

Slides, videos, links and more:

<https://github.com/physicell-training/02-How-to-nanoHUB>

Lesson 2: How to use a PhysiCell nanoHUB app

Paul Macklin, Ph.D.

 @MathCancer

PhysiCell Project

last updated: November 13, 2019



LUDDY

SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING

PhysiCell Project

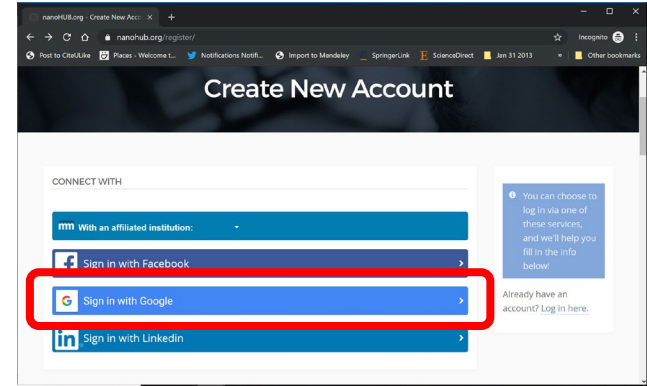
PhysiCell.org

 @PhysiCell

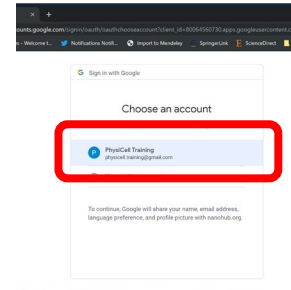
nanoHUB Account

- These tutorials use cloud-hosted PhysiCell models on [nanoHUB.org](https://nanohub.org).
- nanoHUB is **free**, but it requires a one-time registration.
- **Steps:**
 1. Visit <https://nanohub.org/register>
 2. Choose "Sign in with Google"
 3. Choose a Google account
 4. Click "No" (so it doesn't try to associate with some other nanoHIB account)
 5. Finish filling in details, and you're done!
 6. Use your google account for future logins.

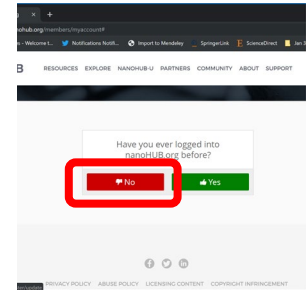
2



3



4



Goals

- Many of the PhysiCell lessons are paired with **educational microapps**:
targeted cloud-hosted models designed to illustrate a core PhysiCell concept.
- All PhysiCell-powered nanoHUB apps share common features.
- This brief lesson will walk you through those features.

Tools versus Apps

- After considerable debate, the nanoHUB community arrived at these definitions:
 - A **tool** is an application with broader feature sets and less constrained use cases. A tool tends to require more extensive training to understand the full feature set.
 - ♦ Future PhysiCell-powered nanoHUB resources may be tools if they allow general-purpose modeling using PhysiCell as the engine.
 - An **app** is purpose-driven model or constrained tool with relatively constrained and focused use cases. Users can generally learn an app in one or two unguided sessions.
 - ♦ Most PhysiCell-powered nanoHUB resources are apps.
- There may be blurred distinctions between tools and apps in some cases, such as PhysiCell apps that can simulate a broad variety of nanotherapies.

Sample nanoHUB app: biorobots

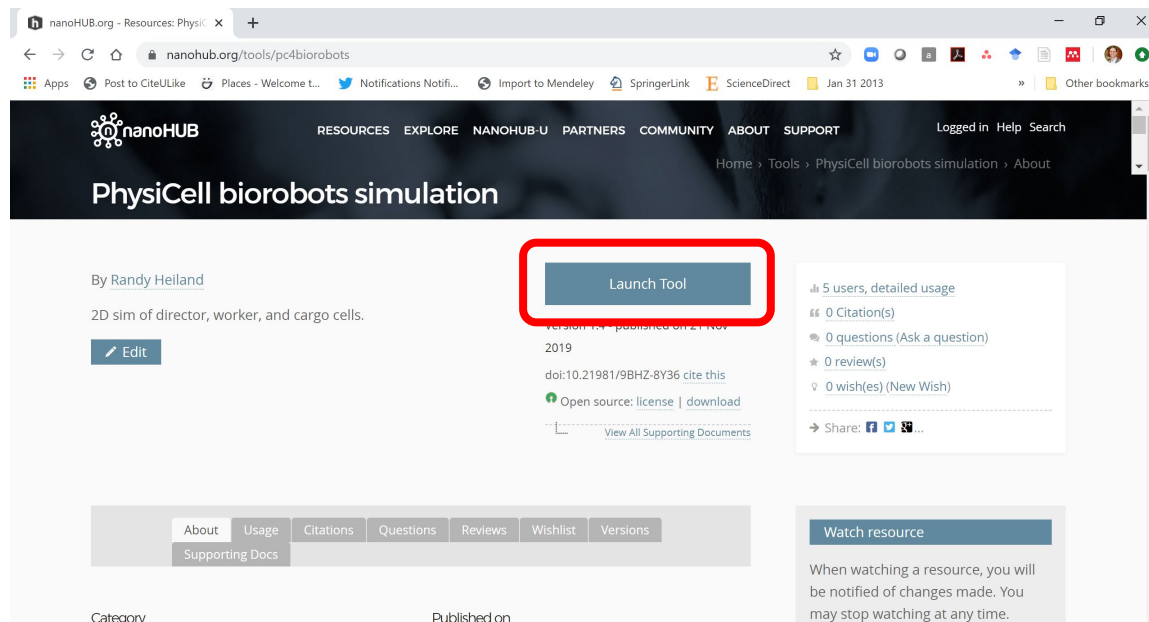
- This sample model explores how engineered cells could be used as a **multicellular cargo delivery system**.
- **Director cells:** Release a chemoattractant to guide delivery of cargo
- **Cargo cells:** Release a chemoattractant to guide worker cells until they are attached.
- **Worker cells:** When unattached to cargo, use chemotaxis to seek cargo cells and adhere to them.

When attached to cargo, use chemotaxis to drag cargo towards directors for delivery.

Splash screen and launching the tool

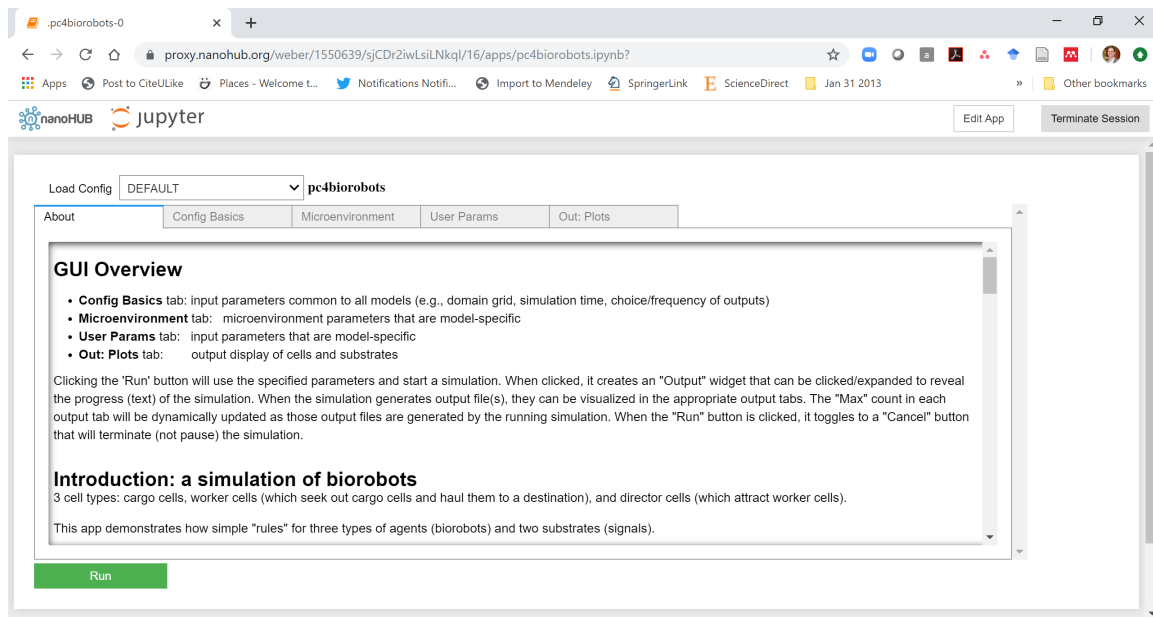
- Each PhysiCell-powered nanoHUB app has a splash page:

- Background information on the tool and model
- Use metrics
- Citation information
- A **LAUNCH TOOL** button. Click this!



About

- The **about** tab gives *in-app* documentation about the mathematics of the model:
 - background
 - diffusing substrates
 - key cell types and behaviors
- This tab also describes the main user parameters of the model.



Configuration basics

- The **config basics** tab lets you set up the simulation:
 - Set the domain size and spatial resolution
 - ♦ Might be disabled in some apps.
 - Set the final simulation time
 - Set the number of threads (capped at 4 on nanoHUB)
 - Specify how often to save cell plots
 - Specify how often to save diffusing substrate data

The screenshot shows a web browser window with the URL `proxy.nanohub.org/weber/1550639/sjCDr2iwLsiLNkq/16/apps/pc4biorobots.ipynb?`. The page features a header with the nanoHUB and Jupyter logos, and buttons for 'Edit App' and 'Terminate Session'. Below the header, a 'Load Config' dropdown is set to 'DEFAULT', and the app name 'pc4biorobots' is displayed. The 'Config Basics' tab is active, showing a form for domain and simulation parameters. The 'Domain (micron):' section includes input fields for Xmin (-1000), Xmax (1000), Ymin (-1000), Ymax (1000), dx (20), and dy (20). The 'Max Time' is set to 2880 min, and the '# threads' is set to 4. The 'Plots:' section has checkboxes for 'Cells' and 'Substrates', both of which are checked. The 'Cells' plot is set to 'every 5 min' and the 'Substrates' plot is set to 'every 10 min'. A green 'Run' button is located at the bottom of the configuration form.

Microenvironment

- The **microenvironment** tab lets you set diffusion options:
 - For each substrate, set:
 - ◆ diffusion coefficient
 - ◆ decay rate
 - ◆ initial condition
 - » applied uniformly throughout the domain
 - ◆ Dirichlet boundary condition
 - » applied uniformly to the boundary
 - » "uncheck" to disable this and revert to Neumann (zero flux) conditions
 - Enable or disable calculation of substrate gradients
 - ◆ needed for cell chemotaxis
 - Enable or disable full tracking of internalized substrates in each cell
 - ◆ needed for some models

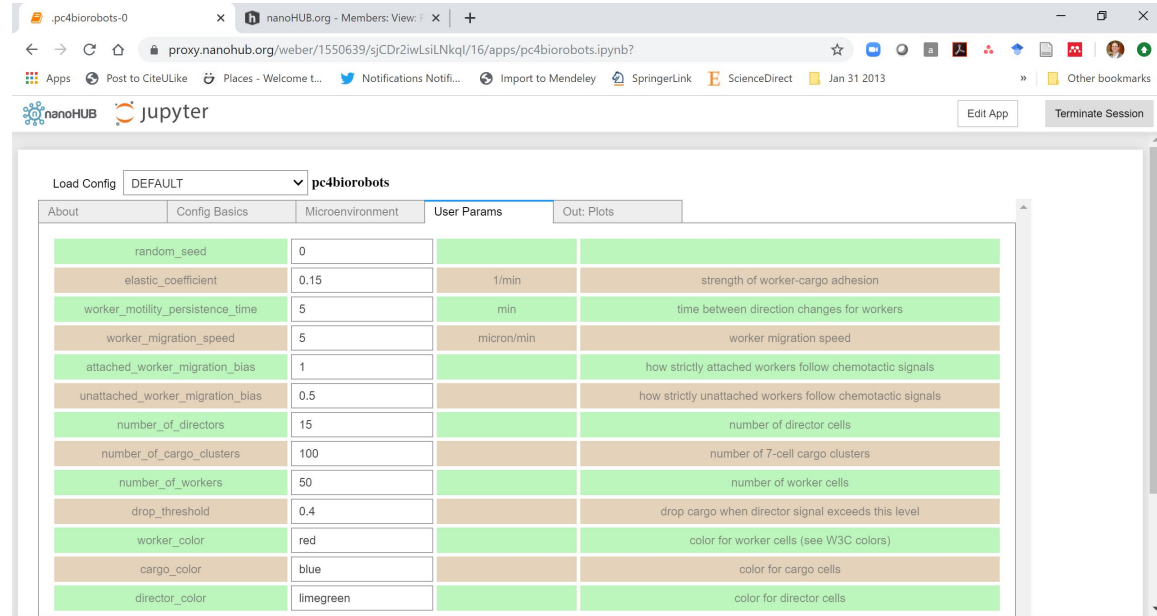
The screenshot shows a web browser window with the URL `proxy.nanohub.org/weber/1550639/sjCDr2iwLsiLNkq/16/apps/pc4biorobots.ipynb?`. The page displays the configuration interface for the `pc4biorobots` model. The `Microenvironment` tab is selected, showing settings for two substrates: `director_signal` and `cargo_signal`.

Substrate	diffusion_coefficient	decay_rate	initial_condition	Dirichlet_boundary_condition	on/off
director_signal	1000	0.1	0	1	<input type="checkbox"/>
cargo_signal	1000	0.4	0	1	<input type="checkbox"/>

Below the substrate settings, there are checkboxes for `calculate_gradients` (checked) and `track_in_agents` (unchecked).

User parameters

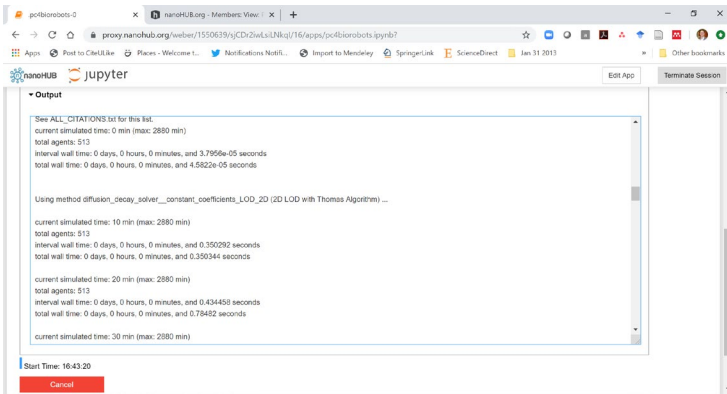
- The **user parameters** tab is where you set parameters that are specific to the app.
- Each row is a parameter:
 - parameter name
 - parameter value
 - units
 - short parameter description
- Consult the **about** tab for more mathematical information on the parameters



Load Config	DEFAULT	pc4biorobots		
About	Config Basics	Microenvironment	User Params	Out: Plots
random_seed	0			
elastic_coefficient	0.15		1/min	strength of worker-cargo adhesion
worker_motility_persistence_time	5		min	time between direction changes for workers
worker_migration_speed	5		micron/min	worker migration speed
attached_worker_migration_bias	1			how strictly attached workers follow chemotactic signals
unattached_worker_migration_bias	0.5			how strictly unattached workers follow chemotactic signals
number_of_directors	15			number of director cells
number_of_cargo_clusters	100			number of 7-cell cargo clusters
number_of_workers	50			number of worker cells
drop_threshold	0.4			drop cargo when director signal exceeds this level
worker_color	red			color for worker cells (see W3C colors)
cargo_color	blue			color for cargo cells
director_color	limegreen			color for director cells

Running

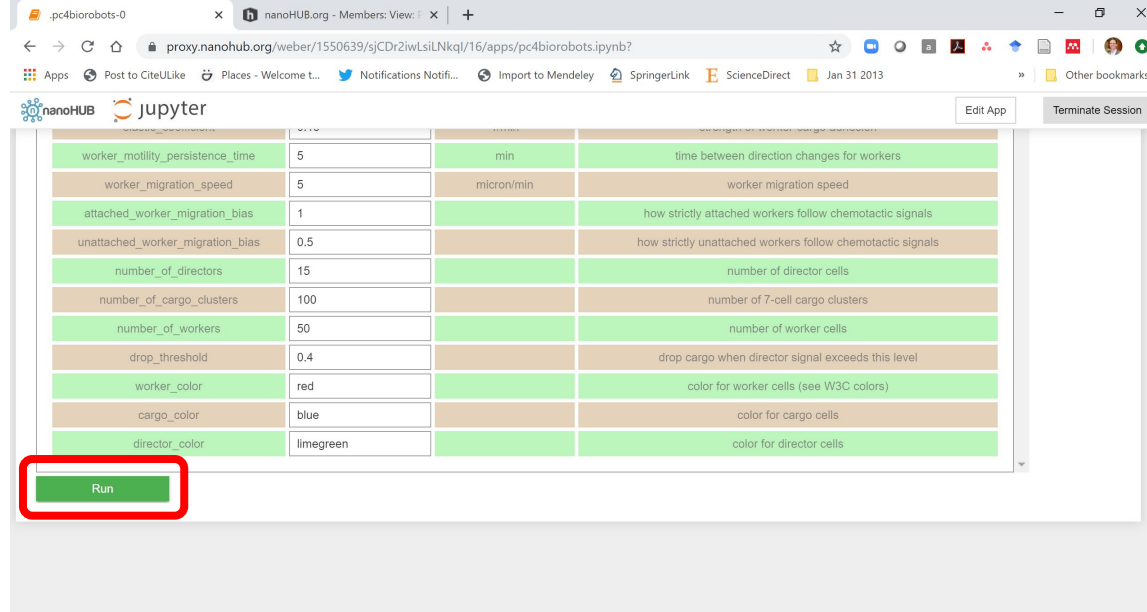
- Once ready, click **Run**
- The standard model outputs will start appearing below
- Click **Cancel** to end early and choose new parameters



The screenshot shows the nanoHUB Jupyter interface. On the left, there is an 'Output' window displaying simulation results. The output text includes:

```
See ALL CITATIONS for this list.  
current simulated time: 0 min (max: 2880 min)  
total agents: 513  
interval wall time: 0 days, 0 hours, 0 minutes, and 3.7956e-05 seconds  
total wall time: 0 days, 0 hours, 0 minutes, and 4.5822e-05 seconds  
  
Using method diffusion_decay_solver___constant_coefficients_LOD_2D (2D LOD with Thomas Algorithm) ...  
  
current simulated time: 10 min (max: 2880 min)  
total agents: 513  
interval wall time: 0 days, 0 hours, 0 minutes, and 0.350292 seconds  
total wall time: 0 days, 0 hours, 0 minutes, and 0.350344 seconds  
  
current simulated time: 20 min (max: 2880 min)  
total agents: 513  
interval wall time: 0 days, 0 hours, 0 minutes, and 0.434456 seconds  
total wall time: 0 days, 0 hours, 0 minutes, and 0.79462 seconds  
  
current simulated time: 30 min (max: 2880 min)
```

At the bottom left, there is a 'Start Time: 16:43:20' and a red 'Cancel' button.

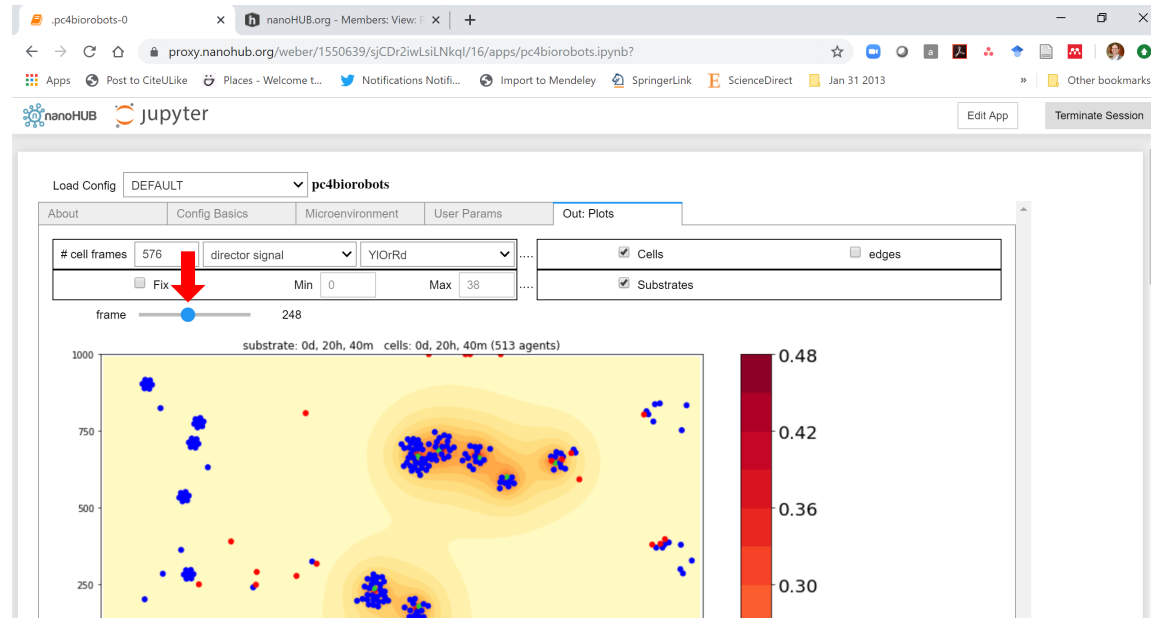


The screenshot shows the nanoHUB Jupyter interface with a table of parameters. The 'Run' button is highlighted with a red rectangle.

Parameter	Value	Unit	Description
worker_motility_persistence_time	5	min	time between direction changes for workers
worker_migration_speed	5	micron/min	worker migration speed
attached_worker_migration_bias	1		how strictly attached workers follow chemotactic signals
unattached_worker_migration_bias	0.5		how strictly unattached workers follow chemotactic signals
number_of_directors	15		number of director cells
number_of_cargo_clusters	100		number of 7-cell cargo clusters
number_of_workers	50		number of worker cells
drop_threshold	0.4		drop cargo when director signal exceeds this level
worker_color	red		color for worker cells (see W3C colors)
cargo_color	blue		color for cargo cells
director_color	limegreen		color for director cells

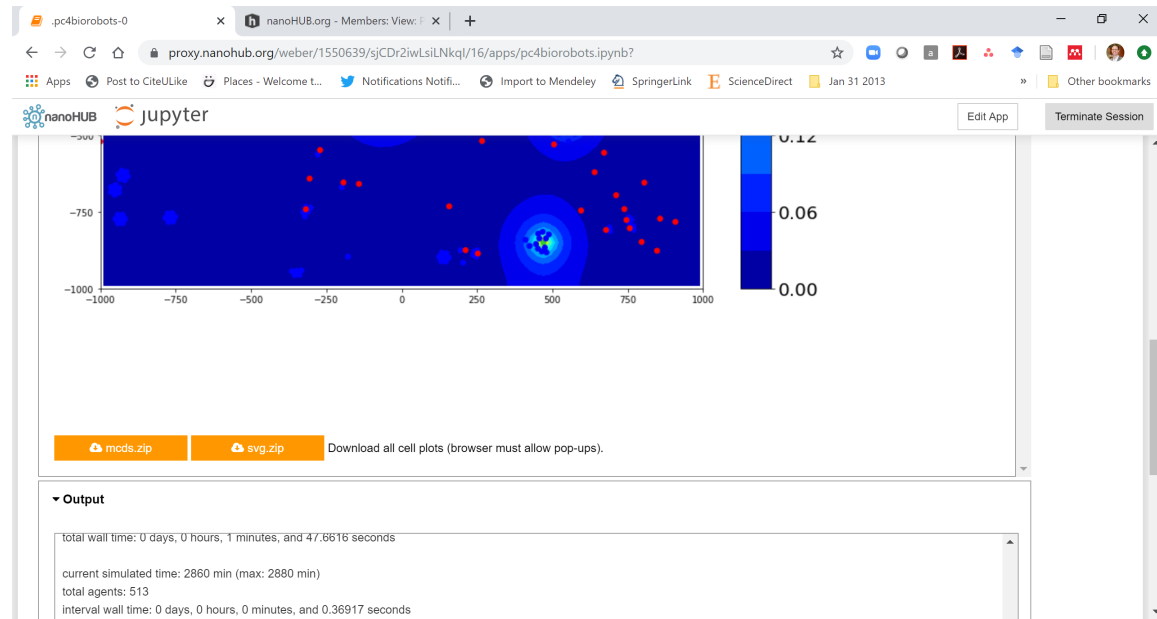
Simulation output

- The **Out: Plots** tab shows the simulation outputs as it runs.
- Slide the "frame" bar to advance time in the simulation.
- Use the dropdown box to choose a different substrate for visualization.
 - Use "fix" to manually set the plot range
 - Click the righthand dropdown box to change coloring (YlOrRd, viridis, jet, ...)
- The checkboxes on the right let you choose plotting options for the cells.
- **Note:** Some older apps may split the output into separate "substrate" and "cell" tabs, rather than overlay them.



Downloading simulation plots and data

- At the bottom of the output tab, you can find options to download your simulation data.
- Click **svg.zip** to download all the SVG cell plots as a single ZIP file.
- Click **mcids.zip** to download all the simulation data (in MultiCellIDS format) as a single ZIP file.
- **Note:** These download buttons only work after a simulation has finished running.
- See the [PhysiCell-Tools](#) ecosystem for tips on loading simulation data in Matlab, Python, and other environments.



Live demo



LUDDY

SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING

PhysiCell Project

PhysiCell.org

 **@PhysiCell**

Next steps

Super fast:

link:

Please proceed to 4 (Introduction to PhysiCell)
<https://github.com/physicell-training/04-PhysiCell-intro>

Intermediate:

link:

Please proceed to 4 (Introduction to PhysiCell)
<https://github.com/physicell-training/04-PhysiCell-intro>

Full training:

link:

Please proceed to 3 (What is an agent-based model?)
<https://github.com/physicell-training/03-What-is-ABM>

More lessons: <https://github.com/physicell-training/master-list>



LUDDY

SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING

PhysiCell Project

PhysiCell.org

@PhysiCell

Credits

Lesson Planning:	Paul Macklin, Drew Willis*, Randy Heiland
Slides:	Paul Macklin, Drew Willis, Randy Heiland
Recording:	Paul Macklin, Drew Willis*
Post-production:	Paul Macklin, Drew Willis*, Kali Konstantinopoulos*
Microapps:	https://www.nanohub.org/tools/pc4biorobots

* denotes undergraduate researcher

Funding:

PhysiCell Development:

- Breast Cancer Research Foundation
- Jayne Koskinas Ted Giovanis Foundation for Health and Policy
- National Cancer Institute (U01CA232137)
- National Science Foundation (1720625)

Training materials:

* Administrative supplement to NCI U01CA232137 (Year 2)



JAYNE KOSKINAS
TED GIOVANIS

Foundation for
Health and Policy



NATIONAL
CANCER
INSTITUTE



BCRF



LUDDY

SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING

PhysiCell Project

PhysiCell.org

@PhysiCell