AEROSOL MICROPHYSICS IN CARMA

Ray Yang

1 .INTRODUCTION

CARMA (Community Aerosol & Radiation Model for Atmospheres) created by NASA Ames Aerosol Modeling group (V1.00 Oct., 1995 , V2.2 Jun, 2001).

- Good organized structure
- Detailed treatment of particle types/phases
- Expandable to multi-component aerosols

2. HIERARCHICAL STRUCTURE OF CARMA

- Functional routines
- Calculation routines
- Initialization routines/output routines

microslow.f
microfast.f

Coagulation
Growth
Evaporation
Nucleation
Vertical transport
Horizontal transport
Radiative transfer

3. MICROPHYSICS

3.0 Some Basic theories

Particle Properties with Size Regimes

$$Kn = \frac{\lambda_g}{r_p}$$

When a particle size is much larger than the mean free path _g of gas molecules, the gas acts as a viscous fluid and the dynamics of particles can be described by continuum theory. On the contrary, the continuum theory couldn't be applied and should be corrected for very tiny particles.

Particle size regimes and Knudsen number (at air 1013 hPa, 20^0C, λ_0=0.066μm)

Size Regime	Free Molecule	Transition	Slip Flow	Continuum (Stokes)
Kn	>10	10 - 0.3	0.3 - 0.1	<0.1
Radius r_p (μm)	<0.005	0.005 - 0.2	0.2 - 0.65	>0.65

Aerosol Size Distribution

Lognormal distribution is widely used.

$$\frac{dN}{d\log r} = \frac{N}{\sqrt{2\pi}\log\sigma}\exp\left[-\frac{1}{2}\left(\frac{\log(r/r_g)}{\log\sigma}\right)^2\right]$$

One advantage of the lognormal distribution is that it can be superimposed to form a new size distribution describing specific atmospheric aerosol modes.

$$n = \sum_{i=1}^{3} \frac{N_i}{\sqrt{2\pi}\log\sigma_i}\exp\left[-\frac{1}{2}\left(\frac{\log(r/r_{gi})}{\log\sigma_i}\right)^2\right]$$

Bin Design

Fixed Bin scheme, volume increase bin design:

$\Delta V(i+1)/\Delta V(i) = c$

3.1 Nucleation

Classical homogeneous nucleation theory:

The change of Gibbs free energy during a formation of cluster

$$\Delta G = 4\pi\, r_p^{\,2}\sigma_p - \frac{4}{3}\pi\, r_p^{\,3}\rho_p \frac{R_c T}{M_g}\ln S_q$$

the critical radius and critical number of molecules:

$$r_c = \frac{2\sigma_p M_g}{\rho_p R_c T \ln S_q}$$

$$J_{\mathrm{hom}o} = 4\pi\, r_c^{\,2}\beta_x Z_n N_x \exp(\frac{-\Delta G^{*}{}_{\mathrm{hom}o}}{k_B T})$$

$$\Delta G^{*}{}_{\mathrm{hom}o} = \frac{4}{3}\pi\, r^2{}_c\sigma_p$$

$$\beta_x = N_x\sqrt{\frac{k_B T}{2\pi m_x}}$$ (number of molecules striking unit surface /s)

Z_n –Zeldovich nonequilibrium factor

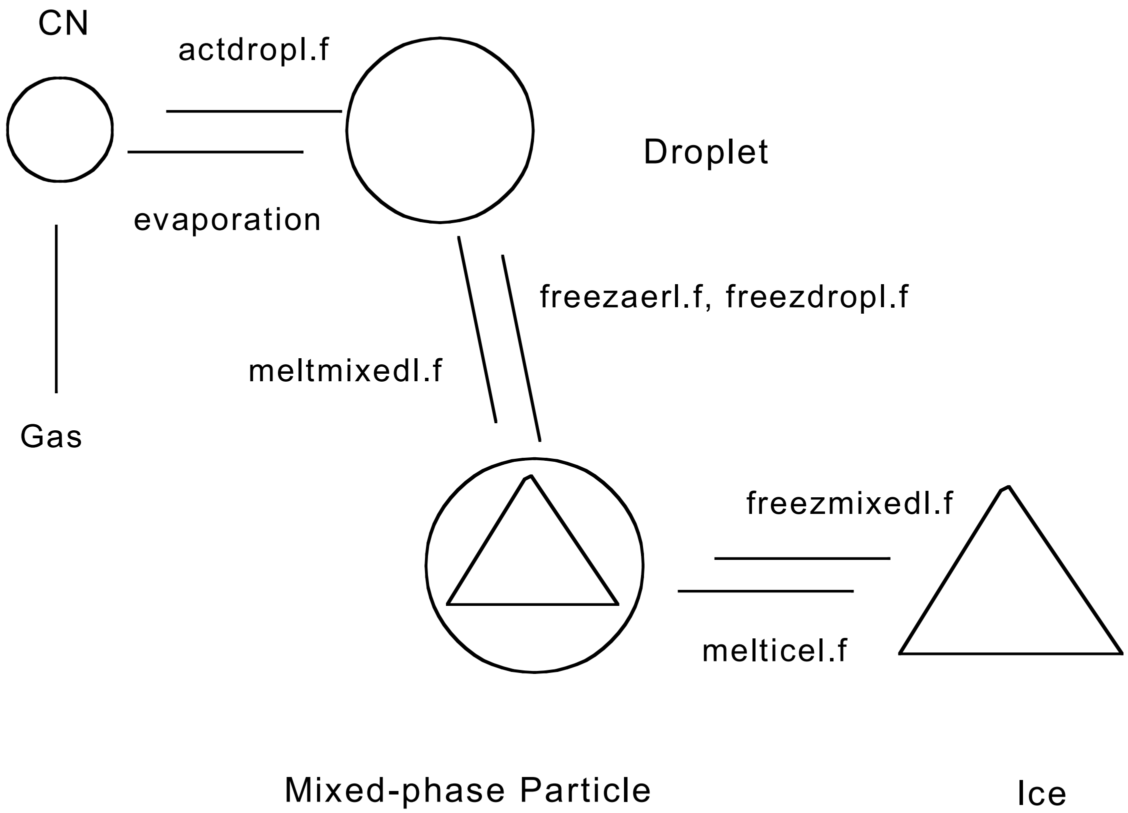
Classic nucleation:

$$J_{hete} = 4\pi\, r_c^{\,2}\beta_x\beta_y\tau\exp(\frac{-\Delta G^{*}{}_{hete}}{k_B T})$$ (x=y, or x≠y)

$$\Delta G^*{}_{hete} = \Delta G^*{}_{\mathrm{hom}\,o} f_h(x_h, m_h)$$

$$x_h = \frac{R_h}{r_c}, \quad m_h = \cos\theta$$

In CARMA:



The rate of ice nucleation in H_2SO_4 solution (Tabazadeh et.al, GRL, 27, 1111, 2000):

$$J = C(T, w_s, V_d)\exp\left(\frac{-\Delta F_g(T, W_s) - \Delta F_{act}(T, W_s)}{kT}\right)$$

$$C(T, w_s, V_d) \cong 2.1 \times 10^{23} V_d \sqrt{\sigma_{sul} / \sigma_{ice}(w_s, T)T}$$

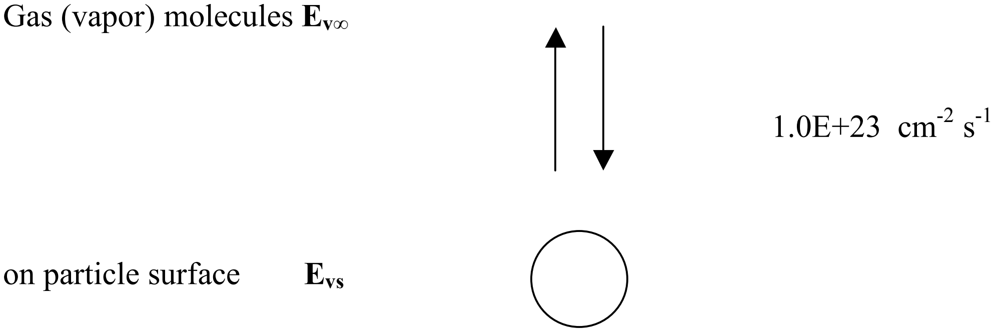
$$\Delta F_g = \frac{4}{3}\pi\sigma_{sul/ice} r_g{}^2$$

$\sigma_{sul/ice}$ is interface energy between sulfate/ice solution; r_g is the critical germ radius; C preexponential factor; ΔF_g is Gibbs free energy for the formation of ice; ΔF_{act} is diffusion activation energy of water molecules across the ice/sulfate solution phase boundary; k is Boltzmann constant.

psolve.f: update particle concentration

3.2 Growth/Condensation/Evaporation

Interaction between aerosol particle and gas molecules:



Gas (vapor) molecules $\mathbf{E_{v\infty}}$

1.0E+23 cm^{-2} s^{-1}

on particle surface $\mathbf{E_{vs}}$

$$\Delta E_v = E_{vs} - E_{v\infty}$$

Condensation: $\Delta E_v < 0$
In balance: $\Delta E_v = 0$
Evaporation: $\Delta E_v > 0$

Saturation vapor pressure over droplet containing dissolved salt (Koehler effect):

$$\frac{E_{s,s}}{E_{\infty,s}} = \left(1 + \frac{3\upsilon\ m_s M_w}{4M_s \rho_w \pi\ r^{*3}}\right)^{-1} \exp\left(\frac{2\gamma\ M_w}{\rho_w RT r^*}\right)$$

$$\frac{E_{s,s}}{E_{\infty,s}} = \left(1 + \frac{B}{r^{*3}}\right)^{-1} \exp\left(\frac{A}{r^*}\right)$$

$$\approx (1 - \frac{B}{r^{*3}})(1 + \frac{A}{r^*}) = 1 + \frac{A}{r^*} - \frac{B}{r^{*3}}$$

Critical supersaturation:

$$S_c = \frac{A}{r^*} - \frac{B}{r^{*3}}$$

Koehler curve, critical radius

Condensational Diffusion Growth:

$$\frac{dm}{dt} = 4\pi\ r_p D_v (\rho_{v,\infty} - \rho_{v,s}) f_c(Kn_c)$$

$$\frac{dq}{dt} = 4\pi\ r_p K_g (T_s - T_\infty) f_h(Kn_h)$$

$$\frac{dm}{dt} = \frac{4\pi\ r_p D_v (p_{v,\infty} - p_{v,s})}{\frac{D_v L_v p_{v,s}}{K_g T_\infty}(\frac{L_v}{R_v T_\infty} - 1) + R_v T}$$

setupnuc.f : to calculate critical S for CN to droplets

setupgkern.f, setupgrow.f

growevapl.f (to calculate dm/dt)***, growp.f (***source term from ***growevapl.f)*** → ***psolve.f***

3.3 Coagulation

Continuous general dynamic equation:

$$\frac{\partial n_v(t)}{\partial t} + \frac{\partial}{\partial v}[I_v(t)n_v(t)] = \frac{1}{2}\int_0^v \beta_{v-v',v'} n_{v-v'}(t)\, n_{v'}(t)\, dv'$$

$$- n_v(t)\int_0^\infty \beta_{v,v'} n_{v'}(t)\, dv' + J_0(t)\delta_{v-v'} + S_v - R_v$$

I_v –rate of a particle volume change with size v; J_0 –production (nucleation), S—emission rate; R—removal rate.

Continuous coagulation equation (used in CARMA):

$$\frac{\partial n_v(t)}{\partial t} = \underbrace{\frac{1}{2}\int_0^v \beta_{v-v',v'} n_{v-v'}(t)\, n_{v'}(t)\, dv'}_{\text{coagulation production}} - \underbrace{n_v(t)\int_0^\infty \beta_{v,v'} n_{v'}(t)\, dv'}_{\text{coagulation loss}}$$

Loop over: ibin (bin), ig (group), ielem (element), ixyz (3D grids)
microslow.f: setupcoag.f → ***coagp.f*** + ***coagl.f*** → ***csolve.f***

Numerical calculation methods:

- Implicit coagulation: iterative computation
- Semi-implicit coagulation scheme: noniterative computation--- used in CARMA (to use n $_{i,t-h}$ replace n $_{i,t}$)

Other coagulation calculation schemes in CARMA***: seinfeld86.f, smoluchowski.f***

Coagulation kernel β (rate coefficient cm^3/particle s)

It depends on particle shape, composition, r.h., etc. The physical processes that result in collision and coagulation include: Brownian movement, gravitational sedimentation, turbulent affect, static electronic force etc.

For Brownian diffusion:

$$\beta_{p1,p2} = 4\pi(r_1 + r_2)(D_{p1} + D_{p2})$$

Overall kernel:

$$\beta_{p1,p2} = \underset{*}{\beta^B{}_{p1,p2}} + \underset{*}{\beta^G{}_{p1,p2}} + \beta^T{}_{p1,p2} + \ldots$$

(* included in CARMA)

To evaluate coagulation kernels in CARMA***: setupckern.f*** and ***setupcoag.f***

3.4 Particle Transport/Diffusion

The advective flux form for a conservative scalar Ω (=ρq, q mixing ratio):

$$\frac{\partial(\Omega)}{\partial t} + \vec{\mathrm{v}} \cdot (\Omega \vec{V}) = 0$$

This equation can be extended by taking into consideration of diffusion, source and sink terms. For a species *i*,

$$\frac{\partial c_i}{\partial t} + u_x \frac{\partial c_i}{\partial x} + u_y \frac{\partial c_i}{\partial y} + u_z \frac{\partial c_i}{\partial z}$$

$$= \frac{\partial}{\partial x}(K_{xx} \frac{\partial c_i}{\partial x}) + \frac{\partial}{\partial y}(K_{yy} \frac{\partial c_i}{\partial y}) + \frac{\partial}{\partial z}(K_{zz} \frac{\partial c_i}{\partial z})$$

$$+ R_i(c_1, c_2, ...c_n) + E_i(x, y, z, t) - S_i\ (x, y, z, t)$$

E—emission flux, S—removal flux, R—chemical production term, K—eddy diffusion coefficient (for gas and particles).

$$\frac{\partial c_i}{\partial t} + u_x \frac{\partial c_i}{\partial x} + u_y \frac{\partial c_i}{\partial y} + u_z \frac{\partial c_i}{\partial z}$$

$$= \frac{\partial}{\partial x}(K_{xx} \frac{\partial c_i}{\partial x}) + \frac{\partial}{\partial y}(K_{yy} \frac{\partial c_i}{\partial y}) + \frac{\partial}{\partial z}(K_{zz} \frac{\partial c_i}{\partial z})$$

Numerical Finite-differencing Scheme: semi-Lagrangian scheme and using cubic spline method (Piecewise Polynomial Method, Colela, 54, 174, 1984, J. Comp. Phys.) to get interpolation.

horizont.f: htranfosl.f/ htranglk.f/ htranppm.f,   hordif.f

vertical.f: vertadv.f, verdif.f → ***versol.f*** (update new concentration)**.**

3.5 Deposition (dry)

Particles impinge and stick to surface. This processes depend not only on particle properties (size, shape, density, surface property) but also on surface to which particles stick, as well as transport process (diffusion, turbulent).

Deposition velocity for particles:

$$V_d = \frac{1}{R_a + R_b + R_a R_b V_s} + V_s$$

R_a—aerodynamic resistance (s/m); R_b—resistance to molecular diffusion through sublayer; V_s—particle sedimentation velocity.

$$R_a = \frac{\ln(Z_R / Z_0) - \psi_h}{\kappa\, u_*}$$

$$R_b = \frac{1}{\varepsilon_0\, u_*(E_B + E_{IM} + E_{IN}) q_1}$$

Z_R—the height at which V_d is evaluated; Z_0—roughness length; ψ_h—stability function; κ--the Von-Karman constant; u_*-- friction velocity

E_B—collection efficiency for Brownian diffusion; E_{IM}-- collection efficiency for impaction; E_{IN}-- collection efficiency for interception; q1—correction factor; ε0—constant.

3.6 Radiation

$$\mu \frac{dI(\lambda,\mu,\varphi)}{d\tau} = I(\lambda,\mu,\varphi) - J^{diff}(\lambda,\mu,\varphi) - J^{dire}(\lambda,\mu,\varphi) - J^{emis}(\lambda,\mu,\varphi)$$

Radiance transfer:

$$\mu \frac{dI(\tau,\mu)}{d\tau} = I(\tau,\mu) - \frac{\widetilde{\omega}}{2}\sum_{l=0}^{N} \widetilde{\omega}_l P_l(\mu)\int_{-1}^{1} P_l(\mu^{'}) I(\tau,\mu^{'}) d\mu^{'}$$

$$- \frac{\widetilde{\omega}}{4\pi}\sum_{l=0}^{N} \widetilde{\omega}_l P_l(\mu) P_l(-\mu_0)\pi F_0 e^{-\tau/\mu_0}$$

$$\widetilde{\omega} = \frac{\beta_s}{\beta_e} = \frac{\beta_s}{\beta_s + \beta_a}$$

Two-stream method: radiance is divided into upward and down ward component.

Two-stream radiative transfer model, using optical properties of droplet/ice (no optical treating for other components and multi-components).

oppr.f -- calculate optical properties: single scattering albedo, asymmetry parameter, optical thickness.

miess.f – Mie scattering calculation

rdtran.f –driver routine

twostr.f ---flux calculation by using two stream method

4. IMPROVEMENT WORK

Aerosol source/emission

surface source, secondary source

Nucleation

Binary (/ternary) nucleation: SO_2+H_2O

Aerosol Chemistry

Chemical interaction between gaseous species and particles

Aerosol-cloud interaction

Aerosol to cloud micro-processes, Particle in- (and below) cloud scavenging

Cloud aqueous phase chemistry

Simultaneous simulation

Radiative transfer in multi-component aerosol layers

This is a brief description of how the default simulation is set up.

As shipped, CARMA is set up for a 1-D simulation of a liquid water cloud forming in a rising bubble. The particle types, specified in aer/setupaer.f, include 3 elements (CN, droplets, and sulfate core mass within droplets). We have also included aer/setupaer.4elem.f, which includes the core mass second moment element, and aer/setupaer.8elem.f, which includes ice and mixed-phase elements. Processes that are active in the default simulation (specified in aer/init.f) include droplet activation from (and evaporation to) CN, condensational growth, coagulation, vertical transport, and radiative transfer. The vertical grid is set up in aer/initatm.f with 35 levels spanning the height range from the surface to 17 km. The dimensions of the spatial and particle grids are set up in include/aerad.h.

The simulation runs 160 time-steps of 10 seconds (the time-step for slow processes such as transport and coagulation), which are divided into 2 to 1000 sub-steps at each grid point for fast microphysical processes (such as nucleation and condensation). Cloud formation is forced by an imposed vertical wind specified in aer/prestep.f -- the initial vertical wind profile has a maximum value of 5 m/s at 5.5 km, and the height of the peak vertical wind moves up at 5 m/s throughout the simulation.

The top-level makefile specifies compilation with g77 with optimization. On a 700 MHz Pentium III system, the simulation takes about 20 seconds to run.
30 /rmounts/home/staff/ryang/carma_2.2/car22/doc:

Simulation of the following processes [NYI = not yet implemented]:

```
   Coagulation.
   Growth.
   Evaporation.
   Nucleation.
   Vertical transport.
   Horizontal transport.
   Thermodynamics.
   Radiative transfer.
   Chemistry.            [NYI]
```

One advantage of CARMA is that it has, therefore detailed design of nucleation and coagulation scheme etc.

The model's top-level calling tree looks like:

```
main          Main overall program (could be replaced by another model).
  init        Defines run vars & controls cold start or restart.
   initnew    Performs cold start initialization.
   initres    Performs a restart initialization.
  step        Controls the taking of 1 timestep.
   prestep    Processing that occurs before each timestep.
   newstate   Controls calc of new values for model state variables.
   postep     Processing that occurs after each timestep.
    outprt    Outputs timestep info to print file at current timestep.
    outhis    Outputs model history to history file at current timestep.
    outres    Outputs model restart info.
  quit        Shuts down the model at the end of a run.
```

```
Calling sequence and brief descriptions of fortran routines.


  main                 Main overall program.
    init               Defines run vars & controls cold or restart.
      initnew          Performs cold start initialization.
        initatm        Initializes model atmosphere variables.
        setupaer       Defines mapping arrays & time-indep params for  microphysics.
          setupbins    Sets up particle bins and shapes.
          setupvf      Calculates particle fall velocities.
          setupgrow    Sets up mapping arrays for particle growth.
          setupgkern   Ct condensational growth kernels.
          setupnuc     Sets up mapping arrays for nucleation.
          setupckern   Ct particle coagulation kernels.
          setupcoag    Sets up mapping arrays for coagulation.
        initaer        Initializes particle size distributions.
        initgas        Initializes gas concentrations.
          vaporp       Ct vapor pressures.
          supersat     Ct supersaturations.
          initrad      Defines time-independent radiative transfer variables
            setuprad   Radiative transfer setup routine (in ../rad subdirectory).
          prerad       Definitions prior to radtran call.
          radtran      Radiative transfer calculation (in ../rad subdirectory).
          postrad      Calculations after radtran call.
          zerorad      Sets radiation variables to zero.
      initres          Performs a restart initialization.
      prtsep           Outputs simple print file visual separator.
      outprt           Outputs timestep info to print file at current timestep.
        prtsep         Outputs simple print file visual separator.
      outhis           Outputs model history to history file at current timestep.
        outhis_ncdf    Outputs history in netcdf format
        outhis_bin     Outputs history in fortran binary format
        prtsep         Outputs simple print file visual separator.
    step               Controls the taking of 1 timestep.
```

```
      prestep          Processing that occurs before each timestep.
        zeromicro      Zeroes fast microphysics sinks and sources.
      newstate         Controls calc of new values for model state variables.
        horizont       Drives horizontal transport calculation.
          glkcoef      Ct coefficients needed for Galerkin method.
          htranglk     Ct horizontal advection rates using Galerkin method.
          htranppm     Ct horizontal advection rates using PPM method.
          htranfosl    Ct horizontal advection rates using FOSL method.
        vertical       Drives vertical transport calculation.
          vertadv      Ct vertical advection transport rates.
          vertdif      Ct vertical diffusion fluxes.
          versol       Solves for new concentrations after vertical diffusion.
        parcel         Ct convective forcings for parcel simulation.
        microslow      Slow calc changes in concentrations due to microphysical
processes.
        coagl        Ct loss rates due to coagulation.
        coagp        Ct production rates due to coagulation.
          csolve       Solve for new particle concen given coag loss & production
rates.
        microfast      Fast calc changes in concentrations due to microphysical
processes.
          zeromicro    Zeroes fast microphysics sinks and sources.
          supersat     Ct supersaturations.
            vaporp     Ct vapor pressures.
          actdropl     Ct particle loss rates due to activation of aerosols -->
droplets.
          freezaerl    Ct particle loss rates due to freezing nucleation of aerosols.
          freezdropl   Ct particle loss rates due to freezing nucleation of drops.
          freezmixedl  Ct particle loss rates due to total freezing of mixed hydrom.
          meltmixedl   Ct particle loss rates due to total melting of mixed hydrom.
          melticel     Ct particle loss rates due to initiation of ice melting.
          coremeltl    Ct core mass changes due to core freezing/melting.
          growevapl    Ct loss rates due to conden growth and evaporation.
          growp        Ct particle source terms due to condensation growth.
          upgxfer      Ct particle source terms due to up-grid transfer processes.
          psolve       Solve for new particle concen given loss & production rates.
          evapp        Ct particle source terms due to evaporation.
          downgxfer    Ct particle source terms due to down-grid transfer processes.
          gasexchange  Ct gas loss rates due to nuc, growth, and evap.
          downgevapply Adds evaporation and down-grid production sources to particle
concen.
          gsolve       Ct new gas concentrations.
          tsolve       Ct new temperatures.
          rhopart      Ct new average particle densities.
        hydrostat      Ct new pressures and metric scale factors.
      varstep          Adjusts time-step based on how much concentrations changed.
      postep           Processing that occurs after each timestep.
        prerad         Definitions prior to radtran call.
        radtran        Radiative transfer calculation (in ../rad subdirectory).
        postrad        Calculations after radtran call.
        rescale        Rescales velocities and diffusion coeffs.
        outprt         Outputs timestep info to print file at current timestep.
          lognormal    Ct lognormal fits to particle size distributions.
          prtsep       Outputs simple print file visual separator.
        outhis         Outputs model history to history file at current timestep.
          outhis_ncdf  Outputs history in netcdf format
          outhis_bin   Outputs history in fortran binary format
          prtsep       Outputs simple print file visual separator.
        outres         Outputs model restart info.
          prtsep       Outputs simple print file visual separator.
    quit               Shuts down the model at the end of a run.
      prtsep           Outputs simple print file visual separator.
```

* ct –calculate(s)