

We thank the referee 2 for their positive comments and thoughtful suggestions below. Each point has been addressed individually, with changes to the manuscript text shown in **orange text**. Unchanged manuscript associated with these changes is shown in **blue text**.

#### Comments from Referee 2

*This paper describes a coarse-graining software, Pysammos, dedicated to the spatial averaging of discrete element modelling data, in particular the one produces by the software MFIX-DEM. The principle of the method and software are explained, as well as its specificities, and its performance are compared with other coarse-graining codes and with different geometries and simulation conditions.*

*The paper is overall well written and detailed, and can be useful as an entry point to coarse-graining, while proposing additional features not often present in already existing solutions. Some of the specific aspects, such as the phase identification and the validation with non-spherical particles, can be useful in a broad range of contexts. I only have some minor comments and questions below.*

- *Comment 1: In the simulations of particles suspended in a magma, it is not clear how the flow is created, and if it is a one-way or two-ways interaction between the fluid and particles. It would be interesting as well to provide the volume fraction field, which may have some spatial variations. Also, the simulations are indicated as 2D, but the particles seem to be three-dimensional?*

Response: Thank you for raising this point. Indeed, further information about the simulation set up would be beneficial for this showcase example. Therefore, we have addressed these details in the results section where the example is introduced:

L533: Hence, we present a DEM-CFD pseudo-2D (i.e., with a thickness of one cell) numerical simulation representing non-spherical crystals suspended in magma flowing cyclically in the positive vertical direction through a narrow conduit with frictional walls (Fig. 12.a). The particles are fully three-dimensional and interact with the fluid through a two-way coupling via drag and pressure forces. The flow is driven by a constant flux, causing the particles to accelerate until they eventually reach steady state. Given the sparsity of the solid particles, we chose a smoothing function half-width four times larger than the default option,  $0.75 \times d_{43}$ .

We agree that we should provide the volume fraction field, since we claim that the average is below 0.1 in line 517. Hence, we have added it to the Supplementary Material, as Figure S8 and cross-referenced it in the text (line ??).

L546: Therefore, for the low concentration of suspended particles ( $\phi \ll 0.1$ ) used in this example (see Fig. S9 of the Supplementary Material), very low effective shear viscosity is to be expected.

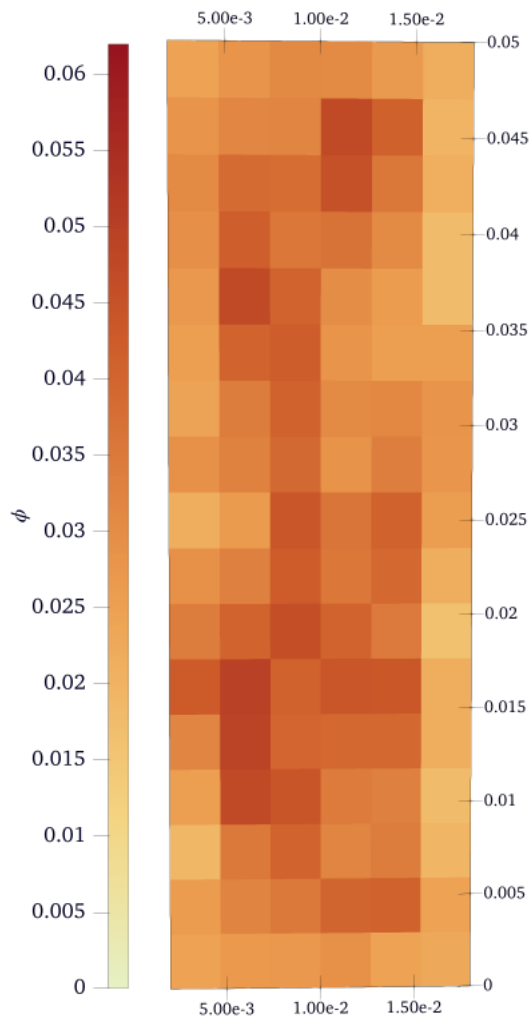


Figure S9: Visualisation of the volume fraction field that corresponds to a DEM-CFD pseudo-2D numerical simulation representing non-spherical crystals suspended in magma.

- Comment 2: *The results on the stress under the pile (4.4.4) are quite surprising. Experimental observations show essentially hydrostatic pressure, sometimes with a dip of pressure below the centre, but both appear quite different from the results reported. Even if frictional redistribution of the stress occurs (similar to the Janssen effect), the weight of the grains still needs to be supported somehow. It would be good to also look at the shear stresses to assess whether this frictional redistribution does occur. Also, Fig 13a may be better plotted with a linear scale, since again the expectation would be a linear stress increase with depth on average.*

Response: Thank you for raising these points, which we address in turn. We agree that a linear scale of Fig. 13.a (now 13.b) enables a more accurate visualisation of the stress distribution across the pile, and hence, we have modified it. Further, to

improve clarity, we have plotted the particle position on a different plot (now Fig. 13.a). We agree that this example would benefit from a more robust and clear display of stress distribution and redistribution. Hence, we have added the map of the total stress tensor magnitude (Fig. 13.b), the map of the shear stress (Fig. 13.c) and the horizontal profile of basal vertical stress (Fig. 13.d). We hope that the combination of these three quantities yield a more diagnostic stress redistribution. The total stress tensor magnitude shows the force chain arching, whereas the basal vertical stress shows a distinct dip beneath the apex of the pile. Notably, peak shear stress occurs not immediately at the walls but within the lower-mid region of the pile, consistent with force chains curving from the crest through the granular bulk before reaching the walls, where the resulting lateral thrust is partially supported, in a Janssen-like manner. These points have been added to the corresponding section of the manuscript:

L549: This example highlights the package's ability to (1) analyse and visualise Pysammos results entirely in Python, (2) resolve individual phases within a polydisperse mixture, and (3) straightforwardly assess each phase's contribution to the bulk properties of the mixture. DEM can also be used to study vertical load redistribution in loose sediment — for instance, in debris-flow deposits posing a risk to nearby communities and infrastructure. Here, we present a 2D DEM simulation of a growing granular pile, in which grains of different sizes are poured from above into a confined space between two walls 50 cm apart (Fig. 13a). This model captures two combined effects: force-chain arching in a free-surface granular pile (Brockbank et al., 1997), and the Janssen effect, whereby vertical stress saturates with depth as it is redistributed laterally through friction at the side walls (Windows-Yule et al., 2019).

L557: Through this example we see that, near the peak of the pile, basal pressure exhibits a local minimum, rather than the maximum that a hydrostatic gradient of a fluid would predict directly beneath the peak (Fig. 13.d) — a phenomenon that has been documented experimentally (Brockbank et al., 1997). This occurs because granular assemblies transmit stress anisotropically along chains of contacting grains, rather than isotropically as in fluids, redirecting stress sideways by "arching" it toward the flanks of the pile rather than accumulating it directly underneath. These arching structures are visible as high-magnitude bands in the total stress tensor magnitude field (Fig. 13b). This effect is even more pronounced in this 2D configuration, compared to a 3D pile.

The mechanism behind this stress redistribution is more explicitly shown by the shear stress field (Fig. 13.c). Unlike the stress magnitude, which illustrates where stress locally intensifies, the sign of the shear stress specifies the direction in which load is being carried: shear stress is close to zero along the vertical centreline and grows with opposite sign toward each flank. This confirms that weight is indeed

transported away from the pile's apex in both lateral directions, rather than straight downward. The largest shear stress magnitudes do not occur at the side walls, but midway between the peak and the walls. This is consistent with force chain curving outward and downward through the pile before reaching the walls. This suggests force chain arching is probably the dominant redistribution mechanism here, with the walls providing at least a partial frictional contribution to support, consistent with a weakly-developed Janssen effect. This is visible in Fig. 13e, where in the wall-confined region (heights of 0.005–0.03 m), horizontal stress increases with depth while vertical stress remains constant — indicating that the walls are supporting part of the pile's weight. We note that this apparent depth-independence of vertical stress is not expected to persist as the pile height grows, given that the wall-supported fraction of the total weight would saturate while the unsupported fraction below saturation depth would grow with height approximately following a hydrostatic profile. Finally, we find that the different phases identified by grain diameter contribute differently to the bulk stress at different heights, although the relatively low polydispersity of this mixture means all phases contribute roughly equally to the bulk field along the profile.

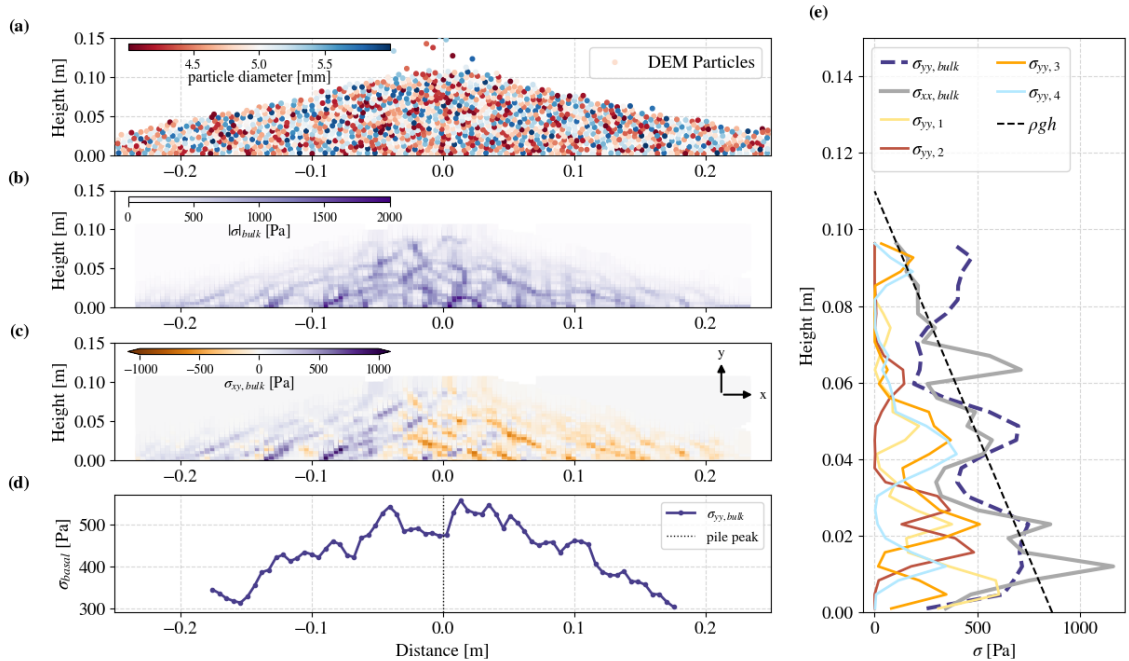


Figure 13: Analysis of a 2D DEM simulation of a growing granular pile carried out with Pysammos's Python-supported visualisation. Simulation particles coloured by size (a), coarse-grained magnitude of the total stress tensor (b) and shear stress (c) across the granular mixture. Coarse-grained basal vertical stresses, taken from the lowest row of CG cells, as a function of horizontal distance, testing for stress arching beneath the pile crest (dotted black line,  $x=0$ ) (d). Vertical profiles, taken at the pile centre ( $x=0$ ) in (d), of the horizontal stress of the mixture (grey solid line), vertical

stress of the mixture (dashed blue line), and vertical stress of each of the phases with diameters 5.7mm, 4.7mm, 5.1mm, and 4.2mm, in yellow, red, orange and blue, respectively (e).

- Comment 3: *The calculation of the contact stress involves integrating the windowing function over the branch vector, as shown in equation 5. Is this approximated through averaging the values at the centre, as is commonly done, or is the integral numerically solved? How? This may explain some of the differences pointed out in  $\sigma_c$  later on in the paper.*

Response: Thank you for raising this point, which was indeed underspecified in the manuscript. The line integral in Eq. 5 is not approximated by evaluating the windowing function at the branch-vector midpoint. Instead, Pysammos solves it via trapezoidal numerical integration, sampling the weight function at multiple points along the branch vector between the particle centre and the contact point, and integrating those samples numerically. We have added this explanation to Section 3.5.3:

The weights are pre-calculated for a range of particle-grid point distances, [...] lookup table algorithm, details of which can be found in Appendix C.

L355: The calculation of contact-related fields (e.g., the contact stress tensor, Eq. (5)) requires the additional line integral of  $\Psi$  along the branch vector connecting a particle centre to a contact point. This integral could, in principle, be solved in analytically for each kernel individually (Weinhart et al., 2012). In Pysammos, we instead evaluate it numerically: the branch vector is discretised into  $N=10$  points, sampled through the same lookup table employed for particle weights, and integrating with the trapezoidal rule. This provides a single, kernel-independent implementation of the integral. Increasing  $N$  beyond 10, up to  $10^5$ , produced no discernible change in the resulting contact stress fields (Fig. S7 of the Supplementary Material), confirming that the trapezoidal integration is already converged at this resolution.

We investigated the effect that the number of trapezoidal integration bins,  $N$ , has on the contact stress results. To assess this, we carried out a test that compared results yielded by  $N=10$  and  $N=10^5$ , at low  $w/d$ , the range that shows larger discrepancy with MercuryCG. This was added to the Supplementary Material and referenced in the Section 3.4.3, to ensure the reader that the choice of  $N$  does not significantly compromise the results.

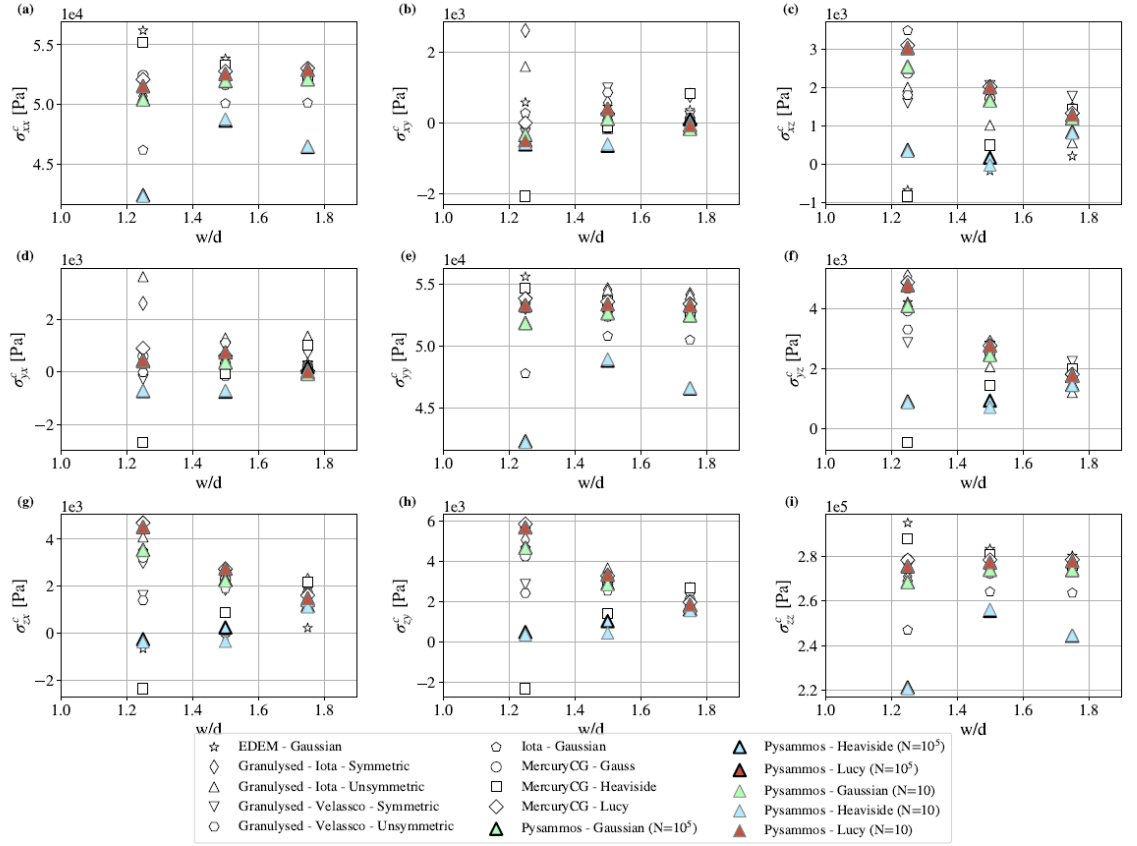


Figure S7: Benchmark of the contact stress for a monodisperse cubic lattice of grains with a diameter of 0.02m and a density of 2500kgm<sup>-3</sup> (g), evaluated at the point  $P = (0, 0, 0.42725)$  m. The stress was coarse-grained using Pysammos with trapezoidal integration and 10 and 105 bins (coloured triangles with black and grey edges, respectively), and using other coarse-graining software (open symbols): EDEM, Granulysed, Iota, and MercuryCG. Where available, results obtained using different smoothing functions (Gaussian, Heaviside, and Lucy) are compared over a range of smoothing half-widths, expressed as multiples of the characteristic particle size ( $w/d$ ). Only values of  $w/d < 2$  are shown, as this is the range where the largest discrepancies between models occur. Panels (a)–(i) show the components of  $\sigma_{ij}$  arranged in matrix form, with rows corresponding to  $i = x, y, z$  and columns to  $j = x, y, z$ .

The contact stress discrepancies we report between Pysammos and MercuryCG at low  $w/d$  (Section 4.1) are therefore likely to reflect the particle-search volume being centred on particle  $i$  rather than on the contact point, which we already identify as a contributing factor. We have honed the Results section, by providing a more precise reason for the discrepancy in the  $\sigma_c$  field at low  $w/d$  values.

L405: The contact tensor field (Fig. 7.d) shows a broadly constant mean value across the analysed  $w/d$  range (see Fig. S6.e for enlarged view), with a greater scatter at the low end. Pysammos and MercuryCG show a degree of discrepancy at lower  $w/d$ . This could be attributed to the difference in the particle-search

criterion between the two codes: in Pysammos, a contact is included in the averaging volume based on the distance from the grid point to the centre of *particle i*, whereas MercuryCG's determines inclusion based on the minimum distance from the grid point to the branch vector itself (Weinhart et al., 2020). This more general criterion allows MercuryCG to include contacts whose branch vector is positioned near the grid point even when the particle centre itself lies outside the cutoff radius, an effect expected to be more pronounced at low  $w/d$ , where the search radius is comparable to the branch-vector length. We note that this discrepancy is unlikely to stem from the numerical evaluation of the branch-vector line integral in Pysammos (Sect. 3.5.3), as the trapezoidal integration is already converged at  $N=10$  sampling points (Fig. S7 of the Supplementary Material) and is not expected to be a significant source of error even at low  $w/d$ . Further, the Heaviside function in Pysammos shows larger disagreement with that of MercuryCG across the whole  $w/d$  range and especially in the diagonal components of [...]

- Comment 4: *There are many different clustering algorithms that exist, how was the particular one used chosen?*

Response: We agree that specifying the clustering algorithm employed in the phase detection is very relevant. In line 278, we already mention that “In Pysammos the phase detection is carried out by clustering particle data using the k-means algorithm, explained in detail in Appendix B1”, and subsequently explain the k-means algorithm in depth in Appendix B1.

- Comment 5: *line 144: the mathematical detail of the Dirac delta function is inaccurate. Technically, the Dirac function is infinity at the particle centre, and 0 otherwise, such as its integral over the whole space is 1.*

Response: Thank you for pointing this out. We agree that the mathematical detail of the Dirac delta function is inaccurate. Therefore, we have modified this sentence to:

L141: Note that  $\delta$  is zero everywhere except at the particle centres, where it is infinitely high, and has a total integral of one.

- Comment 6: *Equations 5 and C4: the tensor product, or Kronecker product that is used for the vector in those equations, should be specified. It is usually denoted with  $\otimes$ .*

Response: Thank you for bringing this up — the notation was indeed ambiguous as written. Because we use component-free (bold tensor) notation throughout, the juxtaposition of two vectors does not carry a single unambiguous meaning the

way it does in index notation, where two free indices (e.g.  $f_p a_q$ ) automatically identify a rank-2 tensor. In bold notation, a reader could plausibly interpret the juxtaposition as an inner product, which would incorrectly yield a scalar rather than the rank-2 tensor we intend. To remove this ambiguity, we have introduced the  $\otimes$  symbol to denote the dyadic (outer) product explicitly. Equations 5 and C4 have been updated accordingly, and we have checked the rest of the manuscript for consistency with this notation:

$$\boldsymbol{\sigma}^c(\mathbf{r}, t) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^{C_{ij}} \mathbf{f}_{ij}^k \otimes \mathbf{a}_{ij}^k \int_0^1 \Psi(\mathbf{r} - \mathbf{r}_i + s \mathbf{a}_{ij}^k) ds$$

$$\mathbf{F}(\mathbf{r}, t) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^{C_{ij}} \nu_i \mathbf{n}_{ij}^k \otimes \mathbf{n}_{ij}^k \int_0^1 \Psi(\mathbf{r} - \mathbf{r}_i + s \mathbf{a}_{ij}^k) ds$$

- Comment 7: *line 193: the use of only equilibrium states to justify the replacement of ensemble averaging with time averaging appears quite restrictive. Most granular systems are out of equilibrium, and time averaging can still be meaningfully performed if the systems are at steady state.*

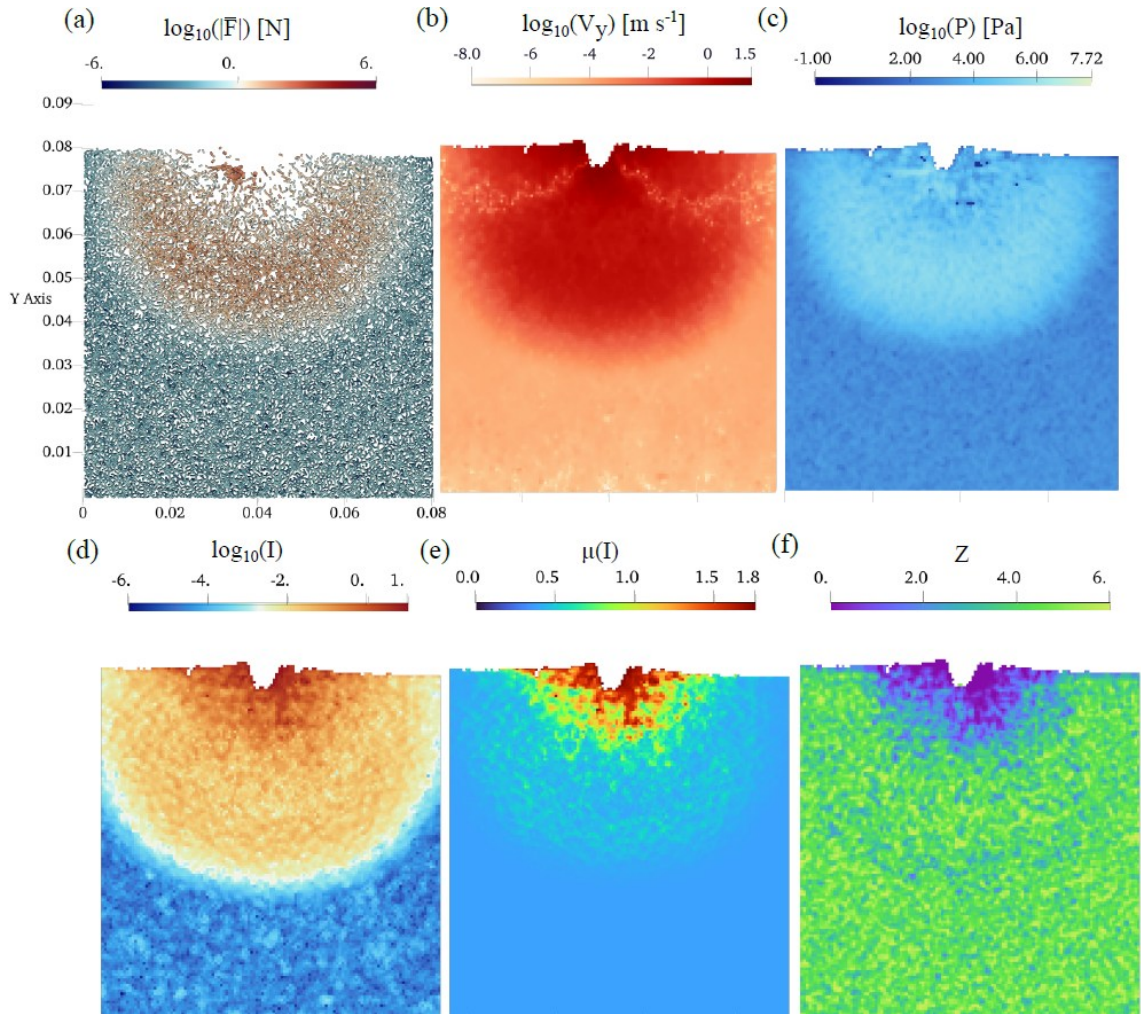
Response: Thank you for this comment. We agree that framing the ensemble-time averaging equivalence purely in terms of equilibrium states is too restrictive, since granular flows are inherently dissipative. We have revised the text to clarify that the relevant condition for this equivalence is statistical stationarity (steady state) rather than equilibrium in the strict thermodynamic sense. We have added a citation to Babic (1997) in support of this point. The manuscript has been modified accordingly, as shown below:

L189: Traditional statistical mechanics considers multiple realisations, or probabilistic copies of a system, over which the corresponding macroscopic quantities are averaged (Jaynes, 1957). This conceptual tool is equivalent to averaging over the total time range, an equivalence classically derived for ergodic systems in equilibrium (Huang, 1987), but which extends to systems at (non-equilibrium) steady state, where the relevant statistics are stationary in time (Babic, 1997). This is advantageous for systems that are complex to simulate over large time scales. In the case of granular media we are able to simulate complex granular systems for longer time scales with DEM. Hence, in the field of computational granular physics, ensemble averaging is not required (Glasser, 2001), and in fact could be problematic due to the inherent lack of scale separation. Instead, the analytical expressions for continuum fields tend to include a time-averaging, as well as a space-averaging, which acts as a low-pass filter on the inherent fluctuations of granular flows, to obtain steady-state solutions (Babic, 1997; Glasser, 2001). Nevertheless, the time averaging is not strictly required, and in fact, even moving window Heaviside temporal averages

could obscure time-dependent trends in highly transient flows, as pointed out by Weinhart et al. (2012). Hence, in this study only spatial averaging is considered for the analytical equations of coarse-graining.

- Comment 8: *Fig. 11b: the scale shows velocities going up to  $10^{30}$  m/s, which is much higher than the speed of light. There is probably an issue with the scale.*

Response: Thank you for spotting this, as it was clearly wrong. We corrected the colour scale range in ParaView and now it presents values that are more reasonable, with an upper bound of  $\log_{10}(V_y)=1.5$ .

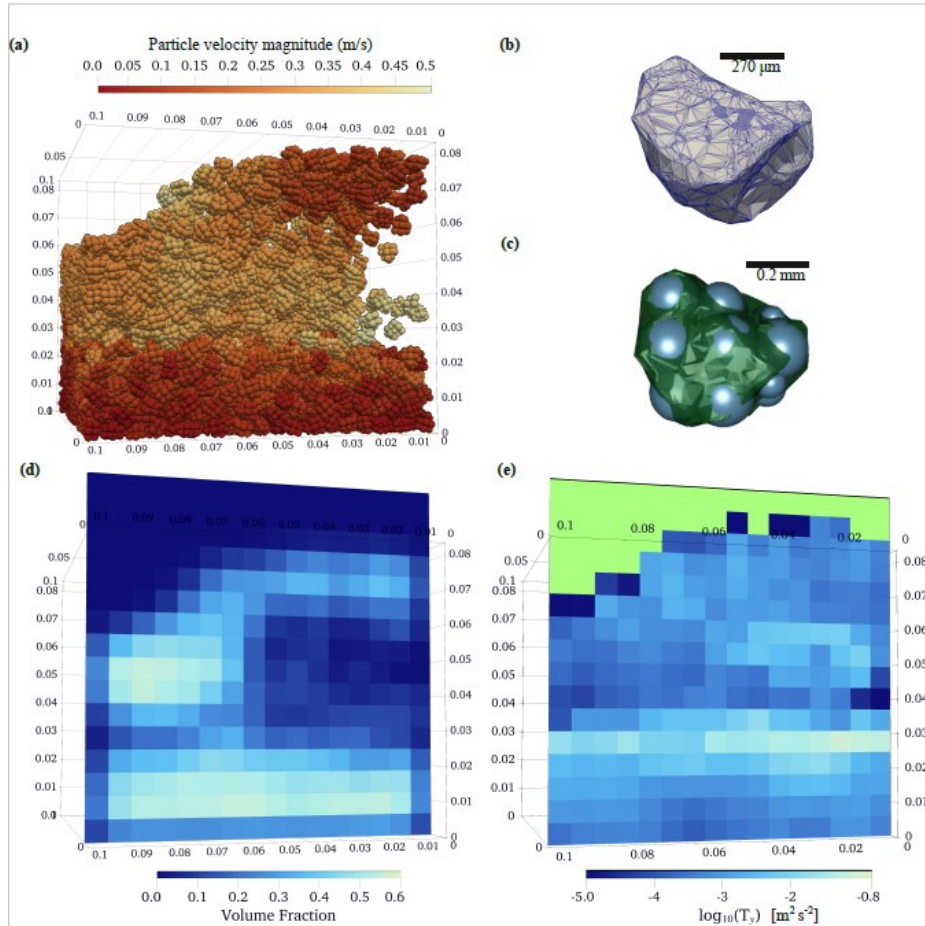


- Comment 9: *Fig 15 caption: the explanation of the "greyish" colour is incorrect.  $T_y$  is not zero at these locations, but undefined, since the volume fraction is zero (no particles). As indicated by equation C13, this would correspond to dividing a zero kinetic stress tensor trace by the zero density at that location, which numerically should result in a Not a Number.*

Response: Thank you for bringing this up, as this was unclear. Indeed, in the regions where the mixture density is zero,  $T$  should be Not a Number (NaN), owing to the indeterminate division of zero by zero. Nevertheless, we force  $T$  to zero when the denominator in Eq. C13 (i.e., the mixture density) is zero. This is a developer choice and it was done with numerical safety in mind, as downstream code may expect finite numbers — for example, the calculator filter in ParaView. Thus, this safety-over-accuracy trade-off avoids NaNs propagating through arrays, plots, and filters. However, we agree that this needs to be explained in the text. We have therefore added an explanation of this design choice to the Appendix section where granular temperature is introduced:

L927: Granular temperature,  $T$ , measures the level of fluctuation [...] It is important to note that in regions with no particle influence (i.e.,  $\rho(r, t) = 0$ ),  $T$  is set to zero, to avoid the undefined division of 0 by 0, and subsequent NaN propagation. Although a NaN would be more mathematically accurate, we have opted for this common numerical-safety practice.

However, it is true that  $\log_{10}(0)$  is also NaN, and therefore it should be made clear in Fig. 15. Previously the NaN values were substituted by a 0 in ParaView, which was very misleading. We have modified Figure 15 to show NaN values in a clearly different colour (pale green).



We have accordingly tweaked the caption of Figure 15 explain that  $T = 0$  is forced, as well as indicating that  $\log_{10}$  of  $T = 0$  are coloured pale green, because they are NaN.

L717: Figure 15: DEM simulation of a granular bed, fluidised with gas [...] Coarse-grained fields of the volume fraction (d) and vertical granular temperature (e) corresponding to the simulation time step shown in (a). Note that the pale green colour in (e) indicates NaN. This occurs at locations where  $T_y$  is set to  $0 \text{ m}^2\text{s}^{-2}$  by convention (see Appendix C). Consequently,  $\log_{10}(0)$  is undefined and is represented as NaN in ParaView.

- Comment 10: line 885: it would be good to define  $D$  right after eq. C13, rather than on line 895.

Response: Thank you for this suggestion. We agree that the definition of  $D$  is more appropriate after the equation that introduces granular temperature  $T$  (C13), rather than after the deviatoric stress equation (C15), later on in the text. Hence, we have moved the definition of  $D$  closer to Eq. C13:

L921: Granular temperature,  $T$ , measures the level of fluctuation of particle velocity about the macroscopic velocity of the granular mixture (Goldhirsch, 2008). It is calculated as a spatial average of the velocity fluctuations squared, since by definition the average of fluctuating velocity (not squared) is zero and hence avoid the cancellation of fluctuating velocities with the same magnitude but opposite sign,

$$T(\mathbf{r}, t) := \frac{1}{D\rho(\mathbf{r}, t)} \text{tr}(\boldsymbol{\sigma}^{k'}(\mathbf{r}, t)),$$

where  $D$  is the number of spatial dimensions considered. Note that the scale-independent term of the kinetic tensor must be used ( $\hat{\sigma}_k$ ), as  $T$  focuses on the local velocity fluctuations,  $V_i$  (Eq. (6) and 7).

- Comment 11: *eq. C19: the equation uses  $\bar{d}$ , which is not defined. Is it the same as  $D_{43}$  (eq. C6)?*

Response: Thank you for this comment. The representative particle size  $\bar{d}$  is defined in section 3.4. Mesh generation:

L296: The mesh is created with a resolution  $w$  (Eq. (9)) specified in `.set_resolution()`. It is calculated as  $w = m_w \bar{d}$ , where  $m_w$  is a scaling and  $\bar{d}$  is a representative particle size. By default,  $m_w$  is 0.75 (Weinhart et al., 2016). In polydisperse granular media the choice of  $\bar{d}$  is not trivial (Polanía et al., 2025), hence, we provide different metrics for the user to choose from, such as the median diameter or  $d_{50}$ , the root mean squared diameter ( $d_{\text{rms}}$ ), volume-weighted average ( $d_{43}$ ) and area-weighted average or Sauter diameter ( $d_{32}$ ).

The reference to Appendix C regarding the rest of continuum fields is made in section 3.5.4 Macroscopic fields. However, we have added a sentence after Eq. C19, redirecting the reader to section 3.4:

L952: where  $\bar{d}$  is the representative particle size (Sect. 3.4). The inertial number is a measure of the dynamic state of a granular medium. It is also understood as the ratio of (1) the microscopic timescale (or scale of rearrangement), representing the time a particle takes to fall in an a slit of a given diameter under a given pressure; and (2) the macroscopic time scale, linked to the average deformation (Midi, 2004).