Functions to work with particle size distributions

vector<double> Convertq2Toq2 (OldGrid, OldDistr, NewGrid)

vector<double> Convertq3Toq3 (OldGrid, OldDistr, NewGrid)

NormalizeDensityDistribution (Grid, qiDistr)

double GetDistributionMedian (Grid, QxDistr)

double GetDistributionValue (Grid, QxDistr, Val)

double GetDistributionMode (Grid, qxDistr)

double GetAverageDiameter (Grid, qxDistr)

double GetSauterDiameter (Grid, q3Distr)

double GetSpecificSurface (Grid, q3Distr)

Several global functions are defined to work with particle size distributions. These functions can be called from any place of the code.
All functions receive grid (Grid) as the input parameter. The grid can be previously obtained with the help of the function GetNumericGrid (DISTR_SIZE) (refer to ‘BaseUnit.pdf’).

d_i – diameter of particle in class i
Δd_i – size of the class i
M_k – k-moment
q – density distribution
$q0$ – number related density distribution
$Q0$ – number related cumulative distribution
$q2$ – surface area related density distribution
$Q2$ – surface area related cumulative distribution
$q3$ – mass related density distribution
$Q3$ – mass related cumulative distribution
w_i – mass fraction of particles of class i
N_i – number of particles of class i
N_{tot} – total number of particles

double GetMMoment (Moment, Grid, InDistr)

Calculates moment of the density distribution:

$$M_k = \sum_i d_i^k q_i \Delta d_i$$

vector<double> ConvertQ0Toq0 (Grid, InDistr)

Performs conversion from Q0 to q0 distributions using information about the size grid:

$$q0_0 = \frac{Q0_0}{\Delta d_i}$$

$$q0_i = \frac{Q0_i - Q0_{i-1}}{\Delta d_i}$$

vector<double> Convertq0ToQ0 (Grid, InDistr)

Performs conversion from q0 to Q0 distributions using information about the size grid:

$$Q0_i = \sum_i q0_i \Delta d_i = Q0_{i-1} + q0_i \Delta d_i$$

vector<double> ConvertQ2Toq2 (Grid, InDistr)

Performs conversion from Q2 to q2 distributions using information about the size grid:

$$q2_0 = \frac{Q2_0}{\Delta d_i}$$

$$q2_i = \frac{Q2_i - Q2_{i-1}}{\Delta d_i}$$

vector<double> Convertq2ToQ2 (Grid, InDistr)

Performs conversion from q2 to Q2 distributions using information about the size grid:

$$Q2_i = \sum_i q2_i \Delta d_i = Q2_{i-1} + q2_i \Delta d_i$$

vector<double> ConvertQ3Toq3 (Grid, InDistr)

Performs conversion from Q3 to q3 distributions using information about the size grid:

$$q3_0 = \frac{Q3_0}{\Delta d_i}$$

$$q3_i = \frac{Q3_i - Q3_{i-1}}{\Delta d_i}$$

vector<double> Convertq3ToQ3 (Grid, InDistr)

Performs conversion from q3 to Q3 distributions using information about the size grid:

$$Q3_i = \sum_i q3_i \Delta d_i = Q3_{i-1} + q3_i \Delta d_i$$

vector<double> Convertq0Toq2 (Grid, InDistr)

Performs conversion from q0 to q3 distributions using information about the size grid:

$$q2_i = \frac{d_i^2 q0_i}{M_2(q0)}$$

vector<double> Convertq0Toq3 (Grid, InDistr)

Performs conversion from q0 to q3 distributions using information about the size grid:

$$q3_i = \frac{d_i^3 q0_i}{M_3(q0)}$$

vector<double> Convertq2Toq0 (Grid, InDistr)

Performs conversion from q0 to q3 distributions using information about the size grid:

$$q0_i = \frac{d_i^{-2} q2_i}{M_{-2}(q2)}$$

vector<double> Convertq2Toq3 (Grid, InDistr)

Performs conversion from q0 to q3 distributions using information about the size grid:

$$q3_i = \frac{d_i \ q2_i}{M_1(q2)}$$

vector<double> Convertq3Toq0 (Grid, InDistr)

Performs conversion from q3 to q0 distributions using information about the size grid:

$$q0_i = \frac{d_i^{-3} q3_i}{M_{-3}(q3)}$$

vector<double> Convertq3Toq2 (Grid, InDistr)

Performs conversion from q3 to q0 distributions using information about the size grid:

$$q2_i = \frac{d_i^{-1} q3_i}{M_{-1}(q3)}$$

vector<double> ConvertMassFractionsToq0 (Grid, InDistr)

Calculates q0 distribution using the functions ConvertMassFractionsToq3() and Convertq3Toq0().

vector<double> ConvertMassFractionsToQ0 (Grid, InDistr)

Calculates Q0 distribution using the functions ConvertMassFractionsToq0() and Convertq0ToQ0().

vector<double> ConvertMassFractionsToq2 (Grid, InDistr)

Calculates q0 distribution using the functions ConvertMassFractionsToq3() and Convertq3Toq2().

vector<double> ConvertMassFractionsToQ2 (Grid, InDistr)

Calculates Q0 distribution using the functions ConvertMassFractionsToq2() and Convertq2ToQ2().

vector<double> ConvertMassFractionsToq3 (Grid, InDistr)

Calculates q3 distribution using the size grid and the distribution of mass fractions:

$$q3_i = \frac{w_i}{\Delta d_i}$$

vector<double> ConvertMassFractionsToQ3 (InDistr)

Calculates Q3 distribution using the distribution of mass fractions:

$$Q3_0 = w_i$$

$$Q3_i = Q3_{i-1} + w_i$$

vector<double> Convertq0ToMassFractions (Grid, InDistr)

Calculates mass fractions from q0 distribution using the functions Convertq0Toq3() and Convertq3ToMassFractions().

vector<double> ConvertQ0ToMassFractions (Grid, InDistr)

Calculates mass fractions from Q0 distribution using the functions ConvertQ0Toq0() and Convertq0ToMassFractions().

vector<double> Convertq2ToMassFractions (Grid, InDistr)

Calculates mass fractions from q2 distribution using the functions Convertq2Toq3() and Convertq3ToMassFractions().

vector<double> ConvertQ2ToMassFractions (Grid, InDistr)

Calculates mass fractions from Q2 distribution using the functions ConvertQ2Toq2() and Convertq2ToMassFractions().

vector<double> Convertq3ToMassFractions (Grid, InDistr)

Calculates mass fractions from q3 distribution using the size grid:

$$w_i = q3_i \Delta d_i$$

vector<double> ConvertQ3ToMassFractions (InDistr)

Calculates mass fractions from Q3 distribution using the size grid:

$$w_0 = Q3_0$$

$$w_i = Q3_i - Q3_{i-1}$$

vector<double> ConvertNumbersToq0 (Grid, InDistr)

Calculates q0 distribution using the number distribution and the size grid:

$$q0_i = \frac{N_i}{\Delta d_i N_{tot}}$$

vector<double> ConvertNumbersToQ0 (Grid, InDistr)

Calculates Q0 distribution using the number distribution and the functions ConvertNumbersToq0() and Convertq0ToQ0().

vector<double> ConvertNumbersToq2 (Grid, InDistr)

Calculates q2 distribution using the number distribution and the functions ConvertNumbersToQ2() and ConvertQ2Toq2().

vector<double> ConvertNumbersToQ2 (Grid, InDistr)

Calculates Q2 distribution using the number distribution and the size grid:

$$Q2_i = \frac{\sum_{j=0}^{i} N_j \pi d_j^2}{\sum_j N_j \pi d_j^2}$$

vector<double> ConvertNumbersToq3 (Grid, InDistr)

Calculates q3 distribution using the number distribution and the functions ConvertNumbersToq0() and Convertq0Toq3().

vector<double> ConvertNumbersToQ3 (Grid, InDistr)

Calculates Q3 distribution using the number distribution and the functions ConvertNumbersToq3() and Convertq3ToQ3().

vector<double> ConvertNumbersToMassFractions (Grid, InDistr)

Calculates mass fractions from the number distribution using the functions ConvertNumberToq0() and Convertq0ToMassFractions().

vector<double> Convertq0Toq0 (OldGrid, OldDistr, NewGrid)

Converts q0 distribution to the same distribution on the modified size grid.

vector<double> Convertq2Toq2 (OldGrid, OldDistr, NewGrid)

Converts q2 distributions to the same distribution on the modified size grid.

vector<double> Convertq3Toq3 (OldGrid, OldDistr, NewGrid)

Converts q3 distributions to the same distribution on the modified size grid.

NormalizeDensityDistribution (Grid, qiDistr)

Normalizes density distribution q0 or q3:

$$q_i = \frac{q_i}{\sum_j q_j \Delta d_j}$$

double GetDistributionMedian (Grid, QxDistr)

Returns median [m] of Q0 or Q3 distribution. Median is a diameter, which corresponds to a value of distribution equal to 0.5.

double GetDistributionValue (Grid, QxDistr, Val)

Returns diameter [m], which corresponds to a specified value of cumulative distribution Q0 or Q3. Input value should be: $0 \leq \mathrm{Val} \leq 1$.

double GetDistributionMode (Grid, qxDistr)

Returns diameter [m], which corresponds to a maximum value of density distribution.

double GetAverageDiameter (Grid, qxDistr)

Returns average diameter [m] of the distribution q0 or q3.

double GetSauterDiameter (Grid, q3Distr)

Calculates Sauter diameter (d32) of q3 distribution [m].

double GetSpecificSurface (Grid, q3Distr)

Calculates specific surface of q3 distribution [m^2].