

# USER GUIDE

---

*BRAINCELL 1.0.*

*Brain cell *in silico**

© University College London, MIT licence

# TABLE OF CONTENTS

## Contents

INTRODUCTION .....	6
BrainCell — First-Time Installation (Windows) .....	Error! Bookmark not defined.
⚡ Before you start .....	Error! Bookmark not defined.
Part A — If NEURON & Anaconda are already installed .....	Error! Bookmark not defined.
Part B — If you don't have NEURON / Anaconda .....	Error! Bookmark not defined.
🔧 Troubleshooting .....	Error! Bookmark not defined.
Experimental Data or Approximations Needed to Build a Realistic Cell Model .....	23
GETTING STARTED .....	23
Installing and Launching BrainCell .....	23
Starting BrainCell .....	24
Astro/ Base. Setting up and running BRAINCELL: Astrocyte configuration. ....	27
GENERATING COMPLETE ASTROCYTE MORPHOLOGY .....	27
Launching the Astrocyte Module .....	27
Generating or Downloading the Astrocyte Stem Tree .....	28
Option 1 — Select Library Stem Tree .....	28
Option 2 — Load from Local Directory .....	28
Option 3 — Load a Reconstructed or Endfoot-Enhanced Tree .....	29
Option 4 — Add Nano-Geometry to the Astrocytic Tree .....	29
Finalising and Preparing the Astrocyte Geometry .....	30
Generating astroglial morphology on the nanoscale .....	31
The geometry of nanoscopic processes .....	31
Populating astrocyte tree with nanoscopic processes .....	31
Tissue-filling properties of astroglial morphology .....	33
"Geometric parameters" button .....	34
Nano Astro — Loading a Pre-existing Astrocyte Morphology .....	35
<b>Loading a Pre-existing Astrocyte (“Astro + Nano”)</b> .....	35
Neuron / Base. Setting up and running BRAINCELL: Neuron module. ....	37
IMPORTING AND GENERATING COMPLETE NEURON MORPHOLOGY .....	37
Building a 3D Neuron Model .....	37
Navigating Neuron Simulation: Exploring Main Windows for Enhanced Analysis .....	39
<b>Key Features of the Main Neuron Window</b> .....	39
Customising Neuronal Connectivity: Spine Set Configuration .....	41
Configuring Spine Sets .....	41
How to Use the Spine Sets Feature .....	41

The editing of spine locations.....	42
Spine Seeding Panels.....	42
“Where to Seed the Nanostructures” Tab (Fig. 11) .....	43
The spine, head and neck geometry. Location over dendritic tree.....	44
Configuring Spine Distributions .....	44
“Nano Neuron”. Download the previously created neuron model.....	46
Manager of biophysical mechanisms.....	47
Manager of Biophysical Mechanisms .....	47
Windows B & C — Managing Mechanisms and Cell Parts .....	48
Split, Merge and Remain option. ....	49
Splitting Cell Compartments .....	49
Merging Cell Compartments .....	50
Renaming Cell Compartments .....	51
Adjust the spatial distribution of mechanisms. ....	53
Synapse distribution. ....	53
Main Interface for Editing Biophysical Mechanisms.....	53
Spatial Inhomogeneity of Biophysical Mechanisms Editor.....	54
Spatial Inhomogeneity Specification.....	54
Visualising Spatial Heterogeneity.....	55
Adjust the stochastic distribution of mechanisms.....	56
Stochasticity Editor .....	57
Manager of synapses. ....	59
Synaptic Panel .....	59
Synapse Location Panel.....	61
Synaptic Structure — Synaptic Manager (Top Part) .....	62
Synaptic Manager — Middle & Bottom Sections .....	62
Spatially - non-uniform sources of extracellular ions. ....	64
Ion Concentration Modelling .....	64
Outside-In Diffusion .....	65
Extracellular Ion Concentration Editor — Overview (Fig. 19) .....	66
Panels & Tools .....	67
Summary .....	67
<b>Outside-In / Inside–Out Diffusion Equations .....</b>	<b>68</b>
<b>1. Fundamental Diffusion Equation .....</b>	<b>68</b>
<b>2. Instantaneous Point-Source Solution .....</b>	<b>68</b>
<b>3. On–Off (Time-Limited) Source .....</b>	<b>68</b>
<b>4. Time-Factor Function.....</b>	<b>69</b>

<b>5. Combined Expression for Extracellular Concentration</b>	69
<b>6. Parameter Notes</b>	69
<b>Summary</b>	69
Manager of Extracellular Sources Panel	70
Single Ion Panel	70
Inside–Out Diffusion	71
Summary	74
Gap Junctions in BrainCell	75
Parameters of gap junctions in Neurons and Astrocytes	77
Exporting and Importing Biophysical Mechanisms	80
Export cell model	82
Model Export Panel	83
Structure of export file	83
BrainCell Software: Model Export. Result of Simulation Parameter Selection	84
Model Export Feature	84
Exporting Computation Run Parameters	85
Summary	86
Managing Computation Results	87
Summary	88
Re-Importing Cell Models	89
Importing External Cell Models from NEURON	90
Simulation modes: Examples of cell simulation inside of BrainCell	91
CA1-Neuron Voltage Simulation (Neuron mode only)	92
Action Potential Dynamics Calculation	92
Adapting Cell Sections	92
Visualising Electrical Dynamics	92
Membrane voltage distribution	94
Exploring Constant Electrical Stimulation	95
Exploring Variable Electrical Stimulation	95
Linescan FRAP experiment: probing intracellular connectivity of Brain Cell	96
FRAP Simulation Overview	96
RAP Panel Controls	96
Bleaching & Recovery Cycles	97
Running FRAP	97
Visualisation & Interaction Tips	97
Notes	97
FRAP with round spot bleaching	98

<b>Ca<sup>2+</sup> Wave Simulations .....</b>	100
<b>Core Calcium Dynamics Equations.....</b>	100
<b>Summary.....</b>	101
<b>Ca<sup>2+</sup> Wave Panel Controls (Fig. 31a).....</b>	102
<b>Ca<sup>2+</sup> Reaction–Diffusion Parameter Settings.....</b>	103
<b>Running &amp; Restarting Simulations .....</b>	103
Simulating microscopic Ca <sup>2+</sup> events .....	104
<b>Calcium Stimulation Parameters Panel.....</b>	104
Simulating Glutamate Transporters.....	105
<b>Important Notes for Glutamate Uncaging Simulations.....</b>	107
<b>Voltage Clamp Configuration.....</b>	107
<b>Monitoring Panels.....</b>	107
<b>Simulating Potassium Dynamics (Intra- and Extracellular).....</b>	110
<b>Important Notes on Potassium Dynamics Simulation .....</b>	111
<b>Monitoring Panels.....</b>	112
<b>Potassium Settings Panels .....</b>	113
Basket cell GABA diffusion (Neuron mode only) .....	114
<b>Building and Analysing Realistic Basket Cell Models with GABA Release Dynamics .....</b>	114
Axon Myelination (Neuron Mode Only).....	117
<b>Myelinated Axon Simulation Panel.....</b>	117
LFP. Recording local field potential for a single cell.....	128
<b>Local Field Potential (LFP) Simulation .....</b>	128
Submitting BrainCell Export Files to the NSG (Neuroscience Gateway).....	131
<b>Configure and Run a Task .....</b>	132
<b>Linux MOD File Requirements .....</b>	133
<b>BrainCell Testing Suite.....</b>	Error! Bookmark not defined.
<b>Available Test Modules .....</b>	Error! Bookmark not defined.
<b>Running Tests.....</b>	135
<b>Creating and Editing Biophysical Mechanisms .....</b>	137
<b>Where the JSON Files Are Located .....</b>	137
<b>Creating a New JSON Mechanism .....</b>	137
<b>Editing an Existing JSON Mechanism .....</b>	138
<b>Available Built-In Mechanisms.....</b>	138
<b>Best Practices.....</b>	138
<b>Applying Your Changes .....</b>	139

## INTRODUCTION

# Getting Started with BrainCell

## Welcome to the BrainCell User Manual

**BrainCell** is a next-generation simulation platform for constructing and analysing **multi-scale models of neurons and astroglia**. It enables the creation of realistic, **multi-compartmental biophysical representations** of brain cells, linking detailed morphology with their physiological function.

Built on the **NEURON–Python computational environment**, BrainCell integrates cellular morphology, membrane dynamics, and intracellular mechanisms into a unified workspace. With this tool, researchers can explore how geometry, ion transport, and cellular interactions shape neuronal and glial behaviour across spatial and temporal scales.

The primary goal of BrainCell is to facilitate the **mechanistic interpretation of experimental data**, allowing users to reproduce, test, and predict complex cellular processes within a biologically grounded computational framework.

This manual provides a step-by-step guide to BrainCell’s features, interface, and workflow — helping you make the most of its modelling capabilities and apply them effectively to your research.

---

## System and Software Requirements

To ensure optimal performance, verify that your system meets the following requirements before installing BrainCell:

**Python (Anaconda)** —Anaconda 2023.09 (Windows, x64): Download from

[https://neuroalgebra.net/assets/Anaconda3-2023.09-0-Windows-x86\\_64.exe](https://neuroalgebra.net/assets/Anaconda3-2023.09-0-Windows-x86_64.exe)

NEURON 8.2.2 (Windows, mingw, Python 3.7–3.11): Download from

<https://neuroalgebra.net/assets/nrn-8.2.2-0-setup.exe>

Ensure that NEURON’s Python environment script (`C:\nrn\bin\nrnpyenv.bat` on Windows) correctly links to your Anaconda installation.

*Note:* NEURON uses the `%APPDATA%` environment variable, which is user-specific — therefore, Anaconda must be installed **for the current user only**, otherwise NEURON will not detect it.

1. **Operating System**
  - Windows 10 or later (64-bit)

# BrainCell Installation Guide

## Choose Your Starting Point

Before installing **BrainCell**, determine which of the following three categories applies to you. Each category has its **own correct installation path**.

---

## About This Manual

This manual guides you through installing BrainCell, a computational neuroscience simulation tool. BrainCell requires NEURON (simulation environment) and Python (Anaconda recommended) to run.

**Platforms covered:** Windows 10/11, macOS 10.15+, Linux (Ubuntu 20.04+, Debian, Fedora)

**Estimated installation time:** 30-60 minutes for clean installation

---

## Quick Start: Identify Your Category

Choose the category that matches your current system:

Category	Description	When to Use
<b>Category 1</b> NEURON + Python already working	You run NEURON simulations regularly	
<b>Category 2</b> Clean system (nothing installed)	Fresh OS or new to NEURON	
<b>Category 3</b> Mixed/incomplete installation	⚠️ NEURON or Python partially installed	

**⚠️ CRITICAL:** If you're in Category 3 (mixed environment), **DO NOT** attempt to patch your installation. Follow the clean installation process—it's faster and more reliable.

---

## System Requirements

### Minimum:

- **RAM:** 4 GB (8 GB recommended)
  - **Disk Space:** 5 GB free
  - **Processor:** 64-bit processor
  - **Internet:** Required for initial download
  - **BrainCell:** Latest version (cloudpackage-v1.zip)
-

# Category 1 — Existing NEURON + Python Users

(Experienced users with working environment)

## Who belongs here?

You are in Category 1 if **ALL** of the following are true:

- NEURON is already installed and launches successfully
- Python (Anaconda or system Python) is installed
- NEURON recognizes Python (you can run simulations via Python)
- You actively run NEURON simulations

If any of these are false, go to Category 2 or Category 3.

---

## Installation Steps

### Step 1: Download BrainCell

#### All Platforms

1. Visit: <https://neuroalgebra.net>
2. Download: `cloudpackage-v1.zip`
  - o Forum thread: <https://forum.neuroalgebra.net/viewtopic.php?t=3>
  - o **Note:** The ZIP file is password-protected. The password is available on the forum thread above.
3. Create a directory for BrainCell:

#### Windows:

`C:\braincell`

#### macOS/Linux:

`~/braincell`

4. **Unblock the downloaded file (important!):**

#### Windows:

- Right-click `cloudpackage-v1.zip` → Properties → Check "Unblock" → Apply

#### macOS:

```
xattr -d com.apple.quarantine ~/Downloads/cloudpackage-v1.zip
```

### **Linux:**

- No action needed (file blocking not applicable)
5. Extract the ZIP file into your BrainCell directory, entering the password when prompted.
- 

## **Step 2: Compile NEURON Mechanisms (If Needed)**

### **⚠ Only perform this step if:**

- You added new .mod files, OR
- You modified existing mechanism files

### **Windows**

Double-click:

```
build_mechs.bat
```

**If compilation fails:** Install Microsoft C++ Build Tools (x64) from:  
<https://visualstudio.microsoft.com/downloads/> (scroll to "Build Tools for Visual Studio")

Then run `build_mechs.bat` again.

### **macOS**

Open Terminal in the BrainCell directory and run:

```
chmod +x build_mechs.sh  
./build_mechs.sh
```

**If compilation fails:** Install Xcode Command Line Tools:

```
xcode-select --install
```

### **Linux**

Open Terminal in the BrainCell directory and run:

```
chmod +x build_mechs.sh  
./build_mechs.sh
```

**If compilation fails:** Install build essentials:

### **Ubuntu/Debian:**

```
sudo apt-get update  
sudo apt-get install build-essential
```

### **Fedor/RHEL:**

```
sudo dnf install gcc gcc-c++ make
```

---

## Step 3: Platform-Specific Setup

### Windows: Unblock Scripts

Right-click each of the following files → Properties → Check "Unblock" → Apply:

- init.bat
- build\_mechs.bat (if you used it)

### If Windows blocks execution:

Open PowerShell and run:

```
Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned
```

### macOS: Make Scripts Executable

Open Terminal in the BrainCell directory:

```
chmod +x init.sh  
chmod +x build_mechs.sh
```

### Linux: Make Scripts Executable

Open Terminal in the BrainCell directory:

```
chmod +x init.sh  
chmod +x build_mechs.sh
```

---

## Step 4: Launch BrainCell

### Windows

Double-click:

```
init.bat
```

### macOS/Linux

Open Terminal in the BrainCell directory and run:

```
./init.sh
```

### Or, using NEURON directly:

```
nrngui init.hoc
```

---

## Verification

After launch, you should see:

1. A console window with NEURON initialization messages
2. The BrainCell main window
3. Options to select **Astrocyte** or **Neuron** mode

**If you see this, installation is complete!**

---

## Category 2 — Clean System Installation

(No NEURON or Python installed)

### Who belongs here?

You are in Category 2 if:

- NEURON is **not** installed
- Python/Anaconda is **not** installed
- Fresh OS or minimal system

**Typical users:** New users, fresh installations, dedicated simulation machines

---

### Recommended Stable Configuration

For reliable operation across all platforms:

- **Anaconda:** 2023.09 or later
- **NEURON:** 8.2.2 or later (with Python 3.7–3.11)

**Discussion and updates:** <https://forum.neuroalgebra.net/viewtopic.php?p=171#p171>

---

## Installation Steps

### Step 1: Install Anaconda

#### Windows

1. Download Anaconda 2023.09:
  - Direct link: [https://neuroalgebra.net/assets/Anaconda3-2023.09-0-Windows-x86\\_64.exe](https://neuroalgebra.net/assets/Anaconda3-2023.09-0-Windows-x86_64.exe)
2. Run the installer:
  - Select "**Just Me**" (not "All Users")
  - Use default installation path

- Check "Add Anaconda to PATH" (important!)
- Accept other default options

### 3. Restart Windows

#### macOS

1. Download Anaconda (Intel or Apple Silicon):
  - Visit: <https://www.anaconda.com/download>
  - Choose the appropriate installer for your Mac
2. Open the downloaded .pkg file and follow the installer
3. Open Terminal and verify installation:

```
conda --version
python --version
```

4. If conda is not recognized, add to PATH:

```
echo 'export PATH="$HOME/anaconda3/bin:$PATH"' >> ~/.zshrc
source ~/.zshrc
```

(Use `~/.bash_profile` if you're using bash instead of zsh)

#### Linux

1. Download Anaconda:

```
wget https://repo.anaconda.com/archive/Anaconda3-2023.09-0-Linux-x86_64.sh
```

2. Install:

```
bash Anaconda3-2023.09-0-Linux-x86_64.sh
```

3. Follow the prompts:

- Accept the license
- Confirm installation location (default: `~/anaconda3`)
- **Yes** to initializing Anaconda

4. Restart terminal or run:

```
source ~/.bashrc
```

5. Verify:

```
conda --version
python --version
```

---

## Step 2: Install NEURON

#### Windows

1. Download NEURON 8.2.2:
  - Direct link: <https://neuroalgebra.net/assets/nrn-8.2.2-0-setup.exe>
2. Run the installer:

- Use default installation options
- Install to a simple path (e.g., C:\nrrn)
- The installer includes Python interfaces for Anaconda

### 3. Restart Windows

4. Verify installation by opening Command Prompt:

```
nrniv -python
```

You should see the NEURON prompt. Type `quit()` to exit.

### macOS

1. Download NEURON for macOS:
  - Visit: <https://neuron.yale.edu/neuron/download>
  - Choose the .pkg installer for your architecture (Intel/Apple Silicon)
2. Run the installer and follow prompts
3. Verify installation:

```
nrniv -python
```

**If not recognized**, add to PATH in `~/.zshrc` (or `~/.bash_profile`):

```
export PATH="/Applications/NEURON/nrn/x86_64/bin:$PATH"
```

Then reload:

```
source ~/.zshrc
```

### Linux

#### Ubuntu/Debian:

```
sudo apt-get update
sudo apt-get install neuron
```

#### Or compile from source:

1. Install dependencies:

```
sudo apt-get install build-essential libncurses-dev libreadline-dev \
libx11-dev libxt-dev bison flex automake libtool
```

2. Download NEURON source:

```
wget https://github.com/neuronsimulator/nrn/releases/download/8.2.2/nrn-
8.2.2.tar.gz
tar -xzf nrn-8.2.2.tar.gz
cd nrn-8.2.2
```

3. Configure and compile:

```
./configure --prefix=$HOME/neuron --with-nrnpython=python3
make
make install
```

4. Add to PATH in `~/.bashrc`:

```
export PATH="$HOME/neuron/x86_64/bin:$PATH"  
export PYTHONPATH="$HOME/neuron/lib/python:$PYTHONPATH"
```

5. Reload:

```
source ~/.bashrc
```

6. Verify:

```
nrniv -python
```

---

### Step 3: Download and Setup BrainCell

Follow the same steps as **Category 1, Steps 1-3** for your platform.

---

### Step 4: Compile Mechanisms

Run the build script for your platform (see Category 1, Step 2).

For clean installations, this step is **usually required** to compile the included mechanism files.

---

### Step 5: Launch BrainCell

Follow **Category 1, Step 4** for your platform.

---

#### Verification

You should see:

1. Console window with initialization messages
2. BrainCell main window
3. Astrocyte/Neuron mode selection

**Installation complete!**

---

## Category 3 — Mixed or Incomplete Environment

## **⚠ CRITICAL: Most Common Failure Scenario**

### **Who belongs here?**

You are in Category 3 if **ANY** of the following apply:

- ⚠ Python/Apache is installed, but NEURON is not
- ⚠ NEURON is installed, but Python is not
- ⚠ NEURON cannot find Python
- ⚠ Multiple Python versions exist on your system
- ⚠ Previous installation attempts failed
- ⚠ NEURON launches but BrainCell doesn't work

**This is the most common cause of installation problems.**

---

## **⚠ CRITICAL RECOMMENDATION**

### **✗ DO NOT:**

- Attempt to patch your current installation
- Manually edit PATH variables
- Install additional Python versions
- Try to "fix" NEURON/Python integration manually

### **✓ DO THIS INSTEAD:**

**Follow the clean installation process below.**

Clean installation is **faster and more reliable** than troubleshooting mixed environments.

---

## **Clean Installation Process**

### **➊ Step 1: Remove ALL Existing Installations (CRITICAL)**

**This step is mandatory. Skipping it will cause problems.**

#### **Windows**

1. **Uninstall NEURON:**
  - Settings → Apps → Find "NEURON" → Uninstall
2. **Uninstall Anaconda/Python:**
  - Settings → Apps → Find "Anaconda" or "Python" → Uninstall all versions
3. **Remove remaining directories:**

- Delete: C:\nrn (if exists)
  - Delete: C:\Users\[YourName]\anaconda3 (if exists)
  - Delete: C:\Users\[YourName]\AppData\Local\Programs\Python (if exists)
4. **Clean PATH variable:**
    - Press Win, type "environment variables"
    - Select "Edit the system environment variables"
    - Click "Environment Variables"
    - In both "User variables" and "System variables", edit PATH
    - Remove any entries containing "python", "anaconda", or "nrn"
  5. **Restart Windows** (critical!)

## macOS

### 1. Remove Anaconda:

```
rm -rf ~/anaconda3
rm -rf ~/.conda
```

### 2. Remove NEURON:

```
sudo rm -rf /Applications/NEURON
rm -rf ~/neuron
```

### 3. Clean PATH:

Edit ~/.zshrc (or ~/.bash\_profile):

```
nano ~/.zshrc
```

Remove lines containing "anaconda", "conda", "neuron", or "nrn"

Save and reload:

```
source ~/.zshrc
```

### 4. Restart your Mac (recommended)

## Linux

### 1. Remove Anaconda:

```
rm -rf ~/anaconda3
rm -rf ~/.conda
```

### 2. Remove NEURON:

If installed via package manager:

```
# Ubuntu/Debian
sudo apt-get remove --purge neuron
sudo apt-get autoremove

# Fedora
sudo dnf remove neuron
```

### If compiled from source:

```
rm -rf ~/neuron
```

#### 3. Clean PATH:

Edit `~/.bashrc`:

```
nano ~/.bashrc
```

Remove lines containing "anaconda", "conda", "neuron", or "nrn"

Save and reload:

```
source ~/.bashrc
```

#### 4. Restart your system (recommended)

---

## Step 2: Verify Clean System

Open a new terminal/command prompt and verify these commands fail:

```
python --version      # Should fail or show system Python only
conda --version       # Should fail
nrniv                 # Should fail
```

**If any of these work, you haven't fully removed the installations. Return to Step 1.**

---

## Step 3: Fresh Installation

Now follow **Category 2 (Clean System Installation)** exactly as written for your platform.

**Install in this exact order:**

1. Anaconda first
  2. NEURON second
  3. BrainCell third
- 

## Advanced Option: Dedicated User Account (Optional)

For the cleanest possible environment, create a new OS user account:

### Windows

1. Press Win+R, type `netplwiz`, press Enter
2. Click "Add" → Create a local account (not Microsoft account)

3. Log in as this new user
4. Follow Category 2 installation steps

## macOS

1. System Preferences → Users & Groups
2. Click the lock, authenticate
3. Click "+" to add a new user (Standard account)
4. Log in as this new user
5. Follow Category 2 installation steps

## Linux

```
sudo adduser braincell
sudo usermod -aG sudo braincell
su - braincell
```

Then follow Category 2 installation steps.

---

# Troubleshooting

## Common Issues & Solutions

### Issue: BrainCell doesn't start

#### Windows

**Symptoms:** Double-clicking `init.bat` does nothing, or window closes immediately

#### Solutions:

1. Ensure ZIP and batch files are unblocked (see Category 1, Step 3)
2. Run PowerShell command:  
`Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned`
4. Try running from Command Prompt:  
`cd C:\braincell\init.bat`

Check for error messages

#### macOS/Linux

**Symptoms:** Script doesn't execute or permission denied

#### Solutions:

1. Make script executable:  
`chmod +x init.sh`
3. Try running:  
`./init.sh`

5. Check for errors in terminal output
- 

## **Issue: Mechanisms fail to compile**

### **Windows**

#### **Solution:** Install Microsoft C++ Build Tools

1. Download from: <https://visualstudio.microsoft.com/downloads/>
2. Scroll to "Build Tools for Visual Studio"
3. Install "Desktop development with C++" workload
4. Re-run `build_mechs.bat`

### **macOS**

#### **Solution:** Install Xcode Command Line Tools

```
xcode-select --install
```

### **Linux**

#### **Solution:** Install build tools

#### **Ubuntu/Debian:**

```
sudo apt-get update  
sudo apt-get install build-essential
```

#### **Fedora:**

```
sudo dnf install gcc gcc-c++ make
```

---

## **Issue: NEURON can't find Python**

#### **Symptom:** NEURON launches but Python commands fail

#### **Platform:** All

#### **Solutions:**

1. Verify both NEURON and Anaconda are 64-bit
2. Check installation order (Anaconda first, then NEURON)
3. Verify Python from terminal:  
4. `python --version`
5. Test NEURON with Python:  
6. `nrniv -python>>> import sys>>> print(sys.version)>>> quit()`

**If still failing:** Go to Category 3 and perform clean installation

---

## **Issue: "Module not found" errors**

**Symptom:** Python import errors when running BrainCell

### **Solutions:**

1. Ensure you're using Anaconda's Python:
2. `which python` # macOS/Linux
3. `where python` # Windows

Should point to Anaconda directory

4. Install missing packages:
  5. `conda install numpy matplotlib scipy`
  6. Verify NEURON Python module:
  7. `python -c "import neuron; print(neuron.__version__)"`
- 

## **Issue: Permission errors (macOS/Linux)**

**Symptom:** "Permission denied" when running scripts

### **Solutions:**

1. Make all scripts executable:
2. `chmod +x *.sh *.hoc`
3. Check file ownership:
4. `ls -la`

Files should be owned by your user, not root

5. If files are owned by root:
  6. `sudo chown -R $USER:$USER ~/braincell`
- 

## **Platform-Specific Issues**

### **macOS: Apple Silicon (M1/M2/M3) Compatibility**

#### **If BrainCell doesn't work on Apple Silicon:**

1. Install Rosetta 2:
2. `softwareupdate --install-rosetta`
3. Use x86\_64 versions of Anaconda and NEURON
4. Run Python in Rosetta mode:
5. `arch -x86_64 python`

### **Linux: Missing GUI libraries**

**Symptom:** NEURON starts but no GUI appears

**Solution:** Install X11 libraries

**Ubuntu/Debian:**

```
sudo apt-get install libx11-dev libxt-dev
```

**Fedora:**

```
sudo dnf install libX11-devel libXt-devel
```

---

## Getting Help

If you're still experiencing issues:

1. **Check the full error message** in the console
2. **Take a screenshot** of the error
3. **Visit the forum:** <https://forum.neuroalgebra.net>
4. **Post your issue** including:
  - o Your operating system and version
  - o Your category (1, 2, or 3)
  - o Complete error message
  - o Steps you've already tried

**Forum support typically responds within 24-48 hours.**

---

## Appendix: Command Reference

### Quick Commands by Platform

#### Windows

```
# Check Python
python --version

# Check Anaconda
conda --version

# Check NEURON
nrniv -python

# Launch BrainCell
cd C:\braincell
init.bat
```

#### macOS/Linux

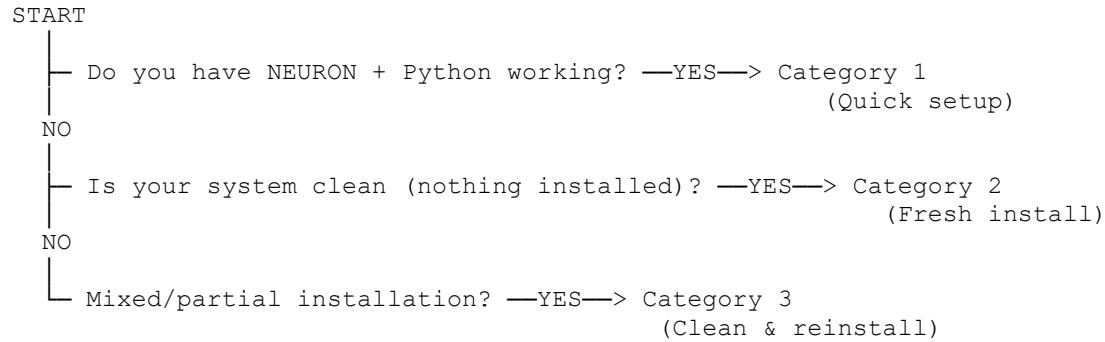
```
# Check Python
python --version

# Check Anaconda
conda --version
```

```
# Check NEURON  
nrniv -python  
  
# Launch BrainCell  
cd ~/braincell  
.init.sh
```

---

## Summary Flowchart



## Document Information

**Version:** 2.0 **Last Updated:** January 2026 **Platforms:** Windows 10/11, macOS 10.15+, Linux (Ubuntu 20.04+) **Feedback:** <https://forum.neuroalgebra.net>

---

## *Experimental Data or Approximations Needed to Build a Realistic Cell Model*

To create a biologically meaningful cell in BrainCell, gather the following data (or use reasonable estimates):

### **1. 3D Main Process Tree**

- Obtain a full 3D reconstruction from [NeuroMorpho.org](https://neuromorpho.org) (SWC, HOC, OBJ, ZIP).
- If unavailable, generate an artificial arbour with typical branching and diameters for the cell type.

### **2. Astrocyte Nano-structure Samples**

- Prepare  $\approx$ 20–50 nanoscopic astroglial processes from serial-section 3D EM.
- BrainCell uses these to derive statistical rules for generating realistic thin astrocytic processes.

### **3. Neuron Nano-structure and Spines**

- BrainCell can auto-generate dendritic spines with adjustable density, size, and synapse placement.
- Synapses can lie on spines or dendritic shafts; users can customise patterns and complexity.

### **4. Tissue Volume Fraction**

- Mean proportion of tissue occupied by astroglia and neurons versus distance from soma.
- Obtain from *in situ* two-photon imaging or published data.

### **5. Membrane Surface and Volume Ratios**

- Mean surface density and surface-to-volume ratio of fine processes, typically from 3D EM.

### **6. Electrophysiological Properties**

- A representative I–V curve from somatic patch-clamp recordings.
- Optional: neurotransmitter-uncaging responses, extracellular ion changes,  $\text{Ca}^{2+}$  wave speed, etc., to refine the model.

- 7. – Other functional data (e.g., responses to neurotransmitter uncaging, extracellular ion changes, calcium wave speed) are optional but can help refine the model.

## **GETTING STARTED**

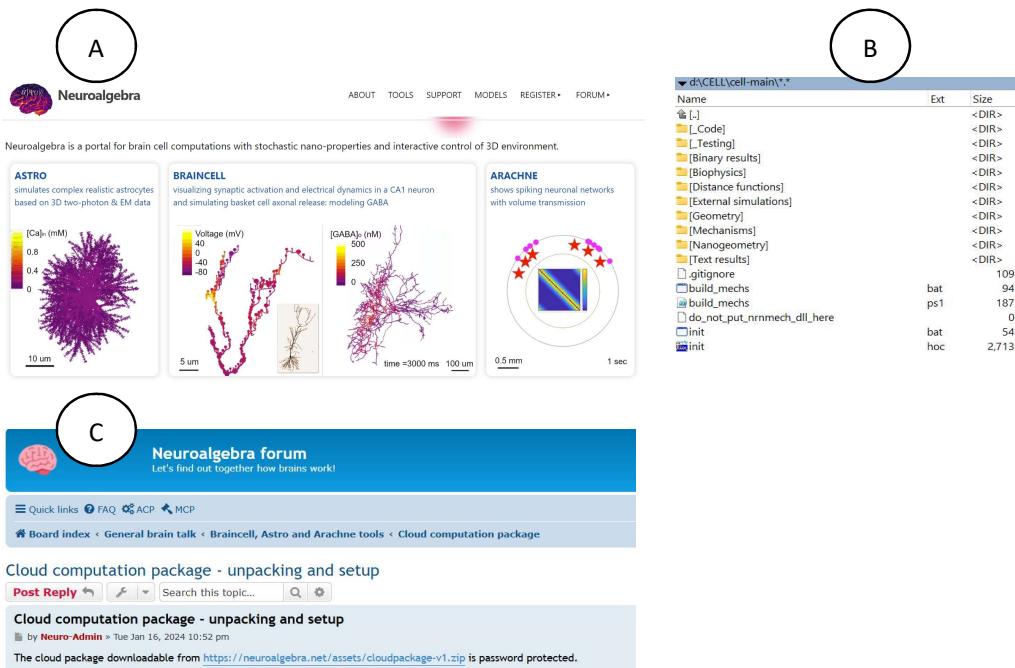
### *Installing and Launching BrainCell*

#### **1. Download the software**

- Visit <https://neuroalgebra.net>.
- On the front page (**Figure 1a**), click “Clone or download” to obtain the package *cloudpackage-v1.zip* (the password is available on the Forum).
- Extract the archive to a convenient folder, for example:  
C:\my\braincell (on Windows) or your home directory (on macOS).
- **Important:** keep the original folder structure exactly as in the archive (see **Figure 1b**).

#### **2. Check prerequisites**

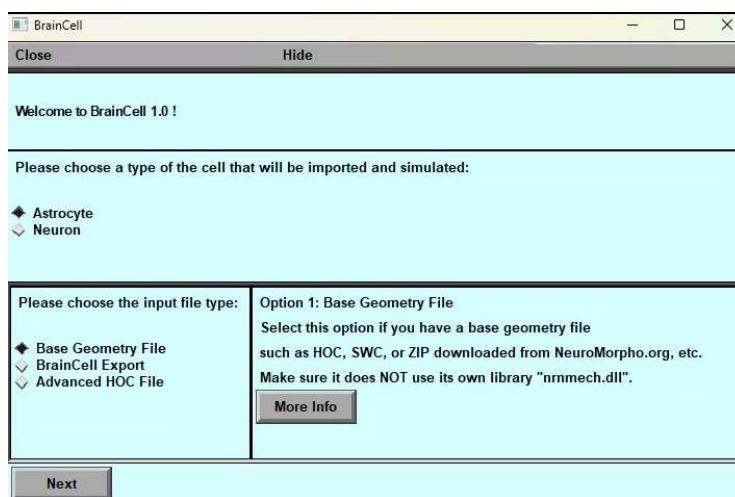
- Ensure that **NEURON 7.0 or later** and **Python 3.x** are installed **before** running BrainCell.



**Figure 1.** Download and installation overview. (a) Screenshot of the BrainCell download page on Neuroalgebra.net. (b) Example folder structure of the BrainCell 1.0 package on the host computer.

### Starting BrainCell

1. Navigate to the folder where you extracted BrainCell.
2. Run the startup script:
  - **Windows:** double-click `init.bat`
  - **macOS / Linux:** run the corresponding startup file (`init.hoc` or a provided terminal command).
3. The **BrainCell** main window will open, displaying the **Simulation Cell Configuration** menu (Figure 2).



**Figure 2.** Introductory menu of BrainCell 1.0 showing the Simulation Cell Configuration panel with options for Astrocyte or Neuron mode, and loading methods (Base Geometry, BrainCell Export, External Simulations).

## *Simulation Cell Configuration*

When the program starts, you can choose the **cell type** and **how to load its model**:

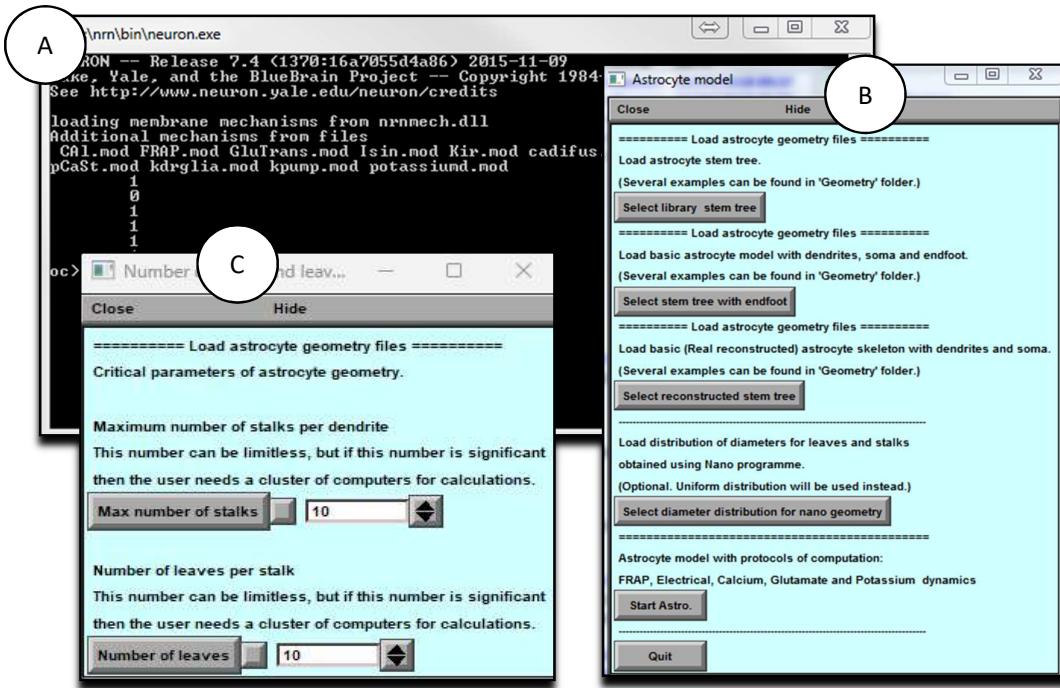
- **Astrocyte or Neuron** — pick the cell type you want to simulate.
- For each type, you have three options:
  1. **Base Geometry** – create or modify a new cell with custom 3D and nanoscale features.
  2. **BrainCell Export** – load an existing, pre-built cell with nanostructures (faster for repeated simulations).
    - *Note: you cannot change the geometry of a loaded export.*
  3. **External Simulations** – load an advanced HOC model with its own biophysics (e.g., from ModelDB) that uses a custom `nrnmech.dll`.

Select the option that matches your workflow to proceed.

1. [Astro / Base](#).
2. [Astro / Nano](#).
3. [Neuron / Base](#).
4. [Neuron / Nano](#).
5. [External Simulations](#).

## Astro/ Base. Setting up and running BRAINCELL: Astrocyte configuration.

### GENERATING COMPLETE ASTROCYTE MORPHOLOGY



*Launching the Astrocyte Module*

**Figure 3.** Astrocyte module control windows opened by the initialisation file `init.hoc`.

- (A) Command Console showing the loading of NEURON mechanisms.  
(B) Astrocyte Model Panel for selecting and loading predefined astrocyte geometries (basic or dendrite–soma combined).  
(C) Parameter Panel for setting nanoscopic morphology parameters — “Max number of stalks” and “Number of leaves.”

When you start BrainCell in **Astrocyte mode**, NEURON automatically opens three control windows that work together to initialise and configure the model (**Figure 3**):

1. **Command Console (A)** — displays system messages and the progress of loading simulation components (`cmd.exe`).
2. **Astrocyte Model Panel (B)** — the main interface for loading or creating astrocyte geometries. Users can select predefined structures or import custom models that include soma and dendritic branches.
3. **Parameter Panel (C)** — allows adjustment of key geometric parameters, such as:
  - o **Maximum number of stalks per dendrite**, and
  - o **Number of leaves per stalk**.These parameters define the density of nanoscopic astrocytic processes. Large values may require a computing cluster for simulation.

## Generating or Downloading the Astrocyte Stem Tree

To build a new astrocyte model, you first need to define the **basic dendritic (stem) tree structure**. BrainCell offers several ways to do this:

### Option 1 — Select Library Stem Tree

Click “**Select Library Stem Tree**” to import an existing 3D reconstruction of an astrocyte. You can use files in general **.zip**, **.hoc**, or **.swc** formats obtained from the [NeuroMorpho.org](https://neuromorpho.org) database. It is recommended to store these files in the directory:  
...\\Geometry\\Astrocyte\\New Style for easier access.



**Figure 4.** Importing a 3D astrocyte structure. (A) Example from the **NeuroMorpho.org** webpage displaying a 3D cell reconstruction. (B) BrainCell operational panels for selecting astrocyte morphology..

### Option 2 — Load from Local Directory

Alternatively, choose a file from your local BrainCell installation folder:

...\\BrainCell\\Geometry\\

Files can be in **SWC** or **HOC** format.

To view a 3D structure:

1. Press “**Select Library Stem Tree**.”
2. In the dialog window, navigate to the folder **Astrocyte/New Style** and select a file containing 3D geometry.
3. A 3D viewer window will open showing the structure.
4. Click “**Use this**” to proceed, or “**Import another**” to try a different file.
5. The **OriginalDendrite** parameter defines the number of main branches (dendrites in NEURON terminology).
6. Once selected, the structure will appear in the preview panel (**Fig. 5A**).

If you change your mind, you can reselect a file at any time — simply click “**Select Library Stem Tree**” again or choose “**Select Stem Tree with Endfoot**.”

When satisfied, click “**Start Astro.**”

⚠ After pressing “Start Astro,” the setup will lock in and cannot be reversed.

#### **Option 3 — Load a Reconstructed or Endfoot-Enhanced Tree**

##### **1. Editing Endfoot Geometry**

Selecting “Select Stem Tree with Endfoot” opens a pop-up window (see **Fig. 5B**) for editing **endfoot geometry**.

Here, you can define the morphology of both main and secondary endfoot branches and assign local biophysical mechanisms.

##### **2. Loading a Reconstructed Stem Tree**

The option “Select reconstructed stem tree” loads a detailed 3D reconstruction (for example, *RealAstrocyteSkeleton1.hoc* in the ...\\Geometry\\ directory).

This model represents a CA1 astrocyte skeleton reconstructed using the **Vaa3D** software (Allen Institute, available at [vaa3d.org](http://vaa3d.org)).

The corresponding window (**Fig. 5**) provides tools for scaling and centring the structure at the coordinate origin, simplifying alignment and compartment placement.

Adjustable parameters include:

- **X-Y scale (pixel /  $\mu\text{m}$ )**
- **Z scale (pixel /  $\mu\text{m}$ )**
- **X-Y shift ( $\mu\text{m}$ )**

The window automatically closes once parameters are applied.

💡 **Note:** In all cases, BrainCell will prompt you to locate the appropriate geometry file. Ensure that required 3D data have been downloaded in advance from NeuroMorpho.org.

#### **Option 4 — Add Nano-Geometry to the Astrocytic Tree**

After defining the 3D stem structure, you can enrich the model with **nanoscopic astrocytic processes**. To do so:

- Select “Select Diameter Distribution for Nano Geometry”, or
- Press “**Start Astro.**”

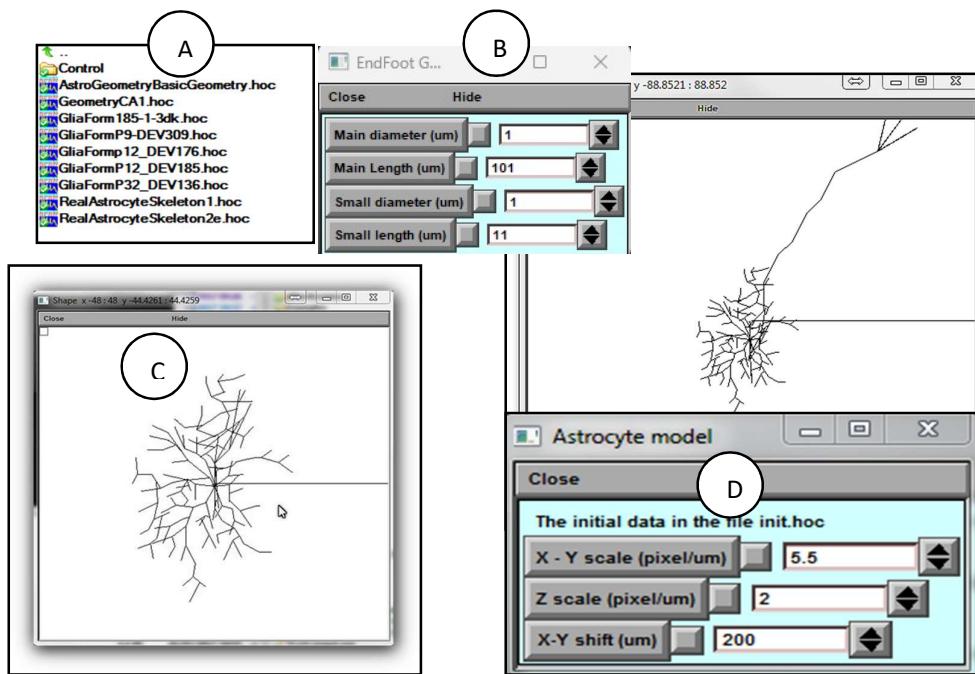
BrainCell will then generate nanostructures randomly, following physiological constraints derived from experimental morphometric statistics.

When learning the software, we recommend using “**Start Astro**” to familiarise yourself with its features and workflow.

## Finalising and Preparing the Astrocyte Geometry

We recommend pressing the “Start Astro” key when first becoming familiar with the BrainCell software.

This option automatically loads the selected geometry and initiates all relevant setup panels, allowing you to explore and understand the program’s functionality and available features.



**Figure 5.** Operational panels for creating and editing astroglial morphology in BrainCell.

(A) **File Navigation Panel** — lists available .hoc files that define 3D astrocytic structures within the user’s directory.

(B) **Endfoot Geometry Panel** — enables editing of astrocyte endfeet and adding associated biophysical mechanisms.

(C) **3D Export Panel** — provides tools for exporting the constructed 3D astrocyte model for later analysis or integration with other simulations.

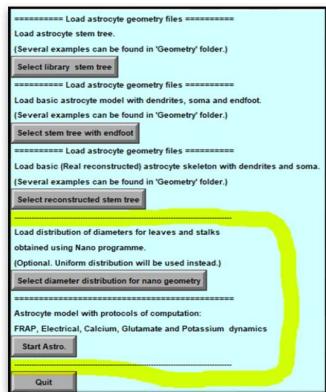
(D) **Morphology Transformation Panel** — allows for precise scaling and spatial shifting of the 3D shape, which is essential for accurate astroglial morphology adjustments.

### Note:

Before proceeding to additional model design steps, ensure that the **cell stem tree geometry** has been successfully uploaded to the designated **GEOMETRY** directory.

## Generating astroglial morphology on the nanoscale

### The geometry of nanoscopic processes



Once the stem tree has been loaded, the next stage is to define the **nanostructure** of the astrocyte. A popup window (highlighted in yellow) offers two options:

#### Option 1 — Load Default Nano-Geometry

##### Press “Diameter Distribution for Nano-Geometry.”

This loads a pre-computed file containing diameter statistics generated by the *Nano (Geometry)* module from sampled 3D-reconstructed astroglial processes.

By default, BrainCell imports the file `testshape.dat_radii_dist.txt`,

which represents the distribution of process diameters measured in the CA1 stratum radiatum. After loading, press “**Start Astro**.”

#### Option 2 — Generate Nano-Geometry Automatically

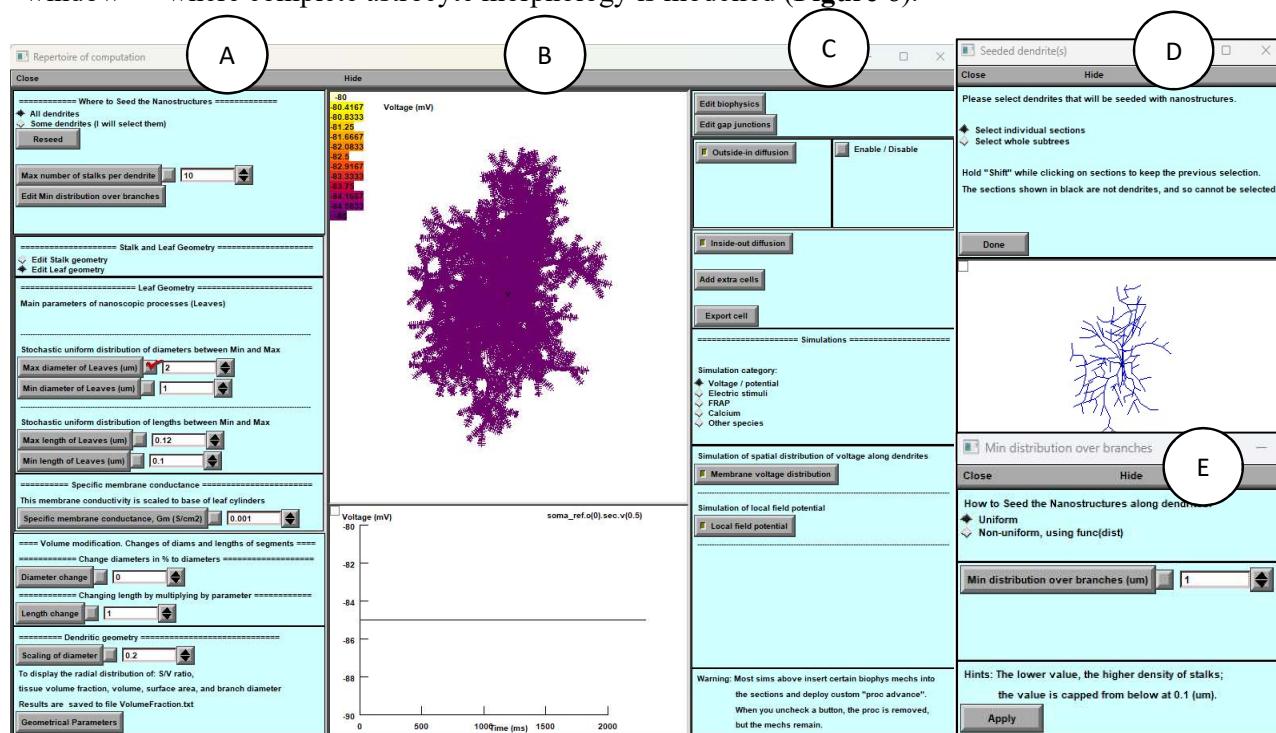
Press “**Start Astro**” without selecting an external file.

In this case, BrainCell automatically generates nanoscopic processes using internal algorithms that replicate experimentally observed morphometric trends.

In both cases, you can later adjust key morphometric parameters of the generated nanostructures. Detailed controls for these adjustments are provided in the chapter **Simulating Astrocyte Physiology**.

### Populating astrocyte tree with nanoscopic processes

Pressing “**Start Astro**” launches the main operational interface — the **Repertoire of Computation** window — where complete astrocyte morphology is modelled (**Figure 6**).



## **Figure 6. Astrocyte Main Window and configuration panels.**

- (A) **Control Panel:** provides detailed settings for fine-tuning astrocyte geometry.
- (B) **Simulated Variable Panel:** displays simulated parameters mapped onto the astrocyte morphology (e.g., membrane voltage) and a live digital output plot.
- (C) **Biophysics, Stochastic, and Extracellular Sources Settings:** configure the cell's biophysical properties, stochastic parameters, and extracellular input conditions.
- (D) **Nanostructure Placement:** defines where nanostructures are placed — across the entire astrocyte or restricted to specific dendrites.
- (E) **Nanostructure Density Distribution:** sets the density pattern of nanostructures (uniform or non-uniform), influencing both spatial coverage and physiological behaviour..

### **Localising Nano-Geometry**

An important control located in the **upper-left corner** of the main window (**Fig. 6A**) allows the user to restrict nano-geometry generation to specific dendrites.

Follow these steps:

1. **Switch to “Some Dendrites”:** change from “All Dendrites” to limit nano-geometry creation to selected dendrites.
2. **Initiate Reseed:** press “Reseed” to open a new window for specifying placement options.
3. **Specify Locations:** define which regions of the dendritic tree will host nanostructures.
4. **Select Dendrites:** choose the target dendrites in the dialog.
5. **Finalize:** press “Done” to generate an astrocyte with nanostructures localised to the chosen branches.

**Note:** Localised nano-geometry significantly increases computational load but is valuable for studying highly specific subcellular domains and local signalling dynamics.

---

### **Nano-Geometry Configuration Panels**

#### **Leaf Geometry Panel (Fig. 6A, top)**

Defines the distribution of **cylindrical leaf compartments** (nanoscopic processes).

When experimental 3D statistics are unavailable, a uniform random distribution can be generated within user-defined lower and upper diameter limits.

**Note:** This panel is ignored if statistical data from a reconstructed nano-geometry file have already been loaded.

#### **Stalk Geometry Panel (Fig. 6A, middle)**

Sets upper and lower limits for **stalk cylinder diameters**, determining how densely the tissue is populated with transitional nanostructures between the main branches and leaves.

#### **Specific Membrane Conductance Panel**

Allows setting of membrane conductance using the control “**Gm (mS/cm<sup>2</sup>)**”, accounting for the total surface area of exposed nanostructures.

The default **resting membrane potential** is **-85 mV**, as defined in NEURON’s **Distributed Mechanism** panel.

#### **Dendritic Geometry Panel (Fig. 6A, bottom)**

Controls **branch diameter scaling**, i.e., how dendrite diameter decreases with distance from the soma, according to empirical measurements. The relationship follows:

$$d(r) \propto [S(r + 1)]^{-1/2}$$

where

- $d(r)$ — branch diameter at distance  $r$  from soma,
- $S$ — scaling coefficient (*scalingDiam*),
- $r$ — distance from soma ( $\mu\text{m}$ ).

### Customising Cell Geometry

Advanced options “Edit Biophysics” and “Edit Morphology” enable rule-based modifications of any morphological or biophysical property as a function of distance from the soma. This feature allows users to construct highly customised cell architectures for specific simulation goals.

**Note:** These editing panels should be skipped if a fully 3D-reconstructed stem tree has already been imported.

### *Tissue-filling properties of astroglial morphology*

#### Tissue Volume Filling and Geometrical Parameters

The **tissue volume-filling properties** and **surface-to-volume ratios** of nanoscopic processes are determined by their individual shapes and by the effective density of simulated nanostructures, as defined in the previous section.

To monitor these parameters, press the “**Geometrical Parameters**” key (**Figure 7**). This command opens several analysis windows that display quantitative characteristics of the generated astrocyte morphology — including total cell volume, surface area, surface-to-volume ratio, and volume-filling fraction within the simulated tissue.

All calculated values are **automatically saved** to the file:  
...\\neuronSims\\Text results\\VolumFraction.txt

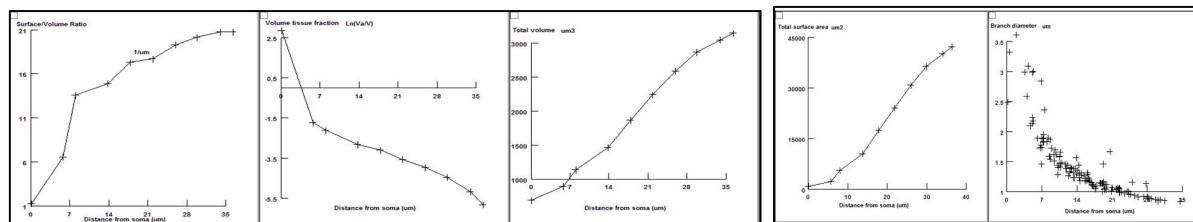
Below are the key features of the main window:

1. **Nano geometry modification**: Users can add and modify spines' nano geometry to study the impact of structural changes on neuron behaviour.
2. **Biophysical mechanisms addition**: The tool enables users to add and change various biophysical mechanisms to neurons to investigate their impact on neuron behaviour.
3. **Synapse distribution**: The tool facilitates the addition and distribution of different types of synapses on the dendritic tree, enabling the study of the neuron's connectivity and behaviour.
4. **Simulation models**: Users can choose various modes to simulate the neuron's behaviour accurately. The simulation models include voltage clamp, current clamp, and dynamic clamp.

The neuron simulation tool's main window provides the necessary features to simulate and analyse neuron behaviour effectively.

### *"Geometric parameters" button*

The “**Geometrical Parameters**” button calculates and visualises the principal **morphometric characteristics** of the modelled astrocyte. When pressed, it generates a series of analytical plots and data windows summarising the cell’s geometry.



**Figure 7.** Geometrical analysis panels generated by the “Geometrical Parameters” command. Displayed windows (from top left):

- Surface-to-volume ratio distribution
- Tissue volume fraction
- Total cell volume (cumulative with distance from the soma)
- Total cell surface area
- Diameters of primary processes

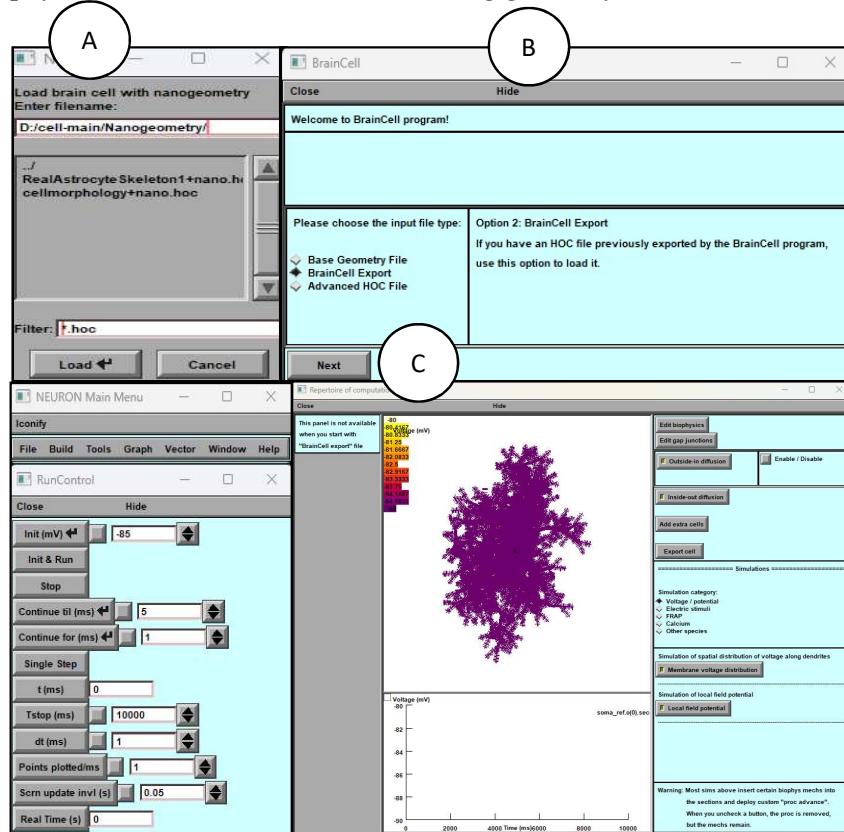
The morphometric characteristics displayed in **Figure 7** can be compared directly with experimental data derived from **3D electron microscopy reconstructions** or **two-photon excitation imaging** of astroglia. Such comparison allows users to evaluate any discrepancy and adjust the **density or scaling of nanoscopic processes** accordingly, using the **Stalk Geometry** and **Dendritic Geometry** options (see **Figure 6**) until the model accurately reproduces measured tissue statistics. These geometric-parameter windows can be accessed **at any stage of modelling** to verify morphological realism and ensure parameter consistency. Once this step is complete, the **astroglial morphology** is fully established. Further fine structural tuning—such as for **FRAP (fluorescence recovery after photobleaching) experiments**—can be performed later, as described in subsequent sections. The model is now ready for simulating astroglial functions with appropriate **membrane and intracellular biophysical mechanisms**, introduced in the following chapters.

1. **Nano geometry modification**: Users can add and modify spines' nano geometry to study the impact of structural changes on neuron behaviour.
2. **Biophysical mechanisms addition**: The tool enables users to add and change various biophysical mechanisms to neurons to investigate their impact on neuron behaviour.
3. **Synapse distribution**: The tool facilitates the addition and distribution of different types of synapses on the dendritic tree, enabling the study of the neuron's connectivity and behaviour.
4. **Simulation models**: Users can choose various modes to simulate the neuron's behaviour accurately. The simulation models include voltage clamp, current clamp, and dynamic clamp.

The simulation tool's main window provides the necessary features to simulate and analyse behaviour effectively.

## Nano Astro — Loading a Pre-existing Astrocyte Morphology

BrainCell allows users to load previously created astrocyte morphologies that already include nanostructural detail. This feature enables users to continue simulations, analyse existing models, or apply new biophysical mechanisms without redefining geometry.



**Figure 8.** Operational panels for loading pre-existing 3D astrocyte structures with nanoscopic detail.

(A) **Download Panel:** provides access to a previously prepared 3D astrocyte morphology. (*Visual: an icon representing data download.*)

(B) **Roadmap Panel:** lets users choose between creating a new astrocyte (Base) or loading a pre-existing nanostructured model (Nano). (*Visual: a roadmap-style diagram branching to both options.*)

(C) **Schematic Illustration Panel:** displays a schematic view of the 3D astrocyte morphology, including its nanoscale architecture. (*Visual: a simplified 3D diagram showing astrocytic nanostructure.*)

## Loading a Pre-existing Astrocyte (“Astro + Nano”)

To open a saved astrocyte model that already includes nano-geometry:

1. In the main menu, click “Astro + Nano” (Fig. 8A).
- This opens the **Download Panel**.
2. Follow the prompts to open the **NEURON Basic Panel**, which helps locate the previously prepared astrocyte file containing nanostructural data.
  3. Select the astrocyte morphology to load.
  4. Once selected, BrainCell will open new options for **simulation setup** and **biophysical mechanism management**.

At this stage, you can:

- Simulate physiological processes using the selected astrocyte, and
- Manage its **biophysical and intracellular mechanisms**.

**Important:** After loading a pre-existing astrocyte model, its **geometry cannot be altered**. All subsequent operations relate only to simulations, parameter tuning, and mechanism management.:.

5. **Nano geometry modification**: Users can add and modify spines' nano geometry to study the impact of structural changes on neuron behaviour.
6. **Biophysical mechanisms addition**: The tool enables users to add and change various biophysical mechanisms to neurons to investigate their impact on neuron behaviour.
7. **Edit Gap Junction**: The tool facilitates the addition and distribution of different types of gap junction on the cell, enabling the study of the cell's connectivity and behaviour.
8. **Synapse distribution**: The tool facilitates the addition and distribution of different types of synapses on the dendritic tree, enabling the study of the neuron's connectivity and behaviour.
9. **Simulation models**: Users can choose various modes to simulate the neuron's behaviour accurately. The simulation models include voltage clamp, current clamp, and dynamic clamp.

The neuron simulation tool's main window provides the necessary features to simulate and analyse neuron behaviour effectively.

# Neuron / Base. Setting up and running BRAINCELL: Neuron module.

## IMPORTING AND GENERATING COMPLETE NEURON MORPHOLOGY

### Building a 3D Neuron Model

The **Neuron module** of BrainCell enables the creation, import, and editing of fully 3D neuronal geometries, independent of the astrocyte module.

When you load or create a neuron model, two main interface windows appear:

- **Cell Structure Editor (Fig. 9A–B):**

Displays the neuron's **soma**, **dendritic tree**, and **axon**.

- If an axon is not defined, BrainCell can automatically generate a simple default one, or you can import it from another project.
- Geometric parameters can be adjusted later through the editor.

Each structural element (soma, dendritic branch, axon) must have a **unique name**; duplicated names will generate an error message.

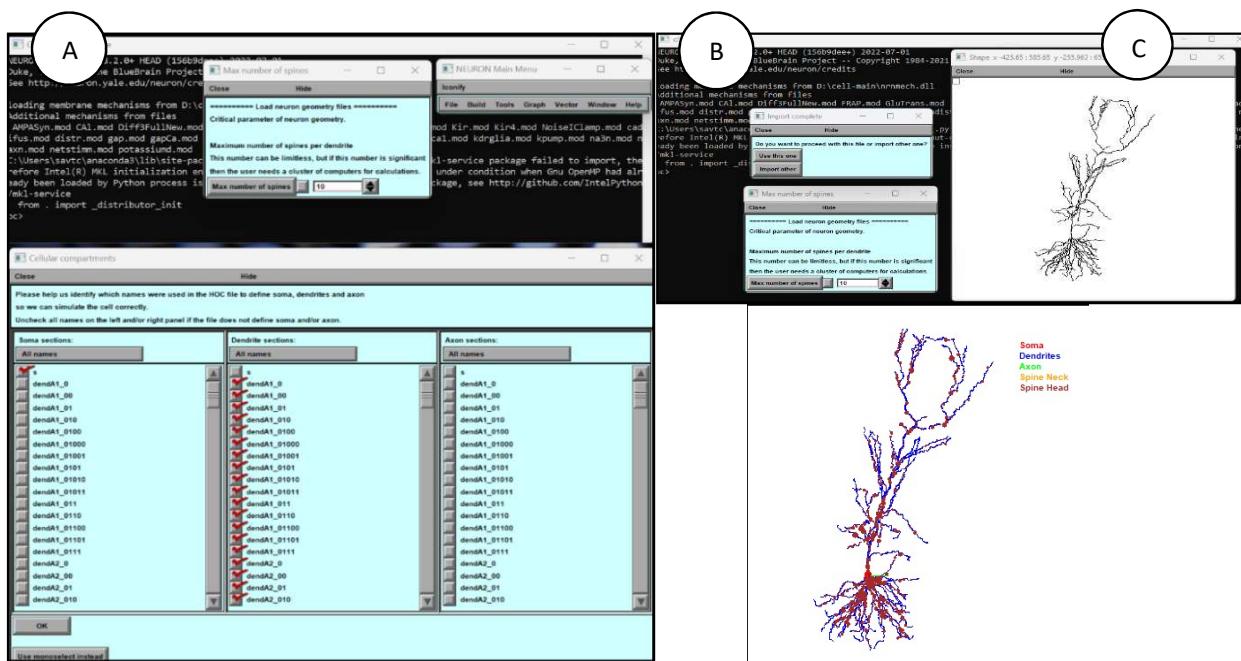
- **Spine Settings (Fig. 9C):**

Allows you to define the **maximum number of dendritic spines** and modify their geometric characteristics.

These parameters can be refined later if required.

To create a new neuron:

1. Choose your desired configuration in the setup window.
2. Click **OK** to confirm.
3. BrainCell will generate the 3D neuron model, which you can:
  - Use directly for simulations,
  - Combine with other neuron or astrocyte models, or
  - Replace later with a different geometry from the database.



**Figure 9.** Import, selection, and finalisation of a 3D neuronal structure.

- (A) **Cell Structure Definition Panel:** defines the soma, dendrites, and axon. Future updates will support custom geometry naming and dendritic subgrouping.  
(B) **Cell Shape Selection Panel:** provides options to choose or modify the desired neuron morphology.  
(C) **Spine Density Control Panel:** adjusts the maximum spine density per dendrite (noting that thicker dendrites generally host fewer spines). Future versions will support continuous spine-density editing.
- 

**Workflow Summary (Fig. 9):**

- **Panels A–B** illustrate the import process, allowing identification of soma, dendrites, and axon, and setting of the maximum number of spines for the principal dendrite.
- **Panel C** shows the selected structure, where you can either accept it or load an alternative morphology.

To explore different neuron shapes, simply repeat the **import and selection** steps. When satisfied with the current morphology, click “**Use this one**” to confirm.

After confirmation, BrainCell opens the **main Neuron Editor window**, where you can:

- Refine and customise cell geometry and morphology,
- Visualise modifications in 3D, and
- Add or configure **biophysical mechanisms** with spatially variable or stochastic properties.

This workflow enables the construction of **realistic, fully customisable neuronal models** ready for advanced simulation.

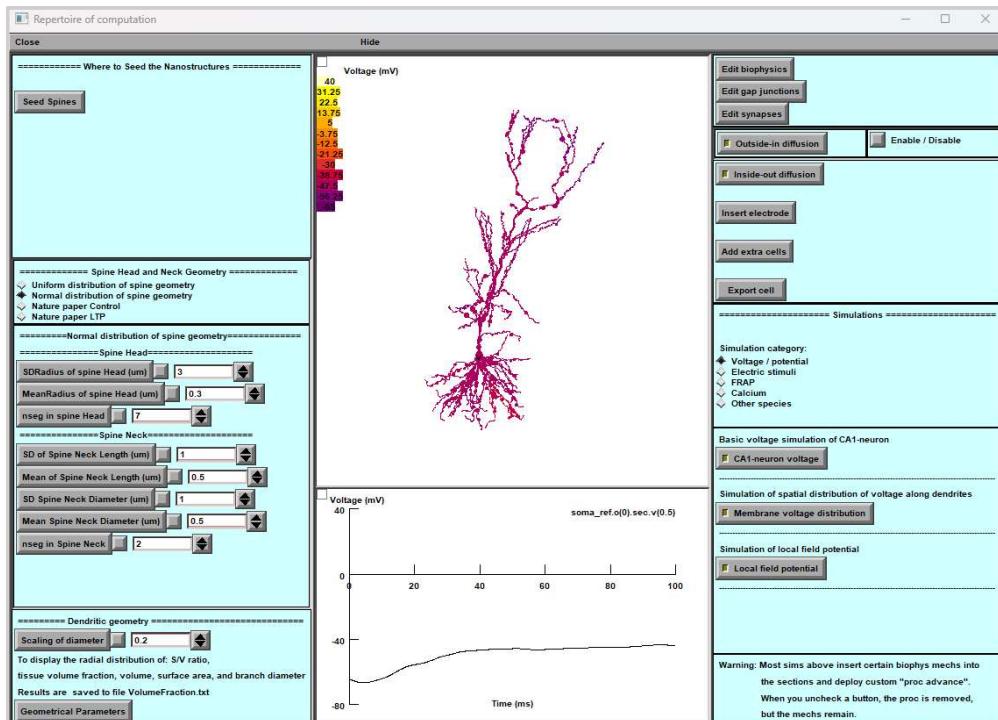
## Navigating Neuron Simulation: Exploring Main Windows for Enhanced Analysis

BrainCell provides a powerful and intuitive environment for **customising and analysing neuronal geometry and biophysical properties**.

Users can define how structural and functional components are grouped or divided during model setup:

- **Combine** the neck and head of dendritic spines into a single functional unit, or
- **Split** dendritic branches into specific regions, such as *proximal* and *distal* segments.

In the **main window** (Figure 10), you can modify the neuron's geometry, apply and test various **biophysical mechanisms**, and explore how these structural and functional choices influence neuronal behaviour.



**Figure 10. Main window for neuron simulation.** The main simulation interface provides all tools required to configure, simulate, and analyse neuronal behaviour.

## Key Features of the Main Neuron Window

- **Nano geometry modification**: “**Seed Spines**”

Enables the addition or modification of spine nano-geometry.

This feature allows researchers to study how fine structural changes in dendritic spines affect synaptic integration and overall neuronal dynamics.

- **Edit Gap Junction**. Opens a dedicated editor for creating new or modifying existing gap junctions. Users can define their spatial distribution, conductance, and connectivity patterns to match specific network configurations or experimental data..

- **Biophysical mechanisms addition**: Provides options to introduce, remove, or adjust membrane and intracellular mechanisms (e.g., ion channels, pumps, transporters). These modifications allow exploration of how molecular mechanisms influence neuronal excitability and signal propagation..
  - **Synapse distribution**: Facilitates the placement and classification of different synapse types across the dendritic tree. This enables detailed investigation of neuronal connectivity and synaptic input integration.
  - **Edit extracellular source**. Allows full customisation of the **extracellular ionic environment** surrounding the neuron. Users can define the concentration and type of ions or neurotransmitters, controlling external conditions for specific simulation scenarios.
  - **Simulation models**: BrainCell supports multiple simulation modes for experimental flexibility, including:
    - **Voltage Clamp** — control membrane potential while measuring ionic currents.
    - **Current Clamp** — inject current to observe membrane voltage responses.
    - **Dynamic Clamp** — apply real-time feedback between experimental and simulated components.
- 

The **Neuron Simulation Tool** in BrainCell thus integrates geometry editing, mechanism management, and electrophysiological simulation within a unified workspace, providing researchers with a complete platform for studying neuronal structure–function relationships.

## Customising Neuronal Connectivity: Spine Set Configuration.

### Configuring Spine Sets

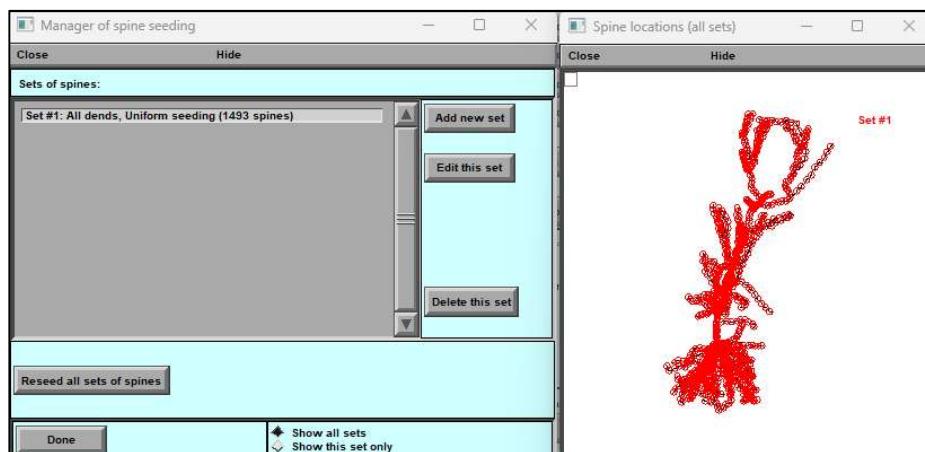
BrainCell lets you **create and customise dendritic spines** — the tiny protrusions where neurons form synapses. This feature helps you build realistic models and control how spines affect electrical and biochemical properties.

With the **Spine Sets** tool, you can:

- **Create different spine distributions** – design unique spine patterns for different dendritic regions.
- **Customise spine properties** – adjust spine size, shape, electrical behaviour, and randomness within each set.
- **Fine-tune control** – edit each spine set individually for precise modelling.

### How to Use the Spine Sets Feature

1. **Open Spine Sets**
  - Locate the “**Spine Sets**” option in the BrainCell interface (its location may vary slightly by version).
2. **Primary Panel**
  - **Add Spine Set** – click to create a new group of spines with chosen characteristics.
  - **Remove Spine Set** – select an existing group and delete it.
3. **Auxiliary Panel**
  - **Visualise Spines** – view your neuron model with spines colour-coded by set to see where they are located.



**Figure 11a.** The panel displayed is used to manage spine seeding.

The **left panel — Manager of Spine Seeding** — lists all existing spine sets and lets you:

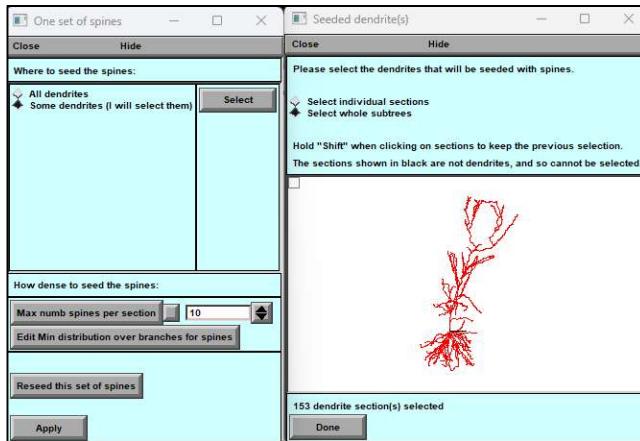
- **Add, edit, or delete** a spine set,

- **Reseed all sets** to reset their placement,
- Toggle between **Show all sets** and **Show this set only**,
- Click **Done** to finish and save changes.

The **right panel — Spine Locations (all sets)** — shows a visual map of spine positions along the dendritic tree, with the **currently selected set** highlighted.

Using these tools, you can add or adjust the **nano-geometry of spines** and explore how structural changes affect neuronal behaviour.

### *The editing of spine locations.*



**Figure 11b. Panels for managing the spatial distribution of spine seeds.**

### *Spine Seeding Panels*

#### **Left Panel — Single Spine Set**

- Choose where to place spines: **all dendrites** or **specific dendrites**.
- Set **spine density**, including the **maximum number of spines per section**.
- Adjust the **minimum distribution** of spines along branches.
- **Reseed** the current set to generate a new random placement.
- Click **Apply** to confirm changes.

#### **Right Panel — Seeded Dendrite(s)**

- Select individual dendrites or sub-sections for spine placement.
  - Use the mouse to click dendrites; hold **Shift** to select multiple.
- See a **3D visualisation** of the neuron with highlighted areas where spines will be placed.
- View the **number of dendritic sections selected**.
- Click **Done** to finalise your selection.

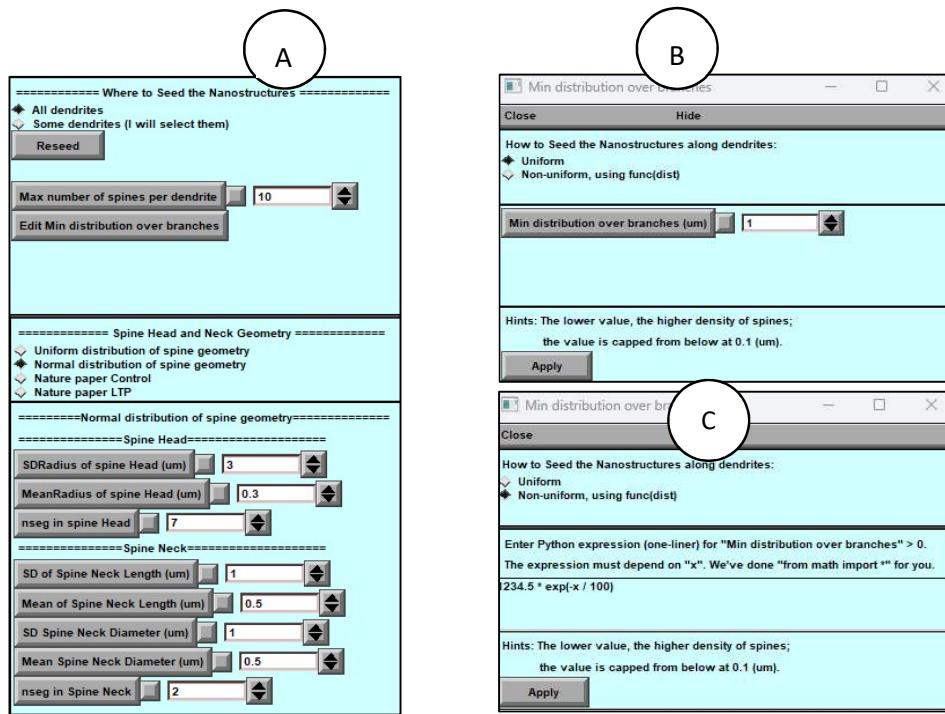
### *“Where to Seed the Nanostructures” Tab*

- (Fig.11b) This tab enables precise spatial control of spine distribution across the neuronal dendritic tree:
    - Click directly on the dendrites in the 3D viewer to define spine locations.
    - Hold **Shift** to select multiple dendrites simultaneously.
    - Once placement is satisfactory, click **Done** to save the configuration.
    - You may revisit this panel at any time to modify or reseed the distribution.
- 

The **Spine Sets** configuration system thus provides an integrated framework for shaping neuronal connectivity at the nanoscale, enabling BrainCell users to model realistic dendritic structures and study how spine organisation impacts synaptic processing and network behaviour.

## The spine, head and neck geometry. Location over dendritic tree.

This part explains how to use the tool to analyse the geometry distribution of dendritic spines (Fig.12). The panel provides two distribution options, regular and uniform, allowing the user to set numerical parameters for each distribution via a window upon selection. The tool also considers the minimum distance between synapses in a dendritic tree as an essential parameter for synapse distribution.



**Figure 12. Spine Geometry Modification Panel**

- A) Main Panel of Spine Geometry: This central panel provides options and controls for modifying the geometry of spines within the dendritic tree.
- B) Panel for Uniform Distribution: This sub-panel is dedicated to achieving a uniform distribution of spines across the dendritic tree and offers precise control over their arrangement.
- C) Panel for Non-Uniform Distribution: In this sub-panel, spines are distributed non-uniformly using a user-defined formula, allowing for customised and intricate spine arrangements.

### Configuring Spine Distributions

#### 1. Selecting a Distribution Type

To begin, choose how spines will be distributed along dendrites.

Two options are available:

- **Regular** – spines are placed at predictable intervals.
- **Uniform** – spines are spaced randomly but evenly overall.

Select your preferred type by clicking the corresponding button (Fig. 12). A parameter window will appear, where you can enter the **numerical settings** for the chosen distribution.

---

## 2. Using Pre-Defined Distribution Types

For convenience, BrainCell includes **experiment-based spine geometry templates** from: *Tønnesen J., Katona G., Rózsa B., et al. Spine neck plasticity regulates the compartmentalisation of synapses. Nat Neurosci 17, 678–685 (2014).* <https://doi.org/10.1038/nn.3682>.

Click the corresponding button on the panel to load one of these **pre-established distributions**. The tool will automatically apply published experimental parameters for the selected type.

---

## 3. Adjusting Spine Complexity

You can control the **complexity** (number of segments) used to model each spine:

- Minimum: **2 segments**
- No maximum limit (but more segments slow down calculations).

Use the **slider** on the panel to set the desired level.

---

## 4. Defining Synapse Distance

Synapse spacing is critical for realistic models. BrainCell lets you:

- Adjust the **minimum distance between synapses**, either uniformly or non-uniformly (Fig. 12 B–C).
- Use a **stochastic element** — placement includes some randomness by default.
- For non-uniform patterns, enter a **custom Python-style formula** in a single line, then click **Apply** to generate the pattern.

This allows both **randomised** and **user-defined** spacing for fine-grained control.

---

## Summary

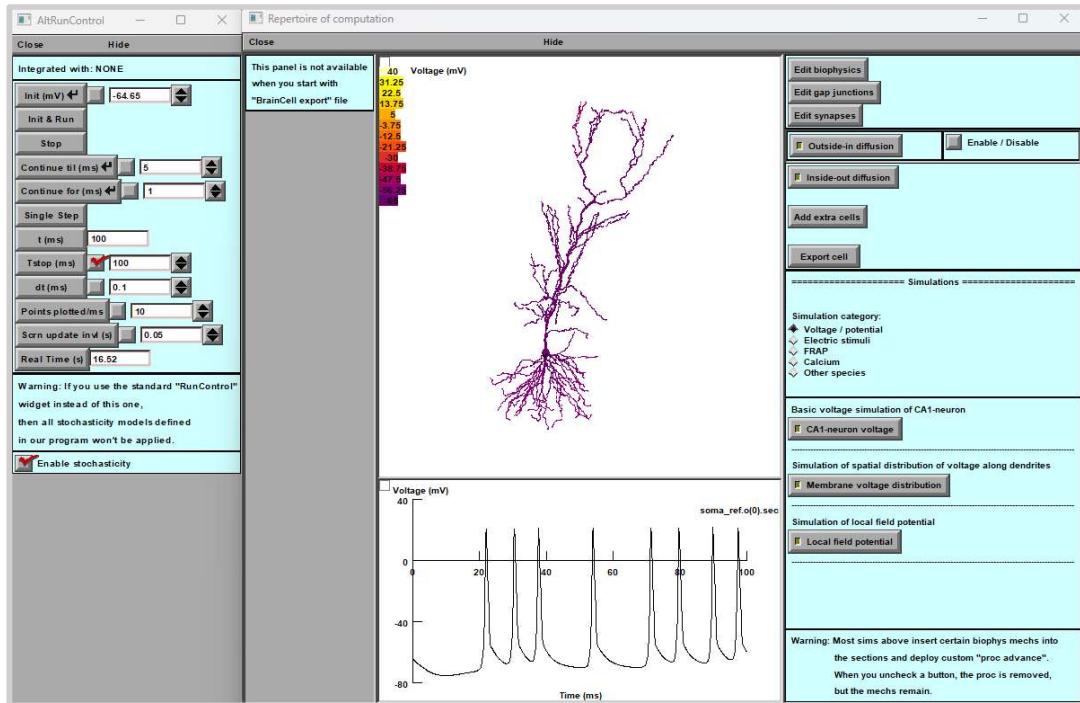
This tool gives full control over **spine geometry and synapse placement**.

- Choose a **regular** or **uniform** distribution.
- Load **predefined experimental templates** when available.
- Adjust **spine complexity** and **synapse spacing** to balance biological realism and simulation speed.

“Nano Neuron”. Download the previously created neuron model.

The combination "Neuron + Nano" creates a new panel. Once the panel appears, you can proceed with the following steps and open the NEURON Basic Panel to locate the previously prepared Neuron with Nanostructure. To proceed with the simulation and management of biophysical mechanisms, select the neuron with Nanostructure in hoc-file. This will take you to a new option for simulation and management (see Fig.13).

**Note:** At this stage, you cannot modify the geometry.



**Figure 13: Left:** The "Alternative Run Control" panel is designed to initialise and run simulations with specific settings like initial membrane voltage, run time, and integration step size, including options for continuous or single-step simulation. It emphasises using this customised run control to maintain the defined stochasticity models.

**Right:** The "Repertoire of Computation" panel, referred to as "Previously created Neuron morphology", appears to be a visualisation and simulation panel. It has sections for editing cell biophysics, gap junctions, synapses, and extracellular sources, as well as a range of simulation categories such as voltage, electric stimuli, and ion-specific dynamics. The panel also warns about integrating custom biophysics mechanisms into the simulation.

## Manager of biophysical mechanisms.

### *Manager of Biophysical Mechanisms*

The **Manager of Biophysical Mechanisms** lets you **add, edit, and organise biophysical mechanisms** within your neuron model to study their effect on neuronal behaviour.

Click the “**Manage the Distance of Mechanisms**” button in the **upper-right corner** of the main window (Fig. 10 . [Main Window](#)). This opens two main panels (Fig. 14):

---

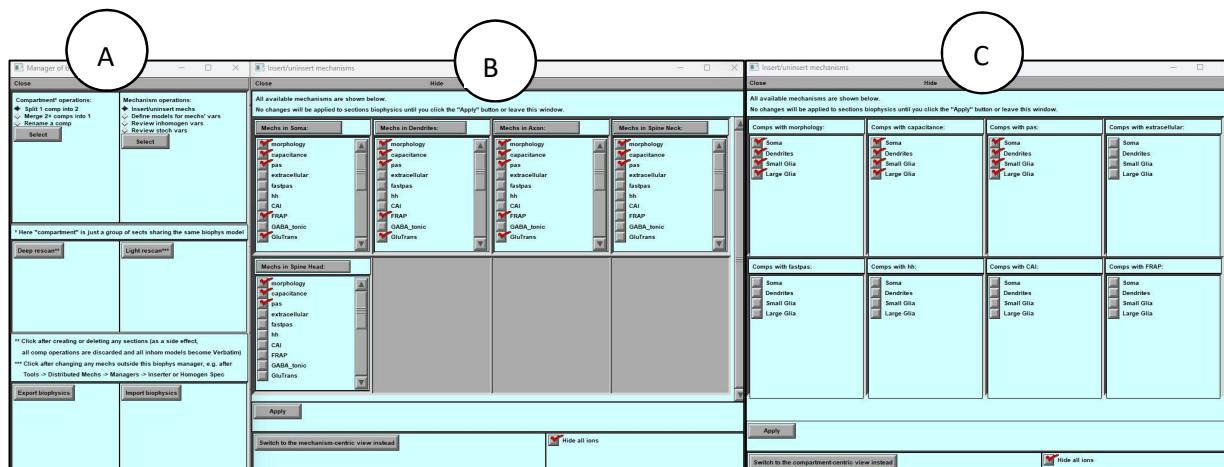
#### *Window A — Cell Compartments & Mechanisms*

This panel provides tools for **editing compartments** and **managing mechanisms**:

1. **Compartment Operations**
  - **Split or merge compartments** to change the model’s structure.
  - **Rename compartments** for better organisation.
2. **Rescanning the Model**
  - **Deep Rescan** – use when you add new cell areas not originally listed by BrainCell; updates the model to include them.
  - **Light Rescan** – use after adding new mechanisms so they integrate correctly.
3. **Export & Import**
  - **Export to JSON** – save your current set of biophysical mechanisms in a JSON file for backup or sharing.
  - **Import from JSON** – quickly load previously saved mechanisms into your current model.

---

These features let you **customise the cell’s structural regions**, integrate **additional mechanisms**, and keep a **reusable library** for future simulations.



**Figure 14. Panels for Managing Biophysical Mechanisms**

## A — Distributed Mechanisms Manager

This panel lets users organise and edit the distribution of biophysical mechanisms across cell compartments. You can add or split regions, scan existing areas for their assigned mechanisms, adjust spatial distributions, and export or import mechanism sets for reuse in other models.

## B — Area-Centric Mechanism List

Shows all mechanisms available for the currently selected cell area. Use this view when you want to see *which mechanisms are present in a particular compartment* and add or remove them as needed.

## C — Mechanism-Centric Area List

Shows all cell areas associated with the currently selected mechanism. Use this view when you want to see *where a given mechanism is applied* and edit its distribution across compartments.

Use the **Apply** button to confirm additions or removals. The **toggle button** at the bottom switches between the *area-centric* (B) and *mechanism-centric* (C) views, providing a flexible way to manage mechanism placement and distribution.

### Windows B & C — Managing Mechanisms and Cell Parts

**Windows B and C** give you full control over **biophysical mechanisms** and the **structural organisation** of your neuron model.

#### 1. Mechanism Management

- **Mechanisms Folder** – shows a complete list of available mechanisms.
- **Displayed Neuron Components** – visualises the cell regions you've already built.

#### 2. Two Display Modes

- **Initial Mode** – shows which mechanisms are present in each neuron segment.
- **Secondary Mode** – highlights the exact location of each mechanism within those segments.

#### 3. Selecting Mechanisms

- Tick the checkboxes next to the desired mechanisms to include or exclude them for your simulation.

#### 4. Cell-Part Operations

- **Subgroup, merge, or rename** structural parts to organise the neuron model more effectively.

#### 5. Mechanism Interaction & Advanced Tools

- **Insert or remove mechanisms** dynamically.
- **Adjust spatial distribution** for precise control of where each mechanism acts.
- **Visualise spatial inhomogeneity** to fine-tune gradients or non-uniform distributions.
- **Stochastic mechanism analysis** – explore random or probabilistic behaviours of mechanisms.

**Windows B and C** provide a powerful, visual way to **edit cell parts and mechanisms**, making it easier to customise, analyse, and refine your neuron model for accurate simulation results.

### *Split, Merge and Remain option.*

#### *Splitting Cell Compartments*

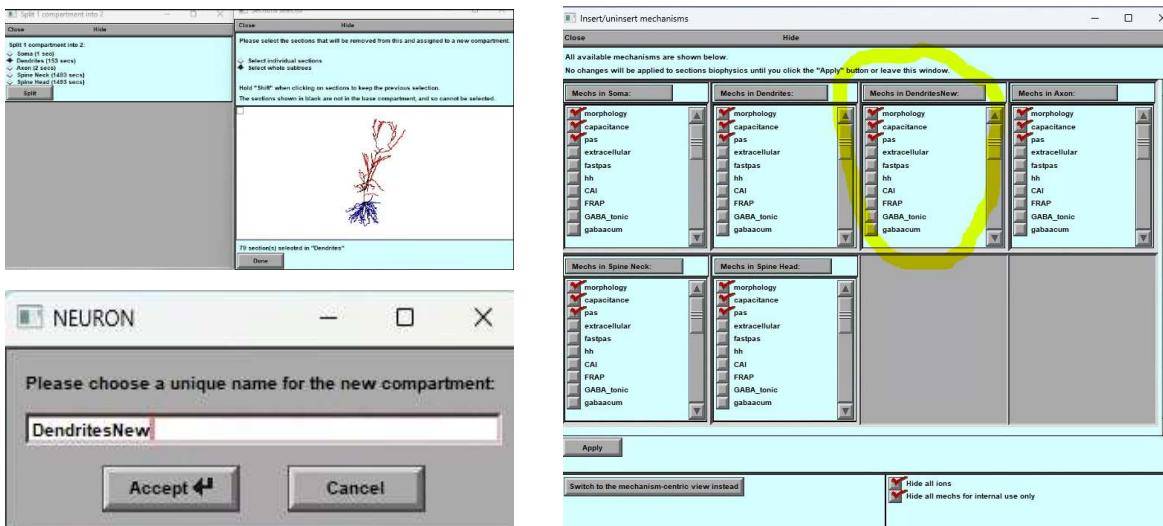
The **Split** function in BrainCell lets you divide any part of a cell into **multiple sub-compartments**, give them **custom names**, and assign **different biophysical mechanisms** to each. This is especially useful when you want to model **distinct properties** within a single region (e.g., dividing a dendrite into sections with different ion channel densities).

#### *Steps to Split a Compartment*

1. **Select the Compartment to Split**
  - o Choose the compartment you want to divide (e.g., a dendrite).
  - o Click the **Split** button.
  - o A graphical view of the selected compartment (such as the dendritic tree) will appear.
2. **Choose the Subparts**
  - o Select the sections to include in the new sub-compartment.
  - o Highlighted areas will appear **red**, making it clear which segments you've selected.
3. **Name the New Subpart**
  - o Click **Done** when your selection is complete.
  - o A prompt will appear to enter a **unique name** for the new sub-compartment (e.g., *DendritesNew*).
  - o Click **Accept** to confirm.
4. **Assign Biophysical Mechanisms**
  - o The new sub-compartment will automatically appear in the **Biophysical Mechanisms** panel.
  - o From here, you can add or edit its properties (e.g., ion channels, capacitance, synaptic inputs) separately from the original compartment.

This feature allows you to model **fine-grained, localised cellular behaviour**.

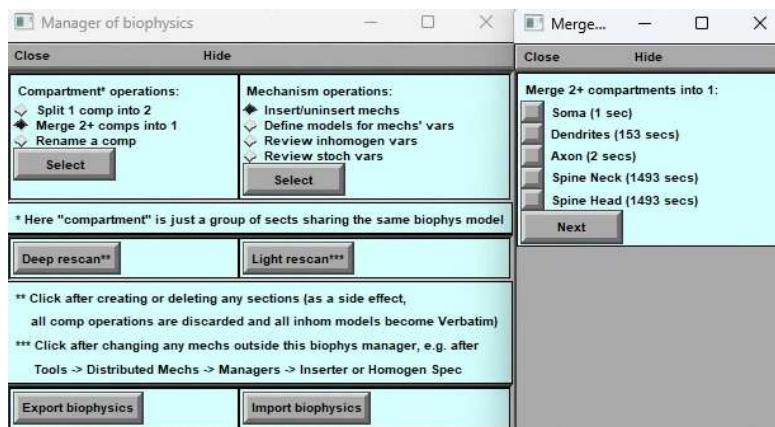
For example, you can divide a single dendrite into **over 150 segments**, giving each its own electrical or synaptic properties to match experimental data or test specific hypotheses.



## Merging Cell Compartments

The **Merge** function lets you combine **two or more compartments** into a single region. This is useful when you want to **simplify a model**, reduce the number of compartments, or make several regions share the same **biophysical mechanisms**.

### Steps to Merge Compartments:



1. **Open the Merge Tool**
  - o Go to the **Manager of Biophysics** panel.
  - o Under **Compartment Operations**, select **Merge 2+ compartments into 1**.
2. **Select Compartments to Merge**
  - o A list of available compartments will appear (e.g., *Soma*, *Dendrites*, *Axon*, *Spine Neck*, *Spine Head*).
  - o Tick the checkboxes next to the compartments you want to combine (for example, merge *Dendrites* and *Axon* into one).
3. **Confirm the Merge**
  - o Click **Next**.
  - o Review the prompt and confirm the operation.
  - o The selected compartments will be combined into one, sharing a **single set of biophysical mechanisms**.
4. **Adjust Biophysical Properties**
  - o After merging, check the new compartment under **Mechanism Operations**.
  - o Add or edit its properties (e.g., ion channels, passive membrane settings).

---

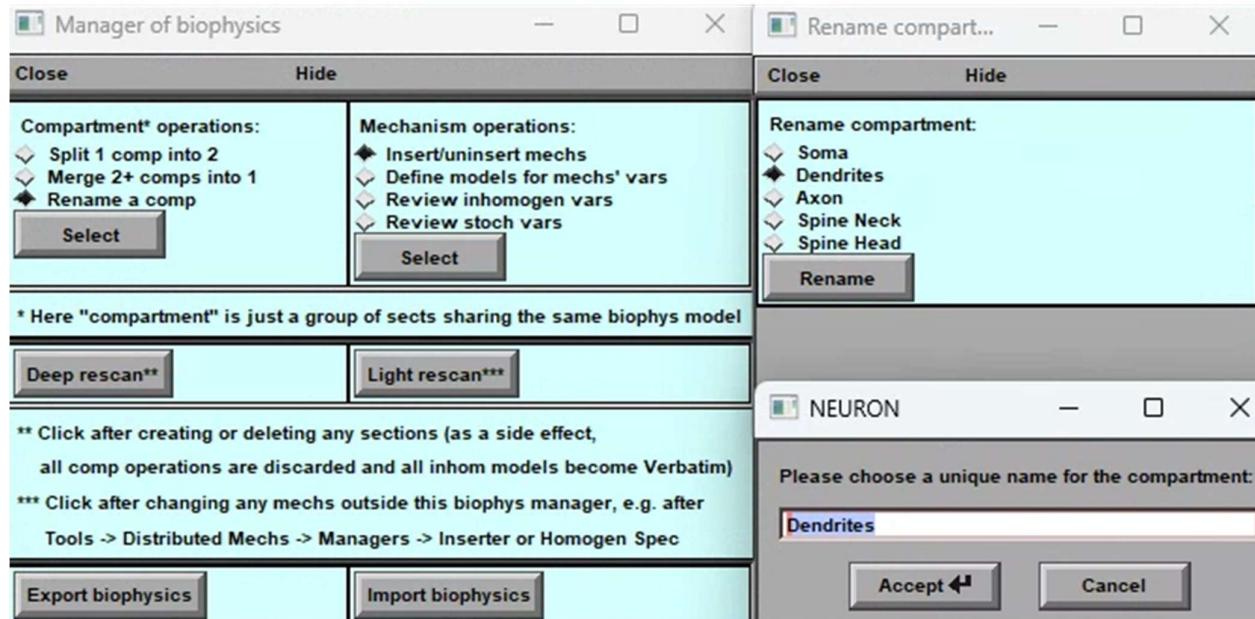
### Notes on Merging

- Use **Deep Rescan** or **Light Rescan** (in the Manager of Biophysics) to ensure that geometry and mechanism updates are correctly applied.
- Always verify the model after merging. Combining regions may alter simulation accuracy or behaviour.
- Merging is helpful when detailed compartmentalisation is unnecessary or when you want to **reduce computational load**.

## Renaming Cell Compartments

The **Rename** function allows you to assign **custom names** to existing compartments, making models easier to navigate and understand.

This is especially helpful when compartments represent specific **functional** or **anatomical** regions.



### Steps to Rename a Compartment

1. **Open the Rename Tool**
  - o Go to the **Manager of Biophysics** panel.
  - o Under **Compartment Operations**, click **Rename a comp**.
2. **Select the Compartment**
  - o A list of current compartments will appear (e.g., *Soma*, *Dendrites*, *Axon*, *Spine Neck*, *Spine Head*).
  - o Choose the one you want to rename.
3. **Enter a New Name**
  - o A pop-up window will ask for a **unique name**.
  - o Type the desired name (e.g., change *Dendrites* to *DendriticBranch1*).
  - o Click **Accept** to confirm.
4. **Check the Update**
  - o The new name will appear everywhere in the model interface.
  - o Perform a **Deep Rescan** or **Light Rescan** to ensure the change is applied throughout the model.

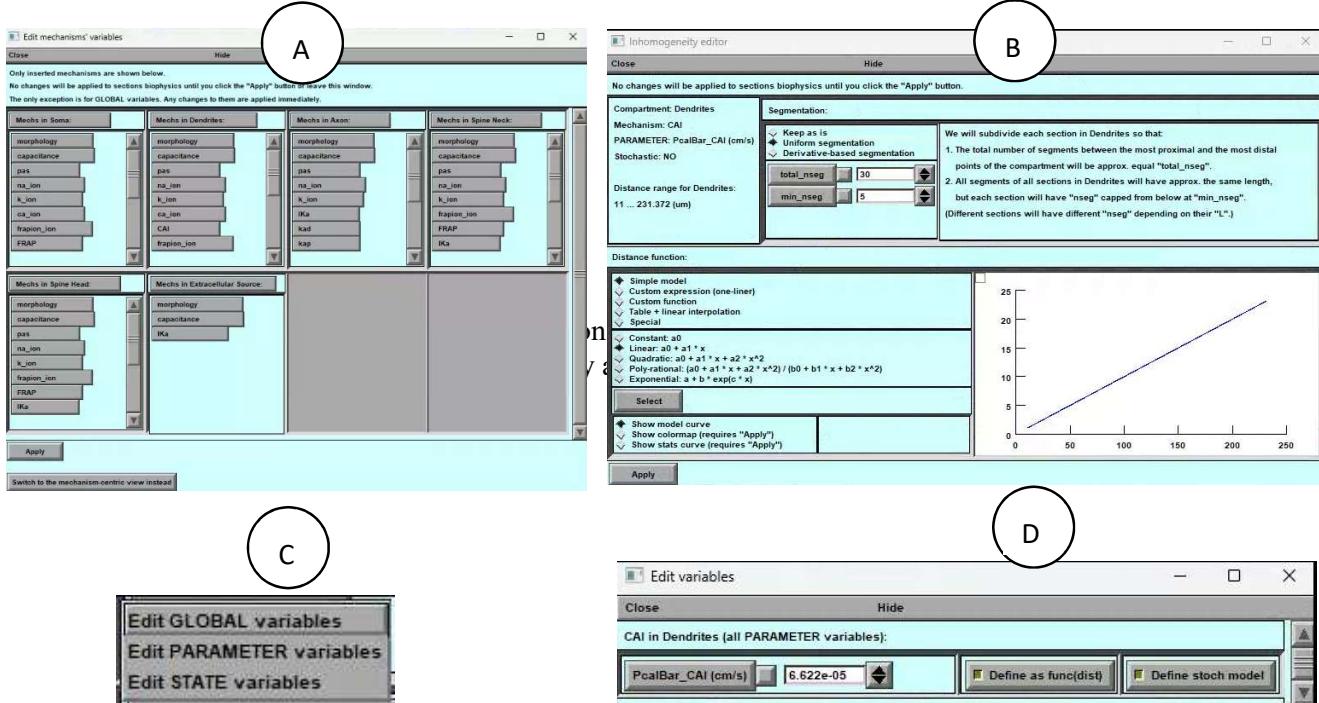
### Notes

- Each compartment name must be **unique** to avoid conflicts.
- Renaming only changes the **label** — it does **not** affect the geometry or biophysical mechanisms.
- Using clear, descriptive names makes it easier to work with **complex models**.



## Adjust the spatial distribution of mechanisms.

The panel (Figure 15) adjusts the spatial distribution of the mechanism across any part of the cell, allowing the user different options to define the mathematical formula for the spatial distribution.



**Figure 15: Panels for adding spatial properties to biophysical mechanisms in a cell.** A) Mechanism Selection Panel: Users can select a biophysical mechanism and choose to modify its spatial. B) Spatial Property Definition Panel: Users can define the spatial properties of the selected parameters or variables. C) Examples Panel: Displays examples of the types of parameters and variables that can be modified for each biophysical mechanism. D) Spatial Model Activation Panel: Users can enable the spatial properties of the selected parameter or variable by pressing the "Define as a function" button.

### Main Interface for Editing Biophysical Mechanisms

The **main interface** (Fig. 15A) provides a central view of the neuron, with each **segment** (soma, dendrites, axon, etc.) shown in its own panel.

Within each segment's panel, you can see and manage all the **biophysical mechanisms** assigned to that region.

#### Editing a Mechanism

1. **Select a Mechanism**
  - o Click on the mechanism you want to change.
  - o A context menu will appear (Fig. 15C).
2. **Choose What to Edit**
  - o The menu allows you to edit:
    - **Global variables**
    - **State variables**

- **Parameters**
    - Click any of these to open a detailed editing window (Fig. 15D).
  - 3. **Adjust Values and Heterogeneity**
    - In the editing window, you can view the **current value** of the selected variable (if it's spatially uniform).
    - To make the variable **spatially inhomogeneous**, click “**Define as a function of distance.**”
    - This opens the **Heterogeneity Editor** (Fig. 15B), where you can define how the variable changes across the cell.
- 

This interface provides direct, visual control over each neuron part and its mechanisms, enabling precise customisation of electrical and biophysical properties.

In the **Heterogeneity Editor**, you can define the variable as a function of distance. This allows you to customise the properties of the variable based on the spatial location. Once you've described it, you can save your changes (by clicking the Apply button) and edit other mechanisms.

#### *Spatial Inhomogeneity of Biophysical Mechanisms Editor.*

The **Heterogeneity Editor** (Fig. 15B) lets you define how a mechanism's properties vary across different **segments** of a neuron.

The window is divided into three main areas:

- The **upper panel** lists the segments of each neuron part (e.g., soma, dendrites, axon).
- Heterogeneity is defined **per segment**, not per exact physical coordinate.

For example:

- If a dendrite has only **one segment**, it will be treated as **uniform**, no matter its physical length.
- To achieve more detailed spatial variation, increase the **number of segments**.
- Keep in mind: **more segments = higher accuracy but longer computation time.**

#### *Spatial Inhomogeneity Specification*

BrainCell lets you define how a **biophysical mechanism** varies across different parts of the neuron. This is done in the **central panel** of the Heterogeneity Editor, which provides **five modes** for setting spatial inhomogeneity.

---

#### *Modes for Specifying Spatial Inhomogeneity*

1. **Simple Model**
  - Choose from predefined mathematical profiles:
    - **Constant** value
    - **Linear**
    - **Quadratic**

- **Polynomial** (2+ parameters)
    - **Exponential**
    - Useful for straightforward, analytical gradients.
  - 2. **Custom Function**
    - Define your own function directly in the interface.
    - Can be written in **NEURON hoc syntax** or **Python**.
    - Enter the function in the pop-up editor.
  - 3. **Custom Function from File**
    - Load a user-written function saved in an external file (NEURON or Python format).
    - Ideal for complex or pre-tested formulas.
  - 4. **Table Function**
    - Use **experimental data** to define heterogeneity.
    - Import values manually or upload a text file containing the data.
  - 5. **Special Function**
    - Visualise the neuron's **segmentation and discretisation**.
    - Gives precise control over how the cell is divided into segments for modelling.
    - Segments are colour-coded for clarity.
- 

### *Visualising Spatial Heterogeneity*

At the **bottom of the panel**, you can choose how to visualise heterogeneity:

- **Option 1 — Distance from Soma**

View heterogeneity as a function of distance from the soma to quickly spot regions with high or low variability.

- **Option 2 — Spatial Colour Map**

Display a **colour gradient** directly on the neuron:

- Cool colours = low heterogeneity
- Warm colours = high heterogeneity

These visualisation tools help you **verify and refine** the spatial variation of your chosen mechanism.

## Adjust the stochastic distribution of mechanisms.

The central panel (**Figure 16**) adjusts the mechanism's stochastic properties across any part of the cell, allowing the user to define the mathematical formula for the stochastic distribution using different options.



**A)** Mechanism Selection Panel: Users can select a biophysical mechanism and choose to modify its stochastic properties.

**B)** Stochastic Property Definition Panel: Users can define the stochastic properties of the selected mechanism's parameters or variables.

**C)** Examples Panel: This panel displays examples of the types of parameters and variables that can be modified for each biophysical mechanism.

**D)** Stochastic Model Activation Panel: Users can enable the stochastic properties of the selected parameter or variable by pressing the "Define stochastic model" button.

To begin editing any mechanism, click on it. This will bring up a menu where you can select what you want to edit. The menu includes options for global variables, state variables, and parameters. You can access another window that provides more detailed information about the selected variable by clicking on any of these options.

In the new window, you'll see the current value of the variable if it's spatially homogeneous and "nan" is spatial non-uniform. To make this mechanism stochastic, click the "**Define as a stochastic model**" bar. Doing this will open a new window called the **Stochasticity Editor**.

## *Stochasticity Editor*

The **Stochasticity Editor** lets you define how a variable behaves as a **stochastic (random) function over time**.

You can choose from built-in noise models, import your own, or use experimental data to describe variability.

Once defined, you can save the settings and continue editing other mechanisms.

The window has **four main sections**:

### *1. Upper Panel — Variable Information & Bounds*

- **Left side:**
  - Shows key information about the selected variable/parameter:
    - Whether it is **inhomogeneous**,
    - The **mechanism** it belongs to,
    - Its **location** (compartment),
    - **Full name and units**.
  - Indicates whether the parameter is already spatially inhomogeneous.
- **Right side — Bounding:**
  - Define the **domain** of the variable's probability density function (PDF):
    - *Keep as is* (no bounds)
    - *Bound from above*
    - *Bound from below*
    - *Bound from both sides*
  - A description of each bounding mode appears on the right side of the panel.

---

### *2. Stochastic Model*

Choose the type of stochasticity and its distribution:

- **Noise Type:**
  - *White noise*
  - *Coloured noise* (with power spectral density (PSD) defined by different  $\alpha$  correlations)
- **Model Options:**
  - **Simple Model** — standard distributions:
    - *Uniform, Normal, Logarithmic, Exponential, Erlang, Weibull*
    - Each selection opens a pop-up panel for entering model parameters.
  - **Custom Expression** — write your own stochastic function inline (in **NEURON hoc** or **Python**, available to premium users).
  - **Custom from File** — load a pre-written stochastic function from an external hoc or Python file.
  - **Table & Interpolation** — import experimental data to define the distribution.
  - **Special Functions** — predefined simple test functions, e.g.:
    - $(1, 0, 0, \dots)$  — delta function with uniform spectrum
    - $(1, 1, 1, \dots)$  — constant (zero frequency)
    - $(1, 0, -1, 1, 0, -1, \dots)$  — cosine at half Nyquist frequency
    - $(1, -1, 1, -1, \dots)$  — cosine at Nyquist frequency
    - *Foo* — simple periodic test function

---

### *3. Visualisation & Modelling*

Tools to explore the selected stochastic model:

- Plot the **probability density function (PDF)** for infinite trials.
  - Generate a **sample** of random numbers from the chosen distribution.
  - Compare the sample's **PDF** to the ideal distribution.
  - Build the **autocorrelation function (ACF)** for the sample.
  - Plot the **power spectral density (PSD)** of the sample.
- 

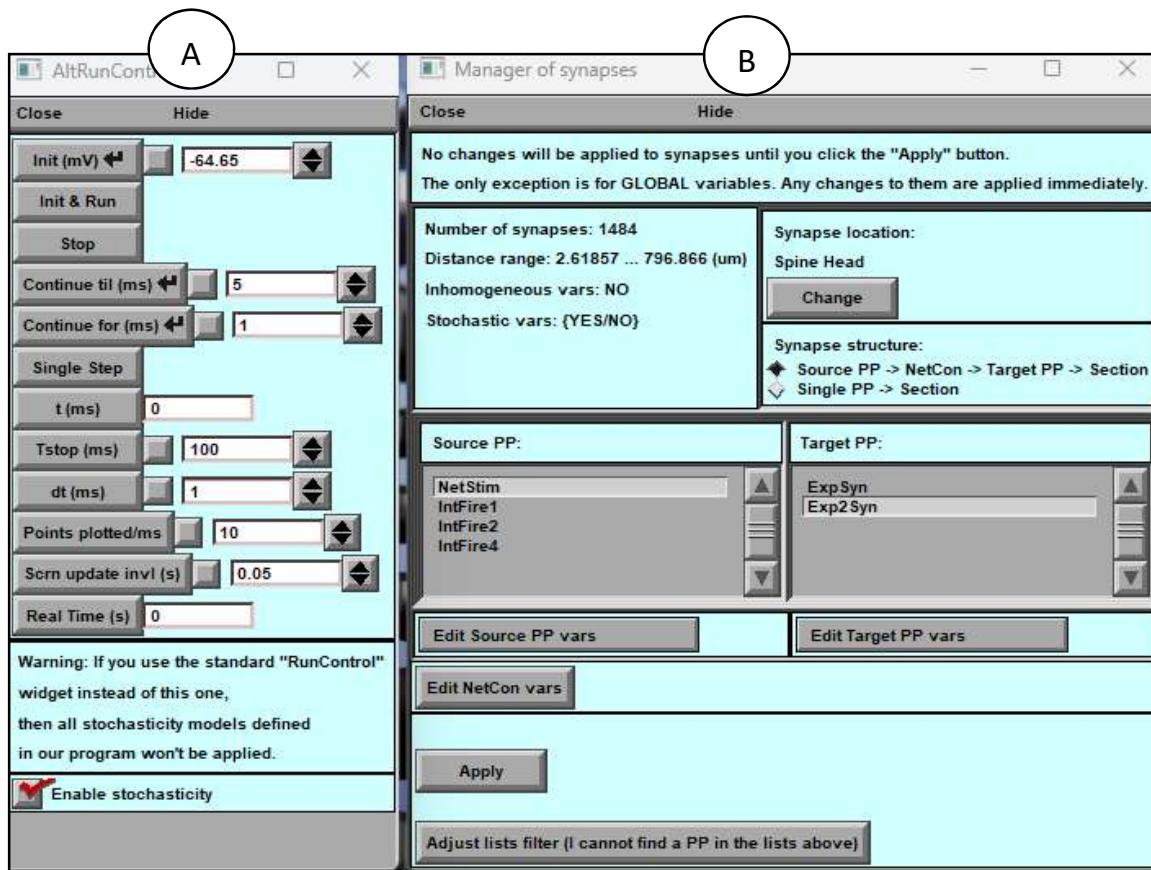
### *4. Apply or Disable*

- **Apply** — confirm and use the stochasticity settings for the variable.
- **Disable** — cancel stochastic behaviour for this variable.

**Note:** Stochastic dynamics can be run and tested using the **special run control panel** “AltRunControl” provided by BrainCell.

## Manager of synapses.

The following information is vital for those who wish to create synaptic inputs.



**Figure 17. The Synaptic Simulations. A) Panel of Alternative Run Simulation for Stochastic Variables:** The panel demonstrate alternative run simulations for stochastic variables within the synaptic environment. **B) Panel of Manager of Synaptic and Presynaptic Mechanisms:** The manager acts as a central control system, orchestrating the interactions between different synaptic elements and ensuring efficient communication within the neural network (Details in the text).

### Synaptic Panel

The **Synaptic Panel** (Fig. 17B) allows you to **create and distribute synaptic inputs** across a neuron's dendritic tree.

The **number of synapses** is automatically matched to the **number of spines** defined in the main geometry panel (Fig. 10).

**⚠ Important:** You **cannot change the number of synapses** here — adjust spine number and density in the main panel instead.

What you **can** do in this panel:

- **Relocate synapses** — place them directly on a spine or position them nearby on the dendrite.

- **Control synaptic distribution** — choose what fraction of synapses are on spines vs. on dendritic shafts.

### Functional effect:

- If a synapse is placed **on a spine**, the spine acts as an **active electrical conductor**.
- If the synapse is placed **next to the spine**, the spine's conductance becomes zero (electrically silent), but its geometry may still affect local signalling.

💡 To include **synaptic release probability** in your simulation, enable stochasticity via **AltRunControl** and check “**Enable stochasticity**.”

---

### *Layout of the Synaptic Panel*

The panel is divided into **three main sections**:

1. **Top Section — Overview & Controls**
    - Displays key information about existing synapses and provides tools to **move or adjust their locations**.
  2. **Middle Section — Synapse Placement**
    - Visualises the dendritic tree and spine distribution.
    - Lets you relocate synapses to the spine head or nearby shaft.
  3. **Bottom Section — Synaptic Properties & Efficacy**
    - Provides controls for setting **synaptic strength** and **release probability** (if stochasticity is enabled).
    - Supports exploration of how synapse placement affects local electrical behaviour.
- 

This panel is designed to help you **understand and fine-tune synaptic organisation and function** in your neuron model.

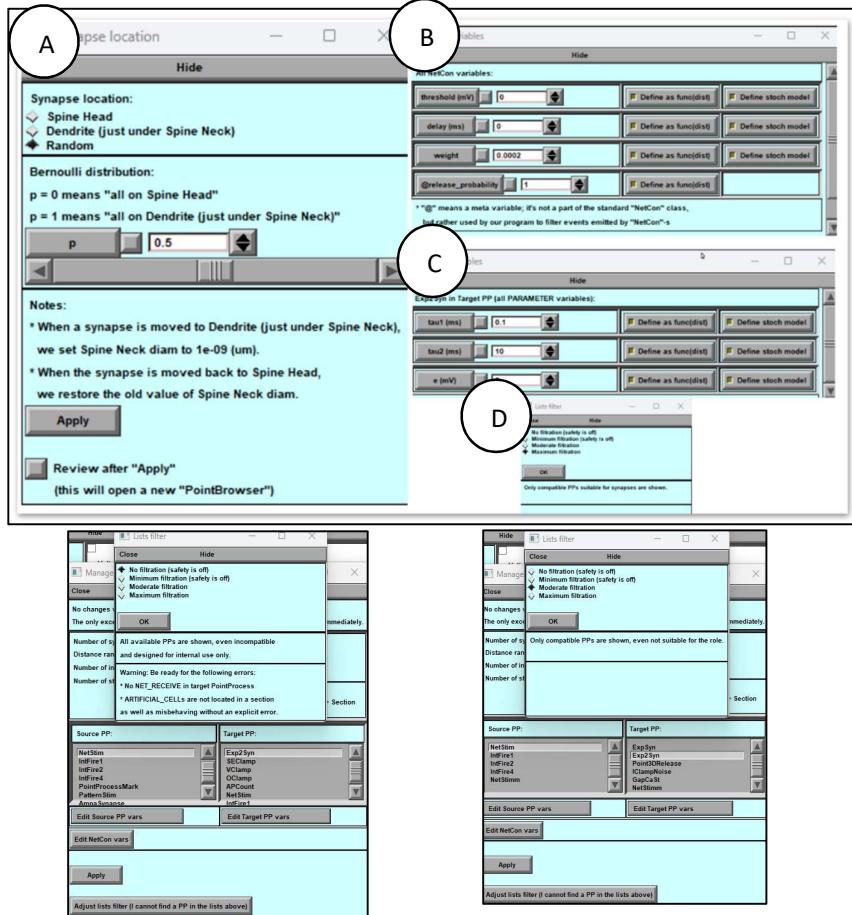
1. [Synaptic manager upper part](#).
2. [Synaptic manager middle part](#).
3. [Synaptic manager bottom part](#).

### *Synaptic manager “upper part”*

The information section in the upper portion of the panel is where you can find valuable data about the synapses on the dendritic tree. This section will indicate the number of synapses distributed throughout the dendritic tree, including the spatial range and whether the parameters are spatially homogeneous. Additionally, it will provide you with stochastic variables of its parameters, which is crucial for understanding the behaviour of the synapses.

To change the location of the synapses on spines or dendrites, use the tools located in the upper right portion of the panel. By manipulating these tools, you can adjust the position of the synapses to your desired location.

The “**Synapse location**” button opens a pop-up panel (**Fig.18A**).



**Figure 18: Pop-Up Panels of Synaptic Manager.** A) shows the synaptic location manager, allowing users to visualise the spatial distribution of synapses. B) displays the manager of presynaptic signals, which provides information about the presynaptic neuron's activity. C) presents the manager of synaptic parameters, where users can modify synaptic properties such as strength and plasticity. D) shows the manager of presynaptic and postsynaptic mechanisms, located in the mechanism directory, which enables users to customise the synaptic transmission and reception mechanisms.

#### Synapse Location Panel

The **Synapse Location** panel (Fig. 18A) is divided into two main sections: **Upper** and **Lower**.

##### Upper Part — Choosing Synapse Placement

Here you can decide how synapses are positioned relative to spines and dendrites. Three placement options are available:

1. **All synapses on spines** – every synapse is placed directly on a spine head.
2. **All synapses on dendrites (at spine locations)** – synapses sit on the dendritic shaft where spines would normally attach.
3. **Mixed distribution** – split synapses between spines and dendrites in a user-defined proportion.

Select the option that best fits your simulation. The software will automatically adjust the synapse locations.

---

#### *Lower Part — Information & Actions*

- **Information display:** shows current synapse **density** and **placement**.
    - Updates to reflect any changes after you choose a new placement option.
  - **Apply** — confirms and saves the new synapse distribution.
  - **Cancel** — discards changes if you don't want to keep them.
  - **Review after Apply** — visualises the updated synapse placement so you can verify that it matches your intended configuration.
- 

This panel makes it easy to **experiment with different synapse placement strategies** and immediately check how they affect your neuron model.

#### *Synaptic Structure — Synaptic Manager (Top Part)*

The **top section** of the Synaptic Manager (Fig. 18B) lets you define the **type of synaptic connections** in your neuron model.

BrainCell supports **two types of synapses**:

1. **Presynaptic + Postsynaptic**
  - Requires an incoming **presynaptic signal**.
  - Connection path:  
Source PP → NetCon → Target PP → Section
2. **Postsynaptic Only**
  - Functions **without a presynaptic input**.
  - Activated by the “**onset**” parameter (defines when the synapse becomes active).
  - Connection path:  
Source PP → Section

Select the desired synapse type in the **top panel**.

The chosen structure will appear in the **middle section** of the Synaptic Manager for further editing.

#### **⚠ Note:**

The “**probability of release**” parameter is **not available** when using the **presynaptic-only** configuration

(Source PP → NetCon → Target PP → Section).

#### *Synaptic Manager — Middle & Bottom Sections*

The **middle section** of the Synaptic Manager (Fig. 18B) is designed to let you **organise and configure presynaptic and postsynaptic mechanisms** with full control over their parameters.

## Middle Section — Presynaptic & Postsynaptic Mechanisms

- The interface has **two panels**:
  - **Left panel — Source PP**: lists **presynaptic mechanisms**.
  - **Right panel — Target PP**: lists **synaptic (postsynaptic) mechanisms**.
  - Both panels display standard **NEURON mechanisms** and unique mechanisms included in the BrainCell library.
- **Finding unique mechanisms:**
  - If you don't see a required mechanism, click "**Adjust lists filter**."
  - A pop-up window will appear, allowing you to apply **filters** and reveal all available **point mechanisms** in the BrainCell folder.
  - Fig. 18 shows two filter levels to help you locate and insert the correct mechanism.
- **Editing parameters:**
  - After selecting the desired mechanisms, use:
    - **Edit source PP vars** – adjust presynaptic parameters.
    - **Edit target PP vars** – adjust postsynaptic parameters.
- **Spatial distribution:**
  - Point-based (synaptic) mechanisms follow the **same distribution rules** as standard density mechanisms (see *Manage Distribution Mechanisms* section).
- **Stochastic behaviour:**
  - Randomness in synaptic parameters is triggered **when a presynaptic signal occurs** (or when the **onset time** is reached).
  - The stochasticity control panel works the same way as for spatial mechanisms.

### Note:

In **Source PP → Section** mode, you can also add other **point mechanisms** such as electrodes (potential clamp), external current injections, or any localised current/potential changes supported by NEURON.

## Bottom Section — NetCon & Advanced Controls

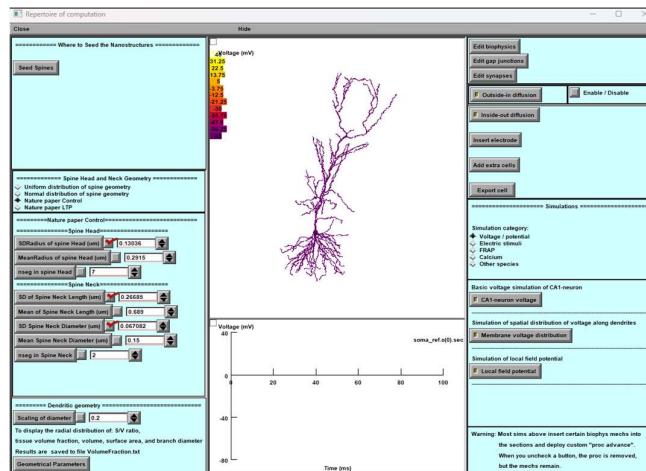
The **bottom section** manages the **NetCon** connection between presynaptic and synaptic mechanisms.

- Click **Edit NetCon vars** to set:
  - **Threshold**
  - **Delay**
  - **Weight**
  - **Release probability**
- Use **Adjust lists filter** (Fig. 18D) to **add custom synaptic mechanisms** not currently shown in the middle panels.
  - New mechanisms should be placed in the **BrainCell/Mechanisms** directory before adding.
- Click **Apply** to confirm all changes to your synaptic configuration

This combined interface gives you **full control** over synapse type, spatial distribution, mechanism parameters, and stochastic behaviour, enabling highly customised synaptic modelling.

## Spatially - non-uniform sources of extracellular ions.

This section provides detailed instructions for organising the calculation of ion concentration near a model cell in BrainCell, regardless of the concentration source.



### Main Panel Functions:

The main panel features two key buttons:  
**Outside-In Diffusion** and **Inside-Out Diffusion**.

#### 1. Outside-In Diffusion:

- This option enables users to create and manage different ion concentration sources outside the cell.
- Steps to use:
  1. Click on "Outside-In Diffusion".
  2. Define the type and properties of the ion concentration source.

3. Configure spatial and temporal parameters to simulate diffusion towards the cell.

#### 2. Inside-Out Diffusion:

- This option calculates the ion concentration dynamics that the cell releases.
- Steps to use:
  1. Click on "Inside-Out Diffusion".
  2. Specify the type of ion and its release dynamics.
  3. Set up the simulation to observe how the released ions affect the surrounding environment over time.

These features allow for comprehensive modelling and simulation of ion concentration changes around a model cell, providing insights into both external and cell-released ion dynamics.

The "Neuron" software starts with a uniform distribution of extracellular ions. Still, sometimes researchers need to study the specific spatiotemporal dynamics of these ions, including different types, neurotransmitters, and neuromodulators. To help with this, BRAINCELL has added a feature

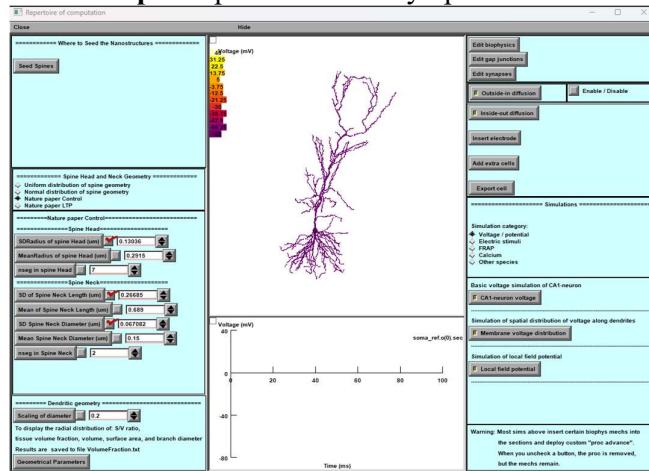
**Edit extracellular sources (in development)** that lets users create different scenarios for ion dynamics. To access this feature, look for the option in the upper right corner of the main panel and select it. This will take you to the extracellular ion distribution editor. It's important to remember that using this editor doesn't consider the potential effect of neuron feedback on ion concentrations.

However, it allows researchers to conduct more detailed investigations into ion dynamics, leading to a better understanding of how neurons behave and interact in the extracellular environment.

### ***Ion Concentration Modelling***

BrainCell includes tools to **simulate ion concentration dynamics** around a model cell — whether ions originate **outside** the cell or are **released** from it.

The main panel provides two key options:



model diffusion toward the cell.

### 1. Outside-In Diffusion

Simulates how external ion sources affect the cell.

#### Steps:

1. Click **Outside-In Diffusion**.
2. Define the **ion type** and its source properties (e.g., concentration, release pattern).
3. Set **spatial** and **temporal** parameters to

### 2. Inside-Out Diffusion

Simulates ion release from the cell and its spread into the extracellular space.

#### Steps:

1. Click **Inside-Out Diffusion**.
2. Specify the **ion type** and its **release dynamics**.
3. Configure the simulation to observe how released ions disperse over time.

These features let you create **custom extracellular ion scenarios**, going beyond NEURON's default uniform ion distribution.

You can model the dynamics of ions, neurotransmitters, or neuromodulators to study their local effects.

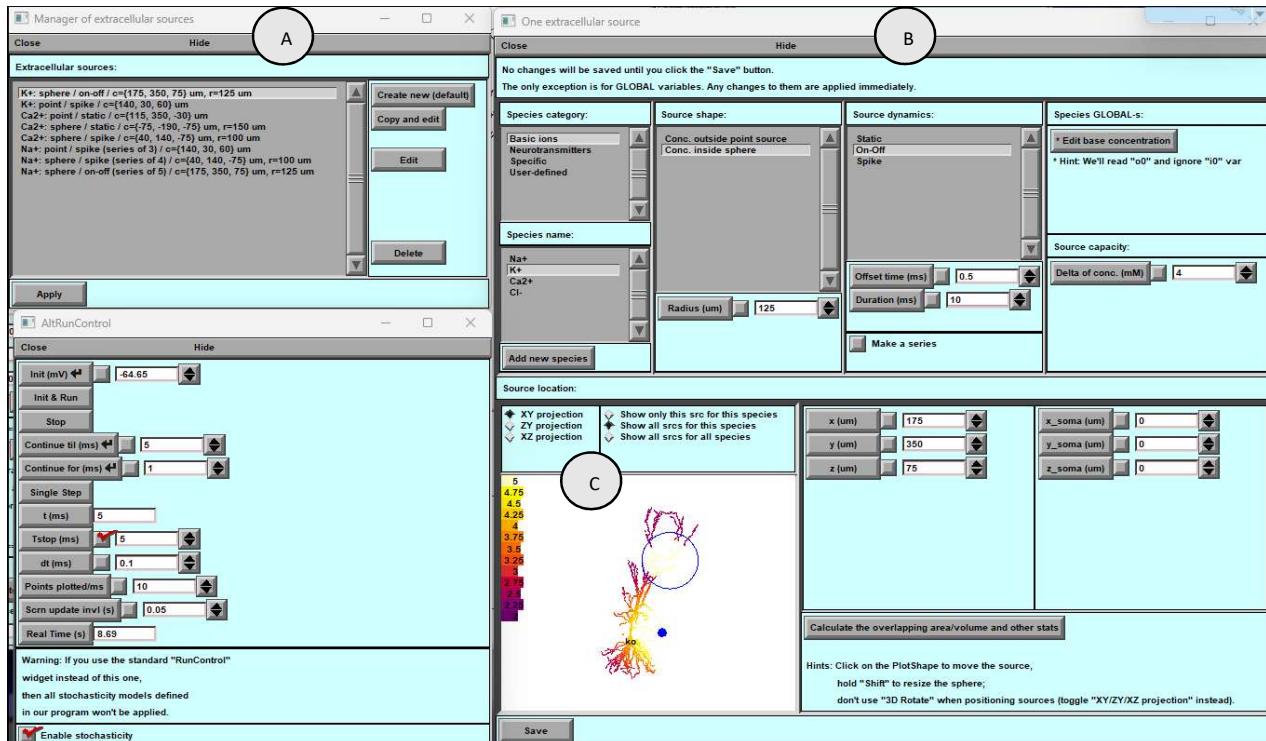
#### ⚠ Important:

The extracellular ion editor does **not** account for **feedback from the neuron itself** on ion concentration changes.

It is intended for **controlled investigations** of ion dynamics rather than full bidirectional coupling.

### Outside-In Diffusion

The Inside-Out Diffusion (Figure 19) is a tool for simulating extracellular ion dynamics release by **external courses**. This chapter will guide you through its key features and how to use them effectively.



**Fig. 19. External ionic concentration**

**A) The existing external sources panel** is a convenient tool for viewing and managing previously created external sources. Users can select an external source from this list to edit or delete it.

**B) The external source creation panel** allows users to design software using three templates: point single release, point constant diffusion, and spherical area of concentration. Users can specify the location and properties of the external source.

**C) The visual graphical panel** allows users to specify the coordinates and area of an external source by dragging and dropping. Users can also use Shift to specify the radius of the external source.

### Extracellular Ion Concentration Editor — Overview (Fig. 19)

The **Extracellular Ion Editor** lets you precisely control and visualise ion concentrations around the model cell, adapting to complex spatial and temporal scenarios.

#### 1. Modifying Ion Concentrations

- You can create **multiple external ion sources** that may **overlap in space or time**.
- It is also possible to define **separate or overlapping sources** of the **same ion type** with different spatial/temporal profiles.

#### 2. Workflow

- After editing any ion source, click **Save**.
- Press **Apply** (Panel A) to implement the changes in your simulation.

---

## Panels & Tools

### Panel A — Extracellular Concentration Changes

- Central view listing all **active ion sources** and their parameters.
- Allows you to **add, edit, or delete** ion sources.

### Panel B — Ion and Concentration Selection

- Choose which **ion or molecule** to simulate.
- Built-in options include:
  - Major ions:  $Na^+$ ,  $K^+$ ,  $Cl^-$ ,  $Ca^{2+}$
  - Neurotransmitters: *acetylcholine*, *glutamate*, *GABA*
  - Special options: *FRAP*, *inositol triphosphate (IP<sub>3</sub>)*
- You can also define **custom ions** (*Ion 1*, *Ion 2*).

### Source Shape

- Define the **spatial distribution** of ions — for example, within or outside a **spherical region**.

### Source Dynamics

- Control how the ion source changes over time:
  - **Constant activation**
  - **Instantaneous activation**
  - **Toggling on/off**
- Each source includes a settings window for adjusting its parameters.

### Concentration Series

- Create a **time series** of concentration changes.
- Customise:
  - Start time and interval length
  - Add **random noise** for more realistic fluctuations.

### Spatial Localisation & Visualisation

- Position ion sources in 3D space.
- View distributions in **XY**, **XZ**, or **YZ planes**.
- Switch views: **single ion**, **ion type**, or **all ions combined**.
- Interact with the **graphical representation** — right-click to open the “**Extracellular Concentration Dynamics**” graph.

---

## Summary

These tools allow you to **design complex ion environments**, combine multiple sources, and visualise how extracellular ion concentrations evolve over time and space.

This helps you explore how neurons interact with their chemical surroundings under different experimental or simulated conditions.

# Outside-In / Inside-Out Diffusion Equations

The extracellular diffusion modules simulate how ions or neurotransmitters spread between sources and the surrounding space.

BrainCell distinguishes two complementary cases:

- **Outside-In:** diffusion **toward** the cell from an external source.
- **Inside-Out:** diffusion **away** from the cell after release.

The following formalism applies to both; only the sign of the flux differs.

---

## 1. Fundamental Diffusion Equation

$$\frac{\partial C(r, t)}{\partial t} = D \nabla^2 C(r, t) + S(r, t)$$

where

- $C(r, t)$ — concentration (mM) at position  $r$  and time  $t$
- $D$ — diffusion coefficient ( $\mu\text{m}^2 \text{ ms}^{-1}$ )
- $S(r, t)$ — source (positive) or sink (negative) term (mM  $\text{ms}^{-1}$ )

For a spherically symmetric point source:

$$\frac{\partial C}{\partial t} = D \left( \frac{\partial^2 C}{\partial r^2} + \frac{2}{r} \frac{\partial C}{\partial r} \right) + S(r, t)$$

---

## 2. Instantaneous Point-Source Solution

For a single pulse of total quantity  $Q_0$  released at  $r = 0$  and  $t = 0$ :

$$C(r, t) = \frac{Q_0}{(4\pi Dt)^{3/2}} \exp\left(-\frac{r^2}{4Dt}\right)$$

This describes how concentration falls off with distance and time.

---

## 3. On-Off (Time-Limited) Source

For a source active between  $t_{start}$  and  $t_{end}$ :

$$C(r, t) = \frac{Q_0}{4\pi r D} \left[ \operatorname{erfc}\left(\frac{r}{2\sqrt{D(t - t_{start})}}\right) - \operatorname{erfc}\left(\frac{r}{2\sqrt{D(t - t_{end})}}\right) \right]$$

where  $\text{erfc}()$  is the complementary error function.

For  $t < t_{start}$ ,  $C = 0$ ; for  $t > t_{end}$ , diffusion proceeds passively.

---

## 4. Time-Factor Function

In BrainCell, the temporal profile of a finite-duration release is wrapped into a dimensionless function `timeFactor(Δt)`:

$$\text{timeFactor}(\Delta t) = H(t - t_{start})[1 - e^{-(t-t_{start})/\tau_{rise}}]e^{-(t-t_{end})/\tau_{decay}}$$

where

- $H()$ — Heaviside step function,
- $\tau_{rise}, \tau_{decay}$ — time constants (ms) for onset and decay of release.

This function smoothly approximates a rectangular pulse and avoids numerical discontinuities.

---

## 5. Combined Expression for Extracellular Concentration

$$C(r, t) = C_0 + \frac{Q_0}{4\pi r D} \text{timeFactor}(\Delta t) \text{erfc}\left(\frac{r}{2\sqrt{Dt}}\right)$$

where  $C_0$  is the baseline extracellular concentration (mM).

---

## 6. Parameter Notes

Symbol	Meaning	Typical value
$D$	Diffusion coefficient	$0.3 \mu\text{m}^2 \text{ ms}^{-1}$ (for $\text{K}^+$ ); $0.5 \mu\text{m}^2 \text{ ms}^{-1}$ (for Glu)
$Q_0$	Released quantity	user-defined
$C_0$	Basal concentration	2.5 mM for $\text{K}^+$ , 1.5 mM for $\text{Na}^+$
$t_{start}, t_{end}$	Activation window	in ms
$\tau_{rise}, \tau_{decay}$	Rise/decay time constants	10 ms / 200 ms typical
$r$	Distance from source	$\mu\text{m}$

---

## Summary

These equations define how ions or neurotransmitters diffuse from (or toward) a point or spherical source over time.

They are implemented in BrainCell's diffusion modules (Outside-In

### ***Manager of Extracellular Sources Panel***

**Description:** When the user activates the extracellular sources function, the first panel displays all the current ion concentration sources in the neuron cell. Initially, several predefined sources are available but remain inactive until manually activated by the user.

#### **Instructions:**

1. **Viewing Sources:** The panel shows a list of all available ion sources.
2. **Activating Sources:** Activate each source to start ion release.
3. **Editing Sources:** Double-click any source to open the main configuration panel (Panel B).
  - o **Panel B Features:**
    - Change ion types
    - Modify spatial distribution
    - Adjust time dynamics
    - Set basic parameters and properties

Use these tools to customize the extracellular ion environment around your model cell effectively.

### ***Single Ion Panel***

The **Single Ion Panel** lets you add and configure **predefined or custom ions** to simulate their dynamics around the model cell.

---

#### ***Predefined Ions***

BrainCell includes a library of commonly used ions and molecules:

- **Main ions:** Potassium ( $K^+$ ), Sodium ( $Na^+$ ), Chloride ( $Cl^-$ ), Calcium ( $Ca^{2+}$ )
- **Neurotransmitters:** Acetylcholine (ACh), Gamma-aminobutyric acid (GABA), Glutamate (Glu)
- **Helper molecules:** Inositol trisphosphate ( $IP_3$ ), FRAP (Fluorescence Recovery After Photobleaching)

---

#### ***Creating Custom Ions***

You can define new ions or molecules by setting their **spatial** and **temporal** properties.

##### *1. Spatial Distribution*

How the ion source is positioned and spreads:

- **Point Source** – ions released from a single point.
- **Spherical Source** – uniform release within a defined sphere.

- **Advanced distributions** – configure more complex patterns as needed.

## 2. Temporal Dynamics

How ion concentration changes over time:

- **Static** – constant concentration.
- **On / Off** – turn release on or off at specified times.
- **Instantaneous** – release a set amount at a single time point.

You can adjust parameters such as **start time**, **release radius**, and duration.

## 3. Global Parameters

- **Diffusion coefficient** – how fast ions spread.
  - **Temperature dependence** – adjust diffusion based on temperature.
  - **Ion absorption** – enable uptake by the extracellular space using a linear absorption model.
- 

### *Positioning Ion Sources*

- **Graphical interface:** drag and drop ion sources directly onto the cell's 3D view.
  - **Numerical input:** enter precise X, Y, Z coordinates.
  - **Move soma:** reposition the entire soma using exact coordinates if needed.
- 

### *Saving Your Ion Setup*

- After defining a custom ion, click **Save** to add it to the **main ion selection panel** for future use.
  - Always click **Save** and then **Apply** to confirm changes and activate them in your simulation.
- 

### *Inside–Out Diffusion*

The **Inside–Out Diffusion** tool simulates how a cell releases ions into the extracellular space and how these ions spread over time.

It is designed to give you full control over **simulation precision, computational cost, and visualisation**.

---

#### *1. General Settings (Panel A)*

- Select the **simulation type** and **computational resource** (local CPU, GPU if available, or external cluster).

- Define **extracellular space parameters** — these strongly affect simulation accuracy.
  - Adjust:
    - **Nano-geometry toggle** – include fine-scale geometry for higher accuracy.
    - **Precision settings** – balance between detailed modelling and faster runtime.
- 

## 2. Diffusion Management (Panel B)

- Choose the **calculation mode**:
    - **On-the-fly** – calculates during the simulation (dynamic but slower).
    - **Deferred** – precomputes diffusion for faster simulation runs.
  - Select the **ion species** to include (e.g., Na<sup>+</sup>, K<sup>+</sup>, Ca<sup>2+</sup>, glutamate).
  - Configure the **spatiotemporal grid** and **recording period** to balance detail vs. computational efficiency.
- 

## 3. Simulation Control (Panel C)

- Use the **AltRunControl** panel to:
    - Set **initial conditions**.
    - Manage **runtime parameters**.
    - Control simulation start, pause, and stop.
    - Enable **stochasticity** for more realistic ion release dynamics.
- 

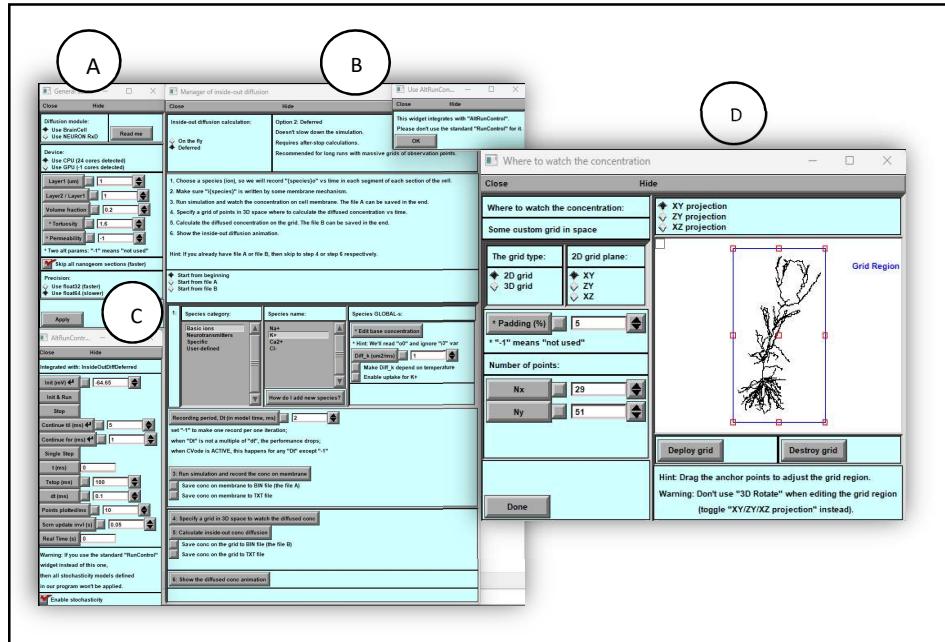
## 4. Visualisation Configuration (Panel D)

- Adjust how diffusion results are displayed:
    - Choose **projection views** (XY, XZ, YZ) for accurate placement.
    - Define your **region of interest** interactively.
    - Select **grid type** and **resolution** for visual clarity.
  - **Tip:** Use **projection views** (XY/YZ/XZ) to define your grid region precisely — avoid relying solely on 3D rotation for placement.
- 

With these panels, you can run **highly detailed extracellular diffusion simulations** that reveal how ions move and interact in complex cellular environments.

Use the “**Read Me**” buttons in each section for more detailed feature-specific instructions.

---



**Figure 20 — Inside–Out Diffusion Control Panel**

The **Inside–Out Diffusion Control Panel** provides all the tools needed to set up, run, and visualise extracellular ion diffusion simulations in BrainCell.

#### A. General Settings

- **Simulation type:** choose **NEURON RxD** or **BrainCell** engine.
- **Computation resource:** run on **CPU** or **GPU**.
- **Extracellular space configuration:**
  - Layer dimensions (pre-membrane and exchange layers)
  - Volume fraction
  - Tortuosity
  - Permeability (*optional*)
- **Nano-geometry toggle:** include or exclude detailed nano-structures.
- **Precision:** select **64-bit** or **32-bit** calculations.

#### B. Inside–Out Diffusion Manager

- **Calculation mode:**
  - *On-the-fly* – compute during the simulation (dynamic but slower)
  - *Deferred* – precompute before running (faster execution)
- **Computation initiation point:** choose where the simulation starts.
- **Ion species selection & configuration:**
  - Main ions: K<sup>+</sup>, Na<sup>+</sup>, Cl<sup>-</sup>, Ca<sup>2+</sup>
  - Neurotransmitters: ACh, GABA, Glutamate
  - Helper ions: IP<sub>3</sub>, FRAP

- **Custom ion creation** supported
  - **Spatiotemporal grid:** define resolution and time steps.
  - **Recording period:** set how often data are saved.
  - **Simulation execution & data storage:** manage where and how results are saved.
  - **Concentration visualisation settings:** configure display options (see Panel 20 D).
  - **Fixed-concentration extracellular space:** option for static environments.
  - **Movie generation:** create animated visualisations of ion diffusion.
- 

#### C. AltRunControl Panel

- Control **simulation runtime parameters**.
  - Define **initial conditions**.
  - Start, pause, or stop simulation progression.
  - Manage **recording periods and data output**.
  - Enable and manage **ion release from cells**.
- 

#### D. Concentration Visualisation Configuration

- **Projection selection:** XY, XZ, or YZ 2D views.
- **Grid type:** choose 2D plane or full 3D grid mapping.
- **Axis visibility:** toggle X, Y, Z axes.
- **Padding adjustment:** fine-tune the border around the analysis region.
- **Grid region definition:** interactive graphical tool;
  - Drag anchor points to adjust region of interest.
  - **Tip:** use XY/YZ/XZ projections for precise placement (avoid rotating in 3D mode).
- **Resolution control:** set the number of points along Nx and Ny for visual detail.
- **Central display:** shows the neuron model with the active grid overlay.

💡 **Tip:** Use the “Read Me” buttons in each section for detailed instructions.

---

#### Summary

These panels work together to let you:

- Precisely define **extracellular space properties**,
- Configure **ion sources and diffusion parameters**,
- Control **simulation runtime**, and
- Create **high-resolution visualisations and animations** of ion release and movement.

This gives you full flexibility for **advanced extracellular ion dynamics modelling**.

## *Gap Junctions in BrainCell*

### *Gap Junction Sets in Astrocytes*

BrainCell allows you to **configure and manage gap junction sets** in astrocytes with full control over their:

- **Biophysical properties**
- **Spatial distribution**
- **Quantity**
- **Stochastic behaviour**

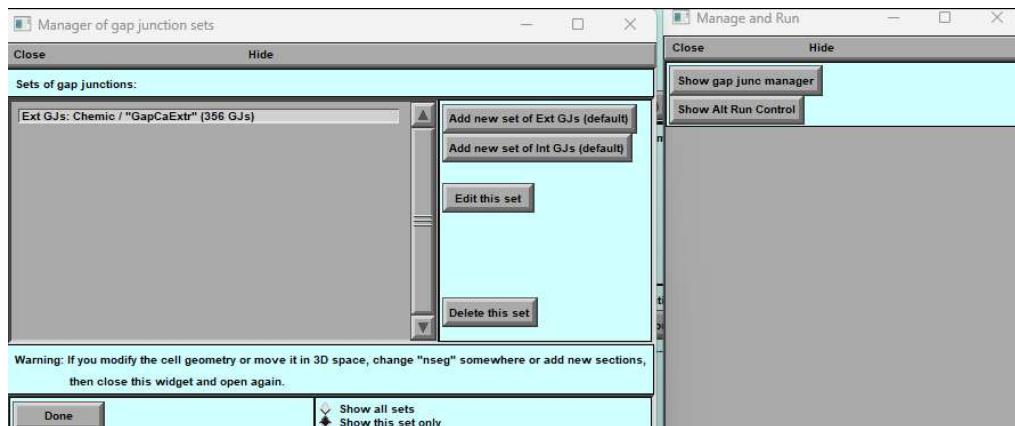
Each gap junction set can be **individually edited**, enabling highly customised astrocyte network simulations.

---

### *Interface Overview*

When you open the **Gap Junction** feature, you will see two panels:

- **Primary Panel** — manage gap junction sets:
  - Add new sets
  - Remove existing sets
  - Edit their parameters
- **Auxiliary Panel** — provides easy navigation between:
  - The **management panel** (for editing and organising sets)
  - The **calculation panel** (optimised for accurate stochastic process modelling)



**Figure 21. Left Panel - Manager of Gap Junction Sets:**

A list of existing Gap Junction sets, each potentially with a unique identifier, like "Ext GJs: Chemic / "GapCaExt" indicating a set of 356 Gap Junctions.

Options to add a new set of external or internal Gap Junctions (GJs) with default settings.

Functionality to edit or delete the selected set of Gap Junctions.

A reminder that if the cell geometry is modified, "nseg" must be changed or new sections added, followed by reopening the widget for updates to take effect.

Buttons are to show either all sets or only the selected set.

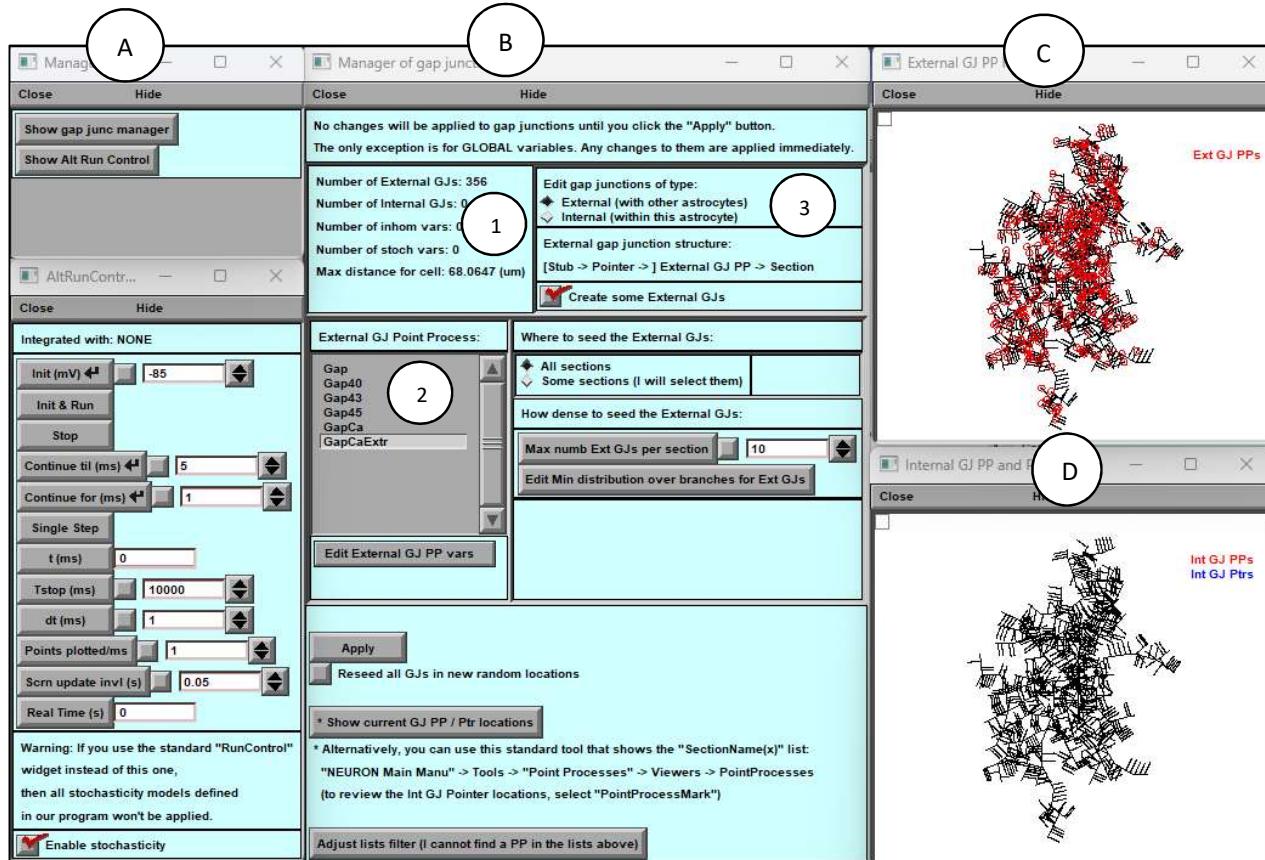
A "Done" button to exit the manager.

**Right Panel - Manage and Run:** Options to show the gap junction manager or the run control panel for simulations.

## Parameters of gap junctions in Neurons and Astrocytes.

This user manual section will provide detailed instructions on creating different types of Gap Junctions (GJ) within cells using BrainCell software. These GJ connections can be established in both astrocytes and neurons. It's important to note that there is a critical distinction between these two cell types: in astrocytes, you can create internal GJ connections within the same cell and external GJ connections directed to other cells. In contrast, in neurons, you can only create external GJ connections.

Let's begin by exploring the main window, which is depicted below.



**Figure 22. Main Window for Gap Junction (GJ) Creation and Editing**

- **A) Support Windows**

Two small support windows keep access to both the **main GJ editor** and the **calculation management window**, even if either is accidentally closed.

- **B) Main Window**

1. Displays **information** about the current number and types of GJs, along with quick reminder instructions.
2. Lets you **select GJ types** defined in your .mod files.
  - The displayed list is filtered by default; use “**Adjust List Filter**” to expand it if your files are not visible.
  - The “**Edit external GJ**” button allows modification of parameters for a selected GJ type.

3. Provides panels to:
  - Choose whether a GJ is **internal** or **external**,
  - **Distribute GJs** across different cell regions,
  - Define **maximum number** and **density** (either uniform or spatially varying).

- **C) Auxiliary Panels**

Visualise the **distribution and arrangement** of different GJ types throughout the cell, helping you verify spatial placement.

### **Adding and Modifying Gap Junctions in BrainCell**

BrainCell software allows you to add and adjust Gap Junctions (GJs) to customize your cell models. Here's a step-by-step guide:

#### **Initial Setup:**

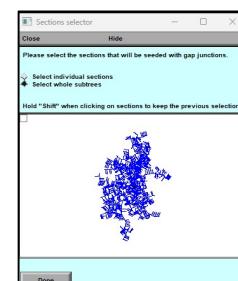
1. A basic set of external GJs is automatically inserted when you start the program. These GJs transfer calcium linearly from external cells to a selected astrocyte. A dedicated MOD file is used for this purpose.
2. The MOD file is designed so that when potassium is in equilibrium inside the cell under normal conditions, there is no calcium flow between this astrocyte and the external cells. The distribution of these GJs is random and follows a typical distribution pattern with specified density and a maximum number, as visible in panel 3.

#### **What You Can Do:**

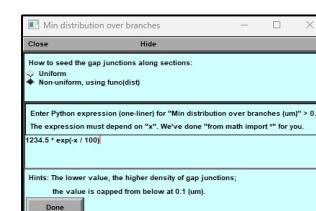
1. **Replace MOD File:** You can replace the original MOD file with your own while keeping the original GJ distribution and density intact.

2. **Change GJ Design and Spatial Distribution:**

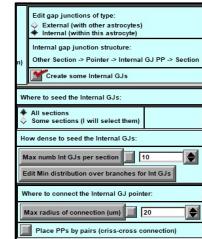
- Navigate to panel 3. Here, you can utilize the distribution function (e.g., for External GJs) and choose between "All section" or "Some section."
- The "Some section" option opens graphical windows where you can select the locations for GJ distribution by clicking on the cell sections. Hold the Alt-key to select multiple sections.



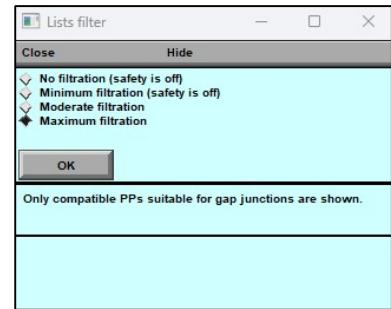
- Adjust the type of density and distribution for GJs. The maximum number is initially set to 10 per section, but you can change it to 0 or Infinity. If you set it to 0, no GJs of this type will be in your cell.
- You can modify the density of GJs (initially set at 8.4376) within a range of 0.1 to Infinity or choose non-uniform density using the "Edit Min distribution of the branches" function.
- For non-uniform spatial distribution of GJs, use the "Non-Uniform" function, which opens a new panel. Here, you can define a formula based on the x distance to distribute the GJs. Click "Done" to see the new distribution.



3. **Create Internal GJs:** Panel 3 functions differently when creating internal GJs. You can specify the minimum distance between cell parts where internal GJs may connect.



4. **Filter MOD Files:** In the MOD-file directory, you'll find various mod files that can be interpreted as GJs. However, not all of them can be used as gap junctions. To assist you, we've introduced a filter that helps you select the appropriate mod file. The default filtration is well-suited for GJs, but you can adjust it if you wish to use a particular mod file.



5. After making your desired adjustments, click the "Apply" button, and your cell will have a set of Gap Junctions according to your specifications.

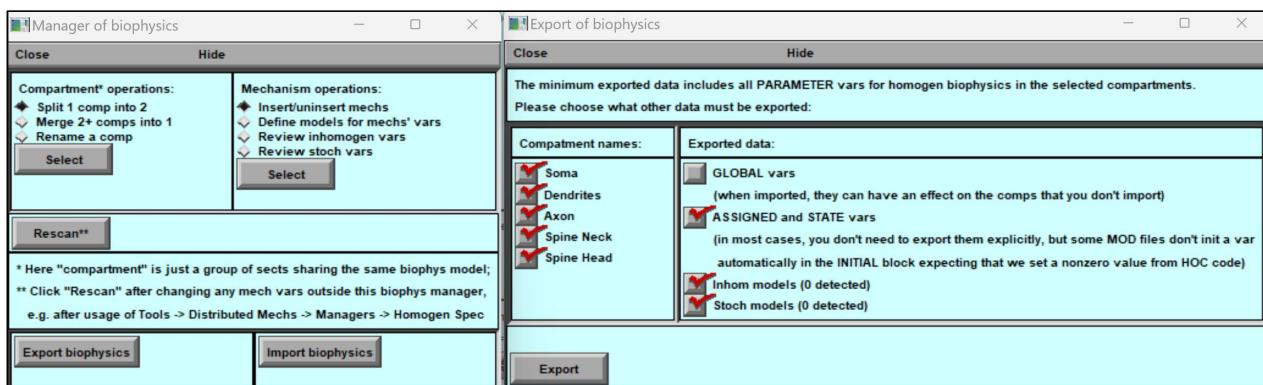
Customizing Gap Junctions in BrainCell allows you to create detailed and biologically relevant cell models tailored to your research needs.

## Exporting and Importing Biophysical Mechanisms

BrainCell includes a **flexible export/import feature** that lets you save and reuse **biophysical mechanism sets** across different cell models.

- You can **export mechanisms** from a current cell model into a file for backup or sharing.
- You can then **import these saved mechanisms** into another cell — for example, to test how the same ion channel configuration behaves on different neuron shapes or to analyse how morphology affects output activity.

This functionality is available in the **Biophysics Manager** panel, making it easy to organise and reuse your customised mechanism setups for future studies.



### Export Functionality

When exporting, the user can selectively choose and export mechanisms associated with specific cells. This information is then stored in a designated text file. The file's structure is outlined below, providing clear insights into the placement of each mechanism and its respective parameter values.

## *Export Text File Structure*

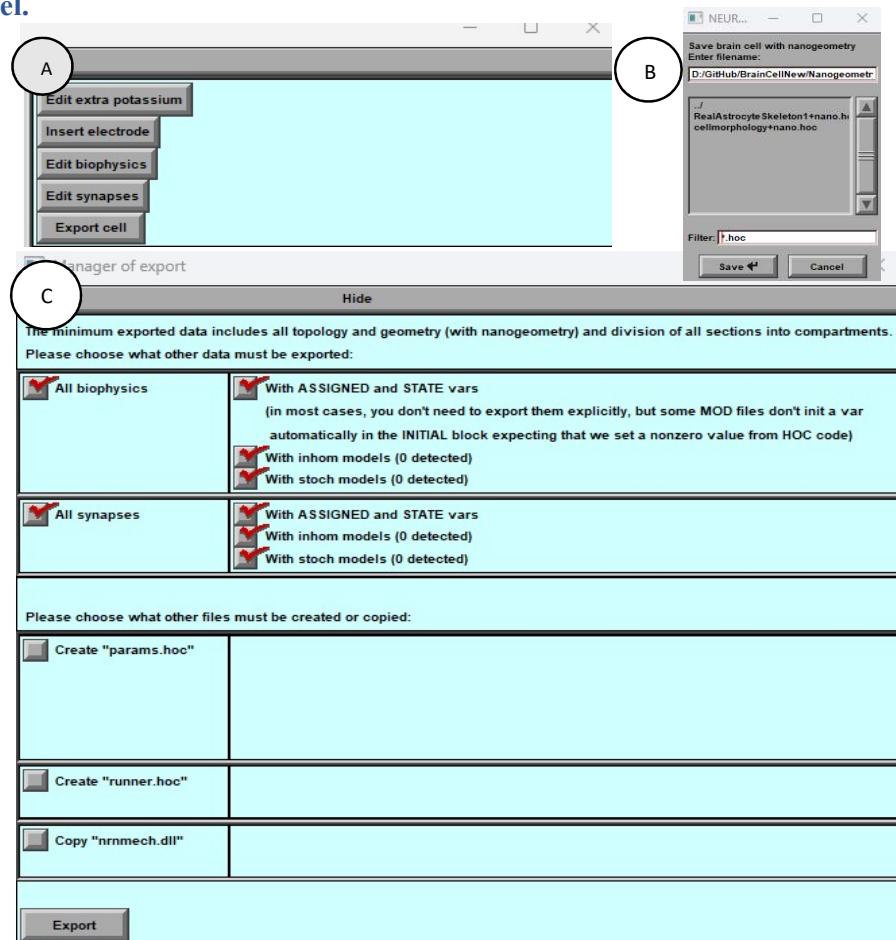
```
        "pas": {
            "PARAMETER": {
                "g_pas": 8.9686098654
                "e_pas": -85.0
            },
            "ASSIGNED": {
                "i_pas": 0.0018251121
            }
        },
        "Dendrites": {
            "capacitance": {},
            "PARAMETER": {
                "cm": 0.8
            }
        },
        "pas": {
            "PARAMETER": {
                "g_pas": 8.9686098654
                "e_pas": -85.0
            },
            "ASSIGNED": {
                "i_pas": 0.0018251121
            }
        },
        "Axon": {
            "capacitance": {},
            "pas": {
                "PARAMETER": {
                    "g_pas": 7.1428571428
                    "e_pas": -65.0
                },
                "ASSIGNED": {
                    "i_pas": 2.4999999999
                }
            }
        },
        "Spine Neck": {
            "capacitance": {},
            "pas": {
                "PARAMETER": {
                    "g_pas": 7.1428571428
                    "e_pas": -65.0
                },
                "ASSIGNED": {
                    "i_pas": 2.4999999999
                }
            }
        }
    }
}
```

## **Import Functionality**

The imported text file presents a straightforward structure that allows users to comprehend and modify the mechanisms' parameter values easily. This flexibility enables users to customise and adapt the parameters as needed. Moreover, the software facilitates the import of already modified text files, streamlining the database creation process for sets of mechanisms about a specific cell type.

With this robust feature, users can effortlessly establish a comprehensive database of mechanism sets associated with cell types, enhancing the software's utility and facilitating in-depth analyses.

## Export cell model.



**Figure 23. Control Panel for Exporting Finished Cell Model.**

**The Control Panel for Exporting Finished Cell Models** is a crucial component of the **BrainCell**, providing users with comprehensive options for exporting their completed cell models. This panel offers convenient access to various export functionalities, facilitating the seamless transition of cell models to external platforms or further analysis.

**A) Key to Call the Model Export Panel:** The Control Panel features a designated key that users can activate to open the Model Export Panel. This key is a quick and intuitive way to access the export functionalities.

**B) Pop-up Export Panel:** Upon activation, the Model Export Panel appears as a pop-up window, displaying a range of export options and settings. This intuitive interface allows Users to navigate and configure their desired export parameters conveniently.

## Model Export Panel

The **Model Export Panel** allows you to save your completed cell model with all its essential features for use in other simulations, studies, or external tools.

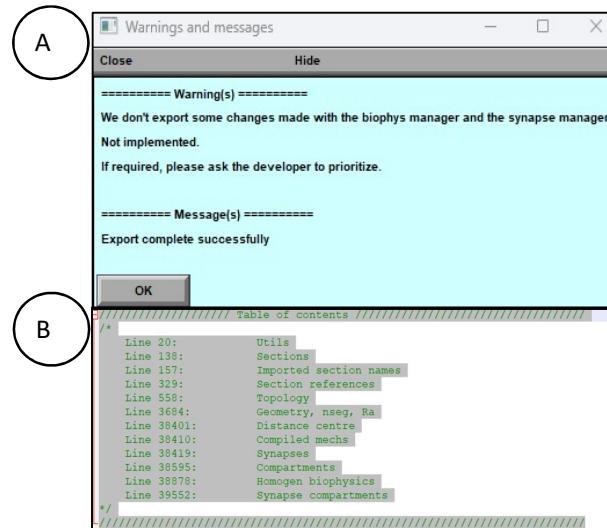
You can export the following components:

- **Cell Shape**  
Save the full 3D morphology of the cell, preserving its geometry and structural details.
- **Biophysical Mechanisms**  
Export all mechanisms configured for the model — including **spatial heterogeneity** and **stochastic properties** — to reproduce the cell's functional behaviour in other studies.
- **Synapses**  
Save the synaptic network, keeping all parameters and spatial/stochastic features.
- **HOC File Export**  
Export the entire model as a **HOC file** for compatibility with **NEURON** and other external software platforms.
- **Cluster Calculation & Parameter Selector (advanced)**  
Provides settings for running **cluster-based simulations** and includes a **parameter selector** to automatically adjust parameters during large-scale or distributed computing.
- **File Export for Biophysical Mechanisms**  
Generate a dedicated file describing all biophysical mechanisms in the model — useful for sharing, reviewing, or running additional analyses.

---

With the **Control Panel for Exporting Finished Cell Models**, researchers can save their work in multiple formats while retaining key features such as **geometry, mechanisms, and synapses**. This makes it easy to **share models, integrate them into other workflows, and run advanced analyses on different platforms or computing clusters**.

### Structure of export file.



**Figure 24. Form and Structure of the Export File.** The panel displays warm messages (A) details about the format and arrangement of the export file (B).

Whenever the user exports a cell model, a notification window (Fig.20) will warn them of any errors or inaccuracies that may have occurred during the exporting process. It is important to note that these errors are usually minor and do not impact the further utilisation of the model. However, they serve as alerts for possible limitations within the model.

A successfully saved file in "hoc" - format will display inside as a commentary (Fig.20) a record of the mechanisms and their respective lines saved to the file. This valuable information facilitates efficient use of the model's features in subsequent calculations or when modifying the model.

## BrainCell Software: Model Export. Result of Simulation Parameter Selection

### *Model Export Feature*

The **Model Export** feature in BrainCell helps you save **specific parameters** and **long calculations** for reuse.

It is especially useful when you want to perform simulations without extra graphics or features, or when you need to run **parameter sweeps** efficiently.

---

#### *1. Purpose*

- Save selected model parameters to avoid reconfiguring them manually.
  - Reuse heavy pre-computations without regenerating graphics or unrelated features.
  - Specify which parameters can change during new calculation runs.
- 

#### *2. Creating a Parameter File*

1. Click “Create File” (Fig. 24A).
  2. A new window opens (Fig. 24B) where you can select which parameters to include.
  3. If you choose “**Export Other Parameters**,” an additional window (Fig. 24C) appears for advanced parameter selection.
- 

#### *3. Export Parameters Window (Fig. 24C)*

The window has:

- A **central panel** showing the selected parameters.
  - A **left toolbar** with five buttons for managing them:
    - **Add Biophys var.** — choose a biophysical parameter from a pop-up list.
    - **Add synapse var.** — select a synapse-related parameter.
    - **Add custom var.** — include custom parameters not visible on the main NEURON panel.
    - **Correct var.** — edit an existing parameter.
    - **Delete var.** — remove a parameter from the list.
-

#### 4. Saving Parameters

- After selecting parameters, click **OK** to create a **parameter file**.
- You can later load this file to run calculations while changing only the saved parameters — speeding up setup and reducing complexity.

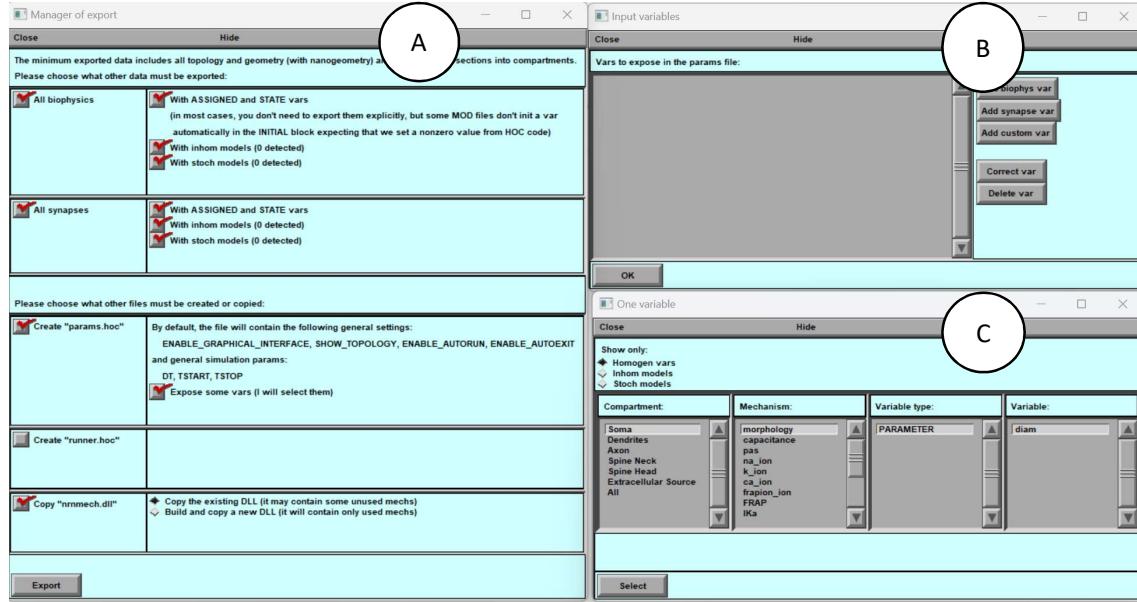
---

#### 5. Running Calculations

By using the **Model Export** feature, you can:

- Save your selected parameters,
- Modify them easily for new experiments,
- Perform calculations without repeating full model setup.

💡 **Tip:** Refer to **Fig. 24A–C** for visual guidance while selecting and saving parameters.



**Figure 25. Exporting Control Parameters using Pop-up Windows**

BrainCell allows users to define and export control parameters using pop-up windows. A) The first window saves specific parameters. B) the second window provides a selection interface for customising the output file. C) the third window gives an example of selecting soma diameter as a parameter. Users can easily create and export parameters for tailored simulations and analyses with these pop-up windows.

#### Exporting Computation Run Parameters

BrainCell allows you to export the results of your simulations by creating a **Result file** that stores selected variables over time.

This feature is ideal for analysing outputs or running automated parameter sweeps.

## *1. Creating the “Runner.hoc” File (Fig. 26A)*

- Click “**Runner.hoc**” to set up result recording.
  - A new file will be created and automatically named based on the **time the calculation finishes**.
  - This prepares the system to capture and save selected variables during simulation.
- 

## *2. Recording & Saving Variables*

- Click “**Record and save some vars**”.
  - A **graph window** will open (Fig. 26B).
  - Use the mouse to select points or traces in the graph and apply NEURON’s standard “**Print what?**” function (Fig. 26B–C).
  - The chosen variables will be saved as a **function of time** in the result file, ready for further analysis.
- 

## *3. Parameter Sweep Calculations*

- Click “**Make a sweep for some vars**” (Fig. 26D) to run calculations across a **parameter grid**.
  - Unlike simple runs (which vary one parameter at a time), this feature lets you define a **set of input parameters** for automated multi-run simulations.
- 

## *4. Configuring the Parameter Grid (Fig. 26E)*

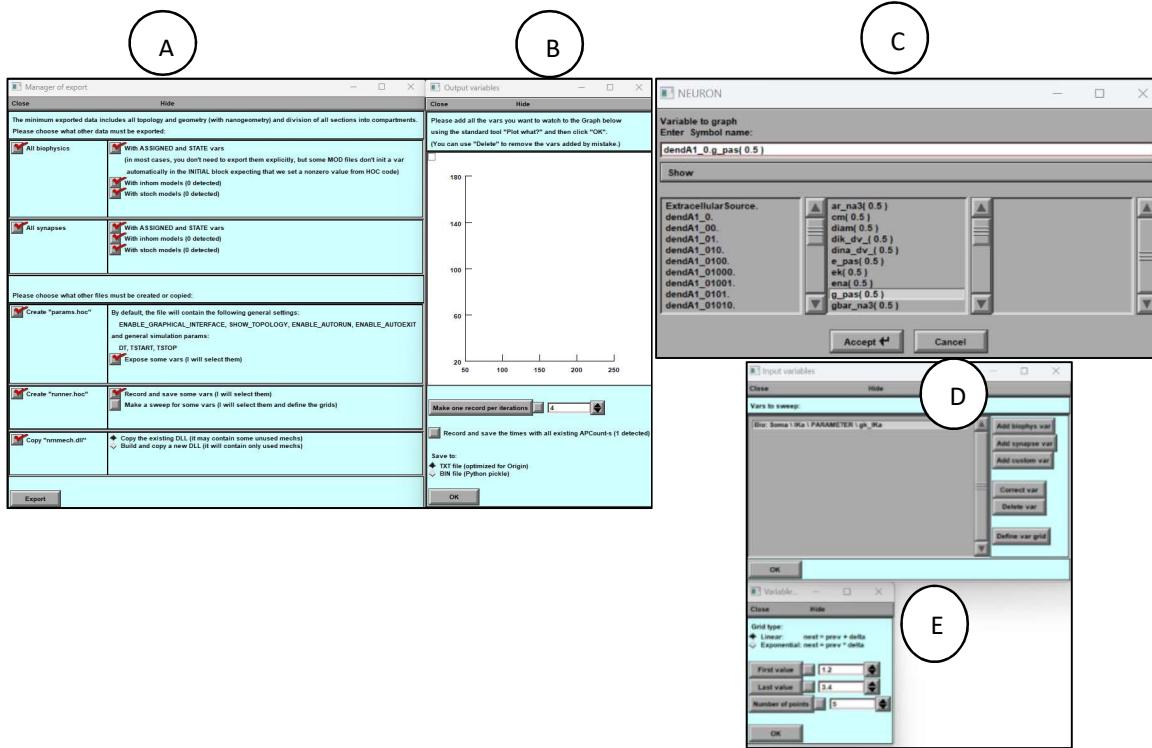
- A pop-up window lets you define grid settings:
    - **Initial and final values** of the parameter to sweep.
    - **Step size** between values.
    - Grid type: **linear** or **non-linear** progression.
  - Confirm to generate the sweep and start automated calculations.
- 

## *Summary*

Using these tools, you can:

- **Record selected variables** during a run and save them over time.
- **Export results** to a named file for later analysis.
- **Automate simulations** by running parameter sweeps with user-defined grids.

This workflow makes it easy to **systematically explore model behaviour** and capture exactly the data you need for further study.



**Figure 26. Exporting of control of computation Pop-up Windows.** BrainCell allows users to define and export the control of computation pop-up windows. A. The first window saves specific parameters. B) the second window provides a plot for customising the output file. C) the third window gives an example of selecting somatic gpas as a variable in the result file. D) window gives an example of selecting soma gk as a parameter. F) the window for the grid definition.

### Managing Computation Results

BrainCell provides an organised way to **export and manage cell models** along with their simulation results.

This structure makes it easy to **store, retrieve, and review** all files related to a particular model and its calculations.

#### Creating a Dedicated Folder for Each Cell Model

- When exporting a cell model, BrainCell lets you create a **separate folder** for that model.
- It's recommended to create a **new folder for each exported cell** to keep your projects organised and make results easy to locate later.

#### Folder Structure and Contents

Each exported cell model folder contains:

- **Geometry files** – full 3D shape of the cell.
- **Biophysical mechanisms** – all ion channels, synapses, and custom parameters.
- **Control files** – scripts and settings used to run simulations.

- **Computation results** – data files generated during simulations.

This structure keeps **all model details and results linked** in one place, so you can trace how outputs were generated and under what conditions.

---

#### *Loading and Inspecting Saved Cell Models*

- Exported cell models can be **imported back** into BrainCell at any time.
  - Once loaded, you can **view the full configuration** of the cell — including geometry, biophysics, and stored simulation data.
  - This makes it easy to **review, analyse, and continue working** with previously created models.
- 

#### *Summary*

By creating a **dedicated folder for each exported model**, you ensure:

- Clean organisation of your simulation projects,
- Quick access to all files and results,
- The ability to **revisit and fully inspect** any cell configuration in BrainCell.

This workflow supports **efficient project management** and thorough analysis of model behaviour and outcomes.

d:\GitHub\BrainCellNew\Nanogeometry\Neuron\*.*			
Name	Ext	Size	Date
[..]	<DIR>		19/06/2023 12:25
[Cell1]	<DIR>		20/06/2023 10:06
[Cell2]	<DIR>		20/06/2023 10:57
cellmorphology+nano	hoc	2,441,843	19/06/2023 12:28
drag&drop_init	bat	42	18/06/2023 15:50
nrmec	dll	597,146	18/06/2023 15:50

d:\GitHub\BrainCellNew\Nanogeometry\Neuron\Cell1\*.*			
Name	Ext	Size	Date
[..]	<DIR>		20/06/2023 10:50
[results]	<DIR>		20/06/2023 10:06
nrmec	dll	597,146	20/06/2023 09:57
runner	hoc	12,266	20/06/2023 09:57
params	hoc	255	20/06/2023 09:57
Cell1	hoc	14,082,988	20/06/2023 09:57

d:\GitHub\BrainCellNew\Nanogeometry\Neuron\Cell1\results\*.*			
Name	Ext	Size	Date
[..]	<DIR>		20/06/2023 10:06
aps 2023-06-20 10.06.15	txt	111	20/06/2023 10:06
vars 2023-06-20 10.06.15	txt	19,135	20/06/2023 10:06
aps 2023-06-20 10.04.31	txt	111	20/06/2023 10:04
vars 2023-06-20 10.04.31	txt	19,135	20/06/2023 10:04
aps 2023-06-20 10.02.47	txt	111	20/06/2023 10:02
vars 2023-06-20 10.02.47	txt	19,135	20/06/2023 10:02
aps 2023-06-20 10.01.02	txt	111	20/06/2023 10:01
vars 2023-06-20 10.01.02	txt	19,135	20/06/2023 10:01
aps 2023-06-20 09.59.18	txt	111	20/06/2023 09:59
vars 2023-06-20 09.59.18	txt	19,135	20/06/2023 09:59

## Figure 27. Organisational Structure of Imported Cell Models and Simulation Results

*Overview:* This figure demonstrates the directory structure export of BrainCell software, showcasing imported cell models and computer simulation results.

### A) Initial Directory View:

Two directories, "Cell1" and "Cell2," house cell models and related files in the "Neuron/NanoGeometry" subdirectory.

### B) Structure of Cell1 Directory:

The directory structure for the "Cell1" folder is displayed in this window. This structure remains consistent for all saved models.

- "exported cell1" - the exported cell model file.
- "nrmec.dll" - a mechanism file containing relevant computational mechanisms.
- "Runner.hoc" - a calculation management file responsible for simulations.
- "Params.hoc" - a file housing parameters modifiable during computation.

### C) Results Directory:

Within this directory, various result files showcase computation outcomes for different cell models. In this instance, ten result files correspond to five calculation parameters. The first file illustrates action potential (AP) dynamics, while the second file (VARS) displays selected variable dynamics during the cell export process.

### *Re-Importing Cell Models*

BrainCell allows you to **reimport previously saved models** for further editing and simulations.

- To reload a saved model, open BrainCell and select the “**Nano**” option (Fig. 2).
- Choose the desired model file to load.
- Once imported, you can **access and modify all parameters** — including geometry, biophysical mechanisms, and simulation settings — to run new calculations or experiments.

**⚠ Tip:** Always export the updated model to a **new folder** to avoid overwriting previous results.

---

### *Importing External Cell Models from NEURON*

BrainCell includes an **experimental feature** that allows importing models written in the **NEURON** simulation language.

This lets you **analyse, modify, and expand** pre-existing neuron models within the BrainCell environment.

 **Note:** This feature is in **active development** and may change in future versions.

---

#### *Steps to Import a NEURON Model*

1. **Compile MOD Files**
  - Combine all required **MOD files** (from both BrainCell and the external model) into a single directory.
  - Use the `mknrndll` utility to compile them into a **Dynamic Link Library (DLL)**.
  - Place the generated DLL in the same directory as the model's entry point (usually `init.hoc`).
2. **Import the Model**
  - In BrainCell, select “**External Simulation**” and choose the main source file (e.g., `init.hoc`).
  - The simulation may start automatically; you can **pause it** to finish the import setup.
3. **Define Cell Components**
  - BrainCell will ask you to identify which parts of the imported model correspond to the **soma, dendrites, and axon**.
4. **Modify After Import**
  - You can **rename or redefine** sections of the imported model as needed to match your experiment.
5. **Review and Export**
  - Check all imported **biophysical mechanisms**, rename if necessary, and export the updated model for future use.

## Simulation modes: Examples of cell simulation inside of BrainCell.

Various computational scenarios are available in BrainCell, all conveniently located on the right side of the main program panel.

1. Voltage
  - [Membrane voltage distribution](#)
  - [CA1-neuron voltage \(Neuron mode only\)](#)
2. Electric stimuli
  - [Constant electric stimuli](#)
  - [Variable electric stimuli](#)
3. FRAP
  - FRAP with round spot bleaching
  - [Line-scan FRAP](#)
4. Calcium
  - [Microscopic calcium dynamics \(Astrocyte mode only\)](#)
  - [Calcium waves](#)
5. Other Species
  - [Membrane biophysics with glutamate transport](#)
  - [Dynamic of intra and extracellular potassium](#)
  - [Basket cell GABA diffusion \(Neuron mode only\)](#)

While sharing some foundational similarities, these scenarios exhibit unique features tailored to astrocytes and neurons. For astrocytes, there is a specific scenario designed to simulate the dynamics of spontaneous calcium responses. For neurons, a dedicated scenario focuses on modeling the dynamics of glutamate release from an array of basket cell terminals.

The following sections provide a detailed breakdown of each scenario, enabling straightforward replication in computational studies. These scenarios are not merely instructional; they serve as versatile source packages capable of simulating a variety of physiological tasks specifically designed to address diverse research objectives.

## **CA1-Neuron Voltage Simulation (Neuron mode only)**

CA1 pyramidal cells (PCs) are a major output of the hippocampus. They integrate signals from the **entorhinal cortex** both directly (via the **temporoammonic pathway**) and indirectly (through the **dentate gyrus → CA3 → CA1 loop**).

The **dendrites** of CA1 PCs contain many types of **voltage-gated ion channels**, distributed in highly non-uniform ways:

- **Sodium channels ( $\text{Na}^+$ ):** Enable **back-propagation** of action potentials (APs) from the axon into dendrites.
  - **Calcium channels ( $\text{Ca}^{2+}$ ):** Control **local calcium influx** following back-propagating APs and synaptic inputs.
  - **Potassium channels ( $\text{K}^+$ ):** Help regulate dendritic excitability.
- 

### **Action Potential Dynamics Calculation**

This feature lets you **simulate action potential dynamics** based on the **loaded cell shape** and **synapses** defined on its spines.

#### **Key considerations:**

- **Cell shape:** The simulation uses the currently loaded geometry (which may be idealised, not a perfect CA1 reconstruction).
- **Synapse definition:** Synapses added to spines are included in AP dynamics.
- **Fixed parameters:** Ion channel kinetics follow **classical distributions** (see *Migliore et al., 1999*).
  - For more details, refer to the linked model documentation.

This setup allows detailed exploration of how **cell morphology and synaptic inputs** shape action potential behaviour.

---

### **Adapting Cell Sections**

If your imported model uses **non-standard section names** (e.g., different terms for *soma*, *dendrites*, *axon*), BrainCell will:

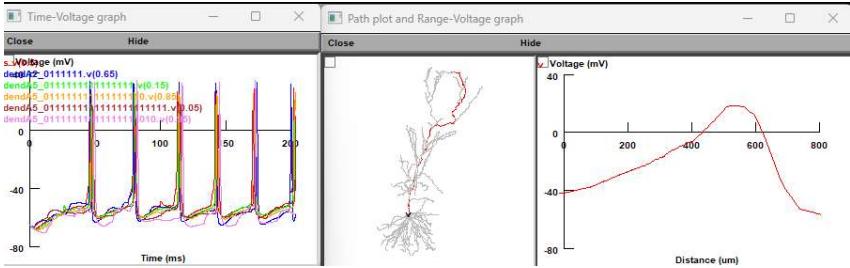
- Display a **warning** if section names don't match expected CA1 features.
- Automatically assign the required mechanisms if you proceed.
- Offer suggestions for **renaming or splitting sections** to match simulation requirements.

This ensures **compatibility and accurate channel placement**, while giving you flexibility to adapt imported morphologies.

---

### **Visualising Electrical Dynamics**

BrainCell provides **three interactive graphs** to explore the cell's voltage behaviour:

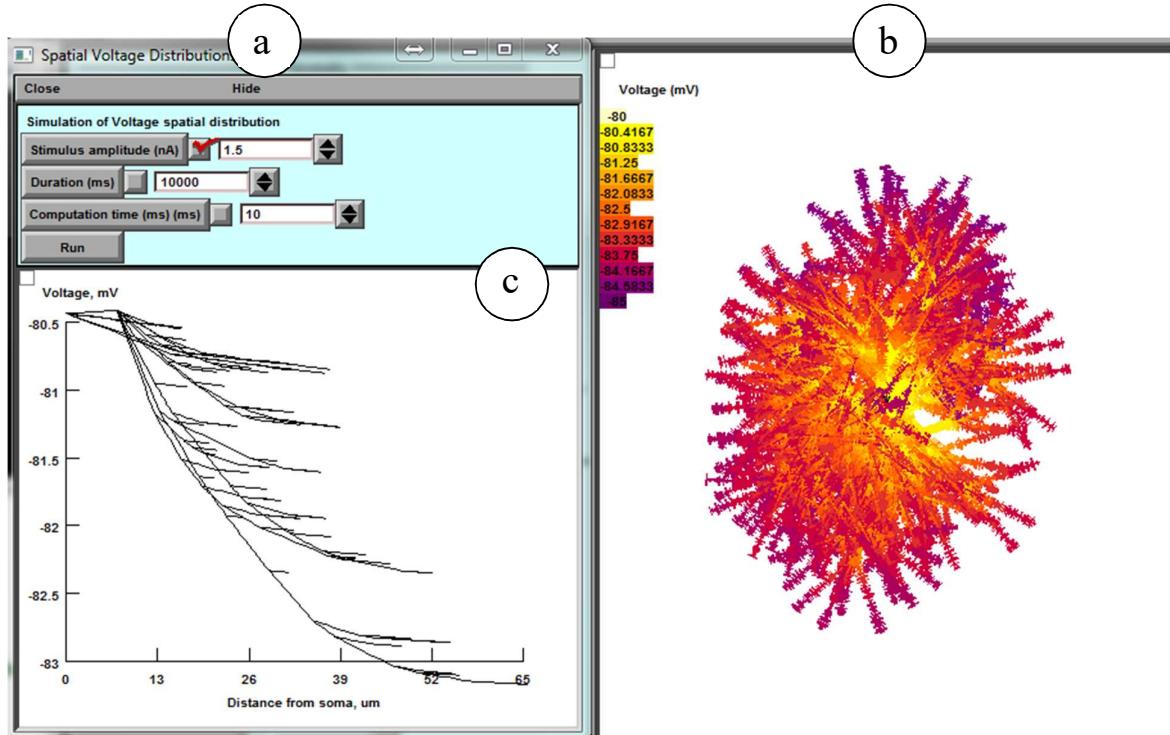


1. **Voltage along a spatial path** — shows potential changes across cell sections (the path is highlighted in red in the figure).
2. Other graphs display **local and global membrane potential dynamics**, helping you analyse activity across the neuron.

These tools make it easy to **observe, compare, and interpret** how action potentials propagate and how dendritic regions behave during simulations.

## Membrane voltage distribution

In the main control panel, pressing '**Compute the spatial voltage distribution**' opens window panels which report membrane voltage across selected cell processes upon a step current injection at the soma (a common electrophysiological test scenario). The default resting potential is -85 mV.



**Figure 28. Window panel to monitor membrane space voltage distribution.** **A**, Settings panel with space profile of voltage after inserting 1.5 nA into the soma. **b**, Dynamic voltage landscape mapped on the cell morphology. **c**, The corresponding voltage profile plot along individual processes ('dendrites' in NEURON).

The initiation panel (Fig. 28a) provides the following settings:

'**Stimulus amplitude (nA)**', the amplitude of constant depolarising current injected into the soma. '**Duration (ms)**', duration of constant depolarising current.

'**Computation time (ms)**', simulation run time (time of display).

'**Run**' to start the simulation.

## Exploring Constant Electrical Stimulation

The "Constant Electrical Stimulation" tool lets to explore the dynamic interplay between current and voltage, unlocking the secrets of electrical signalling in digital neurons.

### Directly Stimulate the Soma:

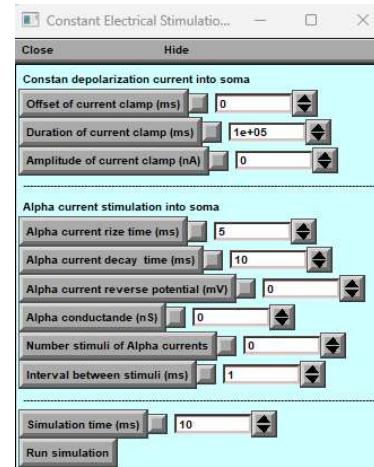
Bypass synaptic complexities and focus on the core of the neuron – the soma. Inject a steady current stream directly into the soma, observing its immediate and sustained impact on the membrane potential—three parameters: Offset, Duration and Amplitude.

### Simulation of Alpha Synapses:

Activate alpha synapses on the soma and witness the interplay between direct and indirect stimulation. See how the neuron integrates these inputs, revealing the intricate dance of voltage fluctuations - 6 parameters: rise and decay time, reverse conductance number of stimuli and interval between stimuli. Potential.

The dedicated panel makes it easy to access current-clamp mode, allowing for quick configuration of stimulation parameters and experimentation.

This demo serves as a starting point for exploring complex neurophysiological phenomena. Adjust parameters, try out different configurations, and better understand how neurons react to electrical stimuli.



## Exploring Variable Electrical Stimulation

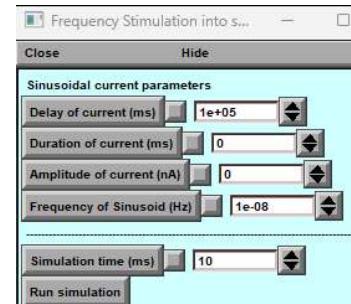
The "Variable Electrical Stimulation" tool lets to explore the sinus curve injected into the soma. Discover membrane potential dynamics with this feature! Experiment with alternating current and learn about voltage fluctuations within neurons.

### Explore demo mode:

- Dedicated Panel: Input the periodic signal frequency and observe the dynamic response in current clamp mode.
- Sinusoidal Playground: Experiment with frequency range and analyse the relationship between stimulus and response.
- Current Clamp Magic: Inject current into soma and mimic real-world effects of electrical stimulation on a neuron.

"Variable Electrical Stimulation" is a tool that can help:

- Visualize action potential generation.
- Investigate membrane properties.
- Design custom electrical stimulation protocols.



**Remember:** As you explore, remember that the specific responses you observe will depend on the chosen cell model and its unique biophysical properties. Don't hesitate to experiment with different models and parameters to unlock the full potential of this versatile tool.

## Linescan FRAP experiment: probing intracellular connectivity of Brain Cell

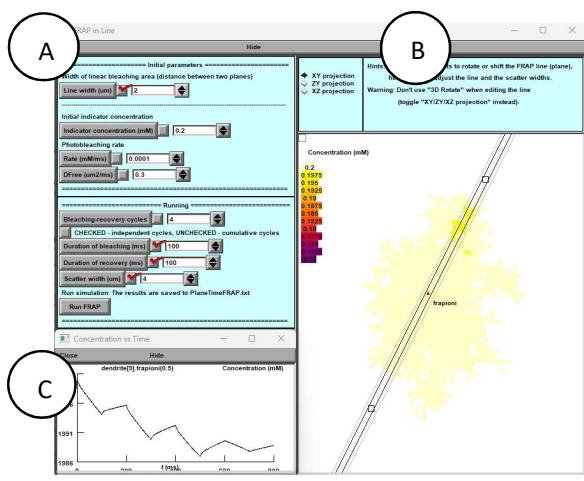
In the "Fluorescence Recovery After Photobleaching (FRAP) for Intracellular Indicators" section of the BrainCell software manual, we show how FRAP can evaluate effective diffusivity in cellular compartments near the bleached area.

FRAP is particularly useful in verifying if the modelled cell morphology accurately replicates empirically documented intracellular diffusivity and connectivity. The BrainCell software enables users to conduct FRAP experiments and corresponding simulations directly from the main window, providing a practical tool to test astrocyte morphology against empirical data.

The software's default settings are based on data from CA1 astrocyte studies using linear photobleaching tests with Alexa Fluor 594 or Alexa Fluor 488.<sup>9</sup>

The FRAP test in BrainCell simulates a photobleaching area represented by a linear segment on the XY plane. Select 'Line-scan FRAP' to access this function, which opens a dedicated control window (refer to Fig. 29 in the manual).

The initiation panel of 'Line-scan FRAP' (Fig. 29a) includes several adjustable settings:



**Figure 29. Panels for controlling and monitoring simulated line-scan FRAP**

- **A — Parameter Settings Panel:** Controls all key FRAP simulation parameters.
- **B — Concentration Map:** Colour-coded display of non-bleached molecule concentrations mapped onto the cell morphology.
- **C — Time Profile:** Example plot of concentration recovery over time.

## FRAP Simulation Overview

- **Default trial duration:** 8000 ms
- **Photobleaching frequency:** every 1000 ms
- In addition to the colour-coded map (panel B), FRAP results can be plotted and saved using standard NEURON tools:
  1. Select the cell branch/region of interest (e.g., *dendrite*) with **PointProcessGroupManager**.
  2. Configure and save plots through the **Graph** menu (panel C).
- After each simulation run, BrainCell automatically saves results to a data file.

## RAP Panel Controls

- **Width of bleaching area:** Distance between two planes ( $\mu\text{m}$ ).
- **Line width ( $\mu\text{m}$ ):** Thickness of the bleaching line.
- **Initial indicator concentration (mM):** Starting concentration of the fluorescent indicator.

- **Photobleaching rate (ms):** Laser-induced bleaching speed.
  - **Diffusion rate ( $\mu\text{m}^2/\text{ms}$ ):** Indicator diffusion rate in the cell.
- 

### Bleaching & Recovery Cycles

- **Independent cycles:** *Checked* — each cycle runs separately.
  - **Cumulative cycles:** *Unchecked* — cycles build on previous ones.
  - **Bleaching duration (ms):** Length of each laser pulse.
  - **Recovery duration (ms):** Time allowed for fluorescence recovery.
- 

### Running FRAP

- **Run simulation:** Starts FRAP and saves results to *Plane1/mFRAP.txt*.
- 

### Visualisation & Interaction Tips

- **Rotate/shift FRAP line:** Drag the **3 anchor points**.
  - **Adjust line/scatter widths:** Hold Shift.
  - **Projection toggle:** Use XY/YZ views instead of 3D Rotate for editing.
  - **Projection options:**
    - XY — 2D map of FRAP line & indicator distribution.
    - YZ — 2D projection in YZ plane.
    - XZ — 2D projection in XZ plane.
  - **Concentration vs Time:** Graphs indicator recovery over time.
  - Avoid **3D Rotate** when editing the FRAP line.
- 

### Notes

- FRAP simulation models **fluorescent indicator diffusion** within cells.
- The FRAP line defines the region to be bleached.
- Recovery dynamics can reveal the **diffusion rate** of the indicator.

## **FRAP with round spot bleaching**

This section introduces a **beginner-friendly computational scenario** for studying cell geometry using **FRAP** (*Fluorescence Recovery After Photobleaching*).

FRAP involves briefly using a laser to “**bleach**” (dim) a small region of a cell, then watching how fluorescence **recovers** as unbleached molecules move back into the area.

BrainCell makes it easy to **simulate FRAP on virtual cells**.

You can:

- Select a **specific region** of the cell,
- Run a **single FRAP test** or create a **series of experiments**,
- Analyse how recovery speed changes with **distance from the cell centre**.

This is particularly useful for exploring **cell geometry** and **molecular diffusion**.

↳ **Default settings** are based on published CA1 astrocyte experiments using linear photobleaching with **Alexa Fluor 594 / 488**.

The photobleaching area is represented as a **spherical segment** within the cell.

To begin, select “**Round Spot FRAP**” to open the dedicated control window (Fig. 30).

---

### *Round Spot FRAP — Adjustable Settings (Fig. 30a)*

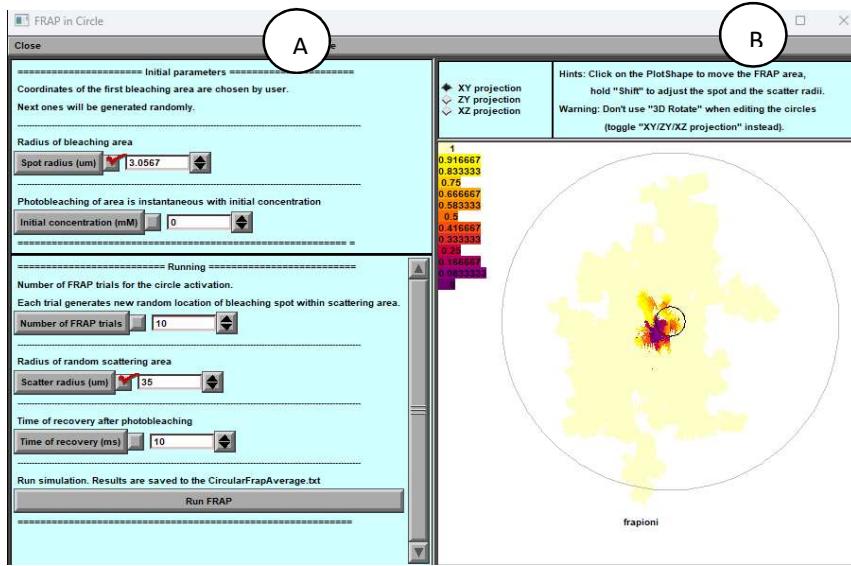
1. **Spot Radius**
  - Defines the size of the bleached area (sphere radius).
  - In two-photon systems, the laser spot is typically **1–1.5 μm**.
  - Adjust the radius using the mouse on the plot to match real experiments.
2. **Initial Concentration (mM)**
  - Sets the dye concentration right after bleaching.
  - Default is **0** (fully bleached area).
3. **Time of Recovery**
  - Duration of the post-bleach recovery period.
  - Controls how long the simulation runs to track fluorescence return.
4. **Number of Trials**
  - Number of FRAP tests to simulate at different positions within the scatter area.
  - Trial positions are randomly chosen.
5. **Scatter Radius**
  - Size of the spherical region where recovery trials will be run.
  - Lets you test how recovery changes at different distances from the cell centre.
6. **Diffusion Coefficient**
  - Default: **0.3 μm<sup>2</sup>/ms**.
  - Determines how fast dye molecules spread back into the bleached area — critical for modelling recovery dynamics.

---

With these options, BrainCell provides a **powerful yet easy-to-use platform** for exploring:

- **Fluorescence recovery dynamics**
- **Molecular diffusivity**
- **Cellular morphology effects**

This tool is valuable for both **researchers** and **students**, offering an intuitive way to simulate and analyse FRAP experiments without advanced programming knowledge.



**Figure 30. Window panels to control and monitor simulated spot FRAP.**

**A. Parameter Settings Panel:** This panel is where you can adjust various settings for the FRAP simulation. Here, you can set parameters like the spot radius, initial concentration, time of recovery, number of trials, scatter radius, and diffusion coefficient. These settings allow you to customize the FRAP simulation according to your specific research needs or experimental design.

**B. Concentration Dynamics Visualization:** This part of the interface shows a color-coded, dynamic map of the concentration of non-bleached molecules on the cell's morphology. It provides a visual representation of how the concentration changes over time during the recovery process post-photobleaching, offering an intuitive understanding of the molecular behavior within the cell.

# Ca<sup>2+</sup> Wave Simulations

Pressing the “Calcium Wave” button on the main control panel opens dedicated windows (Fig. 31) for configuring, running, and visualising intracellular Ca<sup>2+</sup> dynamics.

This module simulates calcium waves triggered by a local increase of IP<sub>3</sub>, which activates Ca<sup>2+</sup> channels.

BrainCell uses the standard NEURON-integrated reaction–diffusion formalism for calcium signalling. The model includes diffusion, buffering, pumps, channels, and leaks.

---

## Core Calcium Dynamics Equations

### 1. Reaction–Diffusion Equation

$$\frac{\partial [Ca^{2+}]}{\partial t} = D_{Ca} \nabla^2 [Ca^{2+}] - \frac{1}{FA_m} (I_{ch} + I_{leak} - I_{pump}) - \sum_i k_{on,i} [B_i] [Ca^{2+}] + \sum_i k_{off,i} [CaB_i]$$

where

- $D_{Ca}$ — diffusion coefficient ( $\approx 220 \text{ } \mu\text{m}^2 \text{ s}^{-1} \approx 0.22 \text{ } \mu\text{m}^2 \text{ ms}^{-1}$ )
  - $F$ — Faraday constant
  - $A_m$ — membrane area per unit cytosolic volume
  - $I_{ch}, I_{leak}, I_{pump}$ — fluxes through Ca<sup>2+</sup> channels, leaks, and pumps
  - $[B_i], [CaB_i]$ — free and bound buffer concentrations
  - $k_{on,i}, k_{off,i}$ — binding and unbinding rate constants
- 

### 2. Buffering Kinetics

For each buffer species  $i$  (mobile or stationary):

$$\frac{d[CaB_i]}{dt} = k_{on,i} [Ca^{2+}] [B_i] - k_{off,i} [CaB_i]$$
$$[B_i]_{total} = [B_i] + [CaB_i]$$

Typical parameters:

- Endogenous buffer (slow, immobile):  $k_{on} = 100 \text{ } M^{-1}ms^{-1}$ ,  $k_{off} = 0.1 \text{ } ms^{-1}$
  - Mobile buffer (Fluo-4):  $k_{on} = 600 \text{ } M^{-1}ms^{-1}$ ,  $k_{off} = 1 \text{ } ms^{-1}$
- 

### 3. SERCA Pump Flux

$$J_{pump} = V_{max} \frac{[Ca^{2+}]^2}{[Ca^{2+}]^2 + K_p^2}$$

where

- $V_{max}$ — maximal SERCA uptake rate
- $K_p$ — pump affinity ( $\approx 0.3 \mu\text{M}$ )

#### 4. Passive Leak Flux

$$J_{leak} = k_{leak}([Ca^{2+}]_{ER} - [Ca^{2+}]_{cyt})$$

where  $k_{leak}$  is the membrane leak constant.

#### 5. IP<sub>3</sub>-Dependent Channel Flux

$$J_{ch} = P_{open} g_{ch} ([Ca^{2+}]_{ER} - [Ca^{2+}]_{cyt})$$

Channel open probability:

$$P_{open} = \frac{[IP_3][Ca^{2+}]_{cyt}^3}{([IP_3] + K_{IP3})([Ca^{2+}]_{cyt}^3 + K_{act}^3)} \cdot (1 - h)$$

Inactivation variable  $h$ :

$$\frac{dh}{dt} = \frac{h_\infty - h}{\tau_h}, h_\infty = \frac{K_{in}^4}{K_{in}^4 + [Ca^{2+}]_{cyt}^4}$$

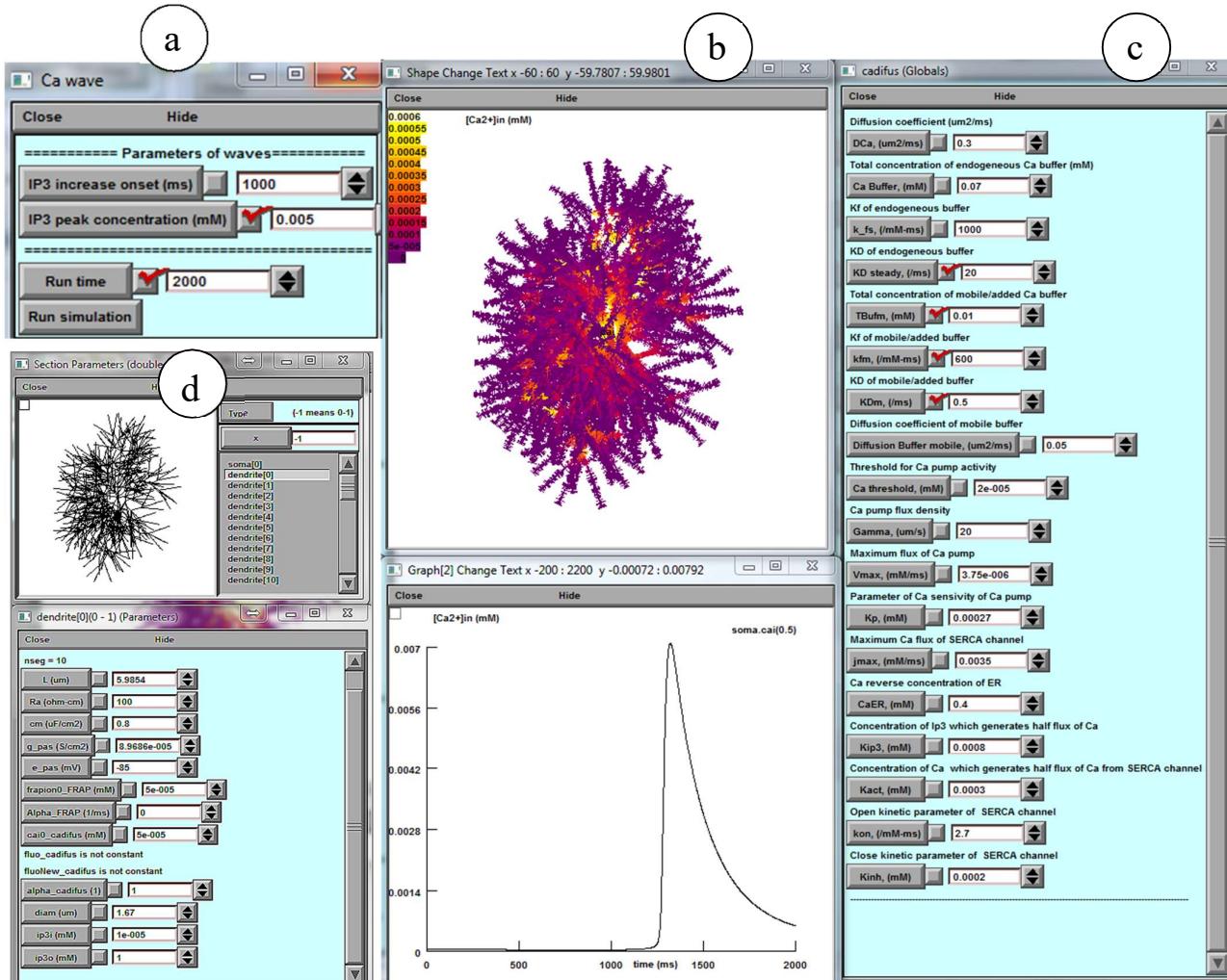
#### 6. Summary of Constants

Symbol	Meaning	Typical value
$D_{Ca}$	Diffusion coefficient	$0.22 \mu\text{m}^2 \text{ ms}^{-1}$
$V_{max}$	SERCA maximum uptake	$5 \times 10^{-3} \text{ mM ms}^{-1}$
$K_p$	SERCA half-activation	$0.3 \mu\text{M}$
$k_{leak}$	ER leak rate constant	$10^{-4} \text{ ms}^{-1}$
$g_{ch}$	IP <sub>3</sub> channel conductance	$5 \times 10^{-3} \text{ mM ms}^{-1}$
$K_{IP3}$	IP <sub>3</sub> dissociation constant	$0.1 \mu\text{M}$

#### Summary

This calcium-wave model reproduces realistic intracellular Ca<sup>2+</sup> propagation through combined diffusion, buffering, and membrane fluxes.

All parameters are editable through the **Ca<sup>2+</sup> Reaction–Diffusion Settings Panel (Fig. 31c)**, and the results are displayed dynamically in the **[Ca<sup>2+</sup>] Landscape (Fig. 31b)**.



**Figure 31. Control and Monitoring Panels for Ca<sup>2+</sup> Wave Dynamics**

- a — Ca<sup>2+</sup> Wave Trigger Settings Panel**

Configure the **trigger parameters** for initiating a Ca<sup>2+</sup> wave.

- b — Dynamic [Ca<sup>2+</sup>] Landscape and Readout Plot**

- Top:** Colour-coded map showing real-time Ca<sup>2+</sup> concentration ([Ca<sup>2+</sup>]) across the cell morphology.
- Bottom:** Readout plot of [Ca<sup>2+</sup>] dynamics, focusing on the **soma** by default.

- c — Parameter Setting Panel for Ca<sup>2+</sup> Reaction–Diffusion Processes**

Define **reaction–diffusion parameters** for intracellular Ca<sup>2+</sup> modelling.

- d — Ca<sup>2+</sup> Homeostasis & Dynamics Monitoring Panel**

Monitor Ca<sup>2+</sup> levels and dynamic changes in **individual cellular processes** (named *dendrites* in NEURON).

### Ca<sup>2+</sup> Wave Panel Controls (Fig. 31a)

- IP<sub>3</sub> increase onset (ms):** Time (ms) when IP<sub>3</sub> concentration starts to rise.
- IP<sub>3</sub> peak concentration (mM):** Peak IP<sub>3</sub> level (e.g., 0.005 mM shown).

- **Run time (s):** Total simulation time (default: **2000 s**).

## Ca<sup>2+</sup> Reaction–Diffusion Parameter Settings

The **Ca<sup>2+</sup> reaction–diffusion settings panel** (Fig. 31c) displays **explanations above each control key**, helping users adjust parameters with confidence.

- **Indicator properties:**

Concentration and kinetic parameters for the **Ca<sup>2+</sup> indicator (Fluo-4)** are **constrained by experimental data**.

- **Pumps & channels:**

Default values for Ca<sup>2+</sup> pumps and channels are based on **estimates from the literature**, but users should refine them to match **their own experimental tests**.

---

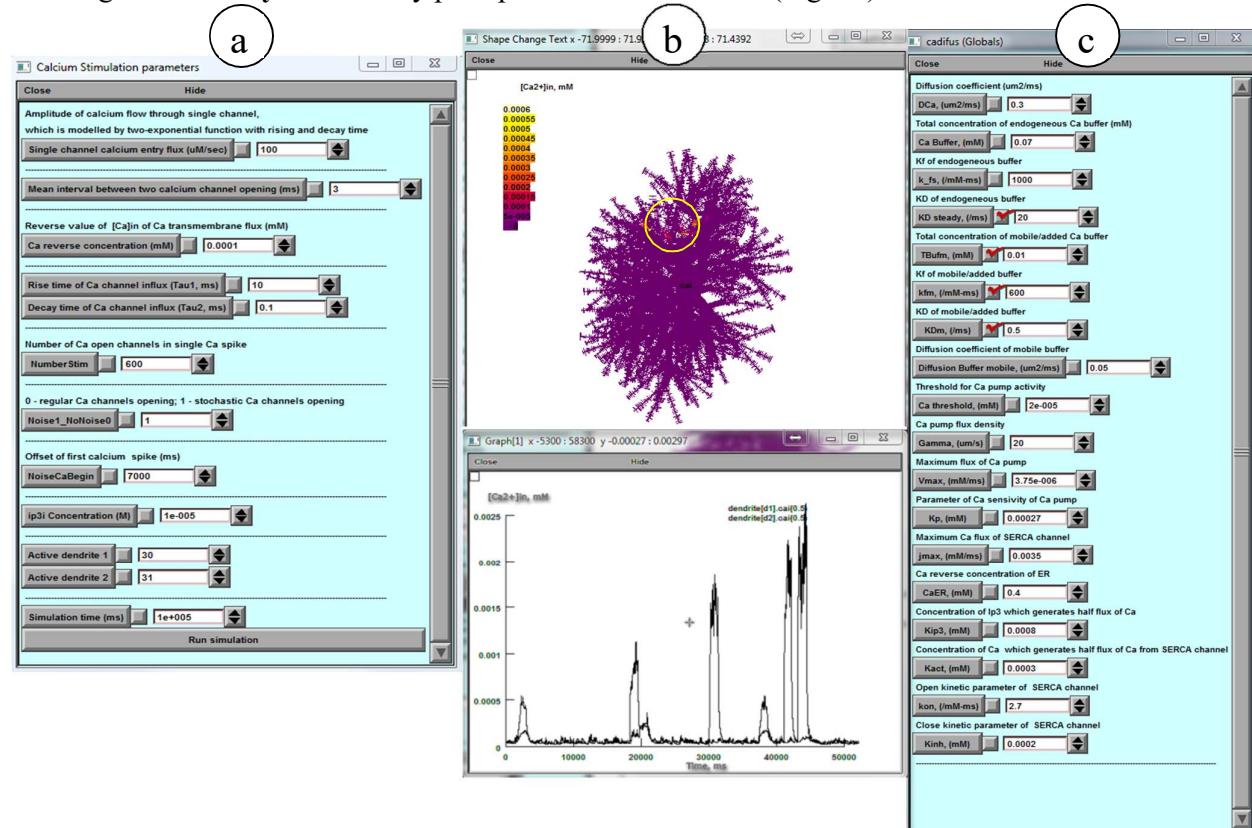
## Running & Restarting Simulations

- After a simulation run finishes, you can **start a new run** with updated parameters.
- The **Graph panel** (Fig. 31b, bottom) shows the **time course of [Ca<sup>2+</sup>] in the soma** for the current run.

↳ *Note:* The current BrainCell version simulates **IP<sub>3</sub> rises at the soma by default**, but it can be adapted to trigger **local IP<sub>3</sub> release** at any chosen region within an astrocyte.

## Simulating microscopic $\text{Ca}^{2+}$ events

Pressing 'Calcium dynamics' key prompts four new windows (Fig. 31):



**Figure 32. Panels for Controlling and Monitoring Microscopic  $\text{Ca}^{2+}$  Dynamics**

- **a — Parameter Settings for Single-Channel  $\text{Ca}^{2+}$  Dynamics**

Configure local  $\text{Ca}^{2+}$  release events, including **sparks** and **puffs**, in two selected *active branches* (example: dendrite numbers **30** and **31** in NEURON nomenclature).

- **b — Visualisation of Internal  $\text{Ca}^{2+}$  Dynamics**

- **Top:** Colour-coded map showing intracellular  $\text{Ca}^{2+}$  dynamics across the cell morphology.

- **Bottom:** Time courses of  $\text{Ca}^{2+}$  changes in two selected dendrites (d1 and d2; here corresponding to dendrites **30** and **31**).

The yellow circle highlights the **area of interest**.

- **c —  $\text{Ca}^{2+}$  Reaction–Diffusion Parameter Panel**

Settings for **reaction–diffusion processes** controlling intracellular  $\text{Ca}^{2+}$ , as described previously (see Fig. 31).

## Calcium Stimulation Parameters Panel

The **Calcium Stimulation Parameters** panel (Fig. 31a) lets you configure key settings that control **microscopic Ca<sup>2+</sup> entry dynamics**.

This feature models Ca<sup>2+</sup> influx as a **stochastic sequence of single-channel openings**, often occurring in **bursts**.

Because many of these microscopic parameters remain experimentally uncertain, the panel is designed to support both **exploratory simulations** and **hypothesis testing**.

---

## Adjustable Parameters

- **Single Channel Calcium Entry Flux**

Sets the rate of Ca<sup>2+</sup> influx during a single channel opening.

- **Mean Interval Between Two Calcium Events**

Defines the average time between individual Ca<sup>2+</sup> entry events.

- **Basal Ca<sup>2+</sup>**

Determines the baseline cytosolic Ca<sup>2+</sup> level, which acts as the **reverse concentration** for linear Ca<sup>2+</sup> efflux.

- **Rise / Decay**

Specify the **rise time** and **decay time** of Ca<sup>2+</sup> flux during a single channel event.

- **Events per Burst**

Sets how many channels open during each burst.

- **Randomness**

- **0:** Uniform random (Poisson-distributed) channel openings.

- **1:** Cyclic (regular) channel opening pattern.

- **Burst Onset**

Time when the first burst of channel openings begins.

- **IP<sub>3</sub> Concentration**

Initial concentration of IP<sub>3</sub>, which modulates Ca<sup>2+</sup> release dynamics.

- **Active Dendrite 1 / Active Dendrite 2**

Select which astroglial processes (*dendrites* in NEURON terminology) will exhibit active Ca<sup>2+</sup> entry.

- **Stimulation Time**

Total simulation duration.

⚠ *Note:* In astroglia, Ca<sup>2+</sup> dynamics are relatively slow; simulations should last **≥100 s** to fully capture the process.

---

Using this panel, you can **precisely shape the temporal and spatial profile** of Ca<sup>2+</sup> entry to investigate how microscopic channel events influence **cellular calcium waves** and other signalling behaviours.

## Simulating Glutamate Transporters

Neuronal membranes are rich in **high-affinity glutamate transporters** that generate **rapid inward currents** when glutamate binds.

In BrainCell, you can model these transporters' behaviour and their effect on membrane voltage.

To access this feature, open the **main panel** and press the “**Glutamate Transporters**” button. This opens a dedicated menu (Fig. 32) for simulating the **dynamic membrane voltage landscape** across the cell in response to **local, volume-limited glutamate application** (*uncaging*) at a chosen site.

---

### Uncaging Glutamate Panel (Fig. 33a, top)

This panel provides **self-explanatory controls** for setting up **glutamate uncaging** within a **defined circular area** of the cell.

Key adjustable parameters include:

- **Uncaging spot size and location**
- **Glutamate concentration**
- **Application duration**
- **Timing of release**

These settings allow you to mimic experimental glutamate application with high spatial and temporal precision.

---

### Transporter Kinetics

BrainCell implements the **six-state kinetic model** of glutamate transporters based on *Bergles & Jahr (1997, Neuron 19: 1297–1308)*.

Each transporter can exist in one of six states  $C_i$  ( $i=1,\dots,6$ ), representing sequential binding, translocation, and release steps.

- The complete kinetic scheme and rate constants are implemented in the `GluTrans.mod` file.
  - This ensures physiologically realistic inward currents associated with **EAAT-type transporters**.
- 

This module enables detailed **simulation of glutamate uptake** and its influence on **membrane voltage and ion homeostasis**, helping researchers explore **synaptic signalling** and **astrocyte–neuron interactions** under controlled glutamate release scenarios.

$$\begin{aligned}
 \frac{dC_1}{dt} &= C_1[Gl]_0 k_{12} u(v, -0.1) + C_2 k_{21} \\
 \frac{dC_2}{dt} &= C_1[Na]_0 k_{23} u(v, 0.5) + C_3 k_{32} \\
 \frac{dC_3}{dt} &= C_3 k_{34} u(v, 0.4) + C_4 k_{43} \\
 \frac{dC_4}{dt} &= C_4 k_{45} + C_5 k_{54}[Gl]_{in} \\
 \frac{dC_5}{dt} &= C_5 k_{56} u(v, 0.6) + C_6 k_{65}[Na]_{in} \\
 \frac{dC_6}{dt} &= C_6 [K]_{in} k_{61} + C_1 k_{16} u(v, 0.6)[K]_o
 \end{aligned}$$

where  $u(th, x) = \exp(\frac{th \times x}{2 \times 26.7})$

## Parameters

$k_{12}$	$= 20 \text{ (mM ms)}^{-1}$	$k_{21}$	$= 0.1 \text{ ms}^{-1}$
$k_{23}$	$= 0.015 \text{ (mM ms)}^{-1}$	$k_{32}$	$= 0.5 \text{ ms}^{-1}$
$k_{34}$	$= 0.2 \text{ ms}^{-1}$	$k_{43}$	$= 0.6 \text{ ms}^{-1}$
$k_{45}$	$= 4 \text{ (mM ms)}^{-1}$	$k_{54}$	$= 10 \text{ (mM ms)}^{-1}$
$k_{56}$	$= 1 \text{ ms}^{-1}$	$k_{65}$	$= 0.1 \text{ (mM ms)}^{-1}$
$k_{16}$	$= 0.0016 \text{ (mM ms)}^{-1}$	$k_{61}$	$= 2 \times 10^{-4} \text{ (mM ms)}^{-1}$

## Important Notes for Glutamate Uncaging Simulations

- **Uncaging onset:**

To maintain **stable membrane kinetics**, the onset of glutamate uncaging should be set to **> 3 ms** after the start of the simulation.

- **Multiple uncaging spots:**

You can add a **second uncaging site** that activates simultaneously with the first.

- Specify the **distance (X)** between the primary and secondary uncaging spots to control spatial stimulation patterns.

## Voltage Clamp Configuration

Transporter currents are calculated under a **somatic voltage clamp** setup:

- Clamp settings are defined in the **Somatic Voltage Clamp Panel** (Fig. 32b), including:
  - **Electrode resistance**
  - **Clamp holding voltage**
  - **Clamp duration**

This configuration ensures stable recording of glutamate transporter-mediated currents.

## Monitoring Panels

- **Membrane Current Landscape (Fig. 32c):**

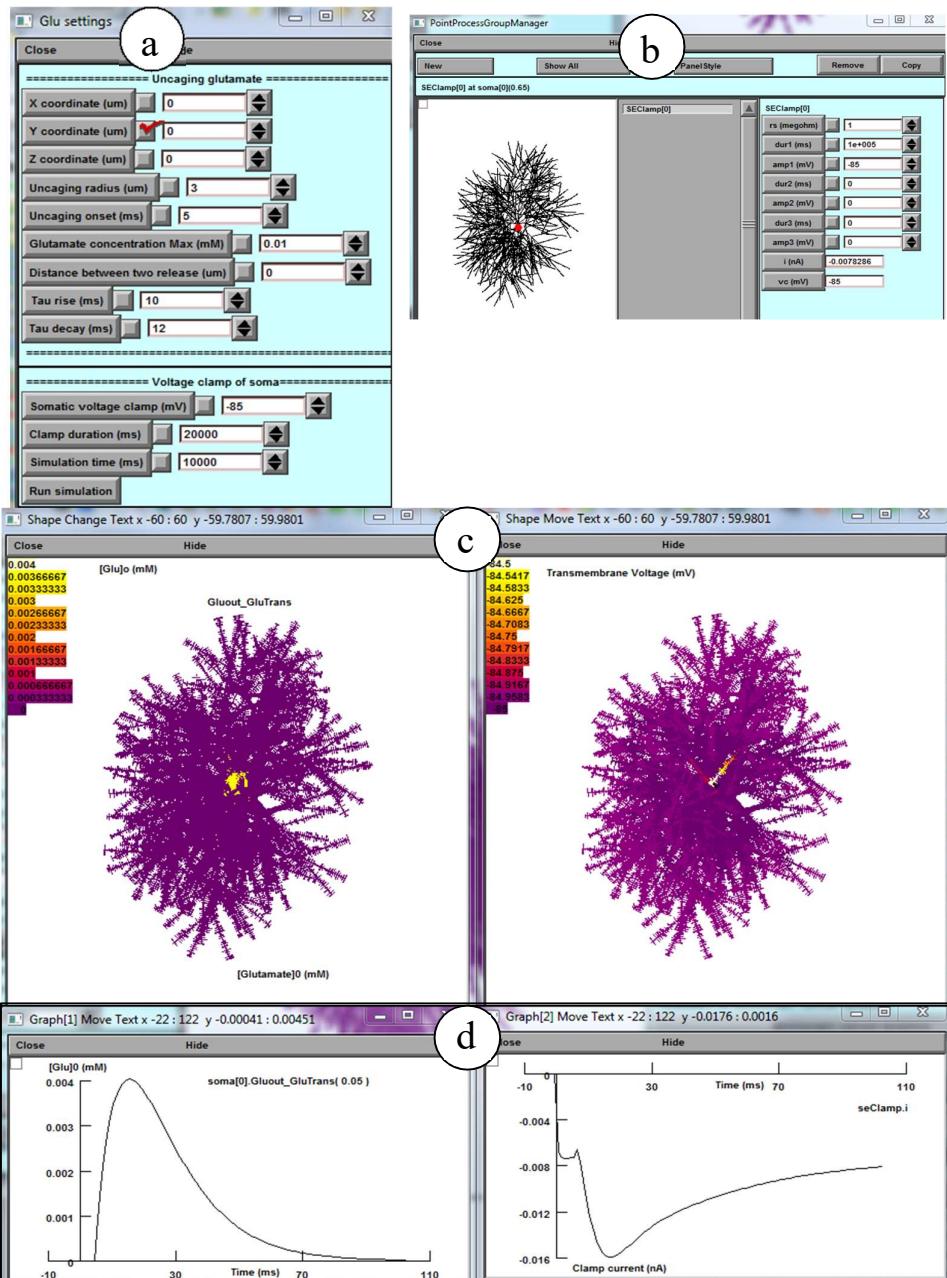
Colour-coded map showing transporter-driven **membrane current density** over the full cell morphology.

- **Graph Panel (Fig. 32d):**

Plots the **time course** of:

- **Extracellular glutamate concentration**, and
- **Clamp current** at selected cell regions.

This setup enables **precise spatiotemporal control** of glutamate applica



**Figure 33. Window panels with parameter settings and readout plots simulate the dynamic membrane voltage landscape mapped onto cell morphology in response to volume-limited application ('uncaging') of glutamate. a, Glutamate uncaging parameter settings. b, Voltage-clamp parameter settings, including electrode positioning. c, Visualisation panel. d, Time course graph panel.**

Window panel 'Glu setting' (Fig. 33a) provides parameters:

'X, Y and Z coordinate (um)' set respectively x, y and z-coordinates of the centre of the uncaging circle. 'Uncaging radius (um)' sets the radius of the uncaging circle.

'Uncaging onset (ms)' sets the onset of uncaging.

'Glutamate concentration Max (mM)' sets the maximum glutamate concentration during uncaging.

'Rise' and 'Decay tau (ms)' set the dynamics of glutamate uncaging.



## Simulating Potassium Dynamics (Intra- and Extracellular)

BrainCell allows you to **simulate potassium ( $K^+$ ) dynamics** inside and outside the cell, including **local  $K^+$  influx, intracellular redistribution, and efflux/leakage** (Fig. 34).

To access this feature:

1. Open the **main BrainCell panel**.
2. Click “**Potassium Dynamics**.”

This will open the dedicated potassium simulation windows (Fig. 34).

---

### Simulation Control & Visualisation

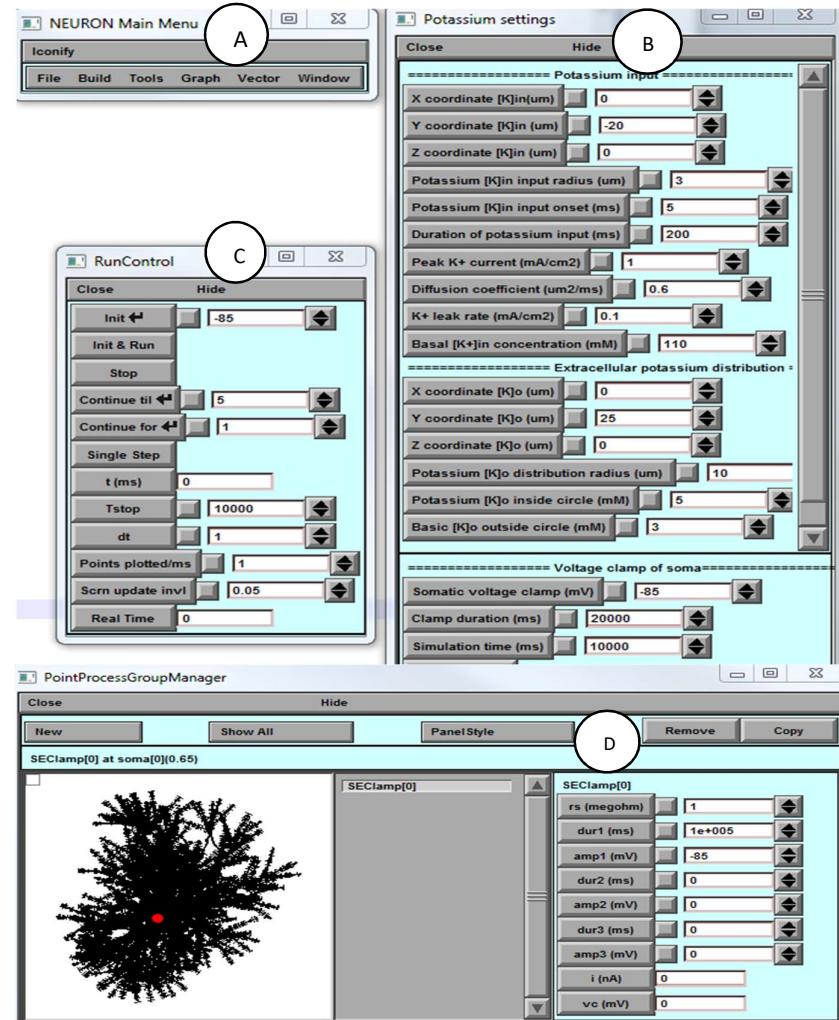
- A **standard NEURON panel** (Fig. 34) is used to:
  - Launch and control the simulation,
  - Display the evolving  $K^+$  concentration and voltage response.

### Potassium Settings Panel (Fig. 34b)

- Provides **self-explanatory controls** for configuring **local, volume-limited potassium application**:
  - Define the size and position of the  $K^+$  input area,
  - Adjust local current injection parameters.
- **$K^+$  dynamics** are calculated under **somatic voltage clamp** conditions.
  - Clamp parameters (electrode resistance, holding voltage, duration) are set in the **Somatic Voltage Clamp panel** (Fig. 34a, bottom).

---

This tool enables **precise modelling of potassium redistribution** under controlled local input and efflux conditions, with real-time visualisation of both **concentration changes** and **membrane response**.



**Figure 34. Window panels to control and monitor intracellular K<sup>+</sup> dynamics and extracellular K<sup>+</sup>-steady-state distribution.** a, Parameter settings for monitoring [K<sup>+</sup>]in and [K<sup>+</sup>]o landscape mapped onto cell morphology. b, Parameter settings pertinent to volume-limited [K<sup>+</sup>]in entry and [K<sup>+</sup>]o distribution. c, Shape visualisation with voltage-clamp positioning. d, Somatic voltage clamp applied (bottom), for the sake of simplicity.

### Important Notes on Potassium Dynamics Simulation

The current **potassium dynamics module** focuses on the **diffusion landscape of intracellular K<sup>+</sup>** and uses **simplified first-order kinetics** for K<sup>+</sup> entry and extrusion.

These reactions are set to be **independent of membrane voltage** to keep the simulation conceptually clear and computationally light.

#### ↳ Advanced modelling:

For more detailed and physiologically realistic simulations, users can include:

- **Voltage- and concentration-dependent K<sup>+</sup> pumps and channels,**
- **Extracellular K<sup>+</sup> dynamics,**
- **Unclamped membrane voltage.**

Such mechanisms — including **astroglial Kir4.1 channels** — are available through the **standard NEURON environment** and can be integrated for higher biological fidelity.

## Monitoring Panels

- **a — Intracellular  $[K^+]$  Landscape**

Colour-coded map of  $[K^+]_n$  (**internal potassium**) across the cell morphology.

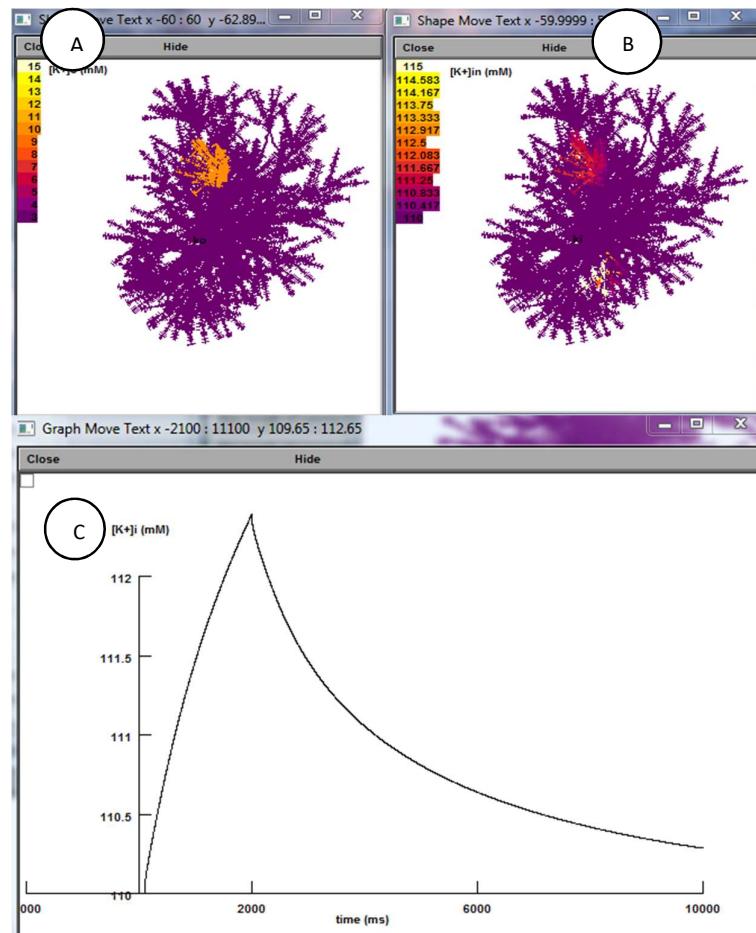
- **b — Extracellular  $[K^+]$  Steady-State Map**

Shows the **final distribution of  $[K^+]_{ext}$**  around the cell.

- **c — Time Course Plot**

Graph of **intracellular  $[K^+]$**  over time at a user-selected location (example: dendrite 51).

This setup allows researchers to **observe intracellular  $K^+$  redistribution** under simplified assumptions and serves as a starting point for **more detailed, biophysically complete potassium models** using NEURON's extended channel and pump libraries.



**Figure 35.** Window panels displaying simulated dynamics of intracellular  $[K^+]_{in}$  and extracellular  $[K^+]_{ext}$  steady-state distribution mapped onto cell morphology (a) and (b), and  $[K^+]_{in}$  in the time course (c) at a selected branch (dendrite), in response to volume-limited injection of potassium current (as in Fig. 34).

## Potassium Settings Panels

### 1. Intracellular Potassium Panel — “*Potassium setting*”

This panel defines the parameters for  $K^+$  entry and intracellular diffusion:

- **X, Y, Z coordinate  $[K^+]_n$  ( $\mu m$ ):** Sets the centroid coordinates of the  $K^+$  entry area.
- **Potassium  $[K^+]_n$  input radius ( $\mu m$ ):** Defines the radius of the  $K^+$  entry area.
- **Potassium  $[K^+]_n$  input onset (ms):** Sets the time (ms) when  $K^+$  entry begins.
- **Duration of potassium input (ms):** Duration of the  $K^+$  entry event.
- **Peak  $K^+$  current ( $mA/cm^2$ ):** Maximum amplitude of  $K^+$  influx current.
- **Diffusion coefficient ( $mm^2/ms$ ):** Intracellular  $K^+$  diffusion coefficient (controls how fast  $K^+$  spreads inside the cell).
- **$K^+$  leak rate ( $mA/cm^2$ ):** Defines  $K_p$ , the  $K^+$  extrusion current density.
- **Basal  $[K^+]_n$  concentration (mM):** Sets the resting intracellular potassium concentration.



The **Basal  $[K^+]_n$  concentration** appears twice in the original UI. It represents the **initial intracellular  $K^+$  level** before simulation starts.

---

### 2. Extracellular Potassium Panel — “*Extracellular potassium distribution*”

This panel controls the **extracellular  $[K^+]$  configuration**:

- **X, Y, Z coordinate  $[K^+]_{ext}$  ( $\mu m$ ):** Sets the centroid coordinates of the extracellular  $K^+$  region.
- **Potassium  $[K^+]_{ext}$  distribution radius ( $\mu m$ ):** Defines the radius of the extracellular  $K^+$  fluctuation area.
- **Potassium  $[K^+]_{ext}$  inside circle (mM):** Sets the  $[K^+]$  level inside the defined extracellular region.
- **Basic  $[K^+]_{ext}$  outside circle (mM):** Sets the baseline extracellular  $[K^+]$  outside the region.

## Basket cell GABA diffusion (Neuron mode only)

### Building and Analysing Realistic Basket Cell Models with GABA Release Dynamics

This tutorial guides you through creating and exploring **realistic basket cell neuron models**, with a particular focus on **GABA ( $\gamma$ -aminobutyric acid) release and uptake dynamics**. It is part of a series of step-by-step scenarios designed to help you construct biologically meaningful neuron models and study inhibitory neurotransmission.

In this tutorial, you will learn how to:

#### 1. Create GABA Synapses on Axon Terminals

Learn how to add and configure multiple **GABAergic synapses** on the axon terminals of the basket cell model. This step is critical for understanding how the **number and distribution of synapses** influence GABA release.

#### 2. Modify the Synaptic Release Function

Explore how to adjust the **release probability and kinetics** of GABA synapses. This modification allows investigation of how different presynaptic release mechanisms affect **GABA diffusion and signalling**.

#### 3. Include GABA Uptake Mechanisms

Incorporate **GABA transporters** into the extracellular space to model **uptake and clearance**. This step helps visualise how reuptake regulates inhibitory signalling and affects the spatiotemporal profile of GABA concentration.

#### 4. Design GABA Concentration Dynamics Visualisations

Learn to create **dynamic visualisations and videos** of GABA concentration changes over time. These visual outputs help reveal how GABA spreads, diffuses, and is taken up during and after synaptic release.

---

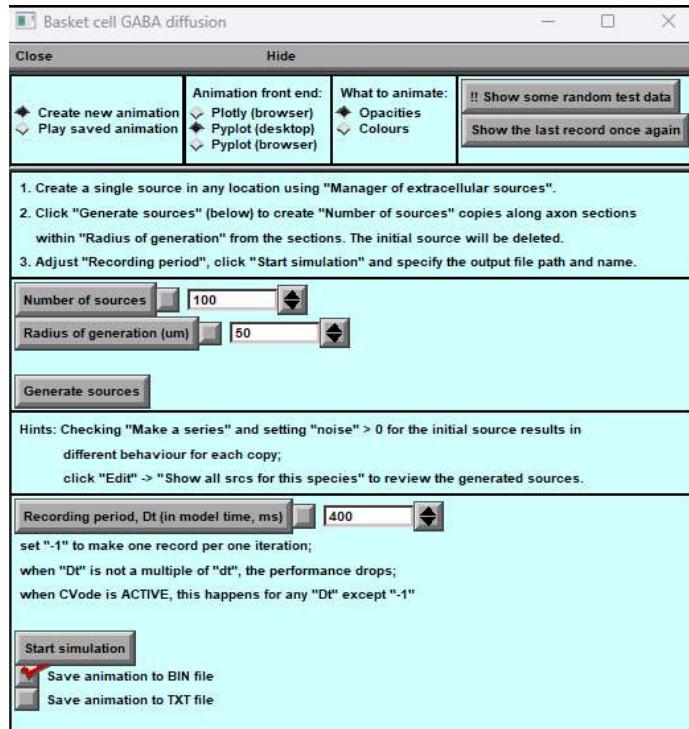
### Purpose and Applications

By completing this tutorial, you will be able to:

- Model **regular and irregular GABA release patterns** from basket cells.
- Study how synapse number, release probability, and uptake alter **inhibitory signalling**.
- Generate **publication-quality videos** showing GABA concentration dynamics in the extracellular space.

This workflow is particularly useful for researchers investigating **inhibitory interneurons, synaptic integration**, and **network behaviour** under varying inhibitory strengths and spatial distributions.

The main panel window.



**Figure 36. Source of Diffusion Panel**

This panel provides a comprehensive interface for creating, running, and visualising diffusion simulations with optional animation export.

## Animation Controls

- **Create new animation** — Start a new animation sequence for the current simulation.
- **Play saved animation** — Replay a previously saved animation.
- **Plotly (browser)** — Use the Plotly graphing library in a web browser for visualisation.
- **Pyplot (desktop)** — Use the Pyplot library on the desktop for visualisation.
- **Pyplot (browser)** — Use the Pyplot library in a browser for visualisation.

## What to Animate

- **Opacities** — Animate opacity changes of simulation elements.
- **Colours** — Animate colour changes over time.
- **Show some random test data** — Display test data for debugging or visual checks.
- **Show the last record once again** — Replay the most recent recorded simulation data.

## Simulation Setup

- **Create a single source in any location** — Create a single extracellular diffusion source via the *Manager of Extracellular Sources*.

- **Generate sources** — Automatically generate multiple copies of a diffusion source distributed along axon sections within a specified radius. (The original source is removed after this operation.)
  - **Adjust Recording Period & Start Simulation** — Define the recording interval and start the simulation; specify the output file location and name.
- 

## Simulation Parameters

- **Number of sources** — Set the number of diffusion sources to generate.
- **Radius of generation ( $\mu\text{m}$ )** — Define the spatial radius for source generation around axon sections.
- **Generate sources** — Execute source generation with the chosen settings.

## Hints

- **Make a series** — Enable to create a series of simulation points for extended analysis.
- **Noise** — Enter a value  $>0$  to introduce variability in each source's behaviour.

## Recording Period (Dt)

- Enter the time step (in ms) for saving simulation data.
  - Setting **Dt = -1** records every simulation iteration (highest temporal resolution).
  - Performance may decrease if **Dt** is not a multiple of the simulation time step (**dt**).
  - When **CVode** is active, only **Dt = -1** avoids performance issues.
- 

## Simulation Controls

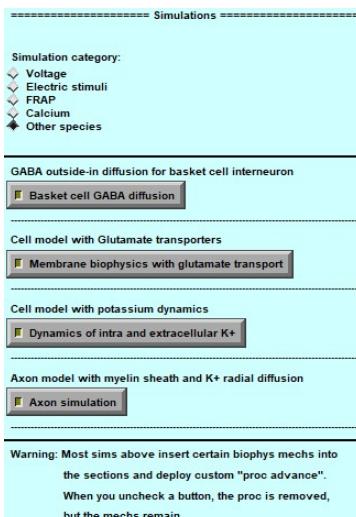
- **Start simulation** — Launch the simulation with the current parameters.
  - **Save animation to BIN file** — Export the animation in binary format.
  - **Save animation to TXT file** — Export the animation in plain text format.
- 

This panel combines **diffusion source configuration**, **simulation control**, and **advanced visualisation tools**, making it easy to run customised diffusion experiments and generate high-quality animations for analysis or presentation.

## Axon Myelination (Neuron Mode Only)

BrainCell allows you to add myelin to the axon of your neuron model, enhancing simulation realism.

1. **Load or Create a Neuron Model**
  - o If your neuron model already includes an axon, you can proceed directly to myelination.
  - o If no axon is present, you can either **add an artificial axon** or **manually create one** before starting the myelination process.
2. **Initiate Myelination**
  - o In the main menu, go to **Other Species → Axon Simulation**.
  - o This opens the myelination tools, allowing you to define the axon's myelin sheath properties and simulate its effect on axonal conduction.



**Figure 37. Myelinated Axon Modelling Panel.**

This panel enables users to create, import, and simulate a **neuron model with a myelinated axon**.

- The **geometry section** allows importing or manually creating an axon.
- **Myelin and Schwann cell parameters** control sheath properties and coverage.
- **Biophysics, clamps, and diffusion settings** enable realistic conduction modelling.
- A **visual preview** (left side) updates interactively as parameters are changed.

Once configured, the user can launch the simulation to analyse **signal propagation and ion dynamics** in a myelinated axon.

## Myelinated Axon Simulation Panel

After clicking “**Axon Modelling**”, the **main control panel** for creating and simulating an axon with myelin (**Schwann cells**) appears (Fig. 37–38).

This panel enables you to **import, design, and configure** a neuron model with a myelinated axon and is divided into several functional sections:

1. **Geometry & Myelin Sheath**
  - Import an existing axon geometry, select from predefined templates, or create one manually.
  - Deploy, scale, or remove the myelin sheath as needed.
2. **Myelin Parameters**
  - Define regions of myelination along the axon.
  - Configure **Schwann cell** properties and myelin coverage.
3. **Axon Trunk Parameters**
  - Set the **axon length, diameter, and segmentation** for conduction modelling.
4. **Axon Sheath Parameters**
  - Specify **Schwann cell diameter, number of radial shells**, and sheath structure.
5. **Biophysics Parameters & Clamps**
  - Choose between **realistic axonal biophysics** or simplified **test sine wave simulations**.
  - Apply **somatic clamps** to control and measure membrane potential.
6. **Diffusion Parameters**
  - Adjust the **diffusion coefficient** and set **intracellular/extracellular potassium concentrations**.
7. **Visualisation Options**
  - Select visualisation modes to analyse the axon's electrical and biophysical behaviour.

The **left-hand side** of the panel shows a **visual representation** of the axon model.

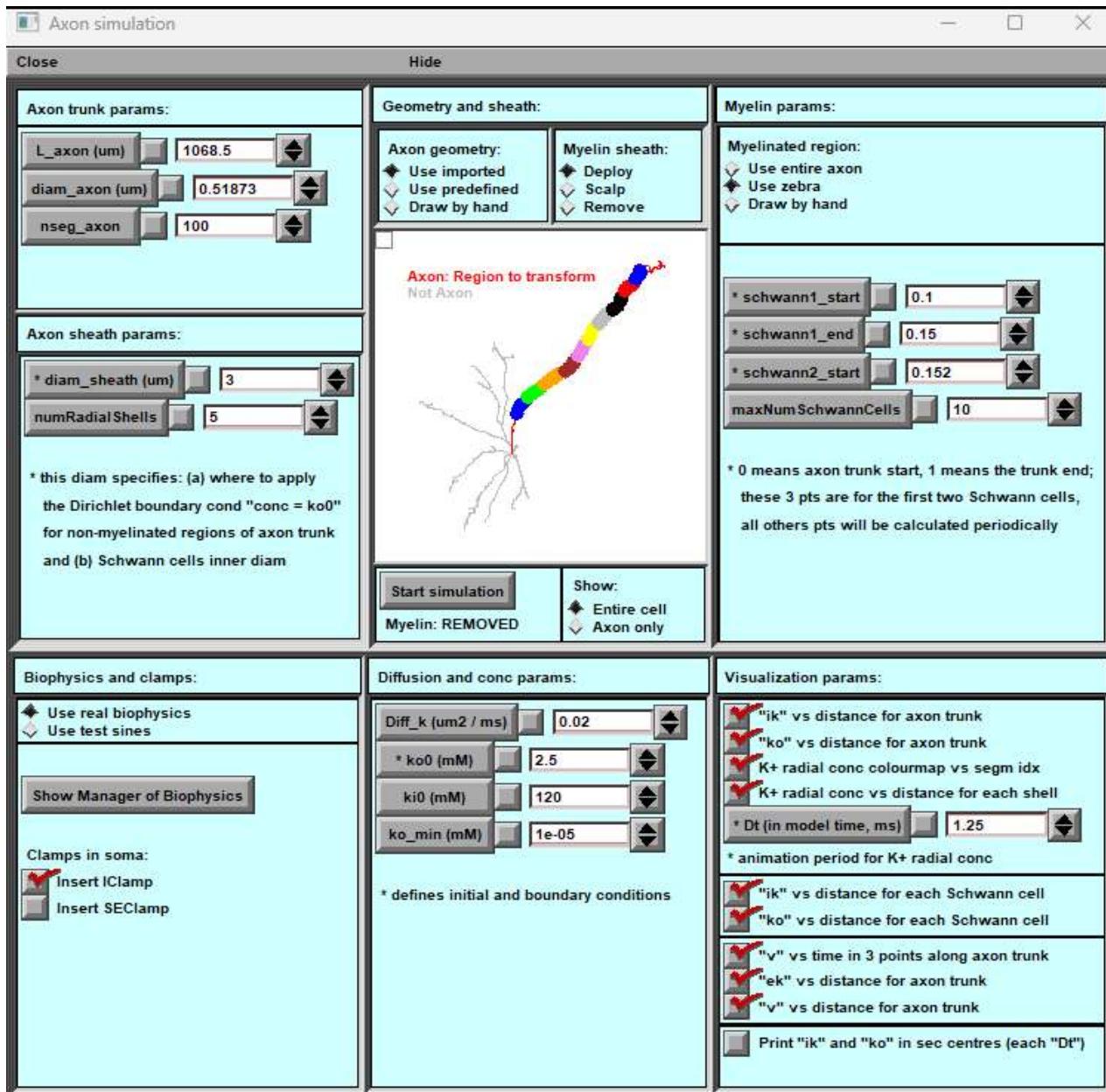
Use the **input fields, sliders, and checkboxes** to modify parameters.

Click “**Start Simulation**” to run the model.

**Tip:** Some parameters are interdependent — changing one may affect the availability or validity of others.

For advanced configuration details, refer to the dedicated sections of this manual.

---



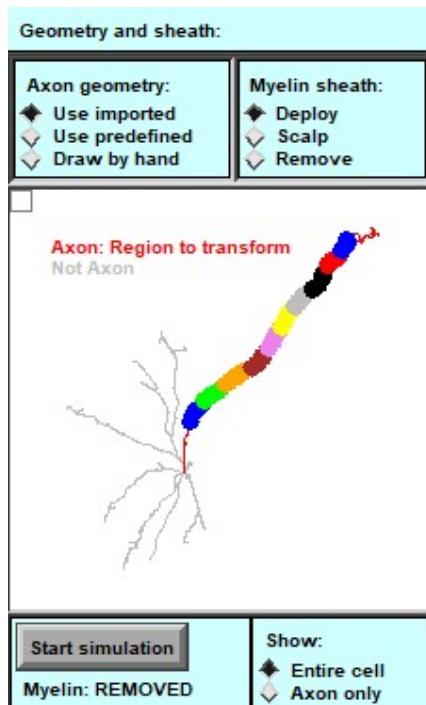
**Figure 38.** Axon Myelination Panel.

The panel provides comprehensive tools for configuring and simulating myelinated axons.

- **Geometry and Sheath:** Define axon geometry (import, use predefined templates, or create manually), apply or remove myelin, and select specific regions to convert into myelinated sections.
- **Myelin Parameters:** Specify regions for myelination, adjust Schwann cell or oligodendrocyte parameters (start and end positions), and set the maximum number of myelinating cells.
- **Axon/Sheath Parameters:** Configure visualisation of the sheath by setting axon length, diameter, number of segments, and radial shell structure.
- **Biophysics and Clamps:** Add or modify biophysical properties, apply predefined parameter sets, and test propagation using clamp simulations.

- **Diffusion Parameters:** Define ion diffusion coefficients and initial concentrations for the simulation environment.
- **Visualisation Parameters:** Select which simulation variables (e.g., ion concentrations or membrane voltage) are displayed dynamically over time and along the axon.

### *Geometry and Sheath Panel*



**Figure 39. Geometry and Sheath Panel Overview.**

This panel defines the structural foundation of the axon and its myelin sheath.

- **Axon Geometry:** Specify axon length, diameter, and segmentation pattern; import existing models or create new ones manually.
- **Sheath Configuration:** Apply or remove myelin along selected axonal segments and define the spatial range of myelination.
- **Region Selection:** Use graphical tools to highlight and transform specific axon regions into myelinated sections.
- **Scaling Controls:** Adjust scaling factors for length and radial thickness to match experimental or theoretical reference data.
- **Preview Display:** Provides a real-time 3D visualisation of the axon and sheath structure, updating dynamically as parameters are modified.

## 2. Myelin Sheath

- **Deploy** — Add a **myelin sheath** to the axon, simulating its insulating properties.
- **Scalp** — Disable myelin's electrical function while **preserving its geometry** (useful for testing geometry-only effects).
- **Remove** — Fully remove the myelin sheath, leaving an unmyelinated axon.

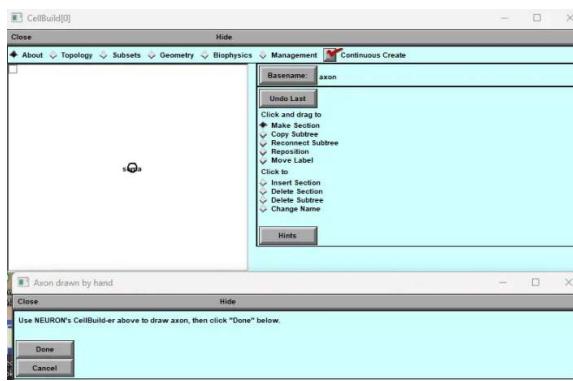
A **visual preview** highlights the axon and its segments, updating in real time as changes are made. Click “**Start Simulation**” to run the model with the selected geometry and myelination settings. A status indicator below the preview shows the current myelination state (e.g., “*Myelin: REMOVED*”).

**Note:** Adjusting these settings may impact other simulation parameters (e.g., conduction velocity, diffusion properties).

Review related panels (e.g., **Myelin Parameters** or **Biophysics Settings**) to ensure the model remains consistent.

#### *Draw Axon Panel Overview*

When you select "Draw by hand" in the **Geometry and Sheath** panel, NEURON’s **CellBuilder** window opens, allowing for manual construction of neuronal structures, including axons.



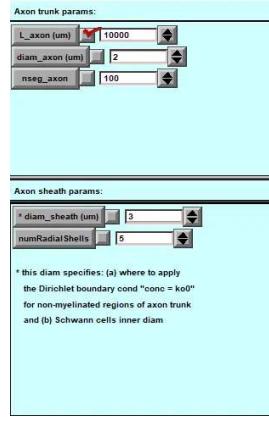
**Figure 40. Draw Axon Panel Overview.**

This panel allows users to construct and edit axonal geometry interactively.

- **Manual Drawing Tools:** Create or modify axon trajectories by drawing directly within the 3D workspace or importing path coordinates.
- **Node Placement:** Add, move, or delete nodes to define bends, branch points, or connection sites along the axon.
- **Length and Orientation Controls:** Adjust overall axon length, curvature, and spatial orientation to match experimental reconstructions or modelling requirements.
- **Integration with Myelination Tools:** The defined axon path can be used immediately for sheath generation in the *Geometry and Sheath* panel.
- **Visual Feedback:** Real-time 3D rendering provides immediate visual confirmation of geometry changes and segment connectivity.

#### *Axon Parameters Panel Overview*

This panel is used to configure the axon's physical properties and its myelin sheath. It consists of two main sections:



## 1. Axon Trunk Parameters:

- **L\_axon (μm):** Defines the axon length in micrometres (default: 10,000 μm).
- **diam\_axon (μm):** Specifies the axon diameter (default: 2 μm).
- **nseg\_axon:** Number of segments modelling the axon (default: 100).

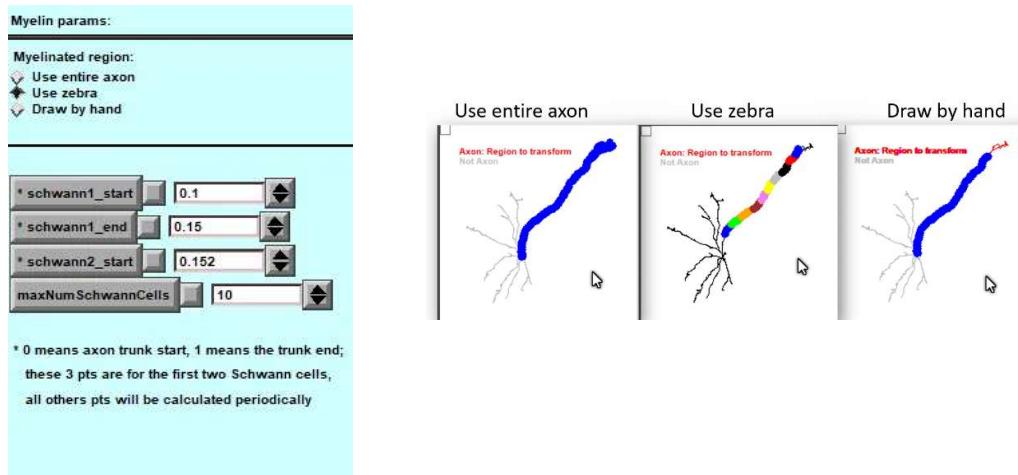
## 2. Axon Sheath Parameters:

- **diam\_sheath (μm):** Diameter of the myelin sheath (default: 3 μm).
- **numRadialShells:** Number of concentric shells modelling the sheath (default: 5).

**Important:** The **diam\_sheath** also applies boundary conditions to non-myelinated axon regions and sets the inner Schwann cell diameter.

Adjust parameters with caution, as they significantly impact the model's accuracy.

## Myelin Parameters Panel



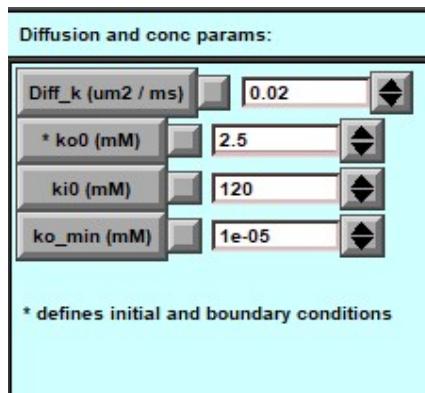
**Figure 41. Myelin Parameters Panel.**

This panel defines the detailed biophysical and structural parameters of the myelin sheath along the axon.

- **Cell Positioning:** Specify the start and end coordinates for individual myelinating cells (Schwann cells or oligodendrocytes) along the axon.

- **Number of Cells:** Set the maximum number of myelinating cells to be deployed per axon segment.
- **Internode and Node Parameters:** Adjust internodal distances, node lengths, and overlap ratios to reproduce realistic myelin organisation.
- **Sheath Thickness and Layers:** Define the radial thickness, number of wraps, and compaction ratio of the myelin sheath.
- **Electrical Properties:** Configure membrane resistivity, capacitance, and periaxonal space parameters to influence signal conduction.
- **Preview Window:** Displays a schematic of the myelin pattern along the axon, updating automatically when parameters are changed.
- .

#### *Diffusion and Concentration Parameters Panel*

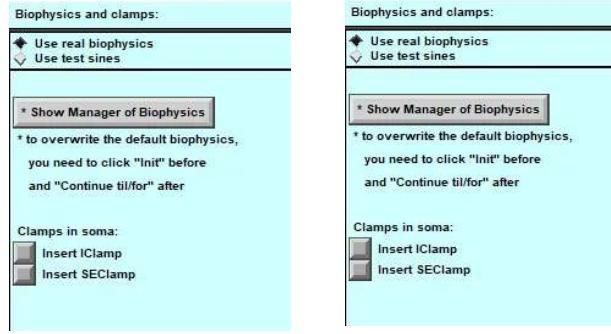


**Figure 42.** This panel lets to set potassium diffusion and concentration values for myelinated axon simulation.

1. **Diff\_k (um<sup>2</sup>/ms):** Sets the potassium ion diffusion coefficient (e.g., 0.02), controlling how quickly potassium spreads.
2. **ko0 (mM):** The initial extracellular potassium concentration (e.g., 2.5), also used as the boundary condition.
3. **ki0 (mM):** Initial intracellular potassium concentration (e.g., 120 mM).
4. **ko\_min (mM):** The minimum extracellular potassium concentration (e.g., 1e-05 mM) to prevent potassium depletion.

Carefully adjust these values to model physiological conditions, as they affect ion movement and action potential propagation during the simulation.

#### *Biophysics and Clamps Panel*



**Figure 43.** This panel allows you to configure the biophysical properties and experimental manipulations for your axon simulation. The panel has two different configurations, which I'll describe below:

- Use real biophysics: Selected
- Use test sines: Not selected

#### Configuration 1: Realistic Biophysics

- **Use real biophysics:** Activates realistic models for axonal behaviour.
- **Show Manager of Biophysics:** Opens an advanced settings panel.
- **Clamps:** Options to insert current (IClamp) or voltage clamps (SEClamp) into the soma.
  - *Tip:* Click "Init" before and "Continue til/for" after adjusting overwrite defaults.

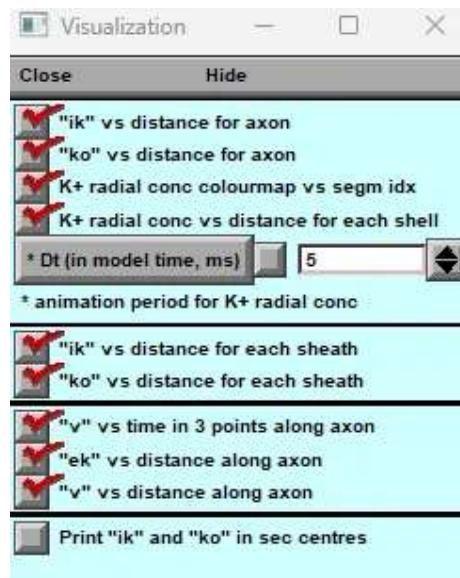
#### Configuration 2: Test Sine Waves

- Use real biophysics: Not selected
- Use test sines: Selected
  - **Use test sines:** Uses simplified sine waves for controlled testing.
  - **Clamps:** Same options available for experimental manipulation.

#### User Notes:

- Use real biophysics for accurate physiological simulations and test sines for simplified experiments.

#### Visualisation of computation



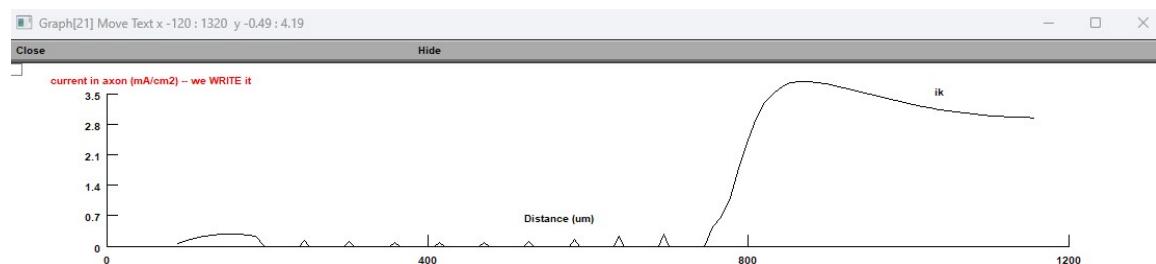
**Figure 44.** Panel for visual results.

- Checkbox (checked): Indicates that the option is enabled or selected
- Checkbox (unchecked): Indicates that the option is disabled or not selected
- ▼ - Dropdown arrow: Allows adjusting numerical values

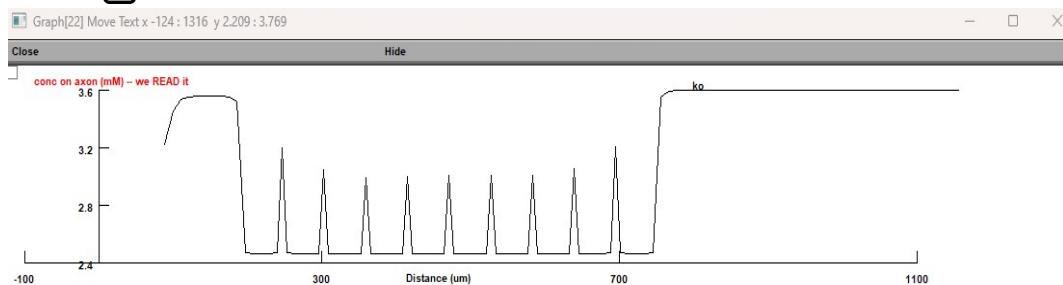
Legend for visualisation options:

1.  "ik" vs distance for axon

Here is an example of plots if the user keeps these options.



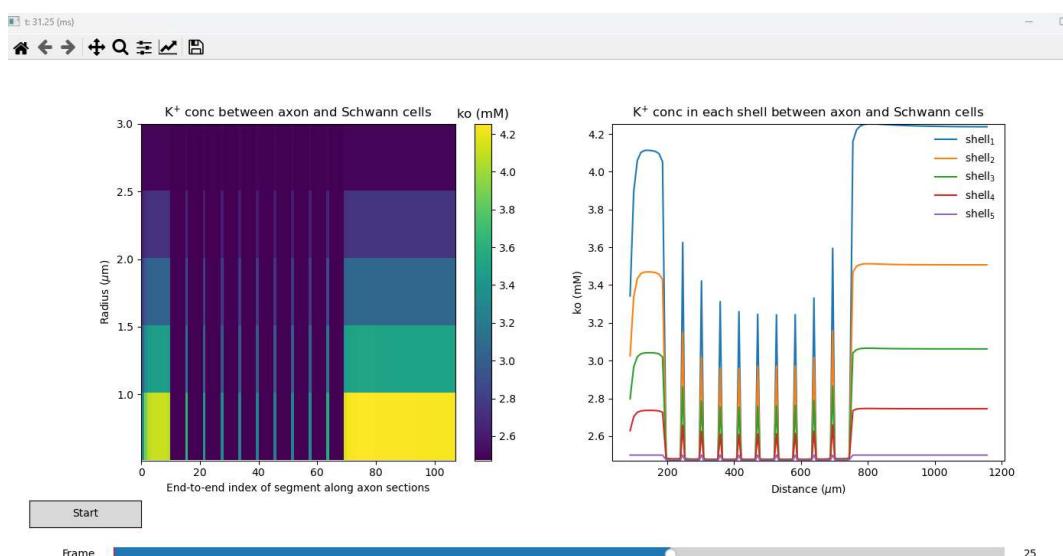
2.  "ko" vs distance for axon



3.  [K]<sup>+</sup> radial concentration colourmap vs segment idx

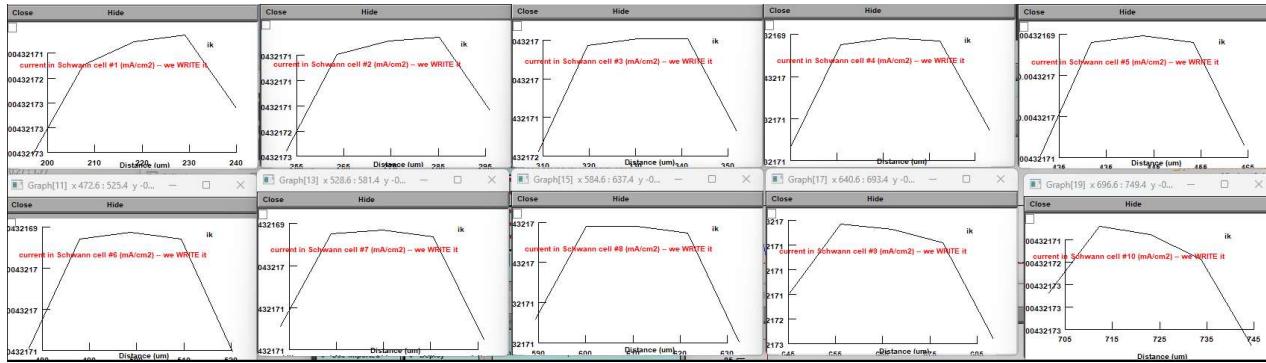
4.  [K]<sup>+</sup> radial concentration vs distance for each shell

5. Here is an example of plots if the user keeps these two options.



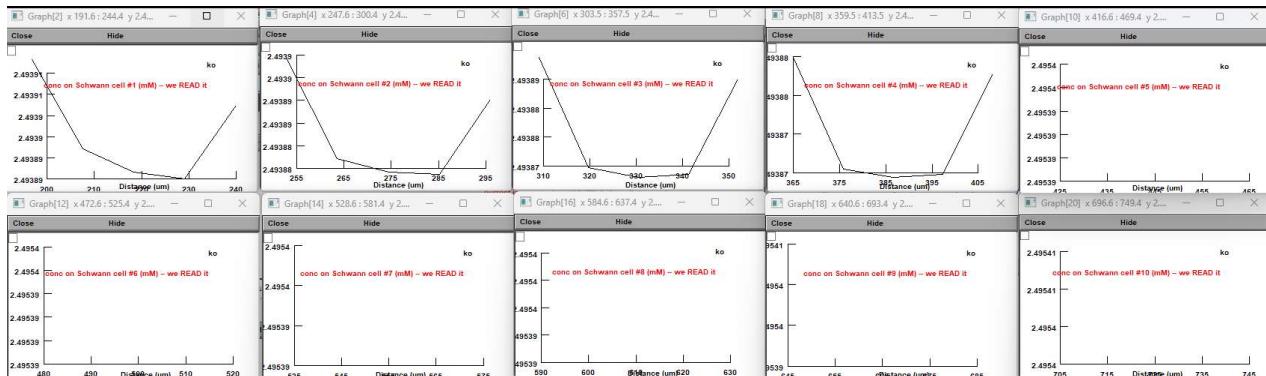
6.  Dt (in model time, ms): 20 [Numerical input field with dropdown] \* animation period for K<sup>+</sup> radial concentration
7.  "ik" density of potassium current vs distance for each myelinated sheath

Here is an example of plots if the user keeps these options.

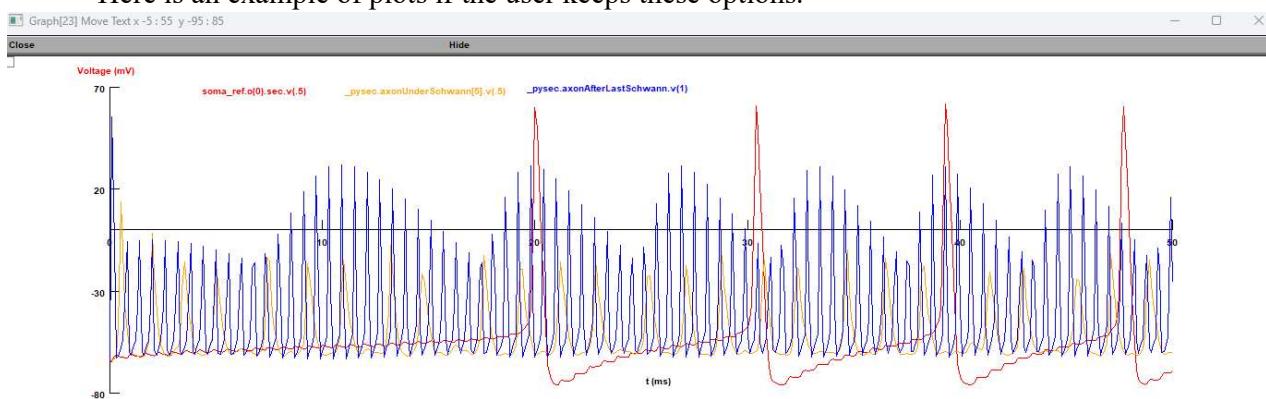


8.  "ko" [K]<sub>o</sub> extracellular potassium concentration vs distance for each myelinated sheath

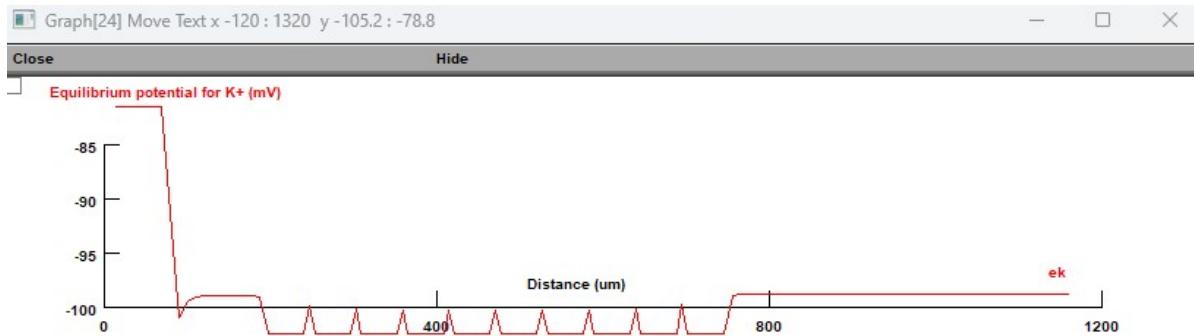
Here is an example of plots if the user keeps these options.



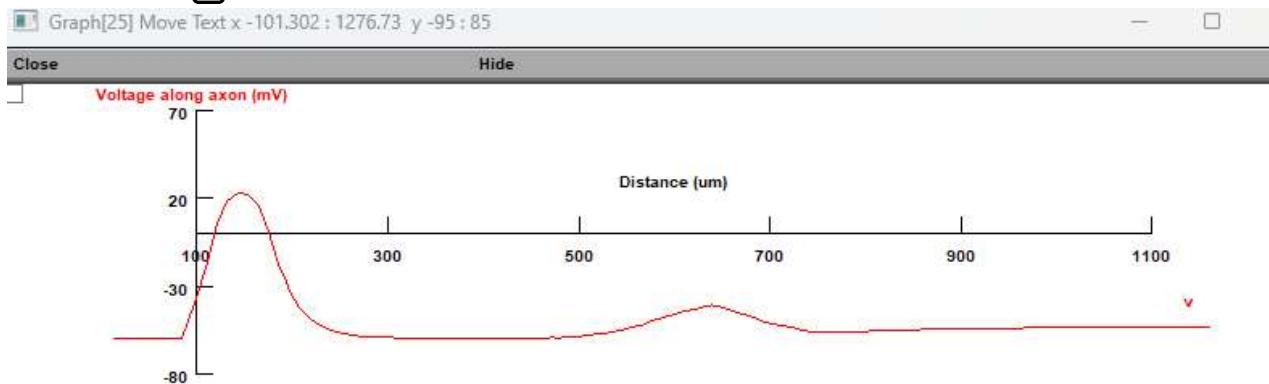
9.  "v" axon membrane voltage vs time in 3 points along axon trunk
- Here is an example of plots if the user keeps these options.



10.  "ek" potassium reverse potential on axon branches vs distance for axon trunk



11.  "v" vs distance for axon trunk



11.  Print "ik" and "ko" in sec centres [Unchecked option]

## LFP. Recording local field potential for a single cell.

### Local Field Potential (LFP) Simulation

**Local Field Potentials (LFPs)** represent the **combined electrical activity** of nearby neurons and are a soft for studying how neural circuits interact and communicate.

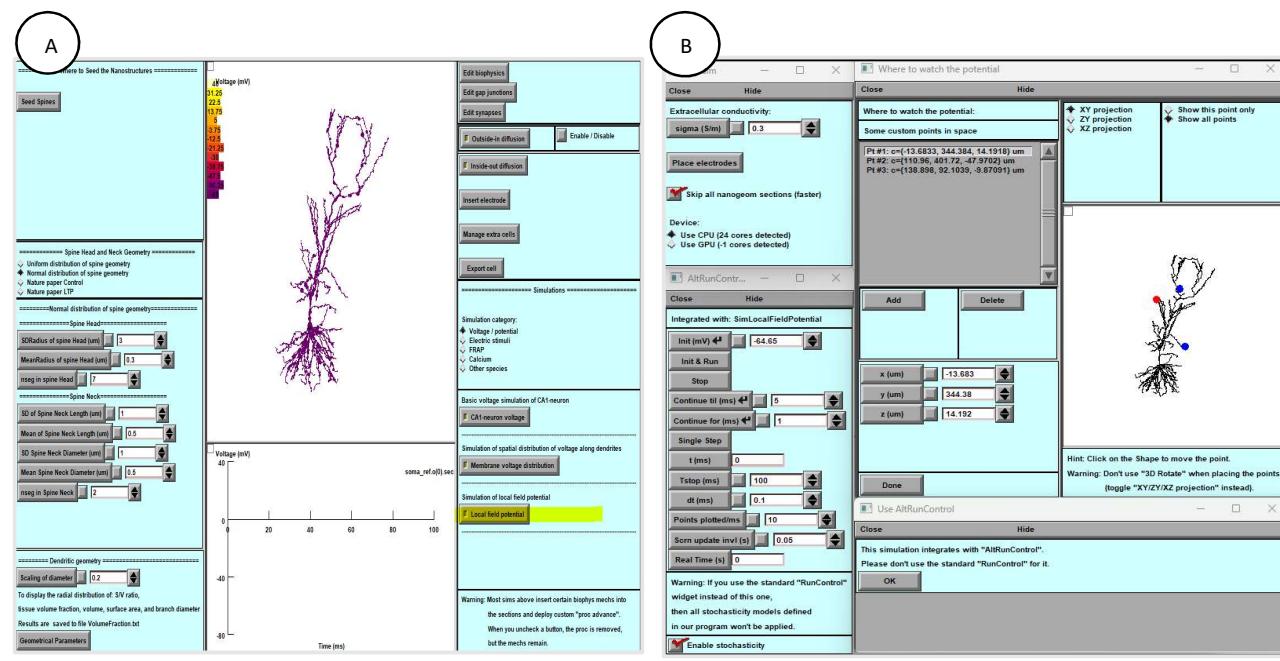
The **LFP simulation module** in BrainCell allows researchers to **calculate extracellular potentials** at user-defined spatial locations around a neural structure.

This feature is particularly useful for investigating how neural activity spreads through tissue and influences the surrounding environment.

The interface is designed to be **intuitive and flexible**, enabling users to:

- Precisely **position virtual recording electrodes** in the extracellular space.
- Define **simulation parameters** and introduce **stochastic variability** to match experimental conditions.
- Run **real-time LFP simulations** and monitor field dynamics as the model evolves.
- Visualise **simultaneous recordings** from multiple points around the neuron or network.

The **interactive workflow** allows immediate feedback: users can adjust parameters, reposition electrodes, and see the impact on LFP signals without restarting the full simulation.



**Figure 45. Local Field Potential (LFP) Interface**

The LFP interface provides tools to:

1. **Configure recording locations** in extracellular space.

2. Run **real-time simulations** of local field potentials around single cells or neural populations.
3. **Visualise multiple simultaneous recordings** across spatially distributed electrodes.
4. Monitor the **dynamics of extracellular fields** during ongoing simulations.

### **Sub panels.**

1. **Extracellular Conductivity ( $\sigma$ ):**  
Adjust the conductivity of the extracellular medium in Siemens per meter (S/m).
  - o Default value: 0.3 S/m.
  - o Modify based on the experimental conditions or tissue type.
3. **Electrode Placement Options:**
  - o **Place Electrodes:** Opens a sub-panel to define custom points in space for potential monitoring.
  - o **Skip All Nanogeom Sections:** Ignoring small geometry sections improves simulation speed.
4. **Device Selection:**
  - o Choose between CPU and GPU for simulation execution. The panel indicates the number of available cores for each.
5. **"Where to Watch the Potential" Sub-panel:**
  - o **Custom Points in Space:** List of coordinates (in micrometres) where potentials will be calculated.
  - o **Visualization Tools:**
    - Toggle between XY, XZ, and YZ projections of the neural structure.
    - Points are shown as coloured dots on the neural structure for reference.
    - Use the "Add" and "Delete" buttons to manage observation points.
6. **Alternative Run Control (AltRunControl):**
7. This customised simulation control panel integrates stochastic modelling features. **Do not use standard RunControl when AltRunControl is active.**

### **Key Parameters:**

- o **Initialization (Init):** Set the initial voltage (in mV).
- o **Run Control:**
  - Define simulation start and stop times (ms).
  - Configure plotting intervals and points displayed per millisecond.
- o **Enable Stochasticity:** Check this box to activate stochastic models in the simulation.

### **Real-Time Monitoring:**

- o Specify the screen update interval (in seconds) to adjust visualisation refresh rates during real-time simulations.

---

### **Visualisation and Controls:**

- **3D Shape Viewer:** Interact with the neural structure to adjust point positions visually. Avoid using "3D Rotate" mode for precise positioning.

### **Warnings:**

- When using AltRunControl, standard RunControl widgets are disabled to ensure compatibility with stochastic models.
-



## Submitting BrainCell Export Files to the NSG (Neuroscience Gateway)

The **NSG Portal** allows users to run BrainCell simulations on high-performance computing clusters. Follow these steps to prepare and submit your BrainCell models for execution on the NSG platform.

### 1. NSG Portal Login Interface

Access the NSG portal and sign in with your account credentials.

Missing results?  
Send us the job handle, and we may be able to help.  
More Information  
About Us  
Usage Statistics  
Enabled Publications  
Latest Workshops

Welcome to NSG. Currently various neuroscience modeling and data processing tools can be run on supercomputing resources provided by the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support program. Please note that if you are using human subject data, then the data should be de-identified.  
First Time Users: Please review the usage policy  
New users who are interested in getting an account should fill out the form

NSG Login:

Username

Password

[Forgot Password?](#)

You can: [Report an issue](#) [Request a feature](#) [Subscribe to Portal News](#)  
SDSC - UC San Diego, MC 0505 - 9500 Gilman Drive - La Jolla, CA 92093-0505

SDSC SAN DIEGO SUPERCOMPUTER CENTER UC San Diego

**Figure 46.** Neuroscience Gateway (NSG) login interface. The NSG (Neuroscience Gateway) portal provides access to high-performance computing (HPC) resources for large-scale neuroscience simulations.

Users can log in with their NSG credentials to run BrainCell and NEURON-based models on supercomputing clusters hosted by the San Diego Supercomputer Center (SDSC) at UC San Diego.

The page allows new users to register, existing users to log in, and includes links to usage policies, statistics, and support documentation.

### 2. Prepare Your Archive

- Create a **single folder** containing **all required BrainCell files** (e.g., your main HOC file, MOD files, and related data).

- **Folder naming:**

Use a simple name with no **Unicode characters, spaces, or special symbols**.

- **Linux compatibility:**

Open the main HOC file (e.g., `runner.hoc`) and add the line:

- `{ load_file("stdrun.hoc") }`

This ensures NEURON recognises standard functions on Linux.

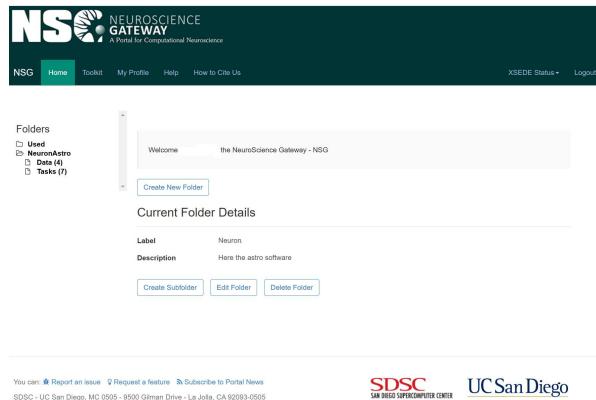
- Make sure the **main input file** is correctly specified in the NSG task settings (usually `runner.hoc`).

**⚠ Cloning an existing task may reset this field — double-check before running.**

### 3. NSG Portal - Folder Management Interface

- Use the **Folder Management Interface** to create and organise folders.

- The panel shows a **hierarchical structure** with the number of files in each folder and allows easy navigation between your data and tasks.



**Figure 47.** Folder management interface on the Neuroscience Gateway (NSG) platform.

After logging in, users can create, edit, or delete project folders that organise simulation files and results.

In this example, the folder “**Neuron**” contains resources for BrainCell astro-neuronal simulations.

The **Folder Details** section displays the folder label and description, while the side navigation panel lists all user directories (e.g., *NeuroAstro*, *Data*, *Tasks*). These folders are used to manage uploaded models, organise input data, and prepare job submissions for execution on the NSG high-performance computing system hosted by the San Diego Supercomputer Center (SDSC).

The Folder Management Interface allows users to organise and manage their computational neuroscience projects within the NSG Portal. **Folders Navigation Panel:** Displays hierarchical folder structure, Shows folder contents with item counts in parentheses, Currently selected: "Used" and "NeuronAstro" folders, Subfolders shown: Data (4 items) and Tasks (7 items).

### 3. NSG Task panel

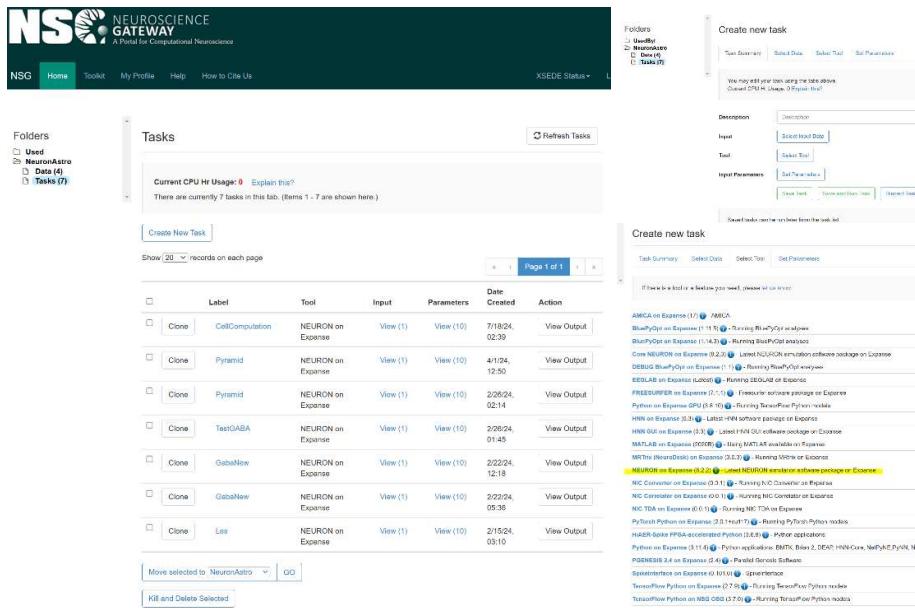
#### Configure and Run a Task

- In the **Task Management Interface**, click **Create New Task**.
- Provide:
  - Tool type:** *NEURON on Expanse* (recommended).
  - Main input file:** usually `runner.hoc`.
  - Adjust **parameters** for computation resources:
    - Number of Nodes = 1**
    - MPI Tasks per Node = 1**
    - OpenMP Threads per Task = 1**
    - Cores per Node = 1**

(⚠ BrainCell currently doesn't support parallel computing — using single-core reduces queue time.)
- Submit the task and wait for it to run (queue times can vary from **30 minutes to 15 hours**).

## ⚠ Note:

- “Core NEURON on Expanse” may cause errors.
- Sometimes NSG submissions are temporarily unavailable; check for “NSG submissions temporarily halted” notices.



**Figure 48.** Task management and job submission interface on the Neuroscience Gateway (NSG).

The **Tasks** panel lists all simulation jobs currently stored or running on the NSG platform. Each task entry displays the model label, computational tool used (e.g., *NEURON on Expanse*), input files, parameters, and submission date.

Users can view outputs, clone existing tasks, or delete completed ones directly from this interface.

The right-hand panels show the **Create New Task** window, where users specify input data, parameter files, and the simulation tool to be executed.

The **Task Summary** and **Set Parameters** tabs provide detailed configuration options, including Python or NEURON execution modes and queue settings.

This workflow enables BrainCell users to run large-scale astrocyte and neuron simulations on NSG’s high-performance computing infrastructure with minimal local setup.

## Linux MOD File Requirements

- Linux uses `libnrnmech.so` instead of Windows’ `nrnmech.dll`.
- Combine MOD files from:
  - Common/
  - Astrocyte/ or Neuron/ (depending on your model)
- Place all MOD files in the **same folder** as your main HOC file.
- NSG will automatically compile them into `libnrnmech.so`.

## 4. Necessary Files

Linux systems require MOD files instead of "nrnmech.dll". Combine MOD files from "Common" and either "Astrocyte" or "Neuron" folders into the same folder as your HOC file. MOD files will compile automatically into "libnrnmech.so".

**1. Parameter Settings:**

Set the following parameters to 1, as BrainCell does not yet support parallel computing:

Number of Nodes

MPI Tasks per Node

OpenMP Threads per Task

Cores per Node Reducing requested resources may decrease queue time.

**2. Additional Observations:**

"NEURON on Expanse" is sufficient for running BrainCell files that require NEURON and Python.

# BrainCell Testing Suite

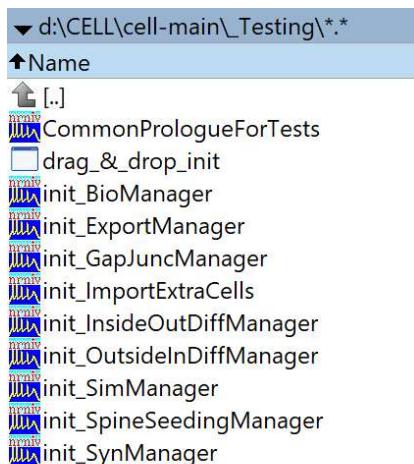
BrainCell includes an integrated **Testing Suite** designed to verify that all core modules are functioning correctly before launching full-scale simulations.

This utility is located in the directory:

```
...\\cell-main\\_Testing\\
```

Each script in this folder initialises or tests a specific BrainCell module (e.g., geometry setup, diffusion, synaptic transmission, or import/export functions).

Running these tests ensures that all dependencies, compiled mechanisms, and inter-module connections are correctly configured—especially after installation or system updates.



**Figure 49.** *BrainCell Testing Suite directory structure.*

The `_Testing` folder contains a collection of initialisation and diagnostic scripts for verifying BrainCell functionality.

Each `init_*` file corresponds to a specific subsystem:

- **init\_BioManager:** checks biophysical mechanism loading.
- **init\_SimManager:** tests the main simulation manager.
- **init\_SpineSeedingManager / init\_SynManager:** validate spine and synapse generation modules.
- **init\_ImportExtraCells / init\_ExportManager:** confirm the correct import/export of multi-cell configurations.
- **init\_GapJuncManager / init\_InsideOutDiffManager / init\_OutsideInDiffManager:** test diffusion and gap-junction connectivity functions.

Users can execute individual scripts to diagnose module-specific problems or run the suite sequentially to perform a complete pre-simulation system check.

## Running Tests

## **Method 1 — Quick Drag & Drop**

1. Navigate to ...\\cell-main\\\_Testing\\
2. Drag any `init_*.hoc` file onto `drag_&_drop_init.bat`
3. The selected test will run automatically.

## **Method 2 — Direct Execution**

1. Go to ...\\cell-main\\\_Testing\\
2. Double-click the desired `init_*.hoc` file.

After running:

- Check **output logs** for any errors.
- Verify that each module runs as expected.

↳ **Tip:** Each test is independent and automatically calls `CommonPrologueForTests` when required — you don't need to run it manually.

---

This section ensures users can **test, prepare, and run BrainCell models on the NSG supercomputing platform** efficiently, reducing compatibility issues and simplifying large-scale simulations.

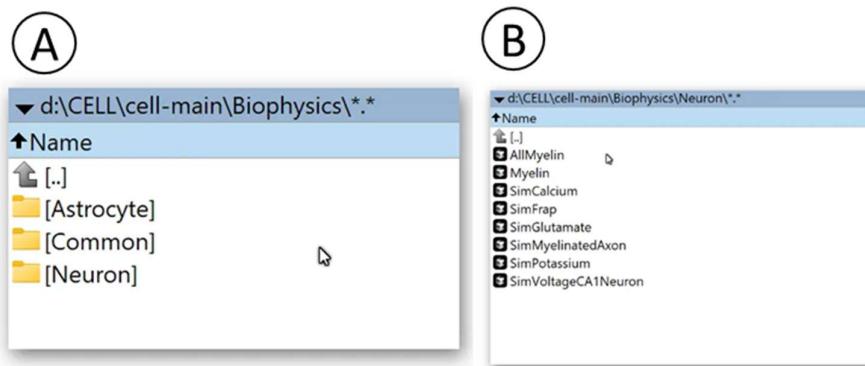
# Creating and Editing Biophysical Mechanisms

BrainCell lets you design and customise **biophysical mechanisms** for your cell models.

You can either use the **built-in Biophysical Mechanisms Manager** or create and edit mechanisms directly by working with **JSON files**.

Editing JSON files gives you a **simple, text-based and visual way** to define mechanisms using any text editor.

## Where the JSON Files Are Located



**Figure 50.** Biophysics directory structure in BrainCell.

(A) The main **Biophysics** folder (`...\\cell-main\\Biophysics\\`) contains three subdirectories: **Astrocyte**, **Neuron**, and **Common**, which store mechanism definitions and scripts shared between modules.

(B) The **Neuron** subdirectory includes functional modules for simulating different neuronal biophysical mechanisms, such as **SimCalcium**, **SimPotassium**, **SimGlutamate**, and **SimMyelinatedAxon**.

These components define the ionic dynamics, synaptic mechanisms, and myelination properties used during neuron-mode simulations, allowing BrainCell to reproduce realistic electrophysiological and biochemical behaviour.

Biophysical mechanism files are stored in:

`...\\cell-main\\Biophysics\\`

Inside this folder, you'll find three subdirectories:

- **Neuron** — mechanisms specific to neurons
- **Astrocyte** — mechanisms specific to astrocytes
- **Common** — mechanisms shared between cell types

## Creating a New JSON Mechanism

1. Open the correct subdirectory (Neuron, Astrocyte, or Common).
  2. Create a **new text file** with a clear name (e.g., MyNewMechanism.json).
  3. Open the file in any text editor (e.g., Notepad++, VS Code, Sublime).
- 

## Editing an Existing JSON Mechanism

1. Open the JSON file you want to modify.
2. Define or change the mechanism using standard **JSON syntax**.
3. Each mechanism should include **type, equations, and parameters**.

### Example JSON file:

```
{  
  "MechanismName": {  
    "type": "voltage_gated_channel",  
    "ion": "na",  
    "equations": {  
      "alpha_m": "0.1 * (v + 40) / (1 - exp(-(v + 40) / 10))",  
      "beta_m": "4 * exp(-(v + 65) / 18)"  
    },  
    "parameters": {  
      "gmax": 120,  
      "e_rev": 50  
    }  
  }  
}
```

---

## Available Built-In Mechanisms

Examples of predefined mechanisms include:

- AllMyelin
- Myelin
- SimCalcium
- SimFrap
- SimGlutamate
- SimMyelinatedAxon
- SimPotassium
- SimVoltageCA1Neuron

You can **modify these templates** or create completely new mechanisms.

---

## Best Practices

- Use clear names** for files and mechanisms (e.g., CaPump\_Modified.json).
- Add comments** using // (single line) or /\* \*/ (multi-line) to explain complex equations.
- Validate JSON syntax** with an online JSON checker before saving.
- Back up original files** before making big changes.
- Ensure scientific accuracy** — double-check ion types, equations, and parameters.

## Applying Your Changes

1. Save your edited or new JSON file in the correct subdirectory.
2. Restart BrainCell or use the **biophysical mechanisms reload option**.

Your new or updated mechanisms will now be available in the **Biophysics Manager** for use in your models.

### ⚡ Tip:

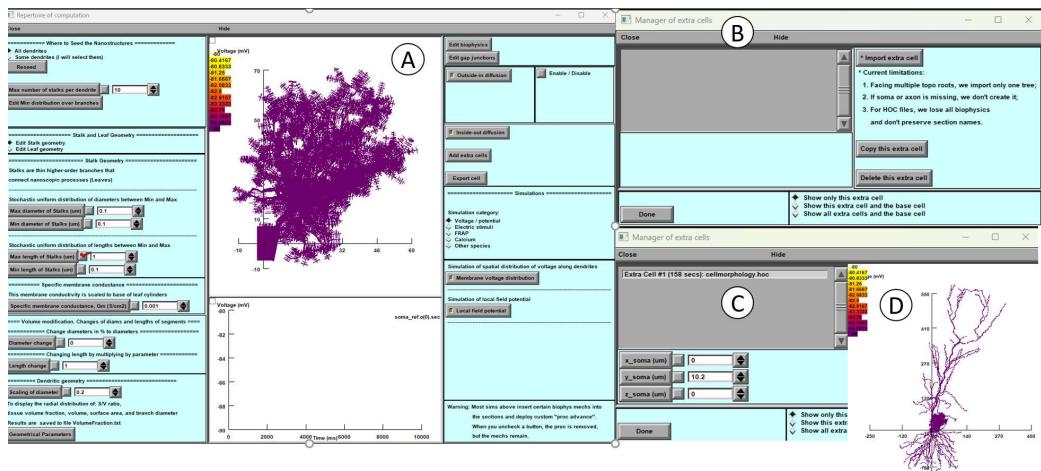
Editing mechanisms via JSON is great for **advanced users** who want fine control over ion channels, pumps, or custom experimental conditions.

It also allows easy sharing of mechanism files between collaborators.

## Multi-Cell Simulations and Shared Extracellular Space

BrainCell now supports simultaneous simulation of multiple cells within a **common extracellular environment**.

In this mode, users can import several different cell morphologies—either neuronal or astrocytic—and observe how their combined activity dynamically alters extracellular ion concentrations and other parameters.



**Figure 51.** Multi-cell simulation interface in BrainCell.

**(A) Main Control Panel (Repertoire of Computation):** the central interface for configuring simulations. Pressing **Add Extra Cells** opens the Manager of Extra Cells window.

**(B) Manager of Extra Cells:** panel for importing and managing additional cell morphologies within a shared extracellular environment.

**(C) Extra Cell Information Panel:** displays details of each imported cell and allows positioning in 3D space by adjusting X, Y, and Z coordinates.

**(D) 3D Geometry Display:** shows all loaded cells simultaneously, illustrating their spatial arrangement and the

resulting extracellular environment. Multiple cells can be added to simulate interactions affecting extracellular ion concentrations and potentials.

---

## Main Window Overview

### Panel A — Repertoire of Computation

This is the central BrainCell control panel.

It includes the **Add Extra Cells** button, which opens the **Manager of Extra Cells** window (Panel B). From here, users can import additional cell morphologies into the same simulation environment.

---

### Panel B — Manager of Extra Cells

When first opened, this panel is empty because the **primary (base) cell** is already loaded by default. To add another cell, click **Import Extra Cell**.

A file-browser window will appear, allowing you to select a morphology file from your computer. The selected file can represent any compatible geometry, including one previously generated in BrainCell.

After choosing the file, allow a few moments for the cell to load fully. Once loaded, it will appear simultaneously in:

- **Panel C** — the *Extra Cell Information Panel*, and
  - **Panel D** — the main 3-D geometry display.
- 

### Panel C — Extra Cell Information

This panel lists the imported cell (e.g., *Extra Cell #1: cellmorphology.hoc*) and provides tools for positioning it in three-dimensional space.

You can shift the cell by adjusting its **X, Y, Z coordinates**, effectively changing the position of its soma or geometric centre.

This feature makes it possible to arrange multiple cells spatially to create realistic multi-cell assemblies.

---

### Panel D — Geometry Display

The imported cell is displayed in the global 3-D view alongside the base cell.

Each additional cell retains its own colour and voltage or concentration map, allowing you to visualise spatial interactions between them.

---

### Performance Note

Users can import **as many cells as desired**, but each additional cell increases computational complexity and simulation time.

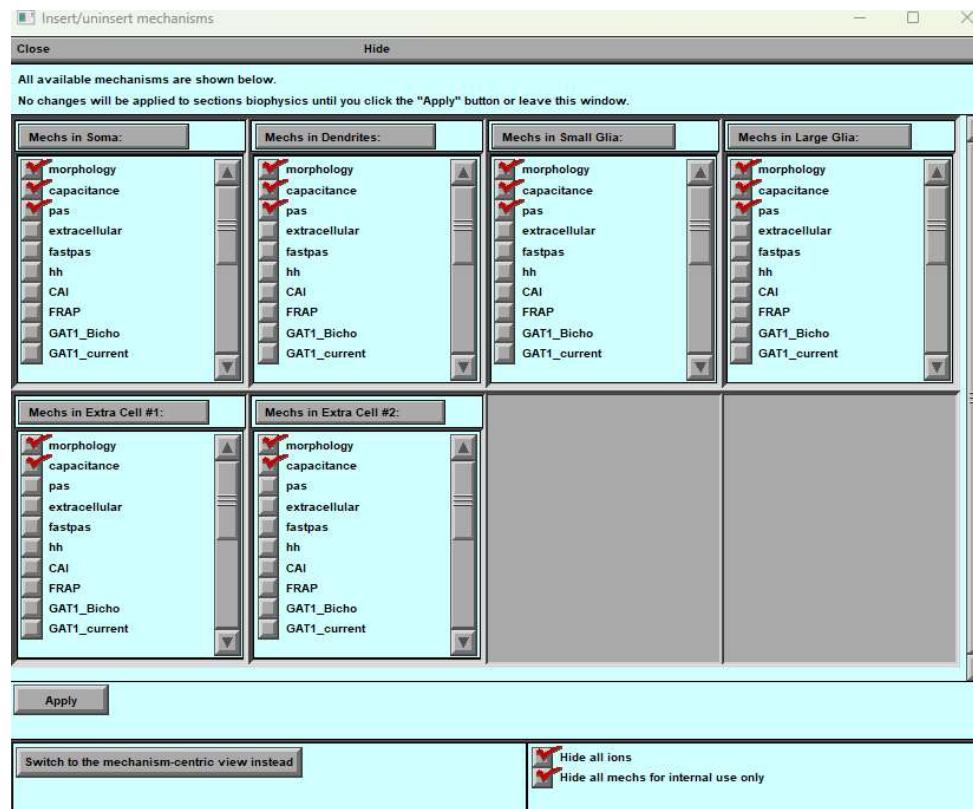
For large assemblies, ensure that hardware performance and memory capacity are sufficient to handle extended calculations.

## Multi-Cell Biophysical Configuration

This section introduces the **multi-cell mechanism editor** in BrainCell, which allows users to manage and synchronise **biophysical properties** across multiple loaded cells.

Each imported cell (Cell 1, Cell 2, etc.) can have its own active and passive mechanisms—such as ion channels, transporters, or diffusion systems—applied to soma, dendrites, or glial compartments.

The panel provides a unified interface for selecting, comparing, and modifying these mechanisms, ensuring consistent physiological behaviour in multi-cell simulations.



**Figure 51.** Insert/Uninsert Mechanisms Panel in multi-cell simulation mode.

This window allows users to manage **biophysical mechanisms** for each cell in a multi-cell simulation. Each section lists the currently available or assigned mechanisms within different cell compartments—**soma**, **dendrites**, **small glia**, and **large glia**—as well as any **additional imported cells** (e.g., *Extra Cell #1*, *Extra Cell #2*).

Users can enable or disable specific modules such as **morphology**, **capacitance**, **passive properties (pas)**, **extracellular space**, **FRAP**, **CAI**, **GAT transporters**, and others.

No changes take effect until the **Apply** button is pressed.

The lower control panel includes filters to **hide ionic species** or **exclude mechanisms reserved for internal use**, ensuring clarity when configuring multiple interacting cells.

This tool enables consistent assignment of mechanisms across all imported cells, supporting complex network simulations in a shared extracellular environment.

## *Coupled Dynamics in a Shared Extracellular Space*

This section describes how BrainCell models the **interaction of multiple cells through a common extracellular environment**.

During a simulation, each cell continuously alters the local ionic composition around it. These extracellular ions diffuse through space, influencing neighbouring cells and creating realistic, bidirectional coupling between neuronal and glial populations.

To begin this type of simulation, open the main BrainCell control panel and press **Inside–Out Diffusion**.

Two windows will appear — **Panel A** and **Panel C** — while **Panel B** (an alternative simulation panel) remains hidden by default.

Panel B offers a more advanced interface and is generally **recommended for routine use**.

On Panels A and C, users can configure the **extracellular environment**, including:

- Selecting which **ions** will be tracked and visualised,
- Adjusting **diffusion coefficients** and concentration ranges, and
- Defining **boundary conditions** for the shared medium.

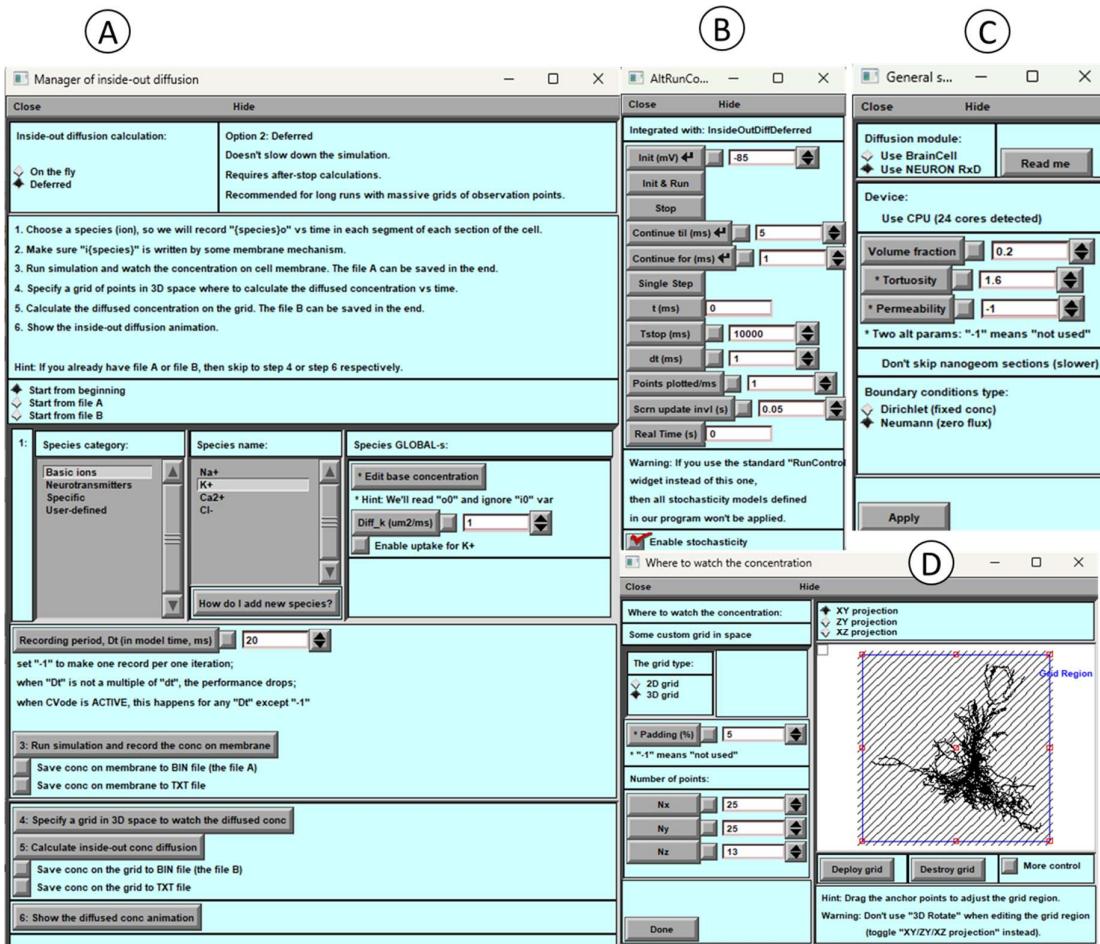
However, for the extracellular space to function properly, it must be **spatially constrained** and **discretised into a computational grid**.

To create this grid, open **Panel A** and click “**Specify a Grid...**”, which opens **Panel D**.

Panel D allows you to define the **grid resolution** and **observation projection**.

Once these parameters are set, return to **Panel B** and press **Compute** to initialise the simulation of extracellular diffusion.

This process establishes a shared dynamic environment in which all loaded cells exchange ions and signals through a common extracellular domain, enabling fully coupled multi-cell simulations.



**Figure 52.** Inside–Out Diffusion control panels for multi-cell extracellular simulations in BrainCell.

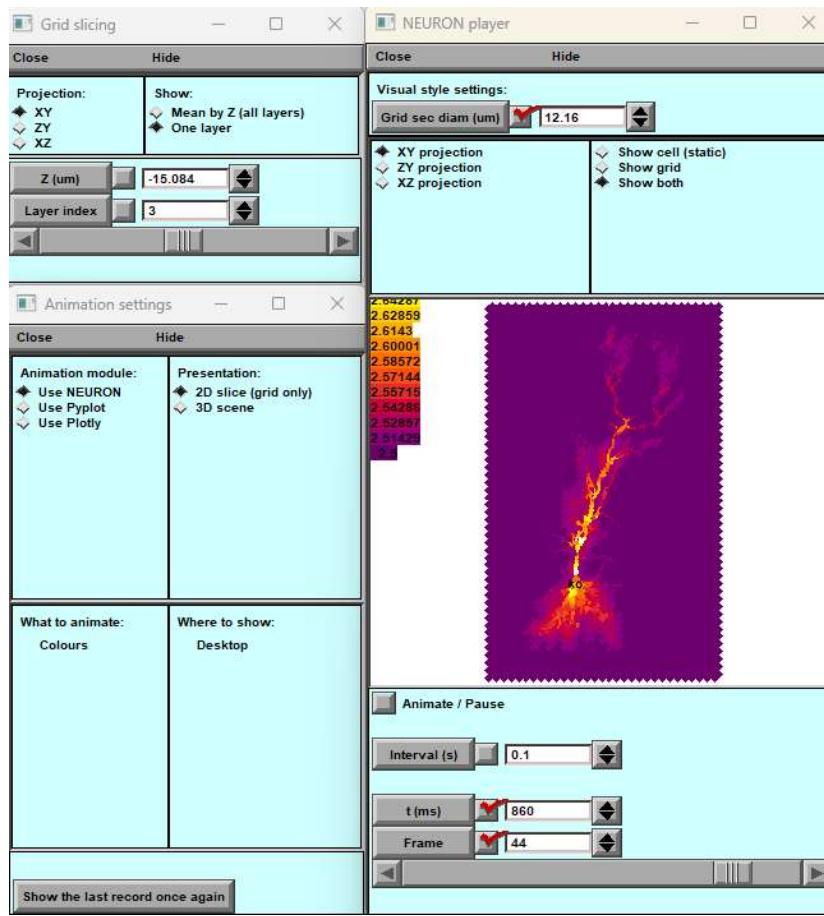
**(A) Manager of Inside–Out Diffusion:** main interface for configuring extracellular ion dynamics. Users can select ionic species (Na<sup>+</sup>, K<sup>+</sup>, Ca<sup>2+</sup>, Cl<sup>-</sup>, etc.), set diffusion coefficients, enable uptake mechanisms, and choose between “on-the-fly” or “deferred” computation modes.

**(B) Alternative Run Controller:** advanced simulation manager that integrates directly with the diffusion module. It defines simulation duration, time step, and update intervals, and provides options for real-time visualisation and stochastic behaviour control.

**(C) General Settings Panel:** specifies the numerical and physical parameters of the diffusion environment, including volume fraction, tortuosity, and boundary conditions (Dirichlet or Neumann). Users can also select the computational device (CPU or NEURON RxD module) and performance options.

**(D) Grid Definition Panel:** used to create and visualise the 3D computational grid where extracellular diffusion is calculated. The panel allows users to define grid dimensions (Nx, Ny, Nz), spatial padding, and projection type (XY, YZ, or XZ), ensuring precise spatial control of the simulated extracellular space.

Together, these panels enable fully coupled simulation of ionic diffusion and cellular interactions within a shared extracellular environment.



**Figure 53.** *Visualisation and animation of extracellular diffusion dynamics.*

This set of panels displays tools for analysing and animating the simulated diffusion processes in the shared extracellular space.

- **Grid Slicing Panel (top left):** defines spatial projection (XY, YZ, or XZ planes) and allows users to select individual or averaged layers for visualisation.
  - **NEURON Player Panel (right):** shows the diffusion results as colour-mapped spatial distributions, with adjustable grid section diameter and projection type. The user can display the grid, the cell morphology, or both simultaneously.
  - **Animation Settings Panel (bottom left):** configures visualisation modules for playback using NEURON, PyPlot, or Plotly. Users can choose between 2D grid-slice and full 3D presentations.
- Playback controls at the bottom of the NEURON Player window enable frame-by-frame animation, setting the time interval, frame number, and total duration of the recording.

At this simulation stage, BrainCell provides several advanced options for managing and resuming extracellular diffusion computations.

If a simulation is paused or interrupted, it can be **continued from the last saved state** using automatically generated checkpoint files that record time-step data and concentration maps.

Simulation results can be **saved in multiple formats**, including **BIN**, **TXT**, and **CSV**.

The **BIN format** is recommended for large datasets, as it allows rapid reloading and continuation of calculations without data loss.

The **TXT** and **CSV** formats are useful for further analysis, visualisation, or export to external tools such as MATLAB or Python.

All saved files can later be **reopened within BrainCell**, enabling users to replay animations, inspect concentration gradients, and compare different diffusion scenarios.

These capabilities ensure flexible control, reproducibility, and efficient handling of long-running multi-cell simulations.

**Note:** At this stage, exporting all cells simultaneously is **not supported**, and therefore **multi-cell diffusion simulations cannot yet be executed on external or free HPC clusters**. All computations must be performed locally within the BrainCell environment.