|  |  |  |
| --- | --- | --- |
| Parameter | Description | Default value |
| SWITCHES |  |  |
| dilu\_on | Defines whether the dilution is on or not. If dilu\_on = 0, the aerosol will not dilute. If dilu\_on = 1, the aerosol will dilute with rate defined by parameter dilu\_coeff. | 1 |
| coag\_on | Defines whether the coagulation is on or not. If coag\_on = 0, the particles won't coagulate. If coag\_on = 1, the coagulation is set on. | 1 |
| sedi\_on | Defines whether the sedimentation is on or not. If sedi\_on = 0, sedimentation is turned off. If sedi\_on = 1, sedimentation will occur. Only usable for sedimentation in SAPPHIR chamber! | 0 |
| gas\_source\_is\_vect | Defines whether the parameter gas\_source is a vector or scalar. This cannot be set by user. Instead, the program checks if gas\_source is an array or not and sets the value to 1 or 0 respectively. |  |
| dilu\_vect\_on | Defines whether the parameter dilu\_coeff is an array or scalar. This cannot be set by user. Instead, the program checks if dilu\_coeff is an array or scalar and sets the value to 1 or 0 respectively. |  |
| part\_source\_is\_vect | Defines whether the parameter part\_source is a time-dependent vector or a constant. This cannot be set by user. Instead, the program checks if part\_source is an array or scalar and sets the value to 1 or 0 respectively. |  |
| coag\_mode | Defines whether the particles coagulate 'normally' or agglomerate. Value should be either 'coag' for normal coagulation or 'aggl' for agglomeration. Agglomeration works only for particles in the free-molecule region. Condensation and deposition might not work correctly for agglomerates. |  |
| coag\_num | Numerical representative of coag\_mode. If coag\_mode == 'coag' => coag\_num = 1. If coag\_mode == 'aggl' => coag\_num = 0. This cannot be set directly by user, but the program sets it based on the value of coag\_mode. |  |
| fixed\_sections | Defines whether the model will use fixed or moving sections. If fixed\_sections == 0, moving sections will be used. Otherwise the sections will be fixed.  When the model uses moving sections, the diameter defining the section grows or shrinks by condensation or evaporation.  When using fixed sections, the diameters defining sections are constant. If particles grow by condensation so that they reach the limit diameter, they will be moved to the next section. The limit diameter is logarithmically halfway between adjacent sections. |  |
| Cvap\_const | Defines whether the vapor concentration is constant or not.  If Cvap\_const == 1, the vapor concentration stays at value Cvap0 during the whole simulation time, so that the value gas\_source has no effect on vapor concentration.  If Cvap\_const ~= 1, the vapour concentration is not kept constant, but its value depends on Cvap0 and gas\_source. |  |
| Parameter | Description | Default value |
| BASIC VALUES |  |  |
| part\_source | Particle source rate (1/cm3/s).  If particle source is constant during the simulation, this parameter is a vector of following form:   |  |  | | --- | --- | | source rate (1/cm3/s) | particle size (m) |   where particle size is the diameter of particles created. For example part\_source=[1, 3e-9] will generate one 3-nanometer particle/cm3 every second during the simulation.  If user wants to create particle sources of different sized particles, this can be done by defining part\_source as a 3D-matrix. For example lines  part\_source(:,:,1) = [1, 3e-9];  part\_source(:,:,2) = [5, 5e-9];  will create one 3-nanometer particle/cm3 and five 5-nanometer particles/cm3 every second during the simulation.  Particle sources can also be defined as time-dependent vectors. In that case, the vector should be of following form:   |  |  |  | | --- | --- | --- | | t\_0 | source rate (1/cm3/s) | particle size (m) | | t\_1 | source rate (1/cm3/s) | NaN | | . . . | . . . | . . . | | t\_end | source rate (1/cm3/s) | NaN |   where the source rate in first line defines the source rate of particles at time t\_0, the next line defines it at time t\_1 etc. Particle size in first line defines the diameter of particles created; the values of column 3 in the rest of lines do not matter. |  |
| gas\_source | The condensing vapor source rate (1/cm3/s). Can be defined as a scalar or two-column array. When defined as scalar, the source rate will be constant during the simulation.  If gas\_source is an array, it must have two columns; the first column is a time vector and the second one tells the gas\_source value at respective time. The time vector doesn't need to have same length as parameter tvect. However, the first and last element of gas\_source's first column must have the same values as respective elements of tvect. If the length of the array is different than tvect's length, gas\_source will be interpolated to same length. |  |
|  |  |  |
| Parameter | Description | Default value |
| dilu\_coeff | Dilution coefficient (1/(cm3s)). Dilution affects particle concentration in following way: dN/dt = -dilu\_coeff\*N (N is particle concentration)  Parameter dilu\_coeff can be either a scalar or an array. When defined as a scalar, dilution coefficient will be constant during the simulation.  If dilu\_coeff is an array, it must have two columns; the first column is a time vector and the second one tells the dilu\_coeff value at respective time. The time vector doesn't need to have same length as parameter tvect. However, the first and last element of dilu\_coeff's first column must have the same values as respective elements of tvect. If the length of the array is different than tvect's length, dilu\_coeff will be interpolated to same length. |  |
| satu\_conc | The condensing vapor saturation concentration (1/cm3). |  |
| lambda | The condensing vapor mean free path (m). |  |
| diff\_coeff | The condensing vapor diffusion coefficient (cm2/s). |  |
| vap\_molmass | Molecular mass of condensing vapor (g/mol). |  |
| particle\_dens | Density of particle matter. (g/cm3). |  |
| stick\_coeff | Sticking coefficient. The probability that vapor molecules will stick to aerosol particles. |  |
| Cvap0 | Initial condensing vapor concentration (1/cm3) |  |
| T | The temperature (K) |  |
| TIME VECTOR |  |  |
| tvect | The time vector (seconds). Define as row vector. The vector spacing does not remarkably affect calculation time, but defines only the spacing of the results. |  |
| DISTRIBUTION PARAMETERS |  |  |
| N0 | Initial total particle concentration (1/cm3).  If user wants to create a distribution that consists of several log-normal distributions, N0 can be defined as a vector. Then the total distribution will be a superposition of log-normal distributions. If N0 is defined as N0 = [1e3, 1e5], the first distribution has particle concentration N0(1) (= 1e3), standard deviation sigma(1) and mean diameter mu(1). The total particle concentration will then be N0(1) + N0(2).  Note that each distribution must have values for sigma and mu, so if N0 is a vector, mu and sigma must be vectors of same length. |  |
| mu | The mean of the lognormal size distribution.  If user wants to create a distribution that consists of several log-normal distributions, mu can be defined as a vector. Then the total distribution will be a superposition of n log-normal distributions, so that each distribution is defined by N0(i), mu(i) and sigma(i). Thus, if mu is a vector, N0 and sigma must be vectors of same length. |  |
|  |  |  |
| Parameter | Description | Default value |
| sigma | Sigma (standard deviation) of log-normal size distribution.  If user wants to create a distribution that consists of several log-normal distributions, sigma can be defined as a vector. Then the total distribution will be a superposition of n log-normal distributions, so that each distribution is defined by N0(i), mu(i) and sigma(i). Thus, if sigma is a vector, N0 and mu must be vectors of same length. |  |
| Dp\_min | The exponent of minimum diameter of the size grid.  The program will create a logarithmically spaced size grid between diameters and . The particle distribution will use this grid as x-axis and therefore Dp\_min and Dp\_max define limits inside which the distribution must stay. Thus, if particles for example grow during simulations, Dp\_max must be defined high enough so that the distribution stays inside the limits. |  |
| Dp\_max | The exponent of maximum diameter of the size grid.  The program will create a logarithmically spaced size grid between diameters and . The particle distribution will use this grid as x-axis and therefore Dp\_min and Dp\_max define limits inside which the distribution must stay. Thus, if particles for example grow during simulations, Dp\_max must be defined high enough so that the distribution stays inside the limits. |  |
| sections | The number of sections the model will use.  The size grid is created between and and the number of grid points is defined by parameter ‘sections’. The grid is logarithmically spaced. Particles are then placed in the sections according to their diameter.  The bigger the number of sections, the more accurate the model will be. However, increasing the number of sections slows down the model. |  |
| output\_sections | Defines the number of sections in output size grid.  After the simulation has run, the distribution at each time point is interpolated to the original size grid. The original size grid has however the same amount of sections as the calculated distributions, which causes error in interpolation. By defining output\_sections bigger than ‘sections’, the original size grid is made denser for the interpolation, minimizing the error. That is why output\_sections should always be bigger than ‘sections’, for example ten times bigger. |  |
|  |  |  |
| Parameter | Description | Default value |
| TOLERANCE PARAMETERS | Defines the tolerance settings for ode45. |  |
| Cvap\_tol | Vapor concentration tolerance. |  |
| N\_tol | Particle concentration tolerance. |  |
| Dp\_tol | Particle diameter tolerance. |  |