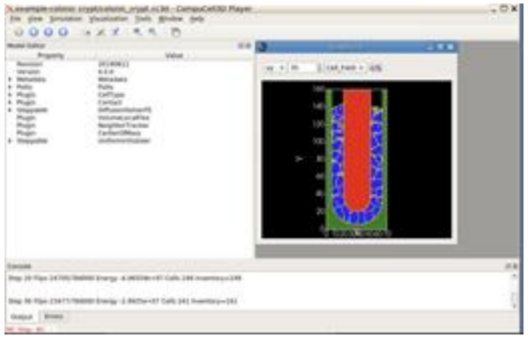
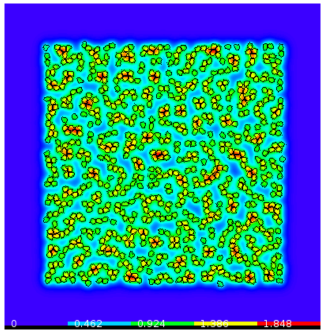
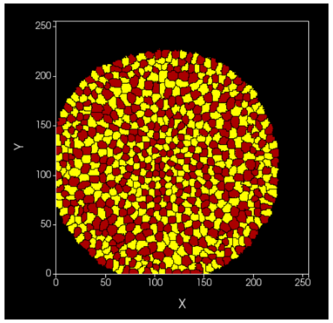
CompuCell3D Workshop: Module 6.4: Parameter Scans and Data Wrangling

Lorenzo Veschini
RA Glazier Lab
lovesc@iu.edu



- This slide deck: https://tinyurl.com/StudentMats24
- Please submit questions/concerns/suggestions via zoom chat
- Please download exercises, slides, cheat-sheets and supporting materials before we start
- Hackathon Idea Sharing Documents: https://tinyurl.com/HackathonDoc2024
- Please join the workshop Slack https://tinyurl.com/CC3DSLACK2024
- Workshop will be live-streamed, recorded and distributed on YouTube
- Please be sure you have a working nanoHUB account

Funding: NIH U24 EB028887, NSF 2120200, NSF 2000281, NSF 2303695
Previous Funding: NIH R01 GM1224241, NSF 1720625

Tentative Course Outline

Times and contents may be adjusted depending on participant feedback

Day 0 Sunday July 28th, 2024: Python Bootcamp
- 0.1 Python Basics [Hayden Fennell]
- 0.2 Lists, Tuples, Strings, Slicing [Hayden Fennell]
- 0.E Mid-Day Q&A Session [Hayden Fennell]
- 0.3 Dictionaries, Functions, Scope, Modules [Hayden Fennell]
- 0.4 Classes, Attributes, Numpy, Scipy, Operations [Hayden Fennell]

Day 1 Monday July 29th, 2024: Basics of CC3D
- 1.1 Introduction to CC3D and Overview [James Glazier]
- 1.2 Intro to Virtual Tissue Modeling [James Glazier]
- 1.E CC3D Examples [Instructor Model Presentations]
- 1.3 CC3D Basics & Intro to Cellular Potts Modeling [Julio Belmonte]
- 1.4 Example Simulation Activities [Joel Vanin; Hayden Fennell; Pedro dal Castel]

Day 2 Tuesday July 30th, 2024: Principles of Modeling, Contact Energies, Cell Growth and Division
- 2.1 Translating Biology into Models [James Glazier]
- 2.2 Cell Properties & Constraints [Gilberto Thomas]
- 2.E CC3D Examples [Instructor Model Presentations]
- 2.3 Cell Growth & Division; Plotting [Gilberto Thomas]
- 2.4 Miscellaneous Features Session [Gilberto Thomas; Peter Fyffe]

Day 3 Wednesday Aug. 3 2023: Chemical Fields, Diffusion, Secretion and Uptake
- 3.1 Chemical Gradients: Diffusion [Pedro dal Castel]
- 3.2 Chemical Gradients: Secretion & Absorption Part 2 [Pedro dal Castel]
- 3.E Turning a Simulation into a Python-only Jupyter Notebook [Lorenzo Veschini]
- 3.3 CC3D Python API [T.J. Sego]
- 3.4 Model-Building Exercise: Avascular Tumor [Pedro dal Castel]

Tentative Course Outline

Times and contents may be adjusted depending on participant feedback

Day 4 Thursday Aug. 1, 2024: Chemotaxis, Chemokinesis and External Potentials
- 4.1 External Potential & Forces on Cells [Gilberto Thomas]
- 4.2 Chemotaxis [Pedro dal Castel]
- 4.E Model-Building Activity: Angiogenesis [Pedro dal Castel]
- 4.3 Breakout Sessions [Various Presenters]
- 4.4 Creating Initial Cell Layouts [Jim Sluka]

Day 5 Friday Aug. 2, 2024: Network Modeling
- 5.1 Network Modeling with Tellurium/Antimony: Part 1 [James Glazier]
- 5.2 Network Modeling with Tellurium/Antimony: Part 2 [James Glazier]
- 5.E Hackathon Prep & Lightning Talks [Hayden Fennell]
- 5.3 Breakout Sessions [Various Presenters]
- 5.4 Boolean Network Modeling [T.J. Sego]

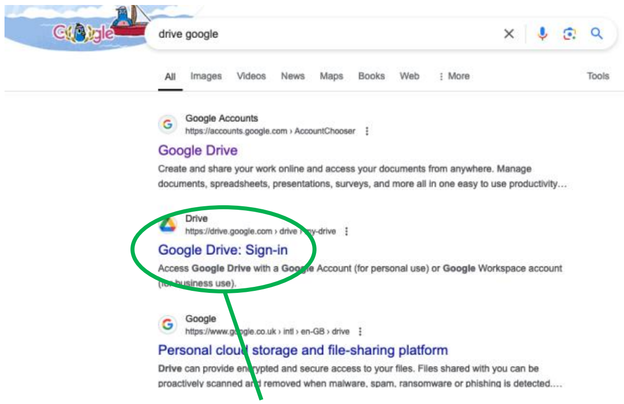
Day 6 Saturday Aug. 3, 2024: Connecting Cells, Data Wrangling, and Best Practices
- 6.1 Compartmental Cells [Julio Belmonte]
- 6.2 Links and FPP [Julio Belmonte]
- 6.E Hackathon Prep & Lightning Talks [Hayden Fennell]
- 6.3 Data Wrangling & Parameter Scans [Lorenzo Veschini]
- 6.4 Best Practices for Parameterization [Jim Sluka]

Day 7 Sunday Aug. 4, 2024: Chemotaxis, Chemokinesis and External Potentials
- 7.1 Vector Fields [Gilberto Thomas]
- 7.2 Planar Cell Polarity [Julio Belmonte]
- 7.E Hackathon Prep & Lightning Talks [Hayden Fennell]
- 7.3 Tissue Folding[Priyom Adhyapok]
- 7.4 Summary, Wrap-Up, & Hackathon Prep/Announcements [James Glazier; Hayden Fenne

Module 6.3 Learning Objectives

1) **Getting ready to work with Colab/Jupyter Notebooks**
    1) Using Gdrive
    2) Access Google Colab
    3) Install all packages in Colab
    4) Local Conda install: Have you linked your env to theJupyter kernel?
    5) Run Jupyter Lab
2) **Create a Python only version of a simulation**
    1) When & why is it useful?
    2) The Angiogenesis simulation
    3) The AngioSim class
    4) The __init__() method
    5) The run() method
3) **Extracting metrics from simulations**
    1) Useful metrics for the Angiogenesis simulation
    2) The AnalysisSteppable()
4) **Running simulations GUI less and in batches (one possible way)**
    1) The AngioSimBatch class
    2) The Instance_exec() method
    3) The run_batch_serial() method
    4) The run_batch_paralle() method
5) **Strategies for parameter searches**
    1) The one_at_a_time() method
    2) The lhs_sampling() method
    3) Fancier methods?
6) **Looking at the data**
7) **Parallel execution**
    1) Why do we want parallel execution?
    2) The AngioSim_PS.py script
8) **Wrapping up simulation results & plotting**

Activity 0: Copy ref code to your GDrive



drive google

All · Images · Videos · News · Maps · Books · Web · ⋮ More · Tools

Google Accounts
https://accounts.google.com › AccountChooser

Google Drive
Create and share your work online and access your documents from anywhere. Manage documents, spreadsheets, presentations, surveys, and more all in one easy to use productivity…

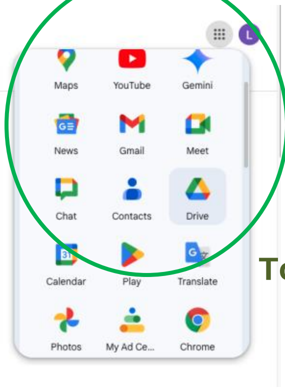
Drive
https://drive.google.com › drive › my-drive

Google Drive: Sign-in
Access **Google Drive** with a **Google** Account (for personal use) or **Google** Workspace account (for business use).

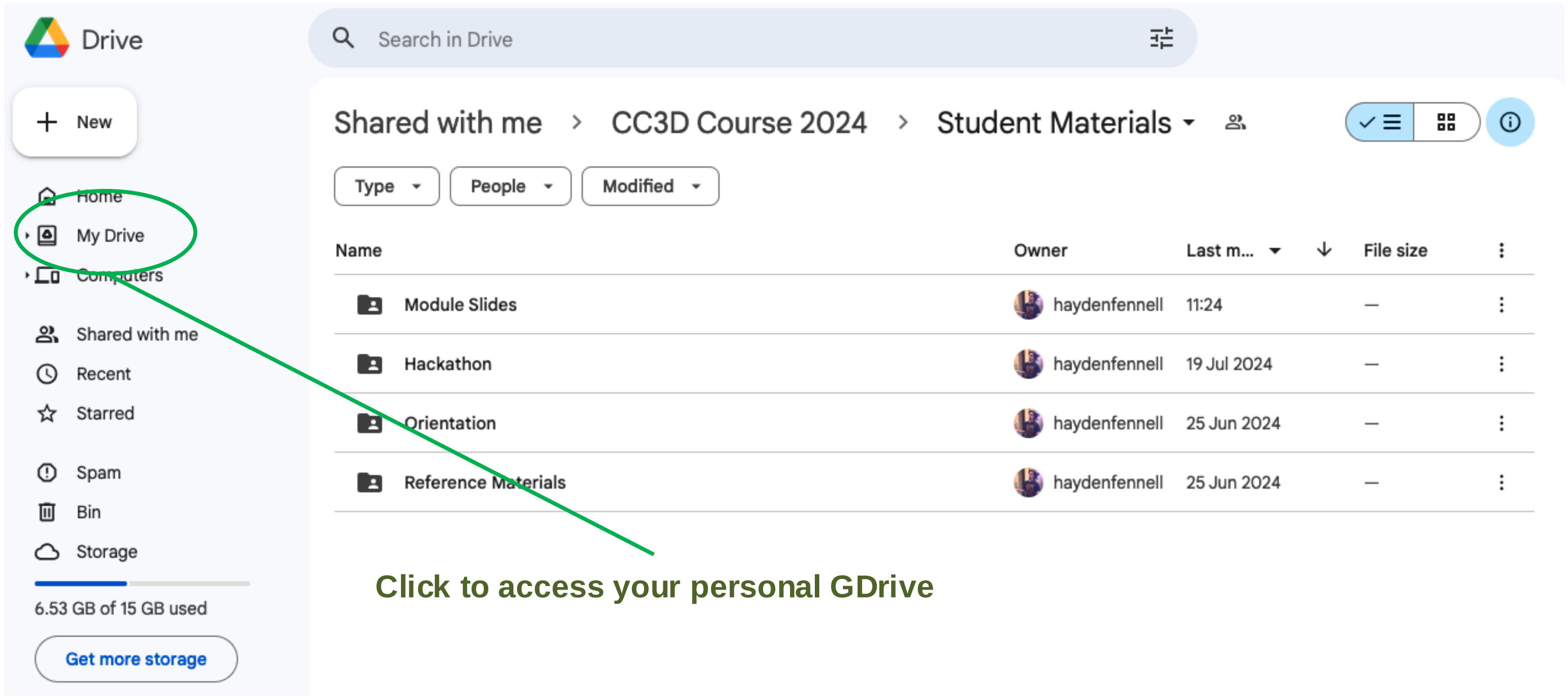
Google
https://www.google.co.uk › intl › en-GB › drive

Personal cloud storage and file-sharing platform
Drive can provide encrypted and secure access to your files. Files shared with you can be proactively scanned and removed when malware, spam, ransomware or phishing is detected.…

Google it (drive google) then Sign-in



If you are signed in in google:
Top right of browser window, click the ⋮⋮⋮ icon,
Then click the Icon
Drive



Drive

Search in Drive

New

Home
My Drive
Computers
Shared with me
Recent
Starred
Spam
Bin
Storage
6.53 GB of 15 GB used
Get more storage

Shared with me › CC3D Course 2024 › Student Materials

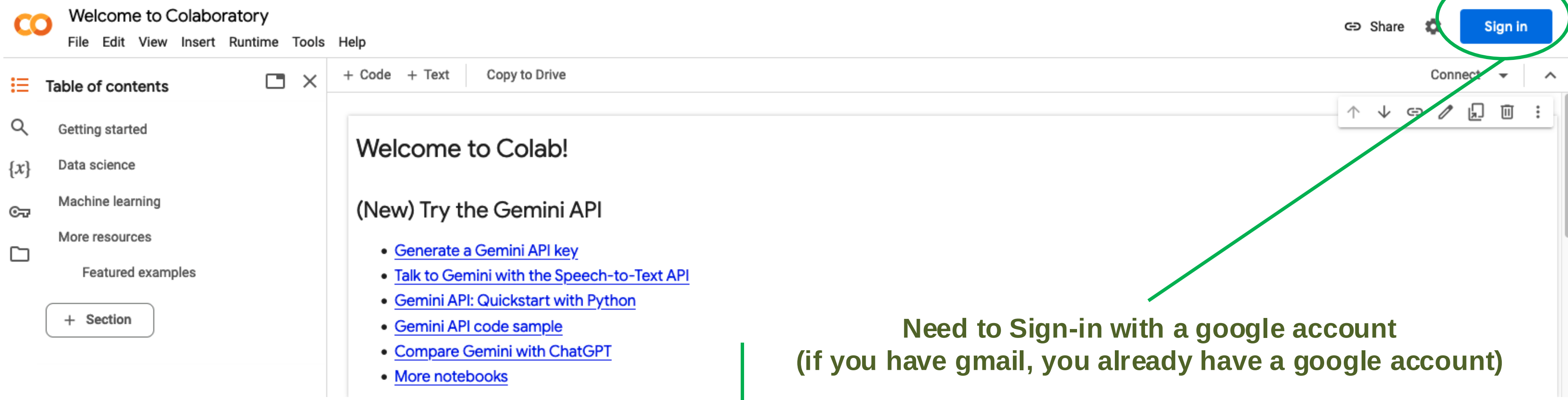
Type · People · Modified

Name	Owner	Last m...	File size
Module Slides	haydenfennell	11:24	—
Hackathon	haydenfennell	19 Jul 2024	—
Orientation	haydenfennell	25 Jun 2024	—
Reference Materials	haydenfennell	25 Jun 2024	—

Click to access your personal GDrive

Activity 0: Accessing Google Colab

https://colab.research.google.com/

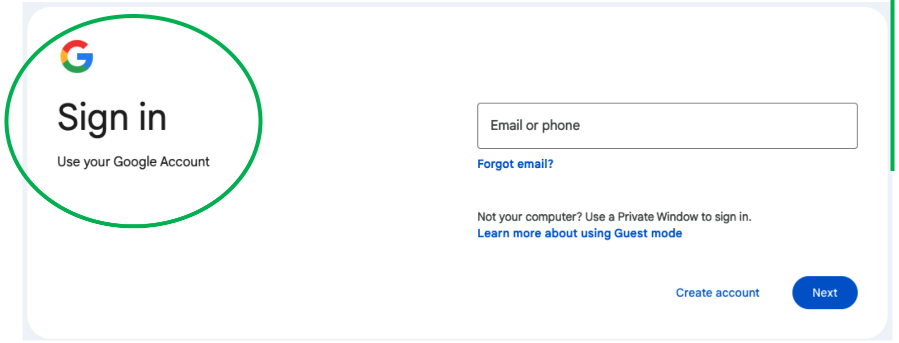


Welcome to Colab!

(New) Try the Gemini API

- Generate a Gemini API key
- Talk to Gemini with the Speech-to-Text API
- Gemini API: Quickstart with Python
- Gemini API code sample
- Compare Gemini with ChatGPT
- More notebooks

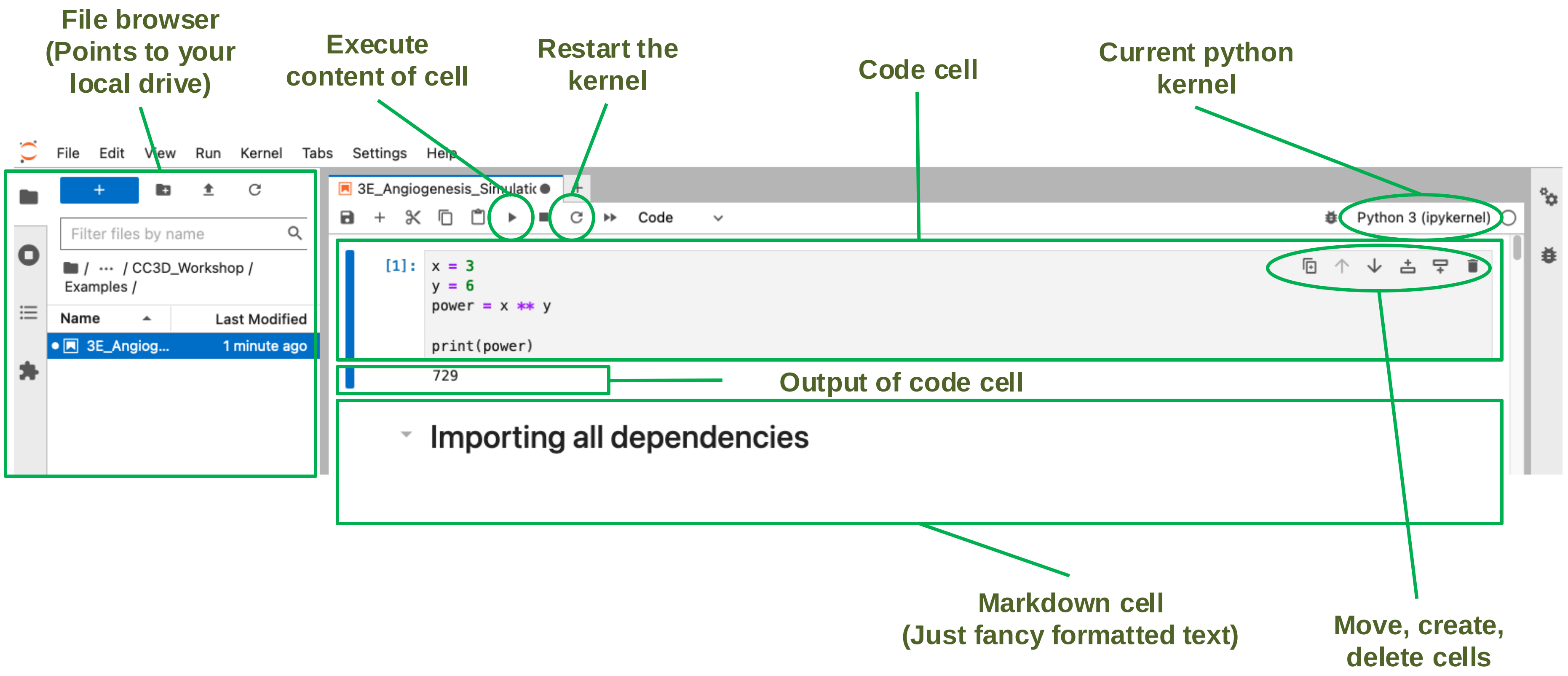
Need to Sign-in with a google account
(if you have gmail, you already have a google account)



Not your computer? Use a Private Window to sign in.
Learn more about using Guest mode

The Jupyter lab api

Familiarising with the Jupyter lab GUI


File browser (Points to your local drive)
Execute content of cell
Restart the kernel
Code cell
Current python kernel
File Edit View Run Kernel Tabs Settings Help
3E_Angiogenesis_Simulatio
Code
Python 3 (ipykernel)
Filter files by name
/ ··· / CC3D_Workshop / Examples /
Name
Last Modified
3E_Angiog...
1 minute ago
Output of code cell
Importing all dependencies
Markdown cell (Just fancy formatted text)
Move, create, delete cells

```
[1]: x = 3
     y = 6
     power = x ** y

     print(power)
     729
```

Cheat sheet: Installing cc3d and other python packages in a Conda environment

```
Type: conda create -n cc3d_ws python=3.10.13
```

-> Create a new **Conda env** named “cc3d_ws” with python 3.10.13

```
Type: conda activate cc3d_ws
```

-> **!!! Activate the new environment !!!**

```
Type: mamba --version
```

-> Check if you have Mamba installed… (Mamba is not a requirement, you can always use ”conda” instead, but mamba can speed-up the “package resolving” process)

```
Output:
mamba 1.4.9
conda 23.7.2
```

```
Type: conda install -c conda-forge mamba
```

-> If you don’t have mamba installed yet, install it…

```
Type: mamba install -c conda-forge -c compucell3d compucell3d=4.6.0
Confirm changes: [Y/n] y
```

-> Install **cc3d** with Mamba

```
Output:
Downloading and Extracting Packages
Preparing transaction: done
Verifying transaction: done
Executing transaction: \
-
done
```

-> Happy days!! All worked fine.
To check that your cc3d install is up and working

```
Type:python –m cc3d.twedit5
```

If all is OK, the familiar Twedit UI opens (it might take a while first time it runs in a new env)

```
Type: mamba install -c conda-forge scikit-image
```

-> Install **scikit-image** used for image analysis in our example

```
Type: mamba install -c conda-forge jupyterlab
```

-> Install **jupyter lab**

```
Type: mamba install -c conda-forge ipykernel
```

-> Install **ipykernel** so we can use our new cc3d_ws env as an ipynb Kernel

```
conda list ipykernel
```

-> Check if ipykernel is installed

```
Type: python -m ipykernel install --user --name cc3d_ws --display-name "cc3d_ws"
```

-> !!! Link our cc3d_ws env to be used as an ipynb Kernel !!!

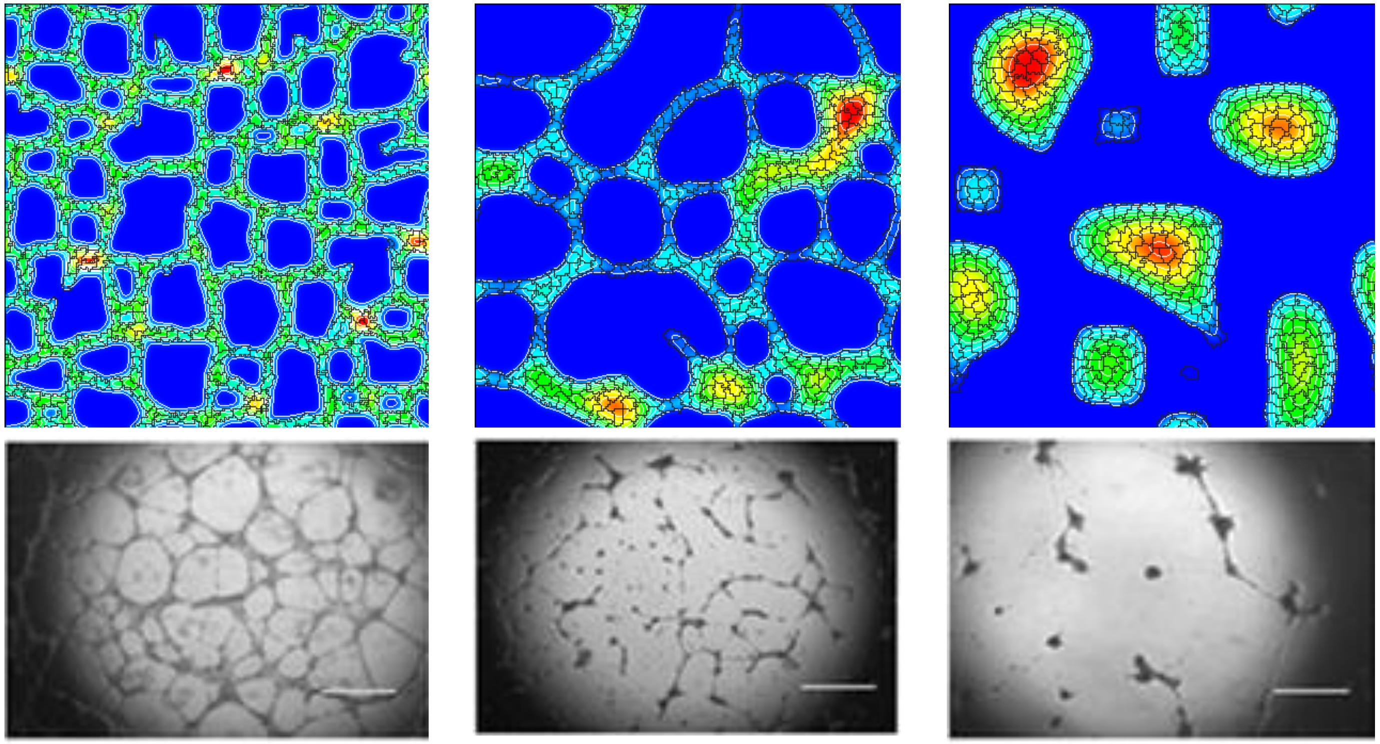
```
Type:jupyter lab
```

-> The **jupyter lab** UI opens in your default browser

The angiogenesis simulation

CC3D Simulations

Matrigel assay



Reference model:
Contact-Inhibited Chemotaxis in De Novo and Sprouting Blood-Vessel Growth, Roeland M. H. Merks, Erica D. Perryn, Abbas Shirinifard, James A. Glazier, Published: September 19, 2008, https://doi.org/10.1371/journal.pcbi.100016

Tweedit vs Python API: When?

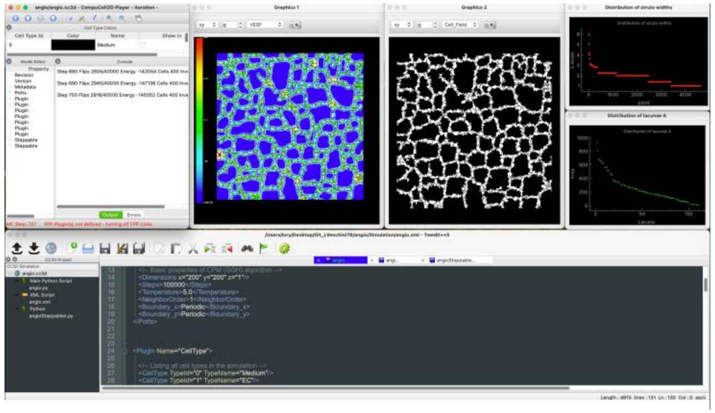
When to use Tweedit/Player?

- When you want to build a CC3D simulation piece by piece, looking at intermediate outcomes.
- When you need convenient and robust facilities to steer simulations runtime.
- When you want to observe the dynamics of simulation metrics over time.

So….almost always!

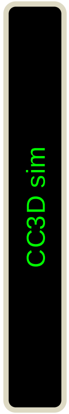
When to use the python API

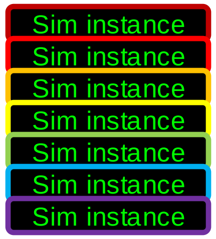
- When you want to run parameter scans/searches efficiently.
- When your main interest is gathering simulation output (vs looking at sim. evolution overtime)
- When you want to run many simulations in parallel in HPC or distributed computing environments

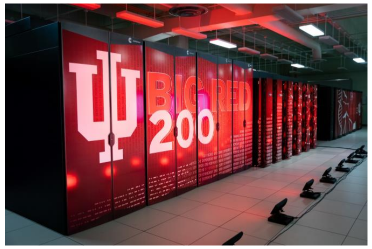


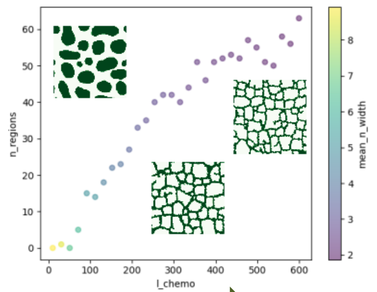
angio/angio.cc3d - CompuCell3D Player - Iteration
Graphics 1
Graphics 2
Distribution of struts widths
Distribution of lacunae A
Output
Errors
/Users/lory/Desktop/Git_LVeschini78/angio/Simulation/angio.xml - Tweedit++5

```
<!-- Basic properties of CPM (GGH) algorithm -->
<Dimensions x="200" y="200" z="1"/>
<Steps>100000</Steps>
<Temperature>5.0</Temperature>
<NeighborOrder>1</NeighborOrder>
<Boundary_x>Periodic</Boundary_x>
<Boundary_y>Periodic</Boundary_y>
</Potts>



<Plugin Name="CellType">

<!-- Listing all cell types in the simulation -->
<CellType TypeId="0" TypeName="Medium"/>
<CellType TypeId="1" TypeName="EC"/>
```









Embarrassingly parallel workflows

Embarrassing or delightful?

In parallel computing, an embarrassingly parallel workload or problem (also called **embarrassingly parallelizable**, **perfectly parallel**, **delightfully parallel** or **pleasingly parallel**) is one where little or no effort is needed to split the problem into a number of parallel tasks. This is due to minimal or no dependency upon communication between the parallel tasks, or for results between them.

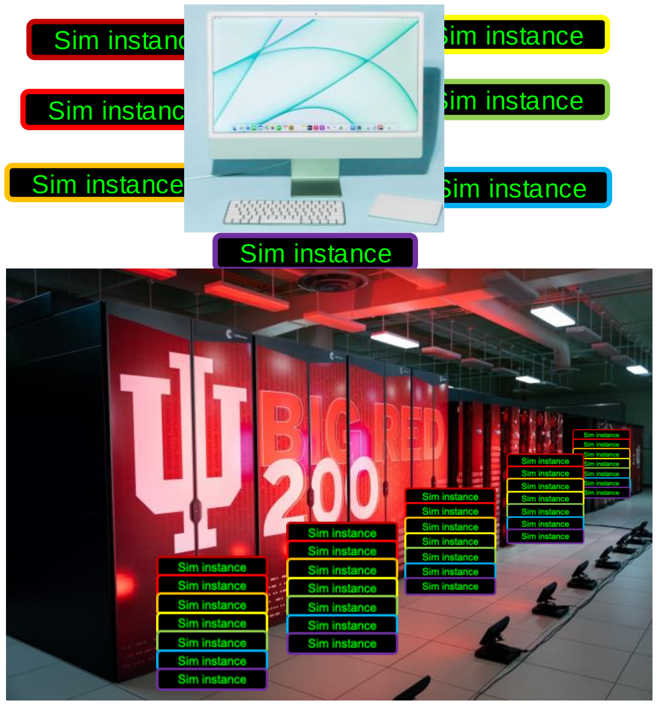
Serial

```
for instance in [Sim instance ×7] :
        inst_results = execute(instance)
        results.append(inst_results)
save(results)
```

Sim instance
Sim instance
Sim instance
Sim instance
Sim instance
Sim instance
Sim instance

Processing_T = P_T(instance) * n_instances

Parallel



Processing_T = P_T(instance) * ((n_instances/n_cores) + Overheads)

https://en.wikipedia.org/wiki/Embarrassingly_parallel

Python API: Wrapup

Which are (some of) the advantages of the cc3dpython API

- Allows generating very portable and reproducible code
- Perform Parameters searches/scans systematically
- Allows leveraging validated optimisation packages (PyPesto, PyMoo)
- Parallelisation is ~~Embarrassingly easy,~~ reasonably accessible
- Simple deployment to High Performance & Cloud Computing
- Simulation, data gathering, wrangling, plotting, and reporting, all in the same place.

Module 3.3 Questionnaire

Please take a minute or two to let us know about your experience with this module by filling out the brief zoom survey

Feel free to provide additional comments and suggestions in the slack or by email to us as well (hfennel@iu.edu)

Funding: **NIH U24 EB028887, NSF 2120200, NSF 2000281, NSF 1720625**
Previous Funding: NIH R01 GM12242413