

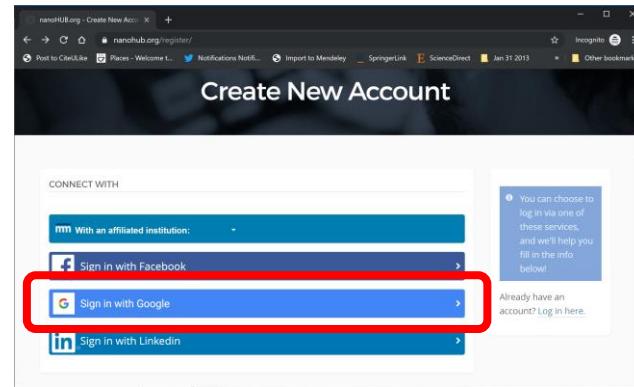
nanoHUB Account

- This talk's online PhysiCell models are cloud-hosted on nanoHUB.org.
- nanoHUB is **free**, but it requires a one-time registration.

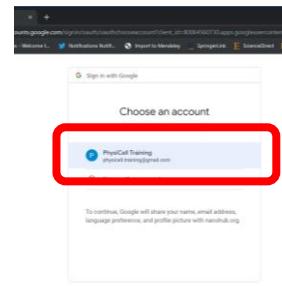
Steps:

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 nanoHUB account)
5. Finish filling in details, and you're done!
6. Use your google account to sign in in the future.

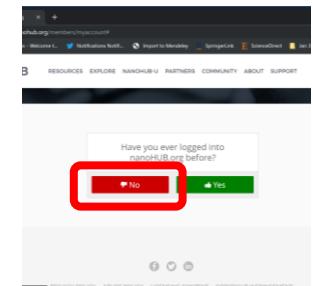
2



3



4



Introduction to Agent-Based Modeling in Cancer (live version)



Paul Macklin, Ph.D.

Intelligent Systems Engineering
Indiana University

Source code, extended details and more:

GitHub.com/PhysiCell-Training/UCI-sysbio-2022

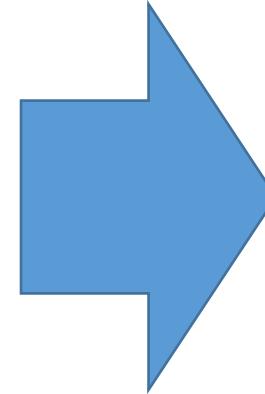
June 13, 2022

Part I:

Introduction to ABMs

From single cells to cancer ecosystems ...

- **Single-cell behaviors:**
 - Growth
 - Division
 - Differentiation
 - Death
 - Consumption
 - Metabolism
 - Secretion
 - Signaling
 - Mutations
 - Motility
 - **Cell-cell interactions:**
 - Adhesion
 - Mechanics
 - Predation
 - Contact communication
 - **Physical constraints:**
 - Diffusion limits
 - Mechanical barriers



Multicellular cancer ecosystem



Multicellular systems biology seeks to *understand* these systems.
Multicellular systems engineering seeks to *control* them.

Source: Hanahan (2022)
DOI: [10.1158/2159-8290.CD-21-1059](https://doi.org/10.1158/2159-8290.CD-21-1059)

Scientists use [models*] to detangle complex systems.

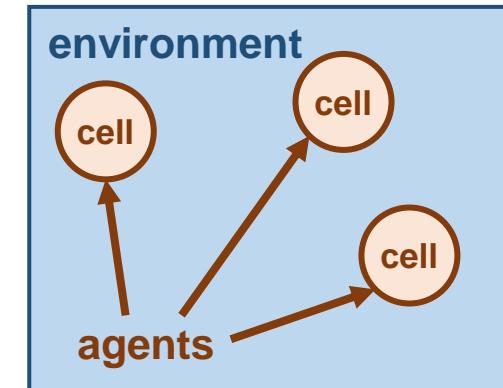
* animal, *in vitro*, engineered, mathematical, ...

Key parts of a multicellular virtual laboratory

- **Model multiple diffusing chemical factors**
 - Growth substrates and metabolites
 - Signaling factors
 - Drugs and therapeutic compounds
- **Model many cells in these chemical environments**
 - Environment-dependent behavior (including molecular-scale "logic")
 - Mechanical interactions
 - Heterogeneity:
 - ◆ individual states
 - ◆ individual parameter values
 - ◆ individual model rules
- **Run many copies of the model in high throughput**
 - Discover the rules that best match observations.
 - Identify and exploit weaknesses that can restore control

What is an agent-based model?

- Individual cells are modeled as discrete (and stochastic) software **agents**.
 - Individual state variables
 - Individual rules
 - ◆ often written as systems of ordinary differential equations (ODEs) or other constitutive laws
 - **The cell agents encode our biological knowledge and hypotheses as rules**
- Cell agents live in a **virtual tissue microenvironment**.
 - Diffusible substrates
 - ◆ modeled as partial differential equations (PDEs)
 - Mechanical barriers
 - **The microenvironment encodes physical constraints**
- Cells agents **interact** with each other and the environment.
 - Mechanics and contact interactions
 - Secretion, uptake and export of diffusible substrates



Types of cell-based models

- **lattice-bound**

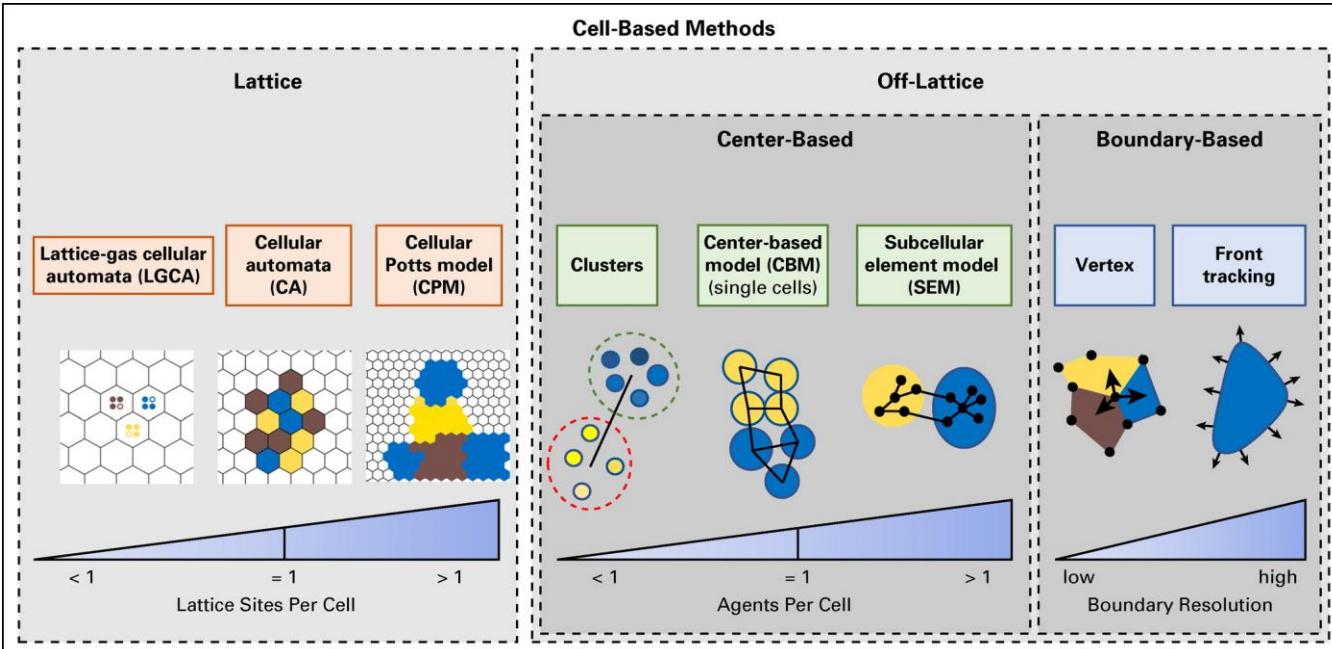
- resolution:

- ◆ < 1 site / cell:
 - » lattice gas
 - ◆ 1 site / cell
 - » cellular automaton
 - ◆ many sites / cell
 - » cellular Potts

- **off-lattice**

- **center-based**

- **boundary-based**



J. Metzcar, Y. Wang, R. Heiland, and P. Macklin. A review of cell-based computational modeling in cancer biology. *JCO Clinical Cancer Informatics* 3:1-13, 2019 (invited review). DOI: [10.1200/CCI.18.00069](https://doi.org/10.1200/CCI.18.00069).

BioFVM: Simulating 3-D biotransport

Design goal: Simulate multiple diffusing substrates in 3D with desktops or single HTC/HPC nodes

Typical use: pO_2 , glucose, metabolic waste, signaling factors, and a drug, on 10 mm^3 at $20 \mu\text{m}$ resolution

Features:

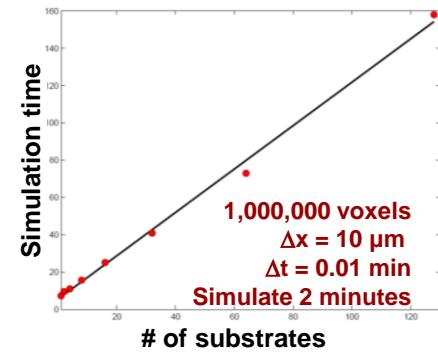
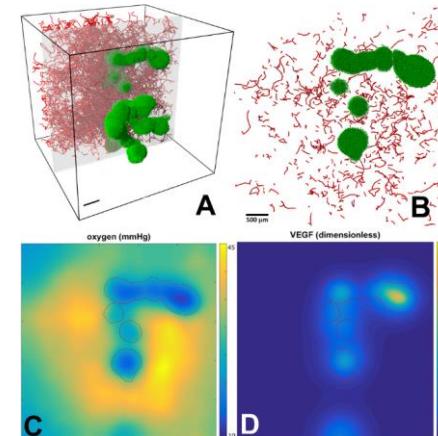
- Off-lattice cell secretion and uptake
- 2nd-order accurate (space), 1st-order accurate (time), numerically stable

Method:

- Operator splitting, LOD, customized Thomas solvers, etc.
- Standard C++11, cross-platform
- OpenMP parallelization
- $O(n)$ cost scaling in # substrates, # voxels
- Easy to simulate 5-10 substrates on 10^6 voxels

Reference: Ghaffarizadeh et al., *Bioinformatics* (2016)

DOI: [10.1093/bioinformatics/btv730](https://doi.org/10.1093/bioinformatics/btv730)



PhysiCell: A multicellular framework

Design goal: Simulate 10^6 or more cells in 2D or 3D on desktops or single HPC nodes

Features:

- Off-lattice cell positions
- Mechanics-based cell movement
- Cell processes (cycling, motility, ...)
- Signal-dependent phenotype
- Can dynamically attach custom data and functions on a cell-by-cell basis

Method:

- Standard C++11, cross-platform
- OpenMP parallelization
- $O(n)$ cost scaling in # cells

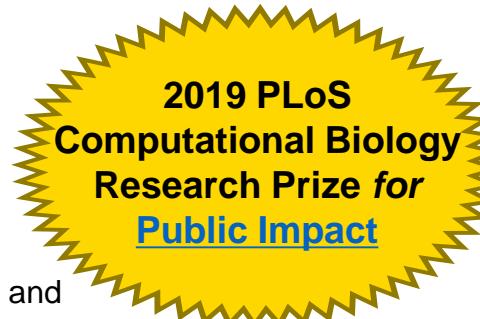
Reference: Ghaffarizadeh et al.,
PLoS Comput. Biol. (2018)

DOI: [10.1371/journal.pcbi.1005991](https://doi.org/10.1371/journal.pcbi.1005991)

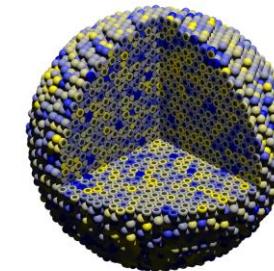


Try this model yourself!

nanohub.org/tools/pc4heterogen



Current time: 0 days, 0 hours, and 0.00 minutes
18317 cells



Competition in a 3-D tumor
[View on YouTube](#) (8K)

Some additional mathematical detail

- Secretion, uptake, and a generalized uptake are centered at cell locations via Dirac delta functions.
- Cell state transitions are stochastic.
 - They can be related to Poisson processes and coarse-grained to classical ODEs.
- Cell motility is a biased random walk.
 - Chemotaxis is modeled through a bias direction.
- Intracellular models can take multiple forms:
 - ODEs, Boolean networks, flux balance, deep neural networks
- See the extended slides on GitHub for more detail.

Phenotype-centric modeling

- We have the key biological processes implemented out-of-the-box:
 - Cycle progression (variety of models available)
 - Apoptotic and necrotic death
 - Mechanics and motility
 - Secretion / Uptake / Export
 - Phagocytosis, fusion, "attack", etc.
 - Mutations
 - ...
- Each process is regulated by corresponding biophysical parameters
 - Generally they are process time scales!
- Modelers write functions to control these process parameters
 - Vary with microenvironmental conditions
 - Vary with cell state
 - Use this to write your biological hypotheses

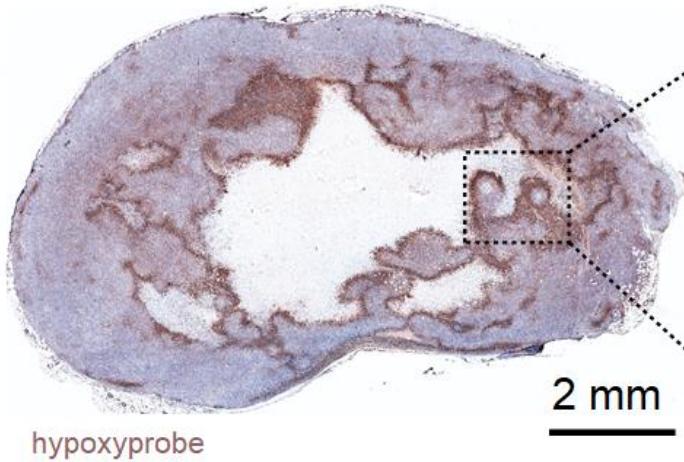
Example: hypoxia-driven breast cancer invasion

Rocha et al., *iScience* (2021)
[DOI: 10.1016/j.isci.2021.102935](https://doi.org/10.1016/j.isci.2021.102935)



Intratumoral hypoxia

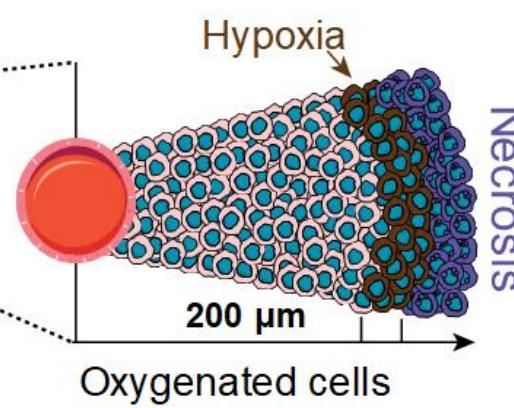
Mouse tumor



Blood vessels



Hypoxia



hypoxyprobe

2 mm

200 μ m

Oxygenated cells

Necrosis

Fate-mapping intratumoral hypoxia

nature communications

View All Nature Research Journals

Search

Login

Explore Content ▾ Journal Information ▾ Publish With Us ▾

Sign Up For Alerts

RSS Feed

nature > nature communications > articles > article

Article | Open Access | Published: 24 October 2019

Fate-mapping post-hypoxic tumor cells reveals a ROS-resistant phenotype that promotes metastasis

Inés Godet, Yu Jung Shin, Julia A. Ju, I Chae Ye, Guannan Wang & Daniele M. Gilkes

Nature Communications 10, Article number: 4862 (2019) | Cite this article

8904 Accesses | 17 Citations | 167 Altmetric | Metrics

Abstract

Hypoxia is known to be detrimental in cancer and contributes to its development. In this work, we present an approach to fate-map hypoxic cells *in vivo* in order to determine their cellular response to physiological O₂ gradients as well as to quantify their contribution to metastatic spread. We demonstrate the ability of the system to fate-map hypoxic cells in 2D, and in 3D spheroids and organoids. We identify distinct gene expression patterns in cells that experienced intratumoral hypoxia *in vivo* compared to cells exposed to hypoxia *in vitro*. The intratumoral hypoxia gene-signature is a better prognostic indicator for distant metastasis-free survival. Post-hypoxic tumor cells have an ROS-resistant phenotype that provides a survival advantage in the bloodstream and promotes their ability to establish overt metastasis. Post-hypoxic cells retain an increase in the expression of a subset of hypoxia-inducible genes at the metastatic site, suggesting the possibility of a 'hypoxic memory.'

Download PDF

Sections Figures References

Abstract

Introduction

Results

Discussion

Methods

Data availability

References

Acknowledgements

Author information

Ethics declarations

Additional information

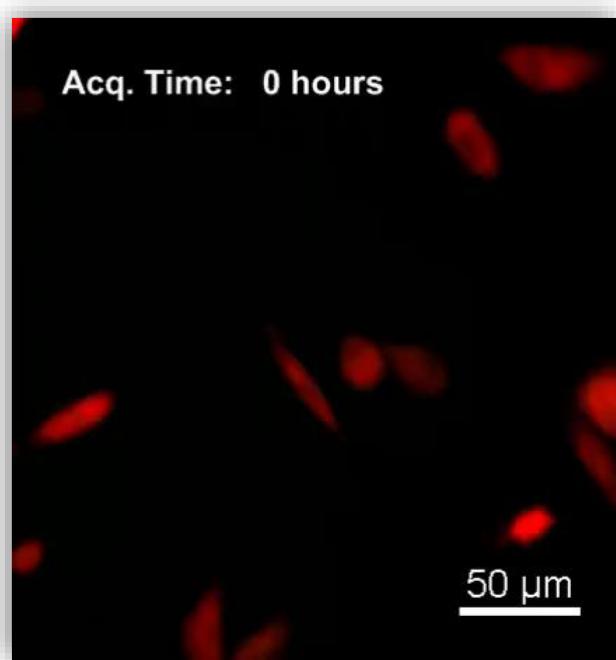
Supplementary information

Rights and permissions

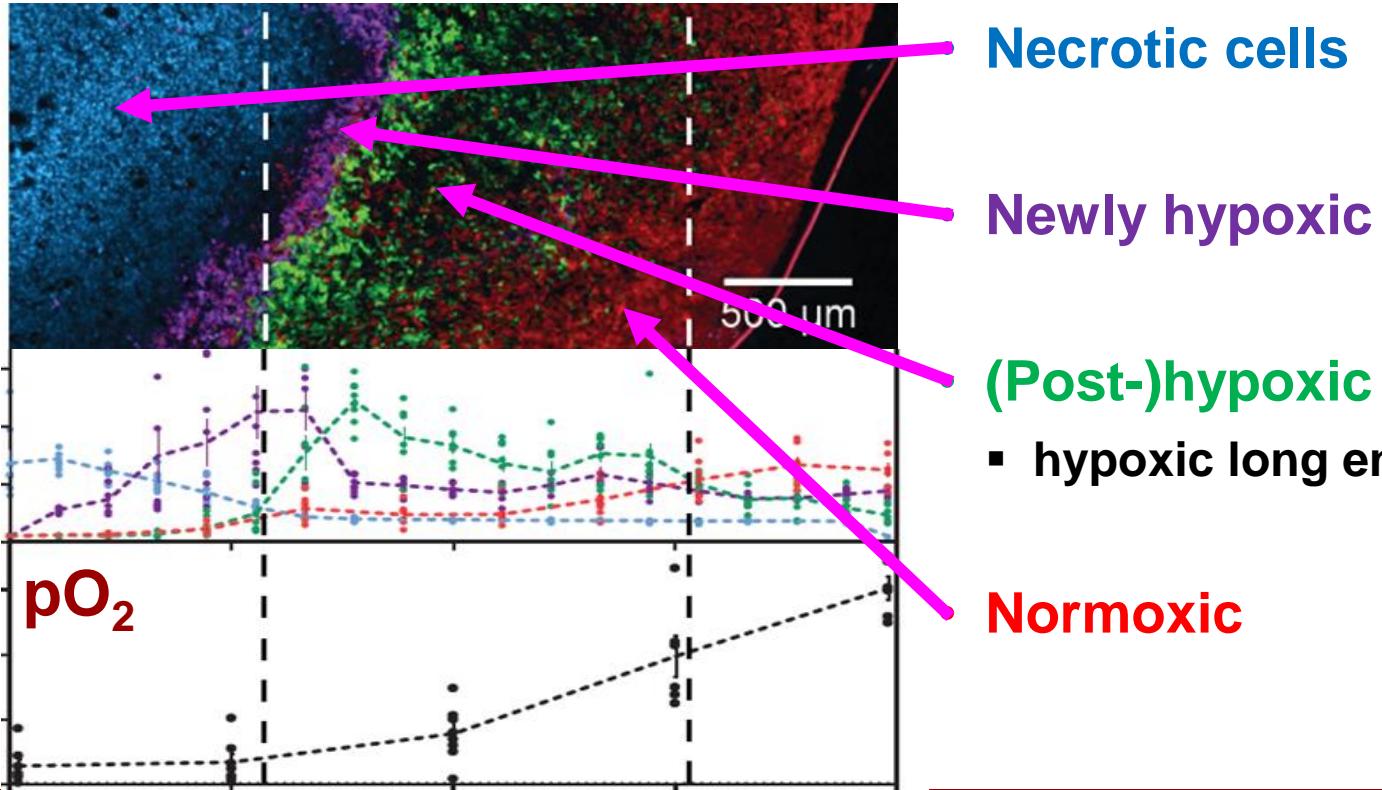
About this article

Further reading

Comments



Hypoxic and post-hypoxic cells



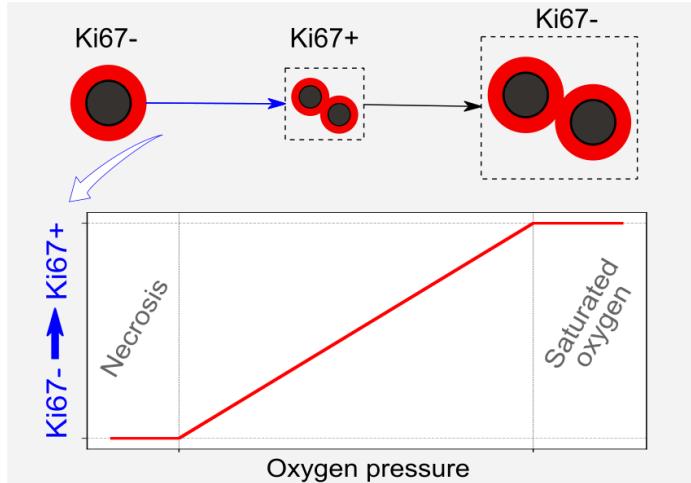
Questions

What are the **rules** of hypoxic cell motility?

How persistent is their response to hypoxic stress?

Proliferation and migration

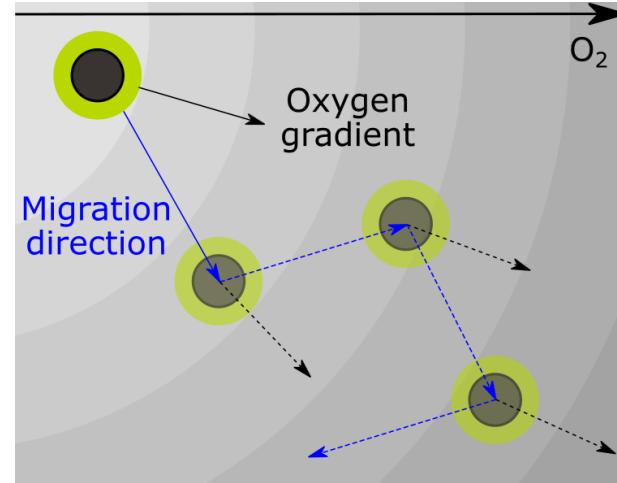
Cell proliferation



Basic Ki-67 model

$$r_{\text{cycle}} = r_{\max} \cdot \left(\frac{pO_2 - pO_2_{\min}}{pO_2_{\max} - pO_2_{\min}} \right) \cdot \left(1 - \frac{p}{p_{\max}} \right)$$

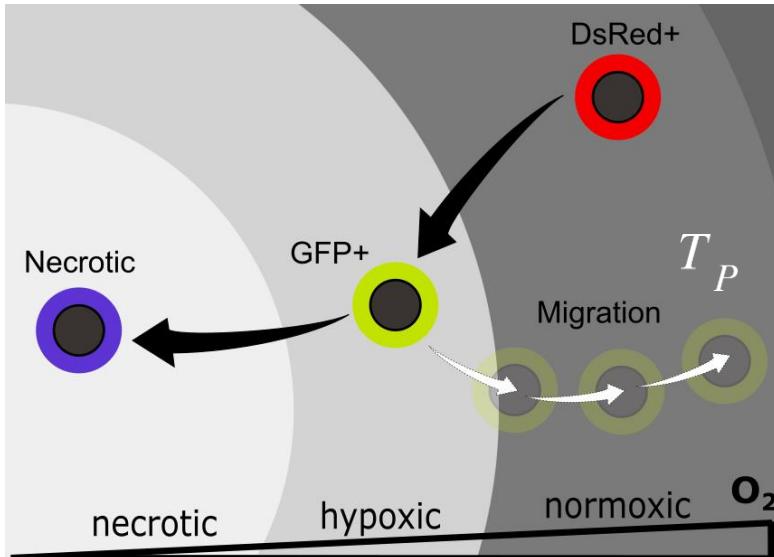
Cell migration



- Speed (s)
- Bias (b)
- Persistence time (t)

Phenotypic transitions

Eventually, cells may undergo phenotypic transitions due to their **motility** or **changes in the microenvironment**.



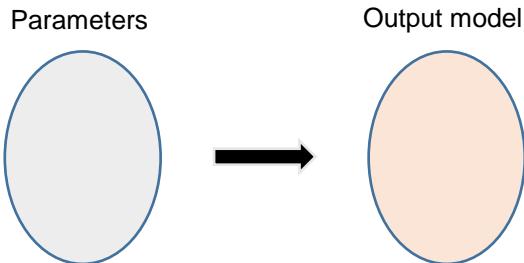
Simple ODE model for protein expression based on the "genes"

$$G = (G_0, G_1)$$

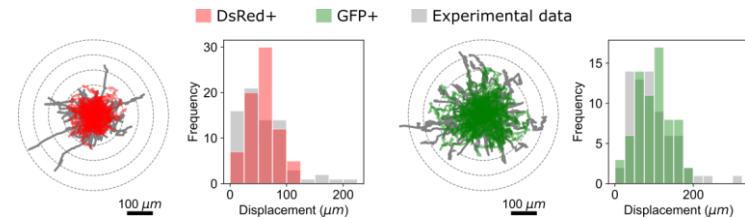
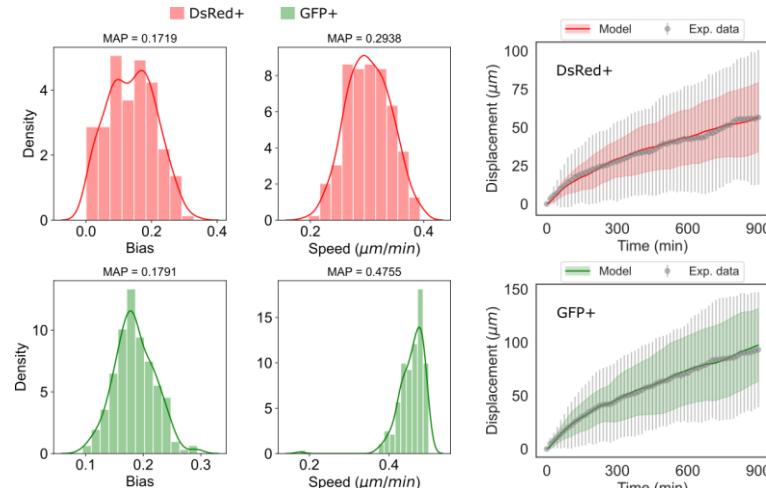
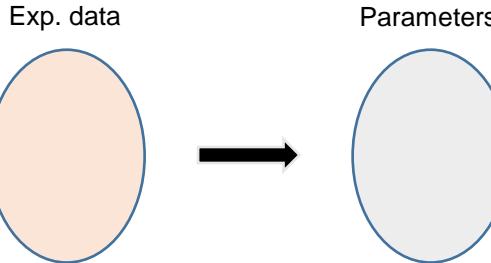
$$\frac{d[DsRed]}{dt} = G_0 \alpha_0 (1 - [DsRed]) + \beta_0 (G_0 - [DsRed])$$
$$\frac{d[GFP]}{dt} = G_1 \alpha_1 (1 - [GFP]) + \beta_1 (G_1 - [GFP])$$

Biological observations calibrate cellular motility in hypoxia computational model

Forward problem



Inverse problem



Phenotypic persistence drives invasion

Phenotypic Persistence:

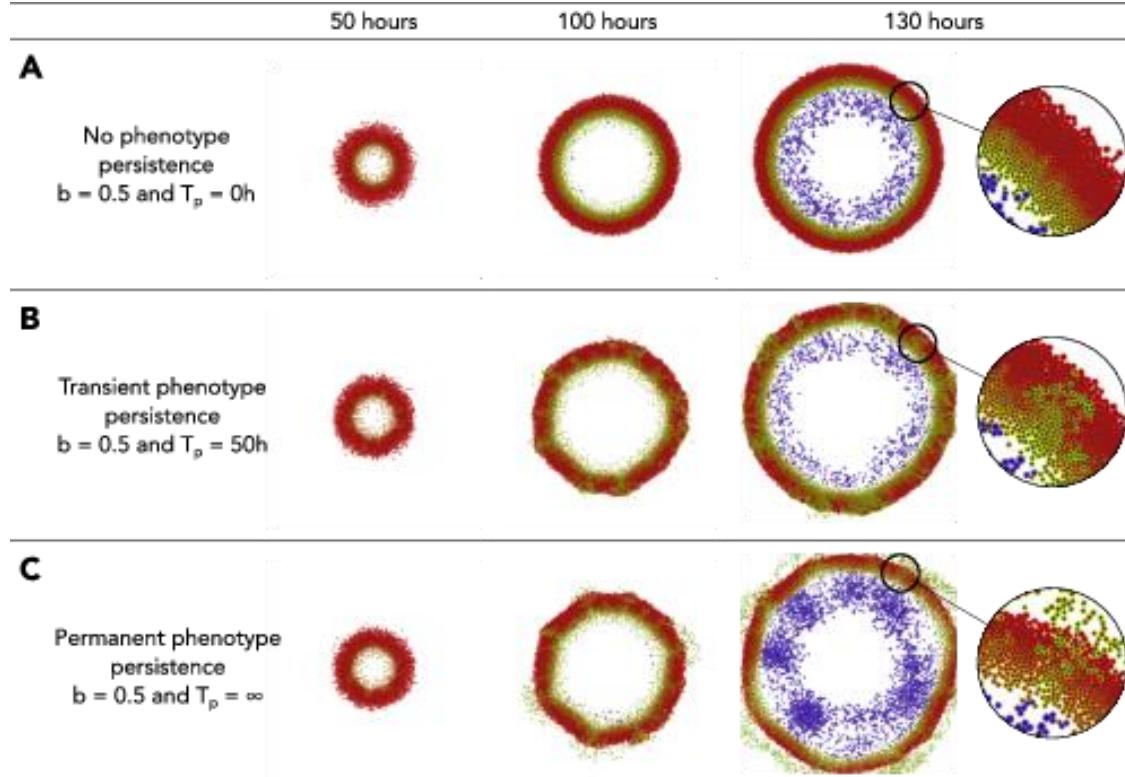
- T_p : duration of hypoxic response

Without persistence ($T_p = 0$) – Row A

- Migration halts at perinecrotic boundary
- Tumors maintain a concentric structure:
 - Oxygenated viable rim (red)
 - hypoxic (or formerly hypoxic) annulus (green)
 - Necrotic core (purple)

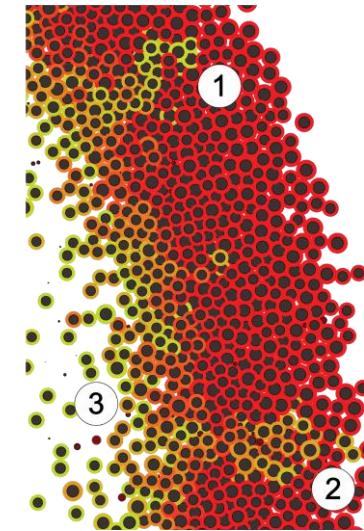
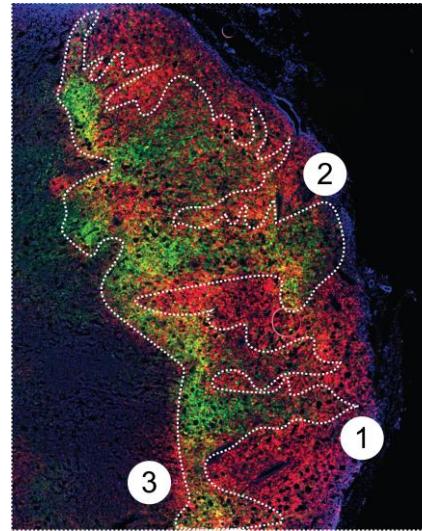
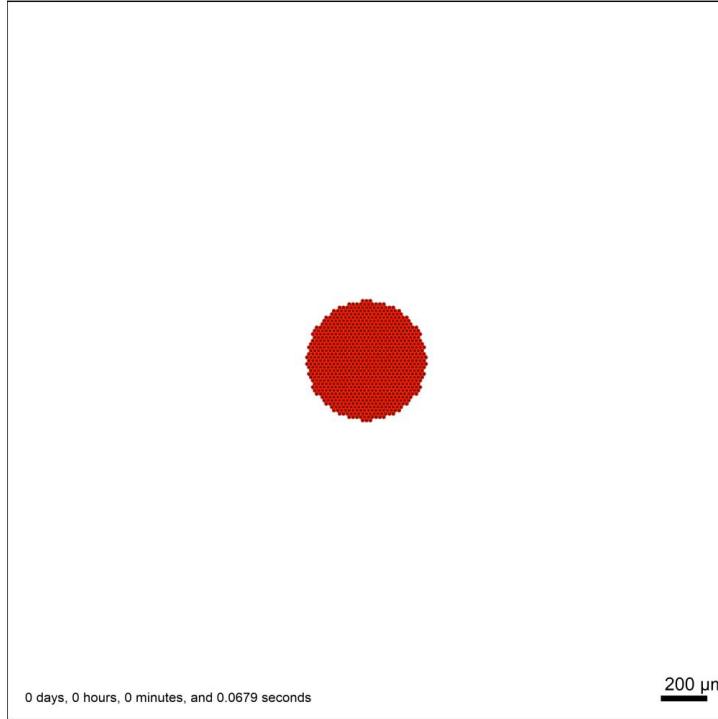
With persistence ($T_p > 0$) – Rows B & C

- Hypoxic cells can continue migrating
- Hypoxic cells "punch through" the oxygenated tumor region

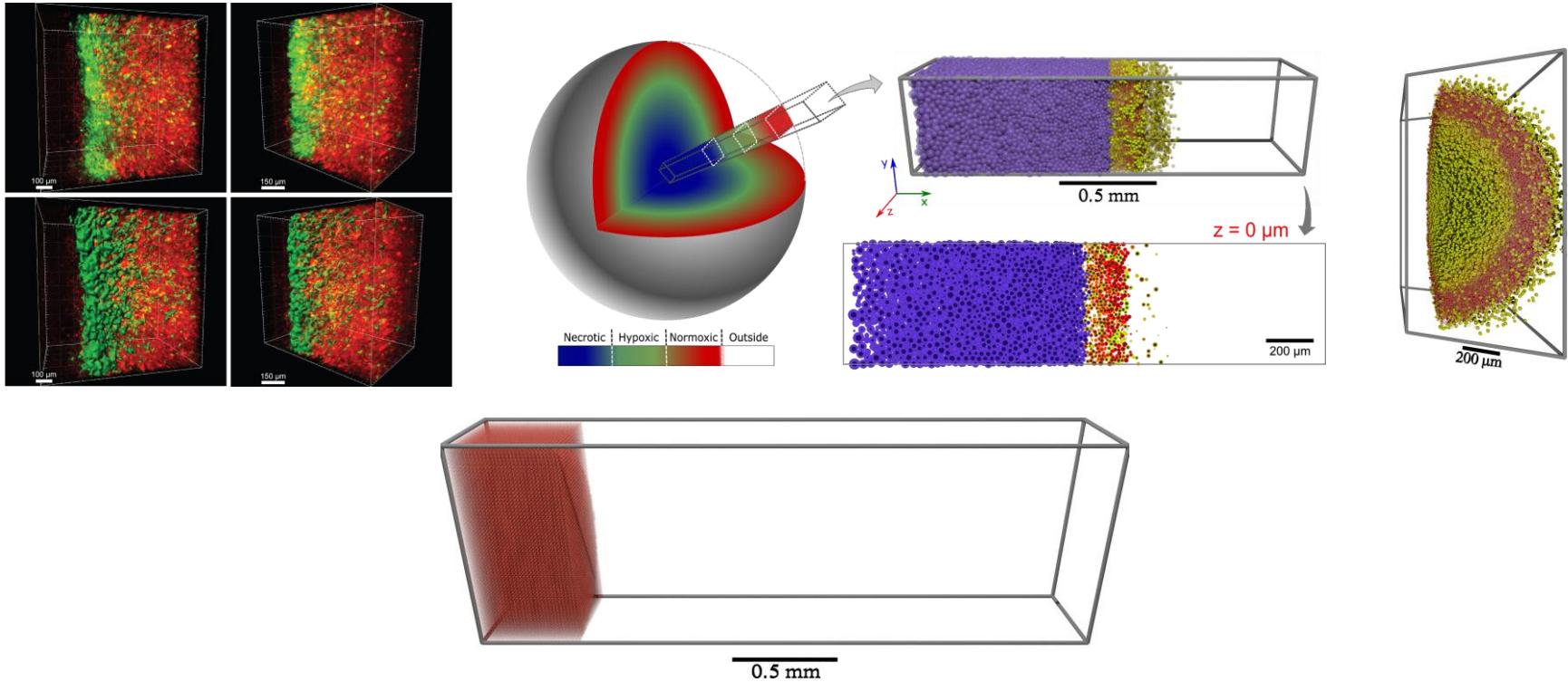


Mathematical model explains biological observations

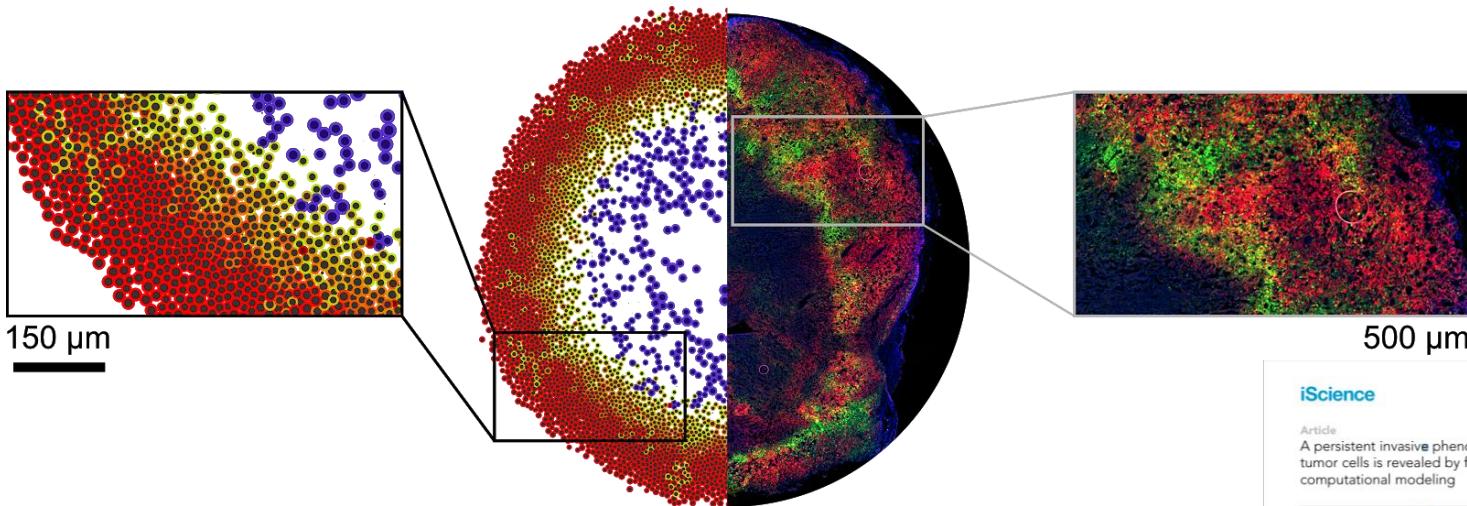
Current time: 0 days, 0 hours, and 0.00 minutes, z = 0.00 μm
889 agents



It also works in 3D



Explore this model



150 μm

500 μm



Try this model yourself!
nanohub.org/tools/pc4tumorhypoxia

Rocha et al., *iScience* (2021)
[DOI: 10.1016/j.isci.2021.102935](https://doi.org/10.1016/j.isci.2021.102935)

iScience CellPress OPEN ACCESS

Article
A persistent invasive phenotype in post-hypoxic tumor cells is revealed by fate mapping and computational modeling

Nelson L. Rocha, Inés Cordero, Daniel J. Sosa, Sébastien Brault, Danièle Demeure, Paul Macklin

Abstract
Post-hypoxic tumor cells exhibit a persistent invasive phenotype. We used fate mapping and computational modeling to reveal the mechanisms underlying this behavior.

Hypoxia Fate-mapping
Imaging serial slices of a hypoxic tumor and tracking individual cells over time reveals a population of cells that persistently migrate away from the hypoxic core.

Post-hypoxic phenotype persistence
Computational modeling predicts that post-hypoxic cells have a persistent invasive phenotype.

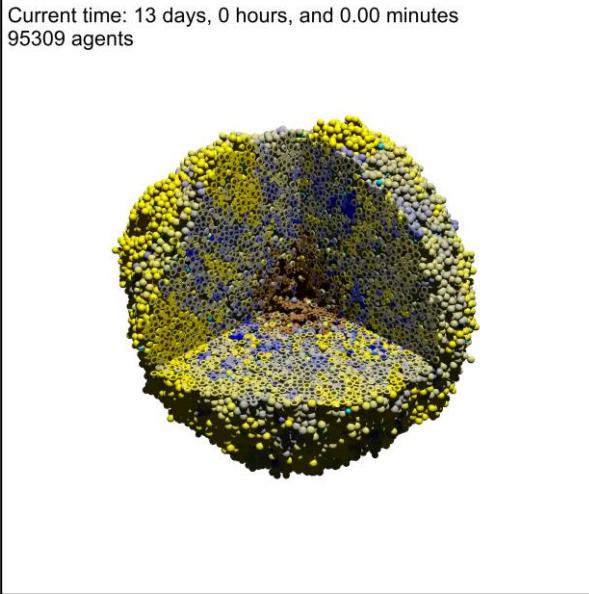
Predicted Invasive Plume
Computational modeling predicts that post-hypoxic cells form an invasive plume.

Observed Invasive Plume
Imaging serial slices of a hypoxic tumor reveals an observed invasive plume.

Contributions
N.L.R. performed the experiments and analyzed the data. I.C. and D.J.S. performed the fate mapping analysis. S.B. and D.D. performed the imaging analysis. P.M. performed the computational modeling. All authors contributed to the manuscript preparation.

Other examples

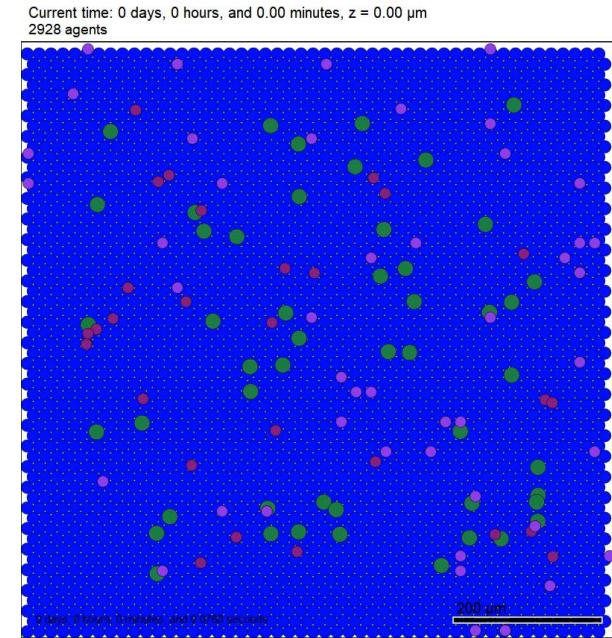
Immune cells attacking a heterogeneous tumor



Micrometastasis growth in liver parenchyma



SARS-CoV-2



Part II: Building a tumor-immune model online

Goal: Build a simple tumor-immune model

- **Tumor:**

- Wild Type tumor cells:

- ◆ Proliferate and apoptose at a constant rate
 - » Oxygen increases proliferation*
 - » Pressure reduces proliferation*
 - » Low oxygen increases necrosis*
 - ◆ Secrete a tumor factor
 - ◆ Dead cells release debris
 - ◆ Randomly mutate
 - ◆ Damage increases apoptosis*

- Mutant tumor cells:

- ◆ Random migration, slower proliferation
 - ◆ Less secretion of tumor factor

* requires additional C++

** see extended version on GitHub

- **Immune:**

- Macrophages

- ◆ Attracted to tumor factor and debris
 - ◆ Secrete inflammatory factor
 - ◆ Phagocytose (consume) dead cells

- CD8+ T cells

- ◆ Attracted to inflammatory factor
 - ◆ Attack (cause damage to) tumor cells
 - » Faster damage rate to WT tumor cells

- Monocytes **

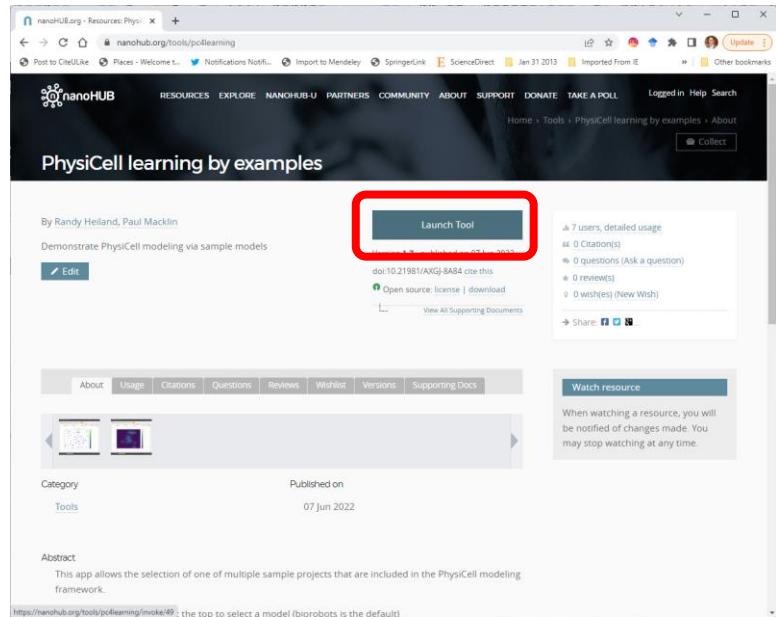
- ◆ Attracted to inflammatory factor
 - ◆ Phagocytose (live) tumor cells
 - » Faster rate for WT tumor cells

Approach

- Use online simulator: <http://nanohub.org/tools/pc4learning>
- Build iteratively:
 - Set the domain
 - Add diffusing substrates
 - Add WT tumor cells and test
 - Add mutant tumor cells and test
 - Add macrophages and test
 - Add CD8+ T cells and test
 - Add monocytes and test

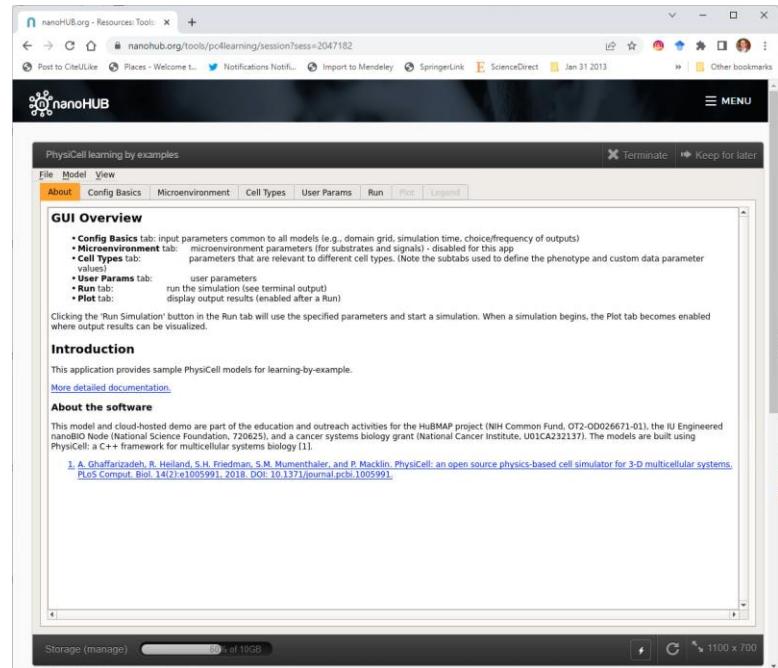
Start the online simulator

- Go to the tool on nanoHUB:
 - <https://nanohub.org/tools/pc4learning>
- Make sure you're logged on.
- Click the “launch tool” button



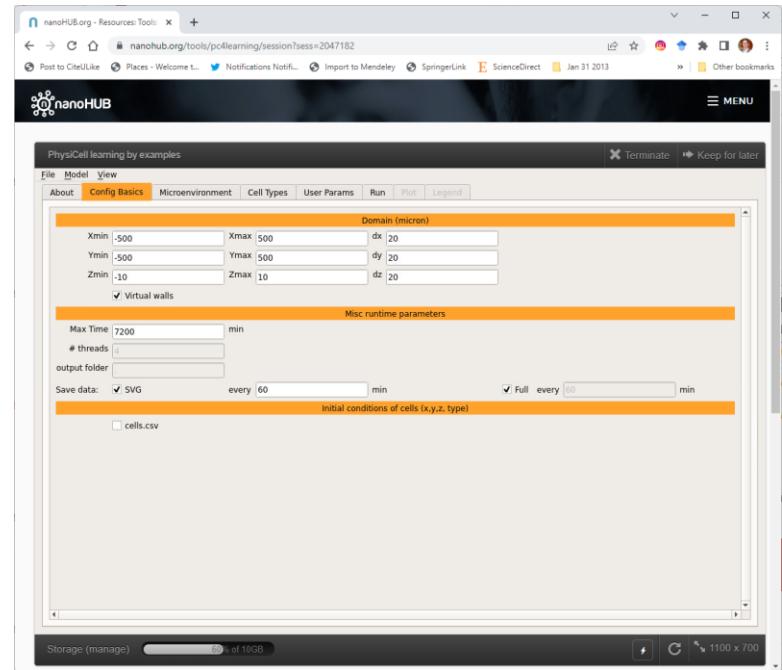
App navigation

- **Overview:**
 - basic overview
- **Config basics:**
 - Domain size, Simulation duration, Data output
- **Microenvironment:**
 - Define diffusing substrates and boundary conditions
- **Cell types:**
 - Define cell types, including their base phenotypes (behaviors)
- **User params:**
 - Model-specific parameters
- **Run:**
 - Start running in the cloud and view (virtual) console output
- **Plot:**
 - Plot the cells and diffusing substrates
- **Legend:**
 - Define the coloring of the plotted cell types



Set up the domain

- Go to the **config basics** tab
- Choose domain settings (in μm)
 - leave $Z_{\min} = -\frac{1}{2}\Delta z$ and $Z_{\max} = \frac{1}{2}\Delta z$ for 2D models
 - Use “virtual walls” to apply a force to keep cells from leaving the domain
- Choose time settings (in min):
 - Max time is the simulation end time
- Choose save settings:
 - SVG: required for plotting cell positions
 - Full: required for plotting diffusing substrates



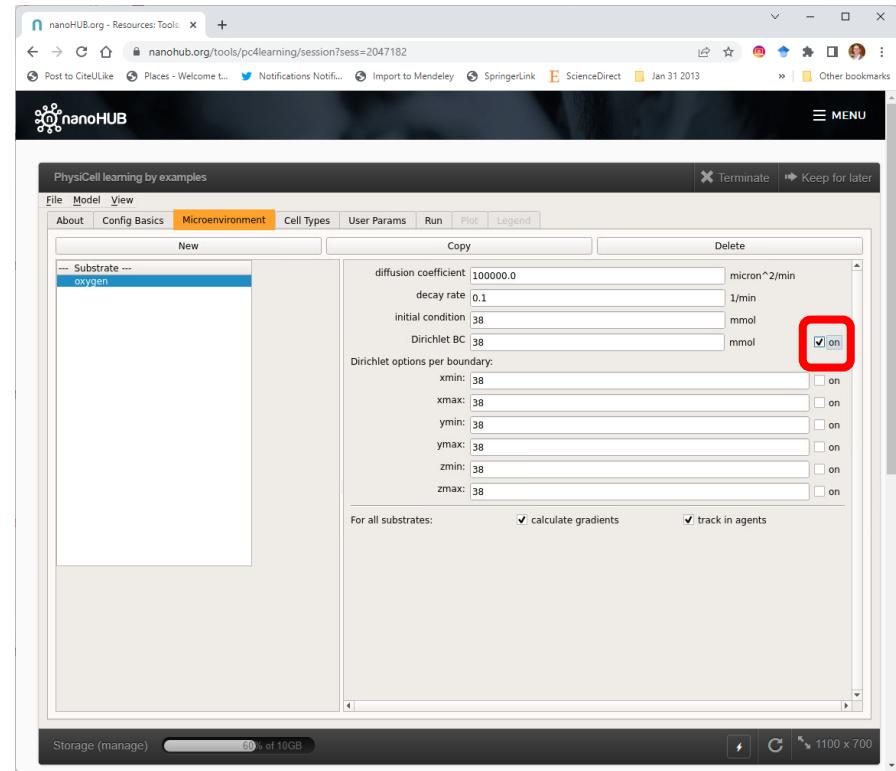
Leave these settings at default values today

Checklist

- Use online simulator: <http://nanohub.org/tools/pc4learning>
- Build iteratively:
 - Set the domain
 - Add diffusing substrates
 - Add WT tumor cells and test
 - Add mutant tumor cells and test
 - Add macrophages and test
 - Add CD8+ T cells and test
 - Add monocytes and test

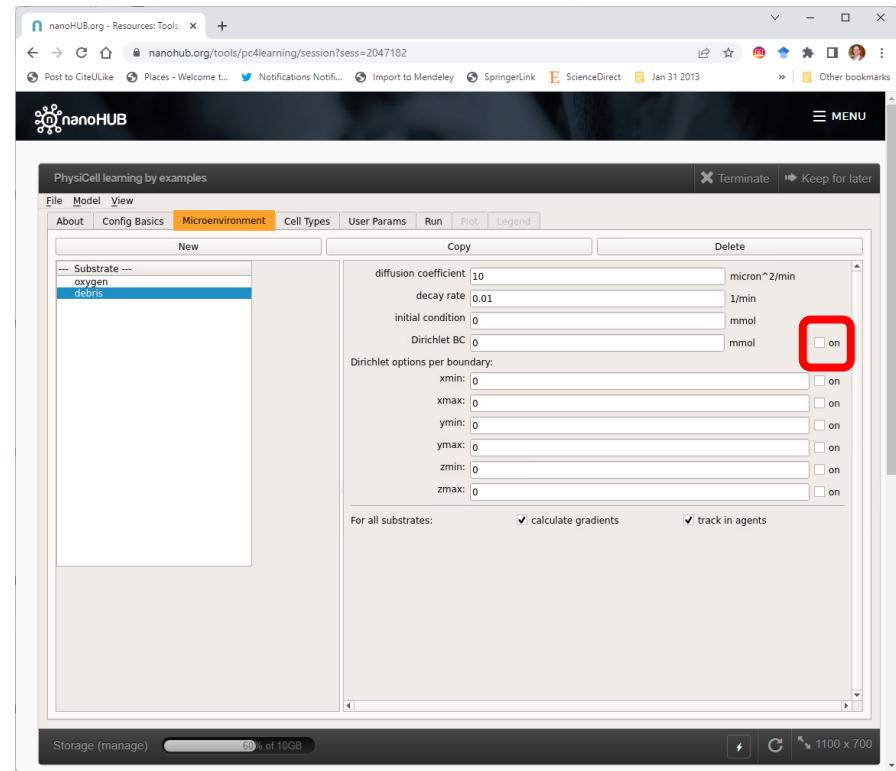
Define substrates (1)

- Go to the **microenvironment** tab
- Double-click **substrate**
 - Rename to **oxygen**
 - Keep diffusion at $100000 \mu\text{m}^2/\text{min}$
 - Set decay at 0.1 min^{-1}
 - ◆ 1 mm diffusion length scale
 - (1 mm length scale)
 - Set Dirichlet boundary conditions to 38 mmHg (5% O₂: physioxic conditions)
 - Set initial condition to 38 mmHg



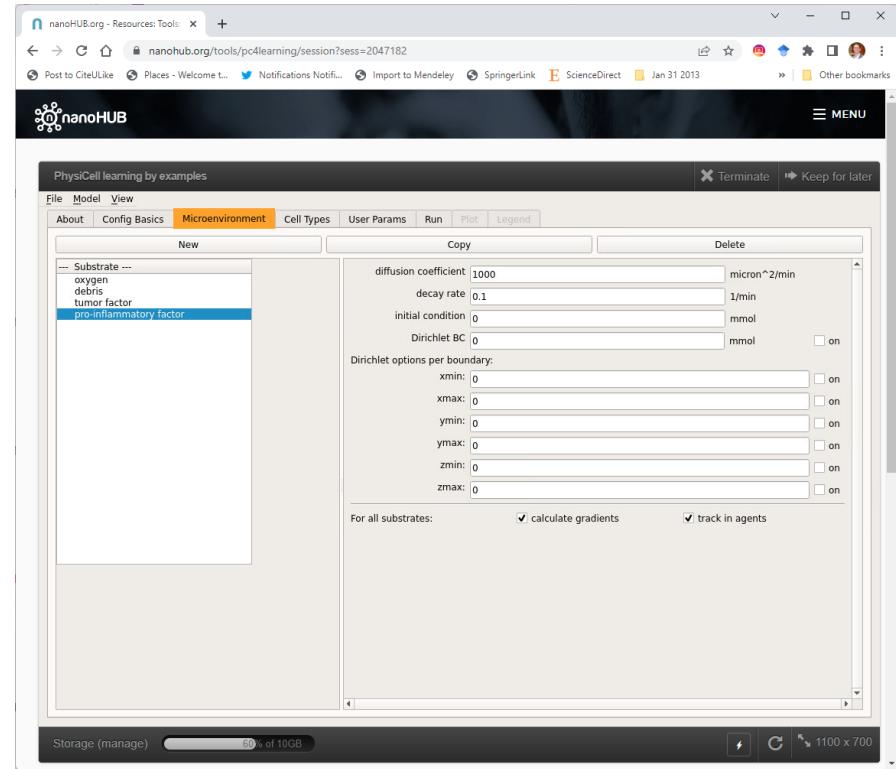
Define substrates (2)

- Make sure **oxygen** is selected
- Click **copy**
- Rename to **debris**
 - Set diffusion at $10 \mu\text{m}^2/\text{min}$
 - Set decay at 0.01 min^{-1}
 - ◆ $100 \mu\text{m}$ diffusion length scale
 - Disable Dirichlet boundary condition
 - Set initial condition to 0



Define substrates (3)

- Make sure **debris** is selected
- Click **copy**
- Rename to **tumor factor**
 - Set diffusion at $1000 \mu\text{m}^2/\text{min}$
 - Set decay at 0.1 min^{-1}
 - ◆ $100 \mu\text{m}$ diffusion length scale
 - Disable Dirichlet boundary condition
 - Set initial condition to 0
- Copy **tumor factor** and repeat to create **pro-inflammatory factor**

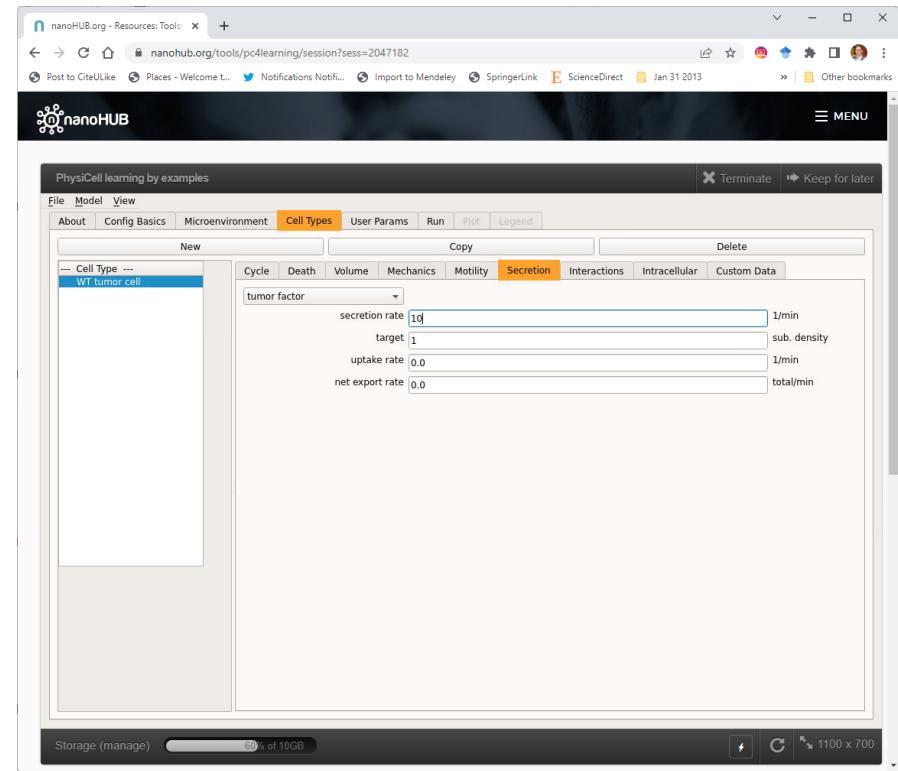


Checklist

- Use online simulator: <http://nanohub.org/tools/pc4learning>
- Build iteratively:
 - Set the domain
 - Add diffusing substrates
 - Add WT tumor cells and test
 - Add mutant tumor cells and test
 - Add macrophages and test
 - Add CD8+ T cells and test
 - Add monocytes and test

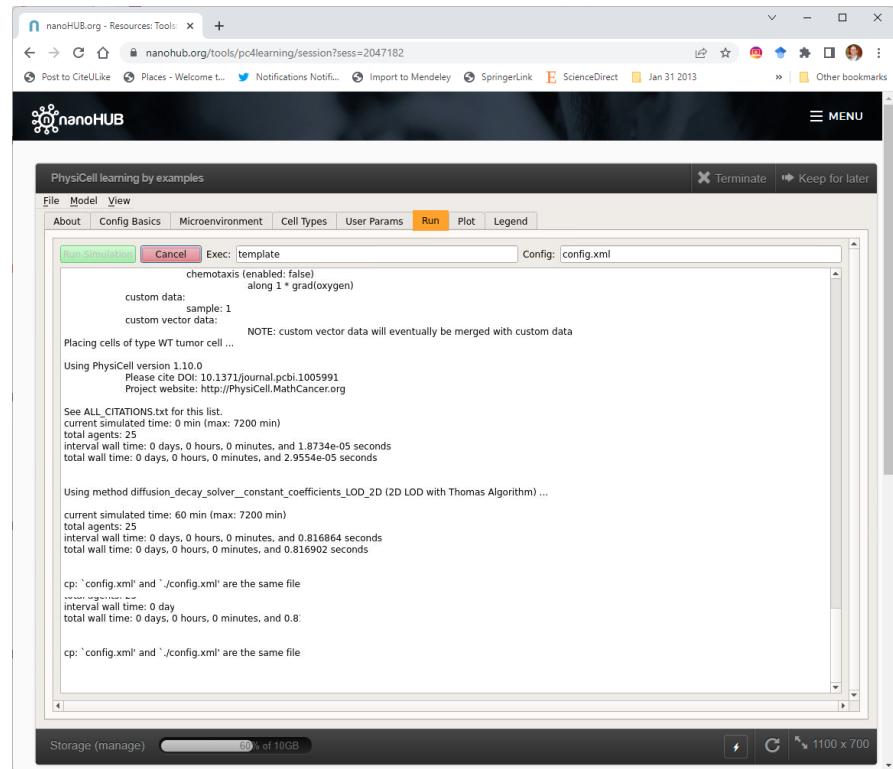
Define WT tumor cells

- Go to the **cell types** tab
- Double-click on **default**
 - Rename to **WT tumor cell**
- Go to **Secretion**
 - Choose **oxygen** from drop-down
 - ◆ Set uptake rate to 10
 - » 100 μm length scale in packed tissue
 - Choose **tumor factor**
 - ◆ Set secretion rate to 10
 - ◆ Set target to 1



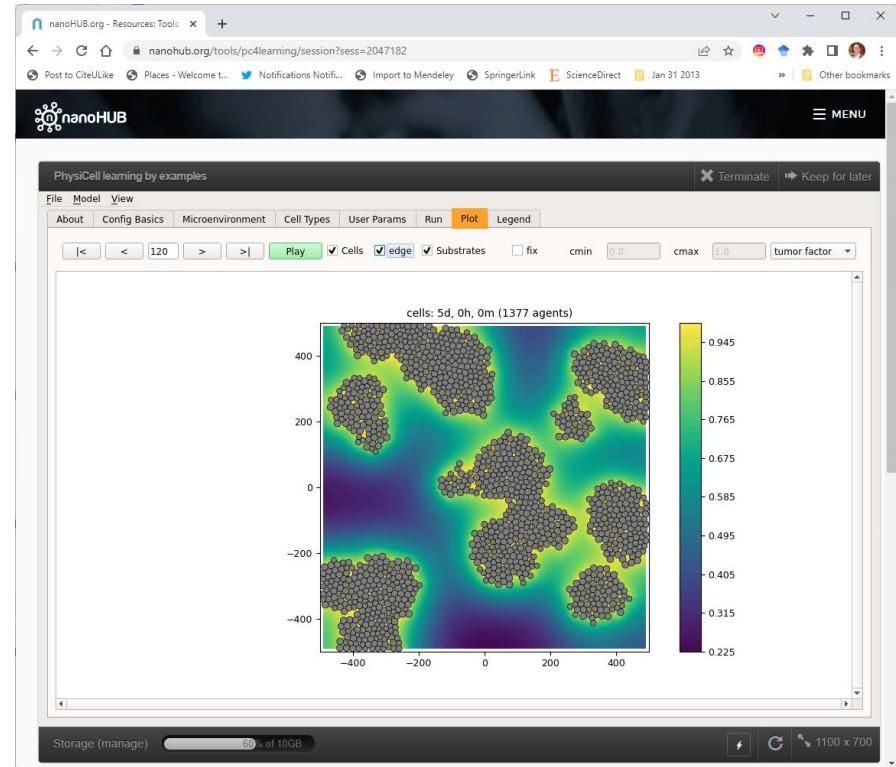
Test the model!

- Go to the **user params** tab
 - Set **number of cells** to 25
- Go to the **run** tab
 - Click **run simulation**



View the results

- Go to the **plot** tab
 - Click **play** to automatically animate
 - Click **pause** to stop playback
 - Click **<** or **>** to advance by 1 frame
 - Click **|< or >|** to go to the start or end
- Check **substrates** to plot substrates below the cells
 - Use the drop-down to choose the substrate

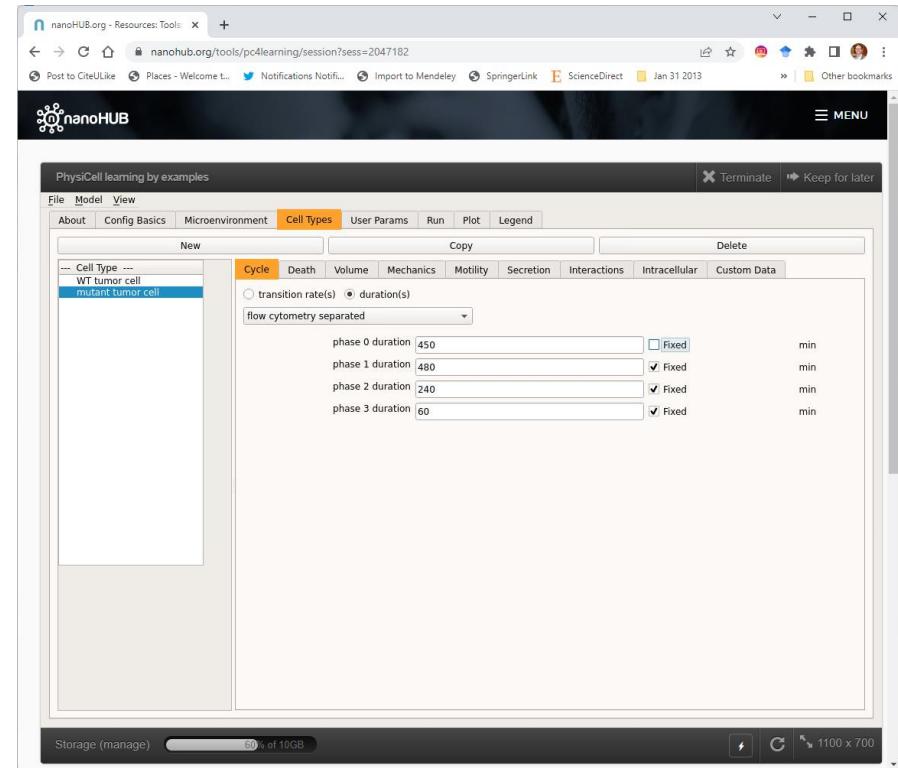


Checklist

- Use online simulator: <http://nanohub.org/tools/pc4learning>
- Build iteratively:
 - Set the domain
 - Add diffusing substrates
 - Add WT tumor cells and test
 - Add mutant tumor cells and test
 - Add macrophages and test
 - Add CD8+ T cells and test
 - Add monocytes and test

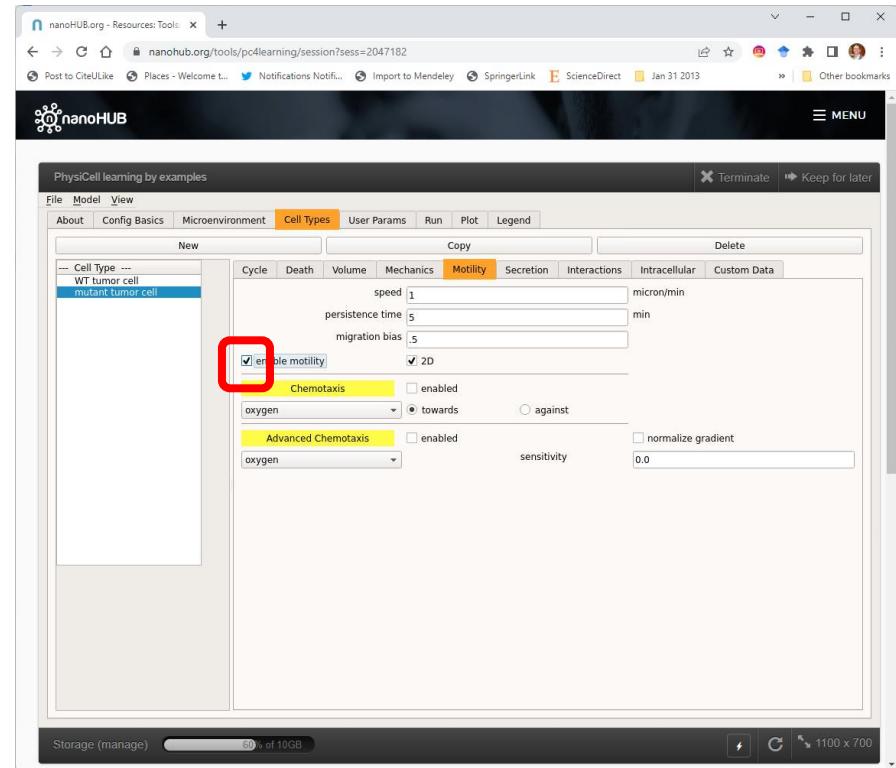
Define mutant tumor cells (1)

- Go to the **cell types** tab
- Click on **WT tumor cell**
- Copy the cell type
 - Rename it to **mutant tumor cell**
- Reduce cycling
 - Click on the **cycle** tab
 - Increase time in G0/G1 to **450** min.



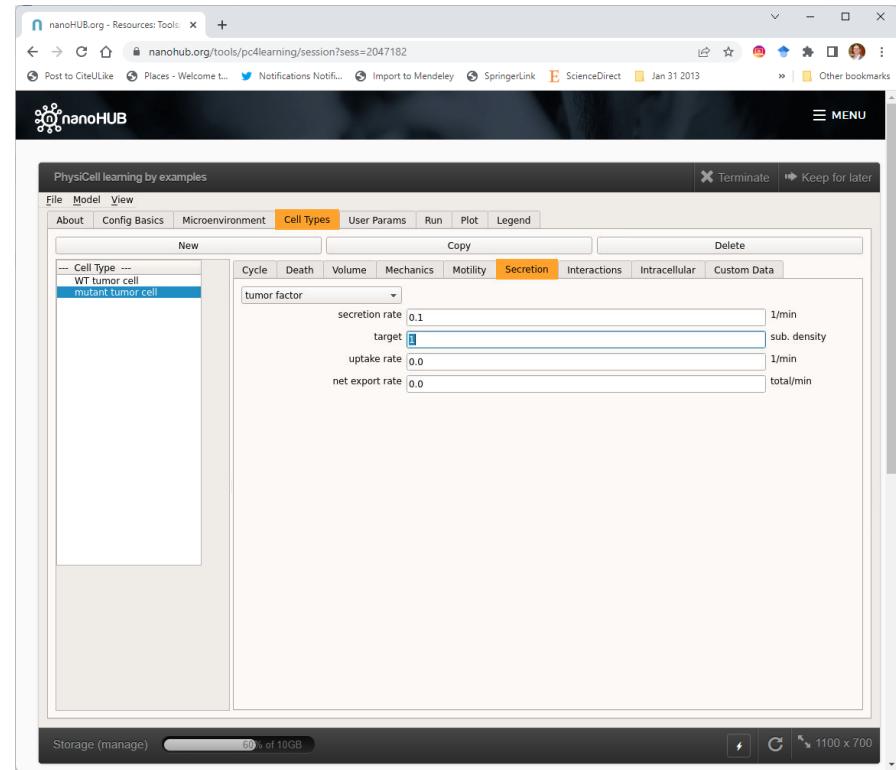
Define mutant tumor cells (2)

- Turn on random motility
 - Click on the **motility** tab
 - Leave mean speed at **1 $\mu\text{m}/\text{min}$**
 - Set persistence time to **5 min**
 - Leave bias at **0.5**
 - ◆ (doesn't affect this result)
 - Make sure to click the checkbox to enable motility!



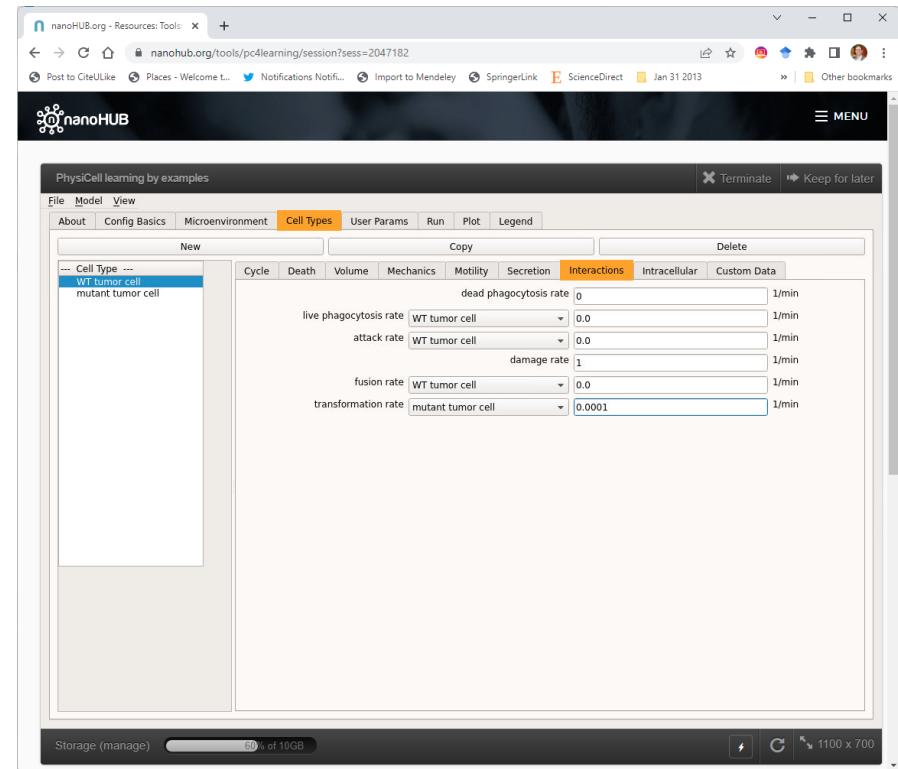
Define mutant tumor cells (3)

- Reduce tumor factor secretion
 - Click on the **secretion** tab
 - Choose **tumor factor** from the drop-down menu
 - Set the value to **0.1**



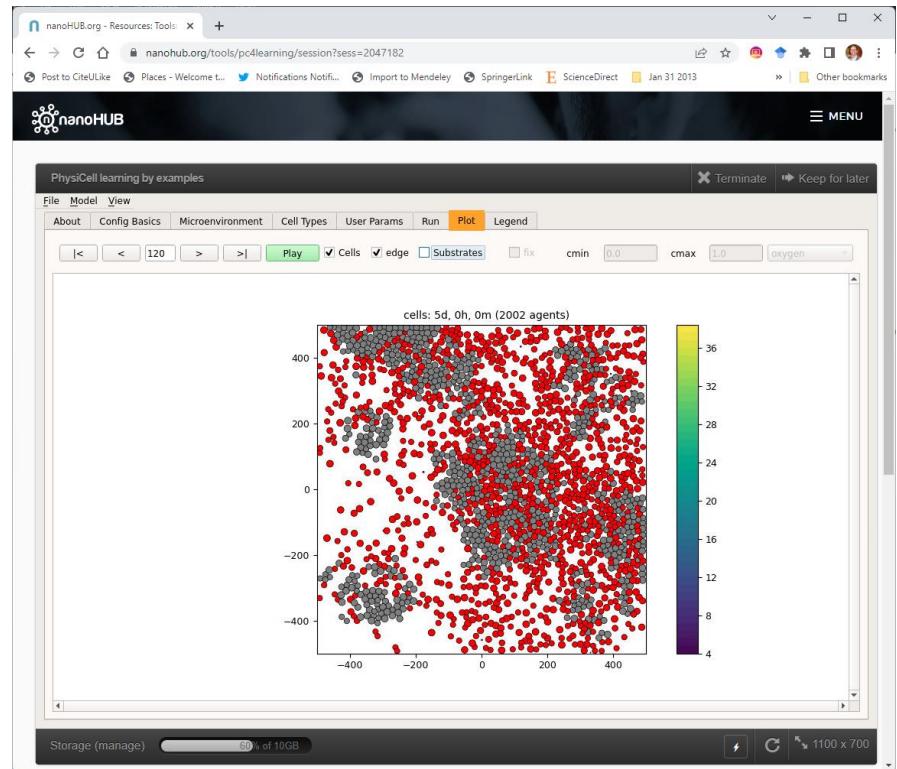
Enable mutations

- Select the **WT tumor cell** type
- Go to the **interactions** tab
- Go to **transformation rate**
- Choose **mutant tumor cell** from the drop-down
- Set the rate to **0.0001**



Test the model!

- Go to the **run** tab
 - Click **run simulation**
- Visualize the results in the **plot** tab
 - Grey: WT tumor cells
 - Red: mutant tumor cells
 - See the **legend** tab to verify colors
- Notice:
 - Red cells are motile
 - Red cells proliferate more slowly
 - Grey cells turn to red cells

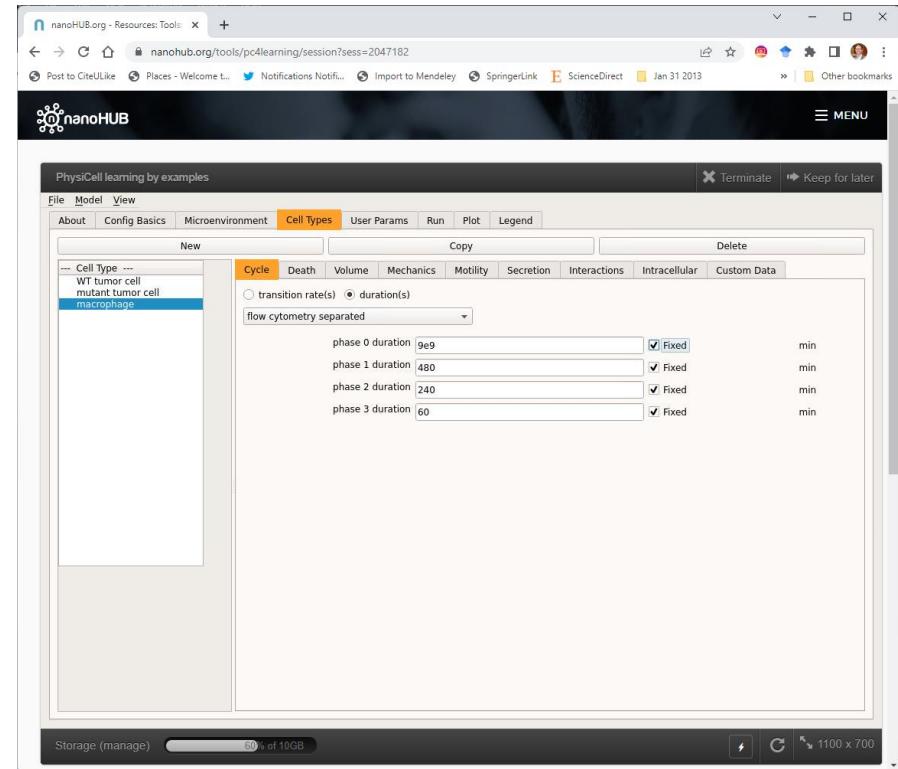


Checklist

- Use online simulator: <http://nanohub.org/tools/pc4learning>
- Build iteratively:
 - Set the domain
 - Add diffusing substrates
 - Add WT tumor cells and test
 - Add mutant tumor cells and test
 - Add macrophages and test
 - Add CD8+ T cells and test
 - Add monocytes and test

Define macrophages (1)

- Go to the **cell types** tab
- Click on **mutant tumor cell**
- Copy the cell type
 - Rename it to **macrophage**
- Disable cycling
 - Click on the **cycle** tab
 - Increase time in G0/G1 to **9e9** min with **fixed duration**



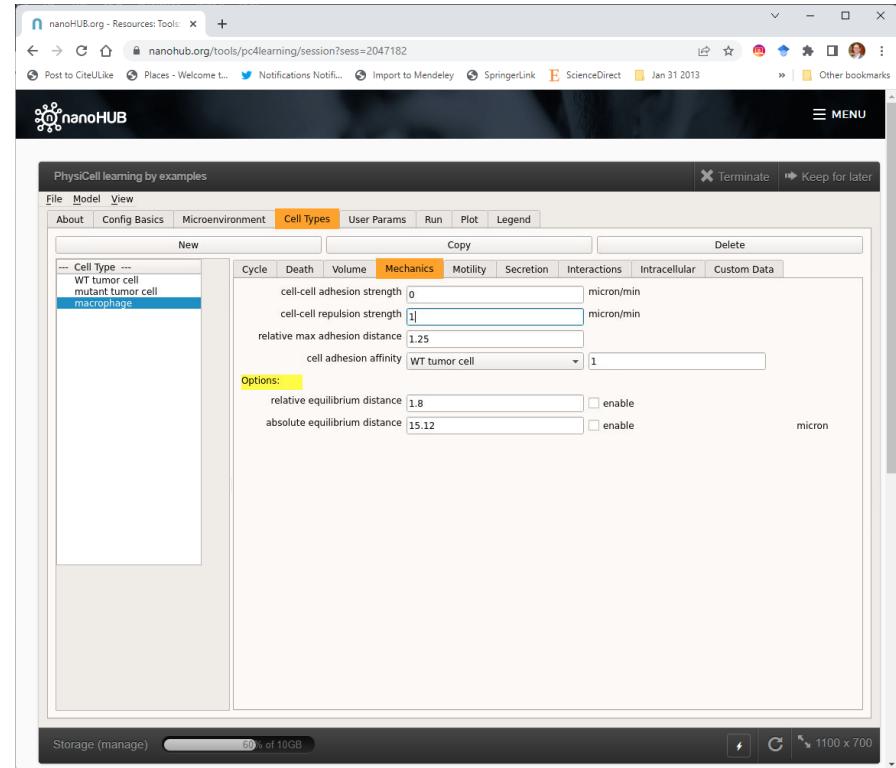
Define macrophages (2)

- Disable apoptotic death
 - Click on the **death** tab
 - Go to the section on **apoptosis**
 - Set the **death rate** to **0**

The screenshot shows the PhysiCell learning by examples software interface on nanoHUB.org. The main window title is "PhysiCell learning by examples". The top menu bar includes "File", "Model", "View", "About", "Config Basics", "Microenvironment", "Cell Types", "User Params", "Run", "Plot", and "Legend". The "Cell Types" tab is selected. A sidebar on the left lists cell types: "WT tumor cell", "mutant tumor cell", and "macrophage", with "macrophage" currently selected. The main panel is titled "Death" under the "Cycle" tab. It contains two sections: "Apoptosis" and "Necrosis". In the "Apoptosis" section, the "death rate" is set to 0. Under "Necrosis", the "death rate" is also set to 0. Other parameters listed include "unlysed fluid change rate", "lysed fluid change rate", "cytoplasmic biomass change rate", "nuclear biomass change rate", "calcification rate", and "relative rupture volume". The "Storage (manage)" bar at the bottom indicates 50% of 10GB used.

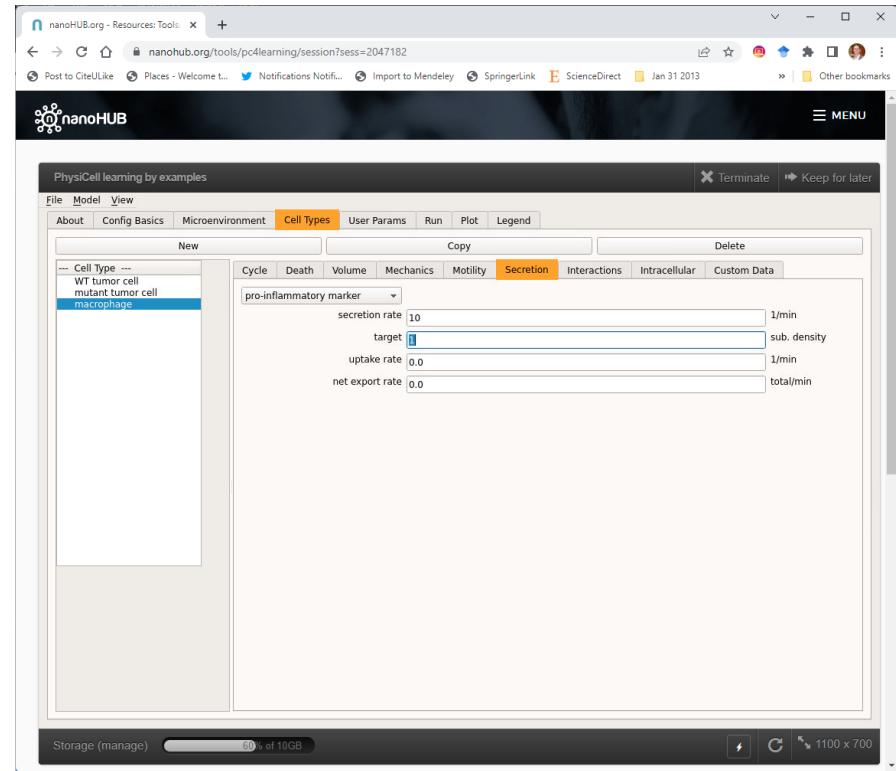
Define macrophages (3)

- Disable adhesion
 - Click on the **mechanics** tab
 - Set **cell-cell adhesion** to **0**
- Make the cells more deformable (allow more overlap)
 - Click on the **mechanics** tab
 - Set **cell-cell repulsion** to **1**



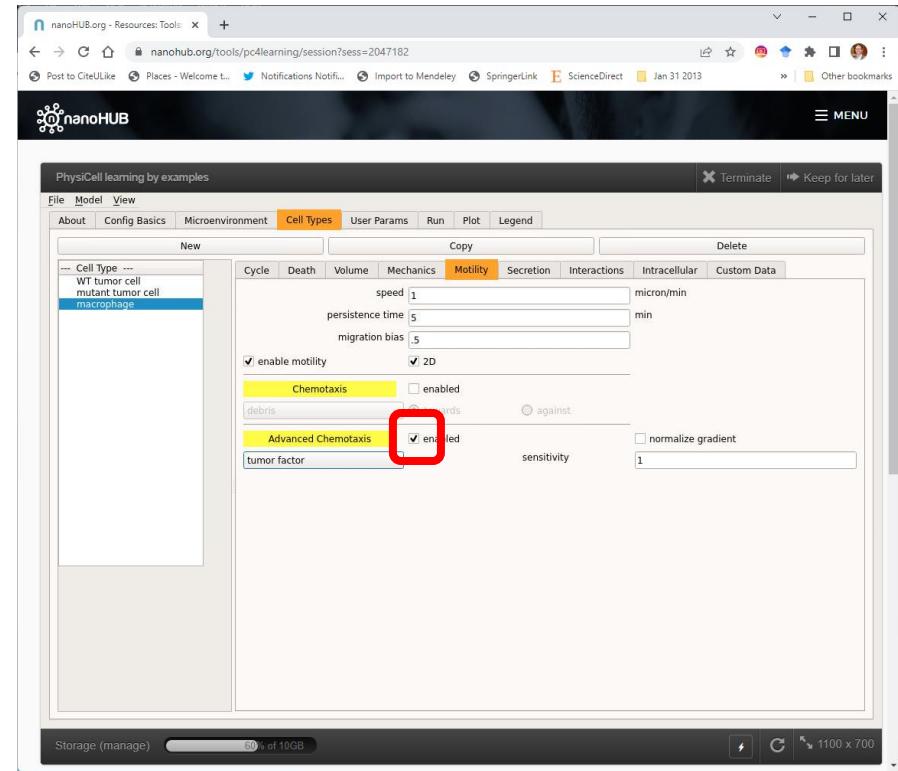
Define macrophages (3)

- Disable secretion of tumor factor
 - Go to the **secretion** tab
 - Choose **tumor factor** from the drop-down
 - Set the **secretion rate** to **0**
- Enable secretion of pro-inflammatory factor
 - Go to the **secretion** tab
 - Choose **pro-inflammatory factor** from the drop-down
 - Set the **secretion rate** to **10**
 - Set the **target** to **1**



Define macrophages (4)

- Turn on chemotaxis up both debris and tumor factor
 - Click on **motility**
 - Set **migration bias** at **0.5**
 - Make sure motility is enabled with the checkbox
 - Go to **advanced chemotaxis**
 - Choose **debris** in the drop-down and set its sensitivity to **0.1**
 - Choose **tumor factor** in the drop-down and set its sensitivity to **1**
 - Make sure you click the checkbox to enable advanced motility



LUDDY

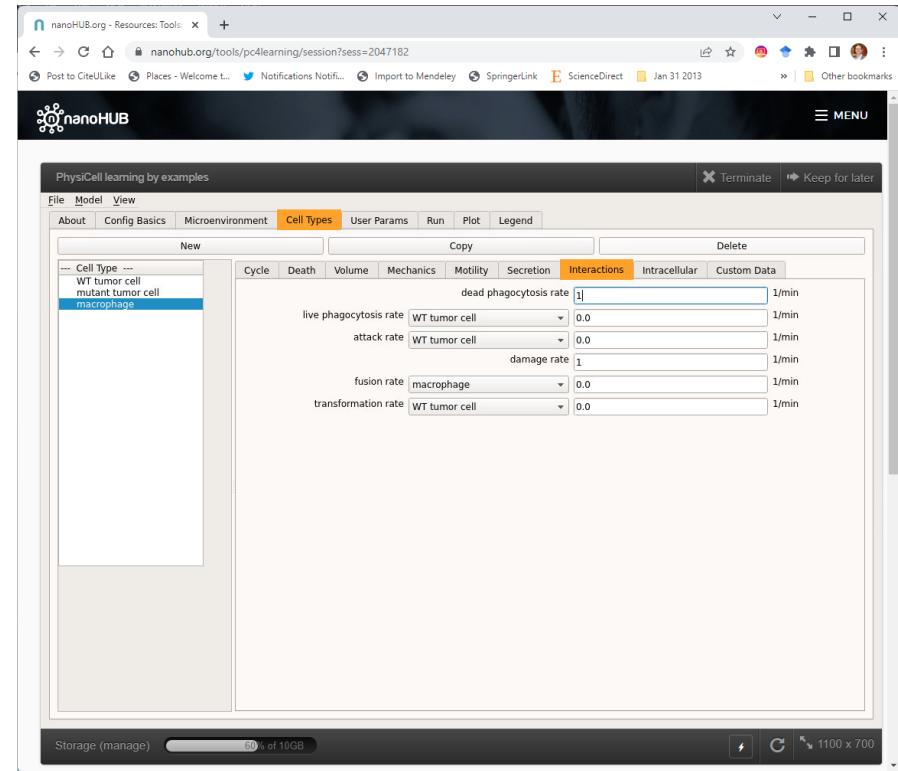
SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING

Macklin Lab
@MathCancer
MathCancer.org

Define macrophages (5)

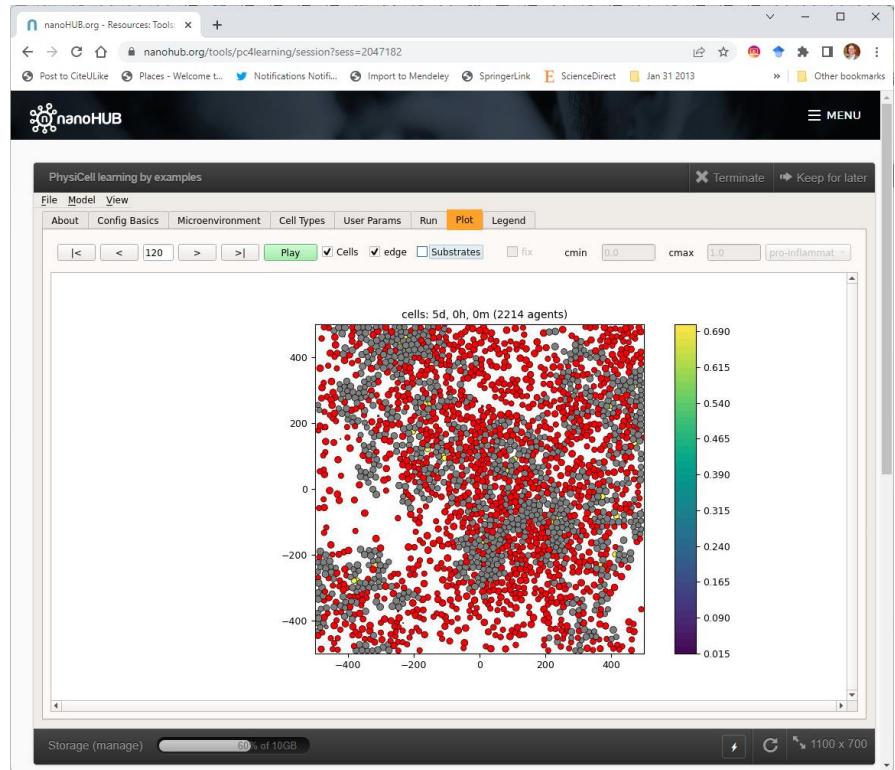
- Enable phagocytosis of dead cells
 - Click on **interactions**
 - Set the **dead phagocytosis rate** to 1

Note: If the dead cell phagocytosis rate is r , then the mean time the macrophage spends in contact with a dead cell before phagocytosing it is $\frac{1}{r}$.



Test the model!

- Go to the **run** tab
 - Click **run simulation**
- Visualize the results in the **plot** tab
 - Grey: WT tumor cells
 - Red: mutant tumor cells
 - Yellow: macrophages
- Notice:
 - Yellow cells will co-localize with tumor cells, with a preference for grey.

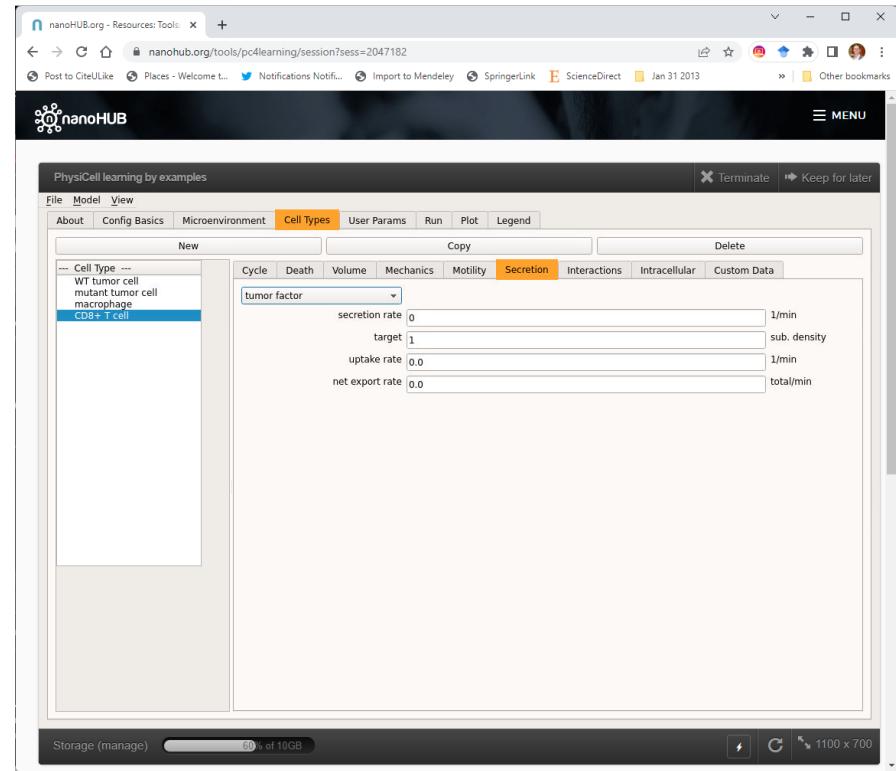


Checklist

- Use online simulator: <http://nanohub.org/tools/pc4learning>
- Build iteratively:
 - Set the domain
 - Add diffusing substrates
 - Add WT tumor cells and test
 - Add mutant tumor cells and test
 - Add macrophages and test
 - Add CD8+ T cells and test
 - Add monocytes and test

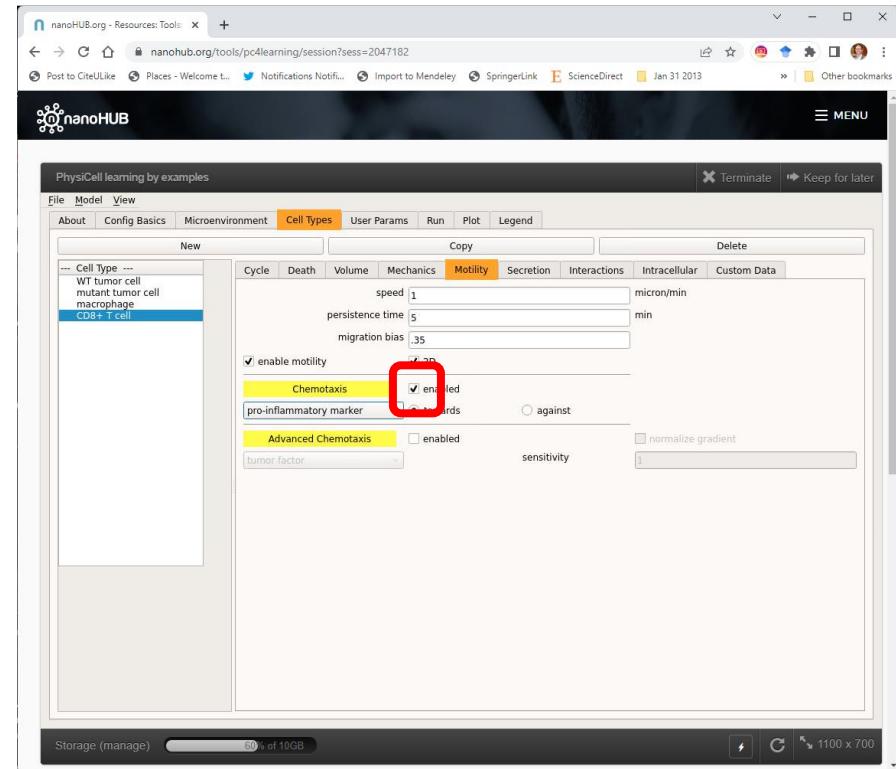
Define CD8+ T cells (1)

- Go to the **cell types** tab
- Click on **macrophage**
- Copy the cell type
 - Rename it to **CD8+ T cell**
- Disable secretion
 - Click on the **secretion** tab
 - Choose **pro-inflammatory factor** from the drop-down menu
 - Set the **secretion rate** to **0**



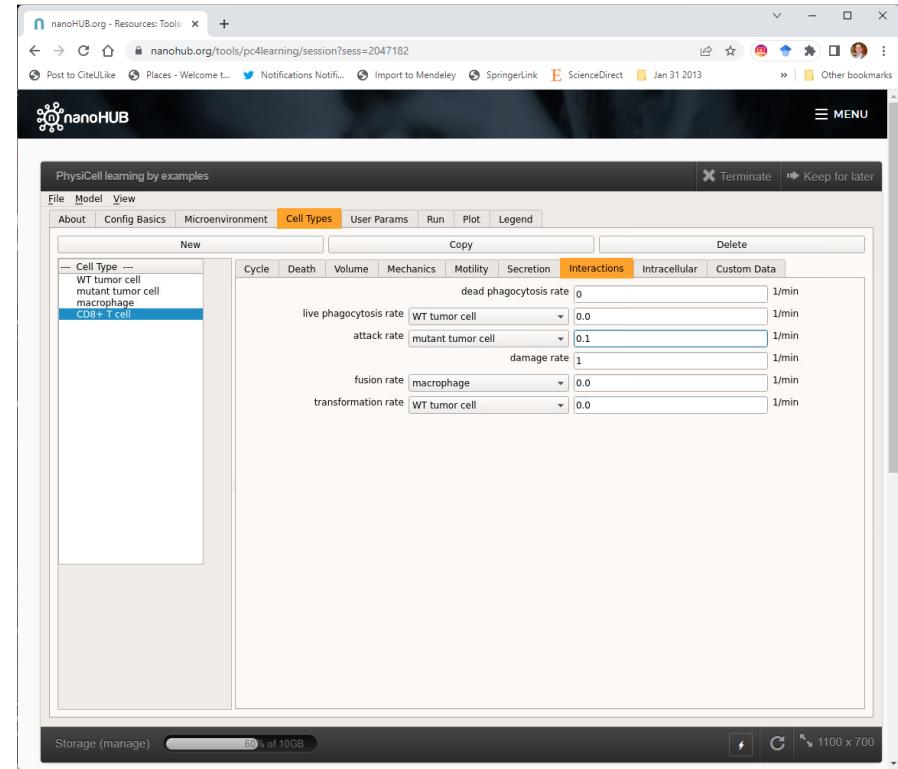
Define CD8+ T cells (2)

- Change to basic chemotaxis towards pro-inflammatory factor
 - Click on the **motility** tab
 - Uncheck to disable advanced chemotaxis
 - Check to enable chemotaxis
 - Set **migration bias** at **0.35**
 - Select **pro-inflammatory factor** from the drop-down menu
 - Make sure **towards** is selected for migration up the gradient



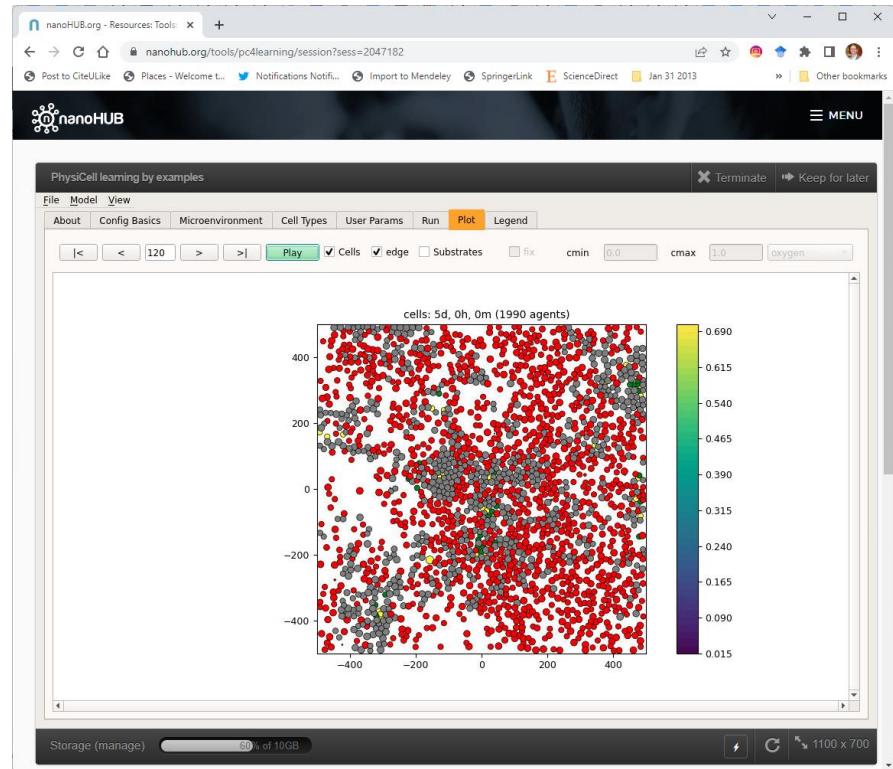
Define CD8+ T cells (3)

- Disable dead cell phagocytosis
 - Click on the **interactions** tab
 - Set **dead cell phagocytosis** to **0**
- Enable attack of tumor cells
 - Click on the **interactions** tab
 - Go to **attack rate**
 - ◆ Choose **WT tumor cell**
 - ◆ Set **attack rate to 10**
 - Use a reduced attack rate for mutants
 - ◆ Choose **mutant tumor cell**
 - ◆ Set **attack rate to 0.1**



Test the model!

- Go to the **run** tab
 - Click **run simulation**
- Visualize the results in the **plot** tab
 - Grey: WT tumor cells
 - Red: mutant tumor cells
 - Yellow: macrophages
 - Green: CD8+ T cells
- Notice:
 - Green cells will co-localize with macrophages
 - No extra cell death because we need a C++ function to tie death to damage.



See the extended version for more

- Define monocytes
 - phagocytose live tumor cells
- Full code for Part III (C++)



Source code, extended details and more:
GitHub.com/PhysiCell-Training/UCI-sysbio-2022

Part III:

Adding dynamical phenotypes with C++

Adding more advanced dynamics

- We are missing a few elements of our model:
 - Tumor cells:
 - ◆ Oxygen promotes cycling
 - ◆ High pressure reduces cycling
 - ◆ Damage increases apoptosis
 - ◆ Low oxygen induces necrosis
 - ◆ Dead cells release debris
- We'll use a C++ phenotype function to accomplish this

Note: We won't have time for this today, but the extended slides give step-by-step details. See the full working code in the GitHub repository:

<https://GitHub.com/PhysiCell-Training/UCI-sysbio-2022>

Approach

- Create a template project in PhysiCell
- Save the config file from our online version
- Declare a custom `tumor_phenotype` function `custom.h`
- Implement the custom model in `custom.cpp`
- Assign the custom phenotype function to WT tumor cells and mutant tumor cells
- Compile and run

End result

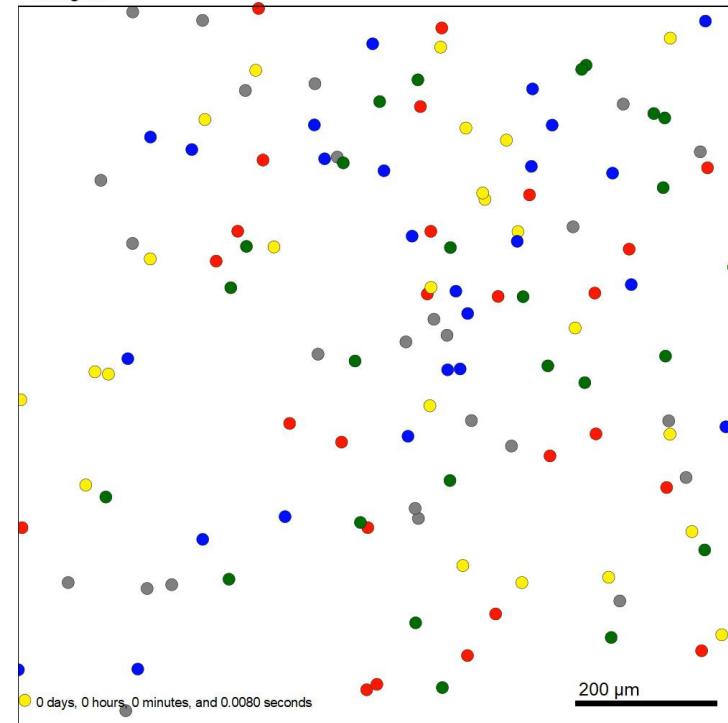
- Full details are in the extended slide deck.
- The completed code is in the session's GitHub repository



- WT tumor cell
- mutant tumor cell
- macrophage
- CD8+ T cell
- monocyte

Source code, extended details and more:
GitHub.com/PhysiCell-Training/UCI-sysbio-2022

Current time: 0 days, 0 hours, and 0.00 minutes, z = 0.00 μm
125 agents



Part IV:

Ongoing work and Opportunities

Continuous refinements for usability

- July 2017 (Version 1.2.0)
 - Restructured code around **cell phenotype**
- February 2018 (Version 1.3.0)
 - Introduced **XML-based configuration files**
- September 2018 (1.4.0)
 - **XML-based domain and user parameters**
- June 2019 (1.5.0)
 - Tracking of **internalized substrates**
 - Introduced standardized **predation** (ingest)
- August 2019 (1.6.0)
 - **XML-based setup** of the diffusion equations
- May 2020 (1.7.0)
 - Add **standardized chemotaxis** (with XML)
 - **XML-based cell definitions**
- March 2021 (1.8.0)
- July 2021 (1.9.0)
 - Formally released **XML-based cell definitions**
 - Introduced **cell contact functions**
 - Automatic tracking of **neighbor cells**
 - Auto-generated **legend**
 - Introduced **virtual walls**
 - Read a list of **initial cell locations** from CSV
- May 2022 (1.10.0)
 - Formalize **standardized intracellular modeling**
 - First support for **Model Builder GUI**

New work: reproducible model language

- As (hand-coded) model complexity grows:
 - Harder to understand the full model
 - Harder to clearly communicate the current biological hypotheses
 - Harder for domain experts to participate in real time
- **Goal:** Create a formalism for agent rules that:
 - Can be written in human-readable "plain English"
 - ◆ Facilitates tools for easy model construction
 - ◆ *Turns model building into knowledge mapping*
 - Can readily be "translated" to a standard mathematical form
 - ◆ Model can parse the rules without hand-coding
 - ◆ More reusable, maintainable model

Example: chemokine-driven cycling

- Biological hypothesis statement: INFG promotes cell cycling

- Rule: INFG increases cell cycle entry

- Mathematical translation:

$$r_{\text{cycle}} = r_0 + (10r_0 - r_0) \frac{[\text{INFG}]^{1.5}}{0.1^{1.5} + [\text{INFG}]^{1.5}}$$

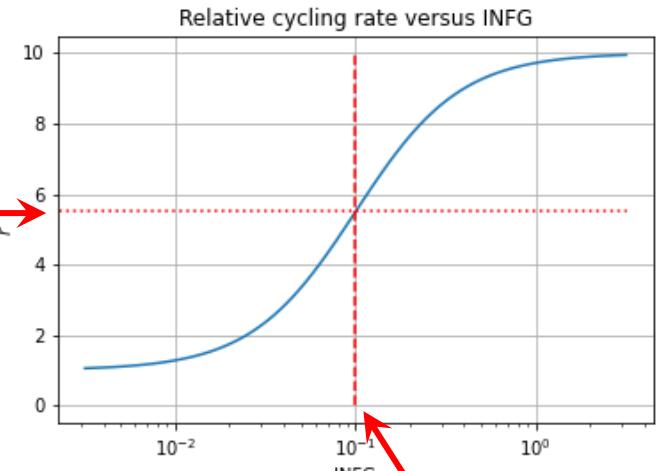
- Refined rule (with parameters):

▪ Hill response function	
▪ Hill power:	1.5
▪ half-max:	0.1
▪ base value:	r_0
▪ tenfold max response:	$r_U = 10 r_0$

- XML markup:

```
<rule>
    <signal name="INFG"/>
    <behavior name="cycle entry"/>
    <response type="increase" form="hill">
        <max_response type="relative">10</max_response>
        <hill_power>1.5</hill_power>
        <half_max>0.1</half_max>
    </response>
</rule>
```

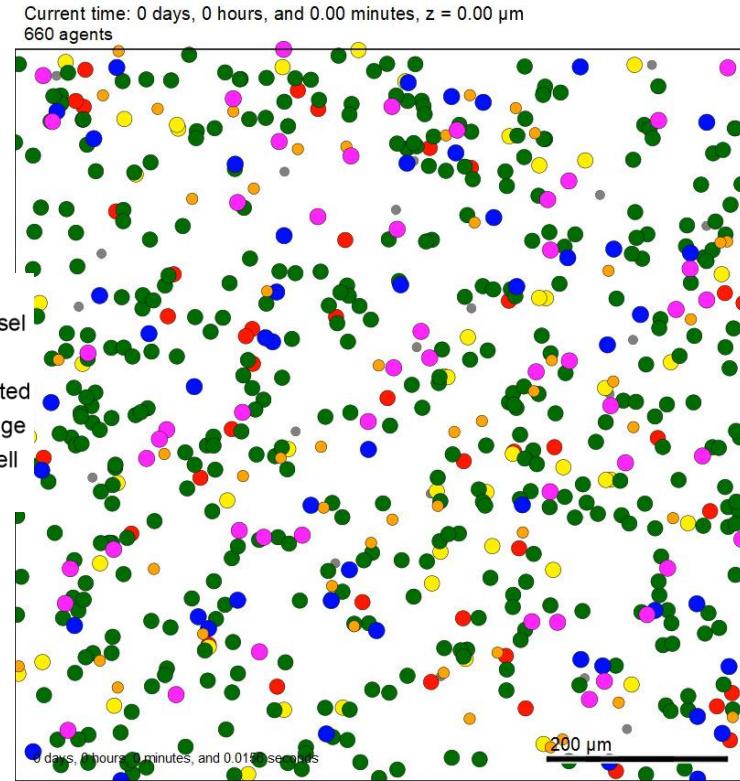
Half of max response



half-max level

Example: Tissue versus virulent bacteria

- Stem cells
 - Divide, differentiate
 - Killed by toxin
- Differentiated cells
 - Divide
 - Killed by toxin
- Blood vessels
 - Release resource
- Bacteria
 - Colonize near resources (via quorum)
 - Release toxin
 - Killed by damage
- Macrophages
 - Phagocytose dead cells
 - Release pro-inflammatory factor
- CD8+ T cells
 - Attracted to pro-inflammatory factor
 - Damage bacteria
- Neutrophils
 - Attracted to pro-inflammatory factor
 - Phagocytose bacteria



LUDDY

SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING

The future is modeling in *real time* with knowledge mapping.

1. Meet with domain experts to formulate behavioral hypotheses
 - Oxygen increases cycle entry in tumor cells
 - Oxygen decreases necrotic death in tumor cells
 - Oxygen decreases secretion of VEGF in tumor cells
 - VEGF increases motility in endothelial cells
 - Contact with endothelial cells inhibits proliferation in endothelial cells
 - Cell debris increases secretion of IL-6 by macrophages
 - ...
2. Immediately import the rules and simulate behavior
3. Get domain expert feedback in real time to improve the hypotheses

Over time, we'll create community-curated library of reusable behavioral hypotheses.

Last Year's Virtual Training Course

15 Virtual Sessions:

- PhysiCell Essentials and Modeling Workflows
- Graphical Model Editor
- Phenotype
- Microenvironment
- Functions
- Chemical Communication / Interactions
- Contact Communication / Interactions
- Intracellular Modeling with ODEs / SBML
- Extensions for high performance computing (HPC)
- Cloud-hosted Model Sharing
- ... and more!

Sessions include:

- Slides (PDF format)
- YouTube recordings
- Source code

github.com/PhysiCell-Training/ws2021



Apply Today! PhysiCell ws2022

- July 24-30, 2022
- Fully virtual
- Tutorial sessions option to general public
- Competitive selection for mentored hackathon



2022 Virtual PhysiCell Workshop and Hackathon
July 24-30, 2022



- Build and explore multicellular agent-based simulations of cancer and other systems
- Learn to share your models online
- Meet other modelers in the CSBC / PS-ON community
- Compete in an exclusive mentored hackathon
- PhysiCell swag for accepted participants
- Application and full agenda at QR code or:
<https://github.com/PhysiCell-Training/ws2022>

