USER MANUAL
Easy-to-use TFM Software

A Matlab package for regularized Fourier transform traction cytometry and Bayesian Fourier transform traction cytometry

2019, Sabass Lab

Table of Contents

I. Introduction

This software provides simple and robust methods to calculate traction forces in a TFM experiment. See References [1-5] for details on TFM and on traction calculation. The software is described in Reference [2]. Please cite us [1-3] if you use this program for your scientific work.

The TFM software is a collection of MATLAB functions that are called via intuitive menus in a graphical user interface.

Figure 1 illustrates the purpose of the TFM software, namely to calculate spatial maps of the cellular traction forces from measured substrate displacements. The calculations can be done either with **Regularized Fourier transform traction cytometry** [3] or with **Bayesian Fourier transform traction cytometry** [1-2]. The difference between the two methods is that data smoothing is either done manually or automatically:

Regularization → manual selection of a regularization parameter

Bayesian Regularization → automatic selection of a regularization parameter

To calculate the traction forces, the following experimental data must be provided:

1. A list of two-dimensional substrate displacements [pixel]. The displacements can be measured, e.g., by tracking the motion of fluorescent marker beads in the substrate.
2. Optionally a sequence cell images in .tif or .jpg format corresponding to the time points at which the displacements were measured.

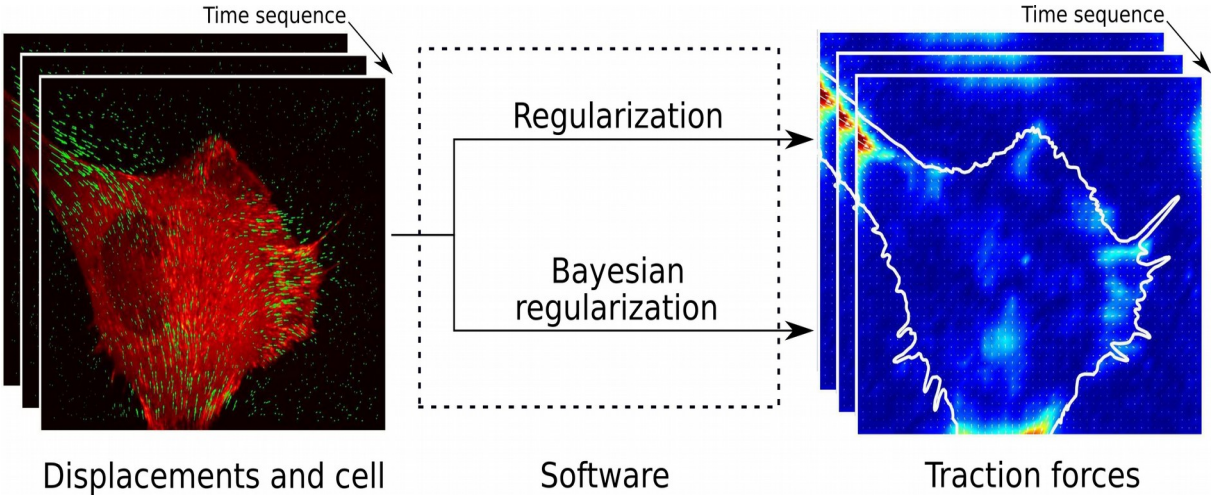


Figure 1. The software provides two methods to calculate cellular traction forces from experimental data

II. Installation

Requirements

This software runs in a MATLAB environment. Therefore, you need a recent copy of MATLAB (https://ch.mathworks.com/), as well as the Image Processing Toolbox (https://ch.mathworks.com/products/image.html). The package has been developed with MATLAB R2017b and should work for versions R2010b and above. It can be run on Windows, Linux, and Macintosh OS X operating systems.

Obtaining the TFM software

You can download this TFM software package for free from Github: https://github.com/CellMicroMechanics. The software will come in the form of a single compressed Easy-to-use_TFM_package-master.zip archive. You can unzip the archive into a folder (**Easy-to-use_TFM_package-master**) on your computer. The folder includes the program functions and a folder **test_data**, which contains exemplary data that can be used to get acquainted with the software.

III. Quick start

Changing into the software folder

To run the TFM software in the MATLAB console, you first need to change to the folder containing the TFM software. As shown in Figure 2, one can click on the 'Browse for folder' icon and switch to the folder: "Easy-to-use_TFM_package-master".

Alternatively, you also can use a command line input, for example:

>> cd /Data/some_folder_here/Easy-to-use_TFM_package-master

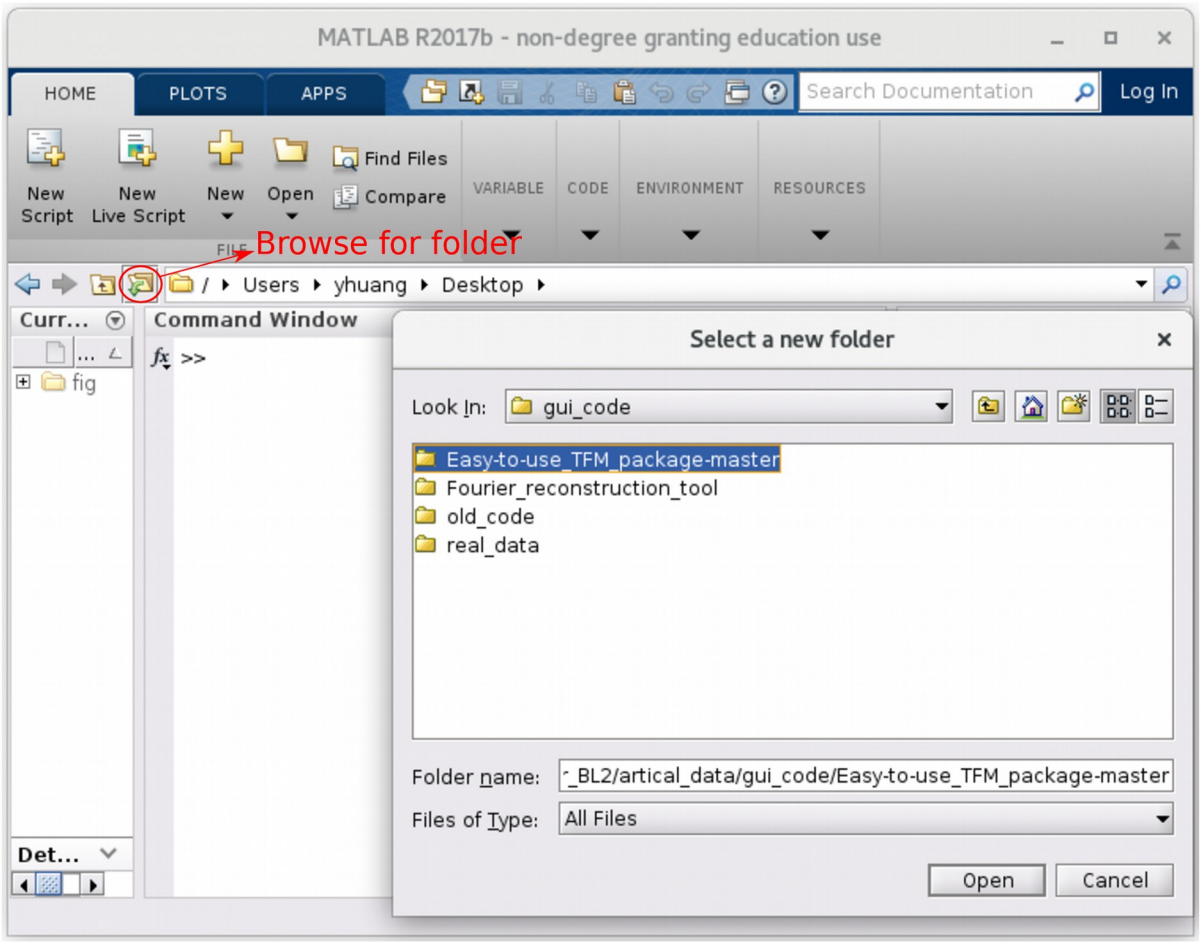


Figure 2. Changing the current folder in MATLAB.

Running the TFM software

To start the TFM software, enter

>> TF_reconstruction

into the command line as shown in Figure 3. Next, a window titled ‘get_data’ will appear and allow you to enter general input data as shown in Figure 4.

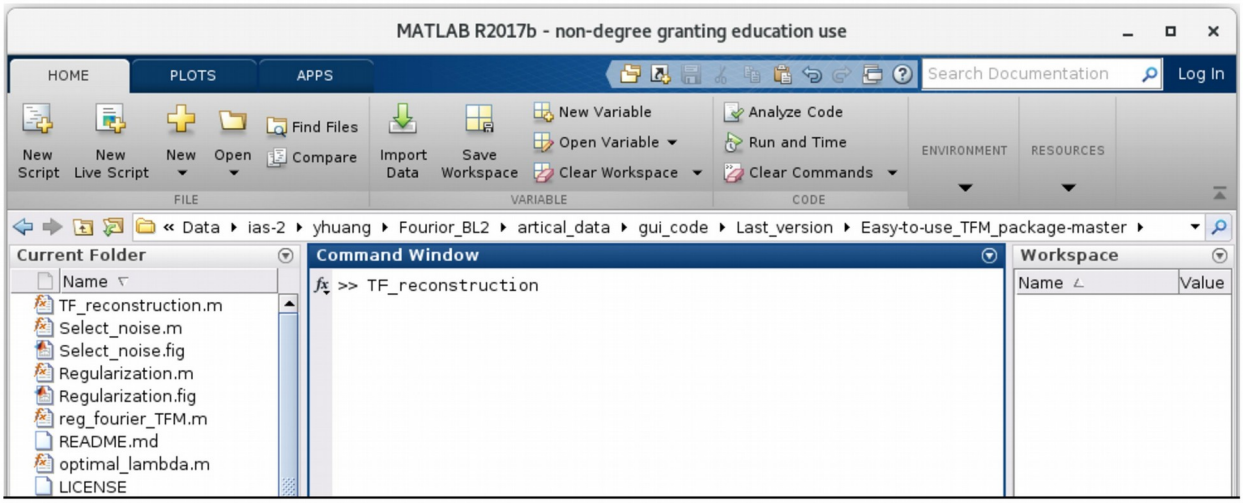


Figure 3. Starting the TFM software.

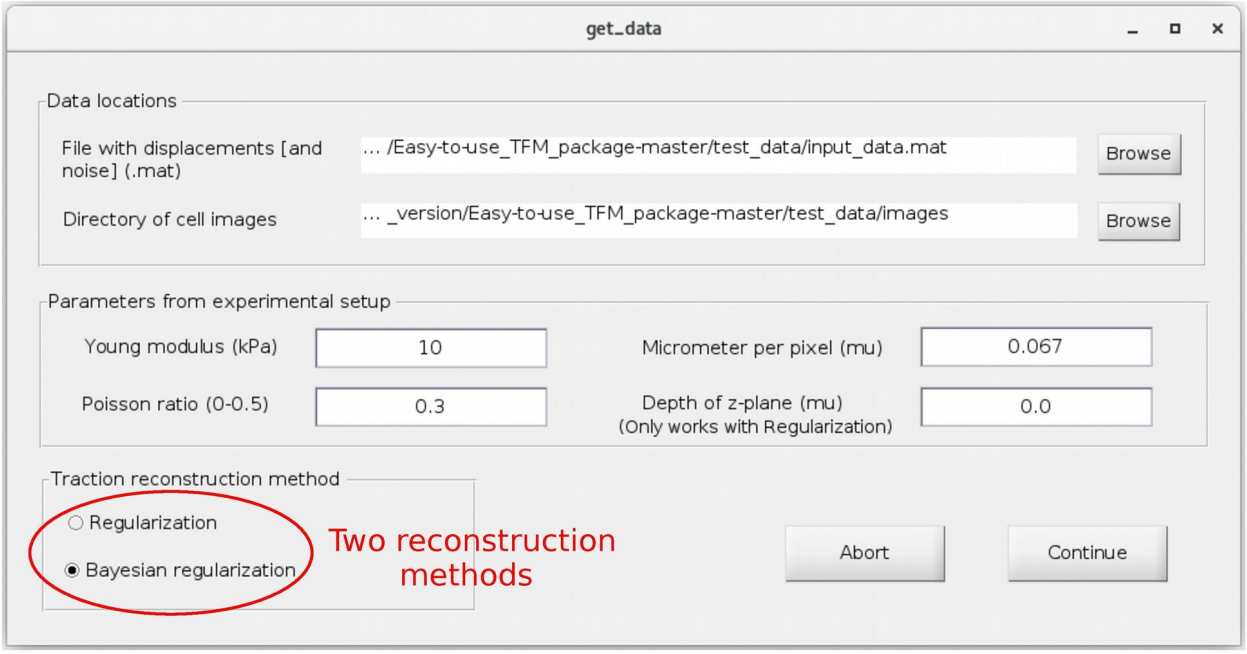


Figure 4. Providing general input data in the window “get_data”.

Setting general input data

Prior to traction calculation, some input data must be provided and a method for traction calculation must be chosen. This is done in the window “get_data”.

- **An input file** containing measured gel displacements is required. Optionally, a sample of the measurement noise can also be provided for use with Bayesian regularization. The input file must contain a MATLAB structure with a defined format that is shown in Figure 5.

  ➔ The fields **input_data.displacement(frame).pos(n,1:2)** contains **the x,y position** of the n-th displacement [pix]. The index ‘frame’ refers to the current dataset number in a movie.

  ➔ the fields **input_data.displacement(frame).vec(n,1:2)** contains the **x,y vector components** of the n-th displacement [pix]. The index ‘frame’ refers to the current dataset number in a movie.

  ➔ A sample of displacements resulting only from measurement noise can be provided in the same format in the structure **input_data.noise.**

  An exemplary input file is provided with **‘input_data.mat’** in the ‘test_data’ folder.

- **A directory containing cell images** in .tif or .jpg format (optional). Such images are of great help to assess the validity of traction calculations.

- **The Young’s modulus** must be provided in units of kilopascals [1000 N/m²]. A comparison of your different data sets will only be possible if this value is provided correctly.

- **The Poisson ratio** is a dimensionless quantity in the range from 0 to 0.5. Usually, the Poisson ratio is close to 0.5.

- **The input ‘Micrometer per pixel’** is the size of each image pixel in micrometers. This quantity is determined by the camera system of the employed microscope. This input is required for the calculation of the strain energies.

- **The input 'Depth of z-plane'** can be used to take into account the vertical offset between the image plane at which the displacements are measured and the surface of the gel. The offset has a positive sign when the image plane is below the gel surface. For example, 0.5 micrometers.

- One of the two methods 'Regularization' or 'Bayesian regularization' can be selected with option buttons.

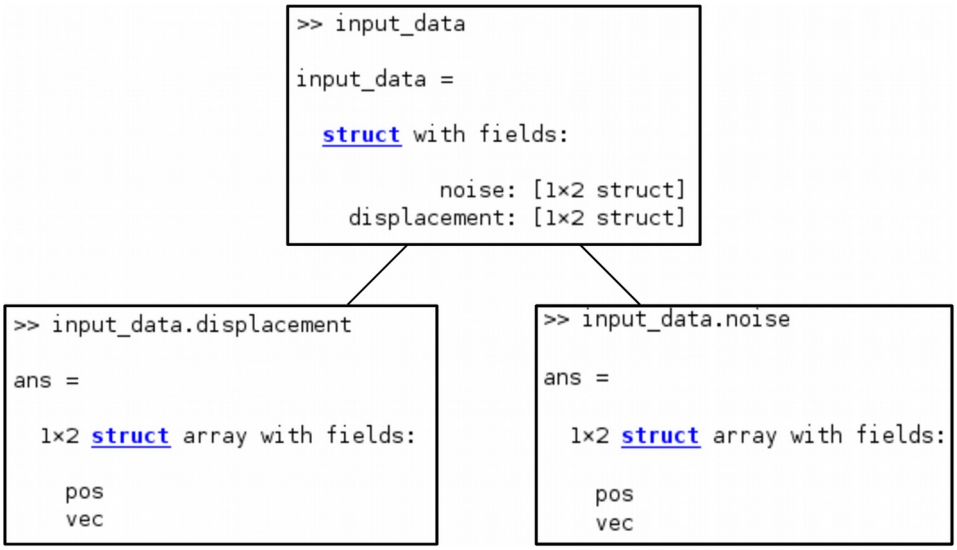


Figure 5. The structure of the input data.

Using the Regularization method

If "Regularization" is selected with the option buttons shown in Figure 4, the next window that appears after clicking "Continue" allows to one to perform traction calculations with the Regularized Fourier transform traction cytometry. The window is shown in Figure 6. The following inputs are required:

- Mesh size [pixel], for example, 10 pixels. Per default, a mesh size is chosen that closely matches the mean distance between the measured displacement vectors.

- The regularization parameter $\lambda_2 E^2$ [pixel²]. For example, 141.3 pixel^2. Increasing the regularization parameter, will produce a smoother traction field at the cost of reduced spatial resolution.

- To find an appropriate regularization parameter, you can preview the traction results. For image sequences, the preview can be done for different, individual frames by changing the entries in ‘dataset for preview’.

- Once an appropriate regularization parameter is chosen, the results for the whole sequence can be calculated by clicking on the ‘Analyze sequence’ button. The result is saved in a file with the name ‘Reg-FTTC_results_date(dd-mm-jj).mat’ (for example: Reg-FTTC_results_21-10-19.mat) in folder that contains the input data.

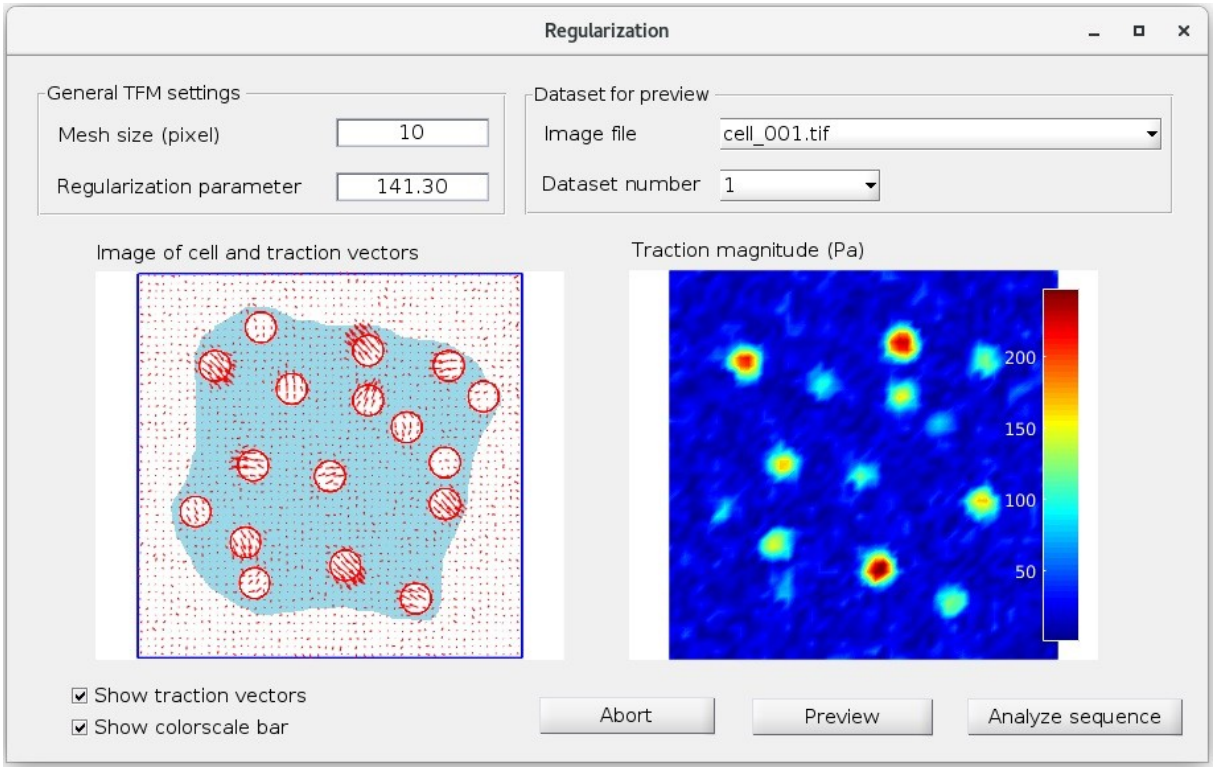


Figure 6. Calculating traction forces in the Regularization menu.

Using the Bayesian Regularization method

If "Bayesian regularization" is selected in the 'get_data' window as shown in Figure 4, the next window that appears after clicking "Continue" allows to one to perform traction calculations with Bayesian Fourier transform traction cytometry. With this method, the optimal smoothing regularization parameter $\lambda_2 E^2$ is determined automatically by maximizing the evidence. The numerically calculated evidence function and the optimal regularization parameter can be displayed together with a preview of the traction forces, see Figure 7. For this method, an estimate of the variance of the noise in the displacement data is required. The technical details of the procedure are described in References [1-2].

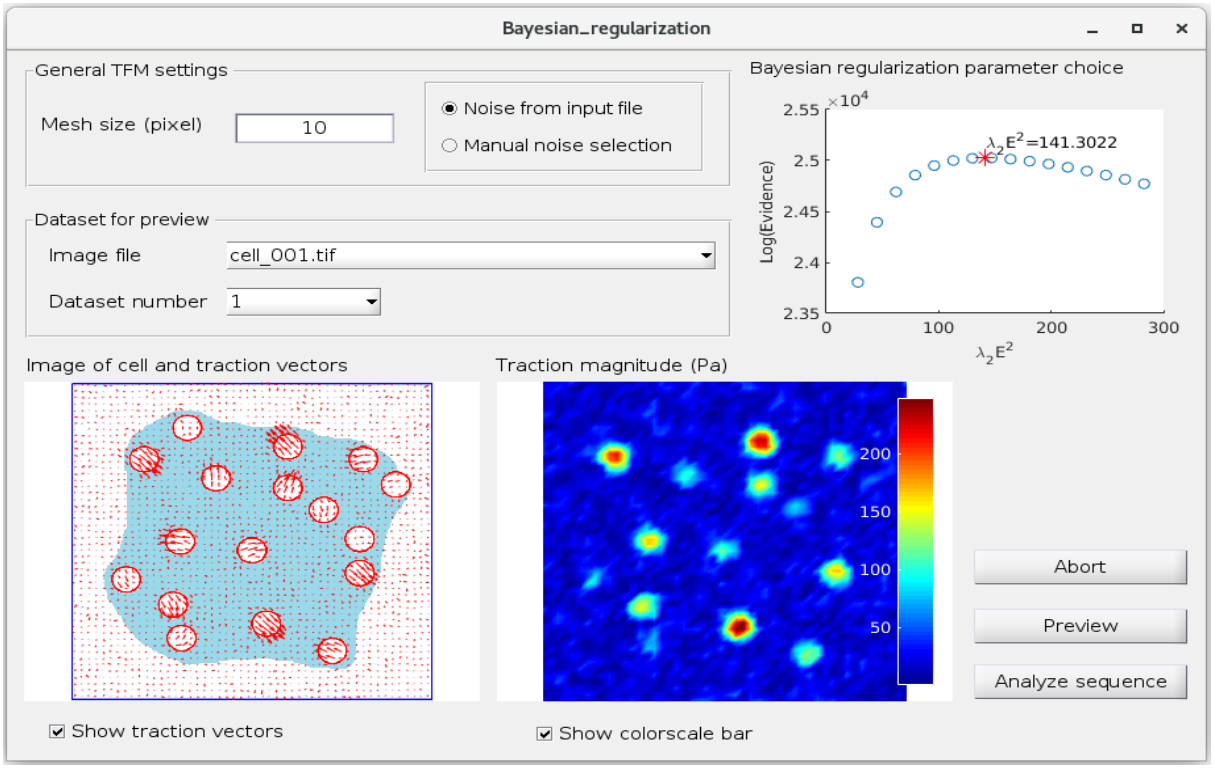


Figure 7. Calculating traction forces in the Bayesian regularization menu.

As illustrated in Figure 7, the following inputs are required:

- Mesh size [pixel], for example, 10 pixels. Per default, a mesh size is chosen that closely matches the mean distance between the measured displacement vectors.

- If your input data contains a sample of the displacement noise, you can select the radio button option ‘Noise from the input file’ as shown in Figure 7.

- If a sample of the displacement noise is not available, you can estimate the noise variance by choosing a region of interest (ROI) where the systematic displacement is small, e.g., far away from the cell. To do this, select the option ‘Manual noise selection’. A window as shown in Figure 8 will appear. Select your ROI and double click to finish. The program uses the last ROI that was chosen to create a separate noise sample for every dataset in a movie.

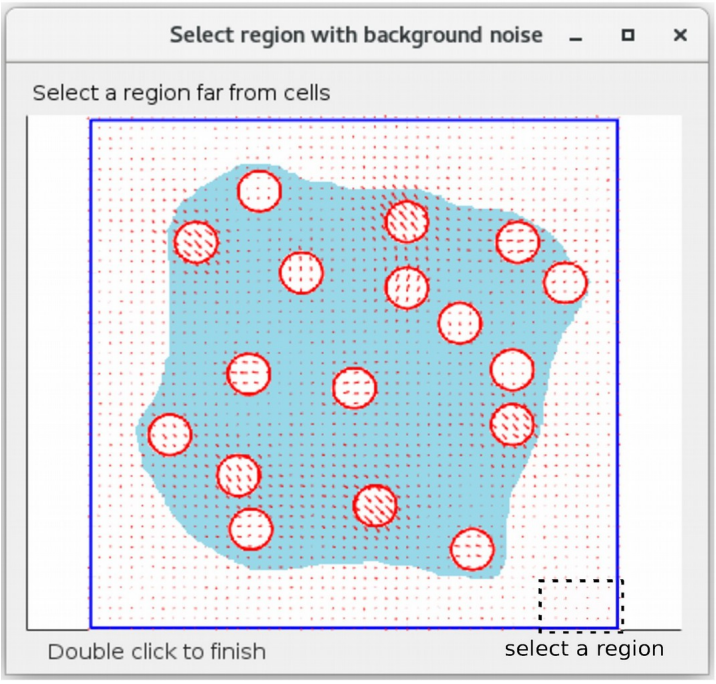


Figure 8. Manual selection of a region to estimate background noise.

- The results and the numerically calculated evidence for the regularization parameter can be previewed by selecting different frames and the corresponding image.

- To generate the traction results, click on the button ‘Analyze sequence’. For an image series, the whole calculation is done with only one setting for the mesh size. If ‘Manual noise selection’ is chosen, you will be asked to select a region for determining the noise variance once and the displacement noise will be estimated in this region separately for each frame in the series. The results of the traction force calculations are saved in a file ‘Bay-FTTC_results_date(dd-

mm-jj).mat' (for example, Bay-FTTC_results_21-10-19.mat) that is located in the folder containing the input data.

III. The program output

The results of the calculations are saved in the files ‘Reg-FTTC_results_date(dd-mm-jj).mat’ and ‘Bay-FTTC_results_date(dd-mm-jj).mat’ for Regularization and Bayesian regularization, respectively. The data format is illustrated in Figure 9.

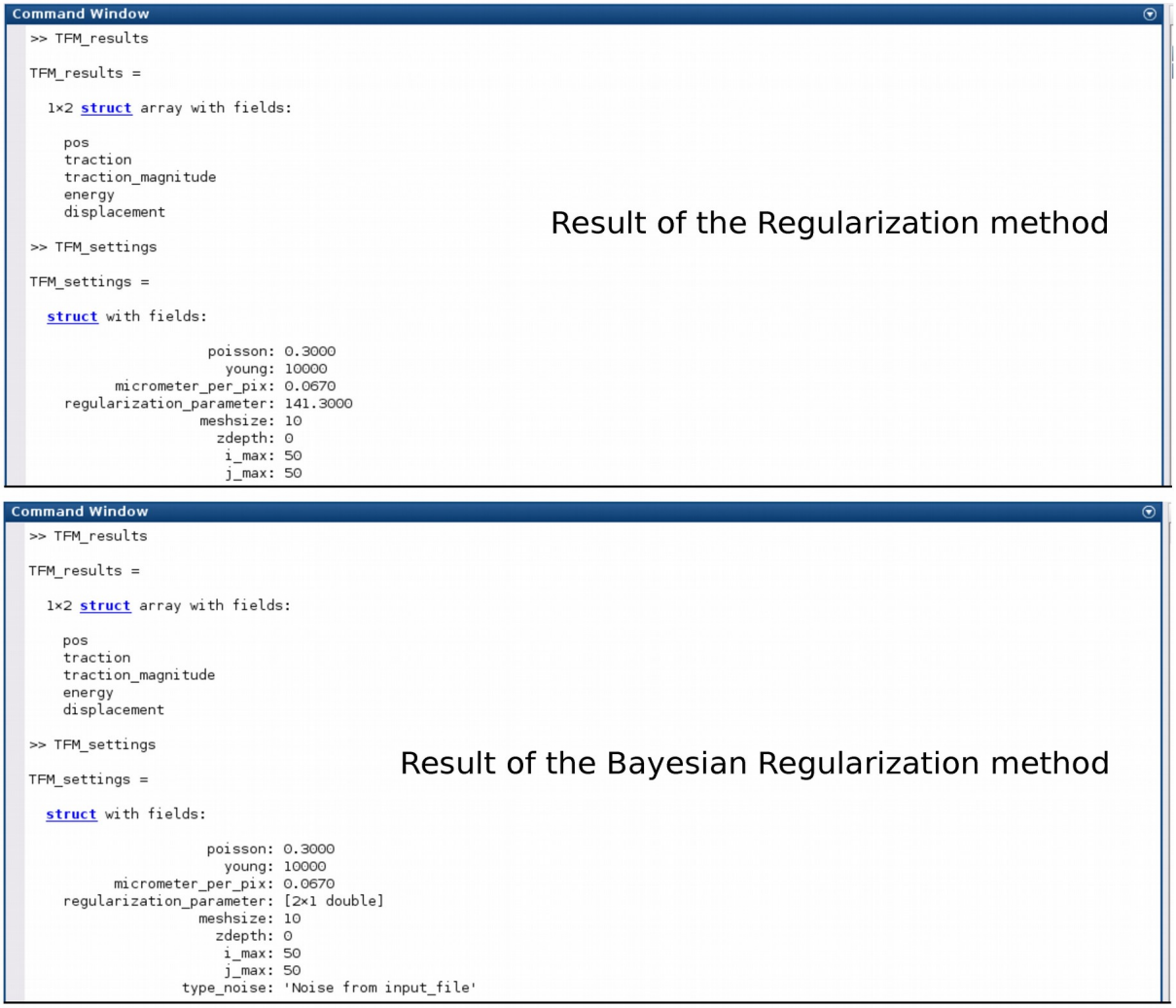


Figure 9. Structure of the stored results of the calculations for different methods.

The results of the traction calculations are saved in two MATLAB structures:

1. **TFM_results**
2. **TFM_settings**

Contents of the output structure TFM_results

- .pos contains the positions (x,y) of the displacement vectors, which are also the positions of the calculated traction vectors. Positions are given in pixels.

  Note: to account for the gel deformation, the positions in the output are shifted with respect to the displacements in the input data as:
  TFM_results.pos = displacement.pos+displacement.vec

- .displacement contains the displacement field interpolated on to a regular grid in units of pixels

- .traction contains the traction vectors (x,y) in units of Pascal [N/m^2]

- .traction_magnitude contains the 2-norms of each of the traction vectors

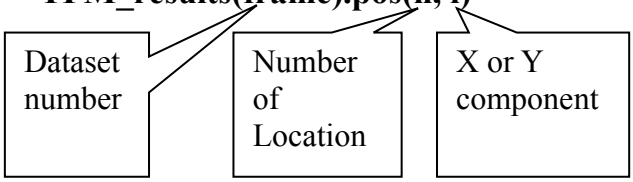
- .energy is the strain energy in pJoule (10^-12 J). This quantity can provide a feeling for 'how strong' the cell is.

  Note: the estimate of strain energy can be inaccurate. Sampling noise and aliasing effects usually affect the strain energy. Therefore, the outer rim of the field is cropped away to remove boundary artifacts in the strain energy. The width of the cropped area is determined by the variable "bnd", whose value can be changed in the code found in the files "Regularization.m" and "Bayesian_regularization.m".

Accessing the Vector fields (.pos, .vec, .force)

The organization of the vector fields is shown exemplary for .pos:

TFM_results(frame).pos(n, i)



For example, if you want to access the x- component of the 12-th traction vector in dataset (frame) 64 you write "**TFM_results(64).traction(12,1)**".

Accessing the scalar fields (.traction_magnitude, .energy)

These fields are organized like above, only lacking the x,y components.

TFM_results(frame).traction_magnitude(n)
TFM_results(frame).energy(n)

V. Using the program from the command line

This TFM package is built around MATLAB functions that can be called individually. This architecture allows users to employ our TFM methods in customized scripts. The main functions used for traction calculation are:

- **interp_vec2grid.m**
  Accepts irregularly spaced 2D displacement data as an argument and interpolates the data onto a regular, rectangular grid for traction calculation

- **reg_fourier_TFM.m**
  Calculation of traction forces

- **fourier_X_u.m**
  Preparation of data for Bayesian TFM using optimal_lambda.m

- **optimal_lambda.m**
  Calculation of optimal regularization parameter in Bayesian TFM

A description of function arguments and outputs is provided in the function files. To illustrate how to use the program from the MATLAB command line, we provide a short example that can be found in "Easy-to-use_TFM_script_example.m".

V. References

[1] Y. Huang, C. Schell, T. B. Huber, A. N. Simsek, N. Hersch, R. Merkel, G. Gompper, B. Sabass. Sci. Rep., 2019, vol. 9, pp. 1–16.
[2] Y. Huang, G. Gompper, B. Sabass. submitted.
[3] B. Sabass, M. L. Gardel, C. M. Waterman, U. S. Schwarz. Biophys. J., 2008, vol. 94, pp. 207–220.
[4] S. Munevar, Y.-L. Wang, M. Dembo. Biophys. J., 2001, vol. 80, pp. 1744-1757.
[5] J. P. Butler, I. Tolic-Nørrelykke, B. Fabry, J. J. Fredberg. Am. J. Physiol. Cell Physiol., 2002, vol. 282(3), pp. C595-C605.