GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

FAKULTÄT FÜR PHYSIK

MAX-PLANCK-GESELLSCHAFT

MASTER'S THESIS

Development, implementation and application of a Stochastic Rotation Dynamics algorithm for granular matter

Arne Wolf Zantop
`arne.zantop@icloud.com`

Advisor & First Referee: Dr. Marco G. Mazza
Second Referee: Prof. Dr. Stefan Klumpp
Due: September 12, 2017

Max-Planck-Institut für
Dynamik und Selbstorganisation

Abteilung Dynamik
Komplexer Fluide

NESM

Abstract

In this work we present an extension of the well-known particle based stochastic rotation dynamics method for the simulation of hydrodynamics of granular gases. We use an effective local coefficient of restitution to render energy dissipation dependent on local macroscopic observables, while locally conserving density and momentum. We derive the granular Boltzmann equation and demonstrate that our model obeys linear granular hydrodynamic equations. Furthermore, we derive a formula for the kinematic viscosity of the model fluid in two dimensions. We present results from simulations with a software implementation for general purpose graphics cards, that we successfully test and benchmarked with analytical predictions for standard stochastic rotation dynamics. For the granular system we observe that our prediction of the kinematic viscosity compares well with the results obtained from simulations. In this context we find that for low shear driving the fluid becomes unstable and develops shear bands. In the simulations of a freely cooling granular gas the temperature evolution follows the prediction of Haff's law over several orders of magnitude in both time and temperature. Furthermore, we observe clustering for lower coefficients of restitution. The emergence and dynamics of the cluster compare well with expectations based on theory, experiments and simulations. The clustering sets in as the global Mach number exceeds one. Subsequently, density fluctuations grow while we observe a change in the power law of the temperature evolution. The clusters exhibit a higher cooling rate than dilute regions, hence, density and temperature become anti-correlated. This locally leads to supersonic flow. After their emergence, clusters move, collide and thus grow further. The velocity distribution function compares well with theoretical predictions. The shape of the reduced velocity distribution function changes with time as predicted, and the evolution of the second Sonine coefficient qualitative matches with analytical predictions. In our discussion we provide criteria for the selection of model parameters, and identify the effects of the finite system size.

Contents

List of Figures

Symbols

α	SRD rotation angle
a	SRD lattice length
$\mathbf{a}$	vector
$\hat{\mathbf{a}}$	unit vector
$\dot{a}$, $\ddot{a}$	first and second time derivative of a
ϵ	coefficient of restitution
$\epsilon_{\xi,\text{eff}}$	GSRD effective coefficient of restitution
$f(\mathbf{v}, \mathbf{x}, t)$	velocity distribution function
m	SRD/GSRD particle mass
$\mathcal{M}$	Mach number
$\nu^{\pm}$	collision frequency
ν_{stream}	streaming viscosity
n, n_ξ	number of particles (in cell $\boldsymbol{\xi}$)
$\tilde{\omega}_\xi$	SRD rotation matrix of cell $\boldsymbol{\xi}$
$\tilde{\omega}_{\xi,\text{eff}}$	dissipative GSRD rotation matrix of cell $\boldsymbol{\xi}$
$\tilde{P}$	pressure tensor
ρ, ρ_ξ	number density
σ	particle cross section
θ_ξ	instantaneous temperature of cell $\boldsymbol{\xi}$
t	time
Δt	SRD/GSRD time step
T	temperature
$\mathcal{V}$	volume of SRD/GSRD cells
$\mathcal{V}_\xi$	set of particles in SRD/GSRD cell $\boldsymbol{\xi}$
$\mathbf{v}$	velocity vector $\mathbf{v} = (v_x, v_y, v_z)^T$
$\bar{\mathbf{v}}$	velocity in center of mass frame
$\mathbf{V}_\xi$	instantaneous center of mass velocity of cell $\boldsymbol{\xi}$
$\boldsymbol{\xi}$	SRD cell index
$\mathbf{x}$	position vector $\mathbf{x} = (x, y, z)^T$

Acronyms

2D two dimensions 25

3D three dimensions 26

CPU central processing unit 41

CUDA compute unified device architecture 41

GB gigabytes 17

GPGPU general-purpose computing on graphic processing unit 41

GPU graphic processing unit 17, 41

GSRD granular stochastic rotation dynamics 25

LHS left hand side 38

MPCD multi-partice collison dynamics 14, 22

RAM random access memory 17

RHS right hand side 36, 37

SRD stochastic rotation dynamics . . . . vi, 14, 22, 29

CHAPTER 1

Introduction

1.1 Motivation

It is often said that water is the only substance that appears on Earth in all three classical states of aggregate. If one, however, extends consideration to nonequilibrium systems one notices that granular materials, sand, for instance, also exhibits such variety of states. Starting with the first classical state of aggregation, sand is solid-like when lying in a desert. One can walk on it since it carries one's weight. In the form of a sandcastle at a beach this is even more obvious. Under different conditions sand also flows like a liquid, like for example in an hourglass, thus existing in the second classical state of aggregation. Lastly, the wind in a desert can also elevate the sand to the third classical state of aggregation, the gas state, and cause a sandstorm.

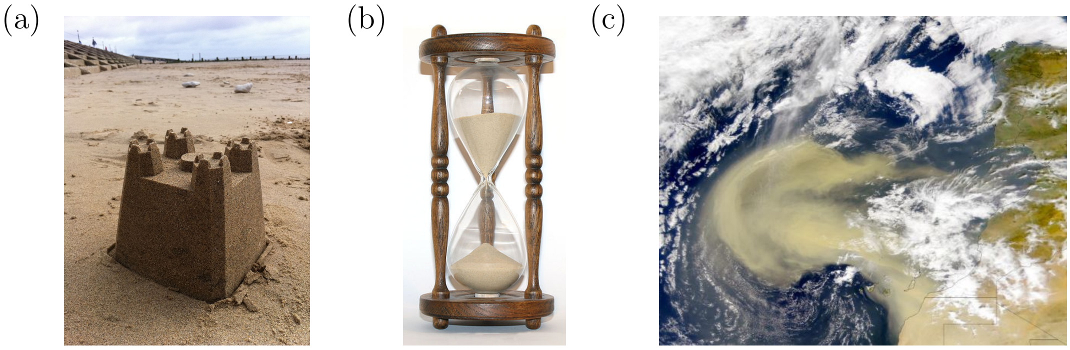


Figure 1.1: Sand as an example of granular matter in different states of aggregation. (a) Solid-like sand castle[1]. (b) Liquid sand flowing in an hourglass[2]. (c) Sand cloud of the size of Spain, blown from the Sahara desert over the Atlantic ocean[3].

The physics of granular matter plays a role in many situations at different length scales. Phenomena of granular matter occur not only in daily life but also in industrial applications and cutting edge research topics. Nearly all raw materials in the pharmaceutical, agricultural and food industries [4] are either powders and grains or solvents like water. Thus, understanding the physics of the granular states has direct repercussions on myriad practical applications.

Granular physics is extremely relevant also in astrophysical problems such as the formation of rocky planets in protoplanetary disks [5, 6, 7] or the dynamics of asteroid belts and planetary ring systems [8].

As we have learned from these examples, granular materials can consist of different materials. They have the common characteristics that instead of microscopic molecules, particles are macroscopic. Beside sand as the basic example, coffee beans, lentils and rice exhibit shape anisotropies. Experiments with artificial grains consider even more complex shapes [9].

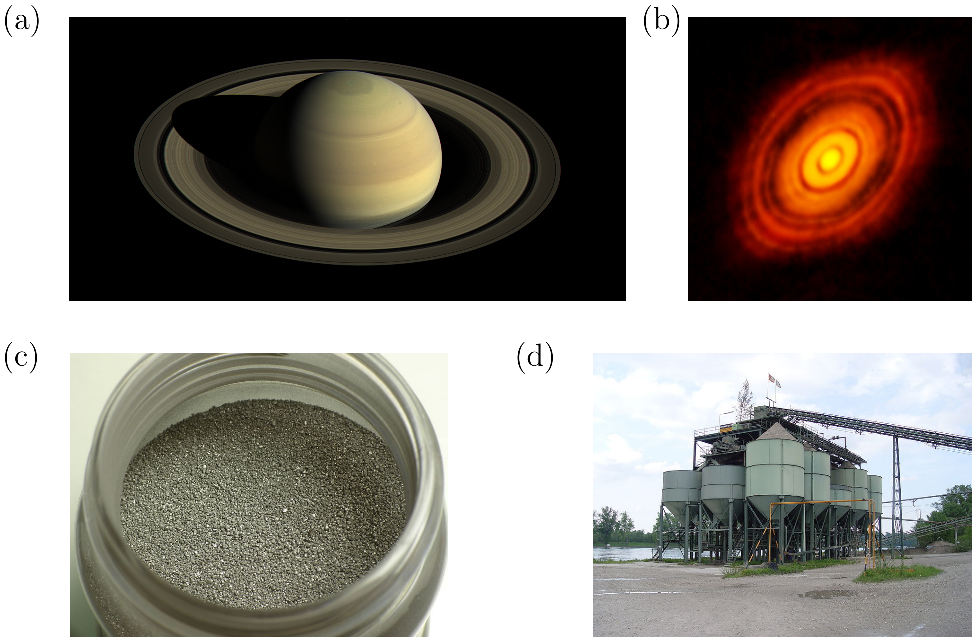


Figure 1.2: Granular matter in nature: In panel (a) we see Saturn and its ring system consisting of icy grains as seen by the Cassini space probe[1]. Panel (b) shows the protoplanetary disk around the HL-Tauri star pictured by the ALMA radio telescope[2]. Granular matter in industry: Panel (c) shows iron powder[3], one of many industrial goods that are processed from an initial powder form. Those and others are stored in silos (panel (d)[4]). In contrast to liquid tanks, the pressure of a granular matter is described by the Janssen equation and thus allows for different states [1].

---

[1] (cc), https://commons.wikimedia.org/wiki/File:Sandcastle1.jpg (M. Harrigan).

[2] (cc), https://commons.wikimedia.org/wiki/File:Wooden_hourglass_3.jpg (User:S Sepp).

[3] (cc), https://apod.nasa.gov/apod/ap000303.html (NASA).

1.2 Granular matter - a phenomenological point of view

In order to approach the phenomena of granular physics, we first ask what are the salient features of the microscopic dynamics. On the microscopic scale matter consists of atoms or molecules. The corresponding basic elements of granular matter are grains, after which the former is actually named.

Although these are fundamentally different from atoms or molecules, we have already recognized that they form the states which share some statistical features of equilibrium, molecular systems. In the simplest case of a solid, grains are packed just as molecules and do not move. In contrast to molecules, there is no long range interaction between grains, so that a sandcastle may break down if shaken. The solid is not of interest in this work and hence we will not consider this state further.

If we consider *gedankenexperiments* of granular liquids or gases, we can quickly realize that granular fluids occur in situations where there is a source of external energy, a driving. Sand grains flowing down a dune are driven by gravity, a sandstorm by wind. When the driving stops, the grains come to rest after some time.

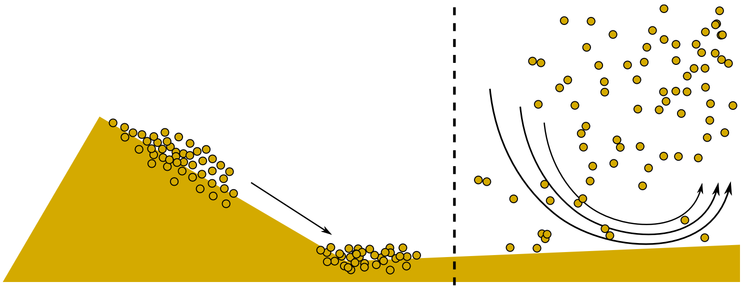


Figure 1.3: Granular liquid (left) and gas (right), set in motion by external driving such as gravity and wind respectively.

The reason for this is that the grains lose energy as they collide. Unlike molecules, the grains are complex macroscopic particles. Upon collision, part of the impact energy is transferred into internal degrees of freedom of the grain itself and eventually is radiated away as electromagnetic energy. Instead of elastic collisions among molecules, grains collide inelastically, and, in a real system, the grains will slightly heat up. This energy transferred into the internal degrees of freedom, however, does not play a role in the observable dynamics any more. In that sense granular matter is a thermodynamically open system and generally out of equilibrium.

A remark is necessary at this point. Although we have said that there are no long range interactions between grains, this statement is not entirely true. Sand

---

[1] (cc), https://saturn.jpl.nasa.gov/resources/7504/?category=images (NASA).

[2] (cc), http://www.eso.org/public/archives/images/large/eso1436a.jpg (ALMA).

[3] (cc), https://commons.wikimedia.org/wiki/File:Iron_powder.JPG (Anonimski).

[4] (cc), https://commons.wikimedia.org/wiki/File:Kieswerk_II.JPG (M. Dürrschnabel).

particles in a sandstorm are known to charge upon collisions and sandstorms are usually accompanied by electrical discharges and lightnings. This field experiences active research, where the nature of the charging and its implications are under investigation. In the following, we will however ignore any charge exchange mechanism among grains. In the next section we will have a detailed look at the nature of the collisions between grains and their implications.

1.2.1 Granular collisions

The kinetic theory of Boltzmann and Enskog for dilute gases can be expanded to the dynamics of granular gases. Like the original, it relates the change of the macroscopic velocity distribution function to the properties of the collisions of pairs of particles. To begin, we consider a pair of granular particles and take a closer look at the details of such a collision.

Two particles i and j of identical size and mass move through space with velocities v_i and v_j, respectively. We may first consider this situation as one-dimensional. The relative velocity between the particles is $v_{ij} \equiv v_i - v_j$. Upon collision, the particles are reflected (see Fig. 1.4).

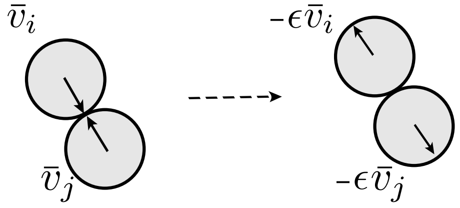


Figure 1.4: Two spherical particles collide inelastically. Total momentum is conserved; the sketch considers the center of mass rest frame. The individual rest frame velocities $\bar{v}_i$ and $\bar{v}_j$ are reflected during the collision and become smaller by the factor of ϵ, the coefficient of restitution.

Because of the inelastic collision, part of the kinetic energy is transferred into the particles' internal degrees of freedom and is hence lost from the dynamics. The relative velocity after the collision v'_{ij} is smaller due to the dissipative nature of the collision. The ration of the relative velocities after and before the collision is called the coefficient of restitution

$$\epsilon \equiv -\frac{v'_{ij}}{v_{ij}}. \tag{1.1}$$

Because of dissipation $0 \leq \epsilon \leq 1$, and therefore total energy is not conserved during the collision. Since only the relative velocity becomes smaller, the center of mass momentum is conserved, though.

In three dimensions, with particle velocities $\mathbf{v}_i$ and $\mathbf{v}_j$, the collision does not have to be head on, due to the spacial extent of the particles. If we define the geometry of the collision with the unit vector

$$\hat{\mathbf{e}} \equiv \frac{\mathbf{r}_i - \mathbf{r}_j}{\|\mathbf{r}_i - \mathbf{r}_j\|} = \frac{\mathbf{r}_{ij}}{\|\mathbf{r}_{ij}\|},$$

the relative velocity $\mathbf{v}_{ij} \equiv \mathbf{v}_i - \mathbf{v}_j$ decomposes into the parts normal $\mathbf{v}_{ij}^{\mathrm{n}}$ and tangential $\mathbf{v}_{ij}^{\mathrm{t}}$ to the contact surface of the particles, in the rest frame of the system (see Fig. 1.5). In general, particles are not completely spherically symmetric and have a friction acting between the surfaces. This makes the collision more complex in the way that the particles' rotational degrees of freedom enter the dynamics and moreover the collision then also affects the tangential motion.

If the reflection is assumed to act only on $\mathbf{v}_{ij}^{\mathrm{n}}$, this is for our case sufficient. In this case only the translational degrees of freedom enter the dynamics. The velocities after the collision are then given by

$$\begin{aligned} \mathbf{v}_i' &= \mathbf{v}_i - \frac{1+\epsilon}{2}(\mathbf{v}_{ij}\cdot\hat{\mathbf{e}})\,\hat{\mathbf{e}}, \\ \mathbf{v}_j' &= \mathbf{v}_j + \frac{1+\epsilon}{2}(\mathbf{v}_{ij}\cdot\hat{\mathbf{e}})\,\hat{\mathbf{e}}. \end{aligned} \tag{1.2}$$

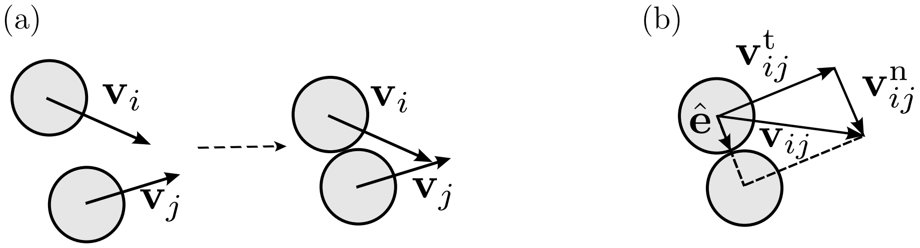


Figure 1.5: (a) Particles i and j move through space and collide. Upon collision, the particles are reflected. In three dimensions the relative velocity decomposes in a normal and tangential to the contact surface.

In our case, the situation becomes analogous to the one dimensional case for the normal component $\mathbf{v}_{ij}^{\mathrm{n}}$ so that

$$\epsilon \equiv -\frac{\mathbf{v}_{ij}^{\mathrm{n}\prime}}{\mathbf{v}_{ij}^{\mathrm{n}}}. \tag{1.3}$$

We stress that in granular collisions, total momentum and mass are conserved, while energy is dissipated.

A consequence of the loss of relative normal velocity is that over time correlations in the velocities build up, and the velocities become more and more parallel. In other words, the angle β between the inbound velocities $\mathbf{v}_i$ and $\mathbf{v}_j$ is always larger than that between the outbound primed velocities $\beta' < \beta$ (see Fig. 1.6).

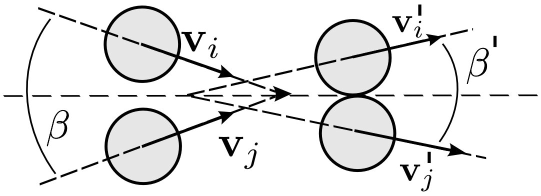


Figure 1.6: Schematic of the aligning effect of the granular collisions. The angle β' after the collision is smaller than the angle β before the collision.

We will discuss the effects of this fact throughout this work. For now, we recognize that this generates an additional correlation between particles and may lead to so-called ring collisions [10].

1.3 Theory

1.3.1 The coefficient of restitution

To derive a kinetic model suitable for analytical treatment and also numerical simulations we consider the simplest geometry: two spherical particles. Their elastic properties are then fully described by their radii and Young modulus. So when particles collide, the impact will lead to a deformation. The energy dissipated in the normal direction will be completely transferred into the viscoelastic deformation of the two particles.

The result of this approach is that the dissipation solely depends on the impact velocity according to

$$\epsilon \sim \left\|\mathbf{v}_{ij}^{\mathrm{n}}\right\|^{1/5}. \tag{1.4}$$

A common, albeit drastic, simplification of this so-called viscoelastic model assumes

$$\epsilon = \text{const.}$$

This assumption considerably simplifies analytical calculations and is moreover well justified in situations where the velocity distribution in the system is narrow and the system in a stationary state.

The stiffer the particles are, the shorter the collision time, *i.e.*, the time that the two particles touch, will be. This is an important condition for the assumption of pairwise collisions only, that we will make in the following. For coefficients of restitution close to one, this assumption holds better but when ϵ is considerably different than one the situation is more problematic.

In general, a classical system consisting of N particles is described by coupled Newton's equations of motion

$$m_i \ddot{\mathbf{x}}_i = \mathbf{F}(\mathbf{x}_1, \dots, \mathbf{x}_N, \mathbf{v}_1, \dots, \mathbf{v}_N) \tag{1.5}$$

for all particles $i = 1, \dots, N$. In the absence of long range interactions (*i.e.* electrostatics) particles will move ballistically on straight trajectories until they collide, and after the collision they will move on straight paths again. If the gas is dilute and the timescale of a single collision sufficiently small, ternary collisions are negligible and collisions will dominantly be binary. Instead of the full Newton's equation of motion, the system is described by a simpler pairwise interaction

$$m_{\mathrm{eff}} \ddot{\mathbf{x}}_{ij} = \mathbf{F}[\mathbf{x}_{ij}, \mathbf{v}_{ij}] \tag{1.6}$$

with $\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$ and the effective mass of the two body system

$$m_{\mathrm{eff}} = \frac{m_i m_j}{m_i + m_j}. \tag{1.7}$$

Together with an initial condition this gives a full description of a single collision. Between collisions the force $\mathbf{F}$ is zero.

Under the restriction of our assumptions, the dynamics are hence completely described by the property of a pair collision. While the dynamics of the single collision are not important, the change of the velocities before and after the collisions is what matters. Since the coefficient of restitution is the key property of a single pair collision, we can conclude that the coefficient of restitution is actually completely sufficient to describe the dynamics of a granular gas. It is moreover the only thing that distinguishes it from an ordinary molecular gas [10].

1.3.2 Temperature of a granular gas - Haff's law

The fact that in a granular gas there is a constant loss of kinetic energy makes it an out-of-equilibrium system. Still, like molecular gases, we can treat the granular gas with the tools of statistical mechanics. For this we consider a system in the thermodynamic limit, *i.e.*, consisting of a large number of particles N contained in some volume V. As for a molecular gas the number density is $\rho \equiv N/V$. Moreover, particle velocities are randomly distributed, and if no macroscopic flow is present, the average velocity $\langle \mathbf{v} \rangle$ will be zero. Though, like in molecular gases there are fluctuations around the mean velocity. In analogy to the temperature of a molecular gas, the granular temperature is defined as the mean energy per degree of freedom, that is the variance of the velocity

$$\frac{3}{2} k_{\mathrm{B}} T \; = \; \left\langle \frac{1}{2} m \| \mathbf{v} - \langle \mathbf{v} \rangle \|^2 \right\rangle, \tag{1.8}$$

where k_{B} is Boltzmann's constant and the angular brackets denote average over the particles. This temperature is of course different from the temperature of the grains themselves, and as we have stated earlier not coupled to the former. The collisions conserve number density and center of mass velocity, but not energy. Hence, the granular gas cools down, even in an externally force free environment. We investigate the granular temperature to monitor the state of the system.

In the early stage the granular gas cools down without the occurrence of any spatial inhomogeneity, which is called the *homogeneous cooling state*. In the following, we assume these conditions to hold in order to derive Haff's law describing the cooling of a homogeneous granular gas. The decay of temperature ΔT during a time interval Δt is described by the number of collisions that occur during this time and the velocity of the colliding particles. In the thermodynamic limit we can work with the averages of those.

The frequency of collisions is obtained via the volume of the so-called collision cylinder multiplied by the number density. With $\langle v_{ij} \rangle \propto \sqrt{T}$, the collision frequency in Δt is

$$\nu(\Delta t) \; \propto \; \rho \sigma^2 \sqrt{T} \Delta t. \tag{1.9}$$

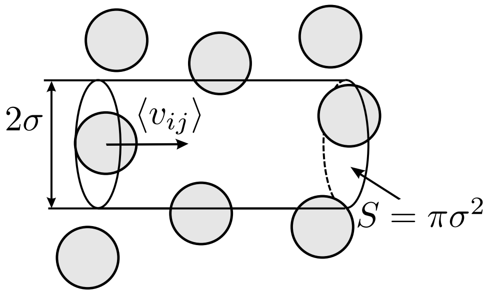


Figure 1.7: The collision cylinder of a reference particle moving with respect to the background. For particles of diameter σ, the volume of this cylinder is $\langle v_{ij} \rangle S \Delta t$. The higher the density the shorter the mean free path and the more frequent particles will collide. The square root of the temperature scales the mean time between collisions.

The volume of the collision cylinder accounts for the space the particle sweeps through during Δt with respect to all other particles. The diameter of two particle diameters 2σ (*cf.* Fig. 1.7) originates from the fact that particles touch with their centers of mass distance being equal to σ.

The average energy difference before and after a collision is given by

$$\frac{1}{2} m_{\text{eff}} \langle v_{ij}'^2 - v_{ij}^2 \rangle \;\propto\; -(1-\epsilon^2)T, \tag{1.10}$$

where we have used Eq. (1.3) and assumed a constant coefficient of restitution. The product of Eq. (1.9) and Eq. (1.10) gives the average amount of thermal energy dissipated during Δt

$$\frac{\Delta T}{\Delta t} \;\propto\; -\rho\sigma^2(1-\epsilon^2)T^{3/2} \;\simeq\; \frac{\mathrm{d}T}{\mathrm{d}t}. \tag{1.11}$$

The solution to Eq. (1.11) yields Haff's law [11], that is, the temperature evolution of the homogeneous granular gas

$$T(t) \;=\; \frac{T(t_0)}{(1+t/\tau_0)^2} \tag{1.12}$$

with $\tau_0^{-1} \propto n\sigma^2(1-\epsilon^2)\sqrt{T(t_0)}$. For systems in which the coefficient of restitution depends on the collision velocity, the exponent of this power law changes. With the dependence of $1-\epsilon^2 \;\propto\; \langle (v_{ij}^{\mathrm{n}})^{1/5} \rangle \;\propto\; T^{1/10}$ obtained for viscoelastic particles, this leads to

$$T(t) \;=\; \frac{T(t_0)}{(1+t/\tau_0')^{5/3}}. \tag{1.13}$$

In both cases, the dissipation leads to a power-law cooling of the granular gas [10].

1.3.3 Boltzmann-Enskog equation for a granular gas

The Boltzmann-Enskog equation relates the dynamics of macroscopic quantities to the properties of the microscopic dynamics, *i.e.*, the collisions of particles. The theory's main concern is the velocity distribution function $f(\mathbf{v}, \mathbf{x}, t)$ which gives the probability to encounter a particle with velocity $\mathbf{v}$ at position $\mathbf{x}$ and time t. It is normalized such that integrating over the whole 6 dimensional phase space gives the total number of particles N in the system

$$\int \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{v} f(\mathbf{v}, \mathbf{x}, t) \ = \ N, \tag{1.14}$$

where we simplified the notation by dropping the integrations interval bounds of $\pm\infty$ and by using a single integral sign. In the presence of an external force $\mathbf{F}$, the Boltzmann-Enskog equation reads

$$\left(\frac{\partial}{\partial t} + \mathbf{v}\nabla_{\mathbf{x}} + \frac{\mathbf{F}}{m}\nabla_{\mathbf{v}} \right) f(\mathbf{v}, \mathbf{x}, t) \ = \ \left. \frac{\partial f}{\partial t} \right|_{\text{coll}}. \tag{1.15}$$

While the left hand side of Eq. (1.15) accounts the change due to streaming and external forces, the right hand side denotes the effect of the change of $f(\mathbf{v}, \mathbf{x}, t)$ due to a collision. The right hand side is an integral, the so-called collision integral, which is a non-linear function of f since multiple particles participate in a collision. In the following we will only consider binary collisions. The dynamics of the moments of the velocity distribution function are obtained by integration. Following [10], we sketch the derivation of the Boltzmann-Enskog equation via direct and inverse collisions for a homogeneous system.

Consider the probability of a state $f(\mathbf{v}, t)$. It can only change in two ways, *i.e.*, either a collision occurs and decreases the probability because its outcome is different or another collision's result equals the current state of interest such that the probability increases. The first possibility is called *direct collision*, the second *indirect collision*. Their frequencies during a time interval Δt are denoted with ν^- and ν^+, respectively. We recall the scattering cylinder for these frequencies. The probability of scatterer in the volume $\mathrm{d}\mathbf{x}_i$, which is determined by the scattering vector $\hat{\mathbf{e}}$, is $f(\mathbf{v}_j, t)\mathrm{d}\mathbf{v}_j\mathrm{d}\mathbf{x}_i$. The cross section of the cylinder is $\sigma^2\mathrm{d}\hat{\mathbf{e}}$ and its length $\mathbf{v}_{ij}\Delta t$.

The total number of collisions is given by the product of the probability of a particle being at $\mathbf{x}$ with $\mathbf{v}_i$ and the volume of the collision cylinder (*cf.* Fig. 1.8) times the probability of a particle residing in it with $\mathbf{v}_j$.

$$\nu^-(\mathbf{v}_i, \mathbf{v}_j, \hat{\mathbf{e}}, \Delta t) \ = \ f(\mathbf{v}_i, t)\mathrm{d}\mathbf{v}_i\mathrm{d}\mathbf{x} f(\mathbf{v}_j, t)\mathrm{d}\mathbf{v}_j\sigma^2\|\mathbf{v}_{ij} \cdot \hat{\mathbf{e}}\|\Delta t \ \mathrm{d}\hat{\mathbf{e}}. \tag{1.16}$$

The frequency of the inverse ν^+ collision which lead to the current state after the collision, has the same shape as ν^-, but instead containing the primed velocities $\mathbf{v}_i''(\mathbf{v}_i, \mathbf{v}_j)$,

$$\nu^+(\mathbf{v}_i'', \mathbf{v}_j'', \hat{\mathbf{e}}, \Delta t) \ = \ f(\mathbf{v}_i'', t)\mathrm{d}\mathbf{v}_i''\mathrm{d}\mathbf{x} f(\mathbf{v}_j'', t)\mathrm{d}\mathbf{v}_j''\sigma^2\left\|\mathbf{v}_{ij}'' \cdot \hat{\mathbf{e}}\right\|\Delta t \ \mathrm{d}\hat{\mathbf{e}}. \tag{1.17}$$

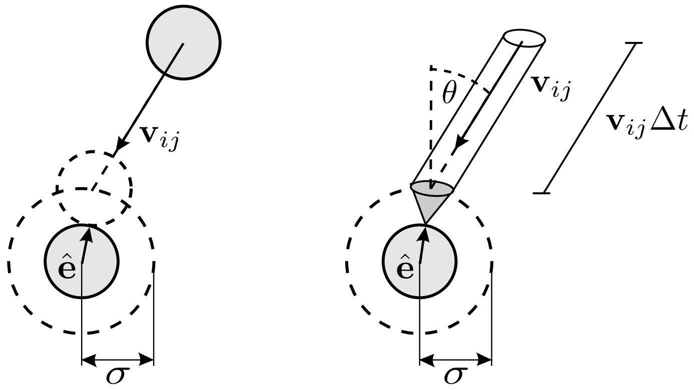


Figure 1.8: Sketch of the geometry leading to the direct collision rate ν^- (Eq. (1.16)). The scattering unit vector $\hat{\mathbf{e}} = \mathbf{x}_{ij}/\|\mathbf{x}_{ij}\|$ is the normal to the scattering surface, the basis of the scattering cylinder. The cylinder accounts for all possible collisions that hit the infinitesimal scattering surface around $\hat{\mathbf{e}}$.

Via the primed velocities that lead the unprimed velocities after the collision

$$\begin{aligned} \mathbf{v}_i'' &= \mathbf{v}_i - \frac{1+\epsilon}{2\epsilon}(\mathbf{v}_{ij}\cdot\hat{\mathbf{e}})\,\hat{\mathbf{e}}, \\ \mathbf{v}_j'' &= \mathbf{v}_j + \frac{1+\epsilon}{2\epsilon}(\mathbf{v}_{ij}\cdot\hat{\mathbf{e}})\,\hat{\mathbf{e}}, \end{aligned} \tag{1.18}$$

Eq. (1.17) is actually a function of the unprimed velocities.

The change in f is given by integration over the frequency of the direct and inverse collision. In order to write down the integral one needs to transform the integration to common variables of integration. The determinant of the Jacobian matrix for the mapping of $\mathbf{v}''$ to $\mathbf{v}$ is

$$\frac{\mathcal{D}\mathbf{v}''}{\mathcal{D}\mathbf{v}} = \frac{1}{\epsilon}.$$

Another factor of $1/\epsilon$ comes from the change of variables in the collision cylinder $\|\mathbf{v}_{ij}''\cdot\hat{\mathbf{e}}\| = \frac{1}{\epsilon}\|\mathbf{v}_{ij}\cdot\hat{\mathbf{e}}\|$. Using the Heaviside step function $\Theta(x)$ to assure that only velocities that lead to collisions are considered, the Boltzmann equation is obtained

$$\begin{aligned} \left.\frac{\partial f(\mathbf{v}_i,t)}{\partial t}\right|_{\text{coll}} &= \sigma^2 \int \mathrm{d}v_j \int \mathrm{d}\hat{\mathbf{e}}\;\Theta(-\mathbf{v}_{ij}\cdot\hat{\mathbf{e}})\|\mathbf{v}_{ij}\cdot\hat{\mathbf{e}}\| \\ &\qquad \times [\frac{1}{\epsilon^2} f(\mathbf{v}_i'',t) f(\mathbf{v}_j'',t) - f(\mathbf{v}_i,t)f(\mathbf{v}_j,t)] \qquad (1.19) \\ &\equiv I(f,f). \end{aligned}$$

The first of the summands on the right hand side represents the inverse collisions, increasing the probability of the current state, the second represents direct collisions. These method of derivation is quite intuitive, though a few facts have to be pointed out separately. First, particle positions were considered mutually independent in the derivation, while in reality, pairs have a joint distribution function $f_2(\mathbf{v}_i, \mathbf{v}_j, t)$ that doesn't necessarily factor out. The assumption of independent positions is valid in dilute gases. A more realistic approximation, however, is $f_2(\mathbf{v}_i, \mathbf{v}_j, t) \approx g(\sigma) f(\mathbf{v}_i, t) f(\mathbf{v}_j, t)$, where $g(\sigma)$ is the pair correlation function at contact distance, hence including excluded volume effects. In this context, the factor of $g(\sigma)$ is called the Enskog factor. Including it into Eq. (1.19), it becomes

$$\left.\frac{\partial f(\mathbf{v}_i, t)}{\partial t}\right|_{\text{coll}} = g(\sigma) I(f, f). \tag{1.20}$$

A second remark comes from the nature of collisions. One condition of the Boltzmann *Stoßzahlansatz* is that the colliding particles' velocities are initially uncorrelated, *i.e.* the assumption of molecular chaos. The aligning effect sketched in Fig. 1.6 though leads to ring collision, such that this assumption is potentially not fulfilled [10].

1.3.4 Granular hydrodynamics

1.3.4.1 Preconditions

The dynamics of the moments of the velocity distribution function, that are directly linked to macroscopic observables, can also be obtained via the Boltzmann equation. While for the derivation of Haff's law, spatial homogeneity is assumed, small gradients are assumed here. This means that the microscopic scales of the mean free path and mean collision time are small with respect to the length L and time scale $\mathcal{T}$, respectively, of the macroscopic dynamics. Gradients are assumed to be small such that over the macroscopic length scales, the variation is of the order of the observable itself, for the temperature $\Delta T \sim T/L$. Another important condition is that the macroscopic flow velocities are sufficiently smaller than the thermal velocities, so that the flow is well in the subsonic regime, *i.e.*, with the local Mach number

$$\mathcal{M} = \sqrt{\frac{\langle \mathbf{v} \rangle^2}{T}} \ll 1 \tag{1.21}$$

being well below one. Asking for this condition of subsonic flow is a rather tricky undertaking, because granular collisions not only lead to an increasing alignment of particle velocities, but also the dissipative nature steadily changes the ratio of Eq. (1.21) to the undermining of the assumption of this condition. Flow can thus become supersonic, and in the case of the viscoelastic model of ϵ becomes subsonic again [10].

1.3.4.2 Hydrodynamic equations

In order to obtain the observables' dynamics, we separately multiply the Boltzmann Eq. (1.15) with the moments

$$\mathcal{J}_i \in \{1, \mathbf{v}, \mathbf{V}^2\} \tag{1.22}$$

with the local velocity $\mathbf{V} = \mathbf{v} - \mathbf{u}$. Their expectation values under the distribution function correspond to the macroscopic observables. Integrating over $\mathbf{v}_i$ thus yields dynamic equations for the latter [10]. For the first two moments there is no change due to the collision integral, since the individual collisions conserve both mass and total momentum. For the temperature ($\mathcal{J}_3 = \mathbf{V}^2$) the collisional contribution doesn't vanish:

$$\int d\mathbf{v} \frac{m\mathbf{V}^2}{2} g(\sigma) I(f,f) \;=\; \frac{3}{2} nT\zeta. \tag{1.23}$$

The cooling rate ζ is defined by

$$\zeta(\mathbf{x},t) \;=\; \frac{\pi m g(\sigma)\sigma^2}{24nT} \int d\mathbf{v}_i d\mathbf{v}_j \mathbf{v}_{ij}^3 f(\mathbf{x},\mathbf{v}_i,t) f(\mathbf{x},\mathbf{v}_j,t)(1-\epsilon^2). \tag{1.24}$$

Note that with $\mathbf{v}_{ij} \sim \sqrt{T}$ the integral is of the order $\sim T^{3/2}$. With the macroscopic flow velocity $\mathbf{u}$, the hydrodynamics equations for granular gases read

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \mathbf{u}) \;=\; 0, \tag{1.25}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{\rho m} \nabla \cdot \tilde{P} \;=\; 0, \tag{1.26}$$

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T + \frac{2}{3\rho} \left(\tilde{P} : \nabla \mathbf{u} + \nabla \cdot \mathbf{q} \right) + \zeta T \;=\; 0, \tag{1.27}$$

where the vector $\mathbf{q}$ denotes the heat flux

$$\mathbf{q} \;=\; \int d\mathbf{v} \frac{m}{2} \mathbf{V}^2 \mathbf{V} f(\mathbf{v},\mathbf{x},t) \tag{1.28}$$

and $\tilde{P}$ the pressure tensor defined by

$$\tilde{P}_{ij} = \int d\mathbf{v}\, m \left(v_i v_j - \frac{1}{3} \delta_{ij} \mathbf{V}^2 \right) f(\mathbf{v},\mathbf{x},t) + nT\delta_{ij}. \tag{1.29}$$

Here, we used the notation of δ_{ij} for the Kronecker delta. Moreover, we use short notation $\mathbf{ab}$ with no space between vectors for the dyadic product $\mathbf{a} \otimes \mathbf{b}$, and for the total contraction of two tensors $\tilde{a}$ and $\tilde{b}$

$$\tilde{a} : \tilde{b} \;\equiv\; \mathrm{Tr}\left(\tilde{a} \cdot \tilde{b} \right).$$

To obtain a closed description of the hydrodynamics, one has to express the pressure tensor $\tilde{P}$ and heat flux $\mathbf{q}$ in therms of the fields ρ, $\mathbf{u}$ and T. In linear order, these read

$$P_{ij} \;=\; p\delta_{ij} - \eta(\nabla_i u_j + \nabla_j u_i - \frac{2}{3}\delta_{ij} \nabla \cdot \mathbf{u}) \tag{1.30}$$

$$\mathbf{q} \;=\; -\kappa \nabla T - \mu \nabla \rho, \tag{1.31}$$

where p is the hydrostatic pressure, η the shear viscosity, κ the thermal conductivity, in general, these are also functions of the hydrodynamic field (for full expressions refer to *e.g.* [12, 13]).

1.3.5 Standard stochastic rotation dynamics

The standard SRD method [14] is a rather recent, well established mesoscopic method to model hydrodynamics. It is computationally efficient and well tunable. Since it is a particle based method, it is especially easy to couple to a solute. Furthermore, because of its mathematical simplicity, the system can be thoroughly treated analytically. Since its introduction, numerous articles about the derivation of hydrodynamic equations and transport coefficients have been published [14, 15, 3]. After its introduction a number of similar and modified algorithms have been published [16]. The family of algorithms that it formed are called multi-partice collison dynamics (MPCD). It reproduces full hydrodynamics with thermal fluctuations. It is used to model solvent dynamics in simulations of colloids, polymers and active swimmers [17].

1.3.5.1 Algorithm

The SRD model considers particles that move in continuous space with continuous velocities and constitute the SRD fluid. Their movement and collisions are described by simplified rules. The dynamic stages of the real fluid consist of two steps: (i) the free streaming, and (ii) the collisions between the particles [14]. One after another, these stages occur with a fixed time interval in between. While the free streaming is treated exactly, the collisions are represented by a simplified coarse-grained collision model. The aim of the streaming is to macroscopically transport mass, momentum and energy. Additionally, the collision redistributes these quantities among the particles.

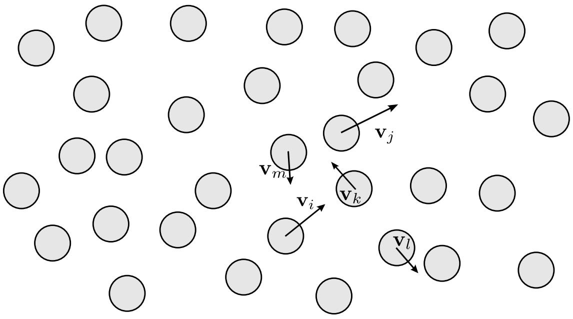


Figure 1.9: SRD particles move for the time Δt with independent continuous velocities in the free streaming step. Generally, particles are point-like and thus can overlap.

We consider a system of N particles of mass m_i, which reside and move in a continuous d-dimensional space with individual positions $\mathbf{x}_i$ and continuous velocities $\mathbf{v}_i$, $i \in [1, N]$ (see Fig. 1.9). In the free streaming step particles are advanced by their individual velocities for the duration of the time step Δt according to

$$\mathbf{x}_i(t + \Delta t) \;=\; \mathbf{x}_i(t) + \mathbf{v}_i \Delta t. \tag{1.32}$$

Next, the collision step is performed. To carry out the collision particles are grouped into Wigner-Seitz cells centered around the nodes of a regular lattice $\mathcal{L}$ with lattice nodes $\boldsymbol{\xi}$. For a cubic lattice this means that the set $\mathcal{V}_\xi$ of particles in cell $\boldsymbol{\xi}$ is defined such that particles for which $|\mathbf{x}_i - \boldsymbol{\xi}| < a/2$ lie in the cell around $\boldsymbol{\xi}$, where $|\mathbf{x}| = \max(x_x, x_y, x_z)$ and a is the lattice constant.

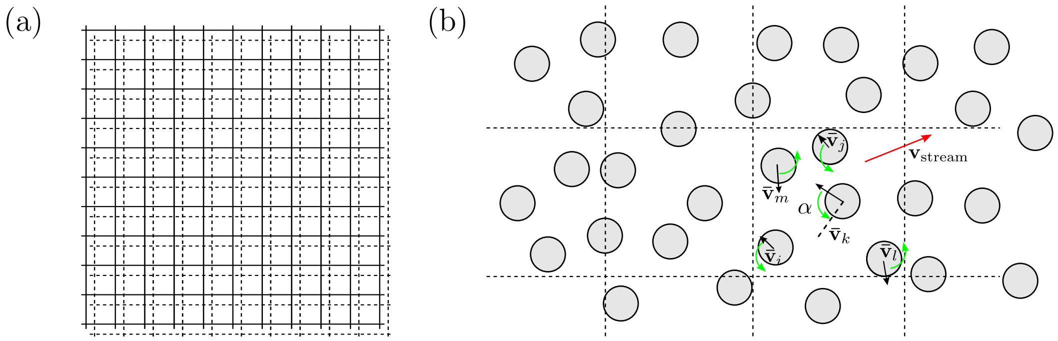


Figure 1.10: (a) A lattice is used to partition the system, and each particle is assigned to lattice cell. The lattice is randomly shifted before each collision step and shifted back afterwards. This ensures Galilean invariance. Panel (b): The thermal velocities inside a cell are obtained by subtracting the mean, *i.e.* streaming velocities. Inside a cell, those thermal velocities are then rotated by a fixed angle α, in the dimensions around a random axis, in two dimension with $\pm\alpha$. Afterwards the streaming velocity is added again.

In this cell the mean velocity $\mathbf{V}_\xi$, corresponding to the streaming velocity, is subtracted from the individual velocities. The leftover thermal velocities are then rotated with a random fixed angle simultaneously in each cell. In two dimensions this means randomly choosing $\pm\alpha$, while in three dimensions a random axis of rotation is chosen (see Fig. 1.10). In three dimensions only $+\alpha$ is considered since the random rotation axis also covers the opposite rotation. The rotation matrix $\tilde{\omega}_\xi$ thereby is independently generated for each cell. After the rotation, the streaming velocity is added back to the individual particles' velocities. The collision step hence reads

$$\mathbf{v}_i(t + \Delta t) \;=\; \mathbf{V}_\xi + \tilde{\omega}_\xi[\mathbf{v}_i - \mathbf{V}_\xi], \tag{1.33}$$

were $\tilde{\omega}_\xi$ is a random rotation from a set Ω. Given the instantaneous cell number density

$$\rho_\xi \;\equiv\; \frac{1}{\mathcal{V}} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} 1, \tag{1.34}$$

the center of mass velocity is defined as

$$\mathbf{V}_\xi \;\equiv\; \frac{1}{\sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} m_i} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} m_i \mathbf{v}_i \;\overset{m_i = m}{=}\; \frac{1}{n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} \mathbf{v}_i, \tag{1.35}$$

where n_ξ is the instantaneous number of particles in cell $\boldsymbol{\xi}$ and $\mathcal{V} = a^3$ the volume of a single cell. Finally, the instantaneous local temperature of a SRD cell is defined as

$$\theta_\xi \equiv \frac{1}{3n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} m_i \|\mathbf{v}_i - \mathbf{V}_\xi\|^2. \tag{1.36}$$

In the following, we generally consider equal masses $m_i = m$. The dynamics generated by these two rules conserve mass, momentum and energy. Mass conservation follows by definition, momentum conservation can be easily proven by

$$\begin{aligned}\mathbf{V}_\xi(t+\Delta t) &= \frac{1}{n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} \mathbf{v}_i(t+\Delta t) \;=\; \frac{1}{n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} (\mathbf{V}_\xi + \tilde{\omega}[\mathbf{v}_i - \mathbf{V}_\xi]) \\ &= \frac{1}{n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} \mathbf{V}_\xi + \underbrace{\tilde{\omega}\frac{1}{n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} \mathbf{v}_i}_{=\tilde{\omega}\mathbf{V}_\xi} - \underbrace{\tilde{\omega}\frac{1}{n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} \mathbf{V}_\xi}_{=\tilde{\omega}\mathbf{V}_\xi} \;=\; \mathbf{V}_\xi.\end{aligned} \tag{1.37}$$

Similarly, energy conservation follows from

$$\begin{aligned}\theta_\xi(t+\Delta t) &= \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} ||\mathbf{v}_i(t+\Delta t) - \underbrace{\mathbf{V}_\xi(t+\Delta t)}_{=\mathbf{V}_\xi}||^2 \;=\; \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} \|\tilde{\omega}[\mathbf{v}_i - \mathbf{V}_\xi]\|^2 \\ &= \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} [\mathbf{v}_i^2 - \mathbf{v}_i^T\mathbf{V}_\xi - \mathbf{V}_\xi^T\mathbf{v}_i + \mathbf{V}_\xi^2] \;=\; \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}_\xi} \|\mathbf{v}_i - \mathbf{V}_\xi\|^2 \\ &= \theta_\xi,\end{aligned} \tag{1.38}$$

where we have used that for two arbitrary vectors $\mathbf{a}$ and $\mathbf{b}$ it holds $(\mathbf{a}\tilde{\omega})^T(\tilde{\omega}\mathbf{b}) = \mathbf{a}^T\mathbf{b}$ because $\det(\tilde{\omega}) = 1$. For the standard SRD method, a Boltzmann equation has been derived from the Liouville's equation and it has been shown that the model yields correct linear hydrodynamic equations [14]. In another approach solely considering transport through virtual surfaces in the fluid, the same result has been obtained [15].

The random grid shift was introduced by Ihle and Kroll [18] to ensure Galilean invariance. In this procedure, first a random shift in the interval $[-a/2, a/2]$ in space is applied to all particle positions. While it was not introduced in the beginning, it has been shown, that not applying a random shift to the grid introduces spurious correlations at low temperatures. Those are spatially anisotropies due to the shape of the lattice and originate from the fact that often the same particles participate in subsequent collisions.

1.3.5.2 Computational complexity of SRD

The SRD method has some major advantages that make it very suitable for numerical simulations. Maybe the most remarkable fact is that SRD is an $\mathcal{O}(N)$ algorithm. This is because there is no pair interaction that has to be calculated. Another important property of the algorithm is that the fundamental quantities used are 1-body observables. Basically the mass and momentum per cell have to be calculated every step. Those are simple sums of particle properties. This gives the algorithm the possibility of an immense parallel speedup on parallel hardware, especially using vector accelerators or graphic cards.

Calculations of the cell-wise quantities $\mathbf{V}_\xi, \rho_\xi, \theta_\xi$ generally needs exclusive or atomic read/write access to the cells memory sections. However, the number of processor units $\#P$ is very small compared to the typical number of particles N on the hardware

$$\mathcal{O}(\#P) = 10^3 \;\ll\; \mathcal{O}(N) = 10^8. \tag{1.39}$$

Typically, there are on average 5 to 25 particles in a cell $\boldsymbol{\xi}$. Hence, the situation where multiple processors access the same memory locations are rare. This is favorable because these accesses have to be exclusive to guarantee memory consistency. Multiple exclusive access attempts on parallel hardware are called *exclusive access collisions*. These accesses have to be performed in sequential order, thus their processing is slow.

Given a fixed size parallel hardware, the algorithm becomes more efficient for increasing number of particles because less exclusive access collisions are occurring. Eventually, the algorithm boils down to summing on parallel hardware plus independent calculation on cells and on particles. The latter is computationally more expensive, the former needs synchronization, though sums are of complexity $\mathcal{O}(\log N)$ on parallel hardware. The bottleneck and size limit are mainly given by the memory size and memory bus. Since not even cells' states fit in any cache, all storage resides in the lowest layer *i.e.* the random access memory (RAM). The Nvidia graphic processing unit (GPU) cards that we used have separate graphic RAM with sizes of either 6 gigabytes (GB) for the Tesla K20Xm card and 12 GB for the Tesla K40 card. With single precision floating point numbers, simulations with 200 million particles and 200^3 cells can be performed within short time.

1.3.5.3 Streaming viscosity of standard SRD

For the SRD fluid, there are two contributions to the kinematic viscosity $\nu = \eta/\rho$ that originate from the two steps of the model. The total kinematic viscosity reads

$$\nu \;=\; \nu_{\text{stream}} + \nu_{\text{coll}}, \tag{1.40}$$

where ν_{stream} and ν_{coll} denote the contribution from the SRD streaming step and collision step, respectively. The streaming viscosity ν_{stream} is usually clearly dominant, and thus the collisional part can be neglected.

In this section we derive expressions for the streaming viscosity of the fluid. Because precise theoretical predictions are available for the viscosity, comparing it with the numerical results provides an important benchmark of our implementation.

The first approach [14] to derive the transport coefficients used the Chapman-Enskog approach, another study was done using the Green-Kubo formula [19]. Another approach by Kikuchi *et al.* [3] shows excellent agreement with simulation results for the viscosity.

If the time between collisions Δt is sufficiently large, the streaming contribution to the viscosity is major and the collisional viscosity negligible. We consider a system under shear with a rate of $\dot{\gamma} = \partial u_x / \partial y$ in the x direction. This system can either be in two or three dimensions. Here we consider two dimensions. The system under this condition relaxes to a steady state where the velocity profile in the sheared direction x is linear in y such that $\mathbf{u} = (\dot{\gamma} y, 0)^T$.

The corresponding off-diagonal entry of the pressure tensor will be determined by the shear rate $\dot{\gamma}$ and shear viscosity η as

$$\sigma_{xy} = \eta \frac{\partial u_x}{\partial y}. \tag{1.41}$$

The pressure tensor element can be evaluated in the simulations following its definition

$$\sigma_{xy} = -(\text{ flux of } x \text{ momentum through a plane of constant } y). \tag{1.42}$$

Or in other words, σ_{xy} equals the x momentum carried by all particles that have velocities v_y large enough to cross the plane in Δt. The velocity v_x at y is different according to the profile, thus σ_{xy} reads

$$\begin{aligned}\sigma_{xy} = & -\frac{\rho}{\Delta t} \int_{-\infty}^{+\infty} \mathrm{d}v_x \int_{-\infty}^{0} \mathrm{d}y \int_{+y/\Delta t}^{+\infty} \mathrm{d}v_y \, v_x f(v_x - \dot{\gamma} y, v_y) \\ & + \frac{\rho}{\Delta t} \int_{-\infty}^{+\infty} \mathrm{d}v_x \int_{0}^{+\infty} \mathrm{d}y \int_{-\infty}^{-y/\Delta t} \mathrm{d}v_y \, v_x f(v_x - \dot{\gamma} y, v_y),\end{aligned} \tag{1.43}$$

for a plane at $y = 0$ and can be reduced to

$$\sigma_{xy} = -\frac{\dot{\gamma}\rho\Delta t}{2}\langle v_y^2 \rangle - \rho \langle v_x v_y \rangle, \tag{1.44}$$

where averages are with respect to the velocity distribution f. There are two contributions, the first from thermal fluctuations and the second from correlation between v_x and v_y.

The correlation changes due to both stages of the SRD method. A closed expression can be found via a self consistency ansatz between the streaming and collisional contribution. We first consider the streaming contribution. Particles from $y > 0$

tend to have a higher x velocity and those from $y < 0$ a lower. To obtain the average after the streaming, we can thus average over the sheared velocity distribution function

$$\langle v_x v_y \rangle_{\text{after streaming}} = \int_{-\infty}^{+\infty} \mathrm{d}v_x \int_{-\infty}^{+\infty} \mathrm{d}v_y v_y v_x f(v_x - \dot{\gamma} y \Delta t, v_y) = \langle v_x v_y \rangle - \dot{\gamma} \Delta t \langle v_y^2 \rangle. \tag{1.45}$$

We see that the streaming operation reduces the correlation between the components by a constant and thus making them anti-correlated. As we will see, collisions reduce correlations. Here, also one calculates the change of correlation $\langle v_x v_y \rangle$ due to the collision rule, taking into account the fluctuations of the number of particles. Eventually one arrives at

$$\langle v_x'' v_y'' \rangle = h(\alpha, \rho_0) \langle v_x v_y \rangle, \tag{1.46}$$

meaning the correlation after the collision is reduced by a constant h. By now requiring self consistency of the two contribution to the viscosity, *i.e.*

$$\left(\langle v_x v_y \rangle - \dot{\gamma} \Delta t \langle v_y^2 \rangle \right) h(\alpha, \rho_0) = \langle v_x v_y \rangle, \tag{1.47}$$

and using the equipartition theorem together with Eq. (1.45), one arrives at equations for the streaming viscosity. In two dimensions

$$\nu_{\text{stream}}^{2D} = \frac{k_B T \Delta t}{a^2} \left[\frac{\rho_0}{(\rho_0 - 1 + e^{-\rho_0})[1 - \cos(2\alpha)]} - \frac{1}{2} \right], \tag{1.48}$$

and in three dimensions applying the same procedure yields

$$\nu_{\text{stream}}^{3D} = \frac{k_B T \Delta t}{a^3} \left[\frac{5\rho_0}{(\rho_0 - 1 + e^{-\rho_0})[4 - 2\cos(\alpha) - 2\cos(2\alpha)]} - \frac{1}{2} \right]. \tag{1.49}$$

1.3.5.4 Numerical shear simulation

It is extremely common to perform computer simulations with periodic boundary conditions in order to reduce finite size effects [20]. This means that there are no walls confining the simulation space, instead particles that leave the simulation space *e.g.* at $\mathbf{x} = (x_{\max}, y, z)^T$ reenter at the opposite side $\mathbf{x} = (-x_{\max}, y, z)^T$. In the case of SRD this also applies to grid shifts. In this way a system of infinite size is mimicked. For the simulation of shear flow one needs to produce a velocity gradient. This is achieved by applying the so-called Lees-Edwards boundary conditions, which are similar to periodic boundary conditions [3, 20]. If the shearing planes are perpendicular to the y-axis, one applies periodic boundary conditions along the x- and z-axis. When a particle crosses the bounds of the y-axis its velocity changes. So for $\mathbf{x} = (x, y_{\max}, z)^T$, the particle reappears at $\mathbf{x} = (x - u\Delta t, -y_{\max}, z)^T$ with the new velocity $\mathbf{v} = (v_x - u, v_y, v_z)^T$.

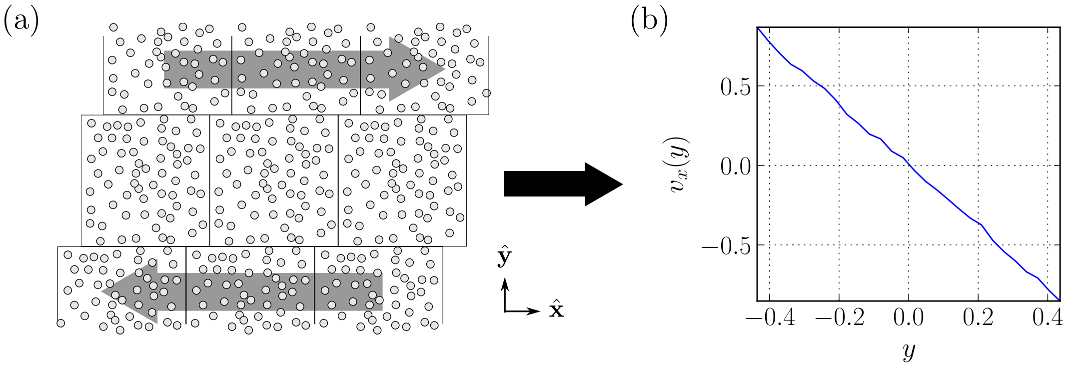


Figure 1.11: (a) Sketch of the shear flow simulation where periodic images are shifted in x and move with velocity $u(y)$. (b) Instantaneous linear velocity profile after some relaxation time obtained in our simulations.

For the opposite case of $\mathbf{x} = (x, -y_{\max}, z)^T$, the particle reappears at $\mathbf{x} = (x + u\Delta t, +y_{\max}, z)^T$ with the new velocity $\mathbf{v} = (v_x + u, v_y, v_z)^T$. This produces a linear velocity profile along the y direction after some equilibration time. Also for this sheared situation, an infinite system is mimicked.

Shear of fluids produces heat because of the internal viscous heating. The system under these conditions will continue to heat up as it is sheared. In order to provide steady conditions for measurements the temperature has to be held fixed. For this purpose a thermostat is added to the simulations. After each step all velocities are rescaled so that the global temperature matches the target temperature. Some studies, *e.g.* Lee *et al.* [21], apply cell-wise rescaling of the velocities to the target temperature. For the present situation of shear flow this creates results drastically deviating from the predictions and, thus, seems to alter the local dynamics too much. Furthermore, global drift is removed via Galilean transformation at each step because it can create artifacts.

The component of the pressure tensor can be obtained by evaluating the pressure tensor via Eq. (1.42). Therefore, we place a high number of planes perpendicular to the y-axis in the simulation space to increase statistics. One needs to exclude the region close to the periodic Lees-Edwards boundary from the measurement, since it creates unwanted artifacts. For the planes we sum the x-component of the velocity of all particles crossing the plane, thereby also taking into account the particles' possibility to cross several planes in one time step. Although the amount of shear is a manually controlled quantity, measuring the slope of the profile is necessary anyway, as the artifacts created by the boundary condition slightly change the slope of the profile.

1.3.5.5 Coupling to boundaries

Stochastic rotation dynamics are especially suitable for coupling to a solute or confining boundaries. For the case of walls, the particles are geometrically colliding in

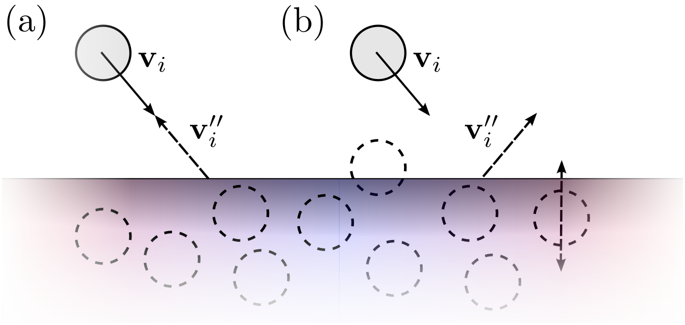


Figure 1.12: Collisions with a wall sketched with the lower gray layer. The walls contain ghost particles that are not translated in the streaming step. (a) Geometry of a bounce back collision. This type leads to no-slip boundary conditions of the wall. (b) Simple reflection creating a slip boundary condition.

the streaming step. Thereby, the shape of the walls is not restricted by the SRD lattice geometry and can thus be arbitrarily curved. With the so-called bounce back rule (Fig. 1.12(a)) no-slip boundary conditions can be generated [21].

In addition to the modification on the streaming step, the rotation step is also modified. This is because on the one hand, the grid shift performed before grouping particles into cells generates voids next to walls and on the other hand the reduced mean particle number in cells next to the walls alter the layer physical properties. To circumvent this, walls are filled with non-moving “ghost” particles. These do not move although they have a thermal velocity that is periodically and randomly reassigned. Their position only changes with the grid shift.

In the rotation step, the ghost particles are then considered for the calculation of the cell-wise quantities ρ_ξ, $\mathbf{V}_\xi$ and θ_ξ, a rotation step is though not performed for them.

The target temperature with which the random thermal velocities of ghost particles are assigned gives the additional freedom to give walls a temperature and construct simulations with thermal heating.

By applying forces to particles a Poiseuille flow experiment can be set up. In practice the simulation space is confined by no-slip walls instead of periodic boundaries, that are applied in the remaining dimensions. To generate streaming in a way that does not affect all simulation space, particles are accelerated in one dimension in a narrow region of the length of a few simulation cells.

1.3.5.6 Interpretation

Coarse graining addresses the interesting point: what is essential to describe the physics of the system under consideration? What can be simplified or neglected while retaining the phenomena of interest. So what is essential for the hydrodynamics of a medium? What microscopic properties does the medium need to have so that the continuum treatment yields the correct equations. If we take a look at the SRD method, we need to consider the two steps of the method. Via the streaming, SRD by definition generates macroscopic transport processes. At this point, *i.e.* considering only the streaming of SRD, it actually equals an ideal gas. The collision rule that is defined changes this because there are no collisions in an ideal gas. Collisions have in common with the streaming step that they conserve mass, momentum and energy. However, the collision rule redistributes them locally among the colliding particles. This leads to the relaxation to equilibrium and to the Maxwell Boltzmann velocity distribution.

As has been shown [14], this produces a system following the Navier-Stokes hydrodynamic equations, so that the essential ingredients necessary to retrieve hydrodynamics are the conservation laws, a streaming transport and local redistribution of the moments of the velocity distribution function via some process similar to collisions. With a look at the Boltzmann-Enskog equation we remark that we have, although the method is particle based, simplified the collision integral part of it while preserving its fundamental properties.

The reduction to these features is even more drastic in other algorithms of the family of multi-particle collision dynamics. In the widely used MPCD-AT [16], *i.e.* the multi-particle collision method with Anderson thermostat, particle velocities after the collision are randomly reassigned such that the moments of the velocity distributions function (mass, momentum and energy) are locally conserved. Hence, after the collisions the mean and variance equal to the pre-collision conditions, using local rescaling to make the local temperature match. This algorithm also fulfills hydrodynamic equations and exhibits a quicker local relaxation time compared to SRD. The SRD algorithm has also been adapted to model complex fluids like liquid crystals [21]. In this work, additional degrees of freedom for the local order and director were given to standard SRD particles and additional dynamics for those degrees of freedom were introduced.

Chapter 2

Granular stochastic rotation dynamics

2.1 Motivation

The standard stochastic rotation dynamics (discussed in Sec. 1.3.5) method has several advantages, such as the good runtime efficiency, the good paralleliseablility, *i.e.* ease of implementations with graphic cards, and, importantly analytical treatability. Compared to methods that discretize continuous equations such as the Navier-Stokes hydrodynamic model, via, for example finite volumes, the method SRD does not suffer from instabilities. Even if those arguments were left aside, the assumptions and simplifications underlying the model are tremendously efficient and fascinating from the physics point of view. An extension of stochastic rotation dynamics to granular systems is highly desirable because it might offer many insights into the physics of granular materials, on the one hand, and the potential of the SRD method, on the other hand. It has been said [10] that the coefficient of restitution ϵ is the only physical difference of granular from molecular gases. Of course, the granular system behaves more and more like a molecular as ϵ approaches one.

We consider the modification of the SRD method, instead of the also popular MPCD-AT method because the latter inherently provides a Maxwell-Boltzmann distribution of the particle velocities. In a granular system this is unsuitable because the velocity distributions of nonequilibrium systems do not follow Maxwell-Boltzmann. In contrast to MPCD, the standard SRD method does not impose *ad hoc* the Maxwell-Boltzmann, rather it dynamically reaches it.

2.2 Dissipative modification

As a first approach to account for the energy loss during collisions of granular particles, we consider a modification of the standard collision rule in Eq. (1.33) firstly because the dissipation is a feature of the collisions, and secondly because once the rotation step of a simulation is reached all local quantities are already summed up and reduced on parallel hardware.

As in standard SRD we want to guarantee mass and momentum conservation. Hence, also in the present case we subtract the mean velocity in each cell to obtain the thermal velocities. To mimic the dissipation we reduce the magnitude the rotated thermal velocities. At this point we reconsider the physics yielding Haff's law and Fig. 1.7. Performing a simple rescaling of the thermal velocities

$$\mathbf{v}_i(t+\Delta t) = \mathbf{V}_\xi(t) + \epsilon\tilde{\omega}_\xi\underbrace{(\mathbf{v}_i(t) - \mathbf{V}_\xi(t))}_{\text{thermal}})$$

generates an exponential cooling law because the instantaneous local temperature at the new time step reads

$$\theta_\xi(t+\Delta t) \; = \; \frac{1}{3}\sum_{i\in\mathcal{V}_\xi} m_i \left\|\mathbf{v}_i(t+\Delta t) - \mathbf{V}_\xi\right\|^2 \; = \; \epsilon^2\theta_\xi(t).$$

Hence

$$\frac{\Delta\theta_\xi}{\Delta t} \; = \; -\,\frac{(1-\epsilon^2)\theta_\xi}{\Delta t} \; \propto \; -\,\theta_\xi,$$

where we used $\det(\tilde{\omega}_\xi) = 1$, because $\tilde{\omega}_\xi$ is an orthogonal matrix. So, apparently this simple-minded attempt fails to generate the correct cooling behavior. The reason for this failure is that the collisions in SRD are occurring at a fixed frequency defined by the time step Δt. In a real granular gas collision rates for fast moving particles are higher. Also in dense regions, particles collide more frequently. Accounting for this via the collision rate is the obvious path in consideration of real granular gases. Introducing a locally variable collision rate to SRD is though not possible. The method would effectively become a multi-particle method but somehow event-driven in nature. We resolve this dilemma by realising that the interpretation of the SRD collision must be changed. So far, we considered the *virtual* particles' collisions as usual collisions, *i.e.* occurring after the average collision time. This now changes such that *granular SRD collisions represent multiple physical granular collisions per virtual particle.* In a hot region more collisions occur, hence if lattice cell $\mathcal{V}_\xi$ is hotter, more energy is dissipated in it per *representative* collision. Moreover the denser a region of the system is, the more collisions will occur. The dissipation in a *representative* collision must also increase with local density.

As we know from Section 1.3.2 about Haff's law, the collision rate changes proportionally to $\nu(\Delta t) \propto \rho\sqrt{T}\Delta t$. With that in mind, we introduce an effective coefficient of restitution that depends on both local temperature θ_ξ and number density ρ_ξ

$$\epsilon_{\xi,\text{eff}} \;=\; 1-(1-\epsilon^2)\,\sigma^2\rho_\xi\sqrt{\theta_\xi}\Delta t. \tag{2.1}$$

In this new definition, ϵ enters squared because $\sqrt{\theta_\xi}$ has the dimension of a velocity. If $\epsilon_{\xi,\text{eff}}$ is multiplied with a velocity, the product has the dimensions of an energy. The product also contains the particles' cross section σ^2 so that it becomes dimensionless. In three dimensions, the modification is designed to mimic the physical collisions.

Let us consider one of the dominant pair collisions *in the physical system*. The two colliding particles can be regarded as isolated from the rest of the system, following our assumptions. The binary system has an angular momentum and a plane perpendicular to it. The relative velocity is parallel to this plane. Similarly *in the SRD system* the axis of rotation defines a plane. To incorporate the fact that the relative velocity lies in a plane perpendicular to the axis of angular momentum, and only this is affected by the dissipative collision, we apply dissipation only on the rotated part of the thermal velocities.

The rule we have introduced has one interesting consequence: by applying it, we assume the system to be rather homogeneous at the length scale a of our observation.

2.3 Granular SRD algorithm

With the above considerations we introduce below the granular stochastic rotation dynamics (GSRD) method. The system consists of N particles with individual continuous positions $\mathbf{x}_i$ and velocities $\mathbf{v}_i$ for $i \in \{1 \dots N\}$. We inherit from SRD the streaming step

$$\mathbf{x}_i(t+\Delta t) \;=\; \mathbf{x}_i(t)+\mathbf{v}_i\Delta t. \tag{2.2}$$

After the streaming step follows the GSRD rotation step. Again, we group particles into Wigner-Seitz cells with lattice constant a. The lattice nodes $\boldsymbol{\xi}$ are randomly shifted by displacements with components distributed in the interval $[-a/2, a/2]$ to preserve Galilean invariance. With the inherited definitions of the local instantaneous density ρ_ξ, streaming velocity $\mathbf{V}_\xi$ and temperature θ_ξ we define the rotation step

$$\mathbf{v}_i(t+\Delta t) \;=\; \mathbf{V}_\xi(t)+\tilde{\omega}_{\xi,\text{eff}}[\mathbf{v}_i(t)-\mathbf{V}_\xi(t)] \tag{2.3}$$

where the granular, dissipative random rotation matrix $\tilde{\omega}_\xi$ is independently chosen for each cell. In two dimensions (2D) the rotation matrix reads

$$\tilde{\omega}^{\text{2D}}_{\xi,\text{eff}} \;\equiv\; \underbrace{\left[1-(1-\epsilon^2)\,\sigma^2\rho_\xi\sqrt{\theta_\xi}\Delta t\right]}_{\epsilon_{\xi,\text{eff}}(\rho_\xi,\theta_\xi)} \begin{pmatrix} \cos(\alpha) & \pm\sin(\alpha) \\ \mp\sin(\alpha) & \cos(\alpha) \end{pmatrix} \tag{2.4}$$

with the fixed angle of rotation α. The random rotations are generated by randomly choosing $\pm\alpha$.

Similarly, we define the rotation matrix in three dimensions (3D) as

$$\tilde{\omega}^{3\mathrm{D}}_{\xi,\mathrm{eff}} \equiv \begin{pmatrix} \epsilon_{\xi,\mathrm{eff}}(\rho_\xi,\theta_\xi)\cos(\alpha) & \epsilon_{\xi,\mathrm{eff}}(\rho_\xi,\theta_\xi)\sin(\alpha) & 0 \\ -\epsilon_{\xi,\mathrm{eff}}(\rho_\xi,\theta_\xi)\sin(\alpha) & \epsilon_{\xi,\mathrm{eff}}(\rho_\xi,\theta_\xi)\cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{2.5}$$

where the axis of rotation $\hat{\mathbf{r}}$ is generated randomly and independently in each cell and at every collision step. For the above definition (2.5) we are free to choose $\hat{\mathbf{r}} = \hat{\mathbf{z}}$ without loss of generality. We note that we don't need to choose randomly between $+\alpha \to -\alpha$ in 3D since it is already included by the possibility of randomly choosing $\hat{\mathbf{r}}$ equally to $-\hat{\mathbf{r}}$.

2.3.0.1 Thoughts on alternative collision rules

If we consider the 2D rule in Eq. (2.4) we might object that we are reducing the magnitude of the 2D thermal velocity vector. In contrast, a real binary collision does not have to be head on, due to the spatial extent of the particles. Hence, the real binary collision velocity splits in two parts normal and tangential to the contact surface. The real collisions thereby do not alter the tangential component. To resolve this and include this as a similar property in our collision, we might consider a random dissipation axis $\hat{\mathbf{d}}$, mimicking the normal component, and dissipating only parallel to it. This *alternative* would read

$$\mathbf{v}_i(t+\Delta t) = \mathbf{V}_\xi(t) + \left\{\hat{\mathbf{d}}(\epsilon_{\xi,\mathrm{eff}}(\rho_\xi,\theta_\xi) - 1)\,\hat{\mathbf{d}} + \mathbf{1}\right\}\tilde{\omega}_\xi[\mathbf{v}_i(t) - \mathbf{V}_\xi(t)].$$

Though it turns out that if we derive the cooling law for this rule, it does not change the dynamics more than changing how the cooling rates depend on ϵ, but still yields Haff's law. Since a single collision dissipates less energy, cooling becomes slower. Lastly, this rule would follow the same linear hydrodynamic equations.

Similar thoughts lead to the insight that as long as generating linear hydrodynamics is the only interest one might *alternatively* consider $\tilde{\omega}^{3\mathrm{D}}_{\xi,\mathrm{eff}} \equiv \epsilon_{\xi,\mathrm{eff}}\tilde{\omega}_\xi$ and thus only changing the cooling rate.

2.3.1 Boltzmann equation

In this section we will derive the Boltzmann equation for the evolution of the one particle probability density and show that the model we introduce obeys the correct linear hydrodynamic equations. Since in the model collisions occur at a constant frequency, the equations describe a discrete time dynamics. We will perform the transition to a continuous treatment of time as late as possible.

Whenever an integration sign without bounds occurs, the integration over the entire accessible phase space is meant.

2.3.1.1 Liouville equation

We start from the Liouville equation for the evolution of the probability density $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ of finding the system in the phase space point $(\mathbf{v}^{(N)}, \mathbf{x}^{(N)})$, where $\mathbf{x}^{(N)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ and $\mathbf{v}^{(N)} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N)$. For the discrete time dynamics collisions occur only once in a given Δt. So, during this time interval, the probability of the state $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ will change. We distinguish two classes of collisions based on their effect on $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$. Direct collisions occur with the probability of current state and reduce the probability since the state is transformed in the collision. The probability of direct collisions $\nu^-(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$, that reduce the probability of $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ is simply $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ since collisions occur independent of the particles' positions and velocities.

Inverse collisions $\nu^+(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ increase the probability of the current state of interest, and correspond to the probability of all the collisions that transform nearby states into the current state through one of the possible physical interactions.

The full dynamics are described by the Liouville equation describing the evolution of the probability density. It reads

$$\frac{P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t+\Delta t) - P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)}{\Delta t} = \mathcal{C}(P(t), P(t)) \tag{2.6}$$

where the collision operator

$$\begin{aligned}\mathcal{C}(P(t), P(t)) = & \frac{1}{\Delta t \, \|\Omega\|^L} \sum_{\Omega^L} \int \mathrm{d}\mathbf{v}'^{(N)} \mathrm{d}\mathbf{x}'^{(N)} P(\mathbf{v}'^{(N)}, \mathbf{x}'^{(N)}, t) \\ & \times \prod_{i=1}^{N} \Big[\underbrace{\delta(\mathbf{v}_i - \mathbf{V}_\xi - \tilde{\omega}_\xi(\rho_\xi, \theta_\xi)[\mathbf{v}'_i - \mathbf{V}_\xi])\delta(\mathbf{x}_i - [\mathbf{x}'_i + \mathbf{v}_i \Delta t])}_{\to\ \nu^+(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)} \\ & - \underbrace{\delta(\mathbf{v}_i - \mathbf{v}'_i)\delta(\mathbf{x}_i - \mathbf{x}'_i)}_{\to\ \nu^-(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)} \Big]\end{aligned}$$

describes the change of $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ due to collisions. The integral over product of δ-functions ν^+ describes the probability of all states that, through a collision, become the current state and hence, increase the probability $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)})$. Thereby, all rotations $\tilde{\omega}_\xi(\rho_\xi, \theta_\xi)$ from the set of possible rotations Ω in all lattice cells L are considered. The product of δ-functions denoted by ν^- describes the opposite, *i.e.*, the probability of the current state to transition into a nearby state in phase space and thus decrease $P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)})$. We can see the fact that the collisions occur in an unconditional fashion, or in other words particles need only to be in the same cell $\boldsymbol{\xi}$ and not in direct contact, because both $\nu^-(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ and $\nu^+(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t)$ are products of δ-functions.

2.3.1.2 Conservation laws

Since the collision operator in the Liouville equation is a continuous linear combination of all possible collision and streaming transformations, it inherits the conservation laws from the individual collisions. The Jacobi determinant of the streaming transformation is

$$\det\left(\frac{\partial x_{i\beta}(t+\Delta t)}{\partial x_{i\gamma}(t)}\right) = 1 \tag{2.7}$$

with $\beta, \gamma \in \{x, y, z\}$. Hence during streaming, all the first three moments of the velocity distribution function are conserved. We next focus on the collision transformation. Number density is conserved by definition. The center of mass momentum in a single SRD-cell changes according to

$$\begin{aligned} n_\xi \mathbf{V}_\xi(t+\Delta t) &= \sum_{i|\mathbf{x}_i\in\mathcal{V}} \mathbf{v}_i(t+\Delta t) \;=\; \sum_{i|\mathbf{x}_i\in\mathcal{V}} \{\mathbf{V}_\xi(t) + \tilde{\omega}_\xi(\rho_\xi, \theta_\xi(t))\underbrace{[\mathbf{v}_i(t) - \mathbf{V}_\xi(t)]}_{\sum(\dots)=0}\} \\ &= \sum_{i|\mathbf{x}_i\in\mathcal{V}} \mathbf{V}_\xi \;=\; n_\xi \mathbf{V}_\xi \end{aligned} \tag{2.8}$$

and hence is conserved, independent of the choice of the matrix $\tilde{\omega}(\rho, \theta)$. We expect the temperature not to be conserved:

$$\begin{aligned} \theta_\xi(t+\Delta t) &= \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i\in\mathcal{V}} \|\mathbf{v}_i(t+\Delta t) - \mathbf{V}_\xi(t+\Delta t)\|^2 \\ &= \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i\in\mathcal{V}} \|\mathbf{v}_i(t+\Delta t) - \mathbf{V}_\xi(t)\|^2 \\ &= \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i\in\mathcal{V}} \|\tilde{\omega}_\xi(\rho_\xi, \theta_\xi)[\mathbf{v}_i(t) - \mathbf{V}_\xi](t)\,\|^2 \\ &= \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i\in\mathcal{V}} ([\mathbf{v}_i(t) - \mathbf{V}_\xi(t)]^T \tilde{\omega}_\xi(\rho_\xi, \theta_\xi)^T \; \tilde{\omega}_\xi(\rho_\xi, \theta_\xi)[\mathbf{v}_i(t) - \mathbf{V}_\xi(t)]). \end{aligned}$$

We consider the rotation axis to coincide with the $\hat{\mathbf{z}}$-axis, so that without loss of generality

$$\tilde{\omega}_\xi^T \tilde{\omega}_\xi \;=\; (\epsilon_{\xi,\mathrm{eff}})^2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/(\epsilon_{\xi,\mathrm{eff}})^2 \end{pmatrix}. \tag{2.9}$$

Using the equipartition theorem we can rewrite the thermal velocities $\mathbf{v}_i^t \equiv (\mathbf{v}_i - \mathbf{V}_\xi)$ components

$$\frac{m}{n_\xi} \sum_{i|\mathbf{x}_i\in\mathcal{V}} (v_{ix}^t(t))^2 \;=\; \theta_\xi(t). \tag{2.10}$$

Using (2.9) and (2.10) we arrive at

$$
\begin{aligned}
\theta_\xi(t+\Delta t) &= \frac{m}{3n_\xi} \sum_{i|\mathbf{x}_i \in \mathcal{V}} (\epsilon_{\xi,\text{eff}})^2 \Big[(v^t_{ix}(t))^2 + (v^t_{iy}(t))^2 + \frac{1}{(\epsilon_{\xi,\text{eff}}(t))^2} (v^t_{iz}(t))^2 \Big] \\
&= \frac{2\,[\,\epsilon_{\xi,\text{eff}}(\rho_\xi, \theta_\xi(t))\,]^2 + 1}{3}\;\theta_\xi(t) \\
&= \frac{3 - 2(1-\epsilon^2)\,\sigma^2 n \sqrt{\theta_\xi}\Delta t + (1-\epsilon^2)^2\,\sigma^4 n^2 \theta_\xi \Delta t^2}{3}\;\theta_\xi(t) \\
&\simeq \frac{3 - 2(1-\epsilon^2)\,\sigma^2 \rho_\xi \sqrt{\theta_\xi}\Delta t}{3}\;\theta_\xi(t),
\end{aligned}
\tag{2.11}
$$

where in the last step we have neglected the second order terms in dissipation $(1-\epsilon^2)$ and time step. Thus, we obtain a decay rate of the thermal energy in a time step that reads

$$
\frac{\theta_\xi(t+\Delta t) - \theta_\xi(t)}{\Delta t} \;=\; -\,\frac{2(1-\epsilon^2)\,\sigma^2 \rho_\xi \theta_\xi(t)^{3/2}}{3}. \tag{2.12}
$$

As we can see, temperature decays upon collisions, in a fashion similar to real granular collisions, where $\mathrm{d}T/\mathrm{d}t \propto nT^{3/2}$, which is the evolution equation that gives Haff's law [11].

2.3.1.3 Boltzmann approximation

We now turn to the calculations of the collision integral in the granular Boltzmann equation. To make progress, we need to assume that there are no correlations among colliding particles in both space and velocity so that we can write the joint probability density as a product of the one particle probability density

$$
P(\mathbf{v}^{(N)}, \mathbf{x}^{(N)}, t) \;=\; \prod_{i=1}^{N} P_1(\mathbf{v}_i, \mathbf{x}_i, t). \tag{2.13}
$$

With this assumption we obtain the one particle density function

$$
f(\mathbf{v}, \mathbf{x}, t) \;=\; N P_1(\mathbf{v}, \mathbf{x}, t), \tag{2.14}
$$

for which we derive the equation of motion from the Liouville equation (2.6). We want to describe the evolution of $f(\mathbf{v}, \mathbf{x}, t)$ completely by the local change of $f(\mathbf{v}, \mathbf{x}, t)$ due to collisions. These collisions take place inside a SRD cell and affect all n particles that are located in this cell. Thus, we focus on the time evolution in a single cell $\boldsymbol{\xi}$ so that $\|\mathbf{x} - \boldsymbol{\xi}\| < 1/2$, considering every case of $n \leq N$ particles residing in the cell. In order to do so, we divide the one particle probability density into two parts inside and outside of the cell

$$
P_1(\mathbf{v}_i, \mathbf{x}_i, t) \;=\; P_1(\mathbf{v}_i, \mathbf{x}_i, t)[\,\Theta(\|\mathbf{x}_i - \boldsymbol{\xi}\| - 1/2) + \Theta(-\|\mathbf{x}_i - \boldsymbol{\xi}\| + 1/2)\,],
$$

where $\Theta(x)$ denotes the Heaviside step function. We can then integrate separately for the $N-n$ particles outside and the n inside the cell. For particles inside, we only integrate over the cell volume $\mathcal{V}(\boldsymbol{\xi})$, while for the remaining we drop the contribution of $\Theta(-\|\mathbf{x}_i-\boldsymbol{\xi}\|+1/2)$. Further, all n out of N particles cases and all possible $\tilde{\omega}\in\Omega$ rotations have to be considered. Multiplying both sides by $\delta(\mathbf{v}-\mathbf{v}_i)\ \delta(\mathbf{x}-\mathbf{x}_i)$ for all i and integrating over all $\mathbf{v}_i,\mathbf{x}_i$, the collision integral in Eq. (2.6) becomes

$$
\begin{aligned}
\frac{f(\mathbf{v},\mathbf{x},t+\Delta t)-f(\mathbf{v},\mathbf{x},t)}{\Delta t} &= \nu_\xi^+ - \nu_\xi^- \\
= \frac{1}{\Delta t}\sum_{n=1}^{N}\binom{N}{n}&\left[\int\prod_{i=n+1}^{N}\mathrm{d}\mathbf{v}_i\mathrm{d}\mathbf{x}_i P_1(\mathbf{v}_i,\mathbf{x}_i,t)\Theta(\|\mathbf{x}_i-\boldsymbol{\xi}\|-1/2)\right]\Bigg\{ \\
+\int_{\mathcal{V}(\boldsymbol{\xi})}&\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{v}_i''\mathrm{d}\mathbf{x}_i P_1(\mathbf{v}_i'',\mathbf{x}_i,t)\right]\frac{1}{\|\Omega\|}\sum_{i,\tilde{\omega}\in\Omega}\delta(\mathbf{v}-\mathbf{V}_\xi-\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i''-\mathbf{V}_\xi]) \\
\times\ &\delta(\mathbf{x}-[\mathbf{x}_i+\mathbf{v}_i\Delta]) \\
-\int_{\mathcal{V}(\boldsymbol{\xi})}&\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{v}_i\mathrm{d}\mathbf{x}_i P_1(\mathbf{v}_i,\mathbf{x}_i,t)\right]\frac{1}{\|\Omega\|}\sum_{i,\tilde{\omega}\in\Omega}\delta(\mathbf{v}-\mathbf{v}_i)\ \delta(\mathbf{x}-\mathbf{x}_i)\Bigg\}.
\end{aligned}
\tag{2.15}
$$

The first integral multiplied with both products accounts for the particles that reside outside of the current cell of interest. Using the normalisation of

$$\int\mathrm{d}\mathbf{v}\mathrm{d}\mathbf{x}P_1(\mathbf{v}_i,\mathbf{x}_i,t)=1,$$

and the fact that we are, in the respective case, considering n particles inside the cell $\boldsymbol{\xi}$, it may be rewritten as

$$
\begin{aligned}
\int\prod_{i=n+1}^{N}&\mathrm{d}\mathbf{v}_i\mathrm{d}\mathbf{x}_i P_1(\mathbf{v}_i,\mathbf{x}_i,t)\Theta(\|\mathbf{x}_i-\boldsymbol{\xi}\|-1/2) \\
&= \left[\int\mathrm{d}\mathbf{v}\mathrm{d}\mathbf{x}P_1(\mathbf{v},\mathbf{x},t)\Theta(\|\mathbf{x}-\boldsymbol{\xi}\|-1/2)\right]^{N-n} \\
&= \left(1-\frac{n_\xi}{N}\right)^{N-n}.
\end{aligned}
$$

In the limit of a large number of particles, we obtain

$$\frac{N!}{n!(N-n)!}\left(1-\frac{n_\xi}{N}\right)^{N-n}\xrightarrow{N\gg n}\frac{N^n(N-n)!}{n!(N-n)!}\left(1-\frac{n_\xi}{N}\right)^{N}\xrightarrow{N\to\infty}N^n\frac{e^{-n_\xi}}{n!}.$$

The factor of N^n can be taken into the definition of $f(\mathbf{v},\mathbf{x},t) = NP_1(\mathbf{v},\mathbf{x},t)$. The collision integral reads

$$\begin{aligned}\frac{f(\mathbf{v},\mathbf{x},t+\Delta t)-f(\mathbf{v},\mathbf{x},t)}{\Delta t} &= \frac{1}{\Delta t}\sum_{n=1}^{N}\frac{e^{-n_\xi}}{n!}\Bigg\{\int\limits_{\mathcal{V}(\boldsymbol{\xi})}\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{v}_i''\mathrm{d}\mathbf{x}_i f(\mathbf{v}_i'',\mathbf{x}_i,t)\right] \\ &\times \frac{1}{\|\Omega\|}\sum_{i,\tilde{\omega}\in\Omega}\delta(\mathbf{v}-\mathbf{V}_\xi-\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i''-\mathbf{V}_\xi])\;\delta(\mathbf{x}-[\mathbf{x}_i+\mathbf{v}\Delta t]) \\ &- \int\limits_{\mathcal{V}(\boldsymbol{\xi})}\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{v}_i\mathrm{d}\mathbf{x}_i f(\mathbf{v}_i,\mathbf{x}_i,t)\right]\frac{1}{\|\Omega\|}\sum_{i,\tilde{\omega}\in\Omega}\delta(\mathbf{v}-\mathbf{v}_i)\;\delta(\mathbf{x}-\mathbf{x}_i)\Bigg\},\end{aligned} \tag{2.16}$$

where the term resulting from the preceding calculation reflects the nature of the number of particles residing in one box during a collision that follows a Poisson distribution. In order to obtain a more intuitive form of the collision integral, similarly to Pöschel and Brilliantov [10], we do a coordinate transformation between the pre- and post-collision velocities. To transform the integration we need to calculate the Jacobi determinant of the collision step. For the collision of n particles in a single cell, we may rewrite the collision transformation as

$$\mathbf{v}_\xi^{(n)} = \mathbf{V}_\xi^{(n)} + \begin{pmatrix}\tilde{\omega}_\xi & 0 & & \\ 0 & \tilde{\omega}_\xi & \ddots & \\ & \ddots & \ddots & 0 \\ & & 0 & \tilde{\omega}_\xi\end{pmatrix}\left[\mathbf{v}_\xi^{(n)\prime\prime} - \mathbf{V}_\xi^{(n)}\right]. \tag{2.17}$$

where $\mathbf{v}_\xi^{(n)} = (\mathbf{v}_1,\ldots,\mathbf{v}_n)$ with the velocities $\mathbf{v}$ of all particles in the cell and $\mathbf{V}_\xi = (\mathbf{V}_\xi,\ldots,\mathbf{V}_\xi)$. The Jacobian for the $\epsilon = \text{const.}$ case reads

$$\det\left(\frac{\partial\mathbf{v}^{(n)}}{\partial\mathbf{v}^{(n)\prime\prime}}\right) = \det\begin{pmatrix}\tilde{\omega}_\xi & 0 & & \\ 0 & \tilde{\omega}_\xi & \ddots & \\ & \ddots & \ddots & 0 \\ & & 0 & \tilde{\omega}_\xi\end{pmatrix}. \tag{2.18}$$

If we, without loss of generality, consider the rotation axis to coincide with the $\hat{\mathbf{z}}$-axis the rotation-dissipation matrix becomes

$$\tilde{\omega}_\xi = \begin{pmatrix}\tilde{\omega}_{\xi,xy} & 0 \\ 0 & 1\end{pmatrix}$$

where $\tilde{\omega}_{\xi,xy}$ is a two dimensional rotation matrix. We can then reorder the entries of $\mathbf{v}^{(n)}$ so that Eq. (2.18) transforms to

$$\det \begin{pmatrix} 1 & 0 & & & & & \\ 0 & \ddots & \ddots & & & & \\ & \ddots & 1 & 0 & & & \\ & & 0 & \tilde{\omega}_{\xi,xy} & 0 & & \\ & & & 0 & \tilde{\omega}_{\xi,xy} & \ddots & \\ & & & & \ddots & \ddots & 0 \\ & & & & & 0 & \tilde{\omega}_{\xi,xy} \end{pmatrix} = \det \begin{pmatrix} \tilde{\omega}_{\xi,xy} & 0 & & \\ 0 & \tilde{\omega}_{\xi,xy} & \ddots & \\ & \ddots & \ddots & 0 \\ & & 0 & \tilde{\omega}_{\xi,xy} \end{pmatrix}.$$

The matrix $\tilde{\omega}_{\xi,xy}$ is of $m \times m$ type with $m = 2$. Since it is a rotation followed by a multiplication, we can use the property of the determinant, that for an $m \times m$ matrix

$$\det(c\,\tilde{\omega}) = c^m \det(\tilde{\omega})$$

to finally arrive at

$$\det\left(\frac{\partial \mathbf{v}^{(n)}}{\partial \mathbf{v}^{(n)\prime\prime}}\right) \;=\; (\epsilon_{\xi,\text{eff}})^{2n}. \tag{2.19}$$

We now transform the first summand of Eq. (2.16) by changing the integration over the pre- to the post-collision velocities, *i.e.* $\mathbf{v}_i'' \to \mathbf{v}_i\ \forall i$. The differentials change according to

$$\prod_{i=1}^{n} \mathrm{d}\mathbf{v}_i'' \;=\; \det\left(\frac{\partial \mathbf{v}^{(n)\prime\prime}}{\partial \mathbf{v}^{(n)}}\right) \prod_{i=1}^{n} \mathrm{d}\mathbf{v}_i \;=\; \frac{1}{(\epsilon_{\xi,\text{eff}})^{2n}} \prod_{i=1}^{n} \mathrm{d}\mathbf{v}_i. \tag{2.20}$$

We obtain

$$\begin{aligned} \frac{f(\mathbf{v},\mathbf{x},t+\Delta t) - f(\mathbf{v},\mathbf{x},t)}{\Delta t} \;=\; & \frac{1}{\Delta t} \sum_{n=1}^{N} \frac{e^{-n_\xi}}{n!} \int\limits_{\mathcal{V}(\boldsymbol{\xi})} \left[\prod_{j=1}^{n} \mathrm{d}\mathbf{v}_j \mathrm{d}\mathbf{x}_j\right] \Bigg\{ \qquad (2.21) \\ & \frac{1}{(\epsilon_{\xi,\text{eff}})^{2n}} \frac{1}{\|\Omega\|} \sum_{i,\tilde{\omega}\in\Omega} \left[\prod_{j=1}^{n} f(\mathbf{v}_j'', \mathbf{x}_j, t)\right] \delta(\mathbf{v}-\mathbf{v}_i)\, \delta(\mathbf{x} - [\mathbf{x}_i + \mathbf{v}\Delta t]) \\ & - \left[\prod_{j=1}^{n} f(\mathbf{v}_j, \mathbf{x}_j, t)\right] \frac{1}{\|\Omega\|} \sum_{i,\tilde{\omega}\in\Omega} \delta(\mathbf{v}-\mathbf{v}_i)\, \delta(\mathbf{x}-\mathbf{x}_i) \Bigg\}. \end{aligned}$$

We can see similarities to the physical Boltzmann-Enskog equation as derived by Pöschel and Brilliantov [10]. Instead of a collision with two particles, we now have an n-particle collision, which results in n integrations and the product of n one-particle distribution function. Similarly, we see the non-equilibrium character in the

imbalance created by the factor
$$\frac{1}{(\epsilon_{\xi,\mathrm{eff}})^{2n}}$$
in front of the term due to inverse collisions. The power of $2n$ can be understood considering that only the two thermal velocity perpendicular to the rotation axis are effected by the dissipation.

2.3.1.4 Hydrodynamic equations

We now consider a granular gas with small initial inhomogeneities. There will be transport phenomena of the zeroth, first and second moments of the velocity. For these we derive transport equations.

In order to do this, we need to consider small gradients in density and temperature. Here, small means that the length scale of the changes of the macroscopic observables L are of the order of the observables and hence much larger than the length scale of the microscopic dynamics, *i.e.* larger than the mean free path of particles l. For the temperature for example, this means that

$$\nabla T \sim \frac{T}{L}, \qquad L \gg l. \tag{2.22}$$

We assume a similar condition for the density. Furthermore, for this treatment we need to restrict ourselves to the regime of subsonic flows. This means that the flow velocities u are much smaller than the square root of the temperature, *i.e.*

$$u \ll \sqrt{\langle v^2 \rangle} \sim v_T \sim \sqrt{T}. \tag{2.23}$$

These conditions are usually fulfilled in a molecular gas. For a granular gas, the situation is different though, because of the dissipative nature of the collisions. Since collisions occur in the center of mass frame, the temperature decays while macroscopic flows persists. This quickly leads to supersonic flows in the inhomogeneous system [10].

Let us now consider the macroscopic fields that can be observed. These correspond to the moments of the velocity distribution function, the zeroth *i.e.* local number density $\rho(\mathbf{x},t)$, the first *i.e.* average local particle velocity $\mathbf{u}(\mathbf{x},t)$ and second the local temperature $T(\mathbf{x},t)$. They are defined as

$$\rho(\mathbf{x},t) \equiv \int \mathrm{d}\mathbf{v} f(\mathbf{v},\mathbf{x},t), \tag{2.24}$$

$$\rho(\mathbf{x},t)\mathbf{u}(\mathbf{x},t) \equiv \int \mathrm{d}\mathbf{v}\ \mathbf{v} f(\mathbf{v},\mathbf{x},t), \tag{2.25}$$

$$\rho(\mathbf{x},t)T(\mathbf{x},t) \equiv \int \mathrm{d}\mathbf{v}\,\frac{m}{3}\,||\underbrace{\mathbf{v}-\mathbf{u}(x,t)}_{=\mathbf{V}}||^2 f(\mathbf{v},\mathbf{x},t), \tag{2.26}$$

Since the temperature is the only collision non-invariant moment of interest, we are especially interested in its behavior. For the following we change back the integration variable of the first term of Eq. (2.21) which is more suitable. If condition

(2.22) is met, we can neglect the change in temperature and density due to streaming in the collision integral via the $\delta(\mathbf{x}_i-[\mathbf{x}+\mathbf{v}\Delta t])$ term. For $\delta(\mathbf{x}_i-[\mathbf{x}+\mathbf{v}\Delta t])\simeq\delta(\mathbf{x}_i-\mathbf{x})$ the Boltzmann equation becomes

$$\begin{aligned}
&\frac{f(\mathbf{v},\boldsymbol{\xi},t+\Delta t)-f(\mathbf{v},\boldsymbol{\xi},t)}{\Delta t} = \qquad\qquad\qquad\qquad (2.27)\\
&\frac{1}{\Delta t}\sum_{n=1}^{N}\frac{e^{-n_\xi}}{n!}\Bigg\{\int\limits_{\mathcal{V}(\boldsymbol{\xi})}\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}_i'' f(\mathbf{v}_i'',\mathbf{x},t)\right]\frac{1}{\|\Omega\|}\sum_{i,\tilde{\omega}\in\Omega}\delta(\mathbf{v}-\mathbf{V}_\xi-\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i''-\mathbf{V}_\xi])\\
&\qquad -\int\limits_{\mathcal{V}(\boldsymbol{\xi})}\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}_i\, f(\mathbf{v}_i,\mathbf{x},t)\right]\frac{1}{\|\Omega\|}\sum_{i,\tilde{\omega}\in\Omega}\delta(\mathbf{v}-\mathbf{v}_i)\Bigg\} \;\equiv\; I(f,f),
\end{aligned}$$

where we replaced the integrals $\int_{\mathcal{V}(\boldsymbol{\xi})}\mathrm{d}x_i\delta(\mathbf{x}-\mathbf{x}_i)=\int_{\mathcal{V}(\boldsymbol{\xi})}\mathrm{d}x$.

Under these conditions we calculate the changes of the expectation values of the moments $\mathcal{J}_j(\mathbf{v}), j\in\{1,2,3\}$ of the velocity distribution function f, due to the collision integral. We multiply (2.27) by each $\mathcal{J}_j(\mathbf{v})\in\{\mathbf{1},\mathbf{v},\mathbf{V}^2\}$ individually and integrate over the velocity $\mathbf{v}$

$$\left\langle\frac{\Delta\mathcal{J}_j}{\Delta t}\right\rangle(\boldsymbol{\xi}) \;=\; \int\mathrm{d}\mathbf{v}\; I(f,f)\;\mathcal{J}_j(\mathbf{v}), \qquad j\in\{1,2,3\}. \tag{2.28}$$

To proceed, we change the notation in the first term of Eq. (2.27) so that both terms can be written under the same integral. Plugging Eq. (2.27) into Eq. (2.28) and reordering terms, one obtains

$$\begin{aligned}
\left\langle\frac{\Delta\mathcal{J}_j}{\Delta t}\right\rangle(\boldsymbol{\xi}) \;=\;& \int\mathrm{d}\mathbf{v}\frac{1}{\Delta t}\sum_{n=1}^{N}\frac{e^{-n_\xi}}{n!}\int\limits_{\mathcal{V}(\boldsymbol{\xi})}\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}_i f(\mathbf{v}_i,\mathbf{x},t)\right] \qquad (2.29)\\
&\times\frac{1}{\|\Omega\|}\sum_{\tilde{\omega}\in\Omega}\sum_{i=1}^{n}\Bigg[\delta(\mathbf{v}-\mathbf{V}_\xi-\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i-\mathbf{V}_\xi])-\delta(\mathbf{v}-\mathbf{v}_i)\Bigg]\mathcal{J}_j(\mathbf{v}).
\end{aligned}$$

The collision rule conserves local density $\mathbf{1}\hat{=}\mathcal{J}_1(\mathbf{v})$ and local momentum $\mathbf{v}\hat{=}\mathcal{J}_2(\mathbf{v})$ independent of the rotation $\tilde{\omega}\in\Omega$ and $\epsilon_{\xi,\mathrm{eff}}$, in the sum over all n particles participating in the collision. This summation occurs in the last sum over i in Eq. (2.29). To evaluate this sum we move the integration over $\int\mathrm{d}\mathbf{v}$ under the sum over i. For the density $\mathcal{J}_1=\mathbf{1}$ we obtain

$$\begin{aligned}
\sum_{i=1}^{n}\int\mathrm{d}\mathbf{v}\Big[\ldots\Big]\mathcal{J}_1(\mathbf{v}) \;&=\; \sum_{i=1}^{n}\int\mathrm{d}\mathbf{v}\Big[\ldots\Big]\mathbf{1} \qquad\qquad (2.30)\\
&=\;\sum_{i=1}^{n}\int\mathrm{d}\mathbf{v}\Bigg[\delta(\mathbf{v}-\{\underbrace{\mathbf{V}_\xi+\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i-\mathbf{V}_\xi]}_{\neq\mathbf{v}_i}\}\;)-\delta(\mathbf{v}-\mathbf{v}_i)\Bigg] \;=\; 0.
\end{aligned}$$

For the momentum this procedure, $\forall\tilde{\omega},\epsilon_{\xi,\mathrm{eff}}$, yields

$$\sum_{i=1}^{n}\int\mathrm{d}\mathbf{v}\Big[\ldots\Big]\mathbf{v} \;=\; \sum_{i=1}^{n}\Bigg[\mathbf{V}_\xi+\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)\underbrace{[\mathbf{v}_i-\mathbf{V}_\xi]}_{\sum_i(\ldots)=0}-\underbrace{\mathbf{v}_i}_{\sum\mathbf{v}_i=\mathbf{V}_\xi}\Bigg] \;=\; 0. \tag{2.31}$$

Consequently, from Eq. (2.30) and (2.31) follows that for $j = 1, 2$

$$\int \mathrm{d}\mathbf{v}\; I(f,f)\; \mathcal{J}_j \;=\; 0, \tag{2.32}$$

as a consequence of the individual collisions' conservation laws.

With Eq. (2.32) we turn back to the dynamics of our macroscopic variables. In the inhomogeneous system collisions are not the only mechanism that changes the velocity distribution function. The transport through mass flux is included into the dynamics by

$$\underbrace{\left(\frac{\partial}{\partial t} + \mathbf{v}\cdot\nabla\right)}_{\text{convective derivative}} f(\mathbf{v},\mathbf{x},t) \;=\; I(f,f). \tag{2.33}$$

In order to obtain the equation describing the dynamics of the density, we multiply Eq. (2.33) by $J_1 = \mathbf{1}$ and integrate both sides
over $\mathbf{v}$ [10]. After standard manipulation, we can transform the left hand side to

$$\frac{\partial}{\partial t}\int \mathrm{d}\mathbf{v}\; f(\mathbf{v},\mathbf{x},t) + \nabla\cdot\int \mathrm{d}\mathbf{v}\; \mathbf{v}f(\mathbf{v},\mathbf{x},t) \;=\; \int \mathrm{d}\mathbf{v}\; I(f,f)\; \mathcal{J}_j.$$

Because of Eq. (2.32), *i.e.* collisions conserve mass, and by using definitions (2.24) and (2.25) we arrive at the continuity equation

$$\frac{\partial}{\partial t}\rho + \nabla\cdot(\rho\,\mathbf{u}) \;=\; 0. \tag{2.34}$$

For momentum the approach is similar [10]. We will perform calculations here since we also want to specifically look at momentum transport in the sheared system and we will encounter the stress tensor on our way.

The left of Eq. (2.33) for this case, after integrating over the velocity, becomes

$$\frac{\partial}{\partial t}\int \mathrm{d}\mathbf{v}\; m\mathbf{v}f + \int \mathrm{d}\mathbf{v}\; m\mathbf{v}\,(\mathbf{v}\cdot\nabla)f \;=\; \frac{\partial}{\partial t}\rho m\mathbf{u} + \nabla\cdot\int \mathrm{d}\mathbf{v}\; m\mathbf{v}\mathbf{v}f, \tag{2.35}$$

where again we use the short notation of $\mathbf{v}\mathbf{v} = \mathbf{v}\otimes\mathbf{v}$ for the dyadic product. The first term can be rewritten as

$$\frac{\partial}{\partial t}\rho m\mathbf{u} \;=\; \mathbf{u}\frac{\partial\rho m}{\partial t} + \rho m\frac{\partial\mathbf{u}}{\partial t} \;=\; \rho m\frac{\partial\mathbf{u}}{\partial t} - \mathbf{u}\nabla\cdot(\rho m\mathbf{u}), \tag{2.36}$$

where the continuity equation (2.34) has been used. By using the definition of the local velocity

$$\mathbf{V} \;\equiv\; \mathbf{v} - \mathbf{u} \tag{2.37}$$

the integral of the second summand of Eq. (2.35) transforms to

$$\begin{aligned}\int \mathrm{d}\mathbf{v}\; m\mathbf{v}\mathbf{v}f \;&=\; \int m(\mathbf{V}+\mathbf{u})(\mathbf{V}+\mathbf{u})f(\mathbf{v})\mathrm{d}\mathbf{v} && (2.38)\\ &=\; \int m\mathbf{V}\mathbf{V}f(\mathbf{v})\mathrm{d}\mathbf{v} + m\mathbf{u}\mathbf{u}\int f(\mathbf{v})\mathrm{d}\mathbf{v} + 2m\mathbf{u}\int \mathbf{V}f(\mathbf{v})\mathrm{d}\mathbf{v}. && (2.39)\end{aligned}$$

The last term in Eq. (2.39) vanishes due to

$$\int \mathbf{V} f(\mathbf{v})\mathrm{d}\mathbf{v} = \int \mathbf{v} f(\mathbf{v})\mathrm{d}\mathbf{v} - \mathbf{u}\int \mathbf{f}(\mathbf{v})\mathrm{d}\mathbf{v} = u - u = 0.$$

The first term is the kinetic definition of the distribution to the pressure tensor

$$\tilde{P}(\mathbf{r},t) \;=\; \int m\mathbf{V}\mathbf{V} f(\mathbf{v})\mathrm{d}\mathbf{v}.$$

Plugging back in these terms into Eq. (2.35), the second term may be further transformed to

$$\begin{aligned}\int \mathrm{d}\mathbf{v}\; m\mathbf{v}\mathbf{v}f \;&=\; \nabla\cdot\tilde{P} + \nabla\cdot m\mathbf{u}\mathbf{u}\rho \\ &=\; \nabla\cdot\tilde{P} + \mathbf{u}\nabla\cdot(m\rho\mathbf{u}) + (m\rho\mathbf{u}\cdot\nabla)\mathbf{u},\end{aligned}$$

so that the second term of Eq. (2.36) disappears and we arrive at the momentum transport equation

$$\frac{\partial\mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} + \frac{1}{\rho m}\nabla\cdot\tilde{P} \;=\; 0, \tag{2.40}$$

which is equal to the equation we have seen for real granular gases.

For the temperature, the situation in general is complex. Since the collisions do not conserve energy, the collision integral on the right hand side (RHS) of Eq. (2.33) does not vanish. Again we multiply both sides by the observable and integrate over the velocity. First, we focus on the sum over i as in Eq. (2.30).
With $\mathbf{v}' = \mathbf{V}_\xi + \tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i - \mathbf{V}_\xi]$ it reads

$$\begin{aligned}\sum_{i=1}^{n}\int \mathrm{d}\mathbf{v}\Big[\dots\Big]\|\mathbf{v}-\mathbf{u}\|^2 \;&=\; \sum_{i=1}^{n}\Big[\mathbf{v}_i'^2 + 2\mathbf{v}_i'\mathbf{u} + \mathbf{u}^2 - (\mathbf{v}_i^2 - 2\mathbf{v}_i\mathbf{u} + \mathbf{u}^2)\Big] \qquad (2.41)\\ &=\; \sum_{i=1}^{n}\Big[\mathbf{v}_i'^2 - \mathbf{v}_i^2,\Big]\end{aligned}$$

where in the last step, we have used momentum conservation (Eq. (2.8)), *i.e.*, $\sum_i^n v_i' = \sum_i^n v_i$. Plugging back in the definition of $\mathbf{v}'$ we obtain

$$\begin{aligned}\sum_{i=1}^{n}\Big[\mathbf{v}_i'^2 - \mathbf{v}_i^2,\Big] \;&=\; \sum_{i=1}^{n}\Big[\mathbf{V}_\xi^2 + \tilde{\omega}_\xi(\rho_\xi,\theta_\xi)\underbrace{[\mathbf{v}_i - \mathbf{V}_\xi]}_{\sum_i(\dots)=0}\mathbf{V}_\xi + \{\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i - \mathbf{V}_\xi]\}^2 - \mathbf{v}_i^2\Big]\\ &=\; \sum_{i=1}^{n}\Big[\{\tilde{\omega}_\xi(\rho_\xi,\theta_\xi)[\mathbf{v}_i - \mathbf{V}_\xi]\}^2 - (\mathbf{v}_i - \mathbf{V}_\xi)^2\Big]\\ &=\; \frac{3n}{m}\,[\,\theta_\xi(t+\Delta t) - \theta_\xi(t)\,] \qquad (2.42)\end{aligned}$$

Using Eq. (2.42) and (2.12) we can transform the integral on the RHS of Eq. (2.33) to

$$\begin{aligned}\int \mathrm{d}\mathbf{v}\; I(f,f)\; \mathcal{J}_3(\mathbf{v}) \;&=\; \int \mathrm{d}\mathbf{v}\; I(f,f)\, \|\mathbf{v}-\mathbf{u}\|^2 \;= \qquad (2.43)\\ -\frac{3}{m}\sum_{n=1}^{N}\frac{e^{-n_\xi}}{n!}\int\limits_{\mathcal{V}(\boldsymbol{\xi})}&\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}_i f(\mathbf{v}_i,\mathbf{x},t)\right]\frac{2\sigma^2 n^2\left(1-\epsilon^2\right)\theta_\xi(\mathbf{v}_1,\ldots,\mathbf{v}_n,t\,)^{3/2}}{3\mathcal{V}}.\end{aligned}$$

The cooling coefficient ζ is defined [10] via

$$-\,\zeta\rho T \;\equiv\; \frac{m}{3}\int \mathrm{d}\mathbf{v}\; \|\mathbf{v}-\mathbf{u}\|^2\; I(f,f). \qquad (2.44)$$

Combining equations (2.43) and (2.44) yields

$$\zeta \;=\; \frac{2(1-\epsilon^2)\,\sigma^2}{Tn_\xi}\sum_{n=1}^{N}\frac{e^{-n_\xi}}{n!}n^2\int\limits_{\mathcal{V}(\boldsymbol{\xi})}\left[\prod_{i=1}^{n}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}_i f(\mathbf{v}_i,\mathbf{x},t)\right]\theta_\xi(\mathbf{v}_1,\ldots,\mathbf{v}_n,t\,)^{3/2}. \qquad (2.45)$$

To solve this averages over the number of particles n and all $\mathbf{v}_i$, we need to perform approximations. We assumed gradients to be small so that the length scale of our observation neither density nor temperature varies strongly. For a sufficiently large number of particles we further expect temperature fluctuations to be independent of fluctuations of n. In other words, for a sufficiently large number of particles, the instantaneous temperature $\theta_\xi = \frac{m}{3n}\sum_{i=1}^{n}\|\mathbf{v}_i-\mathbf{V}_\xi\|^2$ is close to the expectation value of the summands. With $\theta_\xi \approx \frac{m}{3}\|\mathbf{v}-\mathbf{u}\|^2$, we may then write

$$\begin{aligned}\zeta \;&\approx\; \frac{2(1-\epsilon^2)\,\sigma^2}{}Tn_\xi\left\{\sum_{n=1}^{N}\frac{n^2 e^{-n_\xi}}{n!}\Bigg[\underbrace{\int_{\mathcal{V}(\boldsymbol{\xi})}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} f(\mathbf{v},\mathbf{x},t)}_{=n_\xi}\Bigg]^{n-1}\right\}\\ &\quad\times\frac{m}{3}\int_{\mathcal{V}(\boldsymbol{\xi})}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} f(\mathbf{v},\mathbf{x},t)\|\mathbf{v}-\mathbf{u}\|^{3/2}\\ &\overset{N\to\infty}{=}\; \frac{2m(1-\epsilon^2)\,\sigma^2(n_\xi+1)}{3Tn_\xi}\int_{\mathcal{V}(\boldsymbol{\xi})}\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} f(\mathbf{v},\mathbf{x},t)\|\mathbf{v}-\mathbf{u}\|^{3/2}. \qquad (2.46)\end{aligned}$$

This result is similar to the same result for a physical system. Dependencies of density and temperature match. In the factor $\frac{2}{3}(1-\epsilon^2)$ we recognize our dissipation rule, where the dissipation takes place in the two dimensions perpendicular to the random rotation axis. The integral in ζ becomes a constant by rescaling the velocities $\mathbf{c} = \mathbf{v}/\sqrt{T}^3$ and using the so-called Sonine expansion of f (refer to Chapter II.8 of [10]). This procedure yields one equation for the temperature and another for the shape of the rescaled velocity distribution function $\tilde{f}(c)$.

Similar to the derivation of the momentum transport equation one derives the heat transport equation [10]. Only here we obtain a contribution from the collision integral. With the above defined cooling coefficient ζ we arrive at the heat transport equation

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T + \frac{2}{3\rho} \left(\tilde{P} : \nabla \mathbf{u} + \nabla \cdot \mathbf{q} \right) + \zeta T \;=\; 0, \tag{2.47}$$

where the pressure tensor $\tilde{P}$ and the heat flux $\mathbf{q}$ follow from the left hand side (LHS) Eq. (2.33) and are thus the same as for a real granular gas.

2.4 GSRD streaming viscosity

To obtain an experssion for the granular streaming viscosity, we apply the same procedure of the calculation of the velocity correlator of standard SRD to arrive at a corresponding expression for granular SRD. We start with deriving Eq. (1.46), for the 2D system where velocities have the form $\mathbf{v} = (v_x, v_y)^T$ and again we use the short notation $\mathbf{v}''$ for the post-collision velocities

$$\mathbf{v}'' \;=\; \tilde{\omega}\mathbf{v} - \tilde{\omega}\mathbf{V}_\xi + \mathbf{V}_\xi. \tag{2.48}$$

We split the contribution to the center of mass velocity $\mathbf{V}_\xi$ into that of one test particle with velocity $\mathbf{v}$ and the contribution of the remaining $n-1$ particles participating in the collision. The sum of these are written as $\check{\mathbf{v}}$ so that

$$\mathbf{V}_\xi \;=\; \frac{\mathbf{v} + \check{\mathbf{v}}}{n} \tag{2.49}$$

and

$$\mathbf{v}'' \;=\; \tilde{\omega}\mathbf{v} - \frac{1}{n}\tilde{\omega}\mathbf{v} - \frac{1}{n}\tilde{\omega}\check{\mathbf{v}} + \frac{\check{\mathbf{v}}}{n} + \frac{\mathbf{v}}{n}. \tag{2.50}$$

We are interested in how the correlation of the two velocity components v_x and v_y changes during the collision. In order to arrive at an expression, we use the definition of the rotation matrix

$$\tilde{\omega} \;=\; \epsilon_{\xi,\mathrm{eff}} \begin{pmatrix} \cos(\alpha) & \pm\sin(\alpha) \\ \mp\sin(\alpha) & \cos(\alpha) \end{pmatrix}. \tag{2.51}$$

We obtain

$$v_x'' \;=\; \epsilon_{\xi,\mathrm{eff}} \cos(\alpha) \left(v_x - \frac{v_x}{n} - \frac{\check{v}_x}{n} \right) \pm \epsilon_{\xi,\mathrm{eff}} \sin(\alpha) \left(v_y - \frac{v_y}{n} - \frac{\check{v}_y}{n} \right) + \frac{v_x}{n} + \frac{\check{v}_x}{n}$$

and

$$v_y'' \;=\; \mp\epsilon_{\xi,\mathrm{eff}} \sin(\alpha) \left(v_x - \frac{v_x}{n} - \frac{\check{v}_x}{n} \right) + \epsilon_{\xi,\mathrm{eff}} \cos(\alpha) \left(v_y - \frac{v_y}{n} - \frac{\check{v}_y}{n} \right) + \frac{v_y}{n} + \frac{\check{v}_y}{n}.$$

For further simplifications we assume molecular chaos, *i.e.* that the velocities of different molecules are uncorrelated. Hence $\langle \check{v}_x v_y \rangle = \langle \check{v}_y v_x \rangle = 0$.

It also follows that $\langle \check{v}_x \check{v}_y \rangle = (n-1)\langle v_x v_y \rangle$ which proves to be very useful in the following simplifications, because we eventually want to arrive at an expression for $\langle v''_x v''_y \rangle$. The following terms created by the multiplication of the sine and cosine term with the last two terms of each of the velocities' components disappear with this assumption. First

$$\left\langle \left(v_x - \frac{v_x}{n} - \frac{\check{v}_x}{n} \right) \left(\frac{v_y}{n} + \frac{\check{v}_y}{n} \right) \right\rangle \;=\; \left\langle \left(v_x - \frac{v_x}{n} \right) \frac{v_y}{n} \right\rangle - \left\langle \frac{\check{v}_x}{n} \frac{\check{v}_y}{n} \right\rangle \;=\; 0,$$

and secondly for both x and y component

$$\left\langle \left(v_y - \frac{v_y}{n} - \frac{\check{v}_y}{n} \right) \left(\frac{v_y}{n} + \frac{\check{v}_y}{n} \right) \right\rangle \;=\; \left\langle \left(v_y - \frac{v_y}{n} \right) \frac{v_y}{n} \right\rangle - \left\langle \frac{\check{v}_y}{n} \frac{\check{v}_y}{n} \right\rangle \;=\; 0.$$

Also the terms that contain the product $\sin(\alpha)\cos(\alpha)$ vanish due to the different sign, so that the rotation axis or direction does not play a role. If then we multiply the velocities' components and average we arrive at

$$\langle v''_x v''_y \rangle \;=\; \epsilon^2_{\xi,\mathrm{eff}} \left[\cos^2(\alpha) - \sin^2(\alpha) \right] \left\langle \left(v_y - \frac{v_y}{n} - \frac{\check{v}_y}{n} \right) \left(v_x - \frac{v_x}{n} - \frac{\check{v}_x}{n} \right) \right\rangle + \left\langle \frac{v_x v_y}{n} \right\rangle .$$

or

$$\begin{aligned} \langle v''_x v''_y \rangle \;&= \epsilon^2_{\xi,\mathrm{eff}} \frac{n-1}{n} \cos(2\alpha) \, \langle v_y v_y \rangle + \frac{\langle v_x v_y \rangle}{n} \\ &= \; \left[1 - \frac{n-1}{n} (1 - \epsilon^2_{\xi,\mathrm{eff}} \cos(2\alpha)) \right] \langle v_x v_y \rangle \end{aligned} \qquad (2.52)$$

This result does not yet include fluctuations in the number of particles. Before we take them into account, we need to consider that also the effective coefficient of restitution depends on the number of particles. If we define

$$\kappa \;\equiv\; (1-\epsilon^2)\sigma^2 \Delta t \sqrt{\theta_\xi}, \qquad (2.53)$$

the squared $\epsilon_{\xi,\mathrm{eff}}$ becomes $\epsilon^2_{\xi,\mathrm{eff}} = 1 - 2n\kappa + n^2\kappa^2$. The probability of finding n particles in a cell follows the Poisson distribution $P(n) = e^{-n_0} n_0^n / n!$ for the homogeneous system n_0, the probability to find the test particle in a cell with $n-1$ other particles is $nP(n)/n_0$. Hence, to account for fluctuations in the number of particles we rewrite Eq. (2.52) as

$$\begin{aligned} \langle \langle v''_x v''_y \rangle \rangle_n \;&=\; \sum_{n=1}^{\infty} \frac{nP(n)}{n_0} \left[1 - \frac{n-1}{n} (1 - \cos(2\alpha)) - (n-1) 2\kappa \cos(2\alpha) \right. \\ &\qquad \left. + (n^2 - n)\kappa^2 \cos(2\alpha) \right] \langle v_x v_y \rangle \\ &= \left[1 - \frac{n_0 - 1 + e^{-n_0}}{n_0} (1 - \cos(2\alpha)) \right. \\ &\qquad \left. - (2n_0^2 \kappa + (n_0^2 - 2n_0)\kappa^2) \cos(2\alpha) \right] \langle v_x v_y \rangle \;\equiv\; h(\alpha, n) \langle v_x v_y \rangle \, . \end{aligned} \qquad (2.54)$$

We now know that there are two mechanisms acting on the correlation. The streaming step produces an increasing anti-correlation between the velocities' components as can be seen in Eq. (1.45). The collision step reduces the correlations by the factor of $h(\alpha, n)$. Hence, the resulting value strike a balance between these two mechanisms. We thus require self-consistency of Eq. (2.54 and (1.45) yielding Eq. (1.47). With the formula of the component of the stress tensor (1.44) we obtain the viscosity via

$$\eta^{\mathrm{2D}}_{\mathrm{stream}} = \frac{\rho_0 \theta_\xi \Delta t}{a^2} \left(\frac{h}{1-h} + \frac{1}{2} \right),$$

and hence

$$\begin{aligned} &\eta^{\mathrm{2D}}_{\mathrm{stream}} = \\ &\frac{\rho_0 \theta_\xi \Delta t}{a^2} \left\{ \frac{n_0}{(n_0 - 1 + e^{-n_0})[1 - \cos(2\alpha)] - [2 n_0 \kappa + (\rho_0 - 2)\kappa^2] \cos(2\alpha)} - \frac{1}{2} \right\}. \end{aligned} \tag{2.55}$$

We will test the accuracy of this result in Chap. 3.

2.5 Numerical Implementation

2.5.1 General purpose graphics processing units

We utilize GPUs to speed up the SRD simulations by parallelization compared to sequential execution on a central processing unit (CPU). GPUs are capable of performing the same operation on elements of a data array at the same time. As the name tells, the main application of GPU used to be as graphics accelerator. Though with improvements and increasing availability graphic cards are now used for general-purpose computing on graphic processing unit (GPGPU). We use a language extension to C/C++ which is provided by Nvidia for their GPUs, called compute unified device architecture (CUDA). With CUDA functions that are executable on GPUs may be written in C/C++. Usually these GPU functions are called kernels to distinguish them from functions for the CPU. The main features of the CUDA extension is a new syntax to execute these kernels on the GPU. To achieve performance benefits, it is essential to tailor the algorithmic implementation to the hardware. There are a number of features that make programming on systems with a GPU different (*c.f.* Fig. 2.2). GPUs have separate dedicated RAM with their own memory address space. The path between main RAM and GPU RAM generally is a computational bottleneck. For this reason memory transfer is kept at a minimum and hence almost all workload handled is on the GPU with the CPU acting as a manager and bookkeeper.

2.5.2 Details of the hardware-tailored implementation

In order to gain the most parallelization speedup, computational problems have to be examined in terms of their interdependence and size. Dependencies have to be solved sequentially.

The simplest example occurring in our system is the translation and rotation of particles. Given that the rotation axes and the streaming velocities are precalculated, there is no need for communication between threads updating positions and velocities of single particles. Problems of this type are called 'embarrassingly parallel' and, as the name suggest easy to parallelise.

Another problem occurring are sums of a species of values stored in an array. Here, the final result depends on all data elements. This operation (trivial on a CPU) becomes a bit problematic in a parallel environment where different threads might attempt to access the same memory location at the same time. Inconsistent results might arise.

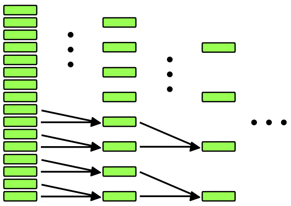


Figure 2.1: Parallel reduction via recursion, for summations, logical evaluation or similar. For an ideal parallel hardware with parallel capabilities as numerous as the data, the runtime becomes $\mathcal{O}(\log N)$ for input data N.

This problem is effectively resolved by using a recursive sum such that at each recursion step the size of the problem is halved. Each thread in one step adds up two successive values; at the next step the number of items left to sum is halved, and so half the number of threads are necessary at this time step.

This procedure yields a computational complexity of $\mathcal{O}(\log N)$ with the problem size N. We encounter this problem for example in the calculation of the global drift velocity, or the pressure tensor. The present GPUs can provide a maximum number of 1024 threads grouped into a block. Threads in blocks can communicate. Communication is the most efficient within one of the 14 streaming multiprocessors, so with blocks of 192 threads each. Only for recursive sums bigger block sizes are beneficial.

For the rotation step the quantities ρ_ξ, $\mathbf{V}_\xi$ and θ_ξ need to be calculated at each time step. To tailor our problem to our architecture we organize the calculations in the following way. In any SRD cell $\boldsymbol{\xi}$, we provide an array of 14 elements for pre-results. We first start a function kernel using 14 blocks with block sizes of 192 threads each. Inside the blocks summations of the pre-results, *e.g.* incrementing one element $\rho_{\xi,i}$ dedicated for the respective block i, is performed using atomic operations. These are operations that guarantee exclusive access to a memory address and are hence essential for parallel computing. This is because today's processors' instructions are divided into multiple stages. These are steamed and may generally be executed out of order. In an atomic operation the multiple stages of an increment, *i.e.* simplified fetch, execute and write back, are guaranteed to be completed before another thread may issue the next fetch on this memory element, so that the up-to-date state of the memory is read.

This synchronization capability is only possible within one streaming multiprocessor, hence we use 14 pre-results and let the blocks process subsets of the whole data. This is an operation concerned with particle properties. Next, we need to address issues at the SRD cell level, therefore less parallel in the sense of a less number of elements. We issue small blocks and take advantage of a hardware function called 'warp shuffle'. Threads are by designed grouped into warps consisting of up to 32 threads. Within a warp these 'shuffle' can be used. It shifts the value of a local variable between threads and hence provides a hardware function for recursive summation which we use here. So, by combining functions for reduction of many cells and within a cell we sum velocities and number density and pre-calculate the rotation axes. With the now known streaming velocities we use a second two-stage reduction to calculate the cell temperatures. Lastly we also compute the effective coefficient of restitution for each cell.

For analyzing the data, there are additional functions that calculate the local properties on a non-shifted grid. To minimize data transfer we average these cell-grid based observables already in the GPU memory space. To efficiently handle memory on the GPU we use the dedicated constant cache structure. Constants can be stored there as read only data. Accessing these data is thus much faster. We mainly store pointers to data as constants, beside some constants.

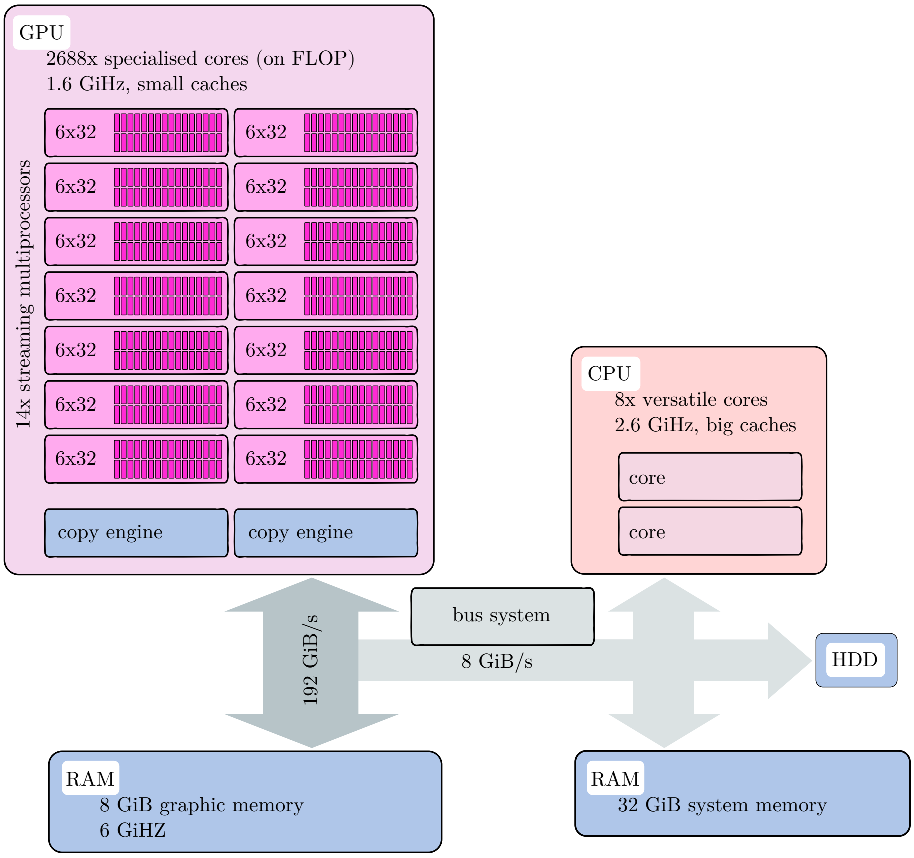


Figure 2.2: Hardware sketch of the used computers. Those feature one Nvidia K20xm GPU with separate graphic RAM, and an octo-core CPU with the system RAM and hard drive storage. GPUs are optimal for simple arithmetic computations and execute those in a highly parallel fashion and thus much faster even though its lower clock rate. In contrast, CPUs are more versatile and thus faster on non-parallelizable tasks. In terms of memory there are several layers. The CPU RAM is effectively infinite in our perspective, just as hard disk drive (HDD) space. Because the SRD is computationally cheap its memory usage becomes more important. As usual for GPGPU applications the interconnection between GPU and CPU is a bottleneck. Here, the connection of the GPU to its RAM becomes a computational bottleneck, too. Parallelism is possible at different stages. The GPU provides two copy engines that can handle data transfer with the CPU asynchronously from computations. On the CPU this can be done, too, using different cores. So if grid based quantities are of interest for writing to disk, the CPU can assign workload, while a second CPU core waits for the asynchronous data transfer to finish. At the same time the GPU can update particle velocities. More precisely said, the GPU combines 14 streaming multiprocessors with 192 cores each. For a more detailed description on intra-GPU concurrency and synchronization abilities please refer to the main text. (The K40 GPU feature 16 streaming multiprocessors and a RAM size of 12 GiB.)

2.5.3 Algorithm summary

We use object oriented programming to benefit from high code re-usability an clarity. The procedure starts with interpreting an input file. (*cf.* Fig. 2.3(a)). Next, an instance of the `simulation_box` class is created (refer to next section). Then, steps and writing data to disk functions are called following the simulation protocol of the input file. A step combines many function calls, one of these are the `step_internal` function of the `simulation_box` object which combines all CUDA kernel calls (Fig. 2.3(b)).

The implementation of the SRD algorithm consists of the following steps

$$\begin{aligned} \mathbf{v}_i^{\mathrm{n}}(t) &= \hat{\mathbf{r}}_\xi(\hat{\mathbf{r}}_\xi \cdot \mathbf{v}_i(t)) \\ \mathbf{v}_i^{\mathrm{t}}(t) &= \mathbf{v}_i(t) - \mathbf{v}_i^{\mathrm{n}}(t) \\ \mathbf{v}_i(t+\Delta t) &= \mathbf{v}_i^{\mathrm{n}}(t) + \epsilon_{\xi,\mathrm{eff}}[\cos(\alpha)\mathbf{v}_i^{\mathrm{t}}(t) + \sin(\alpha)(\mathbf{v}_i^{\mathrm{t}}(t) \times \hat{\mathbf{r}}_\xi)], \end{aligned}$$

where $\hat{\mathbf{r}}_\xi$ is the random rotation axis of cell $\boldsymbol{\xi}$, analogous to the matrix multiplication in Eq. (2.5).

2.5.4 Data structuring

We store physical quantities combined in a struct we call `parameters`. The main function uses an object of the struct `simulation` to hold details of the protocol. The physical space is represented as a class `simulation_box` instance. It has members holding particles' positions, velocities, SRD grid based data, plus counters and buffers. We bundle memory transfers between the different RAMs using the class `memory_manager` working on the data member of the `async_array` objects.

For vector calculus we implement a struct `vektor` with member functions dedicated to the certain operations. For the SRD local grid data there is a struct `srd_cell` holding relevant structure. This approach has the advantage that adding a new parameter passed by the input file is really simple and non-error prone. One can just create a new member in the struct `parameters` while data copying and management of the object remains the same. The same applies for the struct `srd_cell`. Introducing the local effective coefficient of restitution was implemented by simply adding a new member to the `srd_cell`.

The data that we store in the GPU's constant cache, as mentioned earlier, are mostly member pointers of the simulation_box instance. For their variable naming we simply add an underscore to the member name to which the pointer references.

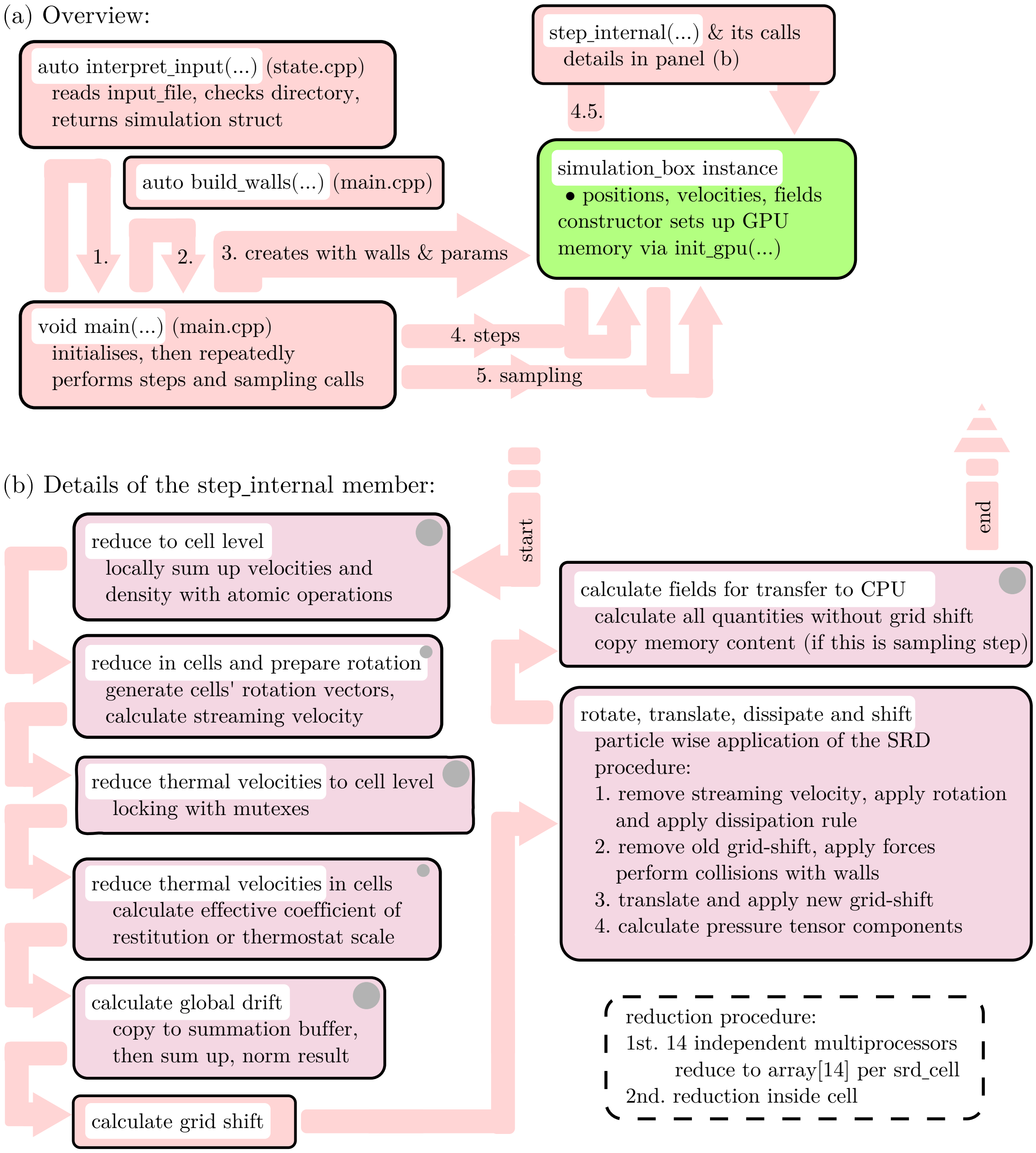
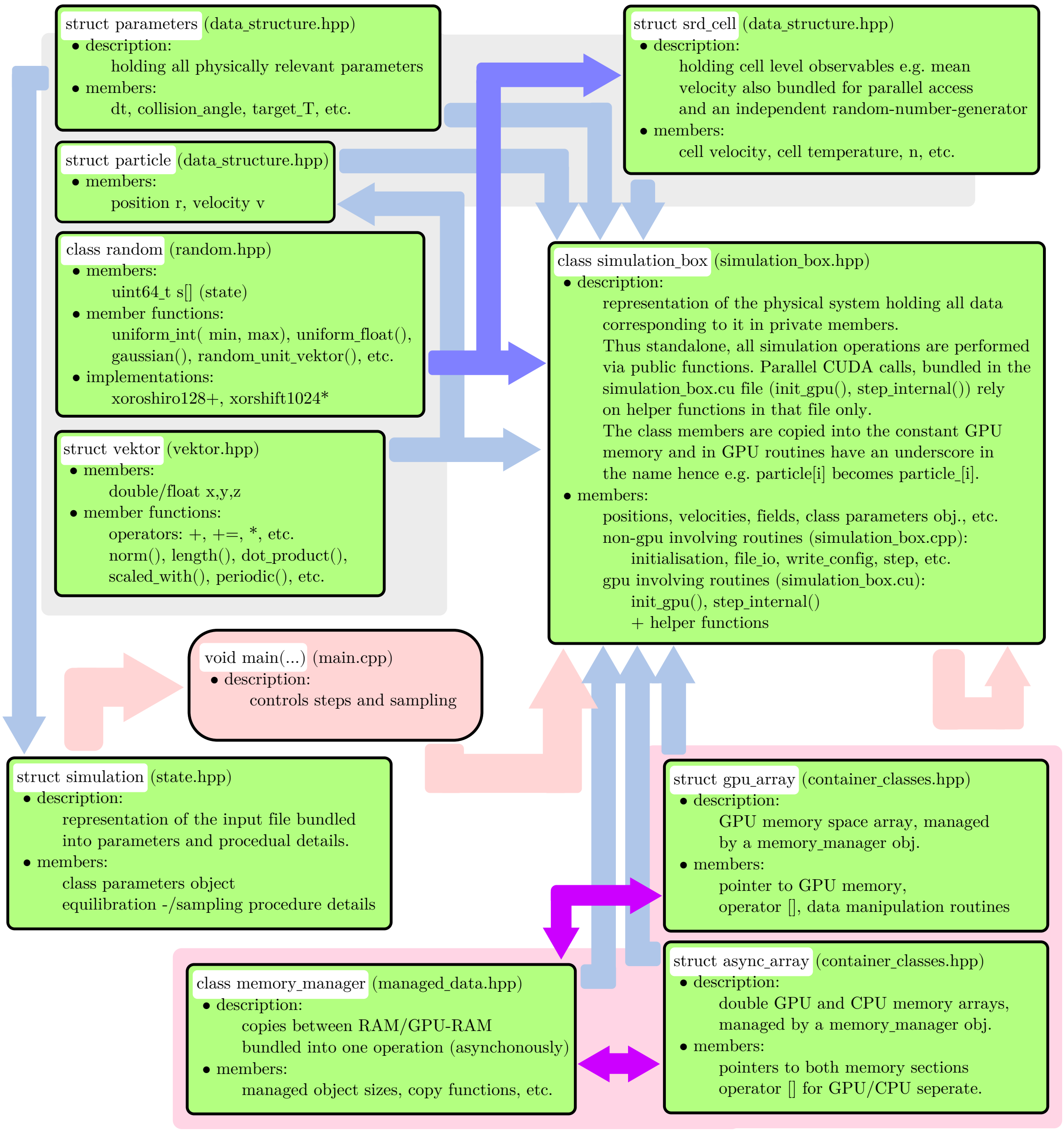


Figure 2.3: Code flowchart divided into an overview in panel (a) and details of the `step_internal(...)` member of the `simulation_box class` in panel (b). Functions and workflow are represented by red cards and red arrows respectively. Parallel CUDA-GPU-functions are represented by purple cards. The central `simulation_box` instance appears in green. All CUDA calls are performed in the `init_gpu(...)` and `step_internal(...)` members. Besides the rotation-translation routine every CUDA-helper-function mostly performs only one task. Functions with a gray dot need synchronization. CPU and GPU compute times are superimposed, also with data transfer. CUDA function calls are launched with individual blocks and grid sizes regarding synchronization necessities of the respective code segment. Thread numbers are chosen to fit processor number so that functions perform loops over subsets of the workload.



struct parameters (data_structure.hpp)
- description:
  holding all physically relevant parameters
- members:
  dt, collision_angle, target_T, etc.

struct particle (data_structure.hpp)
- members:
  position r, velocity v

class random (random.hpp)
- members:
  uint64_t s[] (state)
- member functions:
  uniform_int(min, max), uniform_float(), gaussian(), random_unit_vektor(), etc.
- implementations:
  xoroshiro128+, xorshift1024*

struct vektor (vektor.hpp)
- members:
  double/float x,y,z
- member functions:
  operators: +, +=, *, etc.
  norm(), length(), dot_product(), scaled_with(), periodic(), etc.

struct srd_cell (data_structure.hpp)
- description:
  holding cell level observables e.g. mean velocity also bundled for parallel access and an independent random-number-generator
- members:
  cell velocity, cell temperature, n, etc.

class simulation_box (simulation_box.hpp)
- description:
  representation of the physical system holding all data corresponding to it in private members.
  Thus standalone, all simulation operations are performed via public functions. Parallel CUDA calls, bundled in the simulation_box.cu file (init_gpu(), step_internal()) rely on helper functions in that file only.
  The class members are copied into the constant GPU memory and in GPU routines have an underscore in the name hence e.g. particle[i] becomes particle_[i].
- members:
  positions, velocities, fields, class parameters obj., etc.
  non-gpu involving routines (simulation_box.cpp):
    initialisation, file_io, write_config, step, etc.
  gpu involving routines (simulation_box.cu):
    init_gpu(), step_internal()
    + helper functions

void main(...) (main.cpp)
- description:
  controls steps and sampling

struct simulation (state.hpp)
- description:
  representation of the input file bundled into parameters and procedual details.
- members:
  class parameters object
  equilibration -/sampling procedure details

struct gpu_array (container_classes.hpp)
- description:
  GPU memory space array, managed by a memory_manager obj.
- members:
  pointer to GPU memory,
  operator [], data manipulation routines

class memory_manager (managed_data.hpp)
- description:
  copies between RAM/GPU-RAM bundled into one operation (asynchonously)
- members:
  managed object sizes, copy functions, etc.

struct async_array (container_classes.hpp)
- description:
  double GPU and CPU memory arrays, managed by a memory_manager obj.
- members:
  pointers to both memory sections
  operator [] for GPU/CPU seperate.

Figure 2.4: Object structure and summarized object descriptions of the implemented program. Objects are denoted as green cards, membership usage between objects is symbolized as blue arrows, strongly linked classes are connected with purple arrows. Routines are presented as red cards, control flows as red arrows. Classes are grouped into physics representatives (on gray) and memory managing objects (on pink). The center of the simulation is the `simulation_box` class around which everything is built, the most basic and widely used instance is of the struct `vektor`. After the main function requested a simulation details struct it generates a `simulation_box` instance and calls its public control functions. The class itself handles the details in dedicated private functions, including *e.g.* simulation steps with GPU-addressing and data output.

2.5.5 Random number generators

Artificial pseudo-random numbers are computer-generated numbers using certain algorithms that together with the internal state of the generator are called random number generator. Generating a random number alters the internal state of the generator according to the algorithm. Since the internal state has a binary memory representation, it is simple to recognize that this leads to a finite length sequence of numbers. Measures of the quality of a random number generator may be the length of the period of its sequence or how much subsets resemble uniform distribution of values. It is commonly agreed that the SRD method does not need the statistically best random number generator since only few numbers are necessary. In contrast to that, the MPCD method uses at least $3n$ random numbers per box with n particles and collision steps and thus the quality of the generator is more critical.

For the present implementation we chose a rather new sort of generators of the xorshift family [22] in an exchangeable way to test them against each other. One has a significantly longer period, but the use of a shorter period does not change the simulations.

For the implementation on GPUs we need to guarantee that the random numbers are independently generated. If different threads generate the rotation axes for different SRD cells at the same time, this may cause problems. Instead of providing exclusive access to one common generator we provide independent generators for each SRD cell so that no communication between threads is necessary. For the used xorshift generators so-called 'jump ahead' functions are known, that alter the generators internal state equally to calling a certain high number of single random number generation calls. This gives a number of independent sub-sequences that are safe to call in parallel.

2.5.6 Controlling and using the code

The compiled code is fully flexible in the sense that all of its functions can be switched on and off at runtime. For the appropriate settings, the code uses an input file to read all input parameters, which may be specified by the first input string given to the executable. Hence, there are no pre-compiler definitions (hash-defines), instead consistent declaration of variables with the `const` flag is performed.

The code generates a logfile in ASCII to which it writes *e.g.* the temperature or flux. Specific details are given in the `simulation_box::write_log()` function definition. Structural data is written in binary at specified times.

CHAPTER 3

Results

3.1 Simulation parameters

We perform our simulation using dimensionless equations, where all quantities are given as multiples of the respective characteristic values of the system. Lengths are multiples of the SRD lattice parameter a, mass as multiple of the (identical) SRD/GSRD particle mass m. The time unit is given via a reference thermal energy

$$t_{\mathrm{unit}} = \frac{a}{\sqrt{k_{\mathrm{B}}T_{\mathrm{ref}}/m}},$$

so that a particle moving with velocity $v = \sqrt{k_{\mathrm{B}}T_{\mathrm{ref}}/m}$ at $k_{\mathrm{B}}T_{\mathrm{ref}} = 1$ covers the length a per unit of time. During the initialization of the system, particles are randomly distributed in the system. The mean number of particles per cell n_0 is an important quantity since it controls for instance the viscosity of the fluid.

Other parameters that control the transport and relaxation processes are the SRD rotation angle α, and the mean collision time Δt.

3.2 Code validation via streaming viscosity measurements

Equations (1.48) and (1.49) give analytic predictions for the viscosities of the standard SRD fluid in two and three dimensions [3], respectively, and show an excellent agreement with the results obtained from simulations. Hence, to test our newly developed and implemented code we may also use this comparison to test, verify and benchmark our implementation of the standard SRD model.

We simulate systems of sizes 32×32 in 2D and $32 \times 32 \times 32$ in 3D and initialize each system with on average $n_0 = 5$ particles per cell such that the system in total contains n_0 particles times the number of cells, *i.e.* $N = 5120$ in 2D and $N = 163840$

in 3D. We apply a shear rate of $\dot{\gamma}\Delta t = 2/32 = 0.0625$ at a reduced temperature of $T/T_{\text{ref}} = 0.8$ and $\Delta t = 1$.

The particle velocities are initialized according to the Maxwell-Boltzmann distribution. The velocity profile relaxes to a linear profile induced by the shear quickly. For this reason we equilibrate the system for 10000 time steps. In the following 50000 time steps we measure the flux of momentum every step and write the average to disk every 100 time steps. Configurations are internally averaged on the GPU over 100 states each separated by 10 Δt. During the simulation the density fluctuates around n_0 as can be seen in Fig. 3.1(a), just like we expect. The system remains homogeneous during the whole simulated timespan. As we can see in Fig. 3.1(b), a smooth velocity profile develops. Moreover, we can still see slight density fluctuations which completely agree with our expectations. The SRD rotation angle may be understood as representing an effective mean free path of the fluid, and in that regard similar to the time between multi-particle collisions Δt. This is because it controls how far a particle travels without significantly changing its direction. It also controls how quickly the system locally relaxes to equilibrium. Because of this interesting feature we investigate the streaming viscosity as a function of the rotation angle α here. For the 2D system the streaming viscosity shown in Fig. 3.2(a) naturally diverges as α approaches 0° and because of the geometry also approaching 180°. At a rotation angle of 90° there is a global minimum because the parameter limits for α are 0° and 180°. The computed viscosity is in excellent agreement with the theoretical prediction. We see that because there is a continuous heat production due to the shearing the system settles at higher temperature the higher the viscosity is, even though the thermostat constantly rescales the temperature.

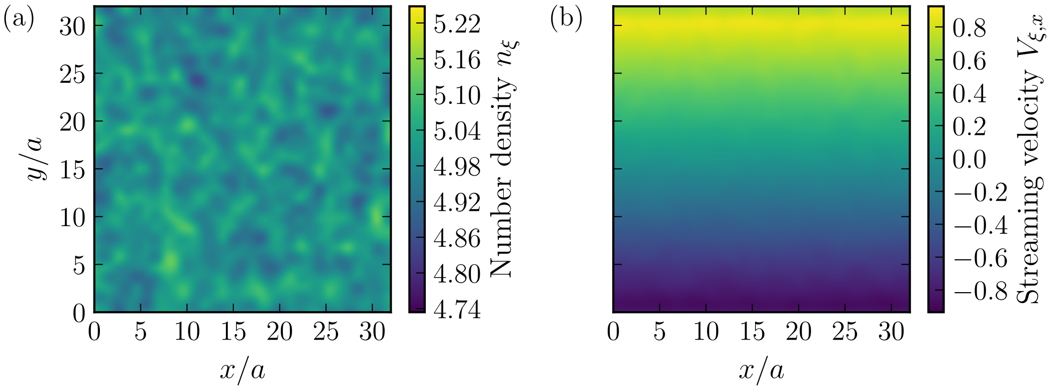


Figure 3.1: Snapshot of the sheared system averaged over 1000 states for a rotation angle $\alpha = 120°$. Panel (a) shows the density in the x/y plane. Since we look at an average, the density does not need to be in integer numbers. Still we see fluctuations, also in panel (b) that shows the cells' streaming velocities $V_{\xi,x}$ in the direction along which the system is sheared.

For the 3D system the viscosity also diverges at $\alpha = 0°$ but not at $\alpha = 180°$ where instead we have a local maximum. The minimum now occurs at $\alpha \approx 11\pi/18$.

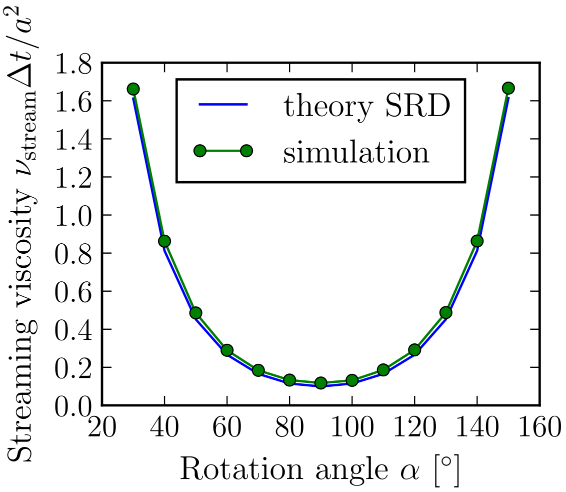


Figure 3.2: Kinetic viscosity $\nu_{\text{stream}} = \eta_{\text{stream}}/n_0$ viscosity of the 2D SRD fluid as a function of the SRD rotation angle α.

Apart from this, the behavior is similar to that of the 2D system. Also here, the streaming viscosity measurement via the momentum transport in the simulations and the theoretical predictions are in perfect agreement.

We performed both of the preceding numerical simulations with the same code that we will use in the next Section for the simulation of granular system in the limit of fully elastic collisions, *i.e.*, by setting $\epsilon = 1$, with the single addition of a thermostat. Hence, with the reproduction of the excellent agreement between theory and simulations we conclude that our implemented code is working properly in the limit of standard SRD for $\epsilon = 1$.

To conclude this section, we sum up a few more properties of the viscosity. The viscosity as a function of other parameters [3] behaves as follows. In terms of the number of particles, both streaming and collisional viscosity increase linearly with the number of particles, and the kinetic contribution is typically an order of magnitude higher than the collisional. The collisional viscosity which we have not discussed so far approaches zero as the rotation angle α goes to zero.

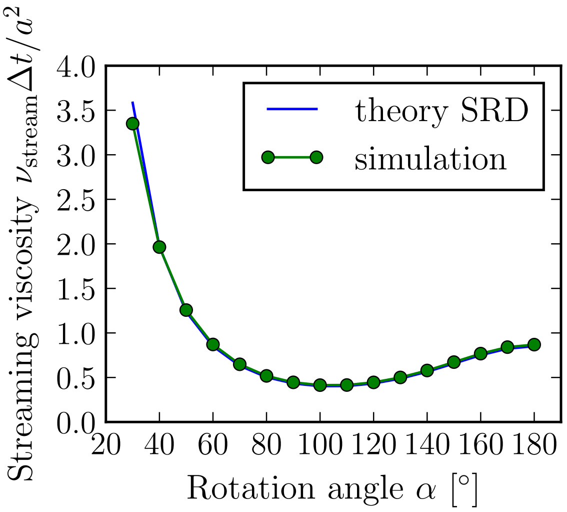


Figure 3.3: Kinematic viscosity $\nu_{\mathrm{stream}} = \eta_{\mathrm{stream}}/n_0$ of the 3D SRD fluid as a function of the SRD rotation angle α. The streaming viscosity diverges at zero and reaches a local maximum at 180°.

3.3 Granular SRD - Haff's law

We now turn to our model of granular fluids, and discuss our results that validate our physical modelling and algorithm. Haff's law is valid for a system in the homogeneous cooling state [11]. Hence, we begin with looking at a system that cools down while remaining in a rather homogeneous state. This means that we choose rather low dissipation with $\epsilon = 0.995$. We simulate a 3D system with $120 \times 120 \times 120$ boxes with mean initial density $\rho_0 = 25/\mathcal{V}$ particles per GSRD cell, hence in total $N = 43.2 \times 10^6$ particles. For the freely cooling simulations we use periodic boundaries in all three dimensions. Granular SRD is more realistic for smaller rotation angles α, hence we chose $\alpha = 20°$ for the present simulation. To obtain features on the length scale of our observations, we initialize the system at a temperature of $T(t_0)/T_{\text{ref}} = 100$ and fix the collision rate at $\Delta t = 0.01$.

Figure 3.4 shows that even after a very long simulation time, the system is rather homogeneous. The spatial variations of density and temperature are never larger than twice thrice the average density and density, respectively. This is still a rather homogeneous state compared to the inhomogeneous cooling state which is clearly characterized by inhomogeneities spanning over several orders of magnitude, as we will see in the following.

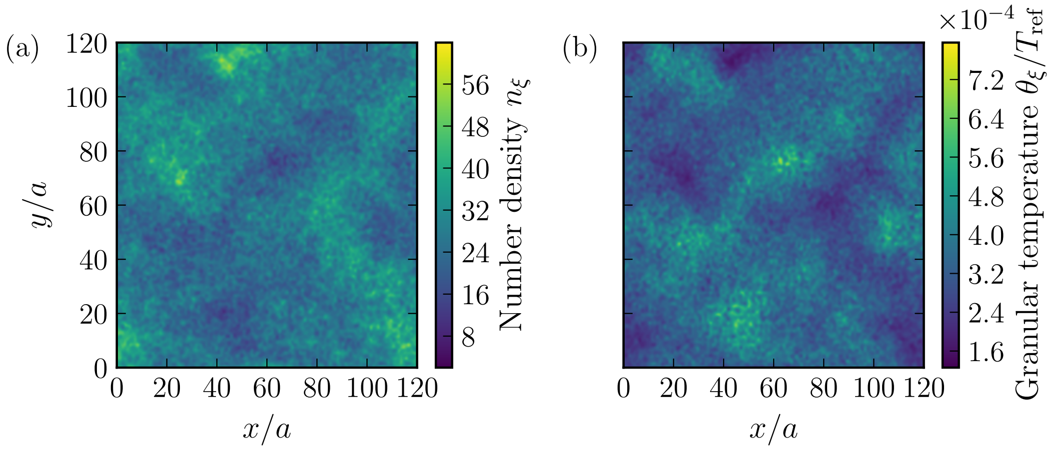


Figure 3.4: State of the freely cooling 3D GSRD gas with $\epsilon = 0.995$ after $\sim 10^5$ collision and streaming steps. We show a cross section perpendicular to the z-axis. In panel (a) we see the instantaneous number density in the non-shifted grid linearly color encoded. In panel (b) we see the instantaneous cell temperature. The system is not perfectly homogeneous anymore though we can see that the variations extent over only one order of magnitude which is actually small. Furthermore, the denser regions are colder than the dilute regions due to their higher $\epsilon_{\xi,\text{eff}}$ mimicking the increased collision rate in those regions.

From Fig. 3.4, we see that the initial fluctuations in the distribution of mass and temperature in the system lead to regions where particles accumulate and then cool down slightly faster than in the less dense regions. The cooling due to dissipative collisions can also be seen in the anti-correlation of density and temperature in Fig. 3.4(a) and (b). This is the result of the effective coefficient of restitution. Eventually, this system will undergo the transition to the so-called inhomogeneous cooling state but very late and only on large length scales, so that we can examine the cooling state and compare it to Haff's law,

$$T(t) = \frac{T(t_0)}{(1 + \tau/T(t_0))^2}.$$

We calculate the global temperature via the mean of the squared fluctuating part of the velocity. Plotted in a double logarithmic plot this should result in a slope of -2 after some time. We can see exactly this expected behavior in Fig. 3.5 over six orders of magnitude in time and eight orders of magnitude in temperature.

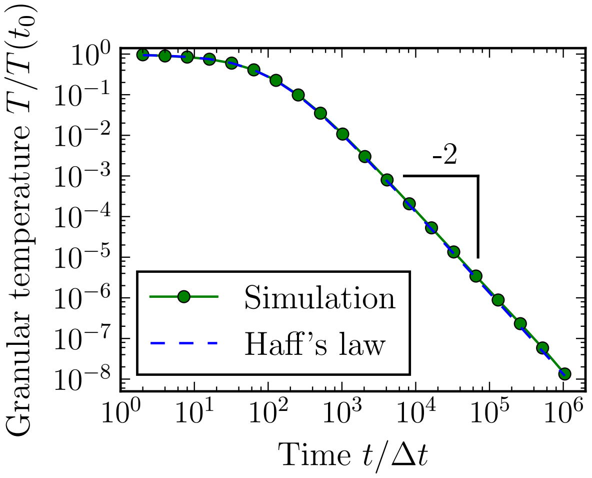


Figure 3.5: Granular temperature of the freely cooling granular system as a function of time. The evolution perfectly follows Haff's law over six orders of magnitude. After some initial time the decay of GSRD temperature has a slope of -2 in the double logarithmic plot.

We can conclude that the algorithm fulfills one of the essential characteristics of granular gases, *i.e.*, Haff's law for the homogeneous cooling state. This is of course not a completely surprising result as we have assumed the system to be locally homogeneous and put in this physics at that level. However, we can already see that there is more happening in the system. We already obtain a hint that the homogeneous cooling state is unstable to density inhomogeneities as indicated in Fig. 3.4.

3.4 GSRD streaming viscosity

We test the expressions in Eq. (3.1) for the GSRD kinematic viscosity with shear simulations, similarly to what we did in Sect. 3.2 for standard SRD. Since the fluid dissipates energy it reaches a stationary state were all the heat induced by the shearing is at the same time dissipated. For this reason, no thermostat is necessary in the simulations. Here, we simulate 2D 32×32-cell systems with density $\rho_0 = 10/\mathcal{V}$ and $\rho_0 = 20/\mathcal{V}$. Again we set $\Delta t = t_{\mathrm{unit}}$. As predicted by Eq. (3.1), similarly to standard SRD, the viscosity is low for rotation angles close to $\alpha = 90°$ Hence for these parameters the heat production is really low and the system becomes more and more unstable to exhibiting shear banding. For this reason, we need to increase the shear rate together with the rotation angle to keep the system in a stable state. However, if initially the system is completely at rest, it does not relax to the stationary sheared state. In this case the cooling immediately leads to a contraction and the contact to the Lees-Edwards boundaries is disrupted. To help the system reach the desired state, we initialize it with a linear velocity profile along the z-axis and a temperature close to the assumed final equilibrium temperature.

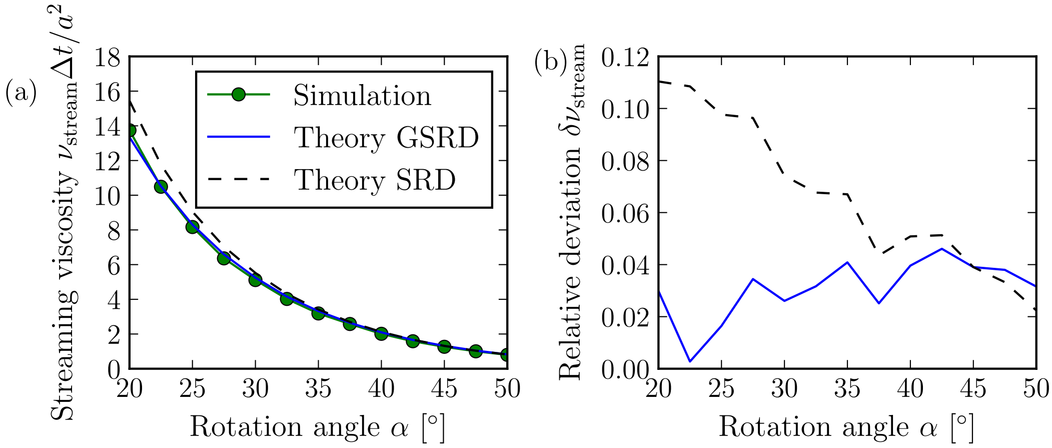


Figure 3.6: Kinematic viscosity of the 2D GSRD fluid with $\rho_0 = 10/\mathcal{V}$. In the left panel (a) we see the kinematic viscosity $\nu_{\mathrm{kin}} = \mu_{\mathrm{kin}}/\rho_0$ as a function of the GSRD rotation angle α. In the right panel (b) we see the relative deviation from the theory for granular fluids, and from the standard theory, for comparison. We see a good agreement with the theory with a maximum deviation of 4%, which represents a good improvement from the standard theory. The effective coefficient of restitution is not constant due to the different equilibrium temperatures resulting from different shear rates. Overall, the values lie around $\epsilon_{\xi,\mathrm{eff}} = 0.998$. The reason for the worse agreement at high α is the instability of the homogeneous state. For increasing α the shear heating becomes less important, so that for angles slightly larger than 40° the system needs a high shear rate to remain stationary. This in turn contradicts our assumptions for the derivation of Eq. (3.1).

If the heat input via the Lees-Edwards boundary is too low, the system forms one, sometimes also more, condensation bands. These cool down much more quickly than the rest of the system, and contract more and more. Thereby in those regions the momentum transport is disrupted and the cooling dominates. This makes shear measurements in this parameter region unfeasible. With mean effective coefficient of restitution $\epsilon_{\xi,\mathrm{eff}} = 0.946$ and $\alpha = 20°$ the system quickly reaches a state as shown in Fig. 3.8, where we see a band with 4 times larger density and with a temperature an order of magnitude lower than the rest of the system.

Hence, all in all, there are two effects that complicate the shear measurement. For larger α, the system is sheared increasingly fast so that our theory loses its validity. On the other end, for small α, the higher temperature also results in a higher $\epsilon_{\xi,\mathrm{eff}}$ which makes the assumption of the velocities of particles being uncorrelated questionable. Also with this condition local mixing and relaxation becomes slower.

In panels (b) of Fig. 3.6 and 3.7 we show the relative deviation

$$\delta\nu_{\mathrm{stream}} = \frac{\nu_{\mathrm{stream,measured}} - \nu_{\mathrm{kin,theory}}}{\nu_{\mathrm{stream,theory}}}$$

of the measurements to the theoretic predictions of Eq. (3.1) and for comparison also to Eq. (1.48).

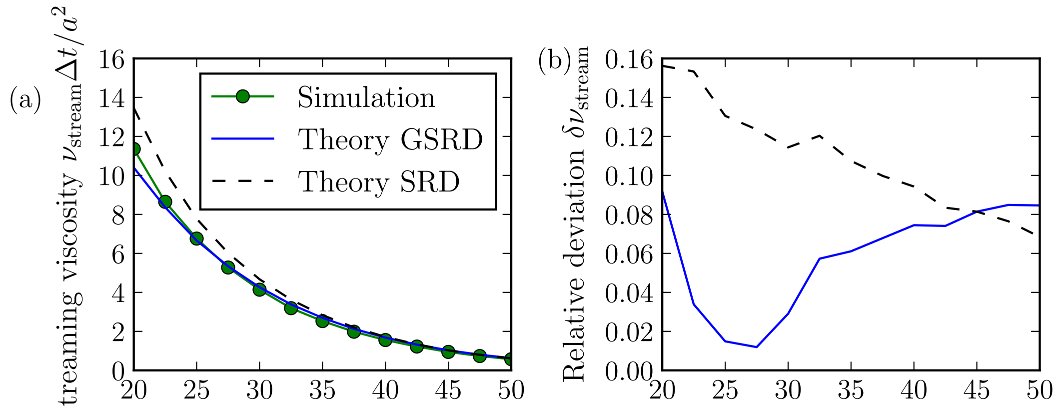


Figure 3.7: Kinematic viscosity of the 2D SRD fluid for a mean density of $\rho_0 = 20/\mathcal{V}$. In the left panel (a) we see the kinematic viscosity $\nu_{\mathrm{kin}} = \mu_{\mathrm{kin}}/\rho_0$ as a function of the SRD rotation angle α. In the right panel (b) we see the relative deviation from the granular theory and the standard theory, for comparison. Also here, we see a good agreement with at maximum 9% error. We attribute the deviation at the small rotation angle α to the resulting slower mixing and higher temperature we obtain in these simulations. In the same range of $\alpha < 27.5°$ also the standard theory exhibits an increasing deviation. The standard theory's deviation from the computed values also increases linearly with decreasing α as we have seen in Fig. 3.6(b).

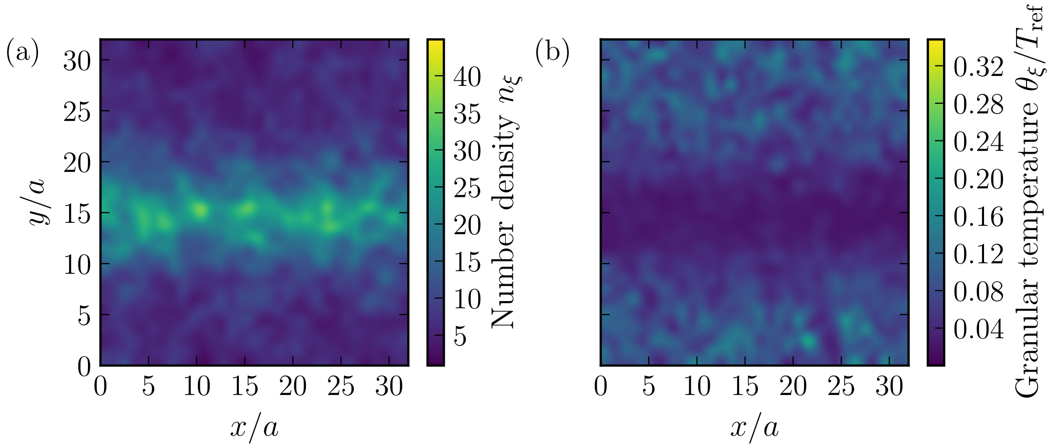


Figure 3.8: Shear simulation of the 2D GSRD fluid that developed a shear condensation for $\alpha = 50°$ and initial $\epsilon_{\xi,\mathrm{eff}} = 0.946$. The momentum transport across the band is disrupted and the band continues to contract. Like in Fig. 3.4, also here we see the result of the higher dissipation rate in denser regions - the shear band is colder than the rest of the system.

Overall, we can see a rather good agreement for both $\rho_0 = 10/\mathcal{V}$ in Fig. 3.6 and $\rho_0 = 20/\mathcal{V}$ in Fig. 3.7. In both cases we see slight deviations at either low or high rotation angles. We attribute this to the slow relaxation at low α that has even more weight in the granular case where collisions also increase correlation between particles. In the present case, this effect increases with temperature as the constant rate effective collisions have to account for a higher dissipation rate.

For the rotation angle larger than $\alpha = 50°$, we have found the system to be unstable and with a strong tendency to form condensation bands that complicate the shear measuring. Even in the stable parameter range, the system exhibits regions of slightly increased density that extend over several cell lengths a. We assume that this condition leads to the deviation of the measurement from Eq. (1.48) for values of $\alpha > 40°$. This condition also complicates the derivation, where we have assumed that the number of particles in a cell follows a Poisson distribution.

An alternative setup to measure the viscosity of the fluid could be a Poiseuille flow experiment, where the fluid flows between two walls with no-slip boundary conditions. With this setup the driving acts in a plane parallel to the flow instead of perpendicular which might stabilize the system considerably. Furthermore, this setup is much more feasible to implement for experiments. Another problem is the large compressibility of the SRD and also GSRD fluid which by its nature has an almost ideal equation of state.

3.5 Inhomogeneous cooling state

3.5.1 Clustering

In this section we want to investigate the dynamical features of granular clusters. For large coefficient of restitution, *e.g.* $\epsilon = 0.995$ the freely-cooling granular gas in the homogeneous cooling state exhibited indications of growing clusters. In order to obtain well-developed clustering we reduce the coefficient of restitution as a first step. To obtain clusters at the length scale of our observation it is further necessary to further reduce the rotation angle to $\alpha = 10°$. This corresponds to a more dilute gas in which particles travel in the same direction longer without colliding, but most importantly it assures that the kinetic contribution to momentum diffusion remains dominant as the system cools down (see Discussion).

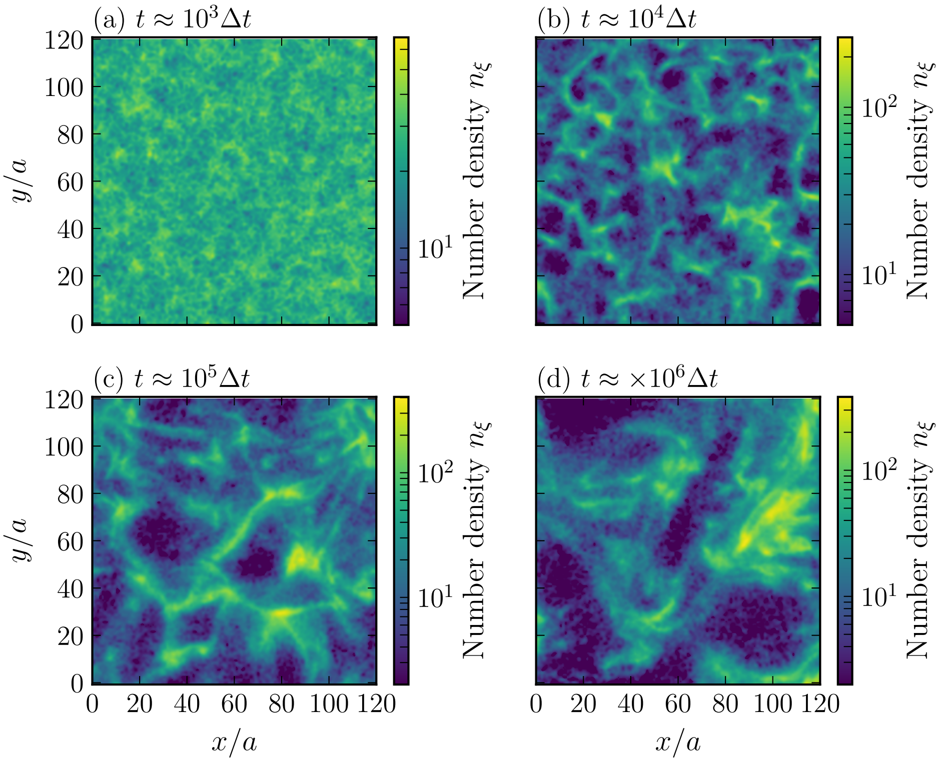


Figure 3.9: Density profile of the freely cooling GSRD system for $\epsilon = 0.98$, with initially $T(t_0) = 100\ T_{\text{ref}}$ and $\Delta t = 0.01$. Panel (a) shows the system close to the transition to the inhomogeneous cooling state. Panel (b): at $t = 10^4\Delta t$ the system has developed clusters, the density varies over 3 orders of magnitude. These clusters continue to grow in size, as can be seen in panels (c) and (b) and also continue to contract.

In the following we investigate a system of 120^3 cells with a initial density of $\rho_0 = 25/\mathcal{V}$, hence $N = 43.2 \times 10^6$ particles. Also here, we simulate using periodic boundary conditions along all dimensions. Caution has to be exercised for this setup because it is theoretically possible to choose parameters for which $\epsilon_{\xi,\mathrm{eff}} \leq 0$, particularly if the time step Δt is to large. For the chosen $\epsilon = 0.98$, $T(t_0) = 100\ T_{\mathrm{ref}}$ and $\Delta t = 0.01 t_{\mathrm{unit}}$ we are at the lower bound so that in the first time step $\epsilon_{\xi,\mathrm{eff}} = 0$. Actually for the 3D system this does not constitute a problem. The system remains in the homogeneous cooling state for approximately $3 \times 10^3 \Delta t$, as we can see in Fig. 3.9(a), where density variations span only one order of magnitude. Subsequently, clusters develop. First their length scales are small (*cf*. Fig. 3.9(b)), then continue growing (*cf*. Fig. 3.9(c),(d)). In Fig. 3.9(d) the maximum density has still grown compared to before (Fig. 3.9(c)).

The clusters all feature a filament-like shape, similar to those that have been observed using various simulation techniques [10, 23, 24, 12].

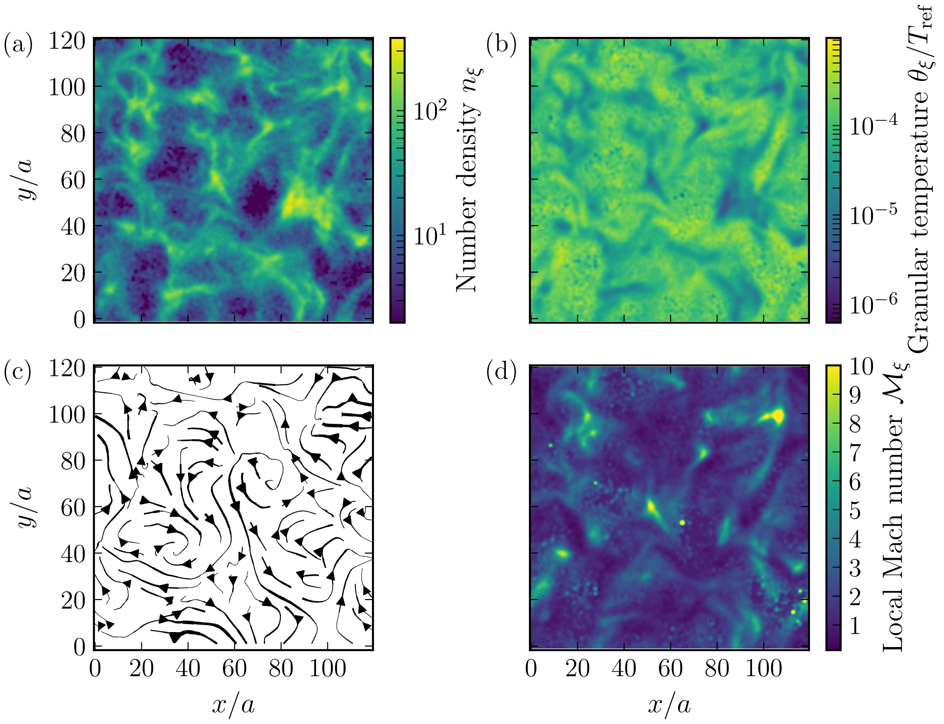


Figure 3.10: Details of the freely cooling system in a cross-section perpendicular to the z-axis at $6.5 \times 10^4 \Delta t$. Panel (a) shows the density, panel (b) the temperature. In panel (c) the streaming profile is visualized with a streamline plot where the line width is proportional to the flow velocity in the plane. Panel (d) shows the local Mach number in the system. We observe that dense clusters are cold and stream macroscopically with supersonic velocities. Also in the streamline plot, we observe vortices in the flow field.

The growth of clusters originates from both accumulation of more particles and collisions of the macroscopic clusters. This means that although the system is rather cold there are still dynamics. Because the thermal velocities decrease rather quickly in time, while the convective velocities decrease to a lesser extent, the granular gas flow becomes quickly supersonic. A measure of this dynamical feature is given by the Mach number $\mathcal{M} = \|v\|/v_T$. These supersonic flows can be seen in the map of the local Mach number shown in Fig. 3.10(d). Additionally, we show a streamline plot in Fig. 3.10(c) and local temperatures in Fig. 3.10(b). Their combination shows that the higher cooling rates in the dense regions lead to bonding of particles to the cluster due to the lower temperature and hence escape velocity. In this state the clusters behave as macroscopic objects themselves because they stream with velocities faster than their internal dynamics, as can be read from the local Mach numbers $\mathcal{M}_\xi > 1$.

The process of clusters growing continues until clusters have reached the size of the system and then saturates. For bigger system sizes this process of course takes longer to finish. The evidence for this fact comes from smaller simulations where we have observed this outcome after $\sim 10^6 \Delta t$. When the saturation sets in, the cooling follows a power law $T \sim (1+t)^{-2}$ again, similar to Haff's law. This is because once clusters have formed, cooling occurs inside them on small scales again, over which the system is homogeneous.

In simulations with a slightly lower coefficient of restitution of $\epsilon = 0.975$ we observe the transition to inhomogeneous cooling slightly earlier at $\sim 10^3 \Delta t$. This simulation was performed with a 200^3 grid, a 3D rendered snapshot after $\sim 10^4 \Delta t$ can be seen in Fig. 3.11. These observations completely agree with findings obtained with an event driven simulation of Luding [23].

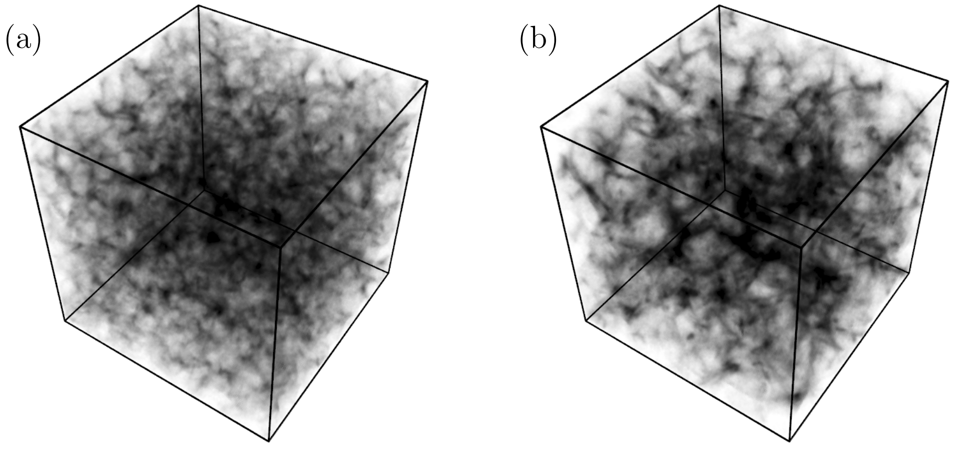


Figure 3.11: 3D density configuration of the GSRD system at (a) $t = 2 \times 10^3 \Delta t$ and (b) $t = 10^4 \Delta t$. The figures are obtained with ray-tracing. The light permittivity of each GSRD cell proportional to its density. Hence, the middle along the diagonal appears darker due to the longer light paths. We can see denser clusters that have filament-like shapes that extend in all directions with no preferred direction.

3.5.2 Cluster growth

In the previous section we have looked at the cluster formation dynamics on macroscopic length scale and have thereby observed that the density, velocity and temperature become inhomogeneous with a rather quick transition. The moment when this transition occurs can be better identified via the standard deviation of the density distribution, that gives a measure of the fluctuations in the density. The time evolution of the fluctuation (*cf.* Fig. 3.12(b)) normalized with the initial value shows a pronounced change from almost constant to a linear growth after approximately $3 \times 10^3 \Delta t$. The standard deviation increases to around 7 times the initial value. Coinciding with that, we see that the mean kinetic energy of the system, which for identical particles is defined as

$$E_{\text{kin}} \equiv \frac{1}{N} \sum_{\boldsymbol{\xi}} n_\xi \mathbf{V}_\xi \tag{3.1}$$

crosses the curve of the temperature in Fig. 3.12(a) at about $3 \times 10^3 \Delta t$.

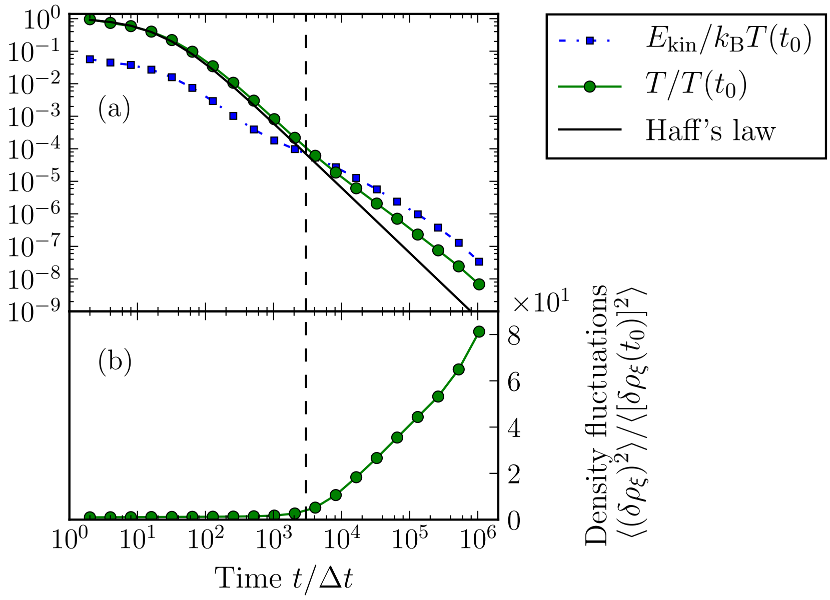


Figure 3.12: Cooling behavior of the GSRD fluid for $\epsilon = 0.98$. Panel (a) shows the development of the mean kinetic energy of the convective degrees of freedom and granular temperature as a function of time with a fit of Haff's law. We see rather good agreement up to $t < 3 \times 10^3 \Delta t$ where the curve of E_{kin} and T cross and hence the global Mach number $\mathcal{M}$ exceeds one. The temperature follows a different power law afterwards. Panel (b) shows the development of the standard deviation of the density distribution $\sigma(\rho)$ relative to $t = 0$. The fluctuations $\sigma(\rho)$ only slowly increase until $t \simeq 3 \times 10^3 \Delta t$ when the behavior changes. In the following, the fluctuations increase quickly hence indicating a transition to the inhomogeneous cooling state, coinciding with the change of slope in the cooling behavior and Mach number $\mathcal{M} > 1$.

Figure 3.13 shows a close-up view of the temporal dependence of the temperature. One can observe that with insetting saturation there appears another change of slope. The global cooling thereby increases. In the middle of the clustering regime the slope of the power law becomes $T(t) \sim t^{1.55}$, then slowly returning to $T(t) \sim t^{1.62}$. The change of slope during the inhomogeneous cooling state has also been observed in previous works [23, 24, 25, 26]. Saturation processes also agree with their observations. Different predictions for the exponent of the new power law $T(t) \sim t^b$ have been proposed. Brito and Ernst [25] predicted and observed a dependence of $T(t) \sim t^{-3/2}$ in three dimension which comes close to what we find here.

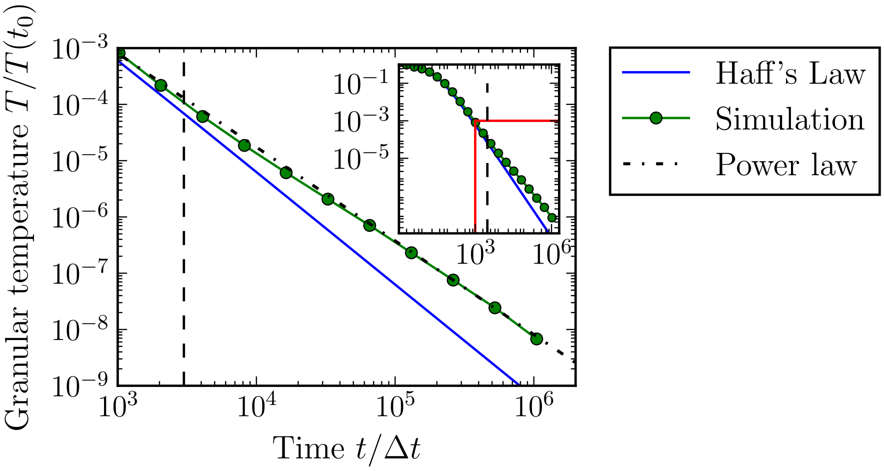


Figure 3.13: Detailed cooling behavior of the GSRD fluid for $\epsilon = 0.98$ in the inhomogeneous cooling state. We show the development of the granular temperature as a function of $t > 3 \times 10^3 \Delta t$, *i.e.*, the region indicated in the inset. In this region the cooling behavior changes to a power law $T(t) \sim t^{1.55}$. With the saturation of the clusters the slope begins to approach -2 again.

Likewise, in the simulations in [25], which are performed using event driven molecular dynamics simulations, a change back to Haff's law is observed after the saturation of clusters, in agreement with what we find.

3.6 Velocity distribution function

The velocity distribution function for nonequilibrium granular gases deviates from the Maxwell-Boltzmann distribution [10]. In particular it decays slower, or in other words, has a longer tail at larger velocities.

We can determine the velocity distribution function also for the GSRD particles. It is convenient to express the distribution as a function of the scaled velocities

$$\mathbf{c} \equiv \frac{\mathbf{v}}{\|\mathbf{v}_T\|}, \tag{3.2}$$

where the thermal velocity is given by $T = m/2\,\mathbf{v}_T^2$. The scaled velocities are distributed following a scaled distribution function

$$f(\mathbf{v},t) = \frac{\rho}{\|\mathbf{v}_T(t)\|^3}\,\tilde{f}(\mathbf{c}). \tag{3.3}$$

It is convenient to write $\tilde{f}(\mathbf{c})$ as an expansion

$$\tilde{f}(v,t) = \phi(c)\left[1 + \sum_{p=1}^{\infty} a_p(t) S_p(c^2)\right], \tag{3.4}$$

where the first term $\phi(c) \equiv \pi^{-3/2}\exp(-c^2)$ is the Maxwell distribution for the scaled velocity. The following terms give corrections to the equilibrium Maxwell distribution. A suitable set of orthogonal functions $\{S_p(x)\}$ are the Sonine polynomials (closely related to the Laguerre polynomials) with corresponding coefficients a_p For the granular velocity distribution function, it is sufficient to consider the first three orders

$$\begin{aligned} S_0(x) &= 1, \\ S_1(x) &= -x + \frac{3}{2}, \\ S_2(x) &= \frac{x^2}{2} - \frac{5x}{2} + \frac{15}{8}. \end{aligned}$$

A plot of the GSRD particles' velocity distribution is presented in Fig. 3.14. We observe a good agreement with observations from molecular dynamics simulations performed by Pathak *et al.* [26]. The second Sonine coefficient is of major importance in theories for the velocity distribution function in granular gases [2].

We observe that the second Sonine coefficient is not constant, but changes with time. In more detail, we report a decrease of a_2 over the first time period up to $t = 5 \times 10^2 \Delta t$, which is followed by an approach of a_2 back towards zero. We can see in Fig. 3.15 that for higher values of ϵ the curves have a lower minimum around $t = 5 \times 10^2 \Delta t$.

These findings are qualitatively in good agreements with the analytical predictions and observations by Brilliantov and Pöschel [2] for a variable coefficient of restitution.

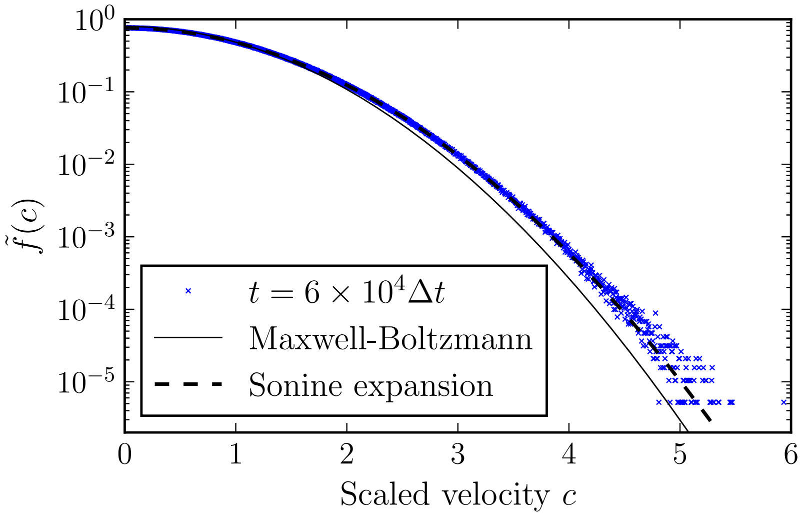


Figure 3.14: Scaled velocity distribution function $\tilde{f}$ of the 3D SRD fluid. The distribution has a stronger tail than the Maxwell-Boltzmann distribution.

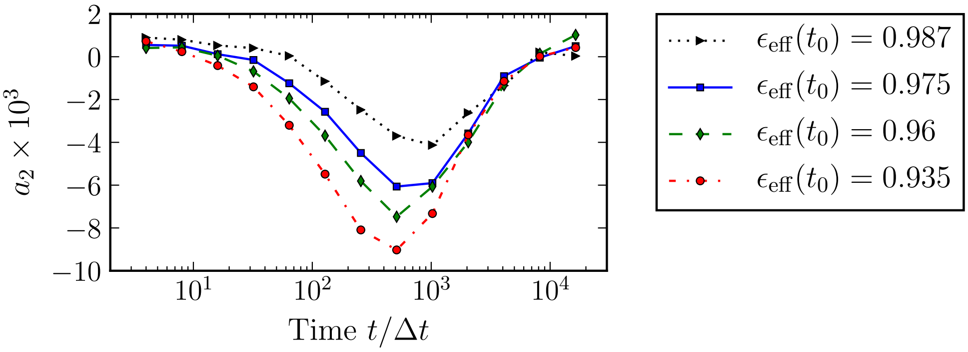


Figure 3.15: Second Sonine coefficient as a function of time for four different coefficients of restitution. First, there is a decrease followed by an approach back towards zero. Minima are lower for higher $\epsilon_{\xi,\text{eff}}(t_0)$. At around $t = 4 \times 10^3 \Delta t$ the systems for the $\epsilon_{\xi,\text{eff}} = 0.935, 0.96$ come close to the transition to the inhomogeneous cooling state where we see an unexpected positive value of a_2. The development of a_2 well agrees with analytical predictions of Brilliantov and Pöschel [2] for a variable coefficient of restitution.

Chapter 4

Discussion

4.1 Coarse-graining via the mean collision time

By applying the SRD and GSRD algorithms we drastically coarse-grain the mean collision time to one single constant, *i.e.*, the time step Δt. This approximation is well justified in homogeneous systems with small gradients and spatial variations.

As we have seen in this work and is also observed elsewhere, the freely cooling granular gas develops spatial variations of the observables that range over several orders of magnitude. Nevertheless, the global Δt remains the same. It is hence not a priori clear if this is a good approximation. Clearly the time step Δt controls how finely the dissipation is resolved in space, and thus the cluster formation.

The system's development of temperature and kinetic energy is well characterized by the global Mach number $\mathcal{M} = v/\sqrt{T}$. Thus, observing the change in the global Mach number in Fig. 4.1, we use the fluctuations in density as a measure for clustering using different Δt. Ironically, as the time step is too small, *i.e.* for $\Delta t\sqrt{T(t_0)} < a/2$ the intrinsic collisional transport phenomena of GSRD leads to unphysical appearance of clusters for $\mathcal{M} < 1$. If, however, $\Delta t\sqrt{T(t_0)} > a/2$ the cluster dynamics does not depend on the choice of Δt. Thus, we can conclude that using a global Δt is a valid approximation and take $\Delta t\sqrt{T(t_0)} > a/2$ as a condition for choosing the input parameters Δt and $T(t_0)$. On other end, the condition that $\epsilon_{\xi,\text{eff}} > 0$ for Δt determines an upper bound for Δt.

4.2 Finite size effects

We consider a freely cooling GSRD system with $\epsilon = 0.98$ as in the previous Section. If we simulate a smaller system using only 30^3 cells, the cluster growth saturates within 10^6 time steps. If this occurs, we see a new interesting behavior. In Fig. 4.2 we see the development of the mean kinetic energy of the convective degrees of freedom

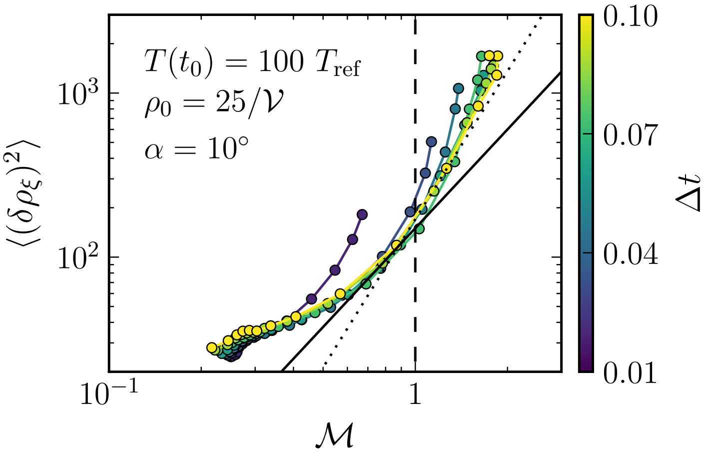


Figure 4.1: Density fluctuations $\sigma(\rho_\xi)^2$ as a function of $\mathcal{M}$ for different Δt. The solid line show a power law with exponent two, the dotted line a power law with exponent 3. The developments start from weak fluctuations. For smaller Δt we observe lower initial fluctuations. For $\Delta t\sqrt{T(t_0)} < a/2$ we observe clustering for $\mathcal{M} < 1$. In this case the algorithm's inherent collisional transport leads to unphysical behavior. For $\Delta t\sqrt{T(t_0)} > a/2$ the dynamics follow a common trajectory. Crossing $\mathcal{M} = 1$ the slope increases due to cluster formation. We conclude $\Delta t\sqrt{T(t_0)} > a/2$ as a necessary condition for choosing the input parameters Δt and $T(t_0)$. The hydrodynamic theory predicts a universal scaling low stating that all trajectories below $\mathcal{M} < 1$ converge to a power law $\sigma(\rho_\xi)^2 \propto \mathcal{M}^2$. Although we see a tendency for this behavior, we find little evidence for that. In the inhomogeneous cooling all curve approach a power law with exponent 3.

and temperature for a 120^3 cell and a 30^3 cell simulation. In the inhomogeneous cooling state both curves follow a power law.

From visual inspections of Fig. 3.9 we can estimate that the typical length scale of clusters reaches $30a$ at approximately $t = 10^5\Delta t$. While in the large simulations the clusters keep growing, in the small system the clusters start interacting with themselves through the periodic boundary conditions. Most of the kinetic energy in the system consists of the movement of clusters. Since the total momentum is zero, the kinetic energy is dissipated faster from this time on, because the whole system connects to an extended cluster that doesn't move. We observe that the kinetic energy drops about three orders of magnitude in one magnitude in time.

We conclude that this effect restricts the further investigation of the system's development.

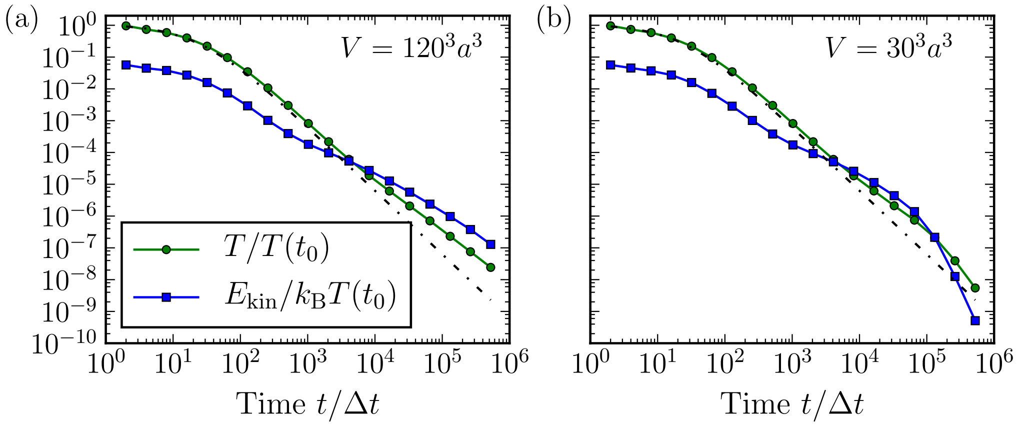


Figure 4.2: Granular temperature and mean kinetic energy of the freely cooling GSRD system with $\epsilon = 0.98$ in (a) a large system size of 120^3 cells and (b) a small system size of 30^3 cells. In both systems the kinetic energy decreases with a power law slower than the temperature. At approximately $t = 10^5\Delta t$ the cluster length scale reaches $30a$, the linear size of the smaller system (b). This system contracts to one big cluster that does not move due to momentum conservation. We observe this process by a sudden drop of kinetic energy in (b) of 3 orders of magnitude in one decade of time.

4.3 Rotation angle and inhomogeneous cooling

The SRD rotation angle α also plays a role in the cluster formation. From the calculation of the viscosity we can conclude that shear heating increases as α decreases. Furthermore, α controls how far a particle travels in a certain direction without being significantly deflected.

We find that in the homogeneous cooling state the power law of the kinetic energy of the convective motion changes with α. For higher α the mean kinetic energy in the system decreases with a power law $E_{\mathrm{kin}} \propto t^{-\tau}$ with a higher exponent τ.

For granular gases, the kinetic energy curve has to cross the temperature curve, *i.e.* it has to become supersonic in order to exhibit clustering. From observations in nature, theory or models as molecular dynamics simulations we know that clustering is an essential feature of granular gases. For this reason one has to choose $\alpha \leq 11°$. The curves of the higher rotation angles in Fig. 4.3(a), (b) and (c) show a decrease of kinetic energy similar to that we have seen in Fig. 4.2(b) in the context of finite size effects. Though now the phenomenon appears for systems in the subsonic state. The reason is similar to that in Section 4.2. The collisional viscosity increases with α but is independent of the temperature [3]. However, for $\alpha \to 0$ the collisional viscosity vanishes. Hence, for α too large, as the system cools down GSRD collisions become dominant at some point of the evolution. This leads to momentum diffusion over the cell length a, thus propagating much faster than streaming.

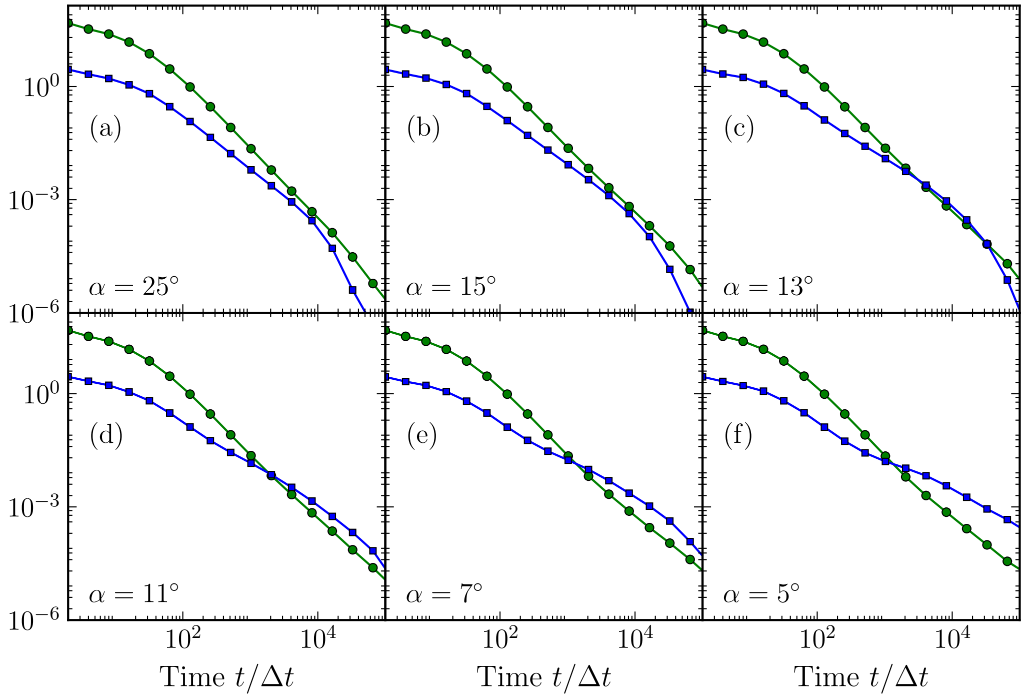


Figure 4.3: Development of the granular temperature T/T_0 (green circles) and mean kinetic energy $E_{\mathrm{kin}}/k_{\mathrm{B}}T_0$ (blue squares) in the freely cooling system as a function of time. The slope in the double logarithmic plot changes with the GSRD rotation angle α, decreasing for higher angles. Due to finite sizes effects the kinetic energy curves exhibit sudden drops at ca. $t \simeq 10^4 \Delta t$ for the systems with $\alpha > 11°$. This is because the collisional viscosity increases with α and, moreover, in contrast to the streaming viscosity, it does not depend on the temperature [3]. Hence, for α large, collisional contributions apparently become dominant at a premature state of the evolution.

As one can see in Fig. 4.4, choosing α low enough prevents that, with decreasing temperature, ν_{coll} exceeds ν_{stream}.

A larger α gives a completely different mechanism for clustering. The clusters lose the typical shape they have due to their formation under hydrodynamic shear instability. Instead they appear more globular, *cf.* Fig. 4.5. Moreover, the cluster formation saturates more quickly as indicated by the quick decay of kinetic energy in Fig. 4.3(a), (b) and (c). All together, the behavior is similar to choosing Δt too low.

A good choice of the rotation angle is hence $\alpha \leq 11°$, though caution has to be exercised in combining α, Δt and $T(t_0)$.

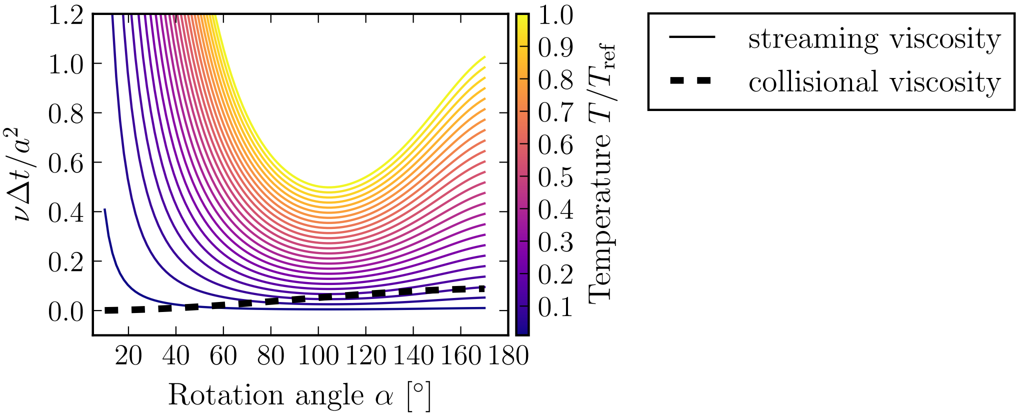


Figure 4.4: Change of streaming viscosity with temperature compared to the collisional viscosity of standard SRD [3]. For larger α eventually the collisional viscosity becomes dominant as the temperature decreases. This also applies for the granular system because qualitatively the curves do not change. In this graph we present the analytical functions for $\rho_0 = 5$ and $\Delta t = 1$.

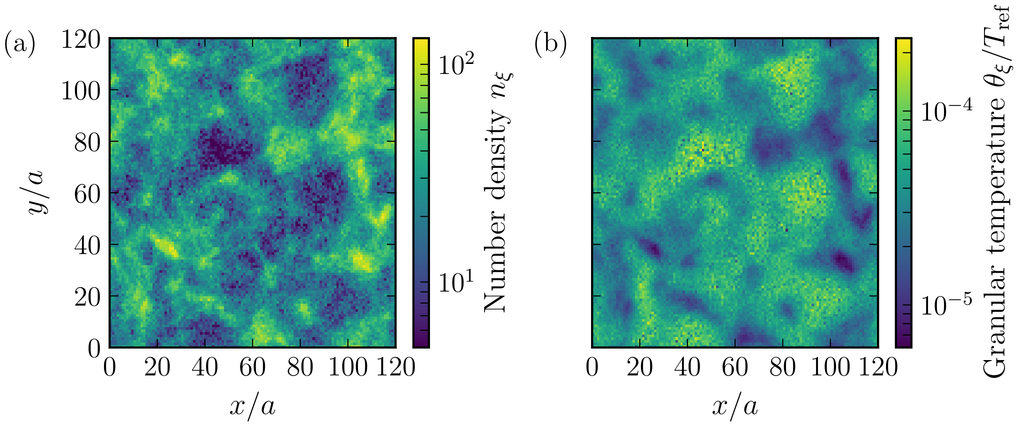


Figure 4.5: Clustering of the 3D GSRD system if collisional viscosity becomes dominant. In the cross sections (a) of the number density and (b) temperature we also observe an anti-correlation. The clusters develop in lumpy shapes because the system does not become supersonic.

CHAPTER 5

Outlook

5.1 Driving by shaking

Driven granular systems are intensively studied because they can reach a steady state and are thus easily controllable and observable. A common experimental setup consists of a granular system under the influence of gravity that is driven by a shaker attached to the system's container. The system undergoes periodic oscillations along the z-axis *cf.* Fig. 5.1. These increase the kinetic energy of particles colliding with the ground.

We have carried out preliminary simulations of a vibrated granular system. We implement this by first adding a gravity force $\mathbf{F}/m = -g\hat{\mathbf{z}}$ that we apply on particles using Euler integration with a time step Δt. While in the x and y dimension we apply periodic boundary conditions, along the z-axis we confine the system between solid walls using the bounce-back rule. The lower wall at $z = 0$ conducts periodic oscillations which serves as an energy source to the system.

During the simulation the system first compresses to a dense layer that remains at some distance from the bottom wall *cf.* Fig. 5.2(a). This dense layer lies on a hot layer, *cf.* Fig. 5.2(b), that is in contact with the vibrating, bottom wall.

In the x- and y-dimension (*cf.* Fig. 5.2(c) and (d)), the system forms clusters that appear to form patterns. These show an increase of their length scale with time. Again we observe an anti-correlation between density and temperature.

A complication of the shaken system is that as the length scale of the apparent pattern grows, the granular gas eventually collapses when the length scale reaches the system size. In other words, because SRD and also GSRD have a high compressibility, the mass in the system continues to accumulate in one point in space. We conclude that further work is necessary to stabilize the steady state of a vibrated system. This should not coma as a surprise, since a vibrated system is subject to strong forces and the granular dissipation is concentrated in a small fraction of the system's volume.

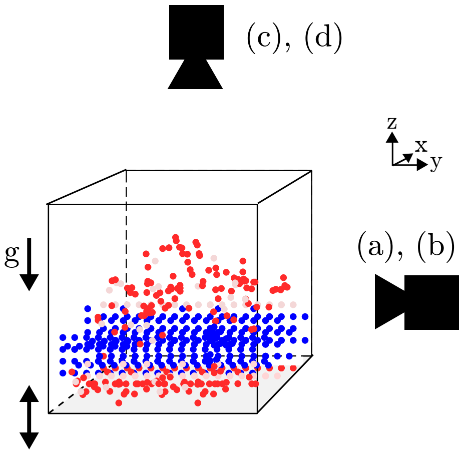


Figure 5.1: Sketch of a granular system under the influence of gravity driven by shaking. In the simulations' setup periodic boundaries are applied along the x- and y-axis. Along the z-dimension the system is confined by walls where the wall at $z = 0$ conducts a periodic oscillation. Lastly, particles are accelerated by a gravity force $\mathbf{F}/m = -g\hat{\mathbf{z}}$. Particles are sketched with red and blue circles representing large and small thermal velocities, respectively.

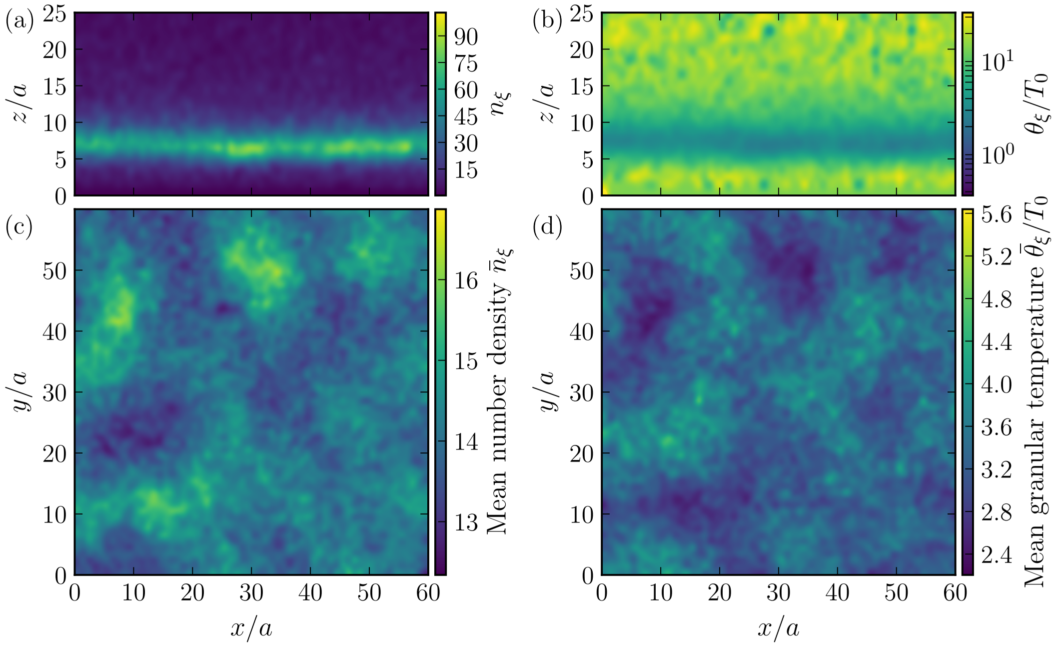


Figure 5.2: Granular SRD in a driven system. (a) The fluid forms a dense layer elevated at some height z over the ground at $z = 0$, towards which the gravitational force is pointing. (b) The layer below the dense layer is in contact with the vibrating ground and is hence hotter than the dense layer lying above. (c) Viewed from above, the dense layer itself show variations in density that form a pattern. (d) The granular temperature also in this system is anti-correlated to the density, *i.e.*, where the fluid is denser the temperature is lower.

Chapter 6

Conclusion

Granular materials are particularly intriguing because of the rich phenomenology they exhibit. From industrial applications, to geophysical and astrophysical problems, understanding their fundamental physics is of paramount importance. In the past decades, considerable experimental, theoretical and numerical work has unveiled myriad effects (often counterintuitive) occurring at different length scales in a granular system. For this reason, simulations often encounter the issue of finite size effects. Having efficient models that can easily span a large range of length and energy scales. The SRD technique offers a twofold advantage: (i) it is extremely computationally convenient as its computational complexity scales linearly with the size of the system; (ii) because of its simplicity, it is amenable to exact analytical predictions. In this Master's thesis we have taken a first concrete step towards a generalization of the particle based SRD method, originally used to simulate the hydrodynamics of simple fluids, to the simulation of granular gases. The algorithm uses an effective local coefficient of restitution to render energy dissipation dependent on local mesoscopic observables, while locally conserving density and momentum. As we were able to show, the model obeys linear granular hydrodynamic equations, which have been rigorously derived by a number of authors in the past three decades. Furthermore, we derived a formula for the kinematic viscosity of the 2D GSRD fluid.

To test our derivations, we developed a software implementation for simulation on general purpose graphics cards that we successfully tested and benchmarked with analytical predictions for standard SRD. Turning to GSRD, we observe that our prediction of the kinematic viscosity compares well with the results obtained from our simulations. In this context we found that for low shear driving the fluid becomes unstable and develops shear condensation.

In the simulations of a freely-cooling granular gas the temperature evolution follows the prediction of Haff's law over several orders of magnitude in time and temperature. Furthermore, we observe clustering for lower coefficients of restitution. The emergence and dynamics of the cluster compare well with expectations based on theory, experiments and simulations. The clustering sets in as the global Mach number exceeds one. Subsequently, the density fluctuations grow while we observe a change in the power law of the temperature evolution. The clusters exhibit a higher cooling rate, hence density and temperature become anti-correlated. This locally leads to supersonic flow. After their emergence, clusters move, collide and thus grow further.

The velocity distribution function compares well with theoretical predictions. The distributions decay slower, a phenomenon known as high energy tail. The shape of the reduced velocity distribution function changes with time as predicted. The second coefficient of the Sonine expansion qualitative matches with analytical predictions.

In our discussion we provide physical criteria for a critical selection of model parameters, and identify the effects of the finite system size. Lastly, in our Outlook we provide detailed information about the application of the GSRD method for the simulation of granular matter under the influence of an external gravity force, driven by a spatially oscillating boundary condition. We believe that the work initiated here is a useful contribution to the research on granular gases.

Appendix

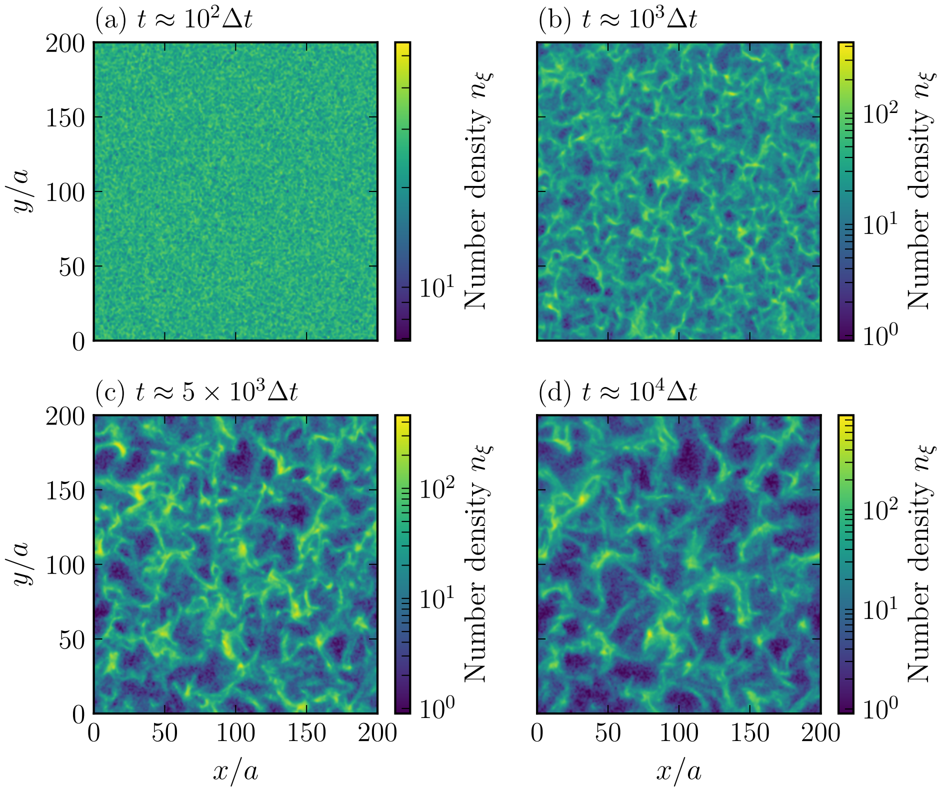


Figure A.1: Evolution of the freely-cooling 3D GSRD fluid for $\epsilon = 0.975$ on a grid of 200^3 cells. The transition to inhomogeneous cooling sets in at approximately $t = 10^3 \Delta t$. Subsequently, the formed clusters merge as they move through the system and form larger clusters. The maximum density grows larger for this bigger system, what we can see in panel (d), since the limit of the cluster length scale is larger here.

Content of the enclosed Compact Disk

filename	description
MasterArbeit.pdf	this document
garular.avi	movie of a freely-cooling granular gas calculated with our code ($n_0 = 25,\ \alpha = 10,\ \epsilon = 0.99,\ \Delta t = 0.01$). Each frame shows a density cross-section 200 Δt after the previous frame.

Bibliography

[1] L. Vanel and E. Clément, *Pressure screening and fluctuations at the bottom of a granular column*, The European Physical Journal B-Condensed Matter and Complex Systems **11**, 525 (1999).

[2] N. V. Brilliantov and T. Pöschel, *Velocity distribution in granular gases of viscoelastic particles*, Physical Review E **61**, 5573 (2000).

[3] N. Kikuchi, C. Pooley, J. Ryder, and J. Yeomans, *Transport coefficients of a mesoscopic fluid dynamics model*, The Journal of Chemical Physics **119**, 6388 (2003).

[4] P. Jop, Y. Forterre, and O. Pouliquen, *A constitutive law for dense granular flows*, arXiv preprint cond-mat/0612110 (2006).

[5] J. Chambers, *Planetesimal formation by turbulent concentration*, Icarus **208**, 505 (2010).

[6] S. Weidenschilling, *Dust to planetesimals: Settling and coagulation in the solar nebula*, Icarus **44**, 172 (1980).

[7] J. E. Chambers, *Planetary accretion in the inner solar system*, Earth and Planetary Science Letters **223**, 241 (2004).

[8] J. Schmidt, H. Salo, F. Spahn, and O. Petzschmann, *Viscous overstability in Saturn's b-ring: II. hydrodynamic theory and comparison to simulations*, Icarus **153**, 316 (2001).

[9] A. G. Athanassiadis, M. Z. Miskin, P. Kaplan, N. Rodenberg, S. H. Lee, J. Merritt, E. Brown, J. Amend, H. Lipson, and H. M. Jaeger, *Particle shape effects on the stress response of granular packings*, Soft Matter **10**, 48 (2014).

[10] N. V. Brilliantov and T. Pöschel, *Kinetic theory of granular gases*, (Oxford University Press 2010).

[11] P. K. Haff, *Grain flow as a fluid-mechanical phenomenon*, Journal of Fluid Mechanics **134**, 401 (1983).

[12] M. Hummel, J. P. Clewett, and M. G. Mazza, *A universal scaling law for the evolution of granular gases*, Europhysics Letters **114**, 10002 (2016).

[13] M. Hummel, *Hydrodynamics of granular gases: clustering, universality and importance of subsonic convective waves*, Phd. Thesis, Universität Göttingen (2016).

[14] A. Malevanets and R. Kapral, *Mesoscopic model for solvent dynamics*, The Journal of Chemical Physics **110**, 8605 (1999).

[15] C. Pooley and J. Yeomans, *Kinetic theory derivation of the transport coefficients of stochastic rotation dynamics*, The Journal of Physical Chemistry B **109**, 6505 (2005).

[16] C. Holm and K. Kremer, *Advanced computer simulation approaches for soft matter sciences III*, vol. 221, (Springer 2008).

[17] J. Blaschke, M. Maurer, K. Menon, A. Zöttl, and H. Stark, *Phase separation and coexistence of hydrodynamically interacting microswimmers*, Soft Matter **12**, 9821 (2016).

[18] T. Ihle and D. Kroll, *Stochastic rotation dynamics: a galilean-invariant mesoscopic model for fluid flow*, Physical Review E **63**, 020201 (2001).

[19] A. Malevanets and R. Kapral, *Solute molecular dynamics in a mesoscale solvent*, The Journal of Chemical Physics **112**, 7260 (2000).

[20] M. P. Allen and D. Tildseley, *Computer Simulations of Liquids*, (Clarendon Press 1987).

[21] K.-W. Lee and M. G. Mazza, *Stochastic rotation dynamics for nematic liquid crystals*, The Journal of Chemical Physics **142**, 164110 (2015).

[22] G. Marsaglia et al., *Xorshift rngs*, Journal of Statistical Software **8**, 1 (2003).

[23] S. Luding, *Structure and cluster formation in granular media*, Pramana **64**, 893 (2005).

[24] S. Miller and S. Luding, *Cluster growth in two-and three-dimensional granular gases*, Physical Review E **69**, 031305 (2004).

[25] R. Brito and M. Ernst, *Extension of Haff's cooling law in granular flows*, Europhysics Letters **43**, 497 (1998).

[26] S. N. Pathak, Z. Jabeen, D. Das, and R. Rajesh, *Energy decay in three-dimensional freely cooling granular gas*, Physical Review Letters **112**, 038001 (2014).

Acknowledgements

I want to thank my friends and family very much for supporting me the whole time. I apologize to my girlfriend who had to bear me a lot, and especially in moments where I was mentally absent thinking about physics. I am really glad about her patience in that regard.

Thanks to Marco G. Mazza all the fruitful discussions and the steady help and encouragement, and Chamkor Singh for coffee breaks and explanations about granular physics. In this manner I also want to very much thank Fabian Schwarzendahl for discussions and even more for the nice time we spend climbing in the woods and dusty climbing gym.

I was also very happy about the nice and very friendly working environment at the MPIDS, the company and help from all the group members, especially mentioning Christine Linne and Hannes Hornischer making the long evenings in the institute much more pleasant. I will miss everyone.

I also want to thank my flatmates for tolerating the dirty kitchen.

Erklärung

Gemäß §12(9) der Prüfungsordnung des Master-Studiengangs Physik an der Universität Göttingen:

Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe und alle Stellen, die wörtlich oder sinn- gemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, im Rahmen einer nichtbestandenen Prüfung an dieser oder einer anderen Hochschule eingereicht wurde.

Des Weiteren versichere ich, dass die schriftliche Version dieser Abschlussarbeit mit der ergänzend vorgelegten Version übereinstimmt.

Göttingen, September 12, 2017

(Arne Wolf Zantop)