Supplementary Information

March 4, 2024

1 Experimental setup

1.1 Mechanics

The mechanical experimental setup used for this research (Fig. S1) is based on a circular horizontal PPMI channel of outer and inner radii $R = 19.2$ cm and $r = 12.5$ cm respectively, and 12 cm height. A bidisperse mixture of 50 % - 50 % glass beads, of radii $r_1 = 1.5\,\mathrm{mm} \pm 10\,\%$ and $r_2 = 2\,\mathrm{mm} \pm 10\,\%$, fills the channel to about 9 cm in height. A horizontal plate fitting the channel can shear the granular medium from the top. It is free to move vertically and can be rotated. A layer of grains is glued to its lower face in order to better drag the underlying granular medium. It does not extend to the whole plate surface, but excludes an annulus close to the border, to prevent free grains from getting stuck between glued grains and container walls (Fig. S1). The plate has mass $M = 1200\,\mathrm{g}$ and moment of inertia $I = 0.026\,\mathrm{kg\,m^2}$, implying that in our experiments the medium is under a nominal pressure $p = Mg/[\pi(R^2 - r^2)] \simeq 176\,\mathrm{Pa}$. The plate can be rotated through a stepping motor, at constant angular speed ω_p in the interval of 0.05 rad/s to 1.5 rad/s.

1.2 Optics

The optical setup is also sketched in Fig. S1. A Nd:YAG laser supplies an elliptically polarized light beam at a wavelength of 532 nm, with a maximum nominal power of 500 mW. The beam is filtered by a polarizing filter and is expanded by a lens with $f = 50$ mm, hitting the channel wall horizontally. The back-scattered light is then focused towards a perpendicular white screen by another $f = 50$ mm lens. To minimize the amount of light reflected by the perspex walls and by the first vertical layer of grains, the beam eventually passes through another polarizer, crossed with the first.

The resulting interference figures have been recorded with a D90 Nikon camera equipped with a CMOS, of 640×424 pixels in the horizontal and vertical direction respectively, and a dynamics of 8 bits. A lens with $f = 50$ mm has been employed, with an acquisition rate of 15 frames/s. Due to the channel curvature and the laser spot dimension, the useful area of the interference pattern on the detector had a size of 250×420 pixels. Calibration measures have shown that

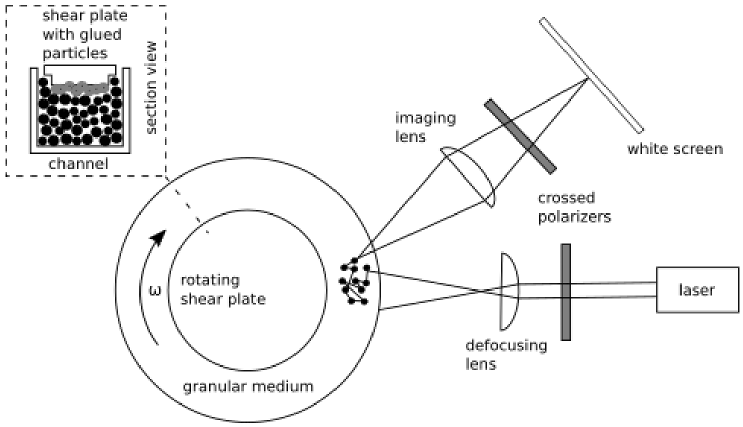

Figure S1: a) Sketch of the experimental set-up with a detail of the top plate.

one pixel corresponds to a region of the size of 0.04 mm on the container wall, so the analyzed region of the granular system has an external surface of 10×16.8 mm^2. The grains diameter is, on average, of about 1.75 mm, corresponding to 44 pixels. The main typical linear sizes characterizing our system are resumed in Tab. S1.

element	mm	pixel
pixel	0.04	1
grain	1.5 - 2.0	38 - 50
interference figure (width $\times$ height)	10.0×16.8	250×420
metapixel	0.4	10

Table S1: Some sizes characterizing the system

2 Preprocessing of video

Scaling of pixel intensities. A video of duration 11 min has been recorded for different speeds of the rotating plate. Each video is encoded as a three-dimensional matrix whose indexes span the two spatial dimensions and the time. This way, the time evolution of the interference pattern is contained in a three-dimensional array of dimensions (250, 420, 9900) whose entries represent the back-scattered light intensity recorded at a pixel (x, z) at a time t. Because the beam intensity is not uniform over the channel wall, one could normalize data with the average intensity of each pixel. However, the presence of very bright spots in the interference pattern makes this kind of normalization ineffective. Thus we proceeded by taking a reference value at each pixel which has been computed in the following way. First, we averaged all the intensities over time obtaining a two-dimensional array (250, 420). Then, we performed averages of this array over rows and columns obtaining two one-dimensional arrays. Finally, the normalization matrix was obtained through the external product of such vectors and taking the square root of each entry (see Fig. S2A).

Spatial coarse-graining and choice of the metapixel. The intensity data collected at every pixel have been averaged in space over squares of side $S = 10$ pixels, the metapixel (MP). To determine the optimal size, a set of 5 min long data has been collected with the plate at rest and the following steps followed:

i) Coarse-graining was performed on each frame by averaging intensities over the pixels of a square block of size S. At the beginning S =1;

ii) for each block, the variance σ^2 over all the frames of the video was computed;

iii) The computed variances were averaged over the blocks of the entire interference image, yielding a value $\langle\sigma^2(S)\rangle$ for each choice of S

iv) the block size S was increased by one and the procedure repeated.

Notice that in the absence of noise on the detector, each variance should be strictly zero since we are considering the case with the plate at rest. Thus, the average variance $\langle\sigma^2(S)\rangle$ indicates the amount of background noise; being uncorrelated, this noise is expected to decrease for increasing sizes of the metapixel S. The results obtained in our setup are shown in Fig. S2B). Panel B shows that the standard deviation decreases as S^{-1} as expected. However, the noise reduction obtained in this way is paid for by a loss of spatial resolution. Indeed, once we construct an interference image with metapixels of size S we cannot resolve anymore any spatial modulation of the intensity over scales $< S$. This is a crucial point since we are interested in the characterization of the shear profile. The criterion we used to stop the coarse-graining procedure is to choose an optimal S which is smaller than the typical spatial correlation length of the intensity pattern in the video with the plate at rest. In this way, we minimize the resolution loss since each metapixel contains on average only correlated pixels which bring the same local information. This procedure allows us to only filter the noise around local intensity values without losing information on spatial intensity modulations. In Fig. S2C we show the spatial correlation functions of the intensity pattern for $\omega = 0\ s^{-1}$. We compared the ones obtained considering distances over both x and y, computed on single frames and then averaged over 100 frames. From this plot, it is seen that a reasonable choice for the metapixel size is $S = 10$ since intensities within this distance are correlated in both directions.

3 Autocorrelation function

For each value of the plate rotational speed, we computed the connected time autocorrelation function for the metapixel centred in (x, z) (horizontal and ver-

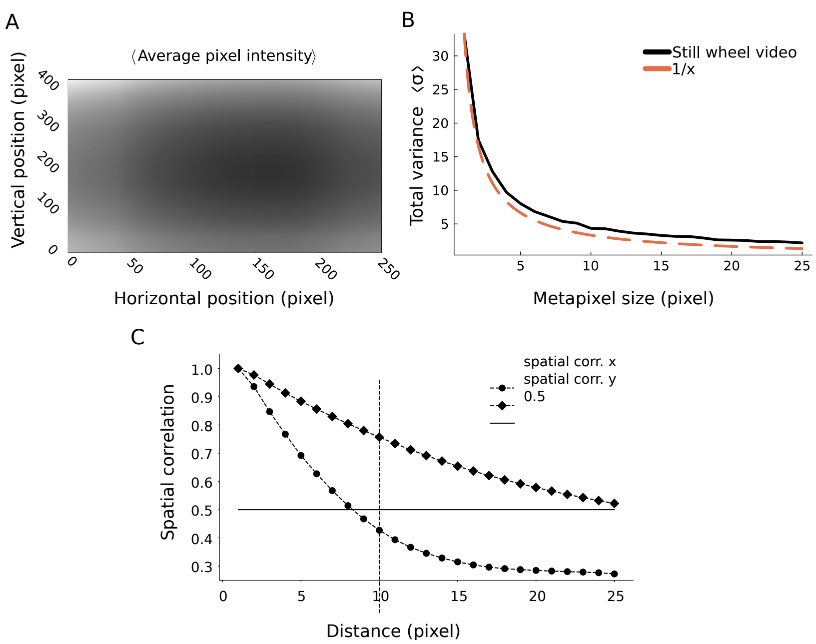

Figure S2: A) Normalization matrix obtained from the external product of intensities averaged over rows and columns (see the text for the specific procedure). B) Total time variance for $\omega = 0$, as a function of the metapixel size S; C) Spatial correlation functions of the intensity pattern for $\omega = 0$ on the x and y directions.

tical coordinates):

$$g(x,z,t) = \frac{\langle I(x,z,t_0)I(x,z,t_0+t)\rangle_{t_0} - \langle I(x,z,t_0)\rangle^2_{t_0}}{\langle I^2(x,z,t_0)\rangle_{t_0} - \langle I(x,z,t_0)\rangle^2_{t_0}} \tag{1}$$

averaged over discrete initial times $t_m = m\Delta t$ as

$$\langle I(x,z,t_0)I(x,z,t_0+t)\rangle_{t_0} = \frac{\sum_{m=1,N_t} I(x,z,t_m+t)I(x,z,t_m)}{N} \tag{2}$$

$$\langle I^2(x,z,t_0)\rangle_{t_0} = \sum_{m=1,N} I^2(x,z,t_m)/N_t$$

$$\langle I(x,z,t_0)\rangle^2_{t_0} = \sum_{m=1,N} I(x,z,t_m)/N, \tag{3}$$

with $\Delta t = 1/15$, $N = 9.9 \cdot 10^3$, corresponding to a time window of 11 min, and $N_t = N - \lceil t/\Delta t \rceil$.

The number of operations required for computing the autocorrelation function grows at least as $N \log N$ if it is done via a fast Fourier transform. In the present case, it was preferred a straight computation, even if it grows as N^2, in order to avoid other drawbacks related to the calculation of the Fourier transform of a finite series. To this aim, we developed a Julia package named *interference.jl* described below. The autocorrelation function is then obtained by first averaging the intensities over each metapixel, thus reducing to a matrix of dimension (25,42,9900), and then as direct products of 3-d matrices (see Sec. 6). Being the system rotationally invariant, the $g(x,z,t)$ have been averaged over x ($x \in [1,25]$), resulting in the $g(z,t)$, reported in the main text.

4 Smoothing and fitting procedure

Using raw intensity matrices over time, we often observed a sharp jump between the first and the second point of the autocorrelation function for some metapixels, suggesting the presence of some residual noise faster than our acquisition rate (15 fps). In order to address this issue, the inverse times $\nu_C = \tau_C^{-1}$, shown in Fig. 5, have been estimated by fitting with the positive part of a Gaussian $\propto \exp\left(-2t^2/\tau\tau_C\right)$ the short-time regime of autocorrelation functions obtained from smoothed intensity signals. Here τ_C is the characteristic physical time related to the autocorrelation decay while τ is the time window duration of the running average used to smooth the signals. In the following, we provide a simple mathematical argument to justify this procedure.

The smoothed intensities can be expressed as

$$I_s(t) = \frac{1}{\tau}\int_t^{t+\tau} dt' I(t') \tag{4}$$

and the non-normalized smoothed autocorrelations read

$$\tilde{g}_s(t) = \frac{1}{T}\int_0^T dt' I_s(t')I_s(t'+t). \tag{5}$$

Here we assume $I(t)$ to be the signals with the time averages subtracted. To simplify it, we avoid including the spatial and the ω dependence in the notation. Let us now show analytically how this smoothing suppresses the fast-decaying spurious contribution and why using a Gaussian fit for $\tilde{g}_s(t)$ is reasonable. Using Eq. (4) we can write:

$$\begin{aligned}\tilde{g}_s(t) &= \frac{1}{T}\frac{1}{\tau^2}\int_0^T dt' \int_{t'}^{t'+\tau} dt'' I(t'') \int_{t'+t}^{t'+t+\tau} dt''' I(t''') \\ &= \frac{1}{\tau^2}\int_0^\tau dt_1 \int_0^\tau dt_2 \frac{1}{T}\int_0^T dt' I(t_1+t')I(t_2+t'+t)\end{aligned} \tag{6}$$

where we changed variables as $t_1 = t'' - t'$ and $t_2 = t''' - t' - t$. The last integral of Eq. (6) can be expressed in terms of the original (i.e. non smoothed) autocorrelation function $\tilde{g}(t)$ only if the argument of one of the two integrated intensities is larger than the other. Indeed we have that

$$\frac{1}{T}\int_0^T dt' I(t_1+t')I(t_2+t'+t) = \begin{cases} \tilde{g}(t_1 - t_2 - t) & \text{if} \quad t_1 \geq t_2 + t \\ \tilde{g}(t_2 + t - t_1) & \text{if} \quad t_1 \leq t_2 + t \end{cases} \qquad \text{with} \quad t > 0 \tag{7}$$

From the above equation we see that the smoothed autocorrelation can be written as a sum of two contributions:

$$\tilde{g}_s(t) = \frac{1}{\tau^2}\int_0^\tau dt_2 \int_{t_2+t}^\tau dt_1 \tilde{g}(t_1 - t_2 - t) + \frac{1}{\tau^2}\int_0^\tau dt_1 \int_{t_1-t}^\tau dt_2 \tilde{g}(t_2 + t - t_1) \tag{8}$$

For $t \leq \tau$, which is representative of the short-time decay of $\tilde{g}_s$, we have to consider both contributions while for $t \geq \tau$ only the second one holds. This is because $t_1 \in [0, \tau]$ so if $t \geq \tau$ the inequality $t_1 \geq t_2 + t$ is never true.

In order to go on with the calculations, we need to assume a functional form for $\tilde{g}(t)$. The simplest choice is taking exponentially decaying correlations, with a characteristic time τ_C: $\tilde{g}(t) = A\exp(-t/\tau_C)$. It coincides with modelling the intensity signals $I(t)$ as Ornstein-Uhlenbeck processes. We make this assumption and later verify *a posteriori* his compatibility with empirical observations. By substituting exponential autocorrelations into Eq. (8) we get

$$\tilde{g}_s(t) = 2A\frac{\tau_C}{\tau}\left[1 - \frac{\tau_C}{\tau}\left(1 - e^{-\tau/\tau_C}\right)\cosh(t/\tau_C)\right] \quad \text{for} \quad t < \tau. \tag{9}$$

The first thing to note is that the above expression is proportional to τ_C/τ when this ratio is small. Thus, if the total signal contains a statistically independent fast spurious component with a characteristic time $\tau_C' \ll \tau \ll \tau_C$, then the autocorrelation of the smoothed signal will be dominated by the component proportional to τ_C/τ. Another important feature of Eq. (9) is that it presents a zero first derivative and a negative second derivative for $t \sim 0$, consistent with the behavior observed in Fig. 4 of the main text (compare panel C1 and C2). Considering now the normalized autocorrelation function $g_s(t) = \tilde{g}_s(t)/\tilde{g}_s(0)$ and expanding it at the leading order we find:

$$\tilde{g}_s(t) = 1 - \frac{2t^2}{\tau\tau_C}. \tag{10}$$

which is equivalent to the second-order expansion of the Gaussian $\exp(-(t/\tau_G)^2)$, where $\tau_G = \sqrt{\tau\tau_C/2}$. Such result justifies the use a Gaussian to fit the short-time part of the experimental autocorrelation function and explains how τ and τ_C appear in it. In principle one could perform the fit directly with Eq. (9), but we have found more efficient the use of a Gaussian since it properly interpolates the $t \sim \tau$ regime of Eq. (8) where Eq. (9) ceases to be valid.

5 Characteristic time and shear band limit detection

For each considered plate velocity, ω_p, the characteristic decorrelation time at each depth z, $\tau_C(z, \omega_p)$, has been determined with two methods, obtaining compatible values.

Gaussian fit

The method that yields the most accurate results is analyzing the autocorrelation function of the smoothed intensity signal. The short-time decay in the autocorrelation is fit with the positive part of a Gaussian function (coherent with Eq. 10), which allows extracting the characteristic time as a function of the metapixel height, $\tau_C(z)$, for each of the eight plate speeds considered. In the analysis, the inverse of the characteristic time $\nu_C(z)$ has been used because less noisy.

We have used a least-square method from the module *LsqFit* of Julia programming language to fit the Gaussian. The code implements the Levenber-Marquardt algorithm for non-linear fitting. The fitting function is a Gaussian with only the characteristic time $\tau_G = \sqrt{\tau\tau_C/2}$ as a parameter:

$$G(x, \tau_C, \tau) = \exp\left[-2\left(\frac{x}{\sqrt{\tau_C\tau}}\right)^2\right] \tag{11}$$

where the τ is the smoothing time, as discussed in the previous section, set equal to 5 (equivalent to 1/3 s). This allows extracting the characteristic $\tau_C(z)$, for each of the eight plate speeds considered. Its inverse, $\nu_C(z)$, expresses the decorrelation rate and has been employed in the analysis

Shear band detection. The decorrelation rates $\nu_C(z)$ allow to individuate the shear band characterizing the quasi solid and liquid phases. As seen in Fig. 5 of the main text, following the first maximum, $\nu_C(z)$ first decays and then enters a plateau associated with the solid phase. We assume the shear band as the region between the maximum of $\nu_C(z)$ and the beginning of the plateau. The starting of the shear band is easy to localize. Conversely, to individuate the plateau is less straightforward because of the noise in the data. We determine, within a certain approximation, the beginning of the plateau by associating it with the first relative minimum of $\nu_C(z)$ after the maximum. Assuming the curve to be smooth, to find the first minimum, one could compute the finite differences of the decorrelation rates $\Delta\nu_C(z) = \nu_C(z+1) - \nu_C(z)$, and look for the first point where the difference is non-negative. However, the data is noisy, and the finite differences can be non-negative also before the actual minimum in the data. Hence, we averaged the finite differences over a window of size w, and looked for the first point where the average is positive. A good estimate is obtained by averaging over all the window lengths in the range 3 metapixels to 13 metapixels. The maxima and the minima individuated with this method are shown in Fig. S3, together with the corresponding $\nu_C(z)$ profiles. The error bars in Fig 8 of the main text represent the standard deviation across the seven window lengths considered.

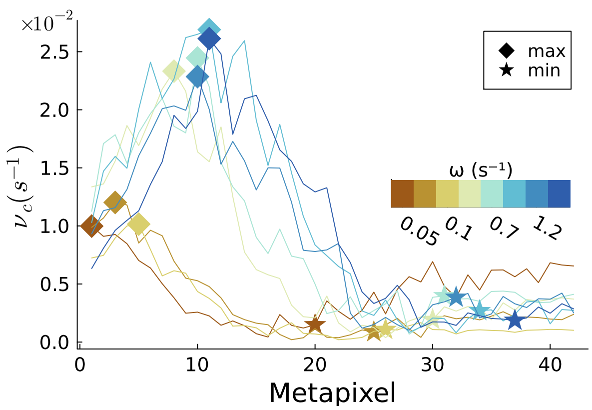

Figure S3: Profiles of the characteristic time for all the plate velocities considered. The minima and maxima of the inverse characteristic time are shown as stars and diamonds, respectively. The shear band is defined as the region between the minimum and the first maximum. The data shown here are the same used for Fig. 6 of the main text.

Half-height method

A second method to analyze the data was also tested; this method requires less analytical insight into the smoothing procedure and can be performed without fitting the auto-correlation decays. To estimate the characteristic time, we used the half-height method. The characteristic time $\tau_H(z)$ is when the (raw) auto-correlation function decays to half its initial value. The half-height method is less accurate than the Gaussian fit and more sensitive to noise; however, it can be performed on non-smoothed data and is therefore simpler and virtually unbiased by any timescale introduced for the analysis. The profiles of $\tau_H(z)$ and their inverse $\nu_H(z)$ are shown in Fig. S4. Because of the relatively coarse temporal resolution (determined by the camera frame rate), the resolution of the half-height method saturates on the short timescales associated with the beginning of the shear band. Consequently, the profile of $\nu_H(z)$ has a plateau near the maximum, and it is impossible to precisely detect the upward shear band limits. The right panel in Fig. S4 illustrates this limitation.

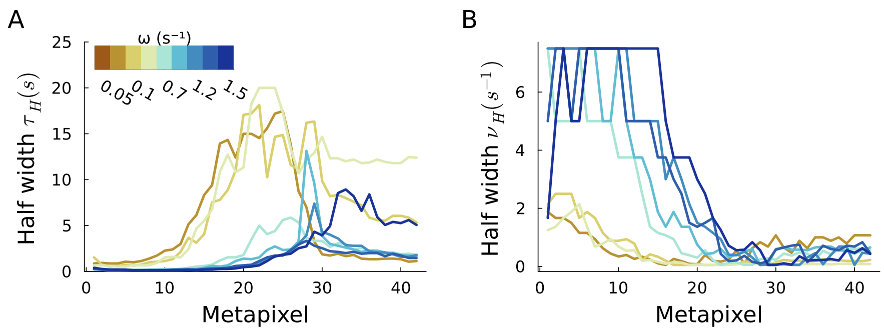

Figure S4: Characteristic time (and its reciprocal in the right panel) computed with the half-height method. The right panel better shows the saturation of the method in the short timescales corresponding to the start of the shear-band.

However, it is possible to compare the two methods and verify whether they yield compatible results. The scatter plots in Fig. 5 illustrate the characteristic times and their inverse for each metapixel and each plate velocity, showing that there is good agreement between the two measures.

6 Code for interference image analysis

The code *interference.jl* has been developed for the experiment itself and is available on GitHub[1]. The software uses bindings to Python Numpy to transform the `.avi` recordings into `.npy` matrices, which are then imported in Julia. We preferred to use Julia because the computation of the coarse-graining and the correlation is faster. The original matrices are sized $250 \times 420 \times$ T, where T is not fixed. The first two dimensions are the corresponding width and height pixel number of the single frame, the third dimension is the number of frames. Each element of the matrix represents the intensity at the corresponding pixel encoded in 8 bits - thus spanning from 0 to 255.

The code *interference.jl* offers a set of tools for data processing.

Signal coarse-graining and autocorrelation computation. The matrices are coarsened in time and space with a non-overlapping average-pooling algorithm - i.e. the value of coarsen matrix is the average of pixels' intensity in the corresponding region of the fine grain matrix. The spatial coarsen reduces raw signal noise, the coarsen in time also removes uninformative correlations on time scales that are irrelevant for the present study. The choice of the coarsening size is left to the user. The coarsen matrices are then stored in `Float32` format to limit the memory occupation.

[1] `https://github.com/aquaresima/interference.jl`

From the obtained three-dimensional matrix, the autocorrelation functions are then computed as direct products of 3D matrices realized in a parallelized way.

Estimation of the correlation time The time correlation function is computed at each metapixel within the assumption of stationary dynamics. Being $v_{ij}(t) = v_{ijt}$ the intensity at each metapixel, the correlation algorithm returns a tridimensional matrix:

$$\hat{C}_{i,j,s} = T^{-1} \sum_{t=0}^{T} \frac{v_{ij}(t)v_{ij}(t+s) - T^{-1}\sum_{t=0}^{T} v_{ij}^2(t)}{T^{-1}\sum_{t=0}^{T} v_{ij}^2(t) - (T^{-1}\sum_{t=0}^{T} v_{ij}(t))^2}. \tag{12}$$

The time dimension T -the maximum correlation time- is determined by the available number of frames N_f and the maximum lag s: $T = N_f - s$. Thus, each matrix element corresponds to the intensity autocorrelation at the metapixel i, j at the discrete time s. Assuming translational symmetry on the x-axis, the correlation function is averaged over the horizontal coordinate i to obtain a 2-d matrix of scalar functions of s, $\hat{C}_{j,s}$.

The characteristic decorrelation time τ_j is then determined with two methods:

- *Half-height characteristic time*: A decaying characteristic time scale of each height is determined as the value of s at which the function assumes half of its initial value (i.e. 0.5). One is then left with a vector of characteristic times: $\hat{\tau}_j$.
- *Gaussian characteristic time*: The first 10 points of the correlation function are fitted with a Gaussian function. The characteristic time is then computed as described in Eqs. (11).

Shear band detection.

The detection of the jump between solid and fluid state, respectively left and right in Fig. S3, corresponds to detect the maximum and first minimum of the function τ_i. The algorithm described in Sec. *Characteristic time and shear band limit detection* is implemented in the code.

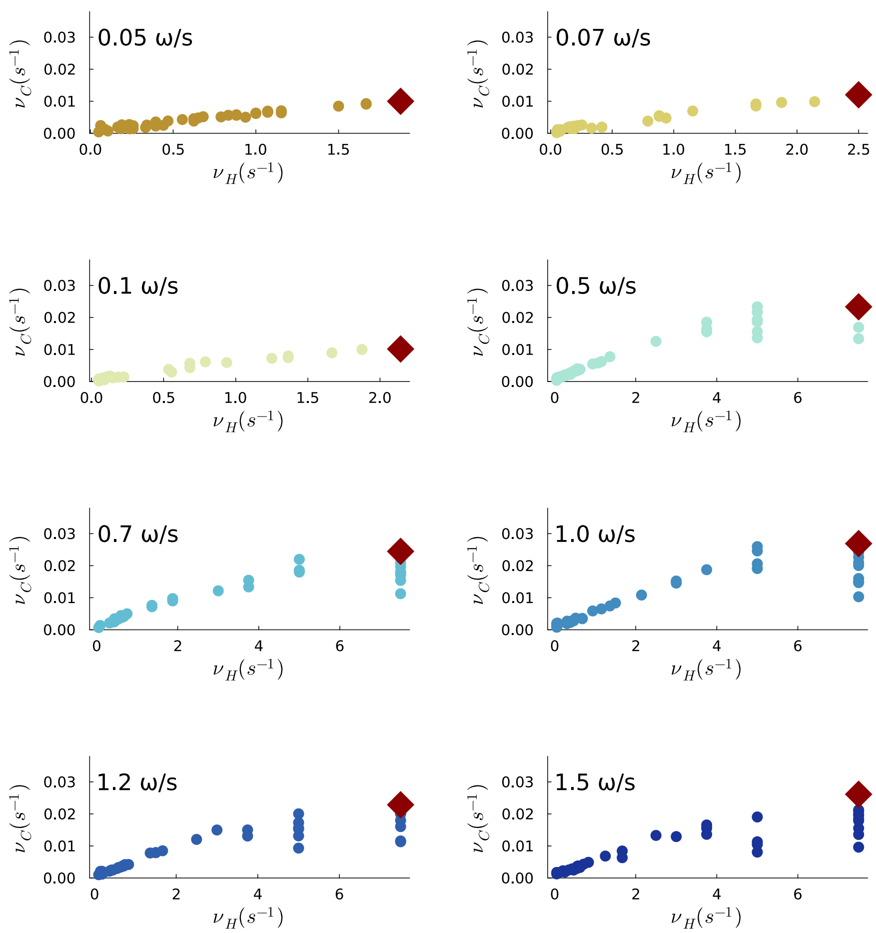

Figure S5: Comparison of the decorrelation rate computed with the Gaussian fit ν_C, and the one computed with the half-height method ν_H. The two methods yield compatible results. The diamond indicates the maximum value of the decorrelation rates, which is used to define the starting of the shear band. For large shear rates, ν_H saturates (see Fig. S4B)