USER'S MANUAL

ASTRO 1.0

2017

ASTRO 1.0. User – friendly software for realistic simulations of astrocytes.

GENERAL INFORMATION

Electrically non-excitable astrocytes appear able in transducing, integrating and propagating physiological intracellular diffusion signals. Decrypting this type of signalling, however, poses a conceptual difficulty because it requires an understanding of molecular interactions in the massive morphological structure of nanoscale-thin leaf-like processes which constitute the bulk of ASTROCYTE geometry. How a particular cell signalling engages a precise type of geometry remains therefore poorly understood.

There have been no attempts to develop an ASTROCYTE model with such a spongy morphology even though this could provide the key to mechanistic insights into astrocytic physiology and Ca2+ signalling.

To understand the role of complex pattern in cell function we have adapted the NEURON modelling environment to build a simulation tool to produce different ASTROCYTE models with the detailed morphology, membrane properties and known molecular signalling mechanisms. The tool enables to design a distributed Ca2+ homoeostasis mechanisms including diffusion, wave propagation, gap-junction escape or channel currents whereas the simulation environment also has the capability to mimic uncaging, membrane physiology, volume current injections or fluorescence recovery after photobleaching (FRAP) experiments in the 3D tissue volume containing the astrocyte.

To our knowledge, this is the first attempt to have a full-scale tool for astroglia simulations, which we believe will attract significant interest among a broad audience of cell biologists and neuroscientists.

This manual describes how to set up and run the computational tool ASTRO.

The basic version of ASTRO consists of three parts, which can work both together and separately.

System Requirements for full version of ASTRO:

- The basic preinstall software: MPIC++ (Linux, Remote cluster), MATLAB not older 2013 (Windows) and Neuron 7.0 (Windows, and Linux)
- Platform: Linux and Windows.
- Type of operation : Sequential and parallel (MPI) computing.

Three following scenarios of simulations are proposed:

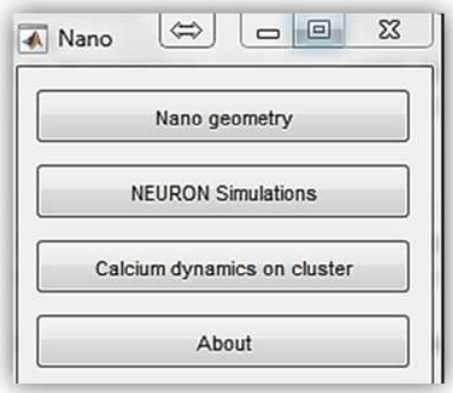


Fig.1. Main panel.

1. **Nano geometry**. Simulation and transformation of the 3D reconstructed nanostructure of astrocytes into a data-file contains statistic of diameters of cylinders suitable for NEURON. This part of software can be used separately from other software. To operate the nano the host computer requires only MatLab not older 2012, Windows 7 or 10.
2. **NEURON simulations.** Design and simulations of the compound digital model of an astrocyte. This option is for creating a real geometry of an astrocyte, combining together various cellular mechanisms and composing different configurations of computations. The host computer requires Neuron7.2 operating under Windows.
3. **Calcium dynamics of the cluster**. Design and simulations the long-term intracellular calcium dynamics of the astrocyte with real geometry using the cloud parallel computers (Matlab → Windows, Neuron → Linux). In this scenario, a user works on a host computer with pre-install MATLAB program. The user creates the MAT-file with instruction for computation and uploads this file to a remote HPC cluster operating under Linux with pre-install NEURON and MPI. Then this cluster starts a simulation of ASTROCYTE without the communications with the host computer. The user's computer connects to the remote process from time to time to a) know time of computation is and b) download intermediate results of the simulation (the results are shown in MATLAB Plots and save to the output file). When simulation completes, MATLAB host computer downloads the output MAT-file and visualises the results of the computation.

Contents

GETTING STARTED

Preparing “ASTRO” for the first launch

The newest installation version currently available and can be downloaded from ***https://github.com/LeonidSavtchenko/Astro***, which should be installed on the host (Windows) and cluster computers (Linux) keeping the following folder structure (Fig.2).

Fig.2 Folder structure of Astro 1.0 over the Windows.

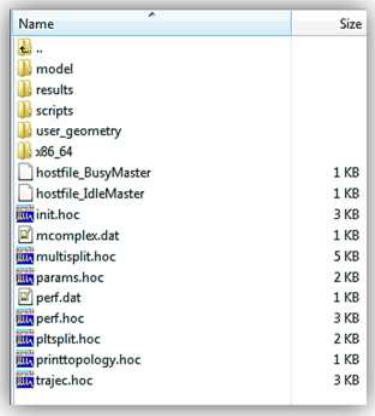
To get started with the "Astro" in the full version, it is necessary that the local computer had a Matlab version release after 2013, Neuron 7.0 and access to the Internet.

When you launch ***Start.m***, you should get a screen that looks like this (Fig.1).

This is the Astro initial screen and gives you a pretty good starting point for how to get commenced with the builder. This initial menu can do just about any major task that is possible in Astro.

On the host computer, a user can work with two parts of Astro: Nano Geometry and NEURON simulations without connection to a remote cluster.

To work with function “Dynamics on cluster” the cluster must be initially prepared for work. To do so the files from GITHUB directory ***Astro/clusterCaSim/hpc/*** should be downloaded on the cluster keeping the structure of catalogues:



Important information: cluster should also have the NEURON 7.0 for Linux.

To connect the local computer and the cluster in the file located on the host computer in the directory

Astro/clusterCaSim/host/Core/scripts/win-lin/params.bat

the user must set the program path to the cluster (HEADNODEIP), the password (PASSWORD), the login (LOGIN) and path (HEADNODEWORKERDIR) of cluster location of folder ***HPC***.

Part of the params.bat file.

```
**********************************************
set HEADNODEIP=144.82.46.83
set LOGIN=my_login
set PASSWORD=my_password
set HEADNODEWORKERDIR=/home/*******/hpc
**********************************************
```

Nanogeometry. MATLAB code.

The MATLAB code "Nano" was designed for the analysis and the transformation of the astrocyte nanostructure, obtained via 3D reconstruction, into a set of coaxial cylinders which is a typical representation of the cell geometry implemented in NEURON.

The Nano code either can be activated from the standard menu (Fig.1) or can be started separately from the directory ***nanoGeometry*** launching the file ***"START_NanoGeometry.m"***:

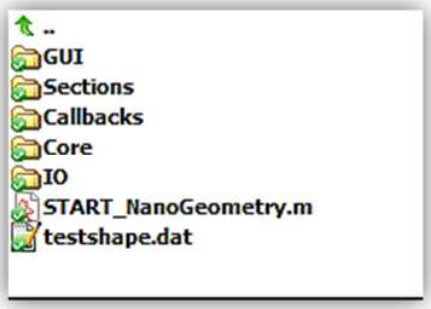


Fig.3. Structure of directory "nanoGeometry " with MatLab code of Nano simulation for ASTRO.1.0

and to upload the following example data file, "***testshape.dat***" contained a typical 3D Nanostructure of astrocytes. The user can upload any file with the 3D reconstruction of an astrocyte in ASCII format.

After downloading the data file, the following two panels will appear:

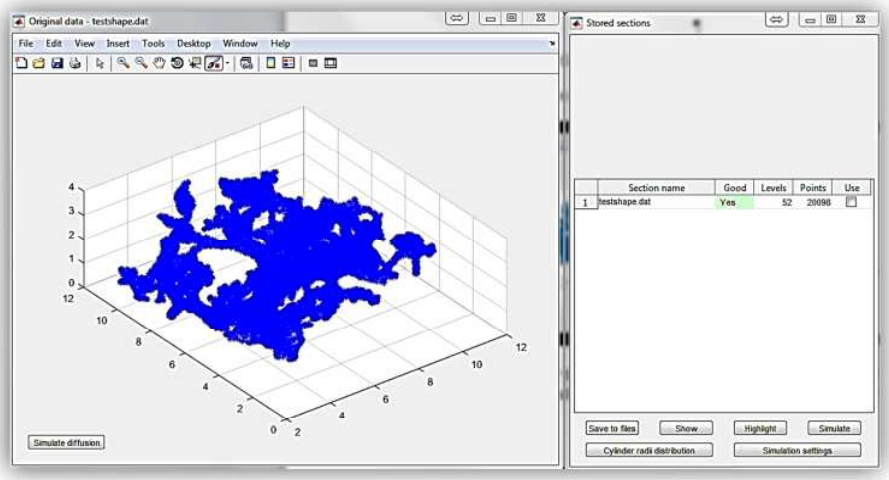


Fig.4. Two panels are for selection and saving "fingers" extracted from the basic downloaded 3D structure (left) and the window with the list containing off-save structures (right).

The first window (Fig.4, left) contains the raw digital data of Astrocyte nanostructure obtained via 3D reconstruction. The second window (Fig.4, right) is for storing and transforming small sections selected from the initial basic structure (Fig.4. left). The selected "finger" can be chosen/looking (Rotation) from the core structure using "Brush" method.

The purpose of this stage is to select as much as possible "fingers" for the production of statistics of cylinder diameters for the complete model of astrocyte recorded in NEURON scripts.

The Fig.5 shows the sample how to choose the tiny area, "finger", of a 3D astrocyte.

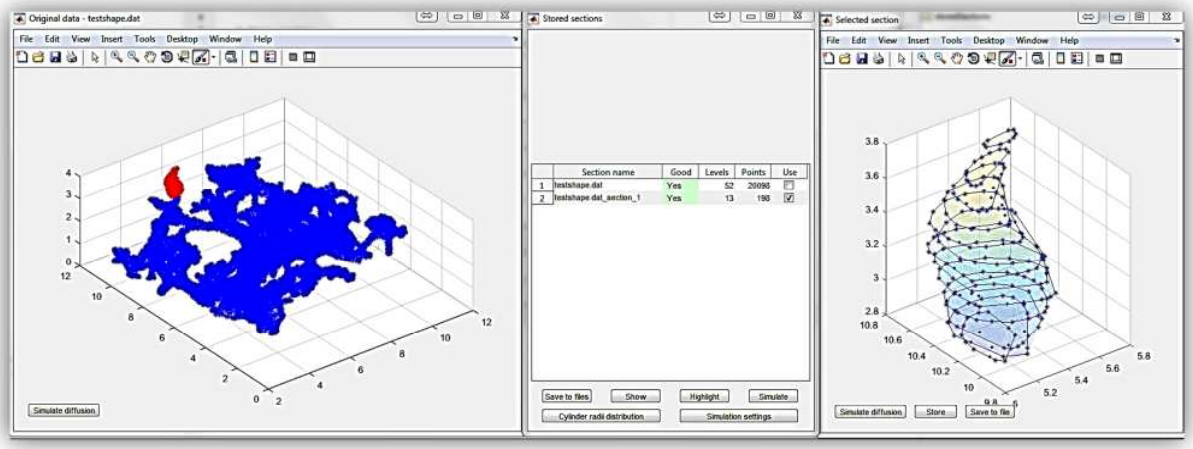


Fig.5. Three windows for storing and analysing data. The two left windows are the same as in Fig.4. The right window is the small selected structure "finger", indicated in red on the far left panel.

Using the method "brush" the user can choose a little "finger" from the whole structure (Fig5,left). The "finger" is a continuous part of the 3D astrocyte and can be selected from any area. The "finger" is

automatically repainted in red, and the chosen points, organised into the layers, appear in a separate window (Fig.5. right). Here, the user can examine the selected “finger” with functions located on the top of the window and can delete unwanted points with “brush” and “delete/remove” features (to select and to the right-click). For some version of Matlab, it is better to use the "NaN" function for removing. The ability to switch points is critical for simulation and for the proper statistic of diameters.

If the user is happy with the finished structure, he can add it, using the key "Store", to the main menu for further calculations of the overall statistics of diameters. If the selected structure is good for statistics, it is displayed (middle panel) as right (yes) and painted in green otherwise it will be indicated as “No Good” and painted in red. For the statistics, the user can combine the different fingers (Fig.6). The more fingers, the statistics of the cylinder diameters is better for use in a complete model of an astrocyte.

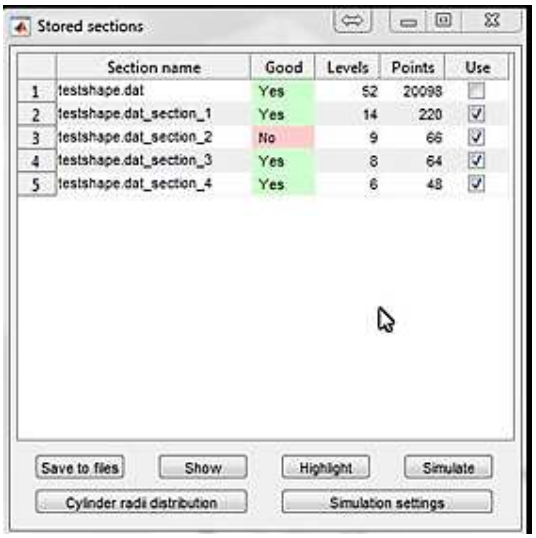


Fig. 6 Example of panel shows four added structures (three compositions are suitable for analysis, the figure number 3 is wrong for reviews). The wrong “finger” must be excluded from the total statistic (unticked). Panel has 6 different options: “Save to files”= Data will be saved to fil; “Show” = The structure ticked by the user will be shown; “Highlight”= Highlighting the indicated structures on the general 3D structure; “Simulation” The begin of Brownian simulation of chosen finger; “Cylinder radii statistic” =The calculation of statistical data of selected structures; “Simulation setting” = The definition of parameters of simulations.

The menu panel (Fig.6) has 6 different options including the most significant opportunity, «Cylinder radii distributions”, for creating statistics of cylinders diameters for the whole astrocyte model composed in NEURON scripts.

Statistic of diameters

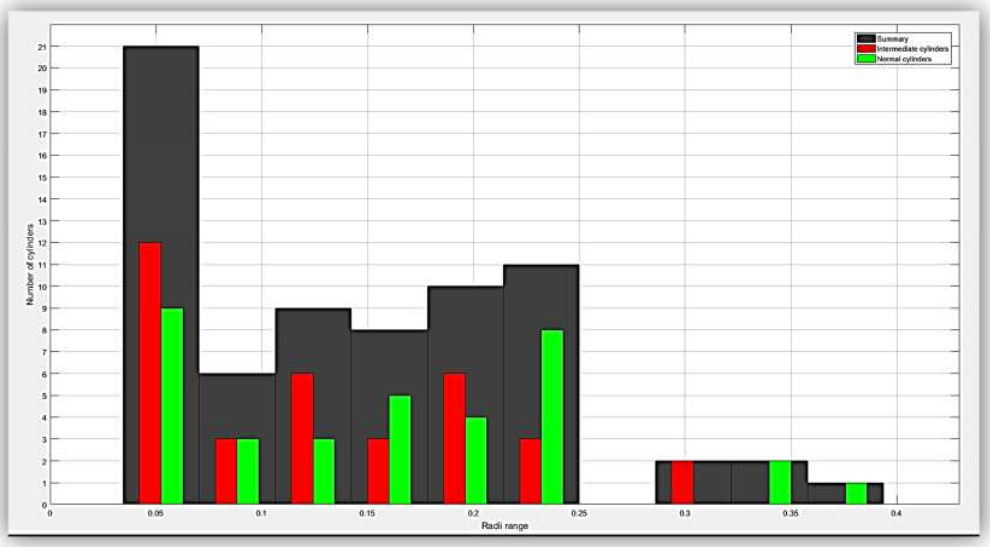


Fig. 7. Example of statistical distributions of cylinders diameters.

As a result of this action, the user will obtain two bar-plots of the statistic of diameters (red and green) for two types of cylinders (regular cylinder and transitional cylinders) and summary (black) statistic corresponding to these two classes of diameters. This example of statistic was calculated for the four "fingers" of the 3D astrocyte as indicated in Fig.6.

The numerical data of this statistic can be saved into the file for later uploading to NEURON script to generate the digital astrocyte.

Procedure for converting the real structure of the fingers into the structures of coaxial cylinders

The following section explains the procedure of transformation of the 3D original reconstructed of nano structure of astrocyte into cylindrical structures. The example of the "**Finger**" is shown on the Fig.8.

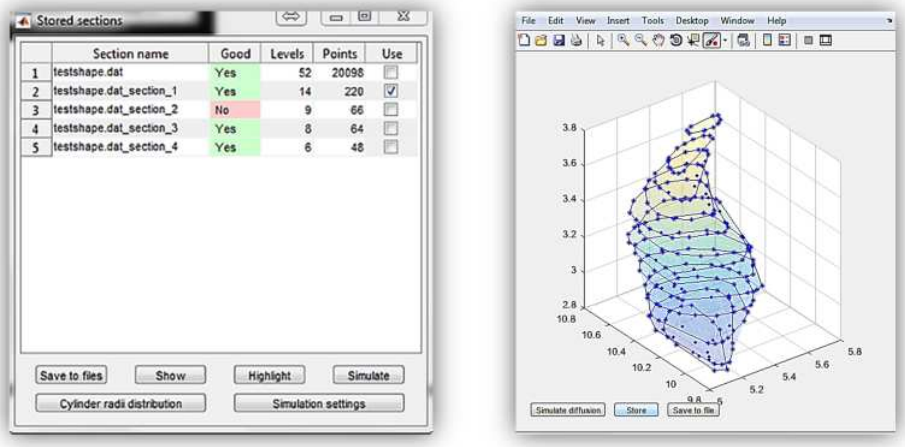


Fig.8. The test of the second structure indicated on the left panel and its three-dimensional reconstruction where each layer is represented by a polyhedron.

Once the user selects a small structure and saves it for further statistics, then at the same time any layer of the “finger” automatically transforms into a polyhedron generating the new structure, the set of N not coaxial polyhedrons (Fig.8, right). The number, N, of polyhedrons matches to the number of layers of the original composition.

The next action of “Nano” is to reconstruct these non-coaxial polyhedrons into 2N co-axial cylinders. The number of cylinders is doubled because “nano” produces two types of cylinders, the standard cylinder and the transitional type, which plays a role to adjust the cylinders along the axis. The radius of the regular cylinder is calculated from the corresponding (the related layer) polyhedron according to the principle of equity of areas. The radius of the transitional cylinder is calculated from the surface of the intersection of two neighbouring areas of polyhedral assuming that this is a circle. In conclusion, “Nano” creates the cylindrical structure (Fig.9) and determines the overall statistics of diameters for cylinders.

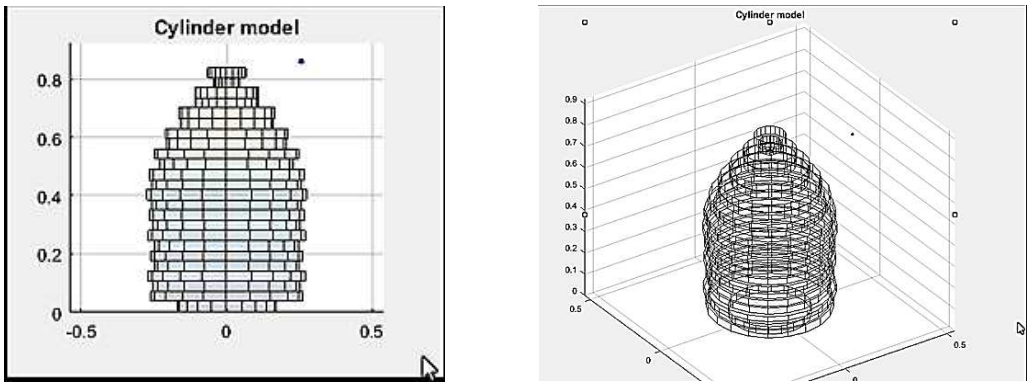
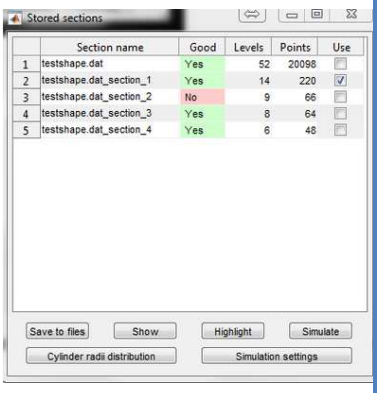


Fig.9. The structure consisting of a set of 2N cylinders, which is equivalent to the structure of the N polyhedral shown in figure 8.

Introduction of transitional cylinders is necessary to mimic a displacement of layers related to the direction of the section plane. The small/large diameter of the transition cylinder means the large/small deflection of between layers.

A biophysical check of the chosen finger.



To examine the quality of the transformation from N non-coaxial polyhedra into 2N coaxial cylinders, “Nano” has an additional opinion based on the study of Brownian diffusion and electrodiffusion (electric current) of free particles. The central qualifying parameter is a passage time of Brownian particle across a cylindrical and a polyhedral structure.

The quality control of the cylindrical structure is as follows: M-particles begin to move at the origin of both structures, cylindrical and polyhedral, and the time of transferring these particles through the structures (fluxes) are calculated. After simulations, two plots of fluxes

versus time are computed for both structures.

For this simulation, the main panel has two keys: “Simulate” and “Simulation parameters”.

Parameters of simulations

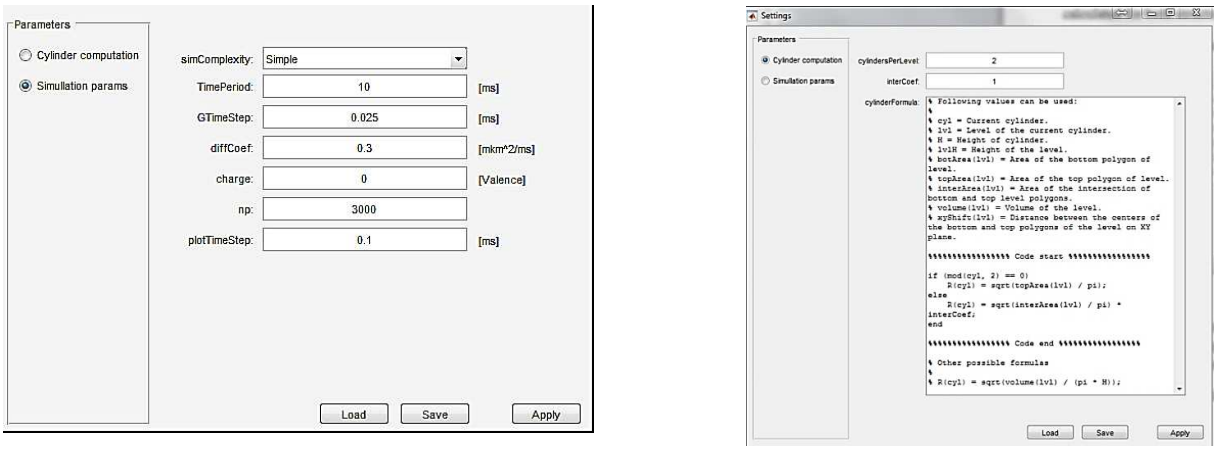


Fig.10. The set of parameter (left panel) to simulate Brownian motion in two different structures, where np=3000 is a number of particles, Charge =0 is a valence, DiffCoef =0.3 □m2/ms is a diffusion coefficient, GstepTime=0.025 ms is a time step of diffusion, TimePeriod=10 ms is diffusion time. The right panel indicates the MatLab code for the parameter of the cylinder. Critical parameter “CylinderPerLevel” determines the number of cylinders per polyhedron. Activating the key “Simulate” generates the simulation panel (from the left).

The panels of the simulation and results of simulations.

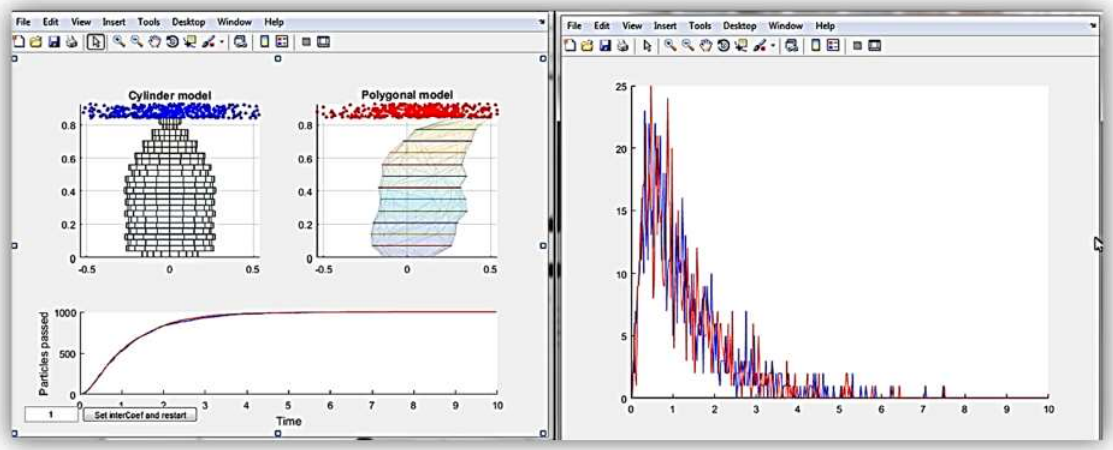


Fig.11. Brownian simulations. The bottom – left plot indicates the dynamic of number particles inside the structures (top left.) The right panel shows the dynamic of fluxes of molecules inside the two structures (top left.). Blue colour corresponds to a cylindrical model, red colour – polygonal model.

If the diffusion fluxes coincide, then the structure remains for the general statistics of the cylinder diameters, both transient and normal.

What happened if the fluxes do not coincide?

For example, see the panels (Fig.12).

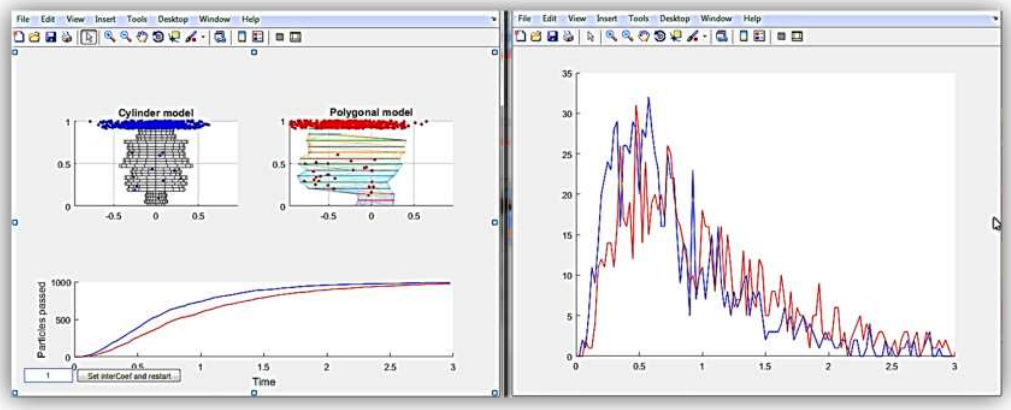


Fig. 12. The results of simulations. The structure of cylinders transmits Brownian particles faster than the structure of polyhedrons. To match these fluxes, one can use the parameter "Set interCoef" which modify the diameters of transitional cylinders, the smaller the diameters, the slower fluxes.

In the example (Fig.12), to decrease the speed of particle through the set of cylinders, the user needs to decrease ***interCoef*** and restart the simulation. In the result, the new plot will be generated.

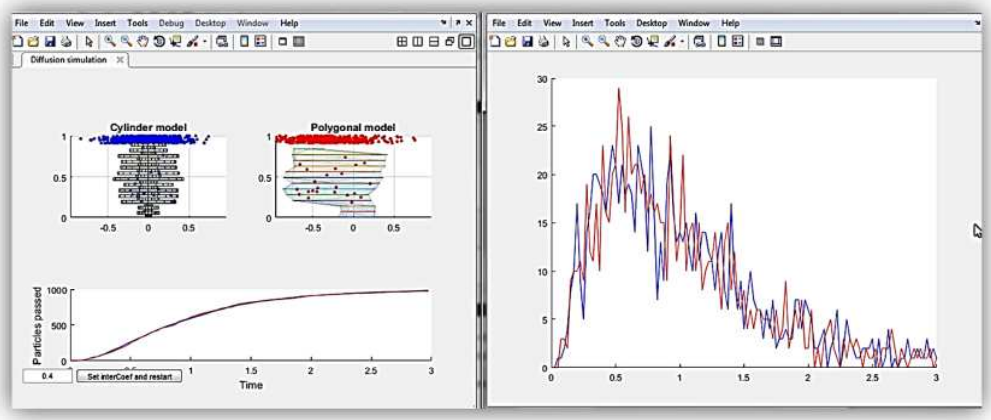


Fig. 13. The result of simulations with decreased transitional diameters shows the nice coincidence of fluxes (see the left-top panel").

In the result, the user needs to recalculate the final statistic of diameters. This modification changes the final statistic of diameters.

The statistic of cylinder's diameters.

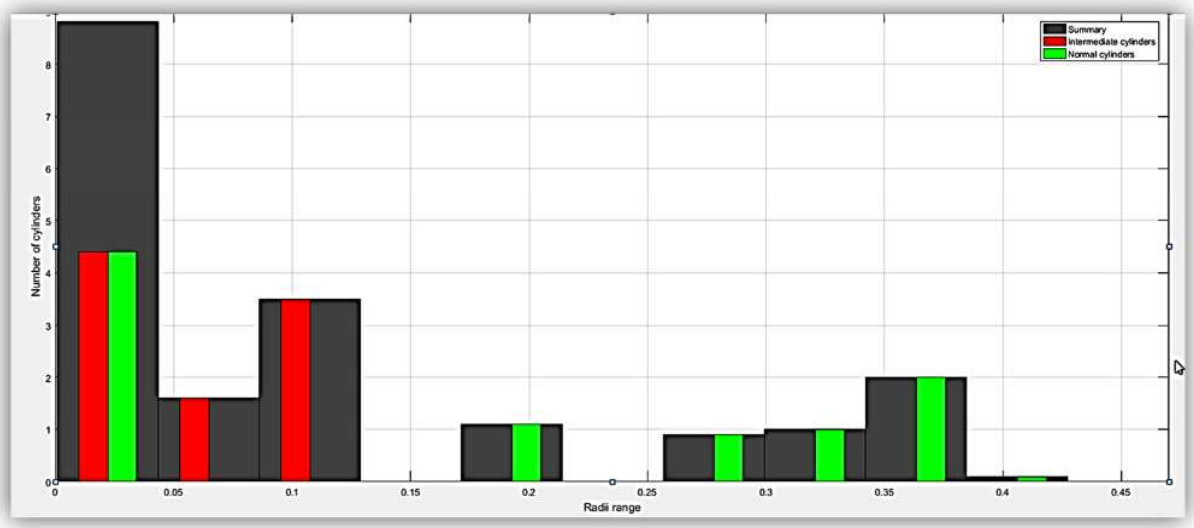


Fig.14. Statistic of two types of diameters: normal (green colour) and transitional (intermediate, red colour). The black bars indicate the mixed statistic of both types of cylinders.

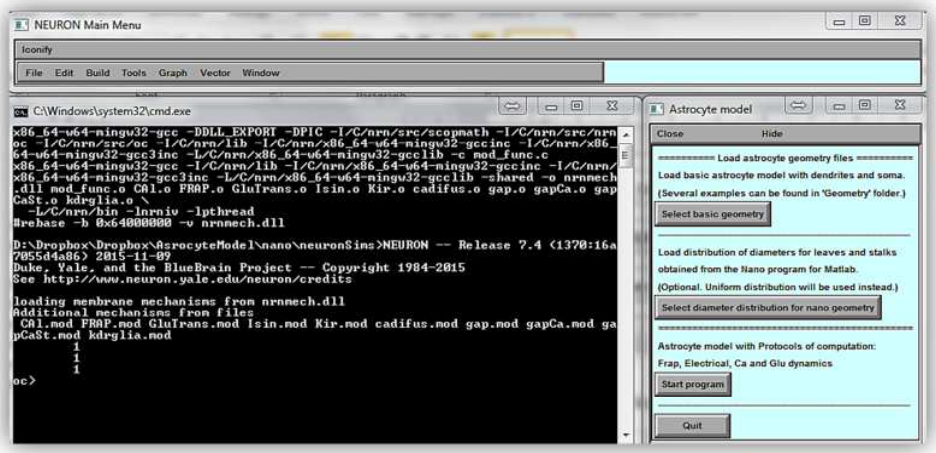
The statistic must be saved into the file for future uploading to NEURON Astrocyte.

Basic Astrocyte simulation in NEURON under Windows.

Preparing "host" local computers (operating under Windows)

1. The user can launch the unit "init.hoc" file located on the host computer in directory …***neuronSims\init.hoc***" or use the panel from the start menu (Fig.1).

2. The file init.hoc activates the following panels:



3. To design a new astrocyte model, first, press the bottom "File of Basic Astrocyte Geometry" and download the file from the directory /Geometry, one of six ready-made dendritic shapes. ***Very important***. The user must download the geometry before start any manipulation with the model. The basic shape of astrocyte "AstroGeometry.hoc" was intensively tested, and the results of these simulations are in the manuscript.
4. The user can create a different topology of dendritic tree using NEURON cell generator or download them from the NEUROMORPHO database (in a format of NEURON).
5. After uploading the basic topology, Astro generates the panel (Fig.15) with the basic astrocyte topology. At the stage, the user can upload one of six different dendrite configurations from the directory ... \Geometry. This topology are schematic without diameters (Fig.15).

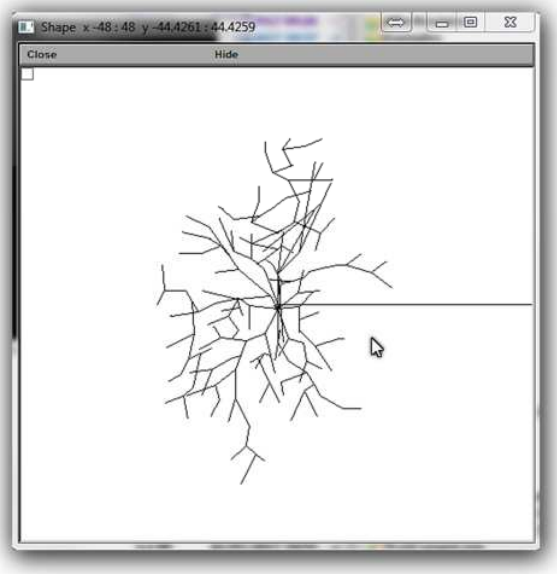


Fig. 15. Panel with basic shape of dendrites.

Select nanostructure

The option “Select nanostructure” lets either to download a file with the statistics of nano diameters produced by “Nano” or to generate a nanostructure of astrocyte by using the built-in tools (See Fig.14 for details). In both cases, the user can adjust nanostructures using preinstalled computational tools.

Basic scenarios of computation

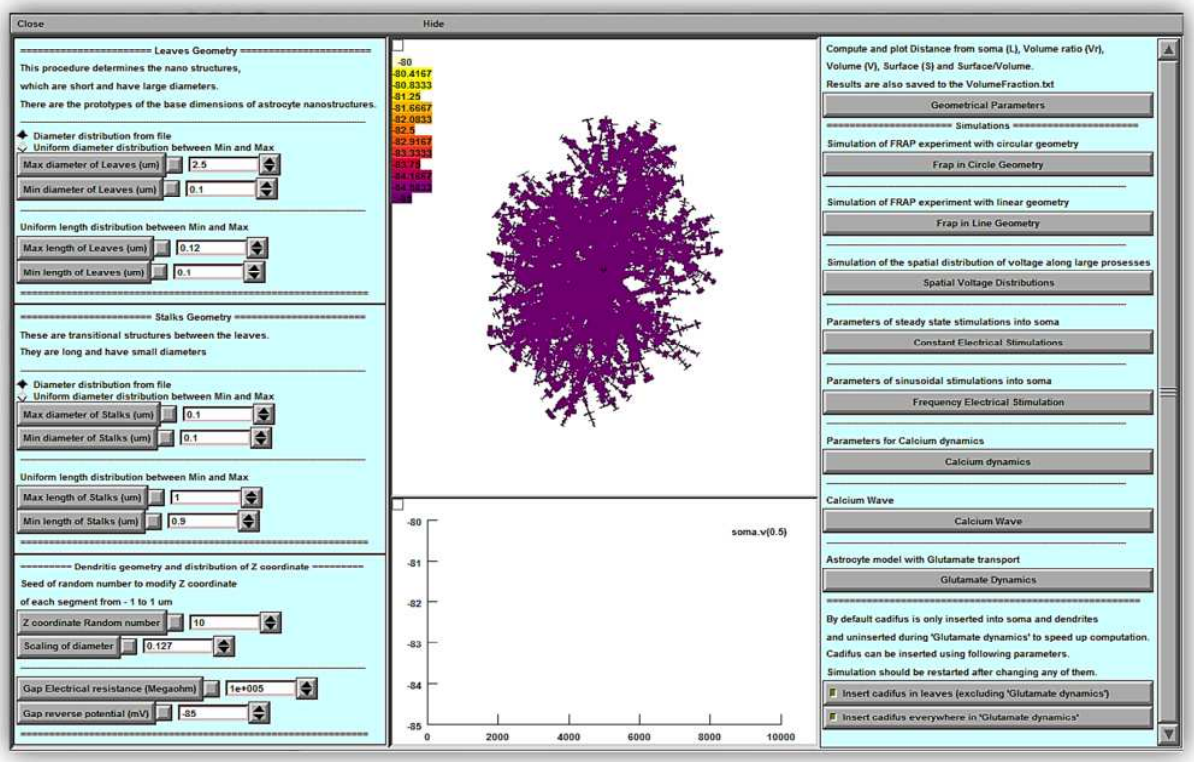
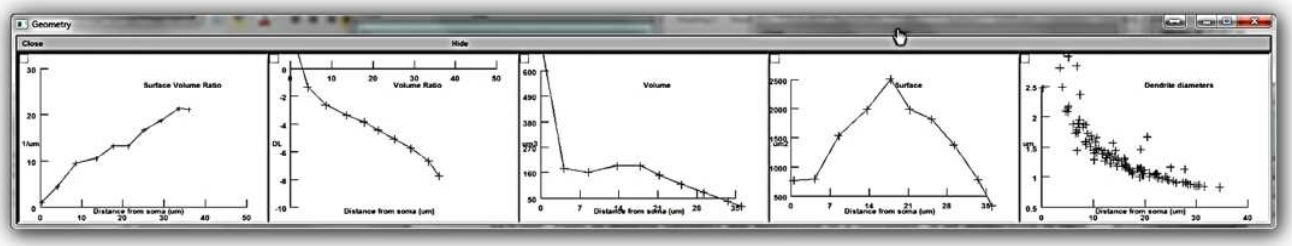


Fig. 16. The central panel of the software “Astro”. The left part of the panel was designed to deal with the geometry of astrocyte. The right part is for scenarios of computations. Two initial graphical areas complete the board, the spatial distribution of the potential over astrocyte (top) and the dynamics of a somatic potential (bottom).

6. Panel "Dendritic Geometry and distribution of Z coordinate" defines the parameters for modification of basic dendritic diameters distribution and Z orientation
    a. "Scaling of Diameter" defines the parameter of diameter scaling. The value of the parameter changes the diameters as a function of distance from the soma. The default value of this parameter is 0.127, which agrees well with the experimental data.
    b. "Z coordinate. Random number". This parameter generates the seed of random number to modify Z coordinate of each segment between - 1 to 1 um

c. "Gap Electrical resistance, Megaohm". The parameter defines the resistance of gap electrical junctions in Megaohm. The preinstall value is so huge, 10^{10} Megaohm, that there is not electrical current through the gap junction.

7. Panel “Repertoire of computations” has four windows: Spatial plot of voltage on the astrocyte model, the plot for the somatic potential and windows with eight different scenarios of computations.
8. The window with Nano parameters allows modifying the nano geometry of astrocyte according to the measuring of the nanostructure in an experiment.
9. Any results of modification of dendritic diameters and nanostructures can be controlled by pressing the button “Compute the Geometrical parameters”. The following windows with the distribution of different geometrical parameters as a function of distance from the soma can be activated :



10. The plotted data also are saved to the file “VolumFraction.txt”.
11. As soon as the user is happy with the total geometry of astrocyte (dendritic and nano), the user can begin to simulate different scenarios of dynamics and to implement the membrane and intracellular biophysical mechanisms.
12. The windows with geometrical parameters can be closed and open whatever user wants it again.

FRAP with circular geometry of photobleaching

13. Using this function, the user can start the simulation of FRAP with a circular bleaching area. After pressing the button “Compute Frap in circle geometry,” the next panel will appear:

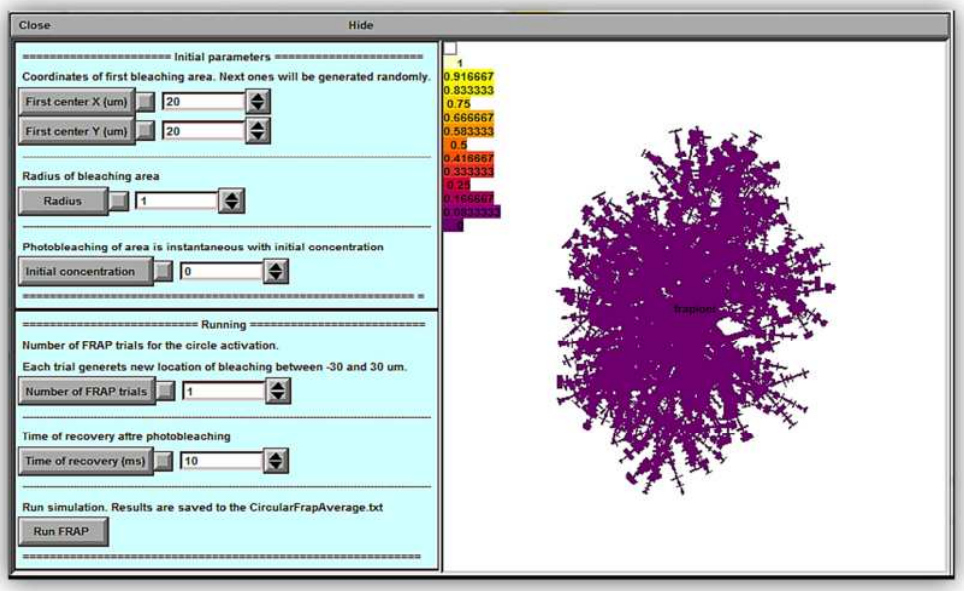


Fig.17. The simulation panel for FRAP with circular geometry. The panel includes the set of parameters (left) and the spatial plot of concentration of non-bleached molecules. See the description of parameters in the text.

14. The following parameters are on the panel (Fig.17).
    a. "First Center X" – X coordinate of centre of bleaching area
    b. "First Center Y" – Y coordinate centre of bleaching area
    c. "Radius" – radius of bleaching area.
15. "A number of trials" defines a number of bleaching trials in various places across the 2D astrocyte area. In this case, the coordinate X and Y are generated stochastically with a uniform distribution between -30 and 30 um.
16. The button "Time of recovery" is a description of a time of recovery after photobleaching. In this option, the discolouration was instantaneous (time = 0 ms) with "the initial concentration".
17. The button "Initial concentration" determines the initial concentration of non-bleached molecules in the area after photobleaching.
18. The button "Run" is for starting of FRAP simulations. The results of any computation are saved to the file "CircularFrapAverage.txt."

FRAP with Linear Geometry

19. The Linear FRAP button is for simulation of FRAP bleaching along the line as a function of time.
20. After lanching the button "Compute FRAP in Line," the following window will appear.

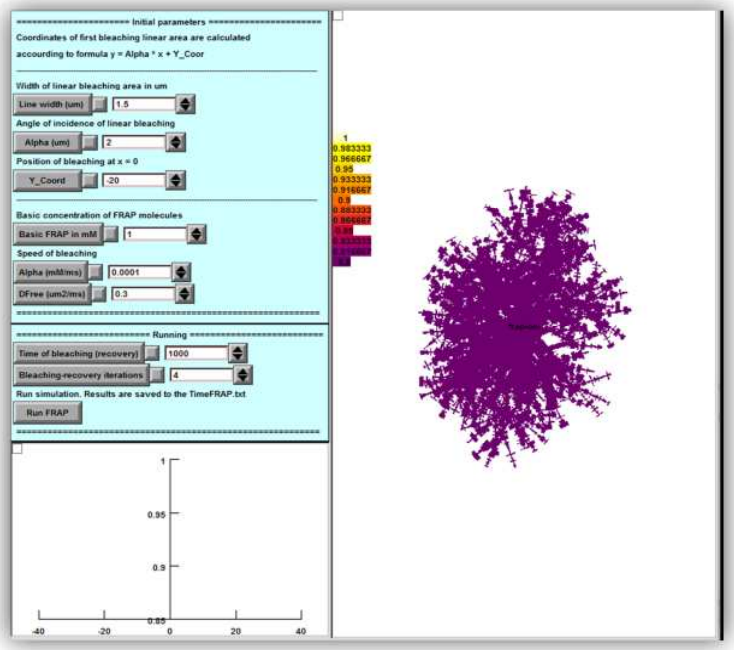


Fig. 18. A panel of line FRAP simulation. See description parameters in the text.

21. Buttons “LineWidth”, “Alpha” and “Y_Corr” set the geometry of the bleaching line.
22. The duration of computation is 8000 ms. Photobleaching is occurring each 1000 ms. The results of computation save to the file “TimeFRAP.txt”
23. The button “Basic concentration” defines the concentration of photobleaching molecules. The value is important to compare with experimental results.
24. The button “Bleaching recovery interactions” defines the number of sequential processes - bleaching – recovery.

The spatial voltage distribution along dendrites

This option of computation is necessary to understand how the geometry of astrocyte influences an electrical communication between dendrites and soma. This procedure computes the distribution of voltage along the dendrites after depolarization of soma by the constant current with amplitude A.

25. The button “Compute the spatial voltage distribution” activates a panel with the plot of voltage as a function of distance from the soma and the set of parameters.

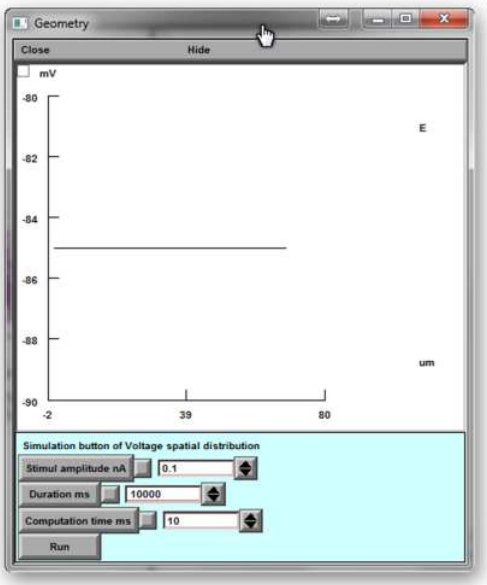


Fig. 19. The panel is for the computation of spatial voltage distribution including the plot and set of parameters (see in the text).

26. The button “Stimulus amplitude” is amplitude of constant depolarisation current in nA injected into the soma.
27. “Durations” defines the time of this current.
28. “Computation time” is a time of computation.
29. “Run” to start the simulation.

Electrical stimulation

30. The button “Electrical stimulation” is the definition of two types of electrical stimulation into the soma. The parameters of these stimuli are from the following table:

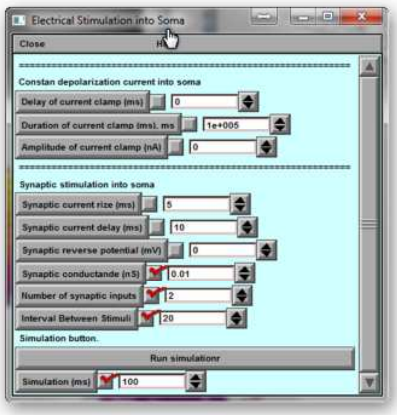
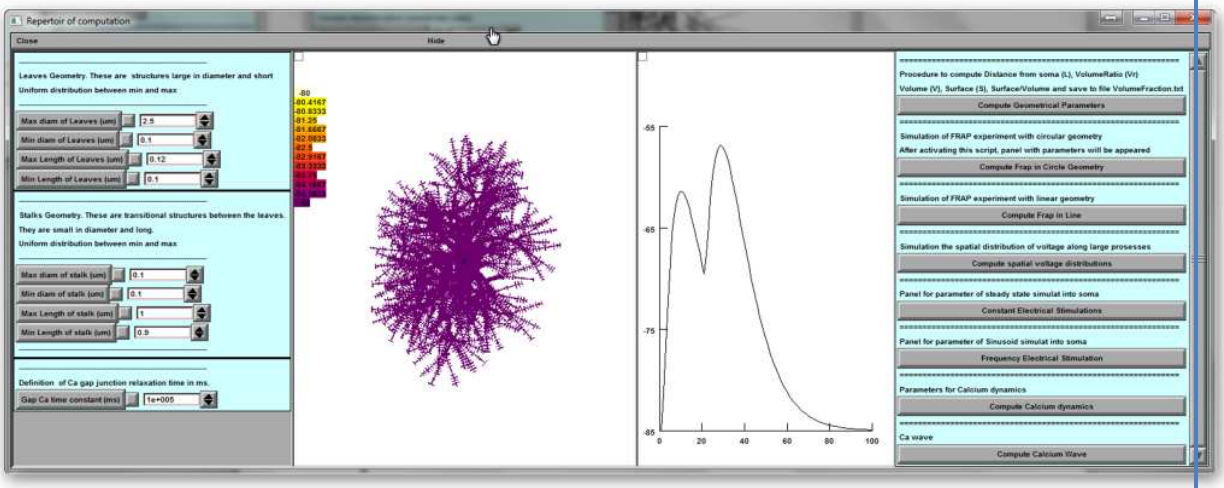


Fig. 20. Constant electrical stimulation. Set of parameters for the simulation of voltage and current dynamics in astrocytes in response to the constant current.

31. First block. Three parameters define the constant current injected into the soma: delay, duration and amplitude.
32. Second block. Five parameters define synaptic current into the soma: rise time, decay time, reverse potential, conductance, the number of synaptic inputs and time between stimuli.
33. The result of the computation is on the main plots with spatial and time scale.



Frequency Electrical stimulation

34. The button “Frequency Electrical stimulations” is the definition of sinusoidal current stimulation into the soma via the following panel:

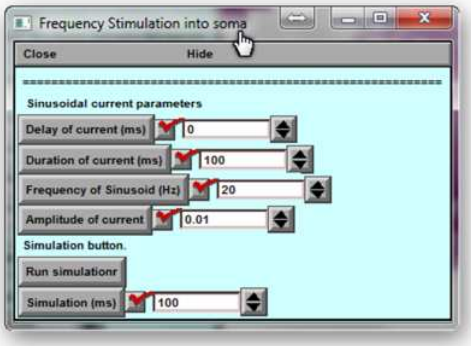
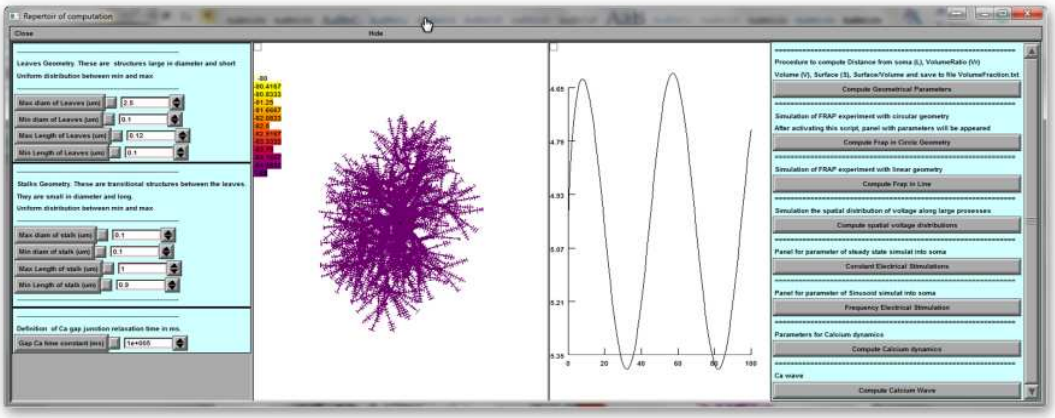


Fig. 21. Set of parameters for the simulation of voltage and current dynamics in astrocytes in response to the Sinusoidal current.

35. The set of parameters describes the sinusoidal current injected into Soma: delay of current, duration of current, a frequency of the current in Hz, the amplitude of current in nA and length of simulations.
36. The main plot shows the results of simulations.



Modelling of Temporal dynamics of calcium

This panel of simulating temporal Ca dynamics.

37. The button “Compute Calcium dynamics” activates 4 new panels

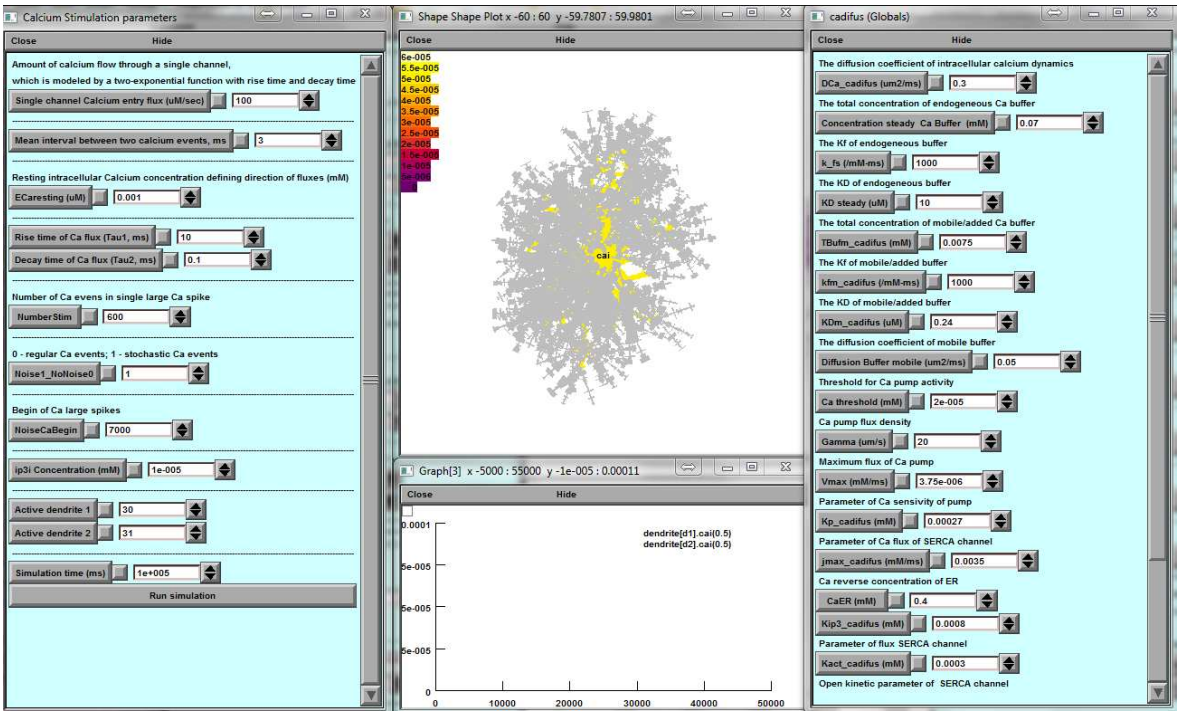


Fig. 22. Panel for Ca dynamic simulation in Astrocyte. The left panel contains the set of parameters which are necessary for the description of single-channel calcium dynamics. The right panel is for a set of parameters describing the local calcium mechanism. The middle plots are for the visualization of spatial (top) and temporal in dendrites d1 and d2 (bottom) calcium dynamics. The numbers of dendrites are indicated by the left panel.

38. First-panel “Calcium stimulation parameters” define the small calcium fluxes through the plasmatic membrane with the following set of parameters
    a. “Single channel calcium entry flux” describes the flux of calcium through a single channel.
    b. “Mean interval between two calcium events” defines the average time interval between two single Ca events.
    c. “ECaresting” is a Ca concentration of linear approximation of Ca fluxes and defines the reverse Ca concentration of flux from inside to outside cell.
    d. “Rise” and “Decay” calcium unitary current of a single event.
    e. “Number of stimuli” defines the number of singles Ca events in single large Ca spike.
    f. “NoiseorNotNoise! It can be either 0 or 1 and defines either random calcium events or regular Ca events.
    g. “NoiseCaBegin” defines the time of the large Ca spike generated.
    h. “ip3 concentration” is an initial concentration of ip3 ions.
    i. “Active dendrite 1 and 2” defines to dendrites where the Ca event happened and recordings on the Plot of Ca dynamics provided.
    j. “Stimulation time” is a time of cell dynamics. Usually, Ca dynamics is long time process and needs at least 100 seconds of computation.
39. Spatial plot defines the spatial dynamics of Calcium.
40. Time plot defines the spatial dynamics of Ca in two dendrites with the coordinates of the area defined inside the panel of an initial set of parameters.
41. A panel of Ca dynamics mechanism.
    a. Panel "The diffusion coefficient of intracellular calcium dynamics."
    b. Panel "The total concentration of endogenous Ca buffer."
    c. Panel "The Kf of endogenous buffer (/mM-ms)"
    d. Panel "The KD of endogenous buffer (uM)"
    e. Panel "The total concentration of mobile/added Ca buffer"
    f. Panel "The Kf of mobile/added buffer (/mM-ms)"
    g. Panel "The KD of mobile/added buffer (uM)"
    h. Panel "The diffusion coefficient of the mobile buffer."
    i. Panel "Threshold for Ca pump activity."
    j. Panel "Ca pump flux density."
    k. Panel "Maximum flux of Ca pump."
    l. Panel "Parameter of Ca sensitivity of pump."
    m. Panel "Parameter of Ca flux of SERCA channel."
    n. Panel "Ca reverse the concentration of ER."
    o. Panel "Parameter of IP3 flux."
    p. Panel "Parameter of flux SERCA channel."
    q. Panel "Open kinetic parameter of SERCA channel."
    r. Panel "Close kinetic parameter of SERCA channel."
    s. Panel "dimensionless numeric value of vrat[i] equals the volume")
    t. Panel "of annulus i of a 1um diameter cylinder multiply by diam^2 to get volume per um length."

Ca wave simulations

42. This function calls the following panels

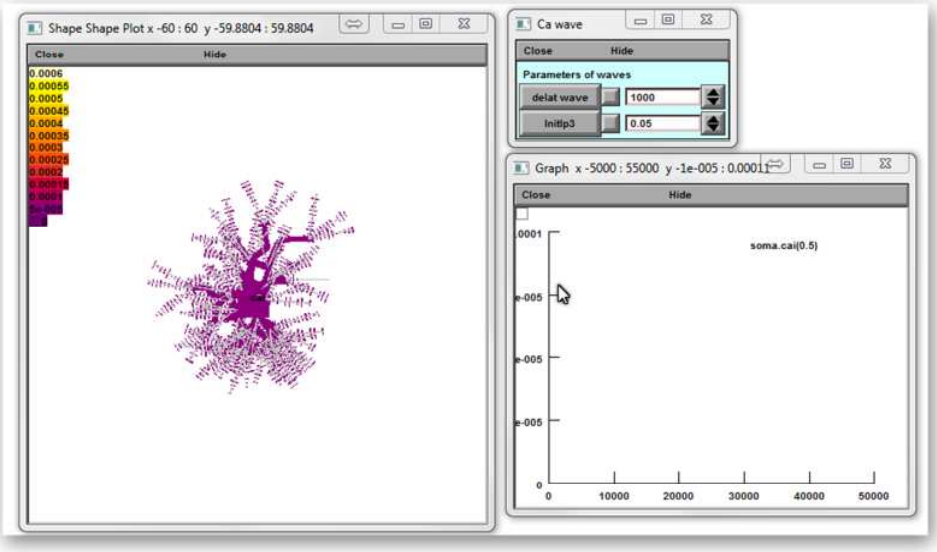


Fig. 23. Panel for Ca Wave dynamic in astrocyte includes spatial plot, parameters and comatic ca dynamics.

43. Panel Ca wave. Calcium wave is caused by an instant increase in time 1000 ms in concentration of ip3=0.05 uM in soma to a level indicated on the panel
44. Simulation time is 10000 ms. New simulation can start ONLY after finishing this simulation.
45. Spatial plot and time plot indicate of the Ca dynamics.

Basic computation with Button "Astrocyte Model with Glutamate transport."

This model has only a single configuration for the study of Astrocyte Glutamate transporters dynamics after uncaging of Extracellular Glutamate.

The typical panels of the configuration are :

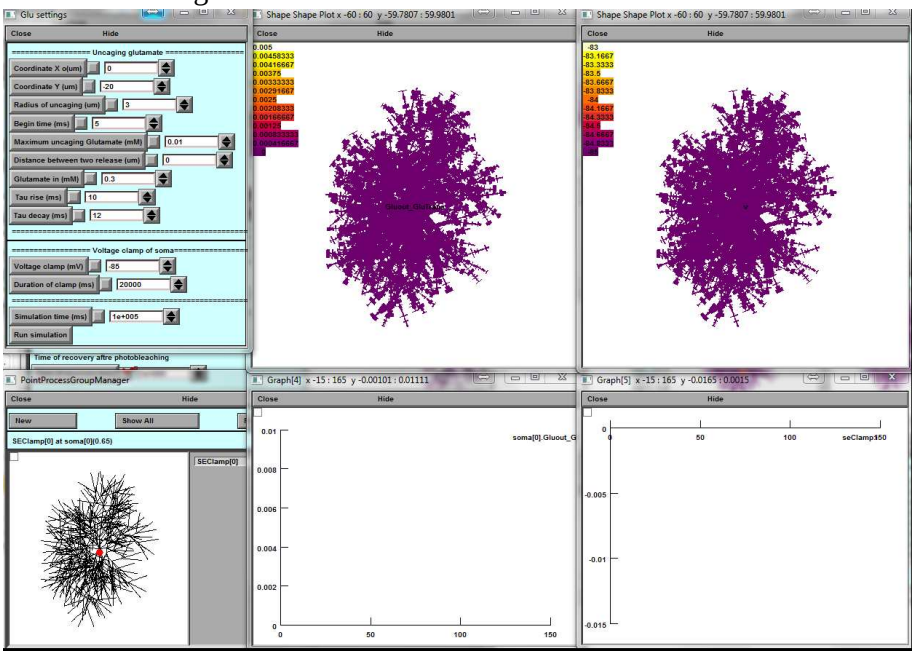


Fig. 24. Set of panels with parameters and plots were designed for the simulation of dynamics of a current of glutamate transporters distributed across the astrocyte. The responses of transporters were due to a glutamate uncaging.

1. All computations of glutamate transporters dynamics are designed in “Voltage Clamp configuration” with parameters are indicated on the panel “Point Process Group Manager”: resistance of the clamp electrode, duration of the clamp and voltage of the clamp.
2. The voltage clamp current on the soma after uncaging of glutamate on the astrocyte is shown on the left top panel.
3. The parameters of uncaging: uncaging inside the circle with Coordinate X and Y and radius, the uncaging begin in time > 3 ms to stabilise the Glutamate pumps. The uncaging is defined with the rise and decay time and maximum concentration of glutamate. The user can also add another simultaneous uncaging spot modifying the distance between first uncaging spot.
4. Two spatial plots with voltage and Extracellular Glutamate indication.

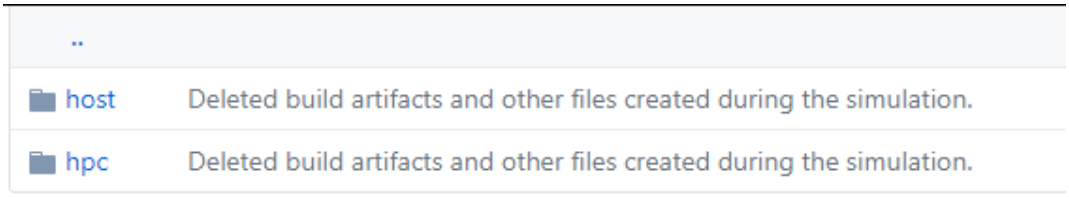
Preparing HPC (OS Linux) cluster for running Calcium simulation

1. Install MPI (Message Passing Interface) on the cluster (most of the clusters have it already preinstalled). Free MPI software can be downloaded from https://www.open-mpi.org/software/ompi/v3.0

2. Install NEURON on the cluster. The latest version can be downloaded from the official site. We recommend using the installation from source code taking the sources from here https://www.neuron.yale.edu/neuron/download/getstd and following steps 1-5 of the next instruction https://www.neuron.yale.edu/neuron/download/compile_linux.
   --with-paranrn option should be added to the configure command for NEURON installation to enable distributed computations.
   NEURON GUI is not required by cluster simulation. If the user wants to remove it from installation, don't download *iv-mm.tar.gz* archive and replace --with-iv=$HOME/neuron/iv with –without-iv when calling configure for NEURON installation.
3. Put the *clusterCaSim/hpc* folder to the place shared between cluster nodes. For example, it can be saved in the directory */home/<username>*. The cluster setup is done.

The user can use PuTTY and WinSCP programs to work with console and file system of the remote cluster. Both programs are free.

The structure of the directors, in GitHub, necessary for working with the cluster has the form:



All files from directory HPC should be downloaded on the cluster, keeping the structure of directories unchanged.

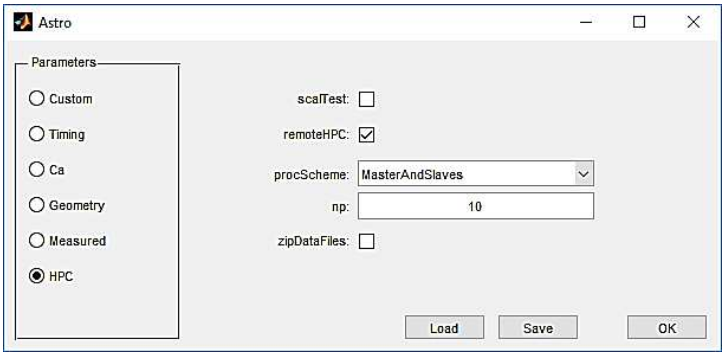
All files from directory HOST should be downloaded on the local computer (OS Windows), keeping the structure of directories unchanged.

Preparing client (OS Windows) machine for running Calcium simulation

1. Open *clusterCaSim/host/scripts/win-lin/params.bat* and set your cluster connection parameters including the path to the *hpc* folder located in the cluster.
2. Open *clusterCaSim/host/core/BasicParams.m* and fill avaliableNodes variable with names of your cluster nodes.
   Also, fill *clusterCaSim/hpc/hostfile_BusyMaster and hostfile_IdleMaster* files with node names in the following manner: each line should contain the name of the node followed by " max_slots=1" without quotes.

Contents of both files should be the same except that *hostfile_IdleMaster* should not include the master node (only slaves).

3. After launching the program, the user can modify following parameters on the *HPC* panel:



scalTest – check if the scalability test is useful here. This test shows how well execution time scales relative to the number of processes
remoteHpc – uncheck if the software will be run on the client PC. The user should also modify NRNDIR and HPCDIR parameters that point to the NEURON and *hpc* folder locations on client PC in the *clusterCaSim/host/scripts/win-win/params.bat* and *params.sh.*
procScheme – processor distribution scheme
np – number of processors

If user compiled executables to run the simulation, then the after all the user should recompile them after changing any parameters in Matlab files using *clusterCaSim/host/BUILD_AllHostExecutables.m.*

Notes about simulations

Nano geometry:
There should be no layers without points for diffusion simulation to work properly.
Because of a significant amount of computations in complex diffusion simulation Matlab cannot handle stopping or restarting it by pressing a button.

Calcium dynamics on the cluster:
There are situations when selected geometry cannot be split into the specified number of processors. In this case, the user will see MPI error before the computations begin. To solve the problem, the user can simply increase or decrease the number of processors.

Dendrites should be connected to the soma only in the 1 position. Otherwise cell splitting fails.
Examples:
Good: soma[0] connect dendrite[125](0), 1
Wrong: soma[0] connect dendrite[125](0), 0.5
Wrong: soma[0] connect dendrite[125](0), 0.1
Wrong: soma[0] connect dendrite[125](0), 0