Simulation of Dielectrophoretic Potential for Single Particles Trapping

July 14, 2024

C. Hamster

***Abstract*—Studying the behavior of individual proteins has significantly advanced drug design, disease diagnosis, and the development of novel nanotechnologies. Trapping proteins is a common research method, but existing techniques often chemically or physically alter their behavior. A promising new technique utilizing dielectrophoresis (DEP) offers label-free, active, detection-less, local trapping of proteins. While DEP has been widely used for cell manipulation, its application in single protein trapping is a recent development. For this method to be effective, a detailed understanding of the DEP potential within the trap is necessary. This requires a combination of experiments and simulations. Research by S. Pud et al. aims to experimentally determine the DEP potential for a dielectrophoretic trap using polystyrene nanoparticles. This paper presents a framework for modeling this setup in silico by solving the Langevin equation for DEP and Brownian motion. The framework will help determine the theoretical DEP potential and pave the way towards DEP study of single proteins.**

***Index Terms*—Dielectrophoresis, Simulation, Langevin eqaution, Brownian Motion, Trapping.**

I. Introduction

The study of single molecules has revolutionized modern biophysics. It enabled studying individual protein behavior, providing insights into fundamental biological mechanisms and driving advancements in drug design, disease diagnosis, and the development of novel nanotechnologies[3]. Over the last decades, many techniques established themselves to study single molecules[16]. One of the crucial factors influencing the accuracy of these techniques is the localization of the molecule under study[4]. Some methods use fluorescence to track the molecules, some confine the movement of the molecule using nanostructures or by attaching a chemical tether, and others confine the movement using external forces[5].

A new technique aims to trap molecules using dielectrophoresis (DEP). DEP is a dielectric force that carries the molecule towards places of field strength where the molecule's free energy is reduced[6]. Standard DEP theory defines it as a force acting on a moment in an electric field gradient. Improvements in manufacturing nanoelectrodes have made it possible to engineer them such that single molecule trapping has come into the realm of possibilities[12]. This method is label-free, active, does not depend on detection, and offers very localized trapping[2][10]. Furthermore, DEP actuation offers a way to study additional parameters of the molecule. Following procedures as proposed by Ma et al. [17], but replacing the tethers with DEP actuation, DEP parameters like size and the Clausius–Mossotti, $K(\omega)$ factor can be determined. This $K(\omega)$ describes the contrast between the medium and molecule in terms of their interaction with the electric field and can reveal information about its dipole moment[6].

Before this can be done, the DEP-induced trapping potential of the trap has to be determined. Simulation can help here by modeling the movement of a molecule in a trap model. This way, the trapping potential can be estimated. Comparing this estimation to real-life experiments can help create a more accurate model of the DEP potential.

This thesis will continue the work done by E. Schwander et al.[2]. Moving from 2D simulation to 3D, expanding on theory research and approximating the trapping potential of a simplified setup. The simulation represents the in silico experiment of the setup shown in Figure 1 and is described in P. Sergii et al. [10]. It aims to trap single spherical polystyrene nanoparticles with a radius between 20 and 50 nm. The design of two bow-tie-shaped nano-electrodes creates a strong and local electric field at the tips. The field induced is in the 10^{-6} to 10^{-7} V/m range for an AC voltage of 1V on the electrodes. This electric field creates a trapping potential strong enough for prolonged trapping of single proteins[19]. The electric field is simulated using COMSOL Multiphysics. This field is imported into a Python environment. In Python, a framework is built to calculate the dielectrophoretic force and model Brownian motion and collisions. Using a simplified electrode model, this framework's trapping potential is compared to its theoretical trapping potential. This way, the accuracy of the framework can be evaluated. It is important to note that in the real setup the molecules are detected using iScat technology. As this thesis aims to model only the DEP potential this part is outside the scope of research.

Ultimately, the research aims to study the DEP potential of the trap for complex molecules like proteins. This thesis, however, serves as a step in guiding the simulation framework in the right direction, providing a foundation that can be built upon.

II. Theory

The setup discussed in the introduction places polystyrene nanoparticles with a 20-50 nm radius in a fluid with two nano-electrodes. These nano-electrodes induce a strong electric field

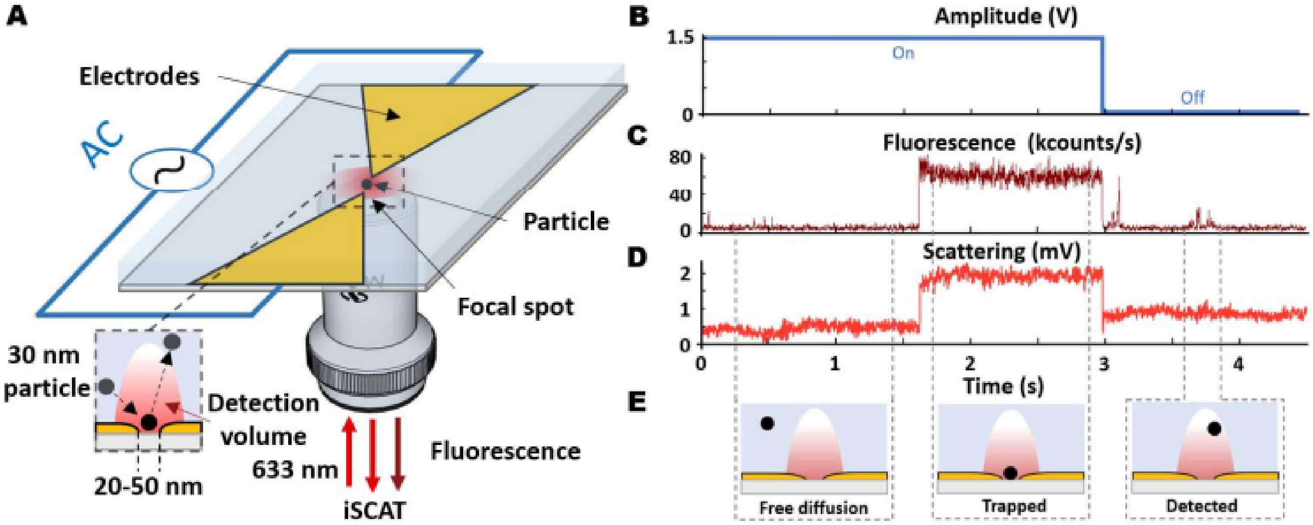

Fig. 1: A) This is the setup used by Sergii Pud et al.[2] that this simulation will be based on. It puts a AC voltage on electrodes with a 20-50 nm gap inbetween them to induce high electric field gradients. iScat dectection to measure the particles. B) The setup shows a detection event for a particle with a 30 nm radius.

gradient, causing a DEP trapping potential. The setup has been designed to stay clear from adverse effects like dielectric breakdown, electrolysis, heat release, reshaping or degradation of particles, or denaturation of proteins [18][10]. Considering this, the movement of the particles is dominated by two main forces: Brownian motion and the dielectrophoretic force. Understanding these forces is crucial for simulation. This section will tackle the theory regarding these forces and discuss the Langevin equation used to model them. Furthermore, it will tackle simplifying the Langevin equation and discuss a simplified electrode model that can be statically evaluated to assess the accuracy of the model.

A. *Brownian Motion*

Brownian motion is the random movement of a particle suspended in a fluid. The thermal energy of the molecules in the fluid causes them to rush and collide with each other. The net force of these collisions creates a random walk. The equipartition theorem describes a molecule's kinetic energy in a fluid, which states that every degree of freedom contributes $\frac{1}{2}k_bT$ of energy. In 3 dimensions, that results in an average translational kinetic energy of:

$$E_p = \frac{3}{2}k_bT \tag{1}$$

where k_b is the Boltzmann constant and T is the temperature. According to A. Einstein the mean displacement of such particle when only taking into account Brownian motion and Stokes friction law is[1]:

$$\langle \mathbf{r}^2 \rangle = 6Dt \tag{2}$$

Where $\mathbf{r}$ is position, t is time, and D is the diffusion coefficient given by Stokes-Einsteins relation[1]:

$$D = \frac{k_bT}{\gamma} \tag{3}$$

Where γ is the stokes drag given by:

$$\gamma = 6\pi\eta R \tag{4}$$

Where η is the viscosity of the surrounding fluid, and R is the particle's radius. So, the movement is affected by temperature T and particle size R.

1) Langevin Equation: The random walk of Brownian particles can be modeled using stochastic differential equations. The Langevin equation is an equation that is used to model Brownian motion. It describes the evolution of a system under the influence of deterministic and fluctuating ("random") forces. For Brownian motion in one dimension, the Langevin equation looks like[2]:

$$m\frac{d^2x(t)}{dt^2} = -\gamma\frac{dx(t)}{dt} + \xi(t) \tag{5}$$

Where ξ is a function representing Gaussian white noise responsible for the random motion of the particle. The magnitude of this white noise is given by $\sqrt{2k_bT\gamma}W(t)$ where $W(t)$ is a random function with normal distribution with mean of 0 and variance of 1.

2) Langevin equation: The random walk of Brownian particles can be modeled using stochastic differential equations. The Langevin equation is such an equation which can be used to model Brownian motion. It describes how a system evolves when subjected to a combination of deterministic and fluctuating (”random”) forces. For Brownian motion in one dimension the Langevin equation looks like[2]:

$$m\frac{d^2x(t)}{dt^2} = -\gamma\frac{dx(t)}{dt} + \xi(t) \tag{6}$$

Where ξ is a function representing Gaussian white noise responsible for the random motion of the particle. The magnitude of this white noise is given by $\sqrt{2k_bT\gamma}W(t)$ where $W(t)$ is a random function with normal distribution with mean of 0 and variance of 1.

B. Dielectrophoresis

The dielectrophoretic force is the force on a dielectric particle polarised by a non-homogeneous electric field. Standard DEP predicts[6] that if the particle's electric polarizability exceeds the polarizability of the fluid medium, it moves towards a higher field strength. Moreover, if the particle's electric polarizability is lower than the polarizability of the fluid medium, it moves towards a lower field strength. This is positive and negative DEP, respectively. When the distance between the poles of the dipole is small compared to the scale of nonuniformity of the field, the dielectrophoretic force is approximated by[2]:

$$F_{dep} = (4\pi\epsilon_0\epsilon_m Re(K(\omega))R^3 E\cdot\nabla)E \tag{7}$$

For a DC field field $(E\cdot\nabla)E = \nabla E^2$ and for an AC field $(E\cdot\nabla)E = \frac{1}{2}\nabla E^2$. The force depends on the permittivity of the medium ϵ_m, the size of the particle R, the electric field gradient, and the Clausius-Mossotti factor, $K(\omega)$. The $K(\omega)$ describes the contrast between the medium and the particle in terms of their polarizability due to an electric field. $K(\omega)$ factor for a sphere like nanoparticle can be described as [6]:

$$K(\omega) = \frac{\epsilon_2^* - \epsilon_1^*}{\epsilon_2^* + 2\epsilon_1^*} \tag{8}$$

Where ϵ_2^* is the complex permittivity of the particle and ϵ_1^* is the complex permittivity of the medium. These complex permittivity are given by [19]:

$$\epsilon_n^* = \epsilon_n - \frac{i\sigma_n}{\omega} \tag{9}$$

where ϵ_n is the relative permittivity of the material and σ_n the conductivity of the material and ω the frequency of the electric field. The $\frac{i\sigma}{\omega}$ part can be ignored when dealing with DC electric fields. The $K(\omega)$ is limited in the range $-0.5 < K(\omega) < 1.0$. Which is negative and positive DEP, respectively.

In the presence of an AC field, this force will alternate with time. When the field changes very fast, it is possible to average this over the timescale. The theoretic time-averaged dielectrophoretic potential energy is:

$$\langle U_{dep}\rangle = -2\pi\epsilon_0\epsilon_m R^3\frac{\epsilon_p - \epsilon_m}{\epsilon_p + 2\epsilon_m}|E^2| \tag{10}$$

With the time averaged force as $\langle F_{dep}\rangle = -\nabla\langle U_{dep}\rangle$. Here, $|E^2|$ is the amplitude of the electric field at maximum amplitude in the AC wave. This will realistically mean that the real potential of the trap is less than this as, at most times, the amplitude is lower, but it would be proportional to it. To model the dielectrophoretic force in simulation, it is added to the Langevin equation. The Langevin equation for Brownian motion under an external force field reads:

$$m\frac{d^2x(t)}{dt^2} = -\gamma\frac{dx(t)}{dt} + \xi(t) + F_{external} \tag{11}$$

In this case, $F_{external}$ is the dielectrophoretic force F_{DEP}.

C. Relaxation Time

This paper aims to simulate sub-50 nm polystyrene particles. These particles have a low mass and reach terminal velocity quickly for a given force. This behaviour is characterized by the momentum relaxation time [7] [8]:

$$\tau = \frac{m}{\gamma} \tag{12}$$

Where m is the mass of the studied particle. If the time step of the simulation is well below this relaxation time, the simulation can transition from the ballistic to the diffusive regime. For 20-50 nm particles, the relaxation time ranges between $8.77 * 10^{-18}$s and $5.48 * 10^{-17}s$ depending on the particle's radius. The timestep of the simulation will be discussed in subsection IV-A. Assuming the timestep is well above the relaxation time, the acceleration term can be discarded, and the resulting Langevin equation will look like:

$$0 = -\gamma\frac{dx(t)}{dt} + F_{dep} + \xi(t) \tag{13}$$

Simplifying the equation:

$$\frac{dx(t)}{dt} = \frac{F_{dep}}{\gamma} + \frac{\xi(t)}{\gamma} \tag{14}$$

D. Statistical Approach To Trapping

The trap will only attract particles if the force towards the trap is greater than the thermal energy propelling the Brownian motion. A boundary will exist where the DEP energy is greater than the particle's thermal energy and remains trapped. For a collection of particles, this boundary can be predicted using the particle-conservation equation[15]:

$$\frac{\delta n}{\delta t} + \bar{v}\cdot\nabla n = \nabla\bar{\mathbf{J}} \tag{15}$$

Here, n is the number of particles per unit volume, and $\bar{\mathbf{J}}$ is the total flux, which in this case is Brownian motion flux and DEP flux. Given by:

$$\bar{\mathbf{J}} = -D\nabla n + \frac{nF_{Dep}}{6\pi\eta R} \tag{16}$$

The particle-conservation equation can be used to analyze the confinement of particles in potential energy minima. For the complex electrode model, the electric field gradients do not always guide particle to the confinement region. This is because the gradient is strongest towards the electrode resulting in particles getting stuck to the electrode instead of moving towards the convergence region. This thesis will explore a simplified case that mimics the complex electric field of the nanogap electrode while remaining easy to study. The most straightforward implementation of such an electrode would be a charged sphere. For a spherical electrode, the confinement region is approximated by a circle. When the particle is within the circle's radius, it is in a steady state. It will be called the capture radius. Here, the dielectrophoretic force is so strong that thermal energy will not carry it outside of the trap. In the steady state, the total flux at any point is zero. In this case, the particle-conservation equation can be written as[15].

$$-D\nabla n + \frac{nF_{DEP}}{6\pi\eta R} = 0 \tag{17}$$

Solving this differential equation for n in terms of dielectrophoretic potential, the following equation is found:

$$\frac{n}{n_0} = \exp(-\frac{U_{dep}}{k_b T}) \tag{18}$$

where $\frac{n}{n_0}$ represents the proportion of particles trapped.

E. Electric Field Gradient

The dielectrophoretic potential depends on the electric field gradient. The electrodes of the complex nano-electrodes create an electric field, which is hard to calculate analytically. COMSOL Multiphysics will be used to calculate it. For the charged sphere, a relatively simple theory can determine The electric field gradient for a sphere. The electric field is the gradient of potential:

$$E = -\nabla V \tag{19}$$

The potential caused by a spherical particle bearing charge q is:

$$V(\mathbf{r}) = \frac{q}{4\pi\epsilon_m\epsilon_0 r} \tag{20}$$

Using spherical coordinates, we find:

$$E = -[\hat{\mathbf{r}}\frac{\delta}{\delta r} + \hat{\theta}\frac{1}{r}\frac{\delta}{\delta\theta} + \hat{\phi}\frac{1}{rsin(\theta)}\frac{\delta}{\delta\phi}]\frac{q}{4\pi\epsilon_m\epsilon_0 r} \tag{21}$$

Because V does not change for θ and ϕ the electric field is:

$$E = -\hat{\mathbf{r}}\frac{\delta}{\delta r}\frac{q}{4\pi\epsilon_m\epsilon_0 r} = \hat{\mathbf{r}}\frac{q}{4\pi\epsilon_m\epsilon_0 r^2} \tag{22}$$

The electric field gradient is thus defined as:

$$\nabla E^2 = (\frac{q}{4\pi\epsilon_m\epsilon_0}^2)\frac{-4}{r^5} \tag{23}$$

III. Simulation Setup

The simulation will solve the Langevin Equation for the polystyrene nanoparticles. It will do this for two electrode geometries. For the spherical model, it is done to assess the trapping potential of the simulation by testing for a capture radius. For the complex electrodes, it is done to prove the framework can model particles in 3D using COMSOL imported electric fields. The framework is coded in Python using code based on previous work done by E. Schwander et al.[2] and Sergii Pud. The process of COMSOL simulation is also explained by E. Schwander et al.[2]. The simulated particles have a radius of between 20 and 50 nm, and the dielectrophoretic force acting on them is approximated by considering the particles as point-like objects. Considering the steep electric field gradient, this model is not correct, but this will remain out of the scope of this thesis as it is a proof of concept for a 3D Langevin equation solver. Furthermore, this thesis will not delve into modeling the real Clausius-Mosotti factor of particles, so a Clausius-Mosotti factor of 1 is used. This is the value often approximated for polystyrene spherical particles[12]. The rest of this section will explain how the Langevin Equation is solved in simulation as well as how the electric field is modeled; it will also explain the need for and implementation of collision detection. Additionally, it will discuss the optimization steps, and lastly, it will explain the setup for the experiments.

A. Numerically Solve Langevin Equation

To simulate the forces on the particle, the Langevin equation has to be solved numerically. As discussed in subsection II-C, the simulation is in the diffusive regime when the timestep is well above the relaxation time of the particle. Equation 13 can be used to solve for the particle's velocity. The Euler method is used:

$$x[n+1] = x[n] + \Delta t \cdot v \tag{24}$$

For the F_{dep} part this yields:

$$x[n+1] = x[n] + \Delta t \cdot \frac{F_{dep}}{\gamma} \tag{25}$$

The white noise term can not be approximated using the Euler method as it would conflict with the variance criteria of the white noise [8]. The following equation approximates the white noise numerically:

$$x[n+1] = x[n] + \sqrt{\Delta t} \cdot W \tag{26}$$

Combining these two equations:

$$x[n+1] = x[n] + \Delta t \cdot \frac{F_{dep}}{\gamma} + \sqrt{\Delta t} \cdot \frac{\xi[t]}{\gamma} \tag{27}$$

The white noise is generated using the numpy random.randn code, which takes a sample from a standard normal distribution [9].

B. Electric Field

The electric field gradient for the complex geometry shown in Figure 2C is calculated using COMSOL. COMSOL is simulation software that uses the Finite Element Method to solve differential equations. It calculates the Electric Field of arbitrary geometries by solving Maxwell differential equations for set boundary conditions. The setup for COMSOL is outside the scope of the thesis and is explained in [2]. The potential of the electrodes is 1 V when running the COMSOL simulation. Because the squared electric field gradient scales with V^2[2], later scaling can be done in Python. This reduces the amount of simulation that has to be done in COMSOL and the amount of data that needs processing, increasing the time spent on the simulation code. When exporting from COMSOL, it generates

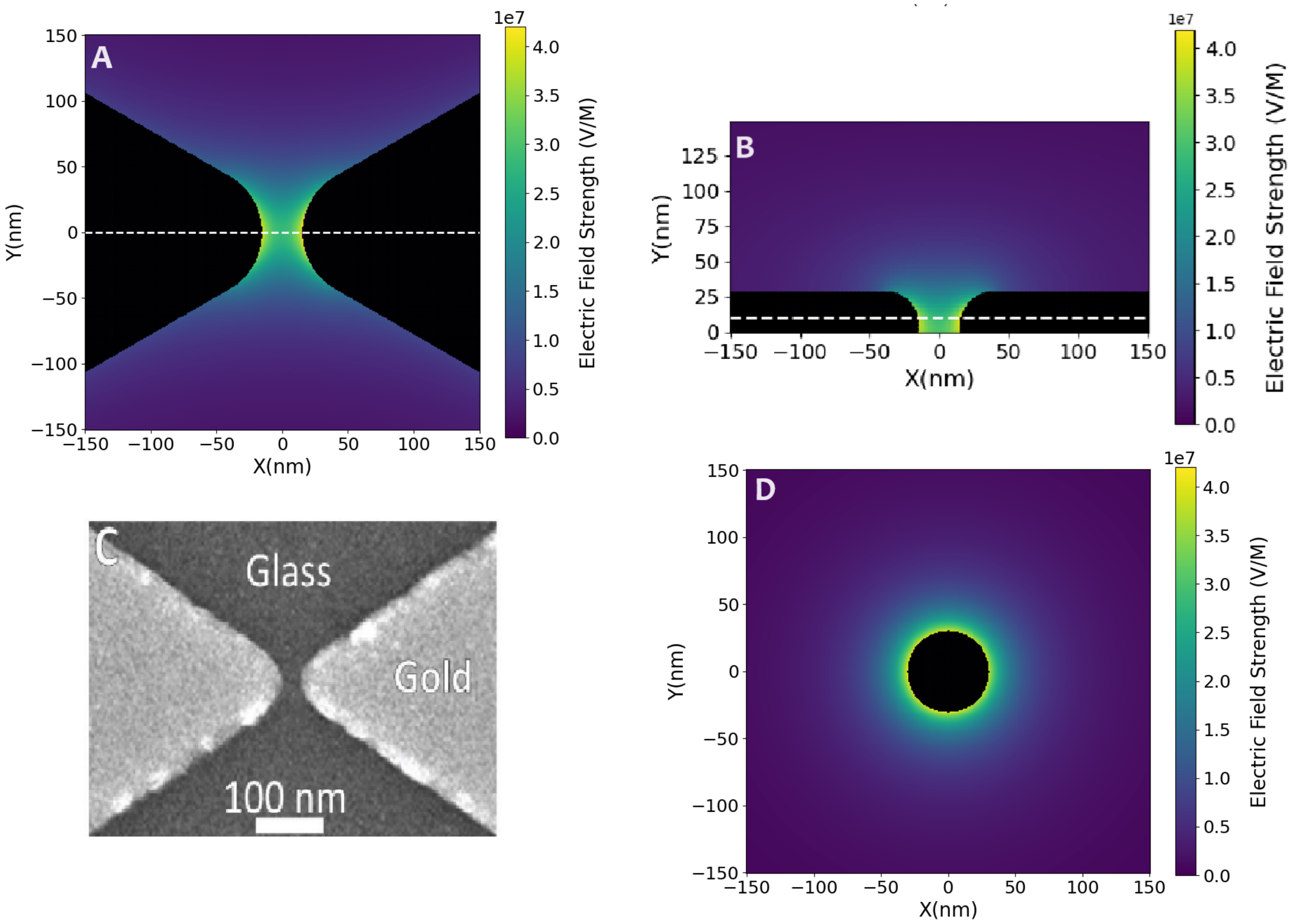

Fig. 2: A) Electric Field of electrodes proposed in Pud Sergii et al. [10] at 10 nm above the bottom. Black indicates the electrodes and the white line indicates the cross-section shown in B. B) Electric field of same electrodes as Y = 0. The white line indicates the cross-section shown in A. C) Electrode figure found in Pud Sergii et al. [10]. D) Electric Field of the simplified electrode.

a .txt file encoding the electric field gradients at a range of coordinates. The resolution of this grid is 1 nm for a 601*601*301 grid. The electric field induced by the electrodes is shown in Figure 2A and Figure 2B. The electric field gradient of the simplified sphere electrodes can be solved using Equation 23. This electric field should be similar to the COMSOL electrode to mimic its behavior. A sphere with a 30 nm radius was chosen to mimic the tips of the COMSOL electrodes. The electric field at the edge of the electrode should be $4 * 10^7 V/m$ as can be seen in figure Figure 2A. The electrode's charge is found by solving Equation 22 at the edge of the electrode for q. This yielded a q of $3.2 * 10^-16$. Using this charge, the electric field is calculated and can be found in figure Figure 2D. The potential is solved with Equation 20 and is 1.2V. The force on the particles is correlated with V^2. Thus, multiple potentials of the electrode will be tested. The ratios considered are 1, 0.5, and 0.25. Respectively being 1V, 0.6V and 0.3V. Furthermore, the electric field gradient is calculated for the same grid as the COMSOL simulation output to keep the method of simulation the same between electrode geometries.

C. *Collision Detection*

Collusion detection is implemented for two reasons. First, the particles should not be able to go within the electrodes or below the floor. Secondly, as the particles are considered point-like, the dielectrophoretic force can become unrealistically strong if the particle gets too close. This is because the electric field gradients are big at the edges of the electrodes. The problem manifested in particles moving through electrodes and exiting on the other side during one timestep. The implementation thus considered the particle's radius, limiting the proximity of particles to the electrodes. For the spherical electrode collision detection, math was simple. Each coordinate in the 601*601*301 grid is looped over. If the distance from the coordinate to the center of the electrode is less than the summed radii of the electrode and particle, the coordinate is a collision coordinate. The collision detection on the COMSOL geometry proved to be more involved. As the arbitrary geometry means no easy mathematical formula would unravel the particle's distance from the electrode. This is solved by looping over all the electrode coordinates and projecting the particle on top of it. All the coordinates on which the particle is projected are considered collision coordinates. Collusion prevention was done in the same way for both electrode models. The future coordinate is checked for collisions before the particle moves. If it is a collision coordinate, the particle's movement is canceled. These non-elastic collisions did result in sticking behaviors, meaning a particle could not move away from the electrode anymore. However, implementing realistic elastic collision would require the normal of the surface at collision[13]. This remained outside the scope of this thesis.

D. Optimization

The simulation involves a lot of calculations at each timestep. To optimize this, all calculations regarding the particle displacement due to the DEP are done before running the simulation. This is done by looping over all grid points and calculating the DEP part of the numerical Langevin equation. The precalculation means that all coordinates in the 601*601*301 grid have predetermined displacement and collision values. During the simulation only two array calls and a numpy.random call for Brownian motion is enough for the movement modeling. This significantly improved the speed of the simulation

To optimize the simulation further, the optimal time step is needed. Too short means that the simulation will take longer than needed, and too large leads to non-realistic behavior in the simulation. The problem with a large timestep is that the particles will move too much in one timestep such that they miss the slope of the electric field. An example of such a problem is that at a large time step, the displacement can get so big that in one timestep, the particle moves from close to the electrodes, where the field gradients are big, to outside the coordinate system. In the physical world, the particle would have collided with the electrode or slowed down due to drag. The relaxation time must also be considered as the Equation 27 assumes that the simulation deals with the diffusive regime. The simulation is run multiple times with multiple timesteps to find the optimal timestep. The optimal stable timestep is seen to scale with the intensity of the electric field[2]. The optimal timestep is found for different electric field ratios. It is also tested for different particle sizes. The simulations are run on the spherical electrode model, as this geometry enables more straightforward distance calculations.

E. Capture Coordinates

The capture coordinates are found by determining the likelihood of trapping the particle at different starting positions. The duration of the experiment is $5 * 10^{-}6$s. This was a balance between runtime and divergence of Brownian motion. The longer the experiment duration, the more the Brownian motion will stay consistent between the simulations. However, the simulation runtime linearly increases with the duration of the experiment. Using limited hardware, $5 * 10^{-}6$s was considered sufficient. For the spherical electrode, the particles are considered trapped when less than 5 nm away from the electrode. This is considered a safe distance as it is clear that the DEP force is much bigger than the thermal energy at this distance. The coordinates where the particle is 50% likely to get trapped are considered capture coordinates. A circle gets fitted over these manually, and this circle represents the capture threshold. It corresponds to a squared electric field strength as approximated by Equation 18. In the complex electrode model, the particle is not considered trapped per se when getting close to an electrode. A capture radius of 50 nm exists at the origin of the grid. This is the place with the lowest trapping potential, and only here Equation 17 is considered valid.

IV. Simulation Results

Voltage Ratio	Radius Particle (nm)	Stable Δt (s)
0.25	20	$1 * 10^{-}7$
0.5	20	$5 * 10^{-}8$
1	20	$5 * 10^{-}9$
	30	$5 * 10^{-}9$
	40	$5 * 10^{-}9$
	50	$5 * 10^{-}9$

TABLE I: Table displaying stable time steps for different simulation configurations

A. Timestep

The timestep on which the simulation runs will, for a large part, determine its speed. To test for the optimal timestep, 300 particles are released at a 15 nm distance from the spherical electrode. This means a different starting position for particle radia as the distance to the electrode is calculated from the closest point of the particle. The time it takes to reach the electrode is measured, and the amount of particles that hit it is also recorded. The graphs displaying the results are found in Figure 5. The stable Δt for the electric field ratios and particle sizes are found in Table I. The other experiment is run with these timesteps.

B. Capture Coordinates

The capture coordinates are starting coordinates at which the particle is 50% likely to be captured by the trap. One hundred particles are released at every 10th index of the 601*601*301 coordinate system to find these coordinates. For the spherical electrode electrode, the particle is considered captured when the distance to the electrode is less than 10 nm. A circle can be fit on these capture coordinates. These circles have a radius and that is called the capture radius. These capture radia represent a squared electric field amplitude. These results are found in Figure 4. The capture radia are found for different particle sizes and voltage ratios. The capture coordinates of the complex electrode are plotted in Figure 3B. It was not possible

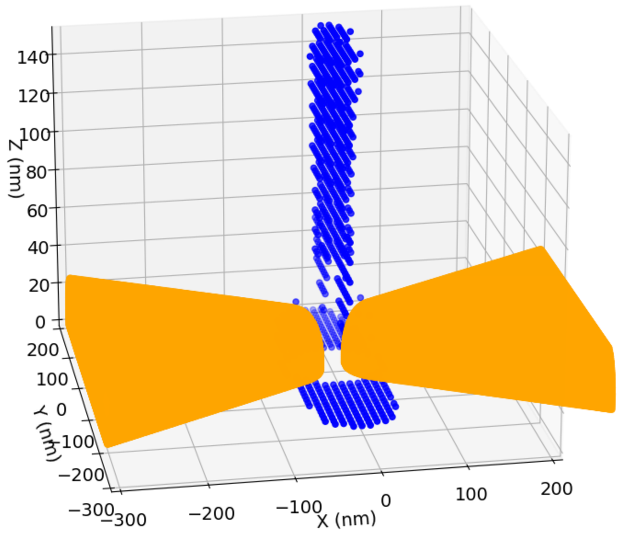

Fig. 3: Capture Coordinates for complex electrode

to determine an capture radius and electric field magnitude for this configuration.

V. Discussion

It is possible to numerically solve the Langevin equation as described in Equation 27. The simulation modeled the dielectrophoretic force and the Brownian motion in 3D for a polystyrene particle with a 20-50 nm radius. This is shown for two electrode models. Their electric field gradients are approximated using a 601*601*301 grid with a 1 nm resolution. Coding of the collision detection was done to improve accuracy, and the timestep and capture coordinate experiments were set up to improve speed and quantify the model's accuracy, respectively. Note that the simulation is run in a diffusive time regime. The timesteps used in the simulation for different electric fields can be found in Table I. These are well above the relaxation time of the particles as calculated in subsection II-C. The timestep is chosen to be the highest possible while remaining stable. The instability caused by bigger timesteps is related to the steep electric field gradients. The forces close to the electrodes become very big due to a high electric field gradient. With a big timestep, the particle might travel through the electrode and come out the other side. It does not raise an error with the collision detection but is unrealistic. The experiment is set up so that the particles are released close to the spherical electrodes. Meaning that a stable timestep is sure to yield realistic results. The timesteps are very small. This meant that for an experiment time of 1 second or more, the simulation would take days to run. For the accuracy experiment, it was thus chosen to run $5*10^-6s$. The accuracy of the trapping potential approximation Equation 10 is then determined using the capture coordinate experiment. It is evident that in terms of absolute trapping potential, the theory and simulation differ a lot. The squared electric field, which captures 50% of the particles according to Equation 18 and Equation 10 is $7.96 * 10^{10}$. Simulation approximated $1.38 * 10^{12}$. It is suspected that the difference between theory and simulation is due to the theory considering the particles as point-like. This simplification can not be done in very steep electric field gradients as the non-uniformity of the electric field is big compared to the distance between the two poles of the induced dipole. This suspicion is further supported by the correlation of the trapping potential to the radius of the particle. The theory states that the trapping potential should be correlated to R^3. In Figure 4, one can see that the simulated trapping potential is approximately correlated to R. This divergence from theory is thus related to the radius, which is the distance between the poles of the dipole. In the future, the particles should be approximated as 3D particles. The simulation seems to agree with the theoretical trapping potentials correlation with V^2. As shown in Figure 4, the capture radius did reduce when lowering the electrode potential remaining at approximately the same threshold squared electric field for the ratio 0.5 and 0.25. The trapping potential of the complex electrode trap could not be approximated in simulation using the same method. Due to the complex electric field and non-elastic collision system, too many starting coordinates lead to particles getting stuck. So, it is reasonable to say that the trapping potential is inaccurate. A major note to this experiment is its short runtime. The relative scale of the Brownian Motion increases as $\Delta t \to 0$[8]. The statistical properties do not change, but in a short time frame, Brownian Motion can diverge, causing a divergence in capture coordinates results. Longer experiment times would thus yield more reliable results. In the future, better optimization or better hardware could be used for these experiments. If this is done, a second experiment, as proposed in F. Ruggeri et al.[11], can determine the potential well of the trap. This potential well depends on the dwell time of the particles in the trapping radius. This experiment does require an experiment time of 1 second or more.

VI. Conclusions

Developing a dielectrophoretic trap to study proteins remains a big task with many hurdles to cross. This thesis demonstrates a 3D simulation framework to determine the trapping potential of such a dielectrophoretic trap. Using the framework, this thesis demonstrated positive dielectrophoresis accuation for particles with a 20 to 50 nm radius and a Clausius-Mossotti factor of 1 and did so for two electrode models. The first model represents the setup of S. Pud[10] in silico. Its induced electric field is calculated using COMSOL Multiphysics and then imported into Python. The second model is a charged sphere. This electric field is calculated in Python. The simulation framework was optimized using precalculations for DEP displacement and collision detection, and the optimal time step was determined for different setups. The accuracy of the trapping potential is determined for the spherical electrode. For the potential on the electrode, the trapping potential correlates as expected by the theory. For

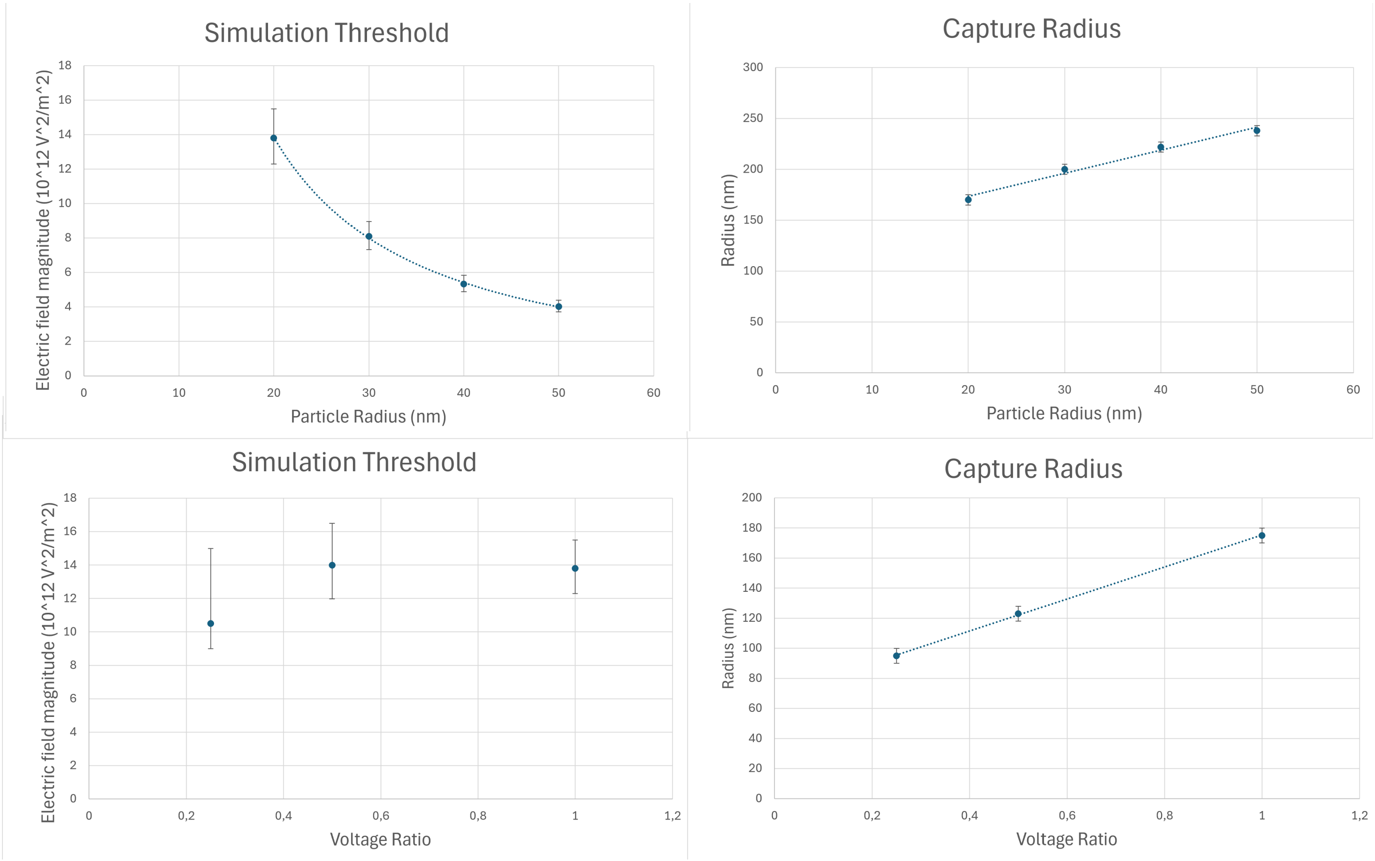

Fig. 4: Results showing the capture radius of both the voltage ratio and particle size capture coordinates experiment. Also it displays the squared electric field magnitude these capture radia represent

the radius of the particle, it does not. This is most likely because the dielectrophotic force is being calculated for point-like particles.

VII. Future outlook

The simulation framework succeeds as a proof of principle for 3D DEP potential modeling. There remain multiple areas where the framework needs improvement. One of such improvements would be longer runtimes. This would allow for less divergence in the results as the mean of Brownian motion over a long time frame is 0. This could be done by running simulations on lower electrode potential, optimizing the code better, or improving the hardware. Secondly, the simulation should move from simulating point-like particles to 3D particles. This can be done by considering the particle as a grid of point-like particles with the particle's radius and averaging out the force on these particles. Furthermore, the collision physics should be improved. This can be done by approximating the normal of all surfaces and implementing elastic collisions. Implementing this will reduce particles getting stuck on electrodes. Also, it might be relevant to consider hindered diffusion. The approximation of the diffusion coefficient as in Equation 3 considers a homogeneous medium for the particle to roam around. This is not realistic in the nanoelectrode setup[14]. To yield the same trapping potential in simulation as in real life, hindered diffusion is needed. Lastly, a new experiment could be added, which proposes another way to find the trapping potential. It is described by F. Ruggeri et al.[11] and would require runtimes of up to 1 second or more.

References

[1] A. Einstein and A. D. Cowper (1956) *Investigations in the theory of Brownian movement, translated by A.D.Cowper*

[2] S. Pud and E. Schwander (2020) *Simulation of a dielectrophoretic trap for single polystyrene beads*

[3] A. Yurt, G. G. Daaboul, J. H. Connor, B. B. Goldberg, and M. Selim Ünlü (2012) *Single nanoparticle detectors for biological applications* Nanoscale 4, 715, https://pubs.rsc.org/en/content/articlehtml/2012/nr/c2nr11562j?casa_token=6UFAbAhzChAAAAAA:RC_c87CaQFkkutEgGJqjrlH7FSr1titgN4IQBDWQnng_4U8K0y8qbs41z3SJGoD04mUFzO6jbZT0eMg (Checked on 10-06-2024)

[4] A. P. Fields and A. E. Cohen (2011) *Electrokinetic trapping at the one nanometer limit* Proceedings of the National Academy of Sciences of the United States of America 108, 8937

[5] M. I. Bespalova, S. Mahanta, and M. Krishnan (2019) *Single-molecule trapping and measurement in solution*, Current Opinion in Chemical Biology 51, 113.

[6] R. Pethig (2022) *Protein Dielectrophoresis: A Tale of Two Clausius–Mossottis or Something Else?*, MDPI, Micromachines 2, 13

[7] T. Li, S. Kheifets, D. Medellin and M. G. Raizen (2010) *Measurement of the Instantaneous Velocity of a Brownian Particle*, American Association for the Advancement of Science, VOL 328

[8] G. Volpe; G. Volpe (2013) *Simulation of a Brownian particle in an optical trap*, American Journal of Physics, 81, 224–230.

[9] *numpy.random.randn*, API reference, https://numpy.org/doc/stable/reference/random/generated/numpy.random.randn.html (Checked on 15-06-2024).

[10] J. Ton, T. Travers, J. Soltani, D. Wijnperle, D. Shavlovskyi, M. Orrit and S. Pud (2024) *Nano-gap electrode dielectrophoresis for tether-free trapping and interferometric-scattering detection of single 20 nm particles*, Achieve, (Checked on 15-06-2024).

[11] F. Ruggeri, F. Zosel, N. Mutter, M. Różycka, M. Wojtas, Andrzej Ożyhar, B. Schuler, and M. Krishnan (2017) *Single-molecule electrometry*, Nature Nanotechnology volume 12, pages 488–495.

[12] R. H¨olzel, N. Calander, Z. Chiragwandi, M. Willander, and F. F. Bier (2005) *Trapping Single Molecules by Dielectrophoresis* PHYSICAL REVIEW LETTERS, PRL 95, 128102.

[13] M. A. Zaman, MoWu, P. Padhy, M. A. Jensen, L. Hesselink and R. W. Davis (2021) *Modeling brownian microparticle trajectories in lab-on-a-chip devices with time varying dielectrophoretic or optical forces* Micromachines, 12(10)

[14] S. Eloul and R. G. Compton (2016) *General Model of Hindered Diffusion* Journal of Physical Chemistry Letters, 4317-4321, 7(21)

[15] A Ramos, H Morgan, N G Green and A Castellanos (1998) *Ac electrokinetics: a review of forces in microelectrode structures* Journal of Physics D: Applied Physics, Volume 31, Number 18

[16] J. Zlatanova, K. van Holde (2006) *Single-Molecule Biology: What Is It and How Does It Work?*Molecular Cell Volume 24, Issue 3, Pages 317-329

[17] G. Ma, H. Zhu, Z. Wan, Y. Yang, S. Wang and N. Tao (2020) *Optical Imaging of SingleProtein Size, Charge, Mobility, and Binding* Nat Commun, 11 (1). https://doi.org/10.1038/s41467-020-18547-w.

[18] Castellanos A, Ramos A, González A, Green N, Morgan H (2003) *Electrohydrodynamics and dielectrophoresis in microsystems: scaling laws*J. Phys. D: Appl. Phys. 36 2584

[19] M. Washizu, S. Suzuki, O. Kurosawa, T. Nishizaka and T. Shinohara (1992) *Molecular Dielectrophoresis of Biopolymers*. Molecular dielectrophoresis of bio-polymers, Conference Record of the 1992 IEEE Industry Applications Society Annual Meeting, Houston, TX, USA, pp. 1446-1452 vol.2

VIII. Appendix

A. AI statement

During the writing of this report AI was used in the form of spelling checkers such as Grammerly and Overleaf. During the research no information was gathered using AI. AI was used in debugging of code, and when coding using the matplotlib package.

B. Figures

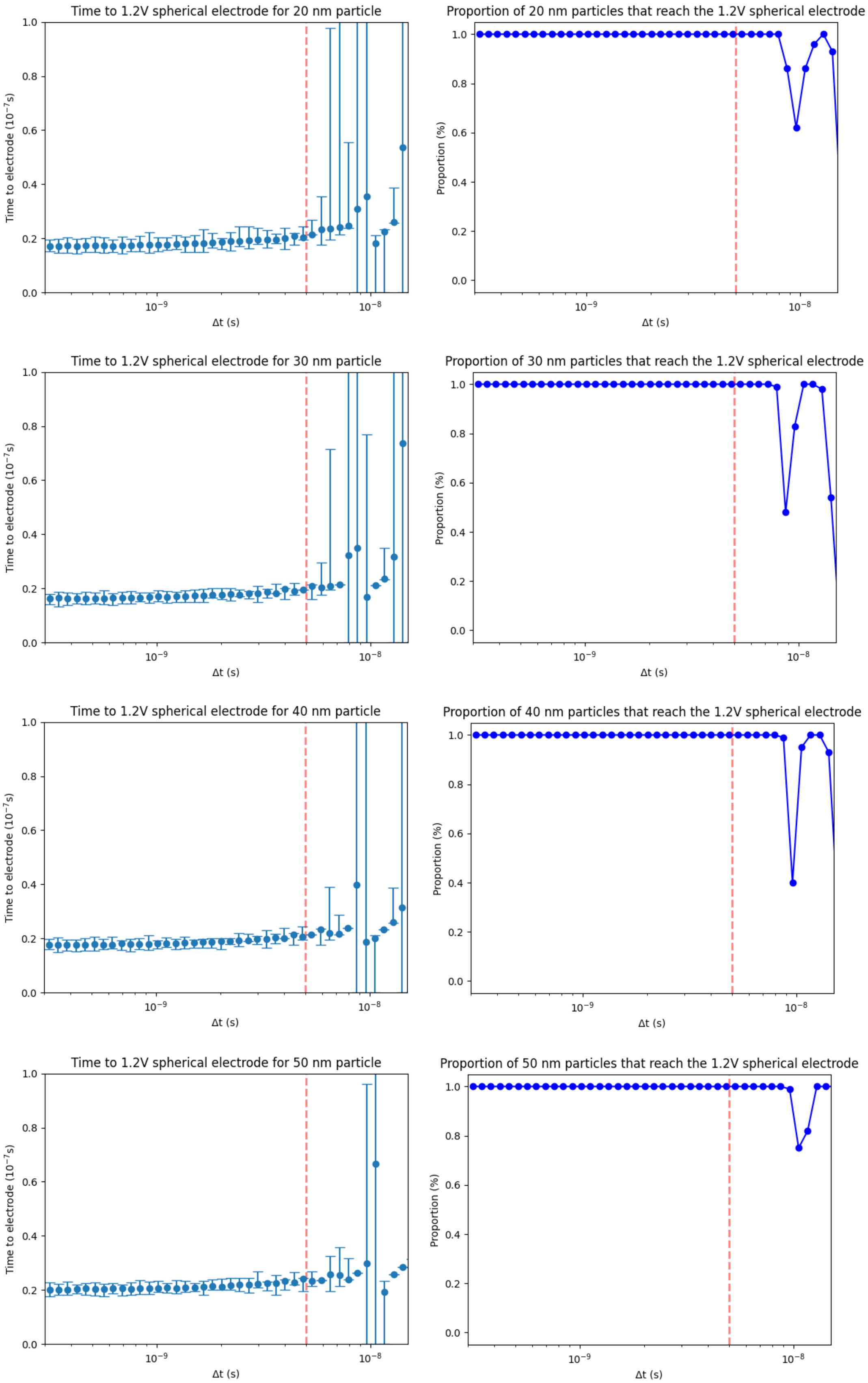

Fig. 5: The results to find the optimal timestep for each particle size. The red line represents the timestep chosen.

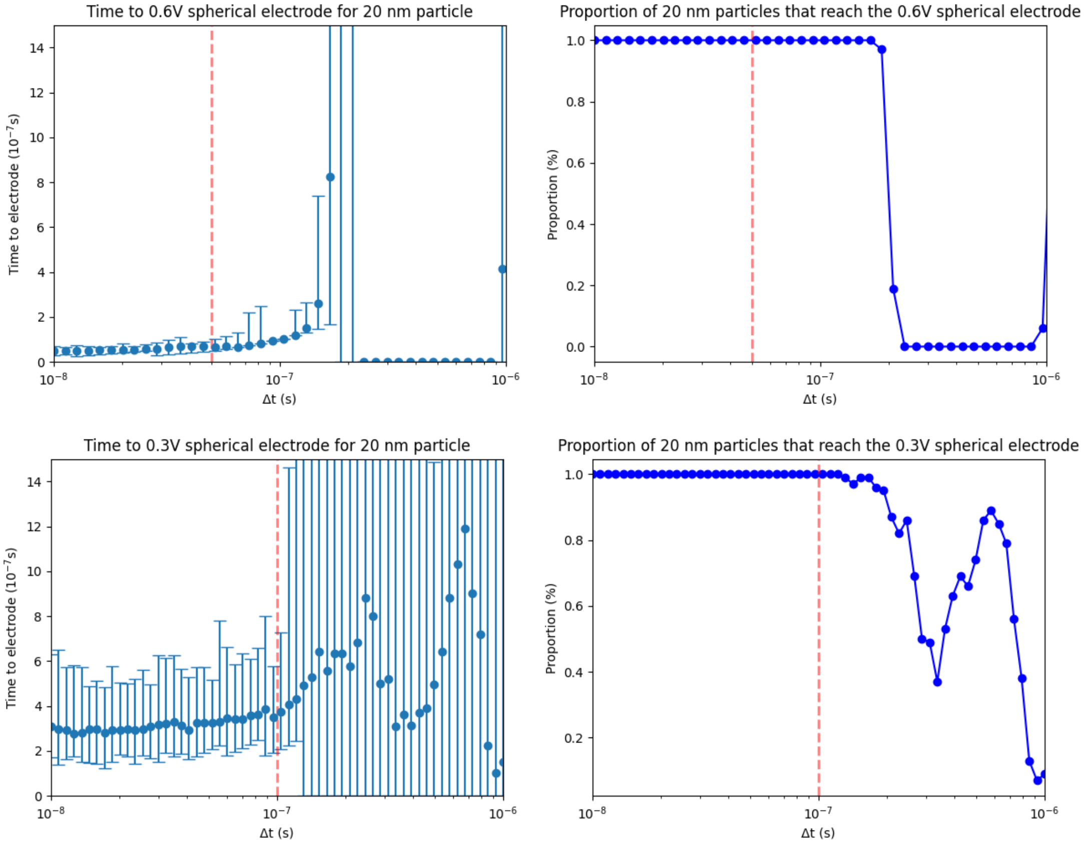

Fig. 6: The results to find the optimal timestep for electrode potential. The red line represents the timestep chosen.