# STIC USER MANUAL – Tristan Wallis

**Last changed 20210618**

## Introduction

STIC = spatiotemporal indexing clustering, which represents the QBI Single Molecule Neuroscience lab's take on molecular clustering in live cells. STIC works by establishing whether entire molecular trajectories overlap in both space and time, and then building up clusters based on this potential overlap. STIC takes a different approach than widely used clustering algorithms which rely on DBSCAN or Voronoi tessellation to establish thresholds for molecular detections (Fig. 1). STIC also differs markedly from these algorithms in that it also determines whether molecules are interacting in time.

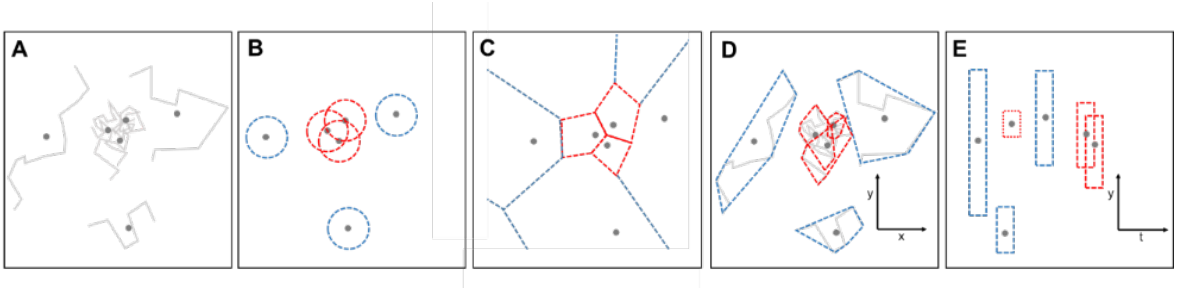



**Figure 1. Schematic representation of clustering algorithms. (A)** Molecular trajectory data, with each trajectory spatial centroid indicated with a dot. **(B)** DBSCAN. Multiple molecule centroids present within a defined radius are considered clustered (red circles). The most effective radius (**ε)** and minimum number of centroids within it (MinPts) are determined empirically. **(C)** Voronoi tessellation. Tiles are drawn around each centroid such that the distance from any point within the tile is closer to its centroid than to any other centroid. Molecular centroids with tile areas less than an empirically determined threshold (red) are considered clustered. **(D)** Spatial indexing. Clustered molecules are determined by overlapping 2D bounding regions (red) defining the spatial extent of each molecular trajectory. **(E)** Spatiotemporal indexing. Each trajectory bounding region is assigned an arbitrary "thickness" in the time dimension. Overlapping 3D bounding regions represent spatiotemporally clustered molecules.

STIC relies on the R-tree spatial indexing algorithm, which is widely used in databases, mapping and especially videogames, where it is used to establish whether in-game objects, such as bullets and bad guys, are interacting. R-tree and other -tree algorithms such as Quad-tree, Oct-tree etc are highly optimised, which allows large numbers of interactions to be calculated quickly so as to not slow the game down unduly. We leverage this ability to rapidly establish whether trajectory bounding boxes are overlapping.

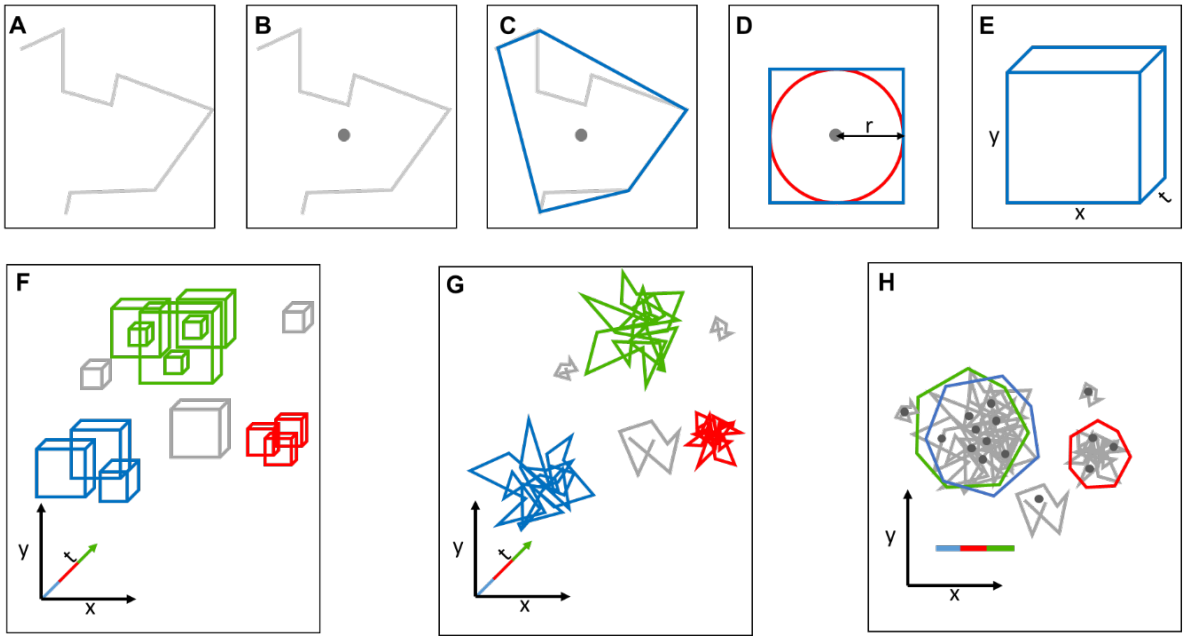



**Figure 2. Schematic representation of spatiotemporal indexing clustering workflow. (A)** Molecular trajectory composed of individual detections. (**B)** Spatiotemporal centroid representing the trajectory's average position in space and time. **(C)** Convex Hull (blue) defining the approximate spatial extent of the trajectory. **(D)** Simplified 2D spatial bounding box (blue square) based on the approximate radius (r) of the Convex Hull (red circle). **(E)** 3D spatiotemporal bounding box of user defined "thickness" in the time dimension. **(F)** R-tree spatiotemporal index of all trajectory bounding boxes. Discrete clusters of overlapping bounding boxes are indicated in colour, unclustered boxes in grey. **(G)** 3D clusters of trajectories associated with overlapping bounding boxes. **(H)** 2D representation of clustered trajectories. Coloured polygons represent the spatial Convex Hull of all detections comprising each of the clustered trajectories. Clusters are coloured according to the averaged detection time of their component trajectories, allowing assignment of overlapping clusters (green and blue) occupying the same spatial extent at different times.

STIC allows us to determine not only whether a trajectory overlaps with another trajectory in space (as do DBSCAN and Voronoi), but also whether the overlap occurs in time too. The idea being that trajectories which overlap spatiotemporally may represent clusters of molecules interacting at a certain space and time on the plasma membrane. Additionally, spatiotemporal metrics also allow us to determine whether clusters repeatedly form and reform on the same region of a plasma membrane – hotspots. The temporal component of STIC, literally gives us another dimension of molecular dynamics data to play with.

STIC was conceived, prototyped and converted into a Python GUI by Tristan Wallis 2020-2021. Sophie Hou assisted with aspects of GUI debugging. Neither of us are trained computer science graduates! Every effort has been made to create a functional, useful and stable program, but bugs and missing features are bound to crop up. Your input with bug reports and feature requests for both the program and this documentation is very important to help STIC mature.

STIC is released under a Creative Commons license. This is a very common open source license which means you are free to use or modify it, and give/sell it to others to use/modify, under the proviso that you or they a) don't release changed versions under a more restrictive license, and b) properly acknowledge the original author (me).

This manual should help you to become a world leader in spatiotemporal cluster analysis. In addition, STIC has popup help for most of its buttons and fields which should help you further. If in doubt, just contact me at t.wallis@uq.edu.au

## Computer requirements

STIC is a Python script and requires Python 3.8 or later to run. Python is available for most computer platforms so you can run STIC on Windows, Linux and Mac (although it is not very fast on Mac for some reason). STIC will not run on the older version of Python 2.7 which is still lingering on a lot of computer systems. You are strongly encouraged to either visit https://www.python.org and download and install the latest version (3.9.5 at the time of writing), or hassle someone in IT to do it for you. You/they will also need to install a number of Python modules, which STIC uses to do a lot of the heavy lifting. This is simple to do from a command line.

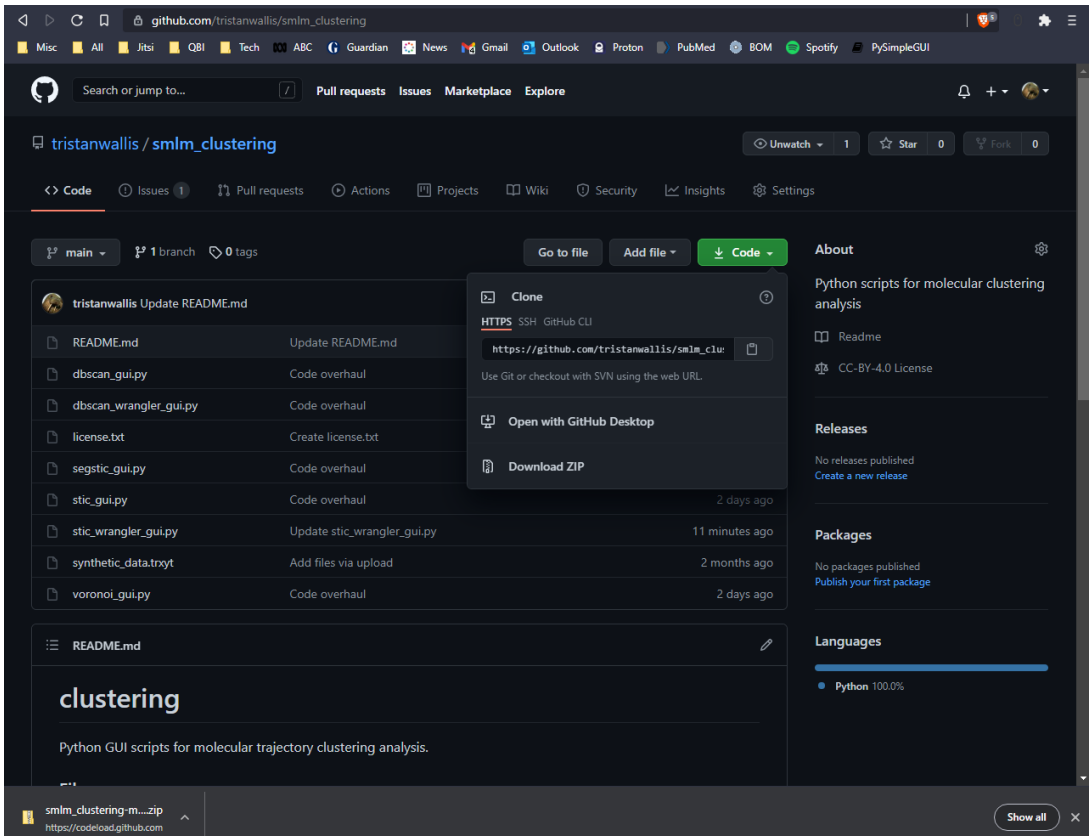
```
Microsoft Windows [Version 10.0.19042.1052]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>python -m pip install matplotlib numpy pandas pysimplegui rtree scipy scikit-learn seaborn
```

STIC is a multithreaded application which means that the intensive number crunching is farmed out to however many cores your computer has. You'll therefore theoretically get better performance on a 4Ghz computer with 8 cores than you will on a 5Ghz computer with 4 cores. But even on a crappy laptop such as mine most analyses should take less than a minute.

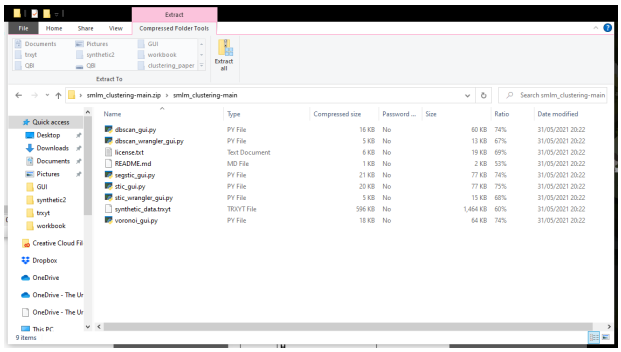
**IMPORTANT:** Because of the multithreading, STIC does not play properly on virtual computers which dynamically assign the user virtual cores depending on usage etc. You will need to run it on a physical computer.

## Obtaining STIC

The latest greatest version of STIC and related clustering software are always available on Github: https://github.com/tristanwallis/smlm_clustering. The total download is less than 1Mb so it only takes a few seconds to download and ensure that you're not running a superseded version. Just click on the green "Code" button and then "Download zip".



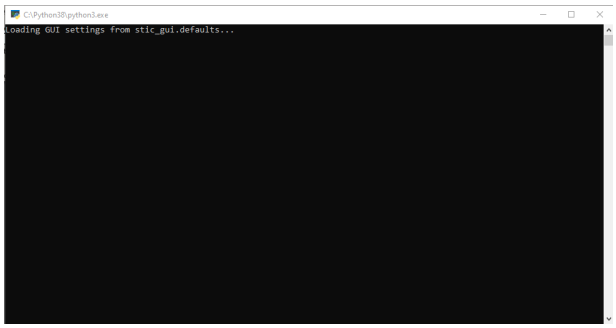

This will then download smlm_clustering_main.zip

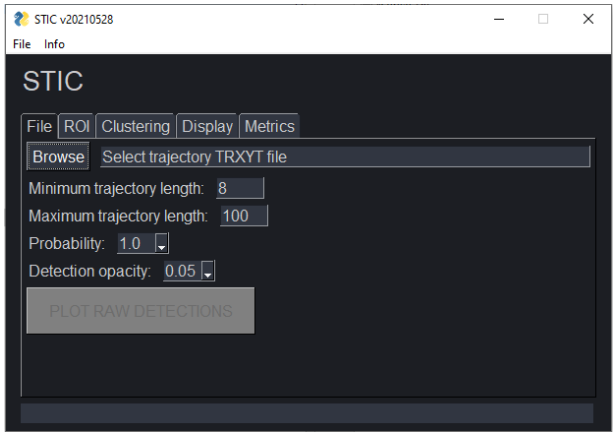


Extract these files into the directory of your choice and you should be good to go!

## Running STIC

Provided that you've installed Python and the required modules correctly, double clicking on **stic_gui.py** should open a command shell, load a brief splash screen and then the main program itself. The command shell is where STIC will print various useful information as it runs. If you close the shell you'll close STIC too.




STIC will display its version number (just the datestamp of its last changes) in the window title, so you'll always know what version you are running.

In order to simplify the potentially confusing clustering workflow, STIC is divided into a number of functional tabs, which it will automatically switch to as appropriate:

- **File:** select trajectory data, screen it based on trajectory length, and display the raw detections
- **ROI:** draw one or more rectangular or free hand regions of interest on the raw detection display (or load in previously saved ROIs) and select encompassed trajectories with optional density screening
- **Clustering:** enter the appropriate parameters for spatiotemporal clustering
- **Display:** plot the results of the clustering with a large range of options for trajectory, centroid and cluster display, and optionally export high resolution images for publication
- **Metrics:** visualise and save a wide range of cluster metrics

Each of these tabs will be detailed below.

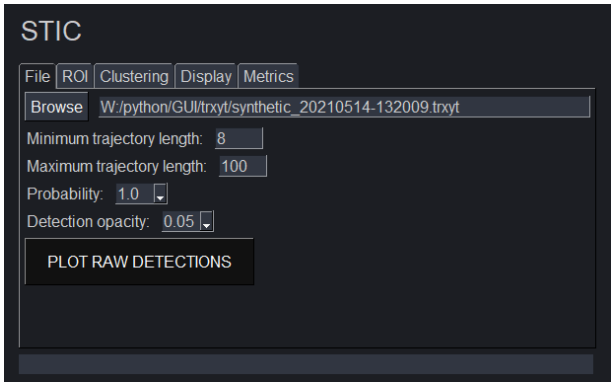
## File

STIC works on simple text files where each line contains the info for a single detection: trajectory number, X co-ordinate (µm), Y co-ordinate (µm) and time (sec), separated by spaces (not commas or tabs). These files are generated by Matlab from PalmTracer output, and should have a **trxyt** suffix.

TRXYT files should contain no headers or comments, just the data eg

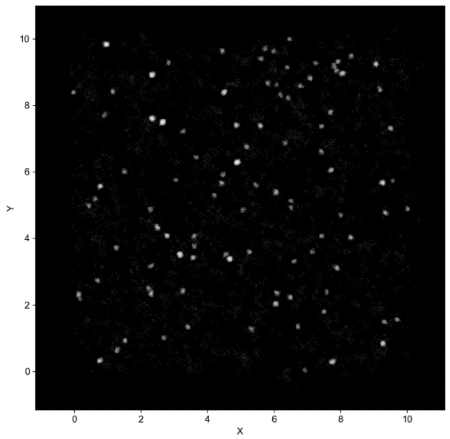
```
1 9.0117 39.86 0.02
1 8.9603 39.837 0.04
1 9.093 39.958 0.06
1 9.0645 39.975 0.08
2 9.1191 39.932 0.1
2 18.9266 39.915 0.12
```

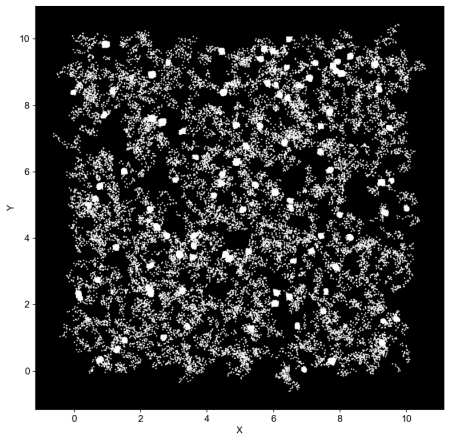
etc




Clicking the **"Browse"** button will open a file selector which will allow you to choose TRXYT files (only). Once you've selected a file its path will be displayed, and you can then opt to change the length filters which by default screen out really short (largely uninformative) trajectories and really long ones (which are usually background noise).

In order to plot this raw data STIC needs to know what % of the raw detections to use(default Probability = 1.0 → 100%) and how opaque each detection should be on the plot. If you have high density data then too high probability and opacity values might result in a solid white plot which could make subsequent ROI assignment tricky. The below images are 0.05 and 1.0 opacity respectively.

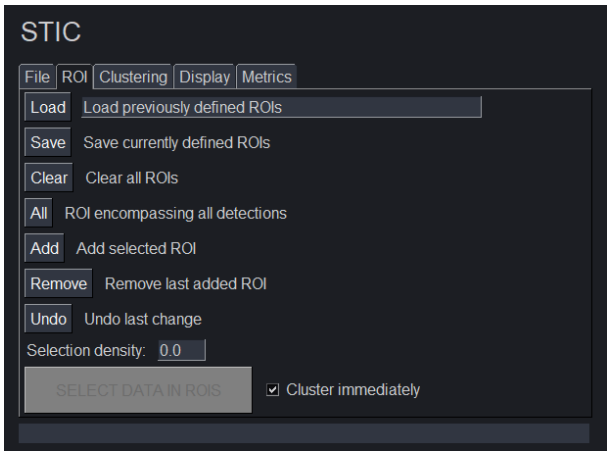



Clicking **"PLOT RAW DETECTIONS"** will, strangely enough, plot the raw detections, and the program will automatically switch to the **ROI** tab. You won't be able to click **"PLOT RAW DETECTIONS"** until you have actually specified an input file. In fact, all of the buttons in STIC will only be active at the appropriate stages of the analysis.

**IMPORTANT NOTE:** It appears that some Matlab processing of trajectory data converts trajectory numbers > 99999 into scientific notation with insufficient decimal points. eg 102103 to 1.0210e+05, 102104 to 1.0210e+05. This can cause multiple trajectories to be incorrectly merged into a single trajectory. For trajectories > 99999 STIC empirically determines whether detections are within 0.32u of each other, and assigns them into a single trajectory accordingly. For trajectories <99999 it honours the existing trajectory number.

## ROI



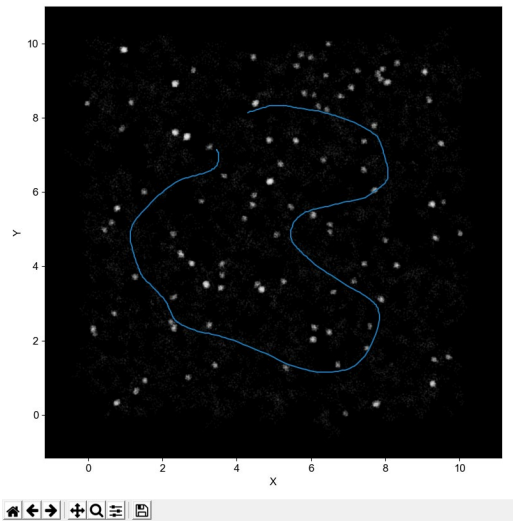

This tab allows you to select single or multiple regions of interest from the raw detections plot.

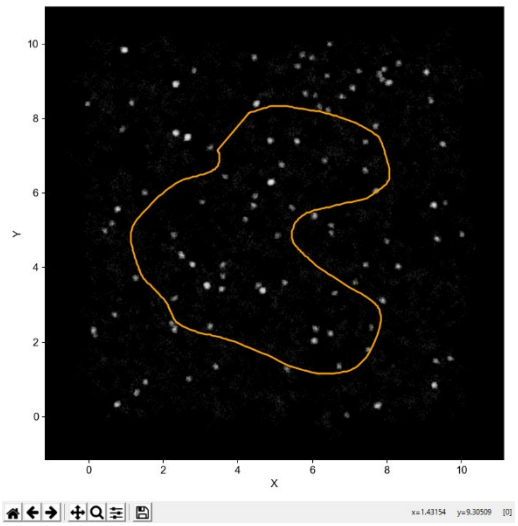
Clicking **"Load"** will allow you to load in automatically saved ROIs from previous STIC analyses.

Clicking **"Save"** will allow you to save the current ROI(s) to a datestamped YYYYMMDD_roi_coordinate.tsv file in the same directory as STIC itself.

Defining your own ROIs can be done either using freehand drawing, or using a rectangular shape. You can add as many ROIs as you'd like, and it doesn't matter if they overlap as the trajectories will only be selected once in overlap areas.

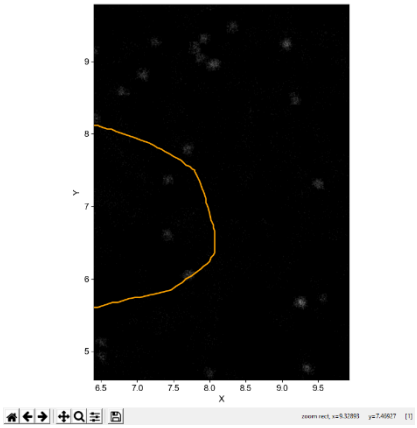
STIC defaults to using the freehand selector. Just hold down the mouse and draw. You don't have to close the shape. Clicking the **"Add"** button will close the shape and add the ROI in yellow.

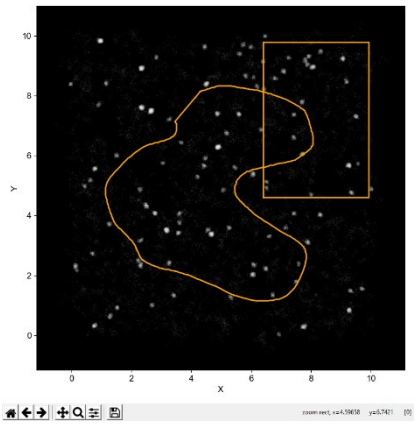



To switch to a rectangle selector, just click on the little magnifying glass icon on the bottom of the display window.

The mouse cursor will change to a crosshair. Hold the mouse to select the region you want, and when you release it the display will zoom in on the selected area. Click the **"Add"** button to add this zoomed area and return to the full sized image.





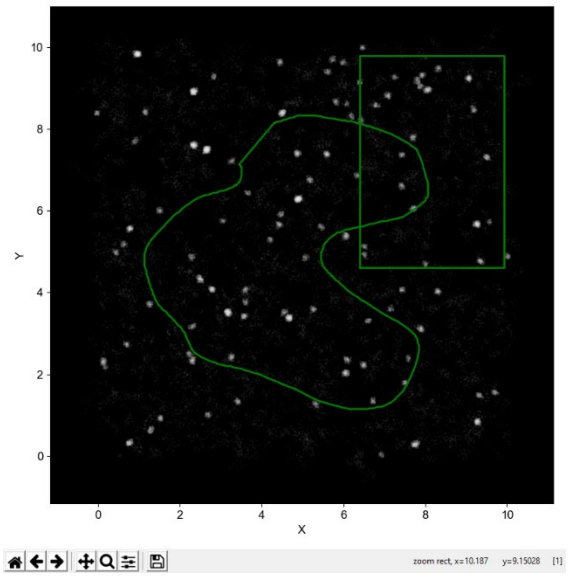
Clicking the **"Remove"** button will allow you to sequentially remove the last added ROI. You can click it more than once to remove more ROIs. This also applies to ROIs loaded in using the "Load" button.

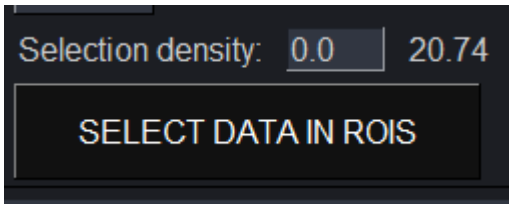
**"Clear"** will completely clear the ROIs.

**"All"** will create a single ROI which encompasses the full spatial extent of all the detections.

**"Undo"** will undo the last change you made to the ROIs. If you have 5 laboriously selected ROIs and accidentally press "Clear" or "All", you can click this button and get back to those 5 ROIs.

You can load a previous set of ROIs, and add more ROIs, etc etc etc. Basically there is a lot of flexibility to allow you to just analyse the areas you think are important. Once you've decided on the perfect set of ROIs then click the **"SELECT DATA IN ROIS"** button, which will now be active. If the "Cluster immediately" box is ticked, trajectories will be selected in ROIs and immediately clustered. If the box is unticked then the trajectories will just be selected. The ROIs will turn green to indicate this.






You'll see a figure next to the **Selection densit**y box, this is the number of selected trajectories per $\mu m^2$. You can adjust this down by entering the value you want in the Selection density box. Clicking **"SELECT DATA IN ROIS"** will reselect the trajectories and randomly drop some to bring the number down close to your entered value. Eg you could enter 15, and the system will return 14.89 trajectories/ $\mu m^2$

If you enter 0 in the box then **"SELECT DATA IN ROIS"** will just go back to selecting everything.

STIC will automatically switch to the **Clustering** tab once trajectories have been selected.

## Clustering


The values on this tab are at the heart of R-tree based spatiotemporal clustering. By default STIC uses th e above values, which generally give good results with synthetic datasets which represent the kind of density and clustering typically seen in PC12 sptPALM experiments.

**Acquisition time:** The length of the acquisition in seconds. 320sec = 16000 frames. The program works internally using seconds, not frames.

**Time threshold:** This is the time "thickness" in seconds that defines whether trajectories are considered to be interacting in time. A time threshold of 20 sec means that two clusters are considered as clustered if they overlap in space, and are within 20 seconds of each other. Each trajectory will be examined to determine if there are other trajectories up to 10 seconds earlier, and up to 10 seconds later. If you set a time threshold of 640 seconds, then STIC will only report spatial clustering (SIC?) and will return clusters more or less the same as DBSCAN or Voronoi.

**Radius factor:** STIC does not use the convex hull of the trajectory (Fig 2c), but rather calculates the ideal radius of the convex hull and builds a box based on this (Fig 2d,e). Unless the trajectory moved within a perfectly circular area, this box does not fully represent the extent of the molecule. So we can expand the box to better account for this. A value of 1.2 works well for starters. You can take it up to 1.5, possibly even 2 if you have a lot of directed trajectories or low density trajectories. But this will of course potentially start showing more spurious overlap. Alternatively, if you are using very high density data you can use a radius factor of <1. If your data is really high density you should consider either changing the selection density in the ROI tab, or perhaps using **segSTIC**, which clusters based on trajectory segment overlap rather than whole trajectory bounding box overlap.

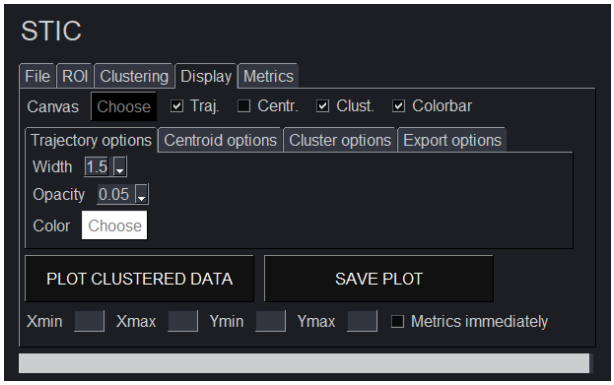
**Cluster threshold:** How many trajectories have to overlap in order to be considered a cluster. 3 is a good start. In high density data you may want to consider increasing this. Before you start complaining, be aware that a cluster threshold like this is at the core of DBSCAN and Voronoi too.

**Cluster size screen:** The general consensus is that protein nanoclusters are usually less than 0.1µm in radius. STIC clusters much larger than this are more likely to be random overlap of higher mobility trajectories. These can be screened out by setting this value appropriately. If you set it much below 0.15µm you might throwing out valid clusters, so use this wisely.

**MSD screen:** This setting may be useful to screen out high mobility background trajectories which could otherwise contribute to large diffuse clusters. Ticking this box will cause STIC to measure the MSD for all trajectories, and establish the average MSD. It will then reject those trajectories whose MSD is greater than the average. Use this carefully – for datasets with lots of low mobility trajectories the average MSD will be lower, and you might end up not using trajectories with low mobility but still higher than the average.

**"CLUSTER SELECTED DATA"** will apply your values and use spatiotemporal indexing clustering on the selected trajectories. If **"Plot immediately"** is ticked then the clusters will be automatically displayed after the analysis is complete.

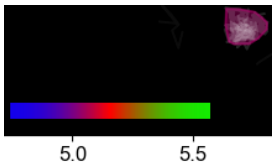
## Display

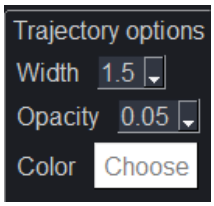



STIC gives you a large amount of flexibility as to how you can display your clustered data, and therefore has a few extra tabs that you can access. At any time, you can change various parameters and hit the **"PLOT CLUSTERED DATA"** button to update the display.

You can use the controls  on the bottom of the display window to zoom in (make sure you've selected the magnifying glass and changed the cursor to a crosshair), the arrows to move back or forward through the various zoom, and the home icon to go back to full view.


Choose the background colour of the plot, and whether you want trajectories, centroids and clusters displayed. The **"colorbar"** checkbox will plot a colorbar onto the display, where blue = earlier in the acquisition and green = later in the acquisition.

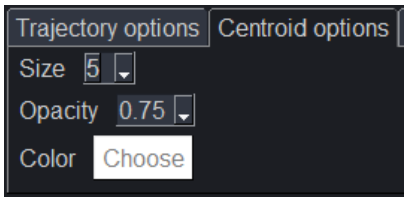



**Width:** how thick the trajectory plot lines are

**Opacity:** 0 = completely transparent, 1 = solid

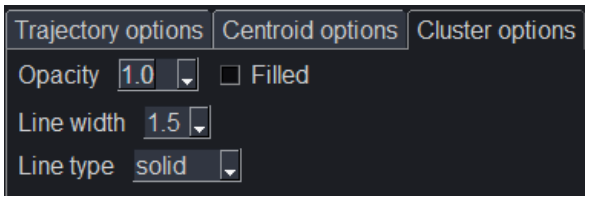
**Color:** all trajectories will be plotted in this color. Trajectories in clusters will also be this color, but will have greater opacity.




**Size:** how large the trajectory centroid will appear

**Opacity:** 0 = completely transparent, 1 = solid

**Color:** All centroids will have the same color, regardless of whether they are in clusters or not.



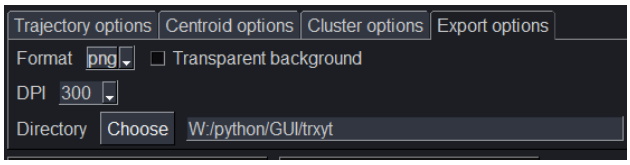

Each cluster represents a convex hull boundary around the greatest extend of the component detections. The color of the cluster represents its approximate time in the acquisition

**Opacity:** 0 = completely transparent, 1 = solid

**Filled:** The cluster will be filled with solid color at half the opacity.

**Line width:** how thick the trajectory border lines are

**Line type:** Style of the trajectory border line




At any time you can export a high quality file representing the contents of the main plot window, by clicking the **"SAVE PLOT"** button. These images may be more suitable for publication etc.

**Format:** eps, pdf, png, ps, svg. Depending on what software you want to use for figure wrangling, png is a common lossless bitmap format which will open in anything.

**Transparent background:** The image will ignore the background canvas color and will be output with a transparent background. This could come in handy if you want to make a fancy image with a gradient background for instance.

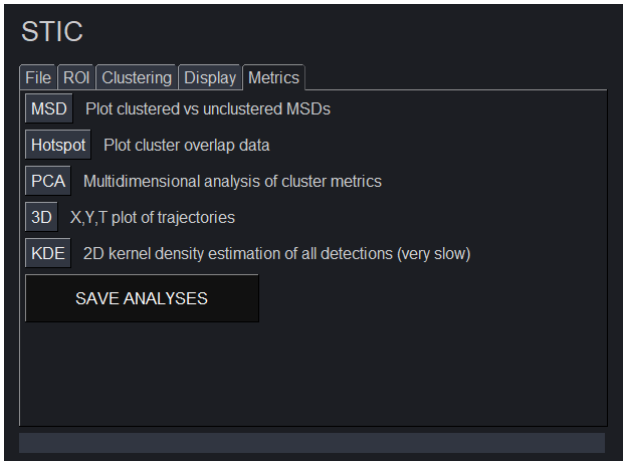
**DPI:** Dots per inch. 50 = lower resolution, 1200 = very high resolution but very large file size

**Directory:** Where to save the image.


Rather than use the mouse to zoom in on a region, you can specify it exactly by entering the corner co-ordinates for the zoom box. This can come in handy if you are eg comparing STIC, DBSCAN and VORONOI (the GUIs of these programs all allow you to zoom like this) of a particular dataset and want to visualise how the clusters differ in the same area. Hit **"PLOT CLUSTERED DATA"** to update the image once you've entered the co-ordinates. You can enter 0 in all boxes to revert to the full plot of all selected trajectories.

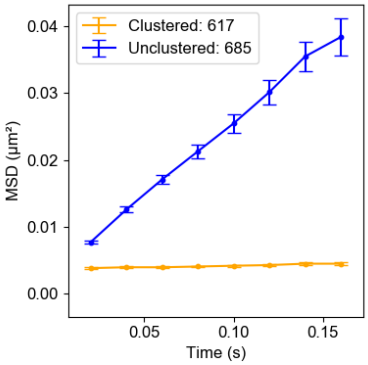
**"Metrics immediately"** will automatically switch to the Metrics tab once the plot has completed. This can be useful if you are analysing large number of TRXYTs all with the same parameters, and don't care much about the plotted cluster image. Ticking **"Cluster immediately" + "Plot immediately" + "Metrics immediately"** and clicking **"SELECT DATA IN ROIS"** in the ROI tab will selected trajectories, cluster them, plot the output, and immediately switch to the Metrics tab so that you can save the analyses or examine them further.

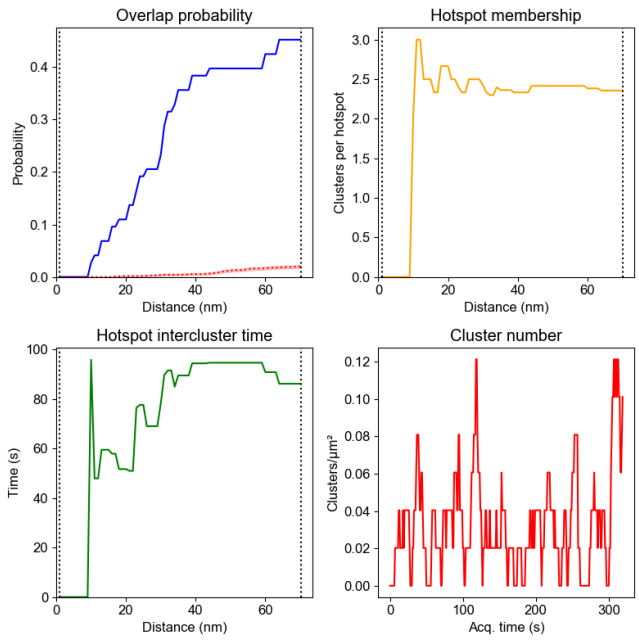
## Metrics




This tab will allow you to examine various cluster metrics to allow further assessment of the data. Because STIC does temporal clustering, many of these metrics are not available when using DBSCAN or Voronoi.

**MSD:** Will plot Mean square displacement vs time for clustered trajectories and unclustered trajectories. Ideally we want to see clustered detections with lower mobility as per the picture on the right. **IMPORTANT NOTE**: There is some discrepancy between STIC MSD calculations (a very "pure" python implementation) vs MSD for the same data in Palmtracer. This results in STIC MSDs generally reporting lower. You are encouraged to use STIC MSDs to determine whether the clustering has actually worked, but perhaps not for publication at this stage.

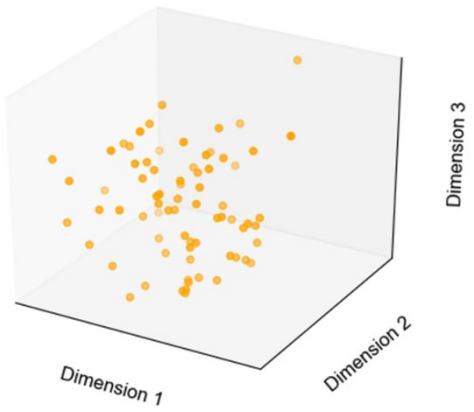


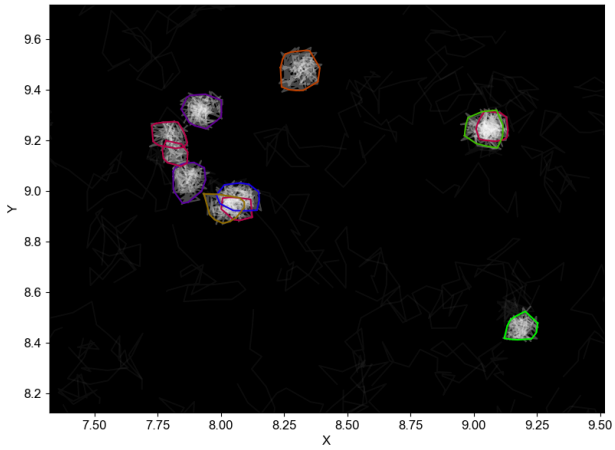

**Hotspot:** Because we can get spatial and temporal clustering information using STIC, we are able to determine the presence of "hotspots" where clusters repeatedly form and dissociate on the same region of the membrane. Overlap probability indicates the likelihood of this. Firstly, the average radius of all clusters is determined. Then, the spatial centroid of each cluster is determined. DBSCAN is then performed multiple times using an epsilon between 0 and the average cluster radius. At each epsilon the number of centroids with another centroid within epsilon is calculated, and converted to a probability such that 1 = all centroids have a neighbor within epsilon. In the figure on the left approximately 45% of all clusters have another cluster within 70nm (the average cluster radius), and ~ 20% have a cluster within ½ the average radius. Remember, they may overlap in space, but occupy a different point in time. Additional information can be extracted from the DBSCAN clusters, such as the number of clusters in each hotspot, and the time between these clusters. The cluster number plot allows you to determine

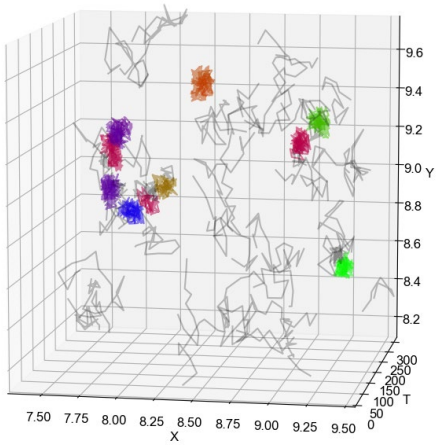
whether the overall number of clusters at a given point in time changed over the course of the acquisition. This could be useful in a system where cells were stimulated ½ way through an acquisition for instance.

**PCA:** Each cluster is assigned multiple metrics such as size, number of trajectories, average MSD, lifetime etc etc. Dimensionality reduction analysis using principal components analysis allow you to determine whether there are subpopulations of clusters with distinctly different metrics. Usually this is not the case, but one could imagine a situation where an analysis of a cell with two distinct membrane regions resulted in a differently behaving clusters. These might show up as distinct groups of dots on the PCA plot.

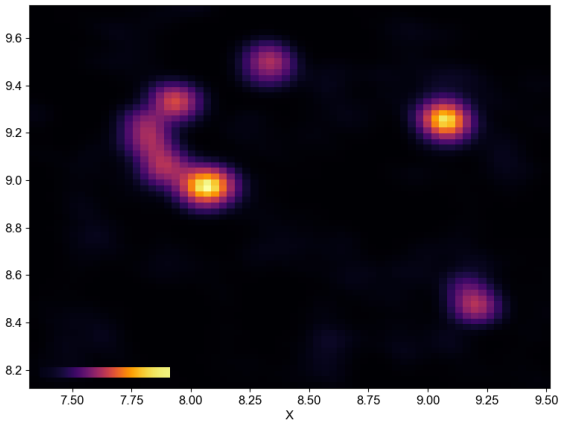



**3D:** STIC works by treating superres data as 3 dimensional [x, y, time]. In the final 2D plots, the time dimension is represented as color. If you click on the "3D" the trajectories on the main display plot will be rendered into 3D space, which will not only allow you to get a better feel for how the clustering picked up things in space and time, but might also make for some pretty pictures for publication.




You can rotate the 3D plot by holding the left mouse button and moving the mouse appropriately. You can zoom in or out by holding the right mouse button and moving the mouse up or down.




**KDE:** Kernel density estimation is basically a 2D histogram of the density of the detections, which shows regions of higher density as brighter color. Not only does it look pretty but it can be very useful for determining whether your clusters correspond to regions of higher detection density. This is not always the case however, since it's possible for a single trajectory with many detections to show up on KDE, but not be selected for STIC. KDE analysis can be very slow with lots of detections so please don't use it to analyse your entire image - start with a 2 µm x 2 µm region and see how long that takes before increasing things.

**"SAVE ANALYSES"** will save any open plot windows, a plot of the raw detections and ROIs, the ROI coordinates themselves, and a TSV file containing the various analysis metrics to a datestamped folder in the same directory as the TRXYT you analysed. Metrics.tsv can be opened in any text editor or Excel

The top lines of the file contain an overview of the analysis:

| | |
|---|---|
| SPATIO TEMPORAL INDEXING CLUSTERING - Tristan Wallis t.wallis@uq.edu.au | |
| TRAJECTORY FILE: | W:/python/GUI/synthetic2/synthetic_20210531-094509.trxyt |
| ANALYSED: | 20210618-153345 |
| TRAJECTORY LENGTH CUTOFFS (steps): | 8 - 100 |
| TIME THRESHOLD (s): | 20 |
| CLUSTER THRESHOLD: | 3 |
| RADIUS FACTOR: | 1.2 |
| MSD FILTER THRESHOLD (um^2): | None |
| CLUSTER MAX RADIUS (um): | 0.15 |
| SELECTION AREA (um^2): | 119.1594 |
| SELECTED TRAJECTORIES: | 2398 |
| CLUSTERED TRAJECTORIES: | 1165 |
| UNCLUSTERED TRAJECTORIES: | 1233 |
| TOTAL CLUSTERS: | 146 |
| HOTSPOTS (CLUSTER SPATIAL OVERLAP AT 1/2 AVERAGE RADIUS): | 11 |
| TOTAL CLUSTERS IN HOTSPOTS: | 26 |
| AVERAGE CLUSTERS PER HOTSPOT: | 2.363636 |
| PERCENTAGE OF CLUSTERS IN HOTSPOTS: | 17.808 |
| | |

The remainder of the file contains the individual and averaged metrics for the clusters identified during the analysis

| INDIVIDUAL CLUSTER METRICS: | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| CLUSTER | MEMBERSHIP | LIFETIME (s) | AVG MSD (um^2) | AREA (um^2) | RADIUS (um) | DENSITY (traj/um^2) | RATE (traj/sec) | AVG TIME (s) |
| 1 | 16 | 9.15 | 0.001867725 | 0.028079762 | 0.094541344 | 569.8054038 | 1.74863388 | 111.80375 |
| 2 | 17 | 13.61 | 0.001861652 | 0.038835033 | 0.11118262 | 437.7490825 | 1.249081558 | 195.1952941 |
| 3 | 10 | 9.59 | 0.00177983 | 0.030512465 | 0.098551607 | 327.7349148 | 1.042752868 | 132.509 |
| 4 | 12 | 9.9 | 0.001530125 | 0.026496711 | 0.09183771 | 452.886405 | 1.212121212 | 259.7658333 |
| 5 | 6 | 6.38 | 0.001356163 | 0.013386859 | 0.065277634 | 448.2007437 | 0.940438871 | 256.64 |
| etc | etc | etc | etc | etc | etc | etc | etc | etc |
| 125 | 4 | 2.93 | 0.001511485 | 0.008466209 | 0.051912215 | 472.4664846 | 1.365187713 | 193.83 |
| 126 | 13 | 6.36 | 0.0016481 | 0.031657953 | 0.100384458 | 410.6393095 | 2.044025157 | 113.5153846 |
| 127 | 9 | 8.51 | 0.002087885 | 0.01981891 | 0.079426412 | 454.111756 | 1.057579318 | 24.23111111 |
| 128 | 12 | 8.63 | 0.001387175 | 0.027698748 | 0.093897739 | 433.2325734 | 1.390498262 | 180.29 |
| AVG | 7.9453125 | 7.32609375 | 0.001681856 | 0.020444714 | 0.077608938 | 405.9038449 | 1.086785969 | 168.3391001 |
| SEM | 0.397641453 | 0.229122877 | 2.23E-05 | 0.001004995 | 0.001945763 | 9.344900604 | 0.039649745 | 7.810406641 |

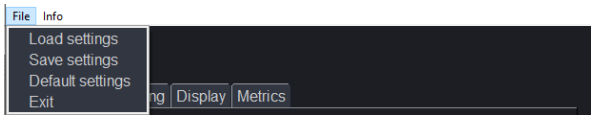
This data can be plotted as is, but the real power comes from being able to combine these metrics files from larger numbers of analyses across multiple conditions. This is handled by **STIC WRANGLER**, which is detailed in a separate document.

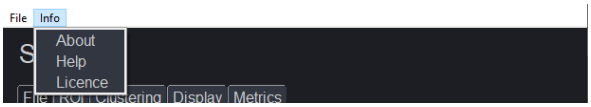
## Multiple analyses

Once you have finished analysing a given TRXYT, you can flick back to the **File** tab and load another one. This will automatically close all other open windows except the main display window. You should never need to close this window, but if you do the program will let you know all about and will open a fresh window and reset the analysis.

## Menus

In addition to the tab and button functionality of STIC, there are some menu items you should get acquainted with.



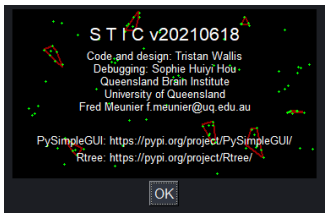

The first time you run STIC it will create a file called **stic_gui.defaults** which contains the default values for the various steps of the analysis. You can load it into a text editor or spreadsheet to view it if you'd like.

| | |
|---|---|
| Trajectory probability | 1 |
| Raw trajectory detection plot opacity | 0.05 |
| Selection density | 0 |
| Trajectory minimum length | 8 |
| Trajectory maximum length | 100 |
| Acquisition time (s) | 320 |
| Time threshold (s) | 20 |
| Radius factor | 1.2 |
| Cluster threshold | 3 |
| Canvas color | black |
| Plot trajectories | TRUE |
| Plot centroids | FALSE |
| Plot clusters | TRUE |
| Plot colorbar | TRUE |
| Trajectory line width | 1.5 |
| Trajectory line color | white |
| Trajectory line opacity | 0.25 |
| Centroid size | 5 |
| Centroid color | white |
| Centroid opacity | 0.75 |
| Cluster fill | FALSE |
| Cluster line width | 1.5 |
| Cluster line opacity | 1 |
| Cluster line type | solid |
| Plot save format | png |
| Plot save dpi | 300 |
| Plot background transparent | FALSE |
| Auto cluster | TRUE |
| Auto plot | TRUE |
| Cluster size screen | 0.15 |
| Auto metric | FALSE |
| MSD filter | FALSE |

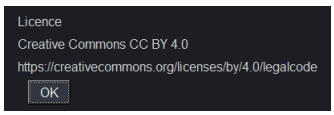
Nearly every value that you can change in STIC can be saved to this file. So if you arrive at a set of values that work well for a given set of data, **"File >> Save settings"** will save them to stic_gui.defaults. The next time you start STIC, they'll be loaded in as the defaults. If you get too far out of whack, **"File >> Default settings"** will restore the defaults. But remember if you don't save them, the next time you load STIC you'll get your messed up values! Similarly if you started with your preferred values, and changed multiple settings, **"File >> Load settings"** will restore your last saved settings.

**IMPORTANT NOTE:** Periodically STIC is updated with new functionality that requires a new **stic_gui.defaults** file. Newer versions of STIC may crash if loading an older defaults file. You are encouraged to delete your existing defaults file and run the new version of STIC to recreate the correct file with default settings.

Clicking **"About"** will pop up this glorious little window with a real time molecular clustering simulation. Warning this is mesmerising to look at and could waste several minutes of your life. Why did I do this? Because I could.




**"Help"** does nothing much at this stage, eventually it will link to a version of this document.




**"Licence"** reiterates the CC BY 4.0 concept embraced by STIC and allows the user to find out more about this powerful open source licence.