Using the RSC programs to fit and subtract vesicle membranes.

F Sigworth 3 September 2024

* Matlab setup. Put the directory aEMCodeRepository into your home directory. Copy the startup.m file to your home directory; on startup Matlab reads this to load the paths to the various library files. It’s best to start Matlab in the home directory, then cd to your working directory from within Matlab. The programs use plain Matlab but the Vesicle\_finding\_GUI uses the image processing toolbox.
* Processing directory structure. We assume we are in the equivalent of a Relion directory. New folders are created in it as processing proceeds, including Info/, Merged/, Jpeg/, etc.

1. rlStarToMiFiles

This converts a Relion-style micrographs\_ctf.star file to a set of micrograph info (mi.txt) files, and creates the various directories. It puts up a file selector for a micrographs\_ctf.star file which contains metadata and also paths to the micrographs. For each micrograph rlStarToMiFiles program writes a \*mi.txt file (and an accompanying binary \*mie.mat file). It also displays and writes a binned (default by 4) micrograph file, Merged\_sm/\*ms.mrc, Jpeg/\*.jpeg (a jpeg version of the same image) and JpegInv, the same to which an inverse CTF filter has been applied.

Once we have vesicle-subtracted micrographs (created in rsRefineVesicleFitsA) those will be written full-sized into the Merged/\*\_v.mrc, downsampled to Merged\_sm/\*mvs.mrc, and also jpeg files.

1. Vesicle\_finding\_GUI

Use this program to locate each vesicle and assign approximate radius and signal amplitude, which are then written to the micrograph’s mi file. On closing the GUI the program saves all the search parameters in the file VFContext.mat in your working directory. This is useful for restarting the GUI, but also is used for **batch processing**. The VFContext.mat file is read by the program in batch mode and searches each micrograph in a list.

A screenshot of a computer

Description automatically generated

**Automatic finder group** (3)

1. The main parameter you need to set is **Min Amp** in section 3. This is the threshold for vesicle finding. (The actual amplitude value is given in yellow in each vesicle on the display.) **Max Amp** is less critical, I usually leave it 2 or 3 times **Min Amp**.
2. **Min R** and **Max R** are the limits of vesicle radii that are searched, in angstroms.
3. The autofinding buttons are:

* **Find**: search for circular vesicle profiles.
* **Find & Track**: the same, but after finding with circular templates, follows the membrane to approximate non-circular profiles.
* **Find Concentric**: The other finding operations do not find vesicles whose centers lie within other vesicles. But this operation subtracts models for each found vesicle, then runs a Find operation to search for any residuals, e.g. concentric vesicles.

**Data group** (1)

Here you **load** an mi.txt file and go forward and backward through the enclosing Info/ directory. Each mi file is updated with the present set of vesicles whenever you move on to another, unless you’ve selected **Read only**. You can set Gaussian **Highpass** and **Lowpass** filters for display purposes. You can set the **W**hite and **B**lack levels of the display, or click **A**utoscale to set them automatically. **RoboFit** toggles an automatic sequence of vesicle fitting, then moving to the next mi file. Click once and wait up to maybe 30 seconds to stop the automatic fitting. Batch operation employs the RoboFit algorithm.

**Masking group** (2)

I use masking to keep carbon areas, lines and other artifacts from triggering the vesicle finder. Best to leave these controls alone unless you really need them.

**Scatterplot** (4)

Plots amplitude (approximate absolute values of radians/VÅ) vs radius in Å. Blue dots are “good” vesicles, red indicate out-of-range but nevertheless fitted “bad” vesicles.

**Show** button

Cycles through various display modes.

**Main display**

* Clicking in the center of a vesicle allows you to switch good -> bad -> erased vesicle, while the details of the vesicle are printed in the command window.
* When Outline mask or Paint mask (2) is active, clicking or dragging in the main window allows you to create a mask manually. No vesicle is found having its center in the mask.

1. rsRefineVesicleFits

Once vesicles have been located and approximate geometry determined in Vesicle\_finding\_GUI, this program performs a least-squares fit of a vesicle model. Select one or many mi.txt files, and the program will operate on each one, updating the vesicle parameters. This program also creates subtracted micrographs (files Merged/\*\_v.mrc) and downsampled versions of the same (Merged\_sm/\*mvs.mrc) and corresponding jpeg files. Starting with a physical model of a spherical membrane, the local radius is allowed to vary with angle about the center of the vesicle; the number of terms in the Fourier expansion in angles depends on the starting radius, and this can be set in parameters passed to the program. The present default is

dpars.rTerms=[120 120 180 240 300 inf]

Which means that vesicles up to 120 Å radius are fitted with 1 or 2 terms; up to 180Å with 3 terms; up to 240Å with 4 terms and so on.

The program is slow: it tries brute-force perturbations of radius, and some operations are not optimized. For substantial datasets I run it as a jobarray of multiple Matlab instances on our cluster.

1. rsPickingPreprocessor4

I use this program to compute geometry-aware normalized cross-correlation functions for particle picking, with templates based on a 3D map. The correlation results are stored in Merged\_sm/\*rscc.mat files.

1. SimpleRSPicker

This is the interactive geometry-aware picker which uses the mi files and rscc files to perform picking, and updating the mi files.

1. rlMisToParticleStar

This program is a script. Make edits in the first lines to set the working directory (or delete the cd line if you’re already in the working directory. Select whether you want to write out particles.star files (only if you’ve used SimpleRSPicker) or (default) just star files for the subtracted micrographs (micrographs\_ctf\_v.star) and (just for convenience) the unchanged unsubtracted micrographs (micrographs\_ctf\_u.star). These star files are all stored in the RSC/ directory, and I import them into Relion. For particle extraction I use particle.star and micrograph\_ctf.star files for “re-extracting” in Relion’s particle extraction job.

V. K3DistributedPipeline.m and k3pRung

A Matlab script (must be edited for paths, filenames and operations desired) and the corresponding bash script that is calls it. K3pRung is the script executed by each job of a Slurm jobarray. K3DistributedPipeline can carry out each step of the processing once everything is set up (vesicle and particle picking parameters set for example) except for the initial creation of mi files. For that first step there is a separate k3DistributedMiToStarFiles.m script.