FOI-R--1462--SE
December 2004
ISSN 1650-1942
Scientific report

Håkan Grahn

Modeling of dispersion, deposition and evaporation from ground deposition in a stochastic particle model

NBC Defence
SE-901 82 Umeå

SWEDISH DEFENCE RESEARCH AGENCY
NBC Defence
SE-901 82 Umeå

FOI-R--1462--SE
December 2004
ISSN 1650-1942
Scientific report

Håkan Grahn

Modeling of dispersion, deposition and evaporation from ground deposition in a stochastic particle model

Issuing organization FOI – Swedish Defence Research Agency NBC Defence SE-901 82 Umeå	**Report number, ISRN** FOI-R--1462--SE	**Report type** Scientific report
	Research area code 3 NBC Defence and other hazardous substances	
	Month year December 2004	**Project no.** A4521
	Sub area code 39 Interdisciplinary Projects regarding NBC Defence and other hazardous substances	
	Sub area code 2	
Author/s (editor/s) Håkan Grahn	**Project manager** Lennart Thaning	
	Approved by Åsa Fällman	
	Sponsoring agency FOI	
	Scientifically and technically responsible Lage Jonsson	

Report title

Modeling of dispersion, deposition and evaporation from ground deposition in a stochastic particle model

Abstract (not more than 200 words)

Often when hazardous substances are released in the atmosphere there will be one fraction of pure gas and another fraction of aerosols. The gas cloud called the primary cloud transports a rather long distance with the wind, disperse due to turbulence and deposits on the ground with a specific dry deposition velocity. The aerosols, which also transport with the wind but in a rather short distance, fall down towards the ground where they are deposited. From this ground deposition layer gas evaporates and creates a gas cloud called the secondary cloud.

In this work evaporation from the ground deposition layer has been implemented in a particle dispersion model based on the Langevin equation and furthermore the parametrization of the dry deposition velocity is documented. A verification of the evaporation rate is also made with the conclusion that the the evaporation error is small and acceptable.

Keywords

Stochastic dispersion model, dry deposition velocity, evaporation from ground deposition

Further bibliographic information	**Language** English

Masters Thesisi in Physics, Department of Physics, University of Umeå. Supervisors: Lage Jonsson (FOI), Leif Persson (FOI).

ISSN 1650-1942	**Pages** 42 p.
	Price acc. to pricelist

Utgivare Totalförsvarets Forskningsinstitut - FOI NBC-skydd 901 82 Umeå	**Rapportnummer, ISRN** FOI-R--1462--SE	**Klassificering** Vetenskaplig rapport
	Forskningsområde 3. Skydd mot NBC och andra farliga ämnen	
	Månad, år December 2004	**Projektnummer** A4521
	Delområde 39 Breda projekt inom skydd mot NBC och andra farliga ämnen	
	Delområde 2	
Författare/redaktör Håkan Grahn	**Projektledare** Lennart Thaning	
	Godkänd av Åsa Fällman	
	Uppdragsgivare/kundbeteckning FOI	
	Tekniskt och/eller vetenskapligt ansvarig Lage Jonsson	

Rapportens titel (i översättning)

Modellering av spridning, deposition och avdunstning från markbeläggning i en stokastisk partikelmodell

Sammanfattning (högst 200 ord)

När farliga gaser frigörs i atmosfären bildas ofta ett gasmoln och ett vätskemoln. Gasmolnet transporteras en längre sträcka med vinden och sprids av turbulens samtidigt som gas deponerar på marken med en specifik torrdepositionshastighet. Vätskan faller å sin sida ner som ett regn och deponerar på marken. Från markbeläggningen avdunstar ett sekundärmoln som kan vara skadligt under en längre tid. I en befintlig spridningsmodell baserad på Langevins ekvation har en metod för avdunstning från en markbeläggning implementerats. En verifikation av avdunstningshastigheten är gjord och modellen ger resultat inom felmarginalen. I rapporten har även en parametrisering av torrdepositionshastigheten beskrivits. Beskrivningen kan användas för att beräkna det värde på torrdepositionshastigheten som behövs som indata i varje specifikt simuleringsfall.

Nyckelord

Stokastisk spridningsmodell, torrdepositionshastighet, avdunstning från markbeläggning

Övriga bibliografiska uppgifter	**Språk** Engelska

Examensarbete i fysik, Fysikinstitutionen, Umeå Universitet. Handledare: Lage Jonsson (FOI), Leif Persson (FOI).

ISSN 1650-1942	**Antal sidor:** 42 s.
Distribution enligt missiv	**Pris: Enligt prislista**

Contents

1 Introduction

When hazardous gases are released in the atmosphere often there will be one fraction pure gas and another fraction of aerosols. The gas cloud, called the primary cloud, is transported by the wind and dispersed due to turbulence. As the primary cloud travels with the wind its concentration decreases both due to turbulent dispersion and due to dry deposition on the ground. The aerosols are transported a short distance by the wind and falls to the ground where they are deposited. From this ground depositionlayer gas evaporates and creates a gas cloud called the secondary cloud.

In the current layout of the particle dispersion model used by the Swedish Defence Research Agency (FOI), this secondary cloud is created manually from the possibly huge and irregular ground deposition layer in a time consuming process. Also, the dry deposition of gases and aerosols are not calculated by the model and must be given as manually calculated input. Other aspects the model doesn't take into account are weather changes during the simulation and complex source movements, like releases from airplanes.

The main purpose of this work is to implement an integrated handling of evaporation from the ground deposition layer and explain and document the parameterization of dry deposition needed by the model. Furthermore, changes in the weather during the simulation as well as complex source movements are implemented in the existing dispersion model.

This report is organized as follows. In chapter two the reader is introduced to the simulation model. In the first part of chapter two, a short introduction to the lowest part of the atmosphere, the mixing layer, where the model is applicable, is given. This is followed by a brief description of the dispersion model and then a documentation of how to calculate the dry deposition parameter needed in the dispersion model. In chapter three the old program structure together with the new and revised routines are presented. In chapter four a verification of the model is given and in chapter five a test run is shown. The test run is then discussed in chapter six. A summary of the report is presented together with conclusions in chapter seven. Finally, in section eight, some suggestions for further development of the model are presented.

2 The simulation model

2.1 Boundary layer meteorology

When gases and aerosols are released in the lower part of the atmosphere they are transported by the wind and dispersed due to turbulence. Almost all turbulence dispersion of pollution in the atmosphere occurs in the boundary layer which is the lowest 100 to 3000 meters of the troposphere. The layer of the atmosphere that is closest to the ground. Above the boundary layer the dispersion is governed by large-scale weather systems not dealt with in this report. The part of the troposphere above the boundary layer is often called the free atmosphere, see figure 1. The height of the boundary layer, and hence the volume depends on the degree of turbulence and its generation inside the layer. [17]

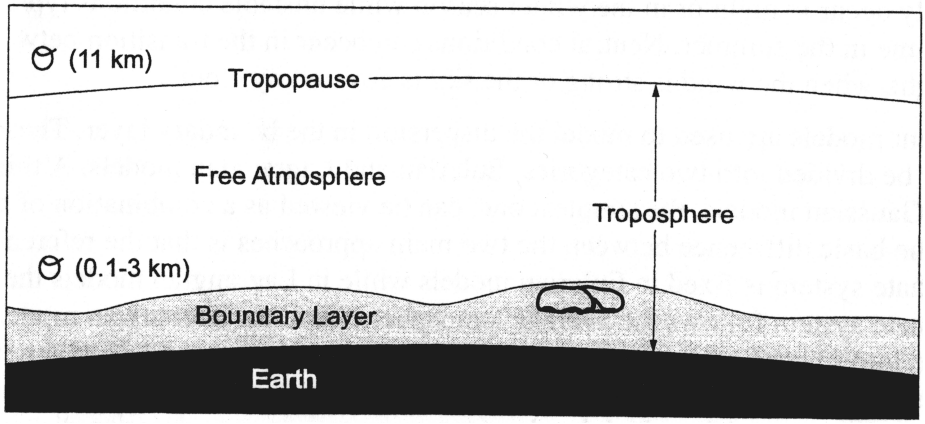

Figure 1. *The troposphere can be divided into two part. A boundary layer near the surface and the free atmosphere above it.[17]*

There are two basic processes that generate turbulence. One is mechanical turbulence generated by the frictional drag on the air flowing over the ground. The other one is the buoyancy. This occurs when an air parcel[1] has a different temperature than the surrounding air thus causing the air to move due to the density difference between cold and warm air. The degree of turbulence in the boundary layer is mainly divided into three broad categories.

The first category is called the *unstable* case which occurs when the temperature is decreasing with height and hence causes thermals of warm air to rise and generate buoyancy turbulence. This effect tends to increase the boundary layer height to well above 1000 meters. The second category is the *neutral* case, where the temperature is nearly constant in the boundary layer and the turbulence is mainly mechanical and weaker. In the last category called the *stable* case the air is colder closer to the ground then at the top and the wind speed is rather low. These effects decrease the boundary layer height.

[1] An imaginary volume of air to which may be assigned any or all of the basic dynamic and thermodynamic properties of atmospheric air.

2.2 The dispersion model

There are a number of different methods to simulate dispersion in turbulent flows e.g., Eulerian, Gaussian and Lagrangian methods. The one used in this work is a Lagrangian particle method based on a probability density function (PDF).

In Lagrangian methods the coordinate system isn't fixed to an inertial frame; instead it is fixed to a *fluid particle* which moves with the local fluid velocity. The fluid particles move continuously, so that the model simulates species concentration at different locations at different times. This could be compared to an Eulerian approach where the species concentration are maintained in an array of fixed computational cells.

The position of a fluid particle is denoted $\mathbf{X}(t, \mathbf{Y})$, where $\mathbf{Y}$ is the position of the particle at a reference time t_0. The velocity is expressed as

$$\frac{\partial}{\partial t}\mathbf{X}(t, \mathbf{Y}) = \mathbf{U}(\mathbf{X}(t, \mathbf{Y}), t). \tag{1}$$

which enlightens the fact that a fluid particle follow the local velocity field. Each fluid particle carries a mass $m = M/N$, where M is the total mass of the pollution and N is the number of particles simulating the pollution.

Langevin Equation

Stochastic differential equations can be useful when ordinary calculus methods fail. This happens for example when the particles velocities undergo diffusion processes which lead to continuous but nowhere differentiable velocity trajectories, see figure 2.2. The types of stochastic differential equations used in this report are the Langevin equations which take the form

$$d\mathbf{U}(t) = \mathbf{a}(\mathbf{U}(t), \mathbf{X}(t), t)dt + b(\mathbf{X}(t), t)d\mathbf{W}(t) \tag{2}$$

$$d\mathbf{X} = \mathbf{U}dt \tag{3}$$

where $\mathbf{U}$ is velocity, $\mathbf{X}$ position, $\mathbf{a}(\mathbf{U}, \mathbf{X}, t)$ is the drift coefficient, $b(\mathbf{X}, t)^2$ is the diffusion coefficient and $\mathbf{W}(t)$ is a Wiener process. In a Wiener process $d\mathbf{W}(t) = \mathbf{W}(t+dt) - \mathbf{W}(t)$ is a normal distributed stochastic variable with mean 0 and variance dt [13]. These types of equations describe a diffusion process and are applicable to particles velocities in turbulent flows. The coefficients $\mathbf{a}$ and b depends on the turbulence model via the Fokker-Plank equation:

$$\frac{\partial}{\partial t}P(\mathbf{U}) = -\frac{\partial}{\partial \mathbf{U}}\left[\mathbf{a}P(\mathbf{U})\right] + \frac{1}{2}\frac{\partial^2}{\partial \mathbf{U}^2}\left[b^2 P(\mathbf{U})\right], \tag{4}$$

where $P(\mathbf{U})$ is the PDF for the particles tracers, i.e. it is the probability that a particle is within the box $[\mathbf{U}, \mathbf{U} + d\mathbf{U}]$. [13]

The Fokker-Plank equation describes how the probability density function for the fluid elements varies with speed and time. Details about the derivation of $\mathbf{a}$ and b when the boundary layer is dominated by buoyancy generated turbulence, unstable case, can be found in Sehlstedt (2000) [15]. For the stable and neutral case, when the boundary layer is dominated by mechanical turbulence due to ground friction, the details of the derivation of $\mathbf{a}$ and b can be found in Schönfeldt (1997) [14].

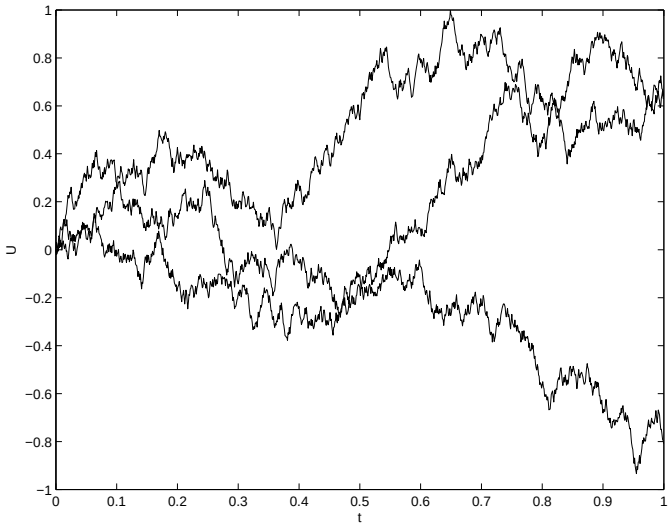

Figure 2. *Three velocities for fluid particles with the same initial conditions under influence by a diffusion process described by the Langevin equation. The coefficients* $\mathbf{a}$ *and* b *are constant.*

2.3 Dry deposition

In the model used in this report the calculation of the dry deposition isn't implemented; instead it is taken as an input to the model, and hence needs to be calculated before the model can start. Dry deposition of particles and gases effect pollution concentration levels in the air, and it is of importance to make an accurate estimate so the model doesn't over- or underestimate the concentration levels.

The purpose with this section is to describe how to calculate dry deposition for a variety of different particles and gases.

Dry deposition refers to the transfer of airborne material, both gases and aerosols, to the ground including vegetation, soil, leafs etc where it is removed from the airborne stream. The transfer process leading to dry deposition can mainly be divided into three processes. The first process involves transport through the atmosphere to the immediate neighborhood of the surface via surface-layer turbulence and is referred to as the aerodynamic component of transfer. [2] The second process is the diffusion of material through the quasi laminar sublayer next to the surface and is dominated by molecular mechanisms. [2] The third process is the absorption of the material and its possible removal from the surface through chemical and biological reactions. This process is called the substrate transfer component. In this final state the absorptivity of the matter at the surface determines how much material that is actually removed during the substrate transfer and deposition process. [2] Dry deposition is in the particle dispersion model modeled by a single parameter v_d (m/s), the deposition velocity.

To calculate the deposition velocity the three exchange processes mentioned are commonly identified as working in series like three resistances in the transport

process, known as the resistance analogy. The deposition velocity is then a measure of conductivity between atmosphere and ground, i.e. the reciprocal of these resistances, $v_d = \frac{1}{R_a+R_b+R_c}$.

Here R_a is the *aerodynamic resistance* determined by the ability of the turbulent eddies to bring material close to the surface. R_b is the *sub-layer resistance* which is the resistance to transfer material across the quasi-laminar sublayer. R_c is the *surface resistance* representing the combined resistance of the surface and transfer to substrate [4].

Dry deposition for gases and aerosols are physically different due to the exchange processes Below are two descriptions of the three resistances, one for gases and the other for aerosols.

2.3.1 Gas deposition

When gases are deposited due to dry deposition in the model, the major causes of resistance generally the sub-layer resistance and the surface resistance. The aerodynamic resistance is often smaller [4]. Figure 3 gives a short description of the resistance analogy.

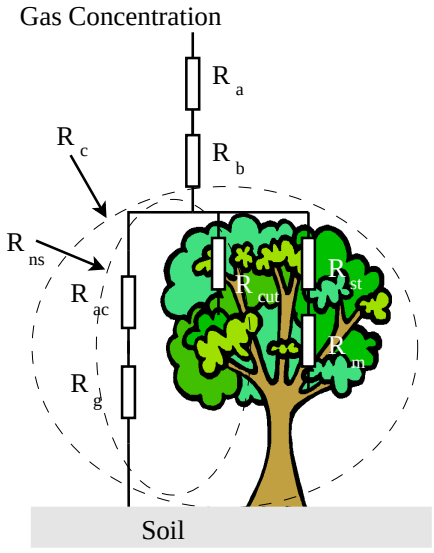

Figure 3. *Scheme of resistance analogy.*

The modeling of the aerodynamic resistance is based on momentum transfer through the atmospheric surface level. According to Arya (1999) [2]

$$R_a = \frac{ln\left(\frac{z_{\text{ref}}}{z_0}\right) - \psi\left(\frac{z_{\text{ref}}}{L}\right)}{\kappa u_*}, \tag{5}$$

where z_{ref} is the reference height for aerodynamic resistance, z_0 the roughness height, ψ the integrated stability function for momentum [4], L the Monin-Obukhovs length defined in Appendix A, κ is von Karman constant (0.4) and u_* is the friction

velocity, also defined in Appendix A. Further more Högström (1988) [5] suggests that the integrated stability function for the momentum can be approximated with the following empirical formulas

$$\begin{aligned} \psi &= -6\tfrac{z_{\mathrm{ref}}}{L}, && \text{for } \tfrac{z_{\mathrm{ref}}}{L} \geq 0 \\ \psi &= \ln\left[\left(\tfrac{1+x^2}{2}\right)\left(\tfrac{1+x}{2}\right)^2\right] - 2\tan^{-1}x + \tfrac{\pi}{2}, && \text{for } \tfrac{z_{\mathrm{ref}}}{L} < 0 \end{aligned} \tag{6}$$

where $x = (1 - 19.3z/L)^{1/4}$.

The equations (6) can be used for most practical applications in which little precision is needed. The approximations are only valid for smooth and moderately rough surfaces, i.e. z_0/L is close to zero. Also note that different authors make different estimates of the integrated stability function. E.g. Arya (1988) [1] estimates a constant of 5 instead of a 6 in the stable case and 15 instead of 19.3 in the unstable case.

The resistance due to molecular diffusion in the thin sublayer near a surface is defined as [7]

$$R_b = 2.0\frac{(\mathrm{Sc})^{2/3}}{\kappa u_*}, \tag{7}$$

where the Schmidt number, Sc, is the ratio of the kinematic viscosity of air, ν, to the particle Brownian diffusivity, D,

$$\mathrm{Sc} = \frac{\nu}{D}, \tag{8}$$

where the Brownian diffusivity, D, is calculated by the Stokes-Einstein equation,

$$D = \frac{k_B T C_c}{6\pi\nu\rho_a r}. \tag{9}$$

Here k_B is the Boltzmann constant, T is temperature, ρ_a is the air density and C_c is the Cunninghan slip correction factor defined as

$$C_c = 1 + \frac{2\lambda}{2r}\left(1.257 + 0.5e^{-\frac{1.1r}{\lambda}}\right), \tag{10}$$

where λ is the mead free path of gas molecules in air ($\lambda = 0.065 \cdot 10^{-6}$ m [4]).

The sublayer resistance increases with increasing surface roughness and decreasing diffusivity of the substance. It is noted that different expressions for R_b are recommended by different authors. For example Hicks *et al.* [3] uses an expression where they divide the Schmidt number by the Prandtl number, which is assumed to be 0.72, thus implying a roughly 24% larger value of R_b.

Dry deposition of gases only takes place as a result of absorption at the surface. Therefore the last resistance, R_c, reflecting the surface properties plays a vital role in the dry deposition process. In a model presented by Zhang [10] R_c is in essence described by

$$\frac{1}{R_c} = \frac{1 - W_{\mathrm{st}}}{R_{\mathrm{st}} + R_m} + \frac{1}{R_{\mathrm{ns}}}, \tag{11}$$

where

$$\frac{1}{R_{\mathrm{ns}}} = \frac{1}{R_{\mathrm{cut}}} + \frac{1}{R_{\mathrm{ac}} + R_g}, \tag{12}$$

where $W_{\rm st}$ is the fraction of stomatal[2] blocking under wet conditions. $R_{\rm st}$ is the stomatal resistance calculated using a sunlit/shade stomatal resistance model [11], where the canopy is subdivided into sunlit leaves and shaded leaves and the resistance is calculated as a function depending on, among other things, temperature, vapour pressure, photosynthesis etc. The mesophyll[3] resistance R_m depends only on the chemical species and data for it can be found in table 1 [11]. $R_{\rm ns}$ is non stomatal resistance which is further decomposed into resistance due to cuticle[4] uptake, $R_{\rm cut}$, and into resistance to soil uptake including the chemical independent in-canopy aerodynamic resistance, $R_{\rm ac}$, and the soil resistance, R_g. The equations (11) and (12) are only for surfaces with canopies. For surfaces without canopies like water, ice and desert $R_{\rm st}$, R_m, $R_{\rm ac}$ and $R_{\rm cut}$ would be inapplicable, but for convenience the same equations are used for all types of surfaces. The only difference when applying the equations to surfaces without canopies is that $R_{\rm ac}$ equals 0 and $R_{\rm st}$, R_m and $R_{\rm cut}$ are assigned a very large value (i.e. 10^{25} s m^{-1}).

According to Zhang (2003) [10] $W_{\rm st}$ is only important during situations when the sun is shining on a wet ground. Typical examples of this are sunny mornings with dew and sunshine immediately after rain. Thus the following formulas are suggested for wet canopies (for dry canopies, $W_{\rm st}$ always equals 0):

$$W_{\rm st} = \begin{cases} 0, & {\rm SR} \leq 200 \text{ Wm}^{-2} \\ ({\rm SR}-200)/800, & 200 < {\rm SR} \leq 600 \text{ Wm}^{-2} \\ 0.5, & {\rm SR} > 600 \text{ Wm}^{-2} \end{cases} \tag{13}$$

where SR is the solar radiation.

According to [10]

$$R_{\rm ac} = \frac{R_{\rm ac0}{\rm LAI}^{1/4}}{u_*^2}, \tag{14}$$

where

$$R_{\rm ac0}(t) = R_{\rm ac0}({\rm min}) + \frac{{\rm LAI}(t) - {\rm LAI}({\rm min})}{{\rm LAI}({\rm max}) - {\rm LAI}({\rm min})} \cdot [R_{\rm ac0}({\rm max}) - R_{\rm ac0}({\rm min})] \tag{15}$$

$R_{\rm ac0}$ can be found in table 1 [10], LAI is leaf area index and can be found in table 2 in Zhang (2002)[11].

R_g and $R_{\rm cut}$ are calculated for SO_2 and O_3 and then scaled for other gaseous species based on the formulas [10]

$$\frac{1}{R_{\rm cut}(i)} = \frac{\alpha}{R_{\rm cut}({\rm SO_2})} + \frac{\beta}{R_{\rm cut}({\rm O_3})} \tag{16}$$

and

$$\frac{1}{R_g(i)} = \frac{\alpha}{R_g({\rm SO_2})} + \frac{\beta}{R_g({\rm O_3})}, \tag{17}$$

where the scaling parameters $\alpha(i)$ and $\beta(i)$ can be found in table 1 in Zhang (2002) [11] for the different substance, i.

[2]Small pores in the outer layer of a leaf or stem through which gases and water vapour pass. Also called stomata.

[3]The photosynthetic tissue of a leaf.

[4]A layer of wax like, water-repellent material, *cutin*, covering the epidermis.

2.3.2 Particle deposition

In the parameterization of particle dry deposition presented in this report R_c is set to zero. This is based on the assumption that all particles hitting a surface stick to it [4]. Furthermore the gravitational settling velocity is included in the expression for dry deposition velocity. This leads to the following expression for particle dry deposition

$$V_d = V_g + \frac{1}{R_a + R_b}. \tag{18}$$

The gravitational settling velocity is calculated numerically according to

$$V_g = \sqrt{\frac{8 \cdot g r_p(\rho_p - \rho_a)\rho_p}{3 \cdot C_D \rho_a}} \tag{19}$$

$$\mathrm{Re} = 2\frac{V_g r}{\nu} \tag{20}$$

$$C_D = \frac{24}{\mathrm{Re}(1 + 0.173 \cdot \mathrm{Re}^{0.657})} + \frac{0.413}{1 + 16300 \cdot \mathrm{Re}^{-1.09}}, \tag{21}$$

where ρ_p and ρ_a is the density of the particle and the density of air respectively, g is the acceleration of gravity, r is the particle radius, C_D the drag coefficient, ν is the kinematic viscosity of air and Re is the Reynolds number. Notice that the settling velocity, the Reynolds number and the drag coefficient are dependent on each other. Several iterations of the equation system are therefore needed to calculate the settling velocity. For a rough estimation of the settling velocity

$$V_g = \frac{\rho_p (2r)^2 g C_c}{18\eta} \tag{22}$$

may be used as an estimation. [9] Here η is the viscosity coefficient of air and C_c is the slip correction factor (Cunninghan correction factor) for small particles, see (10).

The aerodynamic resistance, R_a, is calculated in the same way as for dry deposition of gases, see (5). The surface resistance in (22), R_b, depends on several deposition processes, particle size, atmospheric conditions and surface properties [9]. R_b is parameterized by

$$R_b = \frac{1}{\epsilon_0 u_* (E_{\mathrm{B}} + E_{\mathrm{IM}} + E_{\mathrm{IN}}) R_1}, \tag{23}$$

where E_{B}, E_{IM}, E_{IN} are collection efficiencies from Brownian diffusion, impaction and interaction, respectively; R_1 is a correction factor representing the fraction of particles that stick to the surface. ϵ_0 is an empirical constant and is assumed to be 3. [9]

The collection efficiency due to Brownian diffusion is a function of the Schmidt number given as

$$E_{\mathrm{B}} = \mathrm{Sc}^{-\gamma}. \tag{24}$$

The constant γ varies with land type categories and is value between $1/2$ and $2/3$ [16].

The impaction process, E_{IM}, is controlled by the Stokes number, St, which has the form $\mathrm{St} = V_g u_*/gA$ for vegetated surfaces and $\mathrm{St} = V_g u_*^2/\nu$ for smooth

surfaces. A is a characteristic "radius" of large collectors, e.g. grass blades, stalks, needles etc.

Different expressions for the impaction process are suggested for different types of surfaces. According to Slinn (1982) [16] smooth surface gives

$$E_{\mathrm{IM}} = 10^{-3/\mathrm{St}} \tag{25}$$

and for vegetative canopies the collection efficiency for impaction is

$$E_{\mathrm{IM}} = \frac{\mathrm{St}^2}{1+\mathrm{St}^2}. \tag{26}$$

Impaction efficiency over a spruce forest is according to Peters and Eiden (1992) [6]

$$E_{\mathrm{IM}} = \left(\frac{\mathrm{St}}{\alpha+\mathrm{St}}\right)^{\beta}, \tag{27}$$

where α and β are constants. A good choice for β is 2 according to Peters and Eiden (1992) with α depending on land type category. α varies between 0.8 [6] and 2 for surfaces with canopies. For surfaces without canopies the impaction efficiency is neglectable and hence α has a rather large value between; 50-100 [9].

When a particle passes an obstacle at a distance shorter than its own diameter the mechanisms included in the collection efficiency by interception is of importance. This is especially true for large particles over hairy leaves. Interception deposition during particle flow around surfaces as in canopies of coniferous trees is negligible compared to the other deposition mechanisms. Here the collection efficiency by interception is defined as

$$E_{\mathrm{IN}} = \frac{1}{2}\left(\frac{2r}{A}\right)^2 \tag{28}$$

where r is the radius of the particles and A is defined as a characteristic radius of large collectors. Typically A is 2 for evergreen-needleleaf trees and 10 for urban terrain.

At last the factor R_1, which represent the fraction of particles sticking to the surface in (23), can be simplified to [16]

$$R_1 = e^{-\mathrm{St}^{1/2}}. \tag{29}$$

2.3.3 Implementation of deposition

The mass flux to the surface is defined by the dry deposition velocity concept

$$F_m = C v_d, \tag{30}$$

where C is particle concentration $(1/m^3)$ near the surface. This is a cumbersome approach in the Langevin model since each particle is treated individually and the concentration is therefore unknown to each simulation particle. To overcome this problem a deposition probability is derived [8]

$$P = \frac{F_m}{F_p} = (2\pi)^{1/2}\frac{v_d}{\sigma}, \tag{31}$$

where σ is the standard deviation of a Gaussian velocity distribution and F_p is mass flux through the surface due to a Gaussian velocity distribution,

$$F_p = \int_{w<0} d^3V f(V) V \cos(\theta) = \frac{C\sigma}{(2\pi)^{1/2}}. \tag{32}$$

Where $f(V)$ is the Gaussian velocity distribution with zero mean and standard deviation σ, θ is the angle between the particle velocity and the z-axis and the integration boundary, $w < 0$, means integration over all velocities with negative z-component. The probability that an individual particle is deposited when it hits the surface is proportional to the deposition probability.

2.4 Evaporation from ground deposition

In the implementation of evaporation from the ground deposition the probability for each individual particle to evaporate is calculated during each time step. The probability for evaporation from one particle, P_{evap}, in the time interval $\Delta t = t_1 - t_2$ is defined as

$$P_{\text{evap}} = \text{pppfact} \cdot \int_{t_1}^{t_2} \frac{d\text{Fr}_{\text{re}}}{dt'} dt', \tag{33}$$

where pppfact, *particle per particle factor*, is the factor with unit activity per particle that relates the number of deposited particles to the number of evaporated particles and $\frac{d\text{Fr}_{\text{re}}}{dt}$ $(\%/s)$ is the evaporation. $\frac{d\text{Fr}_{\text{re}}}{dt}$ is defined as

$$\frac{d\text{Fr}_{\text{re}}}{dt} = \frac{\dot{m(t)}}{M(t)}, \tag{34}$$

where $\dot{m}$ is the change in mass and M is the total mass remaining after the particle is deposited. However, the evaporation is often given in terms of $\frac{\dot{m(t)}}{M_0}$ and F_{ev}, where M_0 is the initial mass and F_{ev} the fraction evaporated. This implies that some modifications to (33) must be done to use these values as inputs. Since the remaining mass is equal to the initial mass minus the evaporated mass, i.e.

$$\begin{aligned} \frac{\dot{m(t)}}{M(t)} &= \frac{\dot{m(t)}}{M_0 - \int_{t_0}^{t} \dot{m(t)} dt'} \\ &= \frac{\frac{\dot{m(t)}}{M_0}}{1 - \int_{t_0}^{t} \frac{\dot{m(t)}}{M_0} dt'}, \end{aligned} \tag{35}$$

the (34) can also be written as

$$P_{\text{evap}} = \text{pppfact} \cdot \int_{t_1}^{t_2} \frac{d\text{Fr}_{\text{re}}}{dt'} dt' = \text{pppfact} \cdot \int_{t_1}^{t_2} \frac{\frac{\dot{m(t)}}{M_0}}{1 - F_{\text{ev}}(t)} dt' \tag{36}$$

The values of $\frac{\dot{m(t)}}{M_0}$ and F_{ev} are based on a model for evaporation from a glass surface used by the model *GASSY* [12] which renders a rather imprecise estimation of the real values in some situations. To make the model more independent from this approximation $\frac{\dot{m(t)}}{M_0}$ and F_{ev} are read from a table stored in a separate data file which easily could be updated.

3 Software functions

In this section an introduction to the main flow in the model and some of the developed algorithms are presented. The purpose is to give a short description of the software in general and to explain the advantages achieved with the new subroutines and discuss some of the decisions made during the development process.

In figure 4 a schematic picture of the existing Fortran90 software is presented together with both new and old routines. Two entirely new routines, `evaporate_density` and `init_new_particle`, are constructed.
`Evaporate_density` calculates the number of new evaporated particles from each deposited particle and `init_new_particle` generates the new particles. Some of the modified routines are `out_ptp`, `movepar_ana` and `movepar_ana_convective`.

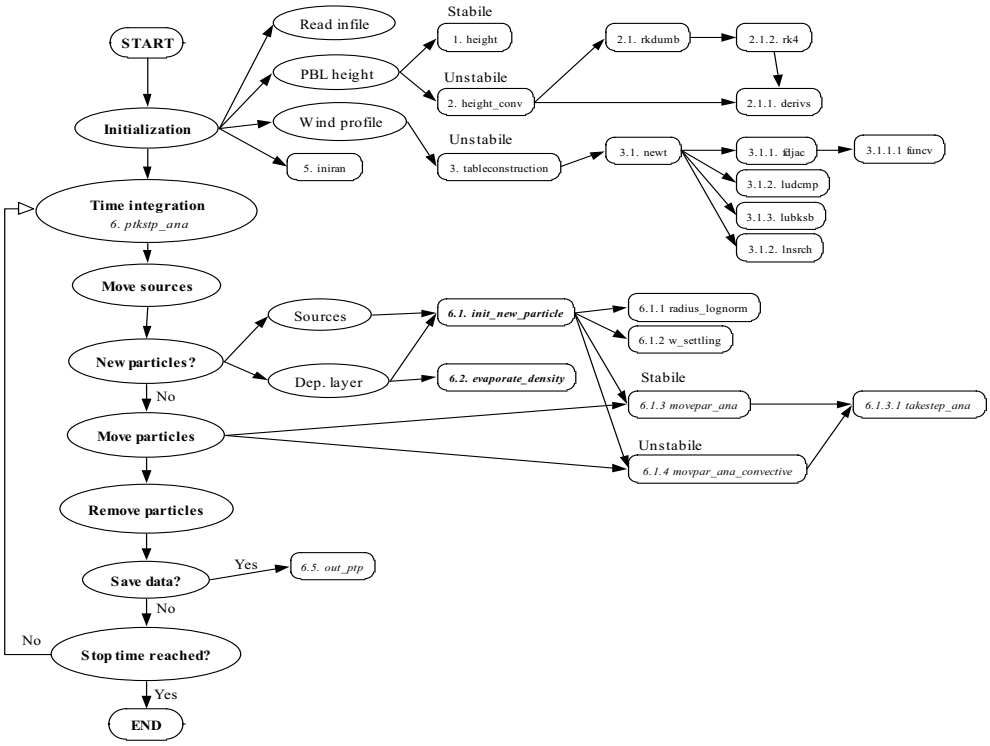

Figure 4. *A schematic picture over the structure of the program with its subroutines and functions. The bold and italic text marks new routines and the italic text marks improved old routines.*

The reason for modifying these routines are mainly due to the introduction of a derived-type structure in the `loop_ana` routine. In Fortran a derived-type structure is a variable that contains other variables called attributes. The derived-type in `loop_ana` controls the particles attributes, e.g. position, velocity, size etc.

The execution of the program starts in the main function `ptk_move_ana` which reads the input variables from the data file `input_ana.nml` and thereafter calculates boundary layer height and aligns the x-axis along with the wind

direction. The main function then calls the routine `ptkstp_ana`, which controls the simulation. This routine consists of a main loop over the total time during which a particle will be moved and three sub loops inside this main loop, the "distribution loop", the "evaporation loop" and the "old particle movement loop".

Before the main loop in `ptkstp_ana` reaches any of the sub loops a time interval during which particles will be moved is decided and all source movements due to waypoints given by the user are made. A waypoint is a position where a source is supposed to be at a given time, and hence the waypoints define a route for the sources. The time interval is adjusted so that no source-start or -stop time or output time as decided by the user is missed. When a time interval is set and all source movements are done `ptksp_ana` reaches the "distribution loop" which is a loop over the sources. In this source loop the total number of particles that is going to be released during the time interval is decided.

When this is known all the new particles are released using the new routine `init_new_particle`. The routine decides initial position, particle radius and deposition velocity by calling `radius_lognorm` and `w_settling`, and then moves the new particles during the time interval by calling either `movepar_ana` if the weather conditions simulated are neutral or stable, or `movpar_ana_convective` if the weather conditions are unstable.

The next loop entered is the "evaporation loop" which is a loop over all the evaporable deposited particles. The purpose of this loop is to decide the number of evaporated particles in the given time interval. For each deposited particle an evaporation probability is determined depending on the time since the particle deposited and the evaporation rate which is tabulated in the input file `evaporation data`. Then the routine `init_new_particle` generates the new gas particles.

Thereafter the rest of the particles that haven't been moved yet are moved in the "old particle movement loop" which moves them with the `movepar_ana` routines. In the last part of the main loop the routine `out_ptp`, where the particles state are written to files, is called every time the actual time coincides with the output time interval.

The routine that calculates the evaporation probability in the "evaporation loop" mentioned above is `evaporate_density`. The routine takes a deposited particle and calculates an evaporation probability by integrating $\frac{dRe}{dt}$ according to equation (33). The integration is performed with a variant of the trapezoidal rule which is a fast method with limited accuracy, especially when a small number of data points are used. However the error caused by the trapezoid rule will be a little bit smaller than expected. The reason for this is that the fewer particles left on the ground, the harder it is to maintain a constant mass flow from the deposition layer. Therefore, a temporary overestimation of the integrand, will at first overestimate the mass flow but later on, when the number of deposited particles has decreased sufficiently much the mass flow will be underestimated. The net mass flow is therefore a little bit lower than suggested by the trapezoid rule. The opposite occurs if the integrand is underestimated.

3.1 New and improved functions

The changes in the program for previous versions are summarized below:

- Modeling of the evaporation for deposited particles is automated which is a major improvement.
- A separate file is created at the end of the simulation so another simulation can continue where the old one stopped by reading all the old particle data from that file. This is useful for changing weather condition during a simulation. It can be done by editing the input data between different simulations.
- The wind direction can change to an arbitrary direction during the simulation.
- The sources can move in any direction throughout the entire simulation. This is made possible by specifying discrete trajectories for the sources. The trajectories are defined by points in time and space, here called waypoints.

3.2 Program flowchart and main structure

Following is a short description of the main algorithm in the program.

1. In `ptk_move_ana`
   - 1.1 Read input variables from `input_ana.nml`
   - 1.2 Calculate boundary layer height.
   - 1.3 Align x-axis along the wind direction.
2. Call ptkstp_ana
   - 2.1 Read the rate of evaporation as a function of time from input table given in `input_ana.nml`
   - 2.2 Start the main-loop
     - 2.2.1 Decide the current time interval so that no source start or stop time is missed and so that each output time given by the user occurs.
     - 2.2.2 Calculate source movement and move them according to the given Waypoints.
     - 2.2.3 Start distribution loop
       - 2.2.3.1 Loop over each source and calculate the number of new particles released from each source during the given time interval.
       - 2.2.3.2 Initiate the new particles with the function `init_new_particle`
       - 2.2.3.3 Call `radius_lognorm` and `w_settling` to calculate particle radius and settling velocity.
       - 2.2.3.4 Call a `movepar_ana` routine to move the particle.
     - 2.2.4 Start evaporation loop.
       - 2.2.4.1 Loop over each deposited particles and check the rate of evaporation during the time interval with the function `evaporate_density`.

2.2.4.2 Integrate the rate of evaporation to calculate the number of new gas particles from the deposited particles.

2.2.4.3 Loop over the newly evaporated particles and initiate them with the `init_new_particle` function.

2.3 Start the old particle movement loop

2.3.1 Loop over each old particle that hasn't been moved yet.

2.3.2 Call some of the `movepar_ana` routines to move the old particles.

2.4 Call outptp to save the particle information to files.

3. Finally create a data file that contains information such as how the simulation was set up, simulation time etc.

4 Model verification

In this section a verification of the model is presented and evaluated. The major evaluation is to confirm that the evaporation from the ground deposition layer is consistent with the evaporation given in the input. Further, since no changes are made in the dispersion model the simulated dispersion has to be unaffected by the new implementations. This is confirmed by comparing the results obtained by the newly developed model with old results.

The first verification done is an evaluation of the evaporation from the ground deposition layer. A number of simulations are done with different substances evaporating from a ground deposition layer during different weather conditions. The evaporation is then compared to the evaporation given by GASSY [12]. GASSY was the theoretical model used to estimate the evaporation rate in the dispersion model during the verification.

The simulated substances are VX and sarin. VX is chosen because of its low volatility and hence long evaporation time and sarin is chosen because it's high volatility and short evaporation time. Both impure and pure variants of the substances are tested. Each substance was simulated with two different $\mathrm{pppfact}$ factors and with different purities in two different weather situations, see table 4. The $\mathrm{pppfact}$ factor specifies the relationship between the source's mass and the number of evaporated particles. If 5 particles are deposited representing 5 kg and $\mathrm{pppfact}$ is chosen to 10, then 50 particles will evaporated from these 5 deposited particles. Each deposited particles will represent a mass of 1 kg and each evaporated particle will represent a mass of 0.1 kg. The $\mathrm{pppfact}$ factor is especially useful when the substance has a very low volatility and more particles are needed to improve the statistic.

Table 1. Values of the different parameters that are used as input to the model for the four different weather types tested.

Substance	Wind (m/s)	Temperature (°C)	Purity (%)	Particle diameter (mm)
VX	0.57	20	100	0.3
VX	1.0	20	50	0.3
Sarin	0.3	13	100	0.3
Sarin	4	13	50	0.3

The results of the evaporation test can be seen in figure 5 and table 4. In each picture there are three different graphs. One with $\mathrm{pppfact}$=1, one with $\mathrm{pppfact}$=10 and the third one is the evaporation given from the GASSY model. Our model tends to slightly overestimate the evaporation compared to GASSY, especially when the substances are impure. For pure substances the error is below 2.2 % in the end of each simulation. In all the four cases a higher $\mathrm{pppfact}$ factor improves accuracy of the model.

The evaporation overestimation is due to the choice of integration scheme. Between each data point a linear fit is made and this approximation of the curve is then integrated, and this leads to the same result as if the trapezoid rule should have been used as integration scheme. The trapezoid rule,

$$\int_{x_1}^{x_2} f(x)dx = \frac{x_2 - x_1}{N}\left[\frac{1}{2}f(x_1) + \frac{1}{2}f(x_2)\right] + O\left(\frac{(x_2 - x_1)^3}{N^3}f''\right), \quad (37)$$

where N is the number if steps, is indeed fast but inaccurate. This is especially true in this case since the known points are few, and in the impure cases unevenly

distributed. Further the integrand is a monotonic decreasing or increasing function and hence the second derivative is non zero, see equation (36). The maximal theoretical relative error due to the trapezoid rule when the substance is VX with 50 % purity is about 9%. This should be compared to the error shown in figure 5 which is about 7%, see table 4.

Table 2. The relative error in evaporation compared to GASSY for different substance, purity and pppfact factor.

Substance	Purity (%)	pppfact	Mass evaporated (%)	Relative error (%)
VX	100	1	90	2.2
VX	100	10	90	2.0
VX	50	1	90	6.5
VX	50	10	90	6.4
sarin	100	1	90	1.7
sarin	100	10	90	2.1
sarin	50	1	90	7.7
sarin	50	10	90	6.4

The second verification of the model is the comparison with simulations done before any changes were made to the model. For that purpose four simulations including the new developments were compared with previous simulations made by Näslund et al [7]. They did two scenarios, a chemical attack with VX, and a chemical attack with sarin. The attacks were supposed to take place in a typical Swedish forest with a roughness factor of 1.0 meter. The chemical substances were released with shells exploding about 10 meters above the ground, creating both a primary and a secondary cloud. In table 4 and table 4 the amount of gas and liquid in the initial state is presented together with the assumed weather.

Table 3. Weather, initial airborne part of 2160 kg sarin. Drop diameter 0.3 mm. Ground contamination area 1 km^2. L is Monin-Obukhovs length and u_* is friction velocity, both explained in appendix A

Weather	L (m)	u_* (m/s)	Initial airborne part
+10 - +15°C, wind 1m/s, stable stratification	350	0.170	1296 kg (60%)
+10 - +15°C, wind 4 m/s, neutral stratification	$1 \cdot 10^6$	0.695	1296 kg (60%)

Table 4. Weather, initial airborne part of 1520 kg VX. Drop diameter 0.3 mm. Ground contamination area 800 × 600 m. L is Monin-Obukhovs length and u_* is friction velocity, both explained in appendix A

Weather	L (m)	u_* (m/s)	Initial airborne part
+10 - +15°C, wind 1m/s, stable stratification	350	0.170	152 kg (10%)
+10 - +15°C, wind 4 m/s, neutral stratification	$1 \cdot 10^6$	0.695	152 kg (10%)

In figure 6 the effects of the chemical attacks are plotted. Note the difference in resolution between the sarin pictures and the VX pictures. This is due to the difference in the number of simulation particles used for VX and sarin. With sarin gas more particles evaporates from the ground deposition and hence a higher resolution is chosen. The levels, or risk distances, in the figure are specified with LD_{50}, LD_{05} and ED_{05} values where ED and LD are short for effect dosage and lethal dosage. The subscript 50 stands for percent and can be interpreted as the

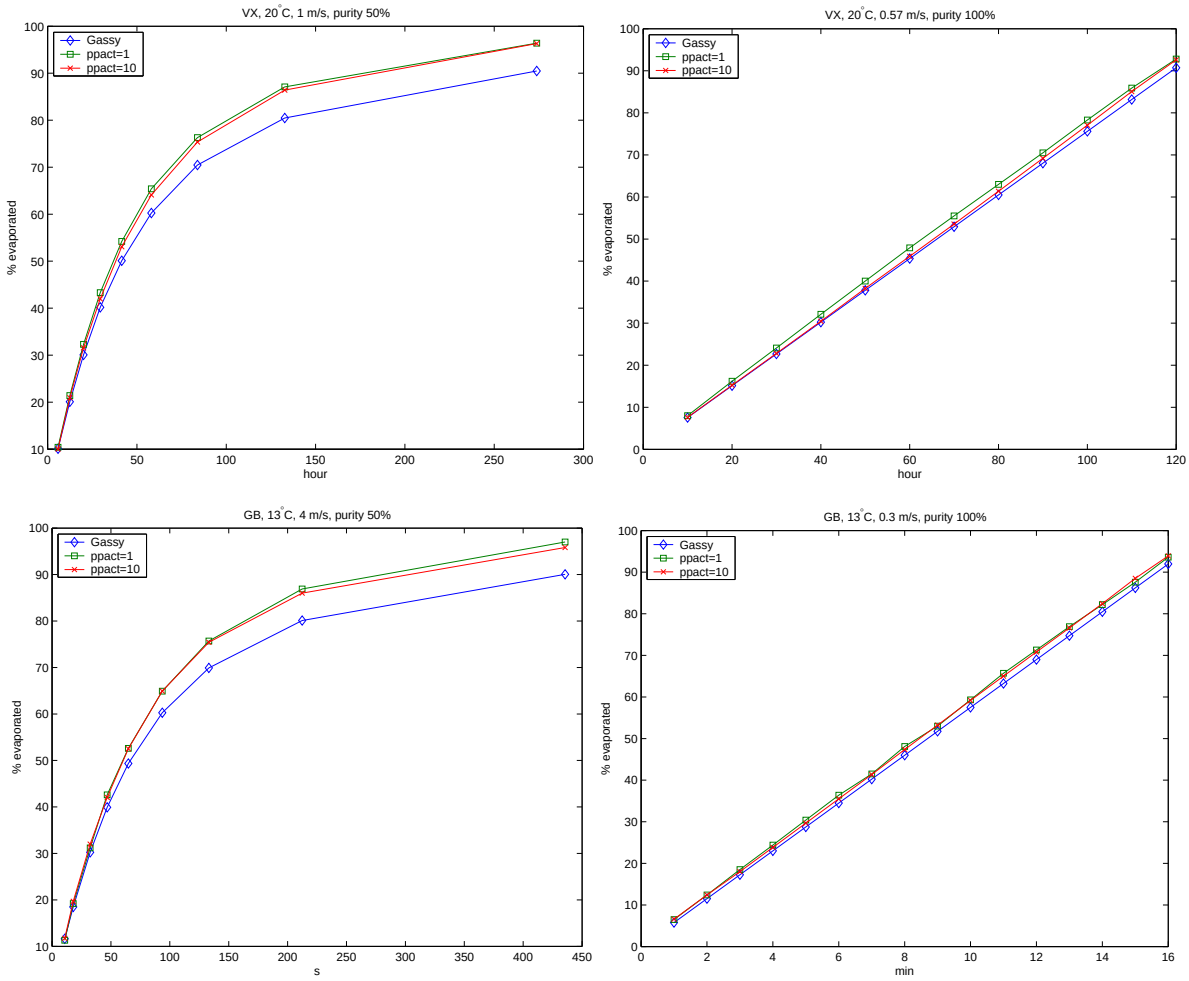

Figure 5. *Percentage evaporated plotted against time. The green line with squares is evaporation with* pppfact*=1, the red line with x:s is evaporation with* pppfact*=10 and the blue line with diamonds is evaporation due to the GASSY model.*

share of people that would show these kind of symptoms if they where exposed to this concentration during, in this case, one hour.

In a comparison of the risk distances attained by Näslund et al [7] who used the old model the LD_{50} and LD_{05} risk distances for VX are shorter when they are calculated with the improved model. The ED_{05} risk distances are on the other hand longer when compared to Näslunds calculations. For sarin the LD_{50} risk distances are shorter, but the LD_{05} and ED_{05} risk distances are longer. These differences are probably due to differently assumed injury levels. See table 4 for a comparison between the two models.

The verification of the model shows that the transport and dispersion model are unaffected by the new developments. Also, when comparing the model to [12] the difference in evaporation from ground deposition is small and below the theoretical error. This means, since the ground evaporation now is implemented, that the revised model offers a significant decrease in preparation time needed to

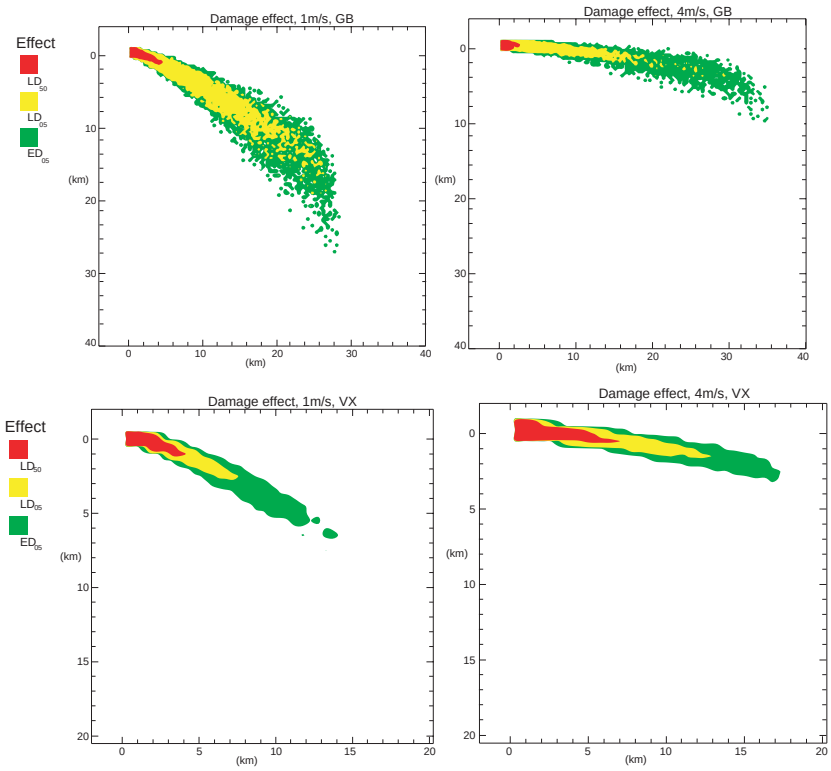

Figure 6. *The two pictures on the top shows damage effect from an attack with 2160 kg sarin during summer conditions. The two pictures on the bottom show damage effect from an attack with 1520 kg VX during summer conditions. Red colors indicates areas with 50% deadly wounded, yellow areas means 5% deadly wounded and green areas means 5% wounded.*

do a simulation.

Table 5. Estimation of risk distance with the model used in this work and previous models.

Substance	Wind speed (m/s)	Dosage	Distance old model (km)	Distance new model (km)	Relative difference (%)
VX	1	LD_{50}	9	5	44
VX	1	LD_{05}	11	8	27
VX	1	ED_{05}	14	15	7
VX	4	LD_{50}	7	6	14
VX	4	LD_{05}	9	10	10
VX	4	ED_{05}	12	20	67
sarin	1	LD_{50}	9	5	44
sarin	1	LD_{05}	20	25	25
sarin	1	ED_{05}	30	36	20
sarin	4	LD_{50}	6	3	50
sarin	4	LD_{05}	17	20	18
sarin	4	ED_{05}	31	38	23

5 Result of Test Simulation

In the preceding section the model were evaluated and compared to previous results. In this section the models capability of running complex scenarios will be demonstrated. As an exercise for a NBC-management team, a good scenario which includes most of the models functionality was constructed in the spring 2004.

An airplane with a load of 300 kg liquid soman unleashes its load over Umeå. The load is released along a path from Rådhuset to the railroad station where the plane turns left along E12 towards the viaduct where the E4 crosses the E12. On this viaduct the entire load has been released and hence the release stops. From the ground contamination caused, gas evaporates and a hazardous secondary cloud is created. The time of the scenario is three hours starting with the 30 seconds flight followed by a three our long evaporation process. A description of the weather during the scenario is given in table 5.

Table 6. Weather during the scenario.

time	weather
12:00-12:55	wind from east 3m/s, cloudy, 12 °C
12.55-13.05	wind from east 2m/s, partly cloudy, 20°C
13:05-14:00	wind from east 2m/s, sunny, 25 °C
14:00-15:00	wind from west 3m/s, sunny, 25 °C

In the simulation two different types of atmospheric stabilities are simulated. Changes in the wind speed and its direction are also included. Together with the source movements in different directions this illustrates all the improvements made of the model i.e., the scenario tests the models capability of handling evaporation from the ground deposition layer, weather changes and complex source movements.

As can be seen in figure 5, the ground concentration in the first hour when the stratification is stable is higher further away than it is during the two last hours when the stratification is unstable. This is the case despite the temperature increase and hence a faster evaporation in the last two hours. At 13.30 when the sun has heated the ground the air concentration raises again compared to the concentration at 13.00. The wind turn at 14.00 spreads the secondary cloud and unhealthy concentration levels occurs further away from the deposition layer. When the wind increases at 14.00 the concentration levels decreases again as can be seen in the picture from 14.30. At 15.00 the ground deposition layer starts to vanish leading to a lower concentration.

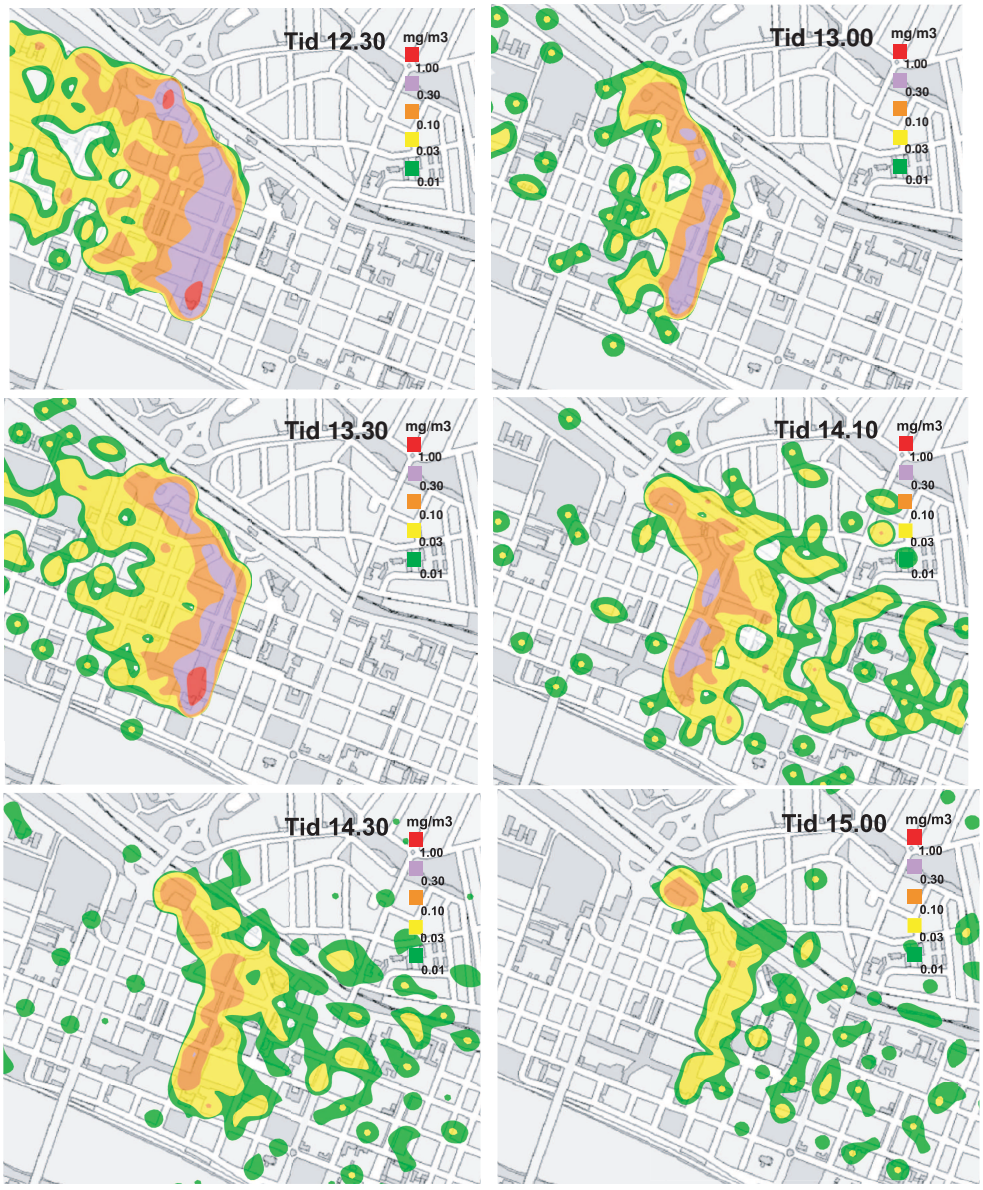

Figure 7. *The figures show different concentration levels at six different times during an attack against Umeå centre with 300 kg liquid soman. Risk for unhealthy effects can occur when the concentration levels are higher then 0.03 mg/m^3 and deadly effects can occur when the concentration is higher then 0.3 mg/m^3. Notice that, how the concentration affects humans is dependent on exposure time. A longer exposure time increases the risk of severe wounds.*

6 Discussion

Bellow is discussed what the model manage to predict in the test simulation (section 5) and what it doesn't predict.

Just like the old model, the model predicts that the highest concentration is during the first hour when the boundary layer height is low and hence the part of the atmosphere where the gas is able to mix is small. The model then predicts the concentration increase between 13.00 and 13.30 which exists due to the temperature increase on the ground. When the wind increases the last hour more air passes the ground deposition layer in a shorter time and hence is the secondary cloud released into a larger volume. This effect is larger than the increase in evaporation rate and hence does the concentration decrease. But this concentration decrease is also an effect of the decreasing amount of soman in the ground deposition layer.

One can notice is that the model neglect individual streets and houses. Instead one average value, called the roughness factor, is calculated as a function of the obstacles heights and used by the model. If the model hadn't neglected objects on the ground, the concentration would have been higher on the streets and some part of the gas might even have traveled a short distance in the opposite wind direction due to whirls around buildings. But as an average estimation of the concentration levels the model should be able to reasonably well predict the airborne concentration of the C-agent.

Furthermore, the evaporation rate is approximately proportional to the temperature, but the model has a constant ground temperature all over the region as long as the air temperature is the same. Hence the evaporation rate is constant in the entire region. In reality different sections on the road have different temperatures due to different heat capacity, color and received solar radiation. Therefore, small errors in the concentration level arise on regions colder and warmer compared to the average ground temperature.

Beside the effects of the homogeneous wind field and temperature, the statistics is also an important factor. In this simulation it is not good enough to predict the geometry and concentration of the gas cloud when the concentration is close to zero. The smallest concentration areas, green and yellow dots, that can be seen in figure 5 consists of single simulation particles. This makes it hard to estimate the concentration and geometry of the gas cloud in these areas due to the lack of statistic. To make a better estimation of the secondary cloud the number of simulation particles has to increase by a factor of ten or more, or several simulations with different seeds to the random number generator must be done to improve the statistics. Notice that the $\mathrm{pppfact}$ was chosen to 10 during the simulation and hence the number of gas particles is 10 times more then the number of fluid particles in the ground deposition layer. Without this choice of $\mathrm{pppfact}$ the number of simulation particles in the contaminated area would have been huge and the simulation time would have increased significantly to arrive at the same result.

But the major drawback of the simulation is the lack of an estimation of the error. Without a good error estimation it is hard to trust the results from the model since the transport and dispersion of the gas depends on several uncertain input parameters like the wind speed, wind direction, boundary layer height etc. An easy way to achieve this would have been to make several simulations with small variations in the input parameters, and from the results calculate mean and standard deviation.

7 Summary and Conclusions

In this master thesis a dispersion model based on a Lagrangian particle method has been enhanced with an integrated handling of evaporation from a ground deposition layer. Further more an explanation of the parametrization of dry deposition velocity for both gases and particles is given. The goals i.e, to simplify the preprocessing needed to start the calculations in the dispersion model, are fulfilled. This is done by both the evaporation implementation and the given documentation of the dry deposition velocity.

In addition to those changes the possibility of complex source movements and weather changes during a simulation are implemented in the model.

Two main verifications of the model have been done. First the evaporation rate was compared to the GASSY [12] model. The results from that comparison were good and all the results were inside the estimated theoretical error. The second verification was a dispersion comparison between the revised model and the old model. The disagreement was acceptable and was probably due to differently assumed injury levels.

Finally, a scenario consisting of complex source movements, weather changes, dry deposition and evaporation from the ground deposition layer was constructed and tested. The model performed well and behaved as expected.

8 Future development

The current implementation of the particle dispersion model has some disadvantage like the horizontally homogeneous wind field and the disability to specify the roughness parameter in more than one direction. Future work on the model could investigate the possibility to include both a wind grid and a roughness grid.

Other possible improvements of the model could be made in the post processing of data obtained from the model. Often the goal with a simulation is to estimate the concentration at a given position (or more likely, many positions), $\mathbf{x}$, from a set of calculated particle positions. Since the concentration at the point $\mathbf{x}$ is either zero or infinite, it is inevitable that the estimate must involve particles in the vicinity of $\mathbf{x}$. In the current model this is done by specifying cubical boxes with the same size through the entire region, counting the number of particles in each box and dividing with the box volumes, hence achieving a concentration, $\text{particles}/(\text{box volume})$.

A better way would be to find a good weighted mean over the particles in the vicinity of $\mathbf{x}$, i.e. a so called kernel estimate, i.e. a weighed mean with the weight being proportional to a specified kernel function. With a smart kernel function, kernel estimation can be used to estimate mean fields, such as concentration, from the particles values. Moreover, the error estimation of the concentration field could be made easier with a kernel function.

References

[1] S. P. Arya. *Introduction to Micrometeorology*. Academic Press Inc., London, 1988. ISBN 0-12-064490-8.

[2] S. P. Arya. *Air Pollution and Dispersion*. Oxford University Press, 1999. ISBN 0-19-507398-3.

[3] T. P. Meyers R. P. Hosker B. B. Hicks, D. D. Baldocchi and D. P. Matt. A preliminary multiple resistance routine for deriving dry deposition velocities from measured quantities. *Water Air Soil Pollut*, 36:311–330, 1987.

[4] S. J. Dyster D. D. Apsley and C. McHugh. Modelling dry depositin. *ADMS 3*, page P17/13D/03, 2003.

[5] U. Högström. Non-dimensional wind and temperature profiles in the atmospheric surface layer: a re-evaluation. *Boundary-Layer Meteorology*, 42: 55–78, 1988.

[6] R Eiden K. Peters. Modelling dry deposition velocity of aerosol particles to a spruce forest. *Atmospheric Environment*, (26):2555–2564, 1992.

[7] E. Karlsson and S. Nyholm. Dry deposition and desorption of toxic gases and from snow surfaces. *Jounral of Hazardous Materials*, 60:227–245, 1998.

[8] Näslund Erik. Karlsson Edvard. and Gryning Sven-Erik. Comparison between experimantal data and a langevin particle dispersion model including dry deposition. Technical report.

[9] J. Padro L. Barrie L. Zhang, S. Gong. A size-segregated particle dry deposition scheme for an athmospheric aerosol module. *Atmospheric Environment*, (35):549–560, 2001.

[10] J. R. Brook L. Zhang and R. Vet. A revised parameterization for gaseous dry deposition in air-quality models. *Atmospheric Chemistry and Physics*, (3): 2067–2082, 28 November 2003.

[11] P. A. Makar J. R. Brook S. Gong L. Zhang, M. D. Moran. Modelling gaseous dry deposition in aurams: a unified regional air-quality modelling system. *Atmospheric Environment*, (36):537–560, 2002.

[12] Nyre'n K. och Broxvall Å. Gassy, datorprogram för beräkning av ångtryck, flyktighet och avdunstning av kemiska stridsmedel och similämnen. Technical report, 1988.

[13] S. B. Pope. *Turbulent Flows*. Cambridge University Press, 2001. ISBN 0-521-59886-9.

[14] Fredrik Schönfeldt. A Langevin equation dispersion model for the stably stratified boundary layer. Technical report, FOI Swedish Defence Research Agency, Division of NBC Defence, 1997.

[15] Stefan Sehlstedt. A Langevin equation dispersion model for the unstably stratified boundary layer. Technical report, FOI Swedish Defence Research Agency, Division of NBC Defence, 2000.

[16] W.G.N. Slinn. Predictions for particle deposition to vegetative surfaces. *Atmospheric Environment*, (14):1013–1026, 1982.

[17] R. B. Stull. *Introduction to Boundary layer Meteorology*. Kluwer Academic Publisher, Dordrecht, Boston, London, 1988. ISBN 90-227-2768-6.

[18] R. B. Stull. *Meteorology for Scientists and Engineers*. Brooks/Cole, second edition edition, 2000. ISBN 0-534-37214-7.

Appendix A

User guide

A.1 Meteorological data

Weather conditions affect the degree of turbulence in the planetary boundary layer.

Relevant weather conditions should be specified for a simulation of a particle dispersion in the planetary boundary layer.

The first thing of interest when particle dispersion in the planetary boundary layer is to be simulated is the weather conditions during the simulation which affect the degree of turbulence.

Below is a parameter by parameter description the parameters that should be specified in the file `input_ana.nml`.

Roughness length

The *roughness length* z_0 is defined to be the height over the displacement plane where the mean wind becomes zero. Typically this value is around $1/10$ of the vegetation and obstacles covering the ground.

z_0 together with the parameter for the lowest allowed z-value can affect the stability of the differential equations solved in the convective case.

Friction velocity

The *friction velocity* $u_* = (|\tau/\rho|)^{1/2}$, where τ is the Reynolds stress and ρ is density, is approximately proportional to the mean velocity. The square of the friction velocity, u_*^2, is called the *kinematic stress* and is stress per unit density of air.

This parameter could be used to edit the wind speed.

Obukhovs length

(Monin-)Obukhovs length defined as, $L = \frac{-u_*^3 T_v}{kgQ_{v_0}}$, where T_v is virtual temperature[5], k is von Karman's constant which is set to a value of 0.4 [18], g is gravitational acceleration and Q_{v_0} is kinematic virtual temperature flux at the surface in units of $K \cdot m/s$. Obukhovs length is interpreted as the height where the buoyancy generated turbulence becomes dominant over the dynamically generated turbulence.

This parameter indicates if the stratification is stable, neutral or unstable. Where L is positive for stable conditions, negative for unstable and 0 for neutral conditions. Obukhovs length can also be used to edit the wind speed.

[5]Virtual temperature is the temperature that dry air would have if its pressure and density where equal to that of moist air.

Vertical virtual potential temperature gradient, γ

The constant γ is the gradient of the virtual potential temperature, $\gamma = \frac{\partial \Theta}{\partial z}$, where Θ is the virtual potential temperature and z is the height above ground. By using Θ, the degree of stratification is divided into the three categories:

1. Unstable, when $\frac{\partial \Theta}{\partial z} < 0$
2. Neutral, when $\frac{\partial \Theta}{\partial z} = 0$
3. Stable, when $\frac{\partial \Theta}{\partial z} > 0$

Brunt-Väsiälä frequency

Brunt-Väisälä frequency, $N = \left(\frac{g}{T_v}\frac{\partial \Theta_v}{\partial z}\right)^{1/2}$ is defined to be the frequency at which a sufficiently small volume of air would oscillate when displaced vertically under stable conditions. $\frac{\partial \Theta_v}{\partial z}$ is the vertical gradient of the virtual potential temperature[6], g is gravitational acceleration and T_v is virtual temperature. The oscillation arises when air parcels are hotter (or cooler) than the surrounding air and experience a buoyancy force will gives them an upward (or downward) velocity towards equilibrium. When the air parcels pass the equilibrium condition the buoyancy force is reversed and an oscillation start.

The frequency is undefined during unstable stratification and equals 0 in neutrals stratification.

Coriolis parameter

The *Coriolis parameter* is derived from the scalar value of the Coriolis force that acts on a moving object on earth. The Coriolis force, $F = 2\Omega \sin \Theta v + \sin \Theta v^2/R$, where Ω is angular frequency of earth given in radians per second and Θ is latitude, R is the earth radius and v is the speed of the object. The last term is so small, it can be neglected compared to the first term. Thus $F = 2\Omega \sin \Theta \cdot v = f \cdot v$, where f is the Coriolis parameter.

Typical value in Sweden is $1.3 \cdot 10^{-4}\ 1/s$.

Large scale vertical velocity

The large scale vertical velocity, W_{BL}, is the vertical velocity on the top of the boundary layer. The W_{BL} tends to increase the boundary layer height.

Wind angle

The *wind angle* rotates the wind clockwise around the z-axis. The wind angle is defined in standard meteorological convention so that 270° corresponds to a west wind or wind in the positive x-direction, i.e. the standard direction for the wind in the model, and a wind angle of 0° is the same as a wind in the negative y-direction, or north wind, see figure A.1.

[6] Virtual potential temperature is the potential temperature where dry air would have the same density as moist air.

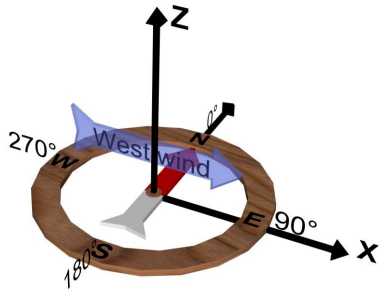

Figure 8. *A west wind in the coordinate system with meteorological angle convention.*

A.2 Evaporation

When estimating evaporation from the ground deposition layer the interesting factor is the change in the relative fraction remaining with time, defined as $\frac{d\mathrm{Fr}}{dt}$. The number of particles evaporated is roughly this factor times Δt. Theoretically $\frac{d\mathrm{Fr}}{dt}$ is constant with time and a function of temperature, particle diameter, substance purity and wind speed at a constant height above the ground. However due to impurities and ground absorption $\frac{d\mathrm{Fr}}{dt}$ is time dependent.

The parameter $\frac{d\mathrm{Fr}}{dt}$ is read from a separate data file containing a table of the form:

1. Number of different temperatures, number of different wind speeds and number of different deposited aerosol drop diameters.

2. For each temperature:

   2.1 For each wind speed:

   2.1.1 For each diameter:

   2.1.1.1 Time in hour to use the specific $\frac{d\mathrm{Fr}}{dt}$, the value of the derivative and the fraction evaporated.

See table A.2 for a complete description of the file layout.

There is a *Python* script called `gassy.py` for generating evaporation tables using the model *GASSY*.

Bellow is a description of parameters necessary to define for evaporation in the input file.

Evaporate

The parameter *Evaporate* is a logical variable set to `true` if deposited non-gas particles are be able to evaporate.

Table 7. Example of an input data table for estimation of $\frac{d\mathrm{Fr}}{dt}$. Comments start with !.

! Number of different temperatures in file 2	Number of different wind speeds in file 2	Number of different diameters in file 2
! temp [°C] 10.0	wind speed $[m/s]$ 1.0	diameter $[m]$ 0.001
! time $[h]$ .339 ⋮ 4.403	$\frac{d\mathrm{Fr}}{dt}$ .0000819 ⋮ .0000239	Mass evaporated $(\%)$ 10.0 ⋮ 90.0
! temp [°C] 10.0	wind speed $[m/s]$ 1.0	diameter $[m]$ 0.002
⋮	⋮	⋮
! temp [°C] 10.0	wind speed $[m/s]$ 2.0	diameter $[m]$ 0.001
⋮	⋮	⋮
! temp [°C] 10.0	wind speed $[m/s]$ 2.0	diameter $[m]$ 0.002
⋮	⋮	⋮
! temp [°C] 20.0	wind speed $[m/s]$ 1.0	diameter $[m]$ 0.001
⋮	⋮	⋮
! temp [°C] 20.0	wind speed $[m/s]$ 1.0	diameter $[m]$ 0.002
⋮	⋮	⋮
! temp [°C] 20.0	wind speed $[m/s]$ 2.0	diameter $[m]$ 0.001
⋮	⋮	⋮
! temp [°C] 20.0	wind speed $[m/s]$ 2.0	diameter $[m]$ 0.002
⋮	⋮	⋮

pppfact

The *pppfact* parameter specifies the relationship between the number of evaporated particles and the number of particles released from the source, i.e. if pppfact, is equal to 10 when 100 fluid particles are released from the source representing a total mass of 100 kg each evaporated particle will represent 0.1 kg of the substance. In concentration pppfact acts as

$$c_m = c\frac{m_{tot}}{N_f}\frac{1}{\text{pppfact}}, \tag{38}$$

where c_m is mass concentration (kg/m^3), c is particle concentration $(1/m^3)$, m_{tot} is the total mass (kg) released from the source and N_f is the number of particles

released from the same source.

resusp_fact

With *resusp_fact* the particle type from different sources can be specified. A source with resusp_fact defined to 0 is a gas source, 1 is a source with fluid drops and 2 is a solid particle source.

create_evap_file

The parameter *create_evap_file* is used to specify what type of out file the program should produce, see table A.2 for the different values of the parameter and their meanings.

release_fact

Dimension less factor used to modulate evaporated particles release height. Evaporated particles release height is equal to

$$\text{height} = \frac{\nu_a z_{\text{rough}}}{u_*} \cdot \frac{10^5}{\text{zlow}} \text{release\_fact}, \tag{39}$$

where ν_a is air viscosity. The first part in equation (39) ($z_{\text{rough}} \cdot \nu_a / u_*$) represent the upper level where the aerodynamic resistance is the dominating deposition resistance.

A.3 Source waypoints

To simulate source movements a number of waypoints can be specified for each source. Each source must have the same number of waypoints. If one source is moving and the other is not then the one not moving must have fixed waypoints.

Here is an example with n waypoints and m sources where subscription denotes the waypoint number and superscription denotes the source number:

$$\begin{array}{rcccccccccccc}
n_waypoints & = & n & & & & & & & & & & \\
x_waypoint & = & x_1^1 & x_2^1 & \dots & x_n^1 & x_1^2 & \dots & x_n^2 & \dots & x_1^m & \dots & x_n^m \\
y_waypoint & = & y_1^1 & y_2^1 & \dots & y_n^1 & y_1^2 & \dots & y_n^2 & \dots & y_1^m & \dots & y_n^m \\
z_waypoint & = & z_1^1 & z_2^1 & \dots & z_n^1 & z_1^2 & \dots & z_n^2 & \dots & z_1^m & \dots & z_n^m
\end{array}$$

A.4 Weather condition changes

To change weather conditions during a simulation the model can write to an out file specified by the variable `save_out_file` that stores the entire final state the particles have at the end of the run. Then another simulation with a different input data can continue from exactly the same position, if the out file from the previous run is renamed to the name of the in file specified be the variable `read_in_file`.

Table 8. Possible values for the variable *creat_evap_file* in the file `input_ana.nml`.

create_evap_file	Out files description
1	Don't separate between any types of particles. Only create the four files:
	NAMN_IN_t.dat — Contains time for output and the number of airborne particles.
	NAMN_IN_xyz.dat — Contains x-, y-, z-, age, radius and settling velocity for the airborne particles.
	NAMN_IN_t_dep.dat — Contains times for output and number of deposited particles.
	NAMN_IN_xyz_dep.dat — Contains x-, y-, z-, age, radius and settling velocity for the deposited particles.
2	Separate between gas and non-gas particles. Creates the four files mention above, but this time they only contains non-gas particles, and additionally the four files:
	NAMN_IN_gas_t.dat — Contains time for output and the number of airborne gas particles.
	NAMN_IN_gas_xyz.dat — Contains x-, y-, z-, age, radius and settling velocity for the airborne gas particles.
	NAMN_IN_gas_t_dep.dat — Contains times for output and number of deposited gas particles.
	NAMN_IN_gas_xyz_dep.dat — Contains x-, y-, z-, age, radius and settling velocity for the deposited gas particles.
3	Separates between evaporated and non evaporated particles in the same way as above for gas and non-gas particles. In addition to the four files in case one, in this case also the files:
	NAMN_IN_evap_t.dat — Specified as in case 2 but for evaporated particles.
	NAMN_IN_evap_xyz.dat — Specified as in case 2 but for evaporated particles.
	NAMN_IN_evap_t_dep.dat — Specified as in case 2 but for evaporated particles.
	NAMN_IN_evap_xyz_dep.dat — Specified as in case 2 but for evaporated particles.

A.5 Input

The input data to the model are stored in a file called `input_ana.nml`, see tables A.5, A.5, A.5, A.5, A.5 and A.5 for the full list of input parameters that

must be stated in the file.

Table 9. Different surface areas variables in the file input_ana.nml.

Variable	Description	# of values	Unit
n_z0	number of roughness areas. I.e. number of different surfaces in the x-direction. In the unstable case n_z0 = 1	1	-
X_Z0	x-coordinates where a new z0 value becomes valid. The first X_Z0 value must be less than the lowest x-value in the simulation.	n_z0	m
Z_ROUGH	Roughness length for the different surfaces. The height above the ground where the mean wind becomes zero.	n_z0	m
UST	Friction velocity.	n_z0	m/s
MO_L	(Monin-)Obukhovs length	n_z0	m
N_FREQ	Brunt-Väisälä frequency	n_z0	$1/s$
F_PARAM	Coriolis parameter	n_z0	$1/s$
WERT_VEL	Large scale vertical velocity	n_z0	m/s
ground_temp	temperature at ground	n_z0	K
h_start	Start value for unstable boundary height calculation	1	m
t_end	Length of time period during which the unstable boundary height grows	1	h
gamma	the stratification above the PBL, potential temperature	1	K/m
XHIGH	the largest x-value in the calculation domain	1	m
ZLOW	the lowest z-value in the calculation domain	1	m
VD	the deposition velocity due to transport through the laminar surface layer and surface resistance	1	m/s
AGE0	age of particles at the release	n_source	s
WIND_ANGLE	rotate the wind clockwise in the xy-plane 0° gives wind from north $\Rightarrow$ wind in $(-y)$-direction 90° gives wind from east $\Rightarrow$ wind in $(-x)$-direction 180° gives wind from south $\Rightarrow$ wind in y-direction 270° gives wind from west $\Rightarrow$ wind in x-direction	1	$^\circ$
RELEASE_FACT	Used to modulate evaporated particles release height	1	-

Table 10. Namelist `method` in `input_ana.nml`.

Variable	Description	# of values	Unit
DIFCOEF	Switch for LEM/RDM. Not in use. Must be = 1 in this model	1	-
DTMAX	Maximum time step	1	s
TOT_TIME	Time limit in the calculations	1	s
PDENS	release rate for particles	n_source	$1/s$
PTK_ST	source release start	n_source	s
PTK_END	source release stop	n_source	s
EVAPORATE	if true then deposited fluid particles can evaporate. If false then evaporation from the ground deposition layer is ignored as in previous models.	1	-

Table 11. Namelist `moving-source` in `input_ana.nml`.

Variable	Description	# of values	Unit
n_waypoints	number of waypoints. Set to 0 if the sources should be fix. The waypoints specifies trajectories for the sources	1	$-$
X_WAYPOINT	x position of waypoints	n_waypoints· n_source	m
Y_WAYPOINT	y position of waypoints	n_waypoints· n_source	m
Z_WAYPOINT	z position of waypoints	n_waypoints· n_source	m
T_WAYPOINT	time to reach waypoint	n_waypoints	s

Table 12. Namelist `settling` in `input_ana.nml`.

Variable	Description	# of values	Unit
s	standard deviation of particle radius	n_source	$-$
r0	median particle radii	n_source	m
rmax	maximum radius in the distribution	n_source	m
rmin	minimum radius in the distribution	n_source	m
p	air pressure used to calculate settling velocity	n_z0	Pa
T	air temperature used to calculate settling velocity	n_z0	K
rho_part	density of the particles	n_source	kg/m^3
g_earth	acceleration due to gravity	1	m/s^2
pppfact	evaporated particles per particle from fluid drop source	n_source	$-$
resusp_fact	particle type, gas=0, fluid=1, solid=2. The current version of the model only calculates evaporation from deposited fluid particles. Different particle types are treated in the same way except from evaporation	n_source	$-$

Table 13. Source position variables `input_ana.nml`.

Variable	Description	# of values	Unit
n_source	number of sources	1	-
XINL1	low x-coordinates for the sources	n_source	m
XINL2	high x-coordinates for the sources	n_source	m
YINL1	low y-coordinates for the sources	n_source	m
YINL2	high y-coordinates for the sources	n_source	m
ZINL1	low z-coordinates for the sources	n_source	m
ZINL2	high z-coordinates for the sources	n_source	m

Table 14. Namelist `utdata` in `input_ana.nml`.

Variable	Description	# of values	Unit
DELT_UT	time interval for output	1	s
NAMN_IN	carries basic part of the name of output files	1	–
TABLE_NAME	name of evaporate data table	1	–
CREATE_EVAP_FILE	see table A.2	1	–
READ_IN_FILE	A file containing particle information from previous run. If empty or don't exist then no file will be read implying that the simulation should start from scratch.	1	-
SAVE_OUT_FILE	A file for storing the particles status at the end of the run. If empty no file will be created. This file can be read by READ_IN_FILE.	1	-