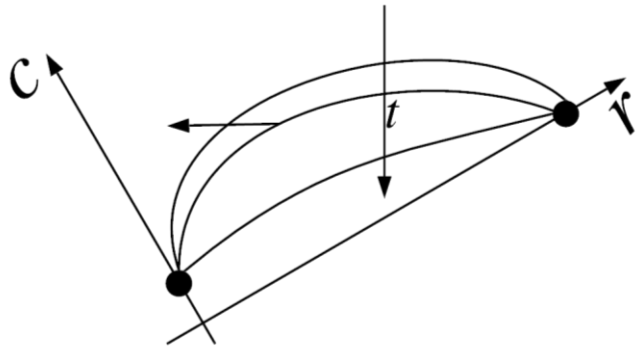
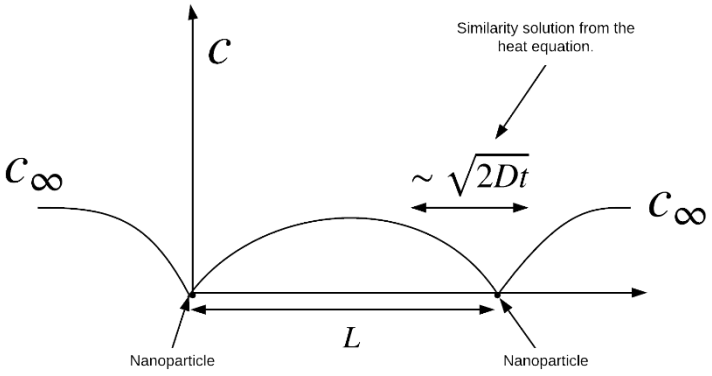
05-03-2020 Meeting Minutes – Nanoparticle Case Study

Supervisor Remarks

- For large times the distribution produced by the PDE model deviates from that produced by the R model.
    - Since the number of particles used in the R model is very large, it is unlikely that this discrepancy is due to numerical errors.
    - It may be worth checking for issues in the underlying mathematics of each model.
- Derivation of expression for $c_\infty(t)$, using a conservation of mass.
    - It is important to note that in reality the density of each nanoparticle is not necessarily constant (i.e. V_m might not be spatially constant).
- Both the R and N models make the assumption that the particles are sufficiently well separated.
    - Nanoparticles occupy a very small volume compares to the bath, so it is valid to assume that they are well separated.
    - This is why the approximation: $\int_{D_3} \hat{c}_3(r,t)dV \approx c_\infty(t)V_0$ is valid in the derivation of our expression for $c_\infty(t)$.
        - $V_0 \in \mathbb{R}$ denotes the volume of the region D_3.
    - Our assumption of well separated nanoparticles is not always physically valid.



        - Assumption is valid, provided that the concentration gradient is small.
        - However, when our gradient is very large then this assumption breaks down.
    - In our models we assume that particles have the same far field behaviour.
        - When this assumption is neglected the models become very complicated.



        - In a more general model, we can find the time at which we expect particles concentrations to interact, so can work out the time at which the particles start to interact.
            - Note: The governing equations become very complicated in such models, because the radial symmetry is broken.
- Asymptotics of the PDE: $\frac{\partial N}{\partial t}(r,t) + \frac{\partial}{\partial r}\left(f(r,t)N(r,t)\right) = 0$.
    - When $\frac{\partial f}{\partial r} \approx 0$, our equation emits travelling wave solutions.
        - During this stage, the distribution N is not focused (i.e. its mean size is changing, but its standard deviation remains constant).
    - Suppose that f has a steady state.
        - f attaining this steady state (i.e. $\frac{\partial f}{\partial r}$ hitting zero) may correspond to the sudden change in behaviour which is observed in the simulations.
    - For the case when $f\frac{\partial N}{\partial r} \approx 0$, we have $\frac{\partial N}{\partial t}(r,t) = -\frac{\partial f}{\partial r}(r)N(r,t)$.
        - In this case we cannot use the original starting distribution N_0 as the initial condition for the simplified PDE problem, since our simplified model is only valid for large time.
        - We need an additional initial condition from somewhere.
        - In this case we get that heavier particles decay faster, which explains the loss of symmetry in the distribution for large times.
    - Further asymptotics:
        - Our PDE and ODE models are both order 1 on both sides, so there is not an obvious small parameter that we can perturb around.
- Could also see if we can derive an ODE for the mean of the distribution.
    - How does $\mu := \int_{r_{min}}^{r_{max}} sN(s,t)ds$ evolve in time?
- Where can we find additional data to compare our model simulation results with?
    - Paper by Time Myers: https://www.sciencedirect.com/science/article/abs/pii/S0021979718312396?fbclid=IwAR10Cyco-l-8woIE9vGHB85wO5FsFvktaWPOVGDV0SoAp75XkjcFmIpIvU0.
- The main objective now is to come up with a conclusion to our original problem and work on producing the presentation.
    - Should look at injecting more solute at later times, by varying $c_\infty(0)$.
        - The best approach may be to simply experiment with injecting solute in different ways (e.g. injecting during the travelling wave phase or during the distribution broadening phase).
        - Could investigate how to optimise the way in which we vary $c_\infty(0)$ in order to produce lots of nanoparticles of a fixed size (e.g. 10 nanometres).
        - Focusing of the distribution.
            - We wish to narrow our distribution in order to produce lots of nanoparticles of the same size.
            - Good to know if we can identify the time at which distribution broadening occurs, so that we know when to stop the experiment.

- Presentation Remarks
    - Model derivation suggestions.
        - State the final governing equations for our model.
            - Don't need to include all the intermediate steps.
        - State any assumptions made in the derivation process.
        - State any corresponding boundary conditions.
        - State and apply the non-dimensionalisation scaling used for our governing equations.
    - Lots of intermediate steps may be overlooked as it is likely they will not be followed by the audience (given the time constraints).
    - It is important to convey the broad aims of the project and mathematical ideas.
    - Should use pictures instead of (or in combination with) equations, whenever possible.
    - For the derivation of the PDE model it would be useful to covey the general ideas, rather than all the precise steps (since it is quite involved).

Additional Notes

- Shyam has subsequently corrected the PDE model simulations so that its results are now consistent with the those from the R model, for very large times.
- All the current source code can be found at: https://github.com/WDash1/MMSC-Nanoparticles-Case-Study
    - The current source code is written in Python 3 and requires the latest version of the SciPy library.
        - The existing source code does not run on the Mathematical Institute Linux PCs.
        - SciPy can be installed in combination with Anaconda, using the hyperlink: https://www.anaconda.com/distribution/
    - All the source code files of interest are located in the directory: "full_runs/alternate" in the Git repository.
        - The file "finite_volume.py" corresponds to simulations of the PDE model.
            - This will output data files to the directory: "full_runs/alternate/fvm_data".
                - If this directory does not exist on your PC, then you will need to create it before running the python code.
        - The file "montecarlo.py" corresponds to simulations of the R model.
            - This will output data files to the directory: "full_runs/alternate/montecarlo_data".
                - If this directory does not exist on your PC, then you will need to create it before running the python code.
        - The file "post_compare.py" will read the data files written by the "montecarlo.py" and "finite_volume.py" programs, re-dimensionalise their data and display it on a graph for each time step.
            - The resulting graphs are written to the directory: "full_runs/alternate/comparison_images".
                - If this directory does not exist on your PC, then you will need to create it before running the python code.

        - It is important to note that you must run both the “montecarlo.py” and “finite_volume.py” programs before you attempt to run the “post_compare.py” program.
    - The file “params.py” specifies all the parameters used by each of the source code simulation files throughout, so parameters should only every be changed in this file.

Objectives

- Use some form of parameter fitting to find a value for D_a that should be used in our model, so that it matches the Peng data or other available data.
- Investigate whether adding solute can increase the time before the distribution starts decaying and broadening (can it prolong the travelling wave phase?).
- Investigate the different ways in which we can add solute to our system and which way is best to produce a narrow distribution of nanoparticles with a mean of 10 nanometres.
    - Could look at how we should add nanoparticles (all instantly or very slowly) to our system.
        - For quickly adding nanoparticles, we could discontinuously vary $c_\infty(t)$ in our simulations.
        - For slowly adding nanoparticles we could smoothly vary $c_\infty(t)$ with time.
        - Could look at using a sigmoid function $S(x) = \frac{1}{1+e^{-(ax+b)}}$ and tweaking the scaling and shifting parameters.
            - $a \in \mathbb{R}$ changes how rapid the transition from 0 to 1 is.
            - $b \in \mathbb{R}$ corresponds to the time at which a transition from 0 to 1 occurs.
    - Could look at when we should add more solute to the system.
        - Should we add them during the travelling wave phase?
        - Should we add them during the distribution broadening phase?
    - Could investigate how much solute we should add.
- All the presentation slides should be completed by no later than Monday morning.
- We should set some time aside next week to practice our presentation (e.g. Monday and Tuesday evening, so that we may refine our delivery).