Package: AeroSampleR (via r-universe)

September 5, 2024

Type Package

Title Estimate Aerosol Particle Collection Through Sample Lines

Version 0.2.0

Author Mark Hogue

Maintainer Mark Hogue <mark.hogue.chp@gmail.com>

Description Estimate ideal efficiencies of aerosol sampling through sample lines. Functions were developed consistent with the approach described in Hogue, Mark; Thompson, Martha; Farfan, Eduardo; Hadlock, Dennis, (2014), ``Hand Calculations for Transport of Radioactive Aerosols through Sampling Systems'' Health Phys 106, 5, S78-S87, <doi:10.1097/HP.0000000000000092>.

License GPL-3

Encoding UTF-8

LazyData true

Imports dplyr, purrr, tidyr, ggplot2, ggthemes, stringr, tidyselect, flextable

RoxygenNote 7.2.1

Depends R (>= 3.4)

Suggests knitr, rmarkdown

VignetteBuilder knitr

Repository https://markhogue.r-universe.dev

RemoteUrl https://github.com/markhogue/aerosampler

RemoteRef HEAD

RemoteSha f5c3436b7549f29893f4ebd2061d50f2d781437e

Contents

`AeroSampleR` *Estimate Aerosol Particle Collection Through Sample Lines*

Description

This package provides a method to estimate sampling efficiency of sampling systems drawing aerosol particles through tubing.

Details

Functions were developed consistent with the approach described in Hogue, Mark; Thompson, Martha; Farfan, Eduardo; Hadlock, Dennis, (2014), "Hand Calculations for Transport of Radioactive Aerosols through Sampling Systems" Health Phys 106, 5, S78-S87, <doi:10.1097/HP.0000000000000092>.

To learn how to use AeroSampleR, start with the vignette: ‘browseVignettes(package = "AeroSampleR")‘

`bend_eff` *bend efficiency*

Description

In order to run this function, first produce a particle distribution with the ‘particle_dist‘ function, then produce a parameter set with the ‘set_params‘ function. Both of these results must be stored as per examples described in the help set with each.

Usage

```
bend_eff(df, params, method, bend_angle, bend_radius, elnum)
```

Arguments

df	is the particle data set (data frame) established with the ‘particle_dist‘ function
params	is the parameter data set for parameters that are not particle size-dependent
method	choice of models: Pui, McFarland, or Zhang
bend_angle	bend angle in degrees
bend_radius	bend radius in m
elnum	element number to provide unique column names

Value

data frame containing original particle distribution with added data for this element

References

A. R. McFarland, H. Gong, A. Muyshondt, W. B. Wente, and N. K. Anand Environmental Science & Technology 1997 31 (12), 3371-3377 <doi:10.1021/es960975c>

Pusheng Zhang, Randy M. Roberts, André Bénard, Computational guidelines and an empirical model for particle deposition in curved pipes using an Eulerian-Lagrangian approach, Journal of Aerosol Science, Volume 53, 2012, Pages 1-20,ISSN 0021-8502, <doi:10.1016/j.jaerosci.2012.05.007>

David Y. H. Pui, Francisco Romay-Novas & Benjamin Y. H. Liu (1987) Experimental Study of Particle Deposition in Bends of Circular Cross Section, Aerosol Science and Technology, 7:3, 301-315, <doi:10.1080/02786828708959166>

Examples

```
df <- particle_dist() # set up particle distribution
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 100,
"T_C" = 25, "P_kPa" = 101.325) #example system parameters
df <- set_params_2(df, params) #particle size-dependent parameters
df <- probe_eff(df, params, orient = 'h') #probe orientation - horizontal
df <- bend_eff(df, params, method='Zhang', bend_angle=90,
bend_radius=0.1, elnum=3)
head(df)
```

---

dat_for_plots *Data from readme file for use in plot examples*

Description

This data was created by running the readme script. It is needed for simple plot examples.

Usage

```
dat_for_plots
```

Format

A `data.frame`

D_p particle diameter in micrometers

dens probability density

dist either log_norm or discrete

C_c Cunningham slip correction factor

v_ts particle terminal velocity

Re_p Reynold's number for particle

Stk Stokes' number for particle

eff_probe aspiration efficiency for probe

eff_bend_2 transport efficiency for the second component, a bend

eff_tube_3 transport efficiency for the third component, a straight tube

---

`particle_dist` *Create a particle distribution*

Description

Needed as a first step in estimating system efficiency. Make the data frame that will be used to estimate efficiency of variously sized aerosol particles' transport through the sampling system. To create your data, save this data to the global environment as shown in the examples.

Usage

```
particle_dist(
  AMAD = 5,
  log_norm_sd = 2.5,
  log_norm_min = 5e-04,
  log_norm_max = 100,
  discrete_vals = c(1, 5, 10)
)
```

Arguments

`AMAD`	default is 5 based on ICRP 66
`log_norm_sd`	default is 2.5 based on ICRP 66
`log_norm_min`	default is 0.0005 based on ICRP 66
`log_norm_max`	default is 100 based on ICRP 66
`discrete_vals`	default is c(1, 5, 10)

Details

All inputs are in micron AMAD, meaning: the aerodynamic diameter of a particle is the diameter of a standard density (1000 kg/m3) sphere that has the same gravitational settling velocity as the particle in question.

Value

a data frame containing a lognormally distributed set of particles and discrete particle sizes

Examples

```
df <- particle_dist() # default
df <- particle_dist(AMAD = 4.4,
                    log_norm_sd = 1.8)
head(df)
```

---

probe_eff *Probe efficiency*

Description

In order to run this function, first produce a particle distribution with the ‘particle_dist‘ function, then produce a parameter set with the ‘set_params‘ function. Both of these results must be stored as per examples described in the help set with each.

Usage

```
probe_eff(df, params, orient = "u", method = "blunt pipe")
```

Arguments

df	is the particle data set (data frame) established with the ‘particle_dist‘ function
params	is the parameter data set for parameters that are not particle size-dependent
orient	orientation of the probe. Options are ’u’ for up, ’d’ for down, and ’h’ for horizontal
method	is the model for the probe efficiency. Default is ’blunt pipe’, based on Su WC and Vincent JH, Towards a general semi-empirical model for the aspiration efficiencies of aerosol samplers in perfectly calm air, Aerosol Science 35 (2004) 1119-1134

Value

data frame containing original particle distribution with added data for this element

Examples

```
df <- particle_dist() # set up particle distribution
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 100,
"T_C" = 25, "P_kPa" = 101.325) #example system parameters
df <- set_params_2(df, params) #particle size-dependent parameters
df <- probe_eff(df, params, orient = 'u') #probe orientation - draws upward
head(df)
```

---

report_basic *report on transport efficiency*

Description

In order to run a report, first produce a model of each individual element. Start with producing a particle distribution with the ‘particle_dist‘ function, then produce a parameter set with the ‘set_params‘ function. Both of these results must be stored as per examples described in the help set with each. Next, add elements in the sample system until all are complete.

Usage

```
report_basic(df, params, dist)
```

Arguments

`df`	is the particle data set (data frame) established with the ‘particle_dist‘ function
`params`	is the parameter data set for parameters that are not particle size-dependent
`dist`	selects the distribution for the report. Options are ’discrete’ for discrete particle sizes or ’log’ for the log-normal distribution of particles that were started with the ‘particle_dist‘ function.

Value

report of system efficiency

Examples

```
df <- particle_dist() # set up particle distribution
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 100,
"T_C" = 25, "P_kPa" = 101.325) #example system parameters
df <- set_params_2(df, params) #particle size-dependent parameters
df <- probe_eff(df, params, orient = 'h') #probe orientation - horizontal
df <- bend_eff(df, params, method='Zhang', bend_angle=90,
bend_radius=0.1, elnum=3)
df <- tube_eff(df, params, L = 100,
angle_to_horiz = 90, elnum = 3)
report_basic(df, params, dist = 'discrete')
```

`report_cum_plots` *report on cumulative transport system efficiency (discrete particle sizes only)*

Description

In order to run a report, first produce a model of each individual element. Start with producing a particle distribution with the ‘particle_dist‘ function, then produce a parameter set with the ‘set_params‘ function. Both of these results must be stored as per examples described in the help set with each. Next, add elements in the sample system until all are complete.

Usage

```
report_cum_plots(df, micron)
```

Arguments

`df`	is the particle data set - after transport analysis by element
`micron`	selects the particle size (aerodynamic mass activity diameter in micrometers). This must be selected from the original distribution of particles that were started with the ‘particle_dist‘ function.

Value

A plot of cumulative transport efficiencies is generated in a plot window

Examples

```
report_cum_plots(dat_for_plots, micron = 10)
```

`report_log_mass` *report relative masses by particle of a log-normal distribution*

Description

This function shows the entire table of results by particle diameter.

Usage

```
report_log_mass(df)
```

Arguments

`df`	is the particle data set - after transport analysis by element

Value

data frame containing mass-based particle fractions in ambient location and in distribution delivered through the system.

Examples

```
df <- particle_dist() # set up particle distribution
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 100,
"T_C" = 25, "P_kPa" = 101.325) #example system parameters
df <- set_params_2(df, params) #particle size-dependent parameters
df <- probe_eff(df, params, orient = 'h') #probe orientation - horizontal
df <- bend_eff(df, params, method='Zhang', bend_angle=90,
bend_radius=0.1, elnum=3)
df <- tube_eff(df, params, L = 100,
angle_to_horiz = 90, elnum = 3)
report_log_mass(df)
```

---

report_plots *plots of individual on transport system elements*

Description

In order to run a report, first produce a model of each individual element. Start with producing a particle distribution with the ‘particle_dist‘ function, then produce a parameter set with the ‘set_params‘ function. Both of these results must be stored as per examples described in the help set with each. Next, add elements in the sample system until all are complete.

Usage

```
report_plots(df, dist)
```

Arguments

`df`	is the particle data set - after transport analysis by element
`dist`	selects the distribution for the report. Options are ’discrete’ for discrete particle sizes or ’log’ for the log-normal distribution of particles that were started with the ‘particle_dist‘ function.

Value

A plot of transport efficiencies is generated in a plot window

Examples

```
report_plots(dat_for_plots, dist = 'discrete')
```

set_params_1 — *Set parameters (not particle size specific)*

Description

Make a set of parameters that will be used throughout this package. ‘set_params_1‘ sets all single parameters. ‘set_params_2‘ adds particle-size-dependent parameters to the particle distribution

Usage

```
set_params_1(D_tube_cm, Q_lpm, T_C = 20, P_kPa = 101.325)
```

Arguments

`D_tube_cm`	Inside diameter of tubing in cm, no default
`Q_lpm`	System flow in lpm, no default
`T_C`	System temperature in Celsius
`P_kPa`	System pressure in kPa (Pa is the MKS unit)

Details

All parameters are to be in MKS units, except as noted.

Value

a data frame with singular parameters

examples params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 100, "T_C" = 25, "P_kPa" = 101.325) t(params)

set_params_2 — *Make a set of particle-size-dependent parameters*

Description

This set of parameters will be used for evaluation of transport efficiency for particle-size-dependent parameters.

Usage

```
set_params_2(df, params)
```

Arguments

`df`	is the particle data set (data frame) established with the ‘particle_dist‘ function
`params`	is the parameter data set for parameters that are not particle size-dependent

Details

No user-selected arguments are needed. Parameters are used in efficiency functions. For each particle diameter, an entry is made in the data frame for the Cunningham slip correction factor, the particle terminal velocity, the particle Reynold's number, and the Stokes factor.

'set_params_1' sets all single parameters. 'set_params_2' adds particle size-dependent parameters to the particle distribution

Value

a data frame starting with the submitted particle distribution with additional columns for particle-size-dependent parameters

Examples

```
df <- particle_dist()
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 100,
"T_C" = 25, "P_kPa" = 101.325)
df <- set_params_2(df, params)
head(df)
```

---

tube_eff *Tube efficiency*

Description

Computation is consistent with the approach described in Hogue, Mark; Thompson, Martha; Farfan, Eduardo; Hadlock, Dennis, (2014), "Hand Calculations for Transport of Radioactive Aerosols through Sampling Systems" Health Phys 106, 5, S78-S87, <doi:10.1097/HP.0000000000000092>, with the exception that the diffusion deposition mechanism is included.

Usage

```
tube_eff(df, params, L_cm, angle_to_horiz, elnum)
```

Arguments

`df`	is the particle data set (data frame) established with the 'particle_dist' function
`params`	is the parameter data set for parameters that are not particle size-dependent
`L_cm`	tube length, cm
`angle_to_horiz`	angle to horizontal in degrees
`elnum`	element number to provide unique column names

Details

In order to run this function, first produce a particle distribution with the 'particle_dist' function, then produce a parameter set with the 'set_params' function. Both of these results must be stored as per examples described in the help set with each.

Value

data frame containing original particle distribution with added data for this element

Examples

```
# Example output is a sample of the full particle data set.

# laminar flow (Reynolds number < 2100)

df <- particle_dist() #  distribution
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 20,
"T_C" = 25, "P_kPa" = 101.325) #example system parameters
df <- set_params_2(df, params) #particle size-dependent parameters
df <- probe_eff(df, params, orient = 'h') #probe orientation - horizontal
df <- tube_eff(df, params, L_cm = 100,
angle_to_horiz = 90, elnum = 2)
(df[sort(sample(1:1000, 10)), ])
# turbulent flow (Reynolds number > 4000)

df <- particle_dist() #  distribution
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 100,
"T_C" = 25, "P_kPa" = 101.325) #example system parameters
df <- set_params_2(df, params) #particle size-dependent parameters
df <- probe_eff(df, params, orient = 'h') #probe orientation - horizontal
df <- tube_eff(df, params, L_cm = 100,
angle_to_horiz = 90, elnum = 2)
(df[sort(sample(1:1000, 10)), ])

# midrange flow (Reynolds number > 2100 and < 4000)

df <- particle_dist() #  distribution
params <- set_params_1("D_tube" = 2.54, "Q_lpm" = 60,
"T_C" = 25, "P_kPa" = 101.325) #example system parameters
df <- set_params_2(df, params) #particle size-dependent parameters
df <- probe_eff(df, params, orient = 'h') #probe orientation - horizontal
df <- tube_eff(df, params, L_cm = 100,
angle_to_horiz = 90, elnum = 2)
(df[sort(sample(1:1000, 10)), ])
```

Index