A quick guide to the program *frap_analysis*

frap_analysis is a program implemented in MATLAB® (The MathWorks™, Natick, MA) to analyze data from photobleaching experiments on a supported lipid bilayer. The theory behind the program is presented in the article "*A Method Improving the Accuracy of Fluorescence Recovery After Photobleaching Analysis*" by Jönsson et al.[1]. All code was developed and tested using MATLAB R2007b. The program analyzes a sequence of images stored as a .tif file (typically a 16-bit grayscale image). At least one pre-bleach image should be included in the .tif file to compensate for uneven illumination over the sample (in case that no pre-bleach image has been included it is also possible to assume a flat illumination profile). Furthermore, the bleached spot in the first post-bleach image should not extend outside of the field of view of the image. Additional information that is required to continue the analysis is the time between the post-bleach images and the size of a pixel in the images (square pixels are assumed). The basic steps in the analysis are presented below.

1) The program is started from the MATLAB command window by typing *frap_analysis*. A window entitled "frap_analysis" is now shown.

2) The .tif file containing all images is opened from the *File* menu (submenu *Open*). The program will now look like in Fig. A1.

3) Each frame of the image stack can be selected by moving the slider under the displayed image. The intensity range shown can be changed by moving the two sliders next to the *Imin* and *Imax* labels. Pushing the *Auto* button sets the intensity range in the displayed image to be between the minimum and maximum intensity in the current image.

4) *First and last pre-bleach frame*. All images between the first and the last pre-bleach frame will be averaged by the program yielding a single pre-bleach image. Frame numbers can either be written directly into the edit fields or can be chosen as the currently displayed image frame by pushing the get button next to each field. It is also possible to not use a pre-bleach image and assume that the illumination is flat. This is done by unchecking *Use pre-bleach frame*. The entry "First pre-bleach frame:" will then change to "Pre-bleach intensity". The user can now either input the pre-bleach intensity (in counts) or use the mean intensity inside a user defined region in the current image (by pushing *Get*). However, if the illumination is not flat then omitting a pre-bleach image will lead to an uncertain analysis.

[1] P. Jonsson, M. P. Jonsson, J. O. Tegenfeldt, F. Hook, *Biophys. J.* **95**, 5334-5348 (2008).

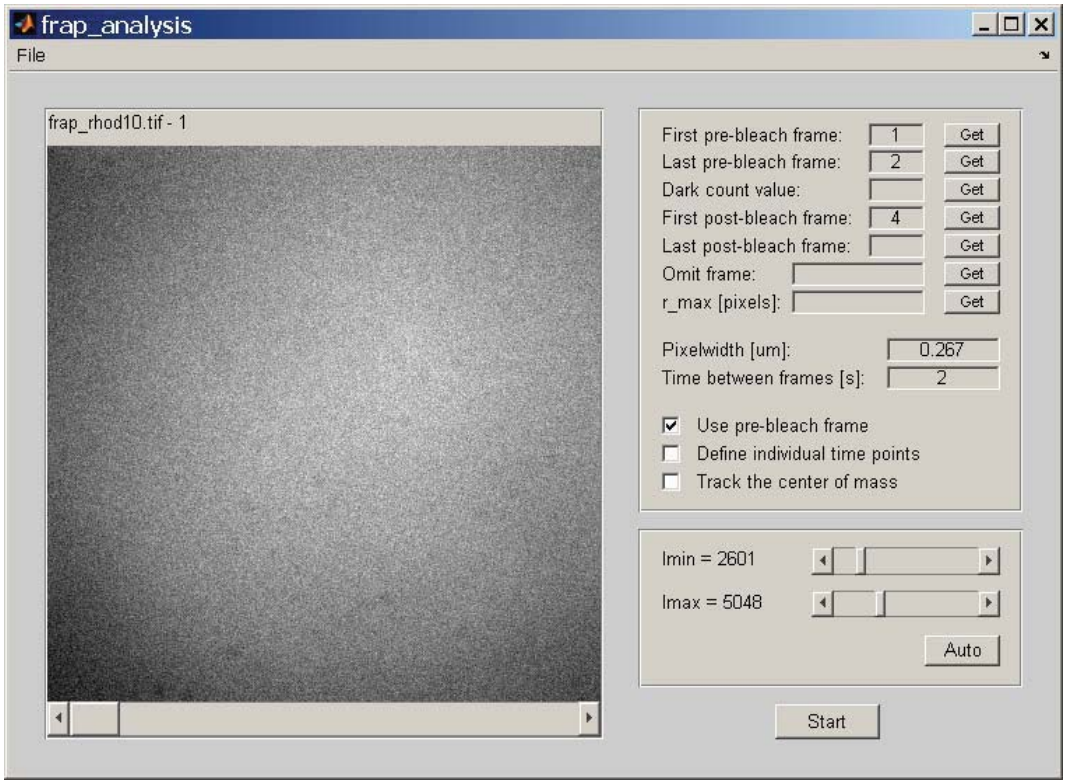

Figure A1. The state of the program after opening the image file to be analyzed.

5) *Dark count value*. The value specified in this field will be subtracted from all images. The dark count intensity can either be given directly in the edit field or be obtained from an image with, e.g. the light on the sample turned off. In the later case the user should push the *Get* button and then draw a region inside which the average dark count value will be determined. The region is drawn from point to point and closed by pushing the right mouse button and double clicking inside the region. If this field is left unassigned then the dark count intensity is set to zero.

6) *First and last post-bleach frame*. The frame number of the first and last post-bleach image that are to be used in the analysis will be set here. If the last post-bleach frame is left unassigned then all images after the first post-bleach frame are used for the analysis. Frame numbers can either be written directly into the fields or can be chosen as the currently displayed image frame by pushing the *Get* button next to each field.

7) *Omit frame*. Image frames that are to be omitted from the analysis, due to errors in the individual images, should be given here. Each image frame should be separated with a comma, which is done automatically if the user pushes the *Get* button to choose which frames that are to be omitted.

8) *r_max [pixels]*. Determines the largest radial value that is used when analyzing the data. If left unassigned the largest value of r is chosen to be the distance between the center of the bleached spot and the edge of the image. However, if the bleached spot is small it may be advisable to reduce the maximum value of r to avoid errors arising from intensity changes at the edge of the image. The maximum value of r can either be given directly or be obtained by pushing the *Get* button. For the latter case a dot appears at the center of the image, which can be drawn out to a line (and moved). Pushing the *Get* button again measures the length of the line (in pixels) and uses this as r_{max}.

9) *Pixelwidth*. The size of a pixel in μm. Square pixels are assumed.

10) *Frame times*. The time for each image frame can be given in two ways depending on whether the checkbox *Define individual time points* is checked or not. If not checked then all image frames are assumed to be separated with a constant time gap and the time between two frames should then be inserted. If the user chooses to use individual times (by checking the *Define individual time points* checkbox) the field "Time between frames" changes to "Frame times". The times for each image frame should now be given instead, where the individual times should be separated with a comma. The first time point will be set to zero and all the other times will be given relative to this value.

11) *Track the center of mass*. If this checkbox is checked then the center of mass will be updated for each frame in the image stack. Otherwise, only the first post-bleach image will be used to determine the center of mass of the bleached spot.

12) The analysis is going into the next phase by pushing the *Start* button.

13) In the next step the user draws a polygon around the region of the bleached spot (the polygon is closed by clicking the right mouse button). After double clicking the inner part of the polygon the program calculates the center of mass of the encircled region, yielding the center of the bleached spot. The program will now open a new window entitled "Fit the data" (see Fig. A2).

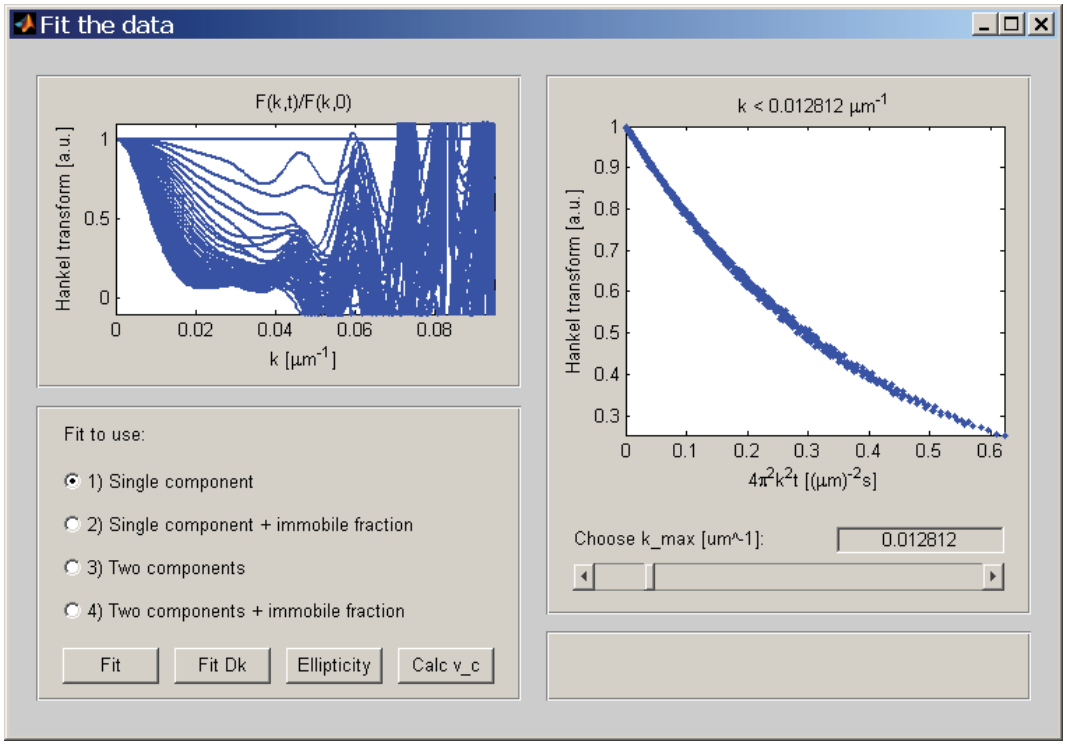

Figure A2. The window in which the curve fitting of the data is controlled.

14) The Hankel transform of the averaged radial data, $F(k,t)$, is given in the upper left corner of Fig. A2. The blue lines correspond to $F(k,t)/F(k,0)$ for each individual image, i.e. with t fixed.

15) The highest value of k that should be used in the analysis of the FRAP images is next chosen. As a rule of thumb k_{max} should be where the blue curves in the top left of Fig. A2 start to deviate from a Gaussian profile. An alternative way of determining a suitable k_{max} is to plot $F(k,t)/F(k,0)$ as a function of $4\pi^2k^2t$ (see the right plot in Fig. A2), which the program does automatically when choosing a new value for k_{max}. Figure A3 shows how $F(k,t)/F(k,0)$ looks for three different choices of k_{max}. If k_{max} is chosen too small (A) then it will be hard to determine the fraction of immobile fluorophores. If k_{max} on the other hand is chosen too high (C) then there will be excessive noise in the data making a curve fit less accurate. Thus, the proper choice of k_{max} is for the case depicted in (B) with k_{max} = 0.024 μm^{-1}. The user can choose different values of k_{max} before proceeding with the analysis. This can be done either by directly giving a value in the edit field or by moving the slider beneath this field.

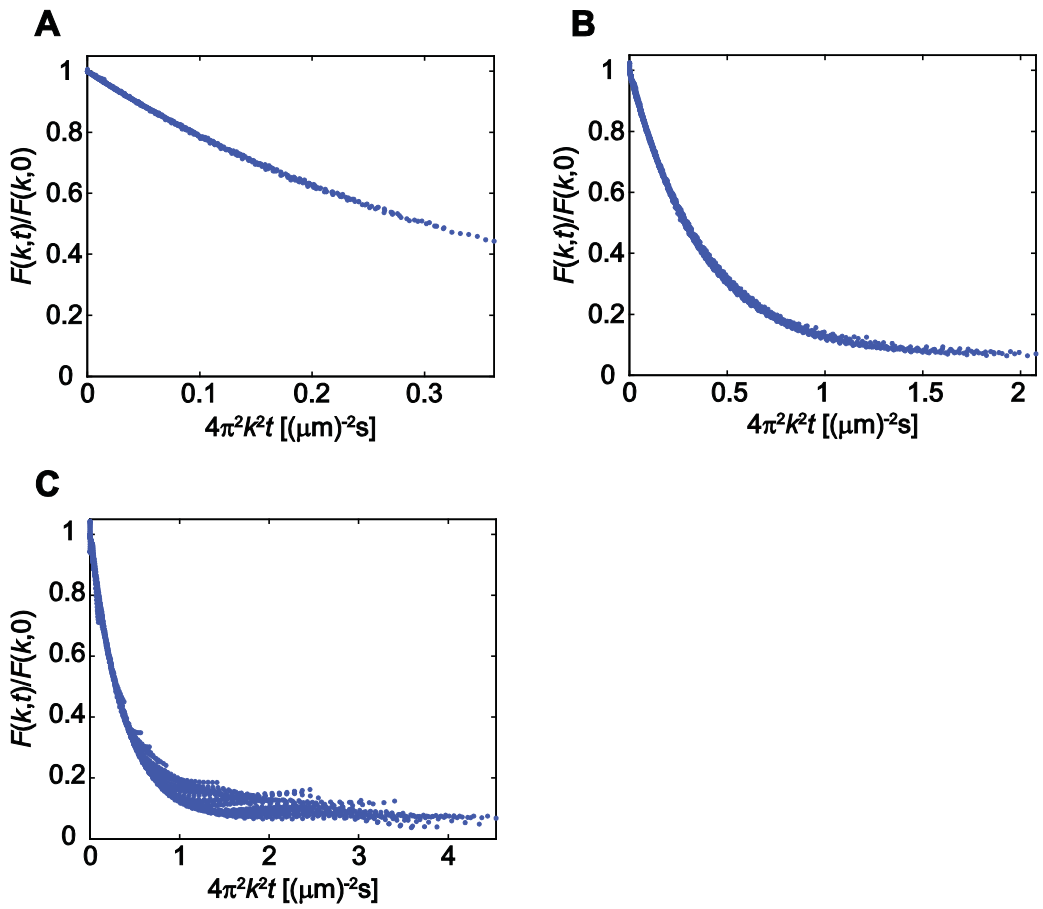

Figure A3. The normalized Hankel transform $F(k,t)/F(k,0)$ vs $4\pi^2k^2t$ for three different choices of k_{max}: (**A**) k_{max} = 0.01 μm^{-1}, (**B**) k_{max} = 0.024 μm^{-1} and (**C**) k_{max} = 0.045 μm^{-1}.

16) $F(k,t)$ is next fitted to an exponential function with the general appearance given in Eq. A1.

$$F(k,t) = F(k,0)\left[(1-\gamma_2-\gamma_0)\exp\left(-4\pi^2k^2D_1t\right)+\gamma_2\exp\left(-4\pi^2k^2D_2t\right)+\gamma_0\right] \quad \text{(A1)}$$

where D_1 and D_2 are two, potentially different, diffusion coefficients, γ_2 is the fraction of the second component and γ_0 is the fraction of immobile molecules. Four different types of curve fits may be chosen: 1) a single diffusing component with no immobile molecules ($\gamma_2 = \gamma_0 = D_2 = 0$), 2) a single diffusing component with an immobile fraction of molecules ($\gamma_2 = D_2 = 0$), 3) two diffusing components with no immobile molecules ($\gamma_0 = 0$) and 4) two diffusing components with an immobile fraction of molecules.

17) The user can choose which of the four fits to use with the radiobuttons in the lower left of Fig. A2. The curve fitting is started by pushing the *Fit* button. When fitting to case 4 the fitted result for case 1, 2 and 3 will also be included (for case 3 then case 1 and 2 will be included and so on). For case 2 to 4 the different curve fits are also compared with each other to determine which fit that is statistically the best. This decision is made on the basis of the squared sum difference between two curve fits to the squared sum error in the fit using the most parameters. If the squared sum difference is smaller than 0.5 times the squared sum error in the fit with the most parameters then the fit with the most parameters is not considered to yield a significant improvement when describing the data. Fig. A4 shows a snapshot of the state of the program after this step.

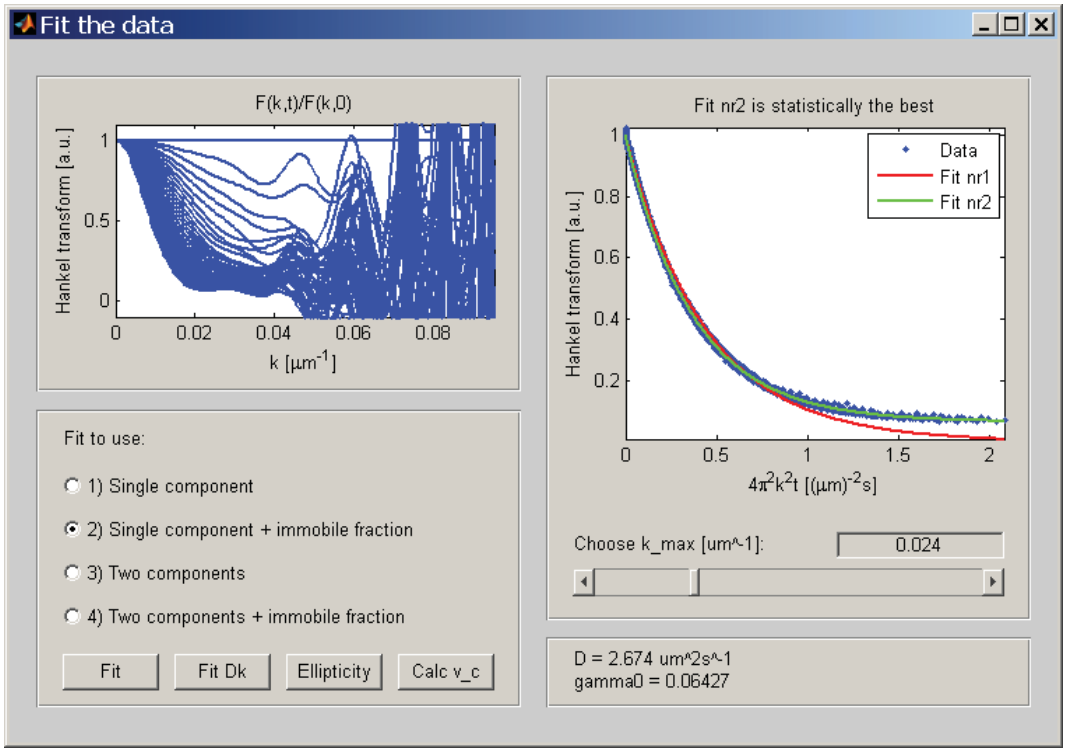

Figure A4. A snapshot of the program after fitting the data to a single diffusing component with an immobile fraction.

18) The fitted value for D, γ_0 etc. (depending on which fit that is chosen) will be given beneath the fitted curve (see Fig. A4). The data points together with the current fit is also given as a separate MATLAB figure, where the data can be saved in various formats.

19) The analysis can also be made with D_1 allowed to vary for each value of k (see Fig. A5). This is done by pushing the button *Fit Dk* after which both the values for $D_1(k)$ (A) and the curve fits (B) are given in separate MATLAB figures.

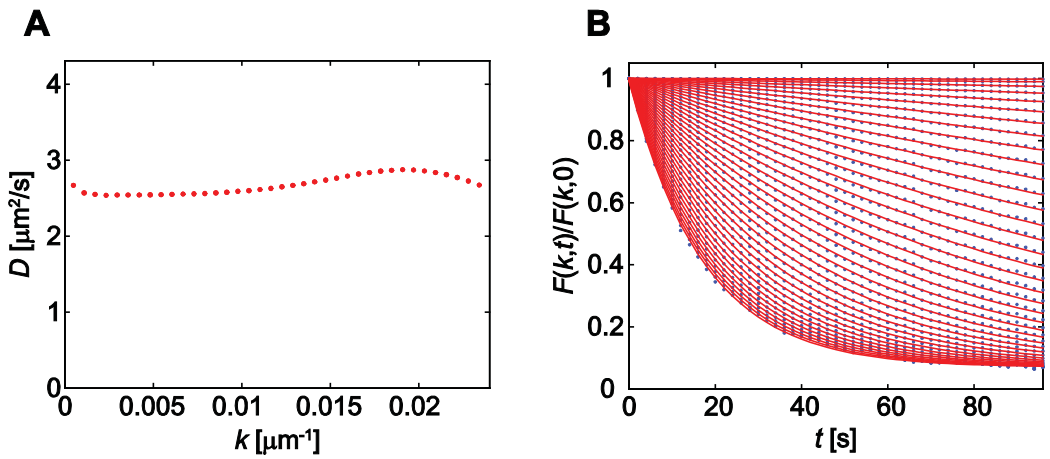

Figure A5. (**A**) Plot showing the value of $D(k)$ determined at each individual value of k. The other parameters in the fit (D_2, γ_2 and γ_0) are effective parameters independent of k. (**B**) Graph showing all the curve fits of $F(k,t)/F(k,0)$ vs t.

20) The *Ellipticity* button in Fig. A2 determines how circularly symmetric the bleached spot is for each frame. This is determined by calculating the maximum and minimum moments of inertia for the bleached spot defined as

$$I_{\min} = \int y'^2 dA \tag{A2}$$

$$I_{\max} = \int x'^2 dA \tag{A3}$$

where y' and x' is the minor and major semi-axis of the bleached spot, respectively. The quotient between $I_{\max}$ and $I_{\min}$ will be proportional to the square of the length of the major and minor semi-axis. Thus, for a circular profile $(I_{\max}/I_{\min})^{1/2} = 1$, while for a bleached ellipse with the major semi-axis twice as long as the minor semi-axis then $(I_{\max}/I_{\min})^{1/2} = 2$. The angle between the x'- and x-axis is also determined for each frame.

21) If the user has chosen to track the center of mass there will also be a button *Calc v_c* (see Fig. A2), which gives the position of the center of mass of the bleached spot at each frame. The program then calculates the mean velocity and direction of the bleached spot from the tracked position of the center of mass. The angle of movement is defined as zero moving to the right and 90° when moving towards the top of the image. Note that the tracking of the center of mass will be uncertain when the bleached spot approaches full recovery.