Flume experiment code written by Justin Nghiem, April 26, 2018; updated May 16, 2018; updated May 31, 2018; updated by Colin Keating June 19th, 2018

The code for processing LISST, sediment trap, and peristaltic pump data is contained in seven R scripts: LISST\_processing.R, k\_calculation.R, bin\_visualization.R, particle\_size\_diagnostics.R, settling\_velocity\_kc\_estimates.R, peristaltic\_pump\_k\_estimation.R, and peristaltic\_pump\_plotter.R. The required R packages are dplyr, magrittr, ggplot2, tidyr, ModelMetrics, and cowplot. If you are missing a package, executing the function install.packages("name of package in quotes") in the console will install it.

Preliminaries:

1. peristaltic pump data

The peristaltic pump data must be in csv format. The data must be arranged in four columns. The headers should be "time," "height," "location," and "concentration" in that order. The first column should contain time of observation (min) for each data point. The second column should contain the height (cm) from the bed for each data point. The third column should contain the location (U = upstream; D = downstream) for each data point. The fourth column should contain the mass concentration (g/L) for each data point.

2. sediment trap data

The sediment trap data must also be in csv format. The data must be arranged in two columns. The headers should be "position" and "mass" in that order. The first column should contain the sediment trap label (e.g. A, B, C, so on). The second column should contain the mass collected (g) from each sediment trap.

Scripts:

1. LISST\_processing.R

This script produces 4 plots from the LISST data, 1) cumulative density, 2) unsmoothed density, 3) smoothed density, and 4) smoothed and normalized density, in a single frame. All information in the plots is aggregated into time intervals, which may be set in the script. The options for the script appear in lines 10-12.

Options:

line 10: file change this variable to the absolute file path of the LISST data

line 11: correction set this variable to TRUE to remove all data points with too high or low transmission; set to FALSE to retain all data points

line 12: time\_interval set this variable to the desired length of data aggregation (in minutes)

Outputs:

plot of particle size distribution densities

2. k\_calculation.R

This script estimates k, the particle capture rate, using nonlinear least squares assuming an exponential decay model of particle concentration (fit may be poor or an error may occur if the data do not follow this assumption) and gives a plot to show the fit vs the actual data. The last portion of the code also calculates effective capture efficiency. The options for the script appear in lines 4-10. Note that the options in lines 9 and 10 are used to compute effective capture efficiency, so you do not need to specify them if you are only interested in k.

Options:

line 4: file change this variable to the absolute file path of the LISST data

line 5: correction set this variable to TRUE to remove all data points with too high or low transmission; set to FALSE to retain all data points

line 6: bin\_k set this variable to the number of the bin for which you want to calculate k (1 to 32)

line 7: time\_start set this variable to the time (in seconds) you want for the starting point for the data used in the estimation (default is 0)

line 8: time\_end set this variable to the time (in seconds) you want for the ending point for the data used in the estimation (default is Infinity, e.g. goes to end of data)

line 9: collector\_density set this variable to the collector density per area in the test section (in m^-2)

line 10: flow\_velocity set this variable to the flume flow velocity (m/s)

Outputs:

k\_exp: estimated particle capture rate k from nonlinear least squares (1/s)

initial\_concentration\_exp: estimated initial particle concentration for the specified bin (microliter/liter)

effective\_capture\_efficiency: estimated effective capture efficiency (%)

plot of fitted exponential to data

3. k\_calculation\_all\_bins.R

This script is similar to k\_calculation.R, except that it fits an exponential decay model on the mean across all bins instead of on a single bin.

4. bin\_visualization.R

This script produces a large frame of 32 plots showing the particle concentration (microliters/liter) over time (seconds) in each bin. The upper-left plot corresponds to bin 1 while the lower-right plot corresponds to bin 32. Bin number increases from left to right. The options appear in lines 4-7.

Options:

line 4: file change this variable to the absolute file path of the LISST data

line 5: correction set this variable to TRUE to remove all data points with too high or low transmission; set to FALSE to retain all data points

line 7: time\_start set this variable to the time (in seconds) you want for the starting point for the data used in the plots (default is 0)

line 8: time\_end set this variable to the time (in seconds) you want for the ending point for the data used in the plots (default is Infinity, e.g. goes to end of data)

Outputs:

plot of particle concentration over time for each bin

5. particle\_size\_diagnostics.R

This script produces three plots from the LISST data, 1) weighted average particle size over time (for each time point, compute weighted average particle size by summing the products of a bin's median particle size and the volume proportion of that bin out of the total volume concentration at that time), 2) cumulative density at each time point (blue corresponds to earlier times, red to later times), and 3) total particle concentration over time. The options appear in lines 10-11.

Options:

line 10: file change this variable to the absolute file path of the LISST data

line 11: correction set this variable to TRUE to remove all data points with too high or low transmission; set to FALSE to retain all data points

Outputs:

p1: plot of weighted average particle size over time (weights produced by volume proportion)

p2: plot of empirical CDF at finest temporal resolution

p3: plot ot total particle concentration over time

6. settling\_velocity\_kc\_estimates.R

This script estimates particle capture rate due to collectors based on sediment trap data and Stokes' law. The options appear in lines 5-11.

Options:

line 5: bin set this variable to the bin that you used to calculate k (in k\_calculation.R)

line 6: depth set this variable to the water depth in the flume experiment (cm)

line 7: particle\_density set this variable to the particle density (kg/m^3)

line 8: total\_time set this variable to the time in over which particles settled in sediment traps (s)

line 9: sediment\_trap\_data change this variable to the absolute file path of the csv file containing sediment trap data

line 12: source() change the argument in this function to the absolute file path of k\_calculation.R on your computer

Outputs:

ST\_estimates: data frame of settling velocities (m/s) and kc (1/s) for all sediment traps

ks\_stokes: Stokes' law estimate of ks for given bin (1/s)

kc\_stokes: kc computed using Stokes' law estimate for ks (1/s)

plots of settling velocity estimates

7. peristaltic\_pump\_k\_estimation.R

This script estimates the particle capture rate k by fitting an exponential model to the peristaltic pump data with nonlinear least squares. The options for the script appear in lines 5-9.

Options:

line 5: file change this variable to the absolute file path of the peristaltic pump data

line 6: loc set this variable to the location in the flume for which you want to estimate k ("upstream" or "downstream")

line 7: ht set this variable to the height in the flume for which you want to estimate k (5, 14, or 27)

line 8: start\_time set this variable to the start time you want for data analysis

line 9: leak disregard this variable

Outputs:

k\_final: final estimate of k for the given height and location

mse: mean-squared error of the exponential curve fit to the data

plot of the model fit and the data is also produced

8. peristaltic\_pump\_plotter.R

This script produces a general plot of peristaltic pump data for all combinations of height and location. The option for the script appears in line 5.

Options:

line 5: file change this variable to the absolute file path of the peristaltic pump data

Outputs:

plot of mass concentration for each height and location

9. mass\_balance.R

This script estimates the mass settled on the test section, the total mass removed (both settling and capture on collectors), and (by subtraction) the mass captured on collectors. The options for the script appear in lines 6-9.

Options:

line 6: pp change this variable to the absolute file path of the peristaltic pump data

line 7: st change this variable to the absolute file path of the sediment trap data

line 8: discharge set this variable to the discharge in the flume (L/s)

line 9: leak disregard this variable

Outputs:

captured\_mass: estimated total mass (capture from settling and on collectors) (g)

collected\_mass: estimated mass directly intercepted by collectors (g)

settled\_mass: estimated mass settled out over test section (g)

Notes:

These scripts assume that the transmission in the LISST data is between 0 and 1. If you are using Kristen Fauria/Stephen Andrews' MATLAB code to process the LISST raw data, the transmission in the processed data will have transmission multiplied by 100 (e.g. it is between 0 and 100). You can compensate for this by appending %>% mutate(V41=V41/100) to the line in each script that reads raw <- read.table(file, header=FALSE).

If you have any further questions, please email Justin Nghiem at justinnghiem@berkeley.edu