Netflux Documentation

*For Netflux version 1.0*

# Overview

Netflux is a user-friendly software for developing dynamic computational models of biological networks. Models are created in Excel format and then simulated using the Netflux graphical interface. No computer programming is required. Netflux is written in MATLAB, but binary versions are available for Windows and MacOS for those that do not have MATLAB installed.

Netflux can export network structures and simulation results for visualization in Cytoscape or other software packages. Netflux can also export MATLAB or Python code for more advanced network simulations. Netflux can import network structures based on SIF or SBML-QUAL formats.

Netflux simulates network models using a logic-based differential equation approach, which we described in detail here:

Kraeutler MJ, Soltis AR, Saucerman JJ. Modeling cardiac β-adrenergic signaling with normalized-Hill differential equations: comparison with a biochemical model. BMC Syst Biol. 2010 Nov 18;4:157.

<https://bmcsystbiol.biomedcentral.com/articles/10.1186/1752-0509-4-157> (open access)

Netflux has been used to develop large-scale computational models of signaling networks for cardiomyocyte hypertrophy, fibroblasts, and mechano-signaling. Netflux has also been used in educational settings from high school (simulations of social networks) through graduate school levels.

Contents

[Overview 1](#_Toc4780678)

[Installation 2](#_Toc4780679)

[Getting Started with Netflux 3](#_Toc4780680)

[Overview of the Netflux Graphical User Interface 5](#_Toc4780681)

[Formatting the Excel network file 7](#_Toc4780682)

[Visualizing a Netflux network model with Cytoscape 10](#_Toc4780683)

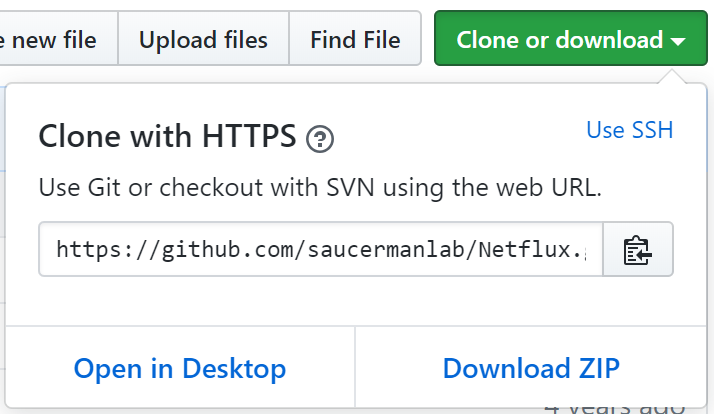
[Visualizing Netflux model simulation results with Cytoscape 11](#_Toc4780684)

[Visualizing Netflux model simulation results with Cytoscape add-on CyAnimator 13](#_Toc4780685)

[Using the SBML-QUAL and SIF Conversion Tools 14](#_Toc4780686)

# Installation

Download Netflux from GitHub (<https://github.com/saucermanlab/Netflux>). Clicking on the button labeled “Clone or download” and then “Download ZIP.”



This ZIP file contains the MATLAB version of Netflux as well as binary versions for Windows or MacOS users that do not have MATLAB installed.

**Users with MATLAB version 2009b or higher**

Extract the “Netflux” folder from within the ZIP file and move it to your desired location.

**Windows users that do not have MATLAB**

Open the “Netflux Compiled” folder and run “Netflux Installer Windows.exe”. This installer will automatically download and install the MATLAB Compiler Runtime, which will take some time as that file is about 800 MB. Netflux will be installed to the C:\Program Files\Netflux folder. Note installation may require administrator privileges on your computer. An exception to Windows firewall may also be needed, but we are checking on this.

**MacOS users that do not have MATLAB**

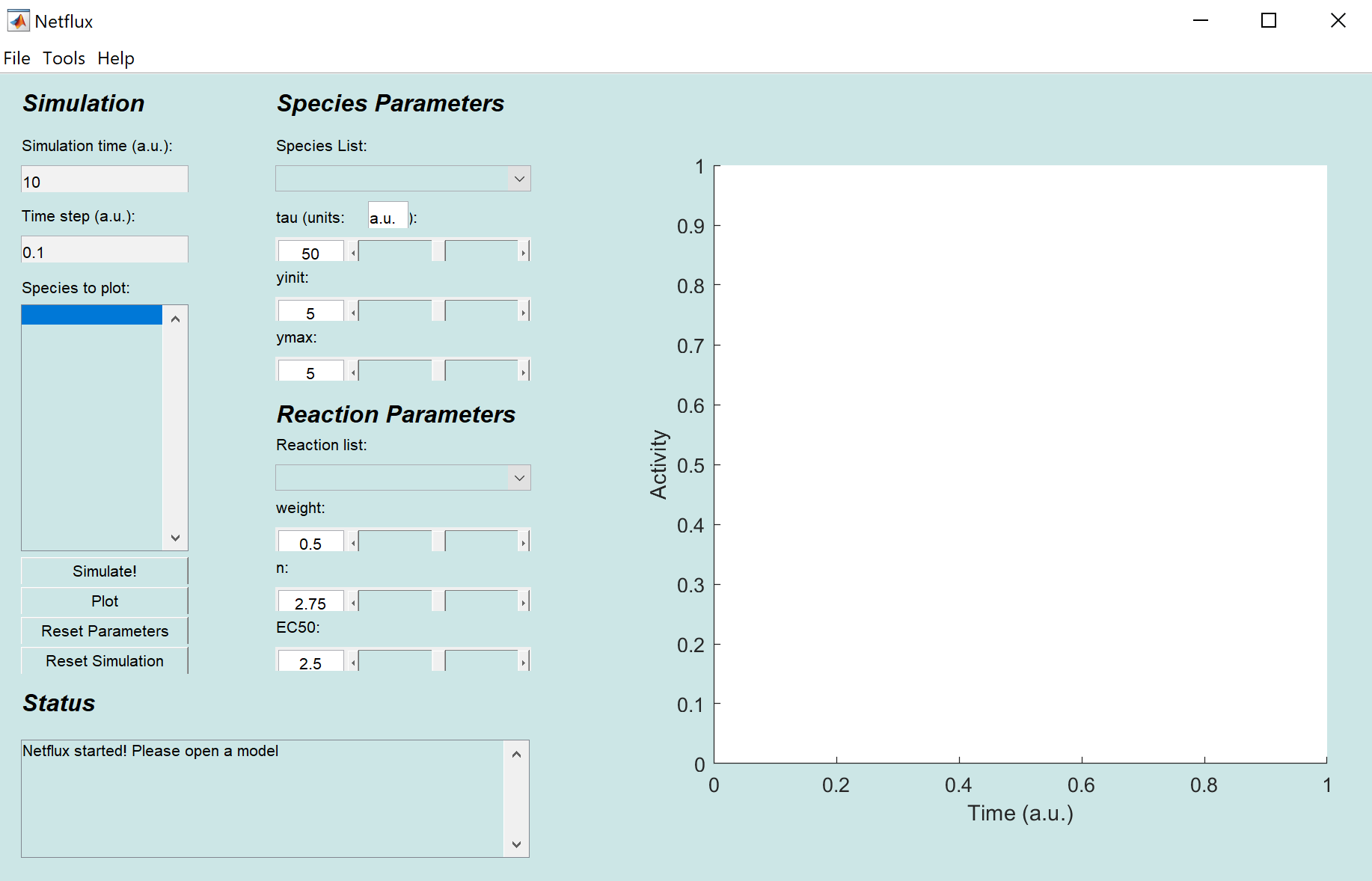
Open the “Netflux Compiled” folder, extract “Netflux\_Installer\_MacOS”, and install “Netflux\_Installer\_MacOS.app”. This installer will automatically download and install the MATLAB Compiler Runtime, which will take some time as that file is about 800 MB. Netflux will be installed to the Applications\Netflux folder.

# Getting Started with Netflux

In this section we’ll get you quickly started running simulations in Netflux with an example network. More details about the Netflux interface are provided in the following section.

1. If you installed the binary Windows or MacOS versions, start Netflux by double clicking the Netflux file. If you are using the MATLAB version, run “Netflux.m” from within MATLAB.
   1. *Note*: The binary versions of Netflux can be slow to load, as they need to also load the MATLAB Compiler Runtime. Be patient and do not double click *Netflux* more than once.

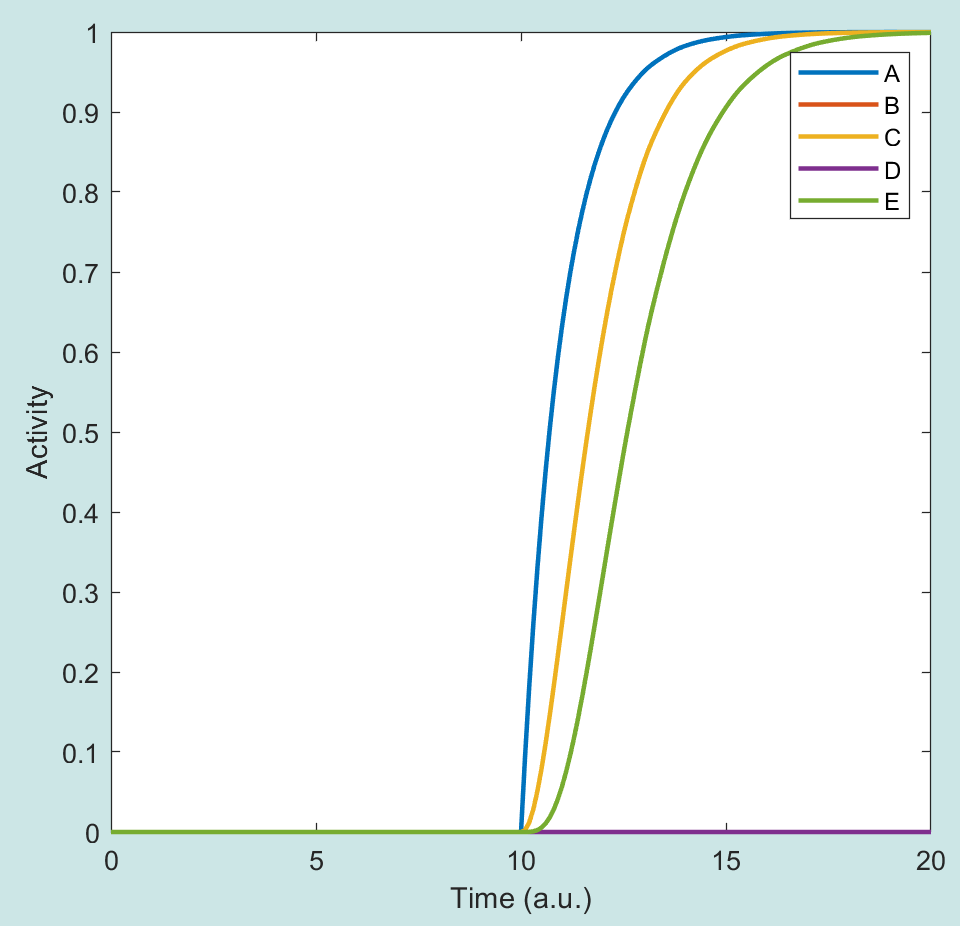
Once Netflux loads, you should see the interface which looks similar to the below image.



1. We’ll start by opening a simple example network model. Select File🡪Open Model. Open “exampleNet.xlsx”, which you can find in the “models” folder. The Status window should indicate that your file was loaded successfully.

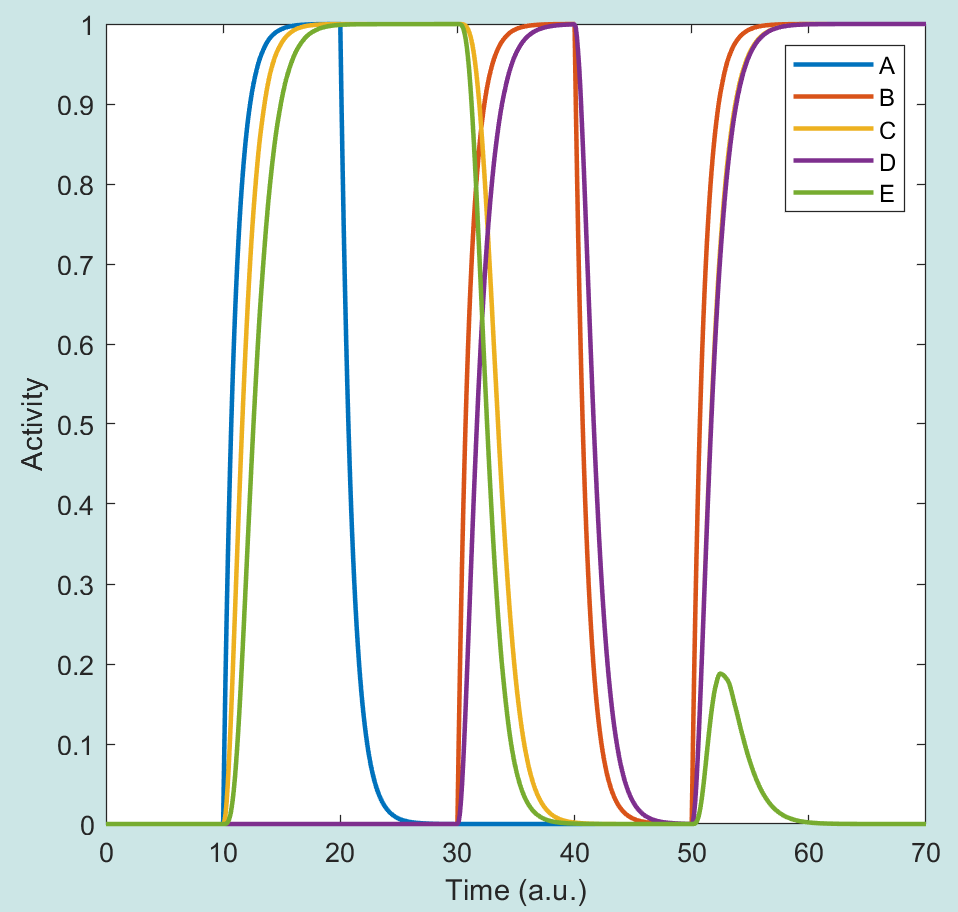
|  |  |
| --- | --- |
| Schematic of the exampleNet model | Netflux interface after loading exampleNet |

1. Press the “Simulate!” button. Congratulations, you have run your first simulation! But as you can see on the plot, nothing happened yet. Click the “Reaction list” to see the reactions that are defined in exampleNet. Change the reaction weight for “r2: => A” to 1, then click “Simulate!” again. You should see that “A” increased.
2. Let’s plot the rest of the species as well. Select all of the species under “Species to plot” using Shift+arrow keys or Ctrl+A. Click “Plot” to replot the simulation results. The plot window should now look like the below figure, showing the predicted sequential activation of A, C, and E.



1. Let’s continue this simulation to make it a bit more interesting. Simulate washout of A by changing the “r2: => A”reaction weight back to 0, and click “Simulate!”. Did the network respond in the way you expected? Why or why not?
2. Simulate stimulation with B by changing the “r1: => B” reaction weight to 1, then click “Simulate!”.
3. Simulate washout of B by changing the “r1: => B” reaction weight back to 0, then click “Simulate!”.
4. Finally, simulate simultaneous stimulation with both A and B by changing of of their input reaction weights to 1, then click “Simulate!” twice.

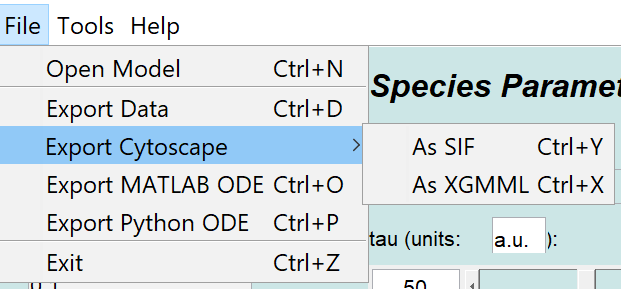
The Netflux plot should now look like the below image. As you can see, we have just simulated a rather complex experiment, which revealed some perhaps unexpected behaviors of this network. Which parts are the simulated data look most interesting to you, and can you explain them based on the network schematic shown above?

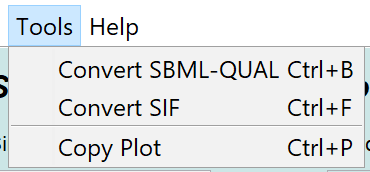


# Overview of the Netflux Graphical User Interface

1. ***Simulation Time*** – Input the time (in seconds) that you want to simulate your reaction. The time starts from the last time point in your previous simulation. For example, if you simulated the reaction for 10 seconds previously, and Simulated again for 10 seconds, the Graph and Data will show for 20 seconds. “Reset Simulation” in order to start from time zero.
2. ***Time Step***—The time interval between plotted solution points.
3. ***Species to Plot*** – The species highlighted will be the species plotted in the graph. Control+click in order to plot multiple species at once. The plotting occurs after you push “Simulate!”
4. ***Simulate***! – Causes the simulation to occur.
5. ***Plot*** – Replot the graph without running through another simulation. For example, you forgot to plot Species B with Species A and you do not want to go through another simulation.
6. ***Reset Parameters*** – Resets all species and reaction parameters.
7. ***Reset Simulation*** – Resets the starting time to zero and clears the graph. Does **NOT** reset the parameters.
8. ***Status Text*** – Displays various messages depending on your actions.
9. ***Species List***  - A drop down menu for you to select which species to alter.
10. ***Tau*** – A slider bar for you to alter the reaction time constant parameter of your selected species.
11. ***Tau Unit Definition –*** A text denoting the units of time for the time constant, which determines the time step, simulation time, and time axes label. Press enter to change the units.
    * *Note*: The units of the time constant determine the units of time in the simulation, but have no effect on the simulations or the solutions because the units are arbitrary.
12. ***Yinit*** – A slider bar for you to alter the initial y parameter of your selected species.
13. ***Ymax*** – A slider bar for you to alter the ‘ymax’ parameter of your selected species.
14. ***Reaction List*** – A drop down menu for you to select which reaction to alter.
15. ***Weight*** – A slider bar for you to alter the reaction weight parameter of your selected reaction.
16. ***n*** – A slider bar for you to alter the hill coefficient parameter of your selected reaction.
17. ***EC50*** – A slider bar for you to alter the half-maximal effective concentration parameter of your selected reaction.

File Menu

1. ***Open Model*** – Loads an Excel file. Doing so will reset all parameters and graphs.
2. ***Export Data*** – Exports the data for each species at intervals defined in the "Time-Step" field into a text file for use in programs such as Cytoscape (<http://www.cytoscape.org/>).
   * Note: the data are space delimited.
3. ***Export Cytoscape*** 
   * As SIF—Exports the network into yournetworkmodel.sif and yournetworkmodel\_nodeAttributes.txt for use in Cytoscape.
   * As XGMML—Exports the network into an XGMML file for use in Cytoscape.
4. ***Export ODE*** – Exports a MATLAB-ready M-file.

Tools Menu

1. ***Convert SBML-QUAL-***Converts an SBML-QUAL file into a Netflux Excel model and SIF/Node Attributes file.
2. ***Convert SIF***-Converts a SIF file to a Netflux Excel model.
3. ***Copy Plot***-Copies the current plot in a separate window.

Help Menu

1. ***About Netflux*** – Displays the About window.

# Formatting the Excel network file

Netflux requires an Excel network file of a certain format. Information is stored in two Excel sheets, ‘reactions’ and ‘species’. To create new networks, it is advised to open ‘exampleNet.xlsx’, save it to a new filename, and then modify the network appropriately. It is advised that Excel spreadsheets are saved as .xlsx file because occasional incompatibilities have been found with older Excel file formats such as .xls. This will ensure that the correct formatting is used and Netflux will be able to read the file correctly. Note that Netflux does not currently have extensive error handling, and incorrect formatting is likely to crash the program; see the end of the page for solutions to common errors.

Some things to keep in mind:

* Data has to start on line 3.
* The preferred file type for Neflux is .xlsx. Some occasional incompatibilities have been found with older Excel file formats such as .xls with the built in MATLAB function xlsread.m. If you are encountering errors with models saved as an .xls file try saving it as an .xlsx file and try again. Files saved as 97-2003 Excel Workbook may or may not read correctly.
* Network names (i.e., the Excel file name) cannot have spaces; use underscores instead.

**Reactions Sheet**

The ‘reactions’ sheet stores the reaction rules, reaction parameters and other reaction-specific information. An example is shown in Figure 2. **Reaction ID’s are stored in column B, reaction rules in column C, reaction weights in column D, hill coefficients in column E, and half-maximal effective concentration in column F. Make sure to name this sheet “reactions”.**

*Reaction ID’s*- this should be a unique name for the reaction with **no spaces or special symbols**

*Reaction Rules*-

1. Input reactions: For an input that activates species ‘A’, use ‘ ‘=> A’.

**Note: Be sure to include an apostrophe before the equals sign, e.g.,** ‘=>**A**

2. Simple activation: If species ‘A’ activates species ‘B’, use ‘A => B’.

3. Simple inhibition: If species ‘A’ inhibits or blocks species ‘B’, use ‘!A => B’.

4. Combinations: If species ‘A’ and ‘B’ are both needed to activate ‘C’, use ‘A & B => C’.

5. Combinations: If species ‘A’ does not inhibit but ‘B’ inhibits ‘C’, use ‘A & !B => C’.

6. Self-activation: If species ‘A’ activates itself, use ‘A => A’.

7. Self-inhibition: If species ‘A’ inhibits itself, use ‘!A => A’.

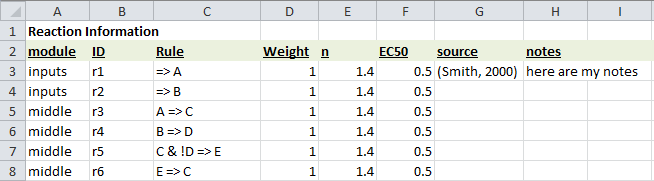
*Weight* – this is a number between 0 and 1 specifying the maximal activity for that pathway. This is normally set at 1. If you would like to fully inhibit a reaction, set the weight to 0. For input reactions (e.g. => I1), this determines the activity of that input (1 for fully activated, 0 for not activated).

*n* – this number is the hill coefficient, which characterizes ligand interactions. The default value for this parameter is 1.4.

*EC50* – this number is the half-maximal effective concentration, which determines at what fraction of the input species would induce half-maximal activation of an output species. The default value for this

parameter is 0.5.

*Notes* – anything to the right of the EC50 column can be used for notes, they are not read by the Netflux program.



**FIGURE 2. The ‘reactions’ sheet from ‘exampleNet.xlsx’.**

**Species Sheet**

The ‘species’ sheet defines the list of ‘species’ to be included in the model, along with associated information. An example is shown in figure 3. **The species ID is stored in column B, species name in column C, Yinit in column D, Ymax in column E, time constant in column F, and type in column G2** **Make sure to name this sheet “species”.**

*Species ID* – Column B is a short unique name for that species that does not contain any spaces. More specifically, ‘ID’ should be an allowable variable name in MATLAB (most users do not have to worry about conflicting naming conventions).

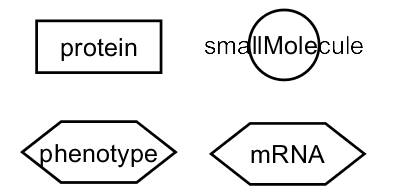
*Species Name* *–* Column C is the ‘name’ field, which is used for the name of the species. This column is currently read but has no use in Netflux; treat it as a note for yourself.

*Yinit* – The initial conditions for your species. The default value is 0.

*Ymax* – The species maximal fractional activation. The default value is 1, but can be changed to reflect changes in protein expression relative to a reference condition1.

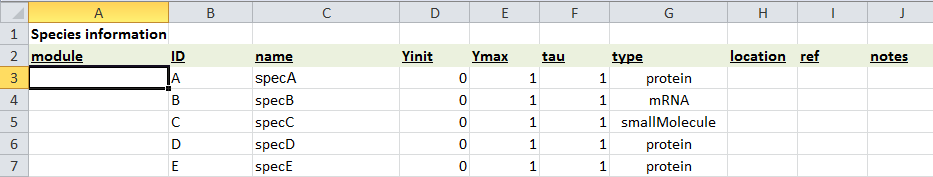
*Tau* – The reaction time constant. The default value is 1.

*Type* – This column defines the node type in the SBGN Process Description format for node formatting in Cytoscape. The SBGN node types and corresponding shapes are shown in Figure 3.



**FIGURE 3**: **Cytoscape node shapes corresponding to the available values in the Type column of the Excel spreadsheet.**

*Notes* – Anything columns to the right of the “type” column can be used for notes, they are not read by Netflux.



**FIGURE 4**: **The ‘species’ sheet from ‘exampleNet.xlsx’.**

Solutions to Common Errors

* Be sure to use the species IDs and not the species names when writing the reaction rules.
* ‘species’ and ‘reactions’ tabs have to be this exact spelling and case.
* Cannot have both regular and bold font in a cell
* UNICODE characters cannot be read
* For Input Species, make sure to input ‘=>A, not just =>A (as this would prompt a function).
* Make sure there are no plots in the sheets.
* For Google Doc Users: After exporting as an Excel file, open up the Worksheet, and re-save it as a .xlsx file using another spreadsheet program, such as Microsoft Excel or OpenOffice.
  + Reason: Exporting from Google seems to save the data into Unicode, which Netflux does not support. Saving the file again using another spreadsheet program will remedy this error.
* For OpenOffice Users: Save as an .xlsx file types
* If the model does not load, try copying and pasting the data from each sheet into a new .xlsx file and resaving. If that doesn't work, try saving as .xls instead of .xlsx.

# Visualizing a Netflux network model with Cytoscape

We will use the software package Cytoscape (<https://cytoscape.org/>) to visualize the network structure of models created in Netflux. Detailed information on the use of Cytoscape can be found at

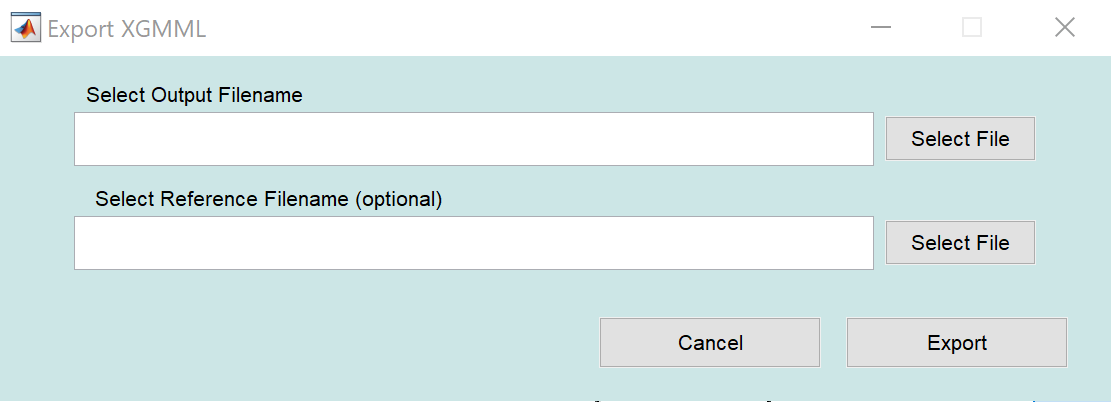
<http://manual.cytoscape.org>

The visual layout we use is similar to the recommendations made for the Systems Biology Graphical Notation for “Activity Flow” diagrams (<http://sbgn.github.io/sbgn/>).

Netflux can export models in either of two Cytoscape file formats: XGMML or SIF. However these instructions focus on the preferred XGMML format, as it stores the network structure, properties, and layout information all in a single file.

Within Netflux

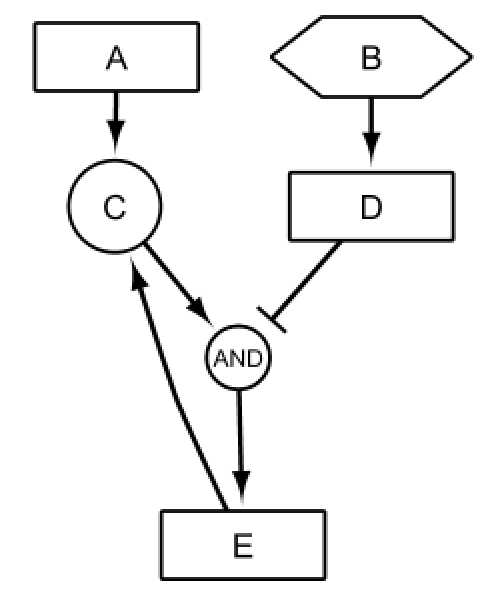
1. Select File🡪Export Cytoscape🡪As XGMML. The following dialogue will be shown:



1. Under “Select Output Filename”, “Select File” to specify the name and location of the XGMML file to be exported.
2. If this is a new network visualization, no entry is required for “Select Reference Filename”. If this is a revised visualization based on a pre-existing XGMML file, select that file at “Select Reference Filename”.
3. Select "Export". You should see a “Model exported to:” message in the Netflux status window.

Within Cytoscape

1. Select File 🡪 Import 🡪 Network from File..., and select the XGMML file that you exported from Netflux.
2. Import the SBGN layout style by selecting File 🡪 Import 🡪 Styles from File…, and select “Netflux SBGN styles.xml” from the installed Netflux directory.
3. Apply the “Netflux SBGN” style by going to the Style tab within the Control Panel. Click on the dropdown menu that says “default”, and change to “Netflux\_SBGN”.
4. Apply the desired layout to the network from the Layout menu. We typically use Layout 🡪 “yFiles Hierarchic Layout”, manually reposition the nodes, and then Layout 🡪 yFiles Organic Edge Router. Node layout can also be adjusted further with Layout 🡪 Node Layout Tools. After some adjustment, your network will now look similar to the below image.



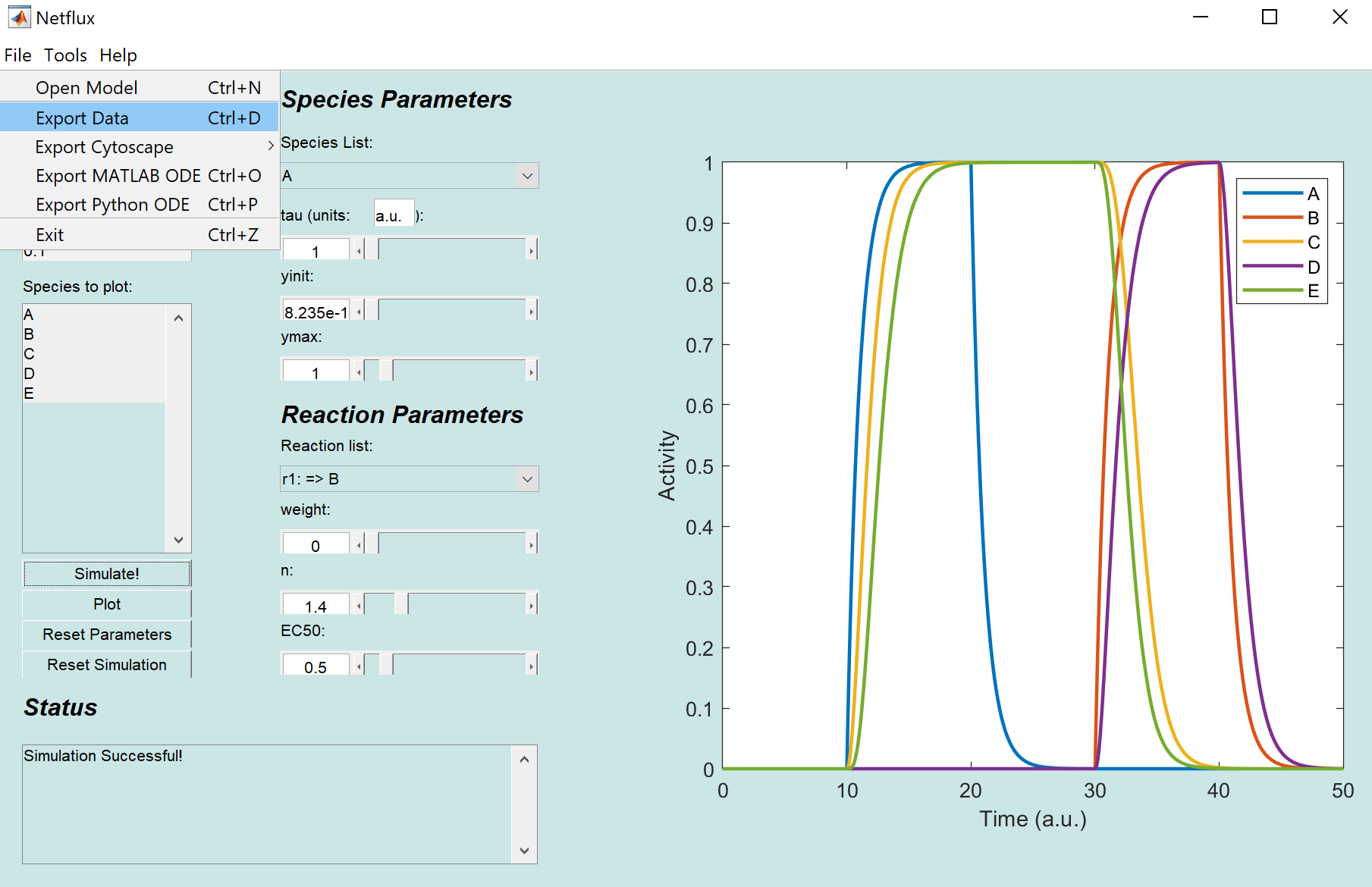
1. Save your Cytoscape session as a .cys file.
2. If you would like to use this network visualization as a Reference for a future model revision, be sure to also export your network in XGMML format using File 🡪 Export 🡪 Network to File…, select XGMML and the desired filename.

# Visualizing Netflux model simulation results with Cytoscape

We can directly visualize Netflux simulation results in Cytoscape by first exporting simulation results in Netflux and then importing those results into a network created using the instructions in the section [Visualizing a Netflux network model with Cytoscape](#_Visualizing_a_Netflux).

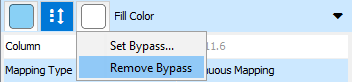
Within Netflux

1. Load a Netflux model and run a simulation (see section **Quick Start**).
2. Select File 🡪 Export Data and save the file in .txt format.

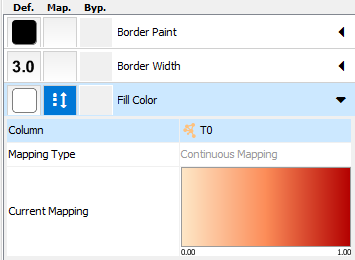


Within Cytoscape

1. Select File 🡪 Import 🡪 Table from File..., and select the .txt data file that you exported from Netflux.
2. Select Advanced Options 🡪 Delimiter: SPACE 🡪 OK 🡪 OK. The Node Table (below the network) will now have additional columns corresponding to the timepoints (e.g. T0, T59.8) that were exported from the Netflux simulation.
3. Select all the nodes using Ctrl+A or Select 🡪 Select All Nodes and Edges. In the Control Panel 🡪 Style, open the “Fill Color” panel. Select the Byp. Icon and then “Remove Bypass”.



1. Select the Map. icon, change “Mapping Type” to “Continuous Mapping”, “Column” to the data corresponding to the desired timepoint (e.g. T0, T59.8). If you would like to change the color scheme, double click on the gradient. Note that Cytoscape’s Continuous Mapping Editor seems to be buggy, and we typically manually adjust the gradients.



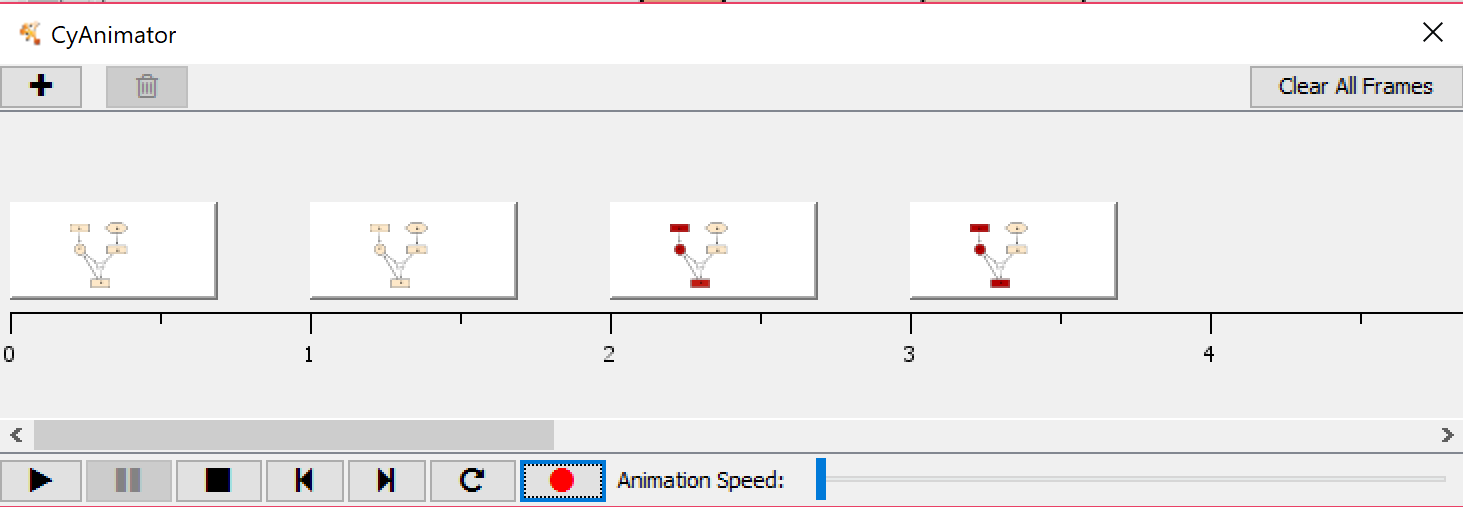
Below are some examples of the visualization of the exampleNet model at various timepoints based on the simulation shown above.

|  |  |  |  |
| --- | --- | --- | --- |
| Time = 0 | Time = 11.5 | C:\Users\jjs3g.BME-PF0YS0YK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\exampleNet simulation T11.5.png  Time = 25 | C:\Users\jjs3g.BME-PF0YS0YK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\exampleNet simulation T35.png  Time = 35 |

# Visualizing Netflux model simulation results with Cytoscape add-on CyAnimator

The Cytoscape plugin CyAnimator (<http://apps.cytoscape.org/apps/cyanimator> or) can be used to create animations of your simulation results.

1. To install CyAnimator, within Cytoscape go to Apps 🡪 App Manager, search for CyAnimator, and install.
2. Import Netflux simulation results and visualize individual timepoints as described in the section [Visualizing Netflux model simulation results with Cytoscape](#_Visualizing_Netflux_model).
3. Open CyAnimator, which should be found under the Apps menu.
4. In the Style tab, change the Fill Color Mapping Column to “T0”. In the CyAnimator window, click  to create a frame corresponding to the initial timepoint.
5. Repeat step #4, but changing the Column to the desired timepoint.
6. Click  to examine your animation.
7. Click  to export your animation as an animated GIF, MP4, or series of individual image frames.



*Notes on use of CyAnimator*

* Generally, you should add timepoints at a constant time interval (e.g. T0, T5, T10, T15). CyAnimator will interpolate in between the saved frames. This means that timepoints in between the frames added to CyAnimator are estimated and may not accurately reflect the simulation. During periods of rapid change in your network you may want to add frames at higher resolution, but positioning these correctly within CyAnimator can be tricky.
* CyAnimator frames are saved when you save a Cytoscape session (.cys), in case you would like to later edit your frames.

Documentation for CyAnimator is available at <http://www.rbvi.ucsf.edu/cytoscape/CyAnimator/index.shtml>

These instructions were tested with Cytoscape version 3.7.1. and CyAnimator version 2.1.0.

# Using the SBML-QUAL and SIF Conversion Tools

The SBML-QUAL conversion tool is compatible with non-metabolic models from the Path2Models Database. This tool converts the SBML-QUAL file into a Netflux excel model, which can be simulated and exported to Cytoscape using the Netflux GUI. A .sif file is also autogenerated from the conversion for direct visualization in Cytoscape.

1. Download an SBML-QUAL file from the non-metabolic section of the Path2Models Database: <http://www.ebi.ac.uk/biomodels-main/path2models?cat=non-metabolic>
2. Select

Tools🡪 Convert SBML-QUAL

Or

Tools🡪 Convert SIF

1. Select the SBML-QUAL or SIF file from step 1 and specify an output folder:
2. Click Convert.
3. The Netflux Excel model will be exported to the location you specified in step 3.

*Documentation last updated 3/29/2019 by Jeff Saucerman*