**README related to the manuscript "Final nanoparticle size distribution under unusual parameter regimes", Sabbioni E., Szabò R., Siri P., Cappelletti D., Lente G., Bibbona E.**

**Corresponding authors**: Elena Sabbioni, elena.sabbioni@polito.it, Rebeka  Szabó, rebekasz@gamma.ttk.pte.hu

The simulation study can be reproduced using the Julia files:

- Gillespie.jl → simulate trajectories of the original process described in (7) in the paper using the Gillespie algorithm.  When the initial number of monomers is bigger than 10^5, we suggest not to run the code on your local machine, but to use a cluster where it can be run in parallel.
- Gillespie_trajectories.jl → simulate trajectories of the original process described in (7)  in the paper using the Gillespie algorithm, following the times at which the jumps occur and the number of available monomers at each jump time. When the initial number of monomers is bigger than 10^5, we suggest not to run the code on your local machine, but to use a cluster where it can be run in parallel.
- TauLeap.jl → simulate trajectories of the original process described in (7)  in the paper using the Tau-Leap with post-leap checks algorithm.  When the initial number of monomers is bigger than 10^5, we suggest not to run the code on your local machine, but to use a cluster where it can be run in parallel.
- ApproxGillespie.jl → simulate trajectories of the approximated process described in (13)  in the paper using the Gillespie algorithm.  When the initial number of monomers is bigger than 10^8, we suggest not to run the code on your local machine, but to use a cluster where it can be run in parallel.
- ApproxGillespie_trajectories.jl → simulate trajectories of the approximated process described in (13) in the paper using the Gillespie algorithm, following the times at which the jumps occur and the number of available monomers at each jump time. When the initial number of monomers is bigger than 10^8, we suggest not to run the code on your local machine, but to use a cluster where it can be run in parallel.
- ApproxTauLeap.jl → simulate trajectories of the approximated process described in (13) in the paper using the Tau-Leap with post-leap checks algorithm.  When the initial number of monomers is bigger than 10^12, we suggest not to run the code on your local machine, but to use a cluster where it can be run in parallel.

All these Julia files load the Julia package "npformation", that contains all the functions and the code that run the Gillespie and the  Tau-Leap algorithm for both the exact and the approximated process.

The folders "ApproxGillespie", "ApproxTauLeap", "Gillespie", "TauLeap" contain respectively the results of the simulations of the approximated process using the Gillespie's algorithm, of the approximated process using the Tau-Leap algorithm, of the exact process using the Gillespie's algorithm and of the exact process using the Tau-Leap algorithm. The folder "traj" contains the results of the simulations of the exact and of the approximated process in which we follow the times at which the jumps occur and the number of available monomers at each step.

The figures presented in the articles can be reproduced with the R files "figure1.r", "figure2.r", "figure3.r". The additional file "plots.r" can be used to produce additional figures (plots for the Tau-Leap counts'distributions,...), while the file "trajectoriesMon.r" can be used to plot the evolution of the monomers'count during the Gillespie algorithm, both for the original and the approximated process.