Journal of Visualized Experiments

3D Modeling of Dendritic Spines with Synaptic Plasticity

--Manuscript Draft--

Article Type:	Invited Methods Article - JoVE Produced Video
Manuscript Number:	JoVE60896R2
Full Title:	3D Modeling of Dendritic Spines with Synaptic Plasticity
Section/Category:	JoVE Neuroscience
Keywords:	AMPA receptor trafficking; reaction-diffusion; synaptic plasticity, dendritic spines; computational modeling; long-term potentiation; long-term depression; heterosynaptic plasticity
Corresponding Author:	Fabio Simoes de Souza Universidade Federal do ABC Sao Bernardo do Campo, Sao Paulo BRAZIL
Corresponding Author's Institution:	Universidade Federal do ABC
Corresponding Author E-Mail:	fabio.souza@ufabc.edu.br
Order of Authors:	Gabriela Antunes
	Fabio Simoes de Souza
Additional Information:	
Question	**Response**
Please indicate whether this article will be Standard Access or Open Access.	Standard Access (US$2,400)
Please indicate the **city, state/province, and country** where this article will be **filmed**. Please do not use abbreviations.	Denver, Colorado/USA

Dear Dr. Aaron Berard,

We are resubmitting the invited article ‘Modeling of Synaptic Plasticity in Three Dimensional Dendritic Spines’. Here, we described our methods to develop a three dimensional model of a dendritic segment with dendritic spines for modeling synaptic plasticity. The constructed mesh can be used for computational modeling of AMPA receptor trafficking in long-term synaptic plasticity using the software Blender with CellBlender and MCell.

In this revised version, we addressed the comments to increase the clarity of the protocol; we reduced the highlighted section and reduced the number of pages of the protocol. We included three supplementary figures in the additional supplementary file.

With my best regards,

Dr. Fabio M. Simoes de Souza

TITLE:

3D Modeling of Dendritic Spines with Synaptic Plasticity

AUTHORS AND AFFILIATIONS:

Gabriela Antunes[1,*], Fabio M. Simoes de Souza[1,*]

[1]Center for Mathematics, Computation and Cognition, Federal University of ABC, São Bernardo do Campo, SP, Brazil

*These authors contributed equally

Corresponding Author:

Fabio M. Simoes de Souza (fabio.souza@ufabc.edu.br)

Email Address of Co-authors:

Gabriela Antunes (gabrian@protonmail.com)

KEYWORDS:

AMPA receptor trafficking, reaction-diffusion, synaptic plasticity, dendritic spines, computational modeling, long-term potentiation, long-term depression, heterosynaptic plasticity

SUMMARY:

The protocol develops a three-dimensional (3D) model of a dendritic segment with dendritic spines for modeling synaptic plasticity. The constructed mesh can be used for computational modeling of AMPA receptor trafficking in the long-term synaptic plasticity using the software program Blender with CellBlender and MCell.

ABSTRACT:

Computational modeling of diffusion and reaction of chemical species in a three-dimensional (3D) geometry is a fundamental method to understand the mechanisms of synaptic plasticity in dendritic spines. In this protocol, the detailed 3D structure of the dendrites and dendritic spines is modeled with meshes on the software Blender with CellBlender. The synaptic and extrasynaptic regions are defined on the mesh. Next, the synaptic receptor and synaptic anchor molecules are defined with their diffusion constants. Finally, the chemical reactions between synaptic receptors and synaptic anchors are included and the computational model is solved numerically with the software MCell. This method describes the spatiotemporal path of every single molecule in a 3D geometrical structure. Thus, it is very useful to study the trafficking of synaptic receptors in and out of the dendritic spines during the occurrence of synaptic plasticity. A limitation of this method is that the high number of molecules slows the speed of the simulations. Modeling of dendritic spines with this method allows the study of homosynaptic potentiation and depression within single spines and heterosynaptic plasticity between neighbor dendritic spines.

INTRODUCTION:

Synaptic plasticity has been associated with learning and memory[1]. Synaptic plasticity, such as long-term potentiation (LTP) and long-term depression (LTD), is associated respectively with the insertion and removal of AMPA receptors (AMPARs) in and out of the synaptic membrane[2]. The AMPAR synapses are located on top of the small volume structures called dendritic spines[3]. Each spine contains a protein dense region in the postsynaptic membrane called the postsynaptic density (PSD). Anchor proteins at the PSD trap AMPARs in the synaptic region. There are few copies of AMPARs within a single synapse and the trafficking and reaction of AMPARs with other species in dendritic spines is a stochastic process[2,4]. There are several compartmental models of synaptic receptor trafficking at dendritic spines[5–8]. However, there is a lack of stochastic computational models of the trafficking of AMPARs associated with synaptic plasticity at the 3D structures of the dendrites and their dendritic spines.

Computational modeling is a useful tool to investigate the mechanisms underlying the dynamics of complex systems such as the reaction-diffusion of AMPARs in dendritic spines during the occurrence of synaptic plasticity[9–12]. The model can be used to visualize complex scenarios, varying sensitive parameters and making important predictions in scientific conditions involving many variables that are difficult or impossible to control experimentally in an in vivo system[12,13]. Defining the level of detail of a computational model is a fundamental step in obtaining accurate information about the modeled phenomenon. An ideal computational model is a delicate balance between complexity and simplicity to capture the essential characteristics of the natural phenomena without being computationally prohibitive. Computational models that are too detailed can be expensive to compute. On the other hand, systems that are poorly detailed can lack the fundamental components that are essential to capture the dynamics of the phenomenon. Although 3D modeling of dendritic spines is computationally more expensive than 2D and 1D, there are conditions, such as in complex systems with many nonlinear variables reacting and diffusing in time and 3D space, for which modeling at a 3D level is essential to obtain insights about the functioning of the system. Further, the complexity can be reduced carefully to preserve the essential characteristics of a lower-dimensional model.

In a stochastic system with few copies of a given species within a small volume, the average dynamics of the system deviates from the average dynamics of a large population. In this case, the stochastic computational modeling of reaction-diffusing particles is required. This work introduces a method for stochastic modeling reaction-diffusion of a few copies of AMPARs in 3D dendritic spines. The purpose of this method is to develop a 3D computational model of a dendritic segment with dendritic spines and their synapses for modeling synaptic plasticity.

The method uses the software programs MCell to solve the model numerically, Blender for constructing 3D meshes, and CellBlender to create and visualize the MCell simulations, including the spatiotemporal reaction-diffusion of molecules in 3D meshes[14–16]. Blender is a suite for the creation of meshes and CellBlender is an add-on for the base software Blender. MCell is a Monte Carlo simulator for the reaction-diffusion of single molecules[17].

The rationale behind the use of this method consists of modeling synaptic plasticity to achieve a better understanding of this phenomenon in the microphysiological environment of the dendritic

spines[14]. Particularly, this method allows the simulation of homosynaptic potentiation, homosynaptic depression, and heterosynaptic plasticity between dendritic spines[14].

The features of this method include modeling the 3D geometric structure of the dendrite and its synapses, the diffusion by random walk, and the chemical reactions of the molecules involved with synaptic plasticity. This method provides the advantage of creating rich environments to test hypotheses and make predictions about the functioning of a complex nonlinear system with a large number of variables. In addition, this method can be applied not only for studying synaptic plasticity but also for studying stochastic reaction-diffusion of molecules in 3D mesh structures in general.

Alternatively, 3D meshes of dendritic structures can be constructed directly in Blender from electron microscope serial reconstructions[18]. Although meshes based on serial reconstructions provide 3D structures, access to the experimental data is not always available. Thus, the construction of meshes adapted from basic geometric structures, as described in the present protocol, provides flexibility to develop customized dendritic segments with dendritic spines.

Another alternative computational method is the bulk simulation of well-mixed reactions in a regular volume[9–11,19–22,22]. The bulk simulations are very efficient in solving the reactions of many species within a single well-mixed volume[23], but the bulk approach is extremely slow to solve the reaction-diffusion of molecules within many well-mixed voxels in a high-resolution 3D mesh. On the other hand, the present method using MCell simulations of reaction-diffusion of individual particles works efficiently in high-resolution 3D meshes[15].

Before using this method, one should ask whether the phenomenon studied requires a stochastic reaction-diffusion approach in a 3D mesh. If the phenomenon has few copies (less than 1,000) of at least one of the reacting species diffusing in a complex geometric structure with small volume compartments such as dendritic spines, then stochastic modeling of reaction-diffusion in 3D meshes is appropriate for the application.

There are several steps required to construct a 3D computational model of a dendritic segment containing dendritic spines with synaptic plasticity. The main steps are the installation of the proper software for the construction of the model, the construction of a single dendritic spine to be used as a template to create multiple spines, and the creation a dendritic segment that is connected with multiple dendritic spines. The step for modeling synaptic plasticity consists of inserting anchors in the PSD region and AMPARs in the dendritic segment and dendritic spines. Then, kinetic reactions between the anchors located in the PSD and AMPARs are defined to produce complexed anchor-AMPAR species that trap the AMPARs in the synaptic region. Respectively, the increase and decrease of the affinity between the anchors and the synaptic AMPARs create the process of LTP and LTD.

PROTOCOL:

NOTE: Please see the **Supplementary file 1** for the glossary of terms used in this protocol.

1. Install Blender, CellBlender, and MCell

NOTE: This protocol requires installation of MCell, Blender, and Cell Blender.

1.1. Download and install the software on the MCell homepage (https://mcell.org/tutorials_iframe.html). Go to downloads on the top of the page and then follow the step-by-step instructions to download and install the software in the environment of choice (e.g., Linux, Mac OSX, or Windows).

NOTE: All computational models and simulations described in this protocol were tested on a CellBlender 1.1 bundle that includes Blender 2.78 with MCell 3.4 and CellBlender 1.1. It worked also on Blender 2.79b. All these software programs are open access and do not require reprint permission to be used. The instructions for the construction and simulation of the model can change slightly from one version to another. Parts of this protocol have been adapted from Czech et al.[16].

2. Create a single dendritic spine

NOTE: This procedure creates a mesh of a single dendritic spine with a spine head and a spine neck using a modified sphere.

2.1. Set up Blender 3D view at the Main Panel.

2.1.1. Open Blender with CellBlender already installed. Press **5** on the keypad to change from **Perspective** to **Orthogonal** view and press **1** to change to front view. The perspective view has depth, but this is not needed now. Changing from **Perspective** to **Orthogonal** view allows better visualization of the mesh. Press **Shift+C** to center the cursor (**Figure 1A**).

2.2. Create the spine head.

2.2.1. Press **Shift+A** to open the mesh palette. Select the **Mesh** and then select **UV Sphere**. A UV sphere is a mesh mapped to the 3D surface of a sphere. The UV sphere represents the spherical head of a mushroom dendritic spine. The software assumes the units of the UV sphere are micrometers.

2.2.2. Change the parameters on the **Add UV Sphere** panel. Change **Size** to ***0.25*** and **Rings** to ***32*** (**Figure 1B**). Press **+** or **–** on the keypad to respectively zoom in and zoom out of the visualization of the mesh. Alternatively, use the scroll button in the mouse to zoom in and out (**Figure 1C**).

NOTE: The **parameter size** scales the size of the original sphere and the **parameter rings** define the resolution of the mesh.

2.3. Make the top of the head flat.

2.3.1. Press **Tab** to switch **Blender** from the **Object Mode**, the **standard object interaction mode**, to **Edit Mode**. Work in **Edit Mode** for modifying the components of an existing mesh.

2.3.2. Once the created mesh has been automatically selected, press **a** to deselect the created mesh. Press **z** to make the mesh transparent, which helps in visualizing the parts that will be edited. Zoom in on the mesh. Press **b** to select the top ¾ of the sphere with the mouse (**Figure 2A**). Press **delete**, select **vertices**, and **enter** to remove the vertices (**Figure 2B**).

2.3.3. Press **b** and select the top. Press **e**, **s**, **0**, and **enter** to seal the top with the vertices still selected. Move the blue arrow down to align to the top of the spine head (**Figure 2C**). Press **z** to change to solid view (**Figure 3A**). Press **7** to change to top view.

NOTE: The top of the sphere is made flat to model the PSD region of the spine head.

2.4. To increase the mesh resolution at the top of the spine first select **Tool and Knife**. Cut a circle with the knife around the center of the top (**Figure 3B**). Select **Tool** and **Loop Cut and Slide**. Repeat this step four times to create four concentric circles around the center of the top (**Figure 3C**).

NOTE: The concentric circles are used to add new voxels that will increase the resolution of the PSD.

2.5. Create the spine neck.

2.5.1. Press ***a*** to deselect the mesh. Press ***1*** to change to the front view. Press ***z*** to make the mesh transparent. Press ***b*** and then select the bottom of the mesh (**Figure 4A**). Press ***delete*** and **vertices** (**Figure 4B**). Press ***b*** and select the bottom of the mesh (**Figure 4C**). Press ***e*** and ***z***, ***-0.45*** to create an extrusion (**Figure 4D**).

NOTE: This creates an extrusion to the z axis position at -0.45 µm. Press ***a*** to deselect the whole mesh.

2.5.2. Press ***b*** and select the bottom of the neck. Press ***e***, ***s***, and ***0*** to seal the bottom (**Figure 4E**). Press ***a*** to select the whole mesh.

2.6. Make the mesh compatible with MCell.

2.6.1. Press ***Crtl+T*** to triangulate the mesh. The mesh is transformed into a set of interconnected triangles. This is a required procedure to make the mesh compatible with MCell. Select **Tool** and **Remove Doubles**. Use the **Remove Doubles** tools to remove duplicated vertices, if any, that have the same coordinates or are very close to each other, to make the mesh compatible with MCell.

NOTE: Double superposed vertices may have been accidentally created during the process of

mesh creation and editing.

2.6.2. Select **Model Objects** on the **CellBlender panel**. Change the name of the **Active Object** to **spine** and press ***+*** to create the object spine. On the **CellBlender panel**, select **Mesh Analysis** and then click on **Analyze Mesh** (**Figure 4F**). This procedure will analyze the properties of the created mesh, including the number of vertices, edges, faces, surface area, volume, and mesh topology.

NOTE: The analysis will print the information in the **Mesh Analysis Panel** and it should be **Watertight**, **Manifold**, and **Outward-Facing Normals**. This step is required to ensure that the mesh will work on MCell. Otherwise, a step was probably missed. In this case, ***delete*** the mesh and start from step 2.1 again.

2.6.3. Press ***z*** to visualize the solid view of the spine. Press **File and Save** to have a copy of your blender file with the spine on the disk.

NOTE: The dimensions (i.e., length, diameter, size) of the meshes are in micrometers. See the glossary for the meaning of each keyboard shortcut.

3. Creating a dendrite with multiple spines

3.1. Generate a spine as described previously in sections 2.1–2.6. Press ***a*** to deselect the spine. Type ***Shift+C*** to center the cursor.

3.2. Create a dendrite. Press ***Shift+A*** to open the mesh palette. Select **Mesh** and then **Cylinder**. Change the parameters on the **Add Cylinder** menu: **Radius** = ***0.3 µm***, **Depth** = ***2 µm***. Press ***Enter***.

NOTE: The parameters radius and depth are defined according to the geometric characteristics of the dendrite.

3.3. Insert a spine in the dendrite.

3.3.1. Press ***r*** and type ***90*** to rotate the cylinder 90° (**Figure 5A**). Use the blue arrow to drag the cylinder down to the bottom of the spine. Press ***3*** on the keypad to have a front view of the cylinder.

3.3.2. Press ***z*** to make the mesh transparent. Use the mouse to move the blue normal arrow of the cylinder downwards to move the base of the spine to the interior of the cylinder (**Figure 5B**). Press ***a*** to deselect all objects.

3.3.3. Use the right button of the mouse to select the dendrite (**Figure 5C**). Select **Modifier** on Blender panel (**Figure 5D**), select **Add Modifier**. Then select **Boolean**, select **Operation Union**, and select **Object spine**. Press **Apply** to create a joint mesh of the dendrite and the spine (**Figure 5E**). This operation creates a new mesh merging two meshes into a single one.

NOTE: The new mesh will be the combined dendrite and spine. The isolated dendrite disappears when the different meshes are combined, but the isolated spine mesh remains overlapped with the new mesh and is used to generate multiple copies of the same spine. Delete all the isolated spines after finishing the mesh. It is critical to have a complete overlap between the spine neck and the dendrite, otherwise, the mesh will not be watertight.

3.4. Set the dendrite object into the CellBlender environment.

3.4.1. Press ***a*** to deselect the meshes. Right-click in the dendrite with the mouse to select the dendrite only. Select **CellBlender**, **Model Objects**, and change **Active Object** to ***Dendrite*** and press ***+*** to create the Dendrite object.

3.5. Insert new spines in the dendrite.

3.5.1. Press ***1*** to change to the side view of the cylinder. Use the mouse to select the mesh of the isolated spine. To insert more spines, follow step 3.3, changing the position and the angle to insert each one to obtain a physiological distribution.

3.6. Make the mesh compatible with MCell. To do so, press ***Tab*** to go to **edit mode**. Press ***a*** to select the whole mesh. Press ***Crtl+T*** to triangulate the mesh. Select **Tool** on the **Blender** panel and select **Remove Doubles**.

3.7. Stylize the meshes.

3.7.1. Smooth the mesh. Press ***Tab*** to change to **object mode**. Select **Tool** on the **Blender** panel and select **Smooth**. Select **CellBlender**, **Model Objects**, and select **Add a Material**.

3.7.2. Make the mesh transparent by selecting **Object Transparent** and **Material Transparent**. Change **alpha** to ***0.5*** and **enter** to make the mesh partially transparent. Press ***z*** to change to solid view.

3.8. Confirm whether the mesh is still compatible with MCell. To do so, select **Mesh Analysis** on the **CellBlender panel** to make sure that the mesh is still **watertight**, **manifold mesh**, and **outward-facing normal**.

3.9. **Save** the blender file as **dendrite_with_spines.blend**.

4. Define surface regions

NOTE: This procedure creates the surface regions of the mesh that later will be used to set up how the regions interact with the molecules.

4.1. Open the file **dendrite_with_spines** in the **Blender environment**. To do so, select **File**, **Open**, **dendrite_with_spines.blend**, and **Open Blender File**.

4.2. Prepare the mesh for defining the surface regions. To do so, press ***Tab*** to change to **edit mode**. Press ***z*** to change to transparent view (**Viewport shading, wireframe**). Press ***a*** to select the whole mesh of the dendrite with spines. Select **Model Objects**. Select **Dendrite**. Press ***t*** to hide the **CellBlender panel** and better visualize the whole mesh in the main panel.

4.2.2. Use ***+*** and ***–*** on the keypad to zoom in and out or scroll with the mouse. This is required for better visualization of the top of the spines to select and define the surface regions. Press ***a*** to deselect the object. Press ***Tab*** to change to **Edit Mode**. Press ***t*** to show the **CellBlender panel** again.

4.3. Define the PSD surface region. To do so, press ***b*** and select the top of a dendritic spine with the mouse (**Figure 6A, 6B**). Press ***+*** on **Defined Surface Regions**. Change the **Region Name** to ***PSD1*** and click **Assign** (**Figure 6C**). Press ***a*** to deselect the object.

4.4. Define the extrasynaptic surface region. To do so, press ***b*** and select the region around the top of the dendritic spine with the mouse (**Figure 6D**). Repeat step 4.3 for the **Region Name** to ***Extra_syn1***. Repeat the step 4.3 for the other spines to define the other regions of the mesh (***PSD2***, ***PSD3***, ***PSD4***, ***Extra_syn2***, ***Extra_syn3***, and ***Extra_syn4)*** (**Figure 6F**). Press ***a*** to deselect the object.

4.5. Define the surface regions of the ends of the dendrite. To do so, press ***b*** and select the left end of the dendrite. Change the **Region Name** to ***Left_end*** and click **Assign**. Press ***a*** to deselect the object. Press ***b*** and select the right end of the dendrite (**Figure 6E**). Change the **Region Name** to ***Right_end*** and click **Assign**.

NOTE: Move the mesh to find the best position to select each **Defined Region**.

5. Create molecules

5.1. Create AMPARs. To do so, select **Molecules** on the **CellBlender Panel**. Select ***+*** on **Defined Molecules** to insert a new molecule and change **Name** to ***AMPAR***. Change **Molecule Type** to ***Surface Molecule*** and **Diffusion Constant** to ***0.05e-8 cm^2/s^{14}*** to define the diffusion constant of AMPARs in the membrane (**Figure 7A**).

5.2. Create Anchors. To do so, select **Molecules** on the **CellBlender Panel**. Select ***+*** on **Defined Molecules** to insert a new molecule and change **Name** to ***anchor***. Change **Molecule Type** to ***Surface Molecule*** and change **Diffusion Constant** to ***0.001e-8 cm^2/s^{14}*** to define the diffusion constant of anchors in the membrane (**Figure 7A**).

5.3. To create Anchors Bound to AMPARs, select **Molecules** on the **CellBlender Panel**. Select ***+*** on **Defined Molecules** to insert a new molecule. Change **Name** to ***anchor_AMPAR***. Change **Molecule Type** to **Surface Molecule**. Change **Diffusion Constant** to ***0.001e-8 cm^2/s^{14}***.

5.4. Create the ***anchor_LTP and anchor_AMPAR_LTP.*** To do so, repeat step 5.2. Name the molecule ***anchor_LTP***. Repeat step 5.3. Name the molecule ***anchor_AMPAR_LTP***.

NOTE: The ***anchor_LTP*** has a high affinity for ***AMPAR***; thus, AMPARs increase in the synaptic regions.

5.5. Create the ***anchor_LTD*** and ***anchor_AMPAR_LTD***. To create an **anchor*_LTD,*** repeat step 5.2. Name the molecule ***anchor_LTD***. Repeat step 5.3. Name the molecule ***anchor_AMPAR_LTD***.

NOTE: The ***anchor_LTD*** has a low affinity for ***AMPAR***; thus, AMPARs decrease in the synaptic region.

6. Define surface classes

NOTE: This procedure defines the classes with the properties that are associated with the surface regions. The extrasynaptic regions reflect the free anchors and anchors bound to AMPAR. The lateral ends of the dendrite reflect all the molecules.

6.1. Define the properties of the extrasynaptic regions.

6.1.1. Press ***Tab*** to change to **object mode**. Select **Surface Classes** on the **CellBlender Panel**. Press ***+*** on the **Surface Class** to define a new surface class.

6.1.2. Make the extrasynaptic region reflect the *AMPAR* bound to the *anchor* molecules.

NOTE: This procedure will trap the *anchors* and everything bound to them within the synaptic region.

6.1.2.1. Change **Surface Class Name** to ***reflective_extra_syn***. Press ***+*** on ***reflective_extra_syn*** **Properties** to associate it with a molecule. Select **Molecules** | ***Single Molecule***. Select ***anchor_AMPAR***. Select **Orientation** = ***Ignore***. Select **Type** = ***Reflective*** to make the region show the ***anchor_AMPAR*** molecules.

6.1.2.2. Repeat step 6.1.2.1 for ***anchor_AMPAR_LTP*** and ***anchor_AMPAR_LTD***.

6.1.3. Make the extrasynaptic region reflect the *anchors*.

6.1.3.1. Press ***+*** on ***reflective_extra_syn*** **Properties** to associate it with a molecule. Select **Molecules** | ***Single Molecule***. Select ***anchor***. Select **Orientation** = ***Ignore***. Select **Type** = ***Reflective*** to make the region reflect the *anchor* molecules.

6.3.1.2. Repeat step 6.1.3.1 for ***anchor_LTP*** and ***anchor_LTD***.

6.2. Define the properties of the dendrite ends. To do so, press ***+*** on **Surface Class** to define a

new surface class. Change **Surface Class Name** to ***reflective_ends***. Press ***+*** on **Properties** to associate it with a molecule. Select **Molecules** | ***All Surface Molecules***. Select **Orientation** | ***Ignore***. Select **Type** | ***Reflective*** to make it reflect all surface molecules.

7. Assign the created classes to each surface region

NOTE: This step assigns the surface classes to the surface regions.

7.1. Assign the properties of the ends of the *dendrite*.

7.1.1. Press ***+*** to **assign** a surface class with a region. Select ***reflective_ends*** for **Surface Class Name** (**Figure 7C**). Select ***Dendrite*** for **Object Name**. Select ***Specified Region*** for **Region Selection**. Select ***Left_end*** for **Region Name**.

7.1.2. Repeat step 7.1.1 for the ***Right_end*** (**Figure 7D**).

7.2 Assign the properties of the extrasynaptic regions.

7.2.1. Press ***+*** to assign a surface class with a region. Select ***reflective_extra_syn*** for **Surface Class Name**. Select ***Dendrite*** for **Object Name**. Select ***Specified Region*** for **Region Selection**. Select ***Extra_syn1*** for **Region Name**.

7.2.2. Repeat step 7.2.1 for ***Extra_syn2***, ***Extra_syn3***, and ***Extra_syn4***.

8. Place molecules on the mesh

NOTE: This step places the *AMPARs*, *anchors*, and *AMPAR* bound to anchors on the mesh.

8.1. To place *AMPAR* **Molecules** on the Mesh, select **Molecule Placement** on the **CellBlender Panel**. Press ***+*** on the **Release/Placement Sites** to create a **new release site**. Change **Site Name** to ***relAMPAR*** (**Figure 7B**). Select **Molecule** = ***AMPAR***. **Object/Region** = ***Dendrite[ALL]-(Dendrite[Left_end]+Dendrite[Right_end])***. **Quantity to Release** = ***1,000***.

8.2. Place *anchor* molecules on the Mesh.

8.2.1. Select **Molecule Placement** on the **CellBlender Panel**. Press ***+*** on the **Release/Placement Sites** to create a **new release site**. Change **Site Name** to ***rel_anchor_PSD1***. Select **Molecule *anchor***. **Object/Region** = ***Dendrite[PSD1]***. **Quantity to Release** = ***200***.

8.2.2. Repeat step 8.2.1 for ***PSD2***, ***PSD3***, and ***PSD4***.

8.3. Place ***anchor_LTP*** **Molecules** on the Mesh. To do so, select **Molecule Placement** on the **CellBlender Panel**. Press ***+*** on the **Release/Placement Sites** to create a **new release site**. Change **Site Name** to ***rel_anchor_LTP_PSD1***. Select **Molecule = *anchor_LTP***. **Object/Region** =

Dendrite[PSD1]*. Quantity to Release** = ***0.

NOTE: ***anchor_LTP*** is an anchor with high binding affinity for *AMPARs*.

8.4. Place ***anchor_LTD* Molecules** on the Mesh by repeating step 8.3 for ***anchor_LTD***.

NOTE: ***anchor_LTD*** is an anchor with low binding affinity for *AMPARs*.

9. Create the chemical reactions

9.1. Creating the **reaction** between *anchor* and *AMPARs*.

9.1.1. Select **Reactions** (**Figure 7D**) to create the reactions. Press ***+*** to include a **new reaction**. **Reactants** = ***anchor' + AMPAR'***. **Reaction Type** = ***<->***. This defines a bidirectional reaction. **Products** = ***anchor_AMPAR'***. **Forward Rate** = ***0.03***. **Backward Rate** = ***0.05***.

9.2. Create the **reaction** between *ANCHOR_LTP* and *AMPARs*. To do so, repeat step 9.1, but replace ***anchor*** with ***anchor_LTP***, and use a **Backward Rate** = ***0.005*** to increase the affinity between the **reactants**.

9.3. Create the **reaction** between ***anchor_LTD*** and *AMPARs* and save the file. To do so, repeat step 9.2, but replace ***anchor*** with ***anchor_LTD***, and use a **Backward Rate** = ***0.5*** to decrease the affinity between the **reactants**. Then **Save** the file.

10. Plot the output of the model

10.1. Plot *anchors* bound to *AMPARs* at the *PSD1* during the basal condition. To do so, select **Plot Output Settings**. Press ***+*** to define the **molecules**. Select ***anchor_AMPAR*** on **Molecule**. Select ***dendrite*** on **Object**. Select ***PSD1*** on **Region**. Repeat step 10.1 for all PSD regions.

NOTE: It is useful to observe the basal number of trapped *AMPARs* to the PSD of each dendritic spine. The number of anchors bound to *AMPARs* can increase or decrease in comparison to the basal conditions during LTP and LTD.

10.2. Plot anchors bound to *AMPARs* in the *PSD1* during LTP. Do this by repeating step 10.1. Replace ***anchor_AMPAR*** with ***anchor_AMPAR_LTP***, then plot anchors bound to *AMPARs* in the *PSD1* during LTD and finally repeating step 10.1, but replace ***anchor_AMPAR_LTP*** with ***anchor_AMPAR_LTD***.

11. Run the simulations

11.1. To run the basal condition, select **Run Simulation.** Select **Iterations** = ***30,000***. Set **Time Step** = ***1e-3* s**. Press **Export & Run**. Wait until the simulation ends. It may take from minutes to hours.

NOTE: In the basal condition, there is no release of *anchor_LTP* and *rel_anchor_LTD* **molecules**. Regarding the parameters of the simulation, the number of iterations needs to be long enough to be able to observe the diffusion of *AMPARs* from the dendrites and their anchoring at PSD. Small time-steps are more precise but slower to complete the simulation.

11.2. Select **Reload Visualization Data**. Select **play animation** to visualize the spatiotemporal results (**Figure 8**). Select **Plot Output Settings**. Press **Plot**.

NOTE: The graphs generated by **CellBlender** are isolated time series of the selected chemical species. Third-party programs can be used to import the data saved from multiple simulations to create overlaid plots of several conditions (e.g., basal, LTP, LTD; see **Figure 8**).

11.3. Run the homosynaptic potentiation condition (i.e., LTP; see **Figure 8**). To do so, select **Molecule Placement** on the **CellBlender Panel**. Select ***rel_anchor_LTP_PSD1*** on the **Release/Placement Sites**.

11.4. Change **quantity to release** = ***200***. Select ***rel_anchor_LTD_PSD1*** on the **Release/Placement Sites**. Change **quantity to release** = ***0***. Select ***rel_anchor _PSD1*** on the **Release/Placement Sites**. Change **quantity to release** = ***0***. Repeat steps 11.1–11.2.

11.5. Run the homosynaptic depression condition (i.e., LTD; see **Figure 8**). To do so, **Release *200* *rel_anchor_LTD_PSD1*** instead of ***rel_ANCHOR_LTP_PSD1***. Set ***rel_anchor*** and ***rel_anchor_LTP_PSD1*** to ***zero***. Repeat steps 11.1–11.2.

REPRESENTATIVE RESULTS:

These results provide the steps for the construction of a 3D mesh that simulates a dendritic spine with a spine head and spine neck (**Figure 1** to **Figure 4**). In addition, multiple dendritic spines can be inserted in a single dendritic segment (**Figure 5**) to study heterosynaptic plasticity of AMPARs[14]. The PSD on the top of the spine head (**Figure 6**) is the place where synaptic anchors bind to AMPARs and trap them temporarily at the synapse (**Figure 7**, **Figure 8**).

Synaptic plasticity could be verified roughly through changes in the number of species of *anchor_AMPAR*, *anchor_AMPAR_LTP*, and *anchor_AMPAR_LTD* at each spine. For the exact calculation of the occurrence of synaptic plasticity, it is recommended to calculate the variation in the total number of anchored and free AMPARs at the synapse. This can be performed using third-party programs to open the saved data of the simulation to summate the time series of the free AMPARs and the anchored AMPARs at each PSD (**Figure 8**).

The release of AMPARs on the mesh allowed the observation of their diffusion by a stochastic random walk along the dendrite and dendritic spines. Factors that modify the affinity of AMPARs for the anchors, such as posttranslational modifications and alterations of the rates of endocytosis and exocytosis, can trap the AMPARs at the PSD[24–26]. The binding of AMPARs with the anchors located at the PSD trapped a high density of AMPARs at the synapse. Homosynaptic potentiation (**Figure 9**) and depression (**Figure 10**) could be verified respectively through

increases and decreases in the number of anchored *AMPARs* caused by changes in the affinity of AMPARs by anchors in comparison to the basal condition (**Figure 11**). Factors that reduced the affinity of AMPARs with the anchors released multiple *AMPARs* from one dendritic spine (i.e., homosynaptic depression) and induced heterosynaptic potentiation at the neighboring spines. Also, factors that increased the affinity of *AMPARs* for the anchors at one spine induced homosynaptic potentiation at that spine and heterosynaptic depression at the neighboring spines[14]. In this way, heterosynaptic plasticity was observed as the opposite effect at the neighboring spines of the homosynaptic plasticity induced at a given spine. For instance, homosynaptic LTP induction at a single spine created a heterosynaptic LTD effect at the neighboring spines (**Figures 8E**,**F**,**G**).

FIGURE AND TABLE LEGENDS:

Figure 1: Creation of the dendritic spine head using a spherical mesh. (**A**) Adding the UV sphere. (**B**) Setting up the sphere dimensions. (**C**) Observing the created sphere.

Figure 2: Construction of the top region. (**A**) Selecting the top region of the sphere. (**B**) Removing the selected region to make it flat. (**C**) Sealing the flat top.

Figure 3: Creating concentric areas on the top of the spine. (**A**) Visualizing the top. (**B**) Using a knife to define a concentric region. (**C**) Creating multiple concentric regions.

Figure 4: Creating the dendritic spine neck. (**A**) Selecting the bottom of the modified sphere. (**B**) Deleting the selected vertices. (**C**) Selecting the bottom. (**D**) Extrusion of the bottom to create the spine neck. (**E**) Sealing the bottom of the spine neck. (**F**) Analyzing the created spine.

Figure 5: Creation of the dendrite with multiple spines. (**A**) Using the cylindrical mesh to create a dendrite. (**B**) Aligning the dendritic spine with the cylinder. (**C**) Joining the cylinder with the spine. (**D**) The Boolean operation to join the meshes. (**E**) The new combined mesh. (**F**) Adding the second spine. (**G**) Adding the third spine. (**H**) Adding the fourth spine.

Figure 6: Defining the PSD region and the perisynaptic zone. (**A**) Selecting the PSD region. (**B**) Detailed view of the created PSD. (**C**) Defining the PSD surface region. (**D**) Selecting and defining the perisynaptic zone around the PSD. (**E**) Selecting and defining the lateral surface of the dendrite. (**F**) Defined surface regions.

Figure 7: Defining the surface molecules. (**A**) Defining AMPAR, anchor, and AMPAR bound to anchor. (**B**) Defining the location and quantity of AMPAR copies. (**C**) Defining the Surface Classes. (**D**) Assigning the Surface Classes. (**E**) Creating the chemical reactions between the molecules.

Figure 8: Representative results of synaptic plasticity. (**A**) Different meshes of a dendritic segment with two, four, or eight spines. (**B**) A different view of the dendritic segment with eight spines. (**C**) Detailed view of a dendritic spine with AMPARs and anchors at the PSD. (**D**) Diagram of the trafficking of AMPARs in and out of the PSD through their interactions with the anchors. (**E-G**) The curves show the number of synaptic AMPARs at each PSD for the basal condition and

during LTP and LTD. The induction of homosynaptic LTP or LTD at a single spine created a heterosynaptic effect in the nearby spines for the mesh with two spines (**E**), four spines (**F**), and eight spines (**G**).

Figure 9: Representative result of the LTP condition. (**A**) The x-axis is the time and the y-axis is the number of the complex anchor_LTP_AMPAR at PSD1. There was a release of 200 free anchor_LTP at the beginning of the simulation. A higher number of bonds with anchors was formed in comparison to the basal condition (**Fig. 9**)

Figure 10: Representative result of the LTD condition. (**A**) The x-axis is the time and the y-axis is the number of the complex anchor_LTD_AMPAR at PSD1. There was a release of 200 free anchor_LTD at the beginning of the simulation. A lower number of bonds with anchors were formed in comparison to the basal condition (**Figure 9**).

Figure 11: Representative result during basal condition. (**A**) The x-axis is the time and the y-axis is the number of the complex anchor_AMPAR at PSD1. There was a release of 200 free anchors at the beginning of the simulation.

DISCUSSION:

This article presents a method for the construction of 3D meshes for modeling reaction-diffusion synaptic plasticity processes in a dendritic segment with dendritic spines. The developed model contains a dendritic segment with few dendritic spines. The lateral diffusion and reaction of AMPARs with synaptic anchors allow the simulation of the basal dynamics. The critical steps in the protocol are cutting the sphere for the creation of the top of the spine head (**Figure 1**, **Figure 2**, **Figure 3**), the extrusion to create the spine neck (**Figure 4**), and the joining of the dendrite and the spines into a single mesh (**Figure 5**). It is critical to have a complete overlap between the spine necks and the dendrite; otherwise, the mesh will not be watertight. Other critical steps are the selection of the membrane regions and the definition of the surface classes (**Figure 6**, **Figure 7**). Save the files for each critical step with a different name.

Use the **mesh analyze** tool to ensure that the mesh is watertight, manifold, and outward-facing normal after creating the single spine and after creating the combined dendrite with the spine. In the mesh fails this analysis, return to the last correct version saved. Some steps may be slightly different depending on the version of the software installed, the operating system, and the type of keyboard.

This protocol simulates the trafficking of AMPAR molecules in the 3D mesh (**Figure 8**, **Figure 9**, **Figure 10**, **Figure 11**), which is key for neuronal excitatory transmission and synaptic plasticity. The trafficking of single molecules in a 3D mesh is an valuable feature of this model with respect to existing methods based on well-mixed volumes with homogeneous distributions of molecules[21,22], which is not the physiological condition at the synapses[27]. A limitation of this technique is the high computational cost and the slow velocity of simulations that use a high number of copies of each molecule and a high number of chemical reactions between them. This constraint can be overcome by reducing the number of copies of each species.

The construction of a system with a realistic 3D mesh and spatiotemporal tracking of molecules is a powerful tool to test mechanical scenarios that can give great insights about the functioning of systems with a high number of nonlinear variables.

ACKNOWLEDGMENTS:

This work was supported in part by the Sao Paulo State Science Foundation (FAPESP) grant #2015/50122-0 and IRTG-GRTK 1740/2, by the IBM/FAPESP grant #2016/18825-4, and by the FAPESP grant #2018/06504-4.

DISCLOSURES:

The authors declare that they have no competing financial interests.

REFERENCES:

1. Sweatt, J. D. Neural plasticity and behavior - sixty years of conceptual advances. *Journal of Neurochemistry*. **139**, 179–199 (2016).
2. Heine, M. et al. Surface mobility of postsynaptic AMPARs tunes synaptic transmission. *Science*. **320** (5873), 201–205 (2008).
3. Buonarati, O. R., Hammes, E. A., Watson, J. F., Greger, I. H., Hell, J. W. Mechanisms of postsynaptic localization of AMPA-type glutamate receptors and their regulation during long-term potentiation. *Science Signaling*. **12** (562), eaar6889 (2019).
4. Nair, D. et al. Super-Resolution Imaging Reveals That AMPA Receptors Inside Synapses Are Dynamically Organized in Nanodomains Regulated by PSD95. *Journal of Neuroscience*. **33** (32), 13204–13224 (2013).
5. Czöndör, K. et al. Unified quantitative model of AMPA receptor trafficking at synapses. *Proceeding of the National Academy of Sciences of the United States of America*. **109** (9), 3522–3527 (2012).
6. Triesch, J., Vo, A. D., Hafner, A.-S. Competition for synaptic building blocks shapes synaptic plasticity. *eLife*. **7**, 37836 (2018).
7. Earnshaw, B. A., Bressloff, P. C. Biophysical model of AMPA receptor trafficking and its regulation during long-term potentiation/long-term depression. *Journal of Neuroscience*. **26** (47), 12362–12373 (2006).
8. Earnshaw, B. A., Bressloff, P. C. Modeling the role of lateral membrane diffusion in AMPA receptor trafficking along a spiny dendrite. *Journal of Computational Neuroscience*. **25** (2), 366–389 (2008).
9. Antunes, G., Roque, A. C., Simoes-de-Souza, F. M. Stochastic Induction of Long-Term Potentiation and Long-Term Depression. *Scientific Reports*. **6**, 30899 (2016).
10. Kotaleski, J. H., Blackwell, K. T. Modelling the molecular mechanisms of synaptic plasticity using systems biology approaches. *Nature Reviews Neuroscience*. **11** (4), 239–251 (2010).
11. Bhalla, U. S. Molecular computation in neurons: a modeling perspective. *Current Opinion in Neurobiology*. **25**, 31–37 (2014).
12. Czöndör, K., Thoumine, O. Biophysical mechanisms regulating AMPA receptor accumulation at synapses. *Brain Research Bulletin*. **93**, 57–68 (2013).
13. Bromer, C. et al. Long-term potentiation expands information content of hippocampal

dentate gyrus synapses. *Proceedings of the National Academy of Sciences of the United States of America*. **115** (10), E2410–E2418 (2018).
14. Antunes, G., Simoes-de-Souza, F. M. AMPA receptor trafficking and its role in heterosynaptic plasticity. *Scientific Reports*. **8** (1), 10349 (2018).
15. Kerr, R. A. et al. Fast monte carlo simulation methods for biological reaction-diffusion systems in solution and on surfaces. *SIAM Journal on Scientific Computing*. **30** (6), 3126 (2008).
16. Czech, J., Dittrich, M., Stiles, J. R. Rapid Creation, Monte Carlo Simulation, and Visualization of Realistic 3D Cell Models. *Systems Biology*. **500**, 237–287 (2009).
17. Stiles, J., Bartol, T. Monte Carlo Methods for Simulating Realistic Synaptic Microphysiology Using MCell. *Computational Neuroscience*. CRC Press. Eds. De Schutter et al. **20001945** (2000).
18. Jorstad, A. et al. NeuroMorph: A Toolset for the Morphometric Analysis and Visualization of 3D Models Derived from Electron Microscopy Image Stacks. *Neuroinformatics*. **13** (1), 83–92 (2015).
19. Antunes, G., Roque, A. C., Simoes de Souza, F. M. Modelling intracellular competition for calcium: kinetic and thermodynamic control of different molecular modes of signal decoding. *Scientific Reports*. **6**, 23730 (2016).
20. Antunes, G., Roque, A. C., Simoes-de-Souza, F. M. Molecular mechanisms of detection and discrimination of dynamic signals. *Scientific Reports*. **8** (1), 2480 (2018).
21. Hoops, S. et al. COPASI--a COmplex PAthway SImulator. *Bioinformatics*. **22** (24), 3067–3074, (2006).
22. Faeder, J. R., Blinov, M. L., Hlavacek, W. S. Rule-based modeling of biochemical systems with BioNetGen. *Methods in Molecular Biology*. **500**, 113–167 (2009).
23. Gillespie, D. T. Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry*. **81** (25), 21 (1977).
24. Anggono, V., Huganir, R. L. Regulation of AMPA receptor trafficking and synaptic plasticity. *Current Opinion in Neurobiology*. **22** (3), 461–469 (2012).
25. Matsuda, S., Launey, T., Mikawa, S., Hirai, H. Disruption of AMPA receptor GluR2 clusters following long-term depression induction in cerebellar Purkinje neurons. *EMBO Journal*. **19** (12), 2765–2774 (2000).
26. Ahmad, M. et al. Postsynaptic Complexin Controls AMPA Receptor Exocytosis during LTP. *Neuron*. **73** (2), 260–267 (2012).
27. Sheng, M., Hoogenraad, C. C. The postsynaptic architecture of excitatory synapses: a more quantitative view. *Annual Review of Biochemistry*. **76**, 823–47 (2007).

Figure 1

Click here to access/download;Figure;fig1.png

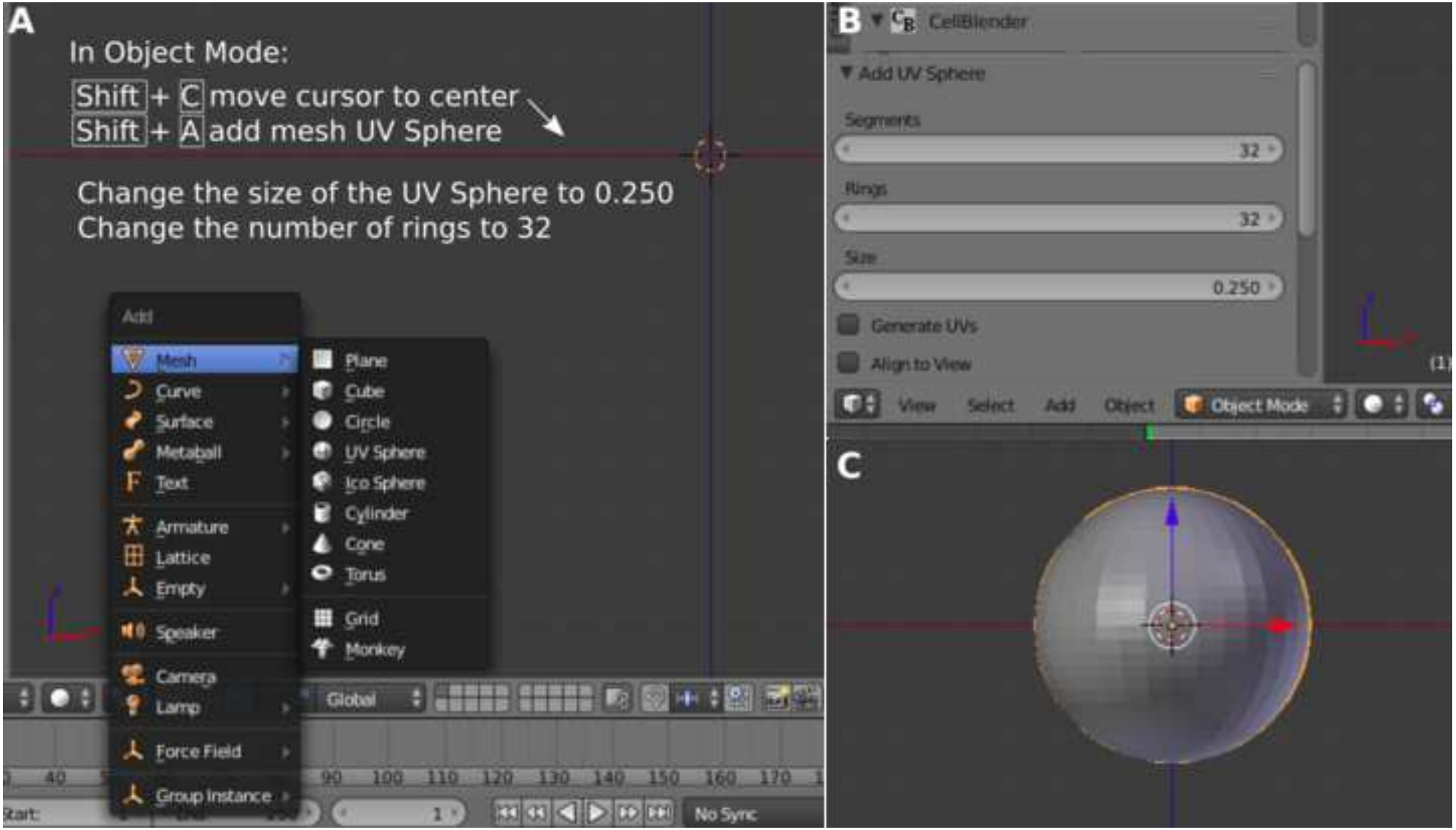


Figure 2

Click here to access/download;Figure;fig2.png

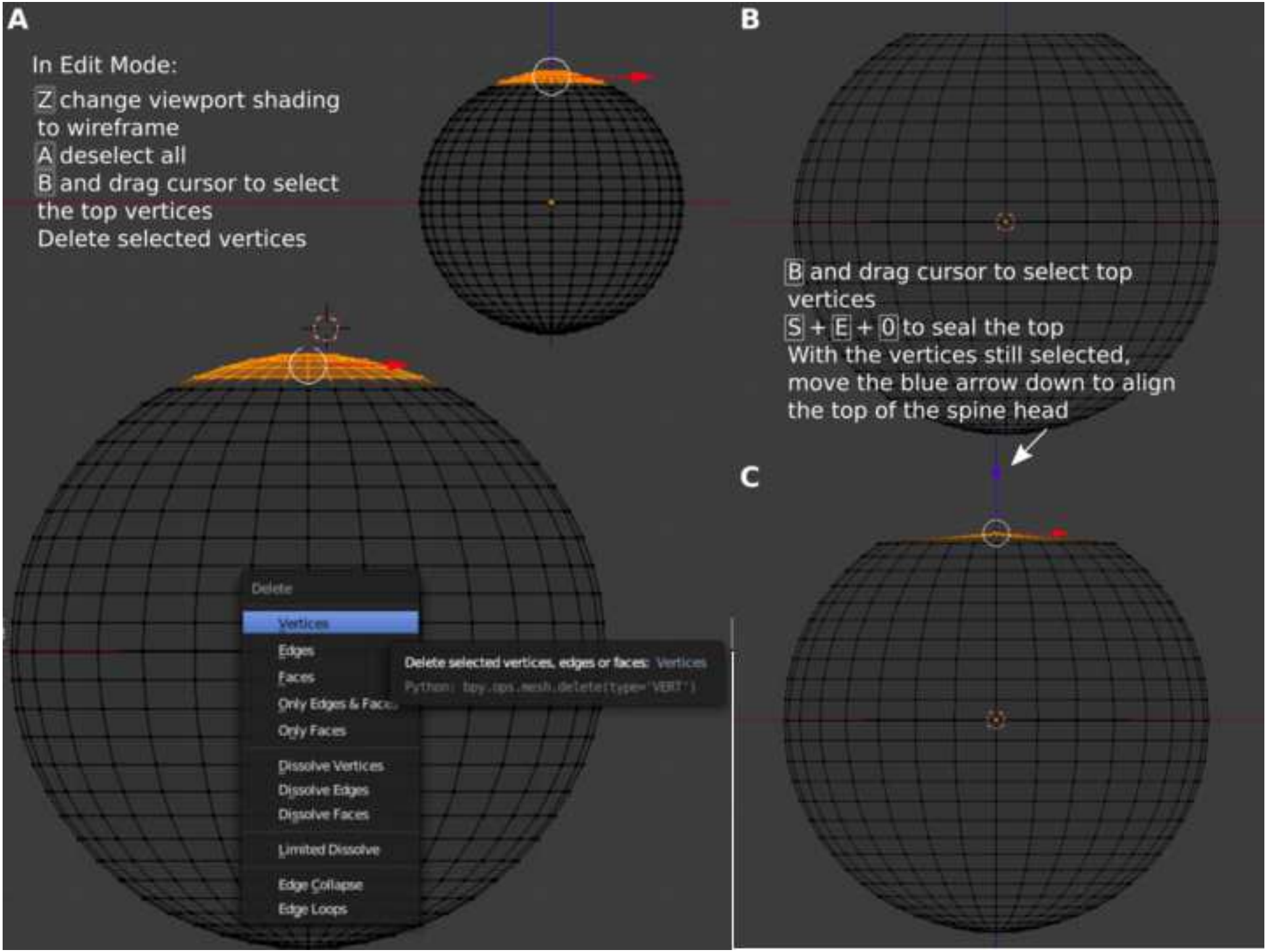


Figure 3

Click here to access/download;Figure;fig3.png

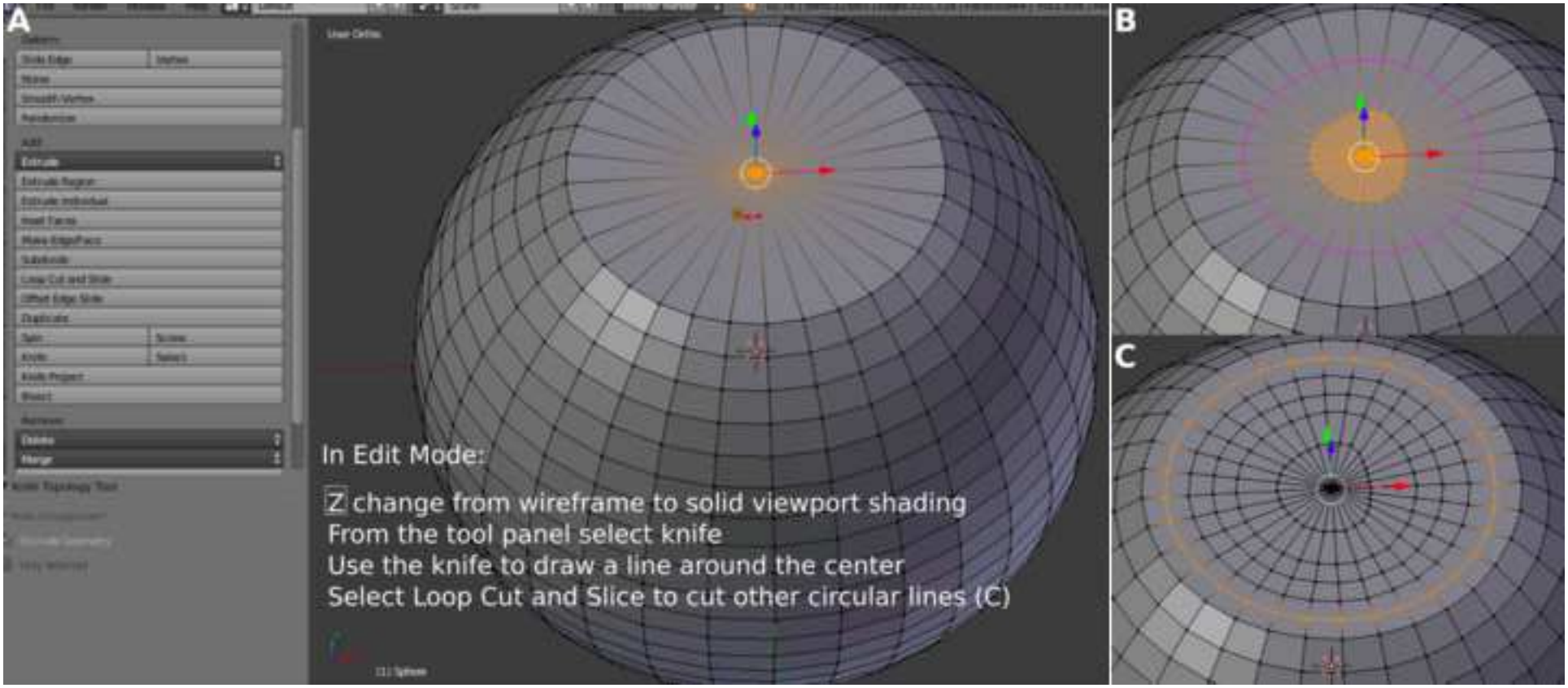


Figure 4

Click here to access/download;Figure;fig4.png

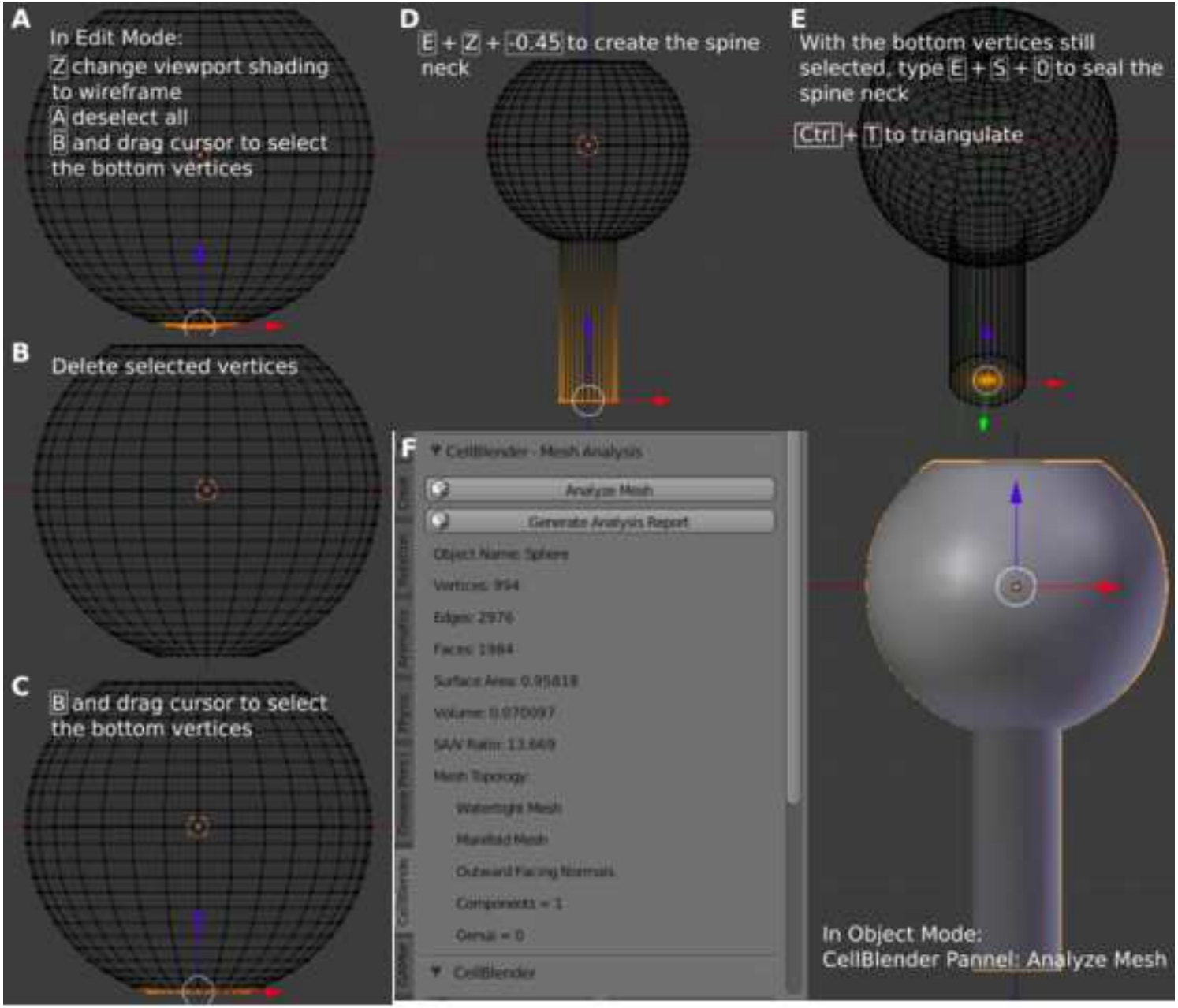


Figure 5

Click here to access/download;Figure;fig5.png

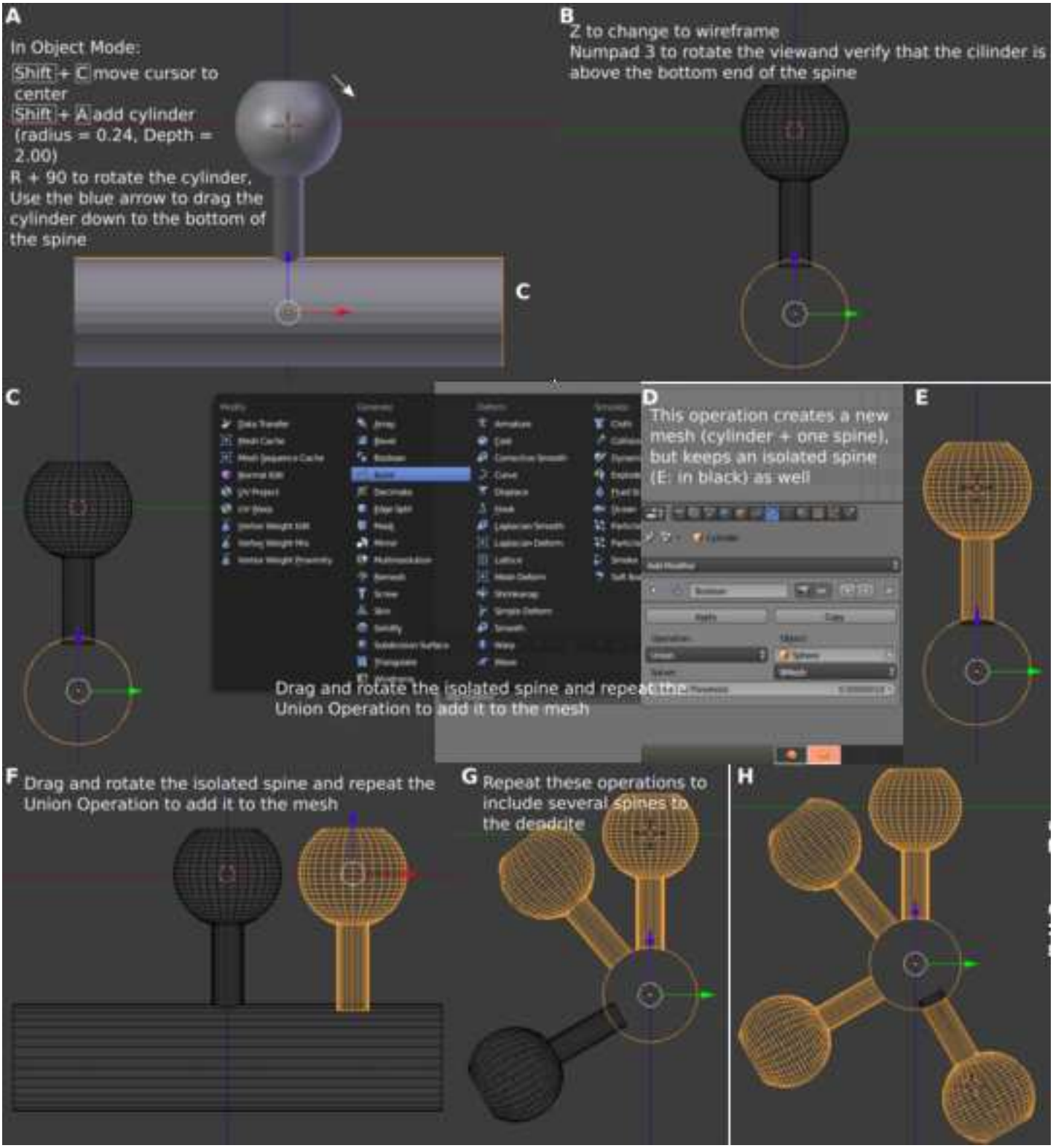


Figure 6

Click here to access/download;Figure;fig6.png

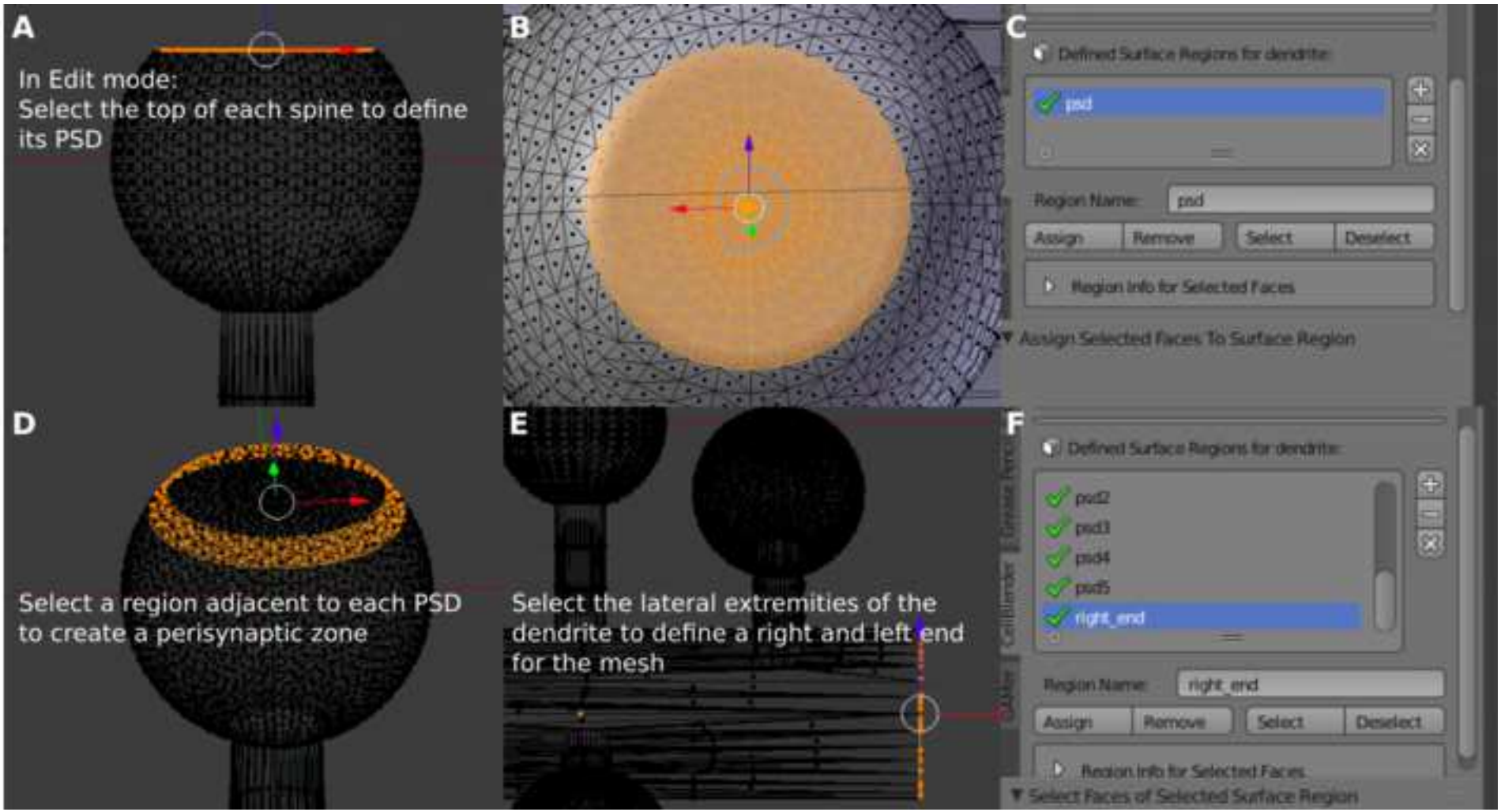


Figure 7

Click here to access/download;Figure;fig7.png

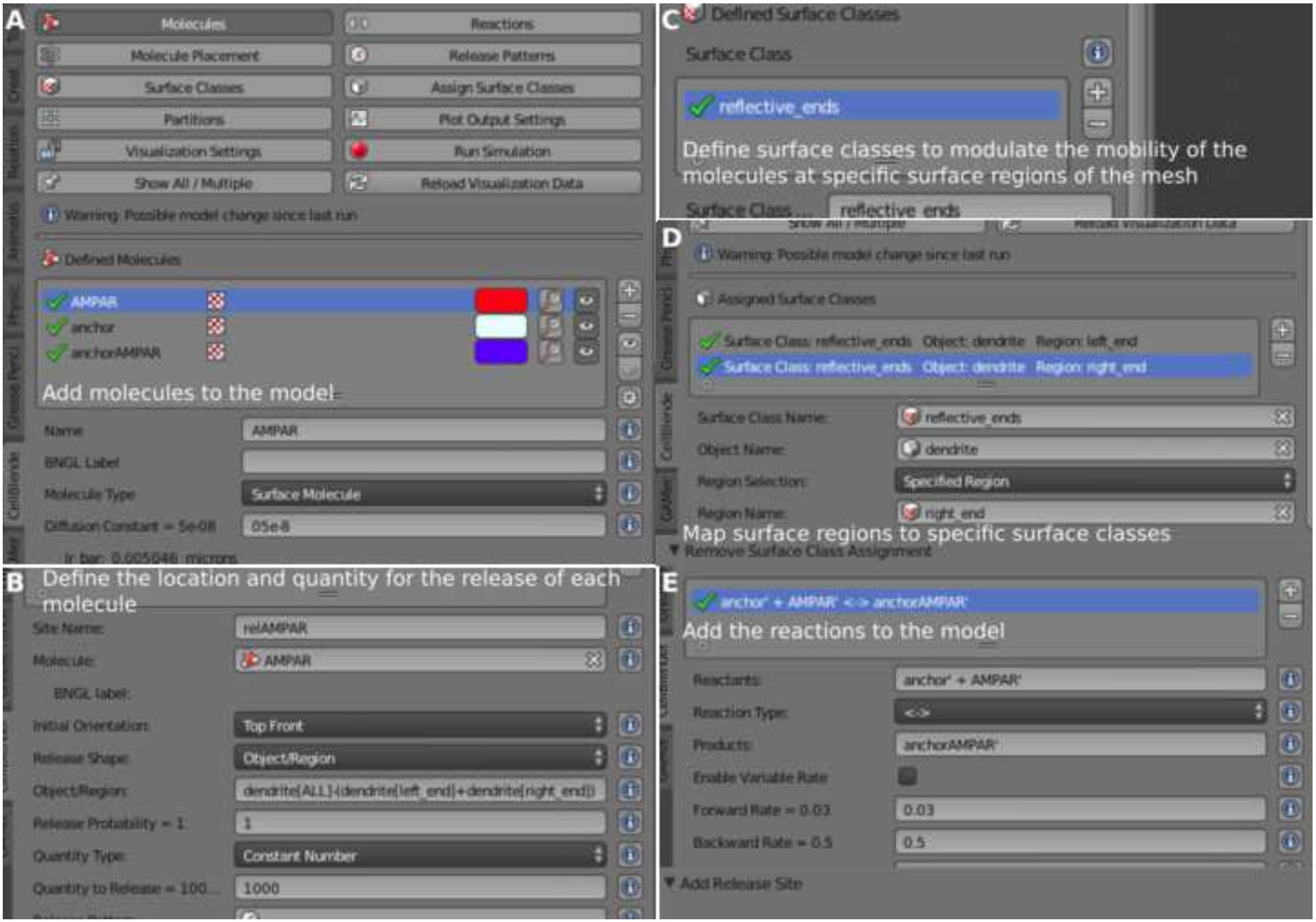


Figure 8

Click here to access/download;Figure;fig8.png

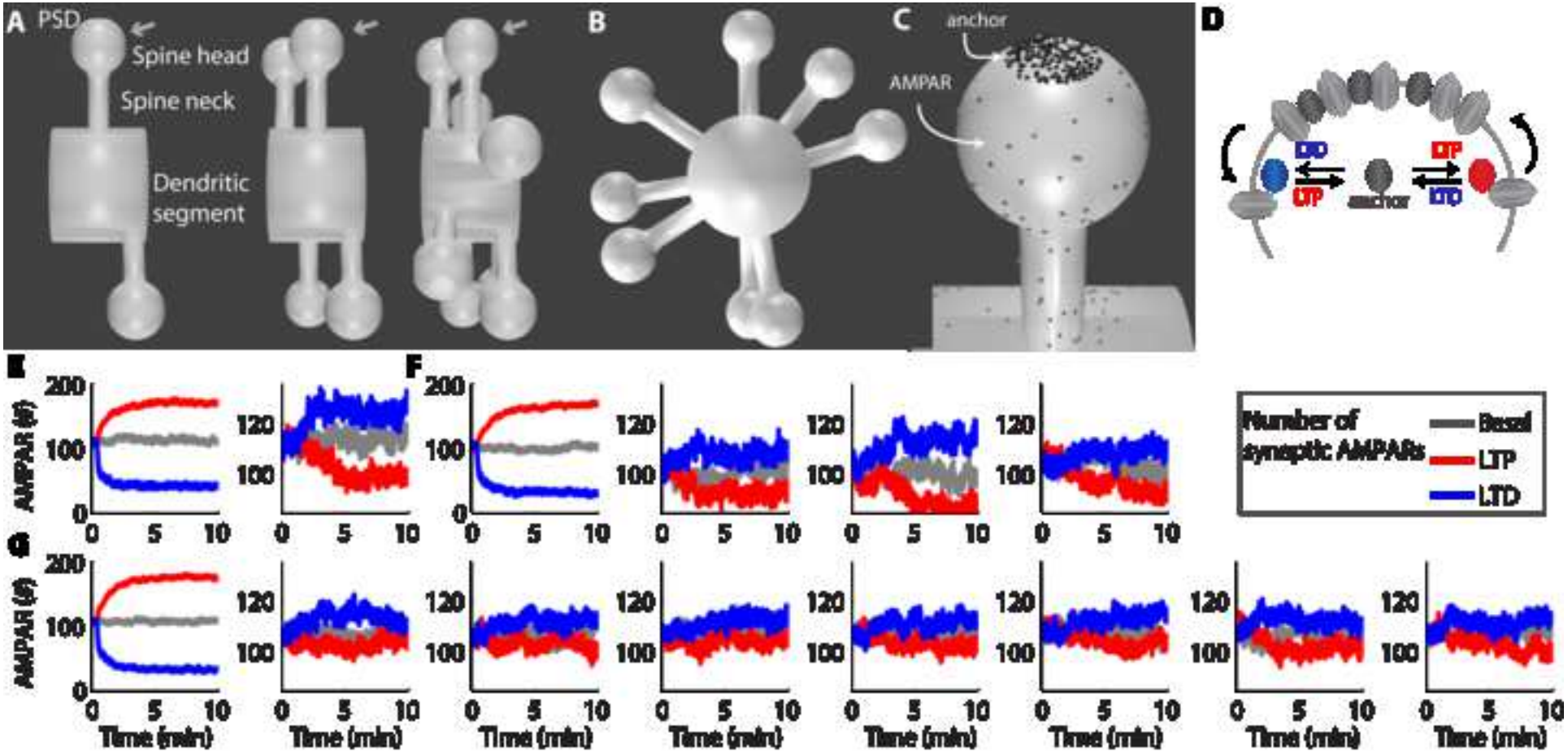


Figure 9

Click here to access/download;Figure;fig9.png

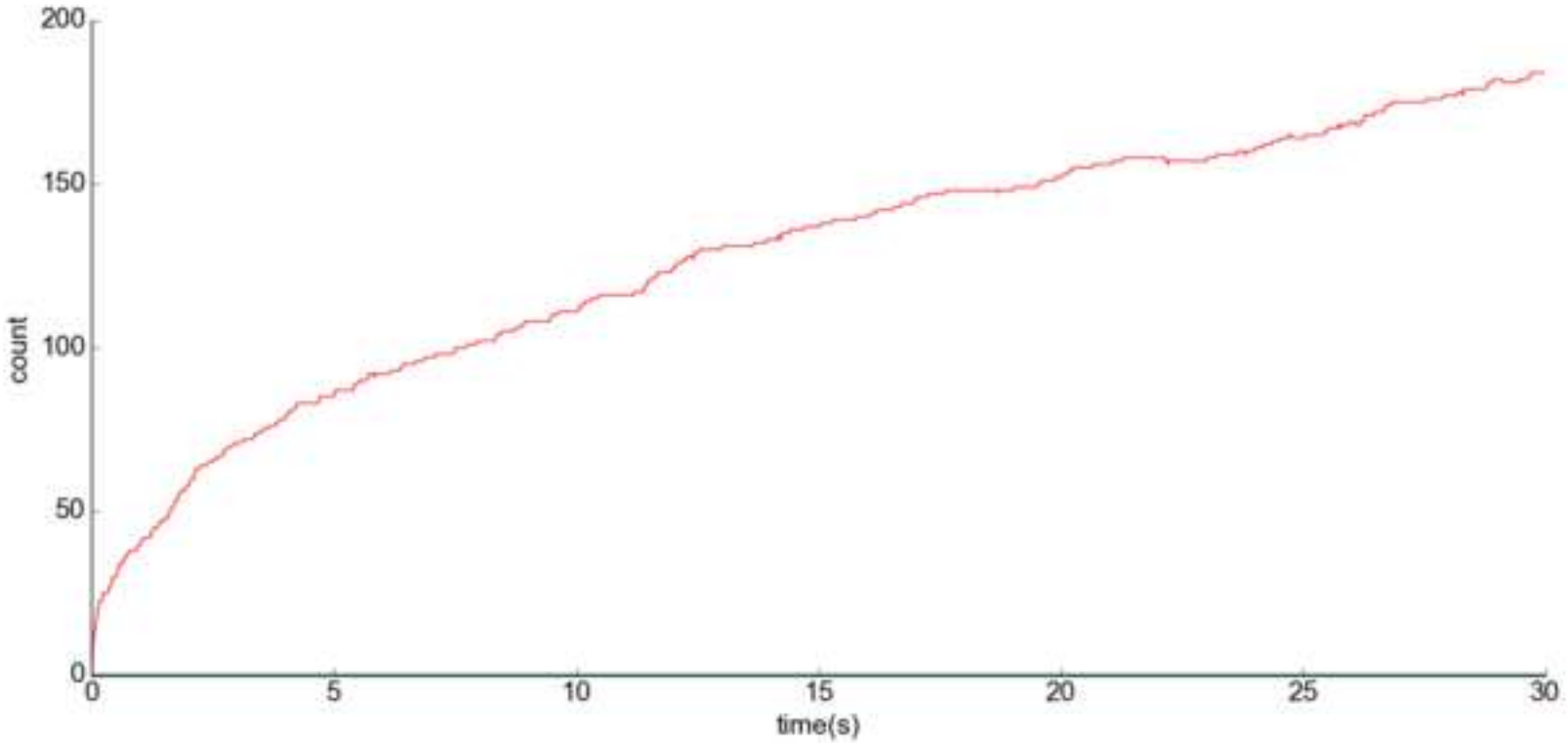


Figure 10

Click here to access/download;Figure;fig10.png

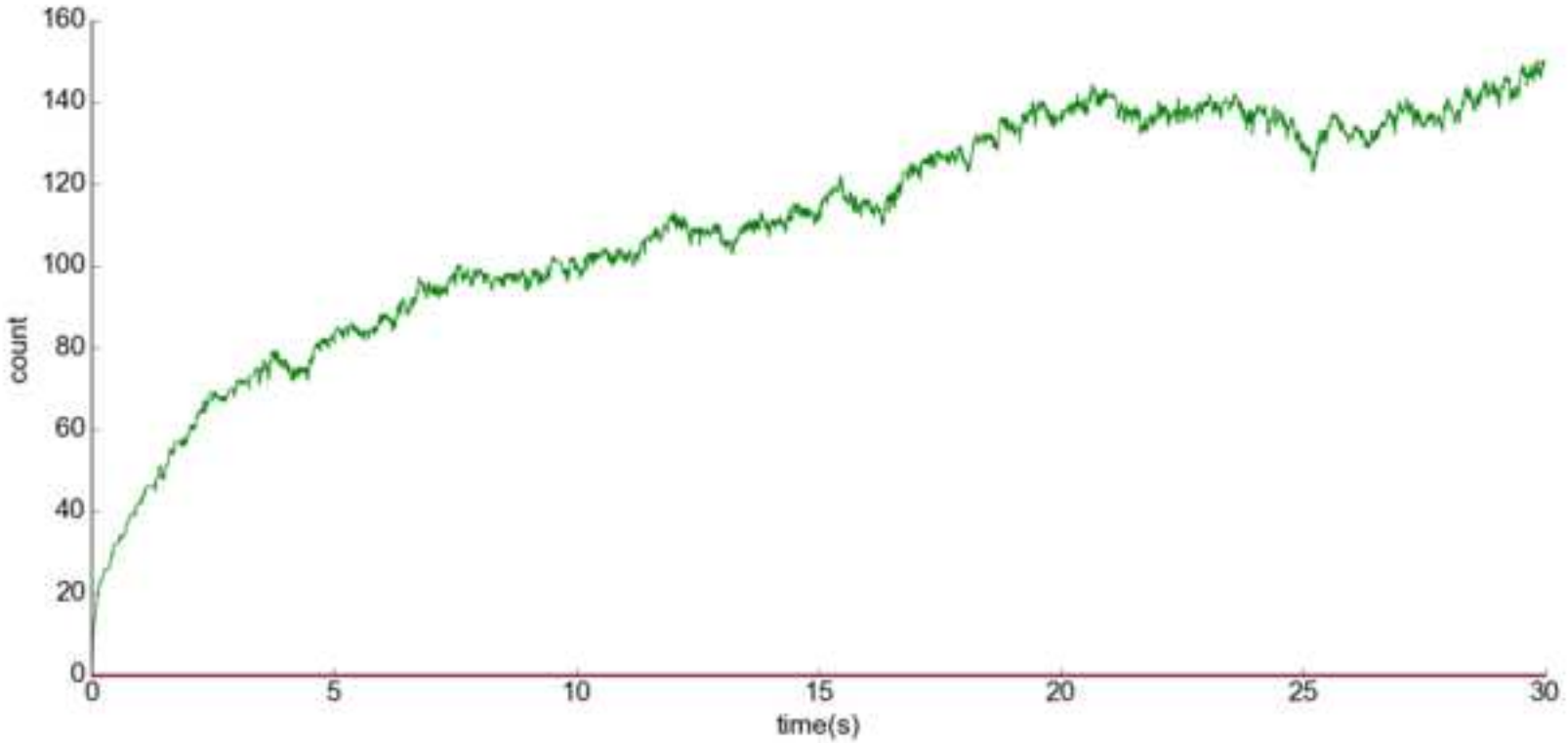
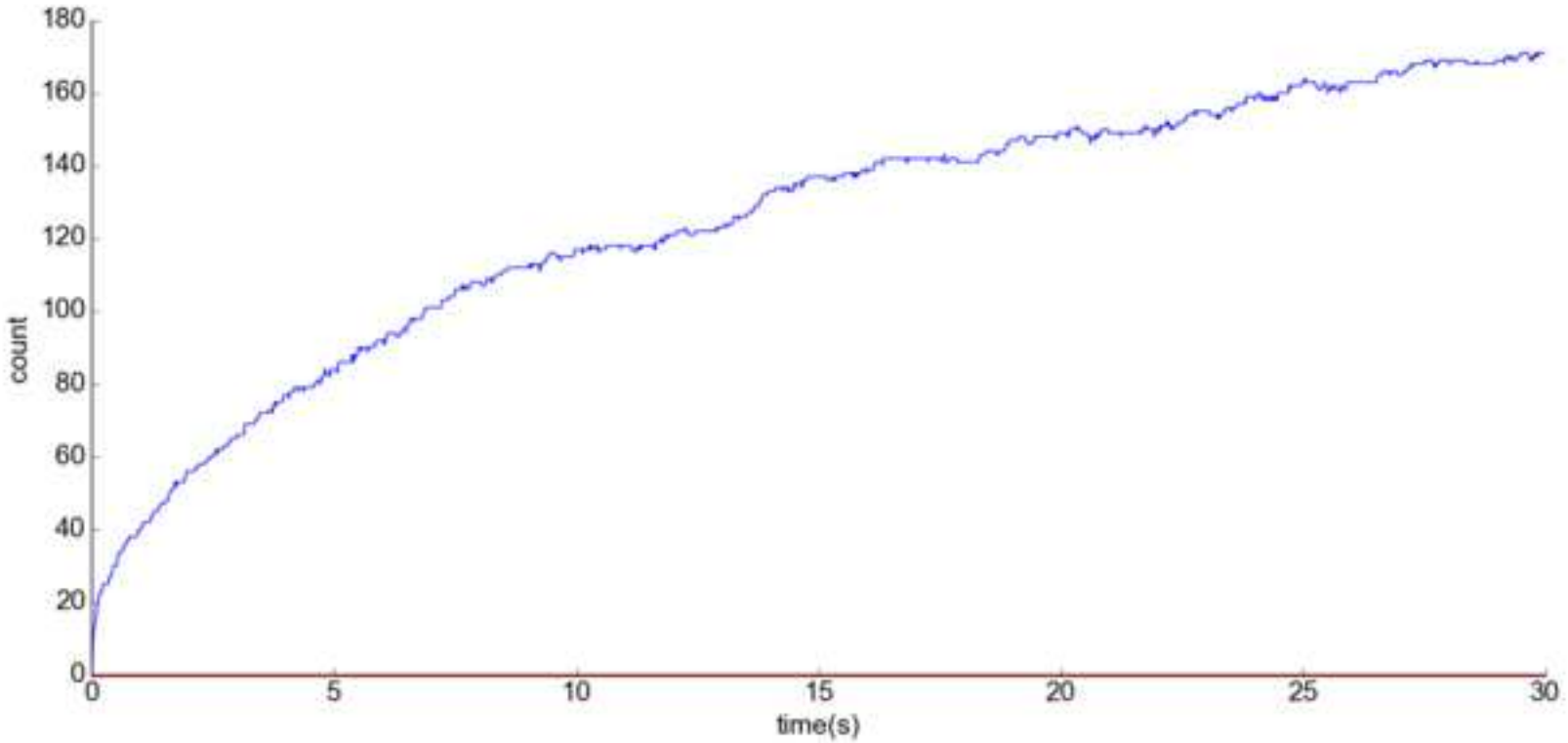


Figure 11

Click here to access/download;Figure;fig11.png



Name of Material/Equipment	Company	Catalog Number	Comments/Description
Blender	Blender Foundation		https://www.blender.org/
CellBlender	University of Pittsburgh		https://mcell.org/
Mcell	University of Pittsburgh		https://mcell.org/

Thank you for the suggestions that improved the manuscript. We revised the manuscript throughout to address the comments. We included more subdivisions and NOTES for clarity. We revised the grammar and increased the number of the subdivisions; we extended the glossary and included a new supplementary file with three figures.

Comment: Title: 3D Modeling of dendritic spines

Answer: 3D Modeling of Dendritic Spines with Synaptic Plasticity. We changed the title but maintained the synaptic plasticity.

Comment: Please reword

Answer: Computational Modeling of diffusion and reaction of chemical species in a three dimensional (3D) geometry is a fundamental method to understand the mechanisms of synaptic plasticity in dendritic spines.

Comment: Citation?
Answer: The citation has been inserted: The AMPAR synapses are located on the top of small volume structures called dendritic spines[3]

3. Buonarati, O.R., Hammes, E.A., Watson, J.F., Greger, I.H., Hell, J.W. Mechanisms of postsynaptic localization of AMPA-type glutamate receptors and their regulation during long-term potentiation. *Science Signaling*. 12 (562), doi: 10.1126/scisignal.aar6889 (2019).

Comment:

Presently the protocol sounds like a manual/tutorial which lists all the features of the software used. Please show this as a tool and how you used this to answer your specific research question. This can be done using a specific example to show how you are creating dendritic spine/s, and how is this being used to model synaptic plasticity.

Presently there is a missing link from one section to other. Please bring out clarity on this.

There is a 10 page limit for the protocol section (including headings and spacings) and there is a 2.75 page limit for the highlighted section (including headings and spacings). Presently both the limits are exceeding. Some of the shoter steps can be combined to 2-3 actions per step. Please see section 1 and 2 and reformat the other sections accordingly.
Please remove the redundancy from the protocol section.
Please define all abbreviations during the first time use. E.g., LTD, LTP, PSD, etc.

Answer:
We presented the protocol using an example as suggested. It created a link between the sections. We defined the abbreviations during the first time that they appear. We reduced the manuscript for the protocol section and reduced the highlighted section.

Comment:
Is it ok to add this here?
All these softwares are open access and does not require reprint permission to be used.
Answer:
OK. All these software are open access and does not require reprint permission to be used.

Comment:
This step can be further divided to in subsections: creating spine head, making the head flat, creating extrusions to match the dendrites etc. to bring out clarity.
Answer:
Done. We included more subsections for clarity

Comment:
This is not an action step so moved here. Also notes cannot be filmed so removed the highlights.
Answer:
OK. We removed the highlights.

Comment:
What will happen once you do this?
Answer:
We removed this sentence, since it was not necessary.

Comment:
Is this ok to include to bring out clarity?
Zoom in and out can also be performed using the scroll button of the mouse (**Figure 1C**).

Answer:
OK.

Comment:

So before this the object(dendrite spine) is created. Now you are editing it, right?
Answer:
Correct. Now I am editing it.

Comment:
How and in which step was the mesh selected previously? Why do you need to make the mesh transparent at this stage
Answer:
The mesh is automatically selected when it has been created. It needs to be transparent for better visualization of the editing parts.

Comment:
Reowrded to bring out clarity.
Answer:
OK.

Comment:
What is being done here with respect to creating the spine head? Please bring out clarity.
Answer: We included new subdivisions for clarity.

Comment:
Added here please check.
to seal the top with the vertices still selected. Move the blue arrow down to align to the top of the spine head
Answer: OK.

Comment:
What is being done here and why?
Select **Tool and Knife**. Cut a circle with the knife around the center of the top (**Figure 3B**). Select **Tool** and **Loop Cut and Slide**. Repeat this step four times to create 4 concentric circles around the center of the top (**Figure 3C**).
Answer: We added a NOTE to clarify it.

Comment:
Moved here please check. Why any citations? Please expand the PSD region.
NOTE: The top of the sphere is made flat to approach the PSD region of the spine head.

Answer: We included new references.

Comment: What is being done here with respect to dendritic spine creation?
Press *b* and select the bottom of the mesh (Figure 4C).
Answer: The creation of the spine neck. More subdivisions were added for clarity.

Comment:
Why this number?
Press *e* and *z*, *-0.45* to create an extrusion (Figure 4D).

Answer: This is the z axis position of the extrusion (-0.45). We made it clear in the text.

Comment:
What is being done here with respect to dendritic spine creation?
Press *crtl+t* to triangulate the mesh.
Answer: We clarified it in the text. It is making the mesh compatible with MCell.

Comment:
When did you create duplicates ?
Answer: Doubles can be created accidentally during the process of mesh editing. It means the creation of superposed vertices. We included a NOTE for clarity.

Comment:
Please use imperative tense if this step needs to be filmed. If not this can be converted to a note instead.
2.36 The analysis will print the information in the **Mesh Analysis Panel** and it should be **watertight**, **manifold** and **outward-facing normals**. This step is required to ensure that the created mesh will work on MCell. Otherwise, probably there is a missed a step. In this case, *delete* the mesh and start from step 2.1 again.

Answer: It has been converted to NOTE.

Comment:

Again this section can be divided into creating multiple spines (refer to the top section for this), creating ectrusions for each, joining each to create a dendrite with multiple spines, etc to bring out clarity for the first time readers.
3. Creating a Dendrite with Multiple Spines.

Answer:

We added new sections for clarity.

Comment:

Added here to bring out clarity.

3.1 Generate spine head as described previously in step 2.1-2.4. Press *a* to deselect the spine. Type *shift+c* to centralize the cursor.

Answer:

OK.

Comment:

How did you come up with tis radius and depth.

3.2. Press *shift+a* to open the mesh palette. Select **Mesh** and then **Cylinder**. Change the parameters on the **Add Cylinder** menu: **Radius** *0.3*, **Depth** *2*. Press *enter*.

Answer:

Those are suggested general values for an ordinary dendrite compartment. We included a NOTE for clarity.

Comment:

Where di you type this?

Type *r* and type *90* to rotate the cylinder 90° (Figure 5A). Use the blue arrow to drag the cylinder down to the bottom of the spine. Press *3* on the keypad to have a front view of the cylinder.

Answer: We meant press instead of type. R is the short for rotate and 90 rotates the mesh 90 degrees. I included the shortcut in the glossary. I changed to press r. After that one need to type a number that is 90 in this case.

Comment:

What is being done here and why? Needs clarity.

3.14 Select **Object spine**.

Answer:

Those are steps to insert a spine in the dendrite. I have created a subtitle to indicate it.

Comment:

3.16 Press *a* to deselect the meshes.

From here on to the end of the protocol, please bring clarity as to what is being done and why. Please refer to my previous comments and work accordingly

Answer:

Done.

Comment:

Please move this note right after the step whis action is being performed.
NOTE: In step 3.15, the new mesh will be the join dendrite and spine. The isolated dendrite disappears when the different meshes are combined, but the isolated spine remains overlapping with the new mesh and is used to generate multiple copies of the same spine. Delete all the isolated spines after finishing the mesh. It is critical to have a complete overlap between the spine neck and the dendrite; otherwise, the mesh will not be watertight.

Answer: We moved it.

Comment:
Press *tab* to change to **edit mode**.

The previous section ends with save the file. Before this step please include a step to show that you reopen the file dendrite_with_spines in the xxx software to perform these actions.

Answer: Done.

Comment:
Press *a* to select the whole mesh.
Of what? Where?

Answer:
Of the dendrite with spines. We clarified it.

Comment:
4.4 Press *t* to hide the **CellBlender panel** and visualize the whole mesh in detail.

Where is the whole mesh?

Answer:
The whole mesh is in the main panel. We clarified it.

Comment:
Use + and – on the keypad to zoom in and out or scroll with the mouse.

Why do you need to do this at this stage?
Answer:
It is required to visualize the surface regions.

Comment:
Why this? Does this have any relationship to AMPAR? Citations if any?
Answer:
It is the creation of the AMPARs. This is the diffusion constant of AMPARs in the membrane. We included a citation for the value (Antunes et al., 2019).

Comment:
Citation to support this value?
Answer:
We included a citation for the value (Antunes et al., 2019).

Comment:
5.1, 5.2, 5.3 are same. Please bring out significance
Answer:
They are all part of the procedure of creating molecules. We included more subdivisions for clarity.

Comment:
What is anchor_LTP? Somewhere in the introduction, please define with citations what are anchors, how these are bound to AMPARs, what are anchor_LTP, anchor LTD, etc. why do you need to create all these. This will bring out clarity in the protocol.

Answer:
We included an explanation in the introduction.

Comment:
Please refer to my comments above and reformat accordingly to bring out clarity.

Answer: We did it.

Comment:

Please refer to my comments above and reformat accordingly to bring out clarity

Answer:
Done. We created more subtitles for clarity.

Comment:
Please refer to my comments above and reformat accordingly to bring out clarity.

Answer: Done.

Comment:
Please refer to my comments above and reformat accordingly to bring out clarity.
Answer: Done. We included more subdivisions and NOTES to address it.

Comment:
How do you generate graphs presented in this figure 8 and how do you link it to synaptic plasticity. This is still not clearly explained.

Answer:
We included more Notes to explain it in the protocol.

Comment:
Legend of Figure 8.
How is this measured? Please explain and detail these observations somewhere in the result.

Answer:
It is measured by the observation of the opposite effect at the neighbor spines of the homosynaptic plasticity at a single spine. For instance, homosynaptic LTP induction at a single spine will create a heterosynaptic LTD effect at the neighbor spines. We included this information in the results.

Term	Meaning
Mesh	Object type used in a 3D scene. A mesh is a representation of a geometric domain by smaller discrete cells.
UV Sphere	Mesh mapped to the 3D surface of a sphere.
Knife tool	Blender tool to cut into the mesh along the view axis to interactively subdivide geometry.
Active Object	The object that has been selected.
Object Mode	The standard object interaction mode. Supports basic operations such as object creation, joining objects, etc
Edit Mode	The mode for modifying the components of an existing mesh. Used for the majority of mesh editing operations.
crtl+z	Shortcut for undo changes.
a	Shortcut to select and unselect a mesh.
b	Shorcut to select a part of the mesh with the mouse.
c	Shortcut to cut part of the mesh.
z	Shortcut to select between transparent and solid visualization of the mesh.
Tab	Select between **Edit Mode** and **Object Mode.**
r	Shortcut to rotate the mesh.