Slides, videos, links and more:
https://github.com/physicell-training/ws2022

institutCurie

Session 11: Simulating a TNF treatment with PhysiBoSS

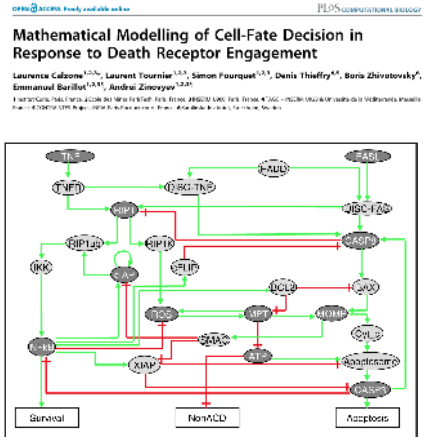
Vincent Noël – Institut Curie
@vincentnoel72

PhysiCell Project

July 27, 2022

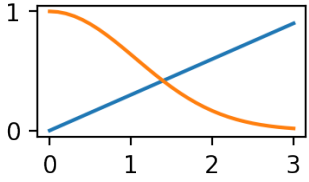
Outline

- Boolean models
- MaBoSS : Continuous time boolean modelling
- PhysiBoSS
- TNF model



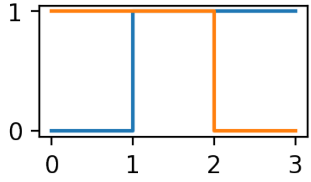
Boolean models

Quantitative (ODEs)



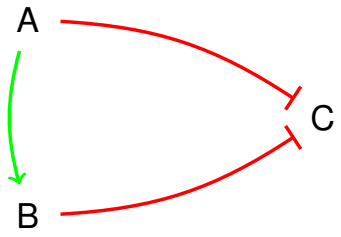
- Values can be any quantity
- Continuous time
- Difficult to write
- Difficult to simulate large models

Qualitative (Boolean)



- Values are 0/1 (false/true)
- Sequences of events
- Easy to write
- Can simulate large models

Boolean models

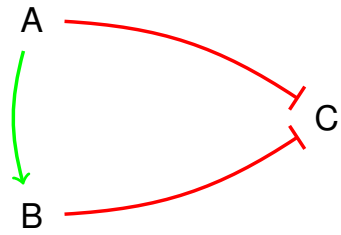


Boolean

Formulas:

- A = input
- B = A
- C = !(A and B)

Boolean models



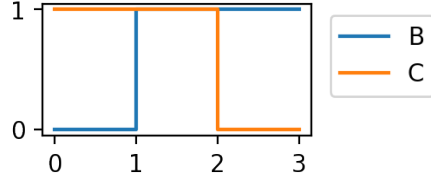
Boolean

Formulas:

- A = input
- B = A
- C = !(A and B)

Initial values:

- $A_0 = 1$
- $B_0 = 0$
- $C_0 = 1$



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¶]

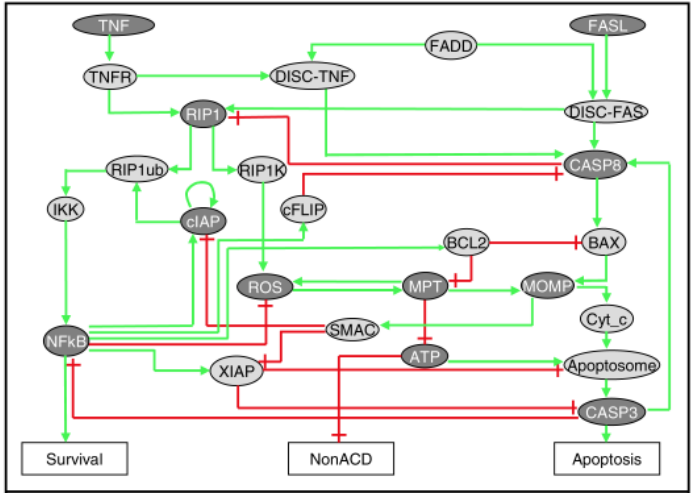
1 Institut Curie, Paris, France, **2** Ecole des Mines ParisTech, Paris, France, **3** INSERM U900, Paris, France, **4** TAGC – INSERM U928 & Université de la Méditerranée, Marseille, France, **5** CONTRAINTES Project, INRIA Paris-Rocquencourt, France, **6** Karolinska Institutet, Stockholm, Sweden

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.

Calzone et al, 2010

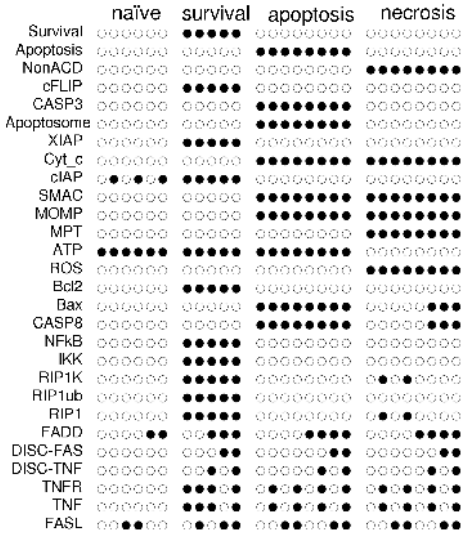
Boolean models



Calzone et al, 2010

Boolean models

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



Calzone et al, 2010

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 naïve 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

MaBoSS : Continuous time boolean modelling

MaBoSS : Continuous time boolean modelling

Methodology article | Open Access | Published: 29 August 2012

Continuous time boolean modeling for biological signaling: application of Gillespie algorithm

Gautier Stoll, Eric Viara, Emmanuel Barillot & Laurence Calzone

BMC Systems Biology 6, Article number: 116 (2012) | Download Citation

6306 Accesses | 32 Citations | 1 Altmetric | Metrics »



Gautier Stoll



Eric Viara



Laurence Calzone



Emmanuel Barillot

MaBoSS : Continuous time boolean modelling

Markovian **Bo**olean **S**tochastic **S**imulator



https://maboss.curie.fr/

- Boolean
- Markovian
- Physical time
- Handle different time scale processes (transcription, phosphorylation, etc.)

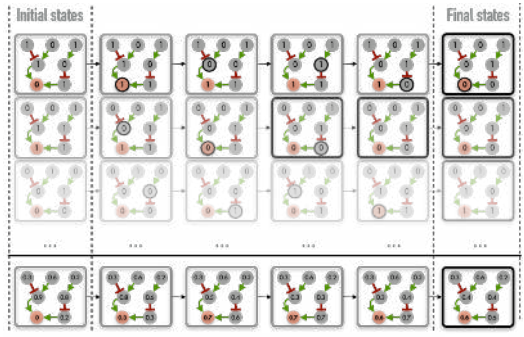
⇒ Fills the gap between ODE and Boolean modeling

MaBoSS : Continuous time boolean modelling

- Simulate transitions from a boolean network state to another using a markov chain
- Transition rate :

$$p(S \to S') = \begin{cases} R_{up}(S), & \text{if} S_i = 0 \\ R_{down}(S), & \text{if} S_i = 1 \end{cases}$$

- These transitions are stochastic : multiple trajectories are possible



Béas et al, 2019

From a set of simulated trajectories, we compute mean trajectories:

⇒ We obtain probabilities per state over time

MaBoSS : Continuous time boolean modelling

- Model definition : BND file

  - Definition of activation and inactivation rates
  - Allows complex cases : multiple rates for multiple states, ...
  - Possibility to use parameters defined in the simulation settings
  - Now allows other boolean model formats : SBML-qual, BNet

```
node A {
    rate_up = 1;
    rate_down = 0;
}

node B {
    logic = A;
    rate_up = @logic ? $act_B : 0;
    rate_down = 0;
}

node C {
    logic = !A | !B;
    rate_up = @logic ? $act_C : 0;
    rate_down = @logic ? 0 : $act_C;
}
```

MaBoSS : Continuous time boolean modelling

› Simulation settings : CFG file

- Initial state of nodes (Fixed, Stochastic)
- Internal/Output variables
- Parameters
- Settings

```
A.istate = TRUE;
B.istate = FALSE;
C.istate = TRUE;

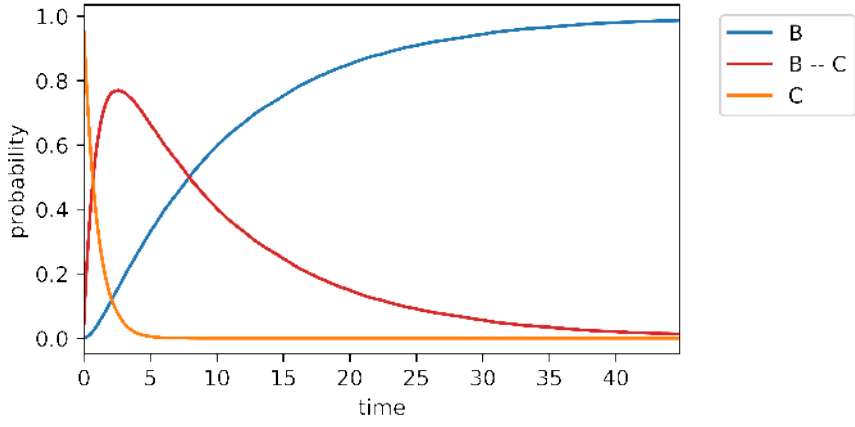
A.is_internal = TRUE;

$act_B = 1;
$act_C = 0.1;

max_time = 45;
sample_count = 10000;
use_physrandgen = 1;
thread_count = 1;
```

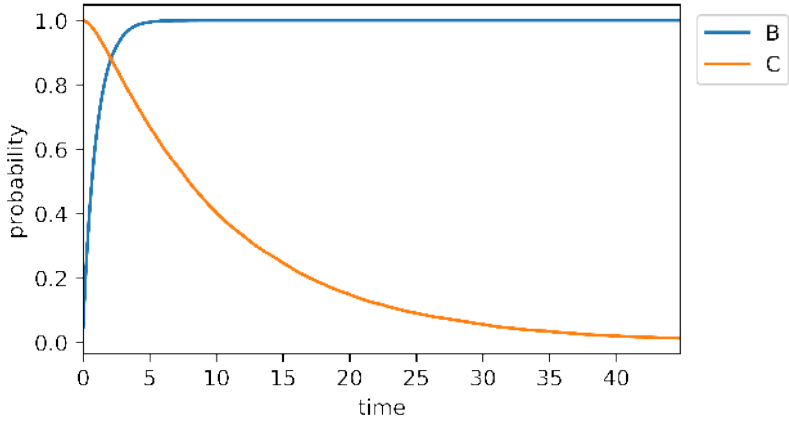
MaBoSS : Continuous time boolean modelling

› State probability trajectories



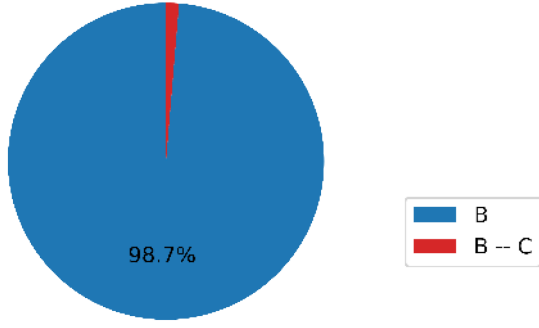
MaBoSS : Continuous time boolean modelling

› Node probability trajectories



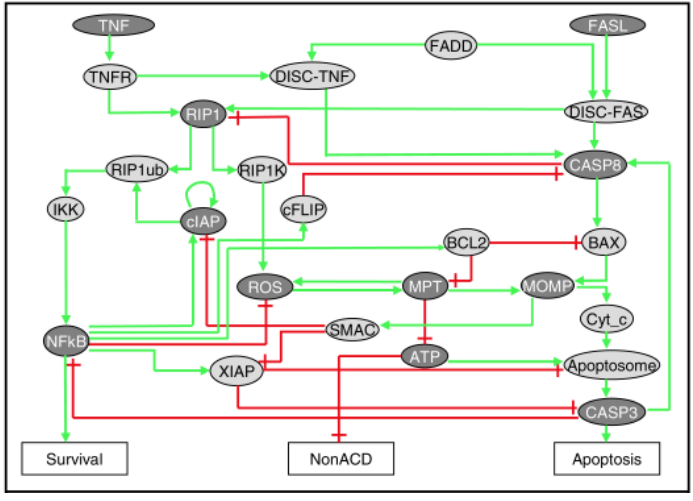
MaBoSS : Continuous time boolean modelling

› Final ($\neq$ steady) state distribution



MaBoSS : Continuous time boolean modelling

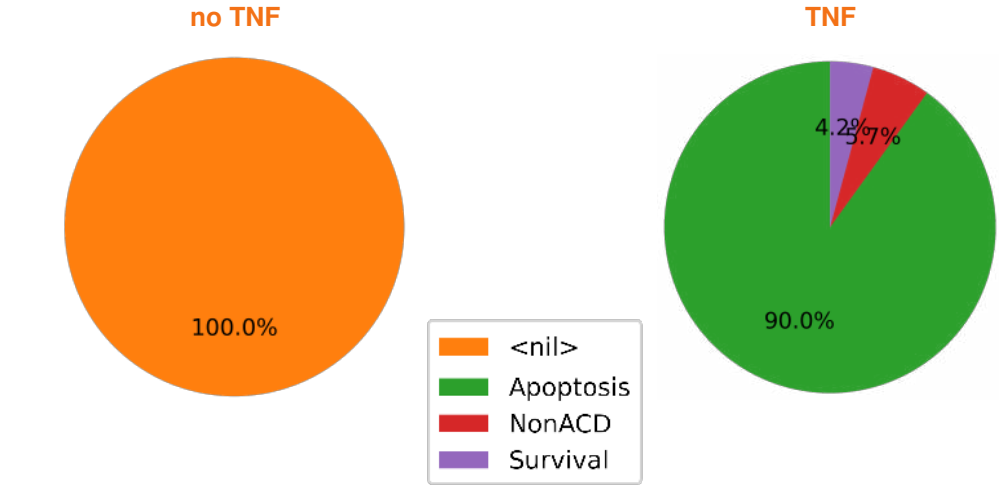
› Cell fate model



Calzone et al, 2010

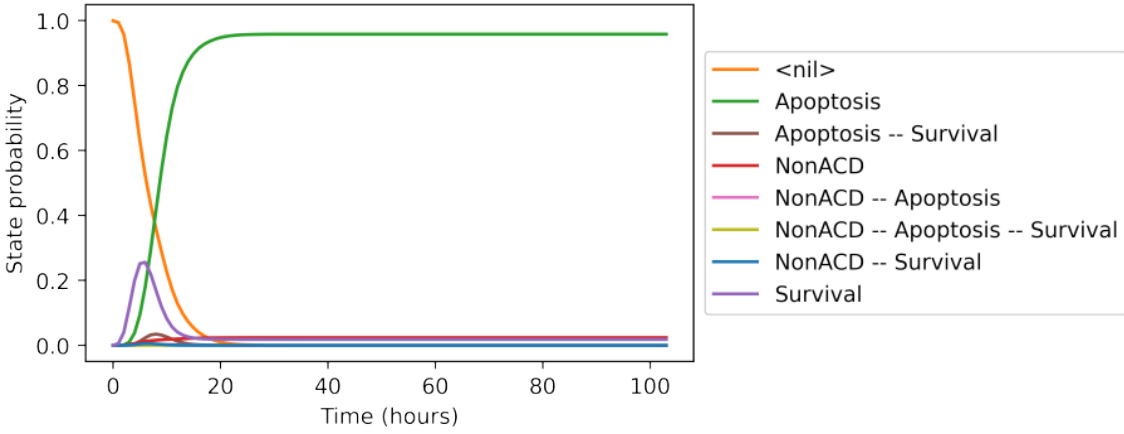
MaBoSS : Continuous time boolean modelling

› Cell fate model



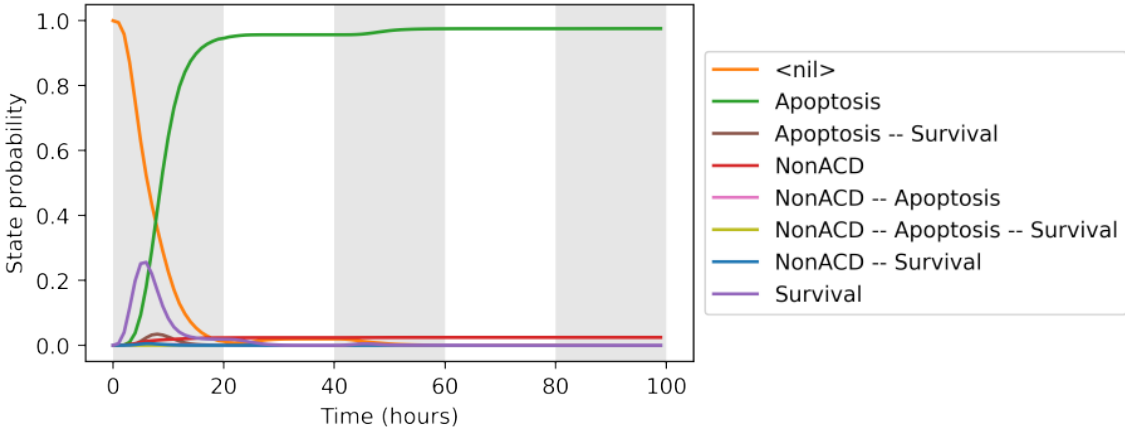
MaBoSS : Continuous time boolean modelling

- Cell fate model



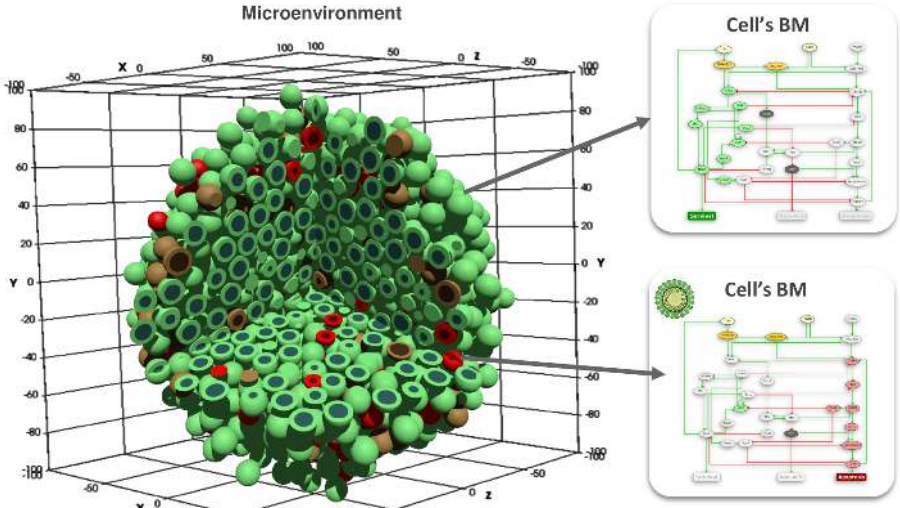
MaBoSS : Continuous time boolean modelling

› Cell fate model



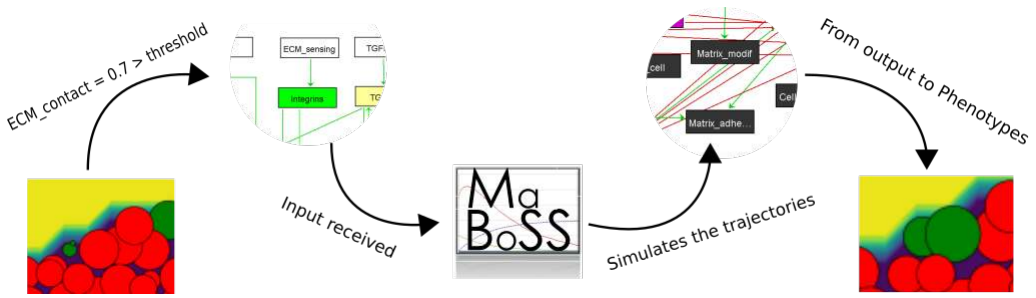
PhysiBoSS

PhysiBoSS



Ponce de Leon et al, in preparation

PhysiBoSS



› Each cell runs one single (stochastic) simulation every *intracellular_dt*

PhysiBoSS

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

```
<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

Inputs : Signals to MaBoSS nodes

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

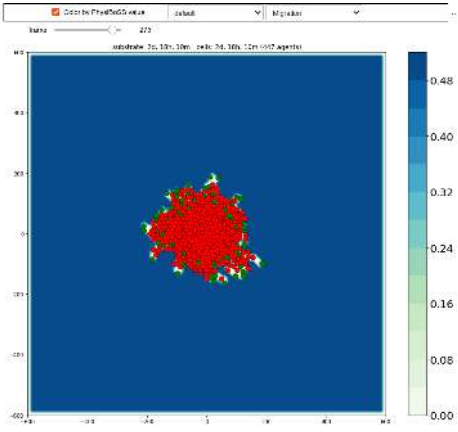
```
<input physicell_name="oxygen" intracellular_name="A">
    <settings>
        <action>inhibition</action>
        <threshold>10</threshold>
        <inact_threshold>10</inact_threshold>
    </settings>
</input>
```

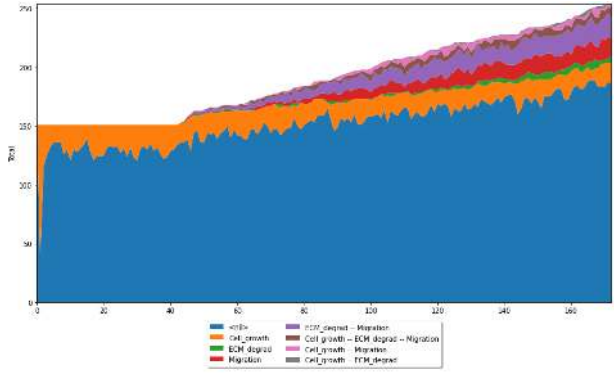
Outputs : MaBoSS nodes to Behaviours

- 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>
```

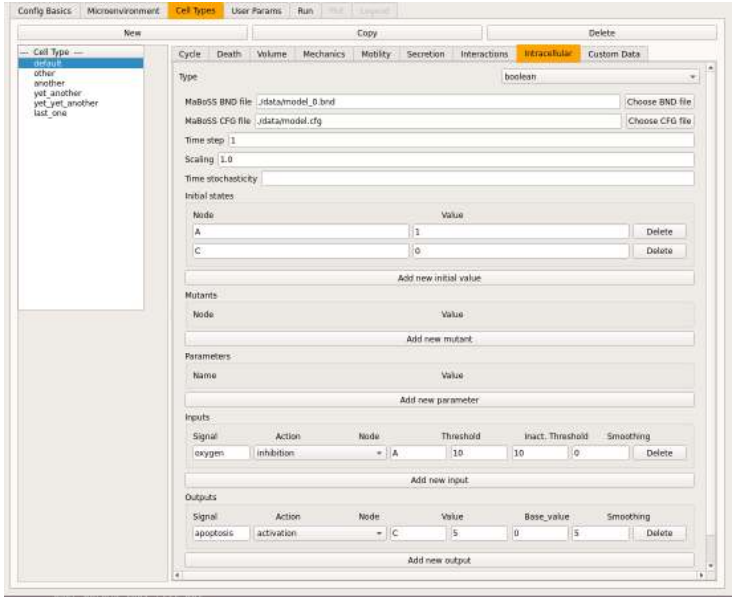
PhysiBoSS



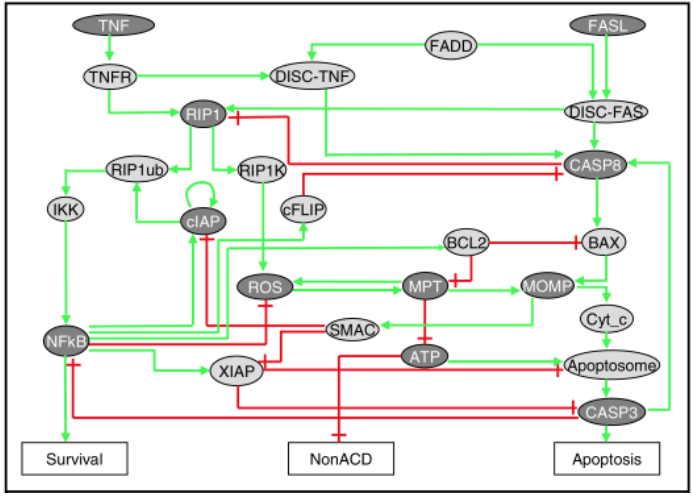


https://github.com/sysbio-curie/PhysiCell-Jupyter-GUI

PhysiBoSS


Config Basics
Microenvironment
Cell Types
User Params
Run
New
Copy
Delete
Cell Type
default
other
another
yet_another
yet_yet_another
last_one
Cycle
Death
Volume
Mechanics
Mobility
Secretion
Interactions
Intracellular
Custom Data
Type
boolean
MaBoSS BND file
./data/model_0.bnd
Choose BND file
MaBoSS CFG file
./data/model.cfg
Choose CFG file
Time step
1
Scaling
1.0
Time stochasticity
Initial states
Node
Value
A
1
Delete
C
0
Delete
Add new initial value
Mutants
Node
Value
Add new mutant
Parameters
Name
Value
Add new parameter
Inputs
Signal
Action
Node
Threshold
Inact. Threshold
Smoothing
oxygen
inhibition
A
10
10
0
Delete
Add new input
Outputs
Signal
Action
Node
Value
Base_value
Smoothing
apoptosis
activation
C
5
0
5
Delete
Add new output

PhysiBoSS



Calzone et al, 2010

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

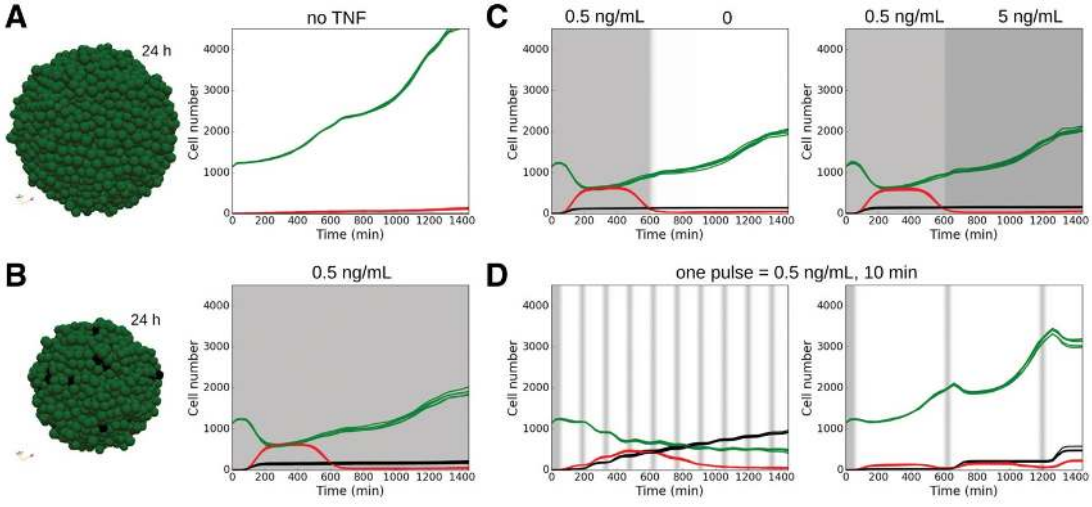
Bioinformatics, Volume 35, Issue 7, 01 April 2019, Pages 1188–1196, https://doi.org/10.1093/bioinformatics/bty766

Published: 30 August 2018 **Article history**



Gaëlle Letort

PhysiBoSS



Letort et al, 2019

PhysiBoSS

ORIGINAL RESEARCH article

Front. Mol. Biosci., 06 April 2022
Sec. Biological Modeling and Simulation
https://doi.org/10.3389/fmolb.2022.836794

This article is part of the Research Topic
Systems Modeling: Approaches and Applications - Volume II
View all 22 Articles >

Optimizing Dosage-Specific Treatments in a Multi-Scale Model of a Tumor Growth

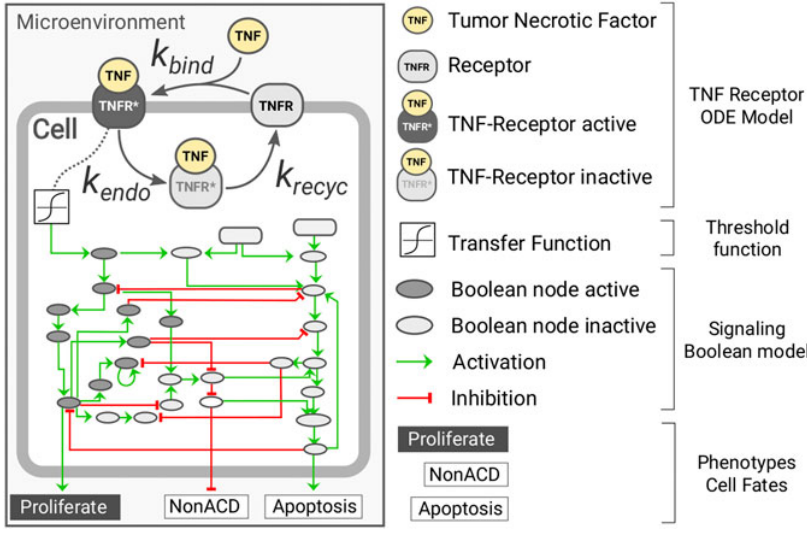
Miguel Ponce-de-Leon[1]*, Arnau Montagud[1], Charilaos Akasiadis[2], Janina Schreiber[1], Thaleia Ntiniakou[1] and Alfonso Valencia[1,3]

[1] Barcelona Supercomputing Center (BSC), Barcelona, Spain
[2] Institute of Informatics and Telecommunications, NCSR "Demokritos", Agia Paraskevi, Greece
[3] ICREA, Pg. Lluis Companys, Barcelona, Spain



Miguel Ponce de Leon

PhysiBoSS



Ponce de Leon et al., 2019

PhysiBoSS

Goals:

- Develop a simple model of TNF treatment
- Only boolean
- Using PhysiBoSS 2.0

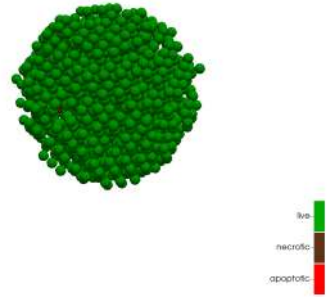
TNF model

- PhysiCell model of spheroid

  - Simple model of tumor spheroid

  - Exponential growth

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



TNF model

› 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;
}
```

TNF model

› 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> cylinder_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;
}
```

TNF model

› 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 max 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);
```

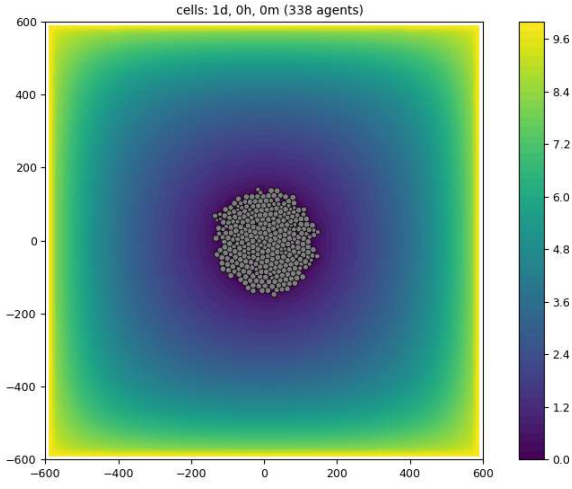
TNF model

› 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>
```

TNF model

› Adding TNF to the microenvironment



TNF model

› Adding TNF to the microenvironment

```
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

TNF model

- 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>
```

TNF model

- Choosing intracellular_dt

Short intracellular dt

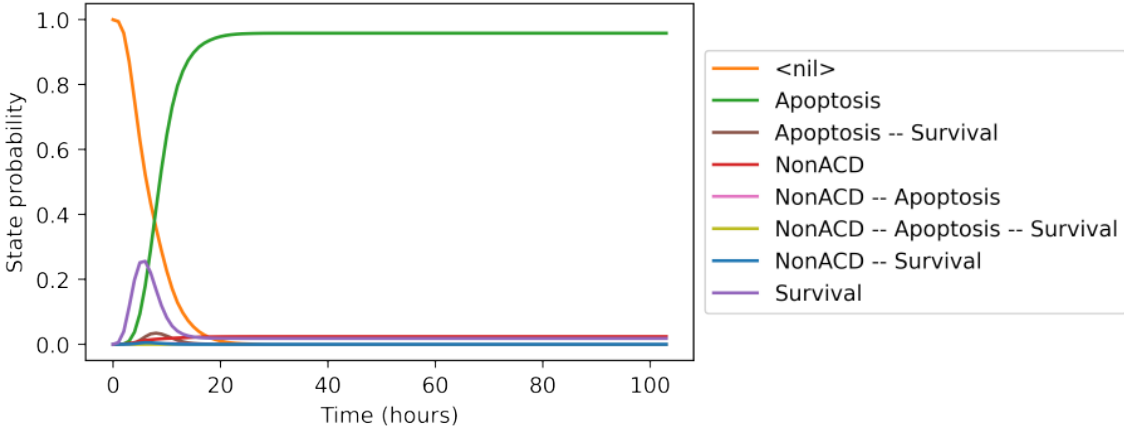
- Integrate transient activation of some nodes
- Needs a well curated model

Long intracellular dt

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

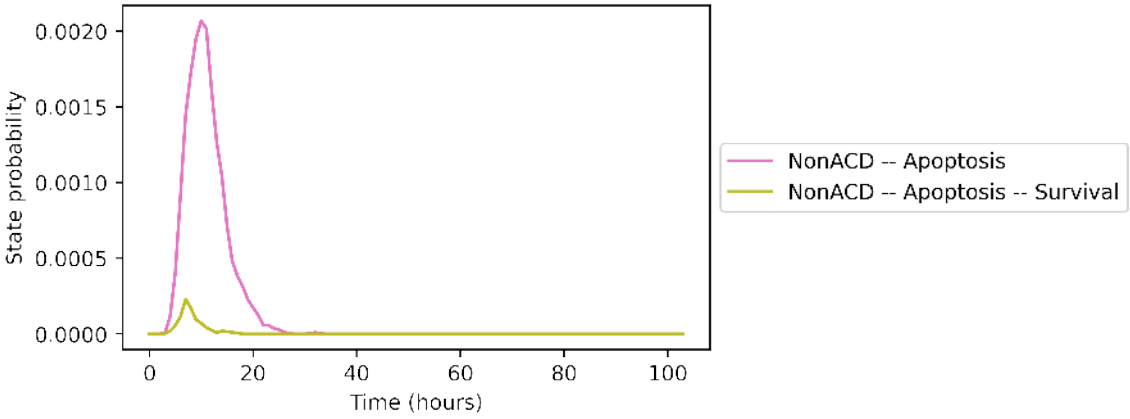
TNF model

› Choosing intracellular_dt



TNF model

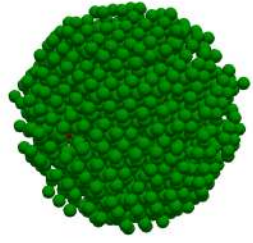
› Choosing intracellular_dt



⇒ To avoid NonACD – Apoptosis, we choose 24h

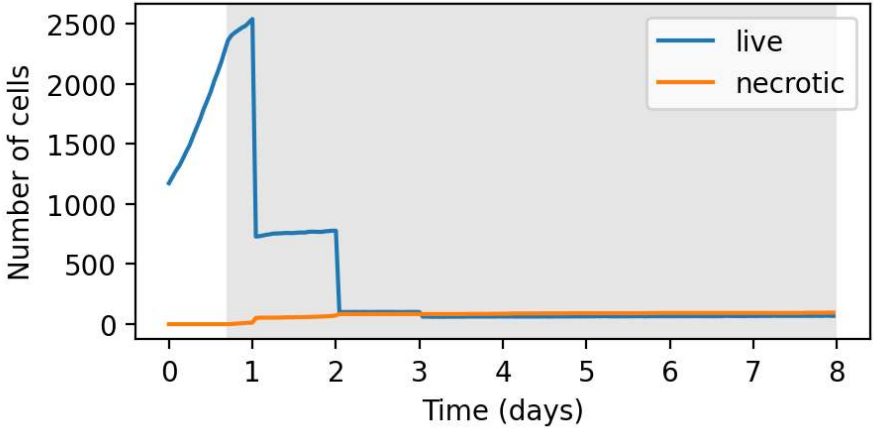
TNF model

› Results with constant TNF



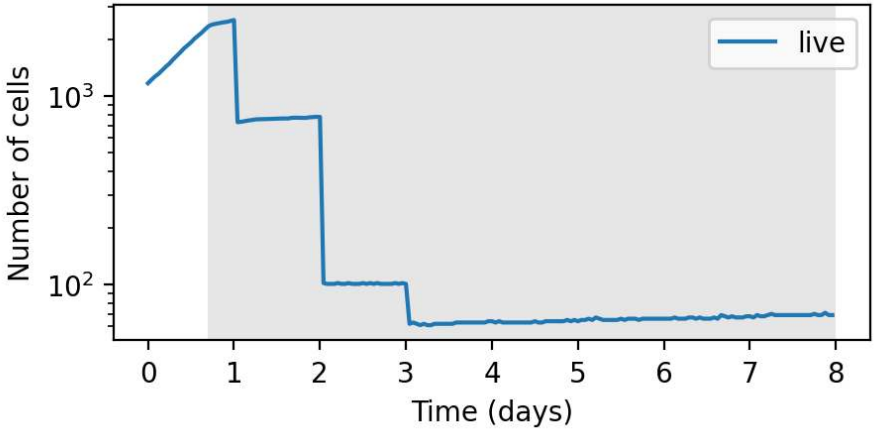
TNF model

› Results with constant TNF



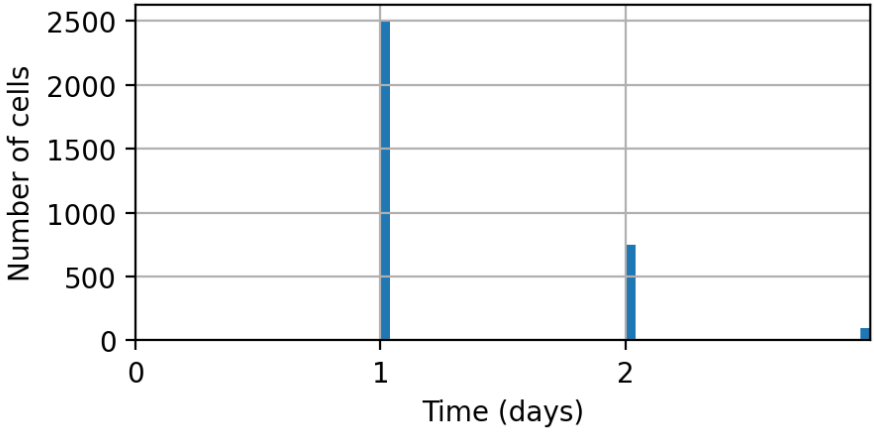
TNF model

› Results with constant TNF



TNF model

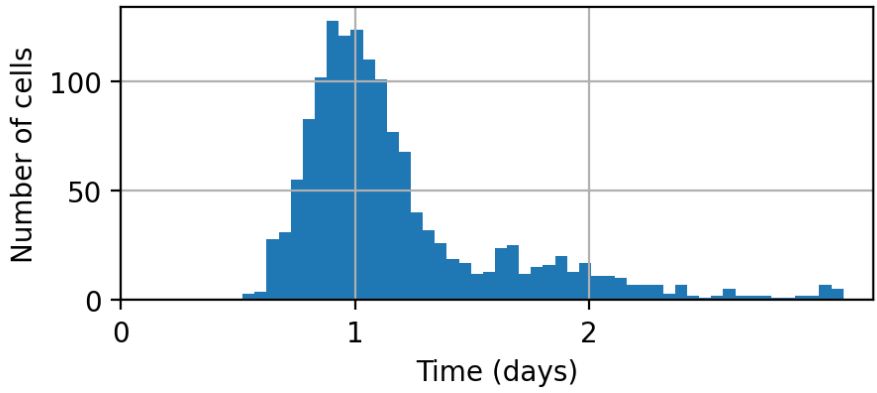
- Time stochasticity



  - time_stochasticity = 0.0

TNF model

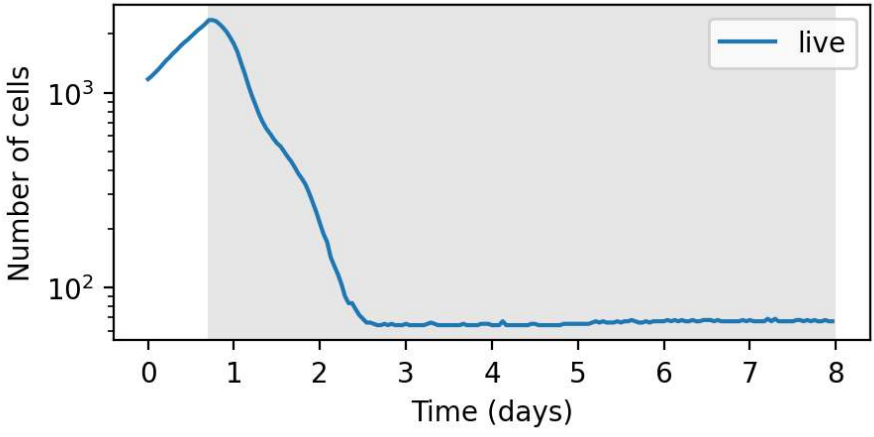
- Time stochasticity



  - time_stochasticity = 0.2

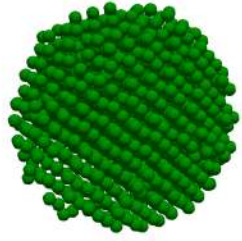
TNF model

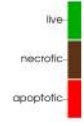
› Time stochasticity



TNF model

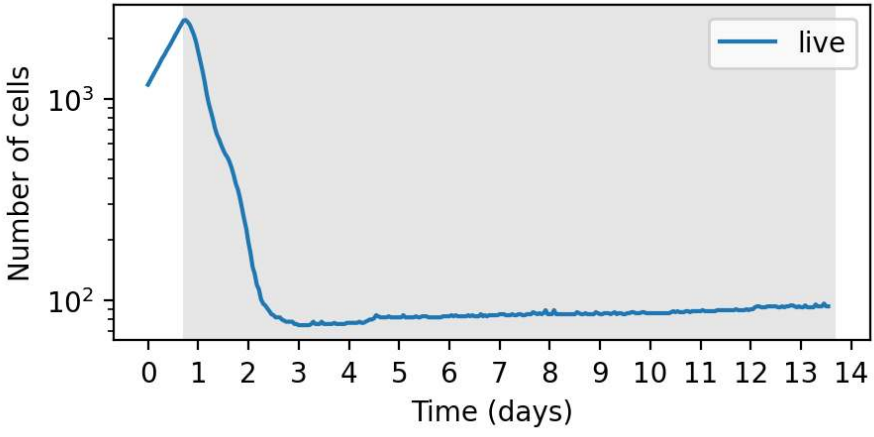
› Time stochasticity





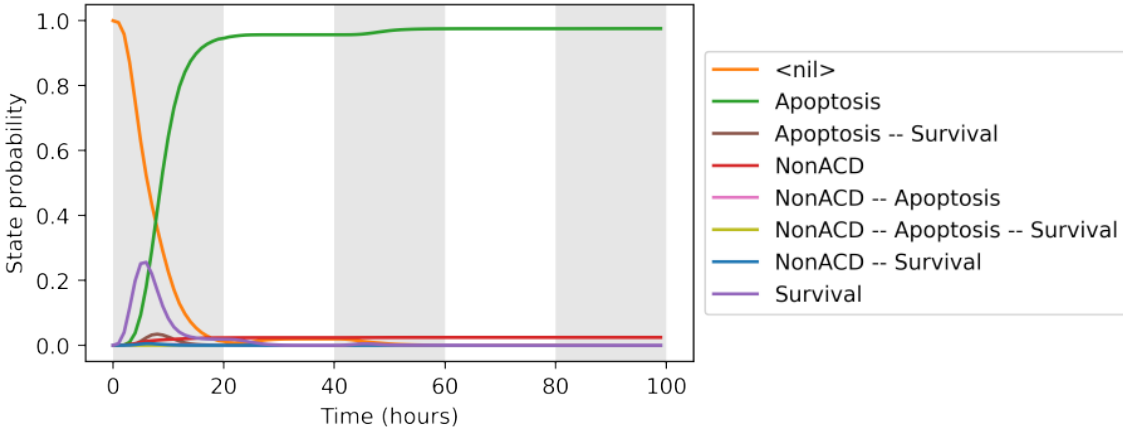
TNF model

› Creating pulses of TNF



TNF model

› Creating pulses of TNF



TNF model

› 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);
    }

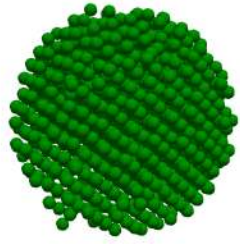
    if (PhysiCell_globals.current_time >= 12520 && PhysiCell_globals.current_time < 15400 && microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF inactivation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, false);
    }

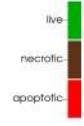
    if (PhysiCell_globals.current_time >= 15400 && PhysiCell_globals.current_time < 19720 && !microenvironment.get_substrate_dirichlet_activation(tnf_index))
    {
        std::cout << "TNF activation" << std::endl;
        microenvironment.set_substrate_dirichlet_activation(tnf_index, true);
    }

    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);
    }
}
```

TNF model

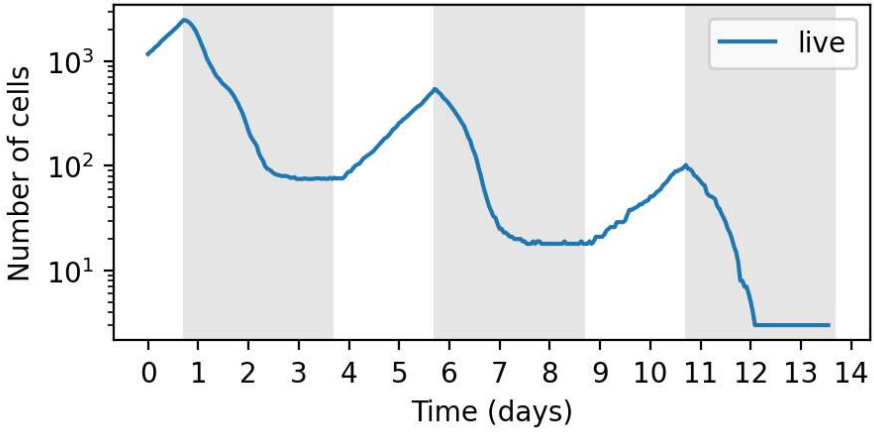
› Creating pulses of TNF





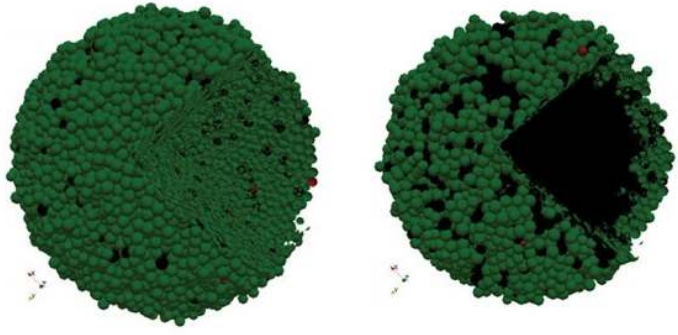
TNF model

› Creating pulses of TNF



TNF model

› Adding tumor necrotic core



Letort et al., 2019

TNF model

- 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 ;
```

TNF model

› 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>
```

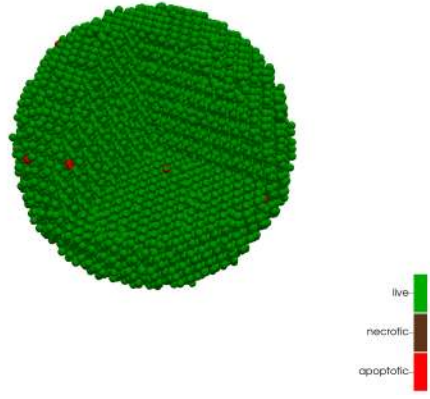
TNF model

› 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>
```

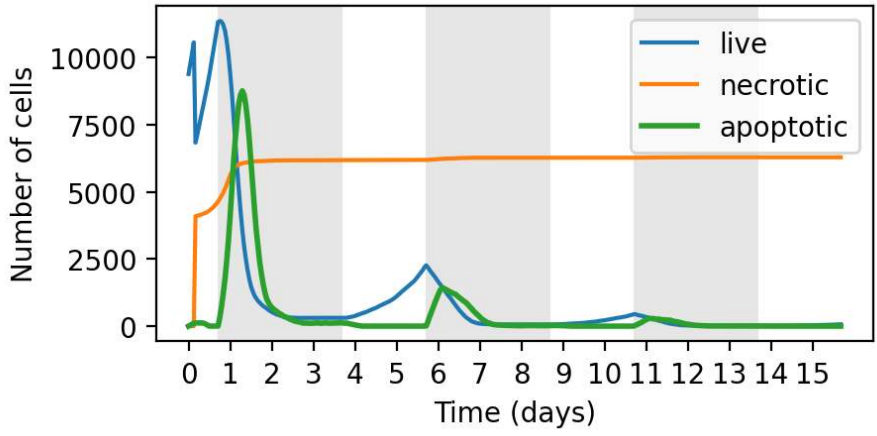
TNF model

› Adding tumor necrotic core



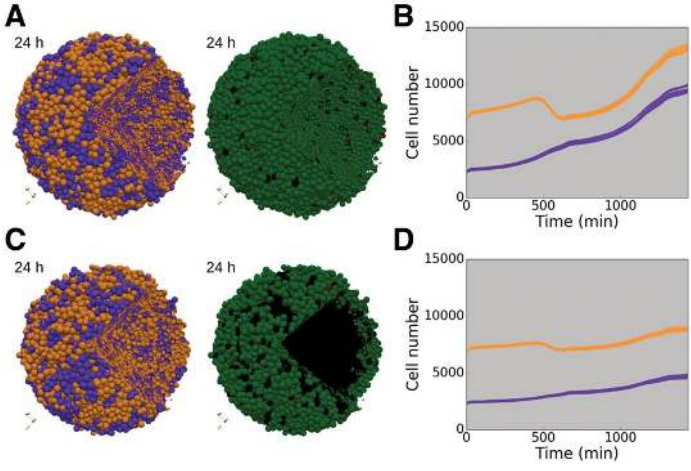
TNF model

› Adding tumor necrotic core



TNF model

› Add IKK/cFLIP mutant population



Letort et al., 2019

TNF model

- 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>
```

TNF model

- 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;

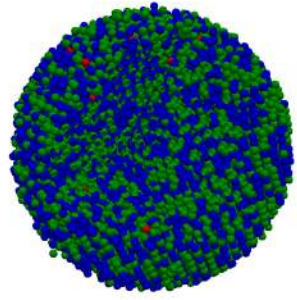

    for( int i=0; i < positions.size(); i++ )
    {
        pCell = create_cell(get_cell_definition(
            (PhysiCell::UniformRandom()*100) > parameters.doubles("percentage_mutants") ? "default":"mutant")
        );

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

    return;
}
```

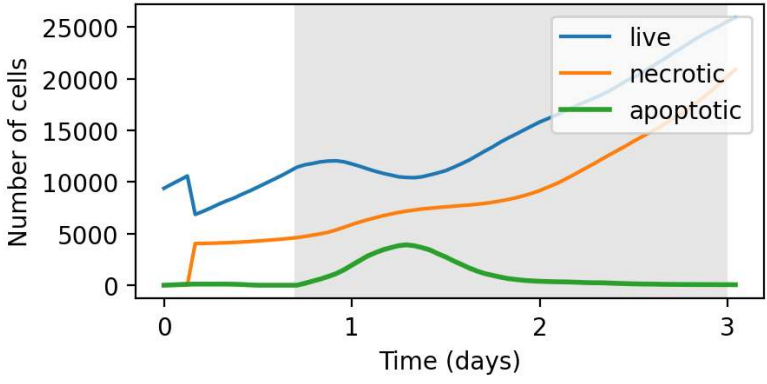
TNF model

› Add IKK/cFLIP mutant population



TNF model

› 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**

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

Funding Acknowledgements

PhysiCell Development:

- Breast Cancer Research Foundation
- Jayne Koskinas Ted Giovanis Foundation for Health and Policy
- National Cancer Institute (U01CA232137)
- National Science Foundation (1720625, 1818187)

Training Materials:

- Administrative supplement to NCI U01CA232137 (Year 2)

Other Funding:

- NCI / DOE / Frederick National Lab for Cancer Research (21X126F)
- DOD / Defense Threat Reduction Agency (HDTRA12110015)
- NIH Common Fund (3OT2OD026671-01S4)

PhysiBoSS Development:

- European Union (PrECISE project: H2020-PHC-668858, INFORE: H2020-ICT-825070, PerMedCoE: H2020-ICT-951773)
- ITMO Cancer (Chemotaxis, INVADE)