Towards designer materials using customizable particle shape

by

Anton Piankov

December , 2021

A thesis submitted to the
Graduate School
of the
Institute of Science and Technology Austria
in partial fulfillment of the requirements
for the degree of
Master of Science

Committee in charge:

Carl Goodrich

Scott Waitukaitis

The Master's thesis of Anton Piankov, titled Towards designer materials using customizable particle shapes, is approved by:

Supervisor: Carl Goodrich, IST Austria, Klosterneuburg, Austria

Signature: _____

___

Committee Member 1: Scott Waitukaitis, IST Austria, Klosterneuburg, Austria

Signature: _____

Signed page is on file

IST Austria Master's Thesis, ISSN: 2791-4585

I hereby declare that this thesis is my own work and that it does not contain other people's work without this being so stated; this thesis does not contain my previous work without this being stated, and the bibliography contains all the literature that I used in writing the dissertation.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee, and that this thesis has not been submitted for a higher degree to any other university or institution.

I certify that any republication of materials presented in this thesis has been approved by the relevant publishers and co-authors.

Signature: _____

Anton Piankov

December , 2021

Signed page is on file

Abstract

Those who aim to devise new materials with desirable properties usually examine present methods first. However, they will find out that some approaches can exist only conceptually without high chances to become practically useful. It seems that a numerical technique called automatic differentiation together with increasing supply of computational accelerators will soon shift many methods of the material design from the category "unimaginable" to the category "expensive but possible". Approach we suggest is not an exception. Our overall goal is to have an efficient and generalizable approach allowing to solve inverse design problems. In this thesis we scratch its surface. We consider jammed systems of identical particles. And ask ourselves how the shape of those particles (or the parameters codifying it) may affect mechanical properties of the system. An indispensable part of reaching the answer is an appropriate particle parametrization. We come up with a simple, yet generalizable and purposeful scheme for it. Using our generalizable shape parameterization, we simulate the formation of a solid composed of pentagonal-like particles and measure anisotropy in the resulting elastic response. Through automatic differentiation techniques, we directly connect the shape parameters with the elastic response. Interestingly, for our system we find that less isotropic particles lead to a more isotropic elastic response. Together with other results known about our method it seems that it can be successfully generalized for different inverse design problems.

Contents

List of Figures

1 Introduction

Our global goal is to be able to find certain particle shapes that would realize the materials with properties we choose in advance. In other words, we are trying to solve an inverse design problem. The second part of this chapter will be about this aspect of our research, the first part will be devoted to the introduction to disordered matter (the material class we consider).

1.1 Disordered matter

In 1915, William Laurance Bragg and his father William Henry Bragg won the nobel prize for showing that atoms in crystals form a periodically repeating lattice. Ever since, structural order has formed the foundation for how we characterize solids as materials with long-range order. On the other hand, in liquids we can find at most short range order determined by a certain amount of nearest neighbours. Therefore, it can be tempting to use structural order to discern solids from liquids. However, there are whole classes of solids that actually lack long-range order, notably the glasses. We call it disordered matter.

However, it would be wrong to assume that disordered matter must lack any kind of order, what we require is the absence of translational periodicity manifesting itself in some level of randomness in atomic positions. On the other hand, even though the definition of crystals does not imply it, in real life they always have imperfections like vacancies and dislocations. Some inhomogeneous materials, such

as composites or porous materials, can be regarded as homogeneous and disordered at sufficiently large length scale. In glasses basic building blocks are local structural motifs but there is some freedom in their orientations and positions. An ordering at the scale of few building blocks from the chosen one soon disappears already at an intermediate scale [38].

Jammed systems as disordered systems are of particular interest for this thesis. Jamming is the onset of the rigidity in amorphous systems. In many disordered particle systems like colloids, foams, emulsions and granular materials a jamming transition from fluid-like to solid-like states [5] can be observed. According to a paradigm presented in [29] all these phenomena could be unified through a jamming diagram. It includes three parameters: thermodynamic temperature T, inverse packing fraction $1/\phi$, and shear stress τ. At low parameters jamming becomes possible and for low enough values even almost certain. Additionally, we have to consider finite size effects as we have to deal with systems with limited amount N of particles.

A true milestone was the discovery of the jamming point J on the zero stress and zero temperature axis. Below it $\phi < \phi_J$ (corresponding to $1/\phi_J < 1/\phi$) jamming is not possible because of mechanical constraints [17]. The probability that system will jam increases with growing $\phi \geq \phi_J$, also the bigger the ϕ the wider the range of shear stresses $0 \leq \tau \leq \tau_\phi$ that allows for jamming [36]. From a theoretical perspective a scaling ansatz was proposed at the critical jamming transition [18], showing possibilities for a renormalization group theory for jamming. However, jamming transition turns out to be of mixed

first-/second-order character, complicating its theoretical description even more [22]. We can conclude that disordered solids have similar behaviour to some extent. Can we actually obtain disordered materials with extremely different properties?

1.2 Designer matter

We work with jammed systems whose particles are soft spheres which are larger than individual atoms but much smaller than the material itself. From this perspective we also deal with soft matter. As its name may suggest these are easily deformable materials with low elastic moduli.

Soft matter can be dissipative, disordered, far from equilibrium, nonlinear, thermal and entropic, slow, observable, gravity affected, patterned, nonlocal, interfacially elastic, memory forming, and active [33]. It has applications in almost all natural sciences: from chemistry and biology to astrophysics.

In soft matter physics some of the problems are especially challenging [19]. Among those not only mentioned jamming (see section 1.1) but also designer matter are related to the topic of this thesis. The last problem can be attacked from different directions by tuning the architecture (selecting the appropriate geometry [13]) or the composition (designing particles, our global goal) of the material. Particles' shape and size can greatly affect jamming probability [21] and even entirely change the class of the material [10].

But what does it actually mean to design particles? It is not working with *a priori* given particles or choosing them through trial and error. An example of such approach is [10] where the authors explore the properties of the materials consisting of particles of shapes that were chosen in advance. Designing particles can be broken down into two parts. First, you need to identify how different particle parameters affect material properties. Second, you target some material property and determine values of particle parameters that would realize it. Both tasks are extremely complex. Different approach in the field can be found in [27, 11, 12, 1]. Speaking about topic of this work, the most important parameter for jammed systems is the contact number, or the average number of the neighbouring particles. It is mainly affected by the shape of the particles and their inner elasticity [26]. Obviously, the contact number hugely depend on the particle shape. Therefore, it is natural to choose a shape as a particle parameter affecting material properties.

Depending on the goals, computational capabilities and chosen means to model systems (about ours see section 2.4) ways to parametrize and approximate the shape vary a lot. Moreover, the wrong choice of the former can significantly limit the possibilities of the latter. In other words, from the practical perspective one must find a balance between the shape generality and the number of the parameters involved in its definition. We will discuss some of the constructive approaches (building the geometry from scratch following an algorithm) and leave aside the whole class of deformation modelling (examples and details in [2]).

Shape parametrization can be purely geometrical [8, 30] decrib-

ing the shape's boundary, for example, as a set of nodes with their coordinates and edges between them as graphs of functions from a chosen class (lines, polynomials etc). On the one hand it may allow to precisely encode a given shape. On the other hand the amount of parameters can be too big, some parameters can be hard to interpret and it completely ignores internal structure of the particles concentrating on the boundary description only. It has to be worked out separately somehow afterwards, for example, using Voronoi tesselation, introducing additional layers of complexity.

In this thesis we do not need to approximate shapes but it is still worth looking at many insightful methods of shape approximations coming from the field of computer graphics. Among them are developable wrapping [25], bounding proxies [7], geometry-aware bases [40] and variational [9] shape approximation algorithms. These models are usually created for purposes different from ours and almost always tuned to 3 dimensions. Their effectiveness is mostly determined by the ability to preserve the deteriorating subjective visual quality balanced by the rendering speed-up, less often by the physical simulation speed-up or even by its accuracy. Similarly to the boundary-based shape parameterization introduced in the previous paragraph, the major drawback for us is the complexity of the introduction of interactions of individual parts of the objects with each other, including those within the same object.

Building on the ideas set out here, one may look at the volume model introduced in [37] and further developed in [41]. Its first part provides an algorithm that could be especially suitable for our model

construction (see section 2.2) representing any shape as a set of overlapping circles so that their union would be similar to the target shape. In principle, it allows to approximate any given shape. But the price of a more detailed approximation will be a significant increase in the required resources (time and memory) for the simulation.

Our particle model (similar to the one described just above) gives control over its shape and inner elasticity simultaneously. Adjusting them – namely distances between circles – we aim to achieve desirable material properties. In this work we start with some particle shape and measure the value of a chosen property in the resulting material. Next we differentiate the property over the particle's parameters. The most interesting feature is that we do it over the whole process of the material creation (see section 2.4.1). After it we would be able to use any gradient descent method to minimize (maximize) the property changing the parameters accordingly. Moreover, dependence of the property on the parameters may be probabilistic. Therefore, this strategy could sound unrealistic but similar approach has already been shown to be effective in designing kinetics of the self-assembly [16]. For this reason we expect that it may be fruitful in our case as well.

2 Materials and methods

This section's purpose is to establish a theoretical and computational ground of the entire thesis. With the notions from this section we will be able to describe the exact setting of our modelling and our results. First, we explain the assumptions we make about our model and present the model itself. Then we provide the general theoretical description of the chosen model and useful definitions. Finally, we present the computational tools we use as well as a theoretical framework behind them.

2.1 Assumptions

For further convenience we present the assumptions in the form of a list.

1. We consider jammed systems in which all its structural blocks are completely identical. To be more precise, all particles are of the same shape, size, internal structure and mechanical properties. Our systems can be also considered as granular materials.

2. Frictionless particles interact with each other via normal contact repulsive forces (see section 2.3.1). They have internal structure allowing them to deform but at an energy cost.

3. We limit ourselves to 2-dimensional case. However, whenever it is possible we will describe the general case. The dimensionality

of the space is denoted with d. Moreover, every system is in the cubic box $L \times ... \times L$ with periodic boundary conditions (in our case it is a box $L \times L$).

4. We consider systems at zero temperature. Generally, it implies that they must be in the state with the minimum possible energy according to the third law of thermodynamics. Our systems end up in local minima. In other words, we deal with athermal systems far from thermodynamical equilibrium. For the details of the realization see section 2.4.3.

2.2 Particle model

In the section 1.2 we have already discussed different approaches to shape parametrizations. The choice of a model is driven by our computational capacities (see section 2.4) as well as our goals. Namely, the model we descibe below is scalable, its parameters and their changes are easily interpretable, particle's internal stucture arises naturally and its architecture is customizable. Consequently, the particles are deformable and their stiffness is adjustable (and even of their individual parts).

It is important to note that the content of this section is applicable to every particle separately. For the particle-particle interactions take a look at the section 2.3.

We approximate a particle (for an example see the section 3.1) of an arbitrary shape as a set of m (possibly overlapping) spheres

$(S_1, S_2, ..., S_m)$ together with their radii $(R_1, R_2, ..., R_m)$ and relative coordinates

$$(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m). \tag{2.2.1}$$

The spheres are linked by the springs of lengths $\mathbf{l} = (l_{1i}, ..., l_{mj})$ between their centers. The spring l_{ij} connects the spheres S_i and S_j. Regarding spheres' centers as vertices of the graph Γ and springs between them as its edges, we can also reasonably demand Γ's connectivity for the "particle" in our defition to be a single particle. The details of the topology of the spring network (for example, connecting only nearest neighbours), amount, relative sizes and placement patterns of the spheres should be chosen according to need. For example, higher amount of spheres can improve the precision of the approximation. As we claimed before, it is clear from the description how such particles can be deformed. Deformations cause stresses within a particle to which its other subparts have to respond.

Additionally we introduce such simplifications:

- All spheres are of the same size

$$R_1 = R_2 = ... = R_m = R. \tag{2.2.2}$$

- We consider relatively small amount of spheres so we can afford connecting all spheres with each other. Therefore, $\mathbf{l}$ becomes $(l_{11}, l_{12}, ..., l_{23}, l_{24}, ..., l_{(m-1)m})$.

- Within a particle spheres interact with each other exclusively via springs (see section 2.3.1, also compare [34]).
- Springs' lengths are equal to the initial distances between centers of the spheres. We can derive their values using 2.2.1 as

$$l_{tt'} = |\mathbf{x}_t - \mathbf{x}_{t'}|. \tag{2.2.3}$$

Again, whenever it is possible, we will attempt to describe the general case beyond the simplifications above.

2.3 Theoretical implementation

Consider a system of N particles in d dimensions. Recall that each particle consists of m spheres connected with each other via springs. In our setting it means that we deal with $N \cdot m$ spheres and $N \cdot (m-1)m$ springs. p's sphere coordinates are denoted by $\mathbf{r}_p$.

The following notation can be a bit complex and lack immediate interpretability but it is not as cumbersome as many simpler options. We explicitly enumerate spheres starting with 1 in a way that first m of them are in the first particle, next m are in the second etc. Moreover, corresponding spheres k and k' in different particles are those whose numbers are congruent modulo m, i.e. if $k \equiv k' (\mathrm{mod}\, m)$. It induces the springs' enumeration. By spring $l_{kk'}$ we mean the spring $l_{kk'}$ such that $k \equiv t (\mathrm{mod}\, m)$ and $k' \equiv t' (\mathrm{mod}\, m)$ and $l_{tt'}$ is a component of $\mathbf{l}$.

2.3.1 Interactions

Any two spheres k and k' from *different* particles interact with a pairwise soft-sphere potential

$$U_{kk'}^{soft}(\mathbf{r_k}, \mathbf{r_{k'}}) = \begin{cases} \frac{\epsilon_1}{\alpha_1}\Big(1 - \frac{r_{kk'}}{R_k + R'_k}\Big)^{\alpha_1} & \text{if } r_{kk'} < R_k + R_{k'} = 2R, \\ 0 & \text{otherwise} \end{cases} \tag{2.3.1}$$

where ϵ_1 sets an energy scale, $\alpha_1 \geq 2$ and $r_{kk'}$ is a distance between their centers

$$r_{kk'} = |\mathbf{r}_k - \mathbf{r}_{k'}|. \tag{2.3.2}$$

Any two spheres k and k' within the same particle interact via a spring $l_{kk'}$ with a potential

$$U_{kk'}^{spr}(\mathbf{r_k}, \mathbf{r_{k'}}) = \frac{\widetilde{\epsilon_2}}{\alpha_2}\Big(l_{kk'} - r_{kk'}\Big)^{\alpha_2}, \tag{2.3.3}$$

where $r_{kk'}$ is defined in 2.3.2.

Notice that the interactions between particles are purely repulsive and the character of the interactions within each individual particle implies that its initial shape is favoured (which is true only due to the simplification about springs' lengths mentioned in the section 2.2).

We consider in both cases harmonic potentials $\alpha_1 = \alpha_2 = 2$. It is easy to see how similar these potentials are. We can replace $\widetilde{\epsilon_2}$ with $\frac{\epsilon_2}{(2R)^{\alpha_2}}$ to work with a dimensionless ratio $\frac{\epsilon_1}{\epsilon_2}$. $\frac{\epsilon_1}{\epsilon_2}$ can be viewed as an approximate measure of the relation between the two forces: from 0

(spheres interact only via springs) to $+\infty$ (only spheres from different particles interact). Then the spring potential can be rewritten as

$$U^{spr}_{kk'}(\mathbf{r_k}, \mathbf{r_{k'}}) = \frac{\epsilon_2}{\alpha_2}\Big(\frac{l_{kk'}}{2R} - \frac{r_{kk'}}{2R}\Big)^{\alpha_2}. \tag{2.3.4}$$

Notice that in our setting every two spheres interact either through a soft sphere potential or though a spring potential. It allows us to introduce a potential between any pair of spheres $U_{kk'}$. To keep it compact we also introduce a function $\mathbb{S}$ such that $\mathbb{S}(k, k') = 1$ if spheres are in the same particle and 0 otherwise

$$U_{kk'} = U^{spr}_{kk'}\mathbb{S}(k, k') + U^{soft}_{kk'}(1 - \mathbb{S}(k, k')). \tag{2.3.5}$$

2.3.2 Useful definitions

For the future convenience we introduce a list of definitions which will be used in the next sections.

1. The total potential U can be expressed using compact form from the end of the previous section as

$$U(\mathbf{r}) = \frac{1}{2}\sum_{k\neq k'}^{k,k'} U_{kk'}(\mathbf{r_k}, \mathbf{r_{k'}}). \tag{2.3.6}$$

where $\mathbf{r}$ is a vector of all $N \cdot m \cdot d$ coordinates of spheres comprising particles. However, we look at the total potential from a slightly

different perspective. One can also consider U as a function of particle parameters, specifically $\mathbf{l}$. Recall that $\mathbf{l}$ determines corresponding springs in *every* particle but not all distinct springs in the system. In other words, changing any component $l_{kk'}$ $(1 \leq k, k' \leq m,\ k \neq k')$ of $\mathbf{l}$ would lead to the simultaneous change of the corresponding springs $l_{tt'}$ in all particles. And it, in turn, would affect U^{spr} which is a component of U. Therefore, we may write $U(\mathbf{r}; \mathbf{l})$. However, we will drop $\mathbf{l}$ from the notation for the most of the time.

2. The stress tensor σ_{ij} is a $d \times d$ stress tensor

$$\sigma_{ij} = \frac{1}{2V} \sum_{k \neq k'}^{k,k'} \Big(\frac{\partial U_{kk'}}{\partial r_{kk'}} \Big) \frac{r_i r_j}{r_{kk'}} \tag{2.3.7}$$

where V is a total volume of the system, r_i and r_j are components of $\mathbf{r}_{kk'} = \mathbf{r_k} - \mathbf{r_{k'}}$.

The pressure p is related to the trace of the stress tensor simpy as

$$p = -\frac{1}{d} \operatorname{Tr} \sigma_{ij}. \tag{2.3.8}$$

3. The global affine deformation to the lowest order is defined by a strain tensor ϵ_{ij} and transforms every vector r_i as

$$r_i = r_i + \sum_j \epsilon_{ij} r_j \tag{2.3.9}$$

where ϵ_{ij} obeys a symmetry requirement $\epsilon_{ij} = \epsilon_{ji}$ and in 2-dimensional

case has a form

$$\epsilon = \begin{pmatrix} \epsilon_{xx} & \epsilon_{xy} \\ \epsilon_{xy} & \epsilon_{yy} \end{pmatrix}. \tag{2.3.10}$$

2.3.3 Linear dilatancy

Here we define the linear dilatancy (one of the elastic moduli), the material property we target in our systems. However, it will require performing a lot of auxiliary work and introducing quite a few technical definitions.

The dilatancy is the phenomenon of changing volume in granular materials subjected to shear deformations. First discovered by Reynolds [3], it has been extensively scrutinized from both theoretical (e.g. [24, 28]), numerical (e.g. [4]) and experimental perspectives (e.g. [20]) since then. Different parameters can be used to measure this effect. We consider linear dilatacy D defined at the end of this section. We will consider D as a function of $\mathbf{l}$ and try to understand the nature of this dependence by examining its gradient's behaviour.

The following is according to [15]. In a stable mechanical system after an affine transformation there is usually a secondary non-affine response. It can be calculated within the harmonic approximation [14]. The change in the total energy is approximately given by

$$\frac{\Delta U}{V^0} = \sigma^0_{ij}\epsilon_{ji} + \frac{1}{2}c_{ijkl}\epsilon_{ij}\epsilon_{kl}. \tag{2.3.11}$$

Where V^0 and σ^0_{ij} are initial state volume and stress tensor respectively. In our case $V^0 = L^d$. c_{ijkl} is the $d \times d \times d \times d$ elastic modulus tensor. Details on its derivation can be found in [14].

We can also rewrite 2.3.11 by introducing an enthalpy-like function $H = U - \sigma^0_{ij}\epsilon_{ij}$

$$\frac{\Delta H}{V^0} = \frac{1}{2} c_{ijkl}\epsilon_{ij}\epsilon_{kl}. \tag{2.3.12}$$

From the symmetry requirement $\epsilon_{ij} = \epsilon_{ji}$ we can derive such symmetries of c_{ijkl}:

$$c_{ijkl} = c_{ijlk} = c_{jikl} = c_{klij}. \tag{2.3.13}$$

Some of the global affine deformations are similar. To understand in which way, imagine you rotate a system and apply the same ϵ as before but in the new coordinate system. However, the manipulations are the same, in the old coordinate system it would correspond to a rotated ϵ. We define $\epsilon(\theta) = \mathrm{M}_{-\theta}\epsilon\mathrm{M}_\theta$ where M_θ is a rotational matrix

$$\mathrm{M}_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}. \tag{2.3.14}$$

In an isotropic system rotations of ϵ correspond to appropriate rotations of the tensor c_{ijkl}. Notice that when $\theta = 90°$ values of c_{ijkl} change in a way that is equivalent to exchanging x and y. Our systems

are finite and we expect them to be anisotropic. Therefore, we can examine dependence of c_{ijkl} on θ to understand a measure of system's anisotropy.

Any modulus $R = 2\frac{\Delta H}{V^0}$ to a linear order is

$$\begin{aligned} R &= c_{ijkl}\epsilon_{ij}\epsilon_{kl} \\ &= c_{xxxx}\epsilon_{xx}^2 + c_{yyyy}\epsilon_{yy}^2 + 2c_{xxyy}\epsilon_{xx}\epsilon_{yy} \\ &+ 4c_{xyxy}\epsilon_{xy}^2 + 4c_{xxxy}\epsilon_{xx}\epsilon_{xy} + 4c_{yyxy}\epsilon_{yy}\epsilon_{xy}. \end{aligned} \tag{2.3.15}$$

Taking this into account, R can be actually considered a function of an angle θ and we define $R(\theta)$ the same way as in 2.3.15 but with $\epsilon_{ij}(\theta)$ instead of ϵ_{ij}. We can average across θ integrating out anisotropic fluctuations in the system

$$R_{DC} = \frac{1}{\pi}\int_0^\pi R(\theta)d\theta. \tag{2.3.16}$$

Similarly, we also find its variance R_{AC} which is, in turn, the measure of fluctuations

$$R_{AC}^2 = \frac{1}{\pi}\int_0^\pi (R(\theta) - R_{DC})^2 d\theta. \tag{2.3.17}$$

To define the linear dilatancy we introduce quantities

$$
\begin{aligned}
G_0 &= c_{xyxy} \\
G_{\frac{\pi}{4}} &= \frac{1}{4}(c_{xxxx} + c_{yyyy} - 2c_{xxyy}) \\
A_2 &= -\sqrt{\frac{1}{4}(c_{xxxx} - c_{yyyy})^2 + (c_{xxxy} + c_{yyxy})^2} \\
\phi_2 &= \tan^{-1}\Big((c_{yyyy} - c_{xxxx})/2(c_{xxxy} + c_{yyxy})\Big) \\
A_4 &= -\frac{1}{2}\sqrt{(c_{xxxy} - c_{yyxy})^2 + (G_0 - G_{\frac{\pi}{4}})^2} \\
\phi_4 &= \tan^{-1}\Big((G_0 - G_{\frac{\pi}{4}})/(c_{xxxy} - c_{yyxy})\Big).
\end{aligned}
\tag{2.3.18}
$$

Shear modulus G can be understood as a response to the affine transformation $\epsilon_{xx} = \epsilon_{yy} = 0$ and $\epsilon_{xy} = \frac{\gamma}{2}$ (γ is the parameter allowing us to control the scale of the deformation). It gives us a family of strain tensors

$$
\epsilon_G(\theta) = \frac{\gamma}{2}\begin{pmatrix} \sin 2\theta & \cos 2\theta \\ \cos 2\theta & -\sin 2\theta \end{pmatrix}. \tag{2.3.19}
$$

We obtain

$$
G(\theta) = \frac{1}{2}\Big(G_0 + G_{\frac{\pi}{4}}\Big) - A_4 \sin(4\theta + \phi_4). \tag{2.3.20}
$$

Also you can notice that $G(0) = G_0$ and $G(\frac{\pi}{4}) = G_{\frac{\pi}{4}}$.

It is easy to see from definitions 2.3.16 and 2.3.17

$$
\begin{aligned}
G_{DC} &= \frac{1}{2}\Big(G_0 + G_{\frac{\pi}{4}}\Big) \\
G_{AC} &= \frac{A_4}{\sqrt{2}}.
\end{aligned}
\tag{2.3.21}
$$

In a similar way we define the uniaxial compression $U(\theta)$ with $\epsilon_{xx} = \gamma$ and $\epsilon_{xy} = \epsilon_{yy} = 0$ leading to

$$\epsilon_U(\theta) = \frac{\gamma}{2} \begin{pmatrix} 1 + \cos 2\theta & -\sin 2\theta \\ -\sin 2\theta & 1 - \cos 2\theta \end{pmatrix}. \tag{2.3.22}$$

We calculate

$$U(\theta) = B + G_{DC} + A_4 \sin(4\theta + \phi_4) + A_2 \sin(2\theta + \phi_2) \tag{2.3.23}$$

so that

$$\begin{aligned} U_{DC} &= B + G_{DC} \\ U_{AC} &= \sqrt{\frac{1}{2}(A_2^2 + A_4^2)}. \end{aligned} \tag{2.3.24}$$

Where B is the bulk modulus defined as the response to the uniform compression with $\epsilon_{xx} = 0$ and $\epsilon_{xx} = \epsilon_{yy} = \frac{\gamma}{2}$. In this case the strain tensor is proportional to the unit one so it does not depend on the angle (also implying $B_{AC} = 0$) and has a simple form

$$\epsilon_B(\theta) = \frac{\gamma}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{2.3.25}$$

The calculation of B is straightforward

$$B = B_{DC} = \frac{1}{4}(c_{xxxx} + c_{yyyy} + 2c_{xxyy}). \tag{2.3.26}$$

Finally, we have all the components to define the linear dilatancy D. First, we need to find a response $R(\theta)$ to the deformation given by $\epsilon_{yy} = 0$ and $\epsilon_{xx} = \epsilon_{xy} = \frac{\gamma}{2}$

$$\epsilon_R(\theta) = \frac{\gamma}{4}\begin{pmatrix} 1 + \cos 2\theta + 2\sin 2\theta & 2\cos 2\theta - \sin 2\theta \\ 2\cos 2\theta - \sin 2\theta & 1 - \cos 2\theta - 2\sin 2\theta \end{pmatrix}. \tag{2.3.27}$$

We can express D in terms of R as

$$D(\theta) = R(\theta) - \frac{1}{4}U(\theta) - G(\theta). \tag{2.3.28}$$

After cumbersome calculations we obtain

$$D(\theta) = -A_4\cos(4\theta + \phi_4) - \frac{A_2}{2}\cos(2\theta + \phi_2) \tag{2.3.29}$$

and again

$$\begin{aligned} D_{DC} &= 0 \\ D_{AC} &= \sqrt{\frac{1}{8}(A_2^2 + 4A_4^2)}. \end{aligned} \tag{2.3.30}$$

Notice that its value is nonzero in almost all directions but its mean D_{DC} is 0 for any system so it is natural to consider D_{AC} as a

value representing fluctuations of D.

2.4 Computational implementation

We model our systems in Python by the means of the libraries *jax* and *jax-md* [39]. They provide us with built-in fuctions producing systems of soft spheres and executing automatic differentiation over the whole minimization process. This section we begin with the notion of the automatic differentiation. Then we proceed to system initialization and minimization. The final paragraphs bring up how much they affect each other.

2.4.1 Automatic differentiation

A prominent property (actually by definition) of a derivative is that it reflects how the rapidly function changes in a given point. Its multidimensional analogue is a gradient, showing componentwise how fast function changes in different directions independently. Gradient as a vector also has an important interpretation. It points in the direction in which the function changes the fastest. For example, we can go in this direction and iteratively increase the value of the function. This way we can (locally) maximize (or minimize) the function finding appropriate arguments.

For us it is important from two perspectives. First, we need gradients to minimize the energy function for every system we handle.

Every system requires thousands of gradient calculations. Moreover, we differentiate these calculations themselves. Therefore, it becomes a crucial task to perform them in the most efficient way. We choose automatic differentiation for these purposes. Second, once we have a minimized system we can measure any property and calculate its gradient with respect to the chosen particle property, namely spring lengths. Similarly, we could use this gradient to minimize (or maximize) the system's property.

Automatic differentiation is an algorithmic technique to calculate derivatives. To understand automatic differentiation conceptually, first, we need to clarify that it is neither a symbolic differentiation nor a finite differentiation. Its advantages are efficiency (cost to compute the derivative is linear in the cost to compute the value of the function that is differentiated), numerical stability and accurateness (derivatives are calculated with machine precision) [35].

Second, we introduce basic concepts necessary to understand how automatic differentiation operates

- Almost all computer programs consist of sequence of basic arithmetic operations (like summation) and elementary functions (such as cos, exp etc).
- We can represent their derivatives using same basic arithmetic operations and elementary functions in a known way. For example, if at some point you need to calculate a sum $a + b$ of previously calculated values a and b, the corresponding calculation of the derivative would be $(a + b)' = a' + b'$ where we would

already have calculated a' and b' (at the stages corresponding to calculations of a and b respectively).

- Using the chain rule (we omit mathematical requirements as they are almost always met) we can calculate the derivative of $\mathbf{y} = h(\mathbf{x}) = f(g(\mathbf{x}))$ as

$$\frac{\partial h}{\partial \mathbf{x}} = \frac{\partial f}{\partial \mathbf{z}} \frac{\partial g}{\partial \mathbf{x}} = \frac{\partial \mathbf{y}}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \mathbf{x}} \tag{2.4.1}$$

  where $\mathbf{z} = g(\mathbf{x})$ and $\mathbf{y}$ is viewed as $\mathbf{y} = f(\mathbf{z})$. For example, if we calculate $e^c = d$ for a previously calculated value c, the corresponding calculation of the derivative would be $(e^c)' = e^c \cdot c'$ $= d \cdot c'$ where we would have already calculated d and c' (at the stage corresponding to the calculation of c).

Finally, we see how to decompose derivative calculation of any complexity into a sequence of simpler operations in an algorithmic way. Depending on the goal and available resources (time and memory) there are two possible implementations: forward (more memory) and reverse (more time) modes. Sometimes it is beneficial to combine them, for example, calculating the Hessian matrix of the function.

In a forward mode after every calculation step we evaluate the derivative corresponding to it and carry both values to the next step so that we obtain the derivative right after obtaining the function value. In a reverse mode we store the result of every calculation (the forward phase). Additionally we need to create a computational graph displaying the relationships between these steps. We use the calculated values to find the derivative during the so-called backpropagation through the

graph (the backward phase). For details and examples of both procedures we refer to [23].

As it is mentioned in the section 2.3.3, we are interested in the behaviour of the derivatives of D_{AC} with respect to $\mathbf{l}$. But it is important to note that we will take them over the whole minimization process. Here we explain how it can be realized.

1. We start with some random configuration $\mathbf{r}^0$ (see section 2.4.2) for which we can calculate $U(\mathbf{r}^0; \mathbf{l})$ (but $\mathbf{r}^0$ is regarded as a variable independent from $\mathbf{l}$, however, in our case there is a clear relationship between them, see section 2.4.2).

2. Next, we change the coordinates of the spheres following the FIRE algorithm until the stopping condition is met (see section 2.4.3) at the M^{th} step. Conceptually every step of the algorithm can be exemplified with the simplest gradient descent step

$$\mathbf{r}^{i+1} = \mathbf{r}^i - \alpha \nabla U(\mathbf{r}^i). \tag{2.4.2}$$

   - We see that $\mathbf{r}^{i+1}$ is a function of the derivatives of $U(\mathbf{r}^i; \mathbf{l})$ and $\mathbf{r}^i$ (other algorithms may include additional terms, α may become a function itself).
   - For $i > 0$ it implies that $\mathbf{r}^{i+1}$ is the function of $\mathbf{l}$ (and $\mathbf{r}^i$).
   - Finally, by induction $\mathbf{r}^{i+1} = \mathbf{f}^{i+1}(U(\mathbf{r}^i(\mathbf{r}^0, \mathbf{l}); \mathbf{l}), \mathbf{r}^i(\mathbf{r}^0, \mathbf{l}))$, or $\mathbf{r}^{i+1} = \mathbf{r}^{i+1}(\mathbf{r}^0, \mathbf{l})$.

3. Having a minimized system we can measure its elastic moduli c_{ijkl} which are the functions of $U(\mathbf{r}^M)$, its first and second derivatives.

Taking into account written above we see that we can consider c_{ijkl} as a function of $U(\mathbf{r}^M(\mathbf{r}^0, \mathbf{l}); \mathbf{l})$ and its derivatives.

D as a certain combination of c_{ijkl} moduli can be considered a function of $\mathbf{r}^0$ and $\mathbf{l}$ following the reasoning above.

2.4.2 Initialization

To initialize a system of N particles in 2 dimensions we require a set of particle positions $(\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_N)$ and their orientations represented by angles $(\psi_1, \psi_2, ..., \psi_N)$, or in total $2N + N = 3N$ numbers. Now it is easy to calculate initial positions of spheres having 2.2.1. For example, sphere k in a particle p (which we enumerate as $v = (p-1)N + k$) will be on a position

$$\mathbf{r}_v = \mathbf{R}_p + \mathrm{M}_{\psi_p}\mathbf{x}_k \tag{2.4.3}$$

where M_ψ is a rotation matrix defined in 2.3.14.

Similar calculations would be in 3 dimensions but we would need at least 3 numbers, e.g. Euler angles, to encode particle's orientation. Also the formulas for the rotations of any objects (e.g. vectors, tensors) change.

Forces between spheres within any particle are vanishing at the initialization step. Notice that it follows from our initial assumptions as spheres in any particle interact only via springs. For two spheres k

and k' within a particle p distance between them is

$$|\mathbf{r}_{(p-1)N+k} - \mathbf{r}_{(p-1)N+k'}| = |\mathbf{R}_p + \mathrm{M}_{\psi_p}\mathbf{x}_k - \mathbf{R}_p - \mathrm{M}_{\psi_p}\mathbf{x}_{k'}| = \tag{2.4.4}$$

$$= |\mathrm{M}_{\psi_p}(\mathbf{x}_k - \mathbf{x}_{k'})| = |\mathbf{x}_k - \mathbf{x}_{k'}| = l_{kk'},$$

which is the corresponding spring's length so the force between them is exactly 0. We used here 2.2.3 and that from 2.3.14 follows $\det \mathrm{M}_{\psi_p} = 1$. Generally, it does not hold during the minimization process when the particles deform (by the deformation we mean the change of relative distances between spheres) interacting with each other.

2.4.3 Minimization

To prepare a new system one would usually start from effectively infinite temperature (random configuration, more details in the end of this section) and end up at zero temperature (in some local energy minimum) using gradient descent algorithm. However, we start with configurations that are not random. We have not proved exactly that it would not change the outcome.

Because we consider systems at zero temperature and all forces are conservative, the minimized state should not depend on the algorithm. However, some methods with added noise may still end up in different local minima. We use the FIRE algorithm [6] which is one of the most efficient in our setting. Default parameters from jax-md library turn out to work sufficiently good.

In a local energy minimum for every sphere force balance must hold. We can use this fact to set a meaningful stopping condition for a minimization procedure

$$\max_{k,i} |f_{k,i}| < f_0 \tag{2.4.5}$$

where $f_{k,i}$ is a component i of the total force acting on the sphere k. f_0 is a constant that we choose to guarantee residual forces to be small comparable to the typical force scale in the system.

Additionally we can impose another condition. We stop the procedure unconditionally after M steps and discard such systems as they do not meet the first stopping condition. The value of M we determine from the empirical experience. It must satisfy several conditions. First, most systems should be minimized by that point. Second, most of the systems which are not minimized by that point will fail to be minimized even after K steps where K must be unreasonably big.

In principle, such limitation must not affect the final ensemble a lot. Because of it we will not specify exact criteria (what is "most systems" and how big K must be). Main purpose of these conditions is to maximize the effectiveness of the procedure - maximum amount of eligible systems in minimal time. Moreover, keeping this number low is essential in our context. For us every additional step increases amount of memory necessary to perform the differentiation over the whole minimization procedure (see section 2.4.1).

In order to inspect the whole phase space of initial configurations, $\mathbf{r}^0$ should be picked from the multidimensional uniform distribution

$\mathfrak{U}_{[0,L]^{dN}}$ or, equivalently, every component of every $\mathbf{R}_p$ from $\mathfrak{U}_{[0,L]}$. But it allows for a variety of extremely overlapping initial configurations. It results in many minimized systems with spheres from one particles stuck in other particles - completely legit systems from minimal energy perspective but totally impossible in practice. For relatively simple geometries it can be fixed by including dummy spheres as in [31] that prevent spheres from other particles to squeeze in. Another problem is that such initial configurations have comparatively huge energy gradients (forces). In this case standard FIRE parameters may fail (the minimization will not end in a meaningful time if ever) causing us to manually adjust them, usually by few orders decreasing the step size as well as its permissible maximum. However, such a minimization can take an overwhelmingly long time.

Our solution is to substantially limit the phase space of initial configurations. In the section 3.1 we specify our approach. General idea is to start from the configurations with minimal amount of overlaps, or locally sparse (by this we mean the system with low local density in every point). There are two justifications for this. Firstly, real-life systems would jam in similar conditions. Secondly, what matters is the final (minimized) state. And there is a certain region of a phase space around the local minimum whose points all can "fall" only to it with a standard gradient descent algorithm. Reasonable final configurations are also locally sparse. Therefore, it is meaningful that during minimization all configurations will become locally sparse from some moment. Therefore, we should not miss any realistic final states with our approach.

3 Results

This section is devoted to the essential part of any research - its results. But first we describe the exact set up so that everyone could reproduce our results.

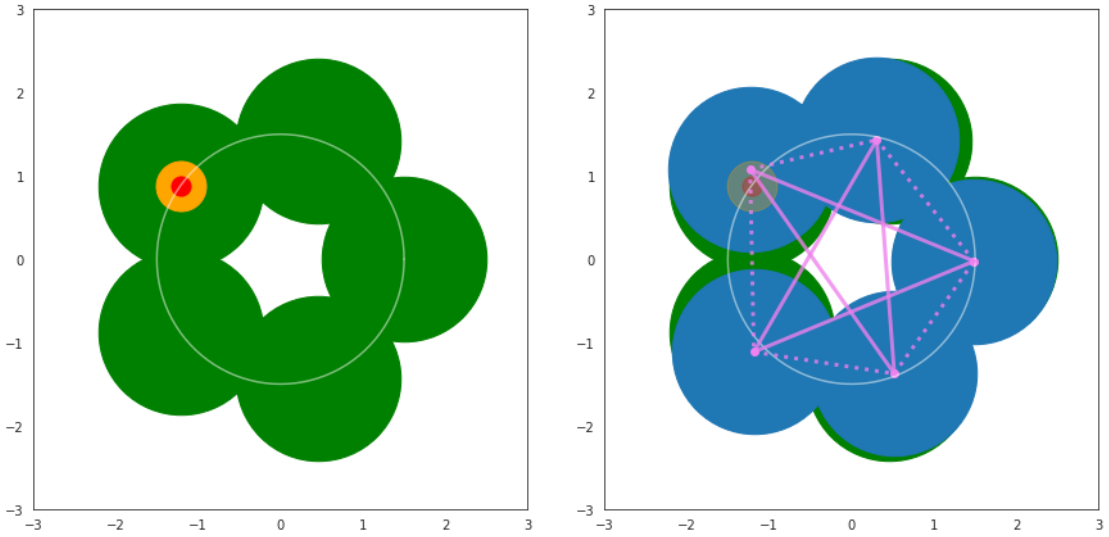
3.1 Setup

- Particle. We work with the particle consisting of 5 spheres of radius $R = 1$ (see figure 1). Their centres are placed equidistantly on the circle of $r = 1.5$ (basically forming a pentagon) and then each of their coordinates is disturbed with a value from the normal distribution $\mathcal{N}(0, 0.1)$ (on the figure 1 for one of the spheres we mark where its centre will end up with probability 50% in red and 99% in orange). There are two types of springs: shorter ones between neighbouring spheres ($l = 1.763$ before disturbance) and longer ones between other pairs ($l = 2.853$).

- System and initialization. System consists of $N = 81$ particles in the square box with sides $L = 37.6$ and periodic boundary conditions so that the density is 0.9. Particles are placed at the square lattice with the spacing $37.6/9 = 4.2$ and then each coordinate is disturbed with a value from the normal distribution $\mathcal{N}(0, 0.25)$. The angles for their orientations are taken from the uniform distribution $\mathfrak{U}_{[0,2\pi]}$ (for the example of the initiation see figure 2a).

- Potential. We set $\epsilon_1 = 1$ and $\epsilon_2 = 40$. It may seem that such particles must be quite stiff. However, a glance at figure 2b allows to notice with the unaided eye that particles are deformed. This values were manually customized to meet two criteria: particles are deformable but most likely will still keep the visual resemblance with the initial state.

- Minimization. For our purposes it is sufficient to set $f_0 = 10^{-12}$ [14]. Empirically derived number of steps after which we uncondinitionally stop the minimization is $M = 2000$. Any system which is not minimized by that moment is skipped. The example



(a) Before **(b)** After

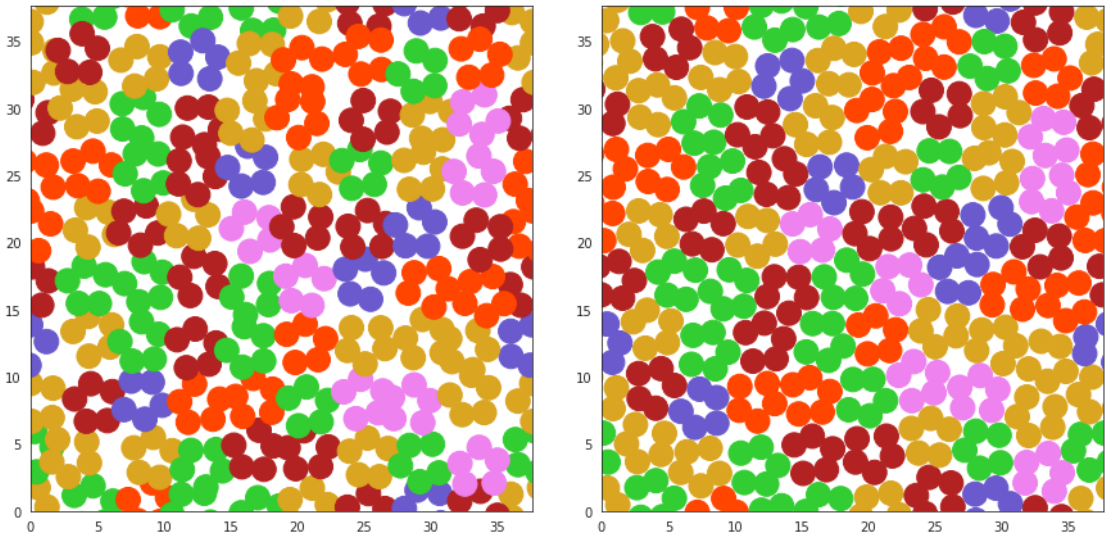
Fig. 1. Particle shape before (green) and after (blue) the gaussian disturbance. Centers of the spheres are marked with violet dots. The springs are violet lines (longer - sharp, shorter - dashed). White circle with $r = 1.5$ is centered at the origin. Red ($r = 0.117$) and orange ($r = 0.303$) spheres show areas where the underlying sphere's centre will end up after disturbance with 50% and 99% probability respectively.

of the minimized state see on the figure 2b.

3.2 Data

In total we consider 23 different randomly generated (see section 3.1) particle shapes.

For every particle shape we produce random initial configurations. We minimize each of them until any stopping condition is met. Again, we always check the condition on the forces and if it is not met, we discard such configuration. We measure the linear dilatancy D (D_{AC}) of the remaining systems. It gives us a distribution of D values as on



(a) Initialized **(b)** Minimized

Fig. 2. Example of the same system in the initial state (**a**) and after energy minimization (**b**). Spheres in the same particle are always the same colour. Different colours are used only to help distinguish individual particles.

the figure 3. It allows us to fix a universal cutoff of 1 regarding systems with higher D as unrealistic.

For each of the remaining systems we run the minimization and an automatic differentiation of D with respect to $\mathbf{l}$ but this time for a fixed amount of steps $M = 2000$ (so that the amount of necessary steps is not the function of $\mathbf{l}$ itself). This way we obtain gradients. We can also check their norms to identify the suspicious outliers. 4.5 is a sensible cutoff (see the figure 4) after examining the overall shape of the distribution of the norms. Again, we discard the systems with gradients of a bigger norm. It continues until we find at least 50 suitable ones (at most around 600 configurations).

Finally, for every shape we can take all the gradients and calculate their mean componentwise (and the variance of their distribution as well). The resulting vector we call the mean gradient (of D_{AC} with respect to $\mathbf{l}$). Similarly to how we used energy gradients in the section

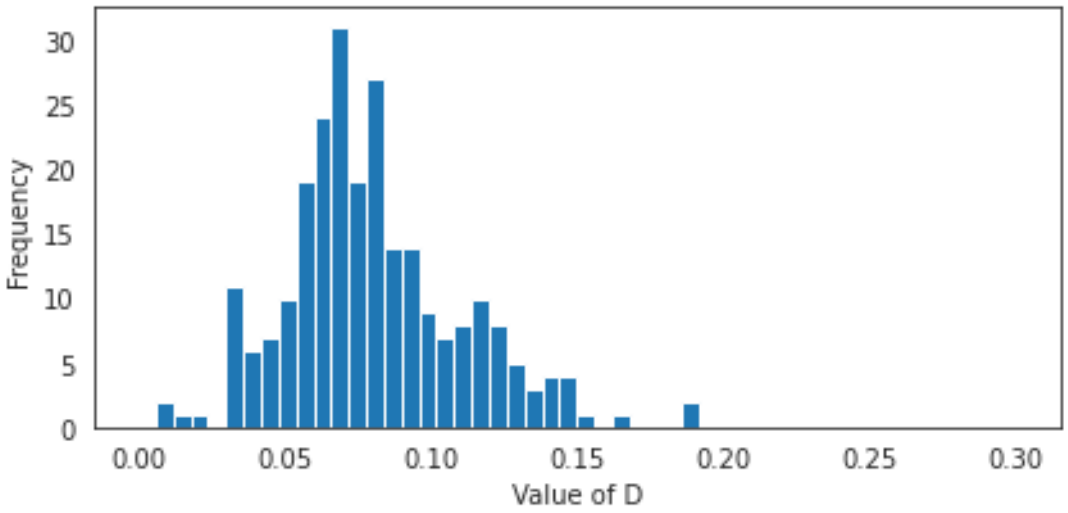


Fig. 3. Distributions of the values of D (252 of 3 different shapes).

2.4.3 to minimize the energy function, we expect that we can minimize D_{AC} changing $\mathbf{l}$ in the direction opposite to the mean gradient (reestimating the mean gradient after each update of $\mathbf{l}$).

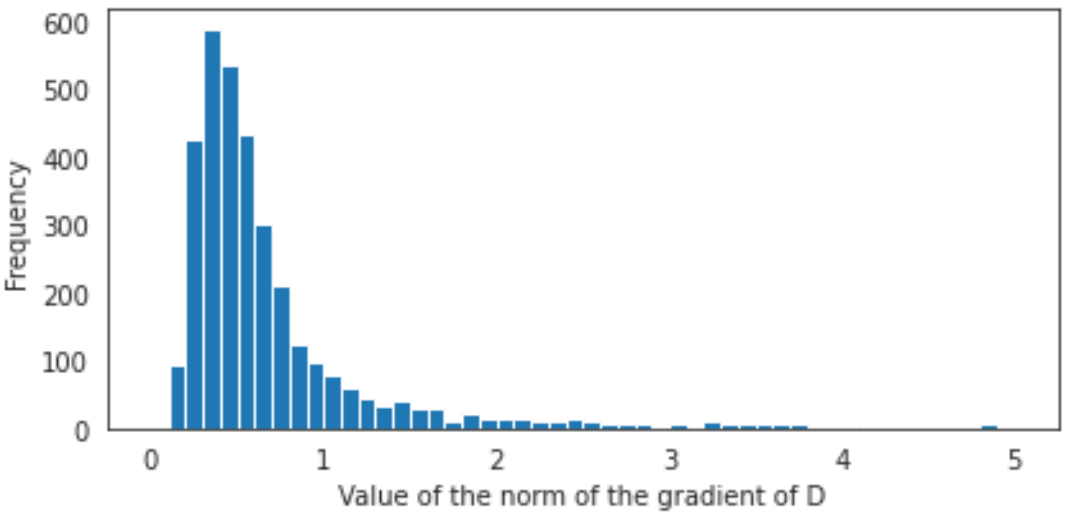


Fig. 4. Distributions of the norms of gradient of D with respect to $\mathbf{l}$ (3485 of all 23 investigated shapes).

4 Discussion

In this final section before the conclusion we analyse and interpret the data from the previous section, critically assess it and provide our view on what could be the further steps to achieve our global goal.

4.1 Analysis

Developing ideas of the previous section we will look at the dependence of the mean gradients on the corresponding **l**. Just as in the figure 1 we will consider shorter and longer springs separately. Along the axis x will be the length of the spring l and along the axis y the corresponding -mean gradient (pointing to the direction minimizing D_{AC}). The result is in the figure 5. Recall that there are 5 shorter and 5 longer springs in each particle so that every particle shape contributes 5 points to each figure. Actually it means that these points should be correlated.

Interestingly, it turns out that for *shorter* springs there is a more general pattern. It can be formulated as "to minimize the fluctuations of the linear dilatancy we need to make the longer springs among them even longer and the shorter ones even shorter". Moreover, the relative magnitude of this change appears to depend approximately linearly on the difference between the spring length and its length before disturbance ($l = 1.763$ marked by the vertical line on the figure 5).

It is possible to give a numerical assessment of this statement. To

do it we calculate the Pearson correlation

$$r = \frac{\mathrm{Cov}(l, dl)}{\sigma_l \sigma_{dl}} \tag{4.1.1}$$

where we denoted dl as the -mean derivative (component of the mean gradient) correspoding to the spring of the length l. Note that for this calculation we consider all data points as independently obtained. It gives us $r = 0.455$ (by its definition $|r| \leq 1$) with the p-value less than 0.00001 (calculated for $r = 0.455$ and the sample size $23 \cdot 5 = 115$). In other words, this correlation should be statistically significant.

Recall that D_{AC} reflects the anistropic fluctuations in the system. Therefore, we come to the counterintuitive conclusion that in order to make the system more isotropic we need to make its constituents less

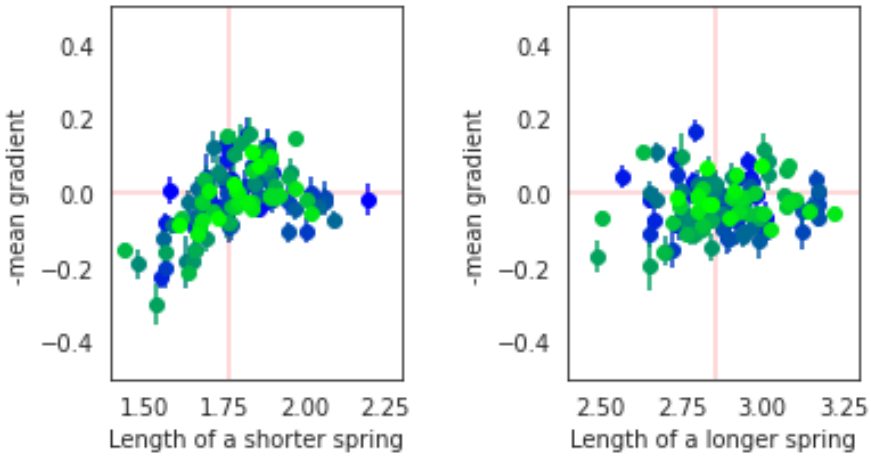


Fig. 5. The relationship between the length of the spring and the correspoding gradient (to be precise, -mean gradient). Points of the same colour correspond to the springs belonging to the same particle type (23 in total). Red vertical lines are at $l = 1.763$ for shorter springs and $l = 2.853$ for longer ones which are the lengths of the respective springs before the disturbance.

symmetric! In our case we can state it because the opposite transformation leads to the perfect pentagon. Explanation of this phenomenon requires further exploration and correct formulations of the questions posed in the process. By now it is hard to say what we observe: a general pattern, a very specific exception, a misleading sign or even a by-product of some conceptual mistake.

Another encouraging observation is that the variance of the gradient distributions is relatively small (see error bars in the figure 5). It can be the sign that following them may consistently change material properties in a desirable manner even though the properties themselves are random variables (consider figure 3).

4.2 Further directions

This research admits a plenty of possible extensions. Here we suggest few of them.

First of all, we chose a very specific setup.

- The amount of the particles in our model is actually too small comparing to real systems. Same calculations for larger systems can both confirm and disprove our findings.
- Different extrema of particles' stiffness can be scrutinized in our context. As well as densities of systems.
- One can actually try to fill the "gap" between longer and shorter

springs (and extend the represented range of l to both sides) by increasing the variance of the disturbance at the stage of particle creation. It would allow to see whether our finding holds in a more general case. By this we mean that we were working in a perturbative mode when the particles still resemble the pentagon.

- The particle's geometry itself is highly customizable. There are many more simple shapes that can be used as toy models with different number of spheres in them, their relative sizes and placements, spring network topologies and assumptions on their lengths etc.

- We considered one specific material property. We mentioned some of the others worth looking like bulk modulus, shear modulus, Poisson's ratio etc.

- We assume particles to be identical. One can consider systems containing slight variations of the same particle, systems with multiple types of particles etc.

- The same approach can be applied not just to jammed systems. For example, systems with variable geometries (you can also look at [32]) can be inspected in a similar way.

Next, we do not follow the gradient to minimize the chosen property. It is interesting for a few reasons.

- It will simply answer whether our approach is fruitful for such problems.

- Shapes that optimize for targeted material properties can be themselves objects of research.
- It can open the way for the study of much more complex particle shapes and sophisticated material properties by means of our approach.
- The last but not least are real-life applications. It may provide us with both materials having improved ordinary qualities (also simpler manufacturing, cheaper or more resource-effective) and metamaterial with unusual properties.

5 Summary and conclusion

In this section we briefly summarize the content of this thesis, point out main results and outline next steps.

Our motivation starting this thesis was to establish a universal framework that would allow to create materials with desirable properties. First, we chose exact material class of jammed systems and how we would like to control their properties through shapes of the particles comprising them. Next, we identified an exact way (gradient-bazed minimization) allowing us to rationally design particle shapes to modify material properties in a desirable way.

The actual procedure includes gradient calculations of the energy function. However, it does not involve conceptually new terms and concepts. It has rather become possible through the good choice of optimization algorithms (automatic differentiation), growing speed of calculations and available memory volumes. As often it happens, combination of these factors led to an approach that was impossible before even for much simpler problems.

Main results of this thesis can be summarized as following. The most obvious result is that we have shown that our approach is realizable in jammed systems. Next, we have obtained enough data to claim that trying gradient descent for our systems will make sense. We justify it by low variance of data points. Probably the most unexpected result is that we may need to decrease the symmetry of the particles in order to increase the system's isotropy.

For anyone who would like to reproduce our results we gave all the technical details about exact model parameters and approaches to data sampling.

This thesis is just the first step in studying the possibilities to control material properties by designing particles' shape by means of automatic differentiation. Ahead we have a lot of work justifying obtained results and developing the ideas presented here. In previous section we already mentioned a lot of ways to extend the exploration horizons. From the perspective of this thesis the most relevant and insightful directions are the implementation of the gradient descent with mean gradients and the range extension for parameters like system size, density and variation of spheres' placement within a particle.

References

[1] M. Agrawal and S. C. Glotzer. Scale-free, programmable design of morphable chain loops of kilobots and colloidal motors. *Proceedings of the National Academy of Sciences*, 117(16):8700–8710, 2020.

[2] G. Anderson, M. Aftosmis, and M. Nemec. Parametric deformation of discrete geometry for aerodynamic shape design. *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2012.

[3] B. Andreotti, Y. Forterre, and O. Pouliquen. Granular media. 2013.

[4] V. Babu, D. Pan, Y. Jin, B. Chakraborty, and S. Sastry. Dilatancy, shear jamming, and a generalized jamming phase diagram of frictionless sphere packings. *Soft Matter*, 17(11):3121–3127, 2021.

[5] R. P. Behringer. Jamming in granular materials. *Comptes Rendus Physique*, 16(1):10–25, 2015. Granular physics / Physique des milieux granulaires.

[6] E. Bitzek, P. Koskinen, F. Gähler, M. Moseler, and P. Gumbsch. Structural relaxation made simple. *Physical Review Letters*, 97(17), 2006.

[7] S. Calderon and T. Boubekeur. Bounding proxies for shape approximation. *ACM Transactions on Graphics*, 36(4):1–13, 2017.

[8] K.-H. Chang and K. K. Choi. A geometry-based parameterization method for shape design of elastic solids*. *Mechanics of Structures and Machines*, 20(2):215–252, 1992.

[9] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. *ACM SIGGRAPH 2004 Papers on - SIGGRAPH 04*, 2004.

[10] P. F. Damasceno, M. Engel, and S. C. Glotzer. Predictive self-assembly of polyhedra into complex structures. *Science*, 337(6093):453–457, 2012.

[11] C. X. Du, G. V. Anders, J. Dshemuchadse, P. M. Dodd, and S. C. Glotzer. Inverse design of compression-induced solid – solid transitions in colloids. *Molecular Simulation*, 46(14):1037–1044, 2020.

[12] K. C. Elbert, W. Zygmunt, T. Vo, C. M. Vara, D. J. Rosen, N. M. Krook, S. C. Glotzer, and C. B. Murray. Anisotropic nanocrystal shape and ligand design for co-assembly. *Science Advances*, 7(23), 2021.

[13] B. Florijn, C. Coulais, and M. V. Hecke. Programmable mechanical metamaterials: the role of geometry. *Soft Matter*, 12(42):8736–8743, 2016.

[14] C. P. Goodrich. Unearthing the anticrystal: criticality in the linear response of disordered solids, 2015.

[15] C. P. Goodrich, S. Dagois-Bohy, B. P. Tighe, M. van Hecke, A. J. Liu, and S. R. Nagel. Jamming in finite systems: Stability,

anisotropy, fluctuations, and scaling. *Physical Review E*, 90(2), Aug 2014.

[16] C. P. Goodrich, E. M. King, S. S. Schoenholz, E. D. Cubuk, and M. P. Brenner. Designing self-assembling kinetics with differentiable statistical physics models. *Proceedings of the National Academy of Sciences*, 118(10), 2021.

[17] C. P. Goodrich, A. J. Liu, and S. R. Nagel. Finite-size scaling at the jamming transition. *Phys. Rev. Lett.*, 109:095704, Aug 2012.

[18] C. P. Goodrich, A. J. Liu, and J. P. Sethna. Scaling ansatz for the jamming transition. *Proceedings of the National Academy of Sciences*, 113(35):9745–9750, 2016.

[19] J. V. D. Gucht. Grand challenges in soft matter physics. *Frontiers in Physics*, 6, 2018.

[20] P. Guo and X. Su. Shear strength, interparticle locking, and dilatancy of granular materials. *Canadian Geotechnical Journal*, 44(5):579–591, 2007.

[21] A. Hafez, Q. Liu, T. Finkbeiner, R. A. Alouhali, T. E. Moellendick, and J. C. Santamarina. The effect of particle shape on discharge and clogging. *Scientific Reports*, 11(1), 2021.

[22] M. V. Hecke. Jamming of soft particles: geometry, mechanics, scaling and isostaticity. *Journal of Physics: Condensed Matter*, 22(3):033101, 2009.

[23] P. H. W. Hoffmann. A hitchhiker's guide to automatic differentiation. *Numerical Algorithms*, 72(3):775–811, 2015.

[24] S. Iai, T. Tobita, O. Ozutsumi, and K. Ueda. Dilatancy of granular materials in a strain space multiple mechanism model. *International Journal for Numerical and Analytical Methods in Geomechanics*, 35(3):360–392, 2011.

[25] A. Ion, M. Rabinovich, P. Herholz, and O. Sorkine-Hornung. Shape approximation by developable wrapping. *ACM Transactions on Graphics*, 39(6):1–12, 2020.

[26] H. M. Jaeger. Celebrating soft matter's 10th anniversary: Toward jamming by design. *Soft Matter*, 11(1):12–27, 2015.

[27] A. Jain, J. R. Errington, and T. M. Truskett. Dimensionality and design of isotropic interactions that stabilize honeycomb, square, simple cubic, and diamond lattices. *Physical Review X*, 4(3), 2014.

[28] N. Kruyt and L. Rothenburg. A micromechanical study of dilatancy of granular materials. *Journal of the Mechanics and Physics of Solids*, 95:411–427, 2016.

[29] A. J. Liu and S. R. Nagel. Jamming is not just cool any more. *Nature*, 396(6706):21–22, 1998.

[30] I. Marinić-Kragić, D. Vučina, and M. Ćurković. Efficient shape parameterization method for multidisciplinary global optimization and application to integrated ship hull shape optimization workflow. *Computer-Aided Design*, 80:61–75, 2016.

[31] R. L. Marson, E. G. Teich, J. Dshemuchadse, S. C. Glotzer, and R. G. Larson. Computational self-assembly of colloidal

crystals from platonic polyhedral sphere clusters. *Soft Matter*, 15(31):6288–6299, 2019.

[32] S. M. Moosavi, H. Xu, L. Chen, A. Cooper, and B. Smit. Geometric landscapes for material discovery within energy-structure-function maps. 2020.

[33] S. R. Nagel. Experimental soft-matter science. *Reviews of Modern Physics*, 89(2), 2017.

[34] A. N. B. Nair, S. Pirker, T. Umundum, and M. Saeedipour. A reduced-order model for deformable particles with application in bio-microfluidics. *Computational Particle Mechanics*, 7(3):593–601, 2019.

[35] U. Naumann. *The Art of Differentiating Computer Programs: An Introduction to Algorithmic Differentiation.* Number 24 in Software, Environments, and Tools. SIAM, Philadelphia, PA, 2012.

[36] C. S. O'Hern, L. E. Silbert, A. J. Liu, and S. R. Nagel. Jamming at zero temperature and zero applied stress: The epitome of disorder. *Physical Review E*, 68(1), 2003.

[37] V. Ranjan and A. Fournier. Volume models for volumetric data. *Computer*, 27(7):28–36, 1994.

[38] P. S. Salmon. Order within disorder. *Nature Materials*, 1(2):87–88, 2002.

[39] S. S. Schoenholz and E. D. Cubuk. Jax, m.d.: A framework for differentiable physics, 2020.

[40] O. Sorkine, D. Cohen-Or, D. Irony, and S. Toledo. Geometry-aware bases for shape approximation. *IEEE Transactions on Visualization and Computer Graphics*, 11(2):171–180, 2005.

[41] S. Stolpner, P. Kry, and K. Siddiqi. Medial spheres for shape approximation. *Advances in Intelligent and Soft Computing Brain, Body and Machine*, page 137–148, 2010.