GEOLOGY
&GEOPHYSICS

ESCUELA
POLITÉCNICA
NACIONAL

E SCIENTIA HOMINIS SALUS

THE
UNIVERSITY
OF UTAH

PARTI-SUITE
PARTICLE TRAJECTORY & CONTINUUM UPSCALING COLLOID TRANSPORT

Abstract

General Description of Parti-Suite
Colloid Transport Codes

William P. Johnson and Eddy F. Pazmino
William.johnson@utah.edu; eddy.pazmino@epn.edu.ec

Parti-Suite
User Manual
Particle Trajectory and Continuum Upscaling Colloid Transport Codes

W.P. Johnson[1] and E. Pazmino[2] Research Groups
[1]University of Utah, USA, Dept. of Geology & Geophysics
[2]Escuela Politecnica Nacional, Ecuador, Dept. of Extractive Metallurgy
william.johnson@utah.edu; eddy.pazmino@epn.edu.ec

Parti-Suite runs on Windows platforms, and includes:

1) Pore scale Lagrangian colloid trajectory simulations with full force and torque balance in three collector geometries:
    a. Traj-Hap: Happel sphere-in-cell collector.
    b. Traj-Jet: impinging jet collector.
    c. Traj-Par: Parallel plate collector.

    All collector geometries include:
    d. Extended DLVO for layered and unlayered systems
    e. Discrete representative nanoscale heterogeneity (DRNH) for attachment and detachment under unfavorable (like-charged) conditions.
    f. Read-in of attachment simulations for subsequent detachment simulations.
2) Upscaling of pore scale colloid trajectories (residence times prior to attachment) to continuum-scale rate coefficients, and Lagrangian continuum-scale simulations for breakthrough-elution concentration histories and profiles of retained colloids with distance from source.
    a. UpscaleContinuum: utilizes residence times from Traj-Hap trajectory simulations to determine continuum-scale rate coefficients for transfer between bulk fluid domain, near surface fluid domain, and surface.
    b. UpscaleContinuum: includes Lagrangian simulator for breakthrough-elution concentration histories and profiles for distribution with distance from source. Options are provided for simulations with implicit and explicit near surface fluid domains.
3) Colloid-surface interaction energy and force profile calculator.
    a. X-DLVO: calculates particle-surface extended-DLVO interaction profile calculations:
        i. Sphere-plate and sphere-sphere geometries;
        ii. Layered and non-layered colloids and collectors;
        iii. Smooth and rough colloids and collectors.
4) Collector efficiency (η) calculations using published correlation equations established under favorable attachment conditions.
    a. EtaCorr: calculates collector efficiencies using several correlation equations from the literature.

Simulations from the software listed above is grounded by comparison to experiments, and is peer-reviewed, as described in Rasmuson et al., 2019; VanNess et al., 2019; and Ron et al., 2019, and references cited therein. Please see W.P. Johnson Research Group publications (http://www.wpjohnsongroup.utah.edu/) for the full set of publications.

Instructions:
Matlab Runtime 9.0 installers are provided in the main distribution drive. Install Matlab Runtime first, and then install the Matlab executables.

Following installation, the Matlab executables are run from (in windows) left-clicking the bottom left windows icon: right click on the recently added program>run as administrator. The command window opens first followed by the graphical user interface (GUI) input window. The GUI takes a few minutes to set up, so please be patient. The GUI has fixed resolution, so text may appear small on high resolution screens and may be cutoff on low resolution screens. Try changing screen resolution if these problems occur on your screen.

The GUI input window may be somewhat intimidating, so try a few runs with default parameters to see how the code and GUI outputs perform. As the simulations develop output GUI windows will display results, although asci files are also produced as described in the Parti-Suite-TrajectoryCodes-Descriptions.

As you change parameters, please keep in mind that these mechanistic force and torque balances are complex, and certain parameter combinations may yield slow simulations. As users test code, errors may arise, so you'll need to periodically check the command window (non-GUI window) for these errors. We'll fix these if you send the error message that occurs by highlighting the text in the command window, clicking right>edit>copy, preferably along with the written input file located as described in Parti-Suite-TrajectoryCodes-Descriptions.

The algorithm we've provided starts with pore scale mechanistic trajectory simulation (Traj-Hap or trajectory code in other collector geometry). The user specifies all physicochemical parameters, the least defined (currently) being nanoscale heterogeneity (heterodomain size and surface coverage under unfavorable attachment conditions. The user adjusts heterodomain size and surface coverage to match their observed collector efficiency (or collision efficiency) under unfavorable conditions (see for example Ron et al., 2019). The user also performs the simulation under favorable attachment conditions.

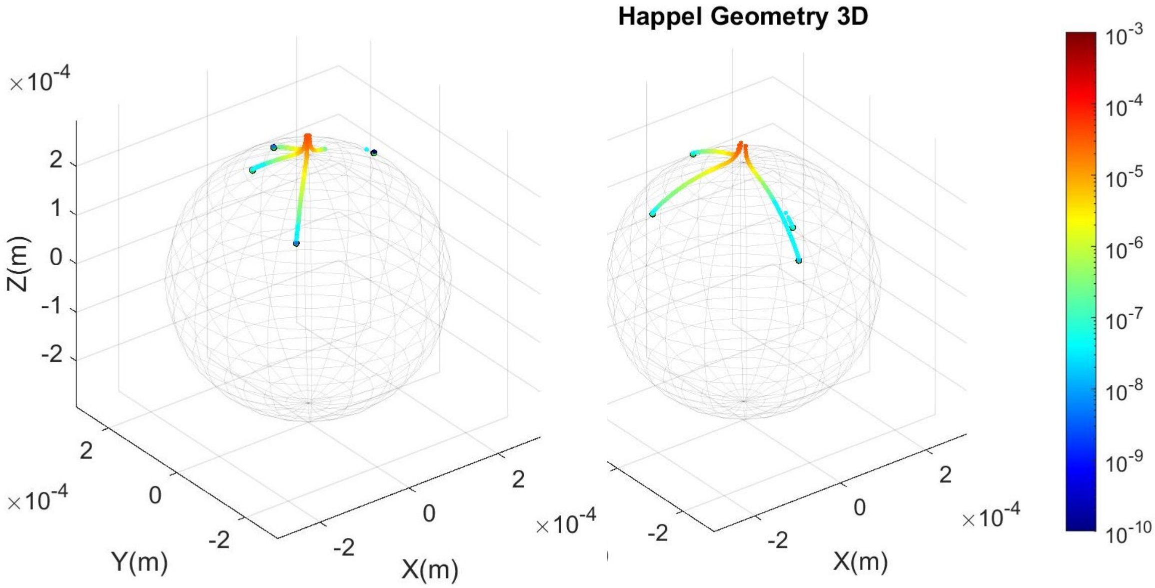

Figure 1: Pore scale trajectory simulation results for favorable conditions (left) and unfavorable conditions (right).

Once the favorable and unfavorable condition simulations are complete , the pore scale trajectory simulation outputs are utilized in UpscaleContinuum (currently configured for outputs from Traj-Hap only) to obtain near surface velocity, collector efficiency corresponding to favorable attachment conditions, and collision efficiencies for unfavorable attachment conditions, as described in Johnson et al. (2018), and in the UpscaleContinuum description manual. In UpscaleContinuum these parameters are upscaled to continuum rate coefficients which are then utilized in continuum scale Lagrangian transport simulations to produce breakthrough-elution concentration histories and profiles of retained colloid concentrations with distance from source.

Since ContinuumUpscale utilizes trajectory information from Traj-Hap, its utility increases with increased numbers of simulated particles in Traj-Hap.

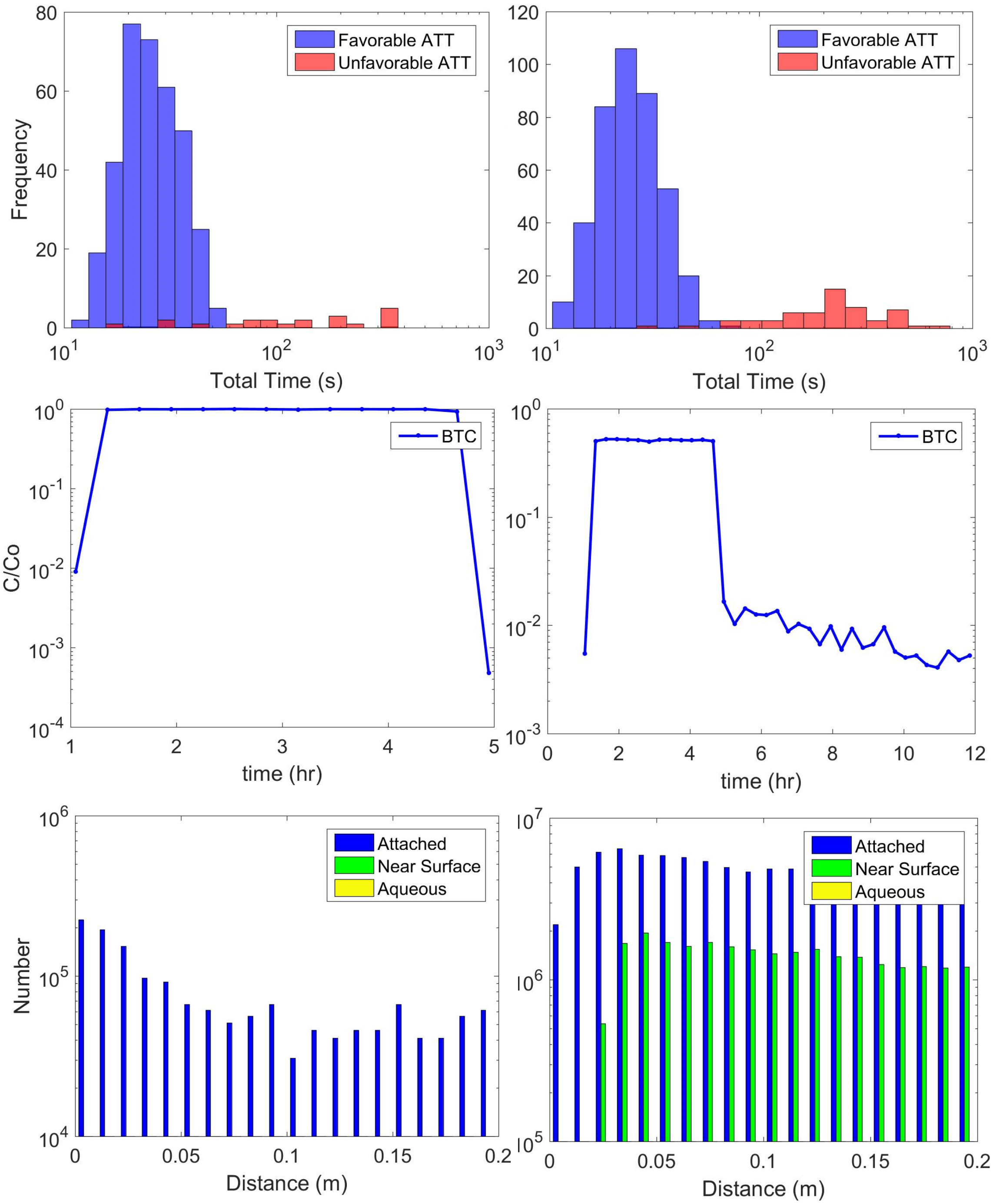

Figure 2: Histograms of residence time prior to attachment (top row), predicted breakthrough-elution concentration histories (middle row), and predicted profiles of retained colloids (bottom row) for 1 um coll. in glass beads (left), quartz sand (right), each at 4 m/day pore water velocity, 6 mM IS. See also Johnson et al., (2018).

We like to think of these codes as our little “Millenium Falcons”. Compared to commercial products they’re not very pretty, but there’s a lot under the hood. The complexity of the simulations requires careful consideration of potential matches and mis-matches between predictions and observations. We ask that you exercise good judgement in drawing conclusions. We welcome your questions and input.

The pore-scale Lagrangian trajectory simulations were originally written in Fortran, from which the codes were parallelized to allow simulation of simultaneous trajectories, since any given trajectory may require upwards of hundreds of trillions of time steps. The software is herein provided as ported MatLab executables for clear visualization of trajectory simulations of colloid populations numbering in the tens to hundreds depending on simulation parameters (which define computational intensity). Once the use has understood the Matlab executables, we encourage use of the Fortran executables, which allow simulation of populations of hundreds to thousands (depending on simulation conditions). To facilitate parallelized simulation of larger particle populations (thousands to tens of thousands) we intend to provide Python-based easily parallelized software in the near future (hopefully early 2020), but we are also willing to collaborate on Fortran-based parallel simulations on our platform for interested users.