Introduction to multiscale modelling with UPMaBoSS and PhysiBoSS

Vincent Noël[1]

[1]Computational Systems Biology of Cancer,
Institut Curie, INSERM U900, Mines Paris, PSL Research University, Paris

BI8040 - Logical Modelling for Experimental Design
in Current and Future Biotechnology and Biomedicine

August 2022









Outline

- Introduction
- UPMaBoSS : Dynamic cell population modeling
  - Cell fate model
  - Immunogenic Cell Death
- PhysiBoSS : Multi-scale, agent based modeling framework
  - Agent based modeling
  - PhysiCell
  - Cell fate model
  - Multiscale model of the different modes of invasion
  - PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2

Introduction

› Boolean models

OPEN ACCESS Freely available online

PLOS COMPUTATIONAL BIOLOGY

Mathematical Modelling of Cell-Fate Decision in Response to Death Receptor Engagement

Laurence Calzone[1,2,3]*, Laurent Tournier[1,2,3], Simon Fourquet[1,2,3], Denis Thieffry[4,5], Boris Zhivotovsky[6], Emmanuel Barillot[1,2,3¶], Andrei Zinovyev[1,2,3¶]

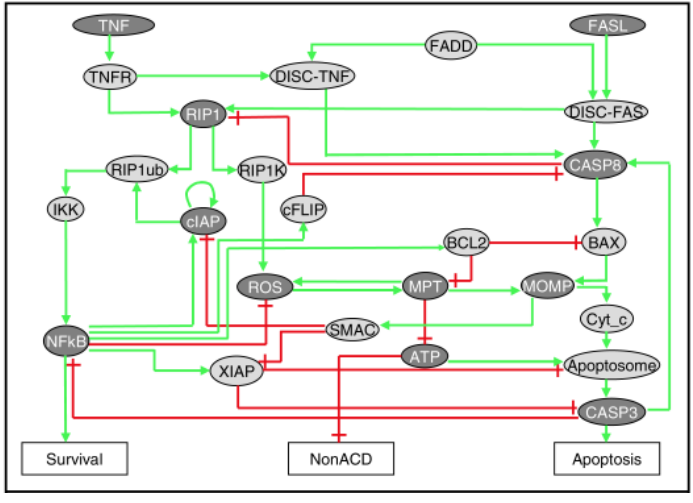
1 Institut Curie, Paris, France, **2** Ecole des Mines ParisTech, Paris, France, **3** INSERM U900, Paris, France, **4** TAGC – INSERM U928 & Université de la Méditerranée, Marseille, France, **5** CONTRAINTES Project, INRIA Paris-Rocquencourt, France, **6** Karolinska Institutet, Stockholm, Sweden

Abstract

Cytokines such as TNF and FASL can trigger death or survival depending on cell lines and cellular conditions. The mechanistic details of how a cell chooses among these cell fates are still unclear. The understanding of these processes is important since they are altered in many diseases, including cancer and AIDS. Using a discrete modelling formalism, we present a mathematical model of cell fate decision recapitulating and integrating the most consistent facts extracted from the literature. This model provides a generic high-level view of the interplays between NFκB pro-survival pathway, RIP1-dependent necrosis, and the apoptosis pathway in response to death receptor-mediated signals. Wild type simulations demonstrate robust segregation of cellular responses to receptor engagement. Model simulations recapitulate documented phenotypes of protein knockdowns and enable the prediction of the effects of novel knockdowns. *In silico* experiments simulate the outcomes following ligand removal at different stages, and suggest experimental approaches to further validate and specialise the model for particular cell types. We also propose a reduced conceptual model implementing the logic of the decision process. This analysis gives specific predictions regarding cross-talks between the three pathways, as well as the transient role of RIP1 protein in necrosis, and confirms the phenotypes of novel perturbations. Our wild type and mutant simulations provide novel insights to restore apoptosis in defective cells. The model analysis expands our understanding of how cell fate decision is made. Moreover, our current model can be used to assess contradictory or controversial data from the literature. Ultimately, it constitutes a valuable reasoning tool to delineate novel experiments.

Introduction

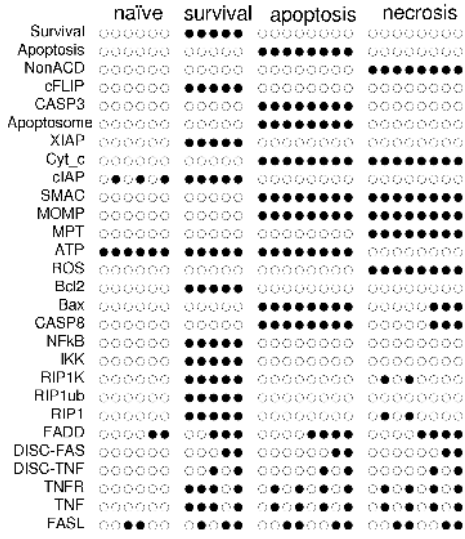
› Boolean models



Calzone et al, 2010

Introduction

- Boolean models
  - Fixed points
  - Depending on input values
  - Can be stochastic (update mode)
  - Possible phenotypes (cell fates)



Calzone et al, 2010

Introduction

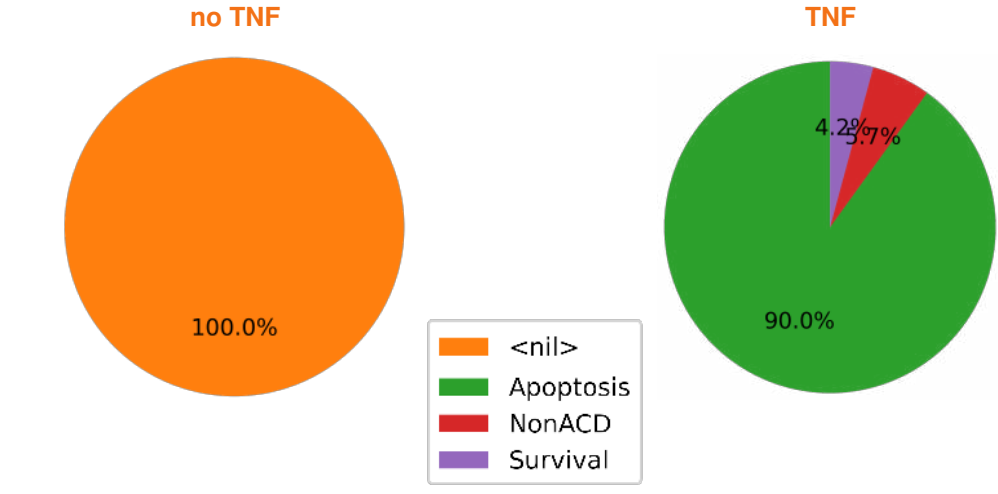
› Boolean models

Name	Modified rules	Expected phenotypes	Qualitative results
Anti-oxidant	ROS′ = (RIP1 OR MPT)	Prediction.	Suppression of NFκB anti-oxidant effect leads to no change in the decision process.
APAF1 deletion	C3′ = 0	*APAF1*$^{-/-}$ mouse thymocytes are not impaired in FAS-mediated apoptosis ([71]).	Apoptosis disappears and replaced by the naïve state. Necrosis and survival are close to the wild type case situations.
BAX deletion	MOMP′ = MPT	*BAX* deletion blocks FAS or TNF+CHX - induced apoptosis in some cell lines, such as HCT116 [72].	*BAX* deletion prevents apoptosis.
BCL2 over-expression	MOMP′ = MPT MPT′ = 0	FAS induces the activation of NFκB pathway [29].	As expected, the survival and naive attractors are preserved while both death pathways are inhibited.
CASP8 deletion	C8′ = 0	*Caspase-8* deficient MEFs [41] or Jurkat cells [42] are resistant to FAS-mediated apoptotic cell death.	Apoptosis disappears. Compared to the wild type, a slight increase of necrosis is observed, while survival becomes the main cell fate.
constitutively activated CASP8	C8′ = 1	Prediction.	Over-expression of caspase-8 leads to a loss of NFκB activation.
cFLIP deletion	C8′ = TNF OR FAS OR C3	*cFLIP*−/− MEFs are highly sensitive to FASL and TNF [73].	The increase of apoptosis is effectively observed in the *cFLIP* mutant; furthermore survival can no longer be sustained.
cIAP deletion	cIAP′ = 0	NFκB activation in response to TNF is blocked [53].	NFκB activation is impaired, and only the apoptotic and necrotic attractors can be reached.
FADD deletion	C8′ = C3 AND NOT NFκB RIP1′ = NOT C8 AND TNF	*FADD*$^{-/-}$ mouse thymocytes are resistant to FAS mediated apoptosis [74]. *FADD*$^{-/-}$ MEFs are resistant to FASL and TNF [75]. In Jurkat cells treated with TNF+CHX, apoptosis is turned into necrosis [43].	FASL signalling is blocked and the 'naïve' attractor is the only reachable one. In response to TNF, apoptosis disappears.
NFκB deletion	NFkB′ = 0	TNF induces both apoptosis and necrosis in *NF-κB p65*$^{-/-}$ cells [76] or in IKKb$^{-/-}$ fibroblasts [35].	This mutant shows a strong increase of necrosis (to be related with concomitant apoptosis/necrosis).
constitutively active NFκB	NFkB′ = 1	Prediction.	Both death pathways are shut down in this mutant.
RIP1 deletion	RIP1′ = 0	*RIPK1*$^{-/-}$ MEFs are hypersensitivity to TNF, no TNF-induced NFκB activation, [77].	Both survival and necrosis states become unreachable. The effect of RIP1 silencing leads to a complete loss of the decision process (apoptosis becoming the only outcome).

Calzone et al, 2010

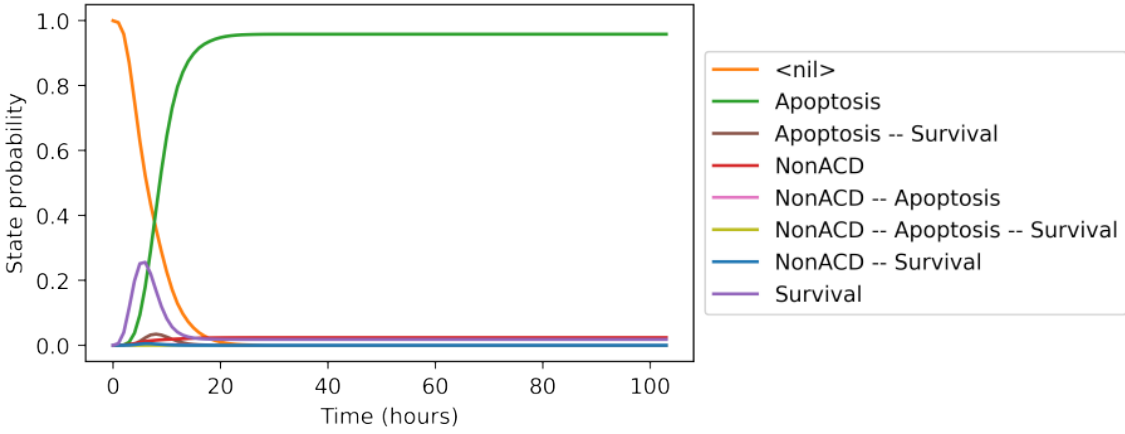
Introduction

› MaBoSS : Continuous time boolean modelling



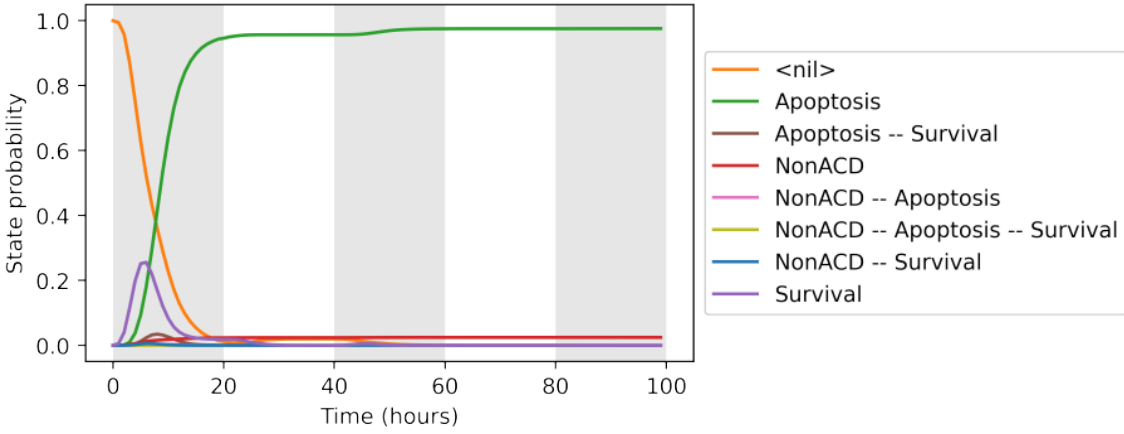
Introduction

› MaBoSS : Continuous time boolean modelling

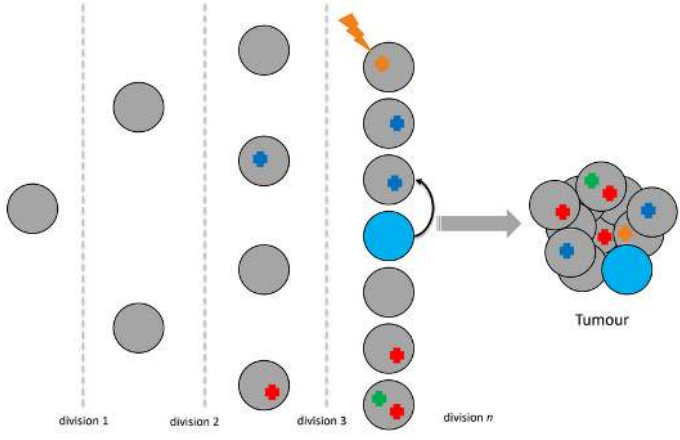


Introduction

› MaBoSS : Continuous time boolean modelling



For some problems, population dynamics are important : We need to consider heterogeneous cell populations, cell divisions, death, mutations, and interactions.



UPMaBoSS : Dynamic cell population modeling

UPMaBoSS : Dynamic cell population modeling

frontiers in Molecular Biosciences

METHODS
published: 02 March 2022
doi: 10.3389/fmolb.2022.800152

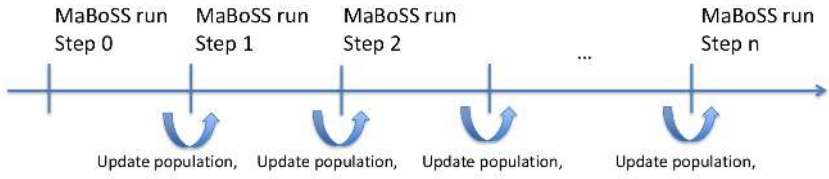
UPMaBoSS: A Novel Framework for Dynamic Cell Population Modeling

*Gautier Stoll[1,2], Aurélien Naldi[3,4], Vincent Noël[5,6,7], Eric Viara[8], Emmanuel Barillot[5,6,7], Guido Kroemer[1,2,9], Denis Thieffry[3,4] and Laurence Calzone[5,6,7]**

[1]Equipe Labellisée Par La Ligue Contre Le Cancer, Université de Paris, Sorbonne Université, INSERM UMR1138, Centre de Recherche des Cordeliers, Paris, France, [2]Metabolomics and Cell Biology Platforms, Gustave Roussy Cancer Campus, Université Paris Saclay, Villejuif, France, [3]Institut de Biologie de L'ENS (IBENS), Ecole Normale Supérieure, CNRS, INSERM, Université PSL, Paris, France, [4]Lifeware Group, Inria Saclay-Ile de France, Palaiseau, France, [5]Institut Curie, PSL Research University, Paris, France, [6]INSERM U900, Paris, France, [7]MINES ParisTech, CBIO-Centre for Computational Biology, PSL Research University, Paris, France, [8]Sysra, Yerres, France, [9]Pôle de Biologie, Hôpital européen Georges Pompidou, AP-HP, Paris, France*

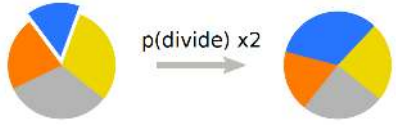
UPMaBoSS : Dynamic cell population modeling

- We start from an existing MaBoSS model
- We add model nodes for division and death
- We add inter-cellular communication
- Chaining MaBoSS simulations, updating population and communication between each



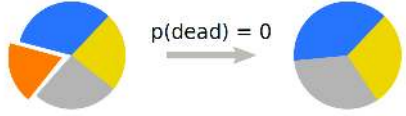
UPMaBoSS : Dynamic cell population modeling

Cell division



$$population = population * (1 + p(divide))$$

Cell death



$$population = population * (1 - p(dead))$$

Intercellular communication

Update the environmental condition of other cells

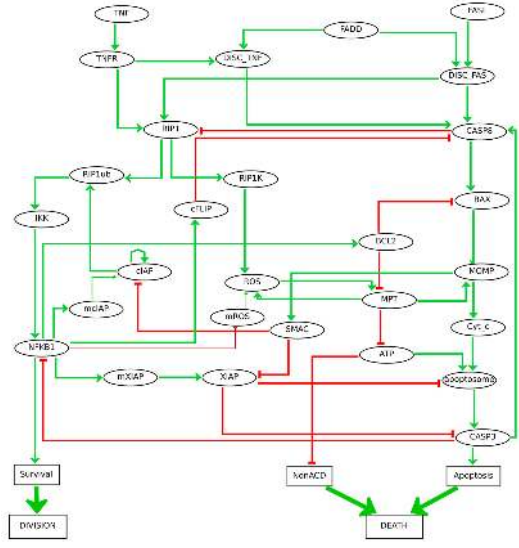
UPMaBoSS : Dynamic cell population modeling

› Cell fate model

We map the cell fates to population updates :

› Survival node : Cell division

› Apoptosis and NonACD : Cell death



Adapted from Stoll et al., 2022

UPMaBoSS : Dynamic cell population modeling

› Cell fate model

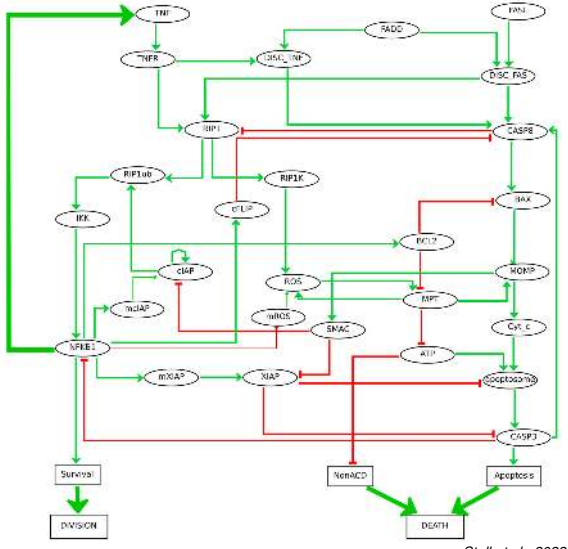
TNF paracrine signaling :

› TNF activation will depend on $TNF_induc parameter

```
node TNF
{
  rate_up = $TNF_induc;
  rate_down = $Degr_TNF;
}
```

› $TNF_induc will depend on population size of non-dead NFkB active cells

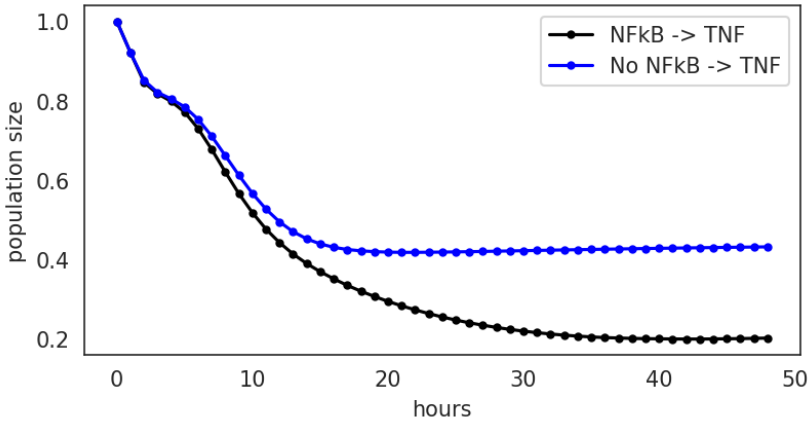
```
$TNF_induc u= $ProdTNF_NFkB*p[(NFkB,Death) = (1,0)];
```



Stoll et al., 2022

UPMaBoSS : Dynamic cell population modeling

› Cell fate model

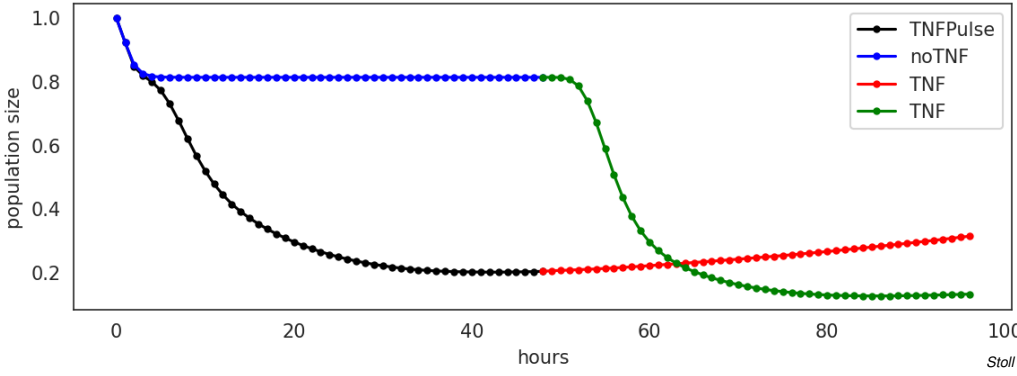


Stoll et al., 2022

› Study the effect of TNF paracrine signaling

UPMaBoSS : Dynamic cell population modeling

› Cell fate model



Stoll et al., 2022

› Study the resistance to TNF treatment

Immunogenic Cell Death

UPMaBoSS : Dynamic cell population modeling

› Immunogenic Cell Death

frontiers in Physiology

ORIGINAL RESEARCH
published: 12 November 2020
doi: 10.3389/fphys.2020.590479

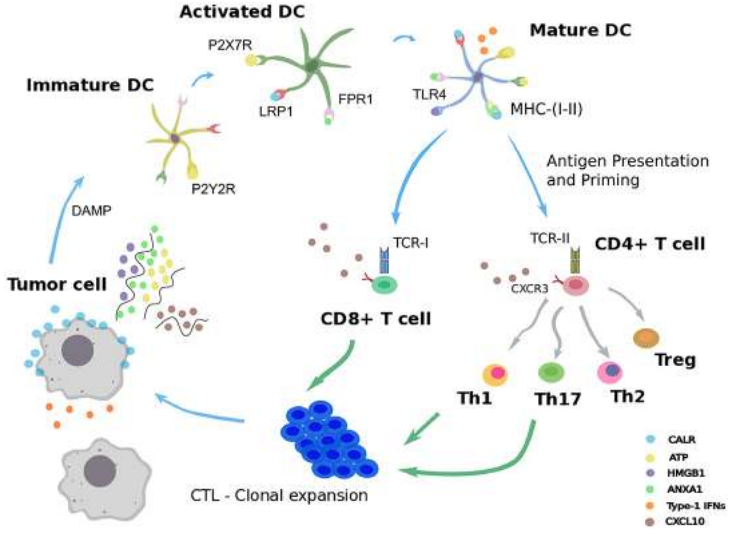
Dynamical Boolean Modeling of Immunogenic Cell Death

Andrea Checcoli[1], Jonathan G. Pol[2,3], Aurelien Naldi[1], Vincent Noel[4,5,6], Emmanuel Barillot[4,5,6], Guido Kroemer[2,3,7,8,9], Denis Thieffry[1], Laurence Calzone[4,5,6†] and Gautier Stoll[2,3*†]*

[1] Institut de Biologie de l'ENS (IBENS), Département de Biologie, École Normale Supérieure, CNRS, INSERM, Université PSL, Paris, France, [2] Equipe 11 labellisée par la Ligue Nationale contre le Cancer, Centre de Recherche des Cordeliers, INSERM U1138, Université de Paris, Sorbonne Université, Institut Universitaire de France, Paris, France, [3] Metabolomics and Cell Biology Platforms, Gustave Roussy Cancer Campus, Villejuif, France, [4] Institut Curie, PSL Research University, Paris, France, [5] INSERM, U900, Paris, France, [6] MINES ParisTech, PSL Research University, CBIO-Centre for Computational Biology, Paris, France, [7] Pôle de Biologie, Hôpital Europeen Georges Pompidou, AP-HP, Paris, France, [8] Suzhou Institute for Systems Biology, Chinese Academy of Sciences, Suzhou, China, [9] Department of Women's and Children's Health, Karolinska University Hospital, Stockholm, Sweden

UPMaBoSS : Dynamic cell population modeling

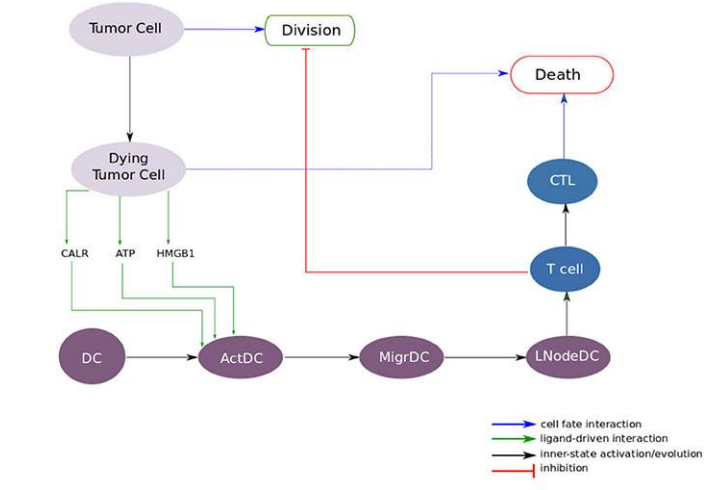
› Immunogenic Cell Death

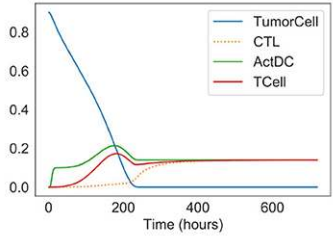


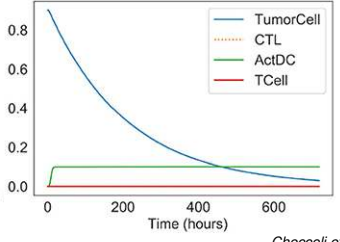
Checcoli et al., 2020

UPMaBoSS : Dynamic cell population modeling

› Immunogenic Cell Death



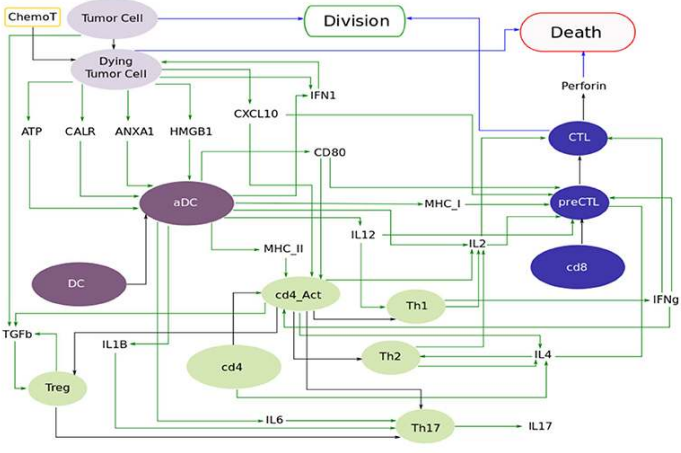


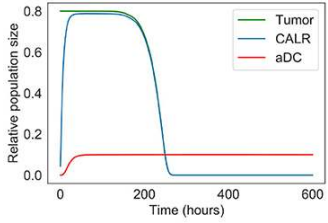


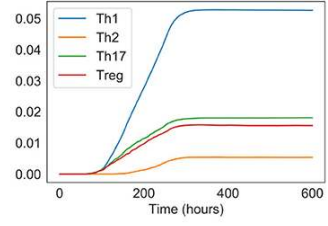
Checcoli et al., 2020

UPMaBoSS : Dynamic cell population modeling

› Immunogenic Cell Death







Checcoli et al., 2020

UPMaBoSS : Dynamic cell population modeling

- Immunogenic Cell Death
  - Experimental, Data analysis, Modeling done by Andrea
  - Will shed light on the triggering of ICD by different immunotherapies


Andrea Checcoli


Gautier Stoll


Jonathan Pol


Denis Thieffry


Laurence Calzone

institutCurie

GUSTAVE ROUSSY
CANCER CAMPUS GRAND PARIS

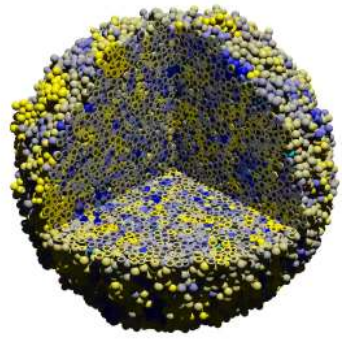
IBENS

UPMaBoSS : Dynamic cell population modeling

- Perspectives
  - A simple yet powerful framework for population modeling
  - Available as a part of pyMaBoSS
  - One hyper parameter : update time step
  - PopMaBoSS : Toward a full probabilistic population modeling framework

For some problems (interaction with microenvironment, tissue organisation), we need spatial information

- Tumor composition
- Tumor growth
- Nutrients (Necrosis)
- Killer cells
- Resistant cells



PhysiBoSS : Multi-scale, agent based modeling framework

› Agent based modeling

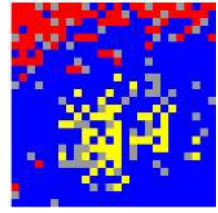
We simulate multiple agents with defined behaviours. In our case we mainly have cells that :

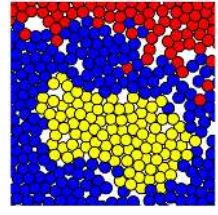
- Can proliferate/die
- Can grow/shrink
- Can move in space
- Can interact with other agents

⇒ Simple behavior for each agent, where complex phenomena can emerge in the system

PhysiBoSS : Multi-scale, agent based modeling framework

- Agent based modeling
  - Lattice-based
    - Each site correspond to one agent (Ex: NetLogo, Chaste)
    - Agents can span several sites (Ex: CompuCell3D)
    - Multiple agents can occupy one site (Ex: COMETS)
  - Lattice-free
    - Vertex models, where space is dynamically divided between agents (Ex: Chaste)
    - Center-based models, where agents are defined by their center position, and their volume (Ex: PhysiCell).





Daly et al., 2019

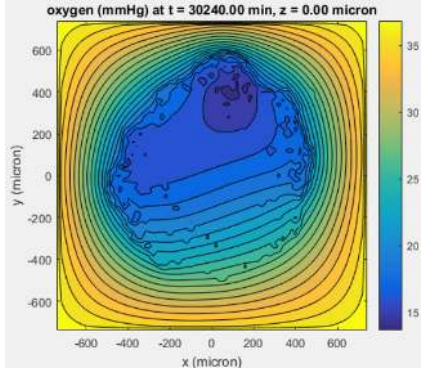
PhysiBoSS : Multi-scale, agent based modeling framework

› Modeling molecules diffusing

› We want to represent molecules floating around

› These molecules can diffuse inside the environment

› These molecules can interact with agents



PhysiBoSS : Multi-scale, agent based modeling framework

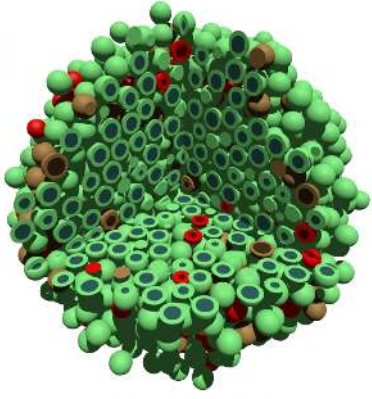
› PhysiCell : Multiscale Modelling Framework



Paul Macklin



Randy Heiland



http://physicell.org
https://github.com/MathCancer/PhysiCell

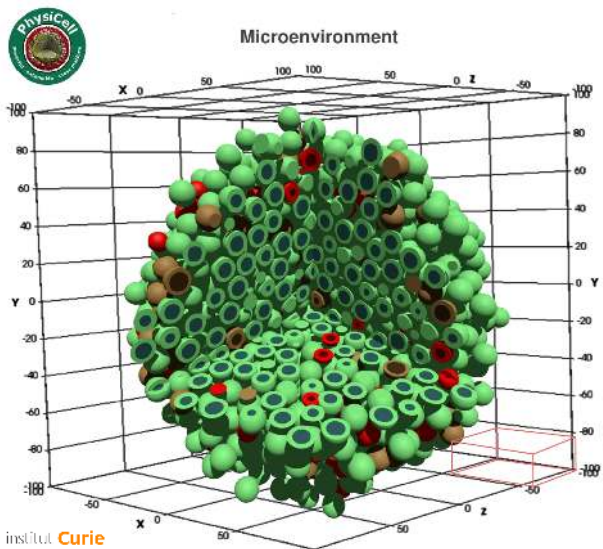
Cell agent properties

- Size (cell volume)
- Position (x, y, z)
- Phenotype
  - Cell cycle Model (G_0, *M, etc*)
  - Death Models
  - Rates
  - Custom Variables
  - Molecular models



PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell : Multiscale Modelling Framework



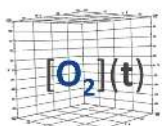
Diffusion equations

$$\frac{\partial \boldsymbol{\rho}}{\partial t} = \overbrace{\mathbf{D}\nabla^2\boldsymbol{\rho}}^{\text{diffusion}} - \overbrace{\boldsymbol{\lambda}\boldsymbol{\rho}}^{\text{decay}} + \overbrace{\mathbf{S}(\boldsymbol{\rho}^* - \boldsymbol{\rho})}^{\text{bulk source}} - \overbrace{\mathbf{U}\boldsymbol{\rho}}^{\text{bulk uptake}} + \overbrace{\sum_{\text{cells } k} \delta(\mathbf{x} - \mathbf{x}_k) W_k [\mathbf{S}_k(\boldsymbol{\rho}_k^* - \boldsymbol{\rho}) - \mathbf{U}_k \boldsymbol{\rho}]}^{\text{sources and uptake by cells}} \text{ in } \Omega$$

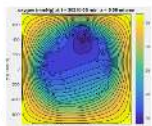
System of PDEs for each molecule:

- Diffusion term
- Decay
- Uptake/Production

PDEs are solved using BioFVM

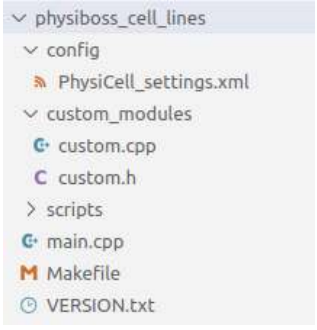






PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell : Project structure




```
while( PhysiCell_globals.current_time < PhysiCell_settings.max_time + 0.1*diffusion_dt )
{
    // save data if it's time.
    if( fabs( PhysiCell_globals.current_time - PhysiCell_globals.next_full_save_time ) < 0.01 * diffusion_dt )
    {
        display_simulation_status( std::cout );
        if( PhysiCell_settings.enable_legacy_saves == true )
        {
            log_output( PhysiCell_globals.current_time , PhysiCell_globals.full_output_index, microenvironment, report_file);
        }

        if( PhysiCell_settings.enable_full_saves == true )
        {
            sprintf( filename , "%s/output%08u" , PhysiCell_settings.folder.c_str(),  PhysiCell_globals.full_output_index );

            save_PhysiCell_to_MultiCellDS_v2( filename , microenvironment , PhysiCell_globals.current_time );

            sprintf( filename , "%s/states_%08u.csv", PhysiCell_settings.folder.c_str(), PhysiCell_globals.full_output_index);

            MaBoSSIntracellular::save( filename, *PhysiCell::all_cells );
        }

        PhysiCell_globals.full_output_index++;
        PhysiCell_globals.next_full_save_time += PhysiCell_settings.full_save_interval;
    }

    // save SVG plot if it's time
    if( fabs( PhysiCell_globals.current_time - PhysiCell_globals.next_SVG_save_time  ) < 0.01 * diffusion_dt )
    {
        if( PhysiCell_settings.enable_SVG_saves == true )
        {
            sprintf( filename , "%s/snapshot%08u.svg" , PhysiCell_settings.folder.c_str() , PhysiCell_globals.SVG_output_index );
            SVG_plot( filename , microenvironment, 0.0 , PhysiCell_globals.current_time, cell_coloring_function );

            PhysiCell_globals.SVG_output_index++;
            PhysiCell_globals.next_SVG_save_time  += PhysiCell_settings.SVG_save_interval;
        }
    }

    // update the microenvironment
    microenvironment.simulate_diffusion_decay( diffusion_dt );

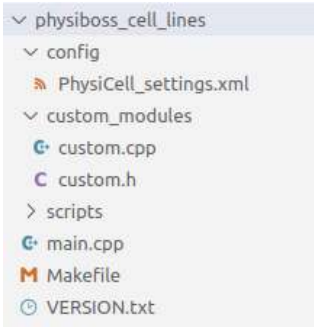
    // run PhysiCell
    ((Cell_Container *)microenvironment.agent_container)->update_all_cells( PhysiCell_globals.current_time );

    /*
      Custom add-ons could potentially go here.
    */

    PhysiCell_globals.current_time += diffusion_dt;
}
```

PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell : Project structure




```
void create_cell_types( void )
{
    /*
       Put any modifications to default cell definition here if you
       want to have "inherited" by other cell types.

       This is a good place to set default functions.
    */

    initialize_cell_definitions_from_pugixml();

    return;
}

void setup_microenvironment( void )
{
    // put any custom code to set non-homogeneous initial conditions or
    // extra Dirichlet nodes here.

    initialize_microenvironment();

    return;
}

void setup_tissue( void )
{
    // place a cluster of tumor cells at the center
    double cell_radius = cell_defaults.phenotype.geometry.radius;
    double cell_spacing = 0.95 * 2.0 * cell_radius;

    double tumor_radius = parameters.doubles( "tumor_radius" ); // 250.0;

    // Parameter<double> temp;

    int i = parameters.doubles.find_index( "tumor_radius" );

    Cell* pCell = NULL;

    std::vector<std::vector<double>> positions = create_cell_sphere_positions(cell_radius,tumor_radius);
    std::cout << "creating " << positions.size() << " closely-packed tumor cells ... " << std::endl;


    for( int i=0; i < positions.size(); i++ )
    {
        pCell = create_cell(get_cell_definition(
            (PhysiCell::UniformRandom()*100) > parameters.doubles("percentage_mutants") ? "default":"mutant")
        );

        pCell->assign_position( positions[i] );
    }

    return;
}
```

PhysiBoSS : Multi-scale, agent based modeling framework

- PhysiCell : Model definition
  - Reducing the dependency on writing C++ code
  - XML definition to encode for basic properties
  - Graphical interfaces being developed for setting properties, simulation
  - No way to easily use boolean models



```
<cell_definitions>
    <cell_definition name="default" ID="0">
        <phenotype>
            <cycle code="0" name="advanced_Ki67_cycle_model">
                <!-- using higher than normal significant digits to match divisions in default code -->
                <phase_transition_rates units="1/min">
                    <!-- 0 to + -->
                    <rate start_index="0" end_index="1" fixed_duration="true">1e30</rate>
                    <!-- + (pre-mitotic) to + (post-mitotic) -->
                    <rate start_index="1" end_index="2" fixed_duration="true">1e30</rate>
                    <!-- + (post-mitotic) to 0 -->
                    <rate start_index="2" end_index="0" fixed_duration="true">0.006666667</rate>
                </phase_transition_rates>
            </cycle>

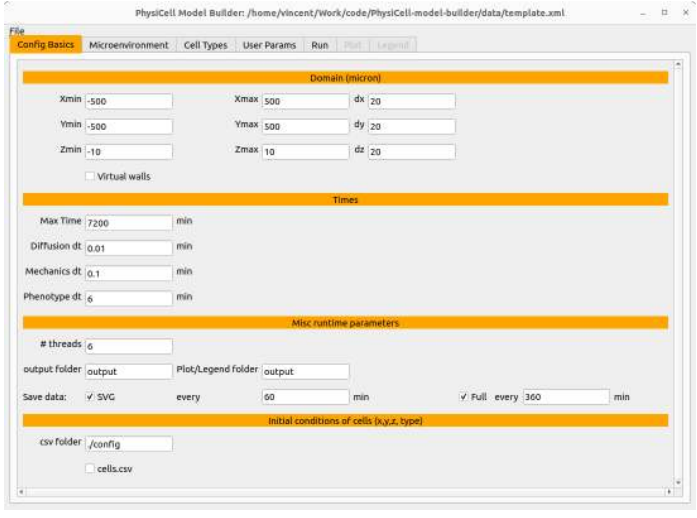
            <death>
                <model code="100" name="apoptosis">
                    <death_rate units="1/min">0.0</death_rate>
                    <phase_transition_rates units="1/min">
                        <rate start_index="0" end_index="1" fixed_duration="true">1e30</rate>
                    </phase_transition_rates>
                </model>

                <model code="101" name="necrosis">
                    <death_rate units="1/min">0.0</death_rate>
                    <phase_transition_rates units="1/min">
                        <rate start_index="0" end_index="1" fixed_duration="false">9e9</rate>
                        <rate start_index="1" end_index="2" fixed_duration="true">1.15741e-5</rate>
                    </phase_transition_rates>
                </model>
            </death>

            <motility>
                <speed units="micron/min">1</speed>
                <persistence_time units="min">1</persistence_time>
```

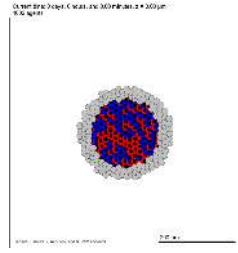
PhysiBoSS : Multi-scale, agent based modeling framework

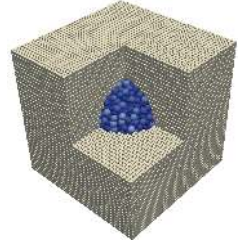
› PhysiCell : Model builder



PhysiBoSS : Multi-scale, agent based modeling framework

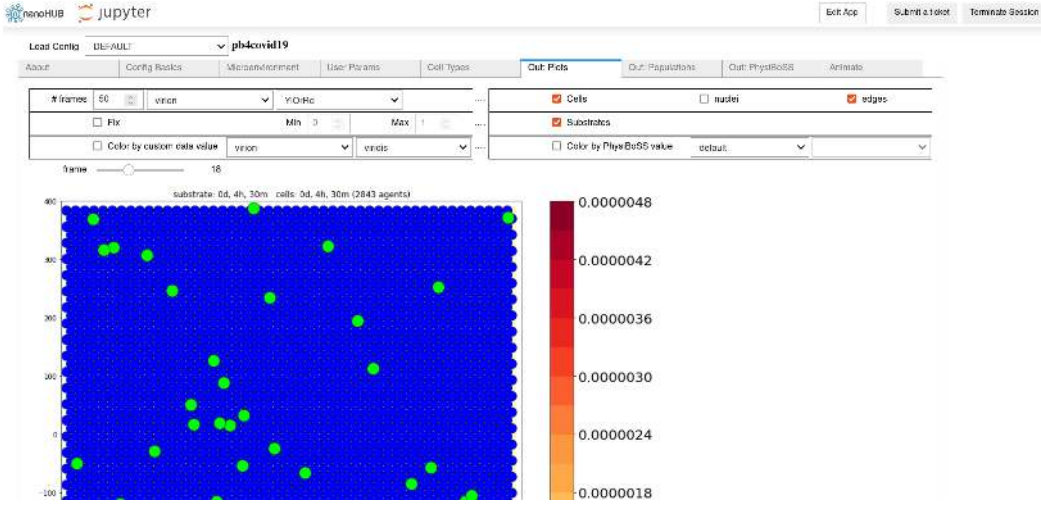
- PhysiCell : Output
  - MAT files with raw data for each cell
  - SVG files of 2D cut
  - Raw data can be used with Paraview, or programmatically





PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell : Jupyter notebooks



PhysiBoSS : Multi-scale, agent based modeling framework

PhysiBoSS : Multi-scale, agent based modeling framework

PhysiBoSS: a multi-scale agent-based modelling framework integrating physical dimension and cell signalling

Gaelle Letort, Arnau Montagud, Gautier Stoll, Randy Heiland, Emmanuel Barillot, Paul Macklin, Andrei Zinovyev, Laurence Calzone

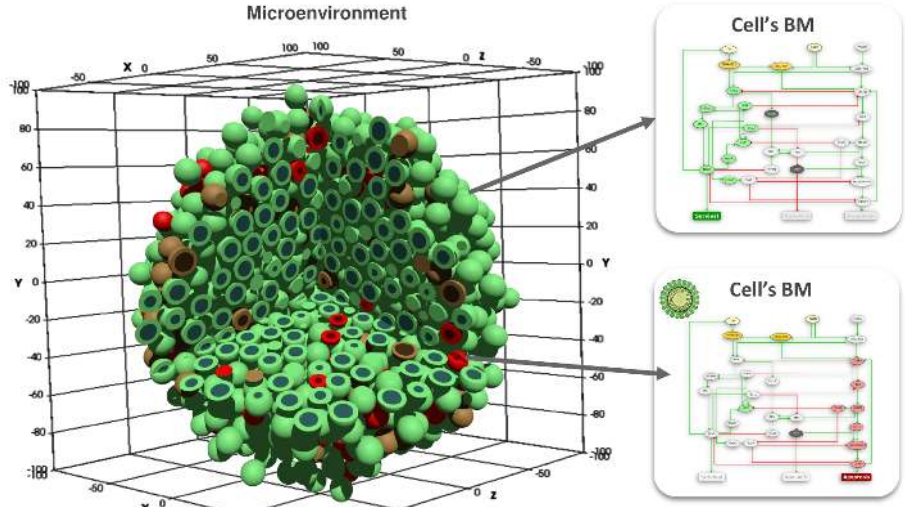
Bioinformatics, Volume 35, Issue 7, 01 April 2019, Pages 1188–1196, https://doi.org/10.1093/bioinformatics/bty766

Published: 30 August 2018 **Article history** ▾



Gaëlle Letort

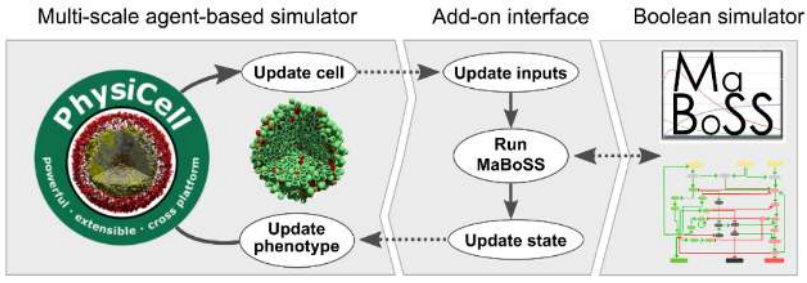
PhysiBoSS : Multi-scale, agent based modeling framework



Ponce de Leon et al, 2022

PhysiBoSS : Multi-scale, agent based modeling framework

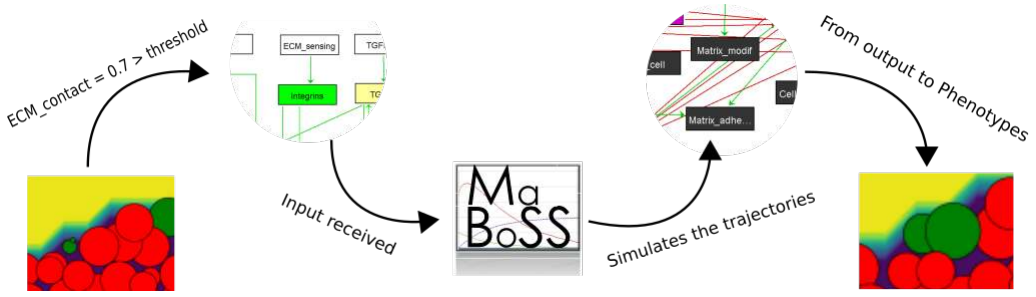
- PhysiBoSS started from a fork of PhysiCell 1.1.0 (Nov 2017)
- First meetings (Curie, BSC, IU) to merge PhysiBoSS into PhysiCell (Feb 2019)
- PhysiBoSS 2.0 was merged into PhysiCell 1.9.0 (July 2021) as an add-on



Ponce de Leon et al, 2022

- PhysiBoSS 2.2.0 release in september (stochastic time step, automatic mapping)

PhysiBoSS : Multi-scale, agent based modeling framework



- We take signals as inputs, and modify behaviors according to outputs
- Each cell runs one single (stochastic) simulation every *intracellular_dt*

PhysiBoSS : Multi-scale, agent based modeling framework

- Model definition (MaBoSS BND and CFG files)
- Intracellular dt
- Scaling, time stochasticity, ...
- Initial values
- Parameter values
- Mutants
- Mapping

```
<phenotype>

    ...

    <intracellular type="maboss">
        <bnd_filename>./data/model_0.bnd</bnd_filename>
        <cfg_filename>./data/model.cfg</cfg_filename>
        <settings>
            <intracellular_dt>1</intracellular_dt>
            <time_stochasticity>0.1</time_stochasticity>
            <scaling>1.0</scaling>
            <parameters>
                <parameter intracellular_name="$time_scale">0.0</parameter>
            </parameters>
        </settings>
        <initial_values>
            <initial_value intracellular_name="A">1</initial_value>
            <initial_value intracellular_name="C">0</initial_value>
        </initial_values>
        <mapping>
            <input physicell_name="oxygen" intracellular_name="A">
                <settings>
                    <action>inhibition</action>
                    <threshold>10</threshold>
                    <inact_threshold>10</inact_threshold>
                </settings>
            </input>
            <output physicell_name="apoptosis" intracellular_name="C">
                <settings>
                    <action>activation</action>
                    <value>5</value>
                    <base_value>0</base_value>
                    <smoothing>5</smoothing>
                </settings>
            </output>
        </mapping>
    </intracellular>

    ...

</phenotype>
```

PhysiBoSS : Multi-scale, agent based modeling framework

Inputs : Signals to MaBoSS nodes

- Oxygen inhibits the node A
- The node A will be
  - inactive when oxygen > 10
  - active when oxygen < 10

```
<input physicell_name="oxygen" intracellular_name="A">
    <settings>
        <action>inhibition</action>
        <threshold>10</threshold>
        <inact_threshold>10</inact_threshold>
    </settings>
</input>
```

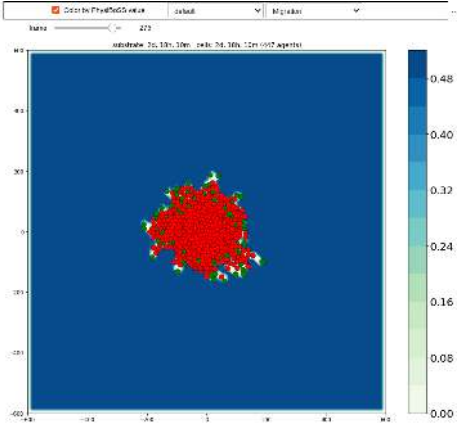
Outputs : MaBoSS nodes to Behaviours

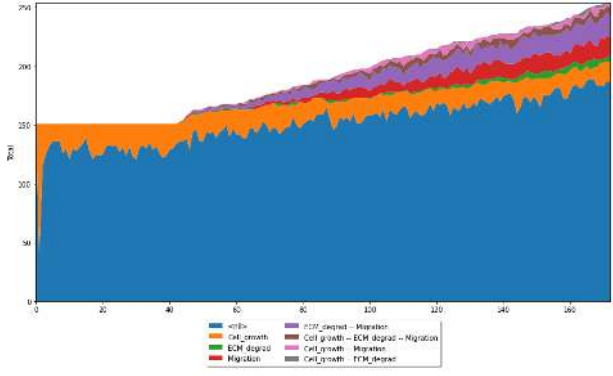
- node C activates Apoptosis
- The apoptosis rate will be set to :
  - 5 if the node C is active
  - 0 if the node C is inactive
- The apoptosis rate will take 5 dt to reach 5

```
<output physicell_name="apoptosis" intracellular_name="C">
    <settings>
        <action>activation</action>
        <value>5</value>
        <base_value>0</base_value>
        <smoothing>5</smoothing>
    </settings>
</output>
```

PhysiBoSS : Multi-scale, agent based modeling framework

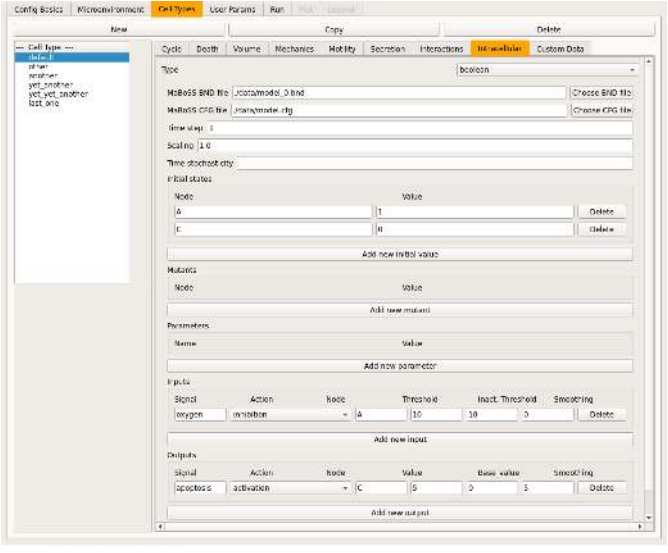
› Jupyter-GUI





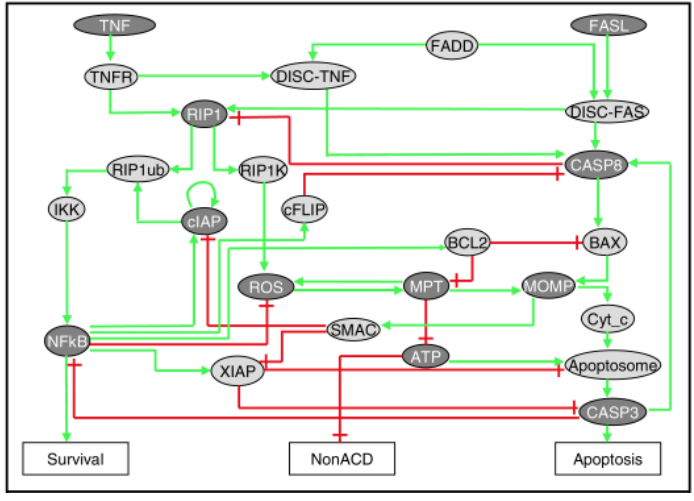
PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell-model-builder



PhysiBoSS : Multi-scale, agent based modeling framework

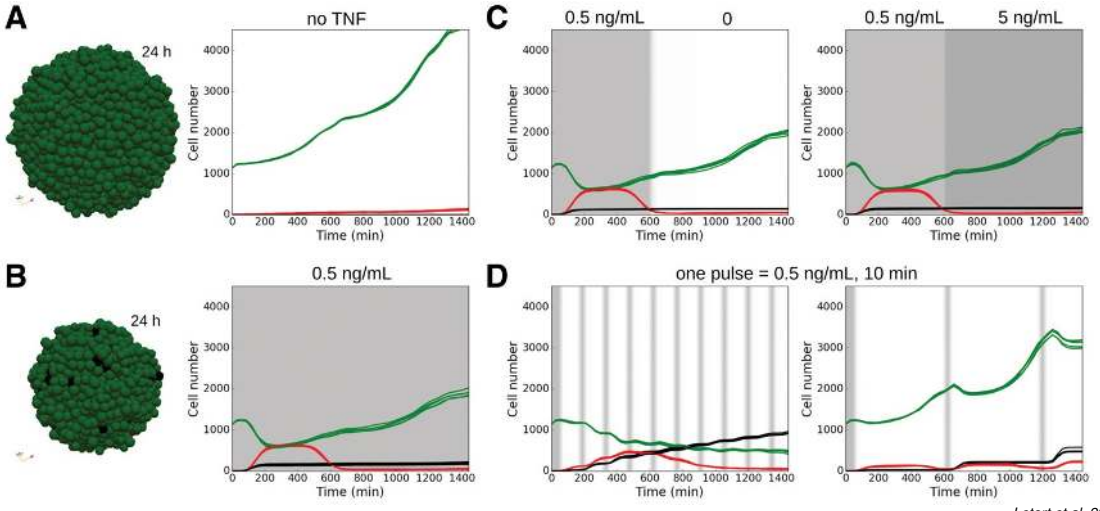
› Cell fate model



Calzone et al, 2010

PhysiBoSS : Multi-scale, agent based modeling framework

› Cell fate model



Letort et al, 2019

PhysiBoSS : Multi-scale, agent based modeling framework

- Cell fate model

Goals:

- Develop a simple model of TNF treatment
- Using PhysiBoSS 2.x
- Using PhysiCell's signals and behaviors

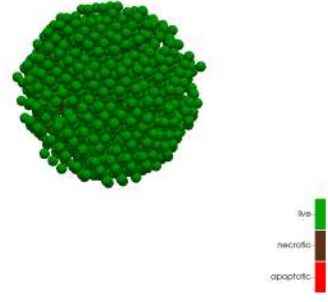
PhysiBoSS : Multi-scale, agent based modeling framework

- PhysiCell model of spheroid

  - Simple model of tumor spheroid

  - Exponential growth

  - template, spheroid 3D, physiboss-cell-lines



PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell model of spheroid

From template

› Latest version of main.cpp and custom.cpp

› Building Signals and Behaviors dictionnaries

custom.cpp

```
void create_cell_types( void )
{
    // set the random seed
    SeedRandom( parameters.ints("random_seed") );

    ...


    /*
       This intializes cell signal and response dictionaries
    */

    setup_signal_behavior_dictionaries();

    ...

    /*
       This builds the map of cell definitions and summarizes the setup.
    */

    display_cell_definitions( std::cout );

    return;
}
```

PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell model of spheroid

From spheroid 3D

› Cell definition

› Create sphere

› Controlled by user param

custom.cpp

```
void setup_tissue( void )
{
    // place a cluster of tumor cells at the center
    double cell_radius = cell_defaults.phenotype.geometry.radius;
    double cell_spacing = 0.95 * 2.0 * cell_radius;

    double tumor_radius = parameters.doubles( "tumor_radius" ); // 250.0;

    // Parameter<double> temp;

    int i = parameters.doubles.find_index( "tumor_radius" );

    Cell* pCell = NULL;

    std::vector<std::vector<double>> positions = create_cell_sphere_positions(cell_radius,tumor_radius);
    std::cout << "creating " << positions.size() << " closely-packed tumor cells ... " << std::endl;


    for( int i=0; i < positions.size(); i++ )
    {
        pCell = create_cell(); // tumor cell
        pCell->assign_position( positions[i] );
    }

    return;
}

std::vector<std::vector<double>> create_cell_sphere_positions(double cell_radius, double sphere_radius)
{
    std::vector<std::vector<double>> cells;
    int xc=0,yc=0,zc=0;
    double x_spacing= cell_radius*sqrt(3);
    double y_spacing= cell_radius*2;
    double z_spacing= cell_radius*sqrt(3);

    std::vector<double> tempPoint(3,0.0);
    // std::vector<double> cylindar_center(3,0.0);

    for(double z=-sphere_radius;z<sphere_radius;z+=z_spacing, zc++)
    {
        for(double x=-sphere_radius;x<sphere_radius;x+=x_spacing, xc++)
        {
            for(double y=-sphere_radius;y<sphere_radius;y+=y_spacing, yc++)
            {
                tempPoint[0]=x + (zc%2) * 0.5 * cell_radius;
                tempPoint[1]=y + (xc%2) * cell_radius;
                tempPoint[2]=z;

                if(sqrt(norm_squared(tempPoint))< sphere_radius)
                { cells.push_back(tempPoint); }
            }

        }
    }
    return cells;
}
```

PhysiBoSS : Multi-scale, agent based modeling framework

› PhysiCell model of spheroid

From PhysiBoSS cell lines

› Makefile

› Export of intracellular data

Makefile

```
### MaBoSS configuration
# MaBoSS max nodes
ifndef MABOSS_MAX_NODES
MABOSS_MAX_NODES = 64
endif

# MaBoSS directory
MABOSS_DIR = addons/PhysiBoSS/MaBoSS-env-2.0/engine
CUR_DIR = $(shell pwd)
CUSTOM_DIR = sample_projects/Arnau_model/custom_modules

ifneq ($(OS), Windows_NT)
    LDL_FLAG = -ldl
endif

LIB := -L$(CUR_DIR)/$(MABOSS_DIR)/lib -lMaBoSS-static $(LDL_FLAG)
INC := -DADDON_PHYSIBOSS -I$(CUR_DIR)/$(MABOSS_DIR)/include -DMAXNODES=$(MABOSS_MAX_NODES)


# If nax nodes > 64, change lib path
ifeq ($(shell expr $(MABOSS_MAX_NODES) '>' 64), 1)
LIB := -L$(CUR_DIR)/$(MABOSS_DIR)/lib -lMaBoSS_$(MABOSS_MAX_NODES)n-static $(LDL_FLAG)
endif
```

main.cpp

```
char filename[1024];
sprintf( filename , "%s/initial" , PhysiCell_settings.folder.c_str() );
save_PhysiCell_to_MultiCellDS_v2( filename , microenvironment , PhysiCell_globals.current_time );

sprintf( filename , "%s/states_initial.csv", PhysiCell_settings.folder.c_str());
MaBoSSIntracellular::save( filename, *PhysiCell::all_cells);
```

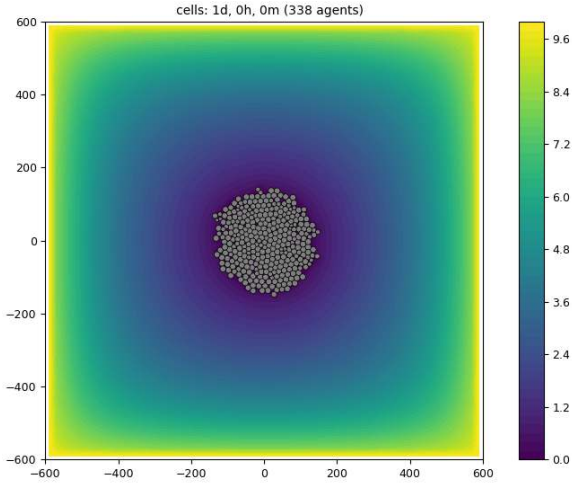
PhysiBoSS : Multi-scale, agent based modeling framework

› Adding TNF to the microenvironment

```
<microenvironment_setup>
  <variable ID="0" name="TNF" units="dimensionless">
    <physical_parameter_set>
      <diffusion_coefficient units="micron^2/min">1200.0</diffusion_coefficient>
      <decay_rate units="1/min">.0275</decay_rate>
    </physical_parameter_set>
    <initial_condition units="mmHg">0.0</initial_condition>
    <Dirichlet_boundary_condition enabled="True" units="mmHg">10.0</Dirichlet_boundary_condition>
    <Dirichlet_options>
      <boundary_value ID="xmin" enabled="True">10.0</boundary_value>
      <boundary_value ID="xmax" enabled="True">10.0</boundary_value>
      <boundary_value ID="ymin" enabled="True">10.0</boundary_value>
      <boundary_value ID="ymax" enabled="True">10.0</boundary_value>
      <boundary_value ID="zmin" enabled="True">10.0</boundary_value>
      <boundary_value ID="zmax" enabled="True">10.0</boundary_value>
    </Dirichlet_options>
  </variable>
```

PhysiBoSS : Multi-scale, agent based modeling framework

› Adding TNF to the microenvironment



PhysiBoSS : Multi-scale, agent based modeling framework

› Adding TNF to the microenvironment

main.cpp

```
int tnf_index = microenvironment.find_density_index("TNF");

if (PhysiCell_globals.current_time < 1000 && microenvironment.get_substrate_dirichlet_activation(tnf_index))
{
    microenvironment.set_substrate_dirichlet_activation(tnf_index, false);
}

if (parameters.bools("tnf_treatment")){
    if (PhysiCell_globals.current_time >= 1000 && PhysiCell_globals.current_time < 5320 && !microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF activation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, true);
    }
}
```

⇒ Activating TNF only after 1000min

PhysiBoSS : Multi-scale, agent based modeling framework

- Adding the Boolean model

- Boolean model
- TNF concentration as input for TNF node
- Apoptosis node triggers apoptosis
- NonACD node triggers necrosis

```
<intracellular type="maboss">
  <bnd_filename>config/boolean_network/CellFateModel.bnd</bnd_filename>
  <cfg_filename>config/boolean_network/CellFateModel.cfg</cfg_filename>
  <settings>
    <intracellular_dt>1440.0</intracellular_dt>
    <time_stochasticity>0.0</time_stochasticity>
    <scaling>60.0</scaling>
  </settings>
  <mapping>
    <input intracellular_name="TNF" physicell_name="TNF">
      <settings>
        <action>activation</action>
        <threshold>0.01</threshold>
        <smoothing>0</smoothing>
      </settings>
    </input>
    <output intracellular_name="Apoptosis" physicell_name="apoptosis">
      <settings>
        <action>activation</action>
        <value>1000000</value>
        <base_value>0</base_value>
        <smoothing>0</smoothing>
      </settings>
    </output>
    <output intracellular_name="NonACD" physicell_name="necrosis">
      <settings>
        <action>activation</action>
        <value>1000000</value>
        <base_value>0</base_value>
        <smoothing>0</smoothing>
      </settings>
    </output>
  </mapping>
</intracellular>
```

PhysiBoSS : Multi-scale, agent based modeling framework

- Choosing intracellular_dt

Short intracellular dt

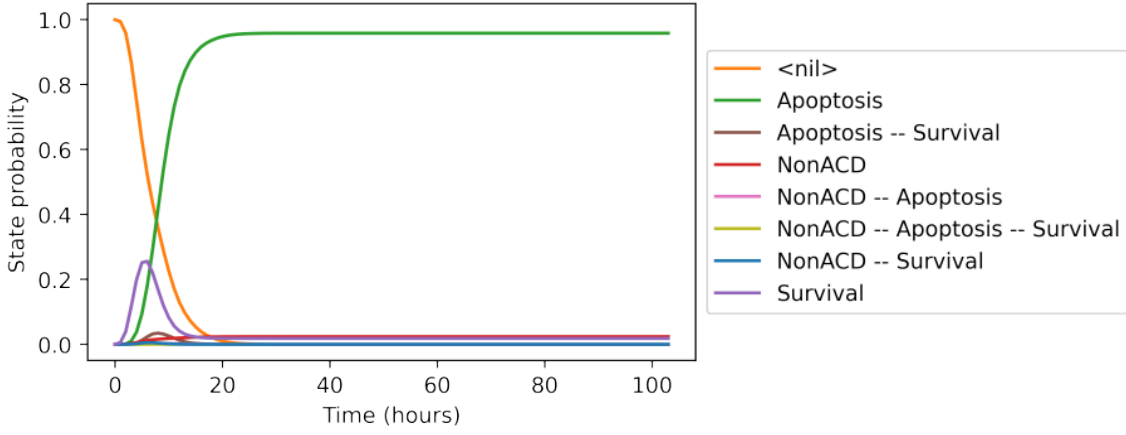
- Integrate transient activation of some nodes
- Needs a well curated model

Long intracellular dt

- Wait for steady state of the model
- Doesn’t integrate possible issues with transient effects

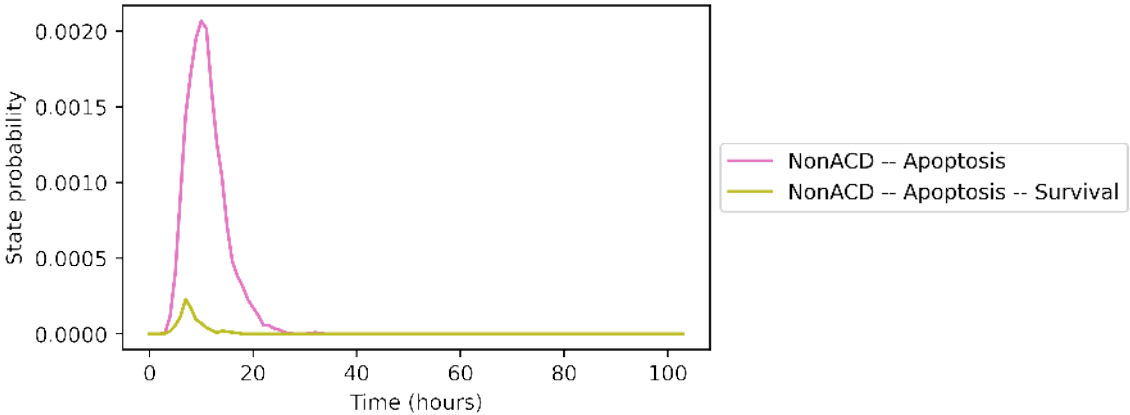
PhysiBoSS : Multi-scale, agent based modeling framework

› Choosing intracellular_dt



PhysiBoSS : Multi-scale, agent based modeling framework

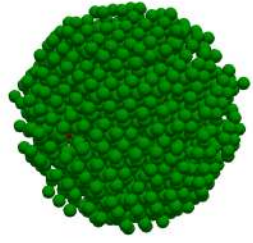
› Choosing intracellular_dt



⇒ To avoid NonACD – Apoptosis, we choose 24h

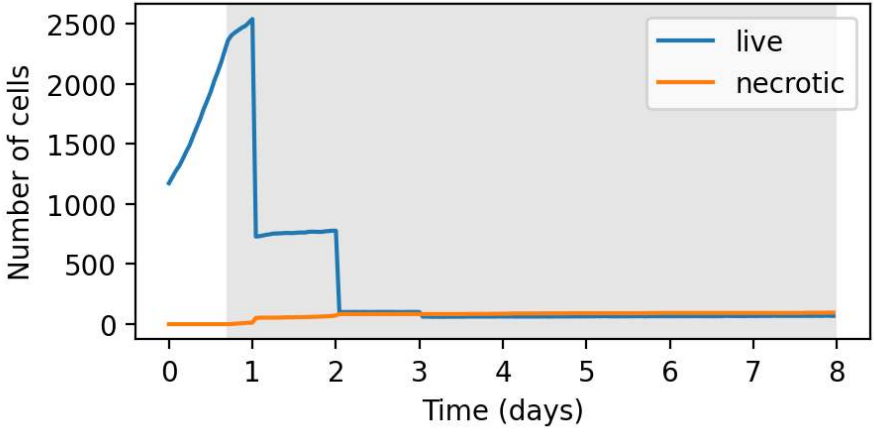
PhysiBoSS : Multi-scale, agent based modeling framework

› Results with constant TNF



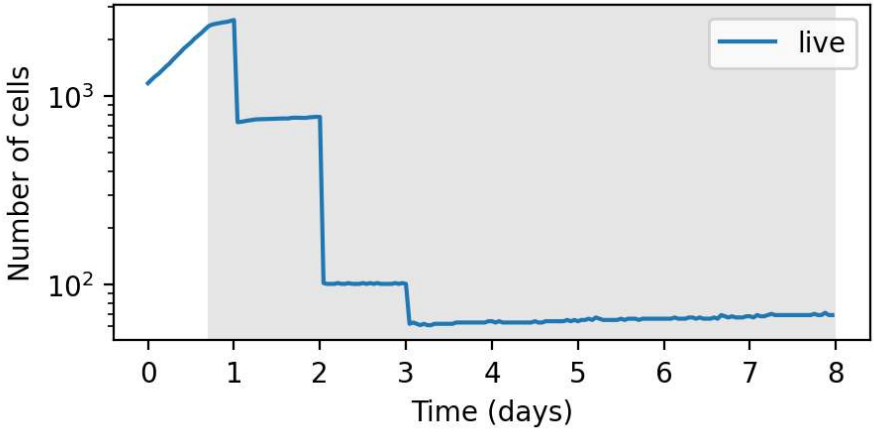
PhysiBoSS : Multi-scale, agent based modeling framework

› Results with constant TNF



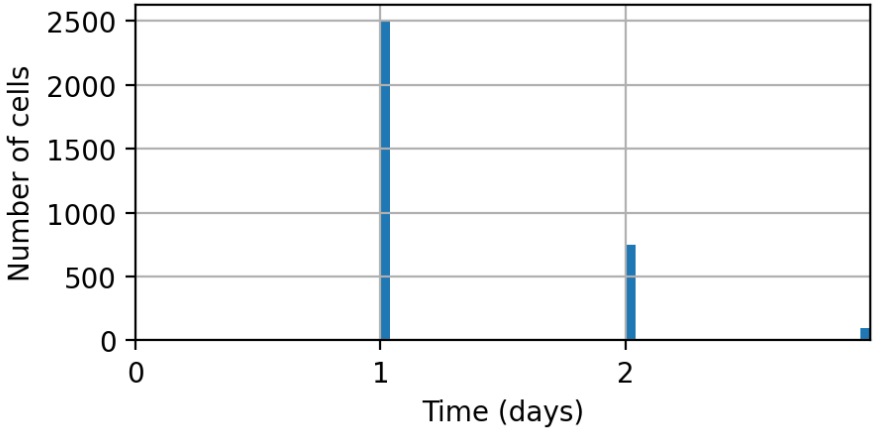
PhysiBoSS : Multi-scale, agent based modeling framework

› Results with constant TNF



PhysiBoSS : Multi-scale, agent based modeling framework

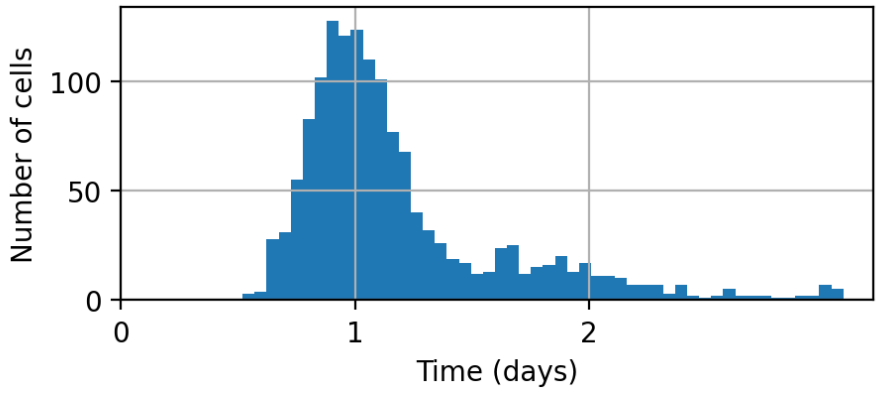
- Time stochasticity



- time_stochasticity = 0.0

PhysiBoSS : Multi-scale, agent based modeling framework

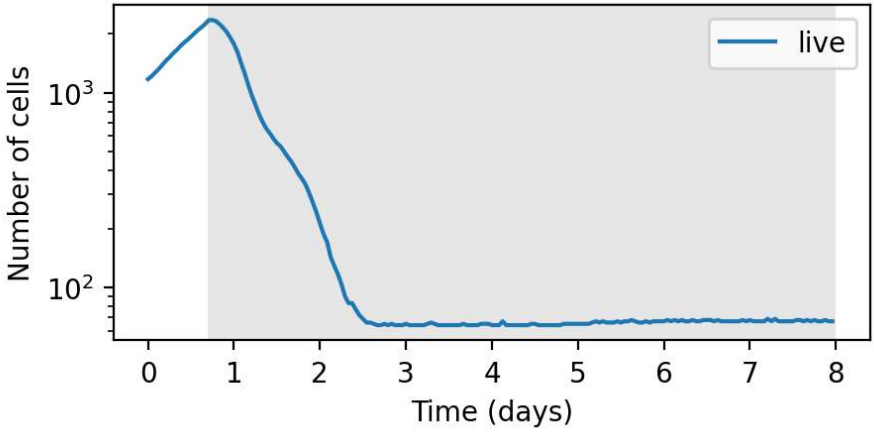
- Time stochasticity



- time_stochasticity = 0.2

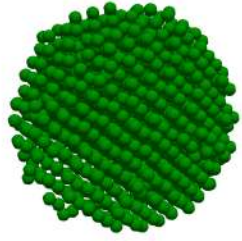
PhysiBoSS : Multi-scale, agent based modeling framework

› Time stochasticity



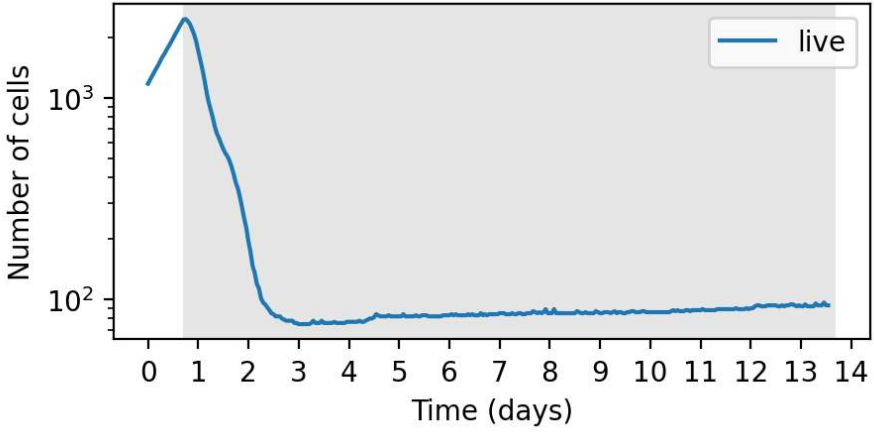
PhysiBoSS : Multi-scale, agent based modeling framework

› Time stochasticity



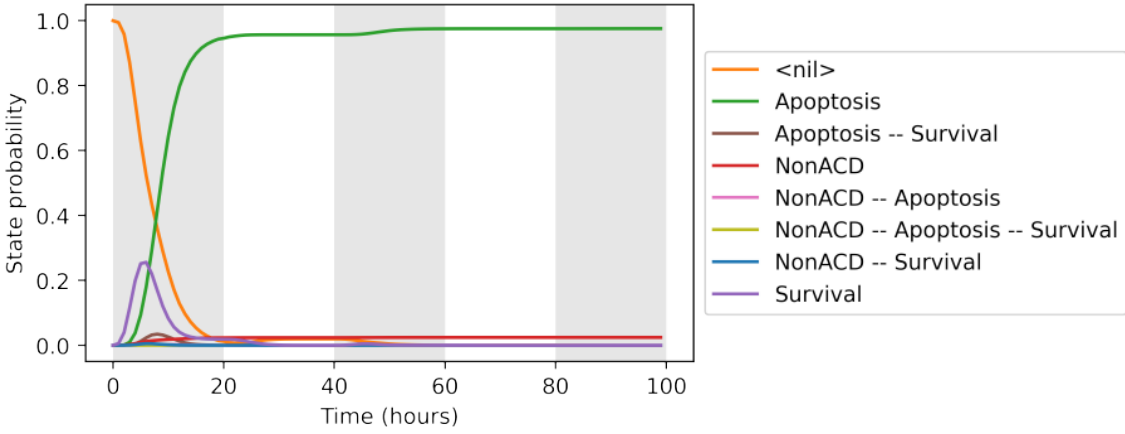
PhysiBoSS : Multi-scale, agent based modeling framework

› Creating pulses of TNF



PhysiBoSS : Multi-scale, agent based modeling framework

› Creating pulses of TNF



PhysiBoSS : Multi-scale, agent based modeling framework

› Creating pulses of TNF

```
if (parameters.bools("pulse_tnf")){

    if (PhysiCell_globals.current_time >= 5320 && PhysiCell_globals.current_time < 8200 && microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF in activation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, false);
    }

    if (PhysiCell_globals.current_time >= 8200 && PhysiCell_globals.current_time < 12520 && !microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF activation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, true);
    }

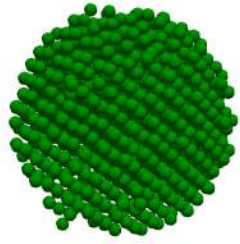
    if (PhysiCell_globals.current_time >= 12520 && PhysiCell_globals.current_time < 15400 && microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF inactivation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, false);
    }

    if (PhysiCell_globals.current_time >= 15400 && PhysiCell_globals.current_time < 19720 && !microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF activation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, true);
    }

    if (PhysiCell_globals.current_time >= 19720 && microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF inactivation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, false);
    }
}
```

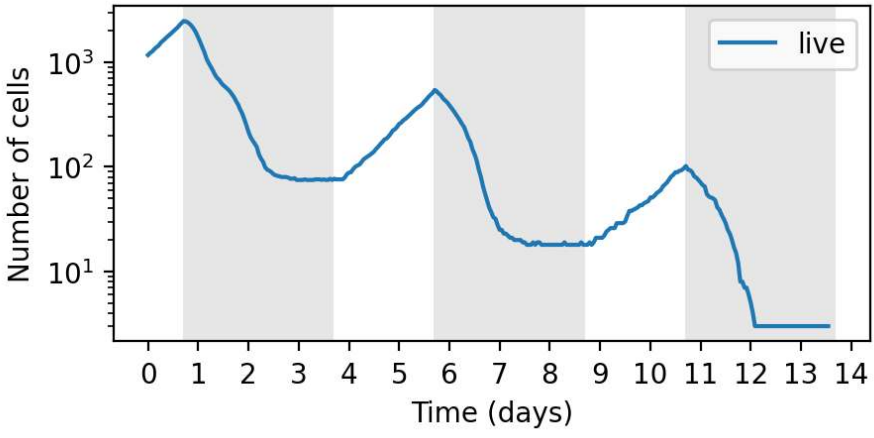
PhysiBoSS : Multi-scale, agent based modeling framework

› Creating pulses of TNF



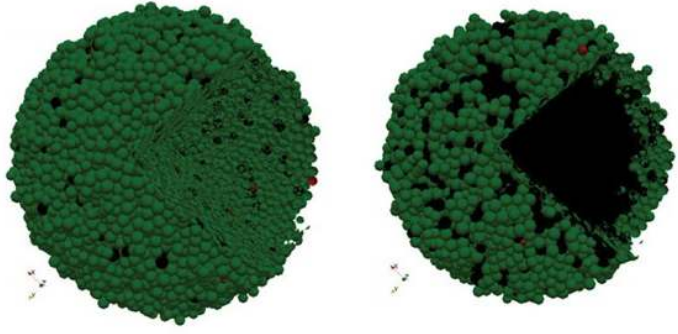
PhysiBoSS : Multi-scale, agent based modeling framework

› Creating pulses of TNF



PhysiBoSS : Multi-scale, agent based modeling framework

› Adding tumor necrotic core



Letort et al., 2019

PhysiBoSS : Multi-scale, agent based modeling framework

› Adding tumor necrotic core

BND file

```
node OXYGEN
{
  rate_up = 0.0;
  rate_down = 0.0;
}

node NonACD
{
  logic = !OXYGEN | !ATP;
  rate_up = @logic ? 1.0 : 0.0;
  rate_down = @logic ? 0.0 : 1.0;
}
node Apoptosis
{
  logic = OXYGEN & CASP3;
  rate_up = @logic ? 1.0 : 0.0;
  rate_down = @logic ? 0.0 : 1.0;
}
node Survival
{
  logic = OXYGEN & NFkB;
  rate_up = @logic ? 1.0 : 0.0;
  rate_down = @logic ? 0.0 : 1.0;
```

CFG file

```
Cyt_c.istate = FALSE ;
XIAP.istate = FALSE ;
apoptosome.istate = FALSE ;
CASP3.istate = FALSE ;
cFLIP.istate = FALSE ;
OXYGEN.istate = TRUE;
NonACD.istate = FALSE ;
Apoptosis.istate = FALSE ;
Survival.istate = FALSE ;
```

PhysiBoSS : Multi-scale, agent based modeling framework

› Adding tumor necrotic core

```
<variable name="oxygen" units="mmHg" ID="1">
  <physical_parameter_set>
    <diffusion_coefficient units="micron^2/min">1000.00</diffusion_coefficient>
    <decay_rate units="1/min">.00001</decay_rate>
  </physical_parameter_set>
  <initial_condition units="mmHg">10.0</initial_condition>
  <Dirichlet_boundary_condition units="mmHg" enabled="true">38.0</Dirichlet_boundary_condition>
  <Dirichlet_options>
      <boundary_value ID="xmin" enabled="True">38.0</boundary_value>
      <boundary_value ID="xmax" enabled="True">38.0</boundary_value>
      <boundary_value ID="ymin" enabled="True">38.0</boundary_value>
      <boundary_value ID="ymax" enabled="True">38.0</boundary_value>
      <boundary_value ID="zmin" enabled="True">38.0</boundary_value>
      <boundary_value ID="zmax" enabled="True">38.0</boundary_value>
    </Dirichlet_options>
</variable>
```

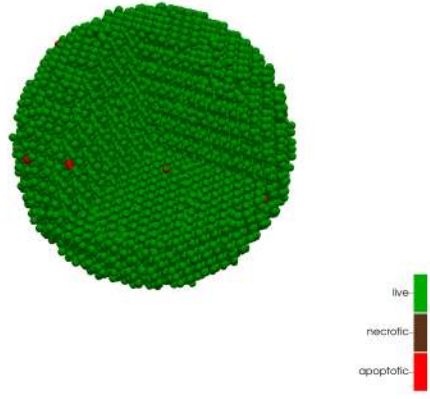
PhysiBoSS : Multi-scale, agent based modeling framework

› Adding tumor necrotic core

```
<mapping>
  <input intracellular_name="TNF" physicell_name="TNF">
    <settings>
      <action>activation</action>
      <threshold>0.01</threshold>
      <smoothing>0</smoothing>
    </settings>
  </input>
  <input intracellular_name="OXYGEN" physicell_name="oxygen">
    <settings>
      <action>activation</action>
      <threshold>0.05</threshold>
      <smoothing>0</smoothing>
    </settings>
  </input>
```

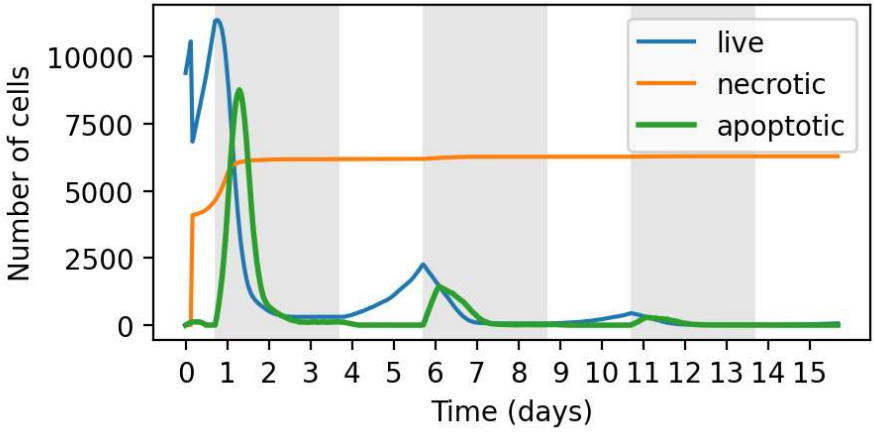
PhysiBoSS : Multi-scale, agent based modeling framework

› Adding tumor necrotic core



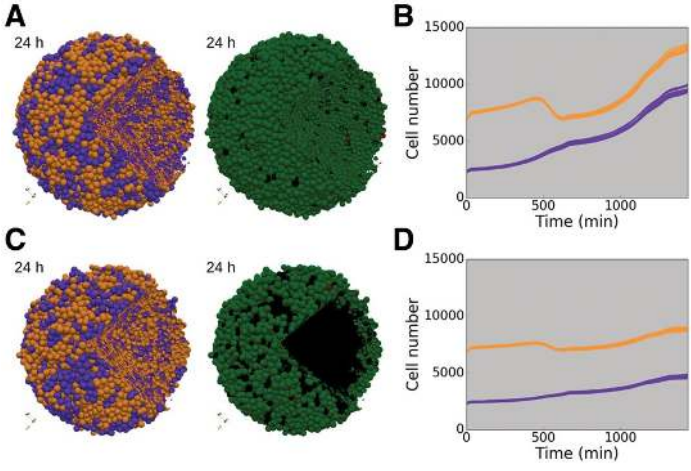
PhysiBoSS : Multi-scale, agent based modeling framework

› Adding tumor necrotic core



PhysiBoSS : Multi-scale, agent based modeling framework

› Add IKK/cFLIP mutant population



Letort et al., 2019

PhysiBoSS : Multi-scale, agent based modeling framework

- Add IKK/cFLIP mutant population
- New cell definition
- Copy from default
- Add mutations to the intracellular model

```
<intracellular type="maboss">
  <bnd_filename>config/boolean_network/CellFateModel_with_oxygen.bnd</bnd_filename>
  <cfg_filename>config/boolean_network/CellFateModel_with_oxygen.cfg</cfg_filename>
  <settings>
    <intracellular_dt>1440.0</intracellular_dt>
    <time_stochasticity>0.2</time_stochasticity>
    <scaling>60.0</scaling>
    <mutations>
      <mutation intracellular_name="IKK">1.0</mutation>
      <mutation intracellular_name="cFLIP">1.0</mutation>
    </mutations>
  </settings>
  <mapping>
    <input intracellular_name="TNF" physicell_name="TNF">
      <settings>
        <action>activation</action>
        <threshold>0.01</threshold>
        <smoothing>0</smoothing>
      </settings>
    </input>
    <input intracellular_name="OXYGEN" physicell_name="oxygen">
      <settings>
        <action>activation</action>
        <threshold>0.05</threshold>
        <smoothing>0</smoothing>
      </settings>
    </input>
    <output intracellular_name="Apoptosis" physicell_name="apoptosis">
      <settings>
        <action>activation</action>
        <value>1000000</value>
        <base_value>0</base_value>
        <smoothing>0</smoothing>
      </settings>
    </output>
    <output intracellular_name="NonACD" physicell_name="necrosis">
      <settings>
        <action>activation</action>
        <value>1000000</value>
        <base_value>0</base_value>
        <smoothing>0</smoothing>
      </settings>
    </output>
  </mapping>
```

PhysiBoSS : Multi-scale, agent based modeling framework

› Add IKK/cFLIP mutant population

```
void setup_tissue( void )
{
    // place a cluster of tumor cells at the center
    double cell_radius = cell_defaults.phenotype.geometry.radius;
    double cell_spacing = 0.95 * 2.0 * cell_radius;

    double tumor_radius = parameters.doubles( "tumor_radius" ); // 250.0;

    // Parameter<double> temp;

    int i = parameters.doubles.find_index( "tumor_radius" );

    Cell* pCell = NULL;

    std::vector<std::vector<double>> positions = create_cell_sphere_positions(cell_radius,tumor_radius);
    std::cout << "creating " << positions.size() << " closely-packed tumor cells ... " << std::endl;

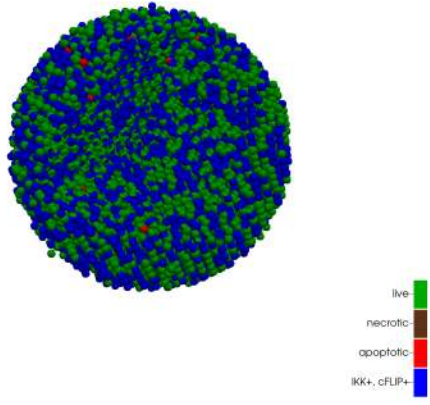

    for( int i=0; i < positions.size(); i++ )
    {
        pCell = create_cell(get_cell_definition(
            (PhysiCell::UniformRandom()*100) > parameters.doubles("percentage_mutants") ? "default":"mutant")
        );

        pCell->assign_position( positions[i] );
    }

    return;
}
```

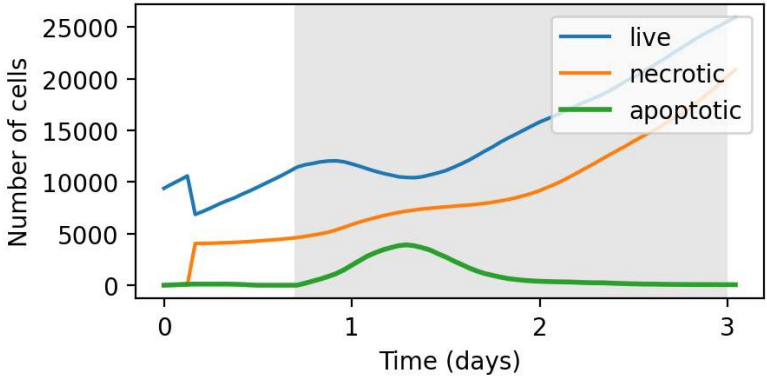
PhysiBoSS : Multi-scale, agent based modeling framework

› Add IKK/cFLIP mutant population



PhysiBoSS : Multi-scale, agent based modeling framework

› Add IKK/cFLIP mutant population



Conclusion

- A small example showing what PhysiBoSS can do
- Parameterization is not easy
- New bugfixes and things learned while preparing it
- This model will be included into PhysiCell's next version !

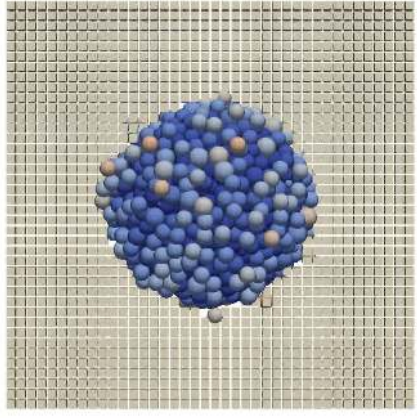
https://github.com/PhysiBoSS/PhysiBoSS
development branch, sample project **simple-spheroid-tnf**

Multiscale model of the different modes of invasion

Conclusion

- Multiscale model of the different modes of invasion
  - Starts from a spheroid tumor
  - Surrounding cells start evading
  - Different migration modes :
    - Single
    - Collective
    - Trail

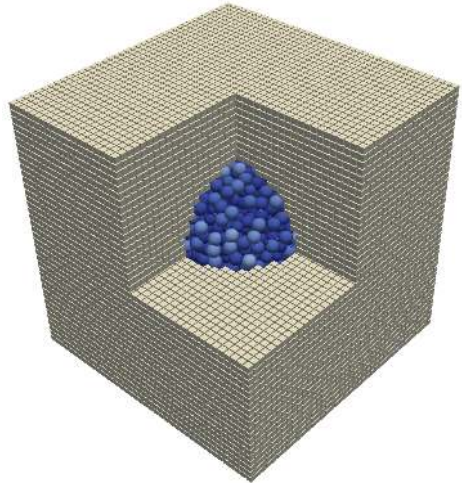




M. Ruscone et al., in preparation

Conclusion

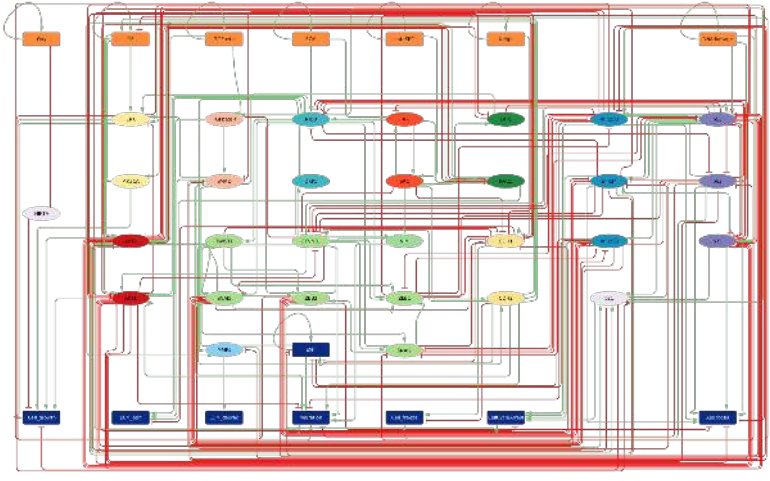
- Multiscale model of the different modes of invasion
  - A substrate of the microenvironment
  - A barrier for the cells
  - Cell-ECM contact triggers signalling
  - Cells produce metaloprotease to destroy ECM



M. Ruscone et al., in preparation

Conclusion

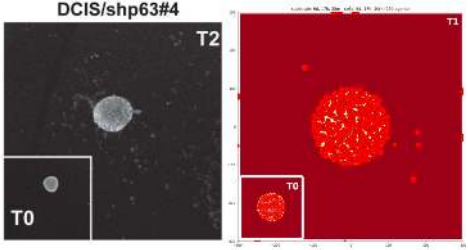
- Multiscale model of the different modes of invasion
  - 7 input nodes and 7 output nodes



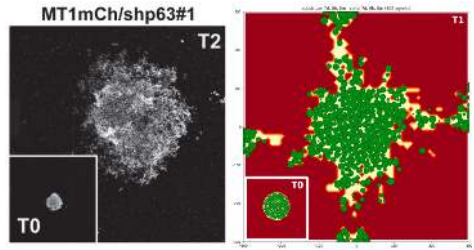
M. Ruscone et al., in preparation

Conclusion

- Multiscale model of the different modes of invasion
  - Comparison with Figure 6.a from *"MT1-MMP and P63 in DCIS to IDC transition" C. Lodillinsky et al*



Red cells have p63 node OFF

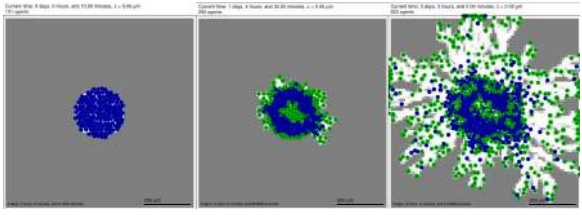


Green cells have p63 node ON

M. Ruscone et al., in preparation

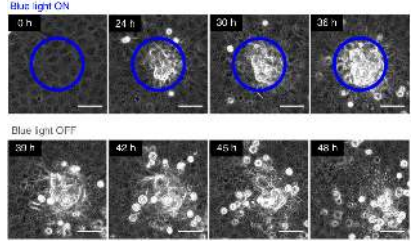
Conclusion

- Multiscale model of the different modes of invasion
  - up/down-regulation induced by light
  - reproduce collective extrusion of cells



Local light-activation of the Src oncoprotein in an epithelial monolayer promotes collective extrusion

Sarah Moitrier, Nastassia Pricoupenko, Adèle Kerjouan, Christiane Oddou, Olivier Destaing, Aude Battistella, Pascal Silberzan & Isabelle Bonnet



M. Ruscone et al., in preparation

Conclusion

- Multiscale model of the different modes of invasion
  - Performing sensitivity analysis
  - Improvement of ECM representation
  - Work on PhysiBoSS model fitting


Marco Ruscone


Gaelle Letort


Arnau Montagud


Laurence Calzone


Andrei Zinovyev


Philippe Chavrier


Olivier Destaing

institutCurie

Barcelona Supercomputing Center

INSTITUT PASTEUR

IAB

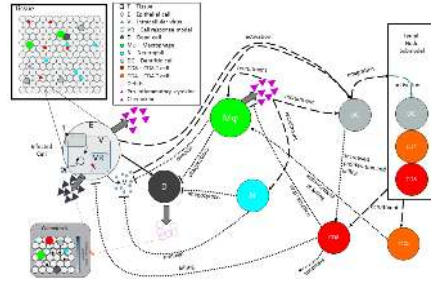
PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2

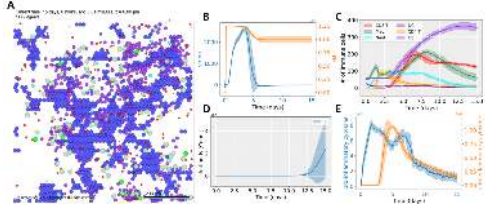
Conclusion

- PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2
  - Build a multi-scale model of COVID infection
  - Using PhysiCell formalism + PhysiBoSS addon
  - Starting with existing model from PhysiCell creators
  - With mechanistic Boolean models produced by the COVID-19 Disease Maps community

Conclusion

- PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2
  - PhysiCell model representing SARS-CoV-2 infection
  - Developed by a consortium led by Paul Macklin's lab
  - Represents Virus, Epithelial cells, CD4, CD8, Macrophages, Dendritic cells, Neutrophils
  - Good phenomenological model
  - Lacks mechanistic details

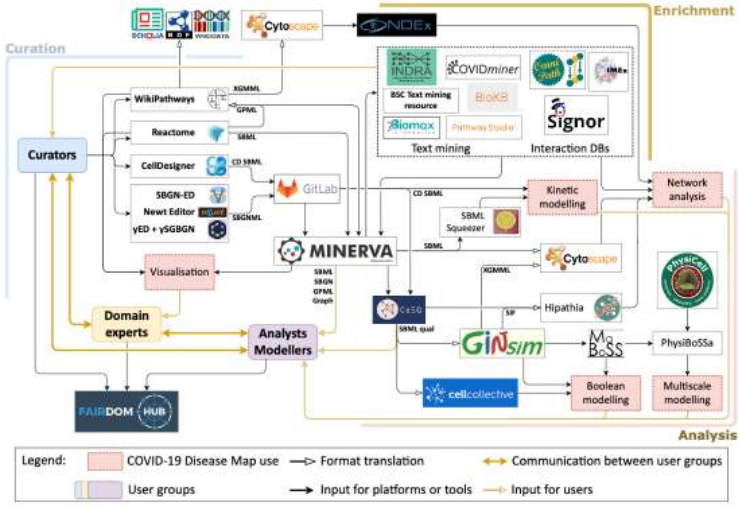




M. Getz et al. (2021)

Conclusion

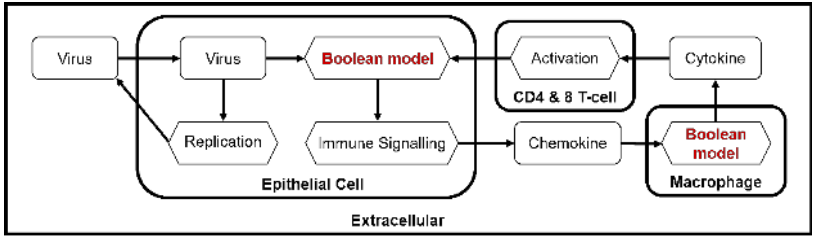
› PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2



M. Ostaszewski et al. (2021)

Conclusion

› PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2

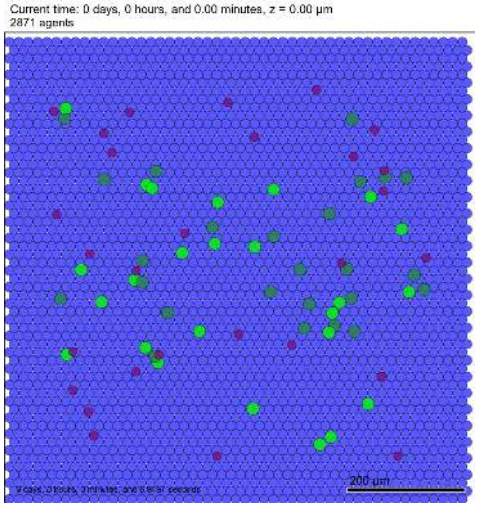


› Epithelial cells Apoptosis model from Anna Niarakis' lab (40 nodes)

› Macrophage model from Martin Kuiper's lab (157 nodes)

Conclusion

› PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2



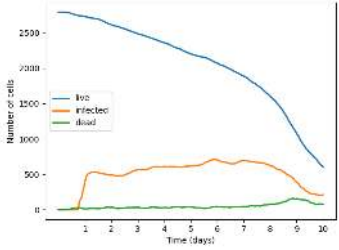
- **lung epithelium:** are colored by their (assembled) virion loads in four colors
  - Cells with 0 assembled virions.
  - Cells with 1-9 assembled virions.
  - Cells with 10-99 assembled virions.
  - Cells with 100-999 assembled virions.
  - Cells with 1000+ assembled virions.
  - Apoptotic (dead from viral load)
- **CD8⁺ T cells:** are colored by their cell cycle status
  - G0G1, S, G2, M cycle
- **CD4⁺ T cells:** are colored by their cell cycle status
  - G0G1, S, G2, M cycle
- **macrophage:** are colored by their cell cycle status
  - G0G1, S, G2, M cycle
  - Activated macrophage
  - Exhausted macrophage
  - Hyperactivated macrophage
- **neutrophil:** are colored by their cell cycle status
  - G0G1, S, G2, M cycle
- **fibroblast:** are colored by their cell cycle status
  - G0G1, S, G2, M cycle
- **Dendritic cells:** are colored by their cell cycle status
  - G0G1, S, G2, M cycle
  - Activated DC

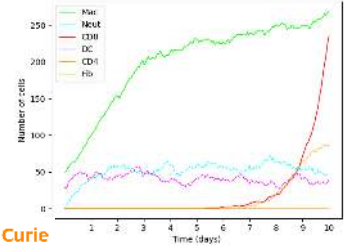
  Apoptotic immune cells
  - Apoptotic
- **Background:** Contour plot of released virus that is diffusing in and above the tissue.

Conclusion

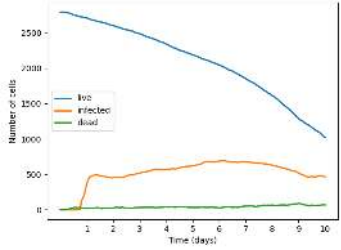
› PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2

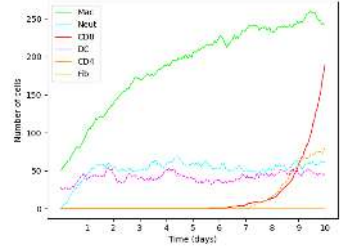
WT





FADD KO

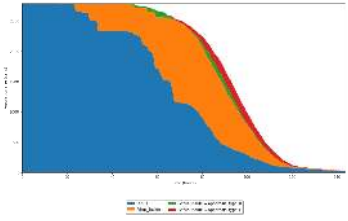




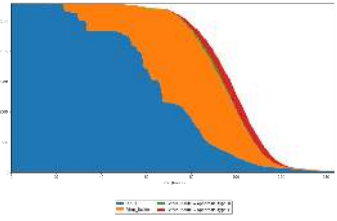
Conclusion

› PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2

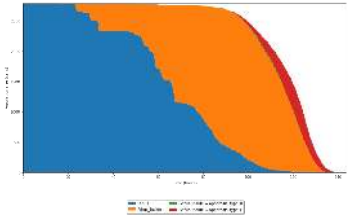
50% M inhibition



80% M inhibition



95% M inhibition



› 100% inhibition blocks the activation of the immune system

› 50% inhibition has little effect compared to no inhibition

› More inhibition adds more delay to the immune system activation

Conclusion

- PhysiBoSS-COVID : Lung tissue infection by SARS-CoV-2
  - Update to latest version of SARS-CoV-2 tissue simulator
  - Improve cell-type-specific boolean models
  - Model personalization with patient's data
  - Simulate possible treatments

Conclusion

- PhysiBoSS started as a fork of PhysiCell, even before it was published
- 5 years later, PhysiBoSS is now tightly included into PhysiCell ecosystem
- Writing models is getting easier !
- Feel free to send me questions/issues/suggestions

https://github.com/PhysiBoSS/PhysiBoSS

vincent.noel@curie.fr

Aknowledgments

institutCurie

GUSTAVE ROUSSY
CANCER CAMPUS GRAND PARIS

INSTITUT PASTEUR

INDIANA UNIVERSITY

bsc Barcelona Supercomputing Center

PerMedCoE


Gautier Stoll


Eric Viara


Laurence Calzone


Emmanuel Barillot


Marco Ruscone


Andrei Zinovyev


Arnau Montagud


Miguel Ponce de Leon


Gerard Pradas


Randy Heiland


Paul Macklin


Furkan Kurtoglu


Gaëlle Letort