

Supplementary information for
A simulation that recapitulates the dynamics of
PER-directed micron-scale colloidal assembly
(Dated: February 20, 2025)

CONTENTS

I. DNA sequences used in DNA-coated colloidal particle	2
II. Protocol for collecting DNA-coated colloidal aggregation microscopes images	2
III. Transport properties estimation	3
IV. Settling velocity and sedimentation equilibrium	3
V. Numerical step size selection	5
VI. Simulation & code setup	5
VII. Characterization of the heterogeneity of core-shell structures	7
VIII. Simulated confocal microscope images	7
References	17

I. DNA SEQUENCES USED IN DNA-COATED COLLOIDAL PARTICLE

The table below shows the domains, their function, and their sequence. All sequences are shown from the 5'-end to the 3'-end.

TABLE I. sequences

Domains	function	sequence
a	starting domain on precursor 1	ACTTCACTT
b	sticky end attached to precursors	ACCATCCTACC
c	starting domain on precursor 2	CTCCTAATTC
b'	sticky end attached to co-assemblers	GGATGGT

The table below shows the sequences ordered from Integrated DNA Technologies (IDT), their name, and their function.

TABLE II. sequences

name	function
$T_{40} \ a$	starting sequence precursor 1 /5DBCON/T40 TACTTCACTT
$T_{40} \ ab$	sequence on precursor 1 after PER /5DBCON/T40 TACTTCACTT ACCATCCTACC
$T_{40} \ c$	starting sequence on precursor 2 /5DBCON/T40 CTCCTAATTC
$T_{40} \ cb$	sequence on precursor 2 after PER /5DBCON/T40 CTCCTAATTC ACCATCCTACC
$T_{40} \ b'$	sticky end attached to co-assemblers /5DBCON/T40 GGATGGT
$T_{a \rightarrow b}$	template to append domain b onto strands ending in domain a ACCATCCTACC GGGCCTTTTGGCCC GGTAGGATGGT AAGTGAAGT /3InvdT/
$T_{c \rightarrow b}$	template to append domain b onto strands ending in domain a ACCATCCTACC GGGCCTTTTGGCCC GGTAGGATGGT GAATTAGGA /3InvdT/

II. PROTOCOL FOR COLLECTING DNA-COATED COLLOIDAL AGGREGATION MICROSCOPES IMAGES

First, we prepared an aqueous solution of colloidal particles. Next, we added the sequence-converting template and started the timer. The solution was then transferred into a capillary tube with dimensions of $10\text{mm} \cdot 1\text{mm} \cdot 100\mu\text{m}$ -sized capillary tube. The tube was placed under a bright-field microscope equipped with a 60x objective lens and a numerical aperture of 0.85. Images were captured at fixed time intervals of 1, 5, or 10 minutes. Due to particle sedimentation in the aqueous solution, the micrographs of the fluid near the bottom of the glass capillary effectively captured the state of the pseudo-2D aggregation process.

III. TRANSPORT PROPERTIES ESTIMATION

The translational diffusion constant D_T and rotational diffusion constant D_R of particles were estimated using the Stokes-Einstein equation. The radius of the particles was $500 \cdot 10^{-9}$ m. The assembly was performed at temperature $T = 30^\circ C = 303K$, where water's viscosity is $\eta = 8 \cdot 10^{-4} (kg/s \cdot m)$:

$$D_T = \frac{k_B T}{6\pi\eta r} = \frac{1.38064 \cdot 10^{-23} (\frac{m^2 kg}{s^2 K}) 303(K)}{6\pi \cdot 8 \cdot 10^{-4} (\frac{kg}{s m}) \cdot 500 \cdot 10^{-9} (m)} \quad (1)$$

$$= 0.56 (\frac{\mu m^2}{s})$$

$$D_R = \frac{k_B T}{8\pi\eta r^3} = \frac{1.38064 \cdot 10^{-23} (\frac{m^2 kg}{s^2 K}) 303(K)}{8\pi \cdot 8 \cdot 10^{-4} (\frac{kg}{sm}) \cdot (500 \cdot 10^{-9})^3 (m^3)} \quad (2)$$

$$= 1.66 (\frac{radian^2}{\mu s})$$

IV. SETTLING VELOCITY AND SEDIMENTATION EQUILIBRIUM

The terminal settling velocity of a particle, denoted by V_g , can be determined by achieving force equilibrium [1]. At this point, the drag force F_d acting on the particle is equal in magnitude and opposite in direction to the force of gravity F_g . One can estimate the terminal velocity V_g of a small particle moving through a fluid using Stokes' law as Eq. (6) shows. Once the terminal velocity is known, the particle's stochastic trajectory can be simulated using a combination of Brownian motion and a settling displacement. We used this terminal velocity for all particles and clusters in our modeling.

$$F_d = 6\pi\eta r V_g \quad (3)$$

$$F_{g'} = -\frac{4}{3}\pi r^3 \Delta\rho g \quad (4)$$

$$F_d = -F_{g'} \quad (5)$$

$$V_g = -\frac{2}{9} \frac{r^2 \Delta\rho g}{\eta}$$

$$= -\frac{2 \cdot (5 \cdot 10^{-7} m)^2 \cdot (1055 - 1000) kg \cdot m^{-3} \cdot 9.8 m \cdot s^{-1}}{9 \cdot 0.8 \cdot 10^{-3} kg \cdot m^{-1} s^{-1}} \quad (6)$$

$$= -37.4 \frac{nm}{s}$$

Sedimentation equilibrium occurs at a macroscopic level when the diffusion flux J given in Eq. (7) (where C is particle concentration) is equal to the gravitational force field flux Eq. (8). The height distribution of particles can be derived from Eq. (10) (11) (13) by integrating the net flux and setting it to zero at equilibrium Eq. (9), where ϕ is the effective gravitational potential $g'h$ [2, 3].

$$J = -D\nabla C \quad (7)$$

$$J = mFC = -mC\nabla\phi \quad (8)$$

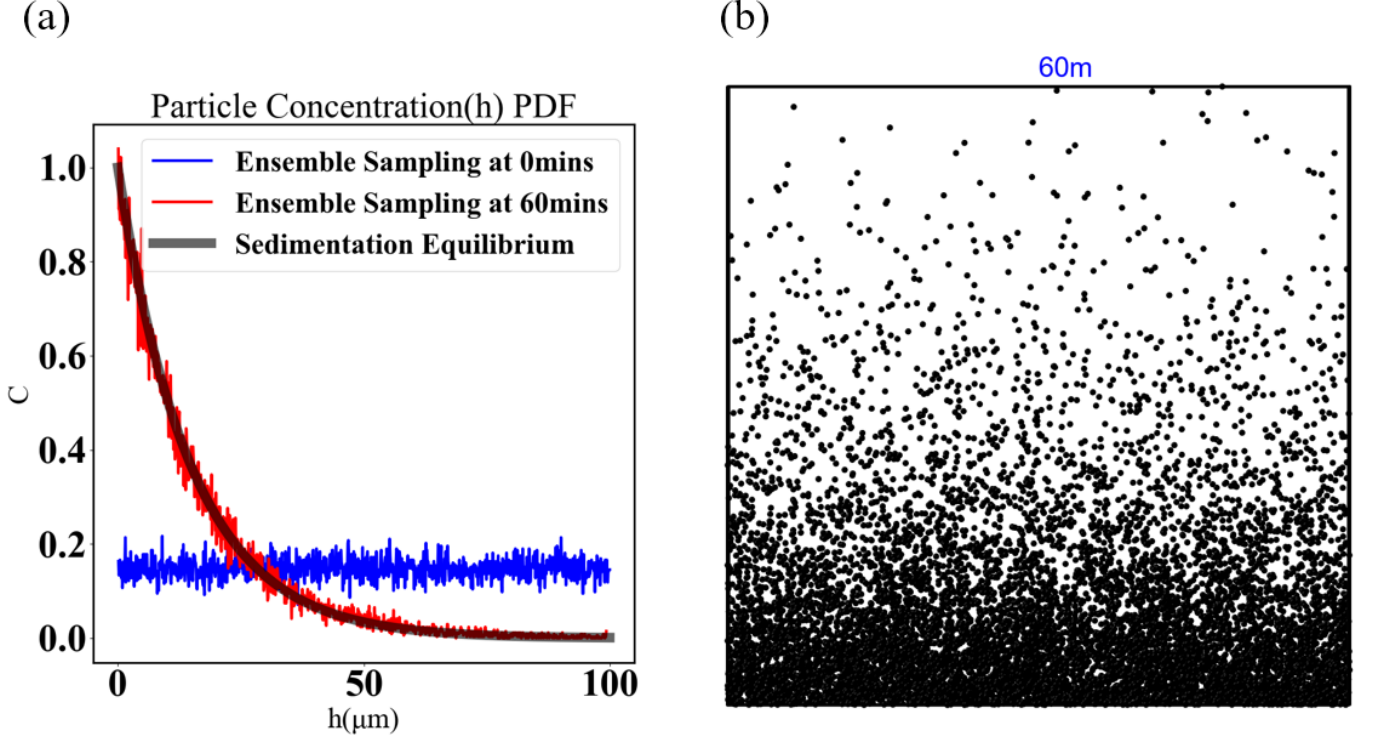


FIG. 1. (a) Particle concentration distribution at different heights h and both initially (0 minutes) and after force equilibration (60 minutes). The ensemble sampling at 60 mins aligns with the sedimentation equilibrium that follows the Boltzmann distribution. (b) A snapshot of particles' distribution after 60 minutes, where the distribution of particles has reached force equilibrium.

$$J = -D\nabla C = -mC\nabla\phi = 0 \quad (9)$$

$$C(h) = C_0 \exp\left(\frac{-m\phi(h)}{D}\right), D = mk_B T \quad (10)$$

$$\int \frac{dc}{c} = -\frac{m}{D} \int d\phi \quad (11)$$

$$C(h) = C_0 \exp\left(-h \frac{mg'}{k_B T}\right) \quad (12)$$

$$C(h) = C_0 \exp\left(-h(m) \frac{1kg((1055 - 1000) \cdot 9.8m \cdot s^{-1})}{1.380649 \cdot 10^{-23} \left(\frac{kg \cdot m^2}{s^2 \cdot K}\right) \cdot 303(K)}\right)$$

We validated the ensemble sampling by comparing it with the sedimentation equilibrium. We sampled the trajectory of 10,000 non-interacting particles for an hour. We then compared the height distribution with the theoretical Boltzmann distribution. Fig. 1 shows that our stochastic sampling agrees well with the theoretical distribution. In simulations tracking follow-up simulation, we let particles reach sedimentation equilibrium for 100 mins and initiate subsequent aggregation.

Fig. 2 shows the sedimentation equilibrium within the spherical boundary after 60 mins. In simulations tracking follow-up simulation, we let particles reach sedimentation equilibrium for 100 mins and initiate subsequent aggregation.

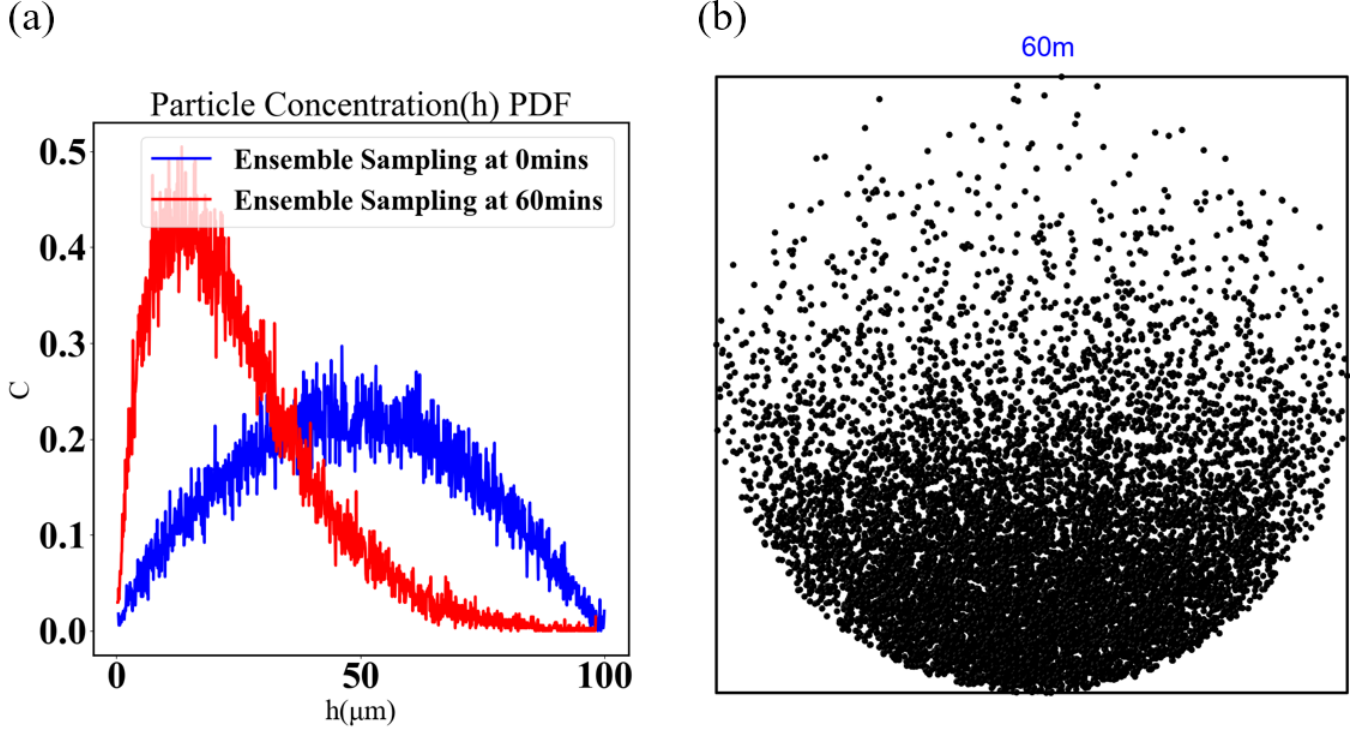


FIG. 2. (a) Particle's concentration distribution within spherical boundaries. (b) A snapshot of particles' distribution at the equilibrium.

V. NUMERICAL STEP SIZE SELECTION

The step size plays a crucial role in determining the outcome of the modeling process, as the self-assembly is integrated numerically using the probabilistic Smoluchowski model. To study a single-component aggregation system, we track the pseudo-2D aggregation of 1,000 colloidal particles over time within $100 \cdot 100 \cdot 100 \mu m^3$ box with different step sizes. In Fig. 3, we present the variation in the single fraction $f_s(t)$ (the fraction of monomer particles) over time for different step sizes, and we find that the results are not significantly different when the time step is either increased or decreased by a factor of two from $\Delta t = 20ms$, considering the relevant timescale of minutes to hours. Thus, we use $\Delta t = 20ms$ for our simulations.

VI. SIMULATION & CODE SETUP

We used and modified NERDSS [4, 5] to replicate and sample colloidal aggregation. For NERDSS code modification, we included the settling term in particles and clusters' z direction displacement (see SI. II for settling displacement quantity), and we used kinetics.t.csv to regulate time-dependent reaction rate constants (see ./main/sample_input).

The volume of a polystyrene (PS) colloidal particle is determined as Eq. (13) shows. Since we have 0.1wt% particles and 0.1wt% co-assembler particles in solution in experiments, we randomly assign 2,014 particles, and 2,014 co-assembler particles shows in a $100 \cdot 100 \cdot 100 \mu m$ simulated box boundary, as Eq. (14) shows.

$$\begin{aligned}
 V_{particle} &= \frac{4}{3}\pi \cdot r^3 \\
 &= \frac{4}{3}\pi \cdot (5 \cdot 10^{-7}m)^3 \\
 &= 5.236 \cdot 10^{-19}m^3 \\
 &= 5.236 \cdot 10^{-1}\mu m^3
 \end{aligned} \tag{13}$$

$$\begin{aligned}
& 100 \cdot 100 \cdot 100 \cdot \mu\text{m}^3(\text{solution}) \\
& \cdot \frac{1g(\text{solution})}{1\text{cm}^3(\text{solution})} \\
& \cdot 0.1\text{wt}\% \cdot \frac{1}{5.236 \cdot 10^{-1} \mu\text{m}^3} \\
& \cdot \frac{1.055\text{cm}^3(PS)}{1g(PS)} = 2014 \text{ particles}
\end{aligned} \tag{14}$$

For full simulation code, please visit:

https://github.com/s106030605/NERDSS_on_DNA_colloid/tree/main.

For single-component aggregation example, please visit :

[./main/NERDSS_simulator_on_DNA_coated_colloid/sample_input/single_component_colloid_aggregation/](#).

For core-shell multi-stage assembly example, please visit :

[./main/NERDSS_simulator_on_DNA_coated_colloid/sample_input/multi-stage-assembly/](#).

For main text 2.4 Exploring protocols for assembly, we modified and used the EXEs/nerdss.cpp code, which can read a file, `positions.csv`, to initialize the particles' configuration. We use the following steps to demonstrate the formation of a ring as a result of the melting of the core within a core-shell structure.

1. We initialized a random allocation of 3005 particles with an approximate composition of W:R:B = 7:2:7 using NERDSS compiled from `NERDSS_simulator_on_DNA_coated_colloid`.
2. We generate a `position.csv` file for core-shell structure profile using `initial_configuration_perfect_core_shell.ipynb` as an example.

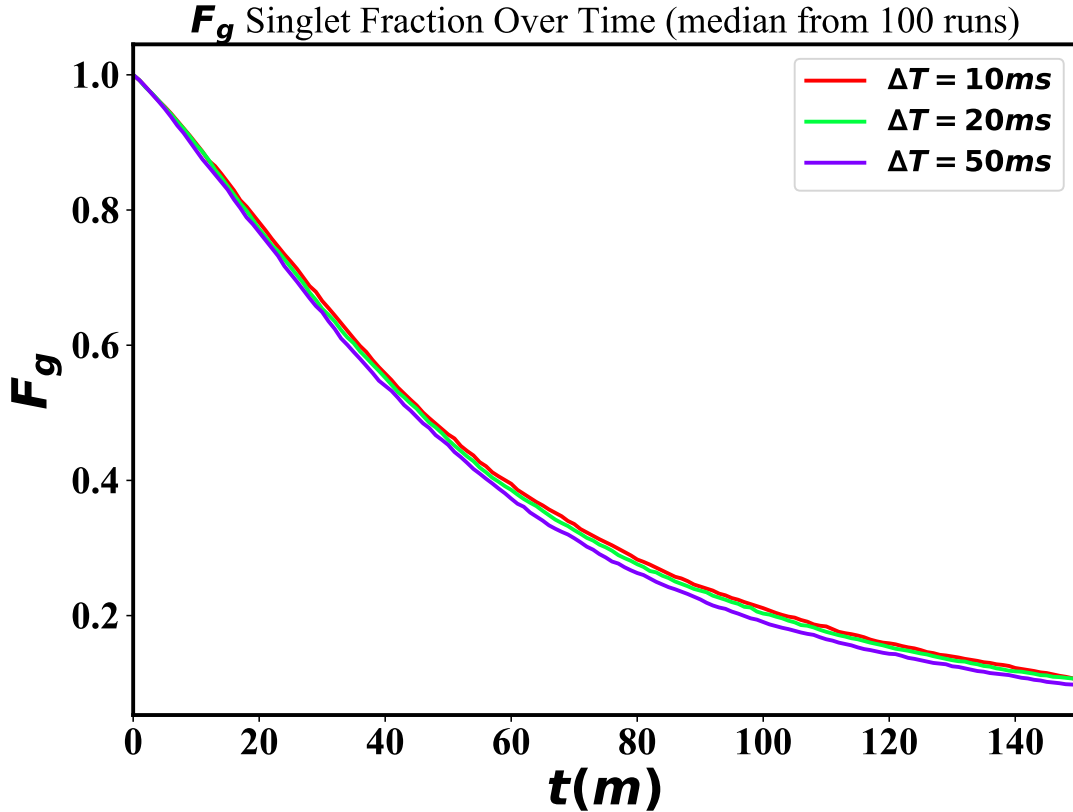


FIG. 3. Fraction of particles in the singlet state $f_s(t)$ over time of 1,000 DNA colloidal particles' aggregation with various step size.

3. We then restarted and continued the aggregation using another NERDSS compiled from `NERDSS_initial_configuration`. By doing so, NERDSS would read the `positions.csv` to reconfigure the particles' allocation. In order to create and maintain tight core-shell structure, the reaction constants for the binding of particles A & B and A & C were both specified as $1,000 \text{ nm}^2/\mu\text{s}$ in `kinetics_t_stage1.csv`. (The core-shell were bound together.)
4. We then restarted and continued the simulation. In order to dissolve the core and leave the ring behind, we kept the rate constant for the binding of A & C while we make the rest of off rate as $1,000\text{s}^{-1}$ in `kinetics_t_stage2.csv`. This enabled us to create a ring formation.

For particle positions initialization and ring formation example, please visit :
`./main/Nerdss_initial_configuration/`.

VII. CHARACTERIZATION OF THE HETEROGENEITY OF CORE-SHELL STRUCTURES

In order to measure the compositional heterogeneity of core-shell structures, we use the Jensen-Shannon Divergence (JSD) as a metric to measure the disparity between the two Radial Distribution Function $s(\text{RDF})$ of green and red pixels for assembled clusters.

The RDF is calculated as follows: first, we consider each pixel in the figure and compute the average position of all these pixels, which serves as the reference center, denoted as C . Next, we calculate the Euclidean distance between each pixel and the reference center C , resulting in a set of distances each denoted as r_i . We determine the largest such distance, referred to as r_0 . To quantify the relative distance for each pixel with respect to this maximum, we divide its individual distance by r_0 , yielding the value $\frac{r_i}{r_0}$. We then set up equally sized 100 bins that cover the range of the distances between the reference pixels to other pixels. The bin thickness was denoted as Δr . Within each shell defined by the range $[r, r + \Delta r]$, we sum the total intensities of all the pixels in this region. To obtain a normalized measure of intensity, we divide this sum of the total intensities by the shell thickness $2\pi r \Delta r$ in the bin. Finally, by converting the pixel count into frequency, we obtain a probability mass function that describes the distribution of pixels within each shell. We can thus obtain the RDF ($I(r)$) and ensure $\int_0^{r_0} I(r) = 1$.

The Jensen-Shannon Divergence (JSD) is a measure of similarity or dissimilarity between two probability distributions. The JSD is a symmetric and smoothed version of the Kullback-Leibler Divergence (KLD). The formula for the Jensen-Shannon Divergence between two discrete probability distributions P and Q is described as follows:

$$JSD(P||Q) = \frac{1}{2}(KLD(P||M) + KLD(Q||M)), \quad (15)$$

where

$$M = \frac{1}{2}(P + Q) \quad (16)$$

$$KLD(P||M) = \sum_i P(i) \cdot \log \left(\frac{P(i)}{M(i)} \right) \quad (17)$$

$$KLD(Q||M) = \sum_i Q(i) \cdot \log \left(\frac{Q(i)}{M(i)} \right) \quad (18)$$

VIII. SIMULATED CONFOCAL MICROSCOPE IMAGES

To quantitatively compare the predictions of simulations with experimental findings, we generate simulated microscope images from the simulation results and compared those with the microscopy images obtained from experiments. Table III presents the protocol for generating micrographs from simulation data (Please refer to the figure 10 in the main text 2.3 Multi-component autonomous aggregation with pre-defined rules.) The σ of Gaussian blur approximation of point spread function for confocal laser scanning microscopy is determined by the following: We used a 20x objective with a 0.4 numerical aperture. The particles were dyed and labeled with Nile Red ($\lambda : 580 - 600\text{nm}$) and pyromethene green ($\lambda : 580\text{nm}$). The FWHM (full width at half maximum) of Gaussian distribution is determined:

$$FWHM = \frac{0.51\lambda}{NA} = \frac{0.51 \cdot 580nm}{0.4} = 740nm \quad (19)$$

where $\sigma = 0.314\mu m$:

$$\sigma = \frac{FWHM}{2\sqrt{2 \cdot \ln(2)}} = 0.314\mu m \quad (20)$$

However, we decided to use σ' that is 2 times the original σ

$$\sigma' : 2 \cdot \sigma = 0.63\mu m \quad (21)$$

TABLE III. Pseudo-code for generating confocal images from simulation

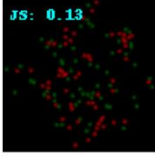
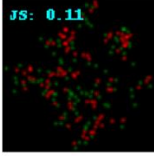
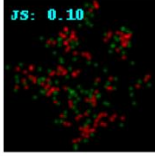
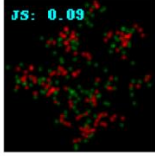
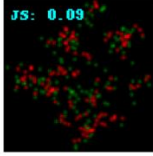
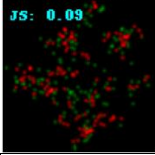
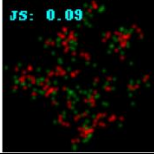
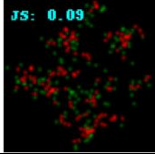
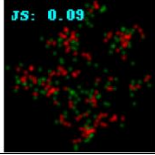
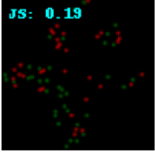
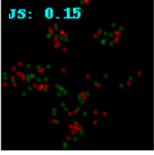
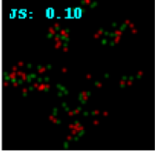
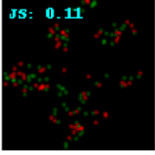
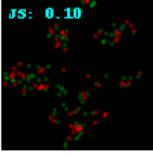
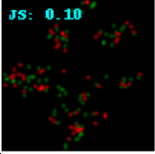
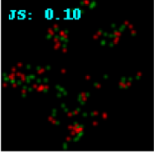
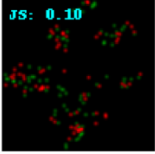
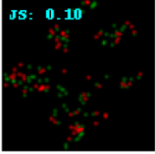
Pseudo-code for generating confocal images from simulation
Input: positions of particles (x, y, z) and their radius r
<div> <div>▷ Project particles' allocation onto various focal planes z and apply Point Spread Function Gaussian blur convolution (Eq. (21)).</div> <div>for z_i in range($4\mu m, 7\mu m$)</div> <div> <div>add circle of radius r at (x,y) if a particle's $z > z_i$ and $z < z_i + 1$</div> <div>convolve the image with Gaussian blur (Eq. (21)) to blur images</div> </div> </div>
<div> <div>▷ Create a plane with summation of the red and green intensities I_R and I_G at every (x,y)</div> <div> $Sum_R(x, y) = \sum_{z=4\mu m}^{7\mu m} I_R(x, y, z)$ $Sum_G(x, y) = \sum_{z=4\mu m}^{7\mu m} I_G(x, y, z)$ </div> </div>
<div> <div>▷ Do linear stretching (scaling of the intensity values in a linear manner) of summation of intensity to 0-255</div> <div> $maxSum_R = max(Sum_R(x, y))$ $maxSum_G = max(Sum_G(x, y))$ $J_R(x, y) = \frac{Sum_R(x, y) - 0}{maxSum_R - 0} \cdot (255 - 0)$ $J_G(x, y) = \frac{Sum_G(x, y) - 0}{maxSum_G - 0} \cdot (255 - 0)$ </div> </div>
<div> <div>▷ create an image and populate RGB values with rescaled sum of intensity J</div> <div> $image[R] = J_R$ $image[G] = J_G$ $image[B] = 0$ </div> </div>
return image

The table below shows various simulated confocal images with different starting z_{bottom} and spacing Δz :

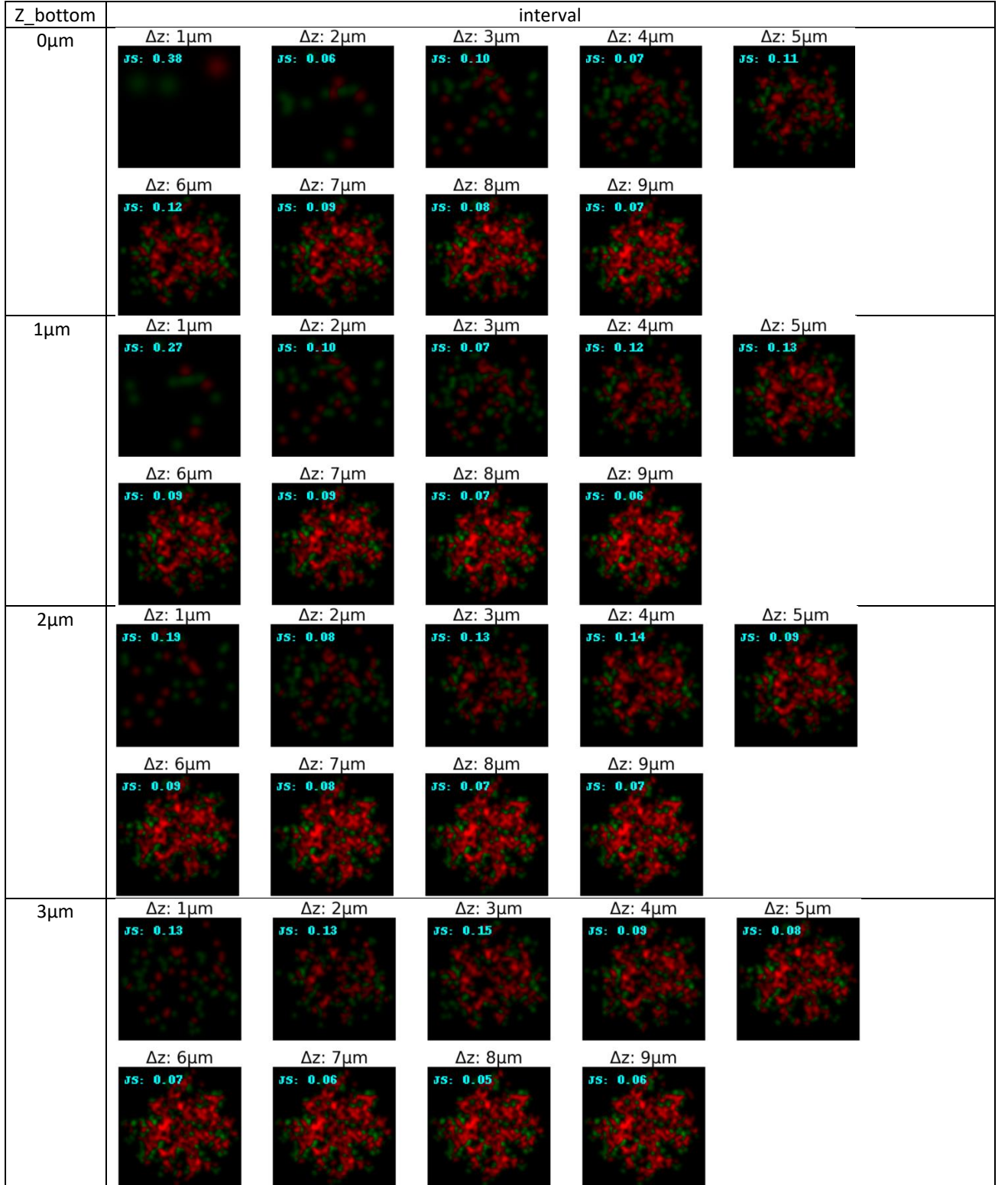
Time delay $\Delta t = 200 \text{ mins}$, binding constant $k = 1 \frac{\text{nm}^2}{\mu\text{s}}$, Gaussian blur approximation $\sigma = 0.31 \mu\text{m}$

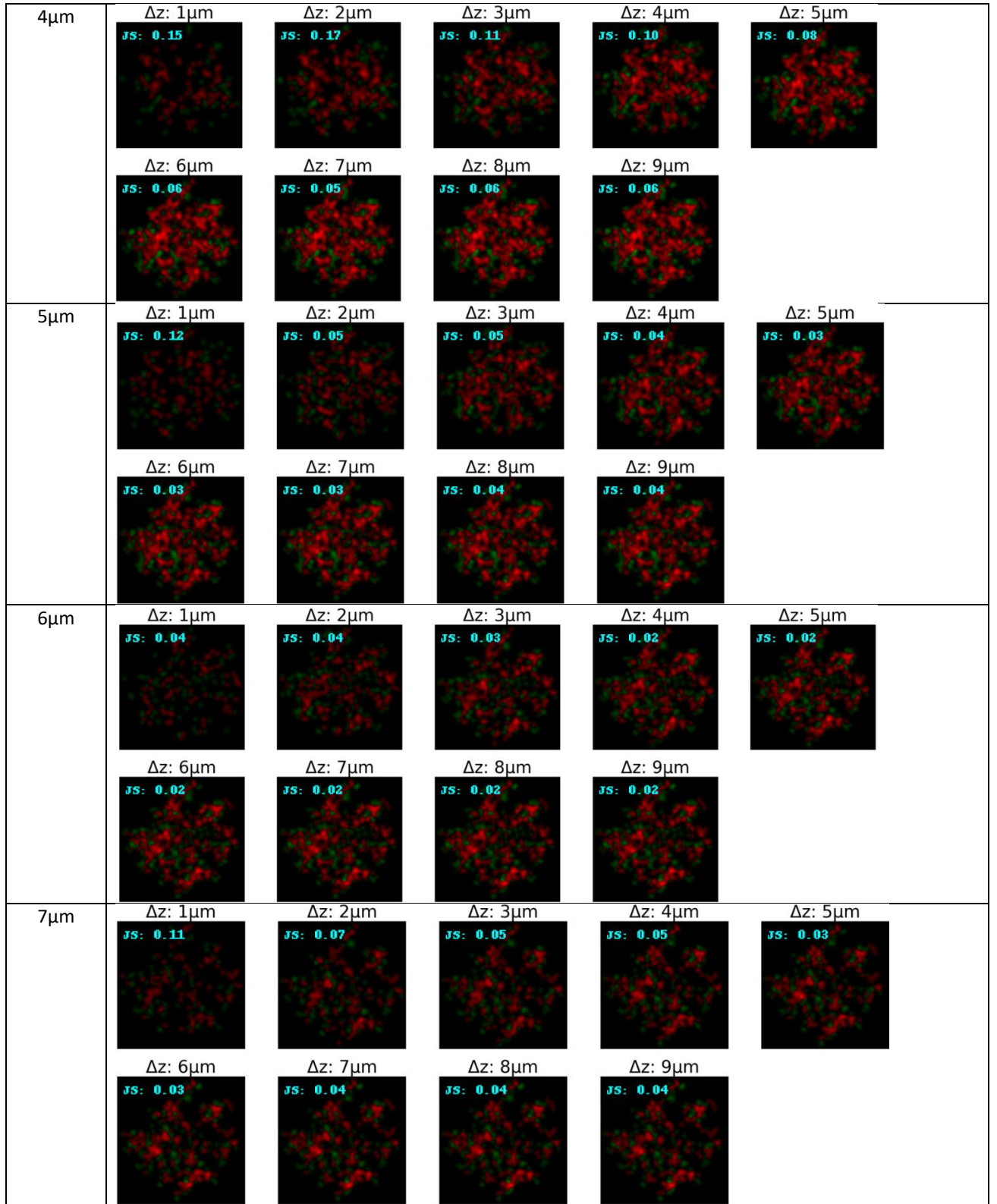
z_{bottom}	interval				
$0 \mu\text{m}$	$\Delta z: 1 \mu\text{m}$ JS: 0.45	$\Delta z: 2 \mu\text{m}$ JS: 0.10	$\Delta z: 3 \mu\text{m}$ JS: 0.16	$\Delta z: 4 \mu\text{m}$ JS: 0.06	$\Delta z: 5 \mu\text{m}$ JS: 0.08
	$\Delta z: 6 \mu\text{m}$ JS: 0.07	$\Delta z: 7 \mu\text{m}$ JS: 0.06	$\Delta z: 8 \mu\text{m}$ JS: 0.04	$\Delta z: 9 \mu\text{m}$ JS: 0.03	
$1 \mu\text{m}$	$\Delta z: 1 \mu\text{m}$ JS: 0.34	$\Delta z: 2 \mu\text{m}$ JS: 0.16	$\Delta z: 3 \mu\text{m}$ JS: 0.06	$\Delta z: 4 \mu\text{m}$ JS: 0.08	$\Delta z: 5 \mu\text{m}$ JS: 0.07
	$\Delta z: 6 \mu\text{m}$ JS: 0.07	$\Delta z: 7 \mu\text{m}$ JS: 0.04	$\Delta z: 8 \mu\text{m}$ JS: 0.03	$\Delta z: 9 \mu\text{m}$ JS: 0.03	
$2 \mu\text{m}$	$\Delta z: 1 \mu\text{m}$ JS: 0.25	$\Delta z: 2 \mu\text{m}$ JS: 0.09	$\Delta z: 3 \mu\text{m}$ JS: 0.09	$\Delta z: 4 \mu\text{m}$ JS: 0.08	$\Delta z: 5 \mu\text{m}$ JS: 0.07
	$\Delta z: 6 \mu\text{m}$ JS: 0.05	$\Delta z: 7 \mu\text{m}$ JS: 0.03	$\Delta z: 8 \mu\text{m}$ JS: 0.03	$\Delta z: 9 \mu\text{m}$ JS: 0.03	
$3 \mu\text{m}$	$\Delta z: 1 \mu\text{m}$ JS: 0.14	$\Delta z: 2 \mu\text{m}$ JS: 0.14	$\Delta z: 3 \mu\text{m}$ JS: 0.09	$\Delta z: 4 \mu\text{m}$ JS: 0.07	$\Delta z: 5 \mu\text{m}$ JS: 0.04
	$\Delta z: 6 \mu\text{m}$ JS: 0.03	$\Delta z: 7 \mu\text{m}$ JS: 0.03	$\Delta z: 8 \mu\text{m}$ JS: 0.03	$\Delta z: 9 \mu\text{m}$ JS: 0.03	

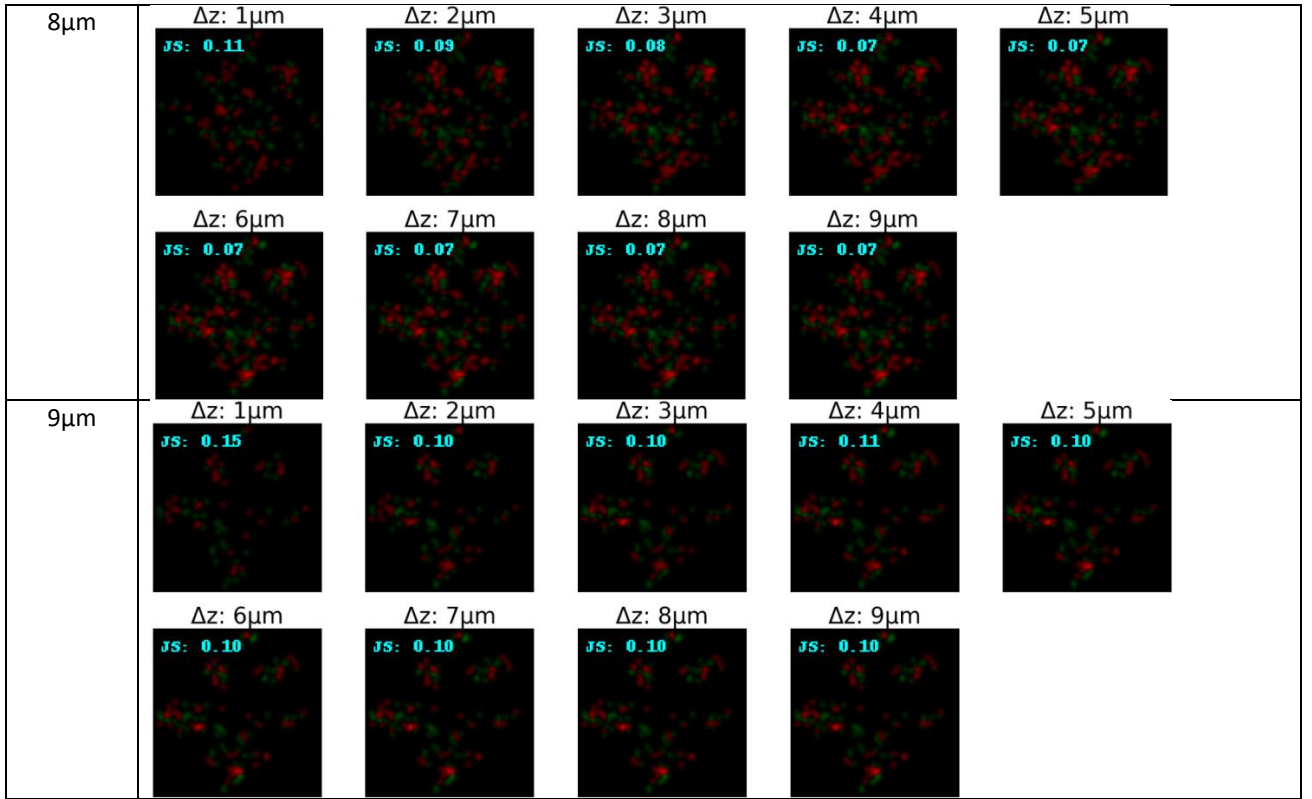
4 μm	$\Delta z: 1\mu\text{m}$ JS: 0.15	$\Delta z: 2\mu\text{m}$ JS: 0.16	$\Delta z: 3\mu\text{m}$ JS: 0.10	$\Delta z: 4\mu\text{m}$ JS: 0.07	$\Delta z: 5\mu\text{m}$ JS: 0.05
	$\Delta z: 6\mu\text{m}$ JS: 0.04	$\Delta z: 7\mu\text{m}$ JS: 0.04	$\Delta z: 8\mu\text{m}$ JS: 0.04	$\Delta z: 9\mu\text{m}$ JS: 0.04	
5 μm	$\Delta z: 1\mu\text{m}$ JS: 0.15	$\Delta z: 2\mu\text{m}$ JS: 0.05	$\Delta z: 3\mu\text{m}$ JS: 0.05	$\Delta z: 4\mu\text{m}$ JS: 0.03	$\Delta z: 5\mu\text{m}$ JS: 0.03
	$\Delta z: 6\mu\text{m}$ JS: 0.03	$\Delta z: 7\mu\text{m}$ JS: 0.03	$\Delta z: 8\mu\text{m}$ JS: 0.03	$\Delta z: 9\mu\text{m}$ JS: 0.03	
6 μm	$\Delta z: 1\mu\text{m}$ JS: 0.05	$\Delta z: 2\mu\text{m}$ JS: 0.04	$\Delta z: 3\mu\text{m}$ JS: 0.04	$\Delta z: 4\mu\text{m}$ JS: 0.04	$\Delta z: 5\mu\text{m}$ JS: 0.04
	$\Delta z: 6\mu\text{m}$ JS: 0.03	$\Delta z: 7\mu\text{m}$ JS: 0.03	$\Delta z: 8\mu\text{m}$ JS: 0.03	$\Delta z: 9\mu\text{m}$ JS: 0.03	
7 μm	$\Delta z: 1\mu\text{m}$ JS: 0.09	$\Delta z: 2\mu\text{m}$ JS: 0.09	$\Delta z: 3\mu\text{m}$ JS: 0.07	$\Delta z: 4\mu\text{m}$ JS: 0.06	$\Delta z: 5\mu\text{m}$ JS: 0.07
	$\Delta z: 6\mu\text{m}$ JS: 0.07	$\Delta z: 7\mu\text{m}$ JS: 0.07	$\Delta z: 8\mu\text{m}$ JS: 0.07	$\Delta z: 9\mu\text{m}$ JS: 0.07	

8 μm	$\Delta z: 1\mu\text{m}$	$\Delta z: 2\mu\text{m}$	$\Delta z: 3\mu\text{m}$	$\Delta z: 4\mu\text{m}$	$\Delta z: 5\mu\text{m}$
					
	$\Delta z: 6\mu\text{m}$	$\Delta z: 7\mu\text{m}$	$\Delta z: 8\mu\text{m}$	$\Delta z: 9\mu\text{m}$	
					
9 μm	$\Delta z: 1\mu\text{m}$	$\Delta z: 2\mu\text{m}$	$\Delta z: 3\mu\text{m}$	$\Delta z: 4\mu\text{m}$	$\Delta z: 5\mu\text{m}$
					
	$\Delta z: 6\mu\text{m}$	$\Delta z: 7\mu\text{m}$	$\Delta z: 8\mu\text{m}$	$\Delta z: 9\mu\text{m}$	
					

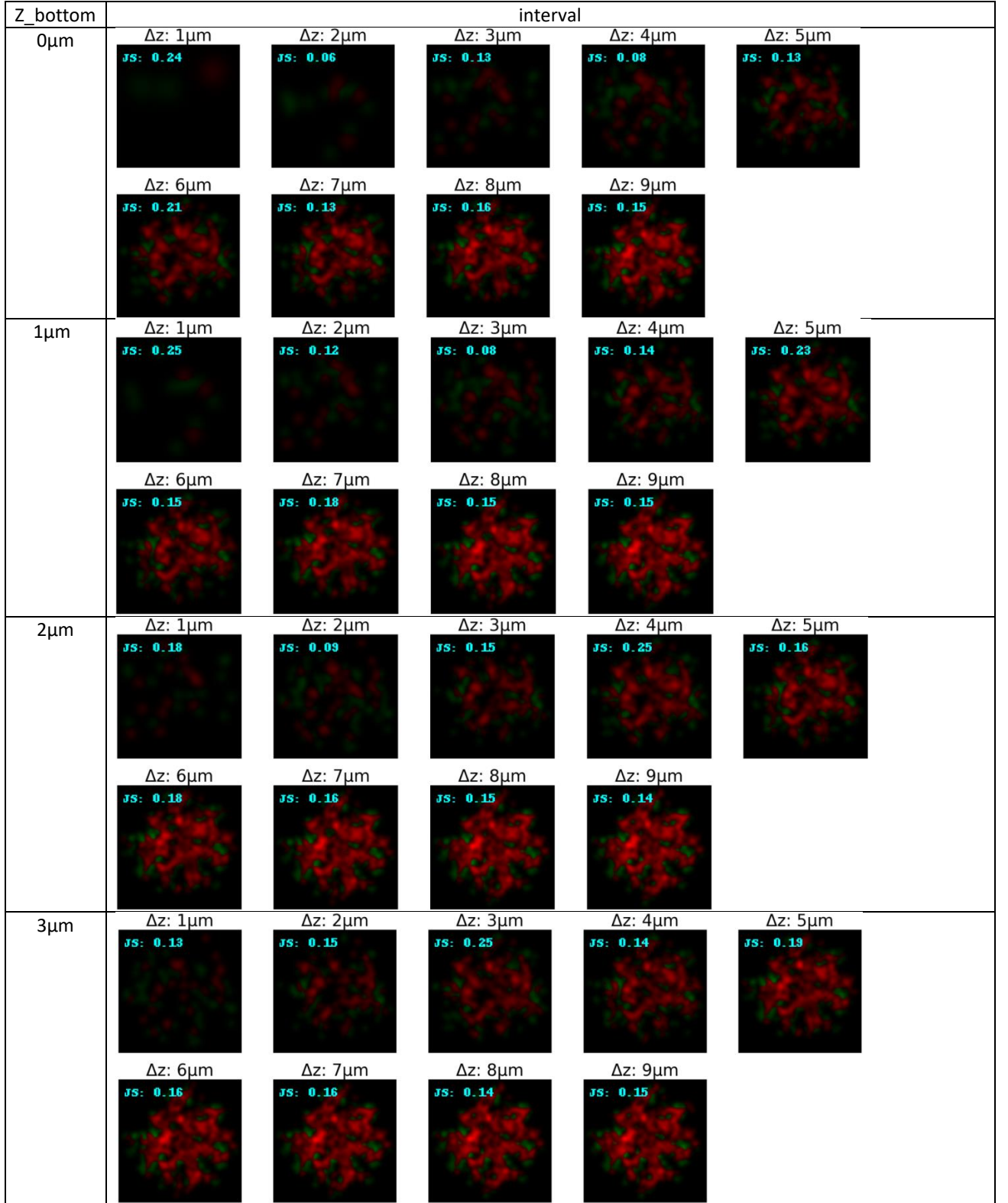
Time delay $\Delta t = 200 \text{ mins}$, binding constant $k = 1 \frac{\text{nm}^2}{\mu\text{s}}$, Gaussian blur approximation $\sigma = 0.63 \mu\text{m}$

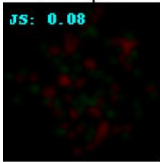
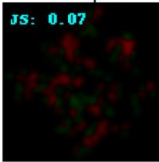
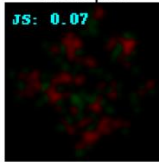
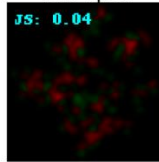
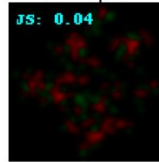
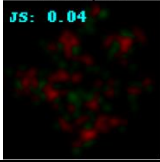
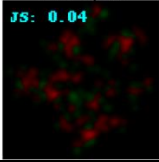
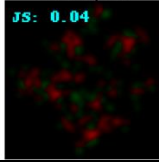
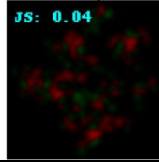
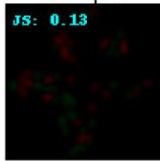
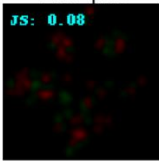
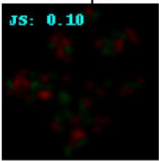
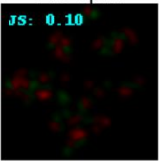
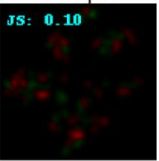
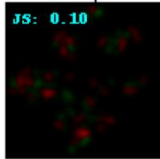
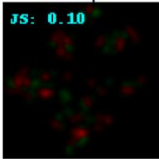
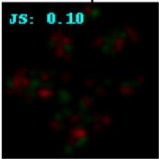
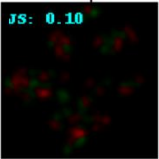






Time delay $\Delta t = 200 \text{ mins}$, binding constant $k = 1 \frac{\text{nm}^2}{\mu\text{s}}$, Gaussian blur approximation $\sigma = 0.94 \mu\text{m}$



8 μm	$\Delta z: 1\mu\text{m}$	$\Delta z: 2\mu\text{m}$	$\Delta z: 3\mu\text{m}$	$\Delta z: 4\mu\text{m}$	$\Delta z: 5\mu\text{m}$
	 JS: 0.08	 JS: 0.07	 JS: 0.07	 JS: 0.04	 JS: 0.04
	$\Delta z: 6\mu\text{m}$	$\Delta z: 7\mu\text{m}$	$\Delta z: 8\mu\text{m}$	$\Delta z: 9\mu\text{m}$	
	 JS: 0.04	 JS: 0.04	 JS: 0.04	 JS: 0.04	
9 μm	$\Delta z: 1\mu\text{m}$	$\Delta z: 2\mu\text{m}$	$\Delta z: 3\mu\text{m}$	$\Delta z: 4\mu\text{m}$	$\Delta z: 5\mu\text{m}$
	 JS: 0.13	 JS: 0.08	 JS: 0.10	 JS: 0.10	 JS: 0.10
	$\Delta z: 6\mu\text{m}$	$\Delta z: 7\mu\text{m}$	$\Delta z: 8\mu\text{m}$	$\Delta z: 9\mu\text{m}$	
	 JS: 0.10	 JS: 0.10	 JS: 0.10	 JS: 0.10	

- [2] M. Lin, H. Lindsay, D. Weitz, R. Ball, R. Klein, and P. Meakin, Universality in colloid aggregation, *Nature* **339**, 360 (1989).
- [3] R. J. Hunter, *Foundations of colloid science* (Oxford university press, 2001).
- [4] M. J. Varga, Y. Fu, S. Loggia, O. N. Yogurtcu, and M. E. Johnson, Nerdss: a nonequilibrium simulator for multibody self-assembly at the cellular scale, *Biophysical Journal* **118**, 3026 (2020).
- [5] M. E. Johnson, Modeling the self-assembly of protein complexes through a rigid-body rotational reaction–diffusion algorithm, *The Journal of Physical Chemistry B* **122**, 11771 (2018).
- [6] J. Perrin, *Brownian movement and molecular reality* (Courier Corporation, 2013).
- [7] Y. e. a. Fu, An implicit lipid model for efficient reaction-diffusion simulations of protein binding to surfaces of arbitrary topology, *The Journal of Chemical Physics* **151**, 12 (2019).
- [8] J. Perrin, Mouvement brownien et réalité moléculaire, in *Annales de Chimie et de Physique*, Vol. 18 (1909) pp. 1–114.
- [9] S. A. Rice, *Diffusion-limited reactions* (Elsevier, 1985).