Friedrich-Alexander-Universität
Naturwissenschaftliche Fakultät

Master's Thesis

in Physics

Classical Density Functional Theory for Active Particles

Jonas Buba

Supervisor: Prof. Dr. Michael Schmiedeberg

Chair for Theoretical Physics I

Submission date: 10 November 2025

Abstract

Active matter systems, composed of self-driven agents, display a wide range of emergent behaviours such as collective motion, clustering, and motility-induced phase separation. Active Brownian particles constitute a minimal model for such self-propelled systems that exhibit complex nonequilibrium behaviour. Understanding the microscopic dynamics of particle collisions is important for linking individual motion to collective effects. In this work, the collision process of active Brownian hard disks is studied within the framework of dynamical density functional theory. Each particle is represented by a Gaussian density peak, introducing positional uncertainty. In this field-based description, the particle interaction is modelled via fundamental measure theory.

The analysis of the motion of density peaks throughout the course of individual collisions allows to quantify the center of mass delay of interacting particles. The results show that the time delay scales approximately as $\Delta t \propto v_0$ with the self-propulsion velocity v_0. The increase in the delay with the Gaussian width σ is faster than $\propto \sigma^{-1}$, and the dependence on the offset h follows a roughly Gaussian shape. The self-propulsion velocity mainly influences the diffusive relaxation after a collision, while the overall positional delay remains nearly independent of v_0. Furthermore, it is demonstrated that the post-collision density profile can be approximated by a convolution of the initial profiles, suggesting a route toward a coarse-grained description of active collisions.

Future work may focus on performance improvements of the numerical implementation to enable larger-scale simulations. Furthermore, the connection to a waiting-time-based model may be possible.

Contents

1. Introduction

Robert Brown first described the type of stochastic motion exhibited by suspended pollen in the year 1827 [1]. Nearly one hundred years later, Albert Einstein provided the theoretical explanation for this phenomenon in one of his famous 1905 papers [2]. He connected the macroscopic diffusion constant D to the microscopic quantities of particles coupled to a thermal bath through the so-called Einstein relation

$$D = \frac{\mu}{\beta}, \tag{1}$$

where μ is the mobility and $\beta = (k_\mathrm{B}T)^{-1}$ involves the Boltzmann constant k_B and temperature T. This idea was formalised by Langevin shortly afterward in the now so-called Langevin equation

$$m\dot{\boldsymbol{v}} = -\gamma\boldsymbol{v} + \sqrt{2D}\xi(t), \tag{2}$$

where γ is the friction coefficient and ξ is Gaussian white noise. Furthermore, it led to the experimental confirmation of the molecular nature of matter by Perrin [3].

1.1. Active Brownian particles

Many systems in nature are not purely passive but continuously consume energy from their surroundings to generate motion. This is true for biological systems like birds, fish, and bacteria, but also for suspended colloids that show phoretic motion [4, 5]. One way to describe such phenomena is by the widely used [6–8] active Brownian particle (ABP) model. In this model, activity is represented by a persistent self-propulsion force acting on each particle. It can be formulated as follows: Consider a system of N particles at positions $\boldsymbol{r}^N = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_N\}$. Each particle is active, meaning it is self propelled in the direction $\boldsymbol{u}^N = \{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_N\}$ with amplitude f_0. The self-propulsion can also be quantified in terms of the self-propulsion velocity $v_0 = f_0/\alpha$ with friction α. The inverse of the friction α^{-1} is the mobility μ. The corresponding overdamped [1] Langevin equation then reads

$$\frac{d\boldsymbol{r}_i}{dt} = \mu\boldsymbol{F}_i(\boldsymbol{r}^N) + v_0\boldsymbol{u}_i + \sqrt{2D}\boldsymbol{\xi}_i(t). \tag{3}$$

Here, $\boldsymbol{F}_i(\boldsymbol{r}^N)$ is the force on particle i due to interactions with other particles or other sources. $\boldsymbol{\xi}_i(t)$ is white noise, i.e. it fulfils $\langle \xi_i^\alpha(t)\xi_j^\beta(t')\rangle = \delta_{ij}\delta_{\alpha\beta}\delta(t-t')$ $(\alpha, \beta \in \{x, y\})$ [2]. The orientation itself may undergo orientational diffusion according to $\dot{\theta}_i = \sqrt{2D_r}\eta_i(t)$, where $\dot{\theta}_i$ is the angle of orientation with respect to an arbitrary axis, D_r is the rotational Diffusion constant, and $\eta_i(t)$ is again white noise. The ratio of the persistence length $l_p = v_0/D_r$ to the particle diameter $2R$ defines the Péclet number

$$\mathrm{Pe} = \frac{v_o}{2RD_r}, \tag{4}$$

[1] Overdamped means that momentum relaxation occurs much faster than position changes. This allows us to drop the inertial term $m\dot{\boldsymbol{v}}$ in eq. 2.

[2] The angle brackets $\langle\ \rangle$ denote the correlation function

which quantifies the level of activity in the system [7].

1.2. Motility induced phase separation

If both the activity and particle density are sufficiently high, ABP systems can undergo a phenomenon called motility induced phase separation (MIPS). In this state, coexisting dilute and dense regions emerge even in the absence of attractive particle interactions [6, 9]. This occurs because particles are slowed down by collisions, locally reducing their motility, which increases their residence time and thereby promotes further accumulation through a positive feedback loop. If the rotational Diffusion D_r is high in comparison to the self-propulsion velocity, the particles reorient quickly, suppressing slowdown and thus hindering motility-induced cluster formation [7, 8].

A minimal kinetic model that captures this slowdown mechanism of colliding particles was proposed by Schieck [10]. The model he proposes uses penetrable particles that experience a waiting time upon collision, which depends on the local density. He demonstrated that this is sufficient for the occurrence of phase separation.

In this work, we aim to quantify the collision delay for a system of hard disks by means of dynamical density functional theory (DDFT), which is a continuum model widely employed for studying the collective dynamics of active particles [11–14]. The specifics of the employed model will be outlined in the following section.

2. Model for active hard disks

In this section, we will begin with the description of a theoretical model for active Brownian particles in the overdamped limit. We consider a two-dimensional system of hard disks as Brownian particles that cannot overlap. In the first step, a dynamical equation for the density field is derived by means of dynamical density functional theory.

2.1. Dynamical density functional theory

The following derivation is primarily based on [15]. As far as activity is concerned, inspiration was taken from [12, 16], and the final form of the model in [13] in the overdamped limit was reproduced. The starting point is the overdamped Langevin equation in 3, where the force on particle i due to external and interaction forces can be written as $\boldsymbol{F}_i(\boldsymbol{r}^N) = -\boldsymbol{\nabla}_i \left(\sum_j V(\boldsymbol{r}_i - \boldsymbol{r}_j) + V_{\text{ext}}(\boldsymbol{r}_i)\right)$ with the external potential $V_{\text{ext}}(\boldsymbol{r}_i)$ and interaction potential $V(\boldsymbol{r}_i - \boldsymbol{r}_j)$. An equation for the partial density operator $\hat{\rho}_i(\boldsymbol{r}, t) := \delta(\boldsymbol{r}_i - \boldsymbol{r})$ can be derived, following the procedure in [15]:

$$\begin{aligned}\frac{\partial \hat{\rho}_i(\boldsymbol{r},t)}{\partial t} =& D\nabla^2\hat{\rho}_i(\boldsymbol{r},t) + \mu\nabla\left[\hat{\rho}_i(\boldsymbol{r},t)\left(\int \mathrm{d}\boldsymbol{r}'\left[\hat{\rho}(\boldsymbol{r}',t)\nabla V(\boldsymbol{r}-\boldsymbol{r}')\right] + \nabla V_{\text{ext}}(\boldsymbol{r})\right)\right] \\ &- v_0\boldsymbol{u}_i\nabla\hat{\rho}_i(\boldsymbol{r},t) + \xi_i(t)\nabla\hat{\rho}_i(\boldsymbol{r},t).\end{aligned}$$

This equation depends on a continuous spatial coordinate, with the positions of the particles represented by the delta-distributions. By using the total density operator $\hat{\rho}(\boldsymbol{r}, t) = \sum_i \hat{\rho}_i(\boldsymbol{r}, t)$ and the total orientation operator $\hat{\boldsymbol{u}}(\boldsymbol{r}, t) := \sum_i \boldsymbol{u}_i\hat{\rho}_i(\boldsymbol{r}, t)$, one finds

$$\begin{aligned}\frac{\partial \hat{\rho}(\boldsymbol{r},t)}{\partial t} =& D\nabla^2\hat{\rho}(\boldsymbol{r},t) + \mu\boldsymbol{\nabla}\left[\hat{\rho}(\boldsymbol{r},t)\left(\int \mathrm{d}\boldsymbol{r}'\left[\hat{\rho}(\boldsymbol{r}',t)\boldsymbol{\nabla} V(\boldsymbol{r}-\boldsymbol{r}')\right] + \boldsymbol{\nabla} V_{\text{ext}}(\boldsymbol{r})\right)\right] \\ &- v_0\boldsymbol{\nabla}\hat{\boldsymbol{u}}(\boldsymbol{r},t) + \boldsymbol{\nabla}\xi(\boldsymbol{r},t)\sqrt{\hat{\rho}(\boldsymbol{r},t)}.\end{aligned}$$

To arrive at a deterministic, noise-independent equation for the total density, we average over possible realisations of the random noise. The resulting equation for the density field reads

$$\begin{aligned}\frac{\partial \rho(\boldsymbol{r},t)}{\partial t} =& D\nabla^2\rho(\boldsymbol{r},t) + \mu\boldsymbol{\nabla}\left[\rho(\boldsymbol{r},t)\boldsymbol{\nabla} V_{\text{ext}}(\boldsymbol{r}) + \int \mathrm{d}\boldsymbol{r}'\langle\hat{\rho}(\boldsymbol{r},t)\hat{\rho}(\boldsymbol{r}',t)\rangle\boldsymbol{\nabla} V(\boldsymbol{r}-\boldsymbol{r}')\right] \\ &- v_0\boldsymbol{\nabla}\rho(\boldsymbol{r},t)\boldsymbol{P}(\boldsymbol{r},t),\end{aligned} \tag{5}$$

with the two-point equal-time correlation function $\langle\hat{\rho}(\boldsymbol{r}, t)\hat{\rho}(\boldsymbol{r}', t)\rangle$ and the relation $\rho(\boldsymbol{r}, t)\boldsymbol{P}(\boldsymbol{r}, t) = \sum_i \boldsymbol{u}_i\rho_i(\boldsymbol{r}, t)$ for the total orientation field $\boldsymbol{P}$. Analogously to [15], one can now introduce the Helmholtz free energy functional

$$\mathcal{F}[\rho] = \mathcal{F}_{\text{id}}[\rho] + \mathcal{F}_{\text{exc}}[\rho] = \beta^{-1}\int d\mathbf{r}\rho(\mathbf{r})\left[\ln\left(\Lambda\rho(\mathbf{r})\right) - 1\right] + \int d\mathbf{r}\rho(\mathbf{r})V_{\text{ext}}(\mathbf{r}) + \mathcal{F}_{\text{exc}}[\rho] \tag{6}$$

with $\beta = \frac{1}{k_{\text{B}}T}$ (temperature T and Boltzmann constant k_{B}), and thermal wavelength Λ. The first term $\mathcal{F}_{\text{id}}[\rho]$ in equation 6 is the contribution from an ideal gas in an external

potential, and the last term $\mathcal{F}_{\text{exc}}[\rho]$ captures the interaction between particles. We do not consider any external potentials V_{ext} in this work, so this term will be dropped in the remainder of this work. Consequently, one finds that the diffusion term in equation 5 can be written in terms of the ideal gas term of the free energy:

$$D\nabla^2\rho(\boldsymbol{r},t) = D\beta\ \nabla\left[\rho(\boldsymbol{r},t)\nabla\frac{\delta\mathcal{F}_{\text{id}}[\rho(\boldsymbol{r},t)]}{\delta\rho(\boldsymbol{r},t)}\right]. \tag{7}$$

We assume in the following that both the stochastic noise and particle mobility originate from coupling to a thermal bath,in which case the Einstein relation (eq. 1) holds. Thus, the prefactor $D\beta = \mu$ of the diffusion, written in terms of a free energy functional, is the same as the prefactor of the interaction term.

Next, we need to find a way to approximate $\langle\hat{\rho}(\boldsymbol{r},t)\hat{\rho}(\boldsymbol{r}',t)\rangle =: \rho^2(\boldsymbol{r},\boldsymbol{r}',t)$ to include the particle interaction. In equilibrium, this is encoded in the excess term $\mathcal{F}_{\text{exc}}[\rho]$ of the Helmholtz free energy functional.

For every density profile $\rho_0(\boldsymbol{r})$, there exists an external potential $u(\boldsymbol{r})$ such that $\rho_0(\boldsymbol{r})$ is the equilibrium density distribution. Thus, one can relate $\rho^2(\boldsymbol{r},\boldsymbol{r}',t)$ to the excess free energy:

$$\int \mathrm{d}\boldsymbol{r}'\rho_0^2(\boldsymbol{r},\boldsymbol{r}')\nabla V(\boldsymbol{r}-\boldsymbol{r}') = \rho_0(\boldsymbol{r})\frac{\delta\mathcal{F}_{\text{exc}}[\rho_0(\boldsymbol{r})]}{\delta\rho_0(\boldsymbol{r})}. \tag{8}$$

We can always find a $\rho_0(\boldsymbol{r}) = \rho(\boldsymbol{r},t)$. If we approximate $\rho^2(\boldsymbol{r},\boldsymbol{r}',t)$ with the equilibrium two-point correlation $\rho_0^2(\boldsymbol{r},\boldsymbol{r}')$. We can insert this back into equation 5 to find

$$\begin{aligned}\frac{\partial\rho(\boldsymbol{r},t)}{\partial t} &= D\Delta\rho(\boldsymbol{r},t) + \mu\boldsymbol{\nabla}\rho(\boldsymbol{r},t)\boldsymbol{\nabla}\frac{\delta\mathcal{F}_{\text{exc}}[\rho(\boldsymbol{r},t)]}{\delta\rho(\boldsymbol{r},t)} - v_0\boldsymbol{\nabla}\rho(\boldsymbol{r},t)\boldsymbol{P}(\boldsymbol{r},t)\\ \Leftrightarrow \frac{\partial\rho(\boldsymbol{r},t)}{\partial t} &= \boldsymbol{\nabla}\left[\rho(\boldsymbol{r},t)\left(\mu\boldsymbol{\nabla}\frac{\delta\mathcal{F}[\rho(\boldsymbol{r},t)]}{\delta\rho(\boldsymbol{r},t)} - v_0\boldsymbol{P}(\boldsymbol{r},t)\right)\right].\end{aligned} \tag{9}$$

This has the form of a continuity equation $\frac{\partial\rho}{\partial t} + \boldsymbol{\nabla}\cdot\boldsymbol{j} = 0$ with the particle flux $\boldsymbol{j} = \rho\boldsymbol{v} = \rho\left(-\mu\boldsymbol{\nabla}\frac{\delta\mathcal{F}[\rho]}{\delta\rho} + v_0\boldsymbol{P}\right)$. The quantity in parentheses defines the effective velocity field $\boldsymbol{v}$.

An equation for the ensemble average can also be derived for each particle, resulting in

$$\frac{\partial\rho_i(\boldsymbol{r},t)}{\partial t} = D\left[\Delta\rho_i(\boldsymbol{r},t) + \boldsymbol{\nabla}\rho_i(\boldsymbol{r},t)\boldsymbol{\nabla}\beta\frac{\delta\mathcal{F}_{\text{exc}}[\rho(\boldsymbol{r},t)]}{\delta\rho(\boldsymbol{r},t)}\right] - v_0\boldsymbol{u}_i\boldsymbol{\nabla}\rho_i(\boldsymbol{r},t), \tag{10}$$

where $\frac{\delta\mathcal{F}_{\text{exc}}[\rho]}{\delta\rho(\boldsymbol{r},t)}$ still depends on the total density. However, it only acts on the partial density $\rho_i(\boldsymbol{r},t)$. The equation for the total density can be recovered by taking the sum over all partial densities.

In this framework of partial densities, a single hard disk at position $\boldsymbol{r}_i$ can be represented by a normalised Gaussian

$$\rho_{i,\text{peak}}(\boldsymbol{r}) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{(\boldsymbol{r}-\boldsymbol{r}_i)^2}{2\sigma^2}\right), \tag{11}$$

where the width σ of the peak is a measure for the positional uncertainty of the disk.

The following section will discuss how the excess free energy functional can be formulated for the interaction of hard disks.

2.2. Fundamental measure theory

Fundamental measure theory (FMT) was first introduced by Rosenfeld in 1989 [17] as a density functional theory for hard sphere liquids. To improve thermodynamic consistency, the White Bear (I) and later White Bear II functionals were introduced [18, 19]. The inclusion of tensorial weight functions by Tarazona (2000) [20] improved the description of densely packed fluids by accounting for anisotropies. In this work, a two-dimensional version of the tensorial FMT for hard spheres is employed, following the formulation described by Roth (2012) [21]. The works [22–24] provided valuable methodological insights for the numerical implementation of the functional.

The key idea of the FMT is that the interaction between hard particles can be represented in terms of their geometric measures. In particular, correlations of the characteristic functions of the particles yield fundamental geometric quantities, such as area and circumference. When such characteristic functions are combined through cross-correlation [3], one obtains overlap measures between particles, which form the basis of the theory. For hard disks, these correlations are particularly useful, since the overlap of two characteristic functions directly encodes whether two particles intersect or not.

The underlying physical problem in FMT is to model the interaction between identical hard disks of radius R. Their interaction potential is given by

$$V_{\text{hard}}(\boldsymbol{r}_i, \boldsymbol{r}_j) = \begin{cases} \infty, & |\boldsymbol{r}_i - \boldsymbol{r}_j| < 2R \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

From this, one defines the Mayer function depending on $r = |\boldsymbol{r}_i - \boldsymbol{r}_j|$ as

$$f(r) = \exp(-\beta V(r)) - 1 = \begin{cases} -1, & r < 2R \\ 0, & \text{otherwise.} \end{cases}. \tag{13}$$

This can be expressed in terms of the Heaviside function $f(r) = -\Theta(2R - r)$ [4]. Therefore, the Mayer function simply indicates whether two disks overlap. Now, one can decompose the Mayer function in terms of cross correlation. For hard disks, this approximately reads

$$-f(r) \approx \omega_2 \otimes \omega_0 + \omega_0 \otimes \omega_2 + \frac{\pi}{2}\omega_1 \otimes \omega_1 - \boldsymbol{\omega}_1 \otimes \boldsymbol{\omega}_1 - \frac{\pi}{4}\overleftrightarrow{\boldsymbol{\omega}}_1 \otimes \overleftrightarrow{\boldsymbol{\omega}}_1. \tag{14}$$

In even dimensions, this decomposition has an infinite number of terms; thus, one has to truncate the decomposition at some point. The corresponding weight functions ω_ν

[3] The cross correlation between two functions $f(\boldsymbol{r})$ and $g(\boldsymbol{r})$ is defined as

$$(f \otimes g)(\boldsymbol{r}) := \int_{\mathbb{R}^n} f(\boldsymbol{r}')g(\boldsymbol{r}' + \boldsymbol{r})d\boldsymbol{r}'.$$

[4] The Heaviside function Θ is defined as

$$\Theta(r) := \begin{cases} 1, & r > 0 \\ 0, & r \leq 0 \end{cases}$$

take the following form:

$$\omega_2(r) = \Theta(R-r) \tag{15}$$
$$\omega_1(r) = \delta(R-r) \tag{16}$$
$$\omega_0(r) = \frac{1}{2\pi R}\delta(R-r) \tag{17}$$
$$\boldsymbol{\omega}_1(r) = \frac{\boldsymbol{r}}{r}\delta(R-r) \tag{18}$$
$$\overleftrightarrow{\boldsymbol{\omega}}_1(r) = \frac{\boldsymbol{r}\boldsymbol{r}^\mathsf{T}}{r^2}\delta(R-r). \tag{19}$$

Here, one can see that weights represent fundamental geometric measures of the disks, such as area ($\omega_2(r)$), circumference ($\omega_1(r)$), and the number of particles ($\omega_0(r)$).

By the use of convolutions [5] the weighted densities

$$n_\nu(\boldsymbol{r}) = (\rho * \omega_\nu)(\boldsymbol{r}) \tag{21}$$

can be defined. With them, we can construct the free energy density:

$$\Phi(\{n_\nu(\boldsymbol{r})\}) = -n_0\ln(1-n_2) + \frac{1}{4\pi(1-n_2)}\left(\frac{19}{12}n_1^2 - \frac{5}{12}\boldsymbol{n}_1\cdot\boldsymbol{n}_1 - \frac{7}{6}\mathrm{Tr}\left(\overleftrightarrow{\boldsymbol{n}}_1^2\right)\right). \tag{22}$$

This finally allows us to write the excess free energy as the integral over the free energy density:

$$\beta F_{\mathrm{exc}}[\rho(\boldsymbol{r})] = \int_{\mathbb{R}^2}\Phi(\{n_\nu(\boldsymbol{r})\})d\boldsymbol{r}. \tag{23}$$

In the previous section, we have seen that we need the functional derivative of the excess free energy to insert into the dynamical equation for the partial/total density (equations 10 and 9, respectively). In the first step, we define another set of weights $T_\nu(\boldsymbol{r}) = \frac{\delta\Phi}{\delta n_\nu(\boldsymbol{r})}$ for which we find

$$T_2(\boldsymbol{r}) = \frac{n_0}{1-n_2} + \frac{1}{4\pi(1-n_2)^2}\left(\frac{19}{12}n_1^2 - \frac{5}{12}\boldsymbol{n}_1\cdot\boldsymbol{n}_1 - \frac{7}{6}\mathrm{Tr}\left(\overleftrightarrow{\boldsymbol{n}}_1^2\right)\right) \tag{24}$$
$$T_1(\boldsymbol{r}) = \frac{19}{24\pi(1-n_2)}n_1 \tag{25}$$
$$T_0(\boldsymbol{r}) = -\ln(1-n_2) \tag{26}$$
$$\boldsymbol{T}_1(\boldsymbol{r}) = -\frac{5}{24\pi(1-n_2)}\boldsymbol{n}_1 \tag{27}$$
$$\overleftrightarrow{\boldsymbol{T}}_1(\boldsymbol{r}) = -\frac{7}{6\pi(1-n_2)}\begin{pmatrix}\frac{1}{2}(\overleftrightarrow{\boldsymbol{\omega}}_1)_{xx} & (\overleftrightarrow{\boldsymbol{\omega}}_1)_{xy}\\ (\overleftrightarrow{\boldsymbol{\omega}}_1)_{yx} & \frac{1}{2}(\overleftrightarrow{\boldsymbol{\omega}}_1)_{yy}\end{pmatrix}. \tag{28}$$

[5] The convolution of two functions $f(\boldsymbol{r})$ and $g(\boldsymbol{r})$ is defined as

$$(f*g)(\boldsymbol{r}) := \int_{\mathbb{R}^n} f(\boldsymbol{r}')g(\boldsymbol{r}-\boldsymbol{r}')d\boldsymbol{r}' \tag{20}$$

We can now write the functional derivative of the excess free energy as a sum of convolutions

$$\beta \frac{\delta F_{\text{exc}}[\rho(\boldsymbol{r})]}{\delta \rho(\boldsymbol{r})} = \sum_{\nu} \xi_{\nu} (T_{\nu} * \omega_{\nu})(\boldsymbol{r}), \tag{29}$$

where the chain rule for functional derivatives was used. The factor ξ_{ν} is $+1$ for symmetric and -1 for antisymmetric weight functions ω_{ν}.

To summarise, we derived a way to represent the derivative of the excess free energy functional $\beta \frac{\delta F_{\text{exc}}[\rho(\boldsymbol{r})]}{\delta \rho(\boldsymbol{r})}$ by multiple consecutive convolutions of the weights ω_{ν} with the total density $\rho(\boldsymbol{r}, t)$. This provides all the necessary equations for a numerical implementation.

3. Numerical implementation

In this chapter, we will discuss how the previously derived model is numerically implemented to simulate the dynamics of the system. The implementation of the simulation box, the region of space where the dynamical equation is solved, and the equations from fundamental measure theory are strongly influenced by the Bachelor thesis of M. Hoffmann [24], which was based on [23], [25], and [26]. All code was written in C++, where the Eigen3 library [27] was used for convenient array handling.

3.1. Rescaling

As a first step towards a numerical implementation, the dynamical equation is rescaled so that only independent parameters remain. The length scale of the system can be defined with respect to the radius of the hard interaction R. Similarly, R^2/D has the dimension of time, and D/R has the dimension of velocity. This means we can define the time and the self-propulsion velocity in terms of these quantities. Thus, after applying the scaling rules

$$\begin{aligned} \boldsymbol{r} &\to \frac{\boldsymbol{r}}{R}, \\ t &\to \frac{D}{R^2} t, \\ v_0 &\to \frac{R}{D} v_0, \end{aligned}$$

the dynamical equation (for the partial densities) reads

$$\frac{\partial \rho_i}{\partial t} = \Delta \rho_i + \boldsymbol{\nabla} \left(\rho_i \boldsymbol{\nabla} \beta \frac{\delta \mathcal{F}_{\text{exc}}[\rho]}{\delta \rho} \right) - v_0 \boldsymbol{u}_i \nabla \rho_i. \tag{30}$$

The self-propulsion velocity v_0 is the only remaining free parameter that allows us to vary the activity of the system.

3.2. Simulation box

Next, we have to set up the simulation box. A convenient choice is a quadratic box with periodic boundaries built from a regular grid. The necessary quantities to set this up are shown in table 1.

box parameters	
Number of sites in x, y direction	$N_x = N_y$
particle number	N
vacancy concentration	n_{vac}
box length	L

Table 1: Parameters of the simulation box

Further, the following relations for the box volume V, the grid spacing Δx,Δy, and the mean density ρ_{bulk} hold:

$$V = L^2, \tag{31}$$

$$\Delta x = \frac{L}{N_x}, \Delta y = \frac{L}{N_y}, \tag{32}$$

$$\rho_{\text{bulk}} = \frac{N(1 - n_{\text{vac}})}{V} = \frac{1}{V} \int_{\text{box}} d\boldsymbol{r} \rho(\boldsymbol{r}). \tag{33}$$

$$\tag{34}$$

From these quantities, one can calculate the packing fraction

$$\eta = \frac{\pi R^2}{V} N(1 - n_{\text{vac}}) = \pi R^2 \rho_{\text{bulk}}. \tag{35}$$

Due to the periodic boundary conditions and the finite grid spacing, only the discrete wavenumbers $k_{n,x/y} = \frac{2\pi n}{L_{x/y}}$ are possible. Therefore, the spacing of the wave numbers is $\Delta k_{x/y} = \frac{2\pi}{L_{x/y}}$.

3.3. Initial density field

As discussed in section 2.1, we initially want to use Gaussian peaks to represent particles with some positional uncertainty (see eq. 11). The total density is then

$$\rho(\boldsymbol{r}) = \sum_i \rho_{i,\text{peak}}(\boldsymbol{r}) + \rho_{\text{bg}}, \tag{36}$$

where the background density ρ_{bg} has been added for reasons we will see later. By using equation 34 and the fact that the peak density $\rho_{i,\text{peak}}(\boldsymbol{r})$ is normalised, one finds

$$\rho_{\text{bg}} = \frac{-N n_{\text{vac}}}{V}. \tag{37}$$

We see that the vacancy concentration can be used to add some density if chosen negatively. This is opposite to the usual use case of removing some density due to vacancies. When considering the partial densities, the background density is equally split, yielding $\rho_{i,\text{bg}} = \rho_{\text{bg}}/N$, which results in the partial density

$$\rho_i(\boldsymbol{r}) = \rho_{i,\text{peak}}(\boldsymbol{r}) - \frac{n_{\text{vac}}}{V}. \tag{38}$$

From the fact that we are considering hard disks follows that the local density must be less than 1. This is also directly visible from T_0 in 28. We can use this to find an upper bound for the background density. For sharp peaks, nearly all the mass is concentrated within one radius R around the peak position; hence, the contributions

from the background and other peaks are limited. If only particle i contributes because no other particle is close, one needs to solve

$$\int_{|\boldsymbol{r}|<R} \rho_{i,\text{peak}}(\boldsymbol{r}, \boldsymbol{r}_0 = \boldsymbol{0}) + \rho_{\text{bg}} d\boldsymbol{r} < 1$$

and thus finds

$$\rho_{\text{bg}} < \frac{1}{\pi} \exp\left(-\frac{1}{2\sigma^2}\right) \tag{39}$$

$$\Leftrightarrow n_{\text{vac}} > -\frac{V}{N} \exp\left(-\frac{1}{2\sigma^2}\right). \tag{40}$$

Additionally, when particles are represented by Gaussian peaks, the vacancy concentration cannot be chosen to be $n_{\text{vac}} > 0$, because that would introduce a background density $\rho_{\text{bg}} < 0$, leading to an unphysical negative density when not at a peak.

Furthermore, a condition for the grid spacing can be derived by requiring that the density change from one pixel to another is significantly smaller than the peak height

$$\Delta x \max |\partial_x \rho_{i,\text{peak}}(\boldsymbol{r})| < a \cdot \rho_{i,\text{peak}}(\boldsymbol{r} = \boldsymbol{0})$$
$$\Delta x < \sqrt{e} a \sigma$$

where a controls the maximum density change and should be $< 1/3$. We can use $L = \Delta x N_x$ to find

$$N_x > \frac{L}{\sqrt{e} a \sigma}. \tag{41}$$

Fulfilling this condition ensures that, for a given box length L, the number of sites is large enough to sufficiently resolve the Gaussian peak.

3.4. Implementation of the orientation

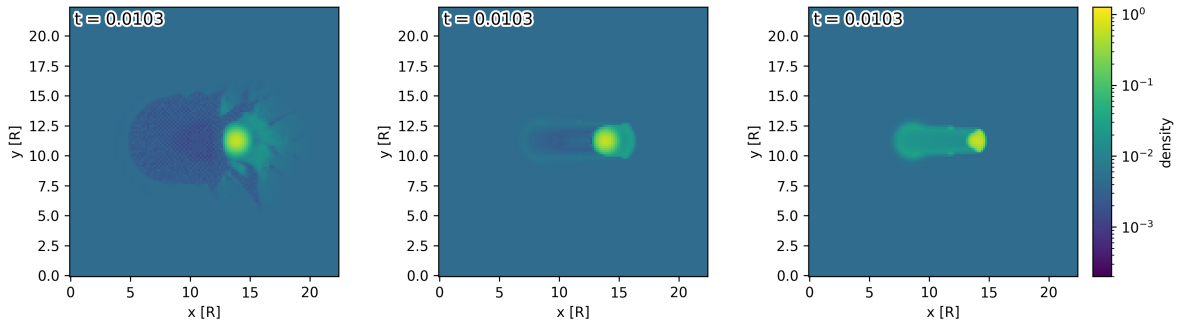
One option for the type of model used is to derive a separate dynamical equation for the orientation field, which can be solved in parallel [13, 14]. However, because neither orientational diffusion nor interaction is considered in this work, two different approaches were investigated.

3.4.1. Total orientation field

If the Gaussian peaks do not overlap significantly ($\rho_i \cdot \rho_j \approx 0$), the region of the orientation field corresponding to one density peak can be set to the orientation of the respective particle. So, every pixel must be attributed to some particle, at least if the density there is greater than a threshold $\rho_{\text{th}} \geq \rho_{\text{bg}}$. This is done using a steepest ascent algorithm, which is outlined in the following:

1. Start from one pixel with $\rho > \rho_{\mathrm{th}}$.
2. Check, which of the four neighbouring pixels has the largest density. Continue with that pixel.
3. Repeat (2) until a local maximum or a pixel included in a previous path is found.
4. Attribute all sites on the path to the maximum (or the maximum associated with the previous path).
5. Repeat from step (1) with the next pixel until all pixels with $\rho > \rho_{\mathrm{th}}$ have been checked.

At $t = 0$, the density field is a superposition of Gaussian peaks; thus, each local maximum corresponds to the center of a peak, which makes it straightforward to identify the associated particle. When the peaks evolve in time, their motion can be tracked by linking each maximum at time t_2, denoted by $\boldsymbol{r}_{\mathrm{max},j}(t_2)\}$, to the closest maximum from the previous set $\{\boldsymbol{r}_{\mathrm{max},j}(t_1)\}$ at an earlier time $t_1 < t_2$. This way of linking peak positions to particles is unambiguous only if the distances between peaks within $\{\mathbf{r}_{\mathrm{max},j}(t_2)\}$ are larger than the distances that the peaks have moved. Good temporal resolution is required to ensure this is the case. However, Gaussian peaks might become distorted due to interactions. Therefore, it is possible for multiple local maxima to form from one Gaussian. The particle areas (determined by steepest ascent) surrounding these maxima must be associated with the particle of the original peak. Thus, after applying the previously described procedure, there are now multiple maxima associated with one particle. However, we track only maxima that are large enough, which means we only consider them for the association check at the next time t_3. This allows the splitting of one Gauss peak into multiple (distanced) peaks, while reducing the probability that a small local maximum at the border of a particle area falsely overtakes areas of another particle. It was found that $\rho_{\mathrm{th}} \approx 2...3\rho_{\mathrm{bg}}$ is a good range. For smaller thresholds, a depletion zone is created behind the particle, resulting in an increasing integral of the density over the region of a single particle. On the other side, for larger ρ_{th}, the integral of the density decreases by leaving density behind. To conclude, the procedure described in this section allows to update the orientation field by tracking the area of one particle and setting the orientation field in this area to the orientation of that particle.



(a) $\rho_{\text{th}} = 0.1\rho_{\text{bg}}$. (b) $\rho_{\text{th}} = 2\rho_{\text{bg}}$ (c) $\rho_{\text{th}} = 10\rho_{\text{bg}}$

Figure 1: Comparison of the effect of different ρ_{th}.

3.4.2. Partial densities

Another option is to treat the total density as the sum of the partial densities. By using the partial density fields ρ_i (with $\rho_i(\boldsymbol{r}, t = 0) = \rho_{i,\text{peak}}(\boldsymbol{r}) + \rho_{i,\text{bg}}$), there is no longer the need for an orientation field. The orientation $\boldsymbol{u}_i$ of each particle can be used directly instead of the orientation field. This makes the implementation of the orientation trivial at the cost of having to solve N equations, connected by the excess free energy term.

In the following, equations involving the dynamical equation will be formulated in terms of partial densities ρ_i and the total density ρ. One can easily recover the case equations for the total density field by taking the sum over all particles i.

3.5. Fourier transformation

For a well performing Fourier transform, the implementation of the discrete Fourier transform from the FFTW3 library [28] is used. However, this implementation is not normalised. Here, we chose to place the normalisation factor in the forward transform so that the (inverse) discrete Fourier transform then reads

$$DFT[f(\boldsymbol{r}_n)](\boldsymbol{k}_j) = \frac{1}{N_x N_y} \sum_{\boldsymbol{r}_n} f(\boldsymbol{r}_n) \exp(-i\boldsymbol{k}_j \boldsymbol{r}_n) = \hat{f}(\boldsymbol{k}_j) \tag{42}$$

$$DFT^{-1}[\hat{f}(\boldsymbol{k}_j)](\boldsymbol{r}_n) = \sum_{\boldsymbol{k}_j} \hat{f}(\boldsymbol{k}_j) e^{i\boldsymbol{k}_j \boldsymbol{r}_n} = f(\boldsymbol{r}_n), \tag{43}$$

where $\{\boldsymbol{r}_n\}$ are the sites and $\{\boldsymbol{k}_j\}$ are the possible wavevectors of the box. Using the (inverse) continuous Fourier transform

$$\mathcal{F}[f(\boldsymbol{r})](\boldsymbol{k}) = \int_{\mathbb{R}^2} f(\boldsymbol{r}) e^{-i\boldsymbol{k}\boldsymbol{r}} d\boldsymbol{r} = \tilde{f}(\boldsymbol{k}) \tag{44}$$

$$\mathcal{F}^{-1}[\tilde{f}(\boldsymbol{k})](\boldsymbol{r}) = \frac{1}{2\pi} \int_{\mathbb{R}^2} \tilde{f}(\boldsymbol{k}) e^{i\boldsymbol{k}\boldsymbol{r}} d\boldsymbol{k} = f(\boldsymbol{r}) \tag{45}$$

we find the relation between them:

$$DFT[f(\boldsymbol{r})](\boldsymbol{k}) \approx \frac{1}{\Delta x \Delta y N_x N_y} \mathcal{F}[f(\boldsymbol{r})](\boldsymbol{k}) = \frac{1}{V} \mathcal{F}[f(\boldsymbol{r})](\boldsymbol{k}). \tag{46}$$

So, every time we initialise a function in Fourier space, we have to divide the analytical expectation by the volume of the box V.

The convolution theorem states that a convolution, as defined in equation 20, becomes a multiplication in k-space:

$$\mathcal{F}[f * g] = \mathcal{F}[f] \cdot \mathcal{F}[g]. \tag{47}$$

By applying the relation for the discrete Fourier transform, we have

$$V \cdot DFT[f * g] = V^2 \cdot DFT[f] \cdot DFT[g] \tag{48}$$

$$\Leftrightarrow DFT[f * g] = V \cdot DFT[f] \cdot DFT[g]. \tag{49}$$

So, when using discrete Fourier transforms to calculate a convolution, we have to multiply by V. This is one drawback of putting the normalisation in the forward transform. We essentially normalised one time more than necessary. This slightly worsens the performance of the fast Fourier transform by $N_x N_y$ more operations. However, this is only linear in the number of sites, while the fast Fourier transform scales with $N_x N_y \log(N_x N_y)$. Therefore, it was decided to trade that performance penalty for the convenience of the $\boldsymbol{k} = 0$ Fourier component being the mean of the transformed real field.

3.6. Excess free energy

It was shown in section 2.2 that the excess free energy term in the dynamical equation 30 can be calculated using convolutions involving the density field and the weight functions. The convolutions, as well as the discrete Fourier transforms needed for them, are calculated as in section 3.5. The first step is the calculation of the weighted densities $n_\nu(\boldsymbol{r}) = (\rho * \omega_\nu)(\boldsymbol{r}) = \mathcal{F}^{-1}\left[[\mathcal{F}[\rho(\boldsymbol{r})] \cdot \mathcal{F}[\omega_\nu(\boldsymbol{r})]\right]$. Due to the discontinuous nature of the weight functions ω_ν, the initialisation of the weight functions cannot easily be done in real space. Analytic calculations of $\mathcal{F}[\omega_\nu]$ show

$$\mathcal{F}[\omega_2(r)](k) = \frac{2\pi R}{k} J_1(kR) \tag{50}$$

$$\mathcal{F}[\omega_1(r)](k) = 2\pi R J_0(kR) \tag{51}$$

$$\mathcal{F}[\omega_0(r)](k) = J_0(kR) \tag{52}$$

$$\mathcal{F}[\boldsymbol{\omega}_1(r)](k) = -i\boldsymbol{k} \frac{2\pi R}{k} J_1(kR) \tag{53}$$

$$\mathcal{F}\left[\overleftrightarrow{\boldsymbol{\omega}}_1(r)\right](k) = -2\pi R \frac{\boldsymbol{k}\boldsymbol{k}^T}{k^2} J_2(kR) + 2\pi \frac{\boldsymbol{I}}{k} J_1(kR). \tag{54}$$

with $J_n(kR)$ the Bessel functions of the first kind [6]. These can now be directly plugged into the calculation of the weighted densities

$$n_\nu(\boldsymbol{r}) = (\rho * \omega_\nu)(\boldsymbol{r}) = DFT^{-1}\left[V \cdot DFT[\rho] \cdot \frac{1}{V}\mathcal{F}[\omega_\nu]\right](\boldsymbol{r})$$

and into the derivative of the excess free energy

$$\beta\frac{\delta F_{\text{exc}}[\rho(\boldsymbol{r})]}{\delta\rho(\boldsymbol{r})} = \sum_\nu \xi_\nu (T_\nu * \omega_\nu)(\boldsymbol{r}) = \sum_\nu \xi_\nu DFT^{-1}\left[V \cdot DFT[T_\nu] \cdot \frac{1}{V}\mathcal{F}[\omega_\nu]\right](\boldsymbol{r}).$$

Now that we know how to calculate the excess free energy term, we can address the numerical implementation of solving the dynamical equation.

3.7. Time evolution

Probably, the simplest option is Euler discretisation, where the updated density field is computed by using a first order approximation of the time derivative:

$$\frac{\rho_i^{n+1} - \rho_i^n}{\Delta t} = \frac{\partial \rho_i^n}{\partial t} \tag{55}$$

with

$$\partial_t \rho_i^n = \Delta\rho_i^n + \boldsymbol{\nabla}\left(\rho_i^n \boldsymbol{\nabla}\frac{\delta\mathcal{F}_{exc}[\rho^n]}{\delta\rho^n}\right) - v_o \boldsymbol{u_i}\boldsymbol{\nabla}\rho_i^n, \tag{56}$$

where ρ_i^n is the density field of particle i at time step n. To investigate under which conditions this Euler discretisation scheme is stable, we consider the case of an advected ideal gas where $\frac{\delta\mathcal{F}_{\text{exc}}[\rho]}{\delta\rho} = 0$. This is sufficient as long as no collision happens (e.g., for a freely moving density peak). We can insert eq. 56 into 55 and transform into Fourier space to find

$$\tilde{\rho}_i^{n+1} = \tilde{\rho}_i^n + \Delta t\left(-\boldsymbol{k}^2\tilde{\rho}_i^n - iv_o\boldsymbol{u_i}\boldsymbol{k}\tilde{\rho}_i^n\right), \tag{57}$$

where the identity $\mathcal{F}[\nabla^n f] = (i\boldsymbol{k})^n\tilde{f}$ was used. For every Fourier mode, the amplification from time step n to step n+1 is therefore given by

$$\zeta_i(\boldsymbol{k}) = \frac{\tilde{\rho}_i^{n+1}}{\tilde{\rho}_i^n} = 1 - \Delta t\left(\boldsymbol{k}^2 + iv_o\boldsymbol{u_i}\boldsymbol{k}\right). \tag{58}$$

A discrete scheme is called stable when $|\zeta_i(\boldsymbol{k})| \leq 1 \forall \boldsymbol{k}$ [29](p. 192-194), which means that no modes grow unconditionally. Using that also $|\zeta_i(\boldsymbol{k})|^2 \leq 1$, yields the following condition:

$$\Delta t \leq \frac{2}{\boldsymbol{k}^2 + v_0^2}, \tag{59}$$

[6] The Bessel function of first kind is defined as

$$J_n(x) = \sum_{l=0}^{\infty} \frac{(-1)^l \left(\frac{x}{2}\right)^{2l+1}}{l!\Gamma(l+n+1)}$$

with the gamma function $\Gamma(x) = \int_0^\infty e^{-t}t^{x-1}dt$.

where $\boldsymbol{k}\boldsymbol{u}_i \leq |\boldsymbol{k}||\boldsymbol{u}_i|$ was used. Hence, for equation 57 to be stable, equation 59 needs to be satisfied. This means that the time step is limited by both advection and diffusion. To remove the diffusion bound, one can treat that part implicitly, as will be shown in the subsequent chapter.

In the case of separate densities, the orientation does not vary spatially. Then, the advection term can also be solved implicitly, completely removing the advection bound. If one wants to consider an orientation field $\boldsymbol{P}(\boldsymbol{r})$, one could, in principle, also improve on the advection bound by using a higher order time discretisation, such as Runge-Kutta methods [30] [31]. While they can increase the possible time step, they require multiple evaluations of $\partial_t \rho_i$. Due to the large number of required FFTs to calculate the excess free energy term, this becomes computationally expensive quickly. As soon as constraints other than advection become limiting, they do not provide any advantage in the time step. On the contrary, they make it more difficult to estimate those other constraints, potentially making it necessary to recalculate the entire time step. For this reason, no higher order time discretisation was used.

Another option would be to treat spatial derivatives in the advection term in real space via an upwind scheme such as [32]. However, a first order upwind scheme suffers from numerical dissipation (adding artificial diffusion), while a 3rd order scheme like in [32] introduces numerical dispersion (resulting in depletion zones in front of and behind density peaks). In principle, they can be combined to be positivity preserving [33], avoiding the formation of depletion zones. It is also possible to match artificial and physical diffusion. This, however, depends on the specific properties of the simulation box and cannot be easily estimated. To avoid the tuning of this for every box, this ansatz was omitted.

3.7.1. Semi-implicit Euler scheme

The idea of the semi-implicit Euler method is to split $\partial_t \rho_i$ into a linear part $\mathcal{L}[\rho_i] = \Delta\rho_i$ and a non-linear part $\mathcal{N}[\rho_i, \rho] = \boldsymbol{\nabla}\left(\rho_i \boldsymbol{\nabla}\frac{\delta \mathcal{F}_{\text{exc}}[\rho]}{\delta \rho}\right) - v_o \boldsymbol{u_i} \boldsymbol{\nabla}\rho_i$. Using the same time discretisation as before and solving for ρ_i^{n+1}

$$\rho_i^{n+1} = \rho_i^n + \Delta t \left(\mathcal{L}[\rho_i^{n+1}] + \mathcal{N}[\rho_i^n, \rho^n]\right) \tag{60}$$

$$\Leftrightarrow \rho_i^{n+1} = \left(I - \Delta t \mathcal{L}\right)^{-1} \Delta t \mathcal{N}[\rho_i^n, \rho^n] \tag{61}$$

$$= \mathcal{F}^{-1}\left[\frac{\Delta t}{1 + \Delta t \boldsymbol{k}^2} \mathcal{F}\left[\mathcal{N}\left[\rho_i, \rho\right]\right]\right] \tag{62}$$

we find the update rule for the density field. Here, the linear part is evaluated at time step n+1, which is why it is called implicit. The non-linear term is still evaluated explicitly at time n. After computing $\mathcal{F}\left[\rho_i \boldsymbol{\nabla}\frac{\delta \mathcal{F}_{\text{exc}}[\rho]}{\delta \rho}\right]$, 2/3-dealiasing is applied to remove spurious high-wavenumber components generated by the multiplication of ρ_i and $\boldsymbol{\nabla}\frac{\delta \mathcal{F}_{\text{exc}}[\rho]}{\delta \rho}$ in real space.

In order to see that the semi-implicit Euler scheme actually removes the diffusion bound on the time step, a stability analysis analogous to the one for the explicit Euler

discretisation in the previous section was performed. In Fourier space, the equation 62 reads

$$\tilde{\rho_i}^{n+1} = \frac{\Delta t}{1 + \Delta t \boldsymbol{k}^2} \left(-i v_o \boldsymbol{u_i} \boldsymbol{k} \tilde{\rho}_i^n\right). \tag{63}$$

For the amplification factor, we find

$$\zeta_i(\boldsymbol{k}) = \frac{\tilde{\rho_i}^{n+1}}{\tilde{\rho}_i^n} = \frac{-i\Delta t \boldsymbol{u_i} \boldsymbol{k}}{1 + \Delta t \boldsymbol{k}^2}, \tag{64}$$

which results in the constraint on the maximum possible time step

$$\Delta t \leq \frac{2}{v_0^2}. \tag{65}$$

Comparing this to the result in equation 59, we see that it was actually possible to remove the $\boldsymbol{k}^2$, which allows a larger time step.

3.7.2. Time step

One constraint on the time step was already derived in the previous section by using the stability condition, resulting in

$$\Delta t \leq \Delta t_{\text{stab}} = \frac{2}{v_0^2}. \tag{66}$$

Another limitation arises from the fact that the density must remain positive everywhere. The linear (diffusive) term in equation 62 only smooths the density field and does not create any sinks; therefore, it can be neglected here. Hence, the following equation must be fulfilled.

$$0 \leq \rho_i^{n+1}(\boldsymbol{x}) = \rho_i^n(\boldsymbol{x}) + \Delta t \mathcal{N}\left[\rho_i, \rho\right](\boldsymbol{x}) \forall \boldsymbol{x}$$

It is sufficient to consider all $\boldsymbol{x}$ where the density decreases, meaning that $\mathcal{N} < 0$. From this, we find

$$\Delta t \leq \Delta t_{\text{pos}} = \min_{\boldsymbol{x} \in \{\boldsymbol{x}: \mathcal{N}(\boldsymbol{x}) < 0\}} \frac{\rho_i^n(\boldsymbol{x})}{-\mathcal{N}\left[\rho_i, \rho\right](\boldsymbol{x})}. \tag{67}$$

Further, because we consider hard disks, it is required that the weighted density $n_2[\rho](x)$ does not exceed 1 at any $\boldsymbol{x}$. Again, the diffusive term can be neglected. So we have to solve

$$\begin{aligned}
1 &\geq n_2[\rho^{n+1}] \\
&= \sum_i n_2\left(\rho_i^n + \Delta t \mathcal{N}[\rho_i^n, \rho^n]\right) \\
&= n_2[\rho^n] + \Delta t \sum_i n_2\left(\mathcal{N}[\rho_i^n, \rho^n]\right) \\
&= n_2[\rho^n] + \Delta t \Delta n_2,
\end{aligned}$$

where the linearity of n_2 was used and $\Delta n_2 := \sum_i n_2\left(\mathcal{N}[\rho_i^n, \rho^n]\right)$. By solving for Δt we find the 3rd constraint

$$\Delta t \leq \Delta t_{\mathrm{n}_2} = \min_{\boldsymbol{x}\in\{\boldsymbol{x}:\Delta n_2(\boldsymbol{x})>0\}} \frac{1 - n_2[\rho^n]}{\Delta n_2} \tag{68}$$

Every iteration step, the size of the time step is chosen adaptively as

$$\Delta t = 0.1 \min(\Delta t_{\mathrm{stab}}, \Delta t_{\mathrm{pos}}, \Delta t_{\mathrm{n}_2}) \tag{69}$$

where an additional safety factor of 0.1 has been added.

3.8. FMT ripples

In section 3.7.1 it was shown that the semi-implicit Euler iteration is stable without the FMT term. However, in the case of a collision (or generally high local density), this term cannot be neglected. When considering equilibrium density fields, it causes crystallisation above some critical density threshold, while for collisions, it results in additional density displacement. Both cases lead to the growth of some modes, so $|\zeta_i(\boldsymbol{k})| > 1$ for some $\boldsymbol{k}$. This is, in principle, expected and wanted behaviour. However, when treating steep gradients or discontinuous functions like the weights ω_ν in the excess free energy term via discrete Fourier transforms, over- and undershoots, so-called Gibbs ripples [34], occur due to the limited resolution. Hence, the density is pushed towards

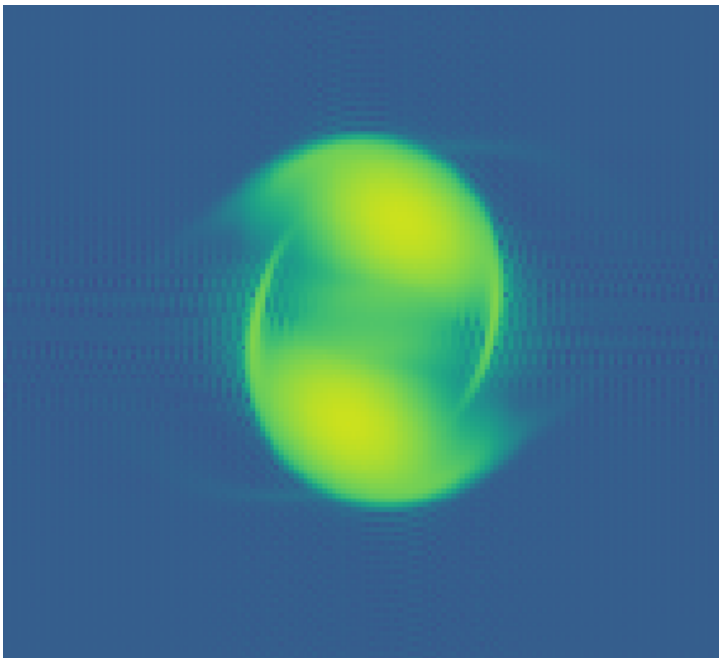


Figure 2: Gibbs ripples visible at the top and bottom, as well as to the left and right of two interacting density peaks.

0 at some sites close to the Gauss peak resulting in the time step vanishing due to the positivity constraint. To tackle this, three approaches were tested:

1. Addition of a small density background to the Gaussian density peaks.

2. Setting the density to a threshold $\rho_{\min}$ if it falls below that threshold. Subsequently, the density field has to be rescaled for mass conservation to be fulfilled.

3. Applying local smoothing where $\rho < \rho_{\min}$

Option one merely delays the vanishing time step a little until the ripples build up. A larger background can, in principle, delay this further, but this is neither limitless possible with the density implementation outlined in sec. 3.3 nor wanted from a physical perspective because the background might alter the collision dynamics. So, this approach on its own is not sufficient; however, it can provide a little more margin and will therefore be used in combination with the other options.

The second option suffers from one major problem: the need to globally rescale the density field. By setting the density to $\rho_{\min}$, if it is undercut, one effectively adds mass. So, the density around the peak (where the ripples occur) increases, while the peaks decrease in height due to the rescaling. This global impact can potentially become large if a significant amount of mass is added, altering the collision dynamics. Hence, one has to limit the amount of mass added (e.g., to 1% of the peak mass) by choosing ρ_{min} sufficiently small. This, however, leads to Δt becoming impractically small.

Option three essentially works by filling low density sites with mass from neighbouring (high density) sites, smoothing the ripples. This has the advantage of introducing only local perturbations away from the density peaks, so they are less likely to interfere with the collision dynamics. The threshold for the smoothing was chosen at $\rho_{i,\min} = 0.5\rho_{i,\mathrm{bg}}$. If this is undercut at one pixel, 1/8 of the density of all for neighbouring sites is redistributed to that pixel. Consider an oscillation in the x-direction with a period of two pixels and an amplitude $\Delta\rho$ satisfying $0.5\rho_{i,\mathrm{bg}} < \Delta\rho \leq \rho_{i,\mathrm{bg}}$, centered around the background density$\rho_{i,bg}$. In this configuration, the minimum is sufficiently small for the redistribution rule to be applied. In the y-direction, the minima exchange equal amounts of mass, so there is no net contribution. In contrast, in the x-direction, one eighth of the mass from each neighbouring maximum is redistributed toward the central minimum. As a result, the new minimum value lies between $0.5\rho_{i,\mathrm{bg}}$ (for $\Delta\rho = \rho_{i,\mathrm{bg}}$) and $(7/8)\rho_{i,\mathrm{bg}}$ (for $\Delta\rho = 0.5\rho_{i,\mathrm{bg}}$). Correspondingly, the new maximum value lies between $1.5\rho_{i,\mathrm{bg}}$ and $(9/8)\rho_{i,\mathrm{bg}}$. This demonstrates that the scheme effectively damps oscillations while conserving mass by construction. In combination with the first option, this allows for time steps of practical size.

4. Particle collision

In this chapter, we discuss the dynamics of the collision of two or more peaks in different configurations from a qualitative perspective. Furthermore, the effects of the different tunable parameters will be discussed. When setting up the initial density field, it is important to start with a configuration where the peaks of the different particles do not overlap. For Gaussian peaks with standard deviation σ, as in equation 11, the initial distance between two peak centers was required to be larger than $\max(10\sigma, 2R)$. For very sharp peaks, we are limited by the hard disk radius. However, σ was always chosen so that $5\sigma \geq R$ since very sharp peaks are hard to simulate with finite resolution. The same vanishing peak overlap has to be true for the final configuration to ensure that the collision process is complete.

4.1. Two-particle collision

For the two-particle collision, the initial configuration is quite simple. The two peaks are positioned at $x_i = L/2 \pm 5\sigma$ and $y_i = L/2 \pm h$ with orientation $\boldsymbol{u}_i = \mp(1,0)^\mathsf{T}$, as depicted in fig. 3. The offset h controls how frontally the particles collide. For an offset $h > R$ we expect no interaction, while the collision should take longer the smaller h. For the peaks to fully lose contact after the collision without colliding another time, a box length of $L = 45\sigma$ proved to be sufficient. This results in a number of sites of $N_x \times N_y = 256 \times 256$ by using eq. 41 with $a = 1/8$ and rounding up to the next power of 2 for efficiency of the Fourier transform. After the particles have collided once, we want to stop the simulation before they collide a second time. So we allow them to move the distance $d = L/2 - 2.5\sigma = 20\sigma$. This way, we take the broadening of the peak up to +50% through diffusion into account. The limit on the diffusion is not a problem, since we want the interaction and advection to be dominant. By considering the time evolution of the width of a Gauss peak under diffusion

$$\sigma(t) = \sqrt{\sigma^2 + 2t} \tag{70}$$

we find a constraint for the time the particle can take to move the distance d:

$$t_{\text{final}} \leq \frac{5}{8}\sigma^2. \tag{71}$$

Hence, the self propulsion velocity has to fulfil

$$v_0 \geq \frac{d}{t_{\text{final}}} = \frac{32}{\sigma}. \tag{72}$$

4.1.1. Collision for different orientation implementations

First, the two implementations of the orientation have to be tested. Therefore, the movement of two density peaks with the following parameters was simulated: The self propulsion velocity was chosen at two times the lower limit derived in the previous

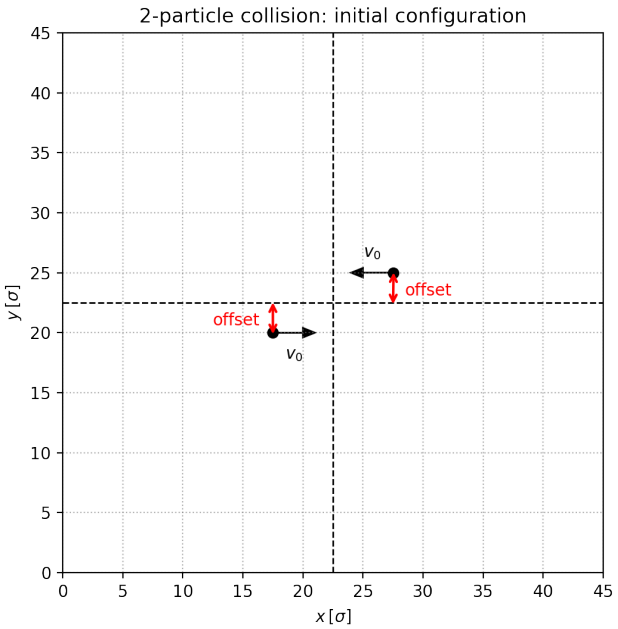


Figure 3: Initial configuration for the two-particle collision

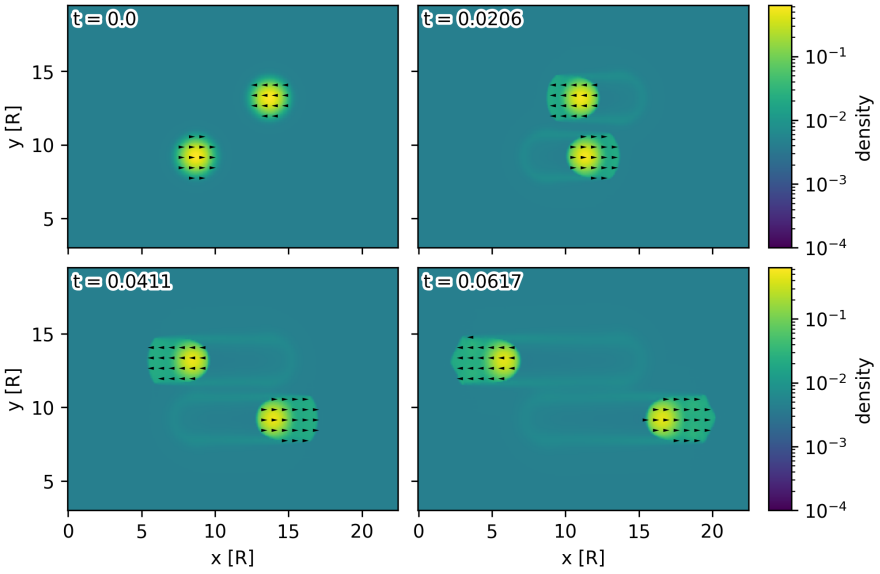
σ	0.5
v_0	$64/\sigma = 128$
ρ_{bg}	$0.1\rho_{\mathrm{bg,max}}$
ρ_{th}	$2\rho_{\mathrm{bg}}$

Table 2: Parameters for the comparison of the orientation implementations.

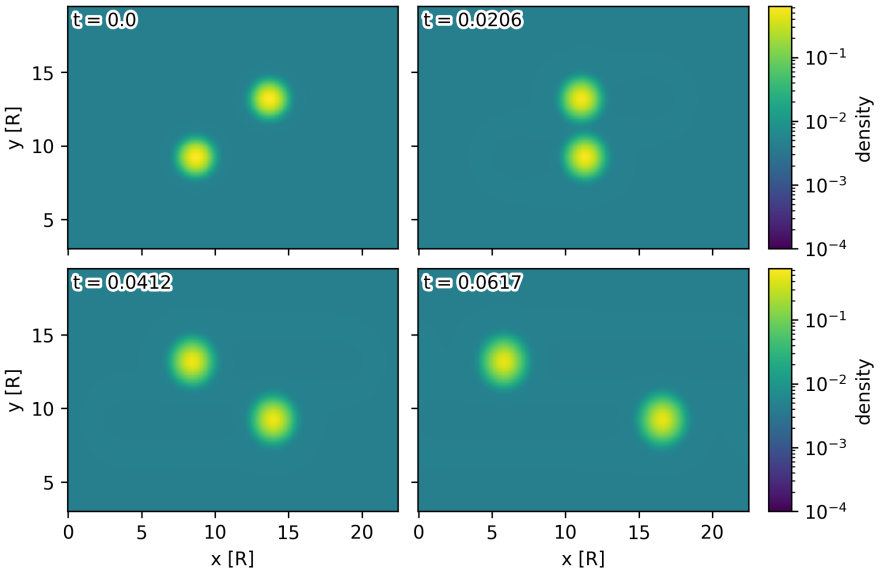
section. For an offset of $h \approx 2$ the peaks pass each other with significant distance, not influencing each other. This case of freely moving particles is shown in figure 4 for both orientation implementations. One can see that both translate the Gauss peaks reasonably well. However, the flanks of the peak stay smoother for the coupled density fields (fig. 4b), while the implementation via steepest has a sharper cut-off due to the orientation only being set down to ρ_{th}. This also leads to the orientation area for each peak extending over time in front of the particle. Also, some density is left behind at the flanks because of the cut-off. The deviations from the original Gaussian form stay small in this case of a large offset, so the steepest ascent implementation would be a feasible option, especially when considering the lower computational effort.

This, however, changes when considering a collision with offset $h = 2\Delta x \approx 0.18$. The need to track multiple local maxima per particle, in combination with the direct contact of the two peaks, makes it possible for regions to switch particle association. This process can be seen in figure 5 and results in the right particle overtaking nearly the entire left particle, while only a small portion of the density continues to move to

the right. But the total mass with one orientation direction should stay constant at ≈ 1. So this behaviour is clearly unphysical.



(a) Orientation via steepest ascent. Orientation field visualised by black arrows.



(b) Orientation via partial density fields.

Figure 4: Freely moving density peaks for different implementations of the orientation. The bottom peak is moving to the right, the top one to the left.

To better understand what happens when the 2 peaks collide, we can analyse the evolution of the individual densities shown in figure 6. Up to around $t = 0.01$, the peaks move freely before they come into contact. During the collision process between roughly $t = 0.01$ and 0.03, the individual densities have a non vanishing overlap, as shown in the

third column of images in figure 6. This behaviour cannot be captured by the steepest ascent implementation of the orientation because each pixel is assigned to either one particle or the other. For that reason, only the implementation via individual density fields was used for the remainder of the work.

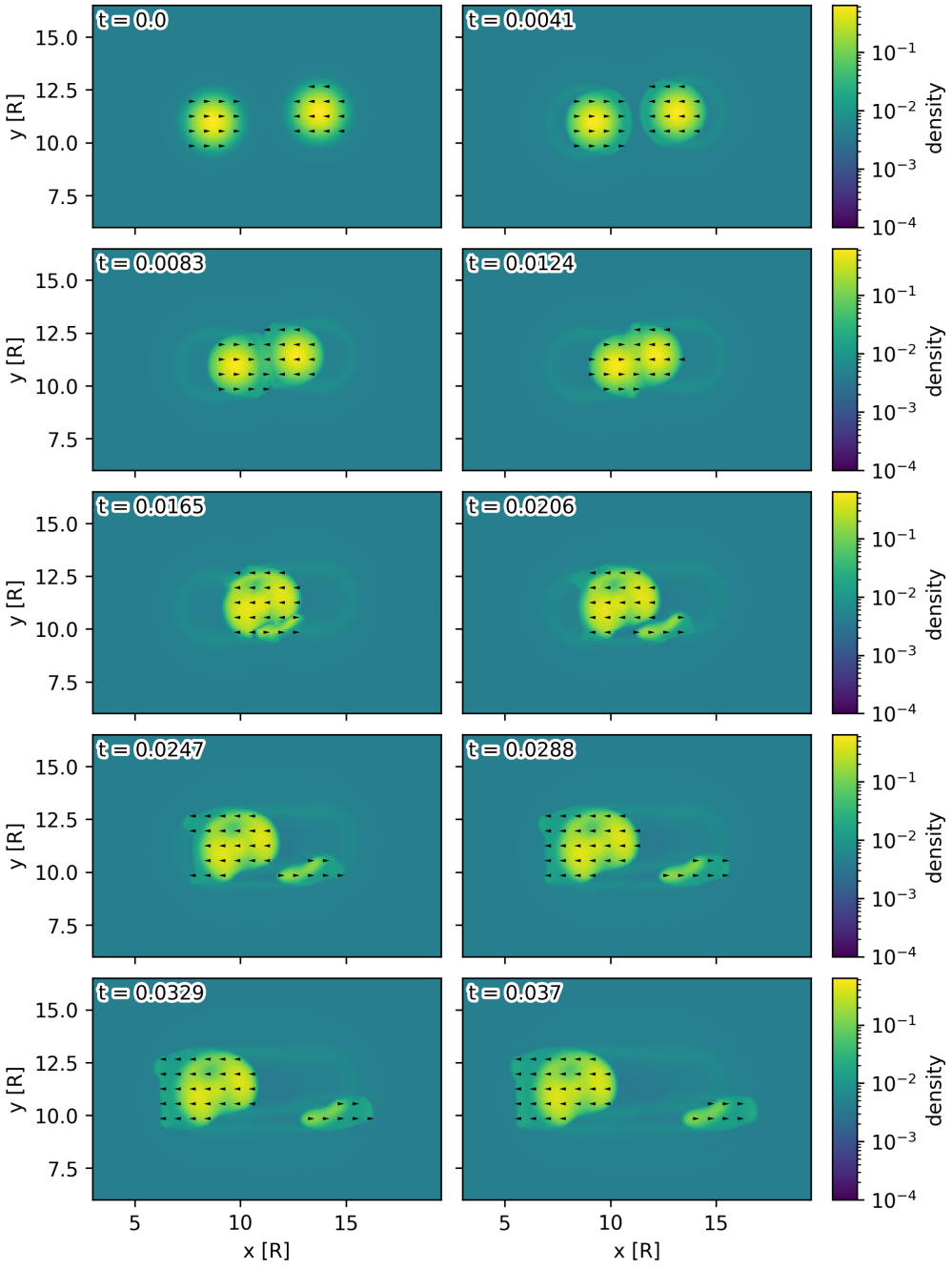


Figure 5: Collision of two peaks with offset $h \approx 0.18$. Orientation is determined by steepest ascent and indicated by black arrows. The density is plotted logarithmically for better visualisation of effects at the particle borders.

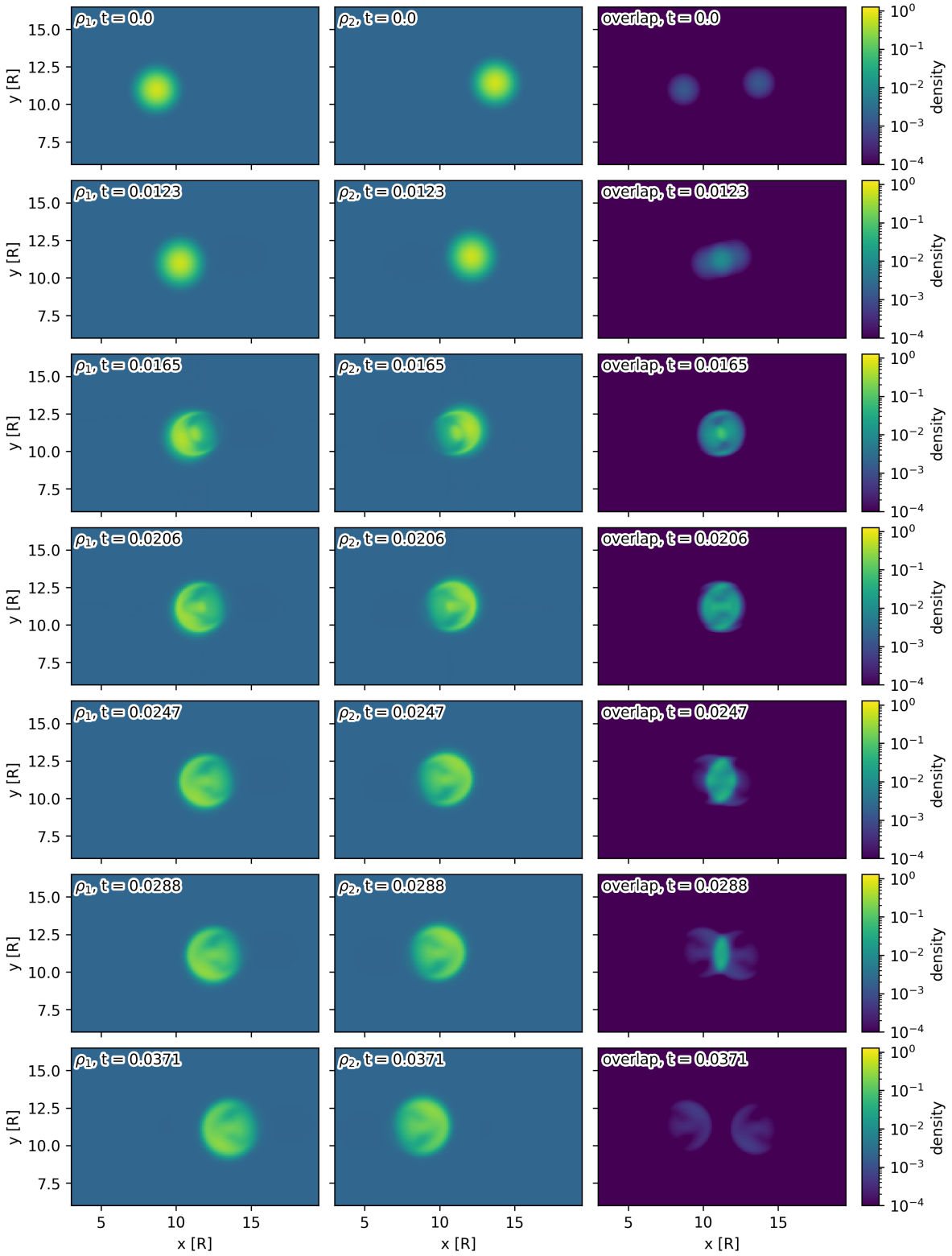


Figure 6: Collision of two peaks with offset $h \approx 0.18$. Orientation implemented via coupled density fields. The individual densities ρ_i are shown in the first two columns. In the third column the overlap $\rho_1 \cdot \rho_2$ is plotted.

In figure 6m we can further see that some of the density of the left particle is pushed up or downm resulting in a crescent shape. For the right particle, this happens analogously, only mirrored. The wider tail on the bottom, where most of the density is pushed to, aligns with the offset direction. When interpreting the density field as a probability density, the chance that the disks move around each other in offset direction is the largest. However, due to the finite width of the peak, there is also a probability to find the left disk at a higher y than the right one. In that case, the left disk will move around the right disk at the top, despite the peak being offset towards the bottom. This results in the slim upper tail. At some point, enough density was pushed to the side so that the remaining density can pass each other. To verify that the peaks are moving with the expected velocity v_0 one can compare the actual with the expected peak position. This, and an indication $4\sigma(t)$ in front, can be seen in figure 7. In case of a collision, as depicted in figure 8, the tails of the half moon stay within the expected range behind the $4\sigma(t)$-line. However, most of the density peak is pushed slightly behind the expected center position.

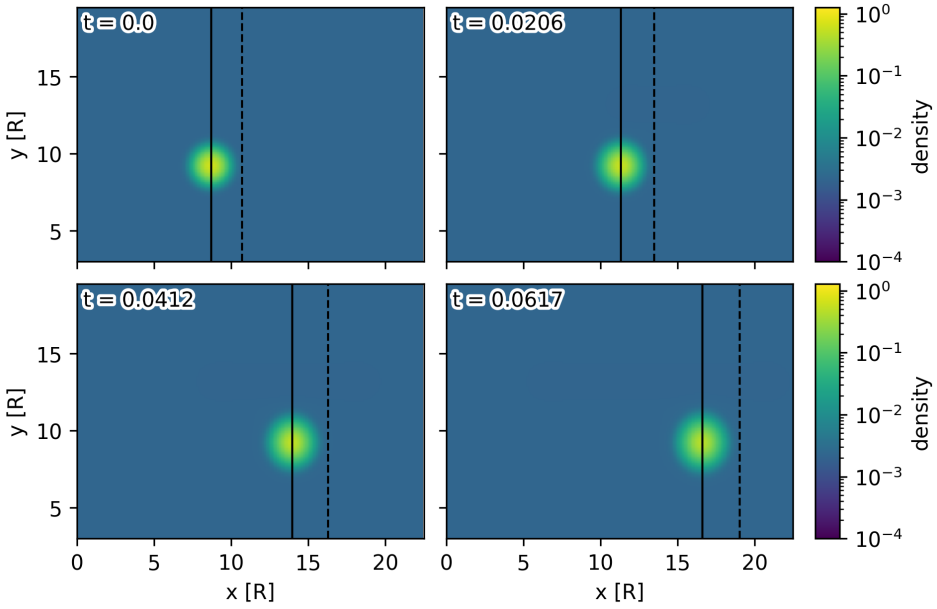


Figure 7: Time Evolution of ρ_1 at $h \approx 2$. The center of the Gaussian peak stays well at the expected position indicated by the black line. The peak also decays visibly before the dashed black line indicating a $4\sigma(t)$ distance to the center.

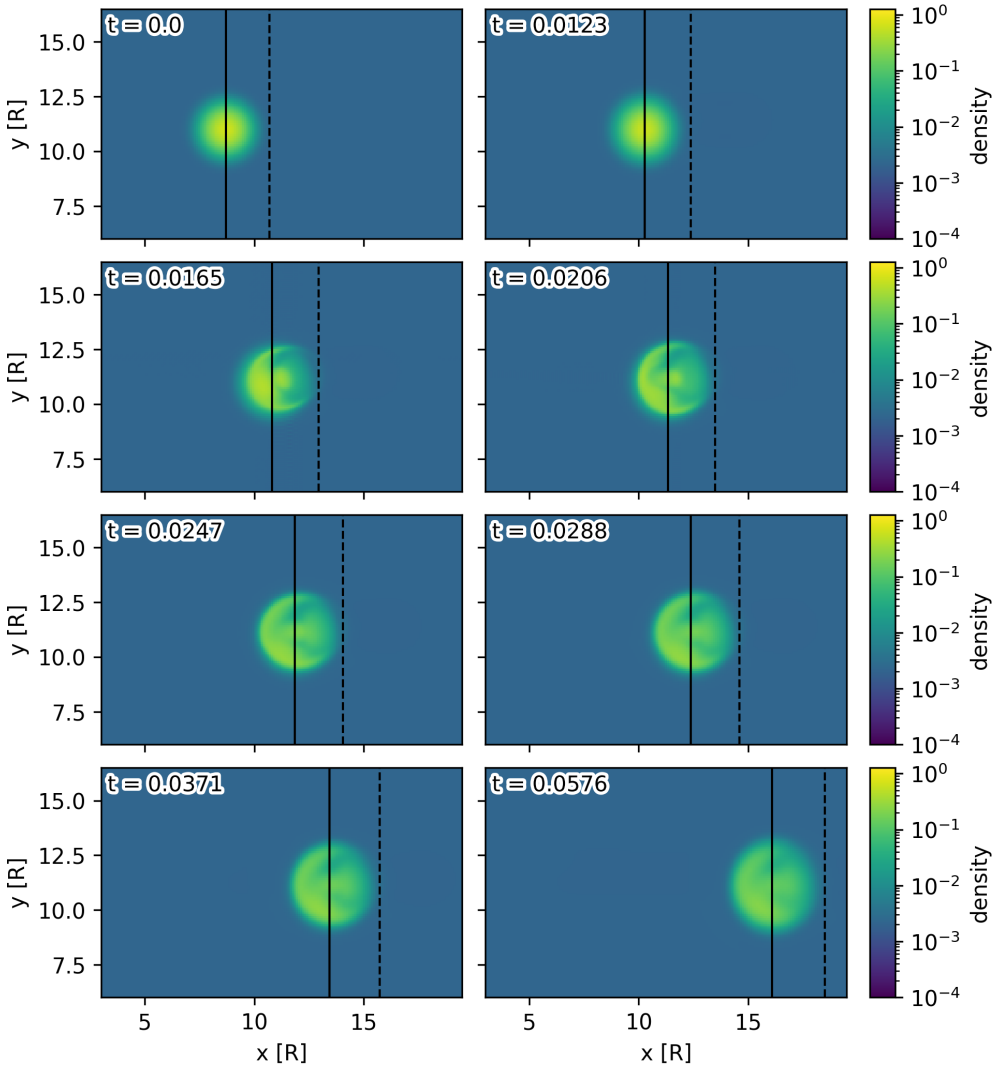


Figure 8: Time Evolution of ρ_1 at $h \approx 0.18$. The black line indicates the expected position of the peak center, while the dashed black line indicates a $4\sigma(t)$ distance to the center.

Additionally, one can look at the different contributions to the density change. For that purpose, the mean value of the absolute excess free energy and activity term are plotted in fig. 9. One can clearly see that the excess term only contributes significantly during the collision process. This validates the previous assumptions for the stability analysis in section 3.7.1.

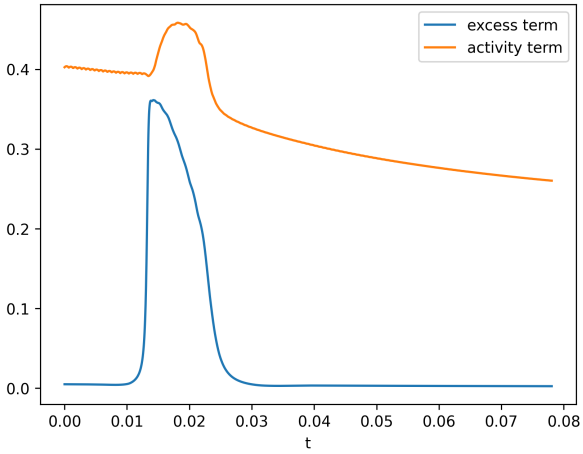


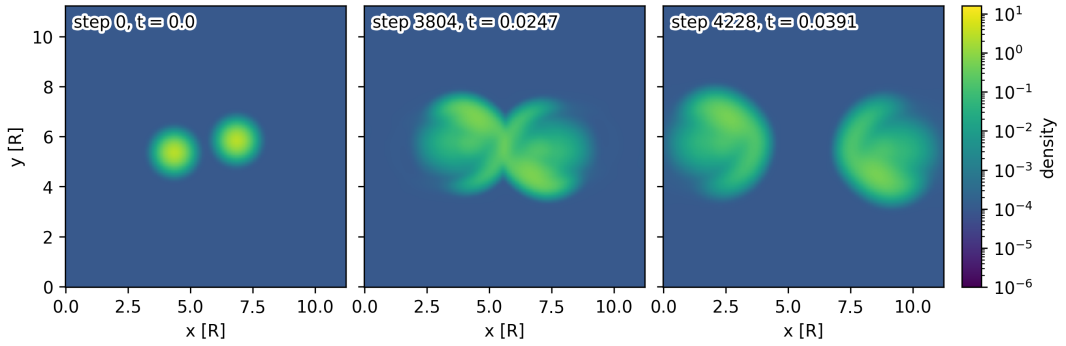
Figure 9: Contribution of the excess and activity term to the density change for the collision of two peaks with $h \approx 0.18$. The plotted excess term is mean($|\boldsymbol{\nabla}\left(\rho_1 \boldsymbol{\nabla} \frac{\delta \mathcal{F}_{\lceil \S \rfloor}[\rho]}{\delta \rho}\right)|$) and the activity term is mean($|v_o \boldsymbol{\nabla}(\boldsymbol{P_1}\rho_1)|$). The collision happened between $t \approx 0.01$ and 0.03.

4.1.2. Role of the self propulsion velocity

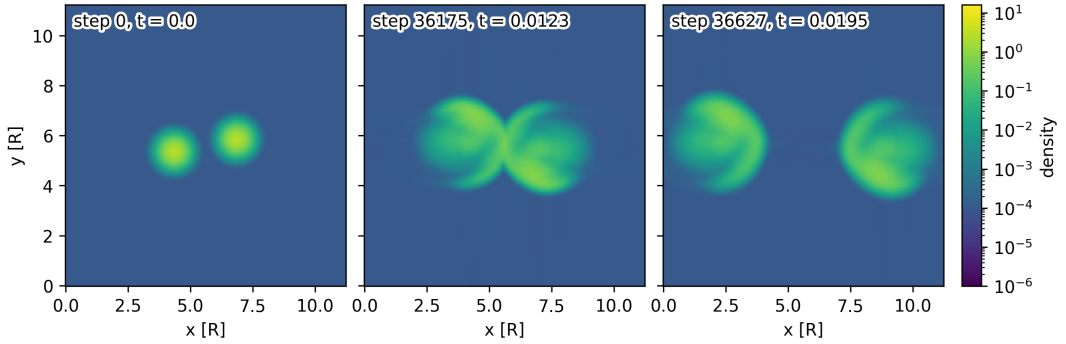
In the given model, v_0 allows for tuning the proportion of diffusion and particle interaction to activity. To investigate the influence of the self propulsion velocity on the collision process, v_0 was varied, while all other parameters were kept constant (see fig. 3). One can see in figure 10 that the final density distribution is more blurred the smaller v_0. This makes sense because the collision takes longer and therefore the influence of diffusion is larger. The higher v_0, the sharper the features become. However, as a drawback, the amount of numerical artefacts from the excess free energy term increases, visible especially for $v_0 = 1024$ (fig. 10d) below and above regions of high density. Further, the number of necessary time steps increases from 4168 at $v_0 = 128$ to 174841 for $v_0 = 2048$. There are two reasons for this: The first is that the final time $t_{\text{final}} \propto 1/v_0$ while $\Delta t_{\text{stab}} \propto 1/v_0^2$, thus the number of time steps to reach t_{final} scales linearly with v_0. But the number of steps increased by the factor ≈ 42, while v_0 only increased by factor 16. This is due to the numerical artefacts at higher velocities that no longer get balanced out by diffusion. These Gibbs ripples also limit the time step via the positivity requirement. Consequently, one has to balance sharper features against more artefacts and the higher computation time.

σ	0.25
h	$5\Delta x \approx 0.22$
ρ_{bg}	$0.9\rho_{\text{bg,max}}$

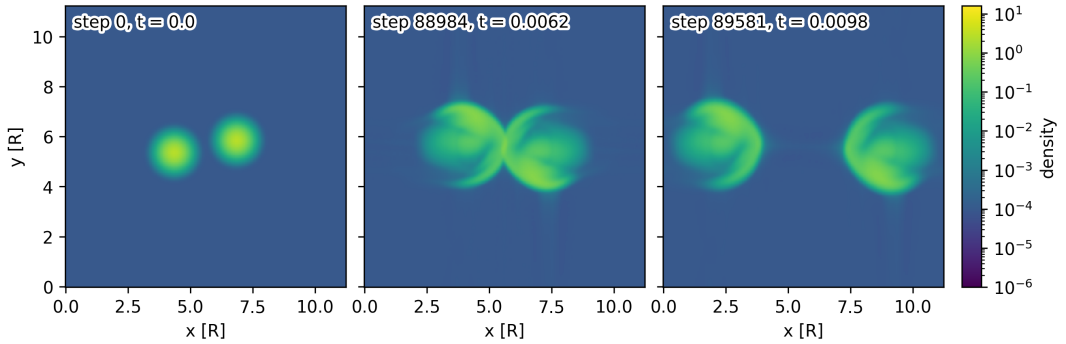
Table 3: Parameters for the two peak collision with varying v_0.



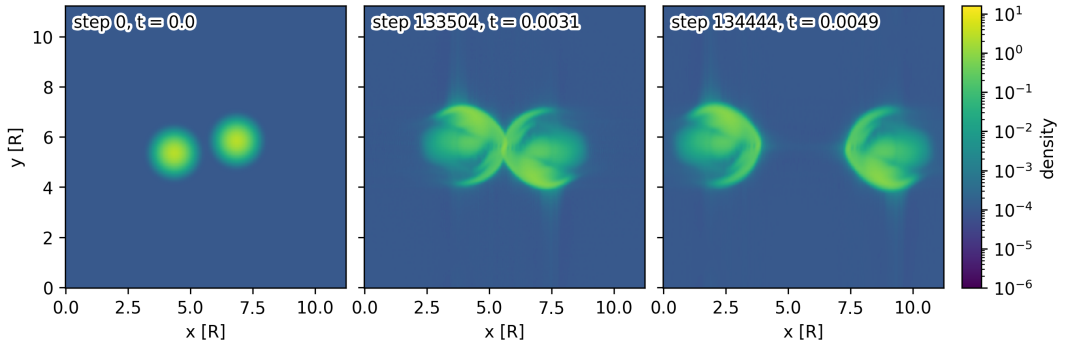
(a) $v_0 = 128$.



(b) $v_0 = 256$.



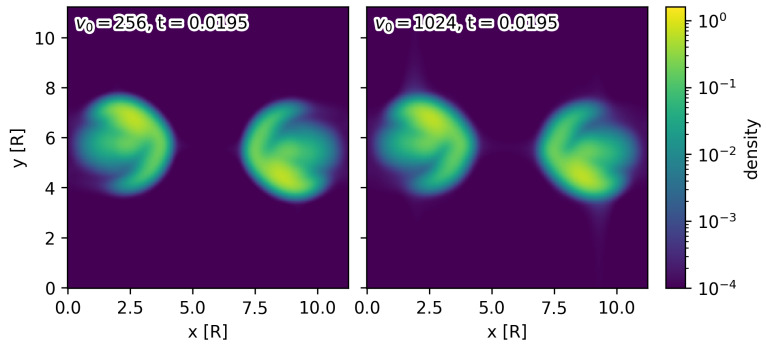
(c) $v_0 = 512$.



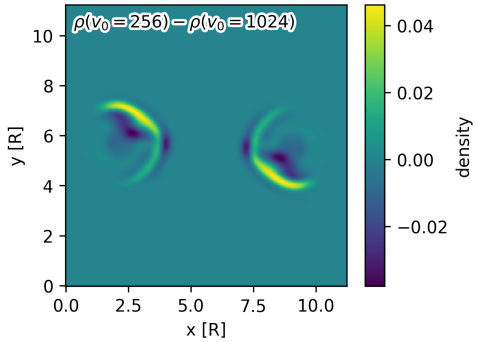
(d) $v_0 = 1024$.

Figure 10: Two-particle collision for varying v_0.

Further, the question arises as to whether there is still a difference in final density when they evolve up to the same time. For that, an additional simulation was run with $v_0 = 256$ and for $v_0 = 1024$. The faster peak arrives at the target position earlier, at which point the self propulsion velocity is set to zero. This allows the density to evolve in place without the need for a larger simulation box to avoid further collisions. The final densities are shown in figure 11. One can see in fig. 11b that more density stays close to the initial trajectory for higher v_0, while more density is spread out and moves around the other particle for the lower v_0. This is consistent with the view that more diffusion occurs before and during the collision the lower the self propulsion velocity. However, even though v_0 changes by a factor 4, the change in the density is only of order 10^{-2}, while the density itself is at least one order of magnitude larger at 10^{-1} to 10^{0}. Consequently, one would need to change the velocity by a larger factor to observe significant differences. Unfortunately, that is not easy to realise without diffusion smoothing the density so much that barely any features are visible or introducing large numerical artefacts. Due to this, only a fixed velocity of $v_0 = 512$ was investigated further, which was found to provide a good balance.



(a) Comparison of the final density field after a collision with $v_0 = 256$ and $v_0 = 1024$.

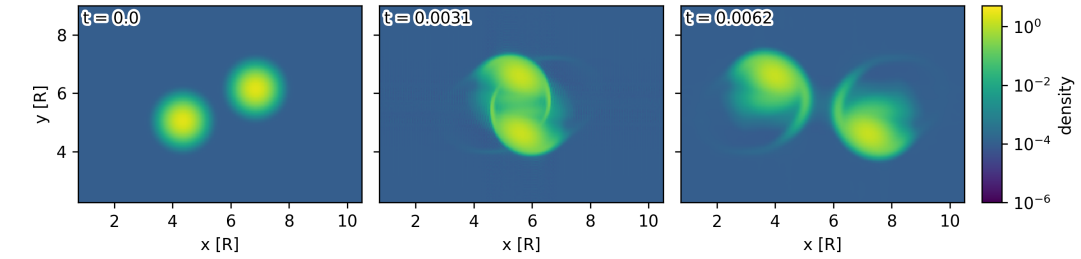


(b) Difference in the final density field after a collision with $v_0 = 256$ and $v_0 = 1024$ at $t = 0.0195$.

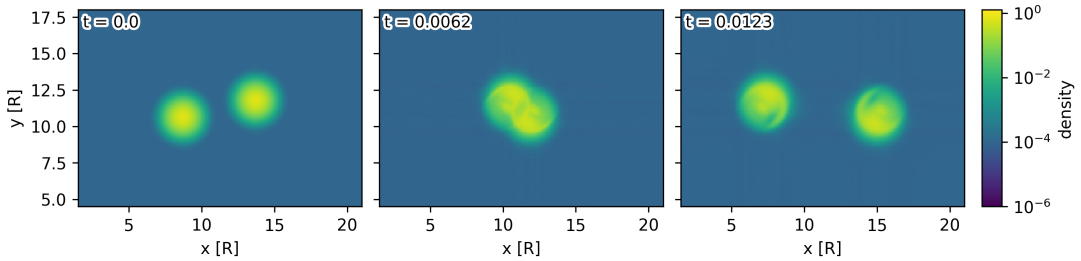
Figure 11: Density field after a collision with $v_0 = 256$ and $v_0 = 1024$ at $t = 0.0195$.

4.1.3. Role of the peak width

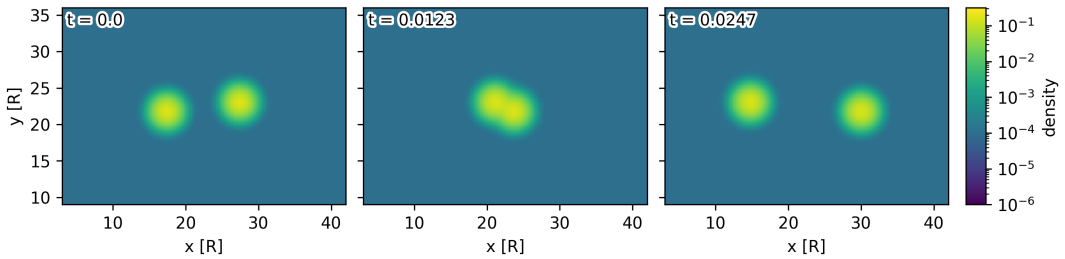
The width σ of the Gaussian density peak is a measure of the localisation of the respective particle. For $\sigma \to 0$, the position of the hard disk is exactly known. However, as the width increases, the uncertainty also increases. The effect of the peak width on the collision is shown in figure 12 for the parameters presented in table 4. One can see that the collision has the strongest effect on the shape of the peak for the smallest width tested of $\sigma = 0.25$, while for $\sigma = 1$, there is no visible effect. One might see a small effect by choosing a smaller offset (e.g., $h = 0$), because that increases the (maximum) overlap; however, that would be compensated by a slightly broader Gaussian peak. Consequently, it is sufficient to simulate collisions up to $\sigma = 1$. Regarding the effect of the width on the shape of the final density, one finds that the smaller σ, the more density is pushed towards the direction of the offset. This is expected because the probability of the left particle being at a y-coordinate larger than that of the right particle gets smaller with decreasing σ.



(a) $\sigma = 0.25$.



(b) $\sigma = 0.5$.



(c) $\sigma = 1$.

Figure 12: Two-particle collision for varying σ.

v_0	512
h	≈ 0.53
ρ_{bg}	$0.9\rho_{\mathrm{bg,max}}(\sigma = 0.25)$

Table 4: Parameters for the two-particle collision with varying σ.

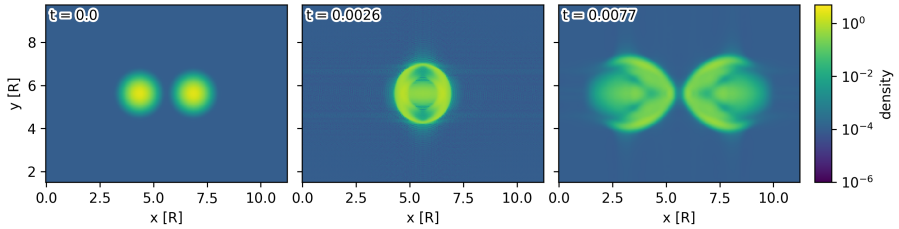
4.1.4. Role of the offset

Finally, the effect of the key parameter of interest, the offset h, is investigated. As in the previous sections, the remaining parameters were fixed at the values shown in table 5, while h was varied in steps of Δy. The collision for selected offsets is shown in figure 13.

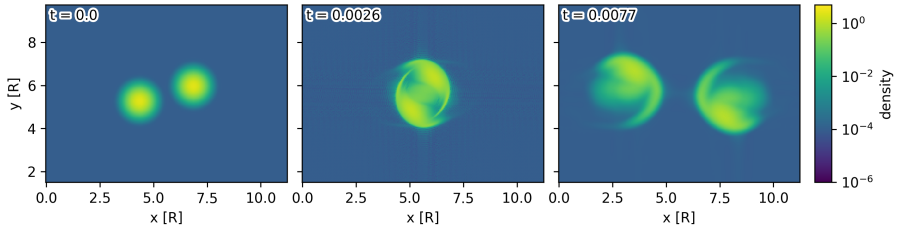
As expected in the $h \approx 1.05$ case, the density peaks pass each other without any noticeable influence (see 13e). This is because the offset is equal to the hard disk radius $R = 1$ and only a small amount of density being closer. Already at an offset of 0.88 (fig. 13d), some density starts to be pushed back and to the side during the collision. This effect gets stronger at $h \approx 0.53$, while there is even a small portion of density passing on the opposite side of the offset direction due to the fact that some density of the left particle started at a larger y-coordinate than a part of the right particle density. Over the course of further decreasing offset, the density evolution becomes increasingly more symmetrical, until, at $h = 0$ (fig. 13a), it is equally likely that one particle squeezes around the other on the top or the bottom. An increase in the peak width decreases the overall effects of the collision, as the more spread-out density reduces the probability of the underlying hard disks colliding. This can be seen in the figures 27 and 28 in the appendix.

σ	0.25
v_0	512
ρ_{bg}	$0.9\rho_{\mathrm{bg,max}}$

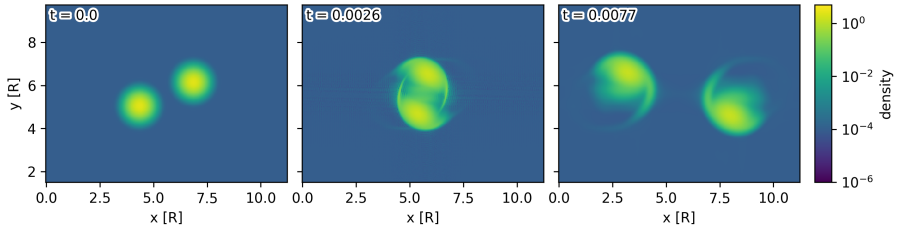
Table 5: Parameters for the two-particle collision with varying h.



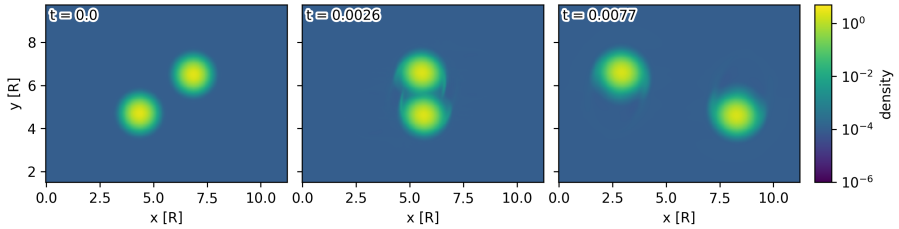
(a) $h = 0$.



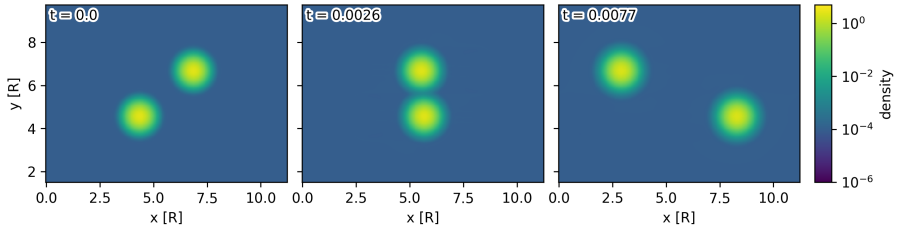
(b) $h \approx 0.35$.



(c) $h \approx 0.53$.



(d) $h \approx 0.88$.



(e) $h \approx 1.05$.

Figure 13: Two-particle collision for varying h. Parameters in tab. 5.

4.2. Multi-particle collision

The delay of the density in the case of a two-particle collision was surprisingly low (around $1R$ in the largest case) when considering that, in the particle dynamics view of two colliding hard disks (without noise), the delay goes to infinity for $h \to 0$. Due to this, we also investigated multi-particle collisions to see if any of them get stuck for a longer time.

4.2.1. Three-particle collision

The setup for the three-particle collision was very similar to the setup for two particles in section 4.1. Without an offset, the peaks were placed at the corners of an equilateral triangle with side length 10σ, with its centroid placed at the $(L/2, L/2)^\mathsf{T}$. The orientation of each peak is towards the centroid. For the addition of an offset, each peak was displaced by h orthogonal to its orientation and counter-clockwise with respect to the centroid. This initial configuration is visualised in figure 14. Due to the fact that we expect a larger delay, the box was chosen to be slightly larger at $L = 55\sigma$, while the number of pixels was kept the same. This results in a slightly lower resolution of the Gauss peaks. The travel distance for the peaks is now $d = L/2 - 2.5\sigma = 25\sigma$.

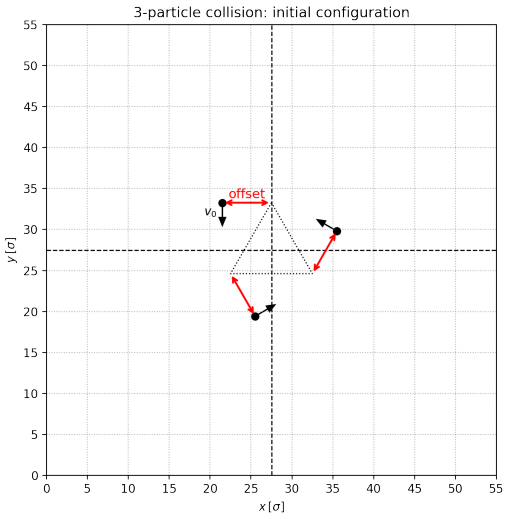
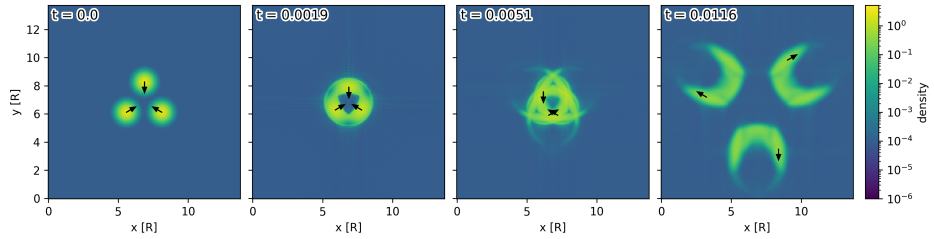


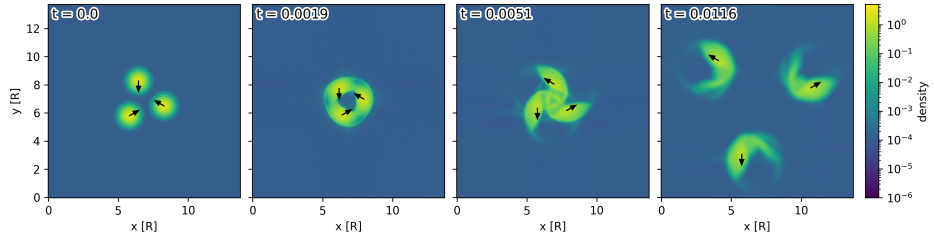
Figure 14: Initial configuration for the three-particle collision

The collision process for varying offsets is visualised in figure 15. Simulation parameters are the same as in the two-particle case (see tab. 5). The first thing to note is that there is now an effect up to $h \approx 1.1$, instead of 0.9 in the two-particle case. This occurs because the orientations (and offsets) of a pair of particles are no longer parallel. Further, there is a larger high density region in the middle, forcing density to be pushed further out. Additionally, due to the fact that there is now one more particle

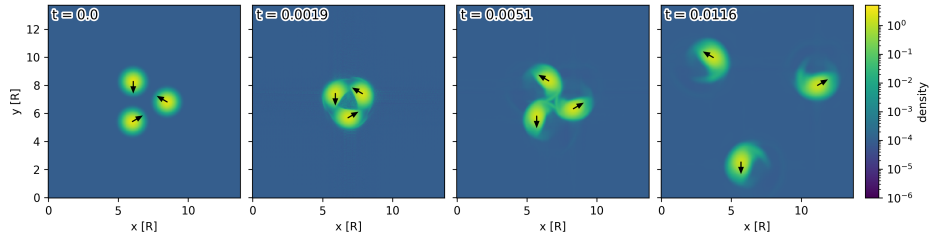
contributing, more density has to be pushed to the sides, which takes longer, before the remaining central density can move through each other. The larger number of particles also leads to a larger effect for broader peaks, as can be seen in figure 29 and 30 in the appendix.



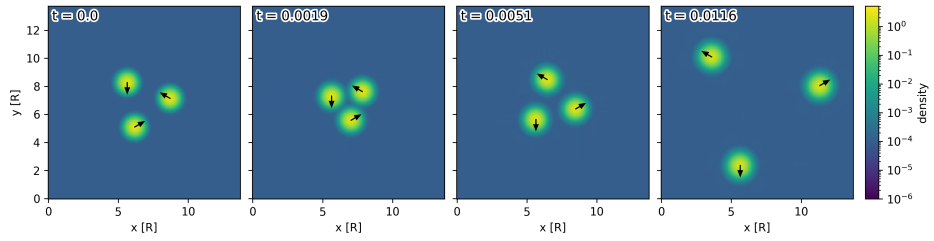
(a) $h = 0$.



(b) $h \approx 0.38$.



(c) $h \approx 0.75$.



(d) $h \approx 1.18$.

Figure 15: Three-particle collision for $\sigma = 0.25$ and varying h. Parameters in tab. 5.

4.2.2. Four-particle collision

The initial configuration for the four-particle collision is shown in figure 16. It is very similar to the configuration for the three-particle collision. The only real difference is that a square, instead of a triangle, is used as the basis for the positions of the peaks.

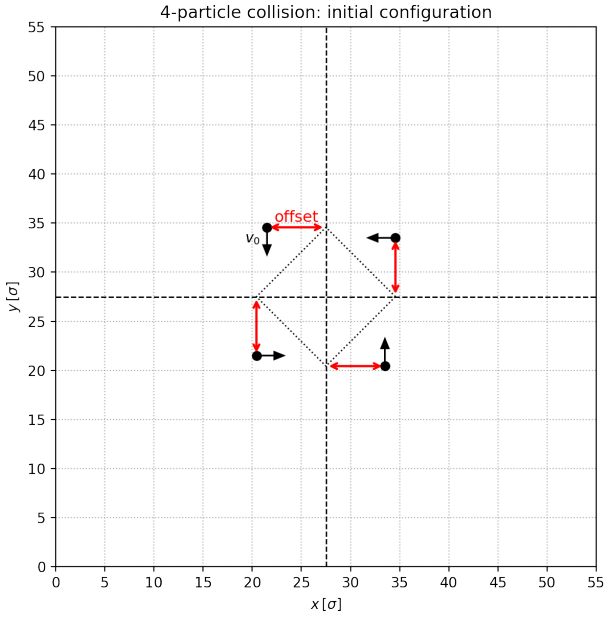
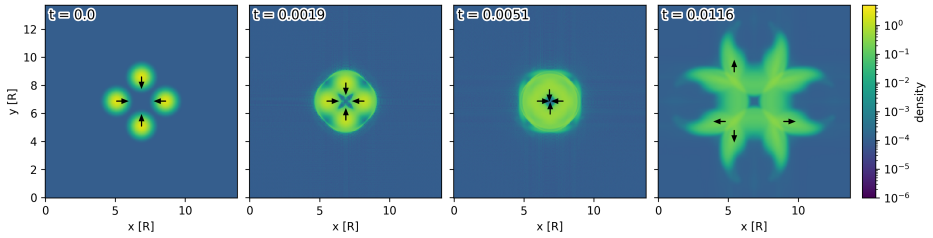
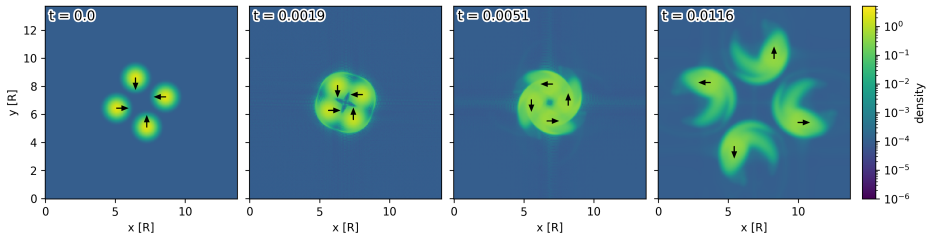


Figure 16: Initial configuration for the four-particle collision

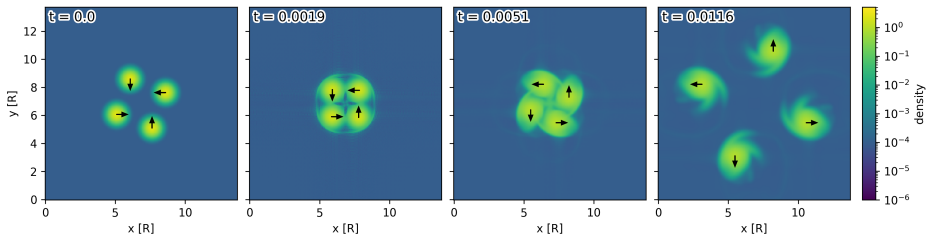
Also, the evolution of the density throughout the course of the collision follows a similar pattern, as can be seen in figure 17. The parameters are kept the same as in the two- and three-particle cases (see tab. 5). The onset of an effect is now at approximately $h \approx 1.2$. The density delay is even larger than that for the three-particle collision because the necessary displacement is increased. Even for $\sigma = 1$, there is now a visible effect (at least at small offsets). Plots for $\sigma = 0.5$ and 1 can be found in the appendix (fig. 31 and 32).



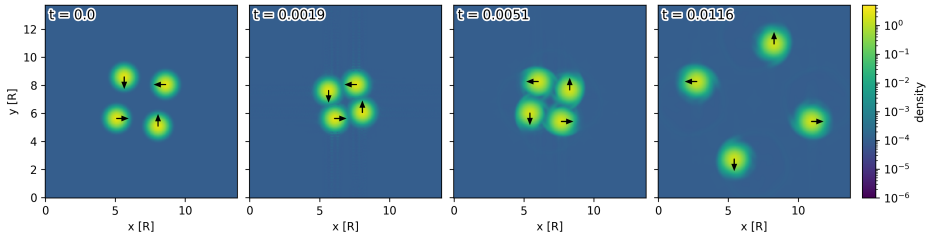
(a) $h = 0$.



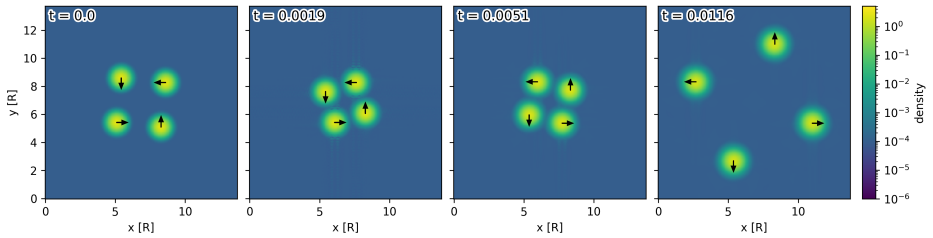
(b) $h \approx 0.38$.



(c) $h \approx 0.75$.



(d) $h \approx 1.18$.



(e) $h \approx 1.4$.

Figure 17: Four-particle collision for varying h. Parameters in tab. 5.

5. Particle delay

In this section, a quantitative analysis of the delay in density due to the peak collision is conducted. For that purpose, the data from section 4 for the two-, three-, and four-particle collisions is analysed further. The goal is to obtain the dependence of the delay on the offset.

5.1. Center of mass delay

As a first step, we can analyse how much the center of mass is delayed. The calculation of center of mass $\boldsymbol{r}_{i,\mathrm{cms}}$ for each sub-density is straightforward as long as the density peak does not extend over the borders of the simulation box. For it to be meaningful, the density distribution must not span the entire box. To fulfil this, we must subtract the background density $\rho_{i,\mathrm{bg}}$ from the respective partial density. Consequently, we have

$$\boldsymbol{r}_{i,\mathrm{cms}} = \frac{\sum_{\boldsymbol{r}\in\mathrm{box}} \boldsymbol{r}\cdot(\rho_i(\boldsymbol{r})-\rho_{i,\mathrm{bg}})}{\sum_{\boldsymbol{r}\in\mathrm{box}}(\rho_i(\boldsymbol{r})-\rho_{i,\mathrm{bg}})} = \Delta x\Delta y \sum_{\boldsymbol{r}\in\mathrm{box}} \boldsymbol{r}\cdot(\rho_i(\boldsymbol{r})-\rho_{i,\mathrm{bg}}), \tag{73}$$

where it was used that the partial densities are normalised without background. The index i will be dropped in this section for better readability. The change of the center of mass from the initial to final state is then $\Delta\boldsymbol{r}_{\mathrm{cms}} = \boldsymbol{r}_{\mathrm{cms}}(t_{\mathrm{final}}) - \boldsymbol{r}_{\mathrm{cms}}(t=0)$. However, we are only interested in the delay in movement direction, so we have to project the center of mass change onto the orientation direction using

$$\Delta\boldsymbol{r}_{\mathrm{cms},u} = (\boldsymbol{u}\boldsymbol{u}^{\mathsf{T}})\Delta\boldsymbol{r}_{\mathrm{cms}} \tag{74}$$

with the orientation $\boldsymbol{P}$. The positional delay is then given by

$$\Delta d = d - \Delta\boldsymbol{r}_{\mathrm{cms},u}, \tag{75}$$

with the expected travelled distance $d = L/2 - 2.5\sigma = v_0/t_{\mathrm{final}}$. This is done for all partial densities separately and then averaged. One can easily calculate the temporal delay $\Delta t = \Delta d/v_0 = \Delta d/512$. In section 4.1.2, it was discussed that the final density is mostly independent of v_0, except for the amount of diffusion. However, diffusion does not change the center of mass. Therefore, the time delay scales approximately like $\propto v_0^{-1}$.

When examining the positional delay for the two-particle collision in figure 18, we see that for a large peak width of $\sigma = 1$, there is no delay. For $\sigma = 0.5$ and 0.25, we observe an onset of delay around $h = 0.8$ and 1, respectively. The delay increases from the point of onset up to $h = 0$, with a larger delay the narrower the peak. This is expected because, in the case of an exactly localised particle ($\sigma \to 0$), the delay should approach $d \to \infty$ (at least without diffusion). Apparently, the increase follows a roughly Gaussian dependence on h. Within the analysed parameter range, the delay also increases more than linearly with the decrease of the peak width. In principle, datasets for different σ are not directly comparable because of different t_{final}. However, pure diffusion leaves the center of mass invariant. Thus, the only difference might arise from diffusion before

and during the collision, which does not significantly influence the resulting density distribution (as discussed in section 4.1.2 in the context of varying v_0). For three- and four-particle collisions (figures 19 and 20), the curves appear very similar. The onset shifts to slightly larger h, and the maximum of d at $h = 0$ increases as well.

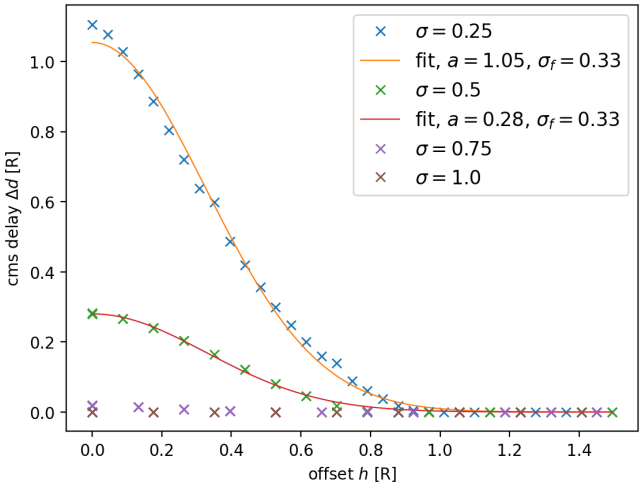


Figure 18: Center of mass delay for two-particle collision. For the data corresponding to $\sigma = 0.25$ and 0.5 also a Gaussian fit of the form $a \exp\left(-\frac{h^2}{2\sigma_f^2}\right)$ is shown.

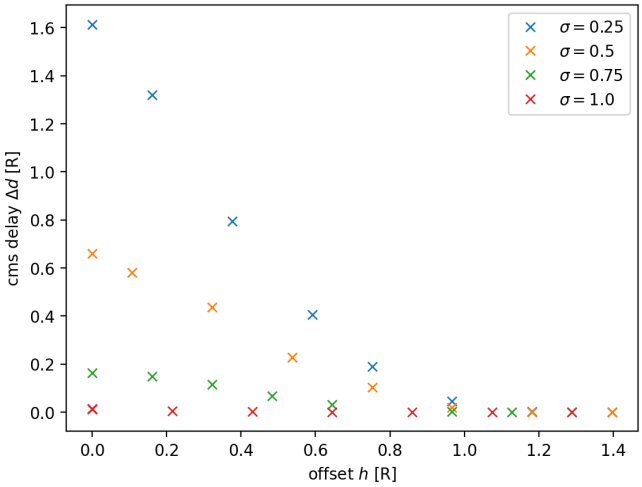


Figure 19: Center of mass delay for three-particle collision.

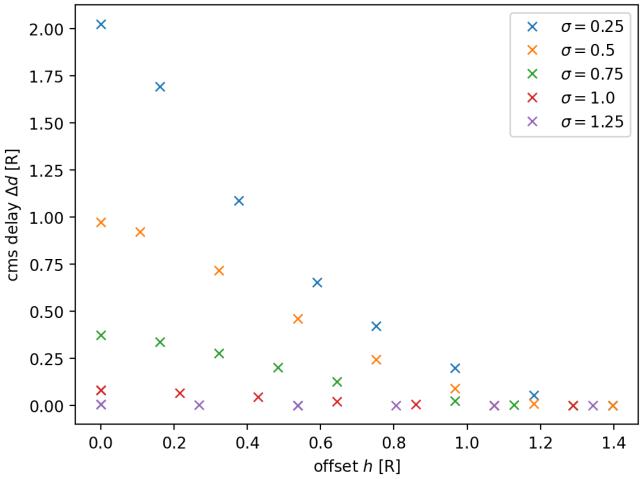


Figure 20: Center of mass delay for four-particle collision.

5.2. Convolutional representation

Further, we want to analyse whether there are different contributions to the delay. For that purpose, the final density distribution $\rho_i(\boldsymbol{r}, t_{\text{final}}) =: \rho_{i,\text{final}}(\boldsymbol{r})$ of the two-particle collision shall be written as a convolution of the initial density $\rho_i(\boldsymbol{r}, 0) =: \rho_{i,0}(\boldsymbol{r})$ with the delay kernel $\kappa(\boldsymbol{r})$:

$$\rho_{i,\text{final}}(\boldsymbol{r}) = (\kappa * \rho_{i,0})(\boldsymbol{r}). \tag{76}$$

We can try to write this in terms of Fourier transforms by using the convolution theorem (stated in 47)

$$\mathcal{F}[\rho_{i,\text{final}}](\boldsymbol{k}) = \mathcal{F}[\kappa](\boldsymbol{k}) \cdot \mathcal{F}[\rho_{i,0}](\boldsymbol{k}). \tag{77}$$

Solving for $\mathcal{F}[\kappa](\boldsymbol{k})$ reveals that this is only well defined if $\mathcal{F}[\rho_{i,0}](\boldsymbol{k}) \neq 0 \forall \boldsymbol{k}$; all Fourier modes are non-zero. In principle, this is the case for Gaussian peaks, due to the fact that their Fourier transform is also a Gaussian. However, high $\boldsymbol{k}$ modes decay exponentially. If the interaction term amplifies these high frequency components during the collision process, we get extremely large $\mathcal{F}[\rho_{i,\text{final}}](\boldsymbol{k})/\mathcal{F}[\rho_{i,0}](\boldsymbol{k})$, which results in a wildly oscillating kernel

$$\kappa(\boldsymbol{r}) = \mathcal{F}^{-1}\left[\frac{\mathcal{F}[\rho_{i,final}]}{\mathcal{F}[\rho_{i,0}]}\right](\boldsymbol{r}). \tag{78}$$

Consequently, a different approach is necessary for the kernel to be interpretable.

If we are only interested in the delay in movement (x-) direction, we can integrate out the orthogonal (y-) direction of the density field, yielding $\rho_i(x, t)$. Additionally, the final partial densities are shifted in the x-direction by the free movement distance d: $\rho_{i,\text{final}}(x - u_{i,x}d)$. Therefore, if the collision had no effect, the initial and shifted final densities would be equal. The initial and shifted partial densities are symmetrical, so one partial density was mirrored before they were averaged, written as $\bar{\rho}_{0/\text{final}}(x)$. The density distributions also depend on the simulation parameters h and σ; therefore, $\kappa(x) = \kappa(x, h, \sigma)$ might depend on them too. To analyse how $\bar{\rho}_{\text{final},\sigma}(x, h)$ depends on

the offset for a given peak width, we aim to calculate the 2-dimensional kernel

$$\kappa_\sigma(x,h) = \mathcal{F}^{-1}\left[\frac{\mathcal{F}[\bar{\rho}_{\text{final},\sigma}(x,h)]}{\mathcal{F}[\bar{\rho}_{0,\sigma}(x,h)]}\right](x,h). \tag{79}$$

Here, only the x-dimension is periodic.

This is where free function fitting comes into play. There, one starts with an initial guess for the kernel and iteratively updates it with a Markov chain Monte-Carlo simulation until a steady state is found. As the optimisation parameters $\boldsymbol{\theta}$, the value of the kernel at every (x,h) was used. Both variables are required to be equally spaced for it to be possible to calculate the convolution via Fourier transforms. To reduce the number of free parameters, a lower resolution than that for the simulation itself of $4\Delta x$ was chosen. For the offset, simulation data from $h = 0$ to $h = 1$ in steps of $\Delta h = 0.05$ was utilised. We define the predicted final density as

$$\bar{\rho}_{\text{pred}}(x,h;\boldsymbol{\theta}) = (\kappa_{\boldsymbol{\theta}} * \bar{\rho}_0)(x,h), \tag{80}$$

where the index σ has been omitted for now. Then, the Likelihood $\mathcal{L}(\boldsymbol{\theta})$ can be used to measure how well the prediction fits the data. If we assume that the simulation results are Gaussian distributed with variance v, we find

$$\log \mathcal{L}(\boldsymbol{\theta}) = -\frac{1}{2v}\sum_{x,h}\left(\bar{\rho}_{\text{final},\sigma}(x,h) - \bar{\rho}_{\text{pred}}(x,h;\boldsymbol{\theta})^2\right) + \log\left(\frac{1}{\sqrt{2\pi}v}\right), \tag{81}$$

where log is the natural logarithm [35]. Furthermore, we can use the knowledge we have about the kernel: it must be normalised in x to be mass conserving, and it should be positive because it represents a mass redistribution. Additionally, it should be smooth in both x and h to avoid encountering the same high frequency oscillations mentioned earlier and to allow for interpretability. This can be encoded into a prior distribution $p(\boldsymbol{\theta})$ that measures how well the kernel matches the expectations. Each constraint adds another term to the logarithm of the prior that measures the deviation from the expectation:

$$\begin{aligned}
\log p_{\text{norm}}(\boldsymbol{\theta}) &= -\lambda_{\text{norm}}\sum_h\left(\sum_x \kappa_{\boldsymbol{\theta}}(x,h) - 1\right)^2 \\
\log p_{\text{pos}}(\boldsymbol{\theta}) &= -\lambda_{\text{pos}}\sum_{x,h}\min\left(\kappa_{\boldsymbol{\theta}}(x,h), 0\right)^2 \\
\log p_{\text{smooth,x}}(\boldsymbol{\theta}) &= -\lambda_x\sum_{x,h}\left(\kappa_{\boldsymbol{\theta}}(x,h) - \kappa_{\boldsymbol{\theta}}(x-\Delta x', h)\right)^2 \\
\log p_{\text{smooth,h}}(\boldsymbol{\theta}) &= -\lambda_h\sum_{x,h}\left(\kappa_{\boldsymbol{\theta}}(x,h) - \kappa_{\boldsymbol{\theta}}(x, h-\Delta h)\right)^2,
\end{aligned}$$

where the sums run over all valid grid points. The higher the deviation, the less likely the respective kernel is. This results in the total log-prior

$$\log p(\boldsymbol{\theta}) = \log p_{\text{norm}}(\boldsymbol{\theta}) + \log p_{\text{pos}}(\boldsymbol{\theta}) + \log p_{\text{smooth,x}}(\boldsymbol{\theta}) + \log p_{\text{smooth,h}}(\boldsymbol{\theta}). \tag{82}$$

We now want to know the probability $p(\boldsymbol{\theta}|\text{data})$ for a specific kernel, given the simulation result (data). By using Bayes' theorem, we can write this as

$$p(\boldsymbol{\theta}|\text{data}) = \frac{p(\text{data}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta})}{p(\text{data})}. \tag{83}$$

The simulation method is deterministic, so $p(\text{data})$ is unknown. However, it is not needed if we want to maximise the posterior distribution $p(\boldsymbol{\theta}|\text{data})$ or its logarithm

$$\log p(\boldsymbol{\theta}|\text{data}) = \log \mathcal{L}(\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) + \text{const.}, \tag{84}$$

where $p(\text{data}|\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta})$ was used. The maximum cannot be easily computed, so another approach is taken: A Markov chain is created by repeatedly sampling from the posterior. Due to the fact that the probability of the target kernel is largest, we should slowly converge towards it. After the chain converged, the expected kernel can be calculated as the average over a number of samples. For the purpose of creating the Markov chain, the Metropolis-Hastings algorithm is used [36, 37]. Here, the parameter space is explored while maintaining detailed balance: The relation of the transition rates between two states $T(\boldsymbol{\theta} \to \boldsymbol{\theta}')$ is given by the relation of the probabilities of the target states

$$\frac{T(\boldsymbol{\theta} \to \boldsymbol{\theta}')}{T(\boldsymbol{\theta}' \to \boldsymbol{\theta})} = \frac{p(\boldsymbol{\theta}')}{p(\boldsymbol{\theta})}. \tag{85}$$

This ensures that the stationary state of the Markov chain is the target kernel. The steps of the algorithm are as follows:

1. Choose initial parameters. Here, for each offset h, a normal distribution around $\mu = 0$ with a width of $\sigma = 2\sqrt{2t_{\text{final}}}$ is used. The width corresponds to two times the expected diffusion. Consequently, in the absence of an effect from the collision, we expect a sharpening of the peak. For a strong effect of the collision, a translation and reshaping of the kernel is expected.

2. Propose a new kernel from a sample distribution. Here, we use $\boldsymbol{\theta}' = \boldsymbol{\theta} + \boldsymbol{\eta}$, where $\boldsymbol{\eta}$ is Gaussian noise with mean $\mu = 0$ and variance σ_η^2.

3. A step is accepted with the probability $\min(1, r)$, where $r = \frac{p(\boldsymbol{\theta}'|\text{data})}{p(\boldsymbol{\theta}|\text{data})}$ is the Metropolis–Hastings acceptance ratio. If rejected, the current kernel is kept.

4. Repeat this procedure from step 2 onwards to create a Markov chain of kernel samples.

For practical implementation, the emcee python library was used [38]. During each run, the variance of the noise was reduced when hitting a plateau and a low acceptance rate ($< 10\%$). The number of steps for each σ_η^2, as well as the weighting factors for each term in the prior, are shown in figure 6. The prior weights were chosen such that all terms of the log-prior contribute similarly. The global prior scaling ensures $\log p(\boldsymbol{\theta})$ and $\log \mathcal{L}(\boldsymbol{\theta})$ are of the same order of magnitude as soon as the data is fitted sufficiently.

phase	steps	σ_η^2
1	500 000	10^{-8}
2	1 000 000	10^{-10}
3	2 000 000	10^{-12}

prior weight	value $[10^5]$
λ_{norm}	40
λ_{pos}	200
λ_x	20
λ_h	1

Table 6: Parameters for running the Markov chain Monte Carlo simulation.

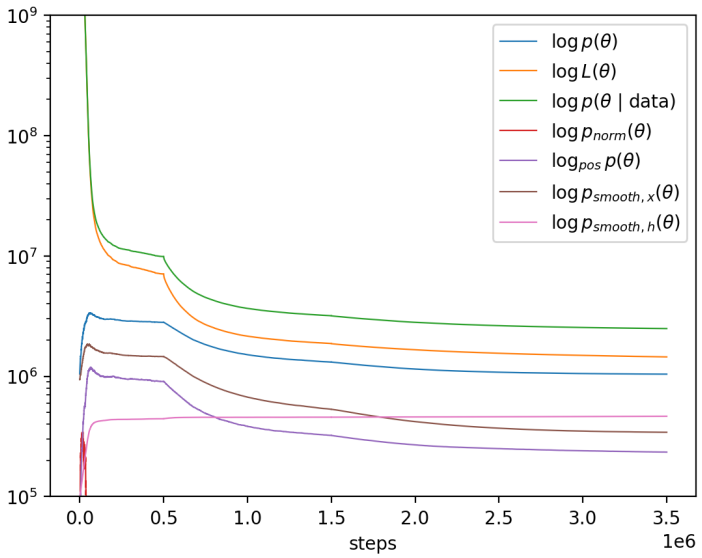


Figure 21: Contributions to the posterior. Chain run for a peak width of $\sigma = 0.25$. Parameters can be found in tab. 6

Otherwise, the prior would have no or too much of an effect. An example plot of the different contributions to the posterior $p(\boldsymbol{\theta}|\text{data})$ can be seen in figure 21.

For the collision of peaks with $\sigma = 0.5$, we see in figure 22 that the initial density profile is independent of h because we integrated over the respective direction. However, in the final density profile, we would expect the line to broaden (due to diffusion) but remain at the same position due to the shift if there was no collision. Here, the effect of the collision is clearly visible. The kernel, averaged over 1000 randomly chosen samples within the last 500000 steps, is shown in figure 23 together with the difference between the predicted and actual final density profile. The deviation of the prediction is very small when compared to the magnitude of the density profiles in fig. 22. Hence, a suitable kernel has been found. From the kernel, we can directly read off the delay: For $h > 0.75$, there is no delay; the finite width simply corresponds to the diffusion. However, for smaller h, a second line at ≈ -1 is formed. With decreasing h, the original line at $x = 0$ is smeared, while the contribution of the new line increases. This corresponds to a smaller amount of density being able to move around the opposing particle unhindered. At the same time, more density is trapped in front of the opposing particle and is only able to continue moving when enough density has moved at the flanks. It is surprising

that the second line is at a constant x, while the delay increases with lower h The expected increase results entirely from the changing contribution of the two lines.

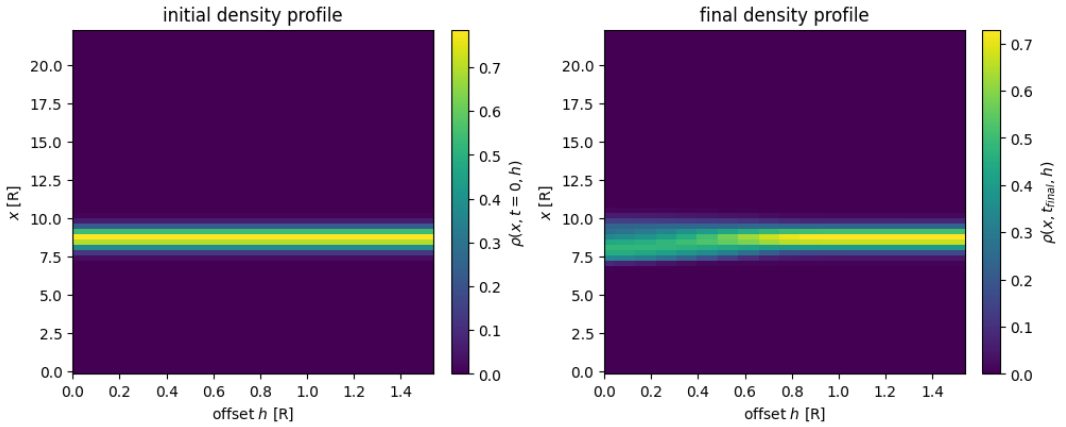


Figure 22: Initial and final density profiles for $\sigma = 0.5$

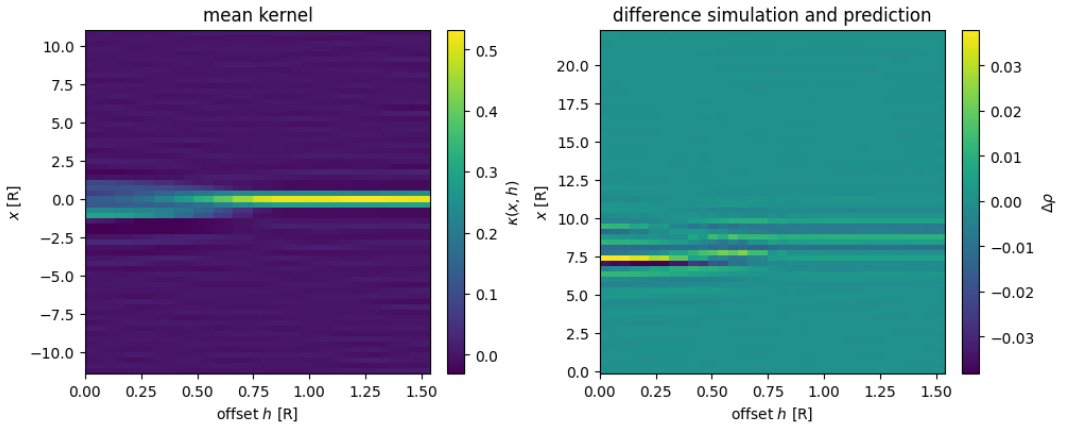


Figure 23: Averaged kernel and difference between final and predicted density profiles for $\sigma = 0.5$

This, however, changes in the $\sigma = 0.25$ case. Here, both lines are still present; however, not at constant x, but they follow a roughly linear dependence on h for $h < 0.8$. Consequently, both the delay of the density moving around the other particle as well as the delay of the trapped density increase with a reduction in the offset. The quality of the obtained kernel is even better than that for $\sigma = 0.5$, as can be concluded by comparing the difference to the density profiles in figure 24.

Consequently, the explicit simulation of the collision process to obtain the post-collision density profile can be substituted by the use of a translation and the convolution with an appropriate kernel. The translation originates from self-propulsion, while the kernel captures the effects of both diffusion and interaction. Consequently, the density

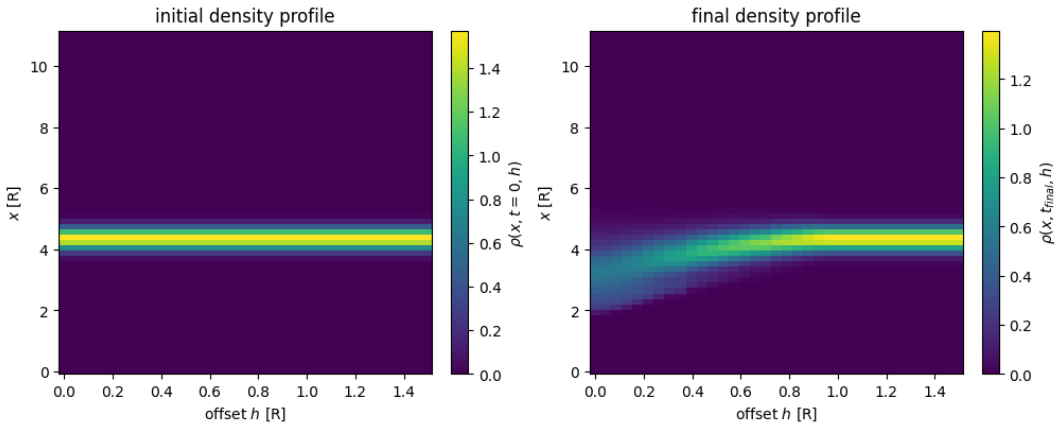


Figure 24: Initial and final density profiles for $\sigma = 0.25$

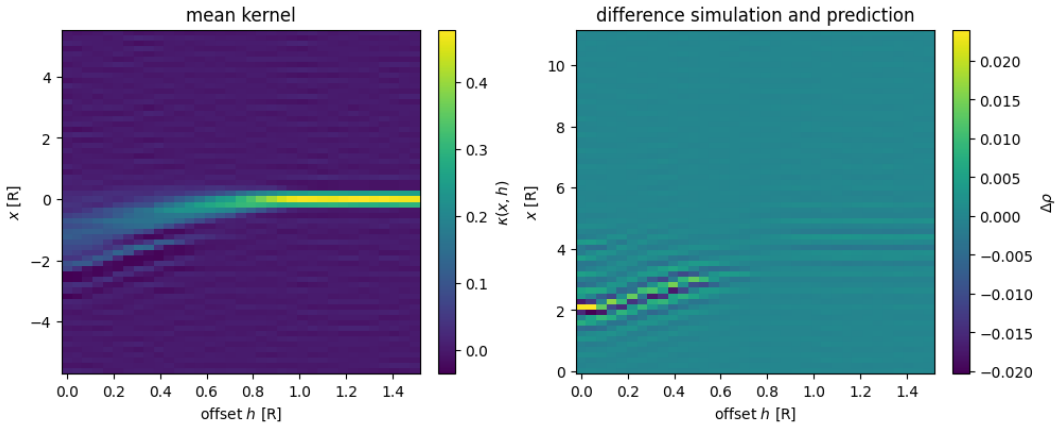


Figure 25: Averaged kernel and difference between final and predicted density profiles for $\sigma = 0.25$

profile after the collision can be written as

$$\rho_{i,\text{final}}(x, h) = (\kappa * \rho_{i,0})(x - u_{i,x} v_0 t_{\text{final}}, h), \tag{86}$$

where $u_{i,x}$ is the x-component of $\boldsymbol{u}$. The expression remains σ-dependent, despite not explicitly stated.

6. Conclusion

To summarise, we demonstrated a way to simulate the collision of active Brownian hard discs by means of DDFT. Each particle was represented by a Gaussian density peak, incorporating uncertainty in the position. The use of a total orientation field via motion tracking of density peaks did not prove successful. However, the use of per particle partial density fields yielded good results, as shown in section 4. Furthermore, it was demonstrated that the self propulsion velocity v_0 has no significant effect on the density after the collision, apart from the occurring diffusion. Hence, the positional delay Δd of the center of mass is close to independent of v_0. Therefore, the time delay scales like $\Delta t v_0^{-1}$.

Additionally, the effect of the Gaussian peak width σ and the offset h was investigated in 5. For the positional and, analogously, the temporal center of mass delay, nonlinear relationships with σ and h were observed (see sec. 5.1. The delay increases faster than σ^{-1} and the dependence on the offset h looks roughly Gaussian. This also holds true for the investigated three- and four-particle collisions, merely the absolute value of the delay increases.

Finally, it was discussed how the simulation of the collision process of two particles can be substituted with an equivalent operation consisting of a translation and a convolution with an appropriate kernel. At least when only the density profiles in the forward direction are considered.

While we were able to simulate and analyse collisions of a few particles, there are still open questions about how an implementation for larger systems can be realised and what implications this yields for the occurrence of MIPS.

7. Outlook

There is further research needed, both regarding the numerical implementation and the data analysis. The limit for the number of particles simulated is currently in the order of 10, due to the fact that each requires the solving of its dynamical equation for the partial density. An example for 16 particles can be seen in figure 26. One can improve this in multiple ways. First, one can change the normalisation convention of the discrete Fourier transform to slightly improve the performance of the convolutions needed for calculating the excess free energy term. However, this does not scale with the particle number because the excess term is computed only once for the total density. Furthermore, both the Fourier transform and the evaluation of the dynamical equations can be parallelised. The first yields a significant speed-up when considering larger boxes than 1000×1000 sites. Additionally, when a large number of particles ($N > 100$) is simulated, one can coarse-grain the particle orientation, allowing the use of one density for all particles of similar orientation. This limits the maximum number of necessary dynamical equations, depending on the desired resolution in the orientation. On top of that, one should implement the advection term implicitly, allowing larger time steps. An additional improvement on the time step might also result from performing a stability analysis on the excess free energy term and, consequently, find a more efficient way to deal with the Gibbs ripples (or avoid them completely). This becomes even more important for large systems, where collisions occur somewhere at all times.

Regarding the delay caused by collisions, one can consider many more interesting configurations. All configurations investigated in this work are rotation-symmetric with respect to the center of the simulation box. However, in the two-particle case, it might also be of interest to analyse collisions with non anti-parallel orientation. For three and four particles, one can also consider mirror-symmetric configurations, for example, with an isosceles triangle or rhombus as the base shape.

Further investigations could focus on the precise dependence of the center of mass delay on the relevant parameters. The time delay is expected to scale with the self-propulsion velocity in a manner slightly different from $\propto v_0^{-1}$. It would also be interesting to analyse collisions for even smaller peak widths σ to determine if the faster-than-linear increase of the delay with σ^{-1} continues. To deduce a functional form $\Delta t(v_0, \sigma, h)$, the specific dependence on σ must be determined. Additional work is needed to connect this to the waiting time in Schieck's thesis [10].

A further goal could be to replace the fundamental measure term in the dynamical equation with an effective convolution kernel, thereby enabling large-scale, spatially coarse-grained simulations. Section 5.2 demonstrated that the resulting density profile after a single collision can indeed be represented as a convolution. The next step would be to derive a two-dimensional, density-dependent version of such a kernel, allowing the DDFT framework to capture collision-induced delays without explicitly resolving individual particle collisions.

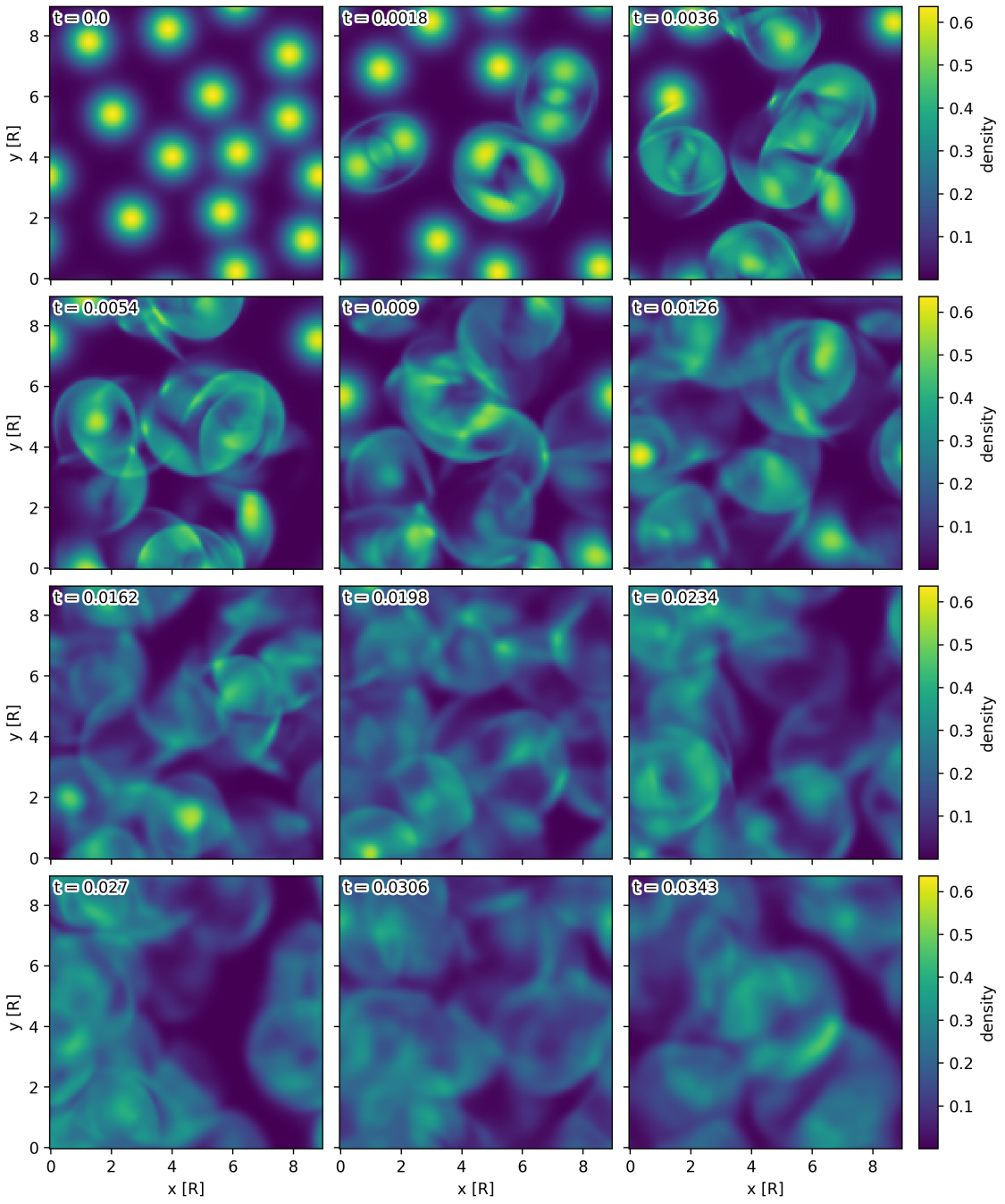


Figure 26: Simulation of 16 particles with randomly initialised positions and orientations for $\sigma = 0.5$, $v_0 = 512$, and $\rho_{\text{bg}} = 0.9\ \rho_{\text{bg,max}}(\sigma = 0.25)$ in a box with $L = 9$ and $N_x = N_y = 128$

References

[1] Robert Brown. Xxvii. a brief account of microscopical observations made in the months of june, july and august 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies. *The philosophical magazine*, 4(21):161–173, 1828.

[2] Albert Einstein. On the motion of small particles suspended in liquids at rest required by the molecular-kinetic theory of heat. *Annalen der physik*, 17(549-560):208, 1905.

[3] Jean Perrin. Mouvement brownien et réalité moléculaire. *Annales de Chimie et de Physique*, 1909.

[4] Sriram Ramaswamy. The mechanics and statistics of active matter. *Annu. Rev. Condens. Matter Phys.*, 1(1):323–345, 2010.

[5] Clemens Bechinger, Roberto Di Leonardo, Hartmut Löwen, Charles Reichhardt, Giorgio Volpe, and Giovanni Volpe. Active particles in complex and crowded environments. *Reviews of modern physics*, 88(4):045006, 2016.

[6] Yaouen Fily and M Cristina Marchetti. Athermal phase separation of self-propelled particles with no alignment. *Physical review letters*, 108(23):235702, 2012.

[7] Gabriel S Redner, Michael F Hagan, and Aparna Baskaran. Structure and dynamics of a phase-separating active colloidal fluid. *Biophysical Journal*, 104(2):640a, 2013.

[8] Ivo Buttinoni, Julian Bialké, Felix Kümmel, Hartmut Löwen, Clemens Bechinger, and Thomas Speck. Dynamical clustering and phase separation in suspensions of self-propelled colloidal particles. *Physical review letters*, 110(23):238301, 2013.

[9] Michael E Cates and Julien Tailleur. Motility-induced phase separation. *Annu. Rev. Condens. Matter Phys.*, 6(1):219–244, 2015.

[10] Emanuel Schieck. Clustering of penetrable, athermal active particles that wait on collisions. Bachelor thesis, 2024.

[11] Andreas M Menzel and Hartmut Löwen. Traveling and resting crystals in active systems. *Physical review letters*, 110(5):055702, 2013.

[12] Andreas M Menzel, Arnab Saha, Christian Hoell, and Hartmut Löwen. Dynamical density functional theory for microswimmers. *The Journal of chemical physics*, 144(2), 2016.

[13] Dominic Arold and Michael Schmiedeberg. Mean field approach of dynamical pattern formation in underdamped active matter with short-ranged alignment and distant anti-alignment interactions. *J. Phys.: Condens. Matter*, 32(315403), 2020.

[14] Dominic Arold and Michael Schmiedeberg. Active phase field crystal systems with inertial delay and underdamped dynamics. *The European Physical Journal E*, 43(7):47, 2020.

[15] Umberto Marini Bettolo Marconi and Pedro Tarazona. Dynamic density functional theory of fluids. *Journal of Physics: Condensed Matter*, 12(8A):A413, 2000.

[16] M Rex, HH Wensink, and H Löwen. Dynamical density functional theory for anisotropic colloidal particles. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 76(2):021403, 2007.

[17] Yaakov Rosenfeld. Free-energy model for the inhomogeneous hard-sphere fluid mixture and density-functional theory of freezing. *Physical review letters*, 63(9):980, 1989.

[18] Roland Roth, Robert Evans, A Lang, and G Kahl. Fundamental measure theory for hard-sphere mixtures revisited: the white bearversion. *Journal of Physics: Condensed Matter*, 14(46):12063, 2002.

[19] Hendrik Hansen-Goos and Roland Roth. Density functional theory for hard-sphere mixtures: the white bear version markii. *Journal of Physics: Condensed Matter*, 18(37):8413, 2006.

[20] P Tarazona. Density functional for hard sphere crystals: A fundamental measure approach. *Physical Review Letters*, 84(4):694, 2000.

[21] Roland Roth, Klaus Mecke, and Martin Oettel. Communication: Fundamental measure theory for hard disks: Fluid and solid. *The Journal of Chemical Physics*, 136(8), 2012.

[22] Roland Roth, Klaus Mecke, and Martin Oettel. Fundamental measure theory of hard-sphere mixtures: a review. *The Journal of Chemical Physics*, 136(8)(081101), 2012.

[23] Tim Neuhaus. Density functional theory for colloidal spheres in various external potentials. PhD thesis, 2013.

[24] Markus Hoffmann. Classical density functional theory for particles with hard cores and soft square shoulders. Bachelor thesis, 2019.

[25] Roland Roth. Introduction to density functional theory of classical systems: Theory and applications. Lecture notes, 2006.

[26] Roland Roth. Fundamental measure theory of hard-sphere mixtures: a review. *J. Phys.: Condens. Matter*, 22, 2010.

[27] Gaël Guennebaud, Benoît Jacob, et al. Eigen v3. `http://eigen.tuxfamily.org`, 2010.

[28] Matteo Frigo and Steven G. Johnson. The design and implementation of fftw3. *Proceedings of the IEEE*, 93(2):216–231, 2005.

[29] Randall J LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.

[30] Sigal Gottlieb and Chi-Wang Shu. Total variation diminishing runge-kutta schemes. *Mathematics of Computation*, 67, 1998.

[31] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM REVIEW*, 43, 2001.

[32] Bram van Leer. Towards the ultimate conservative difference scheme. *Journal of Computational Physics*, 23, 1977.

[33] Xiangxiong Zhang and Chi-Wang Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229, 2010.

[34] KW Cattermole. The fourier transform and its applications, 1965.

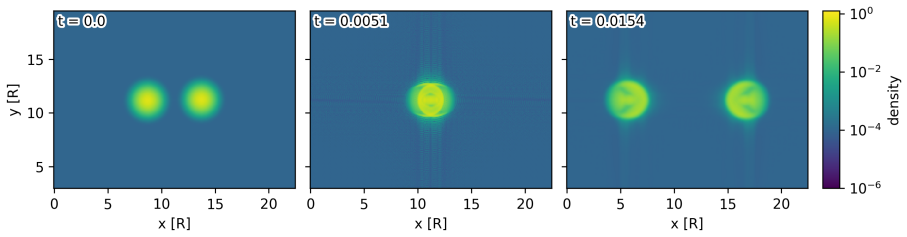
[35] Glen Cowan. *Statistical Data Analysis*. Oxford University Press, 1997.

[36] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

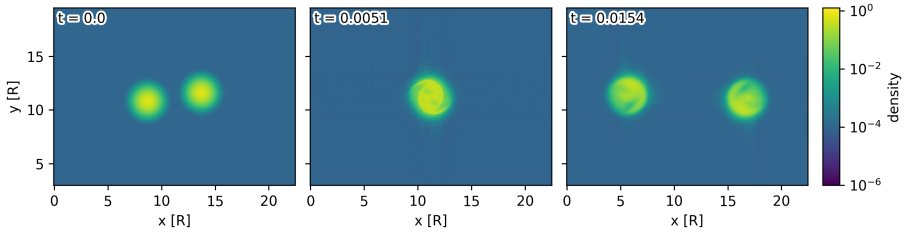
[37] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

[38] Daniel Foreman-Mackey, David W Hogg, Dustin Lang, and Jonathan Goodman. emcee: the mcmc hammer. *Publications of the Astronomical Society of the Pacific*, 125(925):306, 2013.

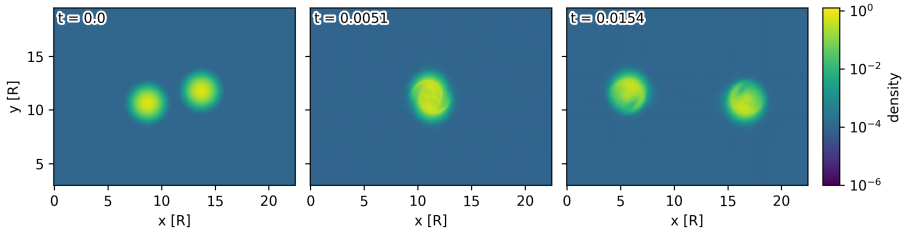
A. Influence of the offset on peaks of different width



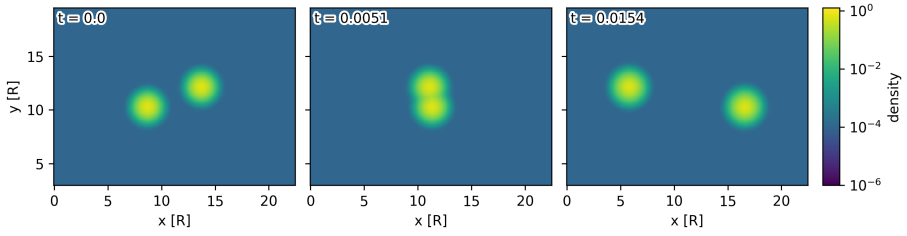
(a) $h = 0$.



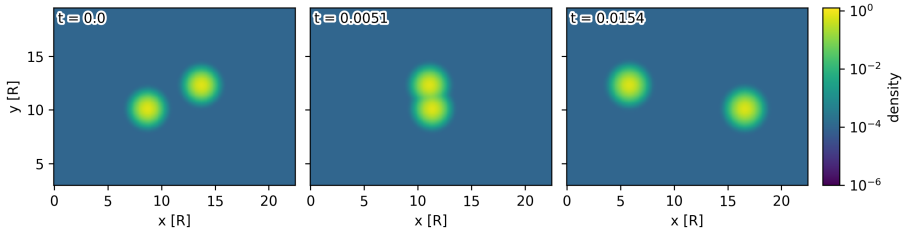
(b) $h \approx 0.35$.



(c) $h \approx 0.53$.

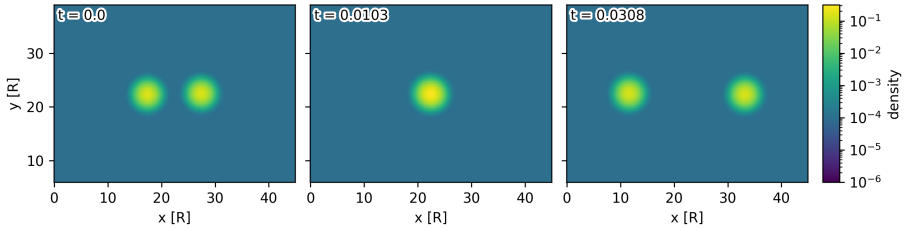


(d) $h \approx 0.88$.

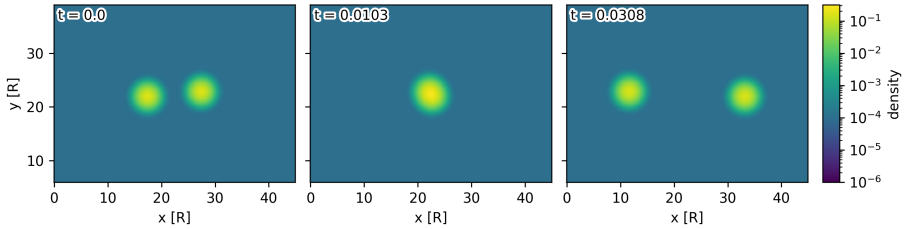


(e) $h \approx 1.05$.

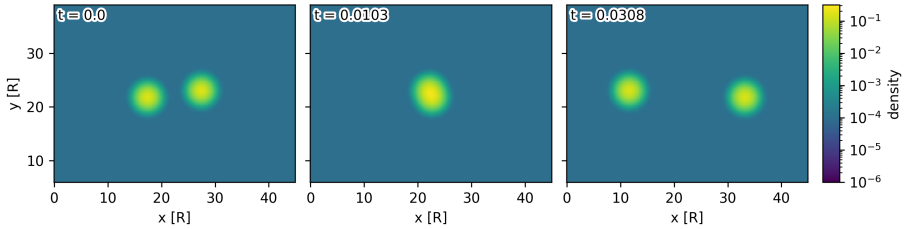
Figure 27: Two-particle collision for $\sigma = 0.5$ and varying h. Other Parameters in tab. 5.



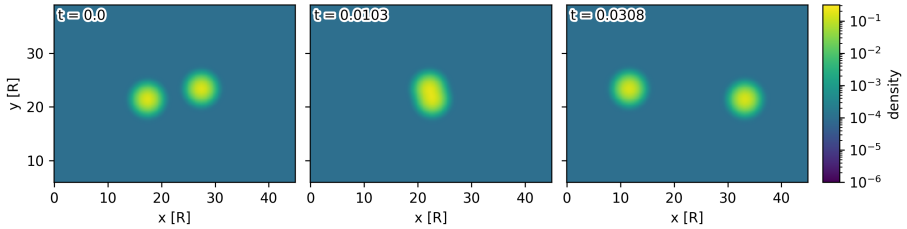
(a) $h = 0$.



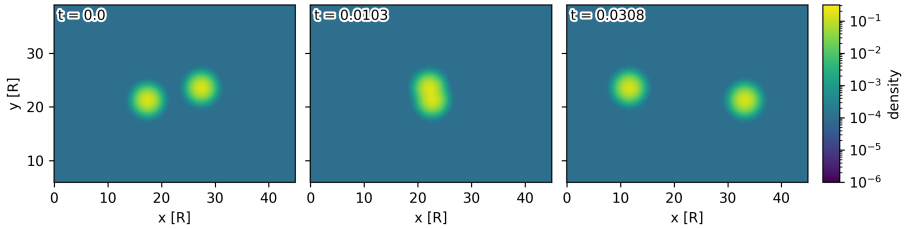
(b) $h \approx 0.35$.



(c) $h \approx 0.53$.

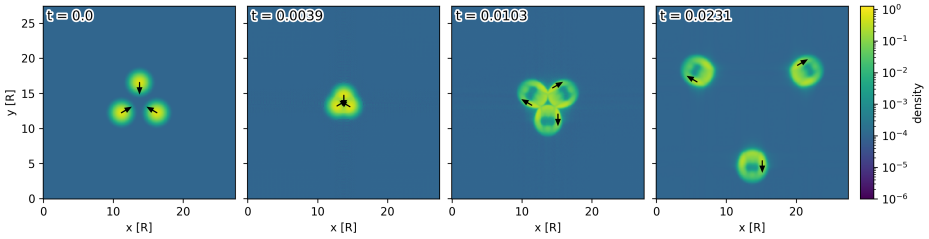


(d) $h \approx 0.88$.

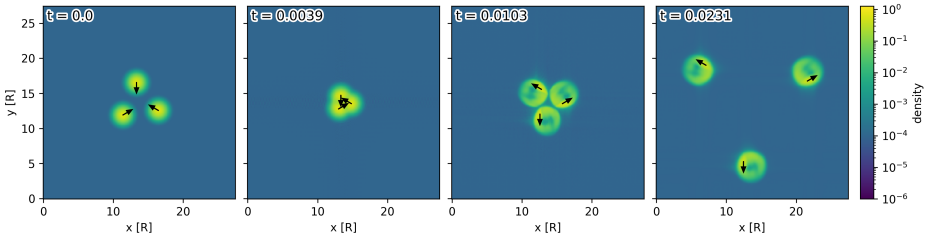


(e) $h \approx 1.05$.

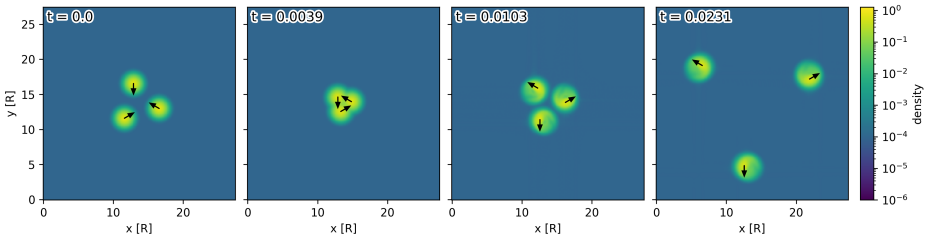
Figure 28: Two-particle collision for $\sigma = 1$ and varying h. Other Parameters in tab. 5.



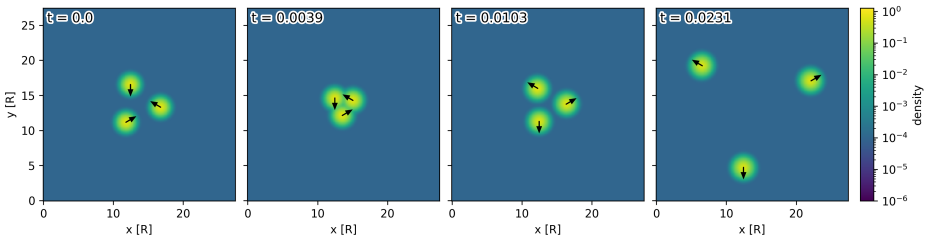
(a) $h = 0$.



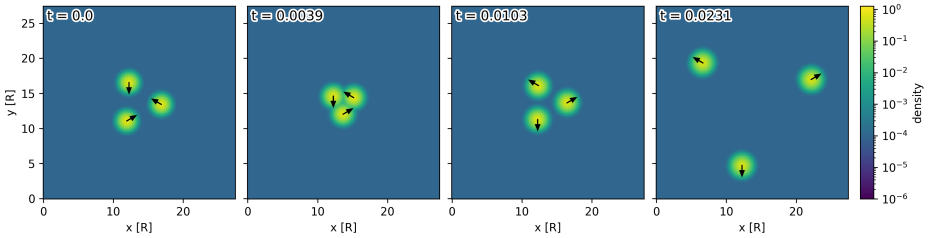
(b) $h \approx 0.32$.



(c) $h \approx 0.75$.

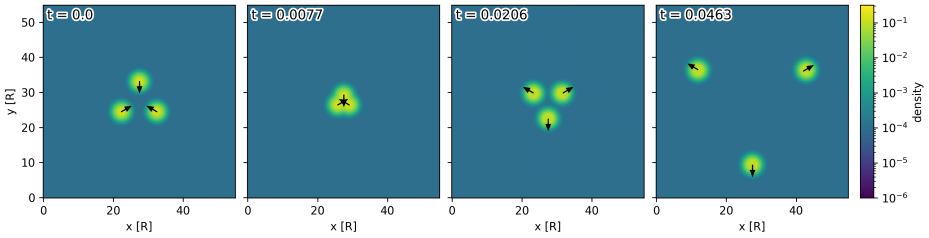


(d) $h \approx 1.18$.

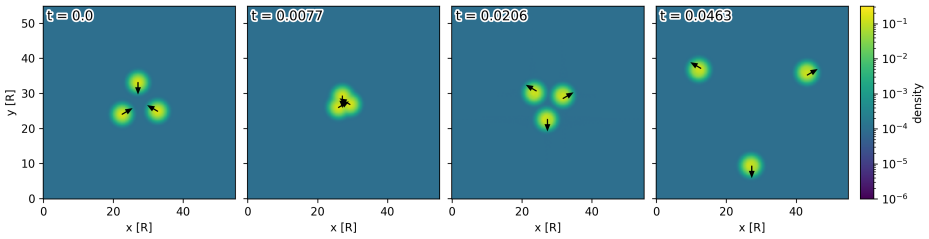


(e) $h \approx 1.4$.

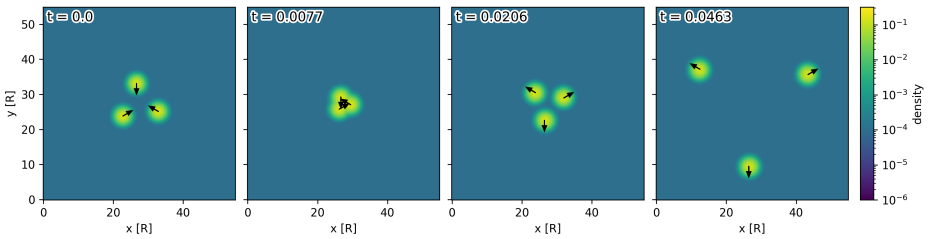
Figure 29: Three-particle collision for $\sigma = 0.5$ and varying h. Other parameters in tab. 5.



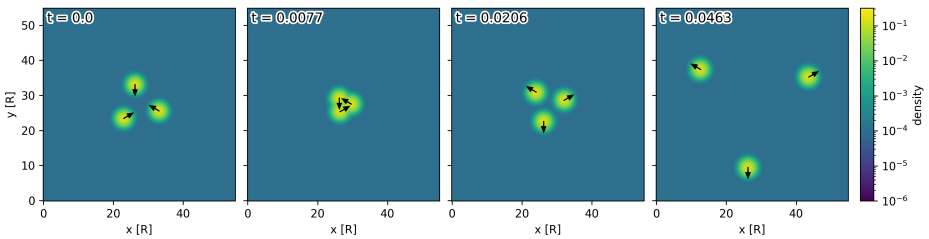
(a) $h = 0$.



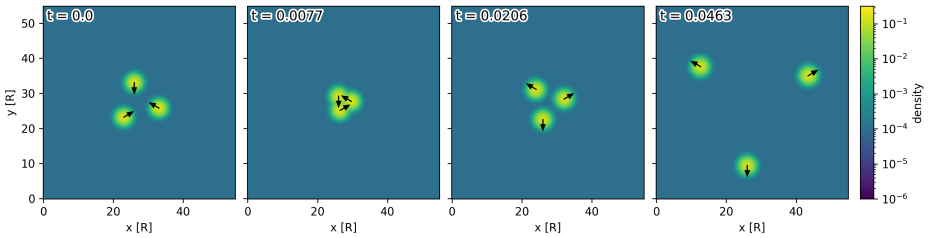
(b) $h \approx 0.21$.



(c) $h \approx 0.64$.

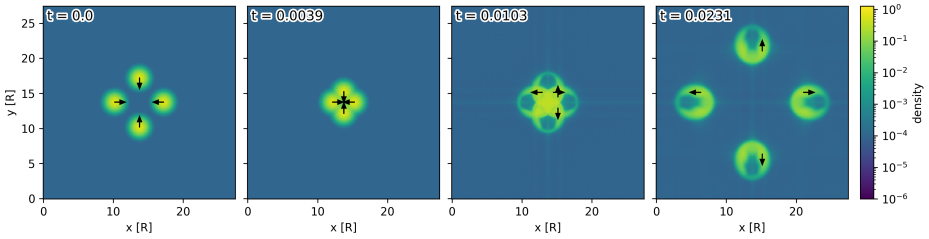


(d) $h \approx 1.07$.

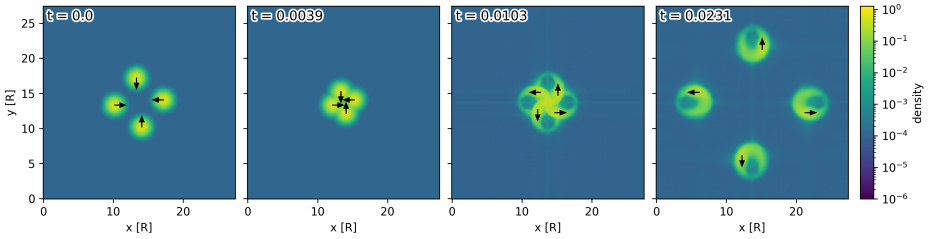


(e) $h \approx 1.29$.

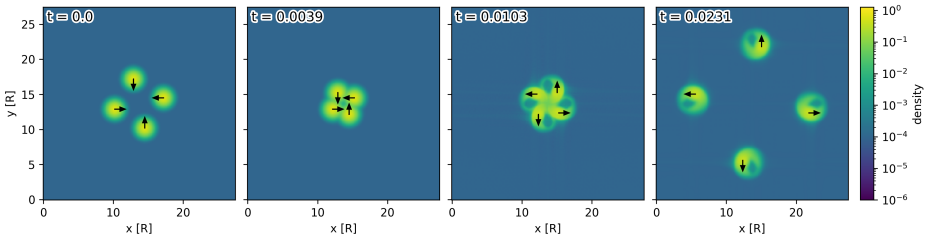
Figure 30: Three-particle collision for $\sigma = 1$ and varying h. Other parameters in tab. 5.



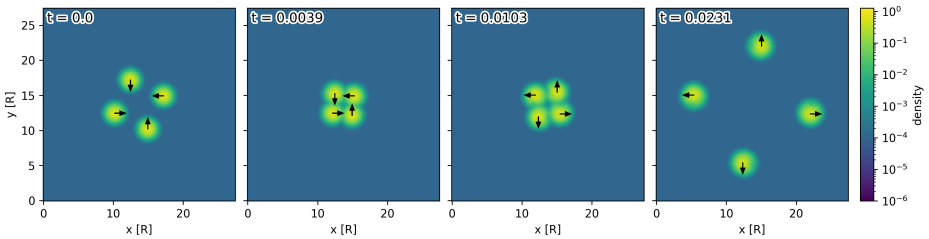
(a) $h = 0$.



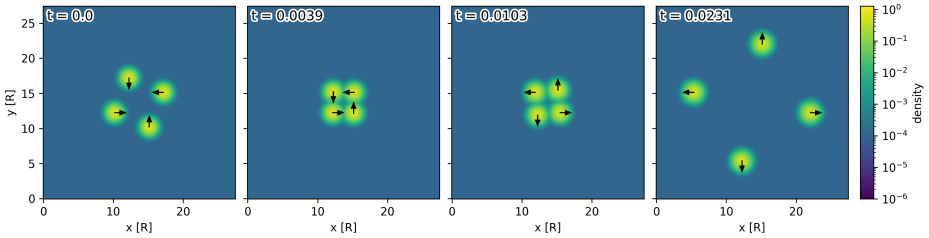
(b) $h \approx 0.32$.



(c) $h \approx 0.75$.

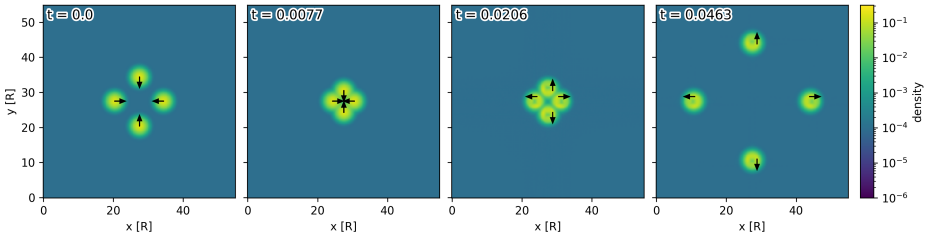


(d) $h \approx 1.18$.

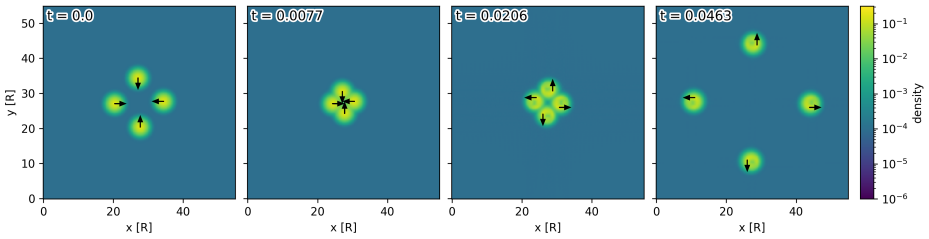


(e) $h \approx 1.4$.

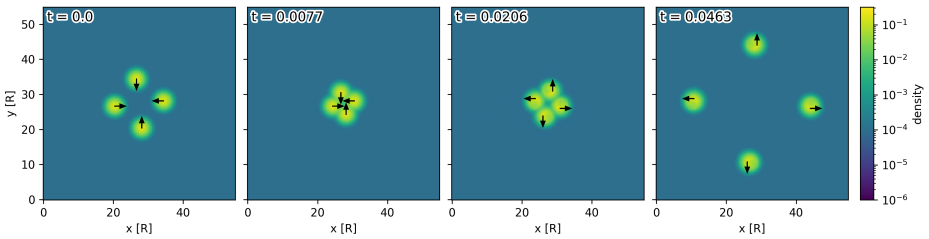
Figure 31: Four-particle collision for $\sigma = 0.5$ and varying h. Other parameters in tab. 5.



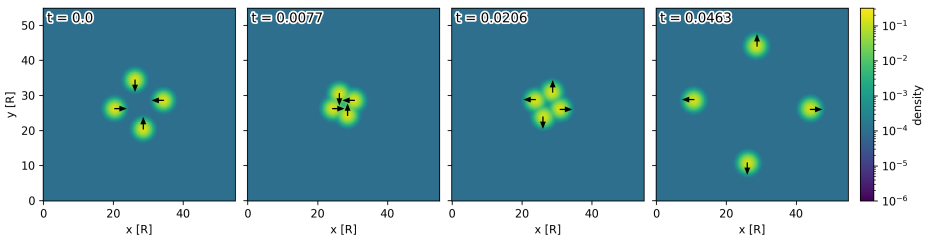
(a) $h = 0$.



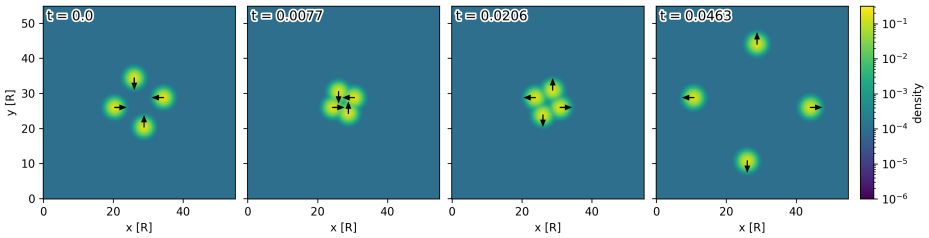
(b) $h \approx 0.21$.



(c) $h \approx 0.64$.



(d) $h \approx 1.07$.



(e) $h \approx 1.29$.

Figure 32: Four-particle collision for $\sigma = 1$ and varying h. Other parameters in tab. 5.

Declaration

I hereby declare that I alone wrote the submitted Master's thesis without illicit or improper assistance. No sources other than those listed in the bibliography were used, and all passages taken from these sources, in part or in full, have been marked as such. I have not submitted this Master's thesis to another institution, and it has never been used for other purposes or to fulfil other requirements, in part or in full.

Erlangen, Jonas Buba