \rho$, the added mass term can be neglected, and $\tau_\mathrm{p} \to \tau_\mathrm{drag}$; furthermore, for Stokes flows for which $f_\mathrm{drag} \approx 1$ (see Appendix B), $\tau_\mathrm{p} \to \tau_\mathrm{Stokes}$. For further information on the calculation of the drag coefficient C_D, drag factor f_drag, and terminal velocity v_t for spherical particles, see Appendix B.

Basset or history force

Whereas the virtual mass force accounts for drag due to differences in particle and fluid acceleration, the Basset or history force accounts for the viscous effects of historical acceleration differences [5]. It addresses the temporal delay in boundary layer development as the relative velocity changes with time and represents the cumulative effect of unsteady motion of the particle on the velocity field of the carrier fluid [5, 7]. As in the case of the second-order correction terms appearing in Eqs. 14b and 14c, the analogous correction term in Eq. 14d is usually neglected [5]:

$$\mathbf{F}_\mathrm{B} = -\frac{3}{2}C_\mathrm{B} d_\mathrm{p}^2\sqrt{\pi\mu\rho}\left\{\int_0^t \left(t - t'\right)^{-1/2}\left[\frac{\mathrm{d}\mathbf{v}_\mathrm{p}(t')}{\mathrm{d}t'} - \left[\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t')}\right]\mathrm{d}t' + t^{-1/2}\left[\mathbf{v}_\mathrm{p}(0) - \mathbf{u}_\mathrm{s}(0)\right]\right\} \tag{29}$$

Owing to the integral form of the Basset force, it turns the momentum conservation ordinary differential equation (ODE) into an integro-differential equation, which is much more difficult to solve. Thus, the Basset force is often neglected for practical reasons, but note that this force can be very large when the body is accelerated at a high rate [6, 7].

Lift forces

The Saffman and Magnus lift forces are due to particle rotation caused by a velocity gradient or some other source such as particle contact and rebound from a surface [5]. The Saffman lift force in particular is due to the pressure distribution developed on a particle in a velocity gradient, whereby the higher velocity on the top gives rise to a low pressure, and the high pressure on the low velocity side gives rise to a lift force [5]. On the other hand, the Magnus force is the lift developed due to particle rotation creating a spin-induced lift. It is a consequence of a pressure difference induced from streamline asymmetry resulting from the rotation of the sphere [5, 6, 7]. The Saffman lift force is a special case of the Magnus force for small rotating particles moving in a uniform shear flow at a low Reynolds number when local gradients of the continuous phase velocity and the pressure difference between the top and bottom of the particle rotating with the fluid are important [6, 7]. Thus, make sure to only account for the lift force once in the particle dynamics equation.

2.1.3.2 Resolved-surface approach

A more detailed approach is to completely resolve the flow surrounding the particle and simply integrate the Cauchy stress tensor $\boldsymbol{\sigma}$ forces over the particle surface:

$$\mathbf{F}_{\mathrm{s}} = \int_{V_{\mathrm{p}}} \nabla \cdot \boldsymbol{\sigma} \mathrm{d}V = \int_{\partial V_{\mathrm{p}}} \boldsymbol{\sigma} \cdot \hat{\mathbf{n}} \mathrm{d}S, \tag{30}$$

where $\partial V_{\mathrm{p}} = S_{\mathrm{p}}$ is the particle surface. In Einstein notation, Eq. 30 can equivalently be written as [5]

$$F_{\mathrm{s},i} = \int_{V_{\mathrm{p}}} \frac{\partial \sigma_{ij}}{\partial x_j} \mathrm{d}V = \int_{\partial V_{\mathrm{p}}} \sigma_{ij} \hat{n}_j \mathrm{d}S. \tag{31}$$

The Cauchy stress tensor for a linear, incompressible Newtonian fluid (such as air [19, 28]) takes the form

$$\sigma_{ij} = -p\delta_{ij} + 2\mu e_{ij}, \tag{32}$$

where μ is the dynamic viscosity, and $e_{ij} = \left(\partial u_i / \partial x_j + \partial u_j / \partial x_i\right)/2$ is the rate of strain tensor. See Appendix A for details.

2.1.4 Collision Forces

Finally, collision forces include both particle–particle collisions and particle–wall collisions [7]. Interparticle collisions result in both translational and rotational movement, where the rotation rate $\boldsymbol{\Omega}_{\mathrm{p}}(t)$ is obtained from the angular momentum equation [6, 24]. There are two commonly used approaches for describing the particle collisions: hard-sphere and soft-sphere [5, 6, 24]. More advanced methods for modeling collisions based on kinetic gas theory and the Boltzmann equation are also possible [29, 30].

2.1.4.1 Particle–particle collisions

For dilute gas–particle flows, particle–particle collisions can be neglected, but as the particle concentration increases, this effect cannot be neglected [5]. In particular, the flow is considered dilute if [5, 6]

$$\frac{\tau_{\text{Stokes}}}{\tau_{\text{collision}}} = \tau_{\text{Stokes}} f_{\text{collision}} \ll 1, \tag{33}$$

where $\tau_{\text{collision}}$ and $\tau_{\text{collision}}^{-1} = f_{\text{collision}}$ are the particle–particle collision time and frequency, respectively. Abrahamson [31] suggested that the collision frequency for solid particles in a gas with number density n_{p} is given by [5, 6]

$$f_{\text{collision}} = 4\sqrt{\pi} n_{\mathrm{p}} d_{\mathrm{p}}^2 \sigma_v, \tag{34}$$

where root mean squared (RMS) fluctuation velocity is

$$\sigma_v = \sqrt{\langle v'v' \rangle} = \frac{\sigma}{(1 + 1.5\tau_\mathrm{p}\varepsilon/\sigma^2)^{1/2}}, \tag{35}$$

with fluid velocity variance $\sigma^2 = 2k/3$ (see Section 3.4.2 for discussion of k and ε), and particle relaxation time $\tau_\mathrm{p} = \tau_\mathrm{Stokes}$ for Stokes flows. Asymptotically, Eq. 35 takes the form $\sigma_v = \sigma + O(\tau_\mathrm{p}\varepsilon/\sigma^2)$ or $\sigma_v = \sigma^2\sqrt{2/3\tau_\mathrm{p}\varepsilon} + O\left[\sigma^4(\tau_\mathrm{p}\varepsilon)^{-3/2}\right]$ for $\tau_\mathrm{p}\varepsilon \ll \sigma^2 \Rightarrow \tau_\mathrm{p}\varepsilon/\sigma^2 \to 0$ or $\tau_\mathrm{p}\varepsilon \gg \sigma^2 \Rightarrow \tau_\mathrm{p}\varepsilon/\sigma^2 \to \infty$, respectively. Thus, small particles with a short relaxation time have a fluctuating velocity on the order of the fluid velocity deviation, whereas large particles with longer relaxation time will somewhat reduce the effect of fluid velocity deviation. In particular, given the discussion in Section 3.1, typically one has $1\,\mathrm{s} \lesssim \sigma^2/\varepsilon = 2k/3\varepsilon \lesssim 10^2\,\mathrm{s}$, which is much greater than the typical particle response time—$\tau_\mathrm{Stokes} \sim 0.1\,\mathrm{s}$ even for solid particles as large as $d_\mathrm{p} \sim 100\,\mu\mathrm{m}$. Thus, typically $\tau_\mathrm{p}\varepsilon \ll \sigma^2 \Rightarrow \sigma_v \sim \sigma \sim 1\,\mathrm{m\,s^{-1}}$. A complete review of various particle collision models and frequencies is provided by Meyer [32], but Eq. 34 serves as an upper bound. Then substituting Eqs. 25 and 34 into Eq. 33 requires

$$\frac{\tau_\mathrm{Stokes}}{\tau_\mathrm{collision}} = \frac{2\sqrt{\pi}\rho_\mathrm{p}n_\mathrm{p}d_\mathrm{p}^4\sigma_v}{9\mu} = \frac{4\rho_\mathrm{p}\alpha_\mathrm{p}d_\mathrm{p}\sigma_v}{3\sqrt{\pi}\mu} \ll 1, \tag{36}$$

where $\alpha_\mathrm{p} = N_\mathrm{p}V_\mathrm{p}/V = \pi n_\mathrm{p}d_\mathrm{p}^3/6$ is the particle volume fraction. Solving Eq. 36 for α_p then requires the particle volumetric fraction be constrained as a function of particle diameter d_p as

$$\alpha_\mathrm{p} = \frac{3\sqrt{\pi}\mu}{4\rho_\mathrm{p}d_\mathrm{p}\sigma_v}\left(\frac{\tau_\mathrm{Stokes}}{\tau_\mathrm{collision}}\right) \ll \frac{3\sqrt{\pi}\mu}{4\rho_\mathrm{p}d_\mathrm{p}\sigma_v}. \tag{37}$$

Figure 2 shows the upper bound on volumetric fraction α_p as a function of particle diameter d_p using Eq. 37 for a range of solid particle densities in the atmosphere with $\tau_\mathrm{Stokes}/\tau_\mathrm{collision} = 0.1 \ll 1$. Looking at Figure 2, we can see that the largest, densest particles should have $\alpha_\mathrm{p} \lesssim 10^{-7}$, but more typically sized particles require $\alpha_\mathrm{p} \lesssim 10^{-5}$. This regime corresponds to a very dilute fallout particle cloud, and some simple DELFIC cloud rise simulations show that the stabilized fallout cloud parcels all appear to be well within this very dilute regime.

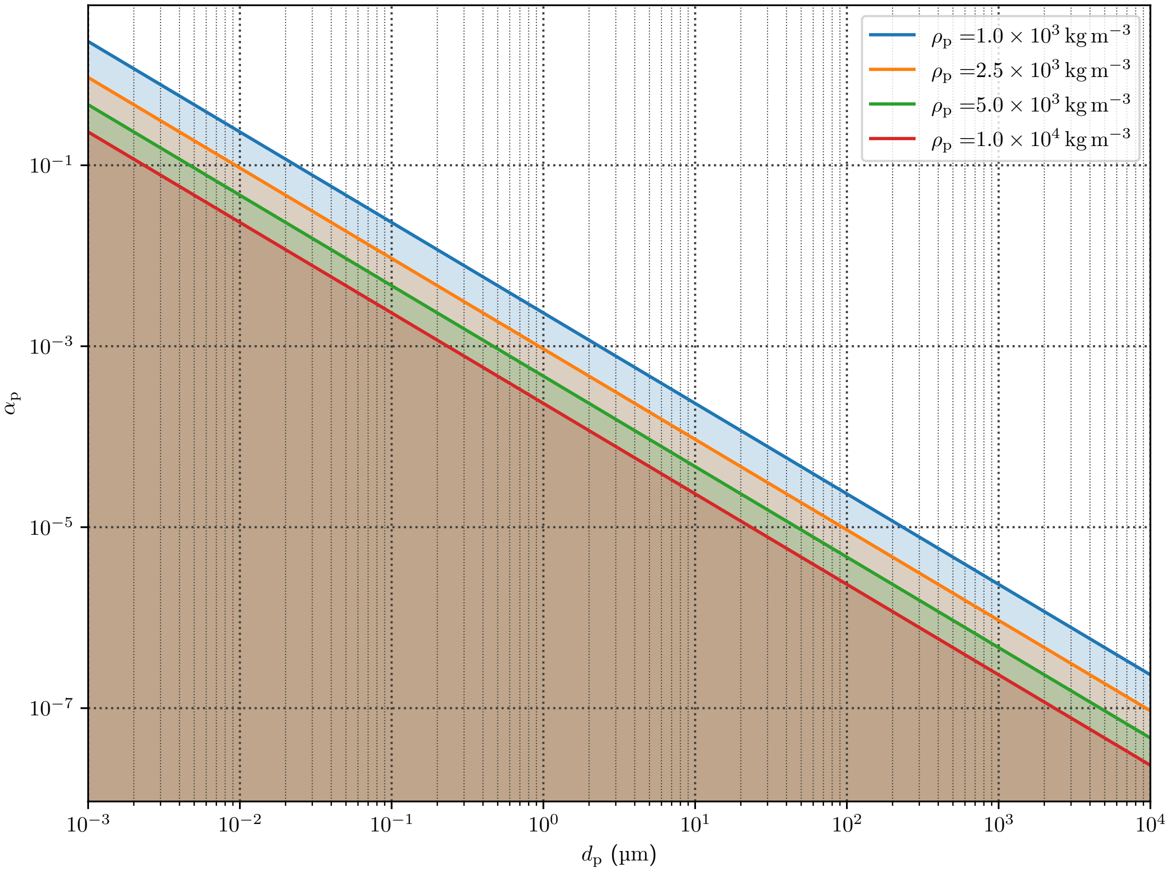

Figure 2. Particle volumetric fraction $\alpha_\mathrm{p} \equiv N_\mathrm{p} V_\mathrm{p}/V = \pi n_\mathrm{p} d_\mathrm{p}^3/6$ for solid spherical particles of mass density $\rho_\mathrm{p} \in \{1, 2.5, 5, 10\} \times 10^3\,\mathrm{kg\,m^{-3}}$ and varying diameter $10^{-3}\,\mathrm{\mu m} \leq d_\mathrm{p} \leq 10^4\,\mathrm{\mu m}$ such that particle–particle collisions may be neglected with Stokes-to-collision time ratio $\tau_\mathrm{Stokes}/\tau_\mathrm{collision} = 0.1 \ll 1$ in air at the altitude $z = 1\,\mathrm{km}$ using the US Standard Atmosphere [1]. The dynamic viscosity of air $\mu = 1.757\,85 \times 10^{-5}\,\mathrm{Pa\,s}$ is calculated using Eq. 9 with $T = 281.651\,\mathrm{K}$, and the RMS fluctuating velocity is given by $\sigma_v = 1\,\mathrm{m\,s^{-1}}$. The shaded regions are the regions for which constraint Eq. 37 is satisfied.

2.1.4.2 Particle–wall collisions

Particle–wall interactions fall into two categories: hydrodynamic forces due to proximity of a wall and the purely mechanical interaction in the absence of a fluid [5, 6]. The hydrodynamic interaction can be neglected if the particle inertia force is so large that collision takes place in a time small compared with the hydrodynamic relaxation time $\tau_\mathrm{p} \sim \tau_\mathrm{Stokes}$ of the particle [5, 6]. For large particles colliding with the wall, the mechanical interaction results in an inelastic collision in which the particle rebounds with kinetic energy loss due to friction and inelasticity effects [5, 6]. When a small particle approaches a wall, however, the molecular forces dominate the inertial forces, resulting in the particle's capture by the wall's cohesive van der Waals forces [5, 6].

2.2 FALLOUT PARTICLE KINEMATICS

In order to rapidly calculate the transport of the Lagrangian discrete parcels, DELFIC uses a steady-state kinematic solution of rigid, spherical particles subject to steady-state drag as well as gravitational forces. Section 2.2.1 first reformulates the particle dynamics equation in terms of a particle velocity relative to that of the local fluid velocity field. This serves as the basis for the derivations and discussions of the kinematic solutions in Sections 2.2.2 to 2.2.5. In particular, Section 2.2.2 derives the particle velocity and displacement with zero initial relative velocity in the presence of a piecewise uniform, time-independent fluid velocity field, placing constraints on the spatial extents of meteorological cells required for particles to reach steady state. Section 2.2.3 then relaxes the initial condition in the horizontal dimensions and assesses the constraints on the temporal variations of the meteorological wind fields. We then examine the requirements on a particle crossing a meteorological spatial cell boundary and the constraints on the spatial variations of the meteorological wind fields in Section 2.2.4. Finally, Section 2.2.5 constructs an analytical solution allowing for arbitrary initial conditions in the presence of a time-dependent wind field, demonstrating that constraints on the particle relative velocity are not required as long as the particle is able to reach steady state within the meteorological spatial cell.

2.2.1 Particle Dynamics and Relative Velocity Transform

The theory behind the Lagrangian transport portion of DELFIC begins with a discrete particle momentum conservation law in the form of Eq. 2 for a particle of constant mass m_p and including only steady-state drag and gravitational forces, excluding buoyancy [10, 33]:

$$\frac{\mathrm{d}\mathbf{v}_\mathrm{p}}{\mathrm{d}t} = -\left(\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right)\phi\left(\left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right|\right) + \mathbf{f}_\mathrm{grav}, \tag{38}$$

where

$$\phi\left(\left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right|\right) = \tau_\mathrm{drag}^{-1} = \tau_\mathrm{Stokes}^{-1} f_\mathrm{drag} = \frac{3\pi\mu d_\mathrm{p}}{m_\mathrm{p}} f_\mathrm{drag} = \frac{\pi\mu d_\mathrm{p} C_\mathrm{D}\mathrm{Re}_\mathrm{r}}{8m_\mathrm{p}} = \frac{\rho C_\mathrm{D} A_\mathrm{D,p}}{2m_\mathrm{p}}\left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right| = \alpha\left|\mathbf{v}_\mathrm{p} - \mathbf{u}_\mathrm{s}\right|, \tag{39}$$

$\alpha = \rho C_\mathrm{D} A_\mathrm{D,p}/2m_\mathrm{p}$, and $\mathbf{f}_\mathrm{grav} = \mathbf{g} = -g\hat{\mathbf{z}}$ is the body force per unit mass (i.e., acceleration) due to gravity. Excluding the buoyancy force for solid fallout particles in the air is probably reasonable since $m_\mathrm{grav} \approx m_\mathrm{p}$ for $\rho/\rho_\mathrm{p} \ll 1$ from Eq. 6. As indicated by the form of Eqs. 8, provided there are no significant temperature gradients or electric fields, gravity may be the only significant real body force. Among the fictitious forces, the linear acceleration, Euler, and centrifugal forces can likely also be neglected compared with gravity for atmospheric transport on Earth. Whether or not the Coriolis force can be neglected will depend on the length and time scales of the particle transport, as discussed in Section 2.1.2. As discussed in Section 2.1.3, as a practical matter, it is typical that only the steady-state drag surface force is considered. Whether or not the inertia, virtual mass, Basset, or lift forces can truly be neglected may require more careful consideration. Consistent with Section 2.1.4, provided that the particles are dilute, particle–particle collisions can be ignored, whereas generally particle–wall collisions must be considered as a boundary condition.

Equation 38 is then rewritten using a variable transform

$$\mathbf{w}_\mathrm{s}(t) = \mathbf{v}_\mathrm{p}(t) - \mathbf{u}_\mathrm{s}(t). \tag{40}$$

The drag force evaluated at the particle position is simply $-\mathbf{w}_\mathrm{s}\phi\left(\left|\mathbf{w}_\mathrm{s}\right|\right)$, and the gravity force is unchanged by the transform. Using Eq. 20, the time derivative becomes

$$\begin{aligned}\frac{\mathrm{d}\mathbf{v}_\mathrm{p}(t)}{\mathrm{d}t} &= \frac{\mathrm{d}\mathbf{w}_\mathrm{s}(t)}{\mathrm{d}t} + \frac{\mathrm{d}\mathbf{u}_\mathrm{s}(t)}{\mathrm{d}t} \\ &= \frac{\mathrm{d}\mathbf{w}_\mathrm{s}(t)}{\mathrm{d}t} + \left[\frac{\partial}{\partial t}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)} + v_{\mathrm{p},i}(t)\left[\frac{\partial}{\partial x_i}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)} \\ &= \frac{\mathrm{d}\mathbf{w}_\mathrm{s}(t)}{\mathrm{d}t} + \left[\frac{\partial}{\partial t}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)} + \left[w_{\mathrm{s},i}(t) + u_{\mathrm{s},i}(t)\right]\left[\frac{\partial}{\partial x_i}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)}. \end{aligned} \tag{41}$$

Then substituting Eq. 41 into Eq. 38, we obtain the dynamic equation for $\mathbf{w}_\mathrm{s}(t)$

$$\frac{\mathrm{d}\mathbf{w}_\mathrm{s}}{\mathrm{d}t} = -\mathbf{w}_\mathrm{s}\phi\left(|\mathbf{w}_\mathrm{s}|\right) + \mathbf{f}_\mathrm{grav} - \left[\frac{\partial}{\partial t}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)} - \left[w_{\mathrm{s},i}(t) + u_{\mathrm{s},i}(t)\right]\left[\frac{\partial}{\partial x_i}\mathbf{u}(\mathbf{x},t)\right]_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)} \tag{42a}$$

$$= -\mathbf{w}_\mathrm{s}\phi\left(|\mathbf{w}_\mathrm{s}|\right) + \mathbf{f}_\mathrm{grav} - \frac{\partial \mathbf{u}}{\partial t} - \left(\mathbf{u}\cdot\nabla\right)\mathbf{u} - \left(\mathbf{w}_\mathrm{s}\cdot\nabla\right)\mathbf{u}, \tag{42b}$$

where we have used a somewhat abbreviated vector notation in Eq. 42b, keeping in mind that operators on the undisturbed fluid velocity $\mathbf{u}(\mathbf{x},t)$ should be evaluated at the particle center of mass $\mathbf{x}_\mathrm{p}(t)$. Using the velocity transform Eq. 40, Eq. 1 for the particle center of mass $\mathbf{x}_\mathrm{p}(t)$ takes the form

$$\frac{\mathrm{d}\mathbf{x}_\mathrm{p}}{\mathrm{d}t} = \mathbf{w}_\mathrm{s}(t) + \mathbf{u}_\mathrm{s}(t), \tag{43}$$

which when combined with Eq. 42 forms the coupled, nonlinear system of ODE of the general form

$$\frac{\mathrm{d}\mathbf{X}_\mathrm{p}}{\mathrm{d}t} + \mathbf{P}(\mathbf{X}_\mathrm{p},t)\mathbf{X}_\mathrm{p} = \mathbf{Q}(\mathbf{X}_\mathrm{p},t), \tag{44}$$

where $\mathbf{X}_\mathrm{p}(t) = \left[\mathbf{x}_\mathrm{p}(t), \mathbf{w}_\mathrm{s}(t)\right]^\mathrm{T}$ is a column vector of the particle position $\mathbf{x}_\mathrm{p}(t)$ and relative velocity $\mathbf{w}_\mathrm{s}(t)$, and

$$\mathbf{P}(\mathbf{X}_\mathrm{p},t) = \begin{bmatrix} 0 & -1 \\ 0 & \phi\left(|\mathbf{w}_\mathrm{s}|\right)\end{bmatrix}, \tag{45a}$$

$$\mathbf{Q}(\mathbf{X}_\mathrm{p},t) = \begin{bmatrix} \mathbf{u} \\ \mathbf{f}_\mathrm{grav} - \frac{\partial \mathbf{u}}{\partial t} - \left(\mathbf{u}\cdot\nabla\right)\mathbf{u} - \left(\mathbf{w}_\mathrm{s}\cdot\nabla\right)\mathbf{u}\end{bmatrix}_{\mathbf{x}=\mathbf{x}_\mathrm{p}(t)}, \tag{45b}$$

Combined with some initial condition, $\mathbf{X}_\mathrm{p}(t_0) = \mathbf{X}_{\mathrm{p},0}$, the initial value problem of Eq. 44 cannot be analytically solved in general. However, it is possible to solve analytically with some additional constraints on $\mathbf{Q}$ and ϕ [34]. Namely, Ray [34] treats the drag factor α as a constant, although density ρ varies approximately exponentially with the vertical coordinate [19], and the drag coefficient C_D will depend on the relative Reynolds number Re_r, as shown in Appendix B. So the derivations presented in Ray [34] and subsequent sections here are restricted to vertical spans that are small compared with the vertical density scale height $H_\rho \sim 9\,\mathrm{km}$ [1, 19, 35] and integration time scales longer than the particle relaxation time. Furthermore, Ray [34] considers either constant fluid velocity $\mathbf{u}$ or time-independent velocity that varies linearly only in the vertical coordinate, $\mathbf{u}(\mathbf{x},t) = \mathbf{u}(z) = [u_x(z), 0, 0]$ with $u_x(z) = \gamma z$.

2.2.2 Zero Initial Condition in Piecewise Spatially Uniform and Time-Independent Velocity: $\mathbf{w}_\mathrm{s}(0) = 0$ and $\mathbf{u}(\mathbf{x},t) = \mathbf{u}_{ik}$ for $\mathbf{x}, \mathbf{x}_\mathrm{p}(t) \in R_i$ and $t_k \leq t \leq t_{k+1}$

To solve the nonlinear dynamic equations, Norment [33] first divides the spatial domain $R = \cup_i R_i$ into disjoint regions R_i of piecewise uniform (albeit temporally varying) velocity such that within such a region

$\mathbf{x} \in R_i$, $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_i(t)$. Then provided that $\mathbf{x}_\mathrm{p}(t) \in R_i$, the spatial derivative terms in Eqs. 42 vanish, leaving

$$\frac{\mathrm{d}\mathbf{w}_\mathrm{s}}{\mathrm{d}t} = -\mathbf{w}_\mathrm{s}\phi\left(|\mathbf{w}_\mathrm{s}|\right) + \mathbf{f}_\mathrm{grav} - \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t}. \tag{46}$$

Then the second row of $\mathbf{Q}(\mathbf{X}_\mathrm{p}, t)$ in Eq. 45b becomes $\mathbf{Q}_2(\mathbf{X}_\mathrm{p}, t) = \mathbf{f}_\mathrm{grav} - \mathrm{d}\mathbf{u}_i/\mathrm{d}t$. Note that if $\mathbf{w}_\mathrm{s}(t)$ is sufficiently slowly varying such that $\mathrm{d}\mathbf{w}_\mathrm{s}/\mathrm{d}t \approx 0$ in comparison to the other acceleration terms, then the solution of Eq. 46, denoted as $\mathbf{w}_{\mathrm{s},\ell}$, is given by an algebraic equation:

$$\mathbf{w}_{\mathrm{s},\ell}\phi\left(|\mathbf{w}_{\mathrm{s},\ell}|\right) = \mathbf{f}_\mathrm{grav} - \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t}. \tag{47}$$

Now consider further the case of piecewise constant temporal functions $\mathbf{u}_i(t) = \mathbf{u}_{ik}$ for $t_k \leq t < t_{k+1}$ within each spatial region R_i. The formulation of the fluid velocity within these spatial regions is discussed in Section 3.4.1. Then for $t_k \leq t < t_{k+1}$, we have a constant velocity $\mathbf{u}$, and $\mathbf{Q}_2(\mathbf{X}_\mathrm{p}, t) = \mathbf{f}_\mathrm{grav} \Rightarrow \nabla_{\mathbf{w}_\mathrm{s}}\mathbf{Q}_2(\mathbf{X}_\mathrm{p}, t) = 0$. Thus, we can use the simple solution for constant $\mathbf{u}$ from Ray [34]. Furthermore, if the z axis is aligned with $\mathbf{f}_\mathrm{grav}$, Eq. 46 can be written as the set of equations:

$$\frac{\mathrm{d}w_{\mathrm{s},x}}{\mathrm{d}t} = -w_{\mathrm{s},x}(t)\phi\left(\sqrt{w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t) + w_{\mathrm{s},z}^2(t)}\right), \tag{48a}$$

$$\frac{\mathrm{d}w_{\mathrm{s},y}}{\mathrm{d}t} = -w_{\mathrm{s},y}(t)\phi\left(\sqrt{w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t) + w_{\mathrm{s},z}^2(t)}\right), \tag{48b}$$

$$\frac{\mathrm{d}w_{\mathrm{s},z}}{\mathrm{d}t} = -w_{\mathrm{s},z}(t)\phi\left(\sqrt{w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t) + w_{\mathrm{s},z}^2(t)}\right) - g. \tag{48c}$$

Equations 48 are equivalent to Ray [34] Eqs. 14 and 15, except with the additional horizontal y dimension and vertical z dimension.

2.2.2.1 Solution method of Norment [33]

With the initial condition $\mathbf{w}_\mathrm{s}(0) = 0$, the horizontal domain solutions are zero at all times $t_k \leq t < t_{k+1}$, $w_{\mathrm{s},x}(t) = w_{\mathrm{s},y}(t) = 0$, as can be seen from Ray [34] Eq. 25. Thus, it remains to only solve Eq. 48c, which is now an uncoupled, separable, nonlinear ODE:

$$\frac{\mathrm{d}w_{\mathrm{s},z}}{\mathrm{d}t} = -w_{\mathrm{s},z}(t)\phi\left(|w_{\mathrm{s},z}(t)|\right) - g. \tag{49}$$

First we will use the approach of [33]. Rearranging Eq. 49 by dividing both sides by the negative of the right-hand side, we have

$$\begin{aligned} \int_0^t \frac{1}{\alpha w_{\mathrm{s},z}(t')\,|w_{\mathrm{s},z}(t')| + g}\frac{\mathrm{d}w_{\mathrm{s},z}(t')}{\mathrm{d}t'}\mathrm{d}t' &= -\int_0^t \mathrm{d}t' \\ \Rightarrow \int_0^{w_{\mathrm{s},z}(t)} \frac{\mathrm{d}w_{\mathrm{s},z}}{\alpha w_{\mathrm{s},z}\,|w_{\mathrm{s},z}| + g} &= -t. \end{aligned} \tag{50}$$

Now if $w_{\mathrm{s},z}(t) > 0$, then $w_{\mathrm{s},z}\left|w_{\mathrm{s},z}\right| = w_{\mathrm{s},z}^2 > 0$ for $0 < w_{\mathrm{s},z} < w_{\mathrm{s},z}(t)$. On the other hand, if $w_{\mathrm{s},z}(t) < 0$, then $w_{\mathrm{s},z}\left|w_{\mathrm{s},z}\right| = -w_{\mathrm{s},z}^2 < 0$ for $w_{\mathrm{s},z}(t) < w_{\mathrm{s},z} < 0$. Thus, the integral in Eq. 50 becomes

$$\begin{aligned}
\int_0^{w_{\mathrm{s},z}(t)} \frac{\mathrm{d}w_{\mathrm{s},z}}{\alpha w_{\mathrm{s},z}\left|w_{\mathrm{s},z}\right| + g} &= \begin{cases} \int_0^{w_{\mathrm{s},z}(t)} \frac{\mathrm{d}w_{\mathrm{s},z}}{\alpha w_{\mathrm{s},z}^2 + g} & w_{\mathrm{s},z}(t) > 0 \\ \int_0^{w_{\mathrm{s},z}(t)} \frac{\mathrm{d}w_{\mathrm{s},z}}{g - \alpha w_{\mathrm{s},z}^2} & w_{\mathrm{s},z}(t) < 0 \end{cases} \\
&= \begin{cases} \left[\frac{v_\mathrm{f}}{g}\tan^{-1}\left(\frac{w_{\mathrm{s},z}}{v_\mathrm{f}}\right)\right]_0^{w_{\mathrm{s},z}(t)} & w_{\mathrm{s},z}(t) > 0 \\ \left[\frac{v_\mathrm{f}}{g}\tanh^{-1}\left(\frac{w_{\mathrm{s},z}}{v_\mathrm{f}}\right)\right]_0^{w_{\mathrm{s},z}(t)} & w_{\mathrm{s},z}(t) < 0 \wedge 0 < g - \alpha w_{\mathrm{s},z}^2(t) \end{cases} \\
&= \begin{cases} \tau_\mathrm{f}\tan^{-1}\left(\frac{w_{\mathrm{s},z}(t)}{v_\mathrm{f}}\right) & w_{\mathrm{s},z}(t) > 0 \\ \tau_\mathrm{f}\tanh^{-1}\left(\frac{w_{\mathrm{s},z}(t)}{v_\mathrm{f}}\right) & w_{\mathrm{s},z}(t) < 0 \wedge \left|w_{\mathrm{s},z}(t)\right| < v_\mathrm{f} \end{cases},
\end{aligned} \tag{51}$$

where

$$v_\mathrm{f} = \sqrt{\frac{g}{\alpha}} = \sqrt{\frac{2m_\mathrm{p}g}{\rho C_\mathrm{D}A_\mathrm{D,p}}} = \frac{1}{d_\mathrm{p}}\sqrt{\frac{8m_\mathrm{p}g}{\pi\rho C_\mathrm{D}}} = \frac{1}{d_\mathrm{p}}\sqrt{\frac{8\rho_\mathrm{p}V_\mathrm{p}g}{\pi\rho C_\mathrm{D}}} = 2\sqrt{\frac{1}{3}\left(\frac{\rho_\mathrm{p}}{\rho}\right)\frac{d_\mathrm{p}g}{C_\mathrm{D}}}, \tag{52a}$$

$$\tau_\mathrm{f} = \frac{v_\mathrm{f}}{g} = \sqrt{\frac{1}{g\alpha}} = \sqrt{\frac{2m_\mathrm{p}}{\rho g C_\mathrm{D}A_\mathrm{D,p}}} = \frac{1}{d_\mathrm{p}}\sqrt{\frac{8m_\mathrm{p}}{\pi\rho g C_\mathrm{D}}} = \frac{1}{d_\mathrm{p}}\sqrt{\frac{8\rho_\mathrm{p}V_\mathrm{p}}{\pi\rho g C_\mathrm{D}}} = 2\sqrt{\frac{1}{3}\left(\frac{\rho_\mathrm{p}}{\rho}\right)\frac{d_\mathrm{p}}{gC_\mathrm{D}}}, \tag{52b}$$

are the fall velocity and time scale, respectively. The former represents the steady-state relative velocity of Eq. 49, and the latter is the time scale of approaching this velocity. Figure 3 shows both the fall velocity and time scales as well as a fall length scale defined as $\Delta z_\mathrm{f} = v_\mathrm{f}\tau_\mathrm{f}$ as a function of particle diameter d_p for various particle density ratios ρ_p/ρ. The drag coefficients used in Eqs. 52 are calculated using Eq. 223 with the Archimedes number given by Eq. 221. See Appendix B where the terminal velocity v_t of particles is explained in more detail and shown in Figure 20. From Figure 3, it is apparent that even for the largest particles, the velocity, time, and length scales are at most on the order of $v_\mathrm{f} \lesssim 20\,\mathrm{m\,s^{-1}}$, $\tau_\mathrm{f} \lesssim 2\,\mathrm{s}$, and $\Delta z_\mathrm{f} \lesssim 40\,\mathrm{m}$, respectively, but they are typically on the order of $v_\mathrm{f} \sim 1\,\mathrm{m\,s^{-1}}$, $\tau_\mathrm{f} \sim 100\,\mathrm{ms}$, and $\Delta z_\mathrm{f} \sim 0.1\,\mathrm{m}$. Thus, meteorological spatial cells of reasonable size and duration should permit the fallout particles to reach steady state without issue, as discussed in Section 2.2.2.3.

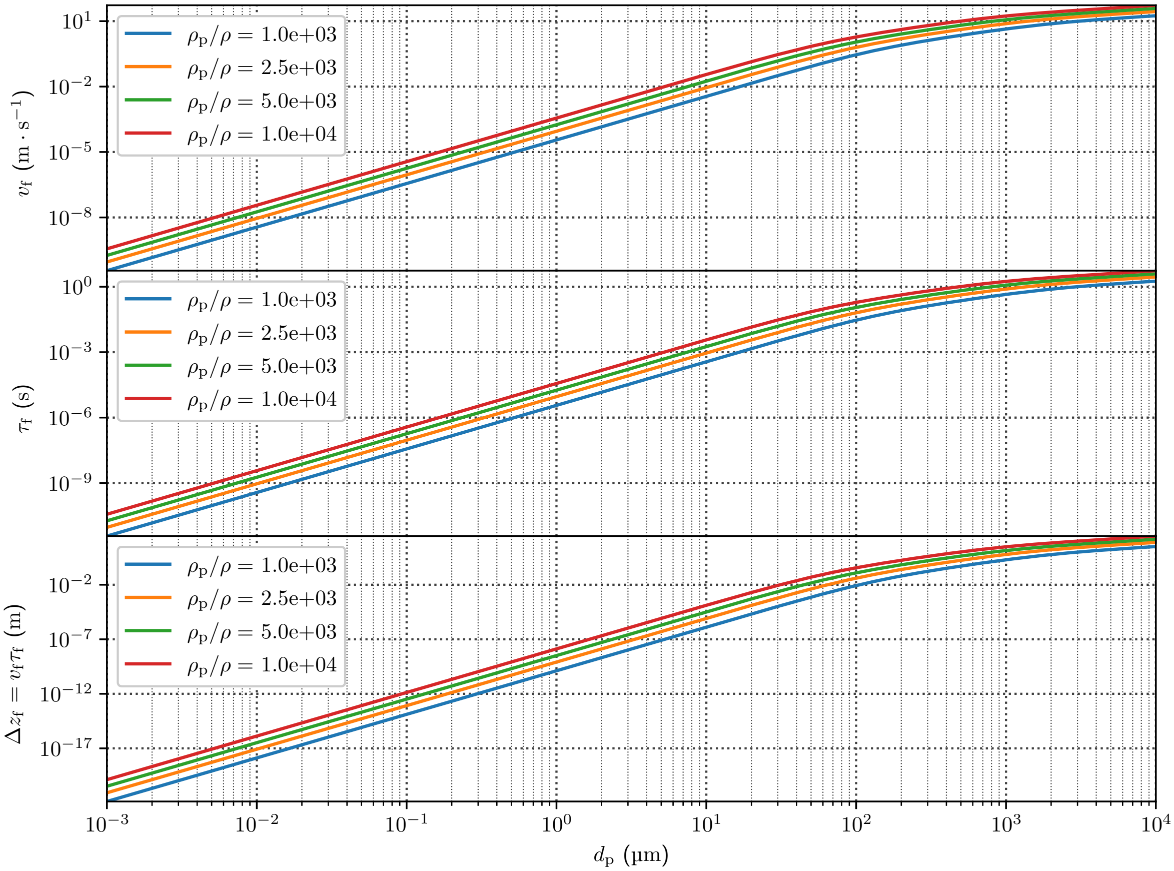

Figure 3. Fall velocity v_f, time τ_f, and length Δz_f scales for solid spherical particles of mass density ratio $\rho_\mathrm{p}/\rho \in \{1, 2.5, 5, 10\} \times 10^3$ and varying diameter $10^{-3}\,\mu\mathrm{m} \le d_\mathrm{p} \le 10^4\,\mu\mathrm{m}$.

Then substituting Eqs. 51 into Eq. 50, we obtain the solutions

$$w_{\mathrm{s},z}(t) = -v_\mathrm{f} \begin{cases} \tan\left(\frac{t}{\tau_\mathrm{f}}\right) & w_{\mathrm{s},z}(t) > 0 \\ \tanh\left(\frac{t}{\tau_\mathrm{f}}\right) & w_{\mathrm{s},z}(t) < 0 \land |w_{\mathrm{s},z}(t)| < v_\mathrm{f} \end{cases} .$$

Because $t > 0$, $\tau_\mathrm{f} > 0$, and $v_\mathrm{f} > 0$, $w_{\mathrm{s},z}(t) < 0$, so we must reject the first solution, meaning that the proper solution is

$$w_{\mathrm{s},z}(t) = -v_\mathrm{f} \tanh\left(\frac{t}{\tau_\mathrm{f}}\right), \tag{53}$$

which is in agreement with Norment [33] Eq. 22. However, Norment [33] did not quite as rigorously derive this or place any constraints on the bounds of integration due to the singularities of the inverse hyperbolic tangent function $\tanh^{-1} x$ at $x = \pm 1$. Note, the constraint $|w_{\mathrm{s},z}(t)| < v_\mathrm{f}$ is indeed properly satisfied by Eq. 53.

2.2.2.2 Solution method of Ray [34]

On the other hand, using the approach of Ray [34], the solution to Eq. 48c is represented using Ray [34] Eq. 26 as

$$w_{\mathrm{s},z}(t) = \frac{1}{\varphi(t)}\left(w_{\mathrm{s},z,0} - g\Phi(t)\right) = -\frac{1}{\varphi(t)}g\Phi(t), \tag{54}$$

where

$$\varphi(t) \equiv \exp\left(\alpha \int_0^t \sqrt{w_{\mathrm{s},x}^2(t') + w_{\mathrm{s},y}^2(t') + w_{\mathrm{s},z}^2(t')}\mathrm{d}t'\right), \tag{55a}$$

$$\Phi(t) \equiv \int_0^t \varphi(t')\mathrm{d}t', \tag{55b}$$

and $\Phi(t)$ satisfies the differential Eq. 29,

$$\frac{\mathrm{d}^2\Phi}{\mathrm{d}t^2} = \alpha\sqrt{w_{\mathrm{s},x,0}^2 + w_{\mathrm{s},y,0}^2 + \left(w_{\mathrm{s},z,0} - g\Phi(t)\right)^2} = \alpha\left|g\Phi(t)\right| = \alpha g\Phi(t), \tag{56}$$

with initial conditions $\Phi(0) = 0$ and $\Phi'(0) = 1$. Note, from the definition of Eqs. 55, and because $\alpha, w_{\mathrm{s},x}(t), w_{\mathrm{s},y}(t), w_{\mathrm{s},z}(t) \in \mathbb{R}$, it is apparent that $\Phi(t) > 0$, allowing us to move $\Phi(t)$ outside the absolute value in Eq. 56. To solve Eq. 56, we multiply both sides by $2\mathrm{d}\Phi/\mathrm{d}t$, use the product rule to form a total differential on both sides, integrate, and then to solve for $\mathrm{d}\Phi/\mathrm{d}t$:

$$\begin{aligned}
2\frac{\mathrm{d}\Phi}{\mathrm{d}t}\frac{\mathrm{d}^2\Phi}{\mathrm{d}t^2} &= 2\alpha g\frac{\mathrm{d}\Phi}{\mathrm{d}t}\Phi(t) \\
\Rightarrow \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}\Phi}{\mathrm{d}t}\right)^2 &= \alpha g\frac{\mathrm{d}}{\mathrm{d}t}\Phi^2 \\
\Rightarrow \int_0^t \frac{\mathrm{d}}{\mathrm{d}t'}\left(\frac{\mathrm{d}\Phi}{\mathrm{d}t'}\right)^2\mathrm{d}t' &= \alpha g\int_0^t \frac{\mathrm{d}}{\mathrm{d}t'}\Phi^2(t')\mathrm{d}t' \\
\Rightarrow \left[\left(\frac{\mathrm{d}\Phi}{\mathrm{d}t'}\right)^2\right]_0^t &= \alpha g\left[\Phi^2(t')\right]_0^t \\
\Rightarrow \left(\frac{\mathrm{d}\Phi}{\mathrm{d}t}\right)^2 - \left[\Phi'(0)\right]^2 &= \alpha g\left[\Phi^2(t) - \Phi^2(0)\right] \\
\Rightarrow \frac{\mathrm{d}\Phi}{\mathrm{d}t} &= \sqrt{\alpha g\left[\Phi^2(t) - \Phi^2(0)\right] + \left[\Phi'(0)\right]^2}.
\end{aligned}$$

Now we can integrate again to solve for $\Phi(t)$:

$$\begin{aligned}
\Rightarrow \int_0^t \frac{1}{\sqrt{\Phi^2(t') - \Phi^2(0) + [\tau_\mathrm{f}\Phi'(0)]^2}} \frac{\mathrm{d}\Phi}{\mathrm{d}t'}\mathrm{d}t' &= \tau_\mathrm{f}^{-1} \int_0^t \mathrm{d}t' \\
\Rightarrow \int_{\Phi(0)}^{\Phi(t)} \frac{\mathrm{d}\Phi}{\sqrt{\Phi^2 - \Phi^2(0) + [\tau_\mathrm{f}\Phi'(0)]^2}} &= \frac{t}{\tau_\mathrm{f}} \\
\Rightarrow \left[\ln\left(\Phi + \sqrt{\Phi^2 - \Phi^2(0) + [\tau_\mathrm{f}\Phi'(0)]^2}\right)\right]_{\Phi(0)}^{\Phi(t)} &= \frac{t}{\tau_\mathrm{f}} \\
\Rightarrow \ln\left(\frac{\Phi(t) + \sqrt{\Phi^2(t) - \Phi^2(0) + [\tau_\mathrm{f}\Phi'(0)]^2}}{\Phi(0) + \tau_\mathrm{f}\Phi'(0)}\right) &= \frac{t}{\tau_\mathrm{f}} \\
\Rightarrow \sqrt{\Phi^2(t) - \Phi^2(0) + [\tau_\mathrm{f}\Phi'(0)]^2} &= \left[\Phi(0) + \tau_\mathrm{f}\Phi'(0)\right] \exp\left(\frac{t}{\tau_\mathrm{f}}\right) - \Phi(t) \\
\Rightarrow \Phi^2(t) - \Phi^2(0) + \left[\tau_\mathrm{f}\Phi'(0)\right]^2 &= \left[\Phi(0) + \tau_\mathrm{f}\Phi'(0)\right] \exp\left(\frac{t}{\tau_\mathrm{f}}\right) \\
&\quad \times \left\{\left[\Phi(0) + \tau_\mathrm{f}\Phi'(0)\right] \exp\left(\frac{t}{\tau_\mathrm{f}}\right) - 2\Phi(t)\right\} + \Phi^2(t) \\
\Rightarrow \Phi(t) &= \frac{1}{2}\left[\Phi(0) + \tau_\mathrm{f}\Phi'(0)\right] \exp\left(\frac{t}{\tau_\mathrm{f}}\right) \\
&\quad - \frac{1}{2}\left(\frac{[\tau_\mathrm{f}\Phi'(0)]^2 - \Phi^2(0)}{\Phi(0) + \tau_\mathrm{f}\Phi'(0)}\right) \exp\left(-\frac{t}{\tau_\mathrm{f}}\right) \\
&= \frac{\Phi(0)}{2}\left[\exp\left(\frac{t}{\tau_\mathrm{f}}\right) + \exp\left(-\frac{t}{\tau_\mathrm{f}}\right)\right] \\
&\quad + \frac{\tau_\mathrm{f}\Phi'(0)}{2}\left[\exp\left(\frac{t}{\tau_\mathrm{f}}\right) - \exp\left(-\frac{t}{\tau_\mathrm{f}}\right)\right] \\
&= \Phi(0)\cosh\left(\frac{t}{\tau_\mathrm{f}}\right) + \tau_\mathrm{f}\Phi'(0)\sinh\left(\frac{t}{\tau_\mathrm{f}}\right) \\
&= \tau_\mathrm{f}\sinh\left(\frac{t}{\tau_\mathrm{f}}\right). \qquad (57)
\end{aligned}$$

Substituting Eq. 57 into Eq. 54 with $\varphi(t) = \mathrm{d}\Phi(t)/\mathrm{d}t = \cosh(t/\tau_\mathrm{f})$, we find

$$w_{\mathrm{s},z}(t) = -g\tau_\mathrm{f}\frac{\sinh(t/\tau_\mathrm{f})}{\cosh(t/\tau_\mathrm{f})} = -v_\mathrm{f}\tanh\left(\frac{t}{\tau_\mathrm{f}}\right), \qquad (58)$$

which is consistent with Eq. 53.

2.2.2.3 Particle velocity and displacement analysis

Because in the horizontal domain $w_{\mathrm{s},x}(t) = w_{\mathrm{s},y}(t) = 0$ for $t_k \leq t < t_{k+1}$ with initial condition $\mathbf{w}_\mathrm{s}(0) = 0$, the horizontal trajectory is simply given by

$$v_{\mathrm{p},j}(t) = u_{ik,j}, \qquad (59a)$$

$$x_{\mathrm{p},j}(t) - x_{\mathrm{p},j}(0) = u_{ik,j}t. \qquad (59b)$$

In the vertical dimension, substituting Eq. 53 or Eq. 58 into Eq. 40, we find the particle vertical velocity is given by

$$\begin{aligned} v_{\mathrm{p},z}(t) &= w_{\mathrm{s},z}(t) + u_{ik,z} \\ &= -v_{\mathrm{f}} \tanh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + u_{ik,z}, \end{aligned} \tag{60}$$

which has a value of zero at $0 < t = \tau_{\mathrm{f}} \tanh^{-1}(u_{ik,z}/v_{\mathrm{f}})$ if $u_{ik,z} > 0$, indicating a change in particle direction of travel. Substituting Eq. 60 into Eq. 43 and integrating both sides, the particle vertical displacement within the region R_i for $t_k \leq t < t_{k+1}$ is given by

$$\begin{aligned} z_{\mathrm{p}}(t) - z_{\mathrm{p}}(0) &= \int_0^t w_{\mathrm{s},z}(t')\mathrm{d}t' + u_{ik,z}t \\ &= -v_{\mathrm{f}}\tau_{\mathrm{f}} \left[\ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right)\right]_0^t + u_{ik,z}t \\ &= -\Delta z_{\mathrm{f}} \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + u_{ik,z}t, \end{aligned} \tag{61}$$

where $z_{\mathrm{p}}(0)$ is the particle initial vertical position, and $\Delta z_{\mathrm{f}} = v_{\mathrm{f}}\tau_{\mathrm{f}}$ is the fall length scale. The first component of Eq. 61 is the displacement solely due to the relative particle and fluid velocity, while the latter is the displacement due solely to fluid velocity. Asymptotically, Eq. 61 takes the following form:

$$\begin{aligned} z_{\mathrm{p}}(t) - z_{\mathrm{p}}(0) &= \begin{cases} u_{ik,z}\tau_{\mathrm{f}}\left(\frac{t}{\tau_{\mathrm{f}}}\right) - \frac{\Delta z_{\mathrm{f}}}{2}\left(\frac{t}{\tau_{\mathrm{f}}}\right)^2 + O\left(\left(\frac{t}{\tau_{\mathrm{f}}}\right)^4\right) & t/\tau_{\mathrm{f}} \ll 1 \\ \left(u_{ik,z} - v_{\mathrm{f}}\right)t & t/\tau_{\mathrm{f}} \gg 1 \end{cases} \\ &= \begin{cases} \left[u_{ik,z} - \frac{v_{\mathrm{f}}}{2}\left(\frac{t}{\tau_{\mathrm{f}}}\right)\right]\tau_{\mathrm{f}}\left(\frac{t}{\tau_{\mathrm{f}}}\right) + O\left(\left(\frac{t}{\tau_{\mathrm{f}}}\right)^4\right) & t/\tau_{\mathrm{f}} \ll 1 \\ \left(u_{ik,z} - v_{\mathrm{f}}\right)t & t/\tau_{\mathrm{f}} \gg 1 \end{cases}. \end{aligned} \tag{62}$$

Looking at Eq. 62 for $t \ll \tau_{\mathrm{f}}$, we can see that in addition to the zero at $t = 0$, the vertical displacement is zero when $0 < t/\tau_{\mathrm{f}} \approx 2u_{ik,z}/v_{\mathrm{f}} \ll 1$, which is due to the change in direction particle for small times when $u_{ik,z} > 0$. More generally, there will be a zero in the vertical displacement for $t > 0$ when $0 < u_{ik,z} < v_{\mathrm{f}}$. To first order at the smallest times, the displacement is linearly proportional to the fluid velocity $u_{ik,z}$. For large times $t \gg \tau_{\mathrm{f}}$, the particle will be vertically displaced linearly with the velocity $u_{ik,z} - v_{\mathrm{f}}$.

The condition that the relative velocity approximately reaches its steady state within the region R_i corresponding to the initial particle position in time $t_k \leq t \leq t_{k+1}$ is given by

$$\begin{aligned} & z_i \leq z_{\mathrm{p}}(t) \leq z_{i+1} \\ \Rightarrow\ & 0 \leq z_{\mathrm{p}}(t) - z_i \leq \Delta z_i \\ \Rightarrow\ & 0 \leq -\Delta z_{\mathrm{f}} \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + u_{ik,z}t + z_{\mathrm{p}}(0) - z_i \leq \Delta z_i, \end{aligned}$$

where $\Delta z_i = z_{i+1} - z_i$ is the vertical span of R_i. Since $z_i \leq z_{\mathrm{p}}(0) \leq z_{i+1} \Rightarrow 0 \leq z_{\mathrm{p}}(0) - z_i \leq \Delta z_i \Rightarrow 0 \leq (z_{\mathrm{p}}(0) - z_i)/\Delta z_i \leq 1$, we can define

$z_{\mathrm{p}}(0) - z_i = \Delta\tilde{z}_{\mathrm{p},0,i}\Delta z_i$ with $0 \leq \Delta\tilde{z}_{\mathrm{p},0,i} \equiv (z_{\mathrm{p}}(0) - z_i)/\Delta z_i \leq 1$. Then we have

$$\begin{aligned}
0 &\leq \frac{1}{\Delta z_i}\left[-\Delta z_{\mathrm{f}} \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + u_{ik,z}t\right] + \Delta\tilde{z}_{\mathrm{p},0,i} \leq 1 \\
\Rightarrow -\frac{1}{2} &\leq \frac{1}{\Delta z_i}\left[-\Delta z_{\mathrm{f}} \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + u_{ik,z}t\right] + \Delta\tilde{z}_{\mathrm{p},0,i} - \frac{1}{2} \leq \frac{1}{2} \\
\Rightarrow &\left|\frac{1}{\Delta z_i}\left[-\Delta z_{\mathrm{f}} \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + u_{ik,z}t\right] + \Delta\tilde{z}_{\mathrm{p},0,i} - \frac{1}{2}\right| \leq \frac{1}{2} \\
\Rightarrow &\left|\frac{\Delta z_{\mathrm{f}}}{\Delta z_i}\left[-\ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + \frac{u_{ik,z}}{v_{\mathrm{f}}}\left(\frac{t}{\tau_{\mathrm{f}}}\right)\right] + \Delta\tilde{z}_{\mathrm{p},0,i} - \frac{1}{2}\right| \leq \frac{1}{2}.
\end{aligned}$$

If we require that $\Delta z_{\mathrm{f}} < \Delta z_i \Rightarrow \Delta z_{\mathrm{f}}/\Delta z_i \equiv \Delta\tilde{z}_{\mathrm{f},i} < 1$ to ensure that the region size is sufficiently large that the fall length scale will fit within the region, then we require for $0 < \Delta\tilde{z}_{\mathrm{f},i} < 1$ that

$$\left|\Delta\tilde{z}_{\mathrm{p},0,i} - \frac{1}{2} + \Delta\tilde{z}_{\mathrm{f},i}\left[-\ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + \frac{u_{ik,z}}{v_{\mathrm{f}}}\left(\frac{t}{\tau_{\mathrm{f}}}\right)\right]\right| \leq \frac{1}{2}. \tag{63}$$

We can see that Eq. 63 is more likely to be satisfied for larger times if $\Delta\tilde{z}_{\mathrm{f},i} \ll 1$, which is to say that the region R_i is large relative to the fall length scale. If $u_{ik,z} < 0$, then the particle will only move downward, and it is more likely for Eq. 63 to be satisfied if $\Delta\tilde{z}_{\mathrm{p},0,i} \sim 1$ and $|u_{ik,z}|/v_{\mathrm{f}} \ll 1$, which ensures that the particle initial position is far from the region's exit boundary. If $u_{ik,z} > 0$, the constraint is more complicated and will also depend on the velocity ratio $u_{ik,z}/v_{\mathrm{f}}$ because the particle changes directions. In particular, it is more likely satisfied if

$$\begin{aligned}
-\ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + \frac{u_{ik,z}}{v_{\mathrm{f}}}\left(\frac{t}{\tau_{\mathrm{f}}}\right) &= 0 \\
\Rightarrow \frac{u_{ik,z}}{v_{\mathrm{f}}} &= \left(\frac{t}{\tau_{\mathrm{f}}}\right)^{-1} \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) \\
&= 1, \quad \frac{t}{\tau_{\mathrm{f}}} \gg 1,
\end{aligned}$$

which is to say $u_{ik,z} \sim v_{\mathrm{f}}$, as long as $\Delta\tilde{z}_{\mathrm{p},0,i}$ is not too close to unity such that there is insufficient space in the region R_i in the positive direction. Given the results of Figure 3, the fall velocity time scale will be large for larger, more dense particles. For such particles, Eq. 63 is thus more likely satisfied if $u_{ik,z} < 0$ and $\Delta\tilde{z}_{\mathrm{p},0,i} \sim 1$ because $|u_{ik,z}|/v_{\mathrm{f}} \ll 1$ is more likely than $0 < u_{ik,z} \sim v_{\mathrm{f}}$ for larger v_{f}, provided that vertical wind speeds are low to moderate, $|u_{ik,z}| \lesssim 5\,\mathrm{m\,s^{-1}}$. Conversely, for smaller, less dense particles, Eq. 63 is probably more likely satisfied if $0 < u_{ik,z} \sim v_{\mathrm{f}}$ and $\Delta\tilde{z}_{\mathrm{p},0,i} \sim 1/2$ for low to moderate vertical wind speeds. More generally for $|u_{ik,z}| \gtrsim v_{\mathrm{f}}$, it is more likely for the particle to remain within R_i if $\Delta\tilde{z}_{\mathrm{f},ik} = (v_{\mathrm{f}} + |u_{ik,z}|)\,\tau_{\mathrm{f}}/\Delta z_i \ll 1$.

In summary, combining the results of Eqs. 59b and 63, the spatial extents $(\Delta x_i, \Delta y_i, \Delta z_i)$ of the fluid region $\mathbf{x}, \mathbf{x}_{\mathrm{p}}(t) \in R_i$ with uniform velocity $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_i(t)$ should be sufficiently large:

$$\Delta\tilde{x}_{\mathrm{f},ik} = \frac{\Delta x_{\mathrm{f},ik}}{\Delta x_i} = \frac{|u_{ik,x}|\,\tau_{\mathrm{f}}}{\Delta x_i} \ll 1, \tag{64a}$$

$$\Delta\tilde{y}_{\mathrm{f},ik} = \frac{\Delta y_{\mathrm{f},ik}}{\Delta y_i} = \frac{|u_{ik,y}|\,\tau_{\mathrm{f}}}{\Delta y_i} \ll 1, \tag{64b}$$

$$\Delta\tilde{z}_{\mathrm{f},ik} = \frac{\Delta z_{\mathrm{f},ik}}{\Delta z_i} = \frac{(v_{\mathrm{f}} + |u_{ik,z}|)\,\tau_{\mathrm{f}}}{\Delta z_i} = \frac{\Delta z_{\mathrm{f}} + |u_{ik,z}|\,\tau_{\mathrm{f}}}{\Delta z_i} \ll 1. \tag{64c}$$

Equations 64 allow particles to approximately reach steady-state velocity within the region R_i during time period $t_k \le t \le t_{k+1}$. In the case that $|u_{ik,z}| \ll v_\mathrm{f}$, Eq. 64c reduces to $\Delta\tilde{z}_{\mathrm{f},i} = \Delta z_\mathrm{f}/\Delta z_i = v_\mathrm{f}\tau_\mathrm{f}/\Delta z_i \ll 1$. As noted in Figure 3 in Section 2.2.2.1, even for the largest and densest particles, the time and length scales are $\tau_\mathrm{f} \sim 2\,\mathrm{s}$ and $\Delta z_\mathrm{f} \sim 40\,\mathrm{m}$, respectively. Then for a horizontal velocity of magnitude $|u_{ik,j}| \sim 10\,\mathrm{m\,s^{-1}}$, we have $\Delta x_{\mathrm{f},ik}, \Delta y_{\mathrm{f},ik} \sim 20\,\mathrm{m}$, whereas typical reanalysis meteorological data are known on a horizontal grid scale of $\Delta x_i, \Delta y_i \gtrsim 10\,\mathrm{km}$. In the vertical dimension, higher resolution reanalysis meteorological data might have a vertical scale $\Delta z_i \sim \Delta z_\mathrm{f} \sim 25\,\mathrm{m}$ for the lower model levels up to, for example, $z \sim 1\,\mathrm{km}$, but lower resolution meteorological data might be sufficiently large above $z \sim 150\,\mathrm{m}$. Even for low resolution meteorological data, though, it is common to have a $10\,\mathrm{m}$ level velocity. Thus, Eqs. 64a and 64b are generally satisfied for most common meteorological data, whereas Eq. 64c will likely only be satisfied above $z \sim 1\,\mathrm{km}$ or $z \sim 150\,\mathrm{m}$, depending on the data set.

2.2.3 Small (but Nonzero) Horizontal Relative Velocity: $w_{\mathrm{s},xy}^2(t) = w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t) \ll v_\mathrm{f}^2$

Again aligning the z axis with $\mathbf{f}_\mathrm{grav}$, Eq. 46 can be written as the set of equations

$$\frac{\mathrm{d}w_{\mathrm{s},x}}{\mathrm{d}t} = -w_{\mathrm{s},x}(t)\phi\left(\sqrt{w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t) + w_{\mathrm{s},z}^2(t)}\right) - \frac{\mathrm{d}u_{i,x}}{\mathrm{d}t}, \tag{65a}$$

$$\frac{\mathrm{d}w_{\mathrm{s},y}}{\mathrm{d}t} = -w_{\mathrm{s},y}(t)\phi\left(\sqrt{w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t) + w_{\mathrm{s},z}^2(t)}\right) - \frac{\mathrm{d}u_{i,y}}{\mathrm{d}t}, \tag{65b}$$

$$\frac{\mathrm{d}w_{\mathrm{s},z}}{\mathrm{d}t} = -w_{\mathrm{s},z}(t)\phi\left(\sqrt{w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t) + w_{\mathrm{s},z}^2(t)}\right) - g - \frac{\mathrm{d}u_{i,z}}{\mathrm{d}t}. \tag{65c}$$

In the case that the horizontal relative velocity components $w_{\mathrm{s},xy}^2(t) = w_{\mathrm{s},x}^2(t) + w_{\mathrm{s},y}^2(t)$ are small compared with the vertical fall velocity scale v_f—i.e., $w_{\mathrm{s},xy}^2(t) \ll v_\mathrm{f}^2$—we can perform a Taylor series expansion of $|\mathbf{w}_\mathrm{s}|$ around $w_{\mathrm{s},xy}^2(t) = 0$:

$$\begin{aligned}
\phi\left(|\mathbf{w}_\mathrm{s}|\right) &= \phi\left(\sqrt{w_{\mathrm{s},xy}^2(t) + w_{\mathrm{s},z}^2(t)}\right) \\
&= \phi\left(\sqrt{w_{\mathrm{s},xy}^2(t) + v_\mathrm{f}^2 \tanh^2\left(\frac{t}{\tau_\mathrm{f}}\right)}\right) \\
&= \phi\left[v_\mathrm{f}\tanh\left(\frac{t}{\tau_\mathrm{f}}\right) + \frac{v_\mathrm{f}}{2}\coth^2\left(\frac{t}{\tau_\mathrm{f}}\right)\left(\frac{w_{\mathrm{s},xy}(t)}{v_\mathrm{f}}\right)^2 + O\left(\left(\frac{w_{\mathrm{s},xy}(t)}{v_\mathrm{f}}\right)^4\right)\right].
\end{aligned}$$

To first order then,

$$\begin{aligned}
\phi\left(|\mathbf{w}_\mathrm{s}|\right) &\approx \phi\left[v_\mathrm{f}\tanh\left(\frac{t}{\tau_\mathrm{f}}\right)\right] \\
&= \alpha v_\mathrm{f}\tanh\left(\frac{t}{\tau_\mathrm{f}}\right) \\
&= \tau_\mathrm{f}^{-1}\tanh\left(\frac{t}{\tau_\mathrm{f}}\right).
\end{aligned} \tag{66}$$

Thus, using the approximation of Eq. 66, Eqs. 65a and 65b are approximated to first order as

$$\frac{\mathrm{d}w_{\mathrm{s},j}}{\mathrm{d}t} = -\tau_\mathrm{f}^{-1} w_{\mathrm{s},j}(t)\tanh\left(\frac{t}{\tau_\mathrm{f}}\right) - \frac{\mathrm{d}u_{i,j}}{\mathrm{d}t} \tag{67}$$

for $j \in \{x, y\}$. Equations 67 are first-order, linear, inhomogeneous, ODEs that can be solved using an integrating factor. First multiply both sides of Eqs. 67 by $\zeta(t)$, and then require that the homogeneous terms form a perfect differential:

$$\zeta(t)\frac{\mathrm{d}w_{\mathrm{s},j}}{\mathrm{d}t} + \zeta(t)P(t)w_{\mathrm{s},j}(t) = \zeta(t)Q_{i,j}(t), \tag{68a}$$

$$\zeta(t)\frac{\mathrm{d}w_{\mathrm{s},j}}{\mathrm{d}t} + w_{\mathrm{s},j}(t)\frac{\mathrm{d}\zeta}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\zeta w_{\mathrm{s},j}\right), \tag{68b}$$

where

$$P(t) = \phi\left(|\mathbf{w}_{\mathrm{s}}|\right) = \tau_{\mathrm{f}}^{-1}\tanh\left(\frac{t}{\tau_{\mathrm{f}}}\right), \tag{69a}$$

$$Q_{i,j}(t) = -\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t}. \tag{69b}$$

Comparing the left-hand side of Eqs. 68 requires that the integrating factor be given by

$$\begin{aligned}
\frac{\mathrm{d}\zeta}{\mathrm{d}t} &= \zeta(t)P(t) \\
\Rightarrow \int_0^t \frac{1}{\zeta(t')}\frac{\mathrm{d}\zeta}{\mathrm{d}t'}\mathrm{d}t' &= \int_0^t P(t')\mathrm{d}t' \\
\Rightarrow \int_{\zeta(0)}^{\zeta(t)} \frac{1}{\zeta}\mathrm{d}\zeta &= \tau_{\mathrm{f}}^{-1}\left[\tau_{\mathrm{f}}\ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right)\right]_0^t \\
\Rightarrow \left[\ln\zeta\right]_{\zeta(0)}^{\zeta(t)} &= \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) \\
\Rightarrow \ln\left(\frac{\zeta(t)}{\zeta(0)}\right) &= \ln\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right) \\
\Rightarrow \zeta(t) &= \zeta(0)\cosh\left(\frac{t}{\tau_{\mathrm{f}}}\right).
\end{aligned} \tag{70}$$

Then using the integrating factor Eq. 70 with the right-hand side of Eqs. 68, we find

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\left(\zeta w_{\mathrm{s},j}\right) &= \zeta(t)Q_{i,j}(t) \\
\Rightarrow \int_0^t \frac{\mathrm{d}}{\mathrm{d}t'}\left(\zeta w_{\mathrm{s},j}\right)\mathrm{d}t' &= \int_0^t \zeta(t')Q_{i,j}(t')\mathrm{d}t' \\
\Rightarrow \int_{\zeta(0)w_{\mathrm{s},j}(0)}^{\zeta(t)w_{\mathrm{s},j}(t)} \mathrm{d}\left(\zeta w_{\mathrm{s},j}\right) &= \int_0^t \zeta(t')Q_{i,j}(t')\mathrm{d}t' \\
\Rightarrow \left[\zeta w_{\mathrm{s},j}\right]_{\zeta(0)w_{\mathrm{s},j}(0)}^{\zeta(t)w_{\mathrm{s},j}(t)} &= \int_0^t \zeta(t')Q_{i,j}(t')\mathrm{d}t' \\
\Rightarrow w_{\mathrm{s},j}(t) &= \frac{1}{\zeta(t)}\left(\zeta(0)w_{\mathrm{s},j}(0) + \int_0^t \zeta(t')Q_{i,j}(t')\mathrm{d}t'\right).
\end{aligned} \tag{71}$$

Substituting Eq. 69b, the (negative) right-hand side integral in Eq. 71 can be integrated using integration by parts, noting that $\int_0^t \zeta(t')\mathrm{d}t' = \zeta(0)\tau_{\mathrm{f}}\left[\sinh\left(t'/\tau_{\mathrm{f}}\right)\right]_0^t = \zeta(0)\tau_{\mathrm{f}}\sinh\left(t/\tau_{\mathrm{f}}\right)$:

$$\begin{aligned}
\int_0^t \zeta(t')\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t'}\mathrm{d}t' &= \left[\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t'}\int_0^{t'}\zeta(t'')\mathrm{d}t''\right]_0^t - \int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\int_0^s \zeta(s')\mathrm{d}s'\mathrm{d}s \\
&= \zeta(0)\tau_{\mathrm{f}}\left[\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t}\right]_{t'=t}\sinh\left(\frac{t}{\tau_{\mathrm{f}}}\right) - \zeta(0)\tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\sinh\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s.
\end{aligned} \tag{72}$$

Then substituting Eq. 72 into Eq. 71, we have the solution for the horizontal relative velocity components

$$\begin{aligned} w_{\mathrm{s},j}(t) &= \operatorname{sech}\left(\frac{t}{\tau_{\mathrm{f}}}\right)\left[w_{\mathrm{s},j}(0) - \tau_{\mathrm{f}}\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t}\sinh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + \tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\sinh\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s\right] \\ &= -\tau_{\mathrm{f}}\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t}\tanh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + \operatorname{sech}\left(\frac{t}{\tau_{\mathrm{f}}}\right)\left[w_{\mathrm{s},j}(0) + \tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\sinh\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s\right]. \end{aligned} \tag{73}$$

Asymptotically, Eqs. 73 have the forms

$$w_{\mathrm{s},j}(t) = \begin{cases} w_{\mathrm{s},j}(0) - \tau_{\mathrm{f}}\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t}\left(\frac{t}{\tau_{\mathrm{f}}}\right) & t/\tau_{\mathrm{f}} \ll 1 \\ -\tau_{\mathrm{f}}\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t} + \exp\left(-\frac{t}{\tau_{\mathrm{f}}}\right)\left[w_{\mathrm{s},j}(0) + \tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\exp\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s\right] & t/\tau_{\mathrm{f}} \gg 1 \end{cases}. \tag{74}$$

Note that Eqs. 74 for $t/\tau_{\mathrm{f}} \gg 1$ differ from Norment [33] Eq. 30 in that Eqs. 74 do not contain a $[\mathrm{d}u_{i,j}/\mathrm{d}t']_{t'=0}$ term. This difference is likely due to what looks like an improper evaluation of the integration by parts at $t' = 0$ in Norment [33] Eq. 29. In particular, the $[\mathrm{d}u_{i,j}/\mathrm{d}t']_{t'=0}$ term should not appear due to the integral $\int_0^{t'} \zeta(t'')\mathrm{d}t''$* vanishing when being evaluated at $t' = 0$ in the first part of Eq. 72. In any case, we do arrive at the same leading order term $-\tau_{\mathrm{f}}\mathrm{d}u_{i,j}/\mathrm{d}t$, which dominates for $t/\tau_{\mathrm{f}} \gg 1$ since the exponential is vanishingly small.

The vertical relative velocity counterpart to Eq. 71 can be solved in much the same fashion, except here $Q_{i,z}(t) = -g - \mathrm{d}u_{i,z}/\mathrm{d}t$. Changing $Q_{i,j} \to Q_{i,z}$ in Eq. 71 then yields the (negative) integral

$$\begin{aligned} \int_0^t \zeta(t')\left(g + \frac{\mathrm{d}u_{i,z}}{\mathrm{d}t'}\right)\mathrm{d}t' &= \left[\left(g + \frac{\mathrm{d}u_{i,z}}{\mathrm{d}t'}\right)\int_0^{t'}\zeta(t'')\mathrm{d}t''\right]_0^t - \int_0^t \frac{\mathrm{d}^2u_{i,z}}{\mathrm{d}s^2}\int_0^s \zeta(s')\mathrm{d}s'\mathrm{d}s \\ &= \zeta(0)\tau_{\mathrm{f}}\left(g + \left[\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t'}\right]_{t'=t}\right)\sinh\left(\frac{t}{\tau_{\mathrm{f}}}\right) - \zeta(0)\tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\sinh\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s, \end{aligned} \tag{75}$$

and thus the vertical relative velocity

$$\begin{aligned} w_{\mathrm{s},z}(t) &= \operatorname{sech}\left(\frac{t}{\tau_{\mathrm{f}}}\right)\left[w_{\mathrm{s},z}(0) - \tau_{\mathrm{f}}\left(g + \frac{\mathrm{d}u_{i,z}}{\mathrm{d}t}\right)\sinh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + \tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\sinh\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s\right] \\ &= -\left(v_{\mathrm{f}} + \tau_{\mathrm{f}}\frac{\mathrm{d}u_{i,z}}{\mathrm{d}t}\right)\tanh\left(\frac{t}{\tau_{\mathrm{f}}}\right) + \operatorname{sech}\left(\frac{t}{\tau_{\mathrm{f}}}\right)\left[w_{\mathrm{s},z}(0) + \tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\sinh\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s\right], \end{aligned} \tag{76}$$

with asymptotic forms

$$w_{\mathrm{s},z}(t) = \begin{cases} w_{\mathrm{s},z}(0) - \left(v_{\mathrm{f}} + \tau_{\mathrm{f}}\frac{\mathrm{d}u_{i,z}}{\mathrm{d}t}\right)\left(\frac{t}{\tau_{\mathrm{f}}}\right) & t/\tau_{\mathrm{f}} \ll 1 \\ -v_{\mathrm{f}} - \tau_{\mathrm{f}}\frac{\mathrm{d}u_{i,z}}{\mathrm{d}t} + \exp\left(-\frac{t}{\tau_{\mathrm{f}}}\right)\left[w_{\mathrm{s},z}(0) + \tau_{\mathrm{f}}\int_0^t \frac{\mathrm{d}^2u_{i,z}}{\mathrm{d}s^2}\exp\left(\frac{s}{\tau_{\mathrm{f}}}\right)\mathrm{d}s\right] & t/\tau_{\mathrm{f}} \gg 1 \end{cases}. \tag{77}$$

Note that as expected for $\mathrm{d}u_{i,z}/\mathrm{d}t = 0$ and $w_{\mathrm{s},z}(0) = 0$, Eq. 76 reduces to Eq. 53 or Eq. 58. Norment [33] does not explicitly provide the vertical relative velocity solution Eq. 76 (or its simpler form for $\zeta(t) = \zeta(0)\exp(t/\tau_{\mathrm{f}})$). However, it is apparent that like the horizontal relative velocity solutions in Eqs. 73, the vertical relative velocity given by Eq. 76 generally contains an additional term accounting for a time-dependent fluid velocity field that should not be neglected to first order unless

$$\begin{aligned} \tau_{\mathrm{f}}\left|\frac{\mathrm{d}u_{i,z}}{\mathrm{d}t}\right| &\ll v_{\mathrm{f}} \\ \Rightarrow \frac{1}{g}\left|\frac{\mathrm{d}u_{i,z}}{\mathrm{d}t}\right| &\ll 1. \end{aligned} \tag{78}$$

*In Norment [33], the integrating factor is $\zeta(t) = \zeta(0)\exp(t/\tau_{\mathrm{f}})$, but this should not affect the integration by parts evaluation.

The condition $w_{\mathrm{s},xy}^2(t) \ll v_{\mathrm{f}}^2$, which we used to derive Eqs. 74, must now be verified. Substituting Eqs. 74 into this condition, we require to first order

$$v_{\mathrm{f}}^2 \gg \begin{cases} w_{\mathrm{s},x}^2(0) + w_{\mathrm{s},y}^2(0) & t/\tau_{\mathrm{f}} \ll 1 \\ \tau_{\mathrm{f}}^2 \left(\left[\frac{\mathrm{d}u_{i,x}}{\mathrm{d}t}\right] + \left[\frac{\mathrm{d}u_{i,y}}{\mathrm{d}t}\right]^2 \right) & t/\tau_{\mathrm{f}} \gg 1 \end{cases}. \tag{79}$$

The former condition on the initial horizontal relative velocity magnitude for $t/\tau_{\mathrm{f}} \ll 1$ is discussed in Section 2.2.4, and Norment [33] treats it as zero, $w_{\mathrm{s},xy}^2(0) = 0$. The latter condition for $t/\tau_{\mathrm{f}} \gg 1$ is equivalent to

$$\begin{aligned} \left[\frac{\mathrm{d}u_{i,x}}{\mathrm{d}t}\right]^2 + \left[\frac{\mathrm{d}u_{i,y}}{\mathrm{d}t}\right]^2 \ll \left(\frac{v_{\mathrm{f}}}{\tau_{\mathrm{f}}}\right)^2 = g^2, \quad t/\tau_{\mathrm{f}} \gg 1 \\ \Rightarrow \frac{1}{g^2}\left(\left[\frac{\mathrm{d}u_{i,x}}{\mathrm{d}t}\right]^2 + \left[\frac{\mathrm{d}u_{i,y}}{\mathrm{d}t}\right]^2 \right) \ll 1, \quad t/\tau_{\mathrm{f}} \gg 1, \end{aligned} \tag{80}$$

which is the horizontal dimension counterpart to Eq. 78 and is nearly always satisfied [33]. Conservatively assuming a horizontal velocity scale on the order of $u_{i,xy} = \sqrt{u_{i,x}^2 + u_{i,y}^2} \sim 10\,\mathrm{m\,s^{-1}}$, Eq. 80 essentially requires a horizontal velocity time-frequency scale of $f_{u_{xy}} \ll g/u_{i,xy} \sim 1\,\mathrm{Hz}$, which appears to be the case at least within the atmospheric boundary layer (ABL) or mixing layer [36,37]. Similarly, the vertical velocity time-frequency scale constraint $f_{u_z} \ll g/u_{i,z} \sim 1\,\mathrm{Hz}$ appears to be satisfied within the ABL [36,37].

2.2.4 Boundary Conditions Crossing from Region R_i to $R_{i'}$

When a particle crosses a boundary, its velocity $\mathbf{v}_{\mathrm{p}}(t)$ must be continuous across the boundary. Using Eq. 40, with the boundary crossing at time t_{c}, we have the constraint

$$\begin{aligned} \mathbf{v}_{\mathrm{p}}(t_{\mathrm{c}}^-) = \mathbf{v}_{\mathrm{p}}(t_{\mathrm{c}}^+) \\ \Rightarrow \mathbf{w}_{\mathrm{s}}(t_{\mathrm{c}}^-) + \mathbf{u}_{\mathrm{s}}(t_{\mathrm{c}}^-) = \mathbf{w}_{\mathrm{s}}(t_{\mathrm{c}}^+) + \mathbf{u}_{\mathrm{s}}(t_{\mathrm{c}}^+), \end{aligned} \tag{81}$$

where t_{c}^- and t_{c}^+ are the times just before and after, respectively, crossing the boundary from region R_i to $R_{i'}$. Thus, $\mathbf{u}_{\mathrm{s}}(t_{\mathrm{c}}^-) = \mathbf{u}_i(t_{\mathrm{c}})$, $\mathbf{u}_{\mathrm{s}}(t_{\mathrm{c}}^+) = \mathbf{u}_{i'}(t_{\mathrm{c}})$, $\mathbf{w}_{\mathrm{s}}(t_{\mathrm{c}}^-)$ is given by Eqs. 73 and Eq. 76, and $\mathbf{w}_{\mathrm{s}}(t_{\mathrm{c}}^+) \equiv \mathbf{w}_{\mathrm{s},0}^+$ is the initial condition for the particle relative velocity within region $R_{i'}$. Substituting the fluid velocities into Eq. 81 then yields

$$\begin{aligned} \mathbf{w}_{\mathrm{s},0}^+ = \mathbf{w}_{\mathrm{s}}(t_{\mathrm{c}}^+) &= \mathbf{w}_{\mathrm{s}}(t_{\mathrm{c}}^-) + \mathbf{u}_i(t_{\mathrm{c}}) - \mathbf{u}_{i'}(t_{\mathrm{c}}) \\ &= \mathbf{w}_{\mathrm{s}}(t_{\mathrm{c}}^-) + \Delta\mathbf{u}_{ii'}(t_{\mathrm{c}}). \end{aligned} \tag{82}$$

Then using the constraint Eq. 79 for* $t_{\mathrm{c}}^+/\tau_{\mathrm{f}} \ll 1$ from Section 2.2.3 on the horizontal velocity initial condition, Eq. 82 requires in the horizontal dimension that

$$w_{\mathrm{s},xy}^2(t_{\mathrm{c}}^+) = w_{\mathrm{s},xy}^2(t_{\mathrm{c}}^-) + 2\mathbf{w}_{\mathrm{s},xy}(t_{\mathrm{c}}^-) \cdot \Delta\mathbf{u}_{ii',xy}(t_{\mathrm{c}}) + \Delta u_{ii',xy}^2(t_{\mathrm{c}}) \ll v_{\mathrm{f}}^2, \tag{83}$$

where $\Delta u_{ii',xy}^2(t_{\mathrm{c}}) = \Delta u_{ii',x}^2(t_{\mathrm{c}}) + \Delta u_{ii',y}^2(t_{\mathrm{c}}) = \left[u_{i,x}(t_{\mathrm{c}}) - u_{i',x}(t_{\mathrm{c}})\right]^2 + \left[u_{i,y}(t_{\mathrm{c}}) - u_{i',y}(t_{\mathrm{c}})\right]^2$. Recalling from Section 2.2.3 and Eq. 79 with $t_{\mathrm{c}}^-/\tau_{\mathrm{f}} \gg 1$ that $w_{\mathrm{s},xy}^2(t_{\mathrm{c}}^-) \ll v_{\mathrm{f}}^2$, Eq. 83 essentially requires to first order

$$\Delta u_{ii',xy}^2(t_{\mathrm{c}}) = \Delta u_{ii',x}^2(t_{\mathrm{c}}) + \Delta u_{ii',y}^2(t_{\mathrm{c}}) \ll v_{\mathrm{f}}^2. \tag{84}$$

*Note that $t = t_{\mathrm{c}}^+ = 0$ corresponds to the time at which the particle enters a new region $R_{i'}$. Thus, we assume that $t_{\mathrm{c}}^-/\tau_{\mathrm{f}} \gg 1$ as the particle leaves R_i, but $t_{\mathrm{c}}^+/\tau_{\mathrm{f}} \ll 1$ as it enters $R_{i'}$.

To evaluate $\Delta u^2_{ii',xy}(t_\mathrm{c})$, we approximate $\Delta u^2_{ii',j}(t_\mathrm{c})$ for $j \in \{x, y\}$ using the average velocity gradients:

$$\Delta u_{ii',j}(t_\mathrm{c}) \approx \left\langle \nabla u_j(\mathbf{x}, t_\mathrm{c}) \right\rangle_i \cdot \Delta \mathbf{x}_{ii'} = \left\langle \frac{\partial u_j(\mathbf{x}, t_\mathrm{c})}{\partial x_{j'}} \right\rangle_i \Delta x_{ii',j'}, \tag{85}$$

where $\Delta x_{ii',j'}$ is the approximate particle displacement in direction $j' \in \{x, y, z\} = \{1, 2, 3\}$ as it traversed region i before entry into i', and $\langle f(\mathbf{x}) \rangle_i \equiv \frac{1}{V_i} \int_{R_i} f(\mathbf{x}) \mathrm{d}\mathbf{x}$ denotes the spatial average over region i of volume V_i. Note that we are using Einstein summation over j' in Eq. 85 but not for i. Then Eq. 84 approximately requires

$$\left| \langle \nabla u_x(\mathbf{x}, t_\mathrm{c}) \rangle_i \cdot \Delta \mathbf{x}_{ii'} \right|^2 + \left| \langle \nabla u_y(\mathbf{x}, t_\mathrm{c}) \rangle_i \cdot \Delta \mathbf{x}_{ii'} \right|^2 \ll v_\mathrm{f}^2. \tag{86}$$

We can estimate the region R_i mean velocity gradient terms as

$$\left\langle \frac{\partial u_j(\mathbf{x}, t_\mathrm{c})}{\partial x_{j'}} \right\rangle_i \approx \frac{\Delta u_{i,j}(t_\mathrm{c})}{\Delta x_{i,j'}}, \tag{87}$$

where $\Delta u_{i,j}(t_\mathrm{c})$ is the change in the j component of the velocity over spatial extent $\Delta x_{i,j'}$ in direction j' for region R_i. This might be taken as an average over the cell face on either end of the j' direction or simply as some point estimate, for example, at the face centers. As an upper bound for the left-hand side of Eq. 86, corresponding to the particle traversing the entire region R_i diagonally, we can let $\Delta \mathbf{x}_{ii'} = \Delta \mathbf{x}_i$ be the region i spatial extents; that is, $\Delta x_{ii',x} = \Delta x_{i,x} = \Delta x_i$, $\Delta x_{ii',y} = \Delta x_{i,y} = \Delta y_i$ and $\Delta x_{ii',z} = \Delta x_{i,z} = \Delta z_i$. Then using Eq. 87, Eq. 86 reduces to the approximate condition

$$\left| \Delta u_{i,x}(t_\mathrm{c}) \right|^2 + \left| \Delta u_{i,y}(t_\mathrm{c}) \right|^2 \ll \left(\frac{v_\mathrm{f}}{3} \right)^2, \tag{88}$$

and we need only estimate the magnitude of the changes in velocity across the regions, $\Delta u_{i,j}(t_\mathrm{c})$. Based on the discussion in Section 2.2.2 and Figure 3, the fall velocity is at most $v_\mathrm{f} \lesssim 20\,\mathrm{m\,s^{-1}}$ for the largest, densest particles, meaning that such particles would require $|\Delta u_{i,j}(t_\mathrm{c})| \ll 7\,\mathrm{m\,s^{-1}}$ for $j \in \{x, y\}$, while for more typically sized particles, we would require $|\Delta u_{i,j}(t_\mathrm{c})| \ll 1/3\,\mathrm{m\,s^{-1}}$. Thus, Eq. 88 and more generally Eq. 83 might not be satisfied for more typically sized particles under typical weather conditions.

2.2.5 Arbitrary Initial Condition in Piecewise Spatially Uniform and Time-Dependent Velocity: $\mathbf{w}_\mathrm{s}(0) = \mathbf{w}_{\mathrm{s},0}$ and $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_i(t)$ for $\mathbf{x}, \mathbf{x}_\mathrm{p}(t) \in R_i$

If possible, it would be nice to be able to lift the constraints used to solve Eq. 46 in Sections 2.2.2 and 2.2.3—namely, the constraint on small horizontal velocity fluctuation. This is especially true since we showed in Section 2.2.4 that the constraint might not be satisfied for typical particle sizes and meteorological conditions. In Section 2.2.3 we already somewhat removed the need for a zero initial condition, which was mandated in Norment [33]. Equations 65 are equivalent to Ray [34] Eqs. 14 and 15, except with the additional horizontal y dimension and vertical z dimension and the addition of a time-dependent forcing term in each equation. We can cast Eqs. 65 into the form of Ray [34] by defining

$$P(\mathbf{w}_\mathrm{s}, t) = \phi\left(|\mathbf{w}_\mathrm{s}|\right) = \alpha \sqrt{w_{\mathrm{s},x}^2 + w_{\mathrm{s},y}^2 + w_{\mathrm{s},z}^2}, \tag{89a}$$

$$\mathbf{Q}(\mathbf{w}_\mathrm{s}, t) = \mathbf{f}_\mathrm{grav} - \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t}. \tag{89b}$$

We first note that $\nabla_{\mathbf{w}_\mathrm{s}} \mathbf{Q}(\mathbf{w}_\mathrm{s}, t) = 0$ is the null matrix, as in Section 2.2.2, so the requirements of Ray [34] are fulfilled. More specifically, $\mathbf{Q}(\mathbf{w}_\mathrm{s}, t)$ is independent of $\mathbf{w}_\mathrm{s}$ altogether. In this case, we again use Eq. 55a but with the solutions more generally given by

$$w_{\mathrm{s},x}(t) = \frac{1}{\varphi}\left(w_{\mathrm{s},x,0} + \int_0^t Q_x(t')\varphi(t')\mathrm{d}t'\right) = \frac{1}{\varphi}\left(w_{\mathrm{s},x,0} - \int_0^t \frac{\mathrm{d}u_{i,x}}{\mathrm{d}t'}\varphi(t')\mathrm{d}t'\right), \tag{90a}$$

$$w_{\mathrm{s},y}(t) = \frac{1}{\varphi}\left(w_{\mathrm{s},y,0} + \int_0^t Q_y(t')\varphi(t')\mathrm{d}t'\right) = \frac{1}{\varphi}\left(w_{\mathrm{s},y,0} - \int_0^t \frac{\mathrm{d}u_{i,y}}{\mathrm{d}t'}\varphi(t')\mathrm{d}t'\right), \tag{90b}$$

$$w_{\mathrm{s},z}(t) = \frac{1}{\varphi}\left(w_{\mathrm{s},z,0} + \int_0^t Q_z(t')\varphi(t')\mathrm{d}t'\right) = \frac{1}{\varphi}\left(w_{\mathrm{s},z,0} - \int_0^t \left(g + \frac{\mathrm{d}u_{i,z}}{\mathrm{d}t'}\right)\varphi(t')\mathrm{d}t'\right). \tag{90c}$$

Substituting Eqs. 90 into Eq. 65a yields

$$\begin{aligned}\frac{\mathrm{d}\varphi}{\mathrm{d}t} = \alpha\Bigg[&\left(w_{\mathrm{s},x,0} - \int_0^t \frac{\mathrm{d}u_{i,x}}{\mathrm{d}t'}\varphi(t')\mathrm{d}t'\right)^2 + \\ &+\left(w_{\mathrm{s},y,0} - \int_0^t \frac{\mathrm{d}u_{i,y}}{\mathrm{d}t'}\varphi(t')\mathrm{d}t'\right)^2 \\ &+\left(w_{\mathrm{s},z,0} - \int_0^t \left(g + \frac{\mathrm{d}u_{i,z}}{\mathrm{d}t'}\right)\varphi(t')\mathrm{d}t'\right)^2\Bigg]^{1/2}.\end{aligned} \tag{91}$$

The final integrals in Eqs. 90 and Eq. 91 can be integrated by parts, similarly to Eq. 72 in Section 2.2.3:

$$\begin{aligned}\int_0^t \frac{\mathrm{d}u_{i,j}}{\mathrm{d}t'}\varphi(t')\mathrm{d}t' &= \left[\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t'}\int_0^{t'}\varphi(t'')\mathrm{d}t''\right]_0^t - \int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\int_0^s \varphi(s')\mathrm{d}s'\mathrm{d}s \\ &= \left[\frac{\mathrm{d}u_{i,j}}{\mathrm{d}t'}\Phi(t')\right]_0^t - \int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\Phi(s)\mathrm{d}s \\ &= \frac{\mathrm{d}u_{i,j}}{\mathrm{d}t}\Phi(t) - \int_0^t \frac{\mathrm{d}^2u_{i,j}}{\mathrm{d}s^2}\Phi(s)\mathrm{d}s\end{aligned} \tag{92}$$

for $j \in \{x, y, z\}$ and where we have used Eq. 55b. Then substituting Eq. 92 into Eq. 91 and recalling that $\varphi(t) = \mathrm{d}\Phi/\mathrm{d}t$, we obtain an integro-differential equation for $\Phi(t)$:

$$\begin{aligned}\frac{\mathrm{d}^2\Phi}{\mathrm{d}t^2} = \alpha\Bigg[&\left(w_{\mathrm{s},x,0} - \frac{\mathrm{d}u_{i,x}}{\mathrm{d}t}\Phi(t) + \int_0^t \frac{\mathrm{d}^2u_{i,x}}{\mathrm{d}s^2}\Phi(s)\mathrm{d}s\right)^2 + \\ &+\left(w_{\mathrm{s},y,0} - \frac{\mathrm{d}u_{i,x}}{\mathrm{d}t}\Phi(t) + \int_0^t \frac{\mathrm{d}^2u_{i,y}}{\mathrm{d}s^2}\Phi(s)\mathrm{d}s\right)^2 \\ &+\left(w_{\mathrm{s},z,0} - \left(g + \frac{\mathrm{d}u_{i,z}}{\mathrm{d}t'}\right)\Phi(t) + \int_0^t \frac{\mathrm{d}^2u_{i,z}}{\mathrm{d}s^2}\Phi(s)\mathrm{d}s\right)^2\Bigg]^{1/2}.\end{aligned} \tag{93}$$

As in Ray [34] Section 3.1.2, we can perform the Taylor series expansion of $\Phi(t)$ around $t = 0$: $\Phi(t) = \sum_{n=0}^{\infty} a_n t^n$, where $a_n = \frac{1}{n!}\left[\frac{\mathrm{d}^n\Phi(t)}{\mathrm{d}t^n}\right]_{t=0}$. The first two coefficients are given by $a_0 = \Phi(0) = 0$ and $a_1 = [\mathrm{d}\Phi(t)/\mathrm{d}t]_{t=0} = \varphi(0) = 1$, while the higher coefficients are generally given by evaluating $a_{n+2} = \frac{1}{(n+2)!}\left[\frac{\mathrm{d}^n}{\mathrm{d}t^n}\left(\frac{\mathrm{d}^2\Phi}{\mathrm{d}t^2}\right)\right]_{t=0}$ using Eq. 93. This leads to, for example,

$$\begin{aligned}a_2 &= \frac{\alpha}{2}\sqrt{w_{\mathrm{s},x,0}^2 + w_{\mathrm{s},y,0}^2 + w_{\mathrm{s},z,0}^2}, \\ a_3 &= -\frac{\alpha}{6\sqrt{w_{\mathrm{s},x,0}^2 + w_{\mathrm{s},y,0}^2 + w_{\mathrm{s},z,0}^2}}\left(w_{\mathrm{s},x,0}\left[\frac{\mathrm{d}u_{i,x}}{\mathrm{d}t}\right]_{t=0} + w_{\mathrm{s},y,0}\left[\frac{\mathrm{d}u_{i,y}}{\mathrm{d}t}\right]_{t=0} + w_{\mathrm{s},z,0}\left(g + \left[\frac{\mathrm{d}u_{i,z}}{\mathrm{d}t}\right]_{t=0}\right)\right),\end{aligned}$$

and so on. Using $\varphi(t) = \mathrm{d}\Phi/\mathrm{d}t$ in Eqs. 90 then solves the general, time-dependent problem with arbitrary initial conditions for spatially uniform regions. This still has the requirements for the particle to reach steady state within a region R_i in order to avoid a velocity gradient as in Eq. 42b—the presence of which would violate the requirements of Ray [34] on **Q**—but there are no longer restrictions on the relative magnitude of the horizontal and vertical velocity displacements $w_{\mathrm{s},xy}(t)$ and $w_{\mathrm{s},z}(t)$ or their initial conditions $w_{\mathrm{s},xy}(0)$ and $w_{\mathrm{s},z}(0)$.

3. DELFIC ATMOSPHERIC TRANSPORT MODEL

There are two types of atmospheric transport models implemented in the early-1970s DELFIC DTM code [8, 9]: the original Lagrangian advective transport model with semi-empirical horizontal dispersion and the DTM with horizontal and vertical diffusion. Both models rely on an empirical Gaussian puff model for the fallout plume horizontal dispersion due to atmospheric turbulence.[*] In the vertical dimension, there are two modes of transport, however. The first is the original "advection plus settling" mode [2, 8, 10] in which vertical diffusive transport is neglected compared with vertical advection and gravitational settling. Later on with the introduction of the DTM [8, 9], the effects of vertical diffusion were incorporated where they could not be neglected. Eventually, however, the DELFIC DTM was simplified by removing the vertical diffusion solver.[†] Thus, the 1979 and later versions use the Lagrangian advective transport model with gravitational settling and horizontal Gaussian dispersion [2, 3].

We begin in Section 3.1 with a basic description of the turbulent atmosphere used by DELFIC and based on the turbulent kinetic energy (TKE) and its dissipation rate. In Section 3.2 we present the Gaussian puff model used by DELFIC for describing turbulent dispersion and its analytical Green's function solutions in the horizontal and vertical dimensions in Section 3.2.1. Specifically, DELFIC begins with a time-dependent, homogeneous advection-diffusion with Gaussian dispersion parameters modeled by a relatively general "four-thirds" scaling law discussed in Section 3.2.2. In Section 3.3, we discuss various types of transport models and the approximate regimes in which they are appropriate, clarifying the hybrid Eulerian–Lagrangian nature of the DTM. Sections 3.4 and 3.5, respectively, then discuss the original advection plus settling transport model with horizontal Gaussian dispersion and the DTM that incorporated diffusion in the vertical dimension. Both of these models build off of the semi-empirical Gaussian puff model with atmospheric dispersion due to turbulence primarily characterized by the TKE dissipation rate.

3.1 DESCRIPTION OF THE TURBULENT ATMOSPHERE

Because of the highly complex, sharp, and irregular structure of turbulent fluid flows [38, 39], it is typically impractical to completely resolve the turbulence via, for example, direct numerical simulation (DNS) in high Reynolds number flows such as the turbulent atmosphere [28]. Thus, one typically begins with a stochastic formulation of the fluid dependent variables velocity $\mathbf{u}$, pressure p, density ρ, and scalar concentration C to characterize the turbulent fluid flow. Let the fluid velocity $\mathbf{u}$ be decomposed into both mean and (stochastic) turbulent components, respectively, as

$$\mathbf{u}(\mathbf{x}, t) = \langle \mathbf{u}(\mathbf{x}, t) \rangle + \mathbf{u}'(\mathbf{x}, t). \tag{94}$$

Several quantities of interest can be calculated from the stochastic velocity field to characterize the fluid turbulence. In particular, the TKE k, TKE dissipation rate ε, and turbulence time scale τ, respectively, are

[*] Actually, the original transport considered atmospheric dispersion in the horizontal dimension in a rather rudimentary way because the theory of atmospheric diffusion was still somewhat unsettled at the time [10, 33].

[†] It is unclear whether this was removed because it was not worth the additional complexity and computational cost compared with the original model.

defined* as

$$k(\mathbf{x},t) \equiv \frac{1}{2}\left\langle \mathbf{u}' \cdot \mathbf{u}' \right\rangle = \frac{1}{2}\left\langle u_i' u_i' \right\rangle, \tag{95a}$$

$$\varepsilon(\mathbf{x},t) \equiv 2\nu \left\langle e_{ij}' e_{ij}' \right\rangle, \tag{95b}$$

$$\tau \equiv \frac{k}{\varepsilon}, \tag{95c}$$

where the fluctuating rate of strain tensor is given by

$$e_{ij}' \equiv e_{ij} - \langle e_{ij} \rangle = \frac{1}{2}\left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i} \right). \tag{96}$$

The TKE in the atmosphere is often on the order of $k \sim 1\,\mathrm{m}^2\,\mathrm{s}^{-2}$, depending on time of day and height [28,41]; atmospheric TKE dissipation rates measured in the ABL or atmospheric surface layers (ASL) in various field campaigns show typical dissipation rates of $10^{-2}\,\mathrm{m}^2\,\mathrm{s}^{-3} \lesssim \varepsilon \lesssim 10^{-1}\,\mathrm{m}^2\,\mathrm{s}^{-3}$ [28,40,42], albeit with lower dissipation rates $\varepsilon \lesssim 10^{-3}\,\mathrm{m}^2\,\mathrm{s}^{-3}$ observed at various times of day or heights [28,43]. Radar measurements [44,45,46,47], for example, have shown that vertical profiles of ε above the ABL ($h_{\mathrm{PBL}} \sim 1\,\mathrm{km}$) are on the order of $10^{-6}\,\mathrm{m}^2\,\mathrm{s}^{-3} \lesssim \varepsilon \lesssim 10^{-3}\,\mathrm{m}^2\,\mathrm{s}^{-3}$, with perhaps typical values of $\varepsilon \sim 10^{-4}\,\mathrm{m}^2\,\mathrm{s}^{-3}$. The DELFIC documentation [2] suggests that $10\,\mathrm{s}\,\mathrm{m}^{-2/3} \lesssim \varepsilon^{-1/3} \lesssim 100\,\mathrm{s}\,\mathrm{m}^{-2/3} \Rightarrow 10^{-6}\,\mathrm{m}^2\,\mathrm{s}^{-3} \lesssim \varepsilon \lesssim 10^{-3}\,\mathrm{m}^2\,\mathrm{s}^{-3}$, indicating that its expected TKE dissipation rates are more consistent with values above the ABL.

3.1.1 Hyperbolic TKE Dissipation Rate Profile

When the TKE dissipation rate ε is unavailable from the meteorological data, DELFIC uses a hyperbolic function to estimate it as a function of height z [2,48]:

$$\varepsilon = \frac{(\tau_0/\rho)^{3/2}}{\kappa(z + R_{\mathrm{sfc}})} = \frac{u_\star^3}{\kappa(z + R_{\mathrm{sfc}})}, \tag{97}$$

where τ_0 is the surface shear stress, $u_\star \equiv \sqrt{\tau_0/\rho}$ is the surface layer friction velocity, R_{sfc} is the surface roughness length, and $\kappa \approx 0.40$ is the von Kármán constant. Within the surface layer ($h_{\mathrm{sfc}} \sim 50\,\mathrm{m} \sim 0.1 h_{\mathrm{PBL}}$ [19,49]), the friction velocity can be calculated using the velocity scaling laws of Monin and Obukhov [19,49,50,51]:

$$\frac{\kappa z}{u_\star}\frac{\partial \langle u \rangle}{\partial z} = \phi_{\mathrm{M}}\left(\frac{z}{L}\right), \tag{98}$$

where [50]

$$\phi_{\mathrm{M}}(\zeta) = \begin{cases} 1 + 4.7\zeta & \zeta > 0, \text{ stable} \\ 1 & \zeta = 0, \text{ neutral} \\ (1 - 15\zeta)^{-1/4} & \zeta < 0, \text{ unstable} \end{cases}, \tag{99}$$

*Monin [38] defines $\varepsilon = 2\nu e_{ij} e_{ij}$ in Section 6.1 as the total kinetic energy dissipation rate and denotes the (mean) TKE dissipation rate as $\overline{\varepsilon} = \langle \varepsilon \rangle$ in Monin [38] Eq. 7.91 or Monin [40] Eq. 21.3'. For very large Reynolds number turbulence, the two definitions are approximately equivalent since direct dissipation due to the mean velocity is negligible compared with the turbulent component dissipation [40]:

$$\begin{aligned} 2\nu \langle e_{ij} e_{ij} \rangle &= 2\nu \left\langle \left[\langle e_{ij} \rangle + e_{ij}'\right] \left[\langle e_{ij} \rangle + e_{ij}'\right] \right\rangle \\ &= 2\nu \langle e_{ij} \rangle \langle e_{ij} \rangle + 2\nu \left\langle e_{ij}' e_{ij}' \right\rangle \\ &\approx 2\nu \left\langle e_{ij}' e_{ij}' \right\rangle. \end{aligned}$$

and L is the Monin-Obukhov length. Then integrating the velocity scaling law Eq. 98 from the surface roughness height R_{sfc} to height z with boundary condition $\langle u(R_{\mathrm{sfc}})\rangle = 0$ yields [19]

$$\begin{aligned}
\frac{\partial \langle u\rangle}{\partial z} &= \frac{u_\star}{\kappa}\frac{\phi_{\mathrm{M}}\left(\frac{z}{L}\right)}{z} \\
\Rightarrow \int_{R_{\mathrm{sfc}}}^{z} \frac{\partial \langle u\rangle}{\partial z}\mathrm{d}z &= \frac{u_\star}{\kappa}\int_{R_{\mathrm{sfc}}}^{z} \frac{\phi_{\mathrm{M}}\left(\frac{z}{L}\right)}{z}\mathrm{d}z \\
\Rightarrow \langle u(z)\rangle &= \frac{u_\star}{\kappa}\int_{R_{\mathrm{sfc}}/L}^{z/L} \frac{\phi_{\mathrm{M}}(\zeta)}{\zeta}\mathrm{d}\zeta \\
&= \frac{u_\star}{\kappa}\begin{cases} \ln\left(\frac{z}{R_{\mathrm{sfc}}}\right) + 4.7\left(\frac{z-R_{\mathrm{sfc}}}{L}\right) & L > 0\text{, stable} \\ \ln\left(\frac{z}{R_{\mathrm{sfc}}}\right) & |L| \to \infty\text{, neutral} \\ \ln\left(\frac{z}{R_{\mathrm{sfc}}}\right) + \ln\left(\frac{\left(n_0^2+1\right)(n_0+1)^2}{\left(n_1^2+1\right)(n_1+1)^2}\right) + 2\left(\tan^{-1} n_1 - \tan^{-1} n_0\right) & L < 0\text{, unstable} \end{cases} \\
&= \frac{u_\star}{\kappa}\left[\ln\left(\frac{z}{R_{\mathrm{sfc}}}\right) + \psi_{\mathrm{M}}\left(\frac{z}{L}, \frac{h_0}{L}\right)\right], \qquad (100)
\end{aligned}$$

where

$$\psi_{\mathrm{M}}\left(\frac{z}{L}, \frac{R_{\mathrm{sfc}}}{L}\right) = \begin{cases} 4.7\left(\frac{z-R_{\mathrm{sfc}}}{L}\right) & L > 0\text{, stable} \\ 0 & |L| \to \infty\text{, neutral} \\ \ln\left(\frac{\left(n_0^2+1\right)(n_0+1)^2}{\left(n_1^2+1\right)(n_1+1)^2}\right) + 2\left(\tan^{-1} n_1 - \tan^{-1} n_0\right), & L < 0\text{, unstable} \end{cases} \qquad (101)$$

with

$$n_0 = \left(1 - 15\frac{R_{\mathrm{sfc}}}{L}\right)^{1/4}, \qquad (102a)$$

$$n_1 = \left(1 - 15\frac{z}{L}\right)^{1/4}. \qquad (102b)$$

Given a measurement $u_{\mathrm{ref}} = \langle u(z_{\mathrm{ref}})\rangle$ of the (mean) surface wind speed at a typical reference height $z_{\mathrm{ref}} = 10\,\mathrm{m}$, one can then solve Eq. 100 for the friction velocity:

$$u_\star = \frac{\kappa u_{\mathrm{ref}}}{\ln\left(\frac{z_{\mathrm{ref}}}{R_{\mathrm{sfc}}}\right) + \psi_{\mathrm{M}}\left(\frac{z_{\mathrm{ref}}}{L}, \frac{R_{\mathrm{sfc}}}{L}\right)}. \qquad (103)$$

DELFIC uses a slightly different definition of ψ_{M} based on Barker [2,51]. In particular, it appears that Barker [51] sets $R_{\mathrm{sfc}} = 0$ in Eq. 101, leading to $\psi_{\mathrm{M}} = 4.7 z_{\mathrm{ref}}/L$ for the stable atmosphere, and $n_0 = 1$ for the unstable atmosphere. It is unclear whether this is intentional (i.e., assuming $R_{\mathrm{sfc}} \ll z_{\mathrm{ref}}$) or a typographical error. Based on McRae [19,52], this would be problematic for use of DELFIC in urban or mountainous areas [28]. The user can input the values u_{ref}, z_{ref}, R_{sfc}, and $1/L$ for DELFIC to compute $u_\star$ according to Eq. 103 and the TKE dissipation ε according to Eq. 97 in subroutine `wilkns`. Alternatively, DELFIC can internally set these parameters, in which case, the typical values of ε are given by

$$\varepsilon = \frac{0.03\,\mathrm{m}^3\,\mathrm{s}^{-3}}{z}. \qquad (104)$$

See also Albertson [53] Eqs. 7 and 17 for similar hyperbolic scaling based on the nondimensional dissipation scaling function ϕ_ε in neutral and unstable atmospheres for $|z/L| \ll 1$ and Stull [28] example

9.5.5 for an equation comparable to Eq. 104 in the surface layer. Note that given the various measurement campaigns and radar measurements discussed at the beginning of Section 3.1, the use of the hyperbolic scaling law in Eqs. 97 or 104 is questionable above the ABL. Thus, one should whenever possible provide reliable estimates of ε based on actual meteorological conditions.

3.1.2 Spatiotemporal Averaging

Because of the implicit spatial dependence of ε, DELFIC calculates in the `tranp` subroutine a spatially averaged dissipation (denoted by `dxbar` and `dybar` in the downwind and crosswind directions, respectively) between layers i and $i' > i$ according to Norment [2] Eq. 3.2.2,

$$\begin{aligned}\langle\varepsilon\rangle_{ii'} &= \frac{1}{z_{\mathrm{met},i'} - z_{\mathrm{met},i}}\left[\sum_{j=0}^{i'-1}\varepsilon(z_{\mathrm{met},j})\Delta z_{\mathrm{met},j} - \sum_{j=0}^{i-1}\varepsilon(z_{\mathrm{met},j})\Delta z_{\mathrm{met},j}\right] \\ &= \frac{1}{z_{\mathrm{met},i'} - z_{\mathrm{met},i}}\sum_{j=i}^{i'-1}\varepsilon(z_{\mathrm{met},j})\Delta z_{\mathrm{met},j}, \qquad (105)\end{aligned}$$

where $\Delta z_{\mathrm{met},i} = z_{\mathrm{met},i+1} - z_{\mathrm{met},i}$ is the thickness of meteorological data layer i with (mean) TKE dissipation rate $\varepsilon(z_{\mathrm{met},i})$. This spatially averaged value is then averaged temporally over the course of the parcel trajectory to determine the mean dissipation (denoted by `dsprtx` and `dsprty`):

$$\varepsilon = \frac{\sum_k \langle\varepsilon\rangle_{ii'}\Delta t_k}{\sum_k \Delta t_k}, \qquad (106)$$

where Δt_k is the residence time of a parcel trajectory within a given atmospheric cell [8]. Supposing that we have a single vertical layer and that DELFIC is configured to use Eq. 104 to estimate the TKE dissipation rate spatial dependence, then we have the spatiotemporally averaged TKE dissipation rate

$$\begin{aligned}\varepsilon &= \frac{0.03\,\mathrm{m}^3\,\mathrm{s}^{-3}}{z_0}\int_{z_{\mathrm{met},0}}^{z_0}\frac{\mathrm{d}z'}{z'} \\ &= \frac{0.03\,\mathrm{m}^3\,\mathrm{s}^{-3}}{z_0}\ln\left(\frac{z_0}{z_{\mathrm{met},0}}\right).\end{aligned}$$

Substituting $z_0 = 10^4\,\mathrm{m}$ as the initial parcel height, and $z_{\mathrm{met},0} = 10\,\mathrm{m}$ as the lowest meteorological data level, we have $\varepsilon \sim 2 \times 10^{-5}\,\mathrm{m}^2\,\mathrm{s}^{-3}$.

3.2 GAUSSIAN PUFF MODEL WITH TIME-DEPENDENT DIFFUSION COEFFICIENTS

As the fallout parcels undergo advection in the horizontal and vertical dimensions, they are taken to have a bivariate Gaussian distribution in the horizontal dimensions. These Gaussian distributions have increasing variance as a function of time t to account for the dispersion in the turbulent atmosphere, which is characterized using the dissipation rate ε. The starting point of the Gaussian puff model is the advection-diffusion equation for the mean concentration $\langle C(\mathbf{x},t)\rangle = C(\mathbf{x},t) - C'(\mathbf{x},t)$ of material in an incompressible fluid [19, 39]:

$$\frac{\partial\langle C\rangle}{\partial t} + \langle v_{\mathrm{p},i}\rangle\frac{\partial\langle C\rangle}{\partial x_i} - \frac{\partial}{\partial x_i}\left(K_{ij}^{\mathrm{eff}}\frac{\partial\langle C\rangle}{\partial x_j}\right) = \langle S(\mathbf{x},t)\rangle, \qquad (107)$$

where $K_{ij}^{\mathrm{eff}}(\mathbf{x},t) = D\delta_{ij} + K_{ij}(\mathbf{x},t)$ is the effective diffusion coefficient due to molecular diffusivity D and turbulent diffusivity or eddy diffusivity tensor $K_{ij}(\mathbf{x},t)$, and $S(\mathbf{x},t) = \langle S(\mathbf{x},t)\rangle + S'(\mathbf{x},t)$ is a volumetric mass source density rate. The turbulent diffusivity tensor is the constant of proportionality in the first-order closure approximation or gradient-diffusion hypothesis for turbulent mass transport [19, 38, 39]:

$$\left\langle u_i' C' \right\rangle = -K_{ij} \frac{\partial \left\langle C \right\rangle}{\partial x_j}. \tag{108}$$

As can be seen from Eq. 108, in contrast to molecular diffusion, the turbulent diffusivity is not a physical property of the fluid but rather a statistical property of its fluctuating motion [38]. For high Reynolds number flows such as the atmospheric surface and mixed layers [28], the molecular diffusivity D is negligible compared with the turbulent diffusivity K_{ij}, so molecular diffusion can usually be neglected in K_{ij}^{eff} [38, 39].

Since DELFIC is modeling the advection-diffusion of the solid dispersed phase of particles with Eq. 107, the advection velocity is given by the particle or parcel (center of mass) velocity $\langle \mathbf{v}_\mathrm{p} \rangle$ rather than simply the local mean fluid velocity $\langle \mathbf{u}(\mathbf{x}, t) \rangle$. This appears to presume that the particles form a velocity field of their own, which is to say that there is a sufficiently large set of dispersed particles such that they form a continuum. Putting aside the fact that Section 2 assumes a dilute dispersed phase—contrary to the requirements of a continuum dispersed phase herein proposed—the results of Section 2 yield a kinematic solution for the dispersed phase particle velocity of the form $\langle \mathbf{v}_\mathrm{p} \rangle = \mathbf{w}_\mathrm{s} + \langle \mathbf{u}_\mathrm{s} \rangle$, which implicitly depends on particle position $\mathbf{x}_\mathrm{p}$. If, however, there is a continuum of particles, we can treat the particle position $\mathbf{x}_\mathrm{p}$ as an independently varying spatial parameter $\mathbf{x}$, yielding the explicit Eulerian particle velocity field $\langle \mathbf{v}_\mathrm{p}(\mathbf{x}, t) \rangle = \mathbf{w}(\mathbf{x}, t) + \langle \mathbf{u}(\mathbf{x}, t) \rangle$ with $\mathbf{v}_\mathrm{p}(t) = \mathbf{v}_\mathrm{p}\left[\mathbf{x}_\mathrm{p}(t), t\right]$ and $\mathbf{w}_\mathrm{s}(t) = \mathbf{w}\left[\mathbf{x}_\mathrm{p}(t), t\right]$. Note that $\mathbf{w}_\mathrm{s}$ has an implicit spatial dependence in Eqs. 42 due to the forcing terms depending on spatial coordinate, so one could in principle solve $\mathbf{w}$ not as a function along particle trajectory $\mathbf{x}_\mathrm{p}(t)$ but as the Eulerian field solution to a partial differential equation (PDE). This allows us to create an Eulerian field for the dispersed phase particles $\langle \mathbf{v}_\mathrm{p}(\mathbf{x}, t) \rangle$, which can be used in Eq. 107 from the parameterized solution $\mathbf{w}(\mathbf{x}, t)$.

A more rigorous way to understand this formulation of the advection velocity is through Maxey [54] Eq. 5.9 and the equilibrium Eulerian approach [22, 25, 26] for sufficiently small particles. The assumptions of the equilibrium Eulerian method are quite similar to those of Section 2.2.2.3; namely, they both presume that the particle initial condition effects decay sufficiently rapidly that the particle velocity can be determined by the local fluid flow. As explained in more detail in Section 3.4.2, the equilibrium Eulerian method essentially requires that the particle Stokes number is much less than unity: $\mathrm{St}_\mathrm{p} \equiv \tau_\mathrm{p}/\tau_\mathrm{K} \ll 1$, where τ_K is the Kolmogorov time scale discussed in Section 3.2.2. In the atmosphere, this essentially corresponds to particles with response time scales $\tau_\mathrm{p} \ll 0.1\,\mathrm{s}$ or diameters $d_\mathrm{p} \ll 100\,\mu\mathrm{m}$. On the other hand, particles with $\mathrm{St}_\mathrm{p} \gtrsim 1$ should be modeled as Lagrangian particles and not undergo dispersion according to the advection-diffusion Eq. 107. Thus, it is the time scale of the dispersion parameters DELFIC uses discussed in Section 3.2.2 that essentially permits this hybrid Eulerian–Lagrangian approach for both small and larger particles.

3.2.1 Green's Function Solution

Suppose that the mean velocity $\langle \mathbf{u} \rangle$ and eddy diffusivity K_{ij} are spatially uniform. While these are not spatially uniform in practice, as discussed in Section 3.4.1, the trajectory of each DELFIC parcel is divided up into regions of constant wind velocity. Similarly, DELFIC performs a spatial averaging of the TKE dissipation as described in Section 3.1.2 such that the effective eddy diffusivity Eq. 130 will be spatially uniform over vertical layers of a horizontal meteorological cell. Furthermore, suppose that the eddy diffusivity is diagonal in the basis axis-aligned with the spatial coordinate system. That is, we have

$$\langle \mathbf{u} \rangle = \langle \mathbf{u}(t) \rangle , \tag{109a}$$

$$K_{ij} = K_{ij}(t) = K_i(t)\delta_{ij}, \tag{109b}$$

where no summation is implied in Eq. 109b. This removes several off-diagonal diffusion terms from the general advection-diffusion Eq. 107. In general, the off-diagonal terms for K_{ij} will be nonzero, but the qualitative predictions of the semi-empirical theory will not depend significantly on the presence or absence of these terms in Eq. 107. See Monin [38] Section 10.5 for further discussion on this point.

Then Eq. 107 can be written as

$$\mathcal{L} \langle C \rangle = \frac{\partial \langle C \rangle}{\partial t} + \langle v_{\mathrm{p},i}(t) \rangle \frac{\partial \langle C \rangle}{\partial x_i} - K_i(t) \frac{\partial \langle C \rangle}{\partial x_i \partial x_i} = \langle S(\mathbf{x}, t) \rangle , \tag{110}$$

where we have defined the linear operator

$$\mathcal{L}(\mathbf{x}, t) = \frac{\partial}{\partial t} + \langle v_{\mathrm{p},i}(t) \rangle \frac{\partial}{\partial x_i} - K_i(t) \frac{\partial}{\partial x_i \partial x_i}. \tag{111}$$

The Green's function solution to Eq. 110 in an infinite domain[*] in N_{dim} dimensions with a Dirac delta source $\langle S(\mathbf{x}, t) \rangle = \delta(\mathbf{x} - \mathbf{x}_0)\delta(t - t_0)$ for each DELFIC parcel and homogeneous boundary conditions[†] is [55, 56]

$$G(\mathbf{x}, t | \mathbf{x}_0, t_0) = \theta(t - t_0) \prod_{i=1}^{N_{\mathrm{dim}}} \frac{1}{\sqrt{4\pi \kappa_i (t - t_0)}} \exp\left(- \frac{(x_i - x_{0,i} - \int_0^{t-t_0} \langle v_{\mathrm{p},i}(\tau + t_0) \rangle \, \mathrm{d}\tau)^2}{4\kappa_i (t - t_0)} \right), \tag{112}$$

where $\theta(t)$ is the Heaviside theta function, and [56]

$$\kappa_i(t - t_0) = \frac{1}{2} \sigma_i^2 (t - t_0) = \int_0^{t - t_0} K_i(\tau) \mathrm{d}\tau \tag{113}$$

is half of the Gaussian variance $\sigma_i^2(t - t_0)$. Equation 112 is a Gaussian distribution, which gives the Gaussian puff and plume models their names. It immediately follows from Eq. 113 that the diffusivity is given by

$$K_i(\tau) = \frac{1}{2} \frac{\mathrm{d}\sigma_i^2}{\mathrm{d}\tau}. \tag{114}$$

Using the Green's function Eq. 112, the general solution for an arbitrary source term $\langle S(\mathbf{x}, t) \rangle$ with inhomogeneous initial condition $\langle C(\mathbf{x}, t_0) \rangle = \langle C_0(\mathbf{x}) \rangle$ is thus given by [19, 55]

$$\langle C(\mathbf{x}, t) \rangle = \int G(\mathbf{x}, t | \mathbf{x}', t_0) \langle C_0(\mathbf{x}') \rangle \, \mathrm{d}\mathbf{x}' + \int_{t_0}^{t} \int G(\mathbf{x}, t | \mathbf{x}', t') \langle S(\mathbf{x}', t') \rangle \, \mathrm{d}\mathbf{x}' \mathrm{d}t'. \tag{115}$$

As can be seen in Section 3.5.1, it is useful to define the concentration profile in relative coordinates:

$$\langle C(\mathbf{x}, t) \rangle = C_{XYZ}(\mathbf{X}, t), \tag{116}$$

[*] In the horizontal dimensions, an infinite domain is a reasonable approximation for terrain with small surface roughness $R_{\mathrm{sfc}} \lesssim 10\,\mathrm{m}$ (i.e., non-urban, non-mountainous environments). The vertical dimension is treated somewhat differently, as we will see.

[†] Alternatively, the inhomogeneous problem with homogeneous boundary conditions is equivalent to the problem with homogeneous source $\langle S(\mathbf{x}, t) \rangle = 0$ and inhomogeneous Dirac delta initial condition $\langle C(\mathbf{x}, t_0) \rangle = \delta(\mathbf{x} - \mathbf{x}_0)$ according to Duhamel's principle.

where

$$X_i = x_i - x_{0,i} - \int_0^{t-t_0} \langle v_{\mathrm{p},i}(\tau + t_0)\rangle \, \mathrm{d}\tau \tag{117}$$

for $i \in \{1, 2, 3\} = \{x, y, z\}$. Furthermore, suppose that the solution is separable into functions depending on the horizontal and vertical coordinates (as is the Green's function Eq. 112):

$$C_{XYZ}(\mathbf{X}, t) = Q_0 C_{XY}(X, Y, t) C_Z(Z, t), \tag{118}$$

where $Q_0 = M_\mathrm{p} = N_\mathrm{p} m_\mathrm{p}$ is the total parcel mass, and C_{XY} and C_Z are the spatial density functions in the horizontal and vertical coordinates, respectively.

In the case of a homogeneous source $S(\mathbf{x}, t) = 0$ with inhomogeneous boundary conditions, Eq. 115 in our $N_\mathrm{dim} = 3$ transformed coordinate system becomes

$$C_{XYZ}(\mathbf{X}, t) = \int G(\mathbf{X}, t|\mathbf{X}', t_0) C_{0,XYZ}(\mathbf{X}')\mathrm{d}\mathbf{X}', \tag{119}$$

where

$$G(\mathbf{X}, t|\mathbf{X}_0, t_0) = \theta(t - t_0) \prod_{i=1}^{3} \frac{1}{\sqrt{2\pi}\sigma_i(t - t_0)} \exp\left(-\frac{(X_i - X_{0,i})^2}{2\sigma_i^2(t - t_0)}\right). \tag{120}$$

Having a separable solution according to Eq. 118 requires that we can write

$$\begin{aligned} Q_0 C_{XY}(X, Y, t) C_Z(Z, t) &= \int G(\mathbf{X}, t|\mathbf{X}', t_0) C_{0,XYZ}(\mathbf{X}')\mathrm{d}\mathbf{X}' \\ &= Q_0 \int G_{XY}(\mathbf{X}_{XY}, t|\mathbf{X}'_{XY}, t_0) C_{0,XY}(\mathbf{X}'_{XY})\mathrm{d}\mathbf{X}'_{XY} \\ &\times \int G_Z(Z, t|Z', t_0) C_{0,Z}(Z')\mathrm{d}Z', \end{aligned}$$

where $\mathbf{X}_{XY} = (X, Y)$,

$$\begin{aligned} C_{0,XYZ}(\mathbf{X}) &= Q_0 C_{0,XY}(\mathbf{X}_{XY}) C_{0,Z}(Z), \\ G(\mathbf{X}, t|\mathbf{X}_0, t_0) &= G_{XY}(\mathbf{X}_{XY}, t|\mathbf{X}_{0,XY}, t_0) G_Z(Z, t|Z_0, t_0). \end{aligned}$$

Thus, the horizontal and vertical concentration profiles are given by

$$C_{XY}(X, Y, t) = \int G_{XY}(\mathbf{X}_{XY}, t|\mathbf{X}'_{XY}, t_0) C_{0,XY}(\mathbf{X}'_{XY})\mathrm{d}\mathbf{X}'_{XY}, \tag{121a}$$

$$C_Z(Z, t) = \int G_Z(Z, t|Z', t_0) C_{0,Z}(Z')\mathrm{d}Z'. \tag{121b}$$

Given that the horizontal fallout particle motion is of a different kind than the vertical transport, this separability might be a reasonable assumption, even though the vertical and horizontal transport are not completely independent [8]. DELFIC attempts to account for some of this coupling through the use of its Lagrangian advective transport portion discussed in Section 3.4.1. In particular, the DTM authors consider in Norment [8] Appendix A the case of vertically varying, horizontally uniform wind fields in the horizontal direction in an unbounded medium, axis-aligned constant eddy diffusivity ($K_{ij} = K_i\delta_{ij}$), and point or vertical line emission sources. They then compare the analytical advection-diffusion Eq. 107 solution to the approximate solution to Eq. 107 with horizontal advection handled using the advective center of mass transport method of Section 3.4.1. That is, the DELFIC approximate solution method uses the advection plus settling model of Section 3.4.1 and does not include the horizontal advection terms in Eq. 107, consistent with the separation of variables frame moving with the fallout parcel horizontal velocity of Section 3.5.1.

The approximate DELFIC solution is thus a 3D Gaussian distribution but with the center of mass approximated using the advective transport mode. In contrast, the exact analytical solution in this case is not Gaussian but rather an exponential of a more general ellipsoid the axes of which rotate with respect to the coordinate axes in time [8]. Despite the coupling having a large impact on fallout parcel center of mass displacements, the concentration fields show a relatively minor difference when compared with the exact analytical solution, with concentration ratios ranging from only 1 to 10. Furthermore, these larger concentration errors are typically observed at the tails of the concentration distributions, where good accuracy is not expected or required, or for longer transport times (10^3 s to 10^5 s, depending on position and source shape), by which point the concentration distribution has become relatively flat such that the concentration is not as sensitive to the exact center of mass position. However, this coupling between the horizontal and vertical motion can result in much more complex, non-ellipsoidal distributions—possibly even bifurcations—as different parts of the dispersed fallout parcel experience spatially dependent advection and diffusion over longer transport times. This could lead to significantly different simulated concentrations under real meteorological conditions over longer dispersion ranges. Thus, provided meteorlogical spatial cells are sufficiently large (i.e., $\sigma_i \ll \Delta x_i$), the separable advection-diffusion equation Gaussian puff model can provide a reasonable approximation to the concentration distribution of the parcels.

3.2.1.1 Horizontal concentration profile

In the horizontal dimensions with a Dirac-delta initial condition

$$C_{XY}(\mathbf{X}_{XY}, t_0) = C_{0,XY}(\mathbf{X}_{XY}) = \delta(\mathbf{X}_{XY}) \tag{122}$$

at time t_0, Eq. 121a yields

$$C_{XY}(X, Y, t) = \theta(t - t_0) \prod_{i=1}^{2} \frac{1}{\sqrt{2\pi}\sigma_i(t - t_0)} \exp\left(-\frac{X_i^2}{2\sigma_i^2(t - t_0)}\right). \tag{123}$$

3.2.1.2 Vertical concentration profile

In the vertical dimension, DELFIC parcels have the uniform initial condition*

$$C_Z(Z, t_0) = C_{0,Z}(Z) = \frac{\theta(Z - Z_1) - \theta(Z - Z_2)}{Z_2 - Z_1}, \tag{124}$$

with Z_1 and Z_2 defined similarly to Z as

$$Z_i(t) = z_i(t) - z_0 - \int_0^{t-t_0} \langle v_{\mathrm{p},z}(\tau + t_0)\rangle \, \mathrm{d}\tau,$$

where the parcel bottom z_1 and top z_2 are given by

$$z_i(t) = z_i(t_0) + \int_0^{t-t_0} \langle v_{\mathrm{p},z}(\tau + t_0)\rangle \, \mathrm{d}\tau.$$

*See, for example, the initial condition given by Norment [8] Eq. 44 and shown in Norment [8] Figure 3 at $t = 0$.

Thus, it follows that $Z_1(t) \leq Z_2(t) \Leftrightarrow z_1(t) \leq z_2(t) \Leftrightarrow z_1(t_0) \leq z_2(t_0)$, and $Z_i(t) = Z_i = z_i(t_0) - z_0$ are constant[*] offsets relative to the initial source height z_0. Furthermore, we have $Z_2 - Z_1 = z_2(t_0) - z_1(t_0)$ and $Z - Z_i = z - z_0 - \int_0^{t-t_0} \langle v_{\mathrm{p},z}(\tau + t_0)\rangle \,\mathrm{d}\tau - [z_i(t_0) - z_0] = z - z_i(t_0) - \int_0^{t-t_0} \langle v_{\mathrm{p},z}(\tau + t_0)\rangle \,\mathrm{d}\tau$.

The (infinite domain) Green's function in the vertical direction of Eq. 121b is given by

$$G_Z(Z,t|Z_0,t_0) = G_Z^{\infty}(Z,t|Z_0,t_0) = \theta(t-t_0)\frac{1}{\sqrt{2\pi}\sigma_z(t-t_0)}\exp\left(-\frac{(Z-Z_0)^2}{2\sigma_z^2(t-t_0)}\right). \tag{125}$$

Since the removal of the vertical diffusion solver discussed in Section 3.5, the advection plus settling mode discussed in Section 3.4 assumes there is no vertical diffusion: $K_z(\tau) = 0 \Rightarrow \sigma_z(\tau) = 0$. In the limit of vanishing vertical diffusion $K_z \to 0$, the vertical direction Green's function Eq. 125 approaches a Dirac-delta function:

$$\begin{aligned}\lim_{K_z\to 0} G_Z^{\infty}(Z,t|Z_0,t_0) &= \theta(t-t_0)\lim_{\sigma_z\to 0}\frac{1}{\sqrt{2\pi}\sigma_z}\exp\left(-\frac{(Z-Z_0)^2}{2\sigma_z^2}\right)\\ &= \theta(t-t_0)\delta(Z-Z_0).\end{aligned} \tag{126}$$

For the uniform initial condition given by Eq. 124, Eq. 121b using the Green's function of Eq. 126 in the limit of no vertical diffusion yields the parcel uniform vertical concentration profile

$$\begin{aligned}C_Z(Z,t) &= \int \lim_{K_z\to 0} G_Z(Z,t|Z',t_0)C_{0,Z}(Z')\mathrm{d}Z'\\ &= \theta(t-t_0)\int \delta(Z-Z')\left[\frac{\theta(Z'-Z_1)-\theta(Z'-Z_2)}{Z_2-Z_1}\right]\mathrm{d}Z'\\ &= \theta(t-t_0)\left[\frac{\theta(Z-Z_1)-\theta(Z-Z_2)}{Z_2-Z_1}\right],\end{aligned} \tag{127}$$

with $Z = z - z_0 - \int_0^{t-t_0} \langle v_{\mathrm{p},z}(\tau + t_0)\rangle \,\mathrm{d}\tau$. Note that in the limit that $Z_2 \to Z_1 \Rightarrow z_2(t_0) \to z_1(t_0)$, the uniform distribution of Eq. 124 or Eq. 127 approaches a Dirac-delta function:

$$\lim_{Z_2\to Z_1} C_Z(Z,t_0) = \lim_{Z_2\to Z_1}\frac{\theta(Z-Z_1)-\theta(Z-Z_2)}{Z_2-Z_1} = \delta(Z-Z_0) = \delta(Z). \tag{128}$$

3.2.1.3 Full parcel concentration profile

Combining the bivariate Gaussian horizontal and uniform vertical distributions of Eqs. 123 and 127 in Eqs. 118 and 116 yields the DELFIC parcel concentration distribution in transformed and fixed Eulerian coordinates, respectively:

$$C_{XYZ}(\mathbf{X},t) = Q_0\theta(t-t_0)\left[\prod_{i=1}^{2}\frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)}\exp\left(-\frac{X_i^2}{2\sigma_i^2(t-t_0)}\right)\right]\left[\frac{\theta(Z-Z_1)-\theta(Z-Z_2)}{Z_2-Z_1}\right], \tag{129a}$$

[*] In general, the parcel bottom and top would follow a (mean) trajectory independently of the (mean) trajectory of the initial source (i.e., center of mass) $\mathbf{x}_0$, resulting in a time-dependence for Z_1 and Z_2. However, since we are assuming the particles follow the wind field with a constant offset as described in Section 3.2.1, the velocities for the trajectory of $\mathbf{x}_0$, $\mathbf{x}_1 = (x_1, y_1, z_1)$, and $\mathbf{x}_2 = (x_2, y_2, z_2)$ are the same.

$$\langle C(\mathbf{x},t)\rangle = Q_0\theta(t-t_0)\left(\prod_{i=1}^{2}\frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)}\exp\left\{-\frac{\left[x_i - x_{0,i} - \int_0^{t-t_0}\langle v_{\mathrm{p},i}(\tau+t_0)\rangle\,\mathrm{d}\tau\right]^2}{2\sigma_i^2(t-t_0)}\right\}\right)$$
$$\times\left\{\frac{\theta\left[z-z_1(t)\right]-\theta\left[z-z_2(t)\right]}{z_2(t)-z_1(t)}\right\}, \tag{129b}$$

with $z_i(t) = z_i(t_0) + \int_0^{t-t_0}\langle v_{\mathrm{p},z}(\tau+t_0)\rangle\,\mathrm{d}\tau$. The concentration distributions of Eqs. 129 can be written in terms of probability density functions (PDFs) Eqs. 229 as is shown in Appendix C. These PDF formulations are suited for generalizing to rotated concentration profiles due to wind shear using dependent, correlated distributions as shown in Section C.2. In particular, the general horizontal concentration distribution of a parcel when accounting for the effects of wind shear are given (in the fixed coordinate system) by the correlated joint distribution Eqs. 235 or 237 with the correlation function $\rho_{(x,y)}$ and deviations σ_x and σ_y determined by the rotation angle θ and parallel and perpendicular deviations $\sigma_{\parallel} = \sigma_X$ and $\sigma_{\perp} = \sigma_Y$ using Eqs. 239a and 243. Alternatively, it might be simpler to evaluate the uncorrelated distribution Eq. 241 in the frame of reference aligned and centered on the principle axes of the ellipse using the linear transform Eq. 240b with rotation matrix Eq. 238.

3.2.2 Gaussian Puff Model Variances: "Four-Thirds" Scaling Law and Fickian Diffusion

The time dependence of the diffusion coefficient can be related to the particle displacement variance $\sigma_i(\tau)$ and the (time-independent) TKE dissipation rate ε according to [40, 55]

$$K_i(\tau) = C_K\varepsilon^{1/3}\sigma_i^{4/3}(\tau), \tag{130}$$

where [2]*

$$C_K = \left[1+\left(\frac{nT_{\mathrm{L}}\langle v_{\mathrm{t}}\rangle}{L_{\mathrm{E}}}\right)^2\right]^{1/2} = \left[1+\left(\frac{n\beta\langle v_{\mathrm{t}}\rangle}{\sigma}\right)^2\right]^{1/2}, \tag{133}$$

with altitude-averaged particle settling speed $\langle v_{\mathrm{t}}\rangle$, Lagrangian time scale T_{L}, Eulerian turbulence length scale $L_{\mathrm{E}} = \sqrt{2k/3}T_{\mathrm{E}} = \sigma T_{\mathrm{E}}$, Eulerian turbulence time scale T_{E}, velocity variance $\sigma^2 = 2k/3$, $\beta \equiv T_{\mathrm{L}}/T_{\mathrm{E}} \approx 4$ [37, 61], and $n \in \{1, 2\}$ for downwind and crosswind turbulence, respectively. Typically, DELFIC treats the ratio $T_{\mathrm{L}}/L_{\mathrm{E}} = 1\,\mathrm{s\,m^{-1}}$ since this leads to good results without underpredicting dispersion [2]. The general four-thirds scaling law seen by Eq. 130 was first observed by Richardson [62] and later supported through dimensional arguments by Taylor [63]. Lin [64, 65] independently derived this four-thirds scaling law not explicitly based on the TKE dissipation rate ε but rather based on a characteristic of the field of turbulence B, which depends on the stochastic process autocorrelation function.

Regardless of whether one uses Taylor's or Kolmogorov's theory, however, Eq. 130 is contingent on the eddy length or time scales being in the inertial subrange; that is, $(\nu^3/\varepsilon)^{1/4} \equiv \eta_{\mathrm{K}} \ll \ell \ll \ell_0$ or

*Note that C_K is not the same constant as the Kolmogorov constant denoted by C [39], C_1 [40], α [37], or α_{k} [28] for the inertial subrange (3D) TKE energy (wave number) spectrum "five-thirds" scaling law [39, 40, 57, 58]

$$E(k) = C\varepsilon^{2/3}k^{-5/3}. \tag{131}$$

Further confusion might arise since the 1D TKE spectrum (denoted $E_{11}(k_1)$ [39], $E_1(k)$ [40], $S(\kappa)$ [37], or $\phi_1(k_1)$ [59]) constant is usually denoted C_1 [39] but sometimes also denoted C_2 [40], C [37], or C_κ [59], where the 3D spectrum constant is given by $C = 55C_1/18$. There is also the rather unfortunate ambiguity with the original velocity structure function covariance "two-thirds" scaling law [39, 40, 60]

$$D_{LL}(r) = D_{11}(r) = C_2\varepsilon^{2/3}r^{2/3}, \tag{132}$$

whose constant might be denoted C_2 [39] or C [40, 60]. On the other hand, C_K is denoted as c [2, 55] or $a = g^{1/3}/2$ [40], and Monin [40] states that its precise value requires further study.

$(\nu/\varepsilon)^{1/2} \equiv \tau_{\mathrm{K}} \ll \tau \ll \tau_0 \equiv \ell_0/u_0 \sim \ell_0/\sigma$ [40]. Thus, it is a general and direct consequence of the small-scale structure of turbulence and is uniquely determined by ε and $\ell = \sigma_i$ [40]. Fortunately, the turbulent atmosphere should have a sufficiently high Reynolds number [28] and wide inertial subrange that Eq. 130 is widely applicable in the atmosphere [66]. In particular, the Kolmogorov length and time scales range from $0.1\,\mathrm{mm} \lesssim \eta_{\mathrm{K}} \lesssim 10\,\mathrm{mm}$ [66,67] and $0.01\,\mathrm{s} \lesssim \tau_{\mathrm{K}} \lesssim 1\,\mathrm{s}$, respectively. The largest atmospheric length and time scales of eddies that have the most TKE range from $100\,\mathrm{m} \lesssim \ell_0 \lesssim 1\,\mathrm{km}$ [66,67] and $100\,\mathrm{s} \lesssim \tau_0 \lesssim 10^3\,\mathrm{s}$, respectively. Thus, one has typical, relatively conservative inertial subrange bounds of $1\,\mathrm{mm} \sim \eta_{\mathrm{K}} \ll \ell \ll \ell_0 \sim 100\,\mathrm{m}$ or $0.1\,\mathrm{s} \sim \tau_{\mathrm{K}} \ll \tau \ll \tau_0 \sim 100\,\mathrm{s}$, which span 5 and 3 orders of magnitude, respectively, meaning that Eq. 130 is widely applicable in the atmosphere.

Substituting Eq. 130 into Eq. 114 results in the ODE

$$C_K \varepsilon^{1/3} \sigma_i^{4/3}(\tau) = \frac{1}{2}\frac{\mathrm{d}\sigma_i^2}{\mathrm{d}\tau}. \tag{134}$$

Given the initial condition $\sigma_i(0) = \sigma_{0,i}$, which is the parcel standard deviation at cloud stabilization time and is taken to be half the parcel radius $r_{\mathrm{parcel}}/2$ supplied by the initialization and cloud rise module (ICRM) in DELFIC [2], Eq. 134 has the solution given by

$$\begin{aligned}
\sigma_i^{-4/3}(\tau)\frac{\mathrm{d}\sigma_i^2}{\mathrm{d}\tau} = \left[\sigma_i^2(\tau)\right]^{-2/3}\frac{\mathrm{d}\sigma_i^2}{\mathrm{d}\tau} &= 2C_K\varepsilon^{1/3} \\
\Rightarrow \int_0^\tau \left[\sigma_i^2(\tau')\right]^{-2/3}\frac{\mathrm{d}\sigma_i^2}{\mathrm{d}\tau'}\mathrm{d}\tau' &= 2C_K\varepsilon^{1/3}\int_0^\tau \mathrm{d}\tau' \\
\Rightarrow \int_{\sigma_i^2(0)}^{\sigma_i^2(\tau)} \left(\sigma_i^2\right)^{-2/3}\mathrm{d}\sigma_i^2 &= 2C_K\varepsilon^{1/3}\tau \\
\Rightarrow 3\left[\left(\sigma_i^2\right)^{1/3}\right]_{\sigma_i^2(0)}^{\sigma_i^2(\tau)} &= 2C_K\varepsilon^{1/3}\tau \\
\Rightarrow \left[\sigma_i^2(\tau)\right]^{1/3} - \left(\sigma_{0,i}^2\right)^{1/3} &= \frac{2}{3}C_K\varepsilon^{1/3}\tau \\
\Rightarrow \sigma_i^2(\tau) &= \left(\sigma_{0,i}^{2/3} + \frac{2}{3}C_K\varepsilon^{1/3}\tau\right)^3.
\end{aligned} \tag{135}$$

Equation 135 for the variance σ_i^2 is generally cubic in the time s and is what DELFIC uses for $\sigma_{0,i} \leq \sigma_i(\tau) \leq \sigma_\ell$, with $\sigma_\ell^2 = 10^9\,\mathrm{m}^2$ [2]. Asymptotically, there are distinct regions in which the individual polynomial terms in Eq. 135 dominate [55]:

$$\begin{aligned}
\sigma_i^2(\tau) &= \sigma_{0,i}^2 + 2\sigma_{0,i}^{4/3}C_K\varepsilon^{1/3}\tau + \frac{4}{3}\sigma_{0,i}^{2/3}C_K^2\varepsilon^{2/3}\tau^2 + \frac{8}{27}C_K^3\varepsilon\tau^3 \qquad (136) \\
&= \begin{cases}
\sigma_{0,i}^2 & 2\sigma_{0,i}^{4/3}C_K\varepsilon^{1/3}\tau \lesssim \sigma_{0,i}^2 \\
2\sigma_{0,i}^{4/3}C_K\varepsilon^{1/3}\tau & \sigma_{0,i}^2 < 2\sigma_{0,i}^{4/3}C_K\varepsilon^{1/3}\tau \wedge \frac{4}{3}\sigma_{0,i}^{2/3}C_K^2\varepsilon^{2/3}\tau^2 < 2\sigma_{0,i}^{4/3}C_K\varepsilon^{1/3}\tau \\
\frac{4}{3}\sigma_{0,i}^{2/3}C_K^2\varepsilon^{2/3}\tau^2 & 2\sigma_{0,i}^{4/3}C_K\varepsilon^{1/3}\tau < \frac{4}{3}\sigma_{0,i}^{2/3}C_K^2\varepsilon^{2/3}\tau^2 \wedge \frac{8}{27}C_K^3\varepsilon\tau^3 < \frac{4}{3}\sigma_{0,i}^{2/3}C_K^2\varepsilon^{2/3}\tau^2 \\
\frac{8}{27}C_K^3\varepsilon\tau^3 & \frac{4}{3}\sigma_{0,i}^{2/3}C_K^2\varepsilon^{2/3}\tau^2 \lesssim \frac{8}{27}C_K^3\varepsilon\tau^3
\end{cases} \\
&= \begin{cases}
\sigma_{0,i}^2 & \tau \lesssim \frac{1}{2}\sigma_{0,i}^{2/3}C_K^{-1}\varepsilon^{-1/3} \\
2\sigma_{0,i}^{4/3}C_K\varepsilon^{1/3}\tau & \frac{1}{2}\sigma_{0,i}^{2/3}C_K^{-1}\varepsilon^{-1/3} < \tau < \frac{3}{2}\sigma_{0,i}^{2/3}C_K^{-1}\varepsilon^{-1/3} \\
\frac{4}{3}\sigma_{0,i}^{2/3}C_K^2\varepsilon^{2/3}\tau^2 & \frac{3}{2}\sigma_{0,i}^{2/3}C_K^{-1}\varepsilon^{-1/3} < \tau < \frac{9}{2}\sigma_{0,i}^{2/3}C_K^{-1}\varepsilon^{-1/3} \\
\frac{8}{27}C_K^3\varepsilon\tau^3 & \tau \gtrsim \frac{9}{2}\sigma_{0,i}^{2/3}C_K^{-1}\varepsilon^{-1/3}
\end{cases}.
\end{aligned}$$

Outside of the inertial subrange, it is known [37, 38] that for long time scales the eddy diffusion coefficient will become time independent: $K_i(t) = K_i$. Thus, for such long time scales, Eq. 114 yields the solution [38, 55]

$$\sigma_i^2(\tau) = \sigma_{0,i}^2 + 2K_i\tau, \quad \tau \gg T, \tag{137}$$

where T is a suitable time scale such as the Lagrangian time scale T_{L} after which the diffusion coefficient becomes constant. Thus, DELFIC uses an equation of the form Eq. 137 for $\sigma_i(\tau) > \sigma_\ell = \sqrt{10^9\,\mathrm{m}^2}$, where this value of σ_ℓ is quite uncertain [2]. Similarly, the Lagrangian time scale T_{L} can vary over several orders of magnitude: $10^{-1}\,\mathrm{s} \lesssim T_{\mathrm{L}} \lesssim 10^4\,\mathrm{s}$. In the atmospheric boundary layer, however, more typically, $T_{\mathrm{L}} \sim 10^2\,\mathrm{s}$ [68, 69]. The condition of [2] Eq. 3.3.1a,

$$\left(\sigma_{0,i}^{2/3} + \frac{2}{3}C_K\varepsilon^{1/3}\tau\right)^{3/2} = \sigma_i(\tau) \leq \sigma_\ell = \sqrt{10^9\,\mathrm{m}^2},$$

requires that

$$\tau \leq \frac{3}{2}\left(\frac{\sigma_\ell^{2/3} - \sigma_{0,i}^{2/3}}{C_K\varepsilon^{1/3}}\right). \tag{138}$$

Suppose that $\sigma_{0,i} \sim 15 \times 10^3\,\mathrm{ft}$ for a $1\,\mathrm{Mt}$ weapon yield [17], $C_K \sim 1$, and $\varepsilon \sim 10^{-2}\,\mathrm{m}^2\,\mathrm{s}^{-3}$, as discussed in Section 3.1. Then Eq. 138 requires $\tau \lesssim 5 \times 10^3\,\mathrm{s}$. Note that Eq. 138 is relatively insensitive to changes in ε due to the $1/3$ power law, so, for example, Eq. 138 requires $\tau \lesssim 2 \times 10^3\,\mathrm{s}$ or $\tau \lesssim 10^4\,\mathrm{s}$ for $\varepsilon \sim 10^{-1}\,\mathrm{m}^2\,\mathrm{s}^{-3}$ or $\varepsilon \sim 10^{-3}\,\mathrm{m}^2\,\mathrm{s}^{-3}$, respectively. Similarly, there is a relatively weak dependence on the exact initial variance $\sigma_{0,i}$ or cloud radius due to the $2/3$ power law. Thus, the value $\sigma_\ell = \sqrt{10^9\,\mathrm{m}^2}$ used by DELFIC essentially assumes that the eddy diffusivity is constant for $\tau \gg T_{\mathrm{L}} \sim 5 \times 10^3\,\mathrm{s}$, which is probably on the higher side of typical Lagrangian time scales.

Note that when the horizontal Gaussian puff model was originally incorporated via the DTM [8], there were two methods for determining the Gaussian variances $\sigma_i^2(\tau)$, depending on a control variable `mc(10)` and whether the meteorological data supplies the eddy diffusivities K_i directly or the TKE dissipation rate ε:

1. given K_i and `mc(10) != 1`, DELFIC uses Eq. 137;
2. given ε and `mc(10) == 1`, DELFIC uses Eq. 135 with K_i given by Eq. 130.

This former method appears to have been removed in later revisions to the DTM [2], so DELFIC now unconditionally uses the latter method with ε either given by the user in the meteorological data or estimated with Eqs. 97 or 104. In the latter method when $\sigma_i(\tau) > \sigma_\ell$, DELFIC specifically uses a truncated form of Eq. 136 that includes only the constant and linear terms, analogous to the long time behavior of Eq. 137. That is, when ε is given, DELFIC uses [2] Eqs. 3.3.1a and 3.3.1b:

$$\sigma_i^2(\tau) = \begin{cases} \left(\sigma_{0,i}^{2/3} + \frac{2}{3}C_K\varepsilon^{1/3}\tau\right)^3, & \sigma_i(\tau) \leq \sigma_\ell \\ \sigma_\ell^2\left[2C_K\sigma_\ell^{-2/3}\varepsilon^{1/3}\tau + 3\left(\frac{\sigma_{0,i}^2}{\sigma_\ell^2}\right)^{1/3} - 2\right], & \sigma_i(\tau) > \sigma_\ell \end{cases}. \tag{139}$$

Note that the relation for $\sigma_i(\tau) > \sigma_\ell$ can be written in the form of Eq. 137 by defining $\sigma_i(\tau_\ell) = \sigma_\ell$ with τ_ℓ given by the right-hand side of Eq. 138, and $K_i = C_K\sigma_\ell^{4/3}\varepsilon^{1/3}$. Then shifting the origin relative to τ_ℓ, that is, $\tau' = \tau - \tau_\ell = t - t_0 - \tau_\ell$, we have for $\sigma_i(\tau) > \sigma_\ell \Rightarrow \tau' > 0 \Rightarrow \tau > \tau_\ell$

$$\begin{aligned}
\sigma_i^2(\tau) &= \sigma_i^2(\tau' + \tau_\ell) \\
&= \sigma_\ell^2 \left[2C_K \sigma_\ell^{-2/3} \varepsilon^{1/3} (\tau' + \tau_\ell) + 3 \left(\frac{\sigma_{0,i}^2}{\sigma_\ell^2} \right)^{1/3} - 2 \right] \\
&= \sigma_\ell^2 \left\{ 2C_K \sigma_\ell^{-2/3} \varepsilon^{1/3} \left[\tau' + \frac{3}{2} \left(\frac{\sigma_\ell^{2/3} - \sigma_{0,i}^{2/3}}{C_K \varepsilon^{1/3}} \right) \right] + 3 \left(\frac{\sigma_{0,i}^2}{\sigma_\ell^2} \right)^{1/3} - 2 \right\} \\
&= 2C_K \sigma_\ell^{-4/3} \varepsilon^{1/3} \tau' + \sigma_\ell^2 \left[3\sigma_\ell^{-2/3} \left(\sigma_\ell^{2/3} - \sigma_{0,i}^{2/3} \right) + 3 \left(\frac{\sigma_{0,i}^2}{\sigma_\ell^2} \right)^{1/3} - 2 \right] \\
&= 2K_i \tau' + \sigma_\ell^2. \qquad (140)
\end{aligned}$$

Furthermore, by defining $\sigma_i = \tilde{\sigma}_i \sigma_{\mathrm{c},i}$ and $\tau = \tilde{\tau} \tau_{\mathrm{c},i}$ where $\sigma_{\mathrm{c},i} = \sigma_{0,i}$ and $\tau_{\mathrm{c},i} = \sigma_{0,i}^{2/3} / \varepsilon^{1/3}$, we can write Eq. 139 in a nondimensional form as

$$\begin{aligned}
\tilde{\sigma}_i^2(\tilde{\tau}) &= \sigma_{\mathrm{c},i}^{-2} \begin{cases} \left(\sigma_{0,i}^{2/3} + \frac{2}{3} C_K \varepsilon^{1/3} \tilde{\tau} \tau_{\mathrm{c},i} \right)^3 & \tilde{\tau}\tau_{\mathrm{c},i} \leq \frac{3}{2C_K} \left(\frac{\sigma_\ell^{2/3}}{\varepsilon^{1/3}} - \tau_{\mathrm{c},i} \right) \\ \sigma_\ell^2 \left[2C_K \sigma_\ell^{-2/3} \varepsilon^{1/3} \tilde{\tau} \tau_{\mathrm{c},i} + 3 \left(\frac{\sigma_{0,i}^2}{\sigma_\ell^2} \right)^{1/3} - 2 \right] & \tilde{\tau}\tau_{\mathrm{c},i} > \frac{3}{2C_K} \left(\frac{\sigma_\ell^{2/3}}{\varepsilon^{1/3}} - \tau_{\mathrm{c},i} \right) \end{cases} \\
&= \begin{cases} \left(1 + \frac{2}{3} C_K \tilde{\tau} \right)^3 & \tilde{\tau} \leq \frac{3}{2C_K} \left(\frac{\sigma_\ell^{2/3}}{\sigma_{0,i}^{2/3}} - 1 \right) \\ \frac{\sigma_\ell^2}{\sigma_{0,i}^2} \left[2C_K \left(\frac{\sigma_{0,i}^2}{\sigma_\ell^2} \right)^{1/3} \tilde{\tau} + 3 \left(\frac{\sigma_{0,i}^2}{\sigma_\ell^2} \right)^{1/3} - 2 \right] & \tilde{\tau} > \frac{3}{2C_K} \left(\frac{\sigma_\ell^{2/3}}{\sigma_{0,i}^{2/3}} - 1 \right) \end{cases}. \qquad (141)
\end{aligned}$$

Thus, the behavior of the nondimensional Eq. 141 depends only on the ratio $\sigma_{0,i}/\sigma_\ell$ and is plotted for several such values in Figure 4. It is apparent from Figure 4 that one can see the constant, linear, quadratic, cubic, and final linear time dependence of $\tilde{\sigma}_i^2(\tilde{\tau})$ and $\sigma_i^2(\tau)$. Figure 5 similarly shows Eq. 139 for a range of ε values with a fixed initial dispersion parameter of $\sigma_{0,i} = 10^3\,\mathrm{m}$. Typical spatiotemporally averaged values of the TKE dissipation rate $10^{-5}\,\mathrm{m}^2\,\mathrm{s}^{-3} \lesssim \varepsilon \lesssim 2 \times 10^{-4}\,\mathrm{m}^2\,\mathrm{s}^{-3}$, as discussed in Section 3.1, thus correspond closely with the orange or green curves in Figure 5, with time scale $\tau_{\mathrm{c},i} \sim 5 \times 10^3\,\mathrm{s}$ and Fickian diffusion starting at approximately $\tau \sim 5 \times 10^4\,\mathrm{s}$.

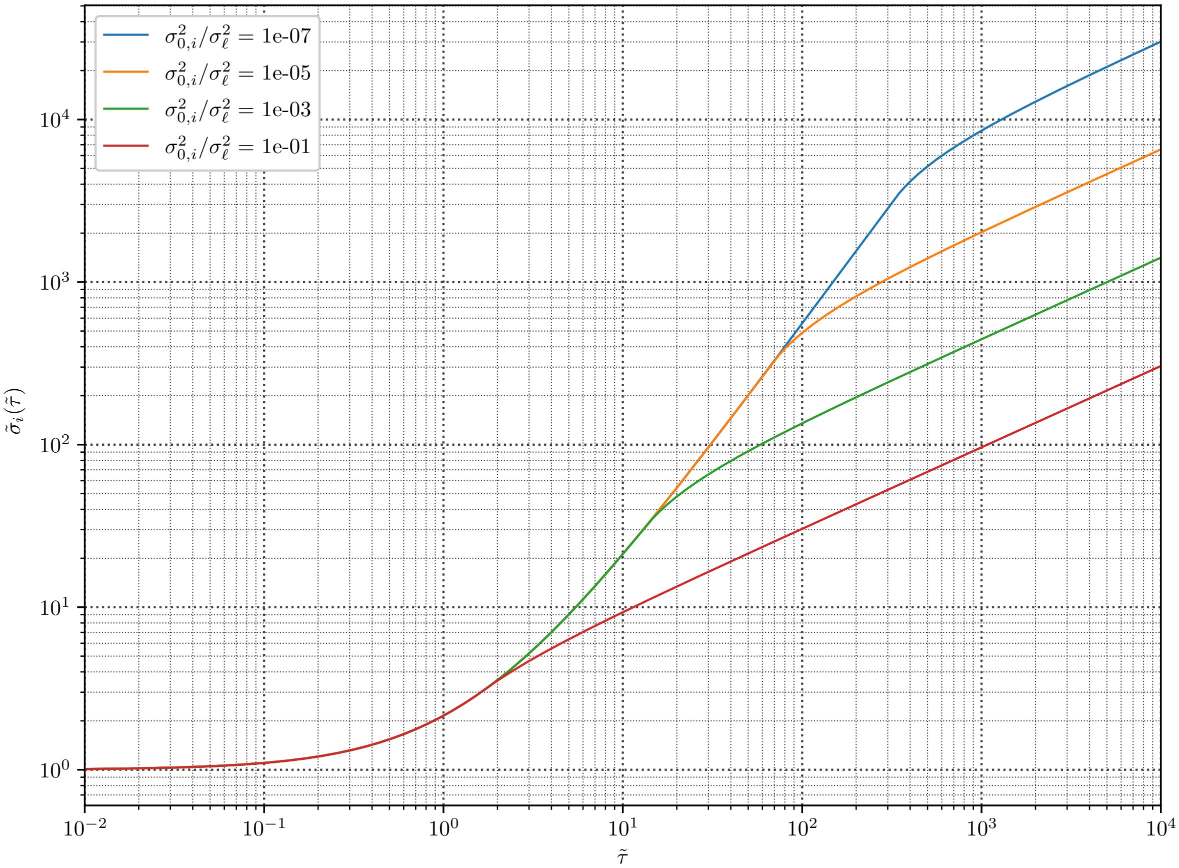

Figure 4. Nondimensional dispersion parameters $\tilde{\sigma}_i$ as a function of nondimensional time $\tilde{\tau}$ according to Eq. 141 with $C_K \sim 1$ and for $\sigma^2_{0,i}/\sigma^2_\ell \in \left\{10^{-7}, 10^{-5}, 10^{-3}, 10^{-1}\right\}$. Note that the Fickian diffusion occurs for $\tilde{\tau} \gtrsim (\sigma^2_{0,i}/\sigma^2_\ell)^{-1/3}$.

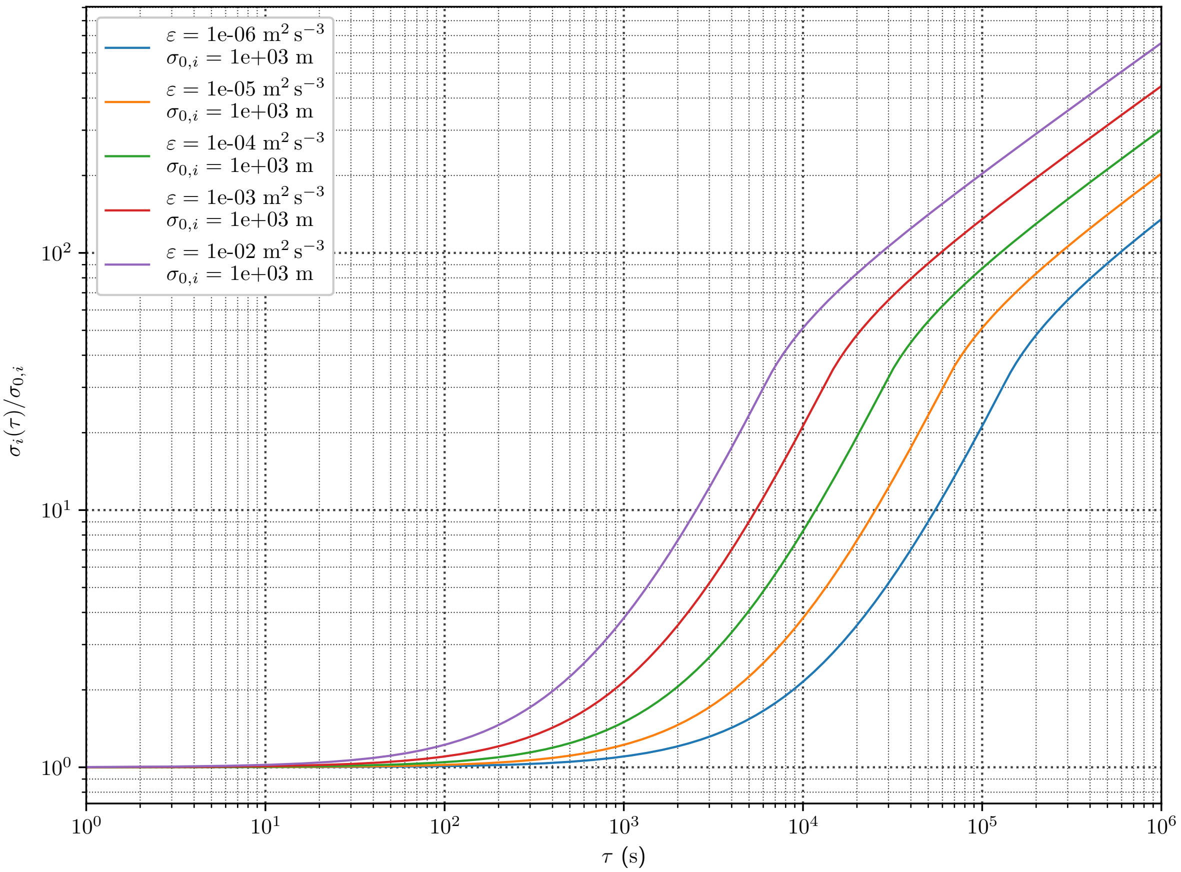

Figure 5. Dispersion parameters σ_i as a function of time τ according to Eq. 139 with $C_K \sim 1$, $\varepsilon \in \left\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\right\}\,\mathrm{m}^2\,\mathrm{s}^{-3}$, $\sigma_\ell^2 = 10^9\,\mathrm{m}^2$, and $\sigma_{0,i} = 10^3\,\mathrm{m}$.

3.3 TRANSPORT MODEL CLASSIFICATION

Recall from the discussion in Section 3.2 that the advection-diffusion model for the fallout particles is applicable only in the Eulerian continuum regime. In addition to the particle volumetric fraction α_p or its associated particle Knudsen number $\mathrm{Kn_p} = \lambda_\mathrm{p}/d_\mathrm{p}$, modeling the fallout particles as a continuum is also closely related to the Stokes number $\mathrm{St_p} \equiv \tau_\mathrm{p}/\tau_\mathrm{K}$, where $\tau_\mathrm{p} \sim \tau_\mathrm{Stokes}$ is the particle relaxation or response time, and $\tau_\mathrm{K} \equiv (\nu/\varepsilon)^{1/2}$ ($0.01\,\mathrm{s} \lesssim \tau_\mathrm{K} \lesssim 1\,\mathrm{s}$ in the atmosphere, as discussed in Section 3.2) is the Kolmogorov time scale [22, 27]. This is because particles with $\mathrm{St_p} \ll 1$ respond to the local fluid flow sufficiently rapidly that they behave as either a dusty gas [22, 70] or an equilibrium Eulerian fluid [22, 25, 26, 27]. These are both monokinetic Eulerian models—that is, the particle distribution function $f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t) = \langle n(\tilde{\mathbf{x}}, t)\rangle\, \delta\left[\tilde{\mathbf{v}} - \langle \mathbf{v}(\tilde{\mathbf{x}}, t)\rangle\right]$ is monokinetic at every point. Thus, $p_{ij} = 0$ in Eq. 250 (Appendix D) such that one need not model the fluctuating velocity covariance $\left\langle v_i' v_j' \right\rangle$ to close the particle momentum conservation equation, and the particle velocity field is uniquely determined by the local fluid flow. In the dusty gas model, the particles follow the fluid perfectly, whereas in the equilibrium Eulerian approach the particle velocity is determined by both the local fluid velocity and its gradients, independent of previous particle history [22, 27].

So there are two key parameters that determine what approach is suitable for turbulent multiphase flows:

1. the particle volumetric fraction α_p or its associated particle Knudsen number $\mathrm{Kn}_\mathrm{p} = \lambda_\mathrm{p}/d_\mathrm{p}$;
2. the particle Stokes number $\mathrm{St}_\mathrm{p} = \tau_\mathrm{p}/\tau_\mathrm{K}$ or its associated particle diameter to Kolmogorov length scale ratio $d_\mathrm{p}/\eta_\mathrm{K}$ [27], with $0.1\,\mathrm{mm} \lesssim \eta_\mathrm{K} \lesssim 10\,\mathrm{mm}$ in the atmosphere.

In the very dilute regime $\alpha_\mathrm{p} \lesssim 10^{-6}$, models can use one-way interphase coupling, which is to say that the dispersed phase particles have negligible effect on the carrier fluid, while the carrier fluid is responsible for the dispersed particle phase properties and behavior [5, 6]. In the dilute (but not very dilute) regime $10^{-6} \lesssim \alpha_\mathrm{p} \lesssim 10^{-4}$, two-way coupling in which there is mass, momentum, and energy transfer between the particle and carrier phase is important [6]. When the particle volume fraction $\alpha_\mathrm{p} \gtrsim 10^{-4}$ increases above the dilute limit such that particle–particle collisions become important, the flow is in the four-way coupled regime [6]. Finally, in the dense regime $\mathrm{Kn}_\mathrm{p} = \lambda_\mathrm{p}/d_\mathrm{p} \ll 0.01 \Rightarrow \alpha_\mathrm{p} \gg \frac{25\sqrt{\pi}}{6}\frac{\langle v_\mathrm{p}\rangle}{\sigma_v}$, the standard Eulerian hydrodynamic equations of the Eulerian–Eulerian two-fluid models (TFM) can be used [71].

Within the very dilute to moderately dense dispersed phases regimes $\alpha_\mathrm{p} \lesssim 0.1$, the particle Stokes number St_p will determine what type of model is suitable. For $\mathrm{St}_\mathrm{p} \ll 1$, Eulerian monokinetic solvers such as dusty gas or equilibrium Eulerian approaches can be used [22, 71]. When $\mathrm{St}_\mathrm{p} \gtrsim 1$, one can use advanced Eulerian models such as quadrature-based moment methods (QBMM) solutions for kinetic theory, Lagrangian models, or fully resolved DNS to model the dispersed phase [22, 71]. Figure 6 approximately quantifies these various regimes according to the two dimensionless parameters. In the atmosphere, then, one requires a very dilute to dilute dispersed phase ($\alpha_\mathrm{p} \lesssim 0.1$) with $\tau_\mathrm{p} \ll 0.1\,\mathrm{s}$ (or $d_\mathrm{p} \ll 100\,\mu\mathrm{m}$ according to Figure 1) to use either the dusty gas or equilibrium Eulerian models.

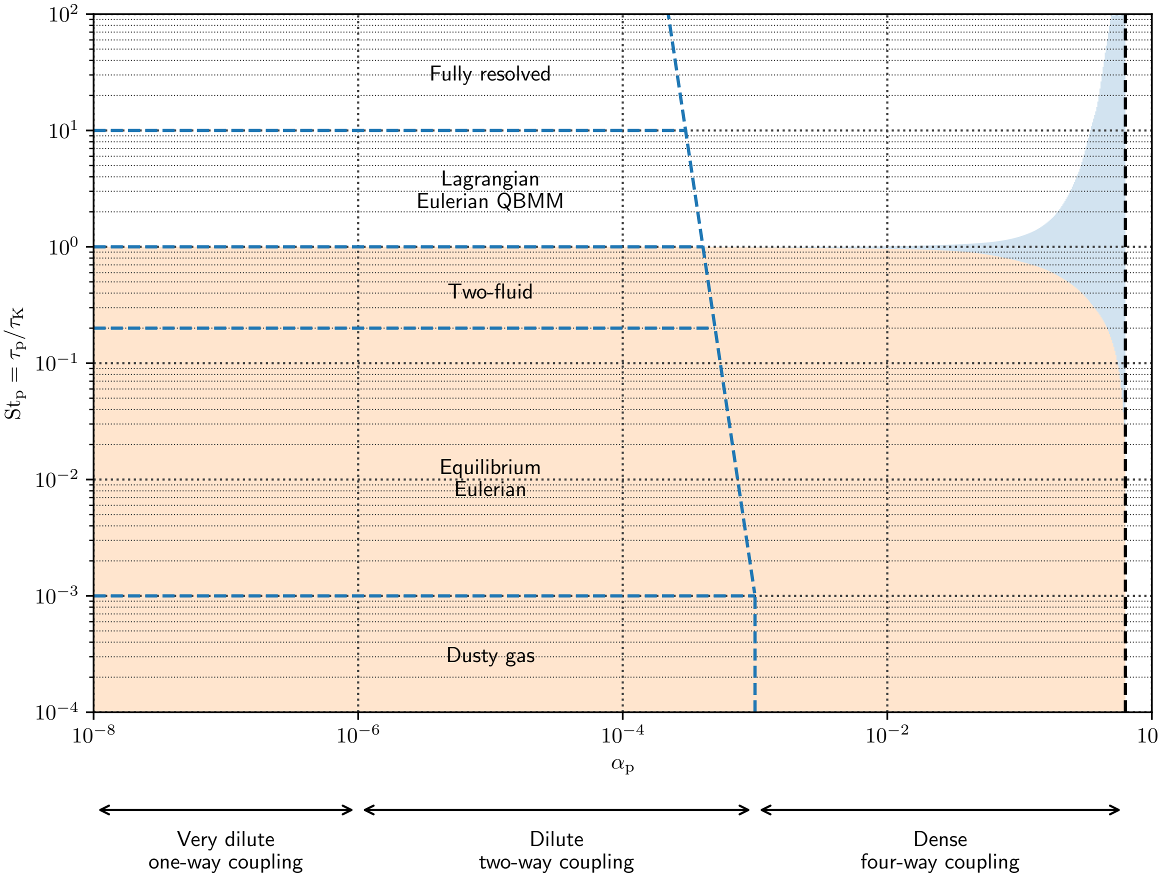

Figure 6. Turbulent multiphase flow model regimes (*abscissa*) particle volumetric fraction α_p from very dilute to dilute and moderately dense (*ordinate*) particle Stokes time scale $\mathrm{St_p} = \tau_\mathrm{p}/\tau_\mathrm{K}$. The curves and regions for the dilute and moderately dense regimes (left and right of the vertical, blue dashed curve, respectively) are adapted from Balachandar [22] Figure 1 and Kong [71] Figure 1, respectively. Kong [71] groups the regimes into Eulerian monokinetic (*orange*), Eulerian hydrodynamic (*blue*), and Lagrangian or Eulerian QBMM regions (*top white*). Balachandar [22] further divides the dilute regimes into specific Eulerian monokinetic and Lagrangian model regimes. Note that the maximum volumetric fraction $\alpha_\mathrm{p,max} < 1$ (*black vertical dashed line*) can generally approach but not reach unity.

As previously mentioned, the DELFIC DTM is a hybrid Eulerian–Lagrangian code that uses both Lagrangian particle models and equilibrium Eulerian approaches for modeling the fallout particle transport. Furthermore, the DELFIC DTM assumes there is one-way interphase coupling, which, as discussed in Section 2.1.4.1, appears to be a reasonable assumption. These models are discussed in more detail in Sections 3.4 and 3.5, with Section 3.4.2 in particular discussing some of the requirements and potential issues with this approach. On the whole, however, the hybrid Eulerian–Lagrangian DTM proves to be an efficient transport model for a wide range of particle sizes.

3.4 ORIGINAL ADVECTIVE TRANSPORT MODEL

The original advective transport model consists of two components: horizontal and vertical advection with vertical particle settling and a horizontal atmospheric diffusion model. The original ATM [10, 33] used a uniform horizontal concentration distribution rather than a Gaussian distribution and approximated the uniform widths using the Pasquill dispersion factors. We do not discuss this original model here but instead focus on the DTM [2, 8] and its advective transport model with horizontal Gaussian dispersion model.

3.4.1 Horizontal and Vertical Advection and Settling

When the vertical diffusive transport can be neglected compared with vertical advection and gravitational settling, DELFIC does not attempt to solve the separable advection-diffusion equation. Instead, DELFIC uses the original "advection plus settling" mode [10] to calculate the fallout parcel center of mass trajectory. In general, the horizontal parcel center of mass position is given by integrating the general trajectory Eq. 1:

$$\mathbf{x}_{\mathrm{p},xy}(t) - \mathbf{x}_{\mathrm{p},xy}(t_0) = \int_{t_0}^{t} \mathbf{v}_{\mathrm{p},xy}(t')\mathrm{d}t', \tag{142}$$

where $\mathbf{x}_{\mathrm{p},xy}(t) = x_{\mathrm{p}}(t)\hat{\mathbf{x}} + y_{\mathrm{p}}(t)\hat{\mathbf{y}}$ and $\mathbf{v}_{\mathrm{p},xy}(t') = v_{\mathrm{p},x}(t)\hat{\mathbf{x}} + v_{\mathrm{p},y}(t)\hat{\mathbf{y}}$ are the horizontal components of the particle position and velocity. Suppose that the horizontal particle velocity is given by the horizontal wind field velocity—that is, $\mathbf{v}_{\mathrm{p},xy}(t) = \mathbf{u}_{\mathrm{s},xy}(t) = u_{\mathrm{s},x}(t)\hat{\mathbf{x}} + u_{\mathrm{s},y}(t)\hat{\mathbf{y}}$—as assumed in Section 2.2.3. Suppose further that the time integration variable t' is parameterized linearly on height z', with the constant of proportionality given by the inverse average velocity from (z_{p}, t_0) to (z, t) for $z_{\mathrm{p}} \neq z$:

$$t' = \varphi(z') = \langle v_{\mathrm{p},z}(z,t)\rangle^{-1} z' = \left(\frac{\Delta t}{\Delta z}\right)^{-1} z' = \left(\frac{t - t_0}{|z_{\mathrm{p}} - z|}\right) z', \quad z_{\mathrm{p}} \neq z. \tag{143}$$

Equation 143 is the assumption of Norment [8] that the traverse time through the layers is proportional to the layer depth. Said differently, the vertical velocity is assumed to be constant on average as the particle travels through all the vertical layers. The velocity $\langle v_{\mathrm{p},z}(z,t)\rangle = |z_{\mathrm{p}} - z| / (t - t_0)$ is what Norment [8] refers to as the "effective settling speed" (or "effective rise speed," depending on z_{p} and z), denoted f_{e} in Norment's notation. This speed is given by the steady-state velocity from Section 2.2.3: $\langle v_{\mathrm{p},z}(z,t)\rangle = |u_{\mathrm{s},z}(t) - v_{\mathrm{f}}(z)|$. Of course, the transform Eq. 143 is undefined for $\langle v_{\mathrm{p},z}(z,t)\rangle = 0 \Leftrightarrow u_{\mathrm{s},z}(t) = v_{\mathrm{f}}(z) \Leftrightarrow z_{\mathrm{p}} = z$, in which case the horizontal trajectory is given directly by Eq. 142 with integration time $\Delta t = t - t_0$ determined from the cell or time boundary. On the other hand, provided that $\langle v_{\mathrm{p},z}(z,t)\rangle \neq 0 \Leftrightarrow u_{\mathrm{s},z}(t) \neq v_{\mathrm{f}}(z) \Leftrightarrow z_{\mathrm{p}} \neq z$, substituting $\mathbf{v}_{\mathrm{p},xy}(t) = \mathbf{u}_{\mathrm{s},xy}(t)$ and the transform Eq. 143 into Eq. 142 yields

$$\begin{aligned}
\mathbf{x}_{\mathrm{p},xy}(t) - \mathbf{x}_{\mathrm{p},xy}(t_0) &= \int_{t_0(z_{\mathrm{p}})}^{t(z)} \mathbf{u}_{\mathrm{s},xy}(t')\mathrm{d}t' \\
&= \int_{z_{\mathrm{p}}}^{z} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right] \frac{\mathrm{d}\varphi}{\mathrm{d}z'}\mathrm{d}z' \\
&= \frac{t - t_0}{|z_{\mathrm{p}} - z|} \int_{z_{\mathrm{p}}}^{z} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z', \quad z_{\mathrm{p}} \neq z.
\end{aligned} \tag{144}$$

Equation 144 is equivalent to Norment [8] Eq. 25 and is the approximate horizontal parcel center of mass given as a function of time t and destination height z.

Suppose that there are layer boundaries defined at height levels z_i and z_{i+1} for region R_i. Then the integral in Eq. 144 can be broken up across the regions R_i as the particle travels from $\mathbf{x}_{\mathrm{p}}(t_0) \in R_i$ to $\mathbf{x}_{\mathrm{p}}(t) \in R_{i'}$:

$$\int_{z_{\mathrm{p}}}^{z} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z' = \begin{cases} \int_{z_{\mathrm{p}}}^{z_{i+1}} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z' + \sum_{j=i+1}^{i'-1} \int_{z_j}^{z_{j+1}} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z' + \int_{z_{i'}}^{z} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z' & z_{\mathrm{p}} < z \\ \int_{z_{\mathrm{p}}}^{z_i} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z' + \sum_{j=i-1}^{i'+1} \int_{z_{j+1}}^{z_j} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z' + \int_{z_{i'+1}}^{z} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right]\mathrm{d}z' & z_{\mathrm{p}} > z \end{cases}. \tag{145}$$

Furthermore, if the regions have uniform, time-independent velocity, $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_{ik}$ for $\mathbf{x}, \mathbf{x}_\mathrm{p}(t) \in R_i$ and $t_k \le t \le t_{k+1}$, then we can further evaluate the integrals of Eq. 144:

$$\int_{z_\mathrm{p}}^{z} \mathbf{u}_{\mathrm{s},xy}\left[\varphi(z')\right] \mathrm{d}z' = \begin{cases} \mathbf{u}_{\mathrm{s},xy,ik}(z_{i+1} - z_\mathrm{p}) + \sum_{j=i+1}^{i'-1} \mathbf{u}_{\mathrm{s},xy,jk}\Delta z_j + \mathbf{u}_{\mathrm{s},xy,i'k}(z - z_{i'}) & z_\mathrm{p} < z \\ \mathbf{u}_{\mathrm{s},xy,ik}(z_\mathrm{p} - z_i) - \sum_{j=i-1}^{i'+1} \mathbf{u}_{\mathrm{s},xy,jk}\Delta z_j + \mathbf{u}_{\mathrm{s},xy,i'k}(z_{i'+1} - z) & z_\mathrm{p} > z \end{cases}, \tag{146}$$

where $\Delta z_i = z_{i+1} - z_i$. If a particle travels between horizontal regions while within the same vertical region, or the velocity field changes temporally while within the region R_j, then the integrals should be broken up further at these boundaries as well. Substituting Eq. 146 into Eq. 144, Eq. 144 in the abbreviated notation of Norment [8] Eq. 32 is

$$\mathbf{x}_{\mathrm{p},xy}(t) - \mathbf{x}_{\mathrm{p},xy}(t_0) = \sum_{z_i}^{z} \frac{\mathbf{u}_{\mathrm{s},xy,ik}\Delta z_i}{|u_{\mathrm{s},z}(t) - v_\mathrm{f}(z)|}, \quad u_{\mathrm{s},z}(t) \neq v_\mathrm{f}(z). \tag{147}$$

In the DELFIC implementation of subroutine `tranp` and as described by Norment [2] Section 3.2.1, Eq. 147 is called the layer-by-layer transport for spatiotemporally varying wind fields. However, for situations in which the vertical wind component is zero $u_{\mathrm{s},z}(t) = 0$, it is possible to use precalculated, vertically weighted sums of the velocity field and their averages from subroutines `sumdat` and `getda` to quickly and directly calculate the parcel transport. In either case, transport is carried out for each parcel by first dividing the parcel up into both bottom or base and top parcels to better account for varying advection and settling as a function of parcel height. The base and top parcel wafers are each represented using an ellipse with different semiaxes, and these parcel wafers are then recombined into a single deposited fallout parcel as described in Section 4.2.2. The time of impact and height of the recombined parcel is taken as the arithmetic mean of the base and top parcel wafer impact times and heights, respectively.

Note that there is a question of how to define what the time-independent velocity values within a given meteorological cell should be. Given the general velocity field $\mathbf{u}(\mathbf{x}, t)$, there are several possibilities:

- a cell-centered, point-wise value at the midpoint of the time interval,

$$\mathbf{u}_{ik} = \mathbf{u}\left(\mathbf{x}_i, \frac{t_k + t_{k+1}}{2}\right), \tag{148}$$

- a weighted interpolation,

$$\mathbf{u}_{ik} = \sum_j w_{ij}\mathbf{u}\left(\mathbf{x}_j, \frac{t_k + t_{k+1}}{2}\right), \tag{149}$$

- a volume- and time-averaged value,

$$\mathbf{u}_{ik} = \frac{1}{V_i \Delta t_k} \int_{t_k}^{t_{k+1}} \int_{R_i} \mathbf{u}(\mathbf{x}, t)\mathrm{d}\mathbf{x}\mathrm{d}t, \tag{150}$$

- and a trajectory-averaged value,

$$\mathbf{u}_{ik} = \frac{1}{\Delta t_k} \int_{t_k}^{t_{k+1}} \mathbf{u}_\mathrm{s}(t)\mathrm{d}t = \frac{1}{\Delta t_k} \int_{t_k}^{t_{k+1}} \mathbf{u}\left[\mathbf{x}_\mathrm{p}(t), t\right] \mathrm{d}t, \tag{151a}$$

$$\mathbf{x}_\mathrm{p}(t) \in R_i, \quad t_k \le t \le t_{k+1}. \tag{151b}$$

The method that DELFIC implements in subroutine `tridin` for 3D resolved data is based on a method by Cressman [72] and uses a weighted interpolation Eq. 149 with weights

$$w_{ij} = \frac{\left(\frac{\alpha^2-\Delta z_{ij}^2}{\alpha^2+\Delta z_{ij}^2}\right)\left(\frac{\beta^2-\Delta x_{ij}^2-\Delta y_{ij}^2}{\beta^2+\Delta x_{ij}^2+\Delta y_{ij}^2}\right)}{\sum_{j'}\left(\frac{\alpha^2-\Delta z_{ij'}^2}{\alpha^2+\Delta z_{ij'}^2}\right)\left(\frac{\beta^2-\Delta x_{ij'}^2-\Delta y_{ij'}^2}{\beta^2+\Delta x_{ij'}^2+\Delta y_{ij'}^2}\right)}, \tag{152}$$

where the sums are over all the input data values, α and β are the maximum vertical and horizontal distances for inclusion of a data value, and $w_{ij} < 0$ are set to zero [2]. Technically, the true particle trajectory will be determined by the 3D analog of Eq. 142. Thus, the correct velocity would be closest to Eq. 151a. However, the constraint Eq. 151b is not unique because the particle might begin at any position on the boundary $\mathbf{x}_\mathrm{p}(t_k) \in \partial R_i$. Thus, there is no unique value for the constant velocity of Eq. 151a unless the velocity itself is spatially uniform $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}(t)$. Additionally, since the meteorological conditions of the parcel are based on the center of mass position $\mathbf{x}_\mathrm{p}(t)$ but the parcel has finite extent due to the Gaussian dispersion of Section 3.4.2, there is some error associated with parcels that span multiple meteorological spatial cells with heterogeneous velocity. This latter effect can be mitigated by using sufficiently large meteorological spatial cells $\sigma_i \ll \Delta x_i$, as mentioned in Section 3.2.1. However, for general velocity fields, it would be more correct to directly integrate over the trajectory of individual (Lagrangian) particles—possibly by interpolating point-wise meteorological data—instead of treating the velocity within each meteorological region and time step as a single constant $\mathbf{u}_{ik}$.

3.4.2 Horizontal Diffusion: Gaussian Puff Model

In Section 3.4.1 we discuss the advection of the fallout parcel, which is based on the particle dynamics and kinematics of Section 2. However, the fallout particles will also undergo diffusion process caused by both molecular diffusion and (predominately) turbulent diffusion. This atmospheric dispersion in the horizontal dimension is described using the Gaussian puff model of Section 3.2. In particular, the horizontal concentration profile is given by Eq. 123. Thus, as the parcels are transported via the advection plus settling mode, they are simultaneously dispersed in the horizontal dimension as a bivariate Gaussian distribution with variances determined as a function of time by Eq. 139, as discussed in Section 3.2.2. As can be seen in Section 3.5.2, this is the same horizontal dispersion model as employed by the original DTM [8, 9].

As discussed in Section 3.3, larger particles with a relaxation time of $\tau_\mathrm{p} \gtrsim 0.1\,\mathrm{s}$ (diameter $d_\mathrm{p} \gtrsim 100\,\mu\mathrm{m}$) cannot be described by a Eulerian continuum model. Such particles must therefore settle on the ground before any significant dispersion occurs according to the Eulerian advection-diffusion Gaussian puff model dispersion parameters. Recalling from Section 3.2.2 the dispersion parameter time scale $\tau_{\mathrm{c},i} \sim 5 \times 10^3\,\mathrm{s}$ and the settling velocity $v_\mathrm{t} \gtrsim 1\,\mathrm{m\,s^{-1}}$, this limits such particles to an approximate height above the ground of $z_0 - z_\mathrm{dep} \lesssim 10\,\mathrm{km}$ at the time $\Delta t = t - t_0 \sim 5\,\mathrm{min}$ of cloud stabilization and handoff from the ICRM to the DTM. As mentioned in Section 2.1.2, typical cloud centroid heights are on the order of $h_\mathrm{cloud} \sim 10\,\mathrm{km}$, so most of these typically sized and larger particles should be able to settle before any significant dispersion occurs. For larger yield explosions or the upper regions of the fallout cloud, however, some of the typically sized particles might experience dispersion. It is thus this intermediate regime $\tau_\mathrm{p} \sim 0.1\,\mathrm{s}$ and $d_\mathrm{p} \sim 100\,\mu\mathrm{m}$ that might prove problematic for DELFIC to transport according to its hybrid Eulerian–Lagrangian model.

3.5 DIFFUSIVE TRANSPORT MODULE (DTM): VERTICAL DIFFUSION

The diffusive transport module [8, 9] is an extension of the original advective transport module that accounts for the effects of diffusion in the vertical dimension as well as the advection plus settling and horizontal diffusion. Like the original transport model, it is based on the general advection-diffusion Eq. 107 for the fallout parcels. It does not attempt to solve the full 3D, time-dependent equation because this would require more input data than are usually available and yields results that are beyond the needs of almost all fallout transport applications [8]. Instead, it makes several simplifications suited for the application to fallout particles. This leads to a set of partial differential equations: a horizontal dimension PDE that is solved semi-empirically, relying on intuitive results from Gaussian puff models, and a vertical dimension PDE that is solved numerically with a finite difference method (FDM).

3.5.1 Analytic Formulation

First, as in Section 3.2, the eddy diffusivity K_{ij} is assumed to be axis-aligned with the spatial coordinate system: $K_{ij} = K_i \delta_{ij}$, where the diffusion coefficients $K_i(\mathbf{x}, t)$ could generally depend on space and time. Then the DTM [8] uses horizontal coordinates relative to the center of mass of the fallout parcels: $\langle C(\mathbf{x}, t) \rangle = C_{XYZ}(\mathbf{X}, t)$, where $X_i = x_i - x_{0,i} - \int_0^{t-t_0} \langle v_{\mathrm{p},i}(\tau + t_0) \rangle \, \mathrm{d}\tau$ for $i \in \{1, 2\} = \{x, y\}$, and $X_3 = X_z = Z = x_3 = x_z = z$. Note that in contrast to the relative coordinates of Eq. 117 in which all three dimensions move with the mean advection, the DTM [8] treats the transformed vertical coordinate as the standard absolute vertical coordinate. Thus, the DTM [8] solves the vertical advection-diffusion equation for $C_Z(Z, t)$ in a fixed Eulerian frame, as we will see in Section 3.5.3. Substituting Eqs. 116 and 117 into Eq. 107 results in the transformed partial derivatives

$$\frac{\partial \langle C \rangle}{\partial t} = \frac{\partial C_{XYZ}}{\partial t} - \langle v_{\mathrm{p},x} \rangle \frac{\partial C_{XYZ}}{\partial X} - \langle v_{\mathrm{p},y} \rangle \frac{\partial C_{XYZ}}{\partial Y}, \tag{153a}$$

$$\frac{\partial \langle C \rangle}{\partial x_i} = \frac{\partial C_{XYZ}}{\partial X_i}, \tag{153b}$$

$$\sum_i \frac{\partial}{\partial x_i} \left(K_i \frac{\partial \langle C \rangle}{\partial x_i} \right) = \sum_i \frac{\partial}{\partial X_i} \left(K_i \frac{\partial \langle C \rangle}{\partial X_i} \right). \tag{153c}$$

Substituting Eqs. 153 into Eq. 107 with negligible molecular diffusion and a zero source* $\langle S(\mathbf{x}, t) \rangle = 0$, we obtain the transformed advection-diffusion equation

$$\frac{\partial C_{XYZ}}{\partial t} + \langle v_{\mathrm{p},z} \rangle \frac{\partial C_{XYZ}}{\partial Z} - \sum_i \frac{\partial}{\partial X_i} \left(K_i \frac{\partial C_{XYZ}}{\partial X_i} \right) = 0. \tag{154}$$

Note that Eq. 154 does not have any advection terms in the horizontal dimension since the horizontal coordinates are following the horizontal trajectory of the parcel. Next, Norment [8] assumes that the horizontal and vertical components depend only on their respective spatial coordinate and time: $K_i = K_i(x, y, t)$ for $i \in \{1, 2\}$, $K_3 = K_z = K_z(z, t)$, and $\langle v_{\mathrm{p},z} \rangle = \langle v_{\mathrm{p},z}(z, t) \rangle$. Suppose then that the solution is separable into functions depending on the horizontal and vertical coordinates, as in Section 3.2: $C_{XYZ}(\mathbf{X}, t) = Q_0 C_{XY}(X, Y, t) C_Z(Z, t)$. Substituting Eq. 118 into Eq. 154 and dividing both sides by C_{XYZ} yields

$$\begin{aligned} &\frac{1}{C_{XY}} \frac{\partial C_{XY}}{\partial t} + \frac{1}{C_Z} \frac{\partial C_Z}{\partial t} + \langle v_{\mathrm{p},z} \rangle \frac{1}{C_Z} \frac{\partial C_Z}{\partial Z} \\ -\frac{1}{C_{XY}} \frac{\partial}{\partial X} \left(K_x \frac{\partial C_{XY}}{\partial X} \right) - \frac{1}{C_{XY}} \frac{\partial}{\partial Y} \left(K_y \frac{\partial C_{XY}}{\partial Y} \right) &- \frac{1}{C_Z} \frac{\partial}{\partial Z} \left(K_z \frac{\partial C_Z}{\partial Z} \right) = 0. \end{aligned} \tag{155}$$

*This essentially requires that no additional fallout particle mass be added into the system after the initial condition $\langle C(\mathbf{x}, t_0) \rangle = \langle C_0(\mathbf{x}) \rangle$ at time t_0, which is a reasonable assumption for an initiating event of finite duration or one in which nearly all mass emission occurs over a short duration compared with the transport process.

The left and right terms on the top and bottom of Eq. 155 can be separated into two eigenvalue equations, depending on time and either the horizontal or vertical coordinates, respectively:

$$\frac{\partial C_{XY}}{\partial t} - \frac{\partial}{\partial X}\left(K_x \frac{\partial C_{XY}}{\partial X}\right) - \frac{\partial}{\partial Y}\left(K_y \frac{\partial C_{XY}}{\partial Y}\right) = \lambda(t) C_{XY}, \tag{156a}$$

$$\frac{\partial C_Z}{\partial t} + \langle v_{\mathrm{p},z}\rangle \frac{\partial C_Z}{\partial Z} - \frac{\partial}{\partial Z}\left(K_z \frac{\partial C_Z}{\partial Z}\right) = -\lambda(t) C_Z, \tag{156b}$$

where $\lambda(t)$ is a time-dependent generalized eigenvalue. Equations 156 correspond to Norment [8] Eqs. 15 and 16 when $\lambda(t) = 0$. This also leads to the same vertical equation as presented by Monin [73], but Monin does not make clear how he derives or justifies this equation.

3.5.2 Horizontal Dimension Semi-Empirical Gaussian Solution

For the horizontal function C_{XY}, Norment [8] begins with Eq. 156a with $\lambda(t) = 0$:

$$\frac{\partial C_{XY}}{\partial t} - \frac{\partial}{\partial X}\left(K_x \frac{\partial C_{XY}}{\partial X}\right) - \frac{\partial}{\partial Y}\left(K_y \frac{\partial C_{XY}}{\partial Y}\right) = 0, \tag{157}$$

which is a source-free diffusion equation following the parcel center of mass in the horizontal dimension. Under the assumptions of an infinite horizontal domain with spatially independent (but still time-dependent) horizontal eddy diffusivities $K_x = K_x(t)$ and $K_y = K_y(t)$, we can use the solution technique of Section 3.2 to solve Eq. 157. In particular, the Green's function solution to Eq. 157 is Eq. 121a, and for a Dirac delta initial condition $C_{0,XY}(\mathbf{X}_{XY}) = \delta(\mathbf{X}_{XY})$, the horizontal concentration distribution is given by Eq. 123, as in Section 3.4.2. As discussed in Section 3.2, DELFIC uses the piecewise Eq. 139 for determining the Gaussian variances $\sigma_i^2(\tau)$, where the eddy diffusivity depends on the TKE dissipation rate ε. When not provided in the meteorological data, ε can be determined using a hyperpolic scaling law with Eqs. 97 or 104. This piecewise function for the Gaussian variances should be valid for both short and long time scales, with the short time scales having a general cubic time dependence and long time scales characterized by Fickian diffusion with linear time dependence and constant eddy diffusivity.

3.5.3 Vertical Dimension Numerical Solution

For the vertical function C_Z, Norment [8] begins with Eq. 156b, again with $\lambda(t) = 0$:

$$\frac{\partial C_Z}{\partial t} + \langle v_{\mathrm{p},z}\rangle \frac{\partial C_Z}{\partial Z} - \frac{\partial}{\partial Z}\left(K_z \frac{\partial C_Z}{\partial Z}\right) = 0, \tag{158}$$

where $\langle v_{\mathrm{p},z}\rangle = -v_{\mathrm{f}} + u_z$. Equation 158 is a source-free advection-diffusion equation with spatiotemporally varying eddy diffusivity $K_z = K_z(z,t)$. Thus, it is a second-order, linear, parabolic PDE with variable coefficients and cannot be solved analytically. Instead, it must be solved using numerical methods—DELFIC in particular uses an implicit FDM that is a generalization of the simple Crank-Nicholson method.

Equation 158 can be written in terms of operators as

$$\mathcal{L}_t C_Z = \mathcal{L}_Z C_Z, \tag{159}$$

where

$$\mathcal{L}_t = \frac{\partial}{\partial t}, \tag{160a}$$

$$\mathcal{L}_Z = -\langle v_{\mathrm{p},z}\rangle \frac{\partial}{\partial Z} + \frac{\partial}{\partial Z} K_z \frac{\partial}{\partial Z}. \tag{160b}$$

First we define a uniformly discretized spatial grid $Z_i = Z_0 + i\Delta Z$ for $0 \leq i \leq N_Z$, and we define the vector

$$\mathbf{C}_Z(t) = [C_{Z,0}(t), \ldots, C_{Z,N_z}(t)] \tag{161}$$

of concentrations $C_{Z,i}(t) = C_Z(Z_i, t)$ evaluated at each spatial grid point i as a function of time t. Then DELFIC defines the discretized spatial operator as [9]

$$\begin{aligned} \tilde{\mathcal{L}}_Z C_{Z,i}(t) = &-\frac{1}{\Delta Z}\left[\langle v_{\mathrm{p},z,i+1}(t)\rangle C_{Z,i+1}(t) - \langle v_{\mathrm{p},z,i}(t)\rangle C_{Z,i}(t)\right] \\ &+ \frac{1}{2\left(\Delta Z\right)^2}\left[\left(K_{z,i+1} + K_{z,i}\right) C_{Z,i+1}(t) - \left(K_{z,i+1} + 2K_{z,i} + K_{z,i-1}\right) C_{Z,i}(t)\right. \\ &\left. + \left(K_{z,i} + K_{z,i-1}\right) C_{Z,i-1}(t)\right], \end{aligned} \tag{162}$$

where $\Delta Z = \Delta Z_i = Z_{i+1} - Z_i$, $\langle v_{\mathrm{p},z,i}(t)\rangle = \langle v_{\mathrm{p},z}(Z_i, t)\rangle$, and $K_{z,i} = K_z(Z_i)$. The diffusion operator is discretized using a second-order (in ΔZ) central difference method, while the advection operator is discretized using a first-order forward difference method. Both the basic diffusion and advection equations and their simple finite difference analyses can be found in Holmes [74]. A first-order method is probably used for the advection term to avoid oscillatory behavior as discussed in, for example, Strikwerda [75] and LeVeque [76]. As was recognized computationally by Norment [8] and shown analytically by Strikwerda [75] and LeVeque [76], using a second-order advection operator introduces numerical diffusion and results in equivalent convergence constraints as the first-order discretized advection operator [74, 75]:

$$\frac{K_z \Delta t}{\left(\Delta Z\right)^2} \leq \frac{1}{2}, \tag{163a}$$

$$\frac{\left|\langle v_{\mathrm{p},z}\rangle\right| \Delta t}{\Delta Z} \leq 1. \tag{163b}$$

Equations 163a and 163b are the convergence conditions for the discretized diffusion and advection operators, respectively, where Eq. 163b is called the Courant-Friedrichs-Lewy (CFL) condition. Using the maximum time step value of $\Delta t = \left(\Delta Z\right)^2 / 2K_z$ from Eq. 163a and substituting into Eq. 163b yields the discretized advection-diffusion operator constraint

$$\frac{\left|\langle v_{\mathrm{p},z}\rangle\right| \Delta Z}{2K_z} \leq 1, \tag{164}$$

which is also noted by Norment [8] Eq. 68. For a detailed and general analysis of the error, stability, and convergence of the diffusion discretization, see, for example, Samarskii [77, 78] or Jovanović [79].

We can then write Eq. 159 for each grid point i as the ODE

$$\begin{aligned} &\mathcal{L}_t C_{Z,i}(t) = \tilde{\mathcal{L}}_Z C_{Z,i}(t) \\ &\Rightarrow \frac{\mathrm{d}C_{Z,i}}{\mathrm{d}t} = F\left[\mathbf{C}_Z(t), t\right], \end{aligned} \tag{165}$$

where $F\left[\mathbf{C}_Z(t), t\right] = \tilde{\mathcal{L}}_Z C_{Z,i}(t)$. Equation 165 can be solved at times $t_{k+1} = t_0 + k\Delta t$ for $0 \leq k \leq N_t$ using the theta method [74, 80] with uniform time step Δt:

$$\mathbf{C}_{Z,k+1} = \mathbf{C}_{Z,k} + \Delta t\left[(1-\theta)F(\mathbf{C}_{Z,k}, t_k) + \theta F(\mathbf{C}_{Z,k+1}, t_{k+1})\right], \tag{166}$$

where $\mathbf{C}_{Z,k} = \mathbf{C}_Z(t_k)$. The theta method of Eq. 166 is second-order in Δt when $\theta = 1/2$ and corresponds to the trapezoidal method; otherwise, the method is first order in Δt. In combination with the spatial discretization of Eq. 162, this FDM can be considered a generalization of the Crank-Nicholson method for spatially varying diffusion coefficients K_z and with the addition of an advection operator.

As noted in Section 1 and the beginning of Section 3, this vertical diffusion solver was eventually removed in the 1979 and later versions of the DELFIC DTM [2, 3]. It is possible, although unclear, whether this was done because the complexity and computational cost was not worthwhile compared with the simpler original advective model with horizontal diffusion. However, as discussed in Section 3.2.1 and as was recognized by the DTM authors [8], there are some circumstances in which the vertical diffusion coupling cannot be neglected and for which the vertical concentration distribution is not Gaussian from the effects of vertical wind shear, atmospheric stability conditions, and the ground boundary. Vertical diffusion might be particularly significant for small particles $d_\mathrm{p} \ll 100\,\mathrm{\mu m}$ in comparison with the vertical advection from particle settling, as demonstrated in Section 4.2.3.3. Nonetheless, since at least the 1979 revision of DELFIC, the concentration distribution of each parcel is treated as a bivariate Gaussian distribution in the horizontal and a uniform distribution[*] in the vertical since vertical diffusion is neglected. Indeed, the vertical diffusion solver was the predominate contribution of the DTM, and its removal essentially reverts back to the original advective transport model except for an improved horizontal dispersion model.

[*] See, for example, Norment [8] Eq. 44, which initializes the vertical concentration profile as a uniform distribution from parcel bottom to top and is depicted in Norment [8] Figure 6.

4. DELFIC DEPOSITION MODEL

Since at least the 1979 revision of DELFIC with the removal of the DTM vertical diffusion solver, the DELFIC deposition model consists of the simpler "advection plus settling" mode of deposition. In this mode, the bottom and top of the parcels are independently transported using the advection plus settling transport model discussed in Section 3.4.1 and depicted by "AS" in Norment [8] Figure 4. The parcel tops and bottoms thus independently undergo their transport and horizontal dispersion and are then recombined to record the ground deposition contribution due to the individual parcel according to Norment [8] Eqs. 87–89 (Norment [2] Eqs. 3.4.1–3.4.3). The resulting ground deposition concentration profile due to an individual parcel is approximated as a bivariate Gaussian distribution the covariances of which are determined geometrically by the ellipse semi-major and semi-minor axes of the combined ellipse depicted in Figure 7. In Section 4.1 we present the method for predicting the deposited concentration profile at ground level due to atmospheric fallout parcels. Then in Section 4.2 we compare the analytical deposition profiles due to the atmospheric concentration fields of a single fallout parcel presented in Section 3.2.1 to those predicted by DELFIC using a geometric approximation.

4.1 DEPOSITION FLUX

The mass deposition (with positive defined upward) at the surface $z = 0$ is given by the vertical mass flux from advection and diffusion (i.e., Fick's law):

$$j_z(x, y, 0, t) = \langle v_{\mathrm{p},z}(t)\rangle \langle C(x, y, 0, t)\rangle - K_z(t) \left[\frac{\partial \langle C\rangle}{\partial z}\right]_{z=0}. \tag{167}$$

Then the deposited ground mass concentration at time t is given by

$$C_{\mathrm{dep}}(x, y, t) = -\int_{t_0}^{t} j_z(x, y, 0, t')\mathrm{d}t'. \tag{168}$$

Note that we negate the flux since the depositing flux is moving in the negative z direction. For notational simplicity we have also neglected the use of ensemble average brackets $\langle\cdot\rangle$ on the (mean) deposition flux and concentration. The deposited concentration profile thus depends on the

- vertical particle velocity $v_{\mathrm{p},z}(\mathbf{x}, t)$,
- mean parcel air concentration $\langle C(\mathbf{x}, t)\rangle$,
- vertical eddy diffusivity $K_z(t)$, and
- mean vertical parcel air concentration gradient $\partial \langle C(\mathbf{x}, t)\rangle / \partial z$,

all evaluated at the ground level $z = 0$.

Although DELFIC employs Eq. 167 for determining the deposited ground concentration as shown in Section 4.2, it is more common to employ an empirical model for dry deposition in which the mass flux $\mathbf{j}_{\mathrm{dep}}$ is proportional to the atmospheric concentration $\langle C(\mathbf{x}, t)\rangle$ at some reference height z_{dep}, with the proportionality constant being the deposition velocity $\mathbf{v}_{\mathrm{dep}}$:

$$\mathbf{j}_{\mathrm{dep}}(x, y, z_{\mathrm{dep}}, t) = -\mathbf{v}_{\mathrm{dep}}(z_{\mathrm{dep}}, t) \langle C(x, y, z_{\mathrm{dep}}, t)\rangle. \tag{169}$$

Note that the deposition flux is negated so that by convention the deposition velocity v_{dep} is positive for a depositing substance moving in the negative (i.e., downward) direction. The deposition velocity can be estimated using the resistance method [19]. For particles, one has an aerodynamic resistance $r_{\mathrm{a}} = r_{\mathrm{a}}(z)$, quasilaminar layer resistance $r_{\mathrm{b}} = r_{\mathrm{b}}(z)$, virtual resistance $r_{\mathrm{a}} r_{\mathrm{b}} v_{\mathrm{t}}$, and a settling velocity $v_{\mathrm{t}} = v_{\mathrm{t}}(z)$ resistance in parallel [19]:

$$v_{\mathrm{dep}} = r^{-1} = (r_{\mathrm{a}} + r_{\mathrm{b}} + r_{\mathrm{a}} r_{\mathrm{b}} v_{\mathrm{t}})^{-1} + v_{\mathrm{t}}. \tag{170}$$

In general, these will also implicitly depend on horizontal position and time and according to atmospheric conditions. By equating the vertical flux components of Eq. 167 and Eq. 169, one has the more general boundary condition

$$\begin{aligned} j_{\mathrm{dep}}(x, y, z_{\mathrm{dep}}, t) &= \langle v_{\mathrm{p},z}(z_{\mathrm{dep}}, t) \rangle \langle C(x, y, z_{\mathrm{dep}}, t) \rangle - K_z(z_{\mathrm{dep}}, t) \left[\frac{\partial \langle C \rangle}{\partial z} \right]_{z = z_{\mathrm{dep}}} \\ &= -v_{\mathrm{dep}}(z_{\mathrm{dep}}, t) \langle C(x, y, z_{\mathrm{dep}}, t) \rangle . \end{aligned} \tag{171}$$

Written in this way, the deposition velocity v_{dep} can be seen as a parameter that characterizes the interaction between the diffusing particles and the surface [73]. Specifically:

- for $v_{\mathrm{dep}} = 0 \Rightarrow j_{\mathrm{dep}}(x, y, z_{\mathrm{dep}}, t) = 0$, the mass concentration remains airborne, corresponding to a perfectly reflecting boundary condition;
- for $v_{\mathrm{dep}} \to \infty \Rightarrow \langle C(x, y, z_{\mathrm{dep}}, t) \rangle = 0$, the airborne concentration vanishes at z_{dep}, corresponding to a perfectly absorbing boundary condition.

Thus, allowing the deposition flux to vary continuously according to Eq. 167 (as DELFIC does) is analogous to having a partially absorbing boundary condition. For example, in the absence of vertical diffusion ($K_z \to 0$), Eq. 171 corresponds to the finite deposition velocity $v_{\mathrm{dep}}(z_{\mathrm{dep}}, t) = - \langle v_{\mathrm{p},z}(z_{\mathrm{dep}}, t) \rangle$.

4.2 DEPOSITION CONCENTRATION PROFILE

Given a mass concentration profile $\langle C(\mathbf{x}, t) \rangle$ for an individual parcel, we can use Eq. 168 to predict the ground-deposited mass concentration $C_{\mathrm{dep}}(x, y, t)$ with vertical mass flux given by either Eq. 167 or Eq. 171. For the original advection plus settling transport mode of DELFIC without vertical diffusion, these are equivalent, with the deposition velocity given by the parcel's settling velocity. These theoretical deposition concentration profiles for individual parcels in the absence of wind shear are presented in Section 4.2.1, while Section 4.2.2 discusses the actual geometric approximation that DELFIC uses. Finally, Section 4.2.3 compares and discusses the differences between these deposition models.

4.2.1 Theoretical

In the absence of vertical diffusion $K_z = 0$, the mass flux is entirely due to advection with no mass flux due to diffusion. We present the theoretical deposition profiles for two different vertical concentration profiles $C_Z(Z, t)$ in the absence of wind shear: a Dirac-delta profile as well as a uniform profile as discussed in Section 3.2.1.2. The former vertical profile will be useful for understanding the DELFIC geometric approximation discussed in Section 4.2.2 because the resulting deposition profile is a bivariate Gaussian distribution. The uniform vertical profile, however, probably results in a more accurate prediction of the theoretical deposited concentration profile since each parcel does in fact have a uniform vertical mass distribution. Finally, to see the effects of vertical diffusion on the deposition of small particles, we look at the deposition of a Dirac-delta profile with vertical eddy diffusivity given by Eqs. 130 and 139.

4.2.1.1 Dirac-delta profile without vertical diffusion

Given the Dirac-delta initial condition in the vertical $C_Z(Z, t_0) = C_{0,Z}(Z) = \delta(Z)$, Eq. 121b using the Green's function of Eq. 126 yields the Dirac-delta vertical profile $C_Z(Z, t) = \delta(Z)$. Then the surface mass flux is given by

$$j_z(x,y,0,t) = \theta(t-t_0)Q_0 \left[\prod_{i=1}^{2} \frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)} \exp\left(-\frac{X_i^2}{2\sigma_i^2(t-t_0)}\right)\right] \\ \times \langle v_{\mathrm{p},z}\rangle \, \delta\left[-z_0(t_0) - \langle v_{\mathrm{p},z}\rangle (t-t_0)\right]. \tag{172}$$

Recalling that X_i of Eq. 117 depend on t for $i \in \{1,2,3\}$, substituting Eq. 172 into Eq. 168 yields the deposited mass concentration

$$C_{\mathrm{dep}}(x,y,t) = -\frac{Q_0}{2\pi} \int_{t_0}^{t} \left[\prod_{i=1}^{2} \frac{1}{\sigma_i(t'-t_0)} \exp\left(-\frac{X_i^2}{2\sigma_i^2(t'-t_0)}\right)\right] \\ \times \langle v_{\mathrm{p},z}\rangle \, \delta\left[-z_0(t_0) - \langle v_{\mathrm{p},z}\rangle (t'-t_0)\right] \mathrm{d}t'. \tag{173}$$

Examining Eq. 173, we can see that the Dirac-delta functions force the integrand to zero everywhere except $-z_0(t_0) - \langle v_{\mathrm{p},z}\rangle (t'-t_0) = 0 \Rightarrow t'-t_0 = -z_0(t_0)/\langle v_{\mathrm{p},z}\rangle$. Equation 173 can then be evaluated using the generalized Dirac-delta scaling property $\int_{-\infty}^{\infty} f(x)\delta(g(x))\mathrm{d}x = \sum_i \frac{f(x_i)}{|g'(x_i)|}$, where x_i are the (simple) roots of g. In particular, for sufficiently large times, Eq. 173 becomes a bivariate Gaussian distribution:

$$C_{\mathrm{dep}}(x,y,t) = -\frac{Q_0 \mathrm{sgn}\,\langle v_{\mathrm{p},z}\rangle}{2\pi} \theta\left[t - t_0 + \frac{z_0(t_0)}{\langle v_{\mathrm{p},z}\rangle}\right] \\ \times \prod_{i=1}^{2} \frac{1}{\sigma_i\left(-\frac{z_0(t_0)}{\langle v_{\mathrm{p},z}\rangle}\right)} \exp\left\{-\frac{\left[x_i - x_{0,i} - \langle v_{\mathrm{p},i}\rangle \left(-\frac{z_0(t_0)}{\langle v_{\mathrm{p},z}\rangle}\right)\right]^2}{2\sigma_i^2\left(-\frac{z_0(t_0)}{\langle v_{\mathrm{p},z}\rangle}\right)}\right\}, \tag{174}$$

where $\mathrm{sgn}x = x/|x|$ is the signum or sign function. Of course, one requires that $\langle v_{\mathrm{p},z}\rangle < 0$ so that we will eventually have some deposition and for which $\mathrm{sgn}\,\langle v_{\mathrm{p},z}\rangle = -1 \Rightarrow C_{\mathrm{dep}}(x,y,t) \geq 0$. Without loss of generality, we can set $t_0 = 0$, $x_0 = y_0 = 0$, and align the x axis with the horizontal velocity such that $\langle v_{\mathrm{p},y}\rangle = 0$. To better understand Eq. 174, we will first nondimensionalize some variables by defining $C_{\mathrm{dep}} = \tilde{C}_{\mathrm{dep}} C_{\mathrm{c}}$ and $t = \tilde{t} t_{\mathrm{c}}$, where $C_{\mathrm{c}} = -Q_0 \mathrm{sgn}\,\langle v_{\mathrm{p},z}\rangle / 2\pi\sigma_x(0)\sigma_y(0)$ and $t_{\mathrm{c}} = -z_0(t_0)/\langle v_{\mathrm{p},z}\rangle$. Then Eq. 174 can be written in terms of these nondimensional variables as

$$\tilde{C}_{\mathrm{dep}}(x,y,\tilde{t}) = \theta\left(\tilde{t} t_{\mathrm{c}} - t_{\mathrm{c}}\right) \prod_{i=1}^{2} \frac{\sigma_i(0)}{\sigma_i\left(t_{\mathrm{c}}\right)} \exp\left\{-\frac{\left[x_i - \langle v_{\mathrm{p},i}\rangle t_{\mathrm{c}}\right]^2}{2\sigma_i^2\left(t_{\mathrm{c}}\right)}\right\}. \tag{175}$$

4.2.1.2 Uniform profile without vertical diffusion

Substituting Eqs. 123 and 127 into Eqs. 118, 116, and 167 with $K_z(\tau) = 0$ yields the surface mass flux

$$j_z(x,y,0,t) = \theta(t-t_0)Q_0 \left[\prod_{i=1}^{2} \frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)} \exp\left(-\frac{X_i^2}{2\sigma_i^2(t-t_0)}\right)\right] \\ \times \langle v_{\mathrm{p},z}\rangle \left\{\frac{\theta\left[-z_1(t_0) - \langle v_{\mathrm{p},z}\rangle (t-t_0)\right] - \theta\left[-z_2(t_0) - \langle v_{\mathrm{p},z}\rangle (t-t_0)\right]}{Z_2 - Z_1}\right\}. \tag{176}$$

Recalling that X_i depend on t for $i \in \{1,2,3\}$ and substituting Eq. 176 into Eq. 168 yields the deposited mass concentration

$$C_{\mathrm{dep}}(x,y,t) = -\frac{Q_0}{2\pi} \int_{t_0}^{t} \left[\prod_{i=1}^{2} \frac{1}{\sigma_i(t'-t_0)} \exp\left(-\frac{X_i^2}{2\sigma_i^2(t'-t_0)}\right)\right] \\ \times \langle v_{\mathrm{p},z}\rangle \left\{\frac{\theta\left[-z_1(t_0) - \langle v_{\mathrm{p},z}\rangle (t'-t_0)\right] - \theta\left[-z_2(t_0) - \langle v_{\mathrm{p},z}\rangle (t'-t_0)\right]}{Z_2 - Z_1}\right\} \mathrm{d}t'. \tag{177}$$

Without loss of generality, we can set $t_0 = 0$, $x_0 = y_0 = 0$, and align the x axis with the horizontal velocity such that $\langle v_{\mathrm{p},y} \rangle = 0$. To better understand Eq. 177, we will first nondimensionalize some variables by defining $C_{\mathrm{dep}} = \tilde{C}_{\mathrm{dep}} C_{\mathrm{c}}$ and $t = \tilde{t} t_{\mathrm{c}}$, where $C_{\mathrm{c}} = Q_0 / 2\pi\sigma_x(0)\sigma_y(0)$ and $t_{\mathrm{c}} = -z_0 / \langle v_{\mathrm{p},z} \rangle$. Then Eq. 177 can be written in terms of these nondimensional variables as

$$\begin{aligned} \tilde{C}_{\mathrm{dep}}(x, y, \tilde{t}) &= \frac{z_0}{z_2(0) - z_1(0)} \int_0^{\tilde{t}} \left[\prod_{i=1}^{2} \frac{\sigma_i(0)}{\sigma_i(\tilde{t}' t_{\mathrm{c}})} \exp\left(-\frac{X_i^2}{2\sigma_i^2(\tilde{t}' t_{\mathrm{c}})} \right) \right] \left\{ \theta\left[z_0 \tilde{t}' - z_1(0) \right] - \theta\left[z_0 \tilde{t}' - z_2(0) \right] \right\} \mathrm{d}\tilde{t}' \\ &= \frac{z_0}{z_2(0) - z_1(0)} \begin{cases} 0 & \tilde{t} \le z_1(0)/z_0 \\ \int_{z_1(0)/z_0}^{\tilde{t}} \prod_{i=1}^{2} \frac{\sigma_i(0)}{\sigma_i(\tilde{t}' t_{\mathrm{c}})} \exp\left(-\frac{X_i^2}{2\sigma_i^2(\tilde{t}' t_{\mathrm{c}})} \right) \mathrm{d}\tilde{t}' & z_1(0)/z_0 \le \tilde{t} \le z_2(0)/z_0 \\ \int_{z_1(0)/z_0}^{z_2(0)/z_0} \prod_{i=1}^{2} \frac{\sigma_i(0)}{\sigma_i(\tilde{t}' t_{\mathrm{c}})} \exp\left(-\frac{X_i^2}{2\sigma_i^2(\tilde{t}' t_{\mathrm{c}})} \right) \mathrm{d}\tilde{t}' & \tilde{t} \ge z_2(0)/z_0 \end{cases} , \end{aligned} \tag{178}$$

where $X = x - \langle v_{\mathrm{p},x} \rangle \tilde{t} t_{\mathrm{c}}$ and $Y = y$. Unfortunately, the integrals in Eq. 178 cannot be analytically solved for general $\sigma_i(\tau)$ or for Eq. 139. Clearly, however, $\tilde{C}_{\mathrm{dep}}(x, y, \tilde{t})$ is generally not a Gaussian deposition concentration profile for a Gaussian horizontal and uniform vertical atmospheric concentration profile.* On the other hand, recalling the limit $Z_2 \to Z_1 \Rightarrow z_2(t_0) \to z_1(t_0)$ for the uniform distribution Eq. 128, Eq. 178 reduces to Eq. 175 in the limit. In practice, this requires that the uniform distribution be sufficiently small compared with the vertical distance:

$$\begin{aligned} Z_2 - Z_1 = z_2(t_0) - z_1(t_0) \ll z_0 \\ \Rightarrow \frac{z_2(t_0) - z_1(t_0)}{z_0} \ll 1, \end{aligned} \tag{179}$$

where z_0 is the initial (center) height of the parcel, and $z_1(t_0)$ and $z_2(t_0)$ are the bottom and top of the parcel at t_0. Then the uniform distribution described by Eq. 127 approximates a Dirac-delta distribution provided that, for example, $z_2(t_0) - z_1(t_0) \sim 100\,\mathrm{m}$ and $z_0 \sim 10\,\mathrm{km}$.

4.2.1.3 Dirac-delta profile with vertical diffusion

Small particles $d_{\mathrm{p}} \ll 100\,\mathrm{\mu m}$ have a much smaller settling velocity, so vertical diffusion can have a more significant affect on their transport than vertical advection, as demonstrated in Section 4.2.3.3. For a semi-infinite domain with perfectly absorbing boundary at $z = 0$, we require that the atmospheric concentration be zero on the $z = 0$ plane. Thus, the semi-infinite domain Green's function is the result of adding the negative of the solution for the image source $\langle S(\mathbf{x}, t) \rangle = \delta(\mathbf{x} - \mathbf{x}_0')\delta(t - t_0)$ located at $\mathbf{x}_0' = (x_0, y_0, -z_0)$ and traveling with the opposite velocity $-\langle v_{\mathrm{p},z}(t) \rangle$ [19]:

$$\begin{aligned} G_z(z, t | z_0, t_0) = \theta(t - t_0) \frac{1}{\sqrt{2\pi}\sigma_z(t - t_0)} \Bigg[&\exp\left(-\frac{\left(z - z_0 - \int_0^{t - t_0} \langle v_{\mathrm{p},z}(\tau + t_0) \rangle \mathrm{d}\tau\right)^2}{2\sigma_z^2(t - t_0)} \right) \\ &- \exp\left(-\frac{\left(z + z_0 + \int_0^{t - t_0} \langle v_{\mathrm{p},z}(\tau + t_0) \rangle \mathrm{d}\tau\right)^2}{2\sigma_z^2(t - t_0)} \right) \Bigg], \end{aligned}$$

or by redefining $Z = z$ and $Z_0 = z_0 + \int_0^{t - t_0} \langle v_{\mathrm{p},z}(\tau + t_0) \rangle \mathrm{d}\tau$,

$$G_Z(Z, t | z_0, t_0) = \theta(t - t_0) \frac{1}{\sqrt{2\pi}\sigma_z(t - t_0)} \left[\exp\left(-\frac{(Z - Z_0)^2}{2\sigma_z^2(t - t_0)} \right) - \exp\left(-\frac{(Z + Z_0)^2}{2\sigma_z^2(t - t_0)} \right) \right]. \tag{180}$$

*If $\sigma_{0,i} \ll \sigma_\ell \ll \sigma_i(\tau)$ such that $\sigma_i(\tau) \sim \sqrt{2K_i\tau}$, Eq. 178 could be evaluated in terms of the error function $\mathrm{erf} z = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} \mathrm{d}t$.

Then analogously to Eq. 123 from Section 3.2.1.1, the vertical concentration profile of Eq. 121b for a Dirac-delta initial condition given by

$$C_Z(Z,t_0) = C_{0,Z}(Z) = \delta(Z) \tag{181}$$

at time t_0 with perfectly absorbing boundary condition at $Z = z = 0$ is

$$C_Z(Z,t) = \theta(t-t_0)\frac{1}{\sqrt{2\pi}\sigma_z(t-t_0)}\left[\exp\left(-\frac{(Z-Z_0)^2}{2\sigma_z^2(t-t_0)}\right) - \exp\left(-\frac{(Z+Z_0)^2}{2\sigma_z^2(t-t_0)}\right)\right]. \tag{182}$$

Since the atmospheric concentration at ground level is by definition zero for the perfectly absorbing boundary at $z = 0$, the mass flux at the ground according to Eq. 167 is entirely due to diffusion with no contribution from advection. Differentiating Eq. 182 with respect to $Z = z$, we obtain the surface mass flux

$$\begin{aligned} j_z(x,y,0,t) &= -Q_0 C_{XY}(X,Y,t)K_z(t)\left[\frac{\partial C_Z(Z,t)}{\partial Z}\right]_{Z=0} \\ &= \theta(t-t_0)Q_0\left[\prod_{i=1}^{2}\frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)}\exp\left(-\frac{X_i^2}{2\sigma_i^2(t-t_0)}\right)\right] \\ &\times \frac{K_z(t)}{\sqrt{2\pi}\sigma_z(t-t_0)}\left[\frac{Z-Z_0}{\sigma_z^2(t-t_0)}\exp\left(-\frac{(Z-Z_0)^2}{2\sigma_z^2(t-t_0)}\right) - \frac{Z+Z_0}{\sigma_z^2(t-t_0)}\exp\left(-\frac{(Z+Z_0)^2}{2\sigma_z^2(t-t_0)}\right)\right]_{Z=0} \\ &= -\theta(t-t_0)Q_0\left[\prod_{i=1}^{2}\frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)}\exp\left(-\frac{X_i^2}{2\sigma_i^2(t-t_0)}\right)\right] \\ &\times \frac{2K_z(t)Z_0}{\sqrt{2\pi}\sigma_z^3(t-t_0)}\exp\left(-\frac{Z_0^2}{2\sigma_z^2(t-t_0)}\right). \end{aligned} \tag{183}$$

Then the deposited concentration is given by substituting Eq. 183 into Eq. 168, keeping in mind that X, Y, and Z_0 are functions of t:

$$C_{\text{dep}}(x,y,t) = \frac{2}{(2\pi)^{3/2}}Q_0\int_{t_0}^{t}\left[\prod_{i=1}^{2}\frac{1}{\sigma_i(t'-t_0)}\exp\left(-\frac{X_i^2}{2\sigma_i^2(t'-t_0)}\right)\right]\frac{K_z(t')Z_0}{\sigma_z^3(t'-t_0)}\exp\left(-\frac{Z_0^2}{2\sigma_z^2(t'-t_0)}\right)\mathrm{d}t'. \tag{184}$$

Unfortunately, as in Section 4.2.1.2, we cannot analytically evaluate the integral of Eq. 184. As in Section 4.2.1.2, we can without loss of generality set $t_0 = 0$, $x_0 = y_0 = 0$, and align the x axis with the horizontal velocity such that $\langle v_{\text{p},y}\rangle = 0$. In this case, we can nondimensionalize Eq. 184 by setting $C_\text{c} = 2Q_0/(2\pi)^{3/2}\sigma_x(0)\sigma_y(0)$ and $t_\text{c} = -z_0/\langle v_{\text{p},z}\rangle$, resulting in the nondimensional equation

$$\tilde{C}_{\text{dep}}(x,y,\tilde{t}) = t_\text{c}\int_0^{\tilde{t}}\left[\prod_{i=1}^{2}\frac{\sigma_i(0)}{\sigma_i(\tilde{t}'t_\text{c})}\exp\left(-\frac{X_i^2}{2\sigma_i^2(\tilde{t}'t_\text{c})}\right)\right]\frac{K_z(\tilde{t}'t_\text{c})Z_0}{\sigma_z^3(\tilde{t}'t_\text{c})}\exp\left(-\frac{Z_0^2}{2\sigma_z^2(\tilde{t}'t_\text{c})}\right)\mathrm{d}\tilde{t}', \tag{185}$$

where $X = x - \langle v_{\text{p},x}\rangle\,\tilde{t}t_\text{c}$, $Y = y$, and $Z_0 = z_0 + \langle v_{\text{p},z}\rangle\,\tilde{t}t_\text{c} = z_0(1-\tilde{t})$.

4.2.2 Geometric Combined Ellipse Profile

DELFIC divides each parcel into both bottom and top portions at altitudes z_1 and z_2, respectively, each portion carrying half of the parcel mass. These parcel portions undergo transport from the central points (x_0, y_0, z_i) with $i \in \{1, 2\}$ until their deposition on the ground centered at $(x_{\text{dep},1}, y_{\text{dep},1})$ and $(x_{\text{dep},2}, y_{\text{dep},2})$ with rotation angles θ_1 and θ_2 due to wind shear. Once deposited on the ground, the correlated, bivariate Gaussian concentration distributions for each portion are geometrically represented as ellipses of the one standard deviation probability curve, as depicted by the red ellipses in Figure 7 with semi-major and semi-minor axis lengths $\sigma_{\|,i}$ and $\sigma_{\perp,i}$, respectively. Appendix C shows the formulation of the bivariate Gaussian concentration distributions written in terms of PDFs, as discussed in Section 3.2.1.3. DELFIC then combines the portions into a single bivariate Gaussian distribution with its variances determined geometrically according to the large, blue ellipse depicted in Figure 7 with semi-major and semi-minor axis lengths $\sigma_{\|}$ and $\sigma_{\perp}$ as well as rotation angle θ.

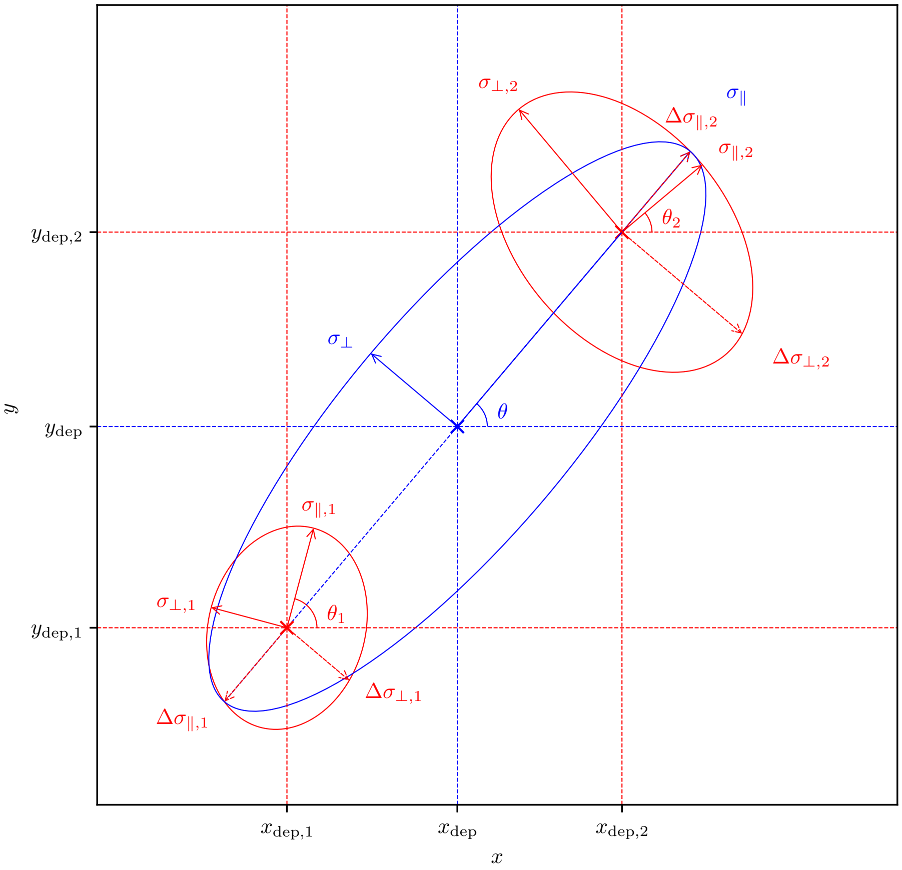

Figure 7. DELFIC geometric combined deposit ellipse based on Norment [2] Figure 2 or Norment [8] Figure 9. The red and blue ellipses represent the the split and combined DELFIC parcel deposit ellipses, respectively, with their centers shown by the red and blue crosses. The combined ellipse semi-major axis length is determined by the outermost points between the bottom and top ellipses on the major axis, while the semi-minor axis length is based on a geometric mean of the distances of the bottom and top ellipses along the minor axis.

When accounting for rotation from nonuniform wind fields, the bottom and top distributions' axes might not align with the combined distribution's axes. Then to construct the combined ellipse Gaussian concentration distribution, the major axis is first aligned with the line connecting the central impact points in the deposition plane of the bottom $(x_{\text{dep},1}, y_{\text{dep},1})$ and top $(x_{\text{dep},2}, y_{\text{dep},2})$ of the parcel at an angle

$$\theta = \arctan\left(\frac{y_{\text{dep},2} - y_{\text{dep},1}}{x_{\text{dep},2} - x_{\text{dep},1}}\right), \tag{186}$$

while the minor axis is perpendicular to this line. The major axis length is determined by the sum of the distance between the bottom and top impact points and the variances of the bottom and top bivariate Gaussian distributions along this major axis. That is, the lengths along the combined distribution's major axis are determined by the outermost points intersecting the bottom and top ellipses along the major axis:

$$\sigma_{\parallel} = \sigma_X = \frac{1}{2}\left[\Delta\sigma_{\parallel,1} + \Delta\sigma_{\parallel,2} + \sqrt{(x_{\text{dep},2} - x_{\text{dep},1})^2 + (y_{\text{dep},2} - y_{\text{dep},1})^2}\right], \tag{187a}$$

$$\Delta\sigma_{\parallel,i} = \left[\frac{\cos^2(\theta_i - \theta)}{\sigma^2_{\parallel,i}} + \frac{\sin^2(\theta_i - \theta)}{\sigma^2_{\perp,i}}\right]^{-1/2}, \tag{187b}$$

where Eq. 187b is the distance from the top or bottom ellipse centers to their outer edges along the combined ellipse major axis using the polar coordinate specification of an ellipse. The semi-minor axis length, on the other hand, is constructed by taking the geometric mean of the perpendicular component of the bottom and top distribution variances:

$$\sigma_{\perp} = \sigma_Y = \sqrt{\Delta\sigma_{\perp,1}\Delta\sigma_{\perp,2}}, \tag{188a}$$

$$\Delta\sigma_{\perp,i} = \left[\frac{\sin^2(\theta_i - \theta)}{\sigma^2_{\parallel,i}} + \frac{\cos^2(\theta_i - \theta)}{\sigma^2_{\perp,i}}\right]^{-1/2}, \tag{188b}$$

Eq. 188b being the distances from the bottom and top ellipse centers to their outer edges along the combined ellipse minor axis. Using Eqs. 186, 187a, and 188a, the approximate combined ellipse deposited concentration profile $C_{\text{dep,DELFIC}}(x, y, t)$ is given by the bivariate Gaussian distribution Eqs. 235 or 237 (or in the frame of the principle axes, Eq. 241) centered at

$$\begin{bmatrix} x_{\text{dep}} \\ y_{\text{dep}} \end{bmatrix} = \begin{bmatrix} x_{\text{dep},1} \\ y_{\text{dep},1} \end{bmatrix} + (\sigma_{\parallel} - \Delta\sigma_{\parallel,1}) \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}. \tag{189}$$

The center and axes lengths of the combined ellipse do not necessarily produce a physically correct deposition concentration profile but rather represent a reasonable geometric approximation to the aggregate distribution of a sufficiently small parcel (in the vertical dimension) in the absence of vertical diffusion, as shown in Section 4.2.3.

4.2.3 Analytical Comparison

For purposes of comparing the theoretical deposition profiles to the deposited concentration predicted by DELFIC, we will make some simplifying assumptions about the particle velocity field. In particular, we will consider a uniform, time-invariant wind field $\langle \mathbf{u}(\mathbf{x}, t)\rangle = \langle \mathbf{u}\rangle$ such that the particle velocity field is a constant $\mathbf{v}_{\text{p}}(\mathbf{x}, t) = \mathbf{u} + \mathbf{w}$ throughout the fallout particle cloud, where $\mathbf{w}$ is the velocity difference due to particle settling. Thus, each point within the parcel follows the same (mean) trajectory with a constant offset determined by the initial position. We furthermore suppose that the vertical wind velocity component u_z is negligible compared with the particle terminal velocity $|u_z| \ll v_{\text{t}} \Rightarrow \langle v_{\text{p},z}\rangle = -v_{\text{t}}$ and assume that we have a relatively calm atmosphere with horizontal wind speed $u_{xy} = 1\,\text{m}\,\text{s}^{-1}$.

In Sections 4.2.3.1 and 4.2.3.2, we consider sufficiently large particles $d_{\text{p}} \gtrsim 100\,\mu\text{m}$ that settle before any significant dispersion occurs. Equivalently, we could use sufficiently small particles $d_{\text{p}} \ll 100\,\mu\text{m}$ such that the dusty gas model or equilibrium Eulerian approach could be applied. However, this significantly increases the transport distance and time scale and is unnecessary for the analytical comparison in the absence of vertical diffusion. For a $d_\mathrm{p} \sim 100\,\mathrm{\mu m}$ particle, according to Figure 20 we have $v_\mathrm{t} \sim 1\,\mathrm{m\,s^{-1}}$. In Section 4.2.3.3, we instead look at small particles $d_\mathrm{p} \sim 10\,\mathrm{\mu m}$ for which $v_\mathrm{t} \sim 10^{-2}\,\mathrm{m\,s^{-1}}$ to show the effects of vertical diffusion where vertical advection is small.

4.2.3.1 Dirac-delta profile without vertical diffusion

Under the conditions described, the theoretical deposition concentration profile $\tilde{C}_\mathrm{dep}(x, y, \tilde{t})$ at $\tilde{t} = 1$ according to Eq. 175 for the Dirac-delta parcel mass vertical profile is shown in Figure 8. Additionally drawn in black are the contours of equal concentration and one standard deviation curves for the DELFIC bottom and top deposit ellipse portions in red as well as the combined deposit ellipse in blue. Similarly, the maxima of Eq. 175 and the DELFIC distributions are shown by the black, red, and blue crosses. Since the initial parcel mass vertical profile is represented here as a single Dirac-delta distribution at $z_0(0) = 10\,\mathrm{km}$, the "top" and "bottom" parcel increments are actually the same, depositing at the same time and location. Thus, the combined ellipse is also the same as that of the individual portions in this case. Furthermore, the combined ellipse exactly matches that of the one-sigma contour for $\tilde{C}_\mathrm{dep}(x, y, \tilde{t})$, so the DELFIC concentration profile $\tilde{C}_\mathrm{dep,DELFIC}(x, y, \tilde{t})$ exactly matches that predicted by Eq. 175 for a Dirac-delta vertical mass profile without vertical diffusion.

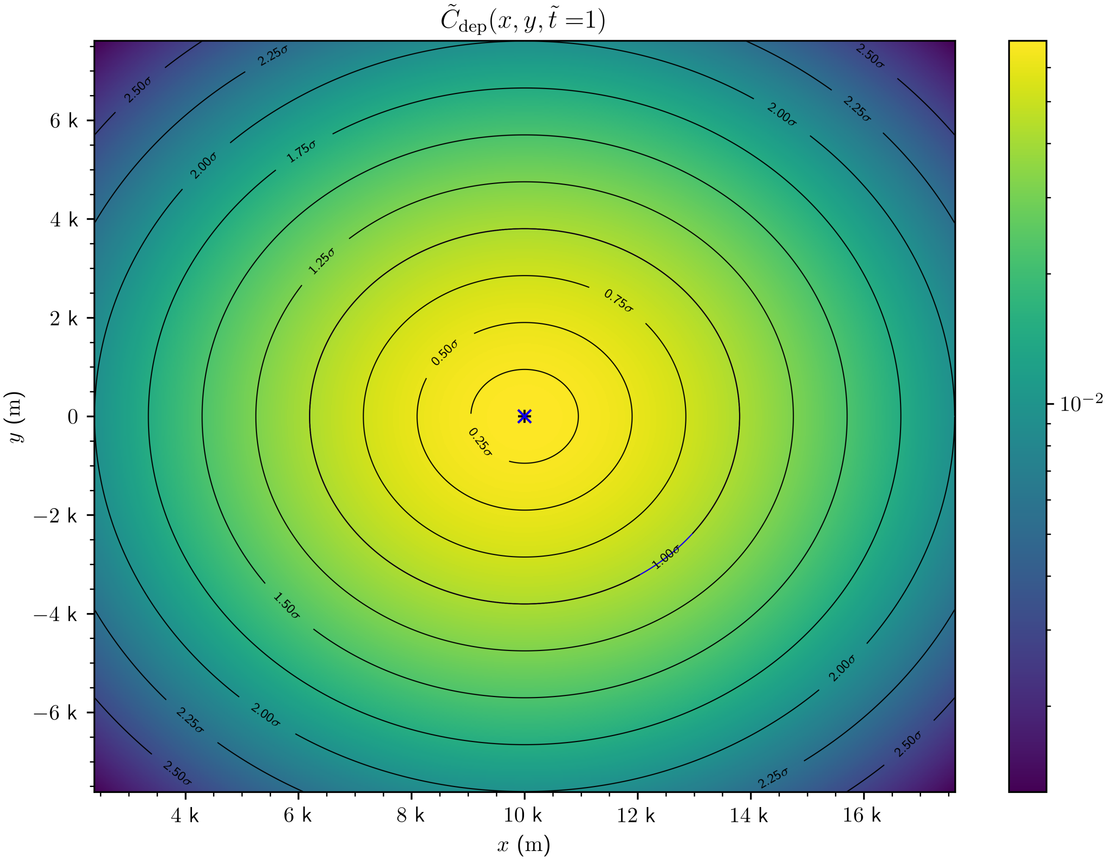

Figure 8. Deposition profile $\tilde{C}_{\rm dep}(x, y, \tilde{t})$ of Eq. 175 at $\tilde{t} = 1$ with $\langle v_{{\rm p},x} \rangle = -\langle v_{{\rm p},z} \rangle = 1\,{\rm m\,s^{-1}}$, $z_0(0) = 10\,{\rm km}$. Contours are drawn in black at concentrations scaled by the standard normal distribution every quarter standard deviation, that is, the curves $\tilde{C}_{\rm dep}(x, y, \tilde{t}) = \sqrt{2\pi}\varphi(\xi_i) \max_{(x,y)} \tilde{C}_{\rm dep}(x, y, \tilde{t})$ for $\xi_i \in \{n/4 : n \in \mathbb{N}^+\}$, where $\varphi(x) = e^{-x^2/2}/\sqrt{2\pi}$ is the standard normal probability density function. Red and blue contours represent the one-sigma concentrations of the split and synthesized DELFIC bivariate Gaussian distributions, respectively. The black cross is located at the maximum, that is, $\arg\max_{(x,y)} \tilde{C}_{\rm dep}(x, y, \tilde{t})$; similarly, the red and blue crosses are the center points of the DELFIC split and synthesized distributions, respectively.

4.2.3.2 Uniform profile without vertical diffusion

We can numerically calculate $\tilde{C}_{\rm dep}(x, y, \tilde{t})$ according to Eq. 178 for the uniform mass vertical profile at times $0 \leq \tilde{t} \leq z_2(0)/z_0$ to cover the full deposition profile behavior. Figure 9 shows in particular the deposition profile $\tilde{C}_{\rm dep}(x, y, \tilde{t})$ at $\tilde{t} = z_2(0)/z_0$ under analogous conditions to those in Section 4.2.3.1, Figure 8. From Figure 9, we can see that the DELFIC synthesized ellipse has a larger semi-major axis length than the predicted one-sigma deviation according to Eq. 178. Figure 10 furthermore shows the relative difference of the normalized DELFIC concentration $\tilde{C}_{\rm dep,DELFIC}(x, y, \tilde{t})$ determined according to the approximated geometric distribution described in Section 4.2.2 compared with the predicted concentration profile of Eq. 178. Thus, it is apparent that DELFIC underpredicts the central peak concentration while overpredicting the concentration outside the one-sigma region in the direction along the wind. Over the region shown by Figure 9, we observe the 2-norm relative difference $\epsilon_2 = \left\|\tilde{C}_{\mathrm{dep,DELFIC}} - \tilde{C}_{\mathrm{dep}}\right\|_2 / \left\|\tilde{C}_{\mathrm{dep}}\right\|_2 = 5.5 \times 10^{-2}$ and the max-norm relative difference $\epsilon_\infty = \left\|\tilde{C}_{\mathrm{dep,DELFIC}} - \tilde{C}_{\mathrm{dep}}\right\|_\infty / \left\|\tilde{C}_{\mathrm{dep}}\right\|_\infty = 5.3 \times 10^{-2}$, indicating a roughly 5% relative difference between the approximate geometric and the theoretical deposited concentration profiles for a uniform vertical mass profile without vertical diffusion.

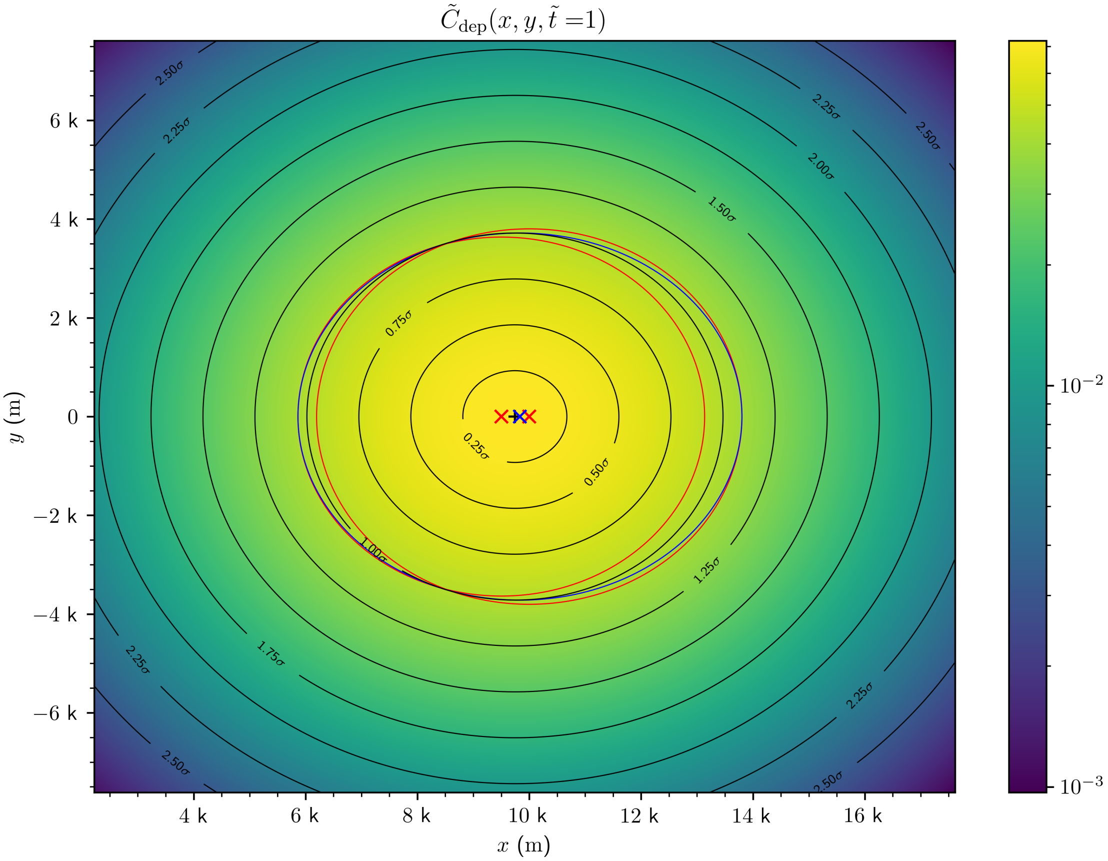

Figure 9. Deposition profile $\tilde{C}_{\mathrm{dep}}(x, y, \tilde{t})$ of Eq. 178 at $\tilde{t} = z_2(0)/z_0 = 1$ with $\langle v_{\mathrm{p},x} \rangle = -\langle v_{\mathrm{p},z} \rangle = 1\,\mathrm{m\,s^{-1}}$**,** $z_1(0) = 9.5\,\mathrm{km}$**,** $z_2(0) = 10\,\mathrm{km}$**.** Contours are drawn in black at concentrations scaled by the standard normal distribution every quarter standard deviation, that is, the curves $\tilde{C}_{\mathrm{dep}}(x, y, \tilde{t}) = \sqrt{2\pi}\varphi(\xi_i) \max_{(x,y)} \tilde{C}_{\mathrm{dep}}(x, y, \tilde{t})$ for $\xi_i \in \{n/4 : n \in \mathbb{N}^+\}$, where $\varphi(x) = e^{-x^2/2}/\sqrt{2\pi}$ is the standard normal probability density function. Red and blue contours represent the one-sigma concentrations of the split and synthesized DELFIC bivariate Gaussian distributions, respectively. The black cross is located at the maximum, that is, $\arg\max_{(x,y)} \tilde{C}_{\mathrm{dep}}(x, y, \tilde{t})$; similarly, the red and blue crosses are the center points of the DELFIC split and synthesized distributions, respectively.

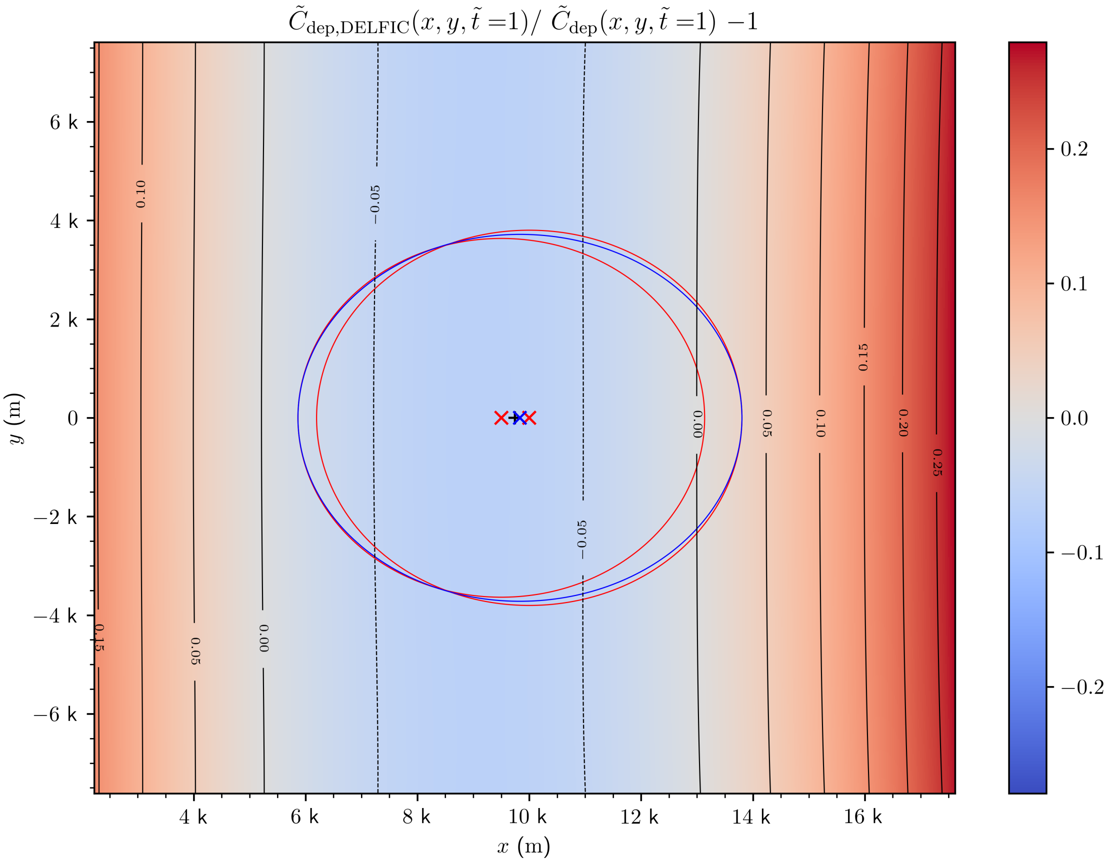

Figure 10. DELFIC deposition profile relative difference $\tilde{C}_{\mathrm{dep,DELFIC}}(x, y, \tilde{t}) / \tilde{C}_{\mathrm{dep}}(x, y, \tilde{t}) - 1$ (relative to Eq. 175) at $\tilde{t} = z_2(0)/z_0 = 1$ with $\langle v_{\mathrm{p},x} \rangle = -\langle v_{\mathrm{p},z} \rangle = 1\,\mathrm{m\,s^{-1}}$, $z_1(0) = 9.5\,\mathrm{km}$, $z_2(0) = 10\,\mathrm{km}$. Red and blue contours represent the one-sigma concentrations of the split and synthesized DELFIC bivariate Gaussian distributions, respectively. The black cross is located at the maximum, that is, $\arg\max_{(x,y)} \tilde{C}_{\mathrm{dep}}(x, y, \tilde{t})$; similarly, the red and blue crosses are the center points of the DELFIC split and synthesized distributions, respectively.

4.2.3.3 Dirac-delta profile with vertical diffusion

We have numerically plotted the corresponding normalized profile shape of Eq. 185 with vertical eddy diffusivity given by Eqs. 130 and 139 in Fig 11. Specifically, the top view shows a zoomed out picture of the hypothetical impact point due to advection of z_0 alone, while the bottom view shows the region where most of the actual deposition occurs due to the diffusive flux. Thus, there is a significantly different deposition profile when accounting for vertical diffusion in addition to advection for such small particles of $d_{\mathrm{p}} \ll 100\,\mu\mathrm{m}$. In particular, most of the deposition occurs much closer to the initial release than would be predicted by advection alone. However, comparing Figures 8 or 9 to Figure 11, we can see that the scale of peak deposition concentrations is about an order of magnitude larger for the larger particles settling in the absence of vertical diffusion than for the smaller particles with vertical diffusion.

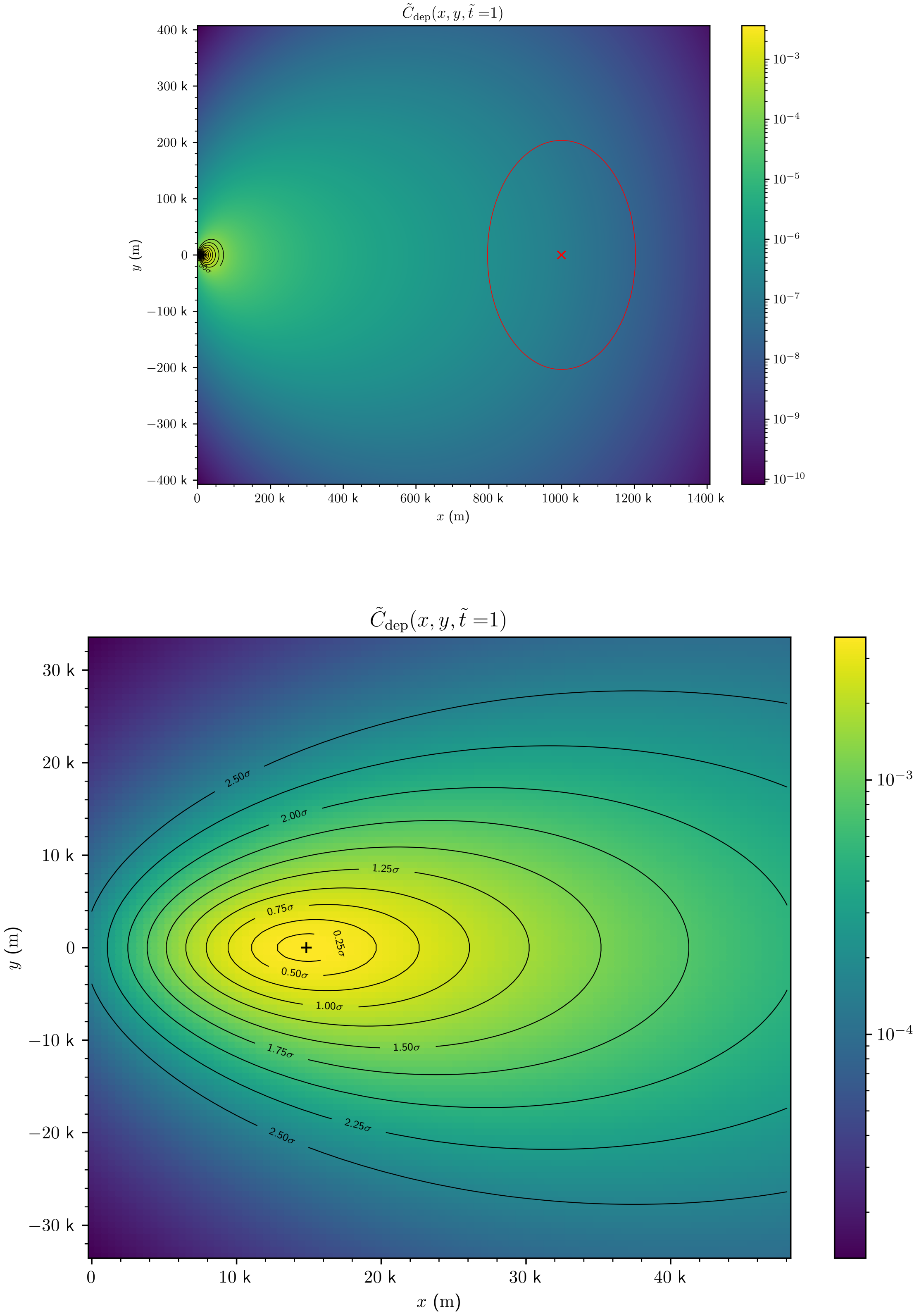

Figure 11. Deposition profile $\tilde{C}_{\mathrm{dep}}(x, y, \tilde{t})$ of Eq. 185 at $\tilde{t} = 1$ with $\langle v_{\mathrm{p},x} \rangle = -100 \langle v_{\mathrm{p},z} \rangle = 1\,\mathrm{m\,s^{-1}}$ and $z_0 = 10\,\mathrm{km}$. (*top*) view showing one-sigma ellipse and "impact point" of z_0 due to advection alone; (*bottom*) zoomed in view of region where most of the actual deposition occurs due to diffusive flux.

5. DELFIC DTM IMPLEMENTATION

In this section we describe the subroutines relevant to the 1979 DELFIC DTM. The 1979 version of DELFIC is organized as shown in Norment [3] Figure 1, with descriptions of the subroutines given in Norment [3] Table 2. Table 1 lists the DTM subroutines in particular along with their descriptions, callers, and corresponding code listing page number in both the current document and the original manual [3].

Table 1. DTM subroutines, their descriptions, and code listing page numbers from the original manual [3].

Subroutine	Description	Called by	Page ([3] printed/PDF)
`advec`	For a particular fallout parcel, `call`s subroutine `tranp` to transport the parcel base and top from the stabilized height to the ground, recombining the base and top as described in Section 4.2.2 to form a deposit increment	`sprvs`	1 on page 69 (104/108)
`boun`	Calculates the horizontal coordinates of the entry point on the boundary into a meteorological spatial cell of a fallout parcel	`tranp`	2 on page 70 (106/110)
`calib`	Returns an index related to the position of a point within a data array	`tranp`	3 on page 71 (107/111)

`cntr`	Returns the horizontal coordinates $(x_{\text{met},j}, y_{\text{met},j})$ of the center of a meteorological spatial cell	`sumdat`
		`tranp`
		`tridin`
			4 on page 71
			(107/111)

`datin`	Processes the input of wind and turbulence meteorological data	`dtmex`	5 on page 72 (108/112)
`dtmex`	DTM executive	`delfic`	6 on page 75 (111/115)
`dtmint`	Initializes the DTM	`dtmex`	7 on page 76 (114/118)
`dumper`	Writes deposit increment data onto unit `ipout` as well as printing to `isout` if requested	`advec` `sprvs`	8 on page 78 (116/120)
`getda`	Computes the vertically averaged meteorological data component from vertically weighted sum data in subroutine `sumdat`	`tranp`	9 on page 79 (117/121)
`getup`	Prepares the horizontal meteorological spatial cell arrays `net` and `netsu` for horizontally resolved meteorological data	`datin`	10 on page 79 (118/122)
`layers`	Constructs base and center altitude arrays `zbh` and `zch`, respectively, of meteorological spatial cells for 3D resolved meteorological data	`datin`	11 on page 80 (120/124)

Subroutine	Description	Called by	Page ([3] printed/PDF)

`nest`	Returns the index of the horizontal meteorological spatial cell of the `net` or `netsu` array and its boundary coordinates for a given coordinate	`boun`
		`sprvs`
		`sumdat`
		`tranp`
			12 on page 82
			(122/126)

`onedin`	Processes the input of wind and turbulence data for a horizontally homogeneous data field	`datin`	13 on page 83 (123/127)

`settle`	Returns the terminal gravitational settling speed of a sphere in air using Davies [81] and Beard [82] Eq. 225 and a slip correction based on the atmospheric conditions	`cpfr`
		`rsxp`
		`sprvs`
			14 on page 85
			(54/58)

`sprvs`	Supervises transport of fallout parcels from the stabilized cloud to impact on the ground	`dtmex`	15 on page 86 (125/129)
`sumdat`	Computes the vertically weighted sums of meteorological data	`dtmex`	16 on page 89 (129/133)
`tranp`	Computes the vertically weighted sums of meteorological data	`advec`	17 on page 91 (131/135)
`tridin`	Processes the input of wind and turbulence data for a 3D resolved data field	`datin`	18 on page 94 (135/139)
`wilkns`	Computes the vertical profile of TKE dissipation rate ε using Eq. 97 or Eq. 104, as described in Section 3.1.1	`datin`	19 on page 97 (139/143)

5.1 ADVEC

The advec subroutine, shown in Listing 1, transports both the bottom and top of an individual fallout parcel from its stabilized height to the ground using subroutine tranp. The impact points of the bottom and top of the parcel are combined as described in Section 4.2.2 to form a single deposit increment that geometrically approximates the deposited mass concentration profile of the parcel. The implementation of this subroutine is described in detail in Norment [8] p. 109.

Listing 1. 1979 DELFIC advec subroutine.

```fortran
      subroutine advec(net,netsu,zbh,timup,usum,vsum,dxsum,dysum,rsum,
     1wfz,tsum,cavs,zch,alt,atp,prs,rho,eta,tmax,
     2icf,jcf,ncf,kbhf,ndatf,ltimf,natf)
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
      common /parcl/  cross,down,dwaf,eddy,ndatp,pmas,psiz,rhop,rwaf,
     1 tp,xp,yp,zlow,zp
      common /space/ wint,xllc,yllc,zmax,zmin,timex
      dimension net(icf,jcf),netsu(ncf),zbh(kbhf),usum(kbhf,ndatf,ltimf)
      dimension vsum(kbhf,ndatf,ltimf),dxsum(kbhf,ndatf,ltimf)
      dimension dysum(kbhf,ndatf,ltimf),timup(ltimf),zch(kbhf)
      dimension cavs(kbhf),wfz(kbhf,ndatf,ltimf),tsum(kbhf)
      dimension rsum(kbhf,ndatf,ltimf)
      dimension alt(natf),atp(natf),prs(natf),rho(natf),eta(natf)
      data eps/0.1/
      mc3=mc(3)
      zp=zlow
      if (   (zp-zmin).gt.eps) go to 1411
      tol=tp
      xol=xp
      yol=yp
      zol=zp
      rol=0.
      sigxl=rwaf
      sigyl=rwaf
      go to 1412
 1411 call       tranp(net,netsu,zbh,timup,usum,vsum,dxsum,dysum,rsum,
     1wfz,cavs,tsum,tmax,xol,yol,zol,tol,sigxl,sigyl,rol,ndatl,
     2icf,jcf,ncf,kbhf,ndatf,ltimf)
 1412 zp=zlow+dwaf
      if( zp-zmin .gt.eps) go to 1414
      tou=tp
      xou=xp
      you=yp
      zou=zp
      rou=0.
      sigxu=rwaf
      sigyu=rwaf
      go to 1415
 1414 call       tranp(net,netsu,zbh,timup,usum,vsum,dxsum,dysum,rsum,
     1wfz,cavs,tsum,tmax,xou,you,zou,tou,sigxu,sigyu,rou,ndatu,
     2icf,jcf,ncf,kbhf,ndatf,ltimf)
 1415 zoutn=(zol+zou)/2.
      toutn=(tol+tou)/2.
      if(abs(xou-xol).ge.1.0e-20) go to 1404
      if(abs(you-yol).ge.1.0e-30) go to 1403
      routn=0.
      go to 1405
 1403 routn=1.57079633
      go to 1405
 1404 routn=atan((you-yol)/(xou-xol))
      if(xou-xol .lt. 0.0) routn=routn - sign(3.141592654,routn)
 1405 r=routn-rol
      sxl=1./sqrt((cos(r)/sigxl)**2+(sin(r)/sigyl)**2)
      syl=1./sqrt((sin(r)/sigxl)**2+(cos(r)/sigyl)**2)
      r=routn-rou
      sxu=1./sqrt((cos(r)/sigxu)**2+(sin(r)/sigyu)**2)
      syu=1./sqrt((sin(r)/sigxu)**2+(cos(r)/sigyu)**2)
      sxotn=(sxu+sxl+sqrt((xou-xol)**2+(you-yol)**2))/2.
      syotn=sqrt(syu*syl)
      xoutn=xol+(sxotn-sxl)*cos(routn)
      youtn=yol+(sxotn-sxl)*sin(routn)
 1450 call dumper(xoutn,youtn,zoutn,toutn,sxotn,syotn,pmas,psiz,routn,0,
     1isout,ipout,mc3)
      return
      end
```

5.2 BOUN

The boun subroutine, shown in Listing 2, returns the horizontal coordinates (xc, yc) where the fallout parcel originating at (xo, yo) will enter a meteorological spatial cell if it were to travel to (xt, yt) in a homogeneous wind field. The implementation of this subroutine is described in detail in Norment [8] p. 132.

Listing 2. 1979 DELFIC boun subroutine.

```
      subroutine boun(net,netsu,xt,yt,xo,yo,xc,yc,icf,jcf,ncf)
      dimension net(icf,jcf),netsu(ncf)
      data eps/0.5/
      adisp=0.
      bdisp=0.
      call nest(net,netsu,xo,yo,ndato,xl,xr,yl,yu,icf,jcf,ncf)
      if(xt.le.xr) go to 102
      xc=xr
      adisp=eps
      go to 104
  102 xc=xl
      if(xt.ge.xl) go to 106
      adisp=-eps
  104 yc=yo+(yt-yo)*(xc-xo)/(xt-xo)
      if((yu.ge.yc).and.(yc.ge.yl)) go to 111
  106 if(yt.lt.yu) go to 108
      yc=yu
  107 bdisp=eps
      go to 110
  108 yc=yl
      if(yt.ge.yl) go to 111
      bdisp=-eps
  110 xc=xo+(xt-xo)*(yc-yo)/(yt-yo)
  111 xc=xc+adisp
      yc=yc+bdisp
      return
      end
```

5.3 CALIB

The calib subroutine, shown in Listing 3, returns an index n related to the position of a point an within a data array a of size nx. The index nn at which the point is checked is offset from n by (1+ns)/2, so a value of ns=-1 leads to no offset, while ns=1 leads to an offset of 1.

Listing 3. 1979 DELFIC calib subroutine.

```
      subroutine calib(a,nx,an,ns,n)
      dimension a(nx)
      eps = 1.e-6 * ns * abs( an )
      n=0
    1 n=n+1
      nn=n+(1+ns)/2
      if((nn.lt.nx+1).and.(a(nn).lt.an+eps)) go to 1
      return
      end
```

5.4 CNTR

The cntr subroutine, shown in Listing 4, returns the horizontal coordinates (xg, yg) of the center of a meteorological spatial cell ndata within arrays net or netsu. The implementation of this subroutine is described in Norment [8] p. 133.

Listing 4. 1979 DELFIC cntr subroutine.

```
      subroutine cntr(net,netsu,ndata,xg,yg,icf,jcf,ncf)
      character*6 progrm
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
      common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      common /space/ wint,xllc,yllc,zmax,zmin,timex
      dimension net(icf,jcf),netsu(ncf)
      data progrm/'cntr  '/
      vint=wint/2.
      ig=0
      jg=0
      ndble=1
      nstor=ndata
    1 do 2 jc=1,jcx
      do 2 ic=1,icx
      if(net(ic,jc).eq.nstor) go to 9
   2  continue
      do 3 nc=1,ncx
      if(netsu(nc).eq.nstor) go to 4
   3  continue
      call error(progrm,-3,isout)
    4 ng=nc-4*(nc/4)+1
      ing=+1
      jng=-1
      nstor=-nc+3
      go to (8,7,6,5), ng
    5 ing=ing+2
      nstor=nstor+1
    6 jng=jng+2
      nstor=nstor+1
    7 ing=ing-2
```

```
2724       nstor=nstor-3
2725     8 ig=ig+ing*ndble
2726       jg=jg+jng*ndble
2727       ndble=2*ndble
2728       vint=vint/2.
2729       go to 1
2730     9 ig=ig+ndble
2731       jg=jg+ndble
2732       xg=wint*float(ic-1)+vint*float(ig)+xllc
2733       yg=wint*float(jc-1)+vint*float(jg)+yllc
2734       return
2735       end
```

5.5 DATIN

The `datin` subroutine, shown in Listing 5, reads and processes meteorological data. For horizontally inhomogeneous, 3D-resolved meteorological data, it first

1. `calls` `getup` to setup the horizontally resolved meteorological data field arrays `net`, `netsu`, and `mary` and
2. `calls` `layers` to construct the arrays `zbh` and `zch` of the atmospheric layer base and center altitudes.

Then it `calls` `tridin` for both the wind and turbulence data separately at lines 2806 and 2833, respectively. For horizontally homogeneous data, these subroutines are skipped since only a single horizontal value is required at a given altitude, and subroutine `onedin` is `called` instead. When processing turbulence data, the user has the option of either providing the TKE dissipation rate ε directly or calculating it using using Eq. 97 or Eq. 104, as described in Section 3.1.1.

Listing 5. 1979 DELFIC `datin` subroutine.

```
2738       subroutine datin(net,netsu,zbh,zch,timup,usum,vsum,rsum,wfz,
2739      1dxsum,dysum,cavs,mary,icf,jcf,ncf,marf,kbhf,ndatf,ltimf)
2740       character*6 progrm
2741       character*4 spec,form,wind,turb,done,meteor,resolv,inpu,wilks
2742       common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
2743       common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
2744       dimension net(icf,jcf),netsu(ncf),mary(marf),zbh(kbhf),zch(kbhf)
2745       dimension timup(ltimf),usum(kbhf,ndatf,ltimf)
2746       dimension vsum(kbhf,ndatf,ltimf),wfz(kbhf,ndatf,ltimf)
2747       dimension dxsum(kbhf,ndatf,ltimf),dysum(kbhf,ndatf,ltimf)
2748       dimension dzsum(35,1,6)
2749       dimension rsum(kbhf,ndatf,ltimf)
2750       dimension cavs(35)
2751       data progrm ,alimit ,wind  ,turb  ,done  ,meteor,resolv
2752      1   /'datin ',999999.,'wind','turb','no m','mete','reso'/
2753       data inpu   , wilks
2754      1  /'inpu' ,'wilk'   /
2755     1 format(a4, 2x, a4, 18x, i2, f10.0)
2756    10 format(///15x'atmosphere update',i4,' for times later than' ,
2757      1 e12.5, 'sec (',f8.3,' hours)',/)
2758    11 format(21x,'* * * * * * * * * windfield data * * * * * * * * *',/)
2759    12 format(21x,' * * * * * * * * turbulence data * * * * * * * *',/)
2760    21 format(1h0,10x,'count of updata data sets does not tally with',
2761      1   'specified update sequence numbers')
2762    22 format(1h0,10x, 22h a data set is missing)
```

```fortran
   23 format(///20x,'update index inconsistent with update time on',
     1    ' an input card')
   25 format( 1h0, 81hfirst update winds must be input first when 1-dime
     1nsional data processing is used)
      do 50 l=1,ltimf
      timup(l)=alimit
      do 50 n=1,ndatf
      usum(1,n,l)=alimit
  50  dxsum(1,n,l)=alimit
      zch(1)=alimit
      if(mc(1).eq.0) go to 500
      call       getup(net,netsu,mary,marf,icf,jcf,ncf,ndatf)
      call       layers(zch,zbh,kbhf)
      go to 1000
 500  icx=1
      jcx=1
      ndatx=1
      net(1,1)=1
 1000 ltimx=0
 1002 read(isin,1)spec,form,ltim,uptimh
      if(spec.eq.done) go to 3000
 1003 if(ltim.lt.1.or.ltim.gt.ltimf)call error (progrm,-1003,isout)
      uptims=uptimh*3600.
 1004 if(timup(ltim) .ne. alimit)if(timup(ltim)-uptims)5003,1050,5003
      timup(ltim)=uptims
 1050 if(mc(2) .ne. 1) write(isout,10) ltim,uptims,uptimh
 1051 if(ltim .gt. 1 .or. spec .eq. turb .and. mc(1) .eq. 0)
     1  if(ltimx-1)1052,1053,1053
      go to 1053
 1052 write(isout,25)
      call error(progrm,-1052,isout)
 1053 if(spec.eq.turb) go to 2000
 1055 if(spec.ne.wind) call error (progrm,-1055,isout)
      if(mc(2) .ne. 1) write(isout,11)
      ltimx=ltimx+1
 1060 if(ltimx.gt.ltimf) call error(progrm,-1060,isout)
      if(mc(1) .ne. 0) go to 1100
      call       onedin(zch,zbh,cavs,usum ,vsum ,ltim,kbhf,ndatf,ltimf,
     1 form,spec)
      do 1070 n=1,ndatx
      do 1070 k=1,kbhx
 1070 wfz(k,n,ltim)=0.0
      go to 1200
 1100 call       tridin(net,netsu,zch,usum ,vsum ,wfz,ltim,icf,jcf,ncf,
     1kbhf,ndatf,ltimf,form,spec)
 1200 continue
      do 1300 n=1,ndatx
      do 1300 k=1,kbhx
      if(abs(usum(k,n,ltim)).ge.1.0e-30)go to 1254
      if(abs(vsum(k,n,ltim)).ge.1.0e-30)go to 1253
      rsum(k,n,ltim)=0.0
      go to 1300
 1253 rsum(k,n,ltim)=sign(1.57079633,vsum(k,n,ltim))
      go to 1300
 1254 rsum(k,n,ltim)=atan(vsum(k,n,ltim)/usum(k,n,ltim))
      if(usum(k,n,ltim) .lt. 0.0) rsum(k,n,ltim) = rsum(k,n,ltim) -
     1  sign(3.141592654,rsum(k,n,ltim))
 1300 continue
      go to 1002
```

```
2000 if(mc(2) .ne. 1) write(isout,12)
     if(form .eq. inpu) go to 2100
2001 if(form .ne. wilks ) call error(progrm,-2001,isout)
     call         wilkns (zch,dxsum,dysum,cavs,timup,kbhf,ndatf,ltimf,
    1 ltim)
     go to 1002
2100 form=resolv
     if(mc(1) .ne. 0) go to 2200
     call       onedin(zch,zbh,cavs,dxsum,dysum,ltim,kbhf,ndatf,ltimf,
    1 form,spec)
     go to 1002
2200 call tridin(net,netsu,zch,dxsum,dysum,dzsum,ltim,icf,jcf,ncf,
    1kbhf,ndatf,ltimf,form,spec)
     go to 1002
3000 continue
     ltim=0
     do 3100 l=1,ltimf
     if(timup(l).eq.alimit)go to 3100
     ltim=ltim+1
3100 continue
     if(ltim.eq.ltimx)go to 3200
     write(isout,21)
3105 call error (progrm,-3105,isout)
3200 do 3250 l=1,ltimx
     do 3250 n=1,ndatx
     if(usum(1,n,l).eq.alimit.or.dxsum(1,n,l).eq.alimit)go to 3275
3250 continue
     return
3275 write(isout,22)
3276 call error(progrm,-3276,isout)
5003 write(isout,23)
     call error(progrm, -1004, isout)
     return
     end
```

5.6 DTMEX

Subroutine dtmex is the DTM executor, initializing several DTM arrays and invoking meteorological data processing and transport subroutines, as shown in Listing 6. Line 2868 specifies the number of vertical atmospheric levels supported by the meteorological data as natf=256. Lines 2869–2872 define the input and output file unit numbers based on the array numtap subroutine argument. Lines 2873–2898 loop through several arrays to zero-initialize them:

1. net, netsu, mary: horizontally inhomogeneous meteorological data space control net, sub-net, and control specification arrays, respectively. These are defined in subroutine getup, which is described in detail on pp. 77–81 of Norment [8]. Note that the topography specification support appears to have been removed somewhere between 1971 and 1979. As is evident from lines 2866–2867, the 1979 version by default supports only horizontally homogeneous meteorological data since icf=jcf=marf=ncf=ndatf=1.

2. cavs, tsum, zbh, zch: for each meteorological data vertical level, the particle gravitational settling velocity ($\mathrm{m\,s^{-1}}$), gravitational settling time sum (s), and base and center heights (m), respectively. The cavs array is actually reused and defined differently depending on where the DTM is in its execution:

   (a) during meteorological data reading in onedin, for processed height level (m)

(b) during meteorological data processing in wilkns, for TKE dissipation rate ($\mathrm{m^2\,s^{-3}}$), calculated according to Eqs. 97 or 104

(c) during transport in sprvs and in settle, for particle gravitational settling velocity ($\mathrm{m\,s^{-1}}$), calculated according to the method discussed in Norment [2] Section 2.2.3 and Appendix B.

3. hdav, timup, wavg: for each meteorological data time period (and level in the case of wavg), the volume-averaged TKE dissipation rate ($\mathrm{m^2\,s^{-3}}$), update time point (s), and horizontally averaged vertical wind velocity ($\mathrm{m\,s^{-1}}$), respectively. hdav is discussed somewhat on pp. 84–90 of Norment [8] under the name davg.

4. wfz, usum, vsum, dxsum, dysum, rsum: for each meteorological data level, horizontal point, and time period, the vertical wind velocity ($\mathrm{m\,s^{-1}}$), vertically weighted x and y velocities ($\mathrm{m^2\,s^{-1}}$), vertically weighted x and y TKE dissipation rates ($\mathrm{m^3\,s^{-3}}$), and wind direction angle (radians). Initially, during datin, usum and vsum store the x and y velocities read from the meteorological data and computed in the onedin or tridin subroutines; these are then vertically weighted in the sumdat subroutine. Similarly, dxsum and dysum initially store the TKE dissipation rates from the meteorological data in the onedin or tridin subroutines or computed in the wilkns subroutine and are vertically weighted in sumdat. The wind direction angle is computed initially in datin based on the x and y velocity components and is vertically weighted in sumdat.

Finally, lines 2899–2906 invoke the data processing and transport subroutines before returning on line 2907 to the main program for subsequent execution of the OPM.

Listing 6. 1979 DELFIC dtmex subroutine.

```
2856       subroutine dtmex(numtap)
2857       common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
2858       dimension numtap(15)
2859       dimension alt(256),atp(256),prs(256),rlh(256),rho(256),eta(256)
2860       dimension net(     1,     1),netsu(     1),wavg(    35,     6)
2861       dimension  usum(    35,     1,     6), vsum(    35,     1,     6)
2862       dimension dxsum(    35,     1,     6),dysum(    35,     1,     6)
2863       dimension  rsum(    35,     1,     6),cavs(    35),hdav(     6)
2864       dimension zbh(  35), zch(  35), timup(   6), mary(   1)
2865       dimension   wfz(    35,     1,     6),tsum(    35)
2866       data    icf   ,jcf   ,marf  ,ncf   ,ndatf ,kbhf  ,ltimf
2867      1       /  1   ,  1   ,   1  ,  1   ,    1 , 35  ,    6 /
2868       natf=256
2869       isin =numtap( 1)
2870       isout=numtap( 2)
2871       ipout=numtap(3)
2872       jparn=numtap( 4)
2873       do 1 n=1,ncf
2874     1 netsu(n)=0
2875       do 2 j=1,jcf
2876       do 2 i=1,icf
2877     2 net(i,j)=0
2878       do 3 m=1,marf
2879     3 mary(m)=0
2880       do 4 k=1,kbhf
2881       cavs(k)=0.
2882       tsum(k)=0.
2883       zbh(k)=0.
2884     4 zch(k)=0.
```

```
      do 5 l=1,ltimf
      hdav(l)=0.
      timup(l)=0.
      do 5 k=1,kbhf
    5 wavg(k,l)=0.0
      do 6 l=1,ltimf
      do 6 n=1,ndatf
      do 6 k=1,kbhf
      wfz(k,n,l)=0.
      usum(k,n,l)=0.
      vsum(k,n,l)=0.
      dxsum(k,n,l)=0.
      dysum(k,n,l)=0.
    6 rsum(k,n,l)=0.
      call      dtmint(alt,atp,prs,rlh,rho,eta,natf)
      call      datin(net,netsu,zbh,zch,timup,usum,vsum,rsum,wfz,
     1dxsum,dysum,cavs,mary,icf,jcf,ncf,marf,kbhf,ndatf,ltimf)
      call      sumdat(net,netsu,zbh,zch,wavg,hdav,usum,vsum,rsum,wfz,
     1timup,dxsum,dysum,icf,jcf,ncf,kbhf,ndatf,ltimf)
      call      sprvs(net,netsu,zbh,zch,timup,usum,vsum,dxsum,dysum,
     1rsum,wfz,cavs,hdav,tsum,wavg,alt,atp,prs,rlh,rho,eta,
     2icf,jcf,ncf,kbhf,ndatf,ltimf,natf)
      return
      end
```

5.7 DTMINT

As shown in Listing 7, subroutine `dtmint` initializes the DTM by the following:

1. Reading DTM cards* 1–4 from the input deck unit `isin=numtap(1)=13` on lines 48–50 and 54.
2. Reading cloud rise, parcel, and atmospheric data written by the ICRM from unit `jparn=numtap(4)=9`. For the purposes of initializing the DTM, this sets the particle density `rhop` (i.e., ρ_p) and the deposition plane altitude `zmin` (i.e., z_dep).
3. Writing the read cloud rise and parcel data to unit `ipout=numtap(3)=11` for later use by the DTM.
4. Writing some DTM configuration information to the printed output unit `isout=numtap(2)=6`.

Listing 7. 1979 DELFIC `dtmint` subroutine.

```
2        subroutine dtmint(alt,atp,prs,rlh,rho,eta,natf)
3        character*6 progrm
4        character*6 detid(12),dtmid(12)
5        common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
6        common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
7        common /parcl/  cross,down,dwaf,eddy,ndatp,pmas,psiz,rhop,rwaf,
8       1 tp,xp,yp,zlow,zp
9        common /space/ wint,xllc,yllc,zmax,zmin,timex
10       dimension alt(natf),atp(natf),prs(natf),rlh(natf),rho(natf)
11       dimension eta(natf),ps(200),diam(200),fmass(200)
12       data progrm/'dtmint'/
13     1 format(12a6)
14     2 format(10x,'initialization and cloud rise module identification',
15      1  12a6,/,10x,'diffusive transport module identification -',
16      2 12a6)
```

[*] See Norment [3] Section 3.2 for details.

```
    7 format(/,15x,'the control variable array, mc(j), has been given',
     1     ' the values -')
    8 format(15x,20i4)
    9 format(/28x28hthe transport time limit is f12.3, 7h sec. (f10.5,
     1 7h hours))
   10 format( 15x,'a plane deposition surface at altitude',f9.3,
     1   '(meters above msl) is assumed')
   14 format(15x,'coordinates of ground zero (xgz,ygz,zgz) are (',
     1 3e13.5,')(meters)',/,42x,'detonation time is',e12.5,' seconds',/)
   21 format(20i4)
   23 format(1h1,///,51x,'* * * * * * * * * *',//,55x,'d e l f i c',//,
     1   12x,'t h e   d e p a r t m e n t   o f   d e f e n s e ',
     2 '    f a l l o u t   p r e d i c t i o n   s y s t e m',
     3  //,51x,'* * * * * * * * * *',////,48x,'diffusive transport',
     4  ' module',///,55x,'prepared by',/,46x,
     5  'atmospheric science associates',/,54x,'bedford, mass.',
     6//// ,41x,'*****  summary of run identifiers  *****')
   27 format(15x,'horizontal coordinates of the south west corner of',
     1 ' the transport space are (',2e13.5,')',/,35x,
     2 'the resolution net spacing is ',e12.5,' (all in meters)')
   40 format(8f10.0)
   43 format(/15x,28hfallout particle density is e12.5,8h kg/m**3,
     1 12h   there arei5, 22h particle size classes)
   45 format(/ 15x, 36hparticle processing begins with the i6, 12h th pa
     1rticle)
   46 format(1h1)
   47 format(/ 15x, 20htransport is by the )
   48 format(1h+ 34x, 12hquick method)
   49 format(1h+, 34x, 21hlayer-by-layer method)
   50 format(28x,57hratic of lagrangian to eulerian turbulence time scal
     1es isf8.3)
      read(isin,1)dtmid
      read(isin,21)mc
      read(isin,21)icx,jcx,nseqo
      if(icx .eq. 0) icx=1
      if(jcx .eq. 0) jcx=1
      if(nseqo .eq. 0)nseqo=1
      read(isin,40) wint,xllc,yllc,timeh,eddy
      if(eddy .eq. 0.0) eddy=4.0
      rewind jparn
      rewind ipout
      read(jparn)fw,ssam,sldtmp,tmsd,sd,w,height,rhop,radmax,zmin
      read(jparn)xgz,ygz,tgz
      read(jparn)(detid(i),i=1,12)
      read(jparn)ndstr
      read(jparn)(ps(j),diam(j),fmass(j),j=1,ndstr)
      read(jparn)nat
      read(jparn)(alt(j),atp(j),prs(j),rlh(j),rho(j),eta(j),j=1,nat)
      write(ipout)fw,ssam,sldtmp,tmsd,sd,w,height,rhop,radmax,zmin
      write(ipout)xgz,ygz,tgz
      write(ipout) (detid(j),j=1,12)
      write(ipout) (dtmid(j),j=1,12)
      write(ipout)ndstr
      write(ipout)(ps(j),diam(j),fmass(j),j=1,ndstr)
      write(isout,23)
      write(isout,2) (detid(j),j=1,12),(dtmid(j),j=1,12)
      write(isout,7)
      write(isout,8)mc
      timex=timeh*3600.
      write(isout,9) timex, timeh
      if(mc(6) .gt. 0) write(isout,50) eddy
      write(isout,14)xgz,ygz,zmin,tgz
      write(isout,27)xllc,yllc,wint
      write(isout,10)zmin
      write(isout,43)rhop,ndstr
      write(isout,47)
      if(mc(4) .eq. 0) write(isout,48)
      if(mc(4) .ne. 0) write(isout,49)
      if(nseqo .ne. 1) write(isout,45) nseqo
      if(mc(2) .ne. 1) write(isout,46)
      return
      end
```

5.8 DUMPER

The dumper subroutine, shown in Listing 8, writes deposit increments to unit ipout as well as isout if requested by setting mc(3)>1.

Listing 8. 1979 DELFIC dumper subroutine.

```
      subroutine dumper(xo,yo,zo,to,sigxo,sigyo,pmas,psiz,ro,incomp,
     1isout,ipout,mc3)
      dimension xout(100),yout(100),zout(100),tout(100),rout(100)
      dimension syot(100),sxot(100),psot(100),pdep(100)
      save xout,yout,zout,tout,rout,syot,sxot,psot,pdep
      data n/0/, nblk/100/
  807 format(5x,9e12.4)
  817 format( 1h0, 23x,'block of',i5,' transported parcel properties',
     1 ' written on ipout tape',/,12x,'xc',10x,'yo',10x,'zo',10x,
     2'to',8x,'sigxo',7x,'sigyo',9x,'rc',9x,'psiz',8x,'pmas',/)
 8023 format(1ho,14x,'resume pre-transport parcel property list for',
     1 ' particle size',e12.5,' meters')
 8024 format( 2x,'nseq',6x,'xp',10x,'yp',10x,'zp',10x,'tp',
     1 9x,'pmas',8x,'rwaf',7x,'zlow',8x,'dwaf',/)
      if(incomp.gt.0) go to 8063
      n = n + 1
      xout(n)=xo
      yout(n)=yo
      zout(n)=zo
      tout(n)=to
      sxot(n)=sigxo
      syot(n)=sigyo
      psot(n)=psiz
      pdep(n)=pmas
      rout(n)=ro
      if(n.lt.nblk) return
 8063 write(ipout) n
      if( mc3 .gt.1) write(isout,817) n
      if(n.eq.0) return
      write(ipout)     (xout(m),yout(m),zout(m),tout(m),sxot(m),syot(m),
     1rout(m),psot(m),pdep(m),m=1,n)
      if(mc3 .le. 1) go to 8064
      write(isout,807) (xout(m),yout(m),zout(m),tout(m),sxot(m),syot(m),
     1rout(m),psot(m),pdep(m),m=1,n)
      write(isout,8023) psiz
      write(isout,8024)
 8064 n=0
      return
      end
```

5.9 GETDA

The getda subroutine, shown in Listing 9, computes the vertically averaged value abar of an array asum, which is the vertically weighted sum of data between vertical levels kbha and kbhb. The vertically weighted sum of data is computed by subroutine sumdat. This is further described in Norment [8] p. 133 and Section 3.1.2 Eq. 105 of this work.

Listing 9. 1979 DELFIC getda subroutine.

```
      subroutine getda(asum,zbh,kbha,kbhb,ndata,ltim, abar ,kbhf,ndatf,
     1ltimf)
      character*6 progrm
      common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
      dimension asum(kbhf,ndatf,ltimf),zbh(kbhf)
      data progrm/'getda '/
      if(kbha-kbhx-1) 3,2,1
    1 i=1
   10 call error(progrm,i,isout)
      abar=0.
      return
    2 abar=asum(kbha-1,ndata,ltim)
      return
    3 abar=asum(kbha-1,ndata,ltim)
      if(kbhb-1) 6,5,4
    6 i=6
      go to 10
    4 abar=abar-asum(kbhb-1,ndata,ltim)
    5 abar=abar/(zbh(kbha)-zbh(kbhb))
      return
      end
```

5.10 GETUP

The getup subroutine, shown in Listing 10, prepares the horizontal meteorological spatial cell arrays net and netsu for horizontally resolved meteorological data. The implementation of this subroutine is described in detail in Norment [8] p. 77.

Listing 10. 1979 DELFIC getup subroutine.

```
      subroutine getup(net,netsu,mary,marf,icf,jcf,ncf,ndatf)
      character*6 progrm
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
      common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      dimension       net(icf,jcf),     netsu(   ncf),      mary(  marf)
      data progrm/'getup ' /
 1000 format( 36i2 )
 1001 format( 1h0, 25x, 31harray mary has been loaded with,i5, 22h eleme
     1nt(s) as follows/)
 1002 format(25x,36i2)
      msect = 4
      mneg = 1 - msect
      ndata = 0
      ic = 0
      jc = 1
      if(icx.ge.icf) call error(progrm,-1,isout)
      if(jcx.gt.jcf) call error(progrm,-1,isout)
      marx = icx * jcx
 1    do 2 mark = 1, marf
 2    mary( mark ) = -9
      if(marx.gt.marf) call error(progrm,-2,isout)
      read(isin,1000) (mary(mark),mark=1,marx)
      write(isout,1001) marx
      write(isout,1002) (mary(mark),mark=1,marx)
      mark = 0
      mctr = 0
 3    mark = mark + 1
      if( mark - marx ) 5, 5, 4
 4    marx = msect * mctr
      if( marx )  6, 14, 1
 5    if( mary( mark ) )  6, 7, 8
 6    call error(progrm,-6,isout)
 7    mneg = mneg + msect
      nqq = - mneg
      mctr = mctr + 1
      go to 9
 8    ndata = ndata + 1
      nqq = ndata
      ndatx = ndata
      if(ndatx.gt.ndatf) call error(progrm,-8,isout)
 9    ic = ic + 1
      if( jc - jcx ) 10, 10, 13
 10   if( ic - icx ) 12, 12, 11
 11   jc = jc + 1
      ic = 0
      go to 9
 12   net( ic, jc ) = nqq
      go to 3
 13   nc = ic
      netsu( nc ) = nqq
      ncx = nc
      if(ncx.gt.ncf) call error(progrm,-13,isout)
      go to 3
 14   return
      end
```

5.11 LAYERS

The layers subroutine, shown in Listing 11, constructs base and center altitude arrays zbh and zch, respectively, of meteorological spatial cells for 3D-resolved meteorological data.

Listing 11. 1979 DELFIC layers subroutine.

```
      subroutine layers(zch,zbh,kbhf)
      character*6 progrm
      character*4 tlayr,basalt,cntalt
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
       common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      common /space/ wint,xllc,yllc,zmax,zmin,timex
      dimension zbh(kbhf),zch(kbhf), entry(8)
      data progrm   ,basalt   ,cntalt   ,epsz     ,alimit   ,irec
     1    /'layers','base','cent', 0.1  , 999999., 8        /
    1 format( 19x,  6hlevels,i4, 5h thru,i4/25x, 8f12.5)
    2 format(8f10.0)
    3 format(1h0, 48x, 25hwind layer base altitudes/)
    4 format(1h0, 48x, 27hwind layer center altitudes/)
    5 format( 11xa4)
   6  format(1h0,25x,31hmaximum wind space altitude is e12.5,7h meters)
   8  format(1h0,10x, 45hzbh(1)and zmin do not agree within tolerance ,e
     112.5)
      read(isin,5)tlayr
      k=0
  200 read(isin,2)(entry(i),i=1,irec)
      do 201 i=1,irec
      if(entry(i).ge.alimit) go to 202
      if(entry(i) .lt. 0.0) go to 201
      k=k+1
      if(k.gt.kbhf) call error(progrm,-201,isout)
      zch(k)=entry(i)
  201 continue
      go to 200
  202 kbhx=k
      kbhm1=kbhx-1
      do 210 i=1,kbhm1
      ip1=i+1
      do 210 j=ip1,kbhx
      if(zch(i).le.zch(j))go to 210
      temp=zch(i)
      zch(i)=zch(j)
      zch(j)=temp
 210  continue
      if(tlayr.eq.cntalt)go to 250
 230  if(tlayr.ne.basalt)call error(progrm,-230,isout)
      if(abs(zch(1)-zmin).le.epsz)go to 235
      write(isout,8)epsz
  234 call error(progrm,-234,isout)
  235 zch(1)=zmin
      do 240 k=1,kbhm1
      zbh(k)=zch(k)
  240 zch(k)= (zch(k) + zch(k+1))/2.0
      zbh(kbhx)=zch(kbhx)
      zch(kbhx)= 2.0*zbh(kbhx) - zch(kbhx-1)
      go to 300
 250  zbh(1)=zmin
      do 260 i=2,kbhx
 260  zbh(i)=2.0*zch(i-1) - zbh(i-1)
 300  zmax=2.0*zch(kbhx)-zbh(kbhx)
      write(isout,6)zmax
      write(isout,3)
      do 301 igo=1,kbhx,irec
      istop=igo+irec-1
      if(istop.ge.kbhx) istop=kbhx
  301 write(isout,1)igo,istop,(zbh(k),k=igo,istop)
      write(isout,4)
      do 303 igo=1,kbhx,irec
      istop=igo+irec-1
      if(istop.gt.kbhx) istop=kbhx
  303 write(isout,1)igo,istop,(zch(k),k=igo,istop)
      return
      end
```

5.12 NEST

The `nest` subroutine, shown in Listing 12, returns the index `ndata` of the horizontal meteorological spatial cell of the `net` or `netsu` array in which the point (`xq`, `yq`) is located and its boundary coordinates (`xl`, `yl`) and (`xr`, `yu`). The implementation of this subroutine is described in detail in Norment [8] p. 135.

Listing 12. 1979 DELFIC `nest` subroutine.

```
      subroutine nest(net,netsu,xq,yq,ndata,xl,xr,yl,yu,icf,jcf,ncf)
      character*6 progrm
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
       common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      common /space/ wint,xllc,yllc,zmax,zmin,timex
      dimension net(icf,jcf),netsu(ncf)
      data progrm/'nest  '/
      ic=(xq-xllc)/wint+1.
      jc=(yq-yllc)/wint+1.
c     write(isout,9000)ic,jc,icx,jcx,xq,yq,xllc,yllc,wint
 9000 format(' nest1:',4i5,1p5e10.3)
      if((ic.ge.1).and.(jc.ge.1).and.(ic.le.icx).and.(jc.le.jcx))go to 1
      ndata=-999
      return
    1 vint=wint
      xl=vint*float(ic-1)+xllc
      xr=vint*float(ic)+xllc
      yl=vint*float(jc-1)+yllc
      yu=vint*float(jc)+yllc
c     write(isout,9001)ic,jc,net(ic,jc),vint,xl,xr,yl,yu
 9001 format(' nest2:',3i5,1p5e10.3)
      if(net(ic,jc)) 4,2,3
    2 call error(progrm,-2,isout)
    3 ndata=net(ic,jc)
c     write(isout,9002)ndata
 9002 format(' nest3:',i5)
      return
    4 nq=-net(ic,jc)
    5 vint=vint/2.
      iq=(xq-xl)/vint
      jq=(yq-yl)/vint
      nq=nq+3*iq+jq-2*iq*jq
      xr=xl+vint*float(iq+1)
      xl=xl+vint*float(iq)
      yu=yl+vint*float(jq+1)
      yl=yl+vint*float(jq)
c     write(isout,9003)iq,jq,nq,netsu(nq),xl,xr,yl,yu,vint
 9003 format(' nest4:',4i5,1p5e10.3)
      if(netsu(nq)) 7,6,8
    6 call error(progrm,-6,isout)
    7 nq=-netsu(nq)
      go to 5
    8 ndata=netsu(nq)
      return
      end
```

5.13 ONEDIN

The onedin subroutine, shown in Listing 13, processes the input of wind and turbulence data for a horizontally homogeneous data field.

Listing 13. 1979 DELFIC onedin subroutine.

```
      subroutine onedin(zch,zbh,cavs,dx   ,dy   ,ltim,kbhf,ndatf,ltimf,
     1 form,spec)
      character*6 progrm,fmt(12)
      character*4 meteor,resolv,wind,turb,spec,form
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
       common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      common /space/ wint,xllc,yllc,zmax,zmin,timex
      dimension zch(kbhf),zbh(kbhf),dx(kbhf,ndatf,ltimf),cavs(kbhf)
      dimension scale(5), ap(3), dy(kbhf, ndatf, ltimf)
      data alimit  , radc      , progrm  , meteor ,resolv, wind , turb
     1  /  999999.,.0174532925,'onedin','mete','reso','wind','turb'/
      data irec/8/
    1 format(4x,'levels',i4,' thru',i4, 8f12.5)
    3 format( ////33x, 25hwind layer base altitudes/)
    4 format(1h0,3x,'maximum wind space altitude is',e12.5,' meters')
 1000 format (12a6)
 1100 format (8f10.0)
 1200 format (20i4)
 1300 format(      16x, 13hraw wind data,33x,19hprocessed wind data//8x,
     11hz, 9x, 10hvx or dir., 3x, 11hvy or speed, 14x, 1hz, 12x,
     2 2hvx, 12x, 2hvy/)
 1400 format (3(2x, 1pe12.5))
 1500 format (1h+ 47x, 3(2x, 1pe12.5))
 1600 format(  10x19hraw turbulence data,28x25hprocessed turbulence data
     1//8x, 1hz, 10x, 4hepsx, 10x, 4hepsy, 17x, 1hz, 11x 4hepsx, 10x,
     2 4hepsy/)
 1700  format( 1h0, 5x, 63hnumber of wind or turbulence input data incon
     1sistent for updatei4)
 1800 format( 1h0, 5x, 59hwind or turbulence strata altitudes inconsiste
     1nt for updatei4)
      if(spec .eq. wind  .and.  form .eq. meteor) go to 25
   20 if(spec .eq. wind  .and.  form .ne. resolv)call error(progrm,-20,
     1  isout)
   25 read(isin, 1000)fmt
      read(isin, 1100) scale
      read(isin, 1200) n1, n2, n3
      do 50 i = 1,3
  50     if(scale(i).eq. 0.0) scale (i) = 1.0
      if(form .eq. meteor) trns=scale(5)*scale(3) - 180.
      if(mc(2) .ne. 1  .and.  spec .eq. wind) write(isout,1300)
      if(mc(2) .ne. 1  .and.  spec .eq. turb) write(isout,1600)
      kbh = 0
 100  read(isin, fmt) ap
      if(ap(n1).ge.alimit)go to 250
      if(mc( 2 )  .ne. 1   ) write(isout, 1400)ap(n1), ap(n2), ap(n3)
      kbh = kbh + 1
      cavs(kbh)= (ap(n1) + scale(4))*scale(1)
      if(form.eq.resolv) go to 150
      dx(kbh,1,ltim)=ap(n3)*scale(2)*sin(radc*(ap(n2)*scale(3) + trns))
      dy(kbh,1,ltim)=ap(n3)*scale(2)*cos(radc*(ap(n2)*scale(3) + trns))
      go to 200
 150  dx(kbh,1,ltim) = ap(n2)*scale(2)
      dy(kbh,1,ltim) = ap(n3)*scale(2)
  200 if(mc( 2).ne. 1)write(isout, 1500)cavs(kbh), dx(kbh, 1, ltim),
     1dy(kbh, 1, ltim)
      go to 100
  250 if(ltim .eq. 1) kbhx=kbh
  251 if(ltim .eq. 1  .or. kbh .eq. kbhx) go to 253
      write(isout,1700)ltim
  253 kbhm1 = kbhx - 1
      do 255 i=1,kbhm1
      ip1=i+1
      do 255 j=ip1,kbhx
      if(cavs(i) .le. cavs(j)) go to 255
      temp=cavs(i)
      cavs(i)=cavs(j)
      cavs(j)=temp
      temp=dx(i,1,ltim)
      dx(i,1,ltim)=dx(j,1,ltim)
      dx(j,1,ltim)=temp
      temp=dy(i,1,ltim)
      dy(i,1,ltim)=dy(j,1,ltim)
      dy(j,1,ltim)=temp
  255 continue
      if(ltim .eq. 1 .and. spec .eq. wind) go to 259
      do 258 i=1,kbhm1
      if(cavs(i) .ge. zbh(i) .and. cavs(i) .le. zbh(i+1)) go to 258
      write(isout,1800)ltim
      call error(progrm,-258,isout)
  258 continue
      return
  259 zbh(1) = zmin
      zch(1)=cavs(1)
      do 260 i=2,kbhx
      zch(i)=cavs(i)
 260  zbh(i)=2.0*zch(i-1) - zbh(i-1)
      zmax=2.0*zch(kbhx)-zbh(kbhx)
      write(isout,3)
      do 270 igo=1,kbhx,irec
      istop=igo+irec-1
      if(istop.gt.kbhx) istop=kbhx
  270 write(isout,1)igo,istop,(zbh(k),k=igo,istop)
      write(isout,4)zmax
      return
      end
```

5.14 SETTLE

The `settle` subroutine, shown in Listing 14, calculates the terminal gravitational settling velocity of spherical fallout particles in air using Davies [81] and Beard [82] Eq. 225 with a slip correction based on the atmospheric conditions:

$$C_\mathrm{C} = 1 + 54.088\,\mathrm{m/s/K}^{1/2}\left(\frac{\mu T^{1/2}}{p d_\mathrm{p}}\right). \tag{190}$$

Using Eqs. 211 and 212, to first order in $\mathrm{Kn} \to 0$, the slip correction can be written as

$$C_\mathrm{C} = 1 + 2A_1\mathrm{Kn} = 1 + \frac{2A_1\lambda}{d_\mathrm{p}}. \tag{191}$$

Comparing Eqs. 190 and 191 shows that DELFIC estimates the mean free path of air as

$$\lambda = 54.088\,\mathrm{m/s/K^{1/2}}\,\frac{\mu T^{1/2}}{2A_1 p}. \tag{192}$$

By comparison, the mean free path of air is approximately [83]

$$\begin{aligned}\lambda &= \sqrt{\frac{\pi}{8}}\left(\frac{\mu}{u}\right)(\rho p)^{-1/2} \\ &= \sqrt{\frac{\pi}{8}}\left(\frac{\mu}{u}\right)\frac{R_{\mathrm{air}}^{1/2} T^{1/2}}{p},\end{aligned} \tag{193}$$

where $u = 0.4987445$, and we have used the dry air ideal gas law $p = \rho R_{\mathrm{air}} T$ with dry air specific gas constant $R_{\mathrm{air}} = R/M_{\mathrm{air}} = 287.06\,\mathrm{J\,kg^{-1}\,K^{-1}}$. Thus, using the factor $A_1 \approx 1.257$ from Davies [81], the coefficient in Eqs. 190 and 192 is approximately

$$\frac{2A_1 R_{\mathrm{air}}^{1/2}}{u}\sqrt{\frac{\pi}{8}} \approx 53.518\,\mathrm{m/s/K^{1/2}}.$$

Listing 14. 1979 DELFIC `settle` subroutine.

```
      subroutine settle(d,rhop,rho,eta,t,p,v,i)
c
c     h. g. norment, atmospheric science associates - december 1978
c
c     computes still-air settling speed of rigid spheres according to
c     the equations of beard (jas33,852(1976)) for small spheres
c     (corr .le. 84.175), and davies(proc.phys.soc.(london)57,256(1945)
c     for larger spheres.
c
c     glossary (si units)
c     c       4.0*g/3.0    where g is acceleration of gravity (9.8)
c     corr    davies number
c     d       sphere diameter
c     eta     viscosity
c     p       pressure
c     rho     fluid density
c     rhop    sphere density
c     t       temperature
c     v       settling speed
c     i       accuracy indicator
c             i = 0 result is accurate
c             i = 1 result is inaccurate, davies number is too large
c
      data c/13.066667/
c
      i = 0
compute davies number
      cdrr = c*(rhop-rho)*rho*d**3/eta**2
check davies number value for routing
      if(cdrr.gt.  0.3261) if(cdrr-84.175) 100,100,200
compute via stokes-law equation
      v = cdrr*eta/(24.0*rho*d)
      go to 500
compute via beards equation
```

```
 100  y = alog(cdrr)
      v = eta/(rho*d)*exp(-3.18657 + y*(0.992696 + y*(-0.153193e-2
     1+y*(-0.987059e-3 + y*(-0.578878e-3 + y*(0.855176e-4
     2-y*0.327815e-5))))))
      go to 500
compute via davies equations
 200  if(cdrr.gt.140.)if(cdrr-4.5e7)400,400,300
      v = eta/(rho*d)*cdrr*(4.16666667e-2 + cdrr*(-2.3363e-4
     1+cdrr*(2.0154e-6-cdrr*6.9105e-9)))
      go to 500
 300  i = 1
 400  y = alog10(cdrr)
      v = eta/(rho*d)*(10.0)**(-1.29536 + y*(0.986 + y*(-0.046677+
     1y*1.1235e-3)))
      return
correct settling speed for slip
 500  v = v*(1.0 + 54.088*eta*sqrt(t)/p/d)
      return
      end
```

5.15 SPRVS

The sprvs subroutine, shown in Listing 15, *supervises* the transport of each individual parcel created by the ICRM until their deposition on the ground. Each block of parcels is read from unit jparn in lines 490–565. For each parcel in the block of a given particle size d_p, the terminal velocity v_t is estimated at line 522 as a function of meteorological layer altitude $z_{\mathrm{met},j}$. When combined with wavg from subroutine sumdat, the maximum time tmax and altitude zlim for the parcel that can deposit at the ground is estimated. If the initial altitude of the bottom of the parcel zlow is less than the maximum deposit altitude zlim, the advec subroutine is called for the parcel at lines 558–563 before looping to the next parcel within the block at line 564. Furthermore, if the average vertical wind velocity wavg is nonzero, the layer-by-layer transport mode is enabled for subroutine tranp.

Listing 15. 1979 DELFIC sprvs subroutine.

```
428       subroutine sprvs(net,netsu,zbh,zch,timup,usum,vsum,dxsum,dysum,
429      1rsum,wfz,cavs,hdav,tsum,wavg,alt,atp,prs,rlh,rho,eta,
430      2icf,jcf,ncf,kbhf,ndatf,ltimf,natf)
431       character*6 progrm
432       common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
433       common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
434       common /parcl/  cross,down,dwaf,eddy,ndatp,pmas,psiz,rhop,rwaf,
435      1 tp,xp,yp,zlow,zp
436       common /space/ wint,xllc,yllc,zmax,zmin,timex
437       dimension alt(natf),atp(natf),prs(natf),rho(natf),eta(natf)
438       dimension net(icf,jcf),netsu(ncf),zbh(kbhf),zch(kbhf),hdav(ltimf)
439       dimension usum(kbhf,ndatf,ltimf),vsum(kbhf,ndatf,ltimf),cavs(kbhf)
440       dimension dxsum(kbhf,ndatf,ltimf),dysum(kbhf,ndatf,ltimf)
441       dimension wfz(kbhf,ndatf,ltimf),wavg(kbhf,ltimf),timup(ltimf)
442       dimension rsum(kbhf,ndatf,ltimf),tsum(kbhf)
443       dimension xpar(100),ypar(100),zpar(100),tpar(100),pdam(100)
444       dimension psam(100),rwfr(100),dwfr(100),zlwf(100),vwfr(100)
445       dimension rlh(256)
446       data progrm/'sprvs ' / ,jf/100/
447  8015 format( 1h+, 102x, 8hairborne)
448  8016 format( 1h , i4, 8e12.4)
```

```
8017 format(1h0,5x, 86h* * * * * short-cut transport is cancelled becau
    1se vertical wind is non-zero * * * * *//)
8019 format(1h+,102x,' impacted')
8020 format( 1h+, 102x, 17houtside windspace)
8021 format( 1h0, 36x, 21hparticle diameter is e12.5, 7h meters)
8022 format( 23x,   9hfall ratee12.5,23h meters/sec at altitudee12.5,
    1  7h meters/ 23x, 37hupper limit altitude for impaction ise12.5,
    2  7h meters)
8024 format( 2x,  4hnseq, 6x, 2hxp, 10x, 2hyp,  10x, 2hzp, 10x, 2htp,
    1 9x, 4hpmas, 8x, 4hrwaf, 8x, 4hzlow, 8x, 4hdwaf/)
8025 format( 1h1, 38x, 31hpre-transport parcel properties/)
     hav=0.
     do 40 l=1,ltimx
  40 hav=hav + hdav(l)
     hav=hav/ltimx
     sigw=eddy
     if(mc(6) .gt. 0) sigw=5.39*(hav)**(1.0/3.0)
     mc3=mc(3)
     if(mc3 .gt. 0) write(isout,8025)
     nseq=0
     pszbe=-2.0
     if(mc( 4 ).ne.0) go to 47
     do 45 l=1,ltimx
     do 45 k=1,kbhx
     if(wavg(k,l).ne.0.0) go to 46
 45  continue
     go to 47
 46  mc( 4 )=1
     write(isout,8017)
 47  continue
     kbhm1=kbhx-1
     wavgk=0.0
     do 51 l=1,ltimx
     do 50 k=1,kbhm1
  50 wavgk=wavgk + wavg(k,l)*(zbh(k+1) - zbh(k))
51   wavgk=wavgk + wavg(kbhx,l)*(zmax - zbh(kbhx))
     wavgk = wavgk/(ltimx*(zmax-zmin))
     if(ndatx-1)70,70,60
  60 slop=1.1
     go to 100
  70 slop=1.0
100  read(jparn)np
     if(np.le.0) go to 806
     if(np.gt.jf) call error(progrm,-100,isout)
     read(jparn)  (xpar(j),ypar(j),zpar(j),tpar(j),pdam(j),psam(j),
    1rwfr(j),dwfr(j),zlwf(j),vwfr(j),j=1,np)
     do 1000 j=1,np
     nseq=nseq+1
     if(nseq.lt.nseqo) go to 1000
     xp=xpar(j)
     yp=ypar(j)
     zp=zpar(j)
     tp=tpar(j)
     psiz=pdam(j)
     pmas=psam(j)
     rwaf=rwfr(j)/2.
     dwaf=dwfr(j)
     zlow=zlwf(j)
     vwaf=vwfr(j)
```

```
      if(abs((psiz-pszbe)/psiz).le.1.0e-10) go to 103
      if( mc3 .gt. 0) write(isout,8021) psiz
      h=(zmin+zmax)/2.
      call trpl(h,nat,alt,atp,t)
      call trpl(h,nat,alt,prs,p)
      call trpl(h,nat,alt,rho,den)
      call trpl(h,nat,alt,eta,vis)
      call settle(psiz,rhop,den,vis,t,p,fav,iaccr)
      fav=fav-wavgk
      do 101 kkz=1,kbhx
      call trpl(zch(kkz),nat,alt,atp,t)
      call trpl(zch(kkz),nat,alt,prs,p)
      call trpl(zch(kkz),nat,alt,rho,den)
      call trpl(zch(kkz),nat,alt,eta,vis)
  101 call settle(psiz,rhop,den,vis,t,p,cavs(kkz),iaccr)
      tmax=tp
      do 1001 iz=1,kbhm1
      tmax=tmax + (zbh(iz+1) - zbh(iz))/(cavs(iz) - wavgk)
      if(tmax.gt.slop*timex  .or.  tmax .lt. 0.0 ) go to 1002
 1001 continue
      tmax=tmax + (zmax - zbh(kbhx))/(cavs(kbhx) - wavgk)
      zlim=5.0e4
      if(tmax.gt.slop*timex  .or.  tmax .lt. 0.0 ) zlim=zmax
      go to 1012
 1002 zlim=zbh(iz+1)
 1012 if(tmax .lt. 0.0) tmax=timex
 1003 if( mc3 .lt. 1) go to 1004
      write(isout,8022) fav,h,zlim
      write(isout,8024)
 1004 continue
      if(mc( 4 ).ne.0) go to 1255
      tsum(1)=0.0
      do 1250 k=2,kbhx
 1250 tsum(k)=tsum(k-1)+(zbh(k) - zbh(k-1))/cavs(k-1)
 1255 continue
      down=(fav*eddy/sigw)**2
      cross=1./sqrt(1.+4.*down)
      down=1./sqrt(1.+down)
      pszbe=psiz
  103 if( mc3 .gt. 0)
     1write(isout,8016) nseq,xp,yp,zp,tp,pmas,rwaf,zlow,dwaf
      if (ifix(dwaf).gt.0) go to 1200
      if( mc3 .gt. 0) write(isout,8019)
      call dumper(xp,yp,zp,tp, rwaf, rwaf, pmas,psiz,0.,0,
     1isout,ipout,mc3)
      go to 1000
 1200 call nest(net,netsu,xp,yp,ndatp,xl,xr,yl,yu,icf,jcf,ncf)
      if(ndatp.gt.0) go to 1260
      if( mc3 .gt. 0) write(isout,8020)
      go to 1000
 1260 if(zlow.lt.zlim) go to 1409
      if( mc3 .gt. 0) write(isout,8015)
      go to 1000
 1409 call       advec(net,netsu,zbh,timup,usum,vsum,dxsum,dysum,rsum,
     1wfz,tsum,cavs,zch,alt,atp,prs,rho,eta,tmax,
     2icf,jcf,ncf,kbhf,ndatf,ltimf,natf)
 1000 continue
      go to 100
  806 call       dumper(0.,0.,0.,0.,   0.,   0.,   0.,  0.,0.,999,
567       1isout,ipout,mc3)
568        call      dumper(0.,0.,0.,0.,   0.,   0.,   0.,  0.,0.,999,
569       1isout,ipout,mc3)
570        rewind jparn
571        endfile ipout
572        rewind ipout
573        return
574        end
```

5.16 SUMDAT

The sumdat subroutine, shown in Listing 16, calculates the vertically weighted sums of the meteorological data fields according to Section 3.1.2 and as mentioned in Section 5.6. In particular, the variables usum, vsum, dxsum, dysum, and rsum store for each meteorological data level, horizontal point, and time period the vertical wind velocity ($\mathrm{m\,s^{-1}}$), vertically weighted x and y velocities ($\mathrm{m^2\,s^{-1}}$), vertically weighted x and y TKE dissipation rates ($\mathrm{m^3\,s^{-3}}$), and wind direction angle (radians), respectively. These quantities are calculated in lines 629–644, similar to Eq. 105 except that they are not then averaged over the vertical span $z_{\mathrm{met},i'} - z_{\mathrm{met},i}$. This average is calculated later in subroutine tranp using subroutine getda. It also calculates a horizontal meteorological spatial cell area-weighted average of the vertical velocity wavg ($\mathrm{m\,s^{-1}}$) for each meteorological data level and time period in lines 656–663.

Listing 16. 1979 DELFIC sumdat subroutine.

```
575        subroutine sumdat(net,netsu,zbh,zch,wavg,hdav,usum,vsum,rsum,wfz,
576       1timup,dxsum,dysum,icf,jcf,ncf,kbhf,ndatf,ltimf)
577        common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
578         common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
579        common /space/ wint,xllc,yllc,zmax,zmin,timex
580        dimension rsum(kbhf,ndatf,ltimf),hdav(ltimf),wavg(kbhf,ltimf)
581        dimension net(icf,jcf),netsu(ncf),zch(kbhf),timup(ltimf),zbh(kbhf)
582        dimension dxsum(kbhf,ndatf,ltimf),dysum(kbhf,ndatf,ltimf)
583        dimension usum(kbhf,ndatf,ltimf),vsum(kbhf,ndatf,ltimf)
584        dimension wfz(kbhf,ndatf,ltimf)
585      1  format(1h0,  5x,  17hupdate time indexi3,  23h.  wind grid cell in
586       1dexi3, 38h with horizontal coordinates (x,y) - (e12.5, 1h,, e12.5,
587       2 8h) meters/)
588       2 format(9x,'layer',8x,'horizontal',6x,'horizontal',
589       1 7x,'crosswind', 7x, 'downwind',6x,'horizontal',/,
590       2 8x,'center',8x,'e.-w. wind',6x,'n.-s. wind',2(6x,'turbulence'),
591       3 6x,'rotation',/,8x,'altitude',4(7x,'component'),9x,'angle')
592       6 format(1h1, 40x 21hweighted, summed data//)
593       8 format(/23x,  6hupdatei4,  6h  meshi4, 32h  average turbulence pa
594       1rameter =e12.5)
595       9 format(1h1, 29x, 57hthree dimensional wind and turbulence data bef
596       1ore summing/)
597      12 format(  9x, 5hlayer, 8x, 10hhorizontal, 6x, 10hhorizontal,
598       1 7x, 8hvertical, 8x, 9hcrosswind, 7x, 8hdownwind,7x10hhorizontal/
599       2 8x, 6hcenter, 8x, 10he.-w. wind, 6x, 10hn.-s. wind, 9x,4hwind,3x,
600       3 2(6x, 10hturbulence), 7x, 8hrotation/ 8x, 8haltitude,
601       4 5(7x, 9hcomponent), 9x, 5hangle)
602      13 format(6e16.4)
603      14 format(7e16.4)
604      15 format(1h0,22x,'turbulence parameter averaged over all space',
605       1   ' for update',i4,' is',e12.5)
606      16 format(1h0,22x,'average vertical wind component for each layer -')
```

```
   17 format( 20x, 6(i5, f8.3, 1h,))
      if(mc(2) .eq. 1  .or.  mc(1) .eq. 0) go to 20
      write(isout,9)
      do 50 l=1,ltimx
      do 50 n=1,ndatx
      call cntr(net,netsu,n,xg,yg,icf,jcf,ncf)
      write(isout,1)l,n,xg,yg
      write(isout,12)
      do 50 k=1,kbhx
   50 write(isout,14) zch(k),usum(k,n,l),vsum(k,n,l),wfz(k,n,l),
     1 dxsum(k,n,l),dysum(k,n,l),rsum(k,n,l)
   20 area=icx*jcx*(wint**2)
      if(mc(2) .eq. 2) write(isout,6)
      do 922 l=1,ltimx
      do 1304 lk=1,kbhx
 1304 wavg(lk,l)=0.0
      hdav(l)=0.
      do 921 n=1,ndatx
      if(mc(2).ne.2) go to 915
      call cntr(net,netsu,n,xg,yg,icf,jcf,ncf)
      write(isout,1)l,n,xg,yg
      write(isout,2)
  915 zstep=zbh(2)-zbh(1)
      usum(1,n,l)=usum(1,n,l)*zstep
      vsum(1,n,l)=vsum(1,n,l)*zstep
      rsum(1,n,l)=rsum(1,n,l)*zstep
      dxsum(1,n,l)=dxsum(1,n,l)*zstep
      dysum(1,n,l)=dysum(1,n,l)*zstep
      hav=    (dxsum(1,n,l) + dysum(1,n,l))/2.0
      kbhm1=kbhx-1
      do 920 k=2,kbhm1
      zstep=zbh(k+1) - zbh(k)
      usum(k,n,l)=usum(k,n,l)*zstep + usum(k-1,n,l)
      vsum(k,n,l)=vsum(k,n,l)*zstep + vsum(k-1,n,l)
      rsum(k,n,l)=rsum(k,n,l)*zstep + rsum(k-1,n,l)
      hav=hav+(dxsum(k,n,l) + dysum(k,n,l))*zstep/2.0
      dxsum(k,n,l)=dxsum(k,n,l)*zstep + dxsum(k-1,n,l)
      dysum(k,n,l)=dysum(k,n,l)*zstep + dysum(k-1,n,l)
920   continue
      hav = (hav + (dxsum(kbhx,n,l)+dysum(kbhx,n,l))*(zmax-zbh(kbhx))/2.
     1  )/(zmax-zmin)
      if(mc(2).eq.2)
     1write(isout,13)(zch(k),usum(k,n,l),vsum(k,n,l),dxsum(k,n,l),
     2 dysum(k,n,l),rsum(k,n,l),k=1,kbhx)
      if(mc(2) .ne. 1  .and.  mc(1) .eq. 1) write(isout,8)l,n,hav
      call cntr(net,netsu,n,xg,yg,icf,jcf,ncf)
      xq=xg
      yq=yg
      call       nest(net,netsu,xq,yq,ndatq,xl,xr,yl,yu,icf,jcf,ncf)
      arean=(xr-xl)*(yu-yl)
      hdav(l)= hdav(l) + hav*arean
      do 9210 kl=1,kbhx
9210  wavg(kl,l)= wavg(kl,l) + wfz(kl,n,l)*arean
921   continue
      hdav(l)=hdav(l)/area
      do 9215 kl=1,kbhx
9215  wavg(kl,l)=wavg(kl,l) / area
      if(mc(2) .ne. 1) write(isout,15)l,hdav(l)
      if( mc(2) .eq. 1  .or.  mc(1) .eq. 0) go to 922
      write(isout,16)
      do 9922 kl=1,kbhx,6
      klp5=kl+5
 9922 write(isout,17) (k, wavg(k,l), k=kl,klp5)
 922  continue
      return
      end
```

5.17 TRANP

The tranp subroutine, shown in Listing 17, is responsible for calculating the advection and dispersion of the bottom and top of a given parcel until its impact on the ground, as described in Section 3.4. Either the layer-by-layer or the quick transport method is used, depending on the value of mc(4), which controls the mode variable. The implementation of this subroutine is described in detail in Norment [8] p. 121.

Listing 17. 1979 DELFIC tranp subroutine.

```
      subroutine tranp(net,netsu,zbh,timup,usum,vsum,dxsum,dysum,rsum,
     1wfz,cavs,tsum,tmax,xo,yo,zo,to,sigxo,sigyo,ro,ndato,
     2icf,jcf,ncf,kbhf,ndatf,ltimf)
      character*6 progrm
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
       common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      common /parcl/  cross,down,dwaf,eddy,ndatp,pmas,psiz,rhop,rwaf,
     1 tp,xp,yp,zlow,zp
      common /space/ wint,xllc,yllc,zmax,zmin,timex
      dimension net(icf,jcf),netsu(ncf),zbh(kbhf),usum(kbhf,ndatf,ltimf)
      dimension vsum(kbhf,ndatf,ltimf),dxsum(kbhf,ndatf,ltimf)
      dimension dysum(kbhf,ndatf,ltimf),timup(ltimf),cavs(kbhf)
      dimension rsum(kbhf,ndatf,ltimf),wfz(kbhf,ndatf,ltimf),tsum(kbhf)
      data progrm   , epsilo   , epsz    , qbrt      , varl
     1   /'tranp ' , .0005    , 0.1     ,.3333333333, 1.0e9   /
    2 format( 6h time=e12.4,  5h alt=e12.4, 7h x-pos=e12.4, 7h y-pos=e12
     1.4,   6h mesh=i4,   8h reached)
    3 format( 6h time=e12.4,  5h alt=e12.4, 7h x-pos=e12.4, 7h y-pos=e12
     1.4,   6h mesh=i4,   10h attempted)
    4 format( 1h0, 38hparcel at initial point (xp,yp,zp,tp) 4e12.4/
     1  31h required channelling at point 4e12.4)
      eps=epsilo*wint
      epst=epsilo*tmax
      xo=xp
      yo=yp
      zo=zp
      to=tp
      sigxo=0.
      sigyo=0.
      ro=0.
      ndato=ndatp
      ndtc1=0
      ndto1=0
      kbhc1=0
      kbho1=0
      ltim=1
 1000 continue
      mode=-1
      if(mc(4) .ne. 0) mode=mode+1
   50 mode=mode+1
```

```
      if(ltimx.gt.1)call calib(timup,ltimx,to,1,ltim)
      call calib(zbh,kbhx,zo,1,kbho)
      if(zo-zbh(kbho).gt.epsz)kbho=kbho+1
 100  wbar=-cavs(kbho-1)
      if(mode.eq.0) go to 210
      wbar=wbar+wfz(kbho-1,ndato,ltim)
      if(wbar)206,110,206
  110 tseg=timex-to
      if(ltim.lt.ltimx)tseg=timup(ltim+1)-to
      kbhc=kbho
      kbho=kbhc-1
      go to 300
  206 if(wbar.lt.0.0.or.abs(zo-zbh(kbho)).gt.epsz)if(wbar)210,210,209
      kbho=kbho+1
      go to 100
 209  if(zo-zbh(kbho).lt.-epsz)kbho=kbho-1
  210 kbhc=kbho+ifix(sign(1.0,wbar))
      tseg = (zbh(kbhc)-zo)/wbar
       if(mode.ne.0 .or. abs(zo-zbh(kbho)) .gt. epsz) go to 300
      tseg=tseg+tsum(kbhc)
      wbar=(zmin-zo)/tseg
      kbhc=1
 300  tc=to+tseg
      if(ltim .eq. ltimx) if(timex-tc)301,350,350
      if(timup(ltim+1)- tc) 305,350,350
  301 tseg=timex-to
      tlim=timex
      go to 306
  305 tseg = timup(ltim+1)-to
      tlim=timup(ltim+1)
  306 if(mode .gt. 0  .or.  kbho-kbhc .eq.1) go to 350
      call calib(tsum,kbhx,tc-tlim,-1,kbhc)
      if(kbho .gt. kbhc) go to 310
      kbhc=kbhc-1
      wbar=-cavs(kbhc)
      go to 350
  310 tseg = tsum(kbho) - tsum(kbhc)
      wbar=(zbh(kbhc)-zo)/tseg
  350 kbha=kbho
      kbhb=kbhc
      if(wbar.lt.0.0) go to 405
      kbha=kbhc
      kbhb=kbho
  405 call getda( usum,zbh,kbha,kbhb,ndato,ltim, ubar,kbhf,ndatf,ltimf)
      call getda( vsum,zbh,kbha,kbhb,ndato,ltim, vbar,kbhf,ndatf,ltimf)
  407 continue
      call getda(dxsum,zbh,kbha,kbhb,ndato,ltim,dxbar,kbhf,ndatf,ltimf)
      call getda(dysum,zbh,kbha,kbhb,ndato,ltim,dybar,kbhf,ndatf,ltimf)
      call getda( rsum,zbh,kbha,kbhb,ndato,ltim, rbar,kbhf,ndatf,ltimf)
      rc=ro+rbar
      sigxc=sigxo+dxbar*tseg
      sigyc=sigyo+dybar*tseg
      tc=to+tseg
      zc=zo+wbar*tseg
      xc=xo+ubar*tseg
      yc=yo+vbar*tseg
      call nest(net,netsu,xc,yc,ndatc,xl,xr,yl,yu,icf,jcf,ncf)
      if(mc(5).eq.1) write(isout,3) tc,zc,xc,yc,ndatc
      if(ndatc.eq.ndato) go to 700
      if (mode.eq.0) go to 50
      xt=xc
      yt=yc
      zt=zc
      call boun(net,netsu,xt,y1,xo,yo,xc,yc,icf,jcf,ncf)
      zc=sqrt(((xt-xc)**2+(yt-yc)**2)/((xt-xo)**2+(yt-yo)**2))
      zc=zt+zc*(zo-zt)
      if(abs(wbar).le.1.0e-30) go to 510
      tseg=(zc-zo)/wbar
      go to 518
 510  if(abs(ubar).le.1.0e-30) go to 513
      tseg=(xc-xo)/ubar
      go to 518
 513  if(abs(vbar).le.1.0e-30) go to 516
      tseg=(yc-yo)/vbar
      go to 518
 516  call error(progrm,516,isout)
      return
  518 continue
      rc=ro+rbar
      sigxc=sigxo+dxbar*tseg
      sigyc=sigyo+dybar*tseg
 521  tc=to+tseg
      call nest(net,netsu,xc,yc,ndatc,xl,xr,yl,yu,icf,jcf,ncf)
      if((kbho1.ne.kbho).or.(kbhc1.ne.kbhc).or.(ndtc1.ne.ndatc).or.
     1(ndto1.ne.ndato)) go to 626
      if(mc(5).eq.1) write(isout,4) xp,yp,zp,tp,xc,yc,zc,tc
      call cntr(net,netsu,ndato,xg,yg,icf,jcf,ncf)
      xq=xg
      yq=yg
      call nest(net,netsu,xq,yq,ndatq,xlo,xro,ylo,yuo,icf,jcf,ncf)
      ndtc1=0
      ndto1=0
      kbhc1=0
      kbho1=0
      spe=2.*eps
      if((abs(xlo-xr).gt.spe).and.(abs(xro-xl).gt.spe)) go to 616
      ubar=0.
      call getda( vsum,zbh,kbha,kbhb,ndato,ltim,vbarc,kbhf,ndatf,ltimf)
      if(abs(vbarc).le.abs(vbar)) go to 407
      vbar=vbarc
      ndato=ndatc
      go to 407
 616  if((abs(ylo-yu).gt.spe).and.(abs(yuo-yl).gt.spe))
     1call error(progrm,616,isout)
      vbar=0.
      call getda( usum,zbh,kbha,kbhb,ndato,ltim,ubarc,kbhf,ndatf,ltimf)
      if(abs(ubarc).le.abs(ubar)) go to 407
      ubar=ubarc
      ndato=ndatc
      go to 407
 626  ndtc1=ndato
      ndto1=ndatc
      kbhc1=kbhc
      kbho1=kbho
 700  zo=zc
      xo=xc
      yo=yc
      to=tc
      ndato=ndatc
      if(mc(5).eq.1) write(isout,2) to,zo,xo,yo,ndato
      sigxo=sigxc
      sigyo=sigyc
      ro=rc
 708  if(ndato.le.0) go to 720
      if ((   (zo-zmin).le.epsz).or.(   (timex-to).le.epst)) go to 720
      go to 1000
  720 r2=rwaf**2
      trip=to-tp
      dsprtx=sigxo/trip
      sigxo = ( r2**qbrt + 2.0 * down * trip * dsprtx**qbrt/3.0 ) ** 3
      if( sigxo .gt.   varl ) sigxo =    varl * ( 2.0 * down * trip *
     1  ( dsprtx/  varl )**qbrt + 3.0 * ( r2/  varl )**qbrt - 2.0 )
      sigxo = sqrt( sigxo )
      dsprty = sigyo/trip
      sigyo = ( r2**qbrt + 2.0 * cross * trip * dsprty**qbrt/3.0 ) ** 3
      if( sigyo .gt.   varl ) sigyo =    varl * ( 2.0 * cross * trip *
     1  ( dsprty/  varl )**qbrt + 3.0 * ( r2/  varl )**qbrt - 2.0 )
      sigyo = sqrt( sigyo )
      return
      end
```

5.18 TRIDIN

The `tridin` subroutine, shown in Listing 18, processes the input of wind and turbulence data for a 3D-resolved data field.

Listing 18. 1979 DELFIC `tridin` subroutine.

```
      subroutine tridin(net,netsu,zch,vx,vy,vz,ltim,icf,jcf,ncf,
     1 kbhf,ndatf,ltimf,form,spec)
      character*6 progrm,fmt(12)
      character*4 meteor,resolv,wind,turb,spec,form
      common/cntrol/ipout,isin,isout,jparn,mc(20),nseqo
      common/indexx/icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      dimension zch(kbhf),net(icf,jcf),netsu(ncf),vx(kbhf,ndatf,ltimf)
      dimension vy(kbhf,ndatf,ltimf),vz(kbhf,ndatf,ltimf)
      dimension xs(200),ys(200),zs(200),sx(200),sy(200),sz(200)
      dimension d2(200),nad(200),scale(8),ap(6)
      data alimit,radc, progrm, meteor ,resolv, wind , turb
     1 /999999. ,.0174532925,'tridin','mete' ,'reso','wind','turb'/
      data  jtipf , big     , gib
     1     / 200   , 1.0e+37   ,  1.0e-37   /
   1  format(/18x, 5halpha, 8x, 4hbeta, 14x, 2hnn/ 15x, 2e12.4, I12)
   3  format(8x,I8,6e16.4)
   4  format(/5x, 62hthe data vector at each space lattice center is com
     1puted usingi4,   7h out ofi4, 15h input vectors./)
  8   format(20i4)
  10  format(8f10.0)

  11  format(12a6)
 17    format(2f10.0,i4)
  20  format(//53x, 22hscaled wind data      / 11x, 5hindex, 11x,
     1 1hx, 15x, 1hx, 15x, 1hy, 14x, 2hvx, 14x 2hvy, 14x, 2hvz)
  21  format(//50x,22hscaled turbulence data/ 11x, 5hindex, 11x,
     1 1hz, 15x, 1hx, 15x, 1hy, 12x, 4hepsx, 10x, 4hepsy)
```

```
24    format(//,' no vectors lie within the specifiec weighting region',
     1 '.  a random selection of',i4,' vectors are equally weighted' ,
     2 /,5x,' for grid point',
     3 5x,'(x,y,z)=(',f12.3,',',f12.3,',',f12.3,')')
  31  format( //50x, 22h    raw wind data      / 11x, 5hindex, 11x,
     1 1hx,15x,1hx,15x,1hy,10x,10hvx or cir., 6x,11hvy or speed,
     2  9x, 2hvz)
  32  format( //47x, 22h   raw turbulence data/ 11x, 5hindex, 11x,
     1 1hz, 15x, 1hx, 15x, 1hy, 12x, 4hepsx, 10x, 4hepsy)
  33  format(8x,i8,5e16.4)
      read(isin,17)alpha,beta,nn
      alfa2=alpha**2
      beta2=beta**2
      if(nn .eq. 0) nn=1
      read(isin,11)fmt
      read(isin,10) scale
      do 9 i=1,3
  9   if( scale(i) .eq. 0.0) scale(i) = 1.0
      if( scale( 6).eq. 0.0) scale( 6)= 1.0
      write(isout,1) alpha,beta,nn
      read(isin,8)n1,n2,n3,n4,n5,n6
  13  if(n1+n2+n3+n4+n5+n6 .lt. 21) call error(progrm,-13,isout)
      if( form .eq. meteor) trns = scale(5)*scale(3) - 180.
      if(mc(2) .ne. 1  .and.  spec .eq. wind) write(isout,31)
      if(mc(2) .ne. 1  .and.  spec .eq. turb) write(isout,31)
      j=0
 100  read(isin,fmt)ap
      if(ap(n1).ge. alimit) go to 101
      j=j+1
      if(mc(2) .ne. 1  .and.  spec .eq. wind) write(isout, 3)j,ap(n1),
     1 ap(n5),ap(n6),ap(n2),ap(n3),ap(n4)
      if(mc(2) .ne. 1  .and.  spec .eq. turb) write(isout,33)j,ap(n1),
     1 ap(n5),ap(n6),ap(n2),ap(n3),ap(n4)
      zs(j) = (ap(n1) + scale(4))*scale(1)
      xs(j) = (ap(n5) + scale(7))*scale(6)
      ys(j) = (ap(n6) + scale(8))*scale(6)
      sz(j) = ap(n4) * scale( 2)
      if( form .eq. resolv  .or.  spec .eq. turb ) go to 50
      sx(j) =ap(n3)*scale(2) * sin( radc*(ap(n2)*scale(3) + trns))
      sy(j) =ap(n3)*scale(2) * cos( radc*(ap(n2)*scale(3) + trns))
      go to 100
  50  sx(j) = ap(n2) * scale(2)
      sy(j) = ap(n3) * scale(2)
      go to 100
 101  jtopv=j
      if(mc(2).eq.1) go to 102
      if(spec .eq. wind) write( isout,20)
      if(spec .eq. turb) write( isout,21)
      if(spec .eq. wind)write(isout, 3)(j,zs(j),xs(j),ys(j),sx(j),sy(j),
     1 sz(j),j=1,jtopv)
      if(spec .eq. turb)write(isout,33)(j,zs(j),xs(j),ys(j),sx(j),sy(j)
     1 ,j=1,jtopv)
 102  if(nn.gt.jtopv .or. nn.lt. 0) nn=jtopv
 115  if(nn.lt.1) call error(progrm,-115,isout)
      write(isout,4)nn,jtopv
      nn1=nn+1
      do 906 ndata=1,ndatx
      call cntr(net,netsu,ndata,xg,yg,icf,jcf,ncf)
      do 905 kbh=1,kbhx
      zg=zch(kbh)
      do 203 j=1,jtopv
      nad(j)=j
      tx=xs(j)-xg
      ty=ys(j)-yg
      tz=zs(j)-zg
      cressz=tz*tz
      cutoff=alfa2-cressz
      if(cutoff.le.0) go to 202
      cressz=cutoff/(alfa2+cressz)
      cressr=tx*tx+ty*ty
      cutoff=beta2-cressr
      if(cutoff.le.0) go to 202
      cressr=cutoff/(beta2+cressr)
      cressz=cressz*cressr
      if(cressz.le.gib) go to 202
      d2(j)=1.0/cressz
      go to 203
 202  d2(j)=big
203   continue
      nadt=1
      kl=nad(nadt)
      dm=d2(kl)
      do 207 j=1,nn
      kl=nad(j)
      if(dm-d2(kl))208,207,207
 208   dm=d2(kl)
      nadt=j
 207   continue
       if (nn1-jtopv)2072,2072,2073
 2072  do 210 j=nn1,jtopv
      kl=nad(j)
      if(dm-d2(kl))210,210,211
 211   ntemp=nad(j)
        nad(j)=nad(nadt)
       nad(nadt)=ntemp
       dm=d2(kl)
       do 212 kkk=1,nn
       kl=nad(kkk)
       if(dm-d2(kl))213,212,212
 213    dm=d2(kl)
       nadt=kkk
 212   continue
 210   continue
 2073  continue
 2080  sum=0.0
      do 214 j=1,nn
      l=nad(j)
      d2(l)=1.0/d2(l)
 214   sum=sum+d2(l)
       if(sum/float(nn) .le. gib) write(isout,24) nn,xg,yg,zg
       vxknl=0.0
       vyknl=0.0
       vzknl=0.0
       do 216 j=1,nn
       l=nad(j)
       vxknl=vxknl+sx(l)*d2(l)
       vyknl=vyknl+sy(l)*d2(l)
 216   vzknl=vzknl+sz(l)*d2(l)
```

```
       vxknl=vxknl/sum
       vyknl=vyknl/sum
       vzknl=vzknl/sum
 2090  vx(kbh,ndata,ltim) = vxknl
       vy(kbh,ndata,ltim) = vyknl
       if(form .eq. turb) go to 905
       vz(kbh,ndata,ltim) = vzknl
 905   continue
 906   continue
      return
      end
```

5.19 WILKNS

The `wilkns` subroutine, shown in Listing 19, calculates the vertical TKE dissipation rate ε profile using Eq. 97 or Eq. 104, as described in Section 3.1.1, when the user does not provide the turbulence data directly.

Listing 19. 1979 DELFIC `wilkns` subroutine.

```
      subroutine wilkns (zch,dxsum,dysum,cavs,timup,kbhf,ndatf,ltimf,l)
      character*6 progrm
      common /cntrol/ ipout,isin,isout,jparn,mc(20),nseqo
       common /indexx/ icx,jcx,kbhx,ltimx,nat,ncx,ndatx
      common /space/ wint,xllc,yllc,zmax,zmin,timex
      dimension dxsum(kbhf,ndatf,ltimf),dysum(kbhf,ndatf,ltimf)
      dimension cavs(kbhf),zch(kbhf),timup(ltimf)
      data progrm , vkk    , wilk    , alimit
     1  /'wilkns' , 0.35   , 0.03    , 999999./
 1000 format(4f10.0)
 5000 format(      19x, 1hk, 8x, 3hzch, 12x, 5hdxsum,10x, 5hdysum)
 5100 format( 15x, i5, 3(3x,e12.5))
 5200 format(15x,'turbulence parameters are calculated by wilkins',
     1' reciprocal altitude function',/,15x,'for update',i3,' at',e12.5,
     2 ' seconds',/)
 5300 format(14x,'surface wind speed is',e12.5,3x,' measured at',
     1 ' height',e12.5,/,14x,'roughness length=',e12.5,3x,'reciprocal',
     2 'monin-obukhov length=',e12.5,3x,'(mks units)',/,14x,
     3 'surface layer friction velocity=',e12.5,3x,'(m/sec)',/)
 5900 format(1h0,9x,'cannot compute turbulence via silkins method',
     1' because zch array has not been constructed',/,10x,
     2 'calculation cannot proceed unless wind data are input',//)
      if(zch(1) .ne. alimit) if(mc(2)-1)50,60,50
      write(isout,5900)
   25 call error(progrm,-25,isout)
   50 write(isout,5200)l,timup(l)
   60 read(isin,1000) u, zm, z0, rl
      if(z0 .eq. 0.0) go to 300
      if(rl .ge. 0.0) go to 100
      xi = (1.0 - 15.0*zm*rl)**0.25
      chi = -alog((xi**2+1.0) * (xi+1.0)**2 /8.0) + 2.0*atan (xi)
     1 - 1.5709796327
      go to 200
  100 continue
      cji = 4.7*zm*rl
  200 continue
      ustar = vkk*u / (alog(zm/z0) + chi)
      c = ustar**3/vkk
```

```
    if(mc(2) .ne. 1) write(isout,5300)u,zm,z0,rl,ustar
    go to 400
300 c = wilk
400 continue
    zgz = zmin
    do 500 K=1,kbhx
500 cavs(k) = c/(zch(k) - zgz + z0)
    do 600 n=1,ndatx
    do 600 K=1,kbhx
    dxsum(k,n,l) = cavs(k)
600 dysum(k,n,l) = cavs(k)
    if(mc(2) .eq. 1) return
    write( isout, 5000)
    do 700 K=1,kbhx
700 write(isout,5100) k, zch(k), dxsum(k,1,l),dysum(k,1,l)
    return
    end
```

6. RESULTS

After having described in detail the theory of the DELFIC atmospheric transport and deposition models in Sections 2 through 4 and the 1979 DTM implementation in Section 5, we present in this section the results of running the test case given in the 1979 documentation with the 1979 version of DELFIC [3]. This serves as simple verification that the DTM and DELFIC generally work as described and presented in 1979. This version of the code is written in fixed form Fortran 66 [84], but we compiled it using GNU Fortran [85], which retains compatibility with much of the legacy Fortran 66, Fortran 77 [86], and later standards. We first present in Section 6.1 the original, transcribed input deck presented in Norment [3], Section 4.1. Then we compare in Section 6.2 the complete output of the test problem produced by running the 1979 code to that presented in Norment [3], Section 4.2.

6.1 INPUT

Listing 20 shows the input file used for the test problem of Norment [3], Section 4.1, with spaces explicitly drawn as well. This input file was used to run the transcribed 1979 version of DELFIC. It represents a ground detonation $z_{\mathrm{hob}} = 0\,\mathrm{m}$ of a $W_{\mathrm{weap}} = 50\,\mathrm{kt}$ $^{239}\mathrm{Pu}$ high-energy neutron weapon at an altitude of $z_{\mathrm{GZ}} = 139\,\mathrm{m}$. The wind field is directed predominately towards the north-northeast direction and is moderately strong $u_{xy} \sim 10\,\mathrm{m\,s^{-1}}$.

Listing 20. DELFIC input file for 1979 test problem.

```
␣DELFIC␣TEST␣PROBLEM␣-␣ICRM
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣3
␣50.␣␣␣␣␣␣␣50.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣139.
␣␣30


␣TEST␣PROBLEM␣ATMOSPHERE
(4F10.0,2F1.0)
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣100.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣273.16
␣␣␣1␣␣␣2␣␣␣3␣␣␣4␣␣␣5␣␣␣6
␣␣␣216.␣␣␣␣␣8.2␣␣␣␣␣1000.␣␣␣␣␣53.
␣␣1548.␣␣␣␣␣3.4␣␣␣␣␣␣850.␣␣␣␣␣70.
␣␣3097.␣␣␣␣-6.7␣␣␣␣␣␣700.␣␣␣␣␣78.
␣␣5688.␣␣␣-15.7␣␣␣␣␣␣500.␣␣␣␣␣19.
␣␣7327.␣␣␣-30.3␣␣␣␣␣␣400.␣␣␣␣␣55.
␣␣9309.␣␣␣-47.1␣␣␣␣␣␣300.
␣10488.␣␣␣-56.5␣␣␣␣␣␣250.
␣11887.␣␣␣-60.5␣␣␣␣␣␣200.
␣13698.␣␣␣-57.1␣␣␣␣␣␣150.
␣16267.␣␣␣-56.5␣␣␣␣␣␣100.
␣18526.␣␣␣-56.5␣␣␣␣␣␣␣70.
␣20665.␣␣␣-56.5␣␣␣␣␣␣␣50.
␣23902.␣␣␣-57.7␣␣␣␣␣␣␣30.
␣26493.␣␣␣-53.7␣␣␣␣␣␣␣20.
␣31023.␣␣␣-45.5␣␣␣␣␣␣␣10.
␣999999.
wind␣␣meteorological
(F12.0,24X,2F12.0)

␣␣␣1␣␣␣2␣␣␣3
␣␣216.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣140.␣␣␣␣␣␣␣␣␣␣8.
␣1548.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣155.␣␣␣␣␣␣␣␣␣13.
```

```
␣3097.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣190.␣␣␣␣␣␣␣␣␣␣5.
␣5688.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣200.␣␣␣␣␣␣␣␣␣15.
␣7327.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣215.␣␣␣␣␣␣␣␣␣19.
␣9309.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣220.␣␣␣␣␣␣␣␣␣16.
10488.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣215.␣␣␣␣␣␣␣␣␣11.
11887.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣220.␣␣␣␣␣␣␣␣␣13.
13698.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣235.␣␣␣␣␣␣␣␣␣12.
16267.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣250.␣␣␣␣␣␣␣␣␣␣9.
18526.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣275.␣␣␣␣␣␣␣␣␣␣7.
20665.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣275.␣␣␣␣␣␣␣␣␣␣7.
23902.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣280.␣␣␣␣␣␣␣␣␣11.
26493.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣270.␣␣␣␣␣␣␣␣␣11.
31023.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣275.␣␣␣␣␣␣␣␣␣25.
␣999999.
␣DELFIC␣TEST␣PROBLEM␣-␣DTM


␣␣␣␣␣1.E10␣␣␣-0.5E10␣␣␣-0.5E10␣␣48.
wind␣␣meteorological␣␣␣␣␣␣␣␣␣1
(F12.0,24X,2F12.0)

␣␣␣1␣␣␣2␣␣␣3
␣␣216.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣140.␣␣␣␣␣␣␣␣␣␣8.
␣1548.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣155.␣␣␣␣␣␣␣␣␣13.
␣3097.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣190.␣␣␣␣␣␣␣␣␣␣5.
␣5688.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣200.␣␣␣␣␣␣␣␣␣15.
␣7327.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣215.␣␣␣␣␣␣␣␣␣19.
␣9309.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣220.␣␣␣␣␣␣␣␣␣16.
10488.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣215.␣␣␣␣␣␣␣␣␣11.
11887.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣220.␣␣␣␣␣␣␣␣␣13.
13698.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣235.␣␣␣␣␣␣␣␣␣12.
16267.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣250.␣␣␣␣␣␣␣␣␣␣9.
18526.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣275.␣␣␣␣␣␣␣␣␣␣7.
20665.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣275.␣␣␣␣␣␣␣␣␣␣7.
23902.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣280.␣␣␣␣␣␣␣␣␣11.
26493.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣270.␣␣␣␣␣␣␣␣␣11.
31023.␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣275.␣␣␣␣␣␣␣␣␣25.
␣999999.
turb␣␣wilkins␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣1

no␣more␣data
␣DELFIC␣TEST␣PROBLEM␣-␣OPM

FFFFFFFF
p239he␣␣␣␣␣␣␣1.4
␣-9000.␣␣␣␣14400.␣␣␣␣-3000.␣␣␣␣51000.␣␣␣␣1800.
␣␣␣␣2
```

Discrepancies between the input listed in Norment [3], Section 4.1, and Listing 20 are shown in the `diff` Listing 21.

Listing 21. DELFIC input file `diff` for 1979 test problem.

```diff
--- 1979-original/delfic.in
+++ 1979-current/delfic.in
@@ -24,7 +24,7 @@
  26493.   -53.7       20.
  31023.   -45.5       10.
  999999.
-WIND  METEOROLOGICAL
+wind  meteorological
 (F12.0,24X,2F12.0)

    1   2   3
@@ -48,7 +48,7 @@


      1.E10   -0.5E10   -0.5E10  48.
-WIND  METEOROLOGICAL         1
+wind  meteorological         1
 (F12.0,24X,2F12.0)

    1   2   3
@@ -68,13 +68,13 @@
 26493.                              270.         11.
 31023.                              275.         25.
  999999.
-TURB  WILKINS                1
+turb  wilkins                1

-NO MORE DATA
+no more data
  DELFIC TEST PROBLEM - OPM

-
-P239HE       1.4
+FFFFFFFF
+p239he       1.4
  -9000.    14400.    -3000.    51000.    1800.
     2

```

These differences are essentially formatting differences:

1. Most characters that control the run should be input in lowercase since the transcription of Norment [3], Section 5, code listings to the 1979 version source code was done in lowercase. This includes:
   (a) ICRM card 13
   (b) DTM cards 5h, 8r, and 5t
   (c) OPM card 4
2. OPM card 3 must explicitly define a logical value (i.e., T or F), whereas Norment [3], Section 4.1, leaves this card blank. The GNU Fortran implementation of Fortran 90 will fail to read this card if left blank, however, which appears to be consistent with Fortran 90 [87], Section 10.5.2, requirements for defining logicals.

6.2 OUTPUT

First we present in Section 6.2.1 the complete output generated by running the input deck of Section 6.1 with the transcribed 1979 version of DELFIC. Then in Sections 6.2.2 through 6.2.4 we present visualizations of the stabilized fallout cloud, as well as the individual deposit increments and fallout map. These are generated by first running the modern DELFIC with the input of Listing 20 and options mc(3)=2 in DTM card 2 and jc=4 in OPM card 6 and then parsing and plotting the resulting output. The former option requests the DTM to output the pretransport parcels as well as the deposit increments, while the latter option forces the fallout map to be in a one-line exponential floating point notation for easier processing.

6.2.1 Printed

Listing 22 shows the output file generated by running the transcribed 1979 version of DELFIC with the input of Listing 20.

Listing 22. DELFIC output file for 1979 test problem.

```
  enter name of input data file
1                                                  * * * * * * * * * *

                                                      d e l f i c

            t h e   d e p a r t m e n t   o f   d e f e n s e   f a l l o u t   p r e d i c t i o n   s y s t e m

                                                   * * * * * * * * * *



                                           initialization and cloud rise module


                                                       prepared by
                                              atmospheric science associates
                                                      bedford mass.



                         ****  run identification ****    DELFIC TEST PROBLEM - ICRM

                      ****  the control variable array, ic(j), was given the following values ****
                      0   0   0   3   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0


                         ****  basic parameters ****
                    yields - total (fission)                     0.50000E+02 ( 0.50000E+02) kt
                    height or depth of burst                     0.00000E+00  meters ( 0.00000E+00  feet relative to gz
                    altitude of gz                               0.13900E+03 meters
                    soil category
+                                                                siliceous


                         ****  initial cloud properties at h + 0.43571E+01 seconds  ****
                    average gas temperature                                     0.26288E+04  degrees kelvin
                    average temperature of condensed phase material in cloud      0.13398E+04  degrees kelvin
                    mass of vaporized soil in cloud                             0.00000E+00  kilograms
                    mass of condensed phase material in cloud                   0.28492E+08  kilograms
                    scaled heights of burst                                       0.00000E+00 feet
                    ↪  ( 0.00000E+00 meters)
                    fraction of the total explosion energy in the cloud at the initial time =0.4500
                    fraction of this energy used to heat air and soil =1.0000
                    fraction used to heat liquid water =0.0000
                    fallout solidification temperature =2200.000 (k)
                    detonation coordinates          xgz             ygz             tgz
                                             0.000000E+00    0.000000E+00    0.000000E+00
                    fallout particle density                      0.26000E+04 kg/m**3

                    particle size frequency distribution
                         a log-normal distribution with -
                              median diameter
                     0.40700E+00  micrometers
                              geometric standard deviation        0.40000E+01
                         this distribution was specified by
+                                                                 the program

                    particle volume frequency distribution
                         a log-normal distribution with -
                              median diameter                     0.12986E+03  micrometers
                              geometric standard deviation        0.40000E+01
0         particle size-mass distribution table (diameters are in meters)
          number of particle size classes =   30
0                    diameter    lower boundry     fraction     upper boundary

              1    0.24830E-02    0.16514E-02    0.33333E-01    0.37331E-02
              2    0.13111E-02    0.10409E-02    0.33333E-01    0.16514E-02
              3    0.89391E-03    0.76766E-03    0.33333E-01    0.10409E-02
              4    0.68190E-03    0.60573E-03    0.33333E-01    0.76766E-03
              5    0.54839E-03    0.49648E-03    0.33333E-01    0.60573E-03
              6    0.45499E-03    0.41696E-03    0.33333E-01    0.49648E-03
              7    0.38534E-03    0.35611E-03    0.33333E-01    0.41696E-03
              8    0.33110E-03    0.30784E-03    0.33333E-01    0.35611E-03
              9    0.28751E-03    0.26852E-03    0.33333E-01    0.30784E-03
             10    0.25163E-03    0.23580E-03    0.33333E-01    0.26852E-03
             11    0.22154E-03    0.20813E-03    0.33333E-01    0.23580E-03
             12    0.19591E-03    0.18440E-03    0.33333E-01    0.20813E-03
             13    0.17381E-03    0.16382E-03    0.33333E-01    0.18440E-03
             14    0.15454E-03    0.14579E-03    0.33333E-01    0.16382E-03
             15    0.13760E-03    0.12986E-03    0.33333E-01    0.14579E-03
             16    0.12257E-03    0.11568E-03    0.33333E-01    0.12986E-03
             17    0.10913E-03    0.10295E-03    0.33333E-01    0.11568E-03
             18    0.97031E-04    0.91456E-04    0.33333E-01    0.10295E-03
             19    0.86085E-04    0.81029E-04    0.33333E-01    0.91456E-04
             20    0.76126E-04    0.71520E-04    0.33333E-01    0.81029E-04
             21    0.67022E-04    0.62807E-04    0.33333E-01    0.71520E-04
             22    0.58659E-04    0.54784E-04    0.33333E-01    0.62807E-04
             23    0.50936E-04    0.47359E-04    0.33333E-01    0.54784E-04
             24    0.43767E-04    0.40447E-04    0.33333E-01    0.47359E-04
             25    0.37067E-04    0.33969E-04    0.33333E-01    0.40447E-04
             26    0.30753E-04    0.27842E-04    0.33333E-01    0.33969E-04
             27    0.24732E-04    0.21969E-04    0.33333E-01    0.27842E-04
             28    0.18866E-04    0.16202E-04    0.33333E-01    0.21969E-04
             29    0.12863E-04    0.10212E-04    0.33333E-01    0.16202E-04
             30    0.67923E-05    0.45176E-05    0.33333E-01    0.10212E-04
0                   cloud subdivision parameters -
                       number of cloud subdivisions in the vertical (kdi) =  18
                       parcel horizontal subdivision parameter (irad) =   0
```

```


                                                     * * * * * * * * * *


 leaving icm
1
                    atmosphere identification - TEST PROBLEM ATMOSPHERE



                                          atmosphere

        alt           atp           prs           rlh           rho           eta

  -0.60000E+03   0.29205E+03   0.10887E+06   0.77000E+02   0.12982E+01   0.18081E-04
   0.00000E+00   0.28815E+03   0.10133E+06   0.77000E+02   0.12250E+01   0.17894E-04
   0.60000E+03   0.28029E+03   0.95618E+05   0.59295E+02   0.11862E+01   0.17512E-04
   0.12000E+04   0.27793E+03   0.88898E+05   0.66070E+02   0.11121E+01   0.17396E-04
   0.18000E+04   0.27462E+03   0.82731E+05   0.70608E+02   0.10466E+01   0.17233E-04
   0.24000E+04   0.27085E+03   0.76842E+05   0.74028E+02   0.98553E+00   0.17045E-04
   0.30000E+04   0.26707E+03   0.70952E+05   0.77447E+02   0.92444E+00   0.16858E-04
   0.36000E+04   0.26493E+03   0.66275E+05   0.64401E+02   0.86920E+00   0.16750E-04
   0.42000E+04   0.26278E+03   0.61599E+05   0.51355E+02   0.81397E+00   0.16642E-04
   0.48000E+04   0.26063E+03   0.56922E+05   0.38308E+02   0.75874E+00   0.16534E-04
   0.54000E+04   0.25849E+03   0.52245E+05   0.25262E+02   0.70350E+00   0.16425E-04
   0.60000E+04   0.25432E+03   0.48211E+05   0.28808E+02   0.65940E+00   0.16211E-04
   0.66000E+04   0.24914E+03   0.44498E+05   0.40651E+02   0.62085E+00   0.15944E-04
   0.72000E+04   0.24396E+03   0.40786E+05   0.52493E+02   0.58231E+00   0.15677E-04
   0.78000E+04   0.23887E+03   0.37717E+05   0.37559E+02   0.54829E+00   0.15407E-04
   0.84000E+04   0.23377E+03   0.34649E+05   0.22625E+02   0.51427E+00   0.15137E-04
   0.90000E+04   0.22868E+03   0.31580E+05   0.76911E+01   0.48026E+00   0.14868E-04
   0.96000E+04   0.22378E+03   0.28832E+05   0.18705E+01   0.44824E+00   0.14603E-04
   0.10200E+05   0.21897E+03   0.26243E+05   0.60664E+00   0.41723E+00   0.14342E-04
   0.10800E+05   0.21610E+03   0.23933E+05   0.13550E+00   0.38565E+00   0.14185E-04
   0.11400E+05   0.21420E+03   0.21762E+05   0.60707E-01   0.35379E+00   0.14081E-04
   0.12000E+05   0.21326E+03   0.19755E+05   0.97022E-02   0.32302E+00   0.14029E-04
   0.12600E+05   0.21425E+03   0.18075E+05   0.62739E-02   0.29442E+00   0.14084E-04
   0.13200E+05   0.21524E+03   0.16394E+05   0.28455E-02   0.26581E+00   0.14138E-04
   0.13800E+05   0.21596E+03   0.14879E+05   0.51797E-03   0.24031E+00   0.14178E-04
   0.14400E+05   0.21613E+03   0.13692E+05   0.39199E-03   0.22101E+00   0.14187E-04
   0.15000E+05   0.21630E+03   0.12506E+05   0.26602E-03   0.20170E+00   0.14197E-04
   0.15600E+05   0.21647E+03   0.11319E+05   0.14004E-03   0.18240E+00   0.14206E-04
   0.16200E+05   0.21664E+03   0.10133E+05   0.14067E-04   0.16309E+00   0.14216E-04
   0.16800E+05   0.21665E+03   0.93245E+04   0.10439E-04   0.15008E+00   0.14216E-04
   0.17400E+05   0.21665E+03   0.85164E+04   0.68099E-05   0.13707E+00   0.14216E-04
   0.18000E+05   0.21666E+03   0.77084E+04   0.31812E-05   0.12406E+00   0.14216E-04
   0.18600E+05   0.21666E+03   0.69781E+04   0.69474E-06   0.11231E+00   0.14217E-04
   0.19200E+05   0.21666E+03   0.64034E+04   0.49288E-06   0.10305E+00   0.14217E-04
   0.19800E+05   0.21666E+03   0.58286E+04   0.29102E-06   0.93805E-01   0.14217E-04
   0.20400E+05   0.21666E+03   0.52538E+04   0.89156E-07   0.84554E-01   0.14217E-04
   0.21000E+05   0.21651E+03   0.48124E+04   0.19219E-07   0.77483E-01   0.14209E-04
   0.21600E+05   0.21630E+03   0.44377E+04   0.15246E-07   0.71501E-01   0.14197E-04
   0.22200E+05   0.21608E+03   0.40630E+04   0.11272E-07   0.65519E-01   0.14185E-04
   0.22800E+05   0.21586E+03   0.36883E+04   0.72983E-08   0.59537E-01   0.14173E-04
   0.23400E+05   0.21564E+03   0.33135E+04   0.33246E-08   0.53555E-01   0.14161E-04
   0.24000E+05   0.21579E+03   0.29847E+04   0.62535E-09   0.48246E-01   0.14169E-04
   0.24600E+05   0.21667E+03   0.27477E+04   0.47485E-09   0.44282E-01   0.14217E-04
   0.25200E+05   0.21756E+03   0.25107E+04   0.32434E-09   0.40319E-01   0.14266E-04
   0.25800E+05   0.21844E+03   0.22737E+04   0.17383E-09   0.36355E-01   0.14314E-04
   0.26400E+05   0.21932E+03   0.20367E+04   0.23328E-10   0.32391E-01   0.14363E-04
   0.27000E+05   0.22041E+03   0.19022E+04   0.20301E-10   0.30175E-01   0.14421E-04
   0.27600E+05   0.22149E+03   0.17676E+04   0.17273E-10   0.27959E-01   0.14480E-04
   0.28200E+05   0.22257E+03   0.16331E+04   0.14245E-10   0.25743E-01   0.14539E-04
   0.28800E+05   0.22365E+03   0.14985E+04   0.11218E-10   0.23527E-01   0.14597E-04
   0.29400E+05   0.22473E+03   0.13640E+04   0.81899E-11   0.21311E-01   0.14656E-04
   0.30000E+05   0.22582E+03   0.12294E+04   0.51622E-11   0.19095E-01   0.14715E-04
   0.30600E+05   0.22690E+03   0.10949E+04   0.21345E-11   0.16878E-01   0.14773E-04
   0.31200E+05   0.22807E+03   0.99542E+03   0.42105E-01   0.15250E-01   0.14836E-04
   0.31800E+05   0.22943E+03   0.96619E+03   0.16842E+00   0.14796E-01   0.14906E-04
   0.32400E+05   0.23079E+03   0.93697E+03   0.29474E+00   0.14342E-01   0.14976E-04
   0.33000E+05   0.23215E+03   0.90775E+03   0.42105E+00   0.13888E-01   0.15047E-04
   0.33600E+05   0.23351E+03   0.87853E+03   0.54737E+00   0.13434E-01   0.15117E-04
   0.34200E+05   0.23487E+03   0.84930E+03   0.67368E+00   0.12980E-01   0.15187E-04
   0.34800E+05   0.23622E+03   0.82008E+03   0.80000E+00   0.12526E-01   0.15257E-04
   0.35400E+05   0.23758E+03   0.79086E+03   0.92632E+00   0.12072E-01   0.15328E-04
   0.36000E+05   0.23894E+03   0.76164E+03   0.10526E+01   0.11618E-01   0.15398E-04
   0.36600E+05   0.24030E+03   0.73241E+03   0.11789E+01   0.11164E-01   0.15468E-04
   0.37200E+05   0.24166E+03   0.70319E+03   0.13053E+01   0.10711E-01   0.15538E-04
   0.37800E+05   0.24302E+03   0.67397E+03   0.14316E+01   0.10257E-01   0.15609E-04
   0.38400E+05   0.24438E+03   0.64475E+03   0.15579E+01   0.98027E-02   0.15679E-04
   0.39000E+05   0.24574E+03   0.61552E+03   0.16842E+01   0.93488E-02   0.15749E-04
   0.39600E+05   0.24710E+03   0.58630E+03   0.18105E+01   0.88949E-02   0.15819E-04
   0.40200E+05   0.24845E+03   0.55708E+03   0.19368E+01   0.84410E-02   0.15890E-04
   0.40800E+05   0.24981E+03   0.52786E+03   0.20632E+01   0.79870E-02   0.15960E-04
   0.41400E+05   0.25117E+03   0.49863E+03   0.21895E+01   0.75331E-02   0.16030E-04
   0.42000E+05   0.25253E+03   0.46941E+03   0.23158E+01   0.70792E-02   0.16100E-04
   0.42600E+05   0.25389E+03   0.44019E+03   0.24421E+01   0.66253E-02   0.16171E-04
   0.43200E+05   0.25525E+03   0.41097E+03   0.25684E+01   0.61713E-02   0.16241E-04
   0.43800E+05   0.25661E+03   0.38174E+03   0.26947E+01   0.57174E-02   0.16311E-04
   0.44400E+05   0.25797E+03   0.35252E+03   0.28211E+01   0.52635E-02   0.16381E-04
   0.45000E+05   0.25933E+03   0.32330E+03   0.29474E+01   0.48096E-02   0.16452E-04
   0.45600E+05   0.26068E+03   0.29408E+03   0.30737E+01   0.43557E-02   0.16522E-04
   0.46200E+05   0.26204E+03   0.26485E+03   0.32000E+01   0.39017E-02   0.16592E-04
   0.46800E+05   0.26340E+03   0.23563E+03   0.33263E+01   0.34478E-02   0.16662E-04
   0.47400E+05   0.26476E+03   0.20641E+03   0.34526E+01   0.29939E-02   0.16733E-04
   0.48000E+05   0.26612E+03   0.17719E+03   0.35789E+01   0.25400E-02   0.16803E-04
   0.48600E+05   0.26748E+03   0.14796E+03   0.37053E+01   0.20861E-02   0.16873E-04
   0.49200E+05   0.26884E+03   0.11874E+03   0.38316E+01   0.16321E-02   0.16943E-04
   0.49800E+05   0.27020E+03   0.89520E+02   0.39579E+01   0.11782E-02   0.17014E-04
1                                    shot-time wind data

                  raw data                                    processed data

        z         vx or dir.   vy or speed             z            vx            vy

   2.16000E+02   1.40000E+02   8.00000E+00
+                                                  2.16000E+02  -5.14230E+00   6.12836E+00
   1.54800E+03   1.55000E+02   1.30000E+01
+                                                  1.54800E+03  -5.49404E+00   1.17820E+01
   3.09700E+03   1.90000E+02   5.00000E+00
+                                                  3.09700E+03   8.68241E-01   4.92404E+00
   5.68800E+03   2.00000E+02   1.50000E+01
+                                                  5.68800E+03   5.13030E+00   1.40954E+01
   7.32700E+03   2.15000E+02   1.90000E+01
+                                                  7.32700E+03   1.08980E+01   1.55639E+01
   9.30900E+03   2.20000E+02   1.60000E+01
+                                                  9.30900E+03   1.02846E+01   1.22567E+01
   1.04880E+04   2.15000E+02   1.10000E+01
```

```
+                                                  1.04880E+04   6.30934E+00   9.01067E+00
   1.18870E+04   2.20000E+02   1.30000E+01
+                                                  1.18870E+04   8.35624E+00   9.95858E+00
   1.36980E+04   2.35000E+02   1.20000E+01
+                                                  1.36980E+04   9.82982E+00   6.88292E+00
   1.62670E+04   2.50000E+02   9.00000E+00
+                                                  1.62670E+04   8.45723E+00   3.07818E+00
   1.85260E+04   2.75000E+02   7.00000E+00
+                                                  1.85260E+04   6.97336E+00  -6.10090E-01
   2.06650E+04   2.75000E+02   7.00000E+00
+                                                  2.06650E+04   6.97336E+00  -6.10090E-01
   2.39020E+04   2.80000E+02   1.10000E+01
+                                                  2.39020E+04   1.08329E+01  -1.91013E+00
   2.64930E+04   2.70000E+02   1.10000E+01
+                                                  2.64930E+04   1.10000E+01   1.97439E-08
   3.10230E+04   2.75000E+02   2.50000E+01
+                                                  3.10230E+04   2.49049E+01  -2.17889E+00




          cloud rise is terminated in cxpn at statement 443 by the  u,ek  switch



1




          cloud rise and expansion history table cx

                          DELFIC TEST PROBLEM - ICRM


                                                   cloud history table

      cloud    cloud    cloud    cloud    cloud    base      top      radial temperature gas
      time   interval   base      top    radius    rate      rate      rate                 density
       (sec)    (sec)     (m)      (m)      (m)     (m/sec)    (m/sec)    (m/sec)     (k)
       ↪  (kg/m**3)

  1)  4.357E+00 1.875E-01 2.149E+02 7.263E+02 3.866E+02 6.412E+01 8.645E+01 1.688E+01 2.629E+03
  ↪  1.284E-01
  2)  4.545E+00 4.375E-01 2.269E+02 7.425E+02 3.897E+02 6.725E+01 9.061E+01 1.766E+01 2.545E+03
  ↪  1.323E-01
  3)  4.982E+00 8.750E-01 2.563E+02 7.821E+02 3.975E+02 6.873E+01 9.258E+01 1.803E+01 2.233E+03
  ↪  1.503E-01
  4)  5.857E+00 1.250E+00 3.165E+02 8.631E+02 4.132E+02 6.170E+01 9.453E+01 2.482E+01 1.621E+03
  ↪  2.053E-01
  5)  7.107E+00 1.500E+00 3.936E+02 9.813E+02 4.443E+02 5.675E+01 9.050E+01 2.551E+01 1.188E+03
  ↪  2.770E-01
  6)  8.607E+00 2.500E+00 4.787E+02 1.117E+03 4.825E+02 5.516E+01 8.233E+01 2.054E+01 9.221E+02
  ↪  3.521E-01
  7)  1.111E+01 3.000E+00 6.166E+02 1.323E+03 5.339E+02 5.317E+01 7.574E+01 1.706E+01 6.811E+02
  ↪  4.669E-01
  8)  1.411E+01 3.750E+00 7.761E+02 1.550E+03 5.850E+02 5.109E+01 7.101E+01 1.506E+01 5.410E+02
  ↪  5.739E-01
  9)  1.786E+01 4.500E+00 9.677E+02 1.816E+03 6.415E+02 4.909E+01 6.702E+01 1.355E+01 4.537E+02  6.646E-01
 10)  2.236E+01 5.500E+00 1.189E+03 2.118E+03 7.025E+02 4.657E+01 6.340E+01 1.273E+01 3.983E+02  7.345E-01
 11)  2.786E+01 6.500E+00 1.445E+03 2.467E+03 7.725E+02 4.365E+01 5.963E+01 1.208E+01 3.599E+02  7.842E-01
 12)  3.436E+01 7.750E+00 1.728E+03 2.854E+03 8.510E+02 4.070E+01 5.574E+01 1.137E+01 3.329E+02  8.135E-01
 13)  4.211E+01 8.750E+00 2.044E+03 3.286E+03 9.391E+02 3.808E+01 5.198E+01 1.051E+01 3.131E+02  8.243E-01
 14)  5.086E+01 1.000E+01 2.377E+03 3.741E+03 1.031E+03 3.579E+01 4.832E+01 9.468E+00 2.988E+02  8.204E-01
 15)  6.086E+01 1.150E+01 2.735E+03 4.224E+03 1.126E+03 3.283E+01 4.474E+01 9.001E+00 2.877E+02  8.124E-01
 16)  7.236E+01 1.300E+01 3.113E+03 4.739E+03 1.229E+03 2.940E+01 4.074E+01 8.575E+00 2.788E+02  7.952E-01
 17)  8.536E+01 1.450E+01 3.495E+03 5.268E+03 1.341E+03 2.570E+01 3.640E+01 8.092E+00 2.715E+02  7.711E-01
 18)  9.986E+01 1.700E+01 3.867E+03 5.796E+03 1.458E+03 2.164E+01 3.164E+01 7.562E+00 2.656E+02  7.425E-01
 19)  1.169E+02 1.750E+01 4.235E+03 6.334E+03 1.587E+03 1.742E+01 2.662E+01 6.955E+00 2.605E+02  7.099E-01
 20)  1.344E+02 2.000E+01 4.540E+03 6.800E+03 1.708E+03 1.346E+01 2.158E+01 6.138E+00 2.567E+02  6.813E-01
 21)  1.544E+02 2.025E+01 4.809E+03 7.232E+03 1.831E+03 1.009E+01 1.751E+01 5.615E+00 2.534E+02  6.604E-01
 22)  1.746E+02 2.500E+01 5.014E+03 7.586E+03 1.945E+03 7.365E+00 1.403E+01 5.041E+00 2.509E+02  6.431E-01
 23)  1.996E+02 2.500E+01 5.198E+03 7.937E+03 2.071E+03 5.197E+00 1.119E+01 4.529E+00 2.488E+02  6.257E-01
 24)  2.246E+02 2.750E+01 5.328E+03 8.217E+03 2.184E+03 3.510E+00 8.932E+00 4.099E+00 2.471E+02  6.122E-01
 25)  2.521E+02 3.000E+01 5.424E+03 8.462E+03 2.297E+03 2.080E+00 6.999E+00 3.719E+00 2.456E+02  6.008E-01
 26)  2.821E+02 3.250E+01 5.487E+03 8.672E+03 2.408E+03 8.383E-01 5.315E+00 3.384E+00 2.445E+02  5.917E-01
 27)  3.146E+02 3.250E+01 5.514E+03 8.845E+03 2.518E+03-2.113E-01 3.852E+00 3.071E+00 2.436E+02  5.849E-01
 28)  3.471E+02 3.750E+01 5.507E+03 8.970E+03 2.618E+03-1.199E+00 2.544E+00 2.829E+00 2.431E+02  5.815E-01
 29)  3.846E+02 4.000E+01 5.462E+03 9.066E+03 2.724E+03-1.628E-01 1.738E-01 3.876E+00 2.429E+02  5.802E-01
 30)  4.246E+02 4.250E+01 5.455E+03 9.073E+03 2.879E+03 0.000E+00 0.000E+00 3.879E+00 2.428E+02  5.803E-01
 31)  4.671E+02 0.000E+00 5.455E+03 9.073E+03 3.044E+03 0.000E+00 0.000E+00 0.000E+00 2.428E+02  5.803E-01





          time of soil solidification at temperature 2200.0000 deg. is   5.0292 sec.
1                        cloud trajectory
      xc            yc            zc            tc            vc
 -0.224053E+02  0.267016E+02  0.470563E+03  0.435707E+01  0.752824E+02
 -0.233695E+02  0.278507E+02  0.484678E+03  0.454457E+01  0.789308E+02
 -0.256193E+02  0.305319E+02  0.519211E+03  0.498207E+01  0.806580E+02
 -0.301188E+02  0.358942E+02  0.589786E+03  0.585707E+01  0.781189E+02
 -0.365467E+02  0.435546E+02  0.687435E+03  0.710707E+01  0.736221E+02
 -0.442601E+02  0.527472E+02  0.797868E+03  0.860707E+01  0.687438E+02
 -0.575647E+02  0.752830E+02  0.969728E+03  0.111071E+02  0.644564E+02
 -0.740469E+02  0.110629E+03  0.116310E+04  0.141071E+02  0.610487E+02
 -0.946495E+02  0.154811E+03  0.139203E+04  0.178571E+02  0.580527E+02
 -0.119373E+03  0.207830E+03  0.165327E+04  0.223571E+02  0.549861E+02
 -0.149590E+03  0.272631E+03  0.195569E+04  0.278571E+02  0.516419E+02
 -0.185301E+03  0.349214E+03  0.229136E+04  0.343571E+02  0.482174E+02
 -0.182681E+03  0.391804E+03  0.266505E+04  0.421071E+02  0.450290E+02
 -0.175084E+03  0.434890E+03  0.305905E+04  0.508571E+02  0.420530E+02
 -0.166401E+03  0.484130E+03  0.347958E+04  0.608571E+02  0.387868E+02
 -0.156417E+03  0.540757E+03  0.392563E+04  0.723571E+02  0.350732E+02
 -0.145129E+03  0.604769E+03  0.438158E+04  0.853571E+02  0.310509E+02
 -0.722386E+02  0.805927E+03  0.483182E+04  0.998571E+02  0.266422E+02
  0.149765E+02  0.104555E+04  0.528474E+04  0.116857E+03  0.220158E+02
  0.104757E+03  0.129222E+04  0.567001E+04  0.134357E+03  0.175217E+02
  0.207363E+03  0.157413E+04  0.602045E+04  0.154357E+03  0.138002E+02
  0.311251E+03  0.185956E+04  0.629990E+04  0.174607E+03  0.106996E+02
  0.471793E+03  0.222016E+04  0.656739E+04  0.199607E+03  0.819257E+01
  0.744242E+03  0.260926E+04  0.677220E+04  0.224607E+03  0.622080E+01
  0.104394E+04  0.303727E+04  0.694328E+04  0.252107E+03  0.453958E+01
  0.137087E+04  0.350418E+04  0.707946E+04  0.282107E+03  0.307648E+01
  0.172506E+04  0.401001E+04  0.717945E+04  0.314607E+03  0.182015E+01
  0.207924E+04  0.451584E+04  0.723860E+04  0.347107E+03  0.672708E+00
  0.248791E+04  0.509948E+04  0.726383E+04  0.384607E+03  0.549942E-02
  0.292383E+04  0.572204E+04  0.726405E+04  0.424607E+03  0.000000E+00
  0.338700E+04  0.638350E+04  0.726405E+04  0.467107E+03  0.000000E+00
1


                                                   * * * * * * * * * *

                                                       d e l f i c

            t h e   d e p a r t m e n t   o f   d e f e n s e    f a l l o u t   p r e d i c t i o n   s y s t e m

                                                   * * * * * * * * * *



                                                diffusive transport module



                                                       prepared by
                                              atmospheric science associates
                                                      bedford, mass.




                                         *****  summary of run identifiers  *****
          initialization and cloud rise module identification DELFIC TEST PROBLEM - ICRM
          diffusive transport module identification - DELFIC TEST PROBLEM - DTM

               the control variable array, mc(j), has been given the values -
                  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0

                            the transport time limit is   172800.000 sec. (  48.00000 hours)
               coordinates of ground zero (xgz,ygz,zgz) are (  0.00000E+00  0.00000E+00  0.13900E+03)(meters)
```

```
                                      detonation time is 0.00000E+00 seconds

                horizontal coordinates of the south west corner of the transport space are (  -0.50000E+10 -0.50000E+10)
                                    the resolution net spacing is  0.10000E+11 (all in meters)
                a plane deposition surface at altitude  139.000(meters above msl) is assumed

                fallout particle density is  0.26000E+04 kg/m**3   there are   30 particle size  classes

                transport is by the
 +                                  quick method
 1



                atmosphere update   1 for times later than 0.00000E+00sec (   0.000 hours)

                      * * * * * * * * * windfield data * * * * * * * * *

                 raw wind data                                processed wind data

         z         vx or dir.   vy or speed              z            vx            vy

    2.16000E+02   1.40000E+02   8.00000E+00
 +                                                2.16000E+02  -5.14230E+00   6.12836E+00
    1.54800E+03   1.55000E+02   1.30000E+01
 +                                                1.54800E+03  -5.49404E+00   1.17820E+01
    3.09700E+03   1.90000E+02   5.00000E+00
 +                                                3.09700E+03   8.68241E-01   4.92404E+00
    5.68800E+03   2.00000E+02   1.50000E+01
 +                                                5.68800E+03   5.13030E+00   1.40954E+01
    7.32700E+03   2.15000E+02   1.90000E+01
 +                                                7.32700E+03   1.08980E+01   1.55639E+01
    9.30900E+03   2.20000E+02   1.60000E+01
 +                                                9.30900E+03   1.02846E+01   1.22567E+01
    1.04880E+04   2.15000E+02   1.10000E+01
 +                                                1.04880E+04   6.30934E+00   9.01067E+00
    1.18870E+04   2.20000E+02   1.30000E+01
 +                                                1.18870E+04   8.35624E+00   9.95858E+00
    1.36980E+04   2.35000E+02   1.20000E+01
 +                                                1.36980E+04   9.82982E+00   6.88292E+00
    1.62670E+04   2.50000E+02   9.00000E+00
 +                                                1.62670E+04   8.45723E+00   3.07818E+00
    1.85260E+04   2.75000E+02   7.00000E+00
 +                                                1.85260E+04   6.97336E+00  -6.10090E-01
    2.06650E+04   2.75000E+02   7.00000E+00
 +                                                2.06650E+04   6.97336E+00  -6.10090E-01
    2.39020E+04   2.80000E+02   1.10000E+01
 +                                                2.39020E+04   1.08329E+01  -1.91013E+00
    2.64930E+04   2.70000E+02   1.10000E+01
 +                                                2.64930E+04   1.10000E+01   1.97439E-08
    3.10230E+04   2.75000E+02   2.50000E+01
 +                                                3.10230E+04   2.49049E+01  -2.17889E+00




                                  wind layer base altitudes

     levels   1 thru   8   139.00000   293.00000  2803.00000  3391.00000  7985.00000  6669.00000  11949.00000  9027.00000
     levels   9 thru  15 14747.00000 12649.00000 19885.00000 17167.00000 24163.00000 23641.00000  29345.00000
0   maximum wind space altitude is 0.32701E+05 meters



                atmosphere update   1 for times later than 0.00000E+00sec (   0.000 hours)

                       * * * * * * * * turbulence data * * * * * * * *

                turbulence parameters are calculated by wilkins reciprocal altitude function
                for update  1 at 0.00000E+00 seconds

                    k        zch           dxsum          dysum
                    1    0.21600E+03    0.38961E-03    0.38961E-03
                    2    0.15480E+04    0.21292E-04    0.21292E-04
                    3    0.30970E+04    0.10142E-04    0.10142E-04
                    4    0.56880E+04    0.54064E-05    0.54064E-05
                    5    0.73270E+04    0.41736E-05    0.41736E-05
                    6    0.93090E+04    0.32715E-05    0.32715E-05
                    7    0.10488E+05    0.28988E-05    0.28988E-05
                    8    0.11887E+05    0.25536E-05    0.25536E-05
                    9    0.13698E+05    0.22126E-05    0.22126E-05
                   10    0.16267E+05    0.18601E-05    0.18601E-05
                   11    0.18526E+05    0.16316E-05    0.16316E-05
                   12    0.20665E+05    0.14616E-05    0.14616E-05
                   13    0.23902E+05    0.12625E-05    0.12625E-05
                   14    0.26493E+05    0.11383E-05    0.11383E-05
                   15    0.31023E+05    0.97138E-06    0.97138E-06
0                      turbulence parameter averaged over all space for update   1 is 0.57080E-05

                                                type of fission is p239he
1                                                  * * * * * * * * * *

                                                        d e l f i c

             t h e   d e p a r t m e n t    o f   d e f e n s e   f a l l o u t   p r e d i c t i o n   s y s t e m

                                                    * * * * * * * * * *

                                                 output processor module

                                                        prepared by
                                              atmospheric science associates
                                                      bedford, mass.


                                          ****  summary of run identifiers  ****
                                          output processor -  DELFIC TEST PROBLEM - OPM
                             initialization and cloud rise -  DELFIC TEST PROBLEM - ICRM
                                       diffusive transport -  DELFIC TEST PROBLEM - DTM

                       ****  the control variable array, ic(j), was given the following values ****
                       0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0

```

```
                                    total yield is  5.0000E+01 kilotons.
                                  fission yield is  5.0000E+01 kilotons.

                                  the height of burst is    0.000 meters.

                       soil induced activity - neutrons emitted per fission are  1.400

              the cloud reached the soil condensation temperature of  2200.0 at   5.0292
              ↪  sec.

                                      there are  30 particle classes

                                printer description  -  characters per inch
                                   horizontal   10          vertical  6
 1


                                ****  output processor task    1 ****



        grid linits and intervals
             xmin          xmax          ymin          ymax          delta x        delta y
            -9000.        14400.        -3000.        51000.        1800.00        1500.00


        combined ground roughness-instrument response factor     1.000     altitude of gz
        ↪  139.000 meters above msl

        undistorted maps are produced by the grid increments printed above




        request #   1


        map type  2          t1=      0.00          t2=      0.00          maschn=   0

        qcut= 0.10000E-03          cutmap= 0.10000E-01




                               table of total activity in each particle size class

     psize          fp           psize          fp           psize          fp           psize
     ↪  fp
   2.4830E-03    6.0769E+09    2.8751E-04    7.4839E+09    1.0913E-04    8.9812E+09    3.7067E-05
   ↪  1.2347E+10
   1.3111E-03    6.3445E+09    2.5163E-04    7.6397E+09    9.7031E-05    9.2362E+09    3.0753E-05
   ↪  1.3248E+10
   8.9391E-04    6.5504E+09    2.2154E-04    7.8010E+09    8.6085E-05    9.5179E+09    2.4732E-05
   ↪  1.4474E+10
   6.8190E-04    6.7229E+09    1.9591E-04    7.9693E+09    7.6126E-05    9.8326E+09    1.8866E-05
   ↪  1.6313E+10
   5.4839E-04    6.8812E+09    1.7381E-04    8.1462E+09    6.7022E-05    1.0188E+10    1.2863E-05
   ↪  1.9684E+10
```

```
    4.5499E-04    7.0330E+09    1.5454E-04    8.3336E+09    5.8659E-05    1.0597E+10    6.7923E-06    2.8232E+10
    3.8534E-04    7.1824E+09    1.3760E-04    8.5334E+09    5.0936E-05    1.1074E+10    0.0000E+00    0.0000E+00
    3.3110E-04    7.3321E+09    1.2257E-04    8.7482E+09    4.3767E-05    1.1645E+10    0.0000E+00    0.0000E+00
0   k factors computed from the fp table -   6.0830E+09                            2.3486E+03
+                                                      (r-m**2)/(hr-kt)                          (r-mi**2)/(hr-kt)


              this map uses the
+                              two-line e format


              the quantity presented is
              exposure rate normalized to time h+1 hour.
              units are roentgens per hour
              ground zero is located at x =       0.0 , y=       0.0
1strip  1      DELFIC TEST PROBLEM - OPM                                              map type   2

   ***       -7200.      -3600.          0.       3600.       7200.      10800.      14400.      18000.      21600.      25200. **


                1     1     1     1     1     1     2     1     1     0     0    -1     0
       51000.   1.307 4.138 3.999 5.705 6.990 9.142 1.001 7.550 3.311 7.395 0.766 0.344 0.000

                1     1     1     1     1     1     2     1     1     0     0    -1     0
       49500.   1.142 2.377 5.799 6.120 7.241 9.673 1.042 7.650 3.200 6.579 0.605 0.234 0.000

                0     1     1     1     1     2     2     1     1     0     0    -1     0
       48000.   4.544 3.050 8.044 6.032 7.676 1.029 1.092 7.751 3.007 5.501 0.438 0.141 0.000

                0     1     1     1     1     2     2     1     1     0     0     0     0
       46500.   3.078 4.392 7.743 5.872 8.287 1.090 1.135 7.614 2.673 4.312 0.298 0.000 0.000

                0     1     1     1     1     2     2     1     1     0     0     0     0
       45000.   2.737 5.451 5.195 6.476 8.821 1.144 1.162 7.283 2.329 3.383 0.208 0.000 0.000

                0     1     1     1     1     2     2     1     1     0     0     0     0
       43500.   2.421 5.619 4.967 7.558 9.140 1.198 1.184 7.014 2.089 2.774 0.152 0.000 0.000

                0     1     1     1     1     2     2     1     1     0     0     0     0
       42000.   3.455 3.130 8.067 8.131 9.497 1.259 1.215 6.888 1.905 2.268 0.108 0.000 0.000

                0     1     2     1     2     2     2     1     1     0    -1     0     0
       40500.   3.307 2.281 1.172 7.875 1.013 1.323 1.252 6.700 1.677 1.749 0.712 0.000 0.000

                0     1     1     1     2     2     2     1     1     0    -1     0     0
       39000.   1.404 2.738 9.744 8.367 1.079 1.386 1.274 6.309 1.415 1.300 0.458 0.000 0.000

                0     1     1     2     2     2     2     1     1     0    -1     0     0
       37500.   0.626 3.915 6.399 1.011 1.123 1.448 1.277 5.861 1.196 0.983 0.304 0.000 0.000

                0     1     1     2     2     2     2     1     1     0    -1     0     0
       36000.   0.407 4.678 9.006 1.123 1.185 1.512 1.279 5.507 1.028 0.752 0.200 0.000 0.000

```

```
              0     1     2     2     2     2     2     1     0     0    -1     0     0
    34500.   0.445 1.967 1.447 1.087 1.289 1.573 1.283 5.153 8.652 0.554 0.124 0.000 0.000

              0     1     2     2     2     2     2     1     0     0     0     0     0
    33000.   0.418 1.155 1.194 1.227 1.382 1.629 1.271 4.693 7.033 0.394 0.000 0.000 0.000

              0     1     1     2     2     2     2     1     0     0     0     0     0
    31500.   0.186 1.566 7.613 1.529 1.458 1.677 1.239 4.210 5.676 0.282 0.000 0.000 0.000

             -1     1     2     2     2     2     2     1     0     0     0     0     0
    30000.   0.908 4.599 1.132 1.570 1.583 1.723 1.206 3.775 4.567 0.198 0.000 0.000 0.000

             -1     1     2     2     2     2     2     1     0     0     0     0     0
    28500.   0.716 2.342 1.603 1.626 1.715 1.773 1.167 3.323 3.552 0.133 0.000 0.000 0.000

              0     0     1     2     2     2     2     1     0    -1     0     0     0
    27000.   0.111 4.758 9.912 2.030 1.829 1.817 1.104 2.824 2.642 0.849 0.000 0.000 0.000

             -1     1     1     2     2     2     2     1     0    -1     0     0     0
    25500.   0.545 1.267 8.717 2.090 2.004 1.838 1.020 2.322 1.879 0.505 0.000 0.000 0.000

             -1     1     2     2     2     2     1     1     0    -1     0     0     0
    24000.   0.156 5.414 1.546 2.116 2.167 1.849 9.207 1.825 1.239 0.272 0.000 0.000 0.000

             -1     1     2     2     2     2     1     1     0    -1     0     0     0
    22500.   0.131 1.321 1.363 2.527 2.327 1.841 7.993 1.332 0.731 0.123 0.000 0.000 0.000

             -1     0     1     2     2     2     1     0     0     0     0     0     0
    21000.   0.375 3.341 9.002 2.544 2.518 1.797 6.565 8.825 0.378 0.000 0.000 0.000 0.000

             -1     1     2     2     2     2     1     0     0     0     0     0     0
    19500.   0.121 5.753 1.668 2.778 2.657 1.702 5.005 5.153 0.165 0.000 0.000 0.000 0.000

              0     1     2     2     2     2     1     0    -1     0     0     0     0
    18000.   0.000 4.158 1.657 3.107 2.782 1.531 3.418 2.518 0.578 0.000 0.000 0.000 0.000

             -1     0     2     2     2     2     1     0    -1     0     0     0     0
    16500.   0.449 4.050 1.301 3.138 2.818 1.275 2.011 0.981 0.151 0.000 0.000 0.000 0.000

             -1     1     2     2     2     1     0     0     0     0     0     0     0
    15000.   0.633 6.724 2.131 3.731 2.766 9.488 9.798 0.295 0.000 0.000 0.000 0.000 0.000

              0     0     2     2     2     1     0    -1     0     0     0     0     0
    13500.   0.000 6.010 1.587 3.969 2.617 6.101 3.772 0.659 0.000 0.000 0.000 0.000 0.000

              0     0     2     2     2     1     0    -1     0     0     0     0     0
    12000.   0.000 4.477 3.898 5.040 2.410 3.465 1.149 0.106 0.000 0.000 0.000 0.000 0.000

              0     0     2     2     2     1     0     0     0     0     0     0     0
    10500.   0.000 0.825 4.543 5.415 2.103 1.809 0.288 0.000 0.000 0.000 0.000 0.000 0.000

              0     0     2     2     2     0    -1     0     0     0     0     0     0
     9000.   0.000 0.278 1.453 6.117 1.843 8.096 0.551 0.000 0.000 0.000 0.000 0.000 0.000

              0    -1     1     3     2     0     0     0     0     0     0     0     0
     7500.   0.000 0.949 3.291 1.006 2.033 4.047 0.000 0.000 0.000 0.000 0.000 0.000 0.000

              0    -1     0     3     2     0     0     0     0     0     0     0     0
     6000.   0.000 0.243 3.806 1.296 2.716 6.933 0.000 0.000 0.000 0.000 0.000 0.000 0.000
```

```

                 0     0     0     2     2     1    -1     0     0     0     0     0     0
        4500.   0.000 0.000 0.256 6.935 4.099 1.731 0.615 0.000 0.000 0.000 0.000 0.000 0.000

                 0     0    -1     2     2     1     0     0     0     0     0     0     0
        3000.   0.000 0.000 0.167 6.515 5.462 2.528 0.214 0.000 0.000 0.000 0.000 0.000 0.000

                 0     0     0     1     1     1     0     0     0     0     0     0     0
        1500.   0.000 0.000 0.000 2.716 8.148 1.814 0.352 0.000 0.000 0.000 0.000 0.000 0.000

                 0     0     0     0     0     0     0     0     0     0     0     0     0
           0.   0.000 0.000 0.000 0.000 1.814 5.548 0.274 0.000 0.000 0.000 0.000 0.000 0.000

                 0     0     0     0    -1     0    -1     0     0     0     0     0     0
       -1500.   0.000 0.000 0.000 0.000 0.149 0.539 0.951 0.000 0.000 0.000 0.000 0.000 0.000

   ***       -7200.      -3600.          0.       3600.       7200.      10800.      14400.      18000.      21600.      25200. **



               sum of map ordinates =  0.284241E+05
 output processing is completed.
```

As was the case for the input file discussed in Section 6.1, there are several formatting differences in the output files:

1. By default, the Fortran 90 [87] and later standards specify that `advance` defaults to `YES`, meaning that input/output statements will advance to the next record by default. This results in extraneous newline characters in parts of the output of Listing 22 compared with Norment [3].
2. Several Hollerith character strings appear in the output of Listing 22, whereas they are not present in Norment [3].
3. Numbers are formatted with a zero explicitly appearing before leading decimals.
4. Zero values are formatted in exponent notation.
5. Whitespace or alignment are different.

To simplify the output comparison, `diff` Listing 23 shows the discrepancies between the output listed in Norment [3], Section 4.2, and Listing 22 when using consistent formatting. That is, Listing 23 only shows the `diff` output attributable to numerical or modeling discrepancies and is formatted according to the output format of the transcribed 1979 code.

Listing 23. DELFIC output file `diff` for 1979 test problem.

```
--- 1979-original/delfic.out
+++ 1979-current/delfic.out
@@ -227,7 +227,7 @@
    2.39020E+04   2.80000E+02   1.10000E+01
 +                                                  2.39020E+04   1.08329E+01  -1.91013E+00
    2.64930E+04   2.70000E+02   1.10000E+01
-+                                                  2.64930E+04   1.10000E+01   1.97438E-08
++                                                  2.64930E+04   1.10000E+01   1.97439E-08
    3.10230E+04   2.75000E+02   2.50000E+01
 +                                                  3.10230E+04   2.49049E+01  -2.17889E+00
```

```
@@ -296,35 +296,35 @@
        xc            yc            zc            tc            vc
  -0.224053E+02  0.267016E+02  0.470563E+03  0.435707E+01  0.752824E+02
  -0.233695E+02  0.278507E+02  0.484678E+03  0.454457E+01  0.789308E+02
- -0.256193E+02  0.305319E+02  0.519210E+03  0.498207E+01  0.806580E+02
- -0.301188E+02  0.358942E+02  0.589786E+03  0.585707E+01  0.781188E+02
- -0.365467E+02  0.435546E+02  0.687435E+03  0.710707E+01  0.736220E+02
+ -0.256193E+02  0.305319E+02  0.519211E+03  0.498207E+01  0.806580E+02
+ -0.301188E+02  0.358942E+02  0.589786E+03  0.585707E+01  0.781189E+02
+ -0.365467E+02  0.435546E+02  0.687435E+03  0.710707E+01  0.736221E+02
  -0.442601E+02  0.527472E+02  0.797868E+03  0.860707E+01  0.687438E+02
- -0.575647E+02  0.752829E+02  0.969727E+03  0.111071E+02  0.644564E+02
+ -0.575647E+02  0.752830E+02  0.969728E+03  0.111071E+02  0.644564E+02
  -0.740469E+02  0.110629E+03  0.116310E+04  0.141071E+02  0.610487E+02
  -0.946495E+02  0.154811E+03  0.139203E+04  0.178571E+02  0.580527E+02
  -0.119373E+03  0.207830E+03  0.165327E+04  0.223571E+02  0.549861E+02
  -0.149590E+03  0.272631E+03  0.195569E+04  0.278571E+02  0.516419E+02
- -0.185301E+03  0.349214E+03  0.229136E+04  0.343571E+02  0.482175E+02
+ -0.185301E+03  0.349214E+03  0.229136E+04  0.343571E+02  0.482174E+02
  -0.182681E+03  0.391804E+03  0.266505E+04  0.421071E+02  0.450290E+02
  -0.175084E+03  0.434890E+03  0.305905E+04  0.508571E+02  0.420530E+02
- -0.166401E+03  0.484130E+03  0.347958E+04  0.608571E+02  0.387869E+02
+ -0.166401E+03  0.484130E+03  0.347958E+04  0.608571E+02  0.387868E+02
  -0.156417E+03  0.540757E+03  0.392563E+04  0.723571E+02  0.350732E+02
  -0.145129E+03  0.604769E+03  0.438158E+04  0.853571E+02  0.310509E+02
- -0.722385E+02  0.805928E+03  0.483182E+04  0.998571E+02  0.266422E+02
-  0.149766E+02  0.104555E+04  0.528474E+04  0.116857E+03  0.220158E+02
+ -0.722386E+02  0.805927E+03  0.483182E+04  0.998571E+02  0.266422E+02
+  0.149765E+02  0.104555E+04  0.528474E+04  0.116857E+03  0.220158E+02
   0.104757E+03  0.129222E+04  0.567001E+04  0.134357E+03  0.175217E+02
   0.207363E+03  0.157413E+04  0.602045E+04  0.154357E+03  0.138002E+02
-  0.311252E+03  0.185956E+04  0.629990E+04  0.174607E+03  0.106996E+02
-  0.471794E+03  0.222016E+04  0.656739E+04  0.199607E+03  0.819258E+01
-  0.744243E+03  0.260926E+04  0.677221E+04  0.224607E+03  0.622082E+01
-  0.104394E+04  0.303727E+04  0.694328E+04  0.252107E+03  0.453959E+01
-  0.137088E+04  0.350418E+04  0.707947E+04  0.282107E+03  0.307650E+01
-  0.172506E+04  0.401001E+04  0.717945E+04  0.314607E+03  0.182016E+01
-  0.207924E+04  0.451584E+04  0.723861E+04  0.347107E+03  0.672723E+00
-  0.248792E+04  0.509948E+04  0.726384E+04  0.384607E+03  0.550127E-02
-  0.292383E+04  0.572204E+04  0.726406E+04  0.424607E+03  0.000000E+00
-  0.338700E+04  0.638350E+04  0.726406E+04  0.467107E+03  0.000000E+00
+  0.311251E+03  0.185956E+04  0.629990E+04  0.174607E+03  0.106996E+02
+  0.471793E+03  0.222016E+04  0.656739E+04  0.199607E+03  0.819257E+01
+  0.744242E+03  0.260926E+04  0.677220E+04  0.224607E+03  0.622080E+01
+  0.104394E+04  0.303727E+04  0.694328E+04  0.252107E+03  0.453958E+01
+  0.137087E+04  0.350418E+04  0.707946E+04  0.282107E+03  0.307648E+01
+  0.172506E+04  0.401001E+04  0.717945E+04  0.314607E+03  0.182015E+01
+  0.207924E+04  0.451584E+04  0.723860E+04  0.347107E+03  0.672708E+00
+  0.248791E+04  0.509948E+04  0.726383E+04  0.384607E+03  0.549942E-02
+  0.292383E+04  0.572204E+04  0.726405E+04  0.424607E+03  0.000000E+00
+  0.338700E+04  0.638350E+04  0.726405E+04  0.467107E+03  0.000000E+00
 1


@@ -405,7 +405,7 @@
    2.39020E+04   2.80000E+02   1.10000E+01
 +                                                2.39020E+04   1.08329E+01  -1.91013E+00
    2.64930E+04   2.70000E+02   1.10000E+01
```

```
-+                                                   2.64930E+04   1.10000E+01   1.97438E-08
++                                                   2.64930E+04   1.10000E+01   1.97439E-08
    3.10230E+04   2.75000E+02   2.50000E+01
 +                                                   3.10230E+04   2.49049E+01  -2.17889E+00

@@ -443,7 +443,6 @@
                   13     0.23902E+05     0.12625E-05     0.12625E-05
                   14     0.26493E+05     0.11383E-05     0.11383E-05
                   15     0.31023E+05     0.97138E-06     0.97138E-06
-
 0                      turbulence parameter averaged over all space for update   1 is 0.57080E-05

                                                 type of fission is p239he
@@ -515,15 +514,15 @@
                                     table of total activity in each particle size class

       psize          fp           psize          fp           psize          fp           psize
↪   fp
-    2.4830E-03    6.1087E+09    2.8751E-04    7.5157E+09    1.0913E-04    9.0129E+09    3.7067E-05
↪   1.2379E+10
-    1.3111E-03    6.3763E+09    2.5163E-04    7.6715E+09    9.7031E-05    9.2679E+09    3.0753E-05
↪   1.3280E+10
-    8.9391E-04    6.5822E+09    2.2154E-04    7.8327E+09    8.6085E-05    9.5497E+09    2.4732E-05
↪   1.4506E+10
-    6.8190E-04    6.7546E+09    1.9591E-04    8.0010E+09    7.6126E-05    9.8644E+09    1.8866E-05
↪   1.6345E+10
-    5.4839E-04    6.9130E+09    1.7381E-04    8.1779E+09    6.7022E-05    1.0220E+10    1.2863E-05
↪   1.9716E+10
-    4.5499E-04    7.0648E+09    1.5454E-04    8.3653E+09    5.8659E-05    1.0629E+10    6.7923E-06
↪   2.8264E+10
-    3.8534E-04    7.2142E+09    1.3760E-04    8.5651E+09    5.0936E-05    1.1106E+10    0.0000E+00
↪   0.0000E+00
-    3.3110E-04    7.3638E+09    1.2257E-04    8.7799E+09    4.3767E-05    1.1677E+10    0.0000E+00
↪   0.0000E+00
-0   k factors computed from the fp table -   6.1020E+09                             2.3560E+03
+    2.4830E-03    6.0769E+09    2.8751E-04    7.4839E+09    1.0913E-04    8.9812E+09    3.7067E-05
↪   1.2347E+10
+    1.3111E-03    6.3445E+09    2.5163E-04    7.6397E+09    9.7031E-05    9.2362E+09    3.0753E-05
↪   1.3248E+10
+    8.9391E-04    6.5504E+09    2.2154E-04    7.8010E+09    8.6085E-05    9.5179E+09    2.4732E-05
↪   1.4474E+10
+    6.8190E-04    6.7229E+09    1.9591E-04    7.9693E+09    7.6126E-05    9.8326E+09    1.8866E-05
↪   1.6313E+10
+    5.4839E-04    6.8812E+09    1.7381E-04    8.1462E+09    6.7022E-05    1.0188E+10    1.2863E-05
↪   1.9684E+10
+    4.5499E-04    7.0330E+09    1.5454E-04    8.3336E+09    5.8659E-05    1.0597E+10    6.7923E-06
↪   2.8232E+10
+    3.8534E-04    7.1824E+09    1.3760E-04    8.5334E+09    5.0936E-05    1.1074E+10    0.0000E+00
↪   0.0000E+00
+    3.3110E-04    7.3321E+09    1.2257E-04    8.7482E+09    4.3767E-05    1.1645E+10    0.0000E+00
↪   0.0000E+00
+0   k factors computed from the fp table -   6.0830E+09                             2.3486E+03
 +                                                         (r-m**2)/(hr-kt)
↪   (r-mi**2)/(hr-kt)


@@ -541,116 +540,116 @@


```

```
                     1     1     1     1     1     1     2     1     1     0     0    -1     0
-          51000.   1.320 4.154 4.015 5.722 7.017 9.181 1.005 7.581 3.325 7.486 0.770 0.346 0.000
+          51000.   1.307 4.138 3.999 5.705 6.990 9.142 1.001 7.550 3.311 7.395 0.766 0.344 0.000

                     1     1     1     1     1     1     2     1     1     0     0    -1     0
-          49500.   1.148 2.395 5.814 6.141 7.272 9.714 1.048 7.682 3.214 6.607 0.607 0.235 0.000
+          49500.   1.142 2.377 5.799 6.120 7.241 9.673 1.042 7.650 3.200 6.579 0.605 0.234 0.000

                     0     1     1     1     1     2     2     1     1     0     0    -1     0
-          48000.   4.587 3.067 8.059 6.058 7.709 1.034 1.098 7.783 3.019 5.524 0.439 0.142 0.000
+          48000.   4.544 3.050 8.044 6.032 7.676 1.029 1.092 7.751 3.007 5.501 0.438 0.141 0.000

                     0     1     1     1     1     2     2     1     1     0     0     0     0
-          46500.   3.115 4.410 7.763 5.900 8.322 1.093 1.140 7.646 2.685 4.330 0.299 0.000 0.000
+          46500.   3.078 4.392 7.743 5.872 8.287 1.090 1.135 7.614 2.673 4.312 0.298 0.000 0.000

                     0     1     1     1     1     2     2     1     1     0     0     0     0
-          45000.   2.769 5.465 5.220 6.504 8.857 1.149 1.167 7.314 2.339 3.397 0.209 0.000 0.000
+          45000.   2.737 5.451 5.195 6.476 8.821 1.144 1.162 7.283 2.329 3.383 0.208 0.000 0.000

                     0     1     1     1     1     2     2     1     1     0     0     0     0
-          43500.   2.454 5.625 4.991 7.589 9.177 1.203 1.183 7.044 2.098 2.786 0.152 0.000 0.000
+          43500.   2.421 5.619 4.967 7.558 9.140 1.198 1.184 7.014 2.089 2.774 0.152 0.000 0.000

                     0     1     1     1     1     2     2     1     1     0     0     0     0
-          42000.   3.495 3.142 8.100 8.164 9.536 1.264 1.220 6.917 1.913 2.278 0.108 0.000 0.000
+          42000.   3.455 3.130 8.067 8.131 9.497 1.259 1.215 6.888 1.905 2.268 0.108 0.000 0.000

                     0     1     2     1     2     2     2     1     1     0    -1     0     0
-          40500.   3.328 2.292 1.177 7.906 1.017 1.329 1.258 6.729 1.684 1.757 0.715 0.000 0.000
+          40500.   3.307 2.281 1.172 7.875 1.013 1.323 1.252 6.700 1.677 1.749 0.712 0.000 0.000

                     0     1     1     1     2     2     2     1     1     0    -1     0     0
-          39000.   1.411 2.748 9.782 8.401 1.083 1.392 1.280 6.336 1.421 1.305 0.460 0.000 0.000
+          39000.   1.404 2.738 9.744 8.367 1.079 1.386 1.274 6.309 1.415 1.300 0.458 0.000 0.000

                     0     1     1     2     2     2     2     1     1     0    -1     0     0
-          37500.   0.629 3.930 6.424 1.015 1.128 1.454 1.282 5.887 1.201 0.988 0.305 0.000 0.000
+          37500.   0.626 3.915 6.399 1.011 1.123 1.448 1.277 5.861 1.196 0.983 0.304 0.000 0.000

                     0     1     1     2     2     2     2     1     1     0    -1     0     0
-          36000.   0.409 4.695 9.843 1.128 1.190 1.518 1.284 5.531 1.032 0.755 0.202 0.000 0.000
+          36000.   0.407 4.678 9.006 1.123 1.185 1.512 1.279 5.507 1.028 0.752 0.200 0.000 0.000

                     0     1     2     2     2     2     2     1     0     0    -1     0     0
-          34500.   0.447 1.975 1.453 1.091 1.295 1.579 1.289 5.175 8.690 0.556 0.124 0.000 0.000
+          34500.   0.445 1.967 1.447 1.087 1.289 1.573 1.283 5.153 8.652 0.554 0.124 0.000 0.000

                     0     1     2     2     2     2     2     1     0     0     0     0     0
-          33000.   0.420 1.160 1.193 1.232 1.387 1.636 1.276 4.714 7.065 0.396 0.000 0.000 0.000
+          33000.   0.418 1.155 1.194 1.227 1.382 1.629 1.271 4.693 7.033 0.394 0.000 0.000 0.000

                     0     1     1     2     2     2     2     1     0     0     0     0     0
-          31500.   0.187 1.572 7.644 1.535 1.464 1.685 1.245 4.229 5.702 0.283 0.000 0.000 0.000
+          31500.   0.186 1.566 7.613 1.529 1.458 1.677 1.239 4.210 5.676 0.282 0.000 0.000 0.000

                    -1     1     2     2     2     2     2     1     0     0     0     0     0
-          30000.   0.912 4.617 1.133 1.577 1.590 1.730 1.212 3.792 4.587 0.199 0.000 0.000 0.000
+          30000.   0.908 4.599 1.132 1.570 1.583 1.723 1.206 3.775 4.567 0.198 0.000 0.000 0.000
```

```

                 -1     1     2     2     2     2     2     1     0     0     0     0     0
-       28500.  0.719 2.351 1.610 1.633 1.722 1.781 1.172 3.333 3.568 0.134 0.000 0.000 0.000
+       28500.  0.716 2.342 1.603 1.626 1.715 1.773 1.167 3.323 3.552 0.133 0.000 0.000 0.000

                  0     0     1     2     2     2     2     1     0    -1     0     0     0
-       27000.  0.111 4.778 9.552 2.038 1.837 1.825 1.103 2.837 2.654 0.853 0.000 0.000 0.000
+       27000.  0.111 4.758 9.912 2.030 1.829 1.817 1.104 2.824 2.642 0.849 0.000 0.000 0.000

                 -1     1     1     2     2     2     2     1     0    -1     0     0     0
-       25500.  .548  1.272 8.754 2.098 2.013 1.846 1.025 2.333 1.887 0.507 0.000 0.000 0.000
+       25500.  0.545 1.267 8.717 2.090 2.004 1.838 1.020 2.322 1.879 0.505 0.000 0.000 0.000

                 -1     1     2     2     2     2     1     1     0    -1     0     0     0
-       24000.  0.157 5.435 1.553 2.125 2.177 1.857 9.249 1.834 1.245 0.273 0.000 0.000 0.000
+       24000.  0.156 5.414 1.546 2.116 2.167 1.849 9.207 1.825 1.239 0.272 0.000 0.000 0.000

                 -1     1     2     2     2     2     1     1     0    -1     0     0     0
-       22500.  0.132 1.326 1.368 2.538 2.337 1.849 8.029 1.338 0.735 0.123 0.000 0.000 0.000
+       22500.  0.131 1.321 1.363 2.527 2.327 1.841 7.993 1.332 0.731 0.123 0.000 0.000 0.000

                 -1     0     1     2     2     2     1     0     0     0     0     0     0
-       21000.  0.376 3.355 9.040 2.555 2.530 1.805 6.595 8.866 0.380 0.000 0.000 0.000 0.000
+       21000.  0.375 3.341 9.002 2.544 2.518 1.797 6.565 8.825 0.378 0.000 0.000 0.000 0.000

                 -1     1     2     2     2     2     1     0     0     0     0     0     0
-       19500.  0.122 5.777 1.673 2.790 2.669 1.710 5.028 5.177 0.166 0.000 0.000 0.000 0.000
+       19500.  0.121 5.753 1.668 2.778 2.657 1.702 5.005 5.153 0.165 0.000 0.000 0.000 0.000

                  0     1     2     2     2     2     1     0    -1     0     0     0     0
-       18000.  0.000 4.176 1.664 3.121 2.794 1.538 3.434 2.532 0.580 0.000 0.000 0.000 0.000
+       18000.  0.000 4.158 1.657 3.107 2.782 1.531 3.418 2.518 0.578 0.000 0.000 0.000 0.000

                 -1     0     2     2     2     2     1     0    -1     0     0     0     0
-       16500.  0.451 4.067 1.307 3.153 2.831 1.281 2.021 0.986 0.152 0.000 0.000 0.000 0.000
+       16500.  0.449 4.050 1.301 3.138 2.818 1.275 2.011 0.981 0.151 0.000 0.000 0.000 0.000

                 -1     1     2     2     2     1     0     0     0     0     0     0     0
-       15000.  0.636 6.755 2.140 3.748 2.779 9.533 9.844 0.296 0.000 0.000 0.000 0.000 0.000
+       15000.  0.633 6.724 2.131 3.731 2.766 9.488 9.798 0.295 0.000 0.000 0.000 0.000 0.000

                  0     0     2     2     2     1     0    -1     0     0     0     0     0
-       13500.  0.000 6.040 1.594 3.987 2.630 6.129 3.790 0.663 0.000 0.000 0.000 0.000 0.000
+       13500.  0.000 6.010 1.587 3.969 2.617 6.101 3.772 0.659 0.000 0.000 0.000 0.000 0.000

                  0     0     2     2     2     1     0    -1     0     0     0     0     0
-       12000.  0.000 4.497 3.915 5.064 2.422 3.481 1.155 0.107 0.000 0.000 0.000 0.000 0.000
+       12000.  0.000 4.477 3.898 5.040 2.410 3.465 1.149 0.106 0.000 0.000 0.000 0.000 0.000

                  0     0     2     2     2     1     0     0     0     0     0     0     0
-       10500.  0.000 0.829 4.564 5.440 2.113 1.818 0.289 0.000 0.000 0.000 0.000 0.000 0.000
+       10500.  0.000 0.825 4.543 5.415 2.103 1.809 0.288 0.000 0.000 0.000 0.000 0.000 0.000

                  0     0     2     2     2     0    -1     0     0     0     0     0     0
-        9000.  0.000 0.280 1.460 6.147 1.852 8.136 0.554 0.000 0.000 0.000 0.000 0.000 0.000
+        9000.  0.000 0.278 1.453 6.117 1.843 8.096 0.551 0.000 0.000 0.000 0.000 0.000 0.000

                  0    -1     1     3     2     0     0     0     0     0     0     0     0
-        7500.  0.000 0.953 3.303 1.011 2.043 4.067 0.000 0.000 0.000 0.000 0.000 0.000 0.000
```

```
229 +        7500.   0.000 0.949 3.291 1.006 2.033 4.047 0.000 0.000 0.000 0.000 0.000 0.000 0.000
230
231                    0    -1     0     3     2     0     0     0     0     0     0     0     0
232 -        6000.   0.000 0.244 3.824 1.302 2.729 6.967 0.000 0.000 0.000 0.000 0.000 0.000 0.000
233 +        6000.   0.000 0.243 3.806 1.296 2.716 6.933 0.000 0.000 0.000 0.000 0.000 0.000 0.000
234
235                    0     0     0     2     2     1    -1     0     0     0     0     0     0
236 -        4500.   0.000 0.000 0.257 6.969 4.120 1.740 0.618 0.000 0.000 0.000 0.000 0.000 0.000
237 +        4500.   0.000 0.000 0.256 6.935 4.099 1.731 0.615 0.000 0.000 0.000 0.000 0.000 0.000
238
239                    0     0    -1     2     2     1     0     0     0     0     0     0     0
240 -        3000.   0.000 0.000 0.168 6.549 5.489 2.540 0.215 0.000 0.000 0.000 0.000 0.000 0.000
241 +        3000.   0.000 0.000 0.167 6.515 5.462 2.528 0.214 0.000 0.000 0.000 0.000 0.000 0.000
242
243                    0     0     0     1     1     1     0     0     0     0     0     0     0
244 -        1500.   0.000 0.000 0.000 2.730 8.188 1.823 0.354 0.000 0.000 0.000 0.000 0.000 0.000
245 +        1500.   0.000 0.000 0.000 2.716 8.148 1.814 0.352 0.000 0.000 0.000 0.000 0.000 0.000
246
247                    0     0     0     0     0     0     0     0     0     0     0     0     0
248 -          0.    0.000 0.000 0.000 0.000 1.824 5.576 0.275 0.000 0.000 0.000 0.000 0.000 0.000
249 +          0.    0.000 0.000 0.000 0.000 1.814 5.548 0.274 0.000 0.000 0.000 0.000 0.000 0.000
250
251                    0     0     0     0    -1     0    -1     0     0     0     0     0     0
252 -       -1500.   0.000 0.000 0.000 0.000 0.150 0.541 0.956 0.000 0.000 0.000 0.000 0.000 0.000
253 +       -1500.   0.000 0.000 0.000 0.000 0.149 0.539 0.951 0.000 0.000 0.000 0.000 0.000 0.000
254
255      ***       -7200.      -3600.          0.      3600.      7200.      10800.      14400.
    ↪  18000.      21600.      25200. **
256
257
258
259 -              sum of map ordinates =  0.285508E+05
260 +              sum of map ordinates =  0.284241E+05
261   output processing is completed.
```

These differences include:

1. Listing 23, lines 7–8 and 70–71 (corresponding to Listing 22, lines 230 and 408, and Norment [3] pp. 42 and 46): processed atmospheric data row 14 (altitude $26.493\,\mathrm{km}$) VY column differs in the final decimal position, representing an absolute relative difference of 5×10^{-6}. This could be explained by differences in rounding mode (i.e., rounding up rather than down) or by different implementations of the cos function.

2. Listing 23, lines 16–62 (corresponding to Listing 22, lines 299–327, and Norment [3] p. 44): cloud trajectory table differs in the final 1–4 decimal places in the xc, yc, zc, and vc columns, representing a maximum absolute relative difference of 3×10^{-4}. This maximum difference occurs for the final nonzero velocity vc—which is on the order of $5.5 \times 10^{-3}\,\mathrm{m\,s^{-1}}$—whereas more typical values of the absolute relative difference are on the order of 10^{-6}. Thus, it is probably reasonable to say that the cloud trajectory differences are minor, albeit within the machine epsilon.

3. Listing 23, lines 87–104 (corresponding to Listing 22, lines 517–525, and Norment [3] p. 48): particle size distribution total activity table fp column differs by approximately $-3.2 \times 10^{7}\,\mathrm{R\,h^{-1}}$, representing a maximum absolute relative difference of 5×10^{-3}. This offset is approximately constant as a function of particle size and might thus be due to a difference in a numerical factor applied equally to all particle sizes within the OPM. That is, it is likely not a difference resulting from any differences in the DTM.

4. Listing 23, lines 112–260 (corresponding to Listing 22, lines 542–654, and Norment [3] pp. 49–50): the deposition contour map differs by a maximum absolute relative difference of 10^{-1}, with typical relative differences on the order of 5×10^{-3}. Similarly, the absolute relative difference in the sum of map ordinates is 4.4×10^{-3}. As in item 3, the output differences are likely due to activity calculation differences from the OPM rather than atmospheric transport or dispersion calculations of the DTM because the cloud trajectory and contour map shape is the same.

6.2.2 Stabilized Fallout Cloud

The parcels in the stabilized fallout cloud from the ICRM serve as the initial condition for the DTM, where the parcel concentration profiles are initialized such that the Gaussian standard deviations are half the parcel radius: $\sigma_X = \sigma_Y = r_{\text{parcel}}/2$, as mentioned in Section 3.2.2. Figure 12 shows the fallout cloud parcels and their mass concentration profiles integrated along each dimension at cloud stabilization time $t = t_0 + 469.0\,\text{s}$. Looking at the top left plot of the parcels, the smallest particles can be found in the upper region of the fallout cloud, while the larger particles have settled much faster and can be found near the base and center of the fallout cloud. The peak concentration of the stabilized cloud in this case is found at $\arg\max_{\mathbf{x}} C(\mathbf{x}, t) = \mathbf{x}_{\max} \approx (3.4\,\text{km}, 6.4\,\text{km}, 5.4\,\text{km})$ and is near the center-bottom of the cloud cap. In the bottom and right plots of Figure 12, the views integrated along each of the coordinate dimensions of the airborne mass concentration field are shown.

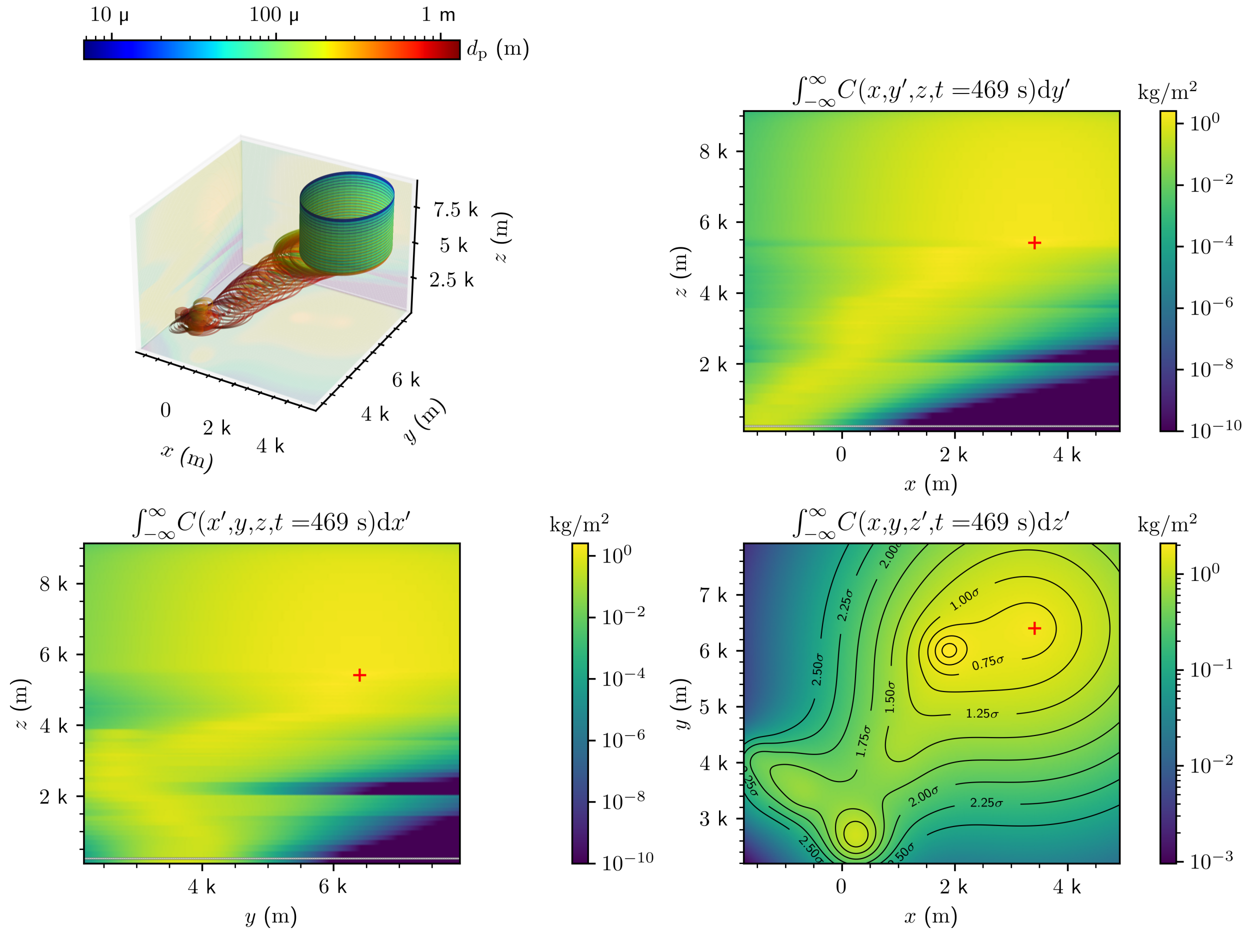

Figure 12. (*Top left*) Stabilized fallout cloud parcels and integrated mass concentration profiles integrated along the (*top right*) y, (*bottom left*) x, and (*bottom right*) z dimensions. Contours in the vertically integrated planar distribution are drawn in black at concentrations scaled by the standard normal distribution every quarter standard deviation, that is, the curves $\int_{-\infty}^{\infty} C(x,y,z',t)\mathrm{d}z' = \sqrt{2\pi}\varphi(\xi_i)\max_{(x,y)}\int_{-\infty}^{\infty} C(x,y,z',t)\mathrm{d}z'$ for $\xi_i \in \{n/4 : n \in \mathbb{N}^+\}$, where $\varphi(x) = e^{-x^2/2}/\sqrt{2\pi}$ is the standard normal probability density function. The red cross is located at the maximum, that is, $\arg\max_{\mathbf{x}} C(\mathbf{x},t)$, and the integrated distributions are also projected onto the 2D axes of the top left 3D plot. The diameters of the particles d_p in each parcel are also indicated by the heat map on the top left plot.

6.2.3 Fallout Parcel Deposit Increments

After the DTM has completed its transport calculation, it records the fallout parcel deposit increments on the ground level, as described in Section 4.2.2. These deposit increments are shown in Figures 13 and 14 for distances up to 10 km and 100 km, respectively, from ground zero (GZ) at time $t = t_0 + 48\,\mathrm{h}$. In total, 2.56×10^7 kg of 2.85×10^7 kg of condensed mass is deposited, resulting in a maximum deposited mass concentration of $\max_{(x,y)} C_\mathrm{dep}(x,y,t) = 0.68\,\mathrm{kg\,m^{-2}}$ at $\arg\max_{(x,y)} C_\mathrm{dep}(x,y,t) = (-1.1\,\mathrm{km}, 2.4\,\mathrm{km})$. The largest particles $700\,\mu\mathrm{m} \lesssim d_\mathrm{p} \lesssim 2.5\,\mathrm{mm}$ deposit within about 10 km, while the particle sizes $100\,\mu\mathrm{m} \lesssim d_\mathrm{p} \lesssim 700\,\mu\mathrm{m}$ deposit within 100 km. For typically sized and larger particles $d_\mathrm{p} \gtrsim 100\,\mu\mathrm{m}$, the parcels experience little horizontal dispersion, as described in Section 3.4.2, while smaller particles $d_\mathrm{p} \lesssim 100\,\mu\mathrm{m}$ settle sufficiently slowly that they will begin to undergo appreciable dispersion. Even smaller particles $d_\mathrm{p} \ll 100\,\mu\mathrm{m}$ deposit at ranges on the order of $1000\,\mathrm{km}$, but these are not shown since these particle sizes and ranges are somewhat beyond what DELFIC is suited to model, as discussed in Sections 3.2.1 and 3.5.3 and as shown in 4.2.3.3.

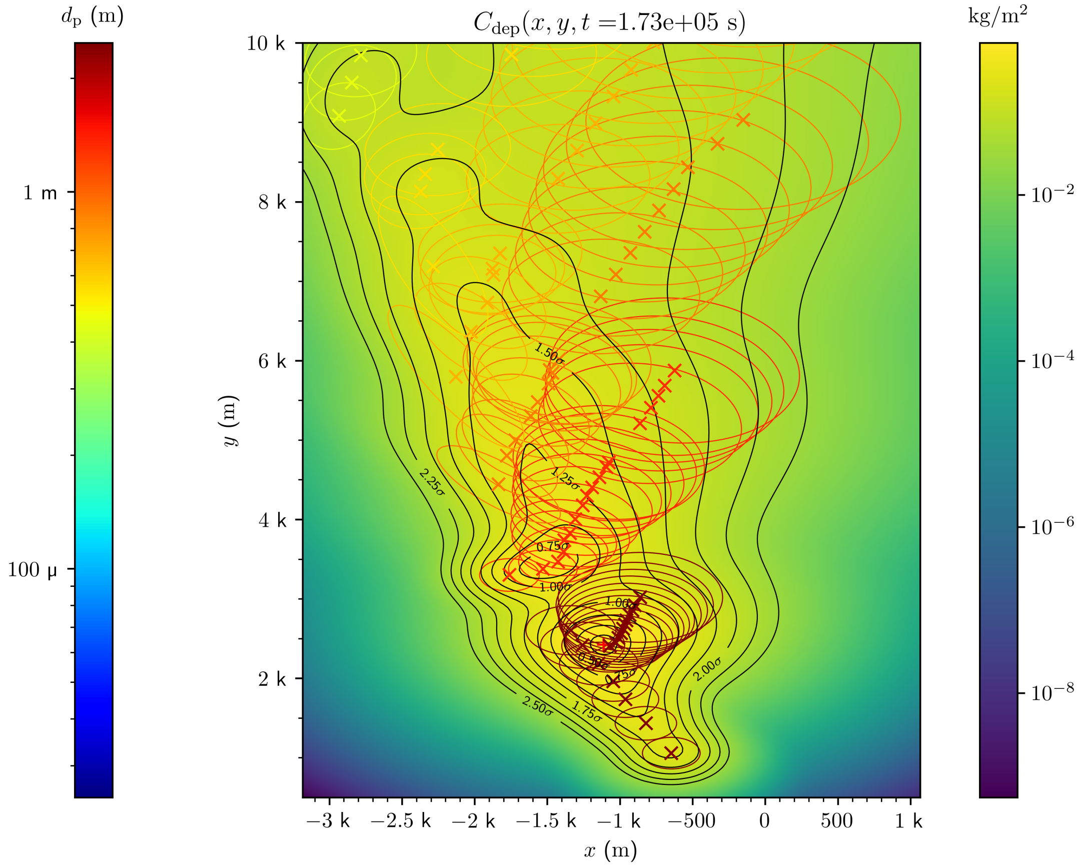

Figure 13. Fallout parcel deposit increments and the deposited mass concentration profile out to a distance of $10\,\mathrm{km}$**.** Contours are drawn in black at concentrations scaled by the standard normal distribution every quarter standard deviation, that is, the curves $C_\mathrm{dep}(x, y, t) = \sqrt{2\pi}\varphi(\xi_i) \max_{(x,y)} C_\mathrm{dep}(x, y, t)$ for $\xi_i \in \{n/4 : n \in \mathbb{N}^+\}$, where $\varphi(x) = e^{-x^2/2}/\sqrt{2\pi}$ is the standard normal probability density function. The red cross is located at the maximum, that is, $\arg\max_{(x,y)} C_\mathrm{dep}(x, y, t)$. The diameters of the particles d_p in each parcel deposit increment are also indicated by the heat map on the left.

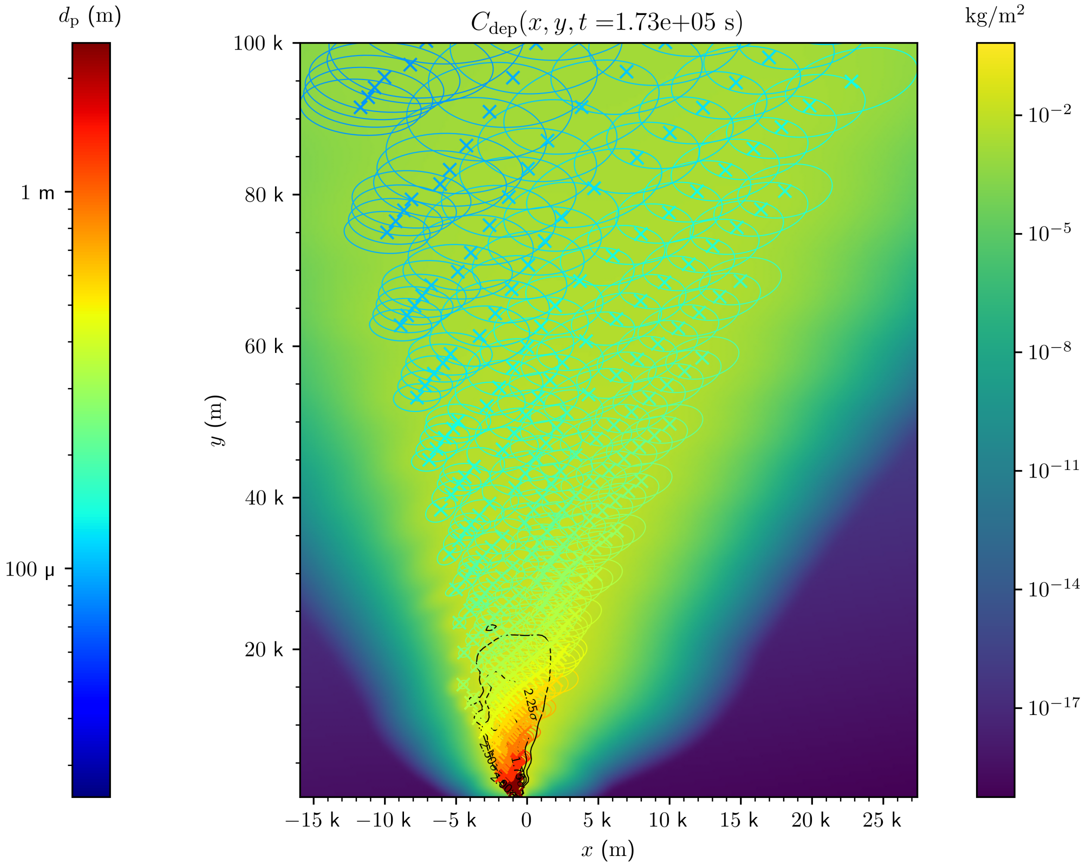

Figure 14. Fallout parcel deposit increments and the deposited mass concentration profile out to a distance of $100\,\mathrm{km}$**.** Contours are drawn in black at concentrations scaled by the standard normal distribution every quarter standard deviation, that is, the curves $C_\mathrm{dep}(x, y, t) = \sqrt{2\pi}\varphi(\xi_i) \max_{(x,y)} C_\mathrm{dep}(x, y, t)$ for $\xi_i \in \{n/4 : n \in \mathbb{N}^+\}$, where $\varphi(x) = e^{-x^2/2}/\sqrt{2\pi}$ is the standard normal probability density function. The red cross is located at the maximum, that is, $\arg\max_{(x,y)} C_\mathrm{dep}(x, y, t)$. The diameters of the particles d_p in each parcel deposit increment are also indicated by the heat map on the left.

6.2.4 Fallout Ground Exposure Rate Distribution

The fallout map exposure rate distribution and contours normalized to $t = t_0 + 1\,\mathrm{h}$ are shown in Figure 15. The maximum ground exposure rate is $\max_{(x,y)} \dot{X}(x, y, t) = 4.4 \times 10^3\,\mathrm{R\,h^{-1}}$ at $\arg\max_{(x,y)} \dot{X}(x, y, t) = (-1.1\,\mathrm{km}, 2.4\,\mathrm{km})$, indicated by the red plus. This closely reflects where the maximum mass concentration is found from Section 6.2.3, although in general the location with the highest mass concentration might not have the highest exposure rate because of the fractionation of the particle size distribution and the exposure rate multipliers of the different nuclides.

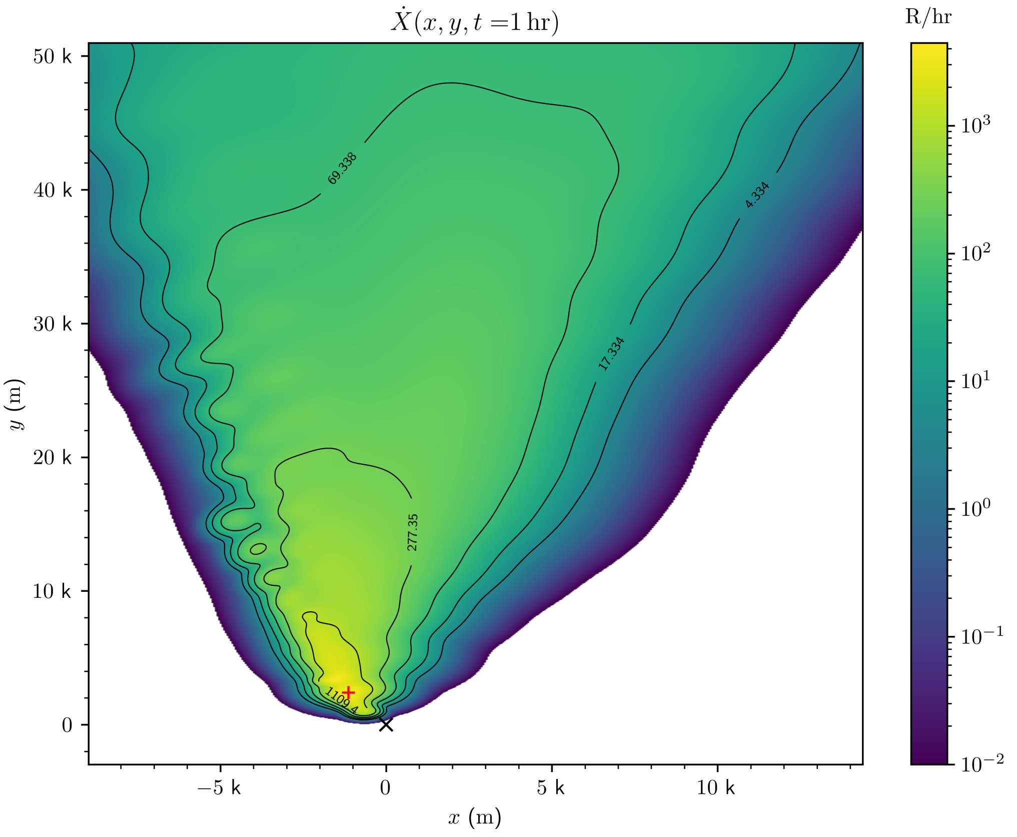

Figure 15. Ground exposure rate $(\mathrm{R\,h^{-1}})$ **distribution and contours normalized to** $t = t_0 + 1\,\mathrm{h}$**.** Contours are drawn in black at concentrations scaled by powers of $1/4$ of the maximum value, that is, $\dot{X}(x, y, t) = 4^{-\xi_i} \max_{(x,y)} \dot{X}(x, y, t)$ for $\xi_i \in \{1 \leq n \leq 5 : n \in \mathbb{N}^{+}\}$. Also drawn are the ground zero (*black cross*) and peak concentration (*red plus*) locations.

7. CONCLUSIONS

The DELFIC DTM is a hybrid Eulerian–Lagrangian transport model that can efficiently transport nuclear debris until its deposition on the ground. In particular, it uses a Lagrangian discrete parcel method in combination with a Gaussian puff model solution to the Eulerian advection-diffusion equation for atmospheric dispersion. Lagrangian transport considers both drag and gravitational forces on solid, spherical fallout particles with constant mass and uses an analytical steady-state solution. Together with the Gaussian puff model solution using time-dependent dispersion parameters, this leads to a rapid and efficient transport solution for fallout particles over a wide range of sizes.

Before the introduction of the DTM in 1975, DELFIC used a simple semi-empirical model of atmospheric dispersion in the horizontal domain. When the DTM was introduced, it improved the horizontal dispersion formulation and added a vertical diffusion solver using an implicit finite difference method. However, by the 1979 version of DELFIC, the vertical diffusion solver of the DTM was removed. As discussed in Sections 3.2.1, 3.5.3, and 4.2.3.3, the lack of a vertical diffusion component might be significant for very small particles of diameters $d_\mathrm{p} \lesssim 10\,\mu\mathrm{m}$, but the vertical advection and gravitational settling of larger particles is usually more important for local deposition. For intermediately sized particles $d_\mathrm{p} \sim 100\,\mu\mathrm{m}$, the hybrid Eulerian–Lagrangian approach may break down since the Eulerian advection-diffusion model does not strictly apply and might overestimate the dispersion of such particles with longer response time scales.

Care must be used when handling heterogeneous meteorological data, in particular by defining the meteorological spatial cells of sufficiently large size. This not only allows the particles to reach steady state within the cells, as required by Section 2.2.2.3, but is also required to prevent the parcels experiencing different conditions throughout their spatial extent, as mentioned in Sections 3.2.1 and 3.4.1. Over long dispersion times and ranges, this issue becomes more significant because of the increasing parcel dispersion parameters of Section 3.2.2. Furthermore, since DELFIC does not account for variations in terrain, it is not well suited for handling urban or mountainous environments, as mentioned in Section 3.1.1. For local fallout this might not be problematic, whereas long-range simulations can produce inaccurate results, especially if not accounting for heterogeneous, time-dependent meteorological conditions. Additionally, as discussed in Section 2.1.2, long-range fallout might be affected by the Coriolis force, which is unaccounted for by DELFIC.

Fallout parcels have a mass distribution that is a bivariate Gaussian in the horizontal dimensions and a uniform in the vertical dimension and can be described using a correlated, joint PDF formulation. The bivariate Gaussian distribution of the parcel has a one-sigma curve that is an ellipse with a rotation angle determined by the wind shear and the dispersion parameters of Section 3.2.2. DELFIC uses a geometric approximation described in Section 4.2.2 that combines the ellipses impacting the ground from the bottom and top of the parcel to model the deposition of a fallout parcel in the absence of vertical diffusion. As shown in Section 4.2.3.1, this approximation is exact in the limit that the parcel vertical span is very small when compared with the vertical distance it travels, that is, a Dirac-delta distribution. Since DELFIC divides up the fallout cloud into many parcels of relatively small vertical extent, this is a reasonable approximation.

Using the input test case published in the 1979 documentation with the 1979 version of the code [3], we were able to mostly reproduce the same results as the 1979 printed output. Specifically, there were some minor differences in rounding or elementary mathematical function implementations resulting in minor relative differences, as well as some minor differences in the total activity or exposure rate in the OPM. This demonstrates and validates that the DTM is behaving as expected according to the 1979 implementation. We demonstrated that the local fallout field predominately consists of large and intermediately sized particles where $d_\mathrm{p} \gtrsim 100\,\mu\mathrm{m}$, which undergo relatively little dispersion. Furthermore, the mass deposition concentration field closely reflects the normalized exposure rate field.

Several potential areas exist for future work on the DTM. Intermediately sized particles could be better handled by adjusting the timing of the dispersion parameters to avoid dispersion on their time scales. For long-range transport, vertical diffusion should be incorporated for handling small particles, and heterogeneous, time-dependent meteorological data should be used. Although the steady-state solution within constant meteorological cells is efficient, it is somewhat limiting and imprecisely reflects the spatiotemporal variation of real meteorological data. Thus, it might be worthwhile to consider explicitly integrating the Lagrangian particle trajectory using spatiotemporal interpolation. For short-range transport, however, the existing DELFIC DTM provides a reasonable and rapid approximation to the overall transport and deposition of nuclear fallout debris.

8. REFERENCES

[1] National Oceanic Atmospheric Administration, National Aeronautics Space Administration, and United States Air Force. 1976. *US Standard Atmosphere, 1976*. Tech. Rep. NOAA–S/T 76-1562.

[2] Norment, H. G. 1979. *DELFIC: Department of Defense Fallout Prediction System. Volume I - Fundamentals*. Tech. Rep. DNA 5159F-1, Atmospheric Science Associates.

[3] Norment, H. G. 1979. *DELFIC: Department of Defense Fallout Prediction System. Volume II - User's Manual*. Tech. Rep. DNA 5159F-2, Atmospheric Science Associates.

[4] van Wachem, B. G. M. and A.-E. Almstedt. 2003. "Methods for multiphase computational fluid dynamics." *Chemical Engineering Journal* 96(1-3): 81–98.

[5] Crowe, C. T., J. D. Schwarzkopf, M. Sommerfeld, and Y. Tsuji. 2011. *Multiphase Flows with Droplets and Particles*. CRC press, 2 ed.

[6] Michaelides, E. E., C. T. Crowe, and J. D. Schwarzkopf. 2016. *Multiphase Flow Handbook*. CRC Press, 2 ed.

[7] Podgórska, W. 2019. *Multiphase Particulate Systems in Turbulent Flows: Fluid-liquid and Solid-liquid Dispersions*. CRC Press.

[8] Norment, H. G. and E. J. Tichovolsky. 1971. *A New Fallout Transport Code for the DELFIC System: The Diffusive Transport Module*. Tech. Rep. DASA 2669, Arcon Corporation.

[9] Norment, H. G. 1972. *A New Fallout Transport Code for the DELFIC System: The Diffusive Transport Module. Supplement*. Tech. Rep. DASA 2669 SU, Mount Auburn Research Associates Inc.

[10] Schwenke, T. W., I. Kohlberg, H. G. Norment, et al. 1967. *Department of Defense Land Fallout Prediction System. Volume IV - Atmospheric Transport*. Tech. Rep. DASA-1800-IV, Technical Operations Research.

[11] Jakobsen, H. A. 2014. *Chemical Reactor Modeling: Multiphase Reactive Flows*. Springer, 2 ed.

[12] Sommerfeld, A. 1964. *Lectures on Theoretical Physics: Mechanics*, vol. 1. Academic Press.

[13] Plastino, A. R. and J. C. Muzzio. 1992. "On the use and abuse of Newton's second law for variable mass problems." *Celestial Mechanics and Dynamical Astronomy* 53(3): 227–232.

[14] Özsoy, E. 2020. *Geophysical Fluid Dynamics I*. Springer Textbooks in Earth Sciences, Geography and Environment, Springer.

[15] Maxey, M. R. and J. J. Riley. 1983. "Equation of motion for a small rigid sphere in a nonuniform flow." *The Physics of Fluids* 26(4): 883–889.

[16] Zhou, L. 2018. *Theory and Modeling of Dispersed Multiphase Turbulent Reacting Flows*. Butterworth-Heinemann.

[17] Glasstone, S. and P. J. Dolan. 1977. *The Effects of Nuclear Weapons*. US Department of Defense and US Department of Energy, 3 ed.

[18] Vallis, G. K. 2017. *Atmospheric and Oceanic Fluid Dynamics*. Cambridge University Press, 2 ed.

[19] Seinfeld, J. H. and S. N. Pandis. 2016. *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change*. John Wiley & Sons, 3 ed.

[20] Holton, J. R. and G. J. Hakim. 2013. *An Introduction to Dynamic Meteorology*. Academic Press, 5 ed.

[21] Prosperetti, A. and G. Tryggvason. 2009. *Computational Methods for Multiphase Flow*. Cambridge University Press.

[22] Balachandar, S. and J. K. Eaton. 2010. "Turbulent dispersed multiphase flow." *Annual Review of Fluid Mechanics* 42: 111–133.

[23] Auton, T. R., J. C. R. Hunt, and M. Prud'Homme. 1988. "The force exerted on a body in inviscid unsteady non-uniform rotational flow." *Journal of Fluid Mechanics* 197: 241–257.

[24] Yeoh, G. H. and J. Tu. 2019. *Computational Techniques for Multiphase Flows*. Butterworth-Heinemann.

[25] Ferry, J. and S. Balachandar. 2001. "A fast Eulerian method for disperse two-phase flow." *International Journal of Multiphase Flow* 27(7): 1199–1226.

[26] Ferry, J. and S. Balachandar. 2002. "Equilibrium expansion for the Eulerian velocity of small particles." *Powder Technology* 125(2-3): 131–139.

[27] Balachandar, S. 2009. "A scaling analysis for point–particle approaches to turbulent multiphase flows." *International Journal of Multiphase Flow* 35(9): 801–810.

[28] Stull, R. B. 2012. *An Introduction to Boundary Layer Meteorology*, vol. 13. Springer Science & Business Media.

[29] Struchtrup, H. 2005. *Macroscopic Transport Equations for Rarefied Gas Flows: Approximation Methods In Kinetic Theory*. Springer.

[30] Cercignani, C. 1988. *The Boltzmann Equation and its Applications*. Applied Mathematical Sciences, Springer.

[31] Abrahamson, J. 1975. "Collision rates of small particles in a vigorously turbulent fluid." *Chemical Engineering Science* 30(11): 1371–1379.

[32] Meyer, C. J. and D. A. Deglon. 2011. "Particle collision modeling–a review." *Minerals Engineering* 24(8): 719–730.

[33] Norment, H. G., T. W. Schwenke, I. Kohlberg, et al. 1966. *Development of an Improved Land-Surface Fallout Model. Interim Report*. Tech. Rep. TO-B 65-99, Technical Operations Research.

[34] Ray, S. and J. Fröhlich. 2015. "An analytic solution to the equations of the motion of a point mass with quadratic resistance and generalizations." *Archive of Applied Mechanics* 85(4): 395–414.

[35] Cavcar, M. 2000. "The international standard atmosphere (ISA)." *Anadolu University, Turkey* 30(9): 1–6.

[36] Caughey, S. J. 1984. "Observed characteristics of the atmospheric boundary layer." In "Atmospheric Turbulence and Air Pollution Modelling," p. 107–158, Springer.

[37] Pasquill, F. and F. B. Smith. 1983. *Atmospheric Diffusion*. Ellis Horwood Series in Environmental Science, E. Horwood, 3 ed.

[38] Monin, A. S. and A. M. Yaglom. 1971. *Statistical Fluid Mechanics: Mechanics of Turbulence*, vol. 1. MIT Press.

[39] Pope, S. B. 2000. *Turbulent Flows*. Cambridge University Press.

[40] Monin, A. S. and A. M. Yaglom. 1975. *Statistical Fluid Mechanics: Mechanics of Turbulence*, vol. 2. MIT Press.

[41] Udina, M., À. Montornès, P. Casso, B. Kosović, and J. Bech. 2020. "WRF-LES simulation of the boundary layer turbulent processes during the BLLAST campaign." *Atmosphere* 11(11): 1149.

[42] Lundquist, J. K., M. Piper, and B. Kosovic. 2004. "Turbulence kinetic energy budgets and dissipation rates in disturbed stable boundary layers." 16th Symposium on Boundary Layers and Turbulence, American Meteorological Society.

[43] Muñoz-Esparza, D., R. D. Sharman, and J. K. Lundquist. 2018. "Turbulence dissipation rate in the atmospheric boundary layer: Observations and WRF mesoscale modeling during the XPIA field campaign." *Monthly Weather Review* 146(1): 351–371.

[44] Cohn, S. A. 1995. "Radar measurements of turbulent eddy dissipation rate in the troposphere: A comparison of techniques." *Journal of Atmospheric and Oceanic Technology* 12(1): 85–95.

[45] Hocking, W. K. and P. K. L. Mu. 1997. "Upper and middle tropospheric kinetic energy dissipation rates from measurements of Cn2—review of theories, in-situ investigations, and experimental studies using the Buckland Park atmospheric radar in Australia." *Journal of Atmospheric and Solar-Terrestrial Physics* 59(14): 1779–1803.

[46] Satheesan, K. and B. V. Krishna Murthy. 2002. "Turbulence parameters in the tropical troposphere and lower stratosphere." *Journal of Geophysical Research: Atmospheres* 107(D1): ACL–2.

[47] Kohma, M., K. Sato, Y. Tomikawa, K. Nishimura, and T. Sato. 2019. "Estimate of turbulent energy dissipation rate from the VHF radar and radiosonde observations in the Antarctic." *Journal of Geophysical Research: Atmospheres* 124(6): 2976–2993.

[48] Wilkins, E. M. 1963. "Decay rates for turbulent energy throughout the atmosphere." *Journal of the Atmospheric Sciences* 20(5): 473–476.

[49] Monin, A. S. and A. M. Obukhov. 1954. "Basic laws of turbulent mixing in the surface layer of the atmosphere." *Contrib. Geophys. Inst. Acad. Sci. USSR* 151: 163–187.

[50] Businger, J. A., J. C. Wyngaard, Y. Izumi, and E. F. Bradley. 1971. "Flux-profile relationships in the atmospheric surface layer." *Journal of the Atmospheric Sciences* 28(2): 181–189.

[51] Barker, E. H. and T. L. Baxter. 01 Jun. 1975. "A note on the computation of atmospheric surface layer fluxes for use in numerical modeling." *Journal of Applied Meteorology and Climatology* 14(4): 620–622.

[52] McRae, G. J., W. R. Goodin, and J. H. Seinfeld. 1982. "Development of a second-generation mathematical model for urban air pollution—I. Model formulation." *Atmospheric Environment (1967)* 16(4): 679–696.

[53] Albertson, J. D., M. B. Parlange, G. Kiely, and W. E. Eichinger. 1997. "The average dissipation rate of turbulent kinetic energy in the neutral and unstable atmospheric surface layer." *Journal of Geophysical Research: Atmospheres* 102(D12): 13423–13432.

[54] Maxey, M. R. 1987. "The gravitational settling of aerosol particles in homogeneous turbulence and random flow fields." *Journal of Fluid Mechanics* 174: 441–465.

[55] Walton, J. J. 1973. "Scale-dependent diffusion." *Journal of Applied Meteorology (1962-1982)* 12(3): 547–549.

[56] Hernandez-Coronado, H., M. Coronado, and D. D. Castillo-Negrete. 2017. "On the anisotropic advection-diffusion equation with time dependent coefficients." *Revista mexicana de física* 63(1): 40–48.

[57] Obukhov, A. M. 1941. "On the distribution of energy in the spectrum of turbulent flow." *Dokl. Akad. Nauk SSSR* 32: 22–24.

[58] Obukhov, A. M. 1941. "Spectral energy distribution in a turbulent flow." *Izvestiya. Akad. Nauk SSSR. Ser. Geogr. i. Geofiz* 4-5: 453–466.

[59] Sreenivasan, K. R. 1995. "On the universality of the Kolmogorov constant." *Physics of Fluids* 7(11): 2778–2784.

[60] Kolmogorov, A. N. 1941. "The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers." *Dokl. Akad. Nauk SSSR* 30(4): 299–303.

[61] Dosio, A., J. V. Guerau de Arellano, A. A. Holtslag, and P. J. Builtjes. 2005. "Relating Eulerian and Lagrangian statistics for the turbulent dispersion in the atmospheric convective boundary layer." *Journal of the Atmospheric Sciences* 62(4): 1175–1191.

[62] Richardson, L. F. 1926. "Atmospheric diffusion shown on a distance-neighbour graph." *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 110(756): 709–737.

[63] Taylor, G. I. 1959. "The present position in the theory of turbulent diffusion." In "Advances in Geophysics," vol. 6, p. 101–112, Elsevier.

[64] Lin, C.-C. 1960. "On a theory of dispersion by continuous movements." *Proceedings of the National Academy of Sciences of the United States of America* 46(4): 566.

[65] Lin, C.-C. 1960. "On a theory of dispersion by continuous movements, II. Stationary anisotropic processes." *Proceedings of the National Academy of Sciences of the United States of America* 46(8): 1147.

[66] Batchelor, G. K. 1950. "The application of the similarity theory of turbulence to atmospheric diffusion." *Quarterly Journal of the Royal Meteorological Society* 76(328): 133–146.

[67] George, W. K. 2013. "Lectures in turbulence for the 21st century." *Chalmers University of Technology* .

[68] Hanna, S. R. 1981. "Lagrangian and Eulerian time-scale relations in the daytime boundary layer." *Journal of Applied Meteorology and Climatology* 20(3): 242–249.

[69] Davis, P. 1983. "Markov chain simulations of vertical dispersion from elevated sources into the neutral planetary boundary layer." *Boundary-layer meteorology* 26(4): 355–376.

[70] Marble, F. E. 1970. "Dynamics of dusty gases." *Annual Review of Fluid Mechanics* 2(1): 397–446.

[71] Kong, B. and R. O. Fox. 2017. "A solution algorithm for fluid–particle flows across all flow regimes." *Journal of Computational Physics* 344: 575–594.

[72] Cressman, G. P. 1959. "An operational objective analysis system." *Monthly Weather Review* 87(10): 367–374.

[73] Monin, A. S. 1959. "On the boundary condition on the earth surface for diffusing pollution." In "Advances in Geophysics," vol. 6, p. 435–436, Elsevier.

[74] Holmes, M. H. 2007. *Introduction to Numerical Methods in Differential Equations*. Texts in Applied Mathematics, Springer.

[75] Strikwerda, J. C. 2004. *Finite Difference Schemes and Partial Differential Equations*. SIAM, 2 ed.

[76] LeVeque, R. J. 2007. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*. SIAM.

[77] Samarskii, A. A. 2001. *The Theory of Difference Schemes*. Monographs and textbooks in pure and applied mathematics, CRC Press.

[78] Samarskii, A. A., P. P. Matus, and P. N. Vabishchevich. 2002. *Difference Schemes with Operator Factors*. Mathematics and Its Applications, Springer.

[79] Jovanović, B. S. and E. Süli. 2013. *Analysis of Finite Difference Schemes: For Linear Partial Differential Equations with Generalized Solutions*, *Springer Series in Computational Mathematics*, vol. 46. Springer.

[80] Quarteroni, A. 2017. *Numerical Models for Differential Problems*, *Modeling, Simulation and Applications*, vol. 16. Springer, 3 ed.

[81] Davies, C. N. 1945. "Definitive equations for the fluid resistance of spheres." *Proceedings of the Physical Society* 57(4): 259.

[82] Beard, K. V. 1976. "Terminal velocity and shape of cloud and precipitation drops aloft." *Journal of Atmospheric Sciences* 33(5): 851–864.

[83] Jennings, S. G. 1988. "The mean free path in air." *Journal of Aerosol Science* 19(2): 159–166.

[84] American National Standards Institute. 1978. *USA Standard FORTRAN*. Standard, 1430 Broadway, New York, New York.

[85] "GNU Fortran - GNU Project - Free Software Foundation (FSF)." https://gcc.gnu.org/fortran/.

[86] American National Standards Institute. 1978. *FORTRAN 77 Full Language*. Standard.

[87] International Organization for Standardization and International Electrotechnical Commission. 1991. *International standard: information, technology, programming languages, Fortran*. Standard.

[88] Kundu, P. K., I. M. Cohen, and D. R. Dowling. 2016. *Fluid Mechanics*. Academic Press, 6 ed.

[89] Landau, L. D. and E. M. Lifshitz. 1987. *Fluid Mechanics*, *Course of Theoretical Physics*, vol. 6. Butterworth-Heinemann, 2 ed.

[90] Batchelor, G. K. 2000. *An Introduction to Fluid Dynamics*. Cambridge University Press.

[91] Proudman, I. and J. R. A. Pearson. 1957. "Expansions at small Reynolds numbers for the flow past a sphere and a circular cylinder." *Journal of Fluid Mechanics* 2(3): 237–262.

[92] Keh, H. J. and S. C. Shiau. 2000. "Effects of inertia on the slow motion of aerosol particles." *Chemical engineering science* 55(20): 4415–4421.

[93] Feng, Z.-G., E. E. Michaelides, and S. Mao. 2012. "On the drag force of a viscous sphere with interfacial slip at small but finite Reynolds numbers." *Fluid Dynamics Research* 44(2): 025502.

[94] Michaelides, E. E. 2014. *Nanofluidics: Thermodynamic and Transport Properties*. Springer.

[95] Schiller, L. and A. Nauman. 1933. “Über die grundlegenden Berechnungen bei der Schwerkraftaufbereitung.” *Verein Deutscher Ingenieure* 44: 318–321.

[96] Clift, R. and W. H. Gauvin. 1971. “Motion of entrained particles in gas streams.” *The Canadian Journal of Chemical Engineering* 49(4): 439–448.

[97] Karamanev, D. G. 1996. “Equations for calculation of the terminal velocity and drag coefficient of solid spheres and gas bubbles.” *Chemical engineering communications* 147(1): 75–84.

[98] Camenen, B. 2007. “Simple and general formula for the settling velocity of particles.” *Journal of Hydraulic Engineering* 133(2): 229–233.

[99] Shen, C. 2005. *Rarefied Gas Dynamics: Fundamentals, Simulations and Micro Flows*. Heat and Mass Transfer, Springer.

[100] Gombosi, T. I. 1994. *Gaskinetic theory*. Cambridge University Press.

APPENDIX A. CAUCHY STRESS TENSOR

APPENDIX A. CAUCHY STRESS TENSOR

The Cauchy stress tensor is a second-order tensor that defines the surface stresses at any point within the continuum fluid. By employing a constitutive relation, we can relate the stress tensor to the fluid deformation. In particular, we consider a linear constitutive relation for a Newtonian fluid, which the air in the atmosphere behaves like to a close approximation [19, 28]. The stress tensor can be divided into an isotropic component and an anisotropic component [88]:

$$\sigma_{ij} = -p\delta_{ij} + \tau_{ij}, \tag{194}$$

where p is the thermodynamic pressure, δ_{ij} is the Kronecker delta tensor, and $\boldsymbol{\tau}$ is the deviatoric or viscous stress tensor. The pressure is generally defined as negative one-third the trace of the stress tensor plus stress due to the divergence of the velocity, $p = \overline{p} + \zeta\nabla \cdot \mathbf{u}$, where $\overline{p} \equiv -\mathrm{tr}(\boldsymbol{\sigma})/3 = -\sigma_{ii}/3$ is the mean hydrostatic stress or the mechanical pressure, and $\zeta = \lambda + 2\mu/3$ is a proportionality constant called the bulk viscosity or second viscosity, which depends on two scalar Lamé parameters—the first parameter λ and the second parameter or dynamic viscosity μ. The second viscosity ζ models the difference between the thermodynamic and mechanical pressures that can arise due to rapid expansion or compression when compared with the thermodynamic equilibrium relaxation time [89].

For a Newtonian fluid, the relationship between the viscous stress τ_{ij} and $\partial u_i/\partial x_j$ is linear [90]:

$$\tau_{ij} = A_{ijk\ell}\frac{\partial u_k}{\partial x_\ell}. \tag{195}$$

The velocity gradients $\partial u_i/\partial x_j$ can be decomposed into symmetric and antisymmetric parts:

$$\frac{\partial u_i}{\partial x_j} = e_{ij} + r_{ij}, \tag{196}$$

where

$$e_{ij} \equiv \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \tag{197}$$

$$r_{ij} \equiv \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right) \tag{198}$$

are the rate of strain and rate of rotation tensors, respectively. The antisymmetric part represents fluid rotation without deformation and cannot by itself generate stress, so stresses must be generated by the strain rate tensor e_{ij}. Assuming a linearly elastic and isotropic fluid with a symmetric stress tensor, one can show that Eq. 195 reduces to a linear function of two coefficients [88, 90]:

$$\tau_{ij} = \lambda\Delta\delta_{ij} + 2\mu e_{ij}, \tag{199}$$

where $\Delta \equiv e_{ii} = \partial u_i/\partial x_i = \nabla \cdot \mathbf{u}$ is the volumetric strain rate or expansion.

Substituting Eq. 199 into Eq. 194, the Cauchy stress tensor for a Newtonian fluid is [89, 90]

$$\sigma_{ij} = -\left(p - \lambda\Delta\right)\delta_{ij} + 2\mu e_{ij} \tag{200a}$$

$$= -\left(p - \zeta\Delta\right)\delta_{ij} + 2\mu\left(e_{ij} - \frac{1}{3}\Delta\delta_{ij}\right), \tag{200b}$$

so the momentum change due to the stress tensor is [89]

$$\frac{\partial \sigma_{ij}}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_i}(\zeta \Delta) + \frac{\partial}{\partial x_j}\left[\mu\left(2e_{ij} - \frac{2}{3}\Delta\delta_{ij}\right)\right]. \tag{201}$$

For many applications the Stokes assumption $\zeta = 0 \Rightarrow \lambda = -2\mu/3$ is sufficiently accurate.* With this assumption, the viscous stress tensor Eq. 199 and Cauchy stress tensor Eqs. 200, respectively, can be written in terms of the dynamic viscosity alone [88, 90]:

$$\tau_{ij} = -\frac{2}{3}\mu\Delta\delta_{ij} + 2\mu e_{ij}, \tag{202}$$

$$\sigma_{ij} = -p\delta_{ij} + 2\mu\left(e_{ij} - \frac{1}{3}\Delta\delta_{ij}\right). \tag{203}$$

Thus, the momentum change due to the stress tensor with the Stokes assumption is [88, 90]

$$\frac{\partial \sigma_{ij}}{\partial x_j} = -\frac{\partial p}{\partial x_i} - \frac{2}{3}\frac{\partial}{\partial x_i}(\mu\Delta) + \frac{\partial}{\partial x_j}(2\mu e_{ij}). \tag{204}$$

The viscosity is generally a function of pressure and temperature and thus varies spatially in the fluid. In most cases, however, the viscosity does not noticeably change in the fluid and can be regarded as constant [28, 88, 89] so that Eq. 204 becomes

$$\begin{aligned}\frac{\partial \sigma_{ij}}{\partial x_j} &= -\frac{\partial p}{\partial x_i} - \frac{2}{3}\mu\frac{\partial \Delta}{\partial x_i} + \mu\frac{\partial}{\partial x_j}(2e_{ij}) \\ &= -\frac{\partial p}{\partial x_i} - \frac{2}{3}\mu\frac{\partial \Delta}{\partial x_i} + \mu\left(\frac{\partial u_i^2}{\partial x_j \partial x_j} + \frac{\partial^2 u_j}{\partial x_i \partial x_j}\right) \\ &= -\frac{\partial p}{\partial x_i} + \mu\left(\nabla^2 u_i + \frac{1}{3}\frac{\partial \Delta}{\partial x_i}\right),\end{aligned} \tag{205}$$

where $\nabla^2 u_i \equiv \partial^2 u_i / \partial x_j \partial x_j$.

The final simplification is to assume the fluid is incompressible: $\Delta = 0$. In this case, the viscous stress tensor Eq. 202 and Cauchy stress tensor Eqs. 203 (or Eq. 199 and Eqs. 200), respectively, simplify to

$$\tau_{ij} = 2\mu e_{ij}, \tag{206}$$

$$\sigma_{ij} = -p\delta_{ij} + 2\mu e_{ij}, \tag{207}$$

while the momentum change Eq. 205 (or Eqs. 201 or 204 with constant viscosity μ) simplifies to [88, 90]

$$\frac{\partial \sigma_{ij}}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \mu\nabla^2 u_i. \tag{208}$$

[*] Exceptions to this include, for example, damping of high-frequency sound waves and the structure of shock waves [90].

APPENDIX B. STEADY-STATE DRAG AND TERMINAL VELOCITY

APPENDIX B. STEADY-STATE DRAG AND TERMINAL VELOCITY

For a solid spherical particle, the drag coefficient C_D and drag factor f_drag discussed in Section 2.1.3.1 will in general depend on the Reynolds number $\mathrm{Re_r}$ [5]. However, for Stokes flow, $\mathrm{Re_r} < 0.1$, and $f_\mathrm{drag} \approx 1$ [7]. Using analytical, asymptotic, or perturbation methods for solving the Navier-Stokes equation for low Reynolds numbers, one can show [6, 7, 91]

$$f_\mathrm{drag} = 1 + \frac{3}{16}\mathrm{Re_r} + \frac{9}{160}\mathrm{Re_r^2}\ln\left(\frac{\mathrm{Re_r}}{2}\right) + O\left(\mathrm{Re_r^2}\right). \quad \mathrm{Re_r} < 1.5, \mathrm{Kn} < 0.01 \tag{209}$$

Note that Eq. 209 is valid for continuum flow regimes, that is, those for which $\mathrm{Kn} = \lambda/d_\mathrm{p} < 0.01 \Rightarrow d_\mathrm{p} > 100\lambda$, where λ is the mean free path of molecules in the surrounding continuum fluid. For dry air below about $10\,\mathrm{km}$, $\lambda \sim 0.1\,\mathrm{\mu m}$ [1], so Eq. 209 requires $d_\mathrm{p} \gtrsim 10\,\mathrm{\mu m}$. Smaller particles $d_\mathrm{p} \lesssim 1\,\mathrm{\mu m}$ will depart from the continuum regime, as discussed in Appendix D, and will experience interfacial slip, resulting in a reduction in the effective drag force and a greater terminal velocity. Thus, in this case, one must use the more general drag factor [92, 93, 94]

$$f_\mathrm{drag} = \frac{1+2\mathrm{Sp}}{1+3\mathrm{Sp}} + \frac{3}{16}\left(\frac{1+2\mathrm{Sp}}{1+3\mathrm{Sp}}\right)^2\mathrm{Re_r} + \frac{9}{160}\left(\frac{1+2\mathrm{Sp}}{1+3\mathrm{Sp}}\right)^3\mathrm{Re_r^2}\ln\left(\frac{\mathrm{Re_r}}{2}\right) + O\left(\mathrm{Re_r^2}\right), \quad \mathrm{Re_r} < 1 \tag{210}$$

where $\mathrm{Sp} = \mu/\beta r_\mathrm{p}$ is the slip ratio, with β the slip coefficient being the constant of proportionality between tangential stress on the surface and the velocity slip, $\tau_{r\theta} = \beta v_\theta$. Thus, $\beta \to \infty \Rightarrow \mathrm{Sp} \to 0$ corresponds to no slip, whereas perfect slip corresponds to $\beta \to 0 \Rightarrow \mathrm{Sp} \to \infty$. These slip parameters are both related empirically to the Knudsen number $\mathrm{Kn} = \lambda/d_\mathrm{p}$, with the result that in the slip flow regime [94]

$$f_\mathrm{drag} = \frac{1}{1+2A\mathrm{Kn}}, \quad \mathrm{Re_r} < 1, 0.1 < \mathrm{Kn} < 42 \tag{211}$$

where

$$A = \frac{1}{2\mathrm{Kn}}\left(\frac{1+3\mathrm{Sp}}{1+2\mathrm{Sp}} - 1\right) = A_1 + A_2\exp\left(-\frac{A_3}{\mathrm{Kn}}\right), \tag{212}$$

with $A_1 \approx 1.15$, $A_2 \approx 0.5$, and $A_3 \approx 0.5$. The quantity $C_\mathrm{C}(\mathrm{Kn}) = f_\mathrm{drag}^{-1}(\mathrm{Kn}) = 1 + 2A\mathrm{Kn}$ is called the Cunningham slip correction factor.

At higher Reynolds numbers, one must employ empirical or semiempirical relationships [5, 6, 7, 95]:

$$f_\mathrm{drag} = \begin{cases} 1 + 0.15\mathrm{Re_r^{0.687}} & 1 < \mathrm{Re_r} < 1000 \\ \left(\frac{0.445}{24}\right)\mathrm{Re_r} & 1000 < \mathrm{Re_r} < 3\times10^5 \\ \left(\frac{0.07}{24}\right)\mathrm{Re_r} & \mathrm{Re_r} > 3\times10^5 \end{cases}. \tag{213}$$

A shortcoming of using Eq. 213 is that there is a discontinuity for f_drag at $\mathrm{Re_r} = 1000$, so often one will instead use [5, 7, 96]

$$f_\mathrm{drag} = 1 + 0.15\mathrm{Re_r^{0.687}} + \frac{0.0175\mathrm{Re_r}}{1 + 4.25\times10^4\mathrm{Re_r^{-1.16}}}, \quad \mathrm{Re_r} < 3\times10^5 \tag{214}$$

for the subcritical Reynolds number range. By equating Eq. 22 with Eqs. 209 through 214, one obtains a functional relationship $f(\mathrm{Re_r}, C_\mathrm{D}) = 0$ between the drag coefficient C_D and the Reynolds number $\mathrm{Re_r}$—namely, the explicit relationship $C_\mathrm{D} = C_\mathrm{D}(\mathrm{Re_r}) = 24f_\mathrm{drag}(\mathrm{Re_r})/\mathrm{Re_r}$. This is the standard drag curve, which is plotted along with the drag factor f_drag in Figure 16 using Eq. 214 in the subcritical range and Eq. 213 for $\mathrm{Re_r} > 3\times10^5$.

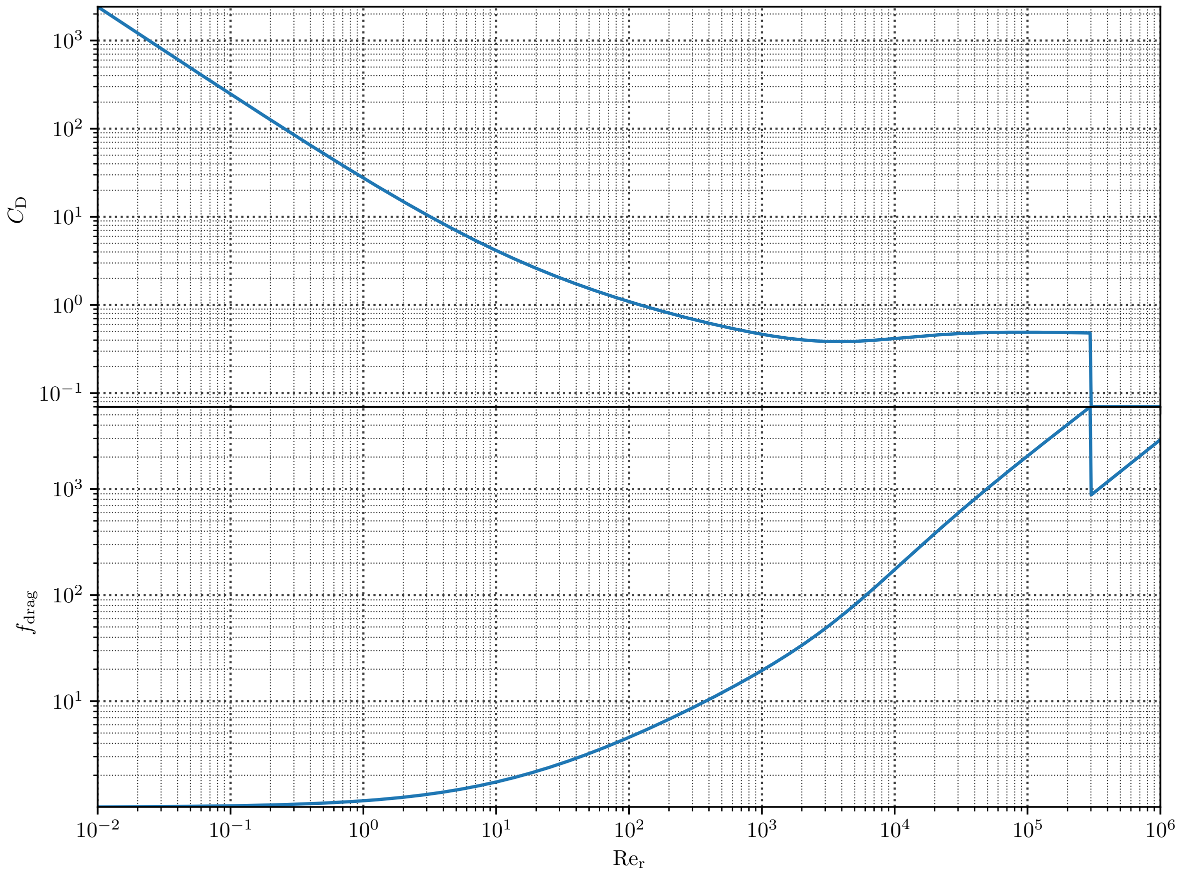

Figure 16. Drag coefficient C_{D} and drag factor f_{drag} for a solid spherical particle as a function of (relative) Reynolds number.

We define the steady state or terminal velocity $\mathbf{v}_{\mathrm{t}} = \lim_{t\to\infty} \mathbf{v}_{\mathrm{p}}(t)$ as the particle velocity such that the gravitational (including buoyancy) and drag forces in a quiescent infinite fluid[*] balance [5]:

$$\begin{aligned} \lim_{t\to\infty} m_{\mathrm{p}} \frac{\mathrm{d}\mathbf{v}_{\mathrm{p}}(t)}{\mathrm{d}t} &= \lim_{t\to\infty} \mathbf{F}_{\mathrm{drag}}(t) + \mathbf{F}_{\mathrm{grav}} \\ \Rightarrow 3\pi\mu d_{\mathrm{p}} f_{\mathrm{drag}}\left(|\mathbf{v}_{\mathrm{t}}|\right)\mathbf{v}_{\mathrm{t}} &= m_{\mathrm{grav}}\mathbf{g} \\ \Rightarrow 3\pi\mu d_{\mathrm{p}} f_{\mathrm{drag}}(v_t) v_{\mathrm{t}} &= \frac{\pi}{6}\left|\rho_{\mathrm{p}} - \rho\right| d_{\mathrm{p}}^{3} g \end{aligned} \tag{215}$$

where we have used Eqs. 5 and 6 with $V_{\mathrm{p}} = \pi d_{\mathrm{p}}^{3}/6$ to write the gravitational force, aligning the velocity with the gravitational vector in steady state and defining v_{t} as positive in the direction of gravity. In the Stokes flow regime $\mathrm{Re}_{\mathrm{r}} < 1$ for which f_{drag} is nearly independent of velocity, we can readily solve Eq. 215 for the terminal velocity:

$$v_{\mathrm{t}} = \frac{\left|\rho_{\mathrm{p}} - \rho\right| d_{\mathrm{p}}^{2} g}{18\mu f_{\mathrm{drag}}} = \left|1 - \frac{\rho}{\rho_{\mathrm{p}}}\right| \frac{\tau_{\mathrm{Stokes}} g}{f_{\mathrm{drag}}}. \tag{216}$$

[*] We define a quiescent infinite fluid as a fluid such that $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}$. Thus, one may choose an inertial frame of reference with $\mathbf{u} = 0$.

For solid particles in air with $\rho \ll \rho_\mathrm{p}$, this reduces to $v_\mathrm{t} \sim \tau_\mathrm{Stokes} g / f_\mathrm{drag}$. Substituting Eq. 211 into Eq. 216 then yields the terminal velocity for small particles $d_\mathrm{p} \lesssim 1\,\mathrm{\mu m}$ in Stokes flows:

$$\begin{aligned} v_\mathrm{t} &= \left|1 - \frac{\rho}{\rho_\mathrm{p}}\right| \tau_\mathrm{Stokes} g \left(1 + 2A\mathrm{Kn}\right) \\ &= \left|1 - \frac{\rho}{\rho_\mathrm{p}}\right| \tau_\mathrm{Stokes} g \left\{1 + 2\mathrm{Kn}\left[A_1 + A_2 \exp\left(-\frac{A_3}{\mathrm{Kn}}\right)\right]\right\}, \quad \mathrm{Re_r} < 1, \quad 0.1 < \mathrm{Kn}. \end{aligned} \tag{217}$$

On the other hand, for general, non-Stokes flows, we may use Eqs. 21b and 22 to write the drag force balance Eq. 215 as

$$\frac{\pi}{8}\rho C_\mathrm{D} d_\mathrm{p}^2 v_\mathrm{t}^2 = \frac{\pi}{6}\left|\rho_\mathrm{p} - \rho\right| d_\mathrm{p}^3 g. \tag{218}$$

Solving Eq. 218 for the terminal velocity yields

$$v_\mathrm{t} = \sqrt{\frac{4g\left|\rho_\mathrm{p} - \rho\right| d_\mathrm{p}}{3\rho C_\mathrm{D}}} = \frac{\nu}{d_\mathrm{p}}\sqrt{\frac{4}{3}\frac{\mathrm{Ar}}{C_\mathrm{D}}}, \tag{219}$$

whereas solving for the Reynolds number $\mathrm{Re_r}(v_\mathrm{t}) = d_\mathrm{p} v_\mathrm{t}/\nu$ yields

$$\mathrm{Re_r^2}(v_\mathrm{t}) = \left(\frac{d_\mathrm{p} v_\mathrm{t}}{\nu}\right)^2 = \frac{4}{3}\frac{\left|\rho_\mathrm{p} - \rho\right| d_\mathrm{p}^3 g}{\rho \nu^2 C_\mathrm{D}} = \frac{4}{3}\frac{\mathrm{Ar}}{C_\mathrm{D}}, \tag{220}$$

where the right-hand side of Eqs. 219 and 220 are written in terms of the Archimedes number [6, 97]

$$\mathrm{Ar} = \frac{g\left|\rho_\mathrm{p} - \rho\right| d_\mathrm{p}^3}{\rho \nu^2} = \frac{g\rho\left|\rho_\mathrm{p} - \rho\right| d_\mathrm{p}^3}{\mu^2}. \tag{221}$$

Thus, Eq. 220 is a dimensionless relationship between the Reynolds number $\mathrm{Re_r}$, the drag coefficient C_D, and the Archimedes number Ar [33, 81, 97]:

$$\mathrm{Re_r^2} = \frac{4}{3}\frac{\mathrm{Ar}}{C_\mathrm{D}}. \tag{222}$$

Note that the Archimedes number Ar in Eq. 221 depends only on physical and geometric parameters of the particle and fluid and is independent of particle or fluid velocity. Figure 17 shows the Archimedes number in air over a range of particle densities for various particle-air density ratios.

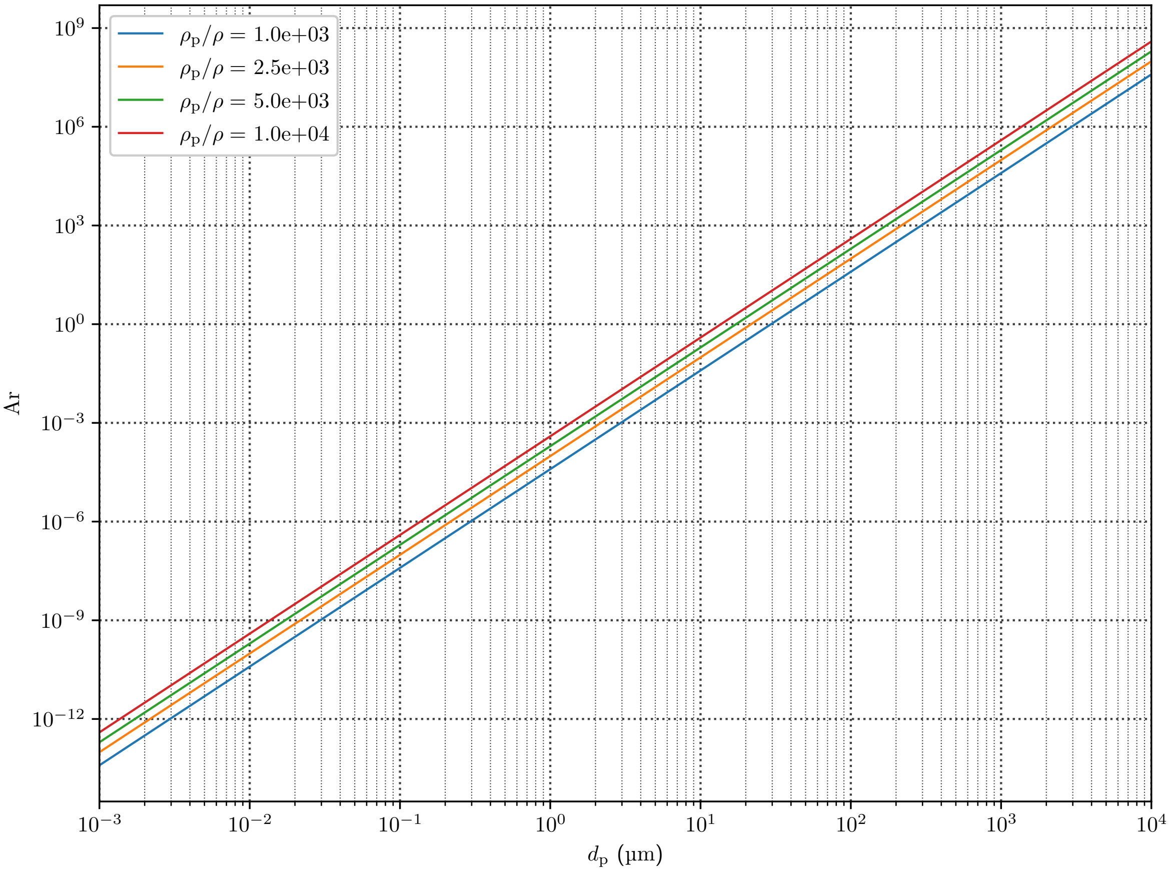

Figure 17. Archimedes number Eq. 221 as a function of particle diameter $10^{-3}\,\mu\mathrm{m} \leq d_\mathrm{p} \leq 10^{4}\,\mu\mathrm{m}$ **for the particle-air density ratios** $\rho_\mathrm{p}/\rho \in \{1, 2.5, 5, 10\} \times 10^{3}$ **in air at the altitude** $z = 1\,\mathrm{km}$ **using the US Standard Atmosphere [1].** The kinematic viscosity of air $\nu = 1.581 \times 10^{-5}\,\mathrm{m}^2\,\mathrm{s}^{-1}$ is calculated using Eq. 9 with $T = 281.651\,\mathrm{K}$ and $\rho = 1.1117\,\mathrm{kg\,m}^{-3}$.

Using Eq. 222, one can eliminate one of the variables Re_r, C_D, or Ar and define a functional relationship between any of the two of these variables: $f(\mathrm{Re}_\mathrm{r}, C_\mathrm{D}) = 0$ (as shown in Figure 16), $f(\mathrm{Ar}, C_\mathrm{D}) = 0$ [97], or $f(\mathrm{Ar}, \mathrm{Re}_\mathrm{r}) = 0$ [5, 98]. These latter two relationships are actually more useful for determining the terminal velocity v_t since the Reynolds number and drag coefficient depend on the velocity either explicitly or implicitly. The relationship $f(\mathrm{Re}_\mathrm{r}, C_\mathrm{D}) = 0$ would thus require solving a complex nonlinear equation (i.e., Eq. 220 or Eq. 222) of the form $\mathrm{Re}_\mathrm{r}^2(v_\mathrm{t})C_\mathrm{D}\left[\mathrm{Re}_\mathrm{r}(v_\mathrm{t})\right] = 4\mathrm{Ar}/3$ for terminal velocity v_t. In contrast, one can readily calculate Ar based on geometric and physical parameters and either:

1. use $f(C_\mathrm{D}, \mathrm{Ar}) = 0$ to determine C_D and then use Eq. 219 to calculate v_t, or
2. use $f(\mathrm{Ar}, \mathrm{Re}_\mathrm{r}) = 0$ to determine Re_r and then use Eq. 23, i.e. $\mathrm{Re}_\mathrm{r}(v_\mathrm{t}) = d_\mathrm{p}v_\mathrm{t}/\nu \Rightarrow v_\mathrm{t} = \nu\mathrm{Re}_\mathrm{r}/d_\mathrm{p}$ to calculate v_t.

The former relationship is given empirically for free-falling spheres by [97]

$$C_\mathrm{D} = \frac{432}{\mathrm{Ar}}(1 + 0.0470\mathrm{Ar}^{2/3}) + \frac{0.517}{1 + 154\mathrm{Ar}^{-1/3}}, \quad \mathrm{Ar} < 2.2 \times 10^{10}, \tag{223}$$

which corresponds to the full subcritical range $\mathrm{Re}_\mathrm{r} < 3 \times 10^5$.

The latter relationship over the same range is given by [5, 98]

$$\mathrm{Re}_\mathrm{r} = \frac{d_\mathrm{p} v_\mathrm{t}}{\nu} = \begin{cases} \left(\sqrt{22 + \sqrt{4.89\mathrm{Ar}}} - \sqrt{22}\right)^2 & \mathrm{Ar} < 4 \times 10^5 \\ 1.74\sqrt{\mathrm{Ar}} & 4 \times 10^5 < \mathrm{Ar} < 3 \times 10^{10} \end{cases}. \tag{224}$$

Another empirical, polynomial relationship of the second type was proposed by Davies [81] and—in addition to a similar relationship by Beard [82] for smaller Ar—is what DELFIC uses in subroutine `settle` [2, 33]:

$$\begin{cases} \mathrm{Re}_\mathrm{r} & \mathrm{Re}_\mathrm{r} < 4 \\ \log_{10} \mathrm{Re}_\mathrm{r} & 3 < \mathrm{Re}_\mathrm{r} < 10^4 \end{cases} = \begin{cases} \frac{1}{24}\left(\frac{4}{3}\mathrm{Ar}\right) - 2.3363 \times 10^{-4}\left(\frac{4}{3}\mathrm{Ar}\right)^2 & \\ +2.0154 \times 10^{-6}\left(\frac{4}{3}\mathrm{Ar}\right)^3 & \\ -6.9105 \times 10^{-9}\left(\frac{4}{3}\mathrm{Ar}\right)^4 & \frac{4}{3}\mathrm{Ar} < 140 \\ -1.295\,36 + 0.986\log_{10}\left(\frac{4}{3}\mathrm{Ar}\right) & \\ -0.046677\left[\log_{10}\left(\frac{4}{3}\mathrm{Ar}\right)\right]^2 & \\ +1.1235 \times 10^{-3}\left[\log_{10}\left(\frac{4}{3}\mathrm{Ar}\right)\right]^3 & 100 < \frac{4}{3}\mathrm{Ar} < 4.5 \times 10^7 \end{cases}. \tag{225}$$

Equations 225 do not cover the full subcritical range, but Davies [81] suggests that for $\mathrm{Re}_\mathrm{r} > 10^4$, the interest becomes purely aerodynamical, and Eqs. 225 together cover spherical particles of density $\rho_\mathrm{p} \in \{1, 10\} \times 10^3\,\mathrm{kg\,m^{-3}}$ up to mass $0.48\,\mathrm{g}$ or diameter $d_\mathrm{p} \in \{5, 10\} \times 10^3\,\mathrm{\mu m}$ in the air. Figure 18 compares the three methods for calculating $\mathrm{Re}_\mathrm{r}(v_\mathrm{t})$ as a function of Ar and shows their relative error compared with the first method:

1. Eq. 222 with Eq. 214 for the drag curve $C_\mathrm{D}(\mathrm{Re}_\mathrm{r}) = 24 f_\mathrm{drag}(\mathrm{Re}_\mathrm{r})/\mathrm{Re}_\mathrm{r}$,

$$\mathrm{Re}_\mathrm{r}^2 C_\mathrm{D}(\mathrm{Re}_\mathrm{r}) = 24\mathrm{Re}_\mathrm{r} f_\mathrm{drag}(\mathrm{Re}_\mathrm{r}) = \frac{4}{3}\mathrm{Ar}, \tag{226}$$

2. Eq. 224, and

3. Eq. 225.

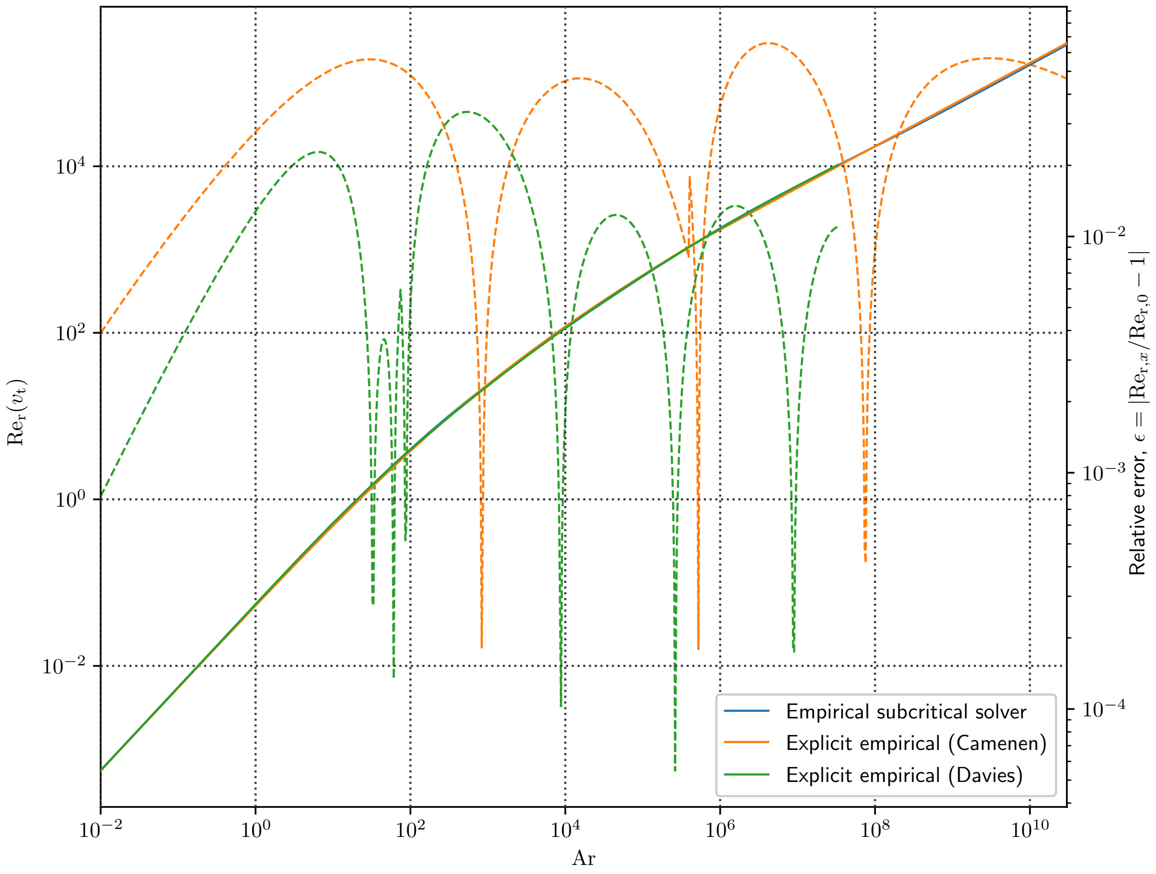

Figure 18. Left axis (*solid*): terminal (relative) Reynolds number $\mathrm{Re_r}(v_\mathrm{t})$ calculated using the (*blue*) nonlinear drag curve Eq. 226 with Eq. 214, (*orange*) Eq. 224, or (*green*) Eq. 225; right axis (*dashed*): relative error of latter two methods compared with the drag curve solution.

The method of Davies [81] has a smaller error relative to the explicit solution of Eq. 222 when compared with Camenen [5, 98], with L^2-norm relative errors $\epsilon_2(\mathrm{Re}_{\mathrm{r},x}) = \|\mathrm{Re}_{\mathrm{r},x} - \mathrm{Re}_{\mathrm{r},0}\|_2 \,/\, \|\mathrm{Re}_{\mathrm{r},0}\|_2$ over the entire domain of $\epsilon_2(\mathrm{Re}_{\mathrm{r,Davies}}) \sim 8.4 \times 10^{-3}$ and $\epsilon_2(\mathrm{Re}_{\mathrm{r,Camenen}}) \sim 5.1 \times 10^{-2}$, respectively. However, this is in reference to the drag curve of Eq. 222, so this comparison assumes that this curve is the best fit to experimental data.

On the other hand, we are typically primarily interested in the terminal velocity v_t itself and not its Reynolds number $\mathrm{Re_r}(v_\mathrm{t})$. Thus, to compare the three methods presented in Figure 18 to Eq. 223, we should compare the drag coefficient C_D calculated either using Eq. 223 directly or using Eq. 220 or Eq. 222 with the results of Figure 18. Additionally, we can compare the terminal velocity v_t of each method using either Eq. 219 or Eq. 220, respectively. Karamanev [97] argues that since Eq. 223 calculates C_D directly and v_t is calculated by the square root of C_D in Eq. 219, if Eq. 223 has an error comparable to the other empirical methods, then the error in v_t will be half that of the methods which use Eq. 220.

This comparison, including the error relative to the empirical drag curve solution, is shown in Figure 19. Table 2 shows the L^2-norm relative errors $\epsilon_2(C_{\mathrm{D},x}) = \|C_{\mathrm{D},x} - C_{\mathrm{D},0}\|_2 \,/\, \|C_{\mathrm{D},0}\|_2$ over the entire domain, from which we can see that indeed the method of Karamanev [97] might yield a more accurate drag coefficient C_{D}. The terminal velocity v_{t} relative errors are more similar between the two methods, but Eq. 223 is valid over a larger range of Ar (and $\mathrm{Re_r}$) than the method of Davies [81], so the method of Karamanev [97] might be a better choice for determining v_{t}. Figure 20 shows the terminal velocity v_{t} plotted using this latter method as a function of both the particle diameter d_{p} and the corresponding Archimedes number Ar for air at the altitude $z = 1\,\mathrm{km}$ for a set of particle density ratios ρ_{p}/ρ. Note that compared with Figure 3, Figure 20 accounts for the effects of both buoyancy forces, as well as the variable drag factor f_{drag}. This latter effect results in the (slightly) density-dependent slopes of the curves for smaller particle diameters or Archimedes numbers, as well as the leveling off of the terminal velocity for larger particle diameters or Archimedes numbers. For submicron particles, the Cunningham slip correction factor $f_{\mathrm{drag}}^{-1}(\mathrm{Kn}) = C_{\mathrm{C}}(\mathrm{Kn})$ results in a greater terminal velocity than would otherwise be expected without the correction.

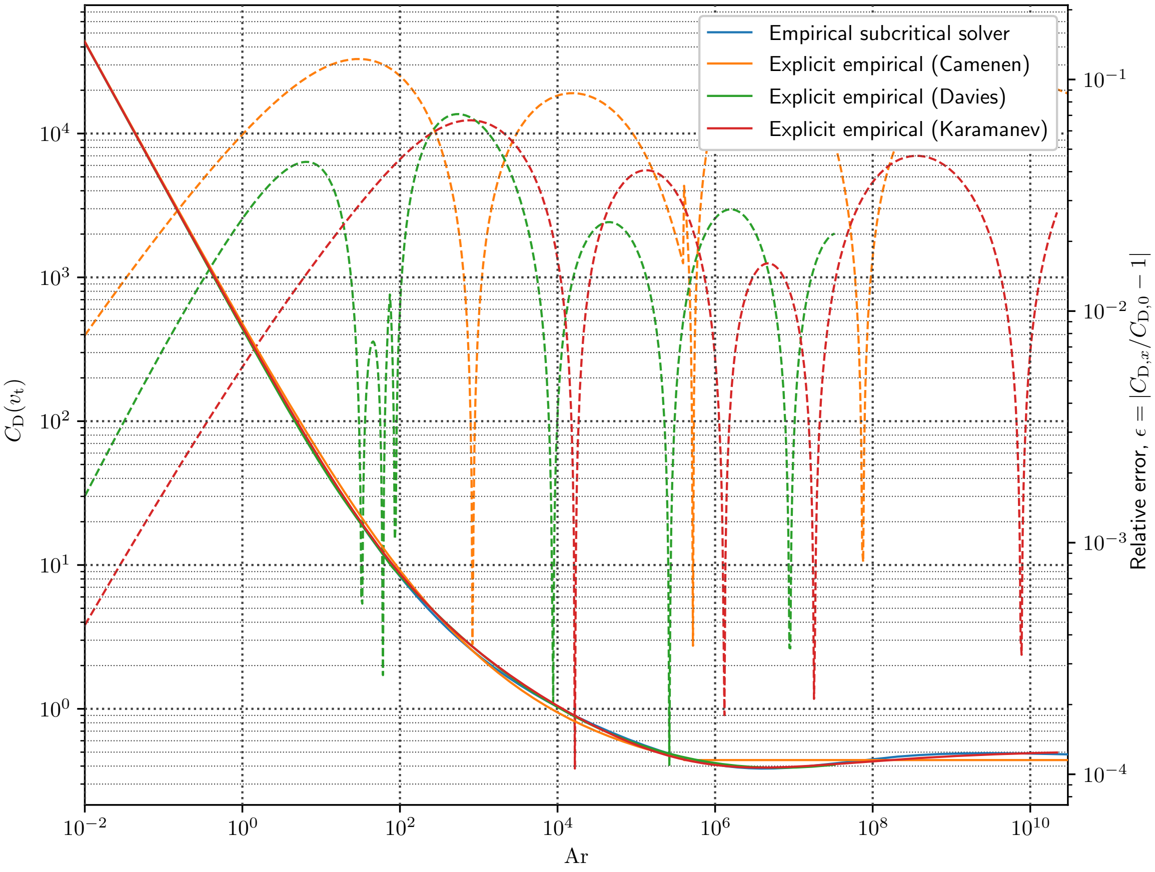

Figure 19. Left axis (*solid*): terminal drag coefficient $C_{\mathrm{D}}(v_{\mathrm{t}})$ calculated using the (*blue*) nonlinear drag curve Eq. 226 with Eq. 214, (*orange*) Eq. 224, (*green*) Eq. 225, or (*red*) Eq. 223; right axis (*dashed*): relative error of latter three methods compared with the drag curve solution.

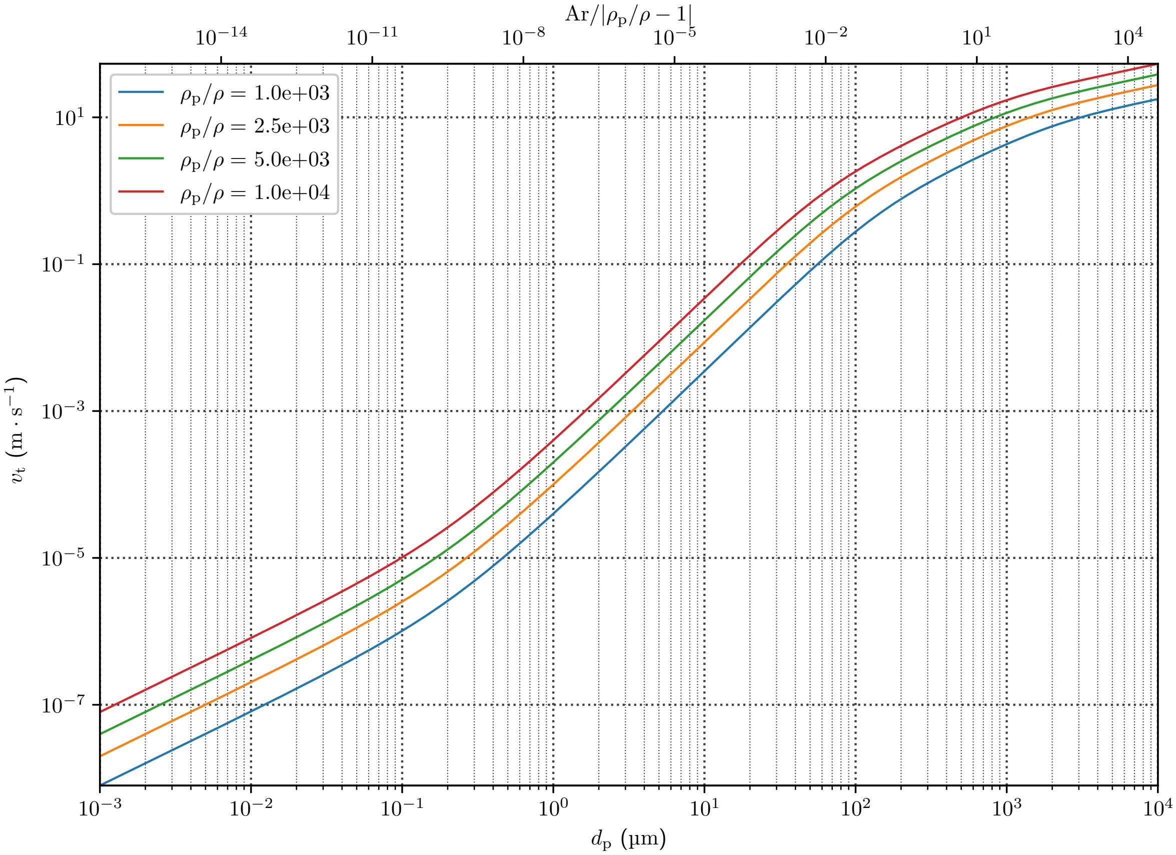

Figure 20. Particle terminal velocity v_t determined using Eq. 223 with Eq. 219 and the slip correction Eq. 211 as a function of (*bottom axis*) particle diameter $10^{-3}\,\mu\mathrm{m} \leq d_\mathrm{p} \leq 10^{4}\,\mu\mathrm{m}$ or (*top axis*) the corresponding density-normalized Archimedes number $4.6 \times 10^{-17} \leq \mathrm{Ar}/|\rho_\mathrm{p}/\rho - 1| \leq 4.6 \times 10^{4}$ for the particle-air density ratios $\rho_\mathrm{p}/\rho \in \{1, 2.5, 5, 10\} \times 10^{3}$ in air at the altitude $z = 1\,\mathrm{km}$ using the US Standard Atmosphere [1]. The kinematic viscosity of air $\nu = 1.581 \times 10^{-5}\,\mathrm{m^2\,s^{-1}}$ is calculated using Eq. 9 with $T = 281.651\,\mathrm{K}$ and $\rho = 1.1117\,\mathrm{kg\,m^{-3}}$.

Table 2. L^2-norm relative errors $\epsilon_2(C_{\mathrm{D},x}) = \|C_{\mathrm{D},x} - C_{\mathrm{D},0}\|_2 \,/\, \|C_{\mathrm{D},0}\|_2$ and $\epsilon_2(v_{\mathrm{t},x}) = \|v_{\mathrm{t},x} - v_{\mathrm{t},0}\|_2 \,/\, \|v_{\mathrm{t},0}\|_2$ relative to the solution using the empirical subcritical drag curve Eq. 226 with Eq. 214.

Method	$\epsilon_2(C_{\mathrm{D},x})$	$\epsilon_2(v_{\mathrm{t},x})$
Camenen [5,98]	1.1×10^{-2}	5.1×10^{-2}
Davies [81]	2.6×10^{-3}	8.4×10^{-3}
Karamanev [97]	6.7×10^{-4}	1.0×10^{-2}

APPENDIX C. DELFIC PARCEL CONCENTRATION DISTRIBUTION: PROBABILITY DENSITY FORMULATION

APPENDIX C. DELFIC PARCEL CONCENTRATION DISTRIBUTION: PROBABILITY DENSITY FORMULATION

Noting that the bivariate Gaussian distributions and uniform distribution of Eqs. 129 correspond to the probability density functions (PDFs) of the normal and uniform continuous distributions, we can write that

$$\begin{aligned}
X &\sim \mathcal{N}\left[0, \sigma_x^2(t-t_0)\right], \\
Y &\sim \mathcal{N}\left[0, \sigma_y^2(t-t_0)\right], \\
Z &\sim \mathcal{U}\left[Z_1, Z_2\right],
\end{aligned}$$

or

$$\begin{aligned}
x &\sim \mathcal{N}\left[\mu_x(t), \sigma_x^2(t-t_0)\right] = \mathcal{N}\left[x_0 + \int_0^{t-t_0} \langle v_{\mathrm{p},x}(\tau+t_0)\rangle \, \mathrm{d}\tau, \sigma_x^2(t-t_0)\right], \\
y &\sim \mathcal{N}\left[\mu_y(t), \sigma_x^2(t-t_0)\right] = \mathcal{N}\left[y_0 + \int_0^{t-t_0} \langle v_{\mathrm{p},y}(\tau+t_0)\rangle \, \mathrm{d}\tau, \sigma_y^2(t-t_0)\right], \\
z &\sim \mathcal{U}\left[z_1(t), z_2(t)\right] = \mathcal{U}\left[z_1(t_0) + \int_0^{t-t_0} \langle v_{\mathrm{p},z}(\tau+t_0)\rangle \, \mathrm{d}\tau, z_2(t_0) + \int_0^{t-t_0} \langle v_{\mathrm{p},z}(\tau+t_0)\rangle \, \mathrm{d}\tau\right],
\end{aligned}$$

where

$$\mu_i(t) = \mathrm{E}\left[x_i(t)\right] = \langle x_i(t)\rangle = x_{0,i} + \int_0^{t-t_0} \langle v_{\mathrm{p},i}(\tau+t_0)\rangle \, \mathrm{d}\tau \tag{227}$$

for $i \in \{1,2\} = \{x,y\}$. Thus, Eqs. 129 can be written as

$$C_{XYZ}(\tilde{\mathbf{X}}, t) = Q_0 \theta(t-t_0) f_X(\tilde{X}) f_Y(\tilde{Y}) f_Z(\tilde{Z}), \tag{228a}$$

$$\langle C(\tilde{\mathbf{x}}, t)\rangle = Q_0 \theta(t-t_0) f_x(\tilde{x}) f_y(\tilde{y}) f_z(\tilde{z}), \tag{228b}$$

where

$$f_{X_i}(\tilde{X}_i) = \frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)} \exp\left(-\frac{\tilde{X}_i^2}{2\sigma_i^2(t-t_0)}\right) = \frac{1}{\sigma_i(t-t_0)} \varphi\left[\frac{\tilde{X}_i}{\sigma_i(t-t_0)}\right], \tag{229a}$$

$$f_Z(\tilde{Z}) = \frac{\theta(Z-Z_1) - \theta(Z-Z_2)}{Z_2 - Z_1}, \tag{229b}$$

$$f_{x_i}(\tilde{x}_i) = \frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)} \exp\left\{-\frac{\left[\tilde{x}_i - \mu_i(t)\right]^2}{2\sigma_i^2(t-t_0)}\right\} = \frac{1}{\sigma_i(t-t_0)} \varphi\left[\frac{\tilde{x}_i - \mu_i(t)}{\sigma_i(t-t_0)}\right], \tag{229c}$$

$$f_z(\tilde{z}) = \frac{\theta\left[\tilde{z} - z_1(t)\right] - \theta\left[\tilde{z} - z_2(t)\right]}{z_2(t) - z_1(t)}, \tag{229d}$$

with $\varphi(x) = e^{-x^2/2}/\sqrt{2\pi}$ the standard normal distribution function. Note that the random variables in each dimension are independent of each other as is evident by the separable nature of the PDFs in Eqs. 228. Thus, there is no correlation between the variables in each dimension.

C.1 INDEPENDENT, UNCORRELATED JOINT DISTRIBUTION

We may write the joint Gaussian distributions Eqs. 229a (f_X and f_Y) or Eqs. 229c (f_x and f_y) as a multivariate (i.e., bivariate) normal distribution:

$$\begin{aligned}
\mathbf{X}_{XY} &= (X, Y)^{\mathrm{T}} \sim \mathcal{N}\left[0, \mathbf{\Sigma}(t)\right], \\
\mathbf{x}_{xy} &= (x, y)^{\mathrm{T}} \sim \mathcal{N}\left[\boldsymbol{\mu}_{xy}(t), \mathbf{\Sigma}(t)\right],
\end{aligned}$$

where

$$\boldsymbol{\mu}_{xy}(t) = [\mu_x(t), \mu_y(t)]^{\mathrm{T}}\,, \tag{230a}$$

$$\boldsymbol{\Sigma}(t) = \begin{bmatrix} \sigma_x^2(t-t_0) & 0 \\ 0 & \sigma_y^2(t-t_0) \end{bmatrix}, \tag{230b}$$

which have the respective joint PDFs

$$f_{\mathbf{X}_{XY}}(\tilde{\mathbf{X}}_{XY}) = \frac{1}{2\pi\sqrt{|\boldsymbol{\Sigma}(t)|}} \exp\left[-\frac{1}{2}\tilde{\mathbf{X}}_{XY}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(t)\tilde{\mathbf{X}}_{XY}\right] \tag{231a}$$

$$= \prod_{i=1}^{2} \frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)} \exp\left(-\frac{\tilde{X}_i^2}{2\sigma_i^2(t-t_0)}\right) = \prod_{i=1}^{2} \frac{1}{\sigma_i(t-t_0)}\varphi\left[\frac{\tilde{X}_i}{\sigma_i(t-t_0)}\right] \tag{231b}$$

$$f_{\mathbf{x}_{xy}}(\tilde{\mathbf{x}}_{xy}) = \frac{1}{2\pi\sqrt{|\boldsymbol{\Sigma}(t)|}} \exp\left\{-\frac{1}{2}\left[\tilde{\mathbf{x}}_{xy} - \boldsymbol{\mu}_{xy}(t)\right]^{\mathrm{T}} \boldsymbol{\Sigma}^{-1}(t)\left[\tilde{\mathbf{x}}_{xy} - \boldsymbol{\mu}_{xy}(t)\right]\right\} \tag{231c}$$

$$= \prod_{i=1}^{2} \frac{1}{\sqrt{2\pi}\sigma_i(t-t_0)} \exp\left(-\frac{[\tilde{x}_i - \mu_i(t)]^2}{2\sigma_i^2(t-t_0)}\right) = \prod_{i=1}^{2} \frac{1}{\sigma_i(t-t_0)}\varphi\left[\frac{\tilde{x}_i - \mu_i(t)}{\sigma_i(t-t_0)}\right] \tag{231d}$$

where

$$|\boldsymbol{\Sigma}(t)| \equiv \det \boldsymbol{\Sigma}(t) = \sigma_x^2(t-t_0)\sigma_y^2(t-t_0), \tag{232a}$$

$$\boldsymbol{\Sigma}^{-1}(t) = \begin{bmatrix} \sigma_x^{-2}(t-t_0) & 0 \\ 0 & \sigma_y^{-2}(t-t_0) \end{bmatrix}. \tag{232b}$$

C.2 DEPENDENT, CORRELATED JOINT DISTRIBUTION

More generally, let us suppose that our bivariate Gaussian distribution in the fixed Eulerian coordinates has a nonzero correlation function

$$\rho_{(x,y)}(t) \equiv \frac{\operatorname{cov}(x,y)}{\sigma_x(t-t_0)\sigma_y(t-t_0)} = \frac{\langle [x-\mu_x(t)]\,[y-\mu_y(t)]\rangle}{\sigma_x(t-t_0)\sigma_y(t-t_0)}. \tag{233}$$

That is, $\mathbf{x}_{xy} \sim \mathcal{N}\left[\boldsymbol{\mu}_{xy}(t), \boldsymbol{\Sigma}_{xy}(t)\right]$, where

$$\boldsymbol{\Sigma}_{xy}(t) = \begin{bmatrix} \sigma_x^2(t-t_0) & \rho_{(x,y)}(t)\sigma_x(t-t_0)\sigma_y(t-t_0) \\ \rho_{(x,y)}(t)\sigma_x(t-t_0)\sigma_y(t-t_0) & \sigma_y^2(t-t_0) \end{bmatrix}, \tag{234}$$

and the joint PDF of $\mathbf{x}_{xy}$ is given by

$$f_{\mathbf{x}_{xy}}(\tilde{\mathbf{x}}_{xy}) = \frac{1}{2\pi\sqrt{|\boldsymbol{\Sigma}_{xy}(t)|}} \exp\left\{-\frac{1}{2}\left[\tilde{\mathbf{x}}_{xy} - \boldsymbol{\mu}_{xy}(t)\right]^{\mathrm{T}} \boldsymbol{\Sigma}_{xy}^{-1}(t)\left[\tilde{\mathbf{x}}_{xy} - \boldsymbol{\mu}_{xy}(t)\right]\right\}, \tag{235}$$

with

$$|\boldsymbol{\Sigma}_{xy}(t)| = \sigma_x^2(t-t_0)\sigma_y^2(t-t_0)\left[1 - \rho_{(x,y)}^2(t)\right], \tag{236a}$$

$$\begin{aligned} \boldsymbol{\Sigma}_{xy}^{-1}(t) &= |\boldsymbol{\Sigma}_{xy}(t)|^{-1} \begin{bmatrix} \sigma_y^2(t-t_0) & -\rho_{(x,y)}(t)\sigma_x(t-t_0)\sigma_y(t-t_0) \\ -\rho_{(x,y)}(t)\sigma_x(t-t_0)\sigma_y(t-t_0) & \sigma_x^2(t-t_0) \end{bmatrix} \\ &= \left[1 - \rho_{(x,y)}^2(t)\right]^{-1} \begin{bmatrix} \sigma_x^{-2}(t-t_0) & -\rho_{(x,y)}(t)\sigma_x^{-1}(t-t_0)\sigma_y^{-1}(t-t_0) \\ -\rho_{(x,y)}(t)\sigma_x^{-1}(t-t_0)\sigma_y^{-1}(t-t_0) & \sigma_y^{-2}(t-t_0) \end{bmatrix}. \end{aligned} \tag{236b}$$

This might happen if, for example, the parcel is not treated simply as a point but experiences the effects of wind shear. Substituting Eqs. 236a and 236b into Eq. 235, we can explicitly write the PDF as

$$
\begin{aligned}
f_{\mathbf{x}_{xy}}(\tilde{\mathbf{x}}_{xy}) = {} & \frac{1}{2\pi\sigma_x(t-t_0)\sigma_y(t-t_0)\sqrt{1-\rho^2_{(x,y)}(t)}} \times \\
& \exp\left(-\frac{1}{2\left[1-\rho^2_{(x,y)}(t)\right]} \left\{ \frac{[\tilde{x}-\mu_x(t)]^2}{\sigma_x^2(t-t_0)} + \frac{[\tilde{y}-\mu_y(t)]^2}{\sigma_y^2(t-t_0)} \right.\right. \\
& \left.\left. -2\rho_{(x,y)}(t)\left[\frac{\tilde{x}-\mu_x(t)}{\sigma_x(t-t_0)}\right]\left[\frac{\tilde{y}-\mu_y(t)}{\sigma_y(t-t_0)}\right] \right\} \right). \qquad (237)
\end{aligned}
$$

From Eq. 237, it is apparent that x and y are independent iff $\rho^2_{(x,y)}(t) = 0$.

Note that $\mathbf{\Sigma}_{xy}(t)$ is a real symmetric matrix (specifically, a positive semi-definite matrix). Thus, it has an eigendecomposition of the form $\mathbf{\Sigma}_{xy}(t) = \mathbf{Q}(t)\mathbf{\Lambda}(t)\mathbf{Q}^{\mathrm{T}}(t)$, where $\mathbf{Q}(t)$ is an orthogonal matrix whose columns are the eigenvectors of $\mathbf{\Sigma}_{xy}(t)$, and $\mathbf{\Lambda}(t)$ is a diagonal matrix whose entries are the eigenvalues of $\mathbf{\Sigma}_{xy}(t)$. Without loss of generality, we can let $\mathbf{Q}(t)$ be the rotation matrix in two dimensions:

$$
\mathbf{Q}(t) = \mathbf{R}(t) = \begin{bmatrix} \cos\theta(t) & -\sin\theta(t) \\ \sin\theta(t) & \cos\theta(t) \end{bmatrix}. \qquad (238)
$$

Then we can write (dropping explicit time dependence for brevity)

$$
\begin{aligned}
\mathbf{\Lambda}(t) &= \mathbf{R}^{\mathrm{T}}(t)\mathbf{\Sigma}_{xy}(t)\mathbf{R}(t) \\
\Rightarrow \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} &= \mathbf{R}^{\mathrm{T}}(t)\mathbf{\Sigma}_{xy}(t)\mathbf{R}(t) \\
&= \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \sigma_x^2\cos\theta + \rho\sigma_x\sigma_y\sin\theta & \rho\sigma_x\sigma_y\cos\theta - \sigma_x^2\sin\theta \\ \rho\sigma_x\sigma_y\cos\theta + \sigma_y^2\sin\theta & \sigma_y^2\cos\theta - \rho\sigma_x\sigma_y\sin\theta \end{bmatrix} \\
&= \begin{bmatrix} \sigma_x^2\cos^2\theta + 2\rho\sigma_x\sigma_y\cos\theta\sin\theta + \sigma_y^2\sin^2\theta & \rho\sigma_x\sigma_y\left(\cos^2\theta - \sin^2\theta\right) + \left(\sigma_y^2-\sigma_x^2\right)\cos\theta\sin\theta \\ \rho\sigma_x\sigma_y\left(\cos^2\theta - \sin^2\theta\right) + \left(\sigma_y^2-\sigma_x^2\right)\cos\theta\sin\theta & \sigma_x^2\sin^2\theta - 2\rho\sigma_x\sigma_y\cos\theta\sin\theta + \sigma_y^2\cos^2\theta \end{bmatrix} \\
&\Rightarrow \begin{cases} \lambda_1 & = \sigma_x^2\cos^2\theta + \rho\sigma_x\sigma_y\sin 2\theta + \sigma_y^2\sin^2\theta \\ \lambda_2 & = \sigma_x^2\sin^2\theta - \rho\sigma_x\sigma_y\sin 2\theta + \sigma_y^2\cos^2\theta \\ 0 & = \rho\sigma_x\sigma_y\cos 2\theta + \frac{1}{2}\left(\sigma_y^2-\sigma_x^2\right)\sin 2\theta \end{cases}.
\end{aligned}
$$

This latter equation can be solved for $\rho_{(x,y)}(t)$ or $\theta(t)$ to show the relationship between the correlation and the rotation angle:

$$
\rho_{(x,y)}(t) = \frac{\left[\sigma_x^2(t-t_0) - \sigma_y^2(t-t_0)\right]}{2\sigma_x(t-t_0)\sigma_y(t-t_0)}\tan 2\theta(t) \qquad (239a)
$$

$$
\Rightarrow \theta(t) = \frac{1}{2}\tan^{-1}\left[\frac{2\rho_{(x,y)}(t)\sigma_x(t-t_0)\sigma_y(t-t_0)}{\sigma_x^2(t-t_0)-\sigma_y^2(t-t_0)}\right] + \frac{\pi k}{2}, \qquad (239b)
$$

where $k \in \mathbb{Z}$. Moreover, we may define the linear transform

$$
\mathbf{x}_{xy}(t) = \boldsymbol{\mu}_{xy}(t) + \mathbf{Q}(t)\mathbf{X}_{XY}(t) \qquad (240a)
$$

$$
\Rightarrow \mathbf{X}_{XY}(t) = \mathbf{Q}^{\mathrm{T}}(t)\left[\mathbf{x}_{xy}(t) - \boldsymbol{\mu}_{xy}(t)\right], \qquad (240b)
$$

for a new normal random vector $\mathbf{X}_{XY}$ with mean and covariance

$$\begin{aligned}
\boldsymbol{\mu}_{XY}(t) &= \langle \mathbf{X}_{XY}(t) \rangle \\
&= \mathbf{Q}^{\mathrm{T}}(t) \left\langle \mathbf{x}_{xy}(t) - \boldsymbol{\mu}_{xy}(t) \right\rangle \\
&= 0, \\
\boldsymbol{\Sigma}_{XY}(t) &= \left\langle \left[\mathbf{X}_{XY}(t) - \boldsymbol{\mu}_{XY}(t)\right] \left[\mathbf{X}_{XY}(t) - \boldsymbol{\mu}_{XY}(t)\right]^{\mathrm{T}} \right\rangle \\
&= \left\langle \mathbf{X}_{XY}(t) \mathbf{X}_{XY}^{\mathrm{T}}(t) \right\rangle \\
&= \left\langle \mathbf{Q}^{\mathrm{T}}(t) \left[\mathbf{x}_{xy}(t) - \boldsymbol{\mu}_{xy}(t)\right] \left[\mathbf{x}_{xy}(t) - \boldsymbol{\mu}_{xy}(t)\right]^{\mathrm{T}} \mathbf{Q}(t) \right\rangle \\
&= \mathbf{Q}^{\mathrm{T}}(t) \left\langle \left[\mathbf{x}_{xy}(t) - \boldsymbol{\mu}_{xy}(t)\right] \left[\mathbf{x}_{xy}(t) - \boldsymbol{\mu}_{xy}(t)\right]^{\mathrm{T}} \right\rangle \mathbf{Q}(t) \\
&= \mathbf{Q}^{\mathrm{T}}(t) \boldsymbol{\Sigma}_{xy}(t) \mathbf{Q}(t) \\
&= \boldsymbol{\Lambda}(t).
\end{aligned}$$

This is to say that given the independent, bivariate normally distributed variables $\mathbf{X}_{XY} \sim \mathcal{N}\left[0, \boldsymbol{\Lambda}(t)\right]$ with PDF

$$f_{\mathbf{X}_{XY}}(\tilde{\mathbf{X}}_{XY}) = \frac{1}{2\pi\sqrt{|\boldsymbol{\Lambda}(t)|}} \exp\left[-\frac{1}{2} \tilde{\mathbf{X}}_{XY}^{\mathrm{T}} \boldsymbol{\Lambda}^{-1}(t) \tilde{\mathbf{X}}_{XY}\right] \tag{241a}$$

$$= \prod_{i=1}^{2} \frac{1}{\sqrt{2\pi\lambda_i(t)}} \exp\left(-\frac{\tilde{X}_i^2}{2\lambda_i(t)}\right) = \prod_{i=1}^{2} \frac{1}{\sqrt{\lambda_i(t-t_0)}} \varphi\left[\frac{\tilde{X}_i}{\sqrt{\lambda_i(t-t_0)}}\right], \tag{241b}$$

we can construct the dependent, normally distributed random variables $\mathbf{x}_{xy}$ with the PDF given in Eqs. 235 or 237 by a simple linear transformation given with the rotation matrix $\mathbf{Q}(t) = \mathbf{R}(t)$ and the translation vector $\boldsymbol{\mu}_{xy}(t)$. Thus, the eigenvectors and eigenvalues of $\boldsymbol{\Sigma}_{xy}(t)$, respectively, in the columns of $\mathbf{Q}(t)$ and the diagonal values of $\boldsymbol{\Lambda}(t)$ give the principal axes and variances of the ellipsoid representing the multivariate normal distribution. Then given

$$\boldsymbol{\Sigma}_{XY}(t) = \begin{bmatrix} \sigma_X^2(t-t_0) & 0 \\ 0 & \sigma_Y^2(t-t_0) \end{bmatrix} = \begin{bmatrix} \lambda_1(t) & 0 \\ 0 & \lambda_2(t) \end{bmatrix} = \boldsymbol{\Lambda}(t) \tag{242}$$

and the rotation angle $\theta(t)$, we have

$$\boldsymbol{\Sigma}_{xy}(t) = \mathbf{Q}(t) \boldsymbol{\Sigma}_{XY}(t) \mathbf{Q}^{\mathrm{T}}(t)$$

$$\Rightarrow \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix} = \begin{bmatrix} \sigma_X^2 \cos^2\theta + \sigma_Y^2 \sin^2\theta & \frac{1}{2}(\sigma_X^2 - \sigma_Y^2)\sin 2\theta \\ \frac{1}{2}(\sigma_X - \sigma_Y)\sin 2\theta & \sigma_X^2 \sin^2\theta + \sigma_Y^2 \cos^2\theta \end{bmatrix}$$

$$\Rightarrow \begin{cases} \sigma_x^2 & = \sigma_X^2 \cos^2\theta + \sigma_Y^2 \sin^2\theta \\ \sigma_y^2 & = \sigma_X^2 \sin^2\theta + \sigma_Y^2 \cos^2\theta \\ \rho\sigma_x\sigma_y & = \frac{1}{2}(\sigma_X^2 - \sigma_Y^2)\sin 2\theta \end{cases}, \tag{243}$$

with the latter relationship in Eq. 243 yielding

$$\begin{aligned}
\rho_{(x,y)}(t) &= \frac{(\sigma_X^2 - \sigma_Y^2)\sin 2\theta}{2\sqrt{\sigma_X^2\cos^2\theta + \sigma_Y^2\sin^2\theta}\sqrt{\sigma_X^2\sin^2\theta + \sigma_Y^2\cos^2\theta}} \\
&= \frac{(\sigma_X^2 - \sigma_Y^2)\sin 2\theta}{2\sqrt{\frac{1}{4}\left(\sigma_X^4 + \sigma_Y^4\right)\sin^2 2\theta + \sigma_X^2\sigma_Y^2\left(\cos^4\theta + \sin^4\theta\right)}},
\end{aligned}$$

which is equivalent to Eq. 239a but written explicitly in terms of σ_X and σ_Y.

APPENDIX D. KINETIC THEORY OF GASES AND RAREFIED GAS DYNAMICS

APPENDIX D. KINETIC THEORY OF GASES AND RAREFIED GAS DYNAMICS

Typically, one models gases (or fluids more generally) as a continuum that is in kinematic equilibrium and for which macroscopic quantities such as (mean) velocity, temperature, and density are continuous and slowly varying throughout the fluid [99]. However, in the case of a fluid with sufficiently low density, discrete particle effects become important and one must abandon the continuum hypothesis and adopt a discrete model [99]. That is, one must instead use the methods of molecular gas dynamics or kinetic theory of gases and rarefied gas dynamics [29, 30]. Formally, a fluid will not be considered a continuum when the coefficients of fluxes appearing in the continuum equations of mass, momentum, and heat conservation can no longer be described by lower-order macroscopic quantities [99]. This will happen when the gradients of macroscopic quantities vary on a length scale L small enough to be comparable with the mean free path λ. The Knudsen number, defined generally as the ratio of the mean free path to the characteristic length scale*

$$\mathrm{Kn} = \frac{\lambda}{L}, \tag{246}$$

is thus a useful quantity for determining the regimes of validity of different models for fallout particle dynamics. In particular, there are four regimes identified [29, 99]:

- $\mathrm{Kn} \lesssim 0.01$: continuum or hydrodynamic regime for which the macroscopic equations of fluid dynamics are applicable.
- $0.01 \lesssim \mathrm{Kn} \lesssim 0.1$: slip flow regime for which fluid dynamics equations boundary conditions should be modified to account for velocity slip or temperature jumps at gas–wall interfaces.
- $0.1 \lesssim \mathrm{Kn} \lesssim 10$: transition regime for which both particle–wall and particle–particle collisions play a significant role and one must model the fluid in greater detail, either through the Boltzmann equation or by extended macroscopic models.
- $\mathrm{Kn} \gtrsim 10$: free molecular regime for which particle–wall interactions dominate compared with particle–particle collisions.

Outside of the continuum or hydrodynamic regime, that is, $\mathrm{Kn} \gtrsim 0.01$, the gases are said to be rarefied [29]. For Knudsen numbers up to about $\mathrm{Kn} \lesssim 1$, macroscopic continuum equations of some form can be used, but these will not be the usual Navier-Stokes and mass continuity equations [29]. The transition regime is probably the most unexplored and difficult regime to model [99, 100].

The Boltzmann equation models the joint distribution $f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t)$ of microscopic particle position $\mathbf{x}$ and velocity $\mathbf{v}$ in a six-dimensional phase space parameterized on time [29, 30]:

$$\frac{\partial f_{\mathbf{x},\mathbf{v}}}{\partial t} + \tilde{v}_i \frac{\partial f_{\mathbf{x},\mathbf{v}}}{\partial \tilde{x}_i} + \frac{\mathrm{d}\tilde{v}_i}{\mathrm{d}t} \frac{\partial f_{\mathbf{x},\mathbf{v}}}{\partial \tilde{v}_i} = Q(f_{\mathbf{x},\mathbf{v}}, f_{\mathbf{x},\mathbf{v}}), \tag{247}$$

where $\mathrm{d}\tilde{\mathbf{v}}/\mathrm{d}t = \mathbf{F}/m$ is the microscopic particle acceleration determined by its equation of motion according to its body and surface forces, and $Q(f_{\mathbf{x},\mathbf{v}}, f_{\mathbf{x},\mathbf{v}})$ is the particle–particle collision integral

*In the case of a continuum, the characteristic length scale L can be written in terms of the local macroscopic gradient of the fastest varying quantity Q [99, 100]:

$$\frac{1}{L} = \left| \frac{\nabla Q}{Q} \right| = \left| \nabla \ln Q \right|. \tag{244}$$

In the case of discrete particles, on the other hand, the length scale is taken as the particle diameter [5, 6]

$$L = d_{\mathrm{p}}. \tag{245}$$

operator. Equation 247 can be derived [30] from the Liouville equations with the "molecular chaos" assumption* for N_p microscopic particles, each in a six-dimensional position $\mathbf{x}_i$ and velocity $\mathbf{v}_i$ phase space for $0 \leq i < N_\mathrm{p}$, and is general enough to model both equilibrium and non-equilibrium flows over the full range of Knudsen numbers—in particular, for rarefied gases. The Boltzmann equation is thus the key equation in the kinetic theory of gases [29].

The macroscopic continuum equations, on the other hand, represent various moments of the Boltzmann equation. In particular, given the distribution $f_{\mathbf{x},\mathbf{v}}$ for particles each of mass m, the (mean) particle mass density and velocity are given, respectively, by [29, 30]

$$\langle C(\tilde{\mathbf{x}}, t)\rangle = m \int f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t)\mathrm{d}\tilde{\mathbf{v}} = m\,\langle n(\tilde{\mathbf{x}}, t)\rangle\,, \tag{248a}$$

$$\langle \mathbf{v}(\tilde{\mathbf{x}}, t)\rangle = \frac{\int \tilde{\mathbf{v}} f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t)\mathrm{d}\tilde{\mathbf{v}}}{\int f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t)\mathrm{d}\tilde{\mathbf{v}}} = \frac{1}{\langle n\rangle} \int \tilde{\mathbf{v}} f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t)\mathrm{d}\tilde{\mathbf{v}}, \tag{248b}$$

where $\langle n(\tilde{\mathbf{x}}, t)\rangle = \int f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t)\mathrm{d}\tilde{\mathbf{v}}$ is the number density. Given a microscopic particle with velocity $\mathbf{v}$, for example, one can calculate its deviation, random, or fluctuating velocity from the mean macroscopic velocity $\langle \mathbf{v}\rangle$ according to $\mathbf{v}' = \mathbf{v} - \langle \mathbf{v}\rangle$. Then taking the zeroth and first velocity moments of Eq. 247 and multiplying by m yields the usual (mean) mass and momentum balance equations [29, 30]:

$$\frac{\partial \langle C\rangle}{\partial t} + \frac{\partial}{\partial x_i} \langle v_i\rangle \langle C\rangle = 0, \tag{249a}$$

$$\frac{\partial}{\partial t} \langle v_i\rangle \langle C\rangle + \frac{\partial}{\partial x_j} \left[\langle C\rangle \langle v_i\rangle \langle v_j\rangle + p_{ij}\right] = \langle C\rangle \frac{F_i}{m}, \tag{249b}$$

where

$$p_{ij} = m \left\langle v_i' v_j'\right\rangle = m \int \tilde{v}_i' \tilde{v}_j' f_{\mathbf{x},\mathbf{v}}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}; t)\mathrm{d}\tilde{\mathbf{v}} \tag{250}$$

is the additional pressure due to random microscopic particle motion analogous to the Reynolds stress term $\left\langle u_i' u_j'\right\rangle$ in the Reynolds-averaged Navier-Stokes (RANS) equation. To solve Eqs. 249, one must obtain a closure for p_{ij} Eq. 250 based on a constitutive relation analogous to the eddy viscosity hypothesis of the RANS equation, that is, $\left\langle u_i' u_j'\right\rangle = 2k\delta_{ij}/3 - \nu_\mathrm{T}\left(\partial \langle u_i\rangle / \partial x_j + \partial \langle u_j\rangle / \partial x_i\right)$. If such a relationship cannot be found, one must return to solving the full Boltzmann Eq. 247. This is indeed the case for the regimes of gas rarefaction for which such constitutive equations are not valid and no general macroscopic theory is possible [30].

*The molecular chaos hypothesis was Boltzmann's key assumption that the velocities of colliding particles are uncorrelated and independent of position. Although somewhat controversial for some time, it is more widely accepted today.