Electronic Supplementary Material (ESI) for Soft Matter.

# Supporting Material

for

# Dynamics of force generation by spreading platelets

Jana Hanke[1#], Dimitri Probst[2,3#], Assaf Zemel[4], Ulrich S. Schwarz[2,3]*, Sarah Köster[1,5]*

[1]Institute for X-Ray Physics, University of Goettingen, Friedrich-Hund-Platz 1, 37077 Göttingen, Germany
[2]Institute for Theoretical Physics, Heidelberg University, Heidelberg, Germany
[3]BioQuant-Center for Quantitative Biology, Heidelberg University, Heidelberg, Germany
[4]Institute of Dental Sciences and Fritz Haber Center for Molecular Dynamics, Hebrew University of Jerusalem, 91120, Israel
[5]German Center for Cardiovascular Research (DZHK), partner site Göttingen, Germany

[#]these authors contributed equally to this work
*correspondence: schwarz@thphys.uni-heidelberg.de and sarah.koester@phys.uni-goettingen.de

## 1. Preparation of PAA Working Solution

**TABLE S1 Amount of acrylamide and bis-acrylamide** used for 10 ml of PAA working solution for the substrate stiffnesses used. The remaining volume is 1X PBS. The elastic modulus was determined by rheology.

| Elastic Modulus (kPa) | 40 % Acrylamide (mL) | 2 % Bis-Acrylamide (mL) |
|---|---|---|
| 19.4 ± 0.5 | 2 | 0.7 |
| 29.3 ± 0.5 | 2.5 | 0.75 |
| 41.2 ± 0.9 | 2.5 | 1.3 |
| 54.1 ± 0.7 | 2.5 | 2.25 |
| 83.1 ± 0.3 | 3.75 | 1.5 |

## 2. Data Analysis

Our aim is to determine the temporal evolution of traction forces from a 30-min-recording of fluorescent beads. We require the algorithm to fulfill a number of criteria. First, it shall be applicable to small cells. Thus, a high resolution in space is needed. Second, it shall be suitable for cells that exert high forces within short time intervals. While this is already taken into account during imaging by choosing a small temporal resolution $\Delta t$, the analysis algorithm must also account for this aspect. Lastly, given the high bead density and large number of frames to be compared to each other, an automated algorithm is necessary. In the following we describe two variants of our method that fulfil these criteria, an algorithm that is based on PIV and data smoothing (*smoothing-based approach*) (Fig. S1A), and one that is based on optical flow and regularization (*regularization-based approach*) (Fig. S1B).

### 2.1 Smoothing-Based Approach

PIV analysis is performed using MATLAB (MATLAB R2009b, The Mathworks, Natick, MA, USA). First, due to the relatively low signal-to-noise (SNR) in the images, we increase the contrast between beads and background (Fig. S1A, step 1). To this end, non-linear signal enhancement is applied. This amplification scales the intensities onto the interval [0,1] such that the smallest intensity is mapped to 0 and the highest intensity to 1. We then define a new intensity distribution such that $s \rightarrow s^p$ for $s \in [0,1]$ is the scaled intensity and $p > 1$. The choice of $p$ depends on the noise in the images; for higher noise, a higher value for $p$ is used. For the data presented here, $p$ is chosen between 1.7 and 2.8. For additional denoising, a Wiener filter of size 3 x 3 pixels is applied. The image smoothing is followed by drift correction to account for small shifts in the field of view (FOV). This is achieved by cross-correlation in Fourier space as described in Ref. (1) (Fig. S1A, step 2).

In general, traction forces are calculated using the integral equation (2,3)

$$d_j(\bar{x}) = \int_A \sum_l G_{jl}(\bar{x} - \bar{x}') f_l(\bar{x}') d\bar{x}' \qquad (1)$$

where $\bar{f}(\bar{x})$ denotes the two-dimensional traction force field, $\bar{d}(\bar{x})$ the displacement field measured by following the beads, $A$ the area on which the forces are studied, and $G_{jl}(\bar{x})$ the Boussinesq Green's function (3). Thus, for determination of the traction forces, we first need to determine the displacement vectors relative to the null image. Given the aforementioned fast force dynamics on a small area at high bead density, direct bead tracking is difficult when working with platelets. At the same time, however, the highly dynamic bead patterns result in so-called loss of pairs common for PIV techniques. Thus, Eq. 1 is solved in three steps (see Fig. S2). First, the velocity field is calculated by PIV (Fig. S1A, step 3) using the MATLAB toolbox mPIV (http://www.oceanwave.jp/software/mpiv). The statistical mean velocity inside a small interrogation sub-window at a certain time point $t_i$ is estimated by calculating the cross-correlation inside the sub-window between the present frame and a reference frame (Fig. S2A). To gain reliable velocities at this stage, the changes in bead pattern between the two images we compare must be sufficiently small in order to reduce bead loss within the single interrogation windows. For cells, which exert large forces, we compare two consecutive images, rather than the present image and the very first one, thereby anticipating large displacements within short time intervals. Additionally, this approach allows for arbitrarily long recording times given a sufficiently small $\Delta t$. This marks the strength of the algorithm presented here compared to other tTFM analysis algorithms (3–5) since the algorithm is adaptable for various cell sizes, force dynamics and observation times.

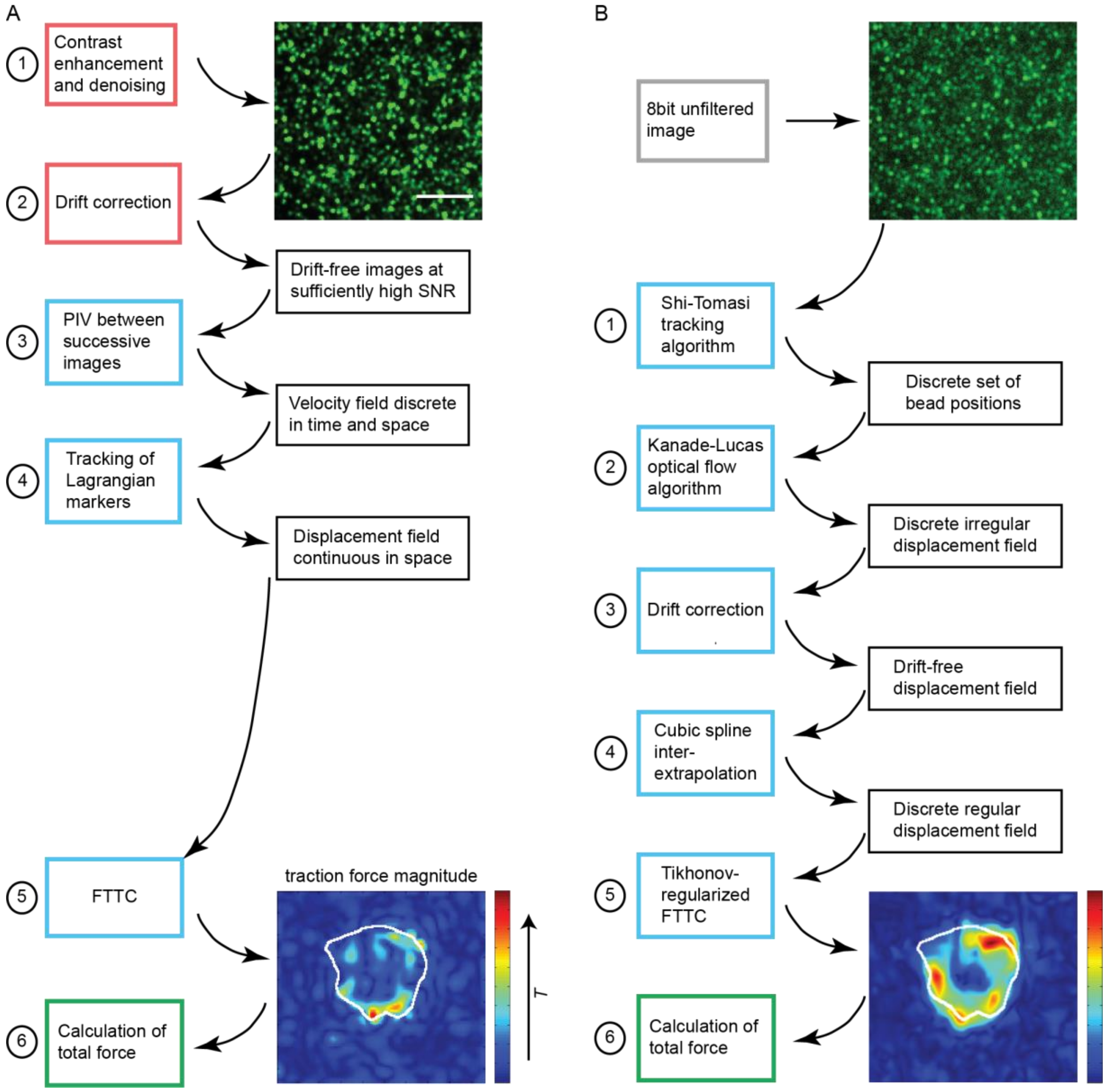

**FIGURE S1 Work flow in the algorithms for traction force computation.** (A) Smoothing-based approach and (B) regularization-based approach. The colored boxes on the left-hand side in each panel show the individual steps in the algorithm. The colors describe the classification of each step: image pre-processing (red: (A), step 1 and 2), main calculation (blue: (A), step 3 to 5 and (B), step 1 to 5) and post-processing (green: step 6). On the right-hand side of each panel, the results for each intermediate step are given. The bottom image shows an example of the traction force magnitude distribution for a given time point together with the corresponding cell outline in white. Scale bar: 5 µm.

The contraction movie of the bead pattern below and near a single cell is first divided into sub-windows in a coarser grid (5.9 or 9.2 µm$^2$ per sub-window depending on the contraction speed), with a 50 % overlap of the individual windows in each direction. In a following recursion step, the edge length of the sub-windows is reduced by a factor of 2 for the final, finer grid. The velocity fields determined on the coarser grid serve as a guide to determine the velocity field of the finer grid. In our experience, this approach yields a higher accuracy than starting directly with a fine grid. By applying a median filter to the velocity field, erratic velocity vectors are removed.

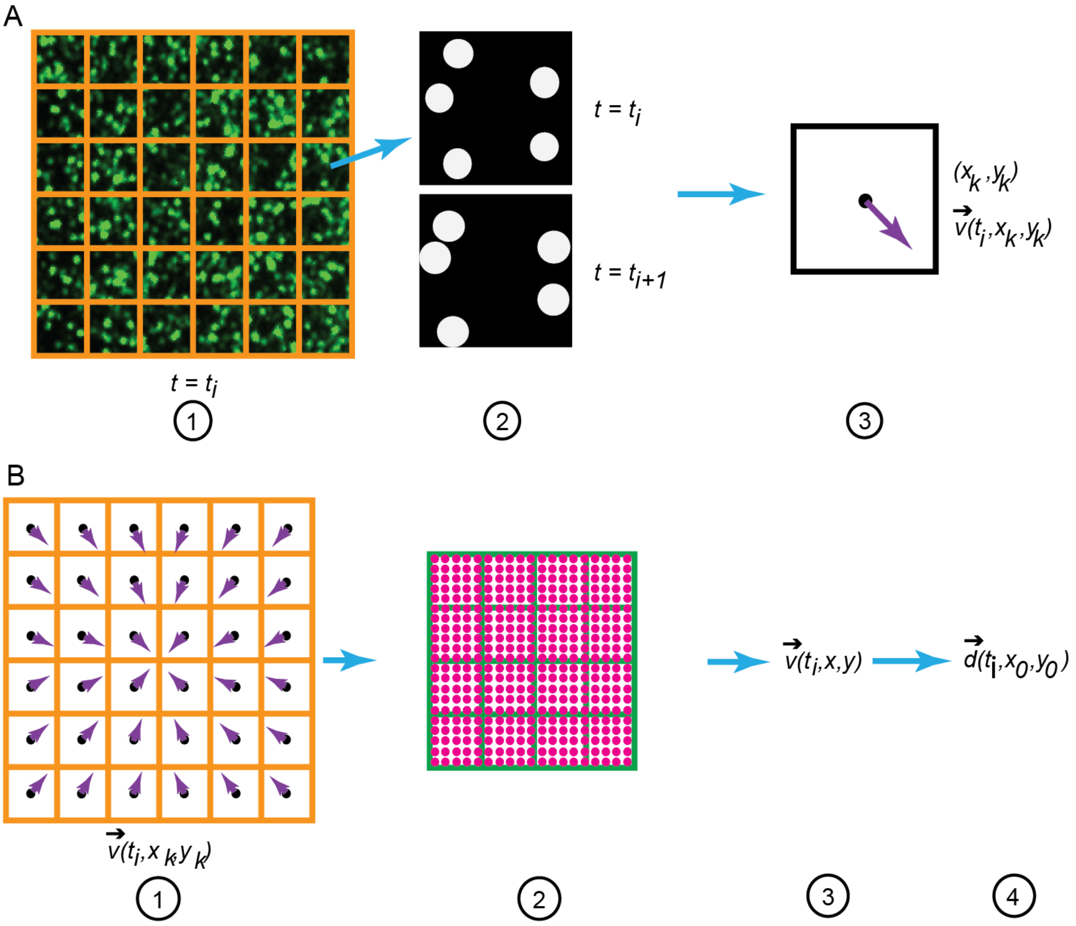

**FIGURE S2 Schematic of the evaluation steps 3 and 4 shown in Fig. S1A.** (A) Detailed description of step 3. The bead image is divided into sub-windows (orange, step 1). Note that the overlap between sub-windows is not shown here for simplicity. For a given sub-window, we study the bead pattern at two subsequent time points (step 2). Note that all beads have moved differently but none left the window. By cross-correlation, we calculate the statistical mean velocity between the two windows (violet arrow, step 3). (B) Detailed description of step 4 in Fig. S1A. For each time point, we study the velocity field for the given grid (step 1). Defining a new evaluation window by disregarding sub-windows at the border, we distribute Lagrangian markers regularly on a 2x2 pixel grid inside the window (magenta, step 2). We locally approximate the continuous velocity field using periodic cubic splines (step 3) and recursively calculate the displacement for each marker (step 4).

It can be shown that statistical errors arising during the calculations are reduced by always comparing frames $t_i$ and $t_{i+1}$ both forward and backwards and averaging the results to obtain the final velocity field. All sub-windows containing no velocity vectors are assigned a velocity by interpolating the neighboring vectors. The advantage of using PIV in combination with interpolation compared to direct bead tracking becomes particularly obvious when bead clusters are present. For small clusters, the sub-windows can still be analyzed while the velocity of slightly larger clusters can be determined by local Kriging interpolation. This approach yields a spatial resolution of 600 nm, given the distance of mid-points between neighboring sub-windows, similar to the definition of resolution used in Ref. (3).

We have now determined an instantaneous velocity field $\bar{v}(t_i, \bar{x}_k)$, where $\bar{x}_k$ is the position of the mid-point of the interrogation window $k$ and $t_i$ is the current time point. To determine the traction forces relative to a relaxed state, we need to calculate the displacement fields relative to $t_0$ such that the displacement $\bar{d}$ at time point $t$ is given as

$$\bar{d}(t) = \bar{x}(t) - \bar{x}(t_0) \tag{2}$$

Mathematically, we need to solve the ordinary differential equation

$$\dot{\bar{x}} = \bar{v}(t, \bar{x}) \tag{3}$$

with the given initial position $\bar{x}(t_0)$. Thus, we solve Eq. 3 numerically by the symplectic implicit midpoint method such that

$$\bar{x}(t_{i+1}) - \bar{x}(t_i) = \Delta t \cdot \bar{v}\left(t_i + \frac{\Delta t}{2}, \frac{\bar{x}(t_i) + \bar{x}(t_{i+1})}{2}\right) \tag{4}$$

We proceed by selecting a finite set of uniformly distributed Lagrangian markers which are sufficiently densely packed inside the complete domain (Fig. S2B, step 2). These markers are defined by their individual starting position $\bar{x}(t_0)$ without further physical properties. The concept is borrowed from fluid dynamics, where Lagrangian markers are the tracer particles studied in the Lagrangian description of a flow field (Fig. S1A, step 4). For simplicity and to avoid additional interpolation steps, the markers are placed on a rectangular grid which will later also be used to determine the local traction forces. The density of the Lagrangian markers influences the approximation quality of the FTTC result: the higher the number of markers the better the approximation. At the same time, the calculation time increases linearly with the number of markers. Here, we found that a 2 x 2 pixel grid for the markers is optimal for our purpose.

As described above, the velocity field is determined by comparing two frames taken at $t_i$ and $t_{i+1}$ and thus corresponds to the average velocity between the time points $t_i$ and $t_{i+1}$. A reasonable assumption is, therefore, that the beads are moving with exactly that velocity during the whole interval $[t_i \ \ t_{i+1}]$, and thus also at $t = t_i + \frac{\Delta t}{2}$. The velocity field can be used immediately in Eq. 4

For each velocity field $\bar{v}(t_i, \bar{x}_k)$, the single velocity entries determined by PIV are assigned to the center of the corresponding interrogation window (Fig. S2B, step 1). As PIV may cause instabilities at the image edges, the interrogation windows near the edges are disregarded in the computations. Given the starting position of each marker, we can now solve Eq. 4 recursively (Fig. S2B, step 4). The initial guess for the next marker position is provided by an Euler forward approach. To gain locally continuous velocity fields, the previously determined velocity fields are interpolated using periodic cubic splines. In contrast to conventional cubic splines, we periodically continue the velocity field by three sub-windows in each direction before locally approximating by a cubic spline. To avoid markers leaving the evaluation frame or giving rise to high disturbances at the edges due to remaining instabilities, we adopt homogeneous Dirichlet boundary conditions, $\bar{v}(t_i, x_{edge}, y_{edge}) = 0$. These boundary conditions can be justified by the assumption that the force field is of finite size and fully enclosed in the studied frame such that the gel is fixed at the boundaries.

As we need to move our markers continuously over the whole studied area, we have to convert the velocity field, which is discrete in space, into a locally continuous field.

After solving Eq. 4 for all time points $t_i$, the displacement field as defined in Eq. 2 is available and the traction forces can be calculated by solving Eq. 1. As this equation is a convolution of the Green's function and the force field, it is often solved in Fourier space where it can be written as

$$\tilde{d}_{j,\bar{k}} = \left\{\sum_l \tilde{G}_{jl}\tilde{f}_l\right\}_{\bar{k}} \tag{5}$$

With the corresponding wave vector $\bar{k}$. Thus, we can now use the FTTC algorithm as described in Refs. (3,6) without additionally regularization as we can show that this step is not necessary in the present application (Fig. S1A, step 5).

To be able to compare two cells with each other concerning the temporal evolution of their contractile behavior, we define the scalar quantity 'total force' as reported previously (5,7) as

$$F_{tot} = \int_{A_{ROI}} |\bar{T}(\bar{x})|\, dA \tag{6}$$

where $\bar{T}(\bar{x})$ is the traction force and $A_{ROI}$ defines a rectangle around the cell large enough to enclose all force contributions stemming from the cell, but sufficiently small to exclude instabilities at the edge (Fig. S1A, step 6). We deliberately do not use the energy as a physical quantity for the comparison as the noise in our data introduces large errors in the energy calculation (see Fig. S3 and S4).

## 2.2 Regularization-Based Approach

Simultaneously, we evaluate our data with an algorithm based on optical flow and regularization (Fig. S1B). Previously, it was observed that optical flow yields higher forces and more precise results than PIV for large cells with small, defined focal adhesions (8). Here, we test if this also holds true for small cells where we cannot distinguish any defined focal adhesions in our data.

For the following algorithm, the open source OpenCV is used (Open Source Computer Vision Library, https://github.com/itseez/opencv, 2015). To use the optical flow algorithm, we need to find features of interest to track. In our case, these features are the beads located in the gel. The beads in the images are characterized through changes in the intensity over neighboring pixels, which can be detected by using an appropriate edge detection algorithm. Here, we apply the Shi-Tomasi corner tracking algorithm (9,10) (Fig. S1B, step 1). To do so, the original 16-bit recordings are truncated to 8-bit images. For each pixel $(x, y)$ in the image, a small surrounding window of size $a \times b$ is considered. Here, we use a square window such that $a = b = 11$ pixels. The change in intensity between the original window and the window moved by a translational vector $(\Delta x, \Delta y)$ is measured by the sum of squared differences (SSD)

$$SSD(x,y) = \sum_{(u,v)} \left(I(u+\Delta x, v+\Delta y) - I(u,v)\right)^2 \tag{7}$$

where $I$ is the intensity of a pixel and the sum is taken over all pixels $(u, v)$ belonging to the window. The intensity of the shifted sub-image $I(u + \Delta x, v + \Delta y)$, can be approximated by a Taylor expansion, reducing Eq. 7 to

$$SSD(x,y) \approx \sum_{(u,v)} \left(\frac{\partial I(u,v)}{\partial u}\Delta x + \frac{\partial I(u,v)}{\partial v}\Delta y\right)^2 \tag{8}$$

Introducing the structure tensor

$$A = \sum_{(u,v)} \begin{pmatrix} \left(\frac{\partial I(u,v)}{\partial u}\right)^2 & \frac{\partial I(u,v)}{\partial u} \cdot \frac{\partial I(u,v)}{\partial v} \\ \frac{\partial I(u,v)}{\partial u} \cdot \frac{\partial I(u,v)}{\partial v} & \left(\frac{\partial I(u,v)}{\partial v}\right)^2 \end{pmatrix}, \tag{9}$$

Eq. 8 reduces to

$$SSD(x,y) \approx (\Delta x\ \Delta y) A \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \tag{10}$$

SSD describes the intensity changes within our window if it is moved into the direction $(\Delta x, \Delta y)$. A localized feature like a bead is characterized by large changes in all directions of movement. The degree of change is characterized by the eigenvalues $\lambda_1$ and $\lambda_2$ of $A$. So for each pixel $(x, y)$, the function $R = \min(\lambda_1(x,y), \lambda_2(x,y))$ is considered. All pixels with a larger $R$ than a given threshold are counted as features. The threshold is set to be $q \cdot R_{max}$, with $R_{max} = \max_{(x,y)} R(x,y)$ and $q$ being a quality level. Thus, image noise is suppressed. Furthermore, to avoid multiple detection of larger beads or bead clusters, a minimum distance of 3 pixels between features is defined. Additionally, features with a $R$ less than 5-6% of $R_{max}$ are discarded and no more than a total amount of 1000 features are considered. The remaining features are designed as the discrete bead positions to be tracked in the following images.

Given the starting position of our beads, we determine their displacement by means of an optical flow algorithm. Here, we use the pyramidal Kanade-Lucas-Tomasi (KLT) algorithm (11,12) (Fig. S1B, step 2). Consider two consecutive frames taken at times $t$ and $t + \Delta t$ and a bead located at position $(x, y)$. Then $I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$, at least approximately, if the bead moves by $(\Delta x, \Delta y)$. In case of an infinitesimal time difference $\Delta t$ the limit $\Delta t \to 0$ yields the optical flow equation

$$\frac{\partial I(x,y)}{\partial x} v_x + \frac{\partial I(x,y)}{\partial y} v_y + \frac{\partial I(x,y)}{\partial t} = 0 \tag{11}$$

Here, $\bar{v} = \left(v_x, v_y\right)$ denotes the velocity of the bead's movement. Once $\bar{v}$ has been determined, the displacement of the bead between the two frames can be estimated by $\bar{d} = \bar{v}\Delta t$. Note that here we assume the velocity to be constant between two frames.

Eq. 11 is not only underdetermined, but even subject to erroneous data. In order to get a more reliable estimate of the velocity, a window of size $s \times s$ around a bead is considered. The general idea of the Kanade-Lucas algorithm is that, similar to PIV, neighboring pixels have a similar motion. Thus, Eq. 11 is required to hold for all $s^2$ pixels of the sub-frame. This provides us with $s^2$ linear equations for the two scalar unknowns $v_x$ and $v_y$. This system is solved in a least-squares sense. The displacement is then calculated for increasingly smaller windows according to the pyramidal Kanade-Lucas-Tomasi algorithm, where the displacements in the larger windows are used as an off-set for the displacement of the smaller windows. Windows are divided into quarters until a given number of iterations is reached. Here, in the first run, we consider a window size of 64 pixels, in the second run, the window size is halved to 32 pixels. In a similar manner as with the previous algorithm, we track our beads between successive frames and move them linearly forward in time. This is equivalent to solving Eq. 3 by the explicit Euler method. Also, as described above, to reduce statistical errors, we always compare the images both forward and backwards and average the results. The result of this step is a set of displacements over time. By choosing an image corner in which no displacement should be observed, we can determine the drift occurring during recording (Fig. S1B, step 3). Image drifts are removed by subtracting the mean displacement of a box of 50 x 50 pixels in a cell-free region from all displacement vectors. This approach has the advantage over standard drift-correction techniques, such as the one provided by ImageJ, that the mean drift is not affected by any imbalance of the displacement field established by the cell itself. Finally, the irregular displacement field is projected to a regular grid using Scipy's bivariate cubic spline algorithm (open source scientific tools for Python, http://www.scipy.org, 2017). In cases in which the displacements at the boundary of the image are significantly larger than 0, we apply a two-dimensional Tukey filter with constant $\alpha = 0.2$ in real space (13).

We calculate the force field again by FTTC which requires a regular grid of discrete displacements. Hence, we extrapolate our discrete, irregular displacements on a regular grid (Fig. S1B, step 4). Contrary to the previous algorithm, for the optical flow, Tikhonov regularization is needed. The FTTC with regularization is performed as described in Ref. (3) except for the determination of the regularization parameter (Fig. S1B, step 5). Instead of using the L-curve criterion as described previously, we use Strong Robust Generalized Cross Validation ($R_1$GCV) (14). Here, for the tuning parameter $\gamma$, a value of 0.95 is used. From the calculated traction force maps, we again determine the total force as described previously (Fig. S1B, step 6).

## 2.3 Error Evaluation

In order to evaluate the error included in our data and analysis independent of the specific algorithm, we define a quantity which takes into account the whole data acquisition process at once. This error can be defined by the vector sum over all traction forces. Theoretically, for adherent, non-motile cells, the summation over all traction forces should yield no total net force. However, practically, this is not the case due to statistical noise and discretization error in the evaluation process. We can thus define the total error as a function in time as

$$e(t) = \frac{|\sum_{\vec{x}} \vec{T}(\vec{x},t)|}{\sum_{\vec{x}} |\vec{T}(\vec{x},t)|} \tag{12}$$

For the presented data, we observe that the error function reaches a stable level after the initial contraction period. For the results presented here, we only consider cells which show an error level of less than 10 % (Fig. S5).

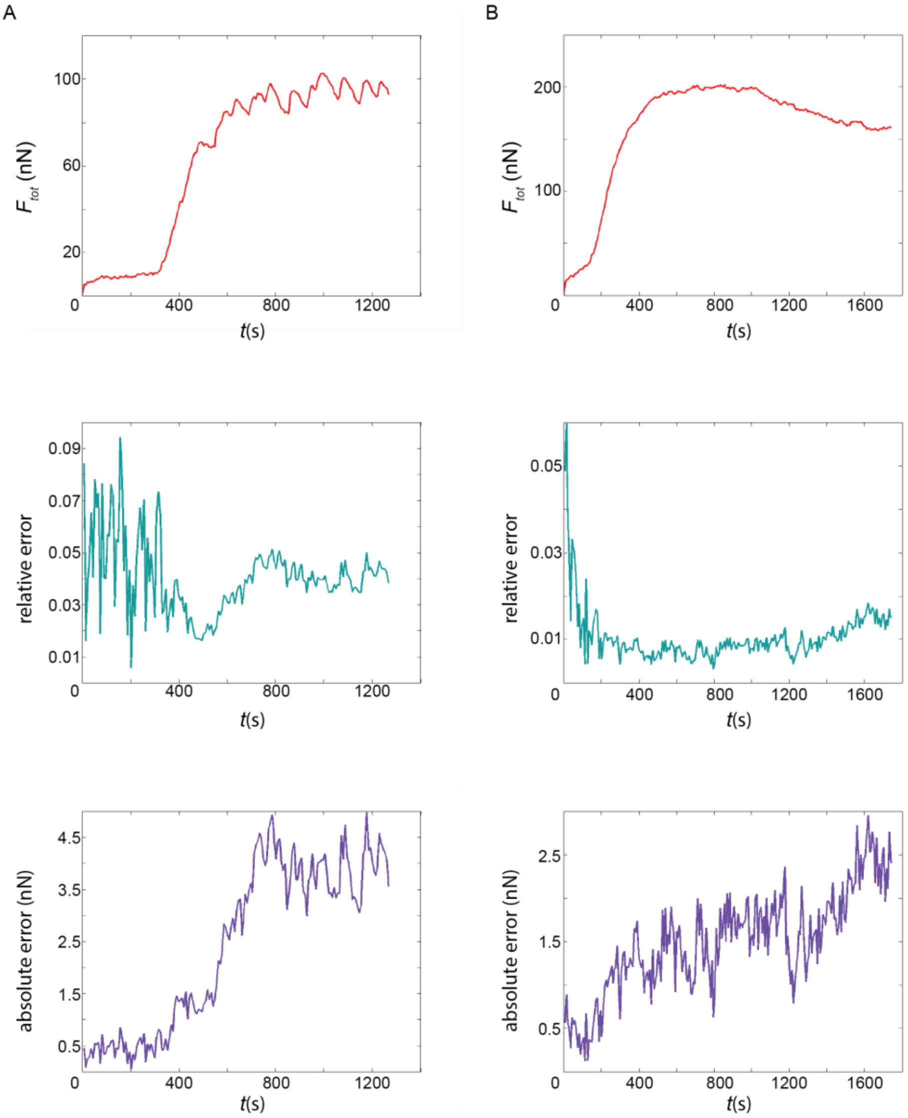

**FIGURE S3 Two examples of the total forces with corresponding errors.** For both data sets, from top to bottom, the total force, the relative error as defined in Eq. 12 and the absolute error are shown. (A) Total force of an oscillating platelet with a relative error of between 0.04 and 0.05, corresponding to 4 % and 5 %, respectively. (B) Total force of a contracting then relaxing platelet. While the relative error increases during the relaxation, the absolute error remains below 3 nN. Only cells with less than 10% error or 0.1 relative error are considered. This bound is carefully chosen to take several phenomena into account. First, platelets may relax again after initial contraction as seen in panel B. Secondly, some platelets contract only slightly. For both situations, an error level of 10 % often corresponds to an absolute error of a few nN which is negligible considering the total force level during the whole contraction. Lastly, as mentioned previously, we work with images of low SNR, analyzing them with several statistical and discrete methods, which leads to an accumulation of small errors.

## 2.4 Direct comparison of both methods

A direct comparison of the results gained by both algorithms for a typical data set is shown in Fig. S6. In this case, the cell shown was allowed to spread and contract on a PAA gel with 19 kPa stiffness and we show the spatial distribution of the force magnitudes $T$ at the end of the recorded data set, *i.e.* after 21 mins after adhesion of the cell to the substrate (Fig. S6A, B), and the calculated total force $F_{tot}$ (see Eq. 6) plotted against time (Fig. S6C, D). Despite minor visible differences between the outputs of both algorithms in the force magnitude maps, the main biophysical results are equally well supported by both methods.

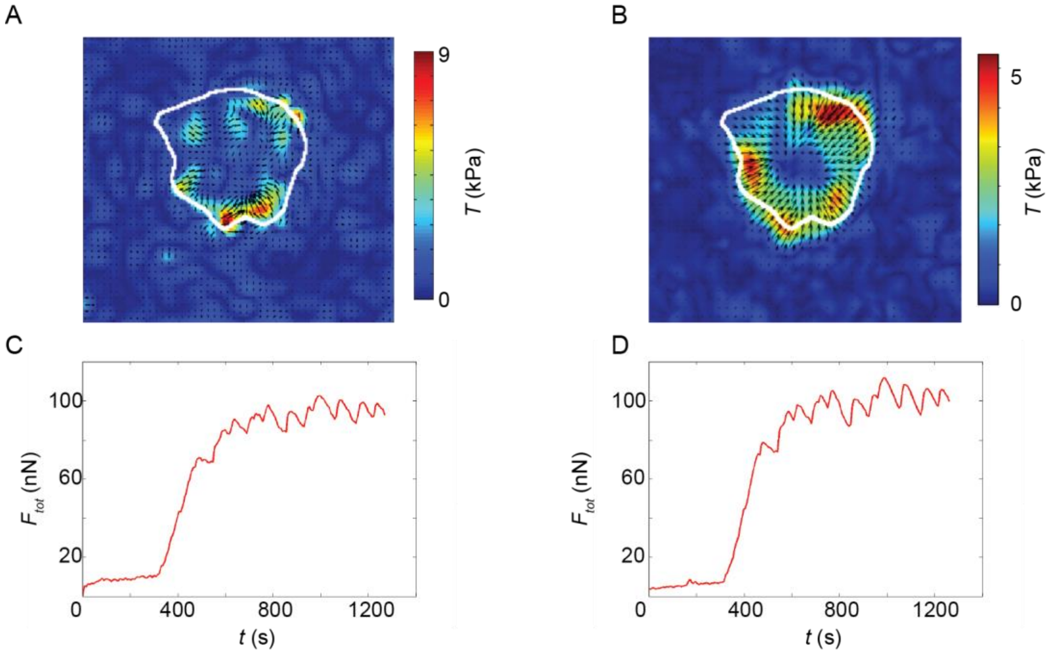

**FIGURE S4 Results for both our algorithms in direct comparison** (example on a 19 kPa substrate, shown is the last frame of the recording after 21 min). (A) Traction force magnitude maps and (C) temporal evolution of the total force $F_{tot}$ as obtained for the PIV-based algorithm. (B, D) Corresponding results for the optical flow-based algorithm.

## 3. Dominant Frequency of Temporal Evolution of Force Data

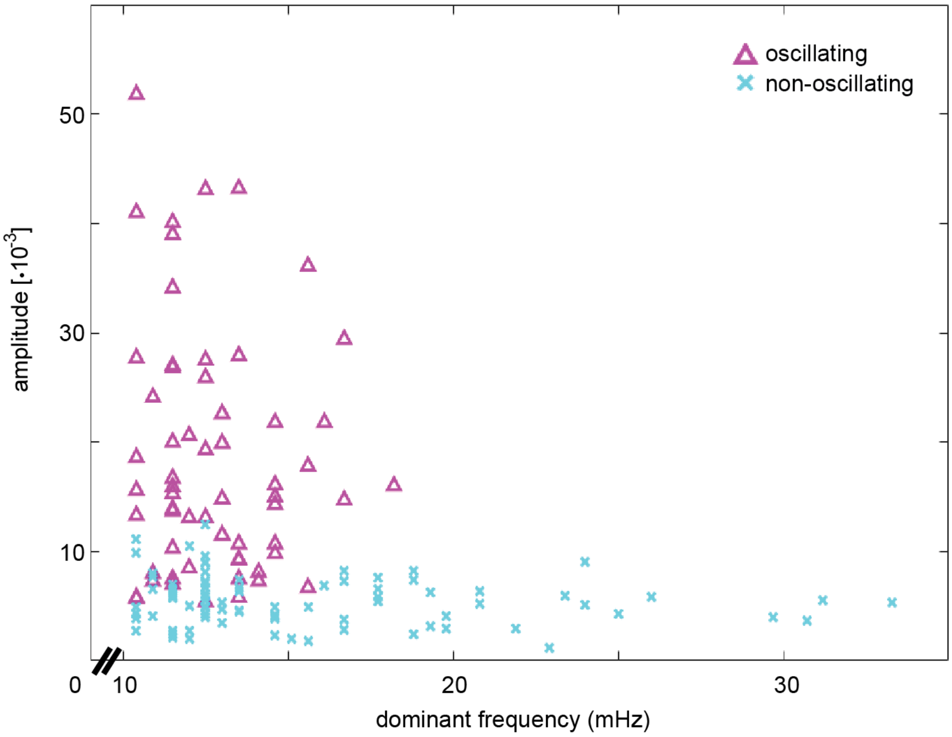

**FIGURE S5 For each platelet, the dominant frequency of the temporal evolution of the total force was determined**. Here, we only considered the data after the initial contraction, *i.e.* only the plateau, oscillating and relaxing parts. Furthermore, we were interested in the changes in the total force relative to the average force and thus divided the force by the mean force measured in the studied part. To avoid the influence of high-frequency noise as well as long-time increases and decreases in force, we filtered the data with a band-pass filter to only include frequencies between 10 and 35 mHz. In general, the platelets we consider to be oscillating from inspecting the movies directly exhibit a higher amplitude in the Fourier domain than non-oscillating cells. While non-oscillating cells very rarely show the relative amplitudes of 10 or higher, oscillating cells show amplitudes as high as 50. Also, the frequency band in which the highest frequency for oscillating platelets is detected is comparably narrow with an average frequency of about 12.8 mHz.

## 4. Temporal evolution of the total forces

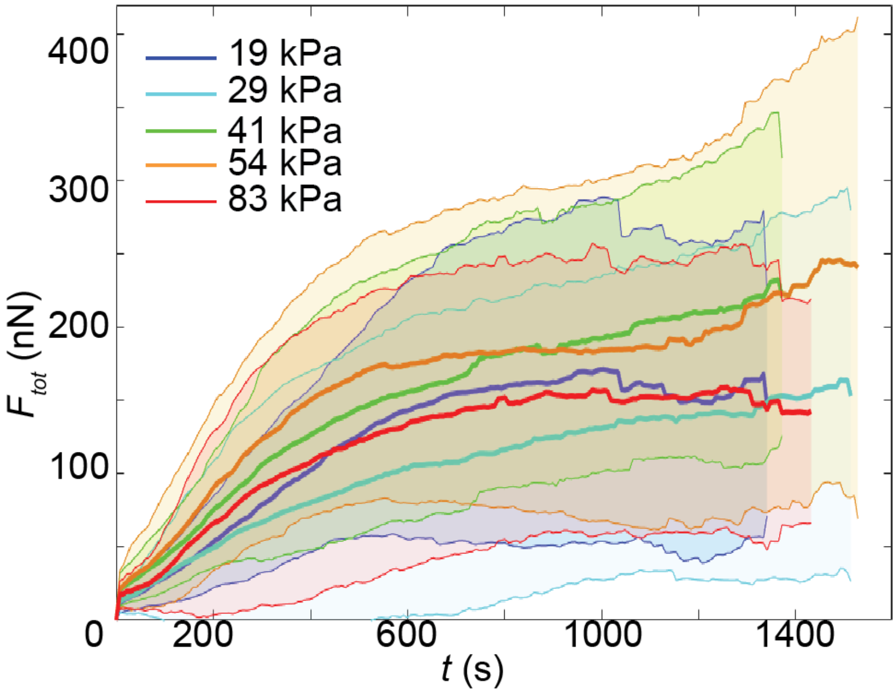

**Figure S6:** Data from Fig. 2B replotted including the standard deviation instead of standard error. The strong overlap between the different data sets becomes even more apparent in this form, suggesting that force generation does not depend on stiffness for fully activated platelets in the stiffness range between 19 and 83 kPa.

# 5. Mathematical Model

The cytoskeleton of blood platelets of height $h_c$ is modelled as a thin elastic disc of radius $r_0$, Young's modulus $E_c$, Poisson's ratio $\nu_c$ and an isotropic active contractile stress $\sigma_0$. In order to model the elastic interaction of the platelet with its environment, the elastic disc is further connected to the underlying substrate using a spring stiffness density $Y$, representing both the substrate stiffness and the stiffness of the focal adhesion (FA) bonds (15). This yields the force balance equation[1]:

$$\sigma_{ij,j} - Yd_i/h_c = 0 \tag{13}$$

We assume small displacements $d_i$ with respect to the cell size and therefore use the linearized strain $\epsilon_{ij} = \frac{1}{2}(d_{i,j} + d_{j,i})$, which leads to the constitutive relation:

$$h_c\sigma_{ij} = 2\mu\epsilon_{ij} + (\lambda\epsilon_{kk} + h_c\sigma_0)\delta_{ij} \tag{14}$$

with the two-dimensional Lamé coefficients

$$\lambda = \frac{h_c E_c \nu_c}{1 - {\nu_c}^2} \tag{15}$$

and

$$\mu = \frac{h_c E_c}{2(1 + \nu_c)} \; . \tag{16}$$

We further assume a constant active stress $\sigma_0$ throughout the platelet, which leads to $\sigma_{0,i} = 0$ within the platelet and manifests itself as the remaining stress normal to the exterior boundary of the platelet, introduced via the boundary conditions. The force balance equation now simplifies to:

$$h_c\sigma_{ij,j} = \lambda d_{k,ki} + \mu\big(d_{i,jj} + d_{j,ij}\big) = Yd_i \tag{17}$$

Rearranging this equation we write: $\frac{1-\nu_c}{2} d_{i,jj} + \frac{1+\nu_c}{2} d_{j,ij} = \frac{d_i}{l_L^2}$, where $l_L$ is the localization length, $l_L = \sqrt{\frac{h_c E_c}{Y(1-{\nu_c}^2)}}$ and can be interpreted as the length, up to which a point force is transmitted. To distinguish between the elastic contributions of the adhesion bonds themselves and substrate stiffness we calculate the localization length via the interpolation formula introduced previously [2] (16):

$$l_L = \sqrt{\frac{E_c h_c}{N_a k_a} + \frac{E_c h_c}{2\pi E_s}\left(\frac{1}{\pi h_s(1 + \nu_s)} + \frac{1}{r_0}\right)^{-1}} \tag{18}$$

Here, $E_S$, and $\nu_s$ are the Young's modulus and Poisson ratio of the substrate, respectively, $N_a$ is the density (per unit area) of adhesion complexes at the cell-substrate interface, and $k_a$ is a local effective spring constant of an adhesion complex. This reveals an expression for the effective stiffness density, $Y$, that accounts for the combined stiffness of the focal adhesion layer and the substrate as follows:

$$Y = \frac{h_c E_c}{l_L^2(1 - {\nu_c}^2)} = \frac{1}{1 - {\nu_c}^2}\frac{N_a k_a E_s/\tilde{h}_s}{N_a k_a + E_s/\tilde{h}_s} \tag{19}$$

where $\tilde{h}_s^{-1} = \frac{1}{2(1+\nu_s)h_s} + \frac{\pi}{2r_0}$.

---

[1] Indices before a comma refer to tensor indices, indices after a comma to derivatives.

[2] The interpolation formula has been derived for a 1D cell and matrix system; we adopt it here as an approximation that correctly captures the effects of substrate stiffness and thickness.

Introducing the strain tensor in cylindrical coordinates into the constitutive relation (Eq. 17) yields

$$r^2\frac{\partial^2 d_r}{\partial r^2}+r\frac{\partial d_r}{\partial r}-\left(1+\frac{r^2}{{l_L}^2}\right)d_r=0 \tag{20}$$

with the boundary conditions $d_r = 0$ at $r = 0$ and

$$\frac{\partial d_r}{\partial r}+\frac{\lambda}{\lambda+2\mu}\frac{d_r}{r}=-\frac{\sigma_0}{\lambda+2\mu} \tag{21}$$

at $r = r_0$. The resulting radial displacement denotes

$$d_r(r)=-l_L\cdot\frac{\sigma_0}{\lambda+2\mu}\cdot\frac{I_1\left(\frac{r}{l_L}\right)}{I_0\left(\frac{r_0}{l_L}\right)-\frac{2\mu}{\lambda+2\mu}\cdot\frac{l_L}{r_0}\cdot I_1\left(\frac{r_0}{l_L}\right)} \tag{22}$$

with modified Bessel functions of first kind $I_0$ and $I_1$. The total traction force $F_{tot}$ can then be calculated via the extension of the elastic connections representing the substrate. In total, this amounts to

$$\begin{aligned}F_{tot}&=Y\cdot\int_{A_0}|d_r(r)|\,dA=\int_0^{2\pi}d\varphi\int_0^{r_0}dr\,r\left(-d_r(r)\right)\\&=\frac{2\pi Y l_L\sigma_0}{\lambda+2\mu}\cdot\frac{\int_0^{r_0}dr\,rI_1\left(\frac{r}{l_L}\right)}{I_0\left(\frac{r_0}{l_L}\right)-\frac{2\mu}{\lambda+2\mu}\cdot\frac{l_L}{r_0}\cdot I_1\left(\frac{r_0}{l_L}\right)}\end{aligned} \tag{23}$$

Carrying out the integration one finds:

$$F_{tot}=\pi^2 r_0 h_c\sigma_0\frac{I_1\left(\frac{r_0}{l_L}\right)L_0\left(\frac{r_0}{l_L}\right)-I_0\left(\frac{r_0}{l_L}\right)L_1\left(\frac{r_0}{l_L}\right)}{I_0\left(\frac{r_0}{l_L}\right)-(1-\nu_c)\cdot\frac{l_L}{r_0}\cdot I_1\left(\frac{r_0}{l_L}\right)} \tag{24}$$

where $L_n(x)$ is the modified Struve function. One has, $L_0(0) = L_1(0) = 0$, hence for soft substrates where $l_L \longrightarrow \left(\frac{h_cE_c}{E_s/\tilde{h}_s}\right)^{1/2} \longrightarrow \infty$, $F_{tot} \to 0$ as expected. For stiff substrates, $E_s \gg N_ak_a\tilde{h}_s, l_L$ decreases to a minimal value, $l_L^{min} = \left(\frac{h_cE_c}{N_ak_a}\right)^{1/2}$, and $F_{tot}$ saturates at its maximum value (as given by Eq. 24 with the replacement of $l_L$ by $l_L^{min}$). The rate at which the total force increases towards the saturation value is dictated by two distinct factors, $N_ak_a$, and $h_cE_c$. As shown in Fig. 3D in the main text, the smaller these factors are, the more quickly $F_{tot}$ reaches its saturation value as $E_s$ is increased.

To gain insight into the functional dependence of $F_{tot}$ on $r_0$ we use the two asymptotic trends of the Bessel functions, $I_n(x)$. For $x \gg n$ one has, $I_n(x) \approx \exp[x]/\sqrt{2\pi x}$. Using this approximation in Eq. 23 and expanding to first order in $l_L/r_0$ we find:

$$\frac{F_{tot}}{2\pi r_0}\approx\sigma_0 h_c\left[1+\left(\frac{1}{2}-\nu_c\right)\frac{l_L}{r_0}+O\left[\left(\frac{l_L}{r_0}\right)^2\right]\right] \tag{25}$$

Thus, for well spread cells with $r_0 \gg l_L$ one finds: $F_{tot}/(2\pi r_0h_c) = \sigma_0$. This is consistent with the tractions being concentrated at the cell periphery and sustaining a uniform tensile stress, $\sigma_0$ in the cytoskeleton, as pointed out in Ref. (17).

In the opposite limit, $x \ll n$, the Bessel functions can be approximated as: $I_n(x) \approx \frac{1}{n!}\left(\frac{x}{2}\right)^n$, and in particular $I_0(x) \approx 1$ and $I_1(x) \approx \frac{x}{2}$. Consequently, in small cells where $r_0 \ll l_L$ the radial tractions underneath the cell $T_r(r) = Yd_r(r)$ increase linearly with the radial distance, $r$, as predicted by Eq. 22. With this traction profile one finds the following scaling of the total force:

$$\frac{F_{tot}}{2\pi r_0} \approx \frac{\sigma_0 h_c}{2(1+\nu_c)}\left(\frac{r_0}{l_L}\right)^2. \qquad (26)$$

Thus, as shown in Fig. S9, plotting the maximum value of the scaled total force, $F_{tot}^{max}/(2\pi r_0)$, as a function of the cell radius, may principally reveal two cell characteristics of interest. In the limit of large cell radii the curve tends to saturate at an asymptotic value $\sigma_0 h_c$ reflecting the uniform tension in the cytoskeleton. In practice we see that this saturation level is reached at cell radii that are beyond a typical platelet size. Moreover, from the scatter plot we cannot rule out the possibility that $\sigma_0$ is a function of the cell radius itself. In the limit of small $r_0$, the rate of increase of $F_{tot}^{max}/(2\pi r_0)$, with $r_0$ is dictated by $1/l_L^2$, which for rigid substrates, is given by, $\frac{h_c E_c}{N_a k_a}$ - the ratio of the cytoskeleton to adhesion-layer stiffness.

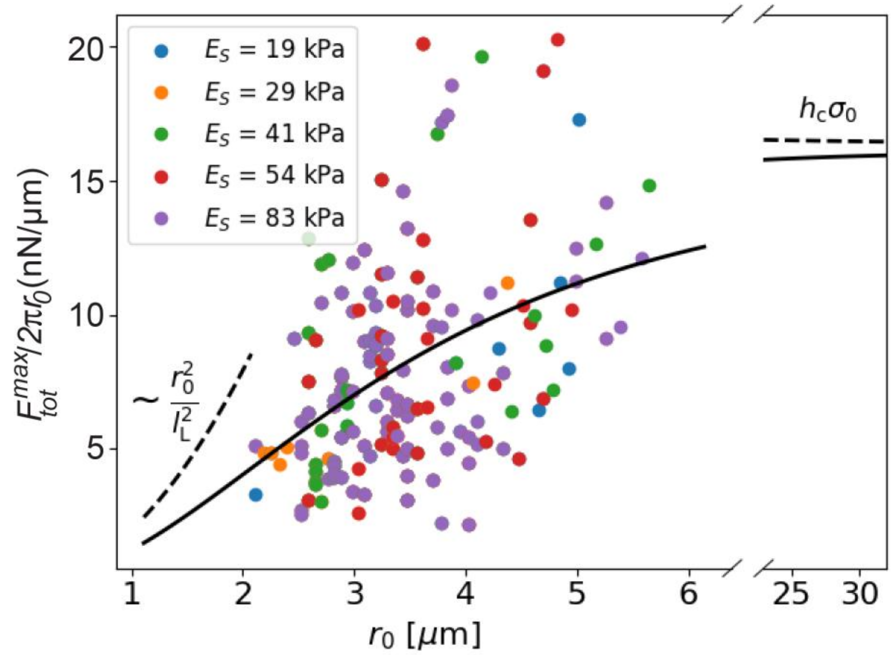

**FIGURE S7: Scaled total force as a function of the platelet radius.** The solid line provides a best fit of Eq. 24 to the data. The dashed lines correspond to Eqs. 25 and 26 and show the respective asymptotes to the mean cell behavior. As noted in the text, the shape of the $F_{tot}^{max}/(2\pi r_0)$ curve is bounded by two platelet characteristics: the saturation level of the scaled force is dictated by $h_c\sigma_{0c}$, the generated platelet contractility, and the initial slope of the curve is dictated by $\frac{h_c E_c}{N_a k_a}$, the ratio of the cytoskeleton to adhesion layer stiffness.

## 5.1 Cell and Substrate Parameters for the Model

**TABLE S2 Parameters used in the calculation of the mathematical model**

| **Quantity** | **Parameter** | **Value** |
|---|---|---|
| *Free fit parameters:* | | |
| Localization length | $l_L$ | Determined using conjugate gradients |
| Active contractile stress | $\sigma_0$ | Determined using conjugate gradients |
| *Fixed parameters:* | | |
| Platelet stiffness (17) | $E_c$ | 5 kPa |
| Cell Poisson ratio (18) | $v_c$ | 0.3 |
| Thickness of the platelet (19) | $h_c$ | 100 nm |
| Typical platelet radius | $r_0$ | 5 μm |
| Adhesion layer stiffness density | $N_a k_a$ | 0.3 nN/μm$^3$ |
| Elastic modulus of the substrate | $E_s$ | 19 - 83 kPa |
| Thickness of substrate | $h_s$ | 50 μm |
| Poisson’s ratio of the substrate (20) | $v_s$ | 0.3 |

The typical platelet size was estimated from the movies. The adhesion layer stiffness density was estimated from the expression for the localization length, given our assumed value for $E_c$, $h_c$ and $v_c$. The substrate stiffness and thickness were determined experimentally.

## 6. Simulation of force patterns

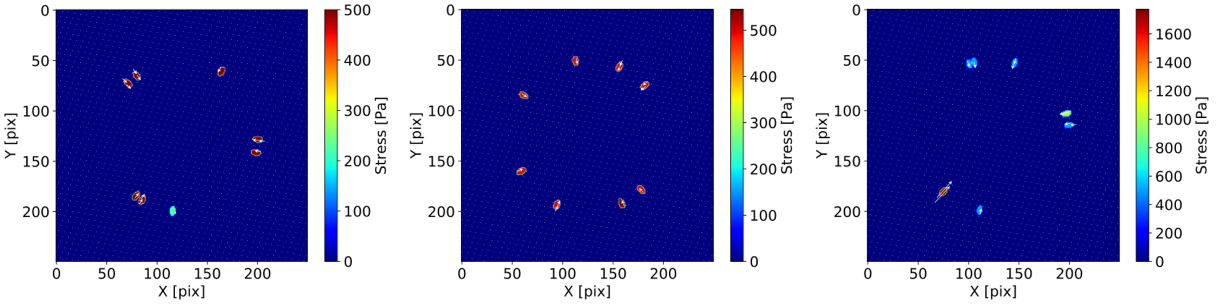

**Figure S8: Typical simulated traction patterns used to reproduce the force anisotropy in Figure 4C and 4D.** The simulated traction patterns consist of 8 stress hot spots which are placed on the same circle line to account for the peripheral location of traction hot spots in the experiments. Further, all force vectors are pointing towards the center of the image. The positions of the first 7 hot spots, each of which has the same stress magnitude, are sampled from a uniform distribution. The last hot spot is placed at the opposite side of the average traction spot position. Its stress magnitude is chosen such that the overall net force becomes zero.

## References

1. Guizar-Sicairos, M., S. T. Thurman, and J. R. Fienup. 2008. Efficient subpixel image registration algorithms. Opt. Lett. 33:156.
2. Dembo, M., and Y.-L. Wang. 1999. Stresses at the Cell-to-Substrate Interface during Locomotion of Fibroblasts. Biophys. J. 76:2307–2316.
3. Sabass, B., M. L. Gardel, C. M. Waterman, and U. S. Schwarz. 2008. High Resolution Traction Force Microscopy Based on Experimental and Computational Advances. Biophys. J. 94:207–220.
4. Munevar, S., Y.-L. Wang, and M. Dembo. 2001. Traction Force Microscopy of Migrating Normal and H-ras Transformed 3T3 Fibroblasts. Biophys. J. 80:1744–1757.
5. Delanoë-Ayari, H., S. Iwaya, Y. T. Maeda, J. Inose, C. Rivière, M. Sano, and J.-P. Rieu. 2008. Changes in the magnitude and distribution of forces at different dictyostelium developmental stages. Cell Motil. Cytoskeleton 65:314–331.
6. Plotnikov, S. V., B. Sabass, U. S. Schwarz, and C. M. Waterman. 2014. High-resolution traction force microscopy. Methods Cell Biol. 123:367–394.
7. Schwarz Henriques, S., R. Sandmann, A. Strate, and S. Köster. 2012. Force field evolution during human blood platelet activation. J. Cell Sci. 125:3914–3920.
8. Holenstein, C. N., U. Silvan, and J. G. Snedeker. 2017. High-resolution traction force microscopy on small focal adhesions - improved accuracy through optimal marker distribution and optical flow tracking. Sci. Rep. 7:41633.
9. Tomasi, C., and T. Kanade. 1991. Detection and Tracking of Point Features, Carnegie Mellon University, Pittsburgh, PA.
10. Shi, J., and C. Tomasi. June 1994. Good Features to track. *In* IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600.

11. Lucas, B. D., and T. Kanade. 1981. An Iterative Image Registration Technique with an Application to Stereo Vision. *In* Intl. Joint Conf. on Artificial Intelligence, Vancouver, British Columbia.

12. Bouguet, J.-Y. 2001. Pyramidal Implementation of the Affine Lucas Kanade Feature Tracker. Description of the Algorithm. Intel Corporation 5.

13. Tukey, J. W. 1967. An introduction to the calculations of numerical spectrum analysis. Wiley, New York.

14. Lukas, M. A. 2008. Strong robust generalized cross-validation for choosing the regularization parameter. Inverse Probl. 24:34006.

15. Edwards, C. M., and U. S. Schwarz. 2011. Force localization in contracting cell layers. Phys. Rev. Lett. 107:128101.

16. Banerjee, S., and M. C. Marchetti. 2012. Contractile stresses in cohesive cell layers on finite-thickness substrates. Phys. Rev. Lett. 109:108101.

17. Lam, W. A., O. Chaudhuri, A. Crow, K. D. Webster, T.-D. Li, A. Kita, J. Huang, and D. A. Fletcher. 2010. Mechanics and contraction dynamics of single platelets and implications for clot stiffening. Nat. Mater. 10:61–66.

18. Trickey, W. R., F. P. T. Baaijens, T. A. Laursen, L. G. Alexopoulos, and F. Guilak. 2006. Determination of the Poisson’s ratio of the cell. Recovery properties of chondrocytes after release from complete micropipette aspiration. Journal of biomechanics 39:78–87.

19. Aquino, D., A. Schönle, C. Geisler, C. V. Middendorff, C. A. Wurm, Y. Okamura, T. Lang, S. W. Hell, and A. Egner. 2011. Two-color nanoscopy of three-dimensional volumes by 4Pi detection of stochastically switched fluorophores. Nat. Methods 8:353–359.

20. Li, Y., Z. Hu, and C. Li. 1993. New method for measuring poisson’s ratio in polymer gels. J. Appl. Polym. Sci. 50:1107–1111.