From multidimensional to multiscale models in Systems Biology – and how to check them!

David Gilbert
Computational Design Group
Synthetic Biology
Brunel University London, UK

david.gilbert@brunel.ac.uk
www.brunel.ac.uk/people/david-gilbert

Overview

- Computational models in systems biology
- Model checking
- Multiscale spatio-temporal model checking
- Multiscale spatio-temporal *meta* model checking
- Implementation
- Case study: Acute inflammation of gut and lung
- Conclusions

Model Checking

david.gilbert@brunel.ac.uk

"Formally check whether
a model of a biochemical system does what we want"

Components:

model

- the current description of a biochemical system of interest

property

- a property which we think the system should have

model checker

- a program to test whether the model has the property

To formally express time properties we use a temporal logic

david.gilbert@brunel.ac.uk

- "*I am hungry*."
- "*I am always hungry*", "*I will eventually be hungry*",
- *"I will be hungry until I eat something*".

Linear time logics restricted to single time line.

Branching logics can reason about multiple time lines.
"*There is a possibility that I will stay hungry forever*."
"*There is a possibility that eventually I am no longer hungry*."

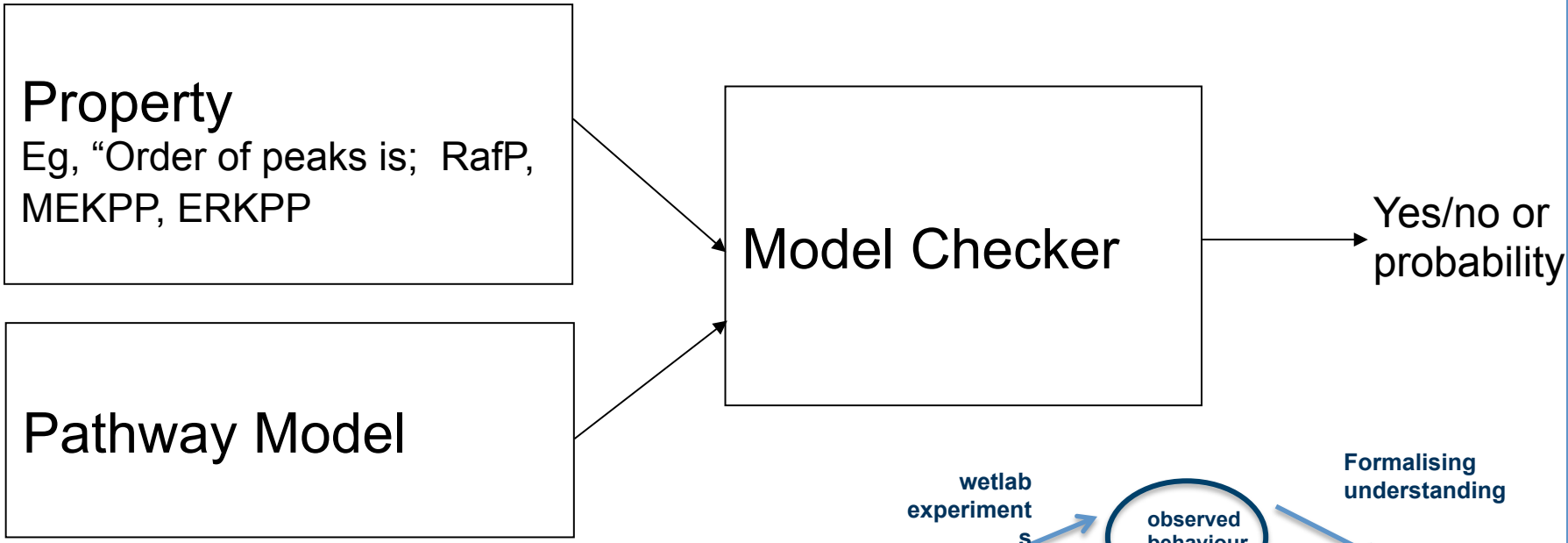
Various logics :

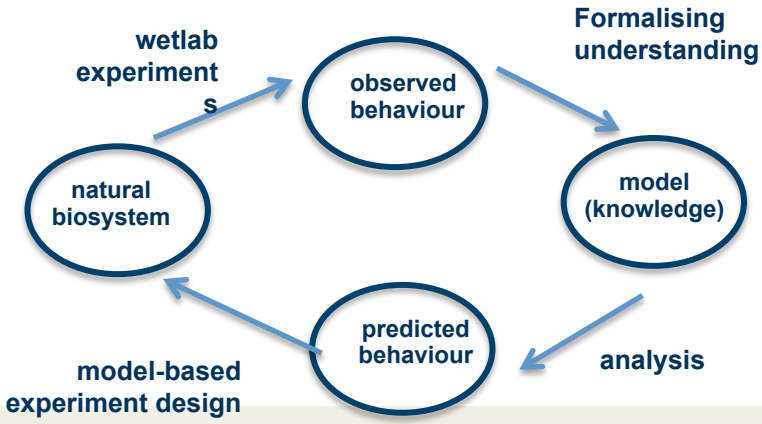
- Computational Tree Logic (CTL)
- Continuous Stochastic Logic (CSL)
- Linear-time Temporal Logic (LTL)

each with different expressivity.

Model Checking Biochemical Pathways

david.gilbert@brunel.ac.uk





Properties…

david.gilbert@brunel.ac.uk

Examples:

- After 100 seconds the concentration of Protein1 is stable
- Protein1 peaks and falls
- Protein1 peaks and stays constant
- Protein1 peaks before Protein2
- Protein1 oscillates 4 times in 5,000 seconds
- Molecules of Protein2 are required for molecules of Protein1 to be created

Analytical vs Simulative Model Checking

david.gilbert@brunel.ac.uk

Analytical:

- **Exact** probabilities & prove properties
- A model state is an association of #molecules/levels to each of the species
  - **Protein1** has 10 molecules & **Protein2** has 20 molecules
- Analytical **assesses every state** that the model can be in (reachable states)
- **State space can grow even worse than exponentially** with increasing molecules, or even be infinite!

Simulative:

Instead of analysing the constructed state space**:**

- **analyse simulation outputs**
- Simulate the model X times and check these simulations
- Simulation run = finite path through the state space
- Can't prove probabilities

Simulative Model Checking

david.gilbert@brunel.ac.uk

In-line: check the observations as they arrive

>Requires complex computational machinery: ‘combine’ simulator & model checker

>Good for biochemical observations

>Don’t always need to finish the experimental run

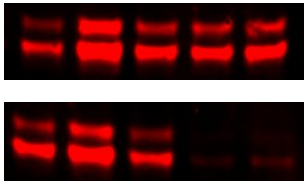
Off-line: check the observations after all have been generated

>Easier to implement computationally (simulate then check)

>Need to always define when to ‘stop’ generating observations

Simulation-based Model Checking Biochemical Pathways

david.gilbert@brunel.ac.uk



Property

Eg, “Order of peaks is RafP, MEKPP, ERKPP”

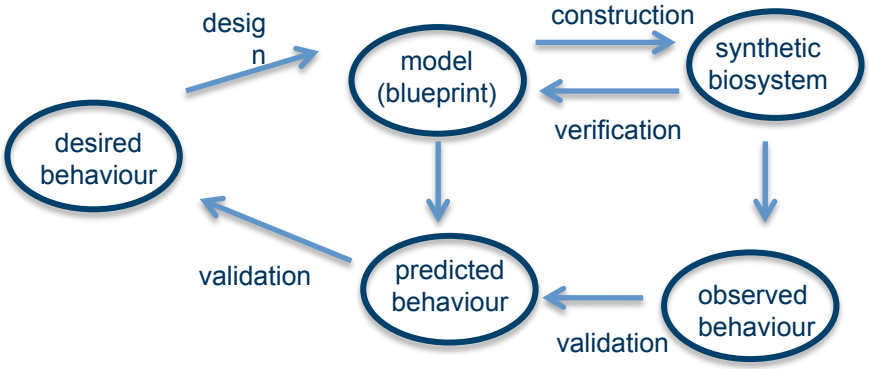
Time series data

Model

Lab

Behaviour Checker

Yes/no or probability



(P)LTL Linear Temporal Logic

G (φ) : φ always happens
F (φ) : φ happens at some time
X (φ) : φ happens in the next time point
φ_1 U φ_2 : φ_1 happens until φ_2 happens

Protein stability:

$P_{=?}$ [time >= 100 → ([Protein] >= 4 ^ [Protein] <= 6)]

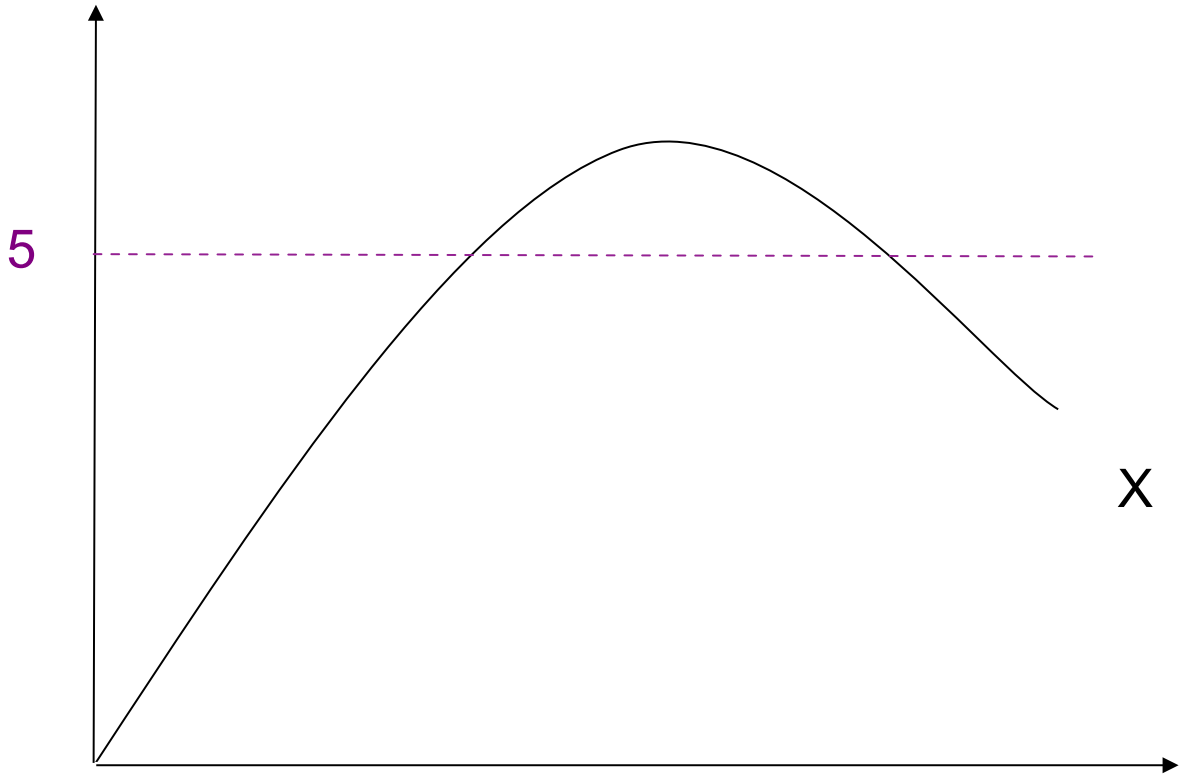
Protein concentration rises to a maximum value and then remains constant:

$P_{=?}$ [(d[Protein]> 0) U (G([Protein] >= 0.99*max[Protein]))]

MC2 with ODE Output

$$P_{=?}[\ F(\ X > 5\ )\ ]$$

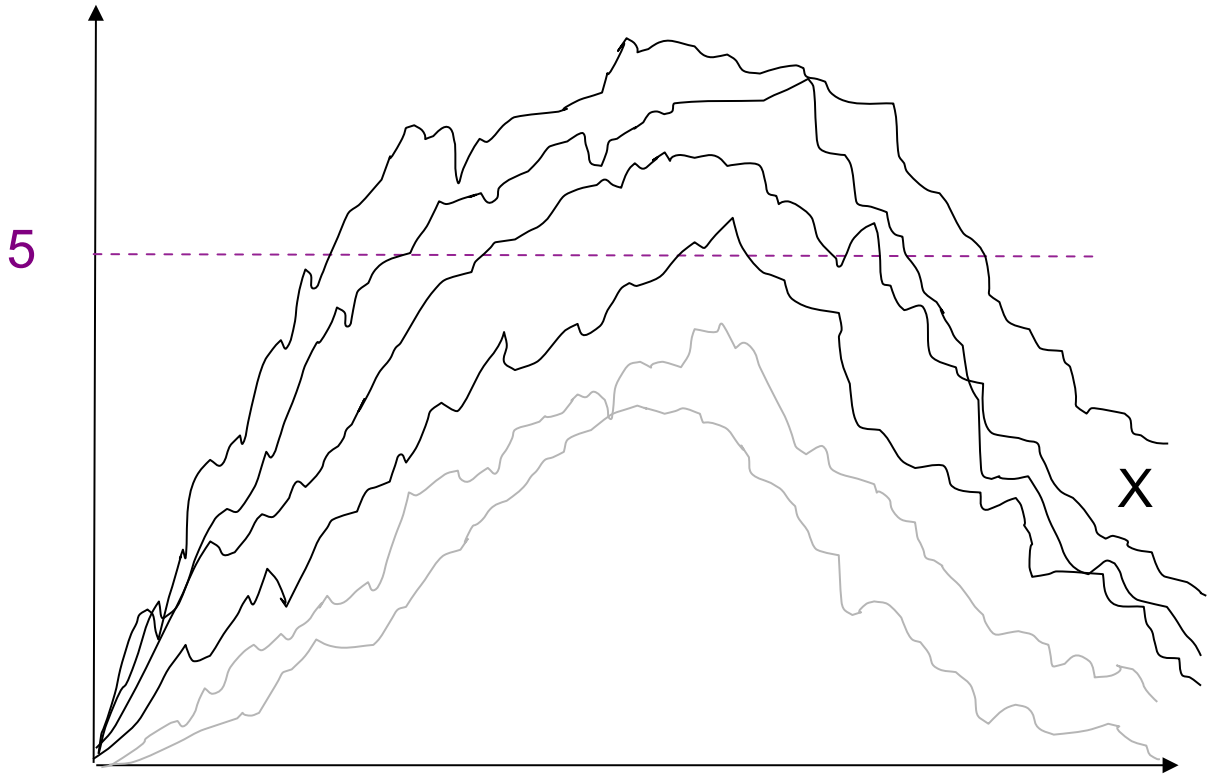
=> P = 1



MC2 with Gillespie Output

$$P_{=?}[\ F(\ X > 5\ )\ ]$$

=> P = 4/6



Qualitative to quantitative descriptions in PLTL

Qualitative:
Protein rises then falls

$$P=?\ [\ (\ d(\text{Protein}) > 0\ )\ U\ (\ G(\ d(\text{Protein}) < 0\ )\ )\ ]$$

Semi-qualitative:
Protein rises then falls to less than 50% of peak concentration

$$P=?\ [\ (\ d(\text{Protein}) > 0\ )\ U\ (\ G(\ d(\text{Protein}) < 0\ ) \wedge F\ (\ [\text{Protein}] < 0.5 * \max[\text{Protein}]\ )\ )\ ]$$

Semi-quantitative:
Protein rises then falls to less than 50% of peak concentration by 60 minutes

$$P=?\ [\ (\ d(\text{Protein}) > 0\ )\ U\ (\ G(\ d(\text{Protein}) < 0\ ) \wedge F\ (\ \text{time} = 60 \wedge \text{Protein} < 0.5 * \max(\text{Protein})\ )\ )\ ]$$

Quantitative:
Protein rises then falls to less than 100µMol by 60 minutes

$$P=?\ [\ (\ d(\text{Protein}) > 0\ )\ U\ (\ G(\ d(\text{Protein}) < 0\ ) \wedge F\ (\ \text{time} = 60 \wedge \text{Protein} < 100\ )\ )\ ]$$

Model searching

Peaks at least once

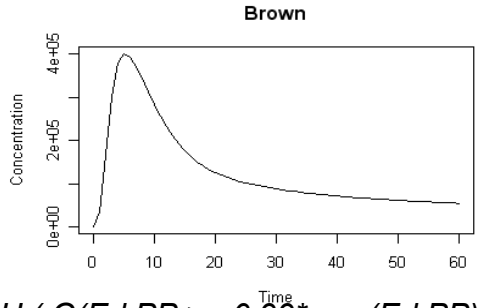
(rises then falls below 50% max concentration)

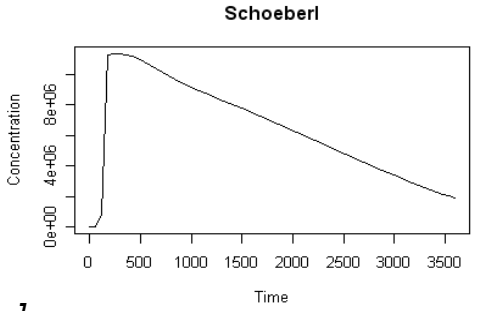
$P_{>=1}[\ ErkPP <= 0.50*max(ErkPP) \wedge d(ErkPP) > 0\ U\ (\ ErkPP = max(ErkPP) \wedge F(\ ErkPP <= 0.50*max(ErkPP)\ )\ )\ ]$

Brown

Kholodenko

Schoeberl



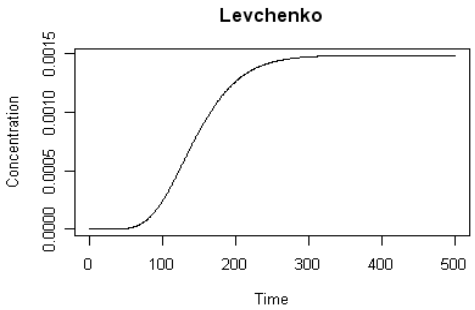


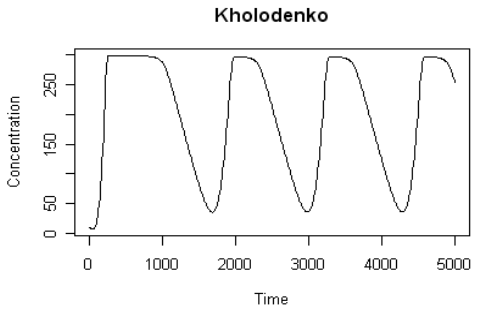
Rises and remains constant

(99% max concentration)

$P_{>=1}[ErkPP <= 0.50*max(ErkPP) \wedge (\ d(ErkPP) > 0\ )\ U\ (\ G(ErkPP >= 0.99*max(ErkPP))\ )\ ]$

Levchenko



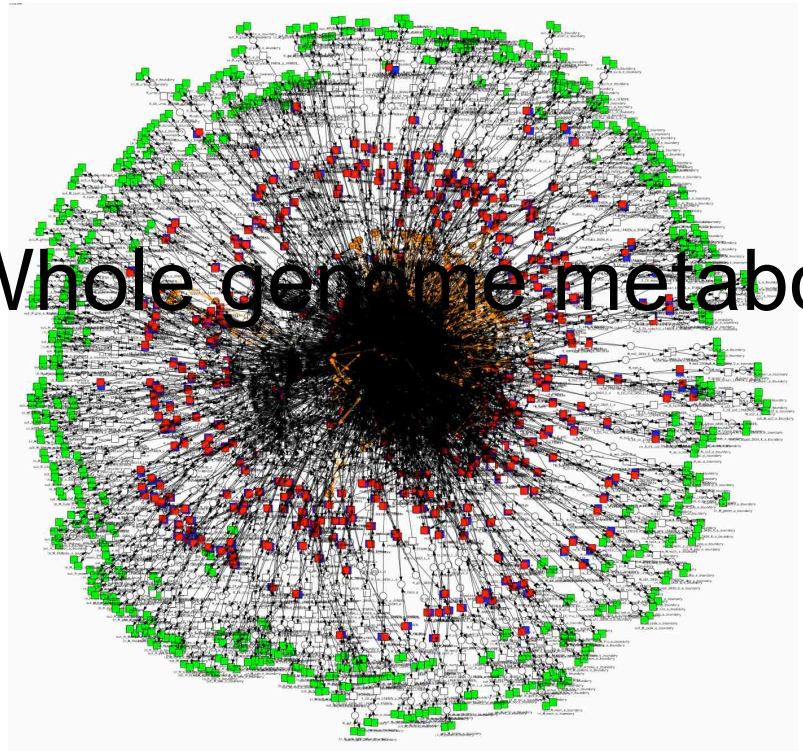


Oscillates at least 4 times

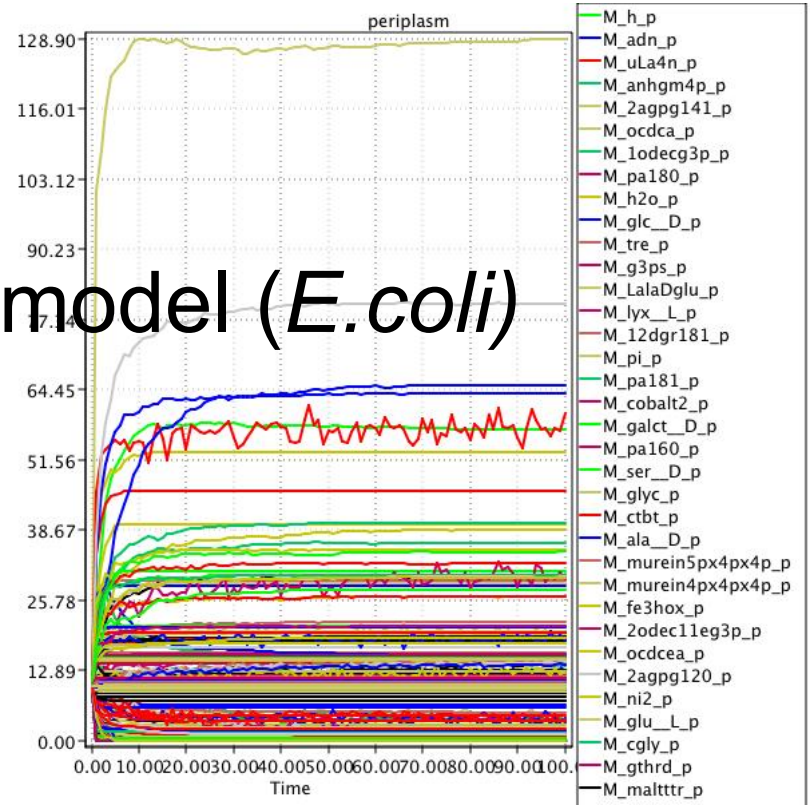
$P_{>=1}[\ F(\ d(ErkPP) > 0 \wedge F(\ d(ErkPP) < 0 \wedge \ldots)\ )\ ]$

Kholodenko

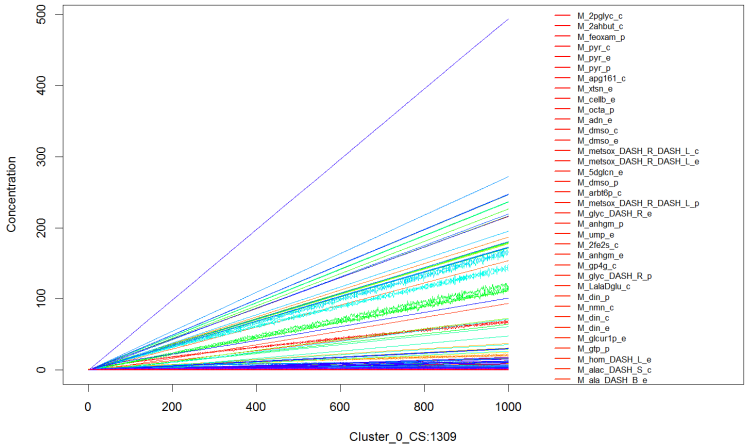
Model checking over large amount of data

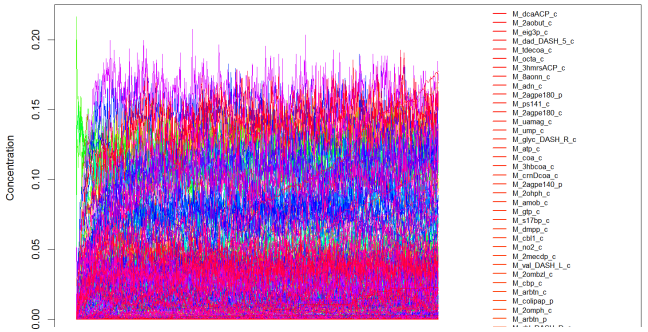


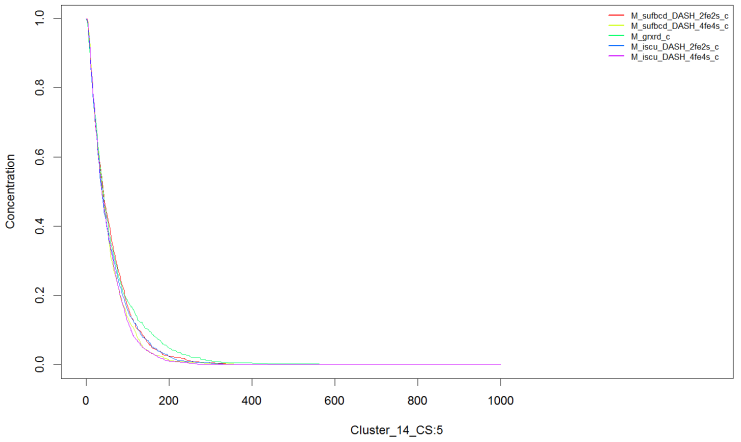
Whole genome metabolic model (*E.coli)*

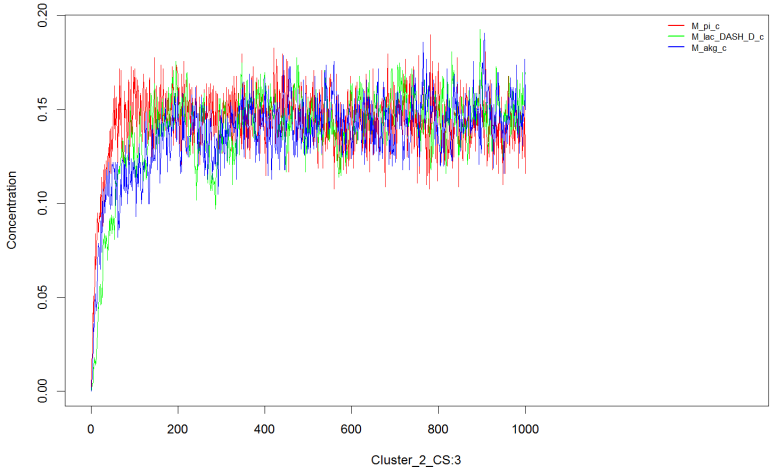


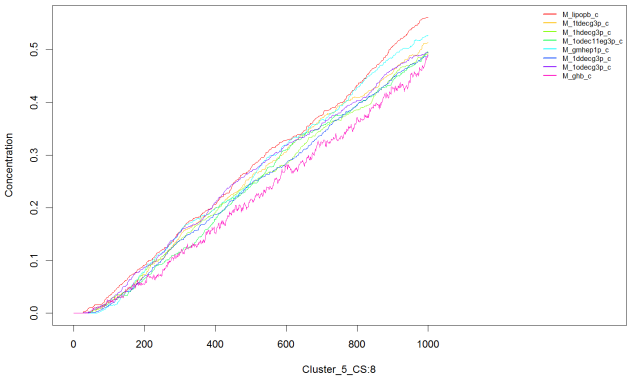
Behaviours


Concentration
Cluster_0_CS:1309


Concentration

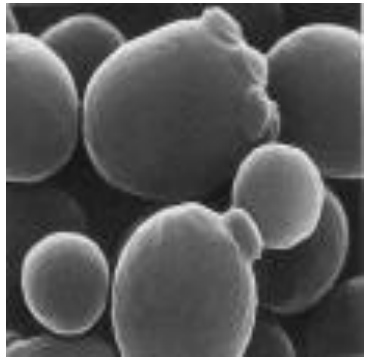

Concentration
Cluster_14_CS:5


Concentration
Cluster_2_CS:3

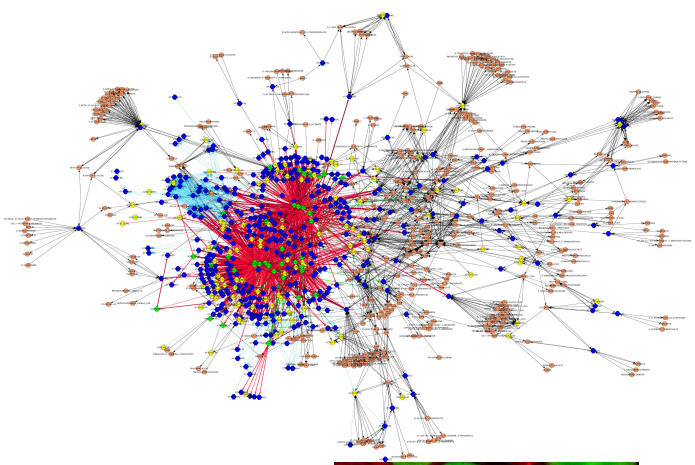

Concentration
Cluster_5_CS:8

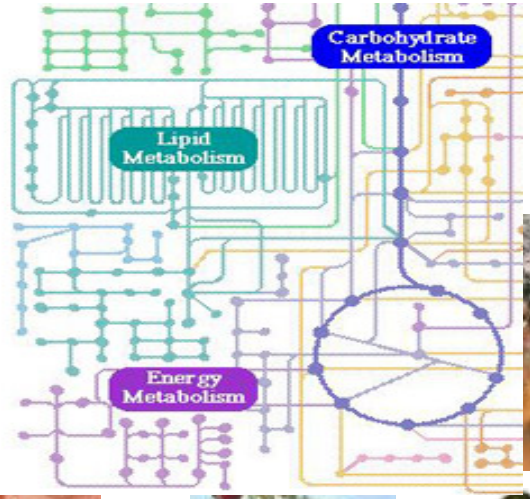
What about scaling up?

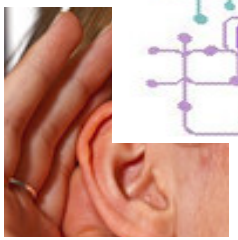
david.gilbert@brunel.ac.uk

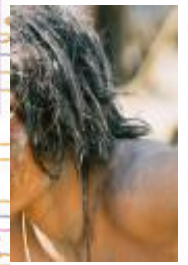


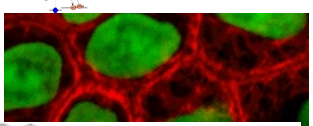
Multiscale Modelling in Biology

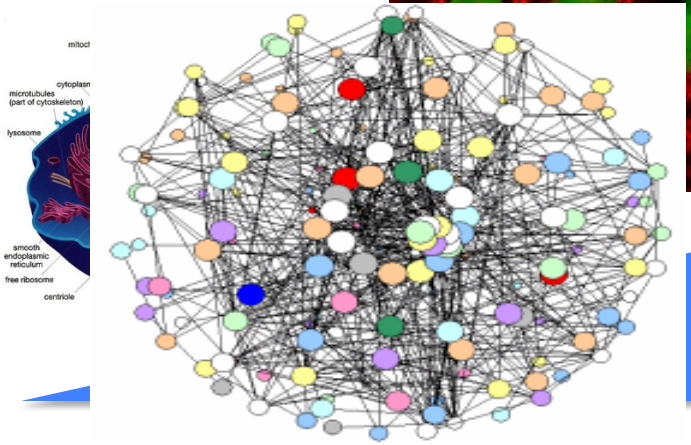


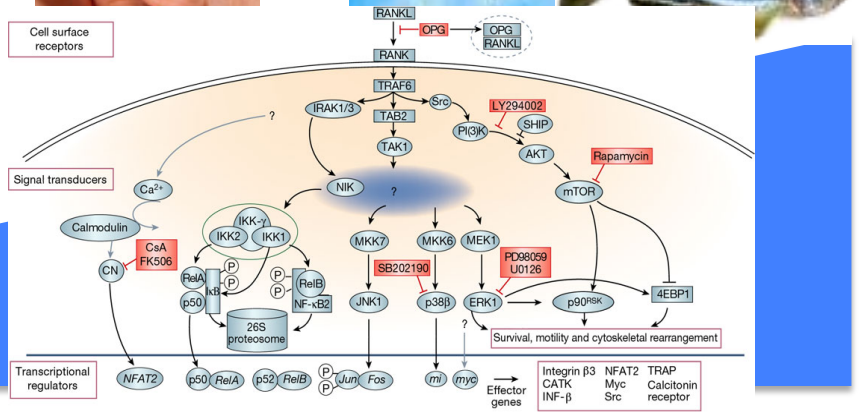






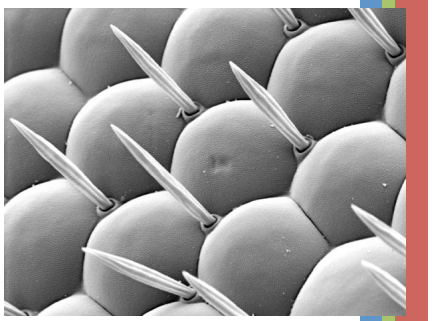






Multiscale Modelling Challenges

david.gilbert@brunel.ac.uk

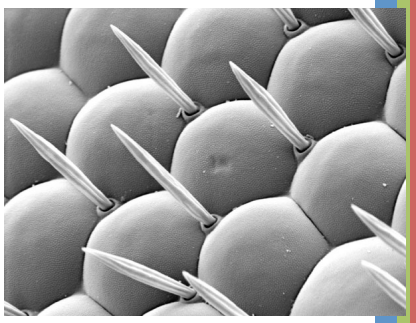


Repetition – multiple components with similar definitions
Variation – genetic mutants; random variants
Organisation - regular / irregular patterns in 1, 2 or 3 dimensions
Communication – short & long distance
Hierarchical organisation – intra or inter cellular (tissues, organs, …)
Movement – mobility (passive) & motility (active)

(Components could be molecules, organelles, cells, tissues, organs, organisms.)

…Multiscale Modelling Challenges

david.gilbert@brunel.ac.uk

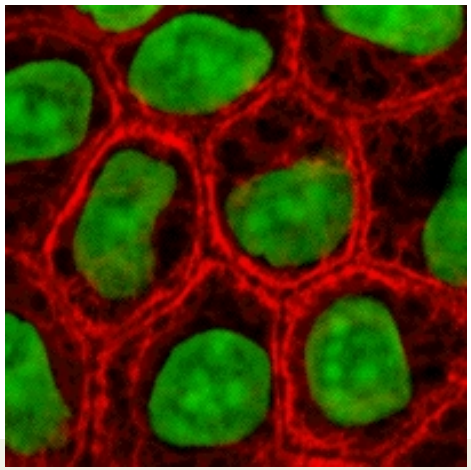


Replication - reproduction
Deletion – cell death
Irregular/semi-regular organisation of components – for example a not-exact honeycomb grid.
Dynamic grid size – for example alter size and/or topology of grid to model development. Also required for ability to insert/remove items.
Differentiation of components - for example, differentiation of embryonic stem cells or immune cells makes a less specialized cell more specialized.
Pattern formation of components - organizing a number of cells in appropriate one, two or three dimensional structures in space and time.

david.gilbert@brunel.ac.uk

Repetition of individual components

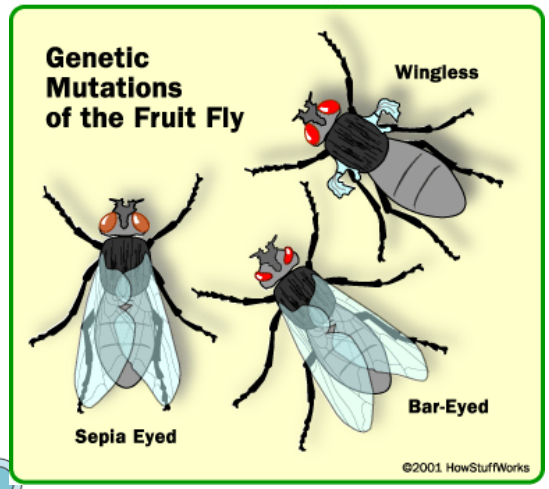
Components within a cell (organelles etc)
Multiple cells each of which having a similar definition
Repeated tissue fragments
Repeated organs (wings,…)
Repeated individual organisms

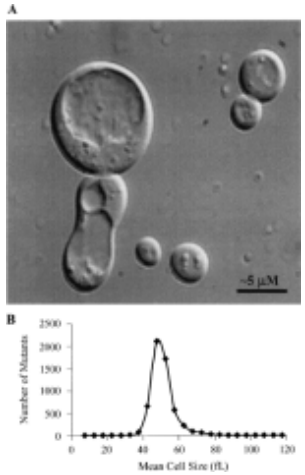


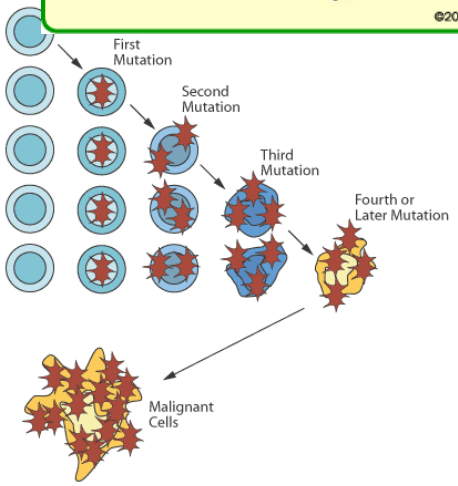
Variation

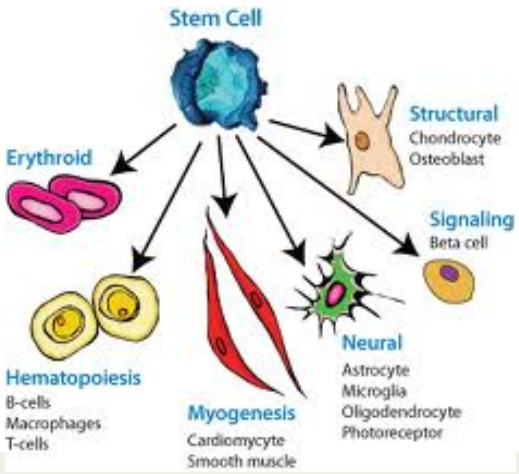
david.gilbert@brunel.ac.uk

Sets of similar components with defined variations
Random mutation
Genetic mutants
Cancerous tissue
Differentiation







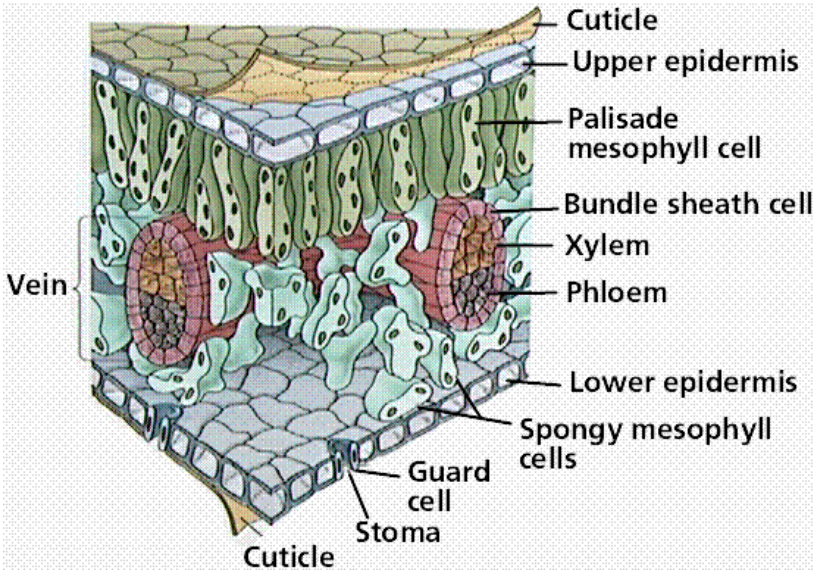


Spatial organisation

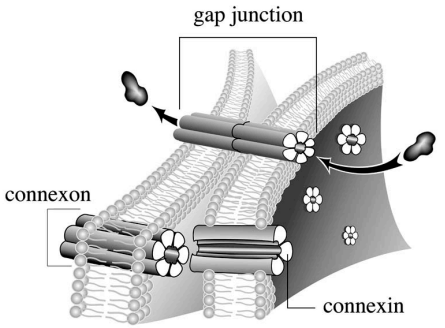
david.gilbert@brunel.ac.uk

Between cells

>how they are organised into regular or irregular patterns over spatial networks in one, two or three dimensions.

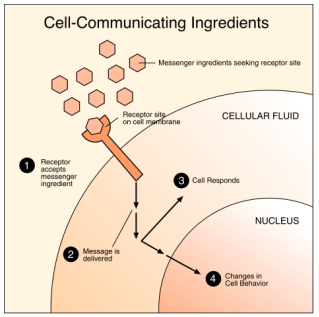


Communication



Between **immediate neighbours** (intracellular complexes)
Long-distance (cytokines etc)

Further constraints:
Type of **relationship** between partners
Type of component(s)
History of component(s)
Position of component(s) in spatial network.



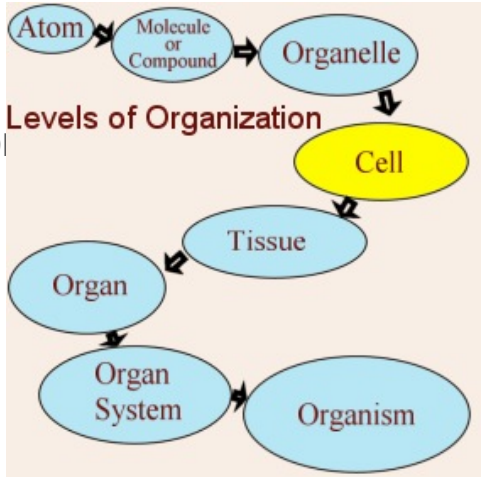
Hierarchical organisation

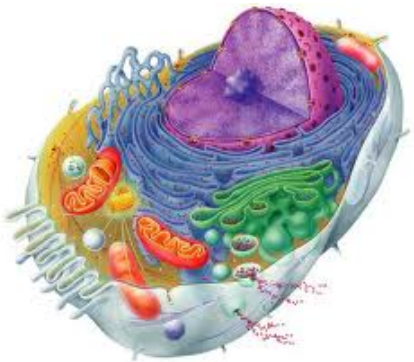
david.gilbert@brunel.ac.uk

Components containing repeated sub-components

Cell containing several compartments /components.

Enables the use of abstraction over level of detail used to describe co





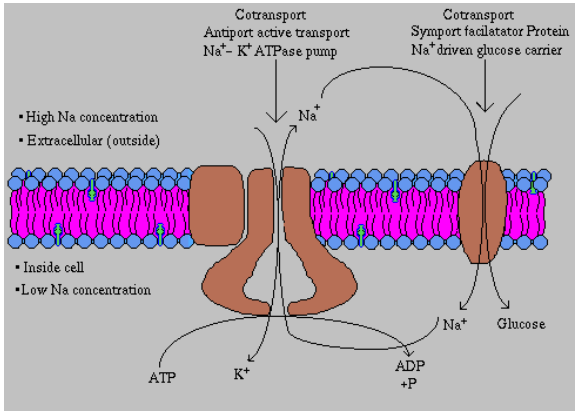
Movement

david.gilbert@brunel.ac.uk

Mobility – passive movement.

Protein transport

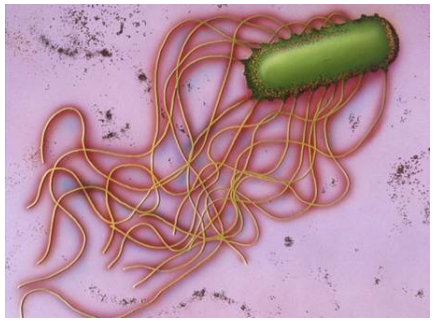
Sodium transport

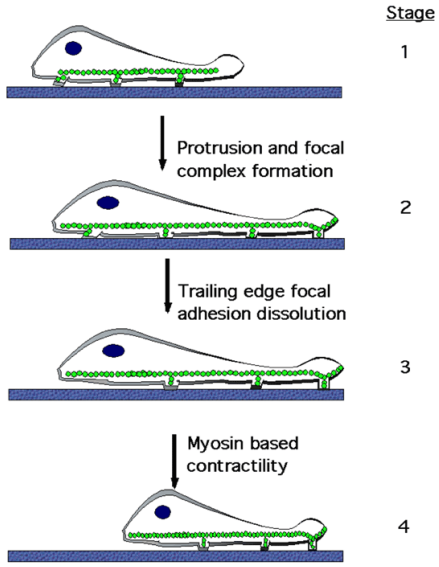


Motility – active movement.

Cells using organelles (flagellae)

General cellular motility



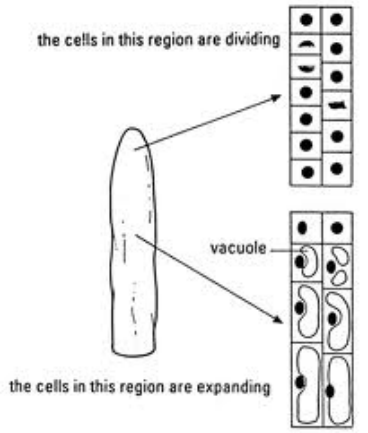


Replication

david.gilbert@brunel.ac.uk

E.g. cell division

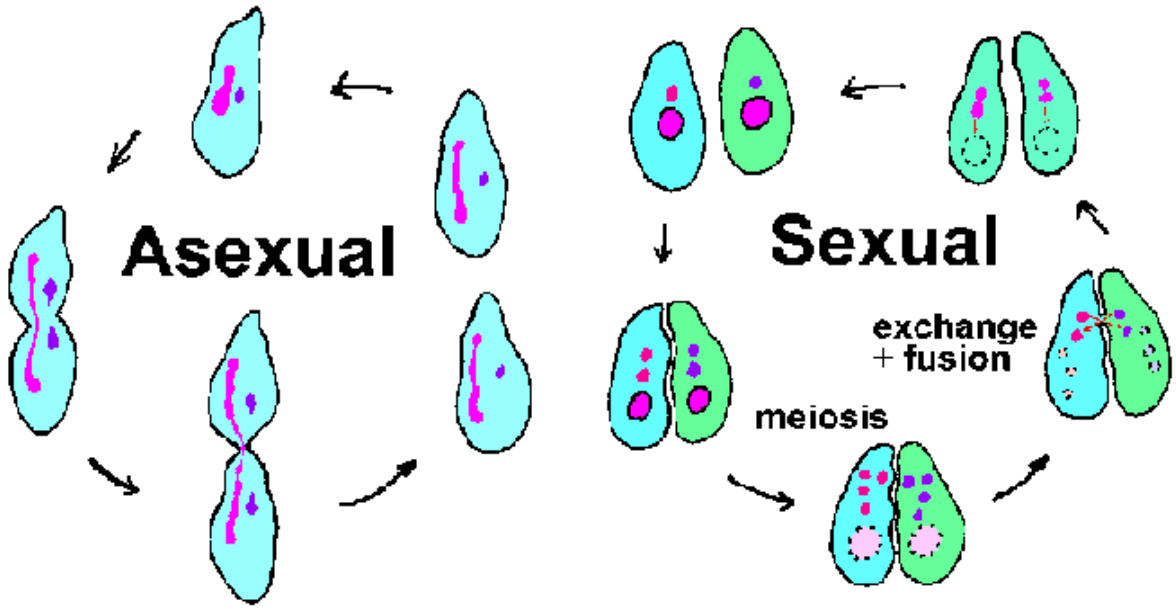
Can take into account:
Mutation
Spatial organisation / position



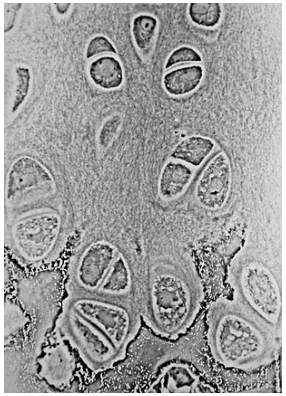
Exchange

david.gilbert@brunel.ac.uk

Exchange of (genetic) information
Sexual
Asexual



Death etc

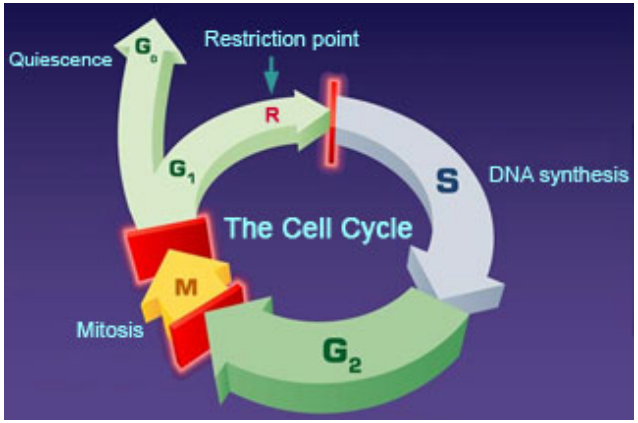


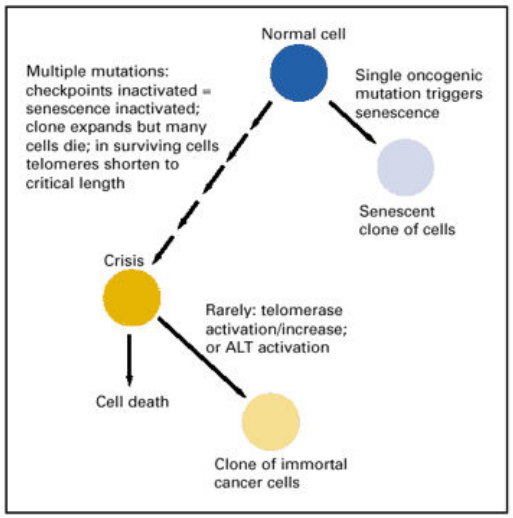
Cell death:

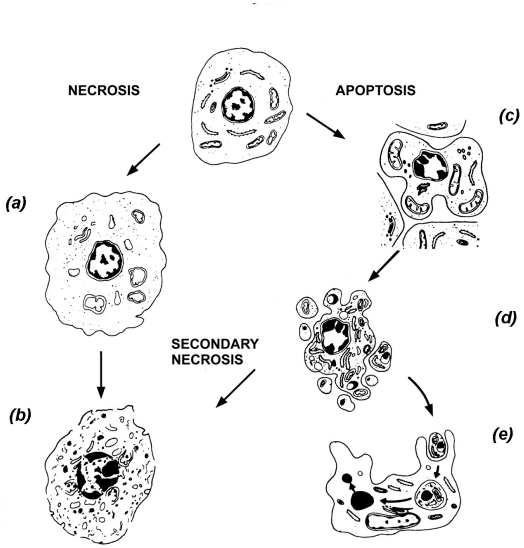
>apoptosis (programmed), necrosis (traumatic)

Quiescence

Senility







Coloured Petri Nets

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Permit
Repeated elements
Discrimination of species (molecules, metabolites, proteins, secondary substances, genes, etc.).
Locality: distinguish between sub-populations of a species in different locations (cytosol, nucleus and so on).

D. Gilbert, M. Heiner, F. Liu and N. Saunders (2013): Colouring Space - A Coloured Framework for Spatial Modelling in Systems Biology. In Proc. PETRI NETS 2013. Springer LNCS, 7927, 230-249.

Mapping space in colour

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Discrete approach

1D : natural numbers Nat [1, 2, 3, … Max]

2D : tuples of Nat [(1,1), (1,2), …. (2,1)… (Max_x, Max_y)]

3D : triples of Nat [(1,1,1), …(Max_x, Max_y, Max_z)]

D. Gilbert, M. Heiner, F. Liu and N. Saunders (2013): **Colouring Space - A Coloured Framework for Spatial Modelling in Systems Biology. In Proc. PETRI NETS 2013. Springer LNCS, volume 7927, pages 230-249, June 2013**

Mapping space in colour

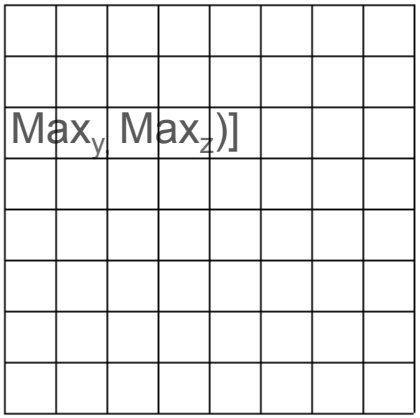
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Discrete approach

1D : natural numbers [1, 2, 3, … Max]

2D : tuples of Nat [(1,1), (1,2), …. (2,1)… (Max_x, Max_y)]

3D : triples of Nat [(1,1,1), …(Max_x, Max_y, Max_z)]



D. Gilbert, M. Heiner, F. Liu and N. Saunders (2013): **Colouring Space - A Coloured Framework for Spatial Modelling in Systems Biology. In Proc. PETRI NETS 2013. Springer LNCS, volume 7927, pages 230-249, June 2013**

Mapping space in colour

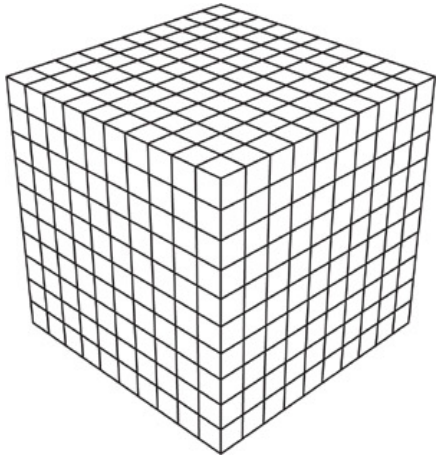
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Discrete approach

1D : natural numbers [1, 2, 3, … Max]

2D : tuples of Nat [(1,1), (1,2), …. (2,1)… (Max_x, Max_y)]

3D : triples of Nat [(1,1,1), …(Max_x, Max_y, Max_z)]

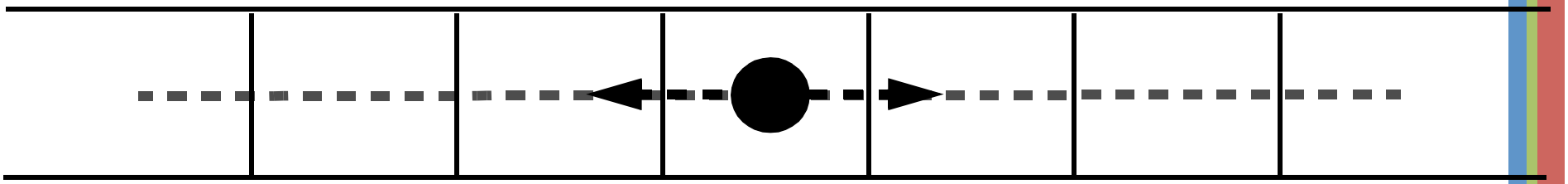


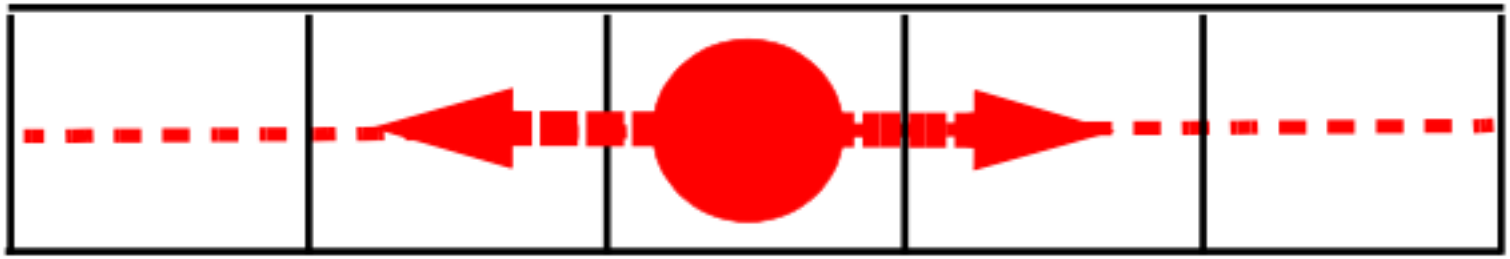
D. Gilbert, M. Heiner, F. Liu and N. Saunders (2013): **Colouring Space - A Coloured Framework for Spatial Modelling in Systems Biology. In Proc. PETRI NETS 2013. Springer LNCS, volume 7927, pages 230-249, June 2013**

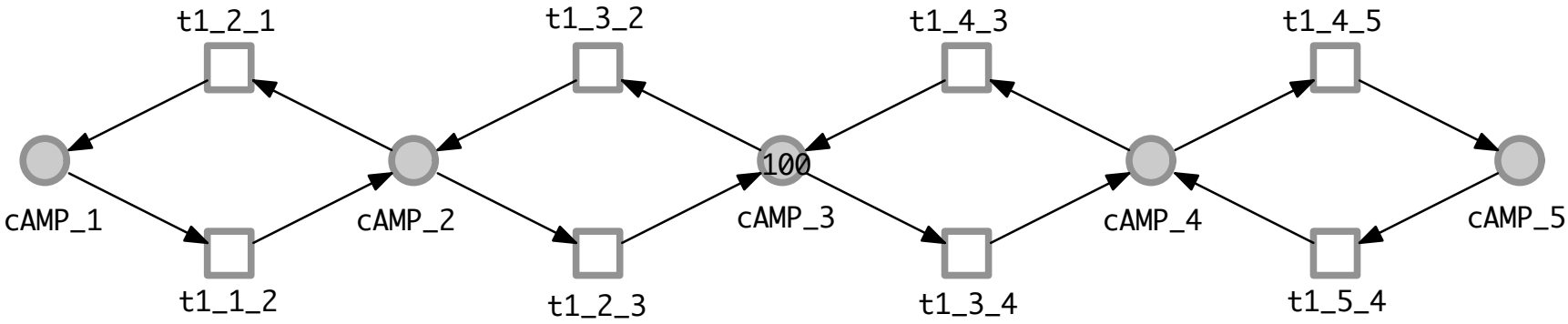
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Diffusion example: 1-D

{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk







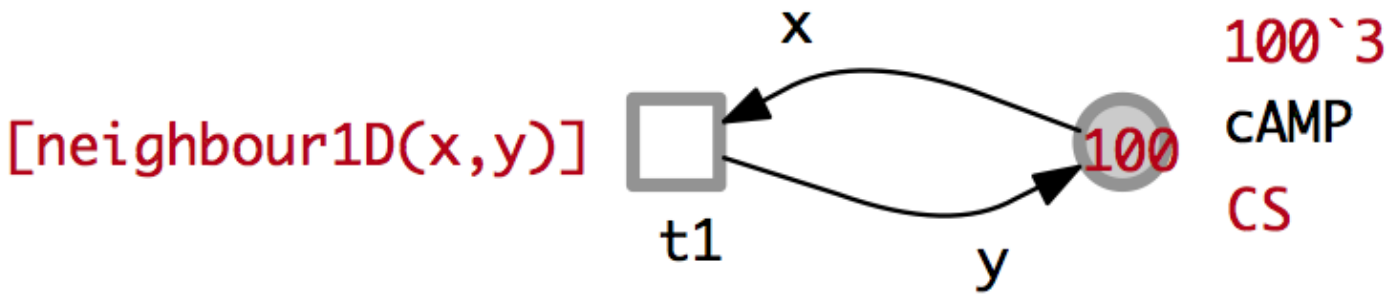
1D

{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk

- **definitions**

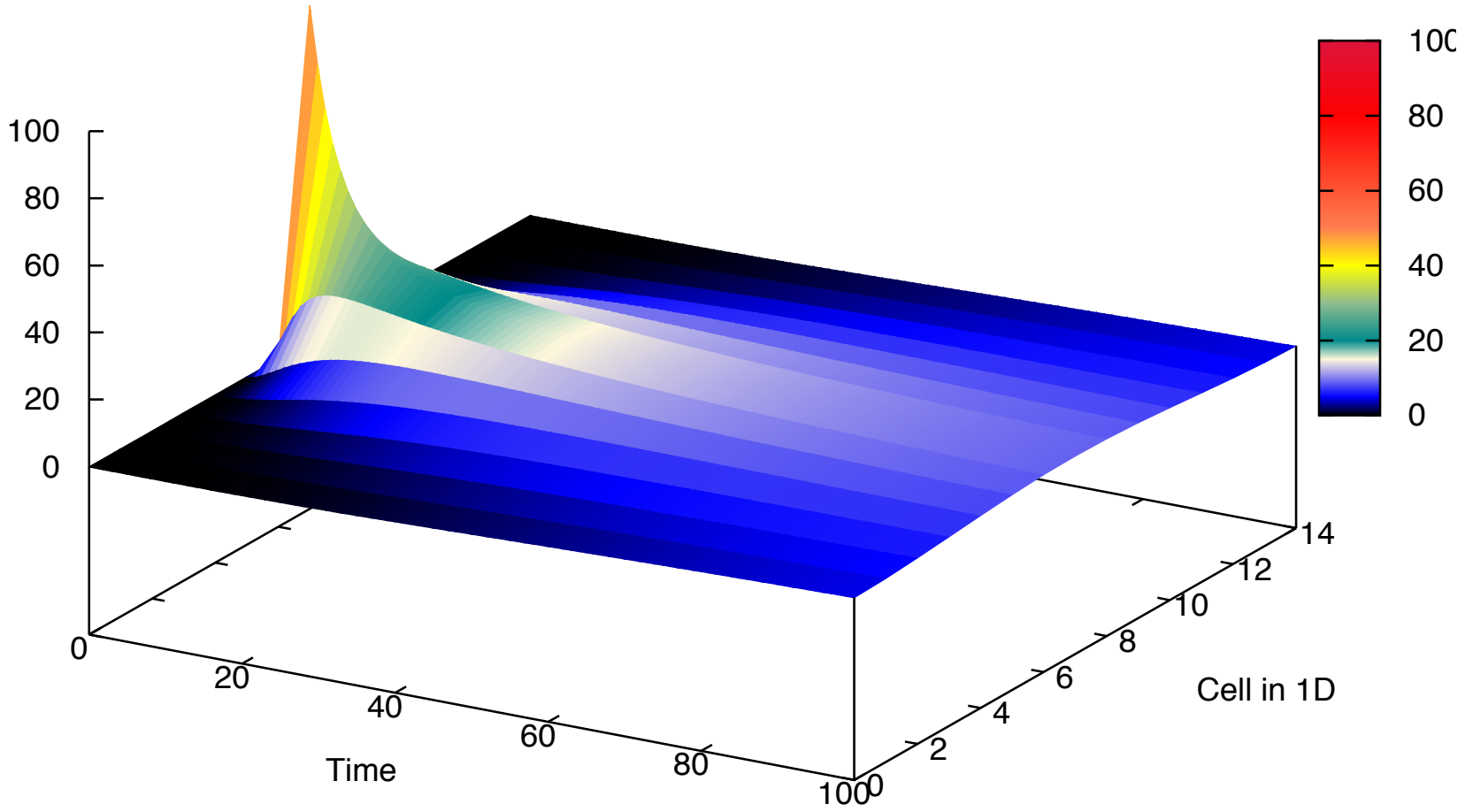
```
const D1 = 5;              // grid size
colorset CS = 1-D1;        // grid positions
var x,y : CS;

function neighbour1D (CS x,a) bool:
  // a is neighbour of x
  ( a=x-1 | a=x+1)  & (1<=a) & (a<=D1);
```

Easy to change spatial resolution



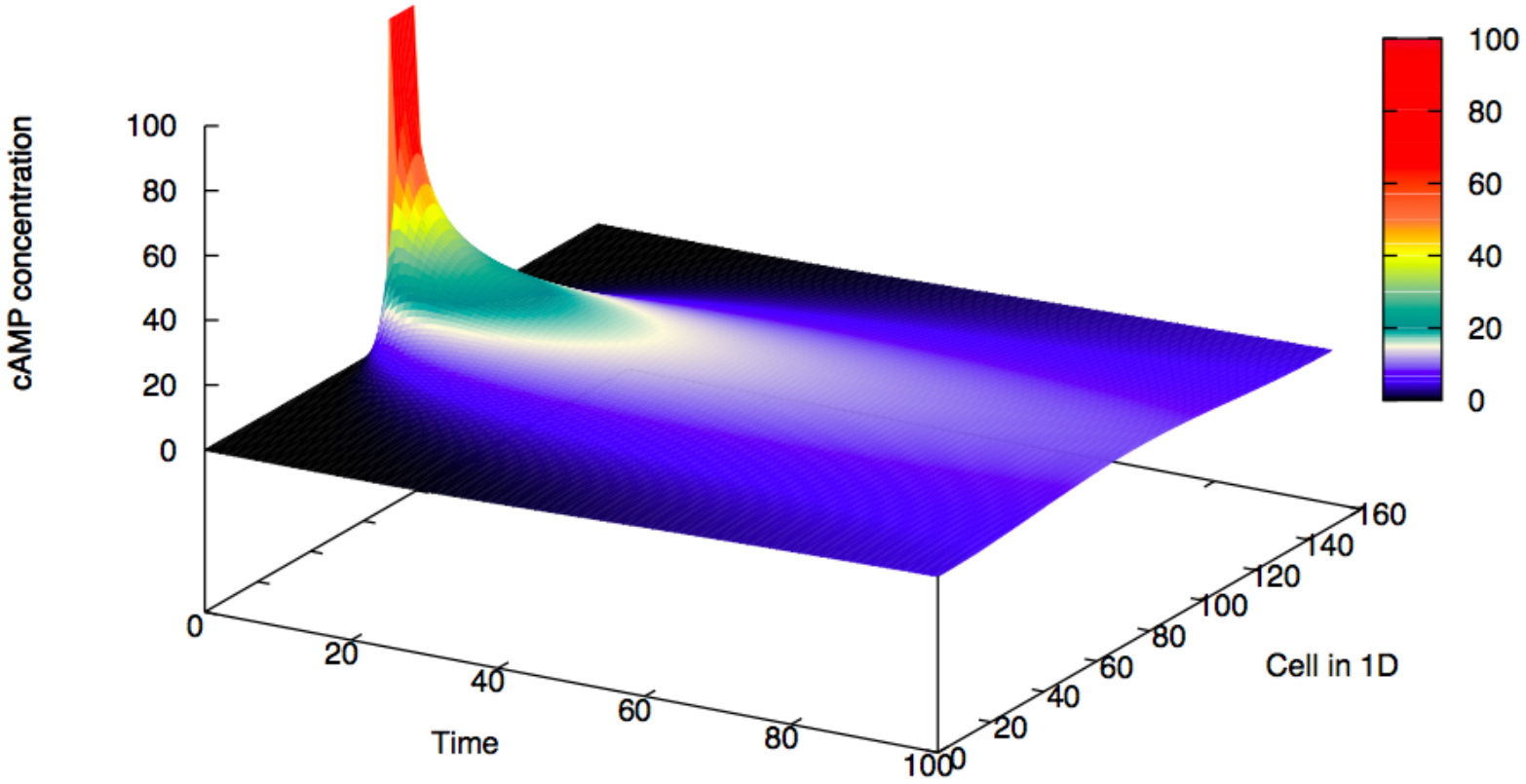
1D, 15 grid positions

{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk



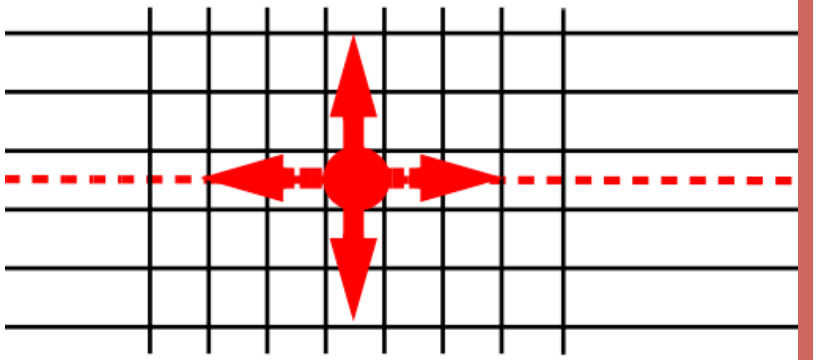
1D, 150 grid positions, scaling

{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk



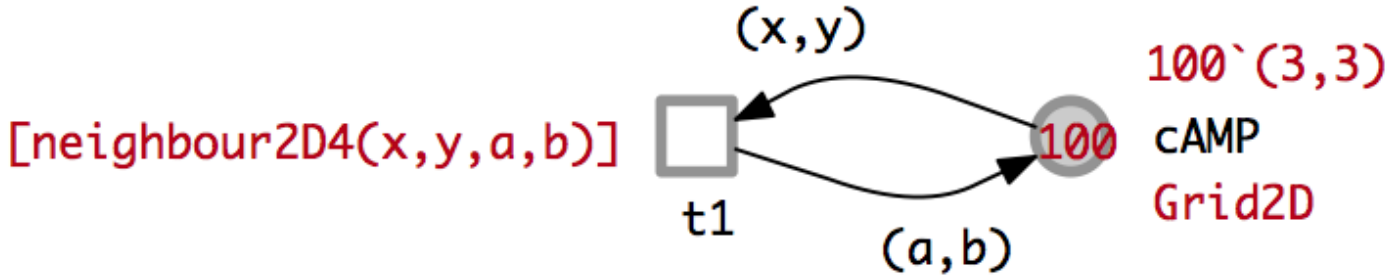
150 GRID POSITIONS, SCALING OF INITIAL MARKING AND RATES

2D (4)

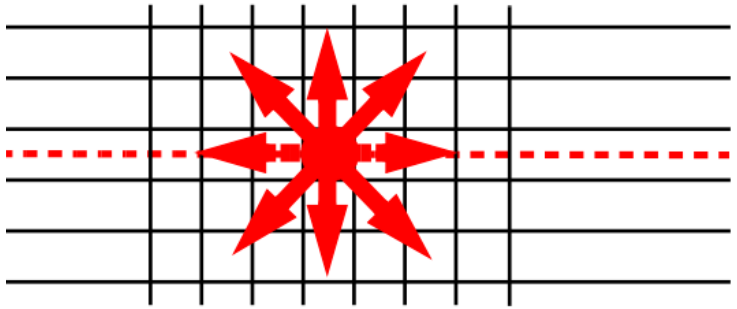


- **four neighbours**

```
function neighbour2D4 (CD1 x, CD2 y, CD1 a, CD2 b) bool:
  // (a,b) is one of the up to four neighbours of (x,y)
     (a=x & b=y-1) | (a=x & b=y+1)
  |  (b=y & a=x-1) | (b=y & a=x+1);
```

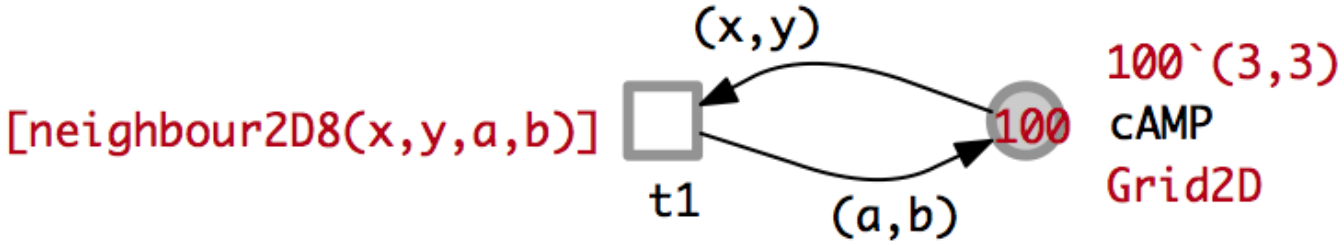


2D (8)



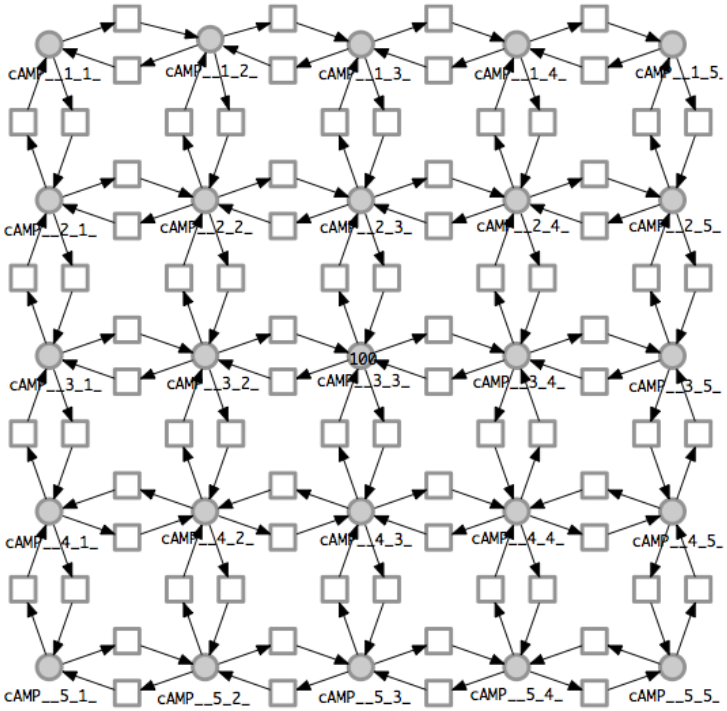
- **eight neighbours**

```
function neighbour2D8 (CD1 x, CD2 y, CD1 a, CD2 b) bool:
  // (a,b) is one of the up to eight neighbours of (x,y)
    (a=x-1 | a=x | a=x+1) & (b = y-1 | b=y | b=y+1)
  & (!(a=x & b=y))
  & (1<=a & a<=D1) & (1<=b & b<=D2);
```

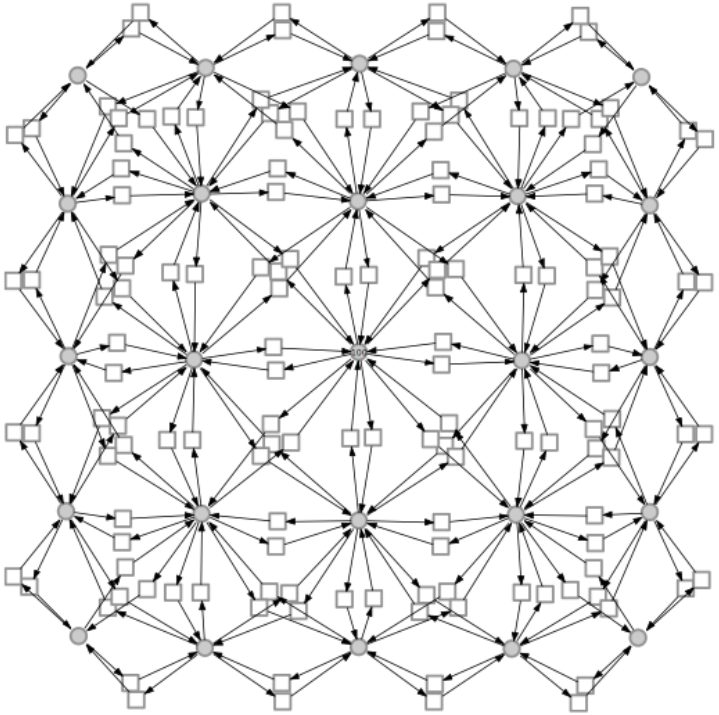


Unfolding…

{david.gilbert,ovidiu.parvu}@brunel.ac.uk



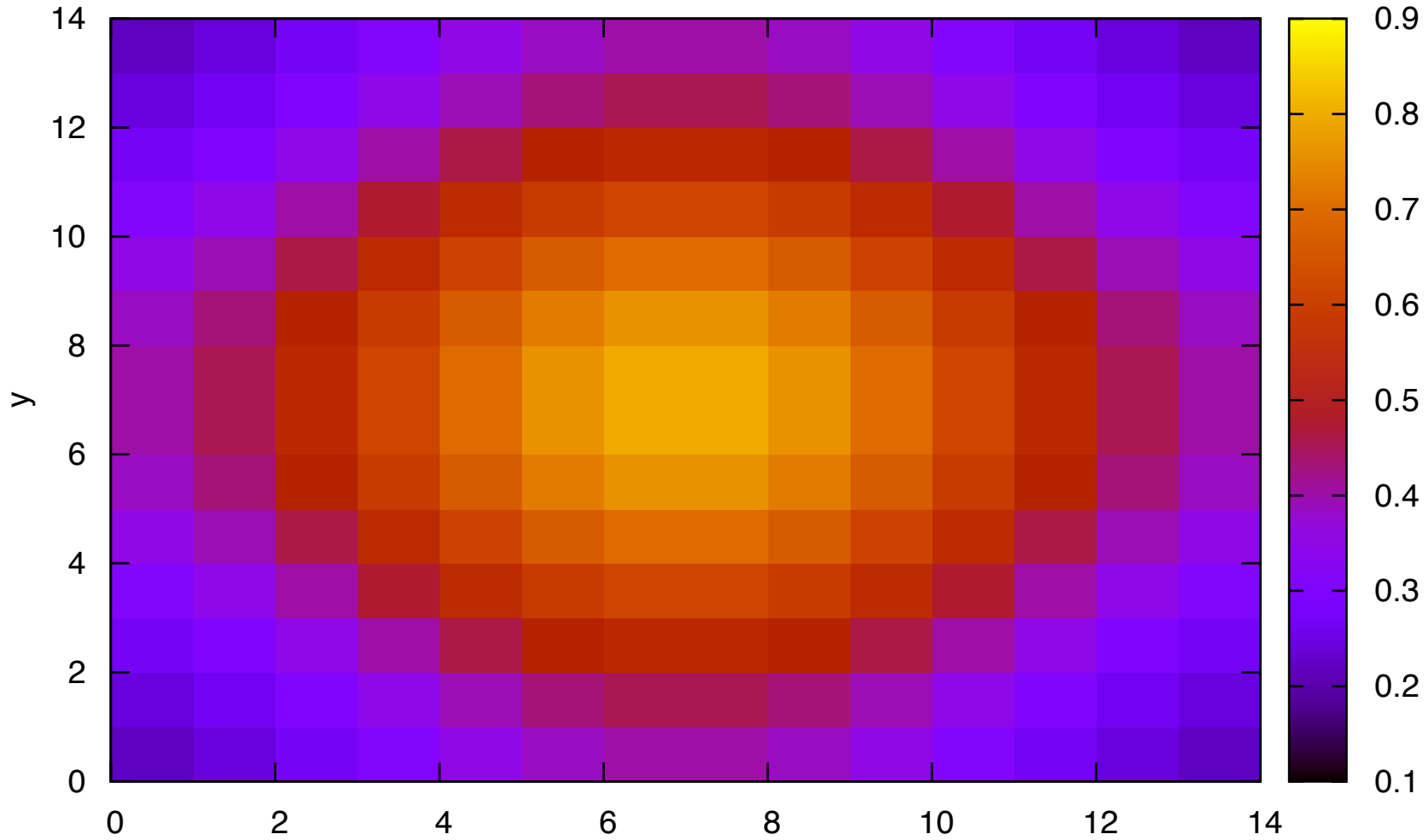
2D4



2D8

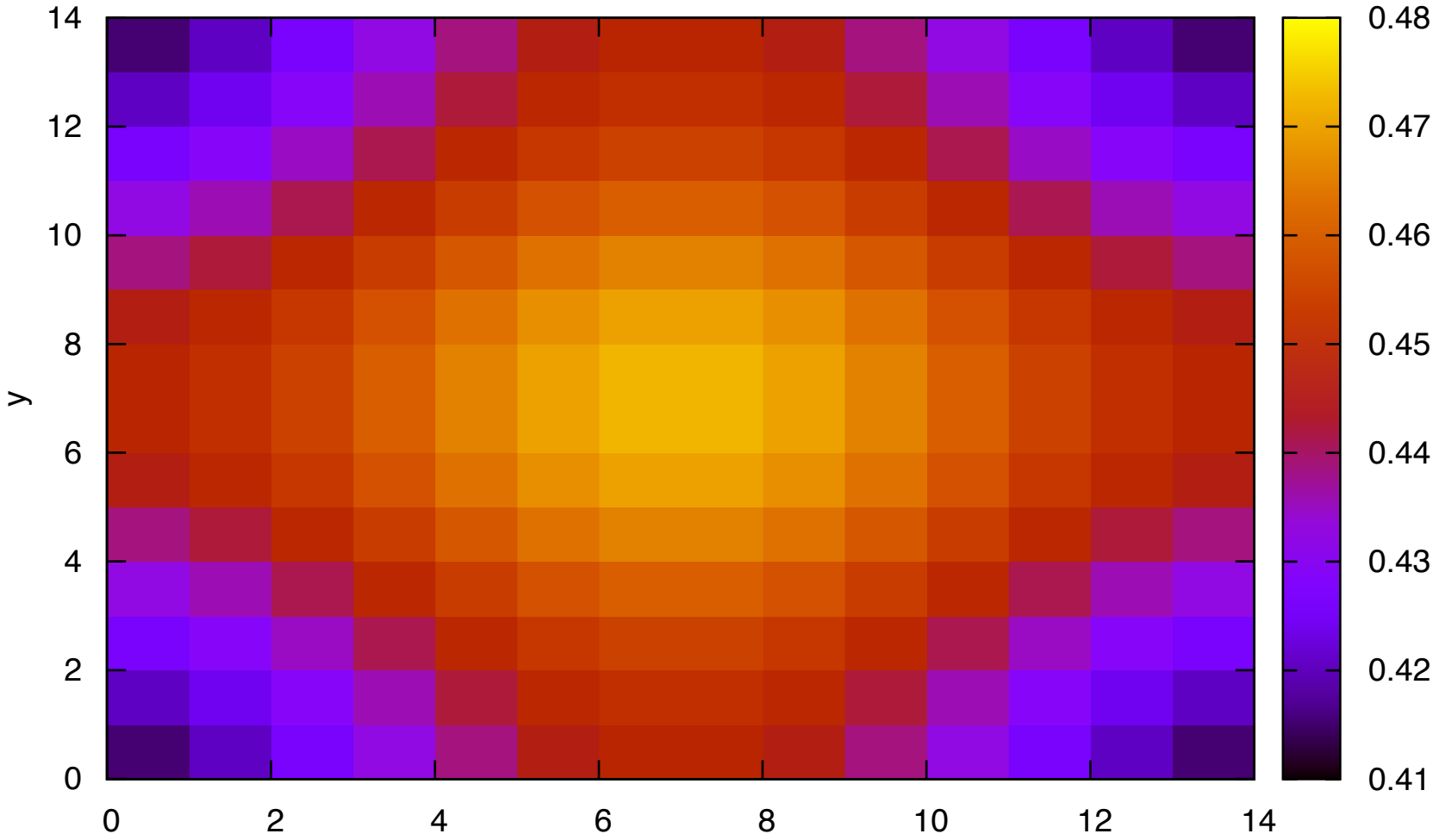
2D (4) gradient

{david.gilbert,ovidiu.parvu}@brunel.ac.uk



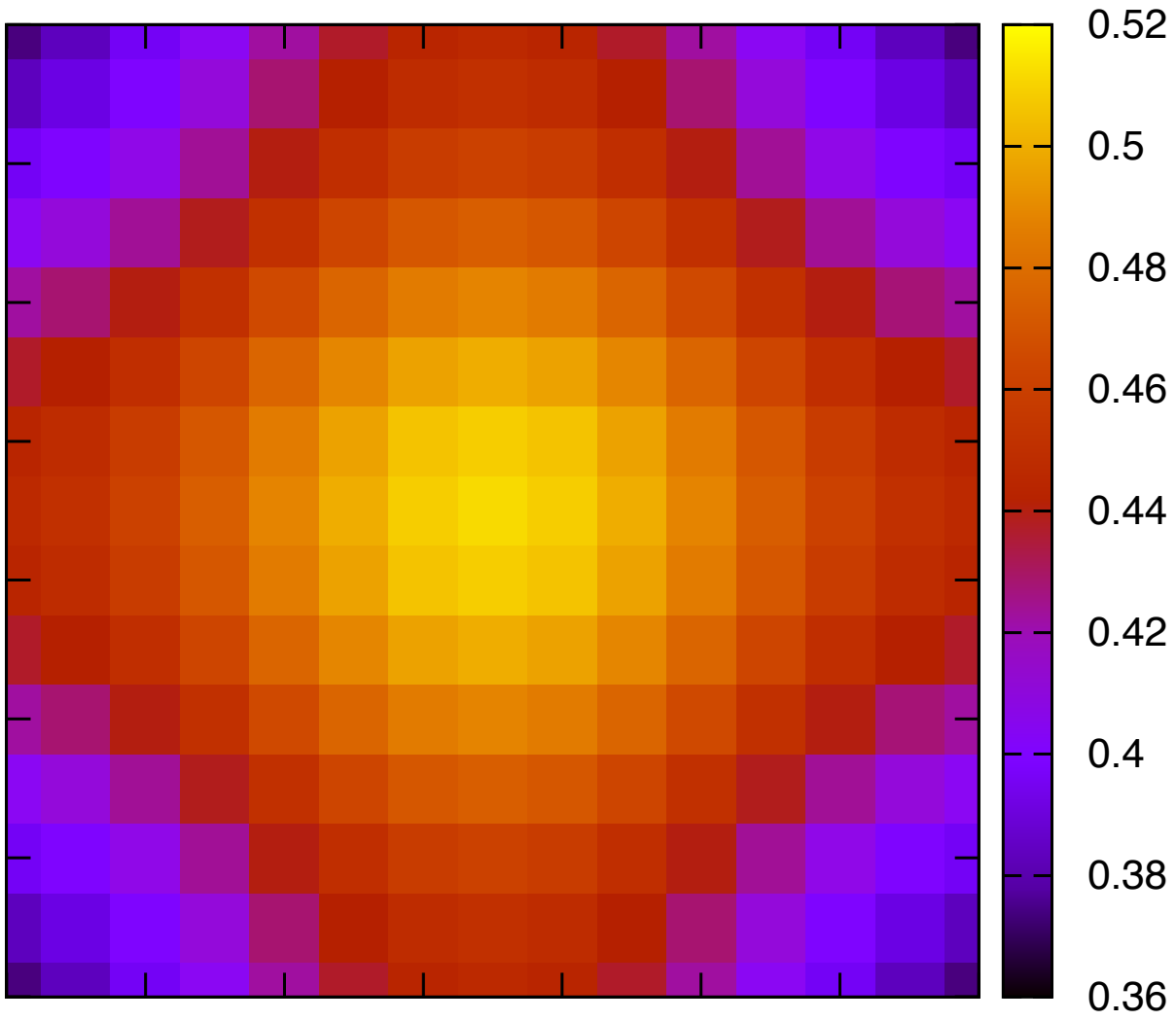
2D (8) gradient

{david.gilbert,ovidiu.parvu}@brunel.ac.uk



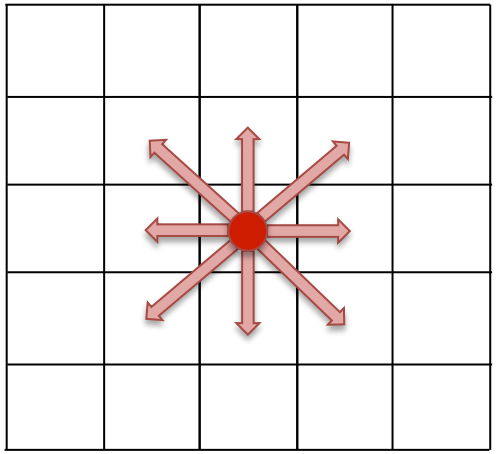
2D (8) gradient, higher resolution

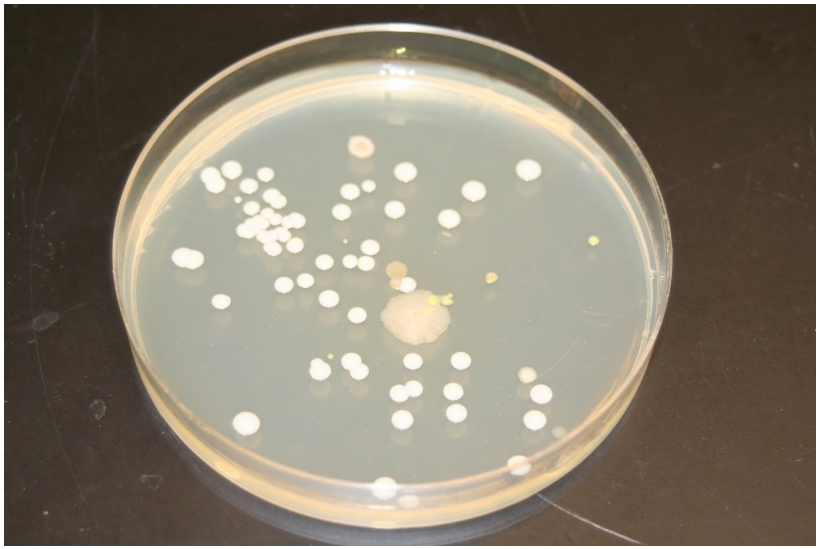
{david.gilbert,ovidiu.parvu}@brunel.ac.uk



{david.gilbert,ovidiu.parvu}@brunel.ac.uk

•**Cell division**: process by which a *parent cell* divides into two or more *daughter cells*.





Bacterial colony

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Specification

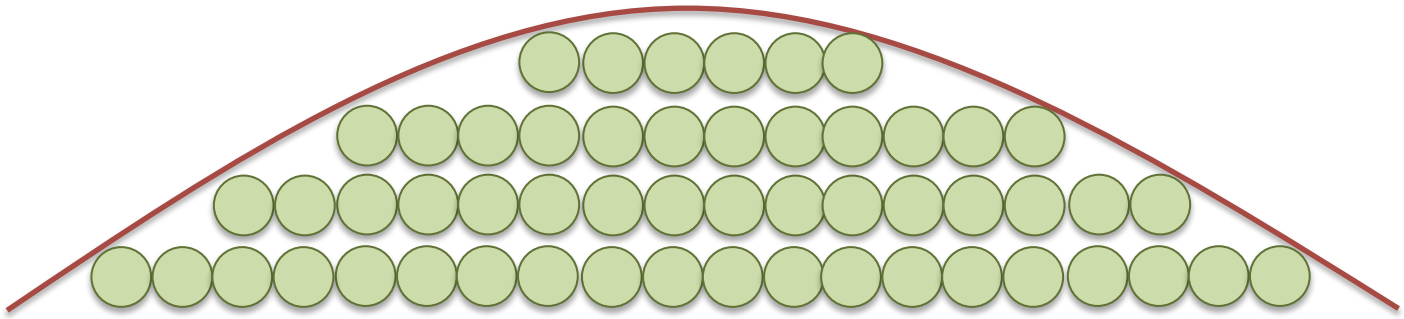
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

- Starts with only **one** cell or organism
- **Dimensions** of the environment are fixed
- Division of the cells/organisms is **random/ stochastic**
- A **maximum capacity** or volume for each position in the grid

Colony profile

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

•The **maximum height** of the colony of bacteria is limited due to:

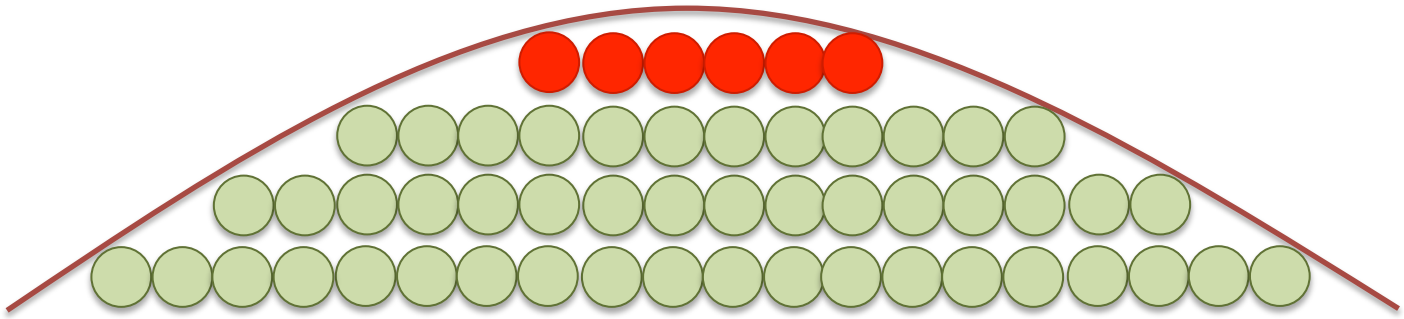


Colony profile

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

- The **maximum height** of the colony of bacteria is limited due to:

1. Lack of **nutrients** for the bacteria at the top

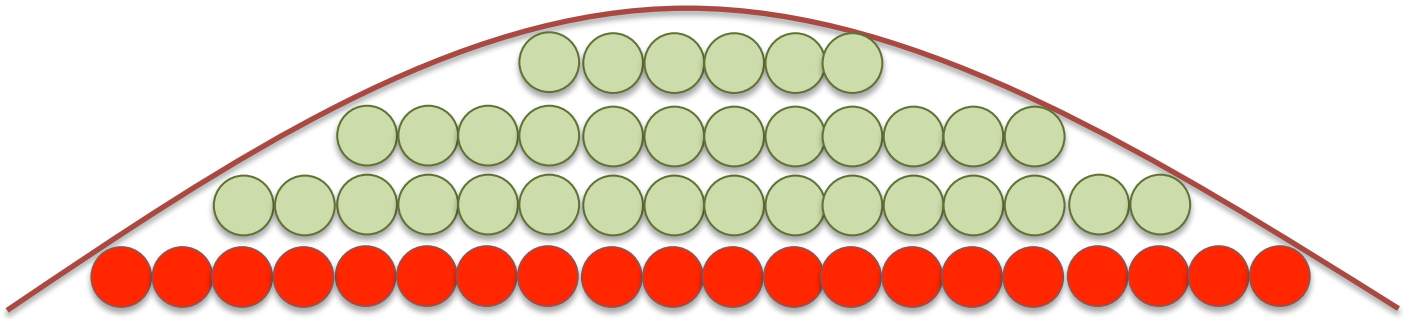


Colony profile

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

•The **maximum height** of the colony of bacteria is limited due to:

1. Lack of **nutrients** for the bacteria at the top
2. Lack of **oxygen** for the bacteria at the bottom



Specification

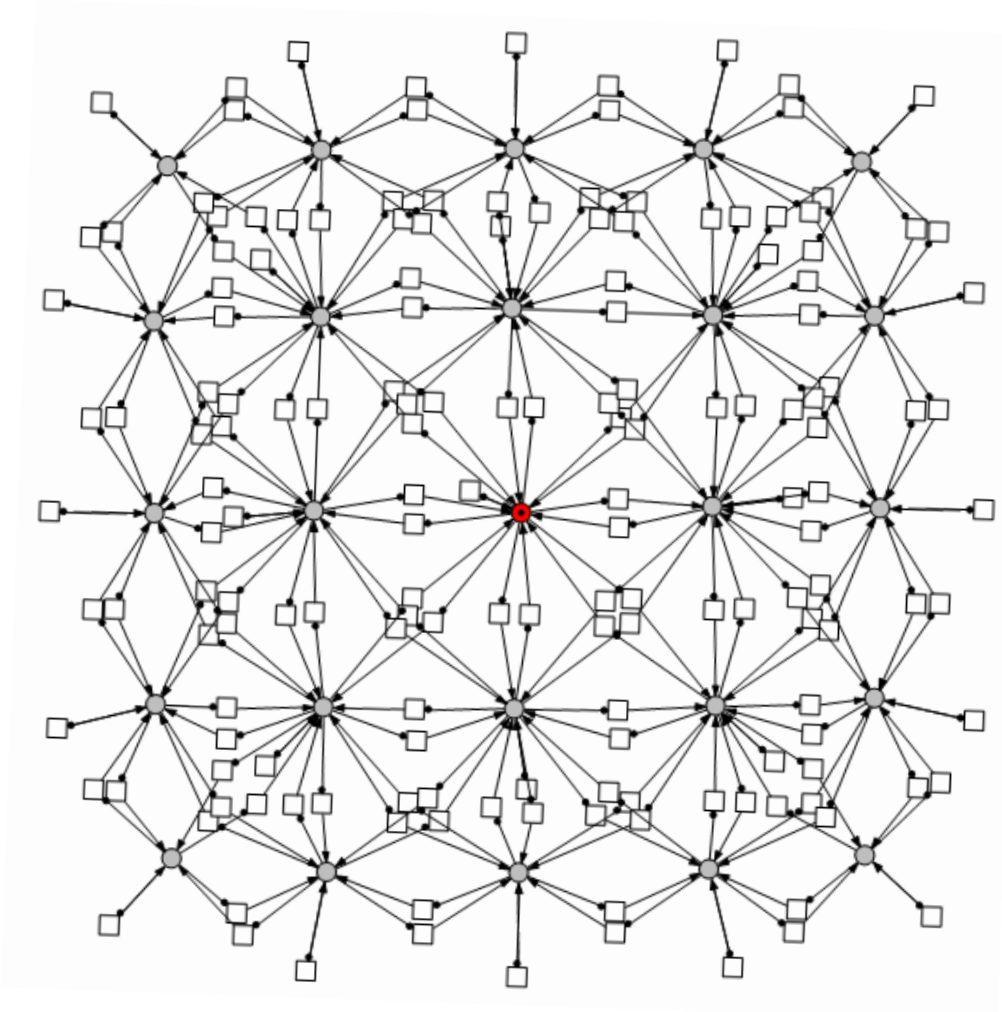
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

- Starts with only **one** cell or organism
- **Dimensions** of the environment are fixed
- Division of the cells/organisms is **random/ stochastic**
- A **maximum capacity** or volume for each position in the grid
- Whenever cellular division occurs the **parent cell** preserves its position, **offspring cell** may be (probabilistically) displaced to neighbouring positions

Constructing the model

{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk

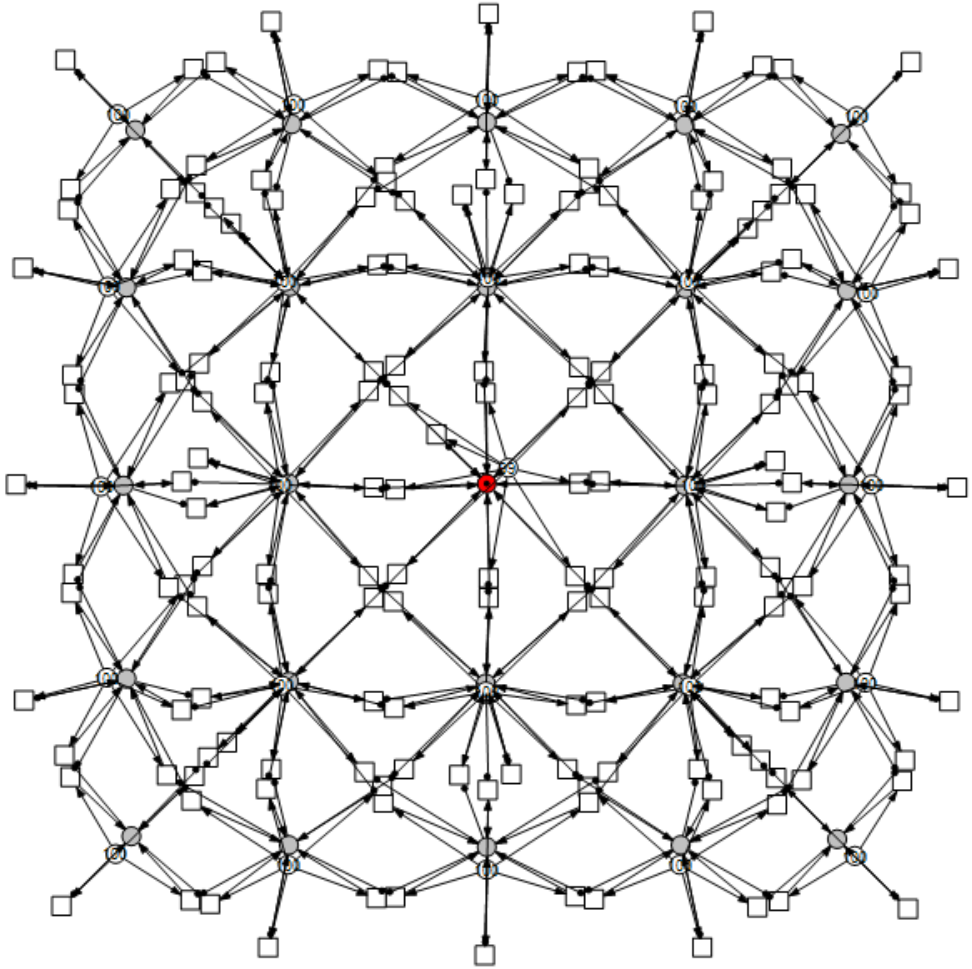
1.Creating the **basic** model for growth



Constructing the model

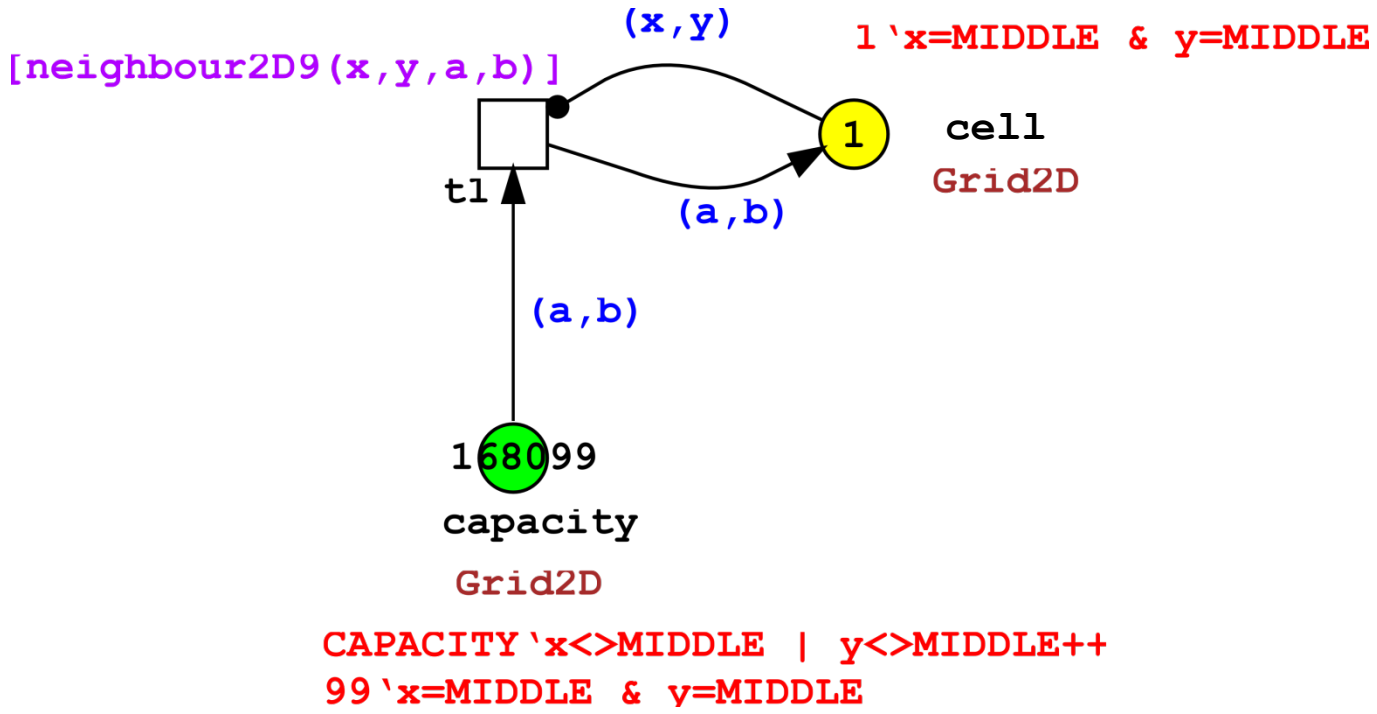
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

1. Creating the **basic** model for growth
2. Adding the **capacity/ maximum height** constraint



Constructing the model

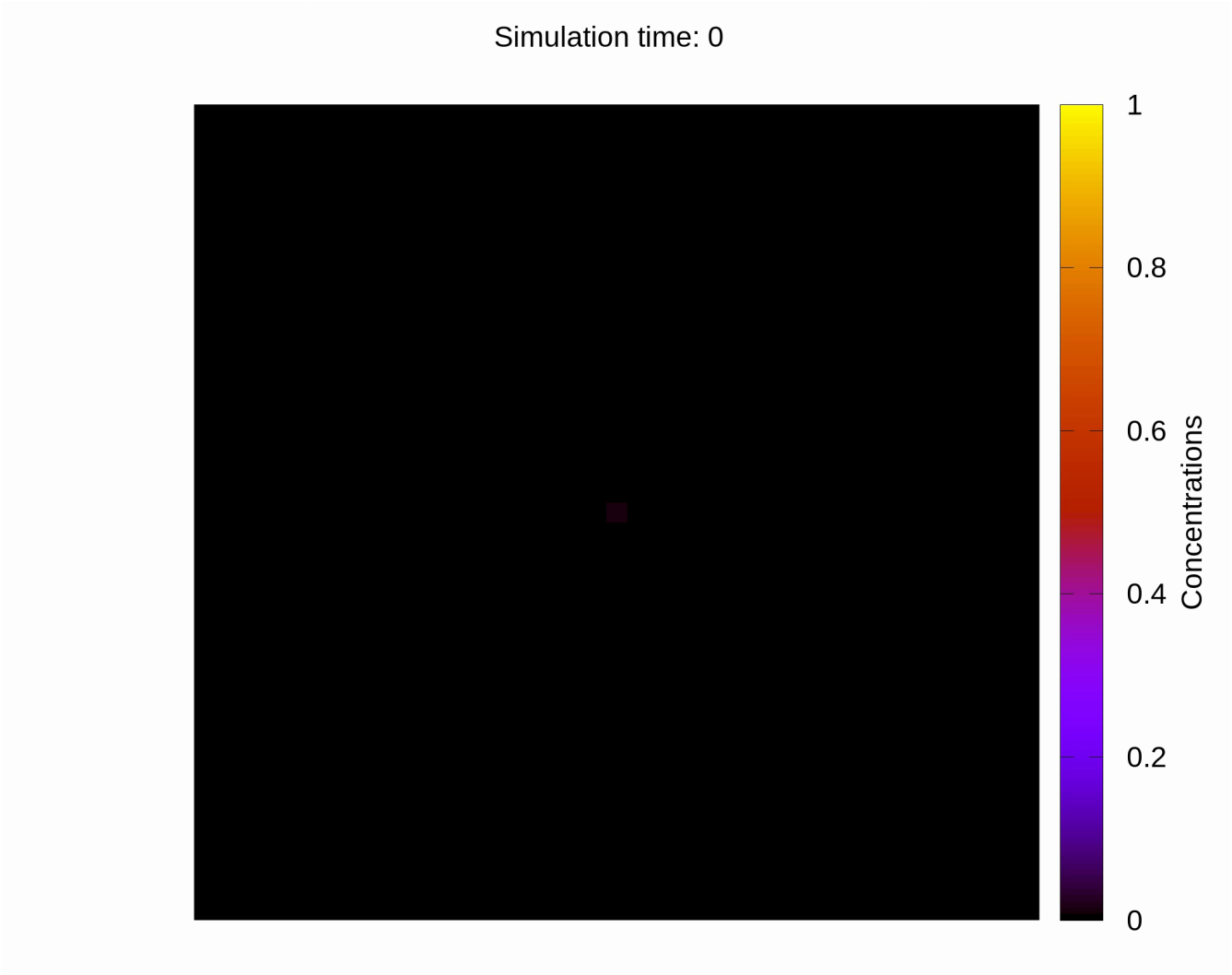
{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk



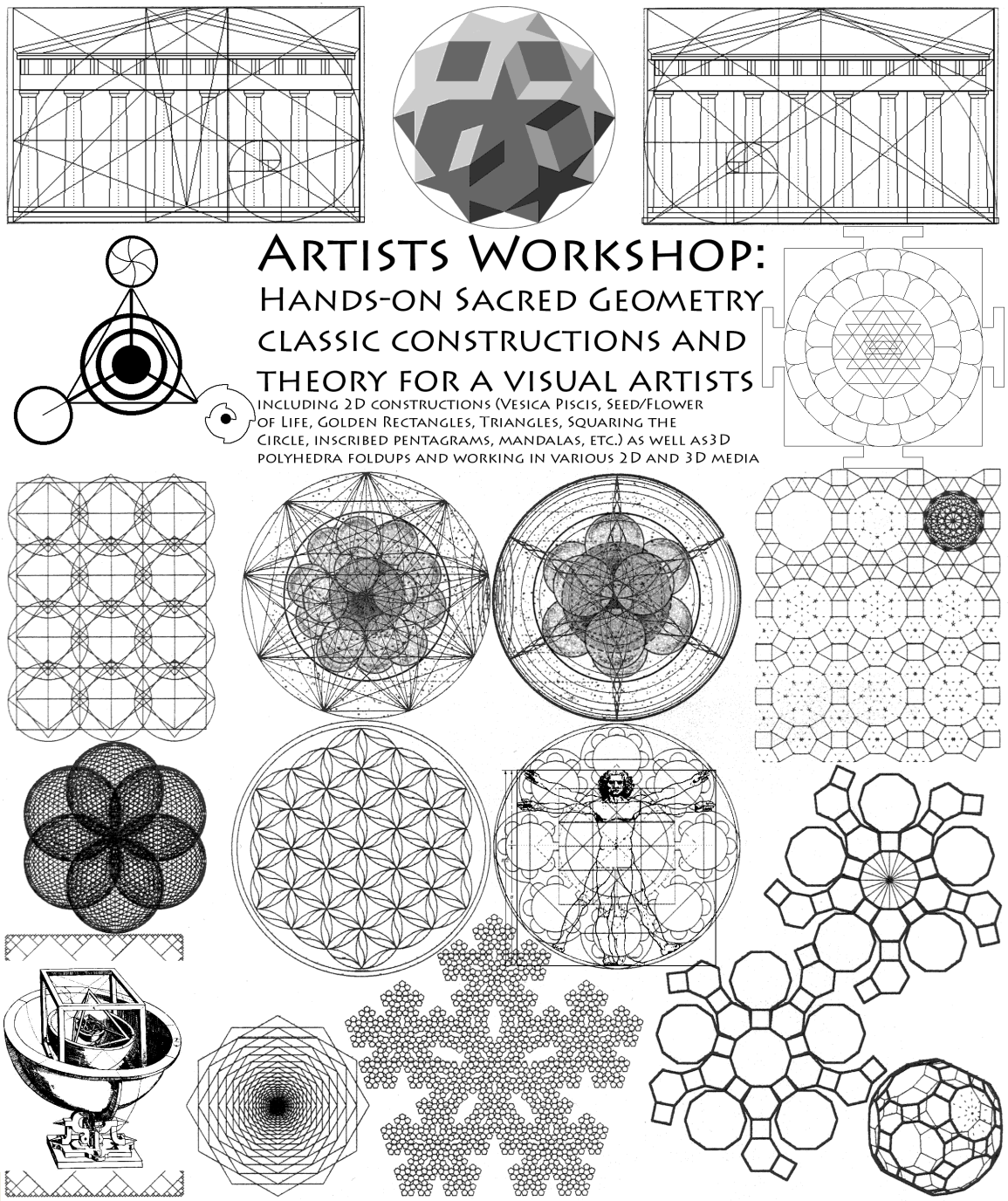
24 hours growth: ~26-27 generations, ~60m cells

Simulating the growth model

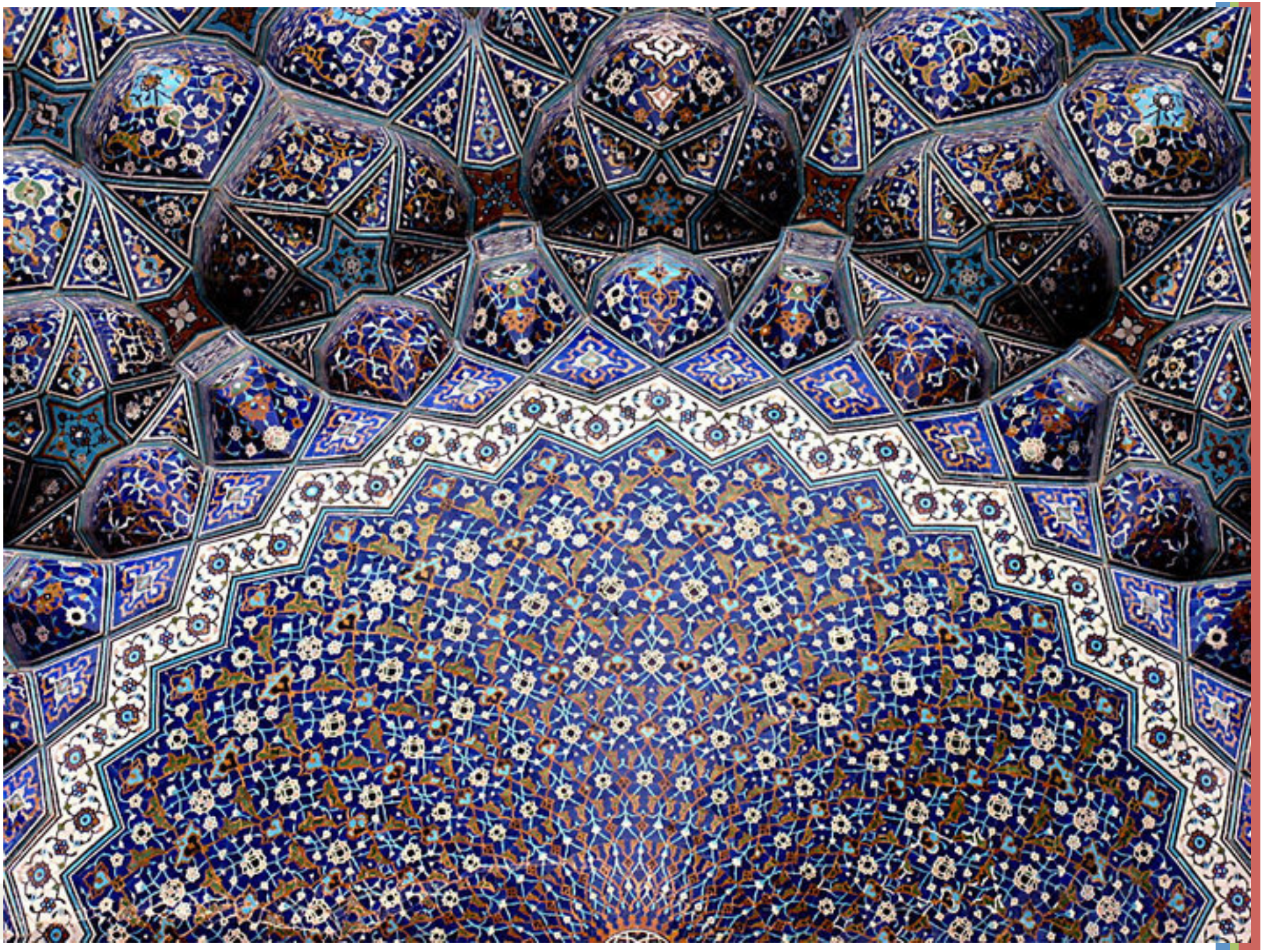
{david.gilbert,ovidiu.parvu}@brunel.ac.uk



{david.gilbert,ovidiu.parvu}@brunel.ac.uk

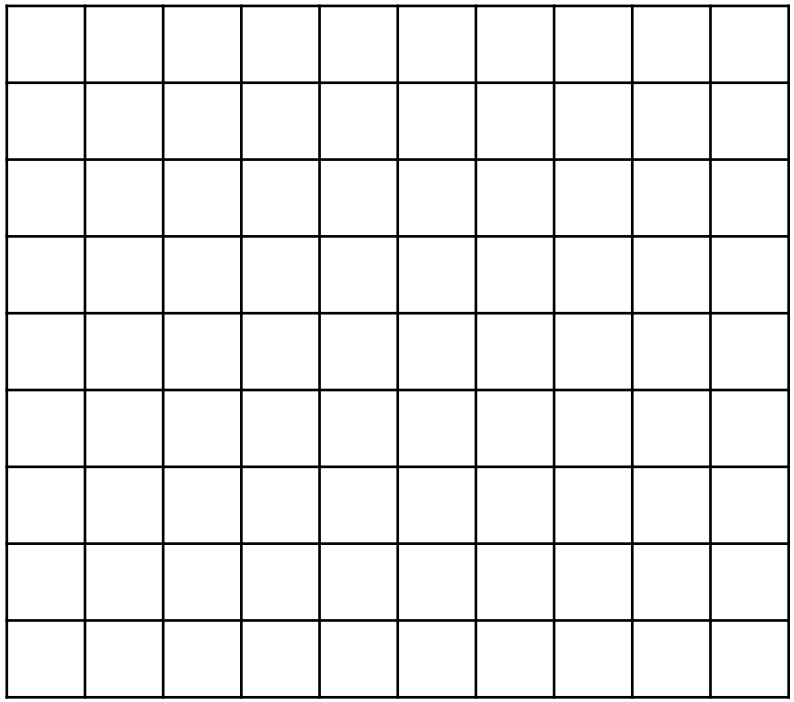


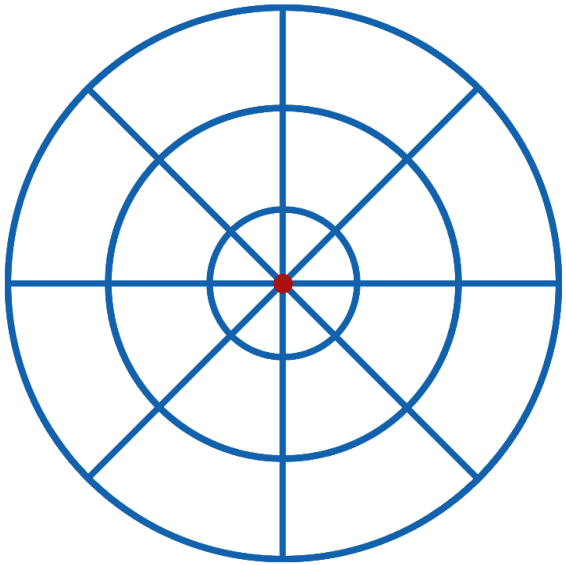
{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk



Rectangular vs circular geometry

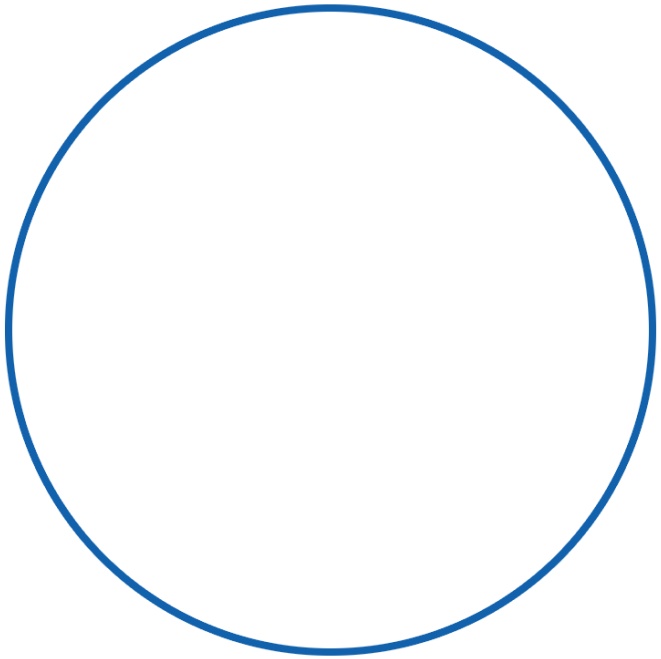
{david.gilbert,ovidiu.parvu}@brunel.ac.uk





Discretising the polar space

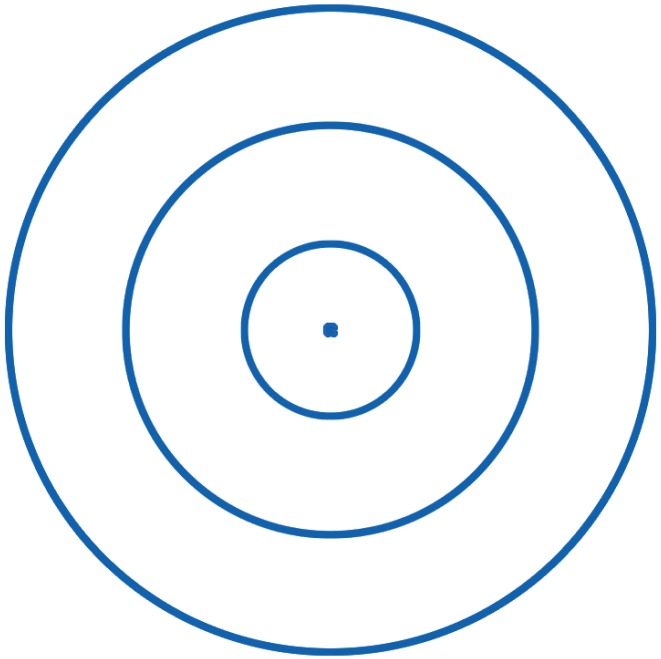
{david.gilbert,ovidiu.parvu}@brunel.ac.uk



Split the initial circle in ***N*** annuli…

Discretising the polar space

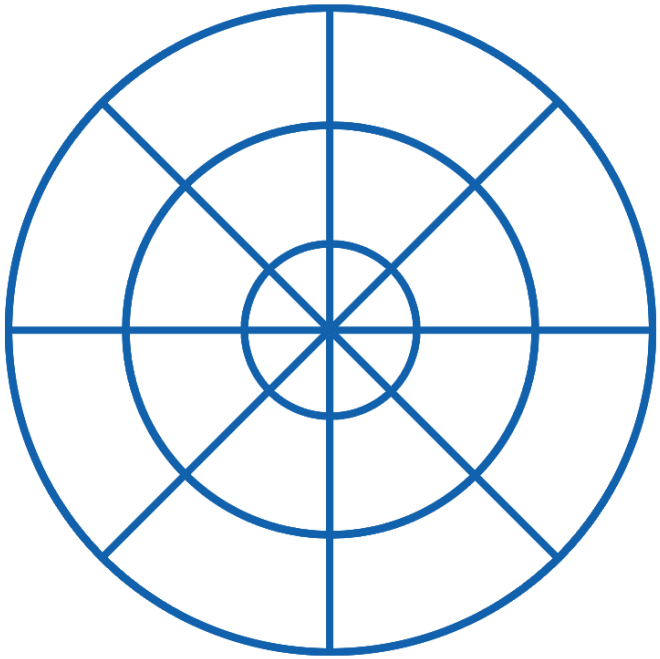
{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk



… and split each annuli in ***M*** sectors...

Discretising the polar space

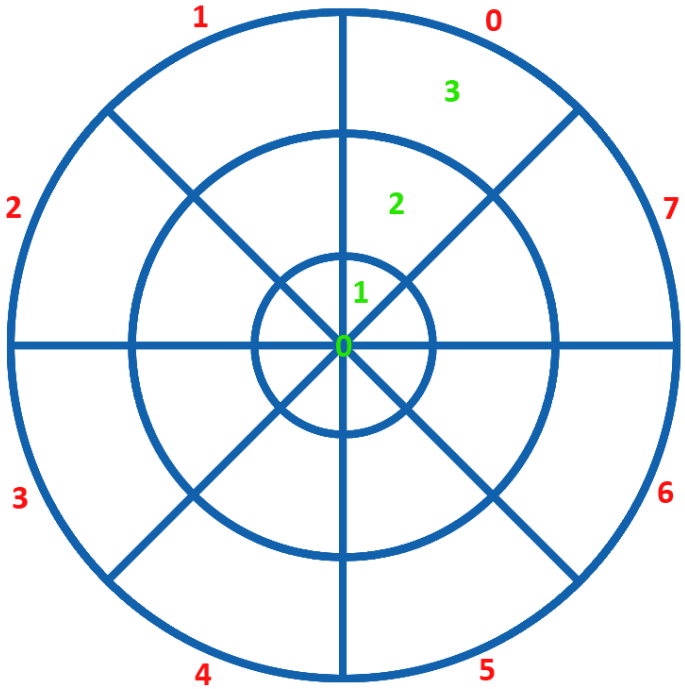
{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk

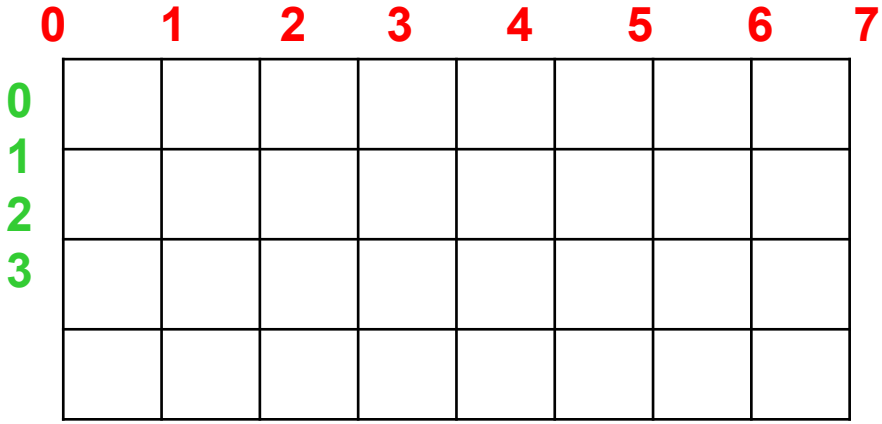


… obtaining ***N x M*** annular sectors.

Mapping polar space to a matrix

{david.gilbert,ovidiu.parvu}@brunel.ac.uk





Remark: Only 1 cell from row 0 will contain a value and this value will represent the concentration of the circle of radius 0.

Details missing

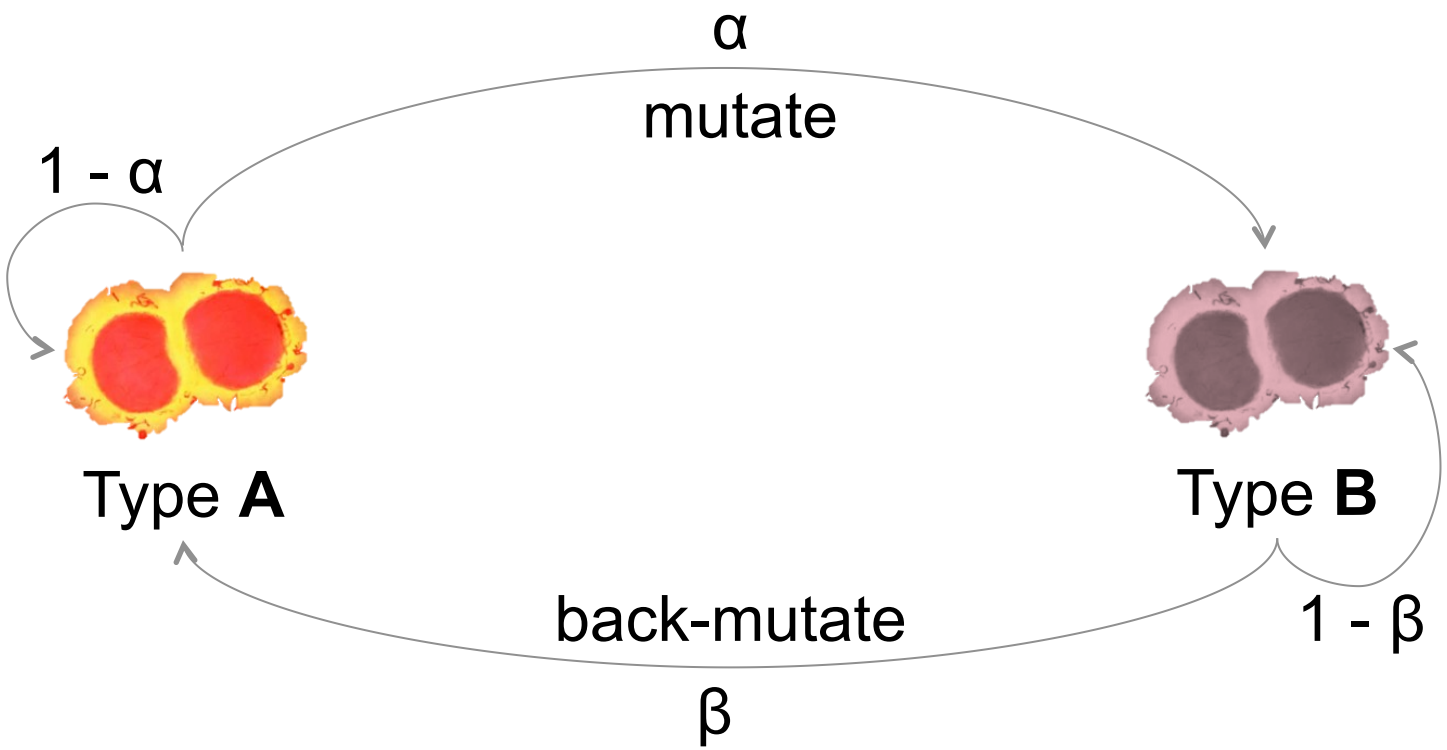
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Ask us for them….

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Alternative geometries: Phase Variation

Stochastic gene expression switching mechanism

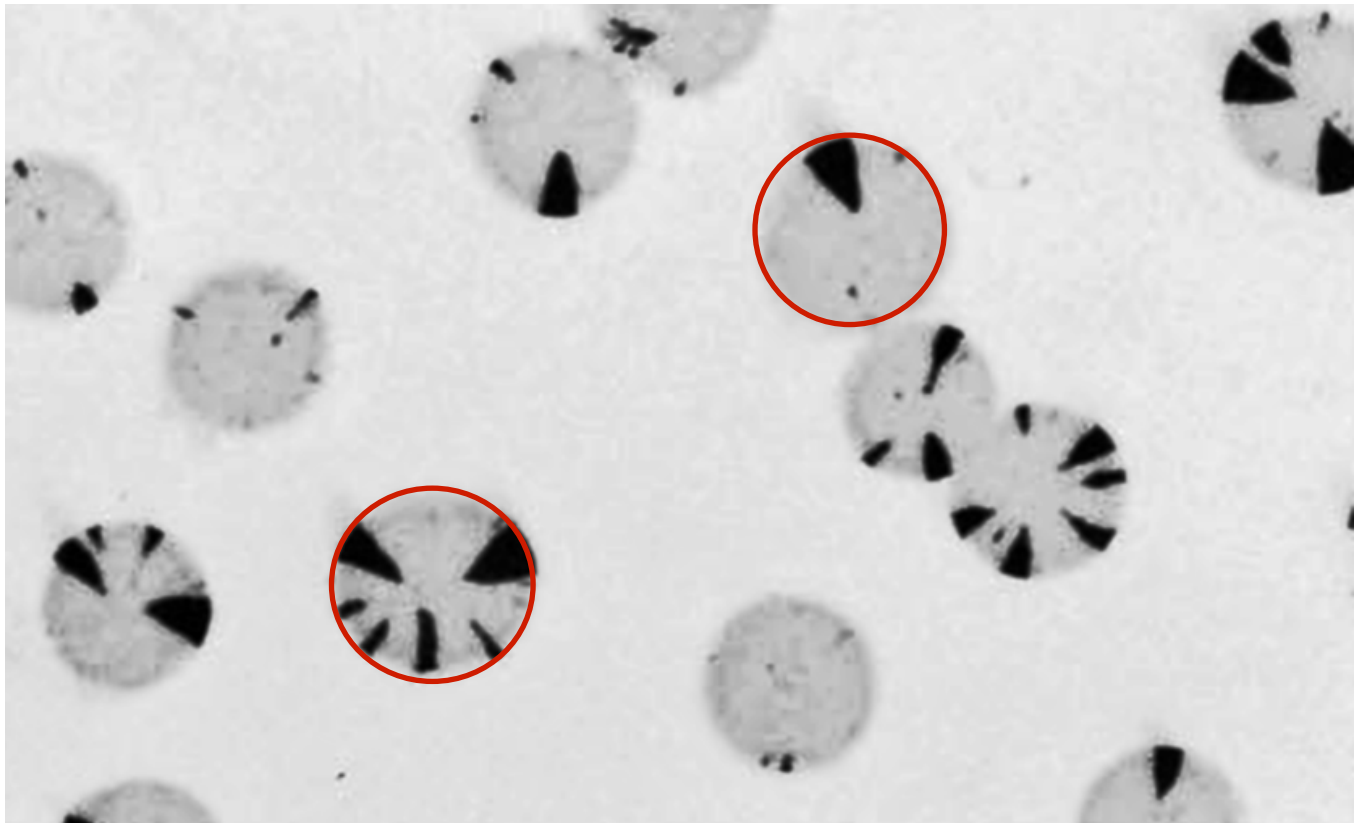


… where α and β are called the **mutation rates** of bacteria.

Alternative geometries: Phase Variation

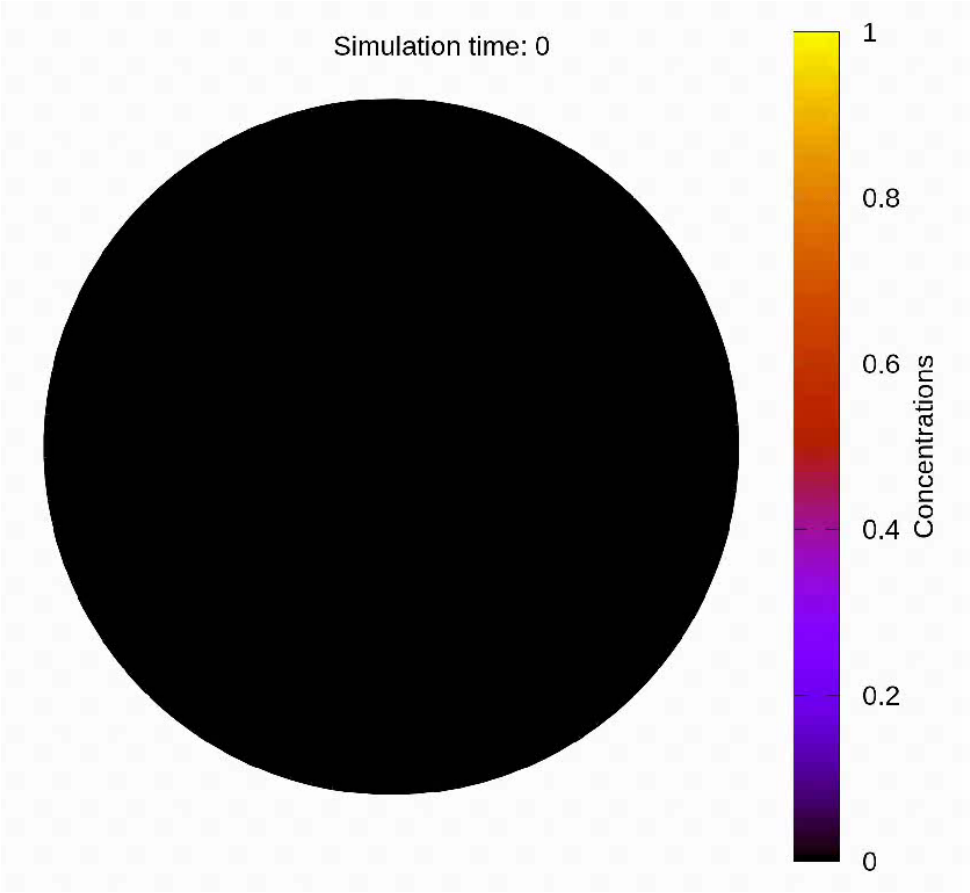
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

Characteristic: development of **sector-like** patterns



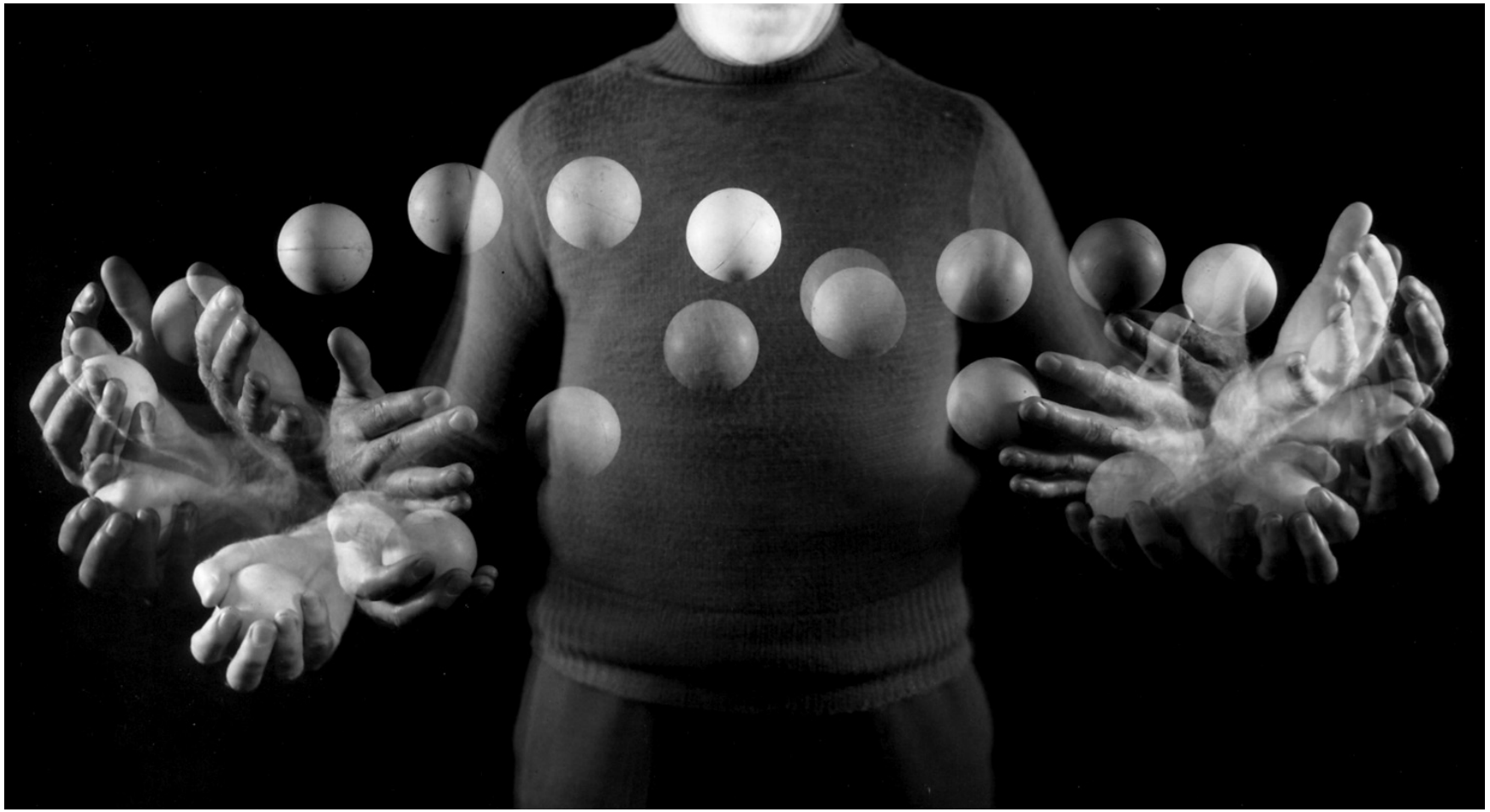
Phase variation model simulations

{david.gilbert,ovidiu.parvu}@brunel.ac.uk



Movement….

{david.gilbert,ovidiu.parvu}@brunel.ac.uk



Movement: Approach 1

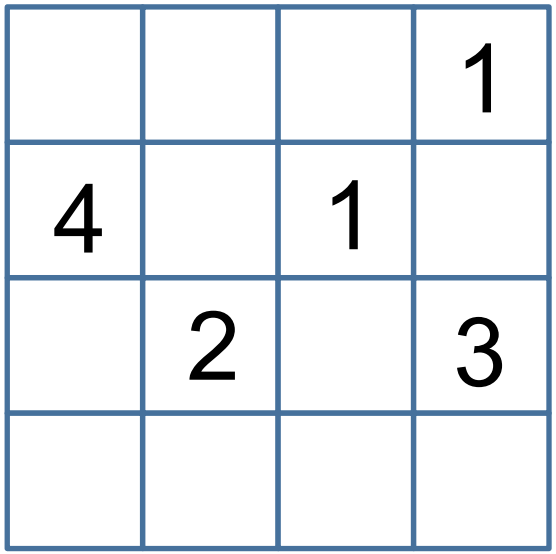
{david.gilbert,ovidiu.parvu}@brunel.ac.uk

2D

Model space using a **grid** with (x, y) colour tuples and use a count for the number of agents per grid position

Advantage:

Memory efficient



Disadvantage:

Agents do not have a state

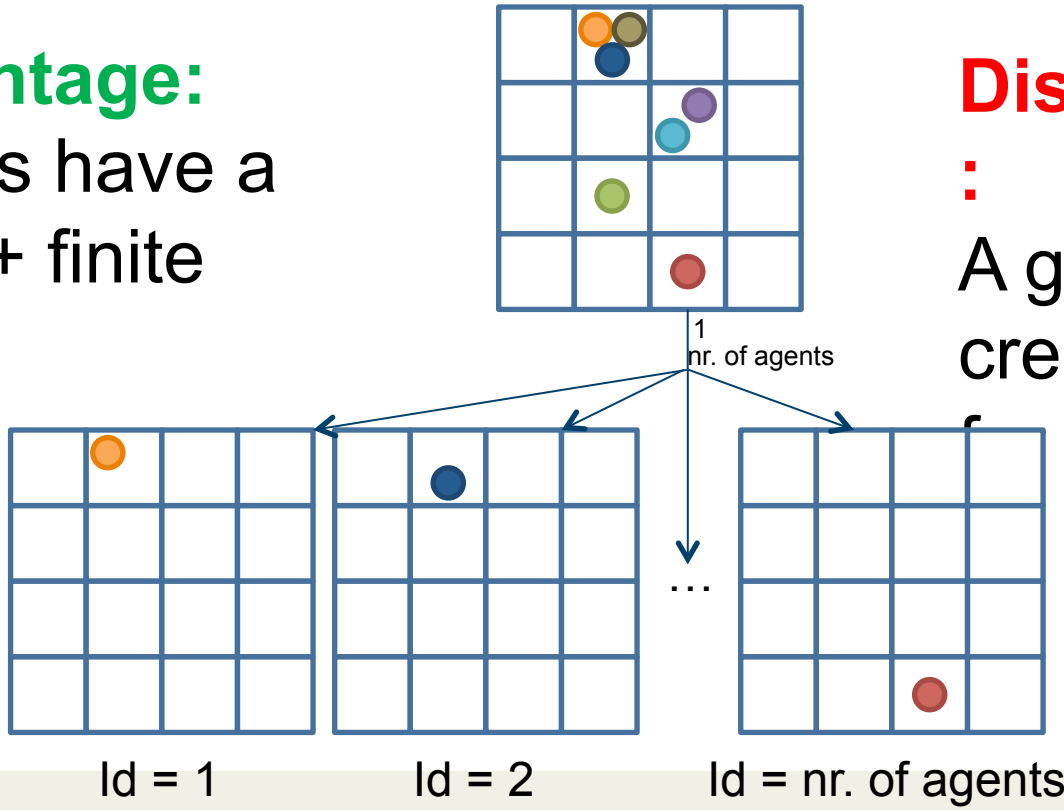
Movement: Approach 2

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

2D

Model each **agent** with an Id and mark the position(s) (X, Y) it occupies

Advantage:
Agents have a state + finite space



Disadvantage:
A grid is created for each agent

Movement: Approach 3

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

2D

Model every **agent** as two places “x” and “y” where each place gets a colour tuple Id

Advantage:

Memory efficient and agents have a state

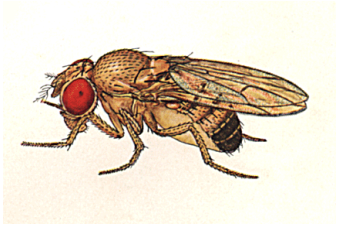
Id	X	Y
1	1	2
2	3	4
…	…	…
Nr. of agents	1	3

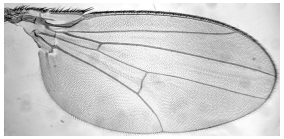
Disadvantage:

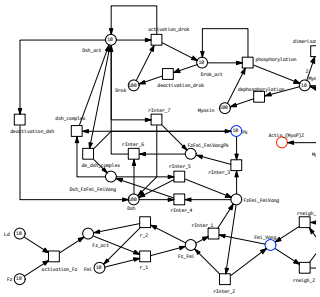
Higher runtime effort (state-dependent rates requires special tool support)

Multiscale: from signalling to organs

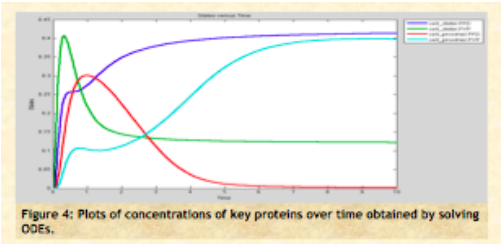
{david.gilbert,ovidiu.parvu}@brunel.ac.uk





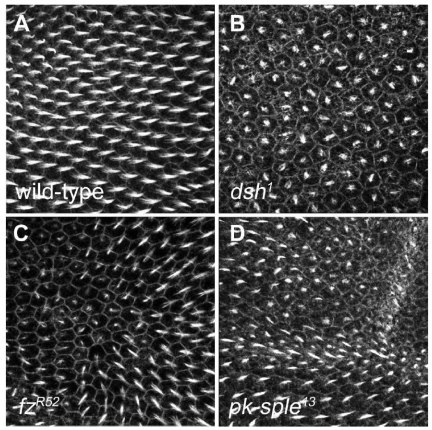


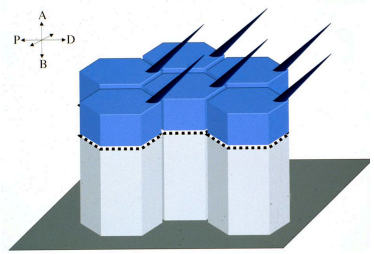
Petri nets (*coloured, hierarchical*)
Monika Heiner

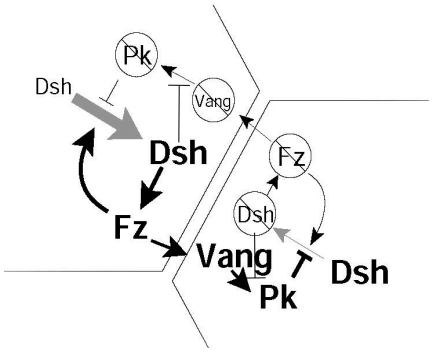


ODEs

Planar Cell Polarity







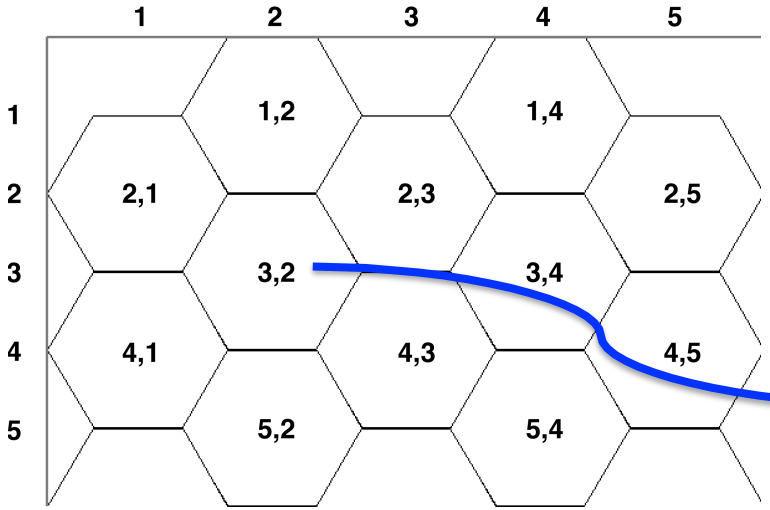
Gao et al (2013). TCCB, 10:2.

Hierarchical Organisation
Tissues – cells – ‘compartments’

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

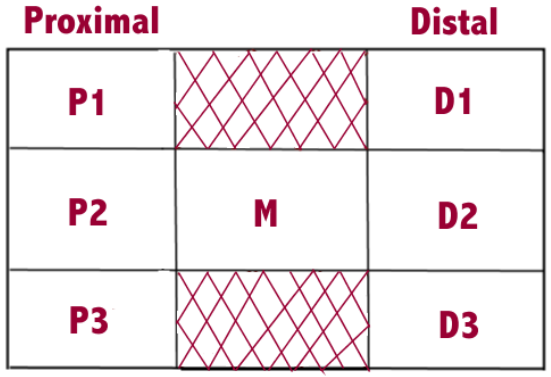
Hierarchically coloured

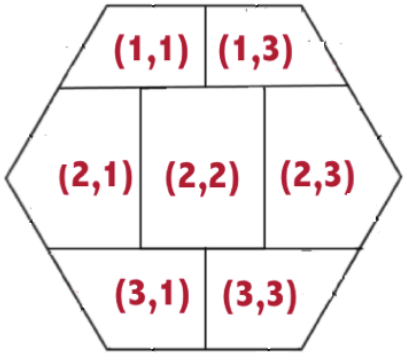
Tissue (Cells)



Cell: (3,2)
Compartment (2,1)

Intracellular compartments



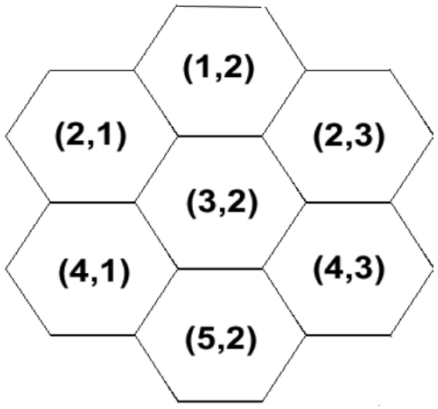


Colourset = {…, {((3,2)(1,1)), ((3,2)(2,1)), ((3,2)(3,1)),……((3,2)(3,3))}, …

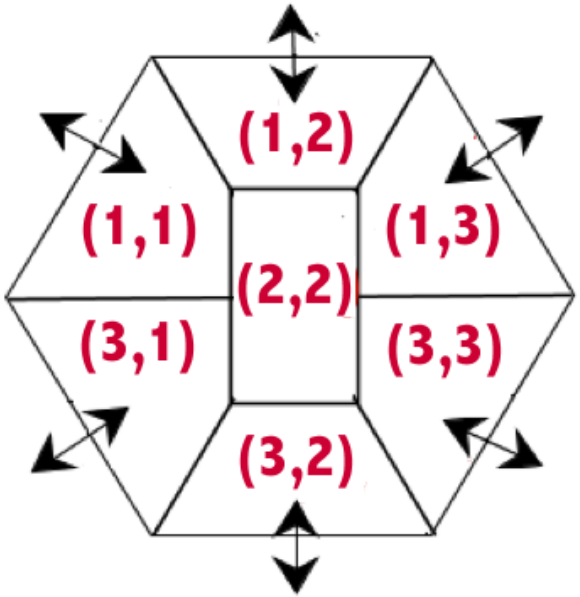
Different Compartmentalisation

david.gilbert@brunel.ac.uk

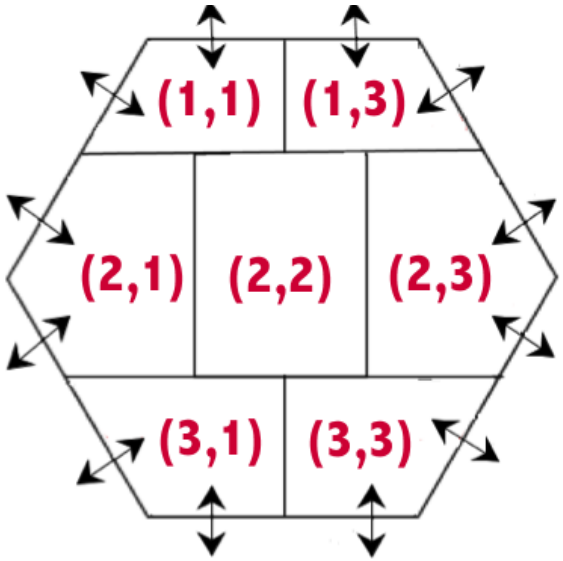
Symmetric Models



1:1

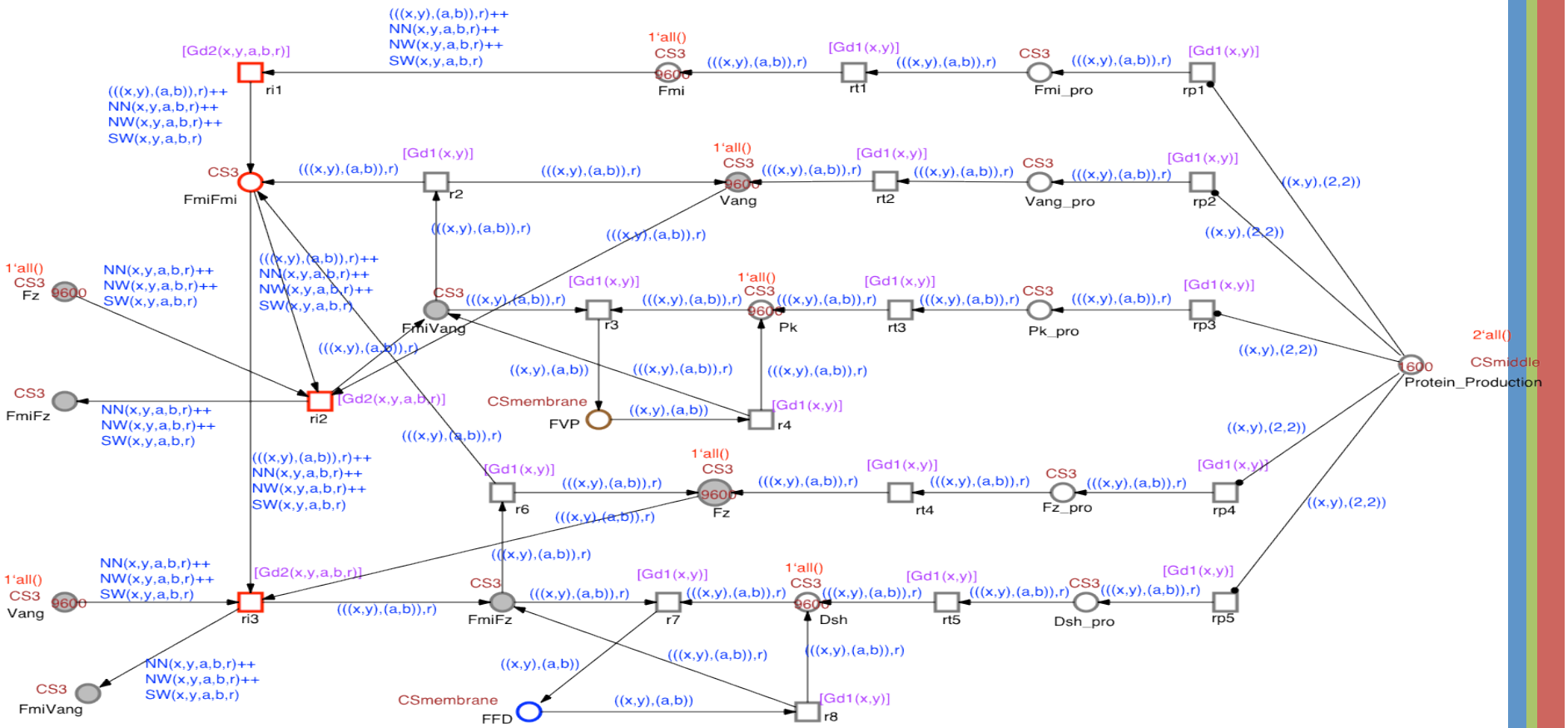


1:2



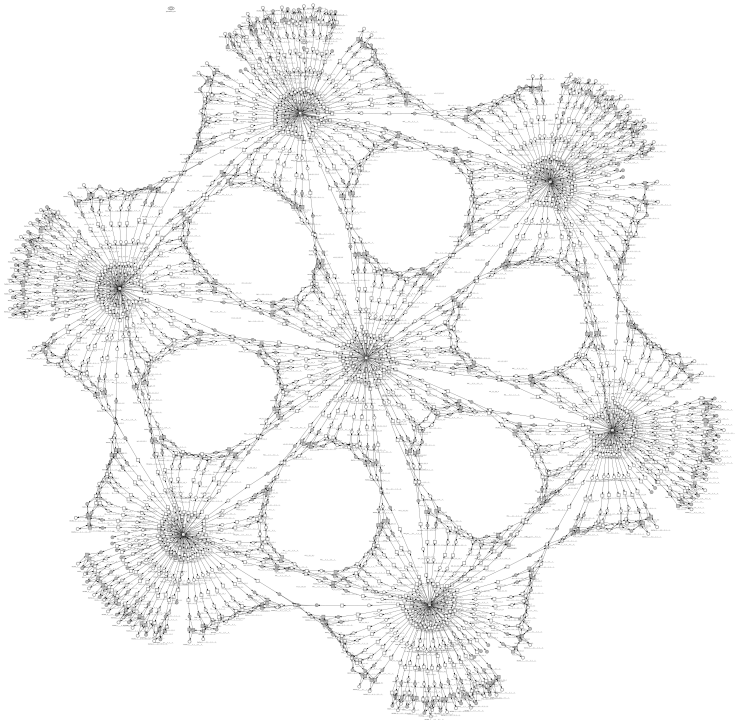
{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk

HCPN Model for Tissue
(Hierarchically Coloured Petri Net)
Detailed level

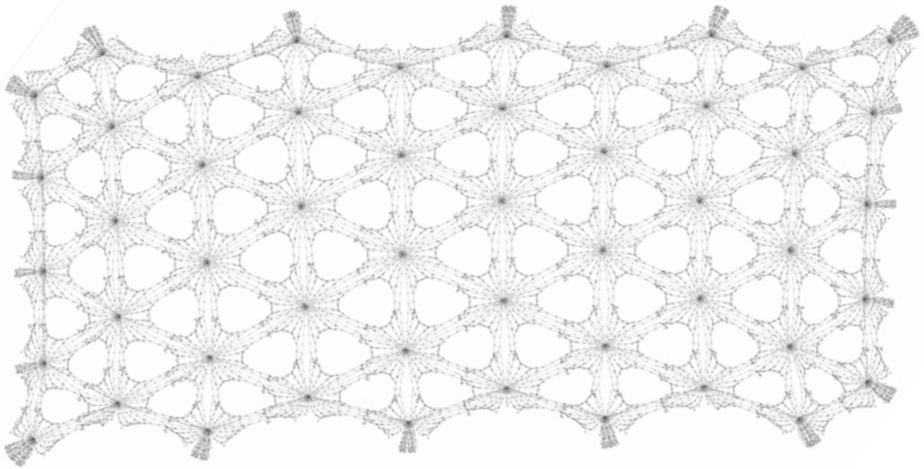


Example unfolding

{david.gilbert,ovidiu.parvu}@brunel.ac.uk

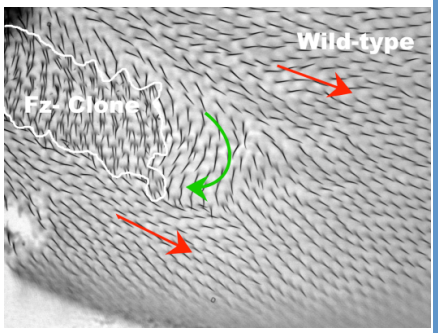


Unfolding at the **tissue-level**



Unfolding at the **cellular-level**

Modelling Mutant Clones



Knock-out: cell clones in which a certain gene is knocked out are induced in the tissue (Biological experiments)

- no corresponding protein produced.

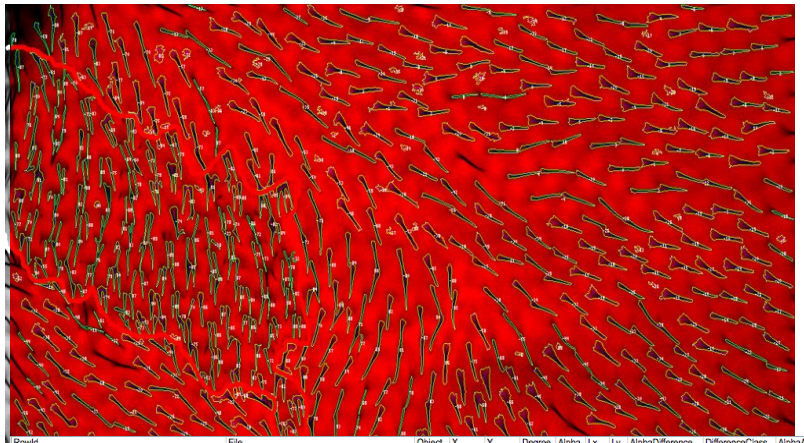
Petri nets: set the protein concentration zero

CPN (repeat, with variations)

- Big enough patch (tissue): 800 cells
- Size / shape of clone: 80 cells (10% of the patch) in a circle-like shape

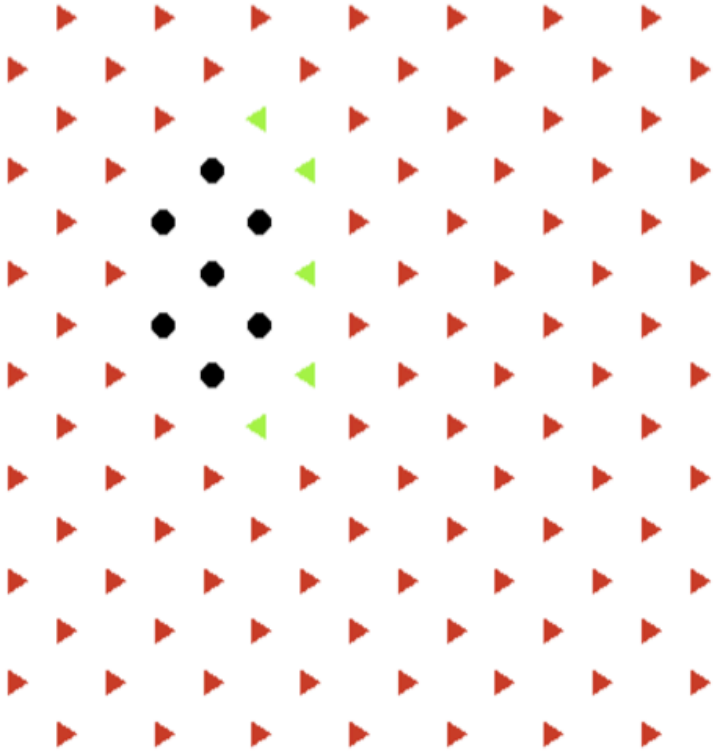
david.gilbert@brunel.ac.uk

Mutated tissue
Experimental vs In-silico



RowId	File	Object	X	Y	Degree	Alpha	Lx	Ly	AlphaDifference	DifferenceClass	AlphaAverage
dsh1-1001005_012.TIF_1	dsh1-1001005_012.TIF	1	23	1104	105	3.40	15	1	0.27	5	3.12
dsh1-1001005_012.TIF_2	dsh1-1001005_012.TIF	2	23	1469	28	2.06	15	1	1.06	2	3.12
dsh1-1001005_012.TIF_3	dsh1-1001005_012.TIF	3	24	922	113	3.55	15	1	0.43	5	3.12
dsh1-1001005_012.TIF_4	dsh1-1001005_012.TIF	4	25	537	158	4.33	15	1	1.21	2	3.12
dsh1-1001005_012.TIF_5	dsh1-1001005_012.TIF	5	25	1602	46	2.37	15	1	0.75	3	3.12
dsh1-1001005_012.TIF_6	dsh1-1001005_012.TIF	6	26	1479	29	2.08	15	1	1.04	2	3.12
dsh1-1001005_012.TIF_7	dsh1-1001005_012.TIF	7	28	1489	20	1.92	15	1	1.21	2	3.12
dsh1-1001005_012.TIF_8	dsh1-1001005_012.TIF	8	28	1545	177	4.65	15	1	1.53	0	3.12
dsh1-1001005_012.TIF_9	dsh1-1001005_012.TIF	9	30	1517	67	2.73	15	1	0.39	5	3.12
dsh1-1001005_012.TIF_10	dsh1-1001005_012.TIF	10	31	1635	11	1.76	15	1	1.36	1	3.12
dsh1-1001005_012.TIF_11	dsh1-1001005_012.TIF	11	32	1619	36	2.19	15	1	0.93	3	3.12
dsh1-1001005_012.TIF_12	dsh1-1001005_012.TIF	12	33	1426	59	2.60	15	1	0.52	4	3.12
dsh1-1001005_012.TIF_13	dsh1-1001005_012.TIF	13	35	1389	40	2.27	15	1	0.85	3	3.12
dsh1-1001005_012.TIF_14	dsh1-1001005_012.TIF	14	35	1563	82	3.01	15	1	0.12	6	3.12
dsh1-1001005_012.TIF_15	dsh1-1001005_012.TIF	15	35	1777	15	1.84	15	1	1.28	1	3.12
dsh1-1001005_012.TIF_16	dsh1-1001005_012.TIF	16	37	835	149	4.17	15	1	1.05	2	3.12
dsh1-1001005_012.TIF_17	dsh1-1001005_012.TIF	17	37	1446	68	2.75	15	1	0.37	5	3.12
dsh1-1001005_012.TIF_18	dsh1-1001005_012.TIF	18	37	1535	15	1.84	15	1	1.28	1	3.12
dsh1-1001005_012.TIF_19	dsh1-1001005_012.TIF	19	39	805	153	4.25	15	1	1.12	2	3.12
dsh1-1001005_012.TIF_20	dsh1-1001005_012.TIF	20	41	1464	50	2.45	15	1	0.68	4	3.12
dsh1-1001005_012.TIF_21	dsh1-1001005_012.TIF	21	44	992	123	3.72	15	1	0.60	4	3.12
dsh1-1001005_012.TIF_22	dsh1-1001005_012.TIF	22	44	1554	16	1.86	15	1	1.27	1	3.12
dsh1-1001005_012.TIF_23	dsh1-1001005_012.TIF	23	44	1608	19	1.90	15	1	1.23	1	3.12
dsh1-1001005_012.TIF_24	dsh1-1001005_012.TIF	24	46	1376	50	2.45	15	1	0.68	4	3.12
dsh1-1001005_012.TIF_25	dsh1-1001005_012.TIF	25	46	1484	53	2.50	15	1	0.62	4	3.12
dsh1-1001005_012.TIF_26	dsh1-1001005_012.TIF	26	47	1087	87	3.08	15	1	0.04	6	3.12
dsh1-1001005_012.TIF_27	dsh1-1001005_012.TIF	27	47	1626	178	4.67	15	1	1.55	0	3.12
dsh1-1001005_012.TIF_28	dsh1-1001005_012.TIF	28	49	846	111	3.51	15	1	0.39	5	3.12
dsh1-1001005_012.TIF_29	dsh1-1001005_012.TIF	29	49	1524	18	1.88	15	1	1.24	1	3.12
dsh1-1001005_012.TIF_30	dsh1-1001005_012.TIF	30	49	1645	40	2.27	15	1	0.86	3	3.12
dsh1-1001005_012.TIF_31	dsh1-1001005_012.TIF	31	50	1502	36	2.20	15	1	0.93	3	3.12
dsh1-1001005_012.TIF_32	dsh1-1001005_012.TIF	32	51	818	144	4.09	15	1	0.97	2	3.12
dsh1-1001005_012.TIF_33	dsh1-1001005_012.TIF	33	51	1164	26	2.02	15	1	1.11	2	3.12
dsh1-1001005_012.TIF_34	dsh1-1001005_012.TIF	34	51	1787	16	1.86	15	1	1.27	1	3.12
dsh1-1001005_012.TIF_35	dsh1-1001005_012.TIF	35	53	1009	59	2.60	15	1	0.52	4	3.12
dsh1-1001005_012.TIF_36	dsh1-1001005_012.TIF	36	53	1145	51	2.47	15	1	0.66	4	3.12
dsh1-1001005_012.TIF_37	dsh1-1001005_012.TIF	37	53	1431	75	2.89	15	1	0.24	6	3.12
dsh1-1001005_012.TIF_38	dsh1-1001005_012.TIF	38	53	1667	10	1.74	15	1	1.38	1	3.12
dsh1-1001005_012.TIF_39	dsh1-1001005_012.TIF	39	53	1683	10	1.74	15	1	1.38	1	3.12
dsh1-1001005_012.TIF_40	dsh1-1001005_012.TIF	40	54	1573	22	1.96	15	1	1.17	2	3.12
dsh1-1001005_012.TIF_41	dsh1-1001005_012.TIF	41	55	922	106	3.42	15	1	0.29	5	3.12
dsh1-1001005_012.TIF_42	dsh1-1001005_012.TIF	42	55	1472	60	2.62	15	1	0.51	4	3.12
dsh1-1001005_012.TIF_43	dsh1-1001005_012.TIF	43	57	1393	39	2.25	15	1	0.87	3	3.12
dsh1-1001005_012.TIF_44	dsh1-1001005_012.TIF	44	57	1451	69	2.77	15	1	0.35	5	3.12

FFD at distal vs FFD at proximal over Tissue



• Q. Gao, F. Liu, D. Gilbert, M. Heiner & D. Tree. **CMSB 2011**, Paris, France.

Analysis & Visualisation

david.gilbert@brunel.ac.uk

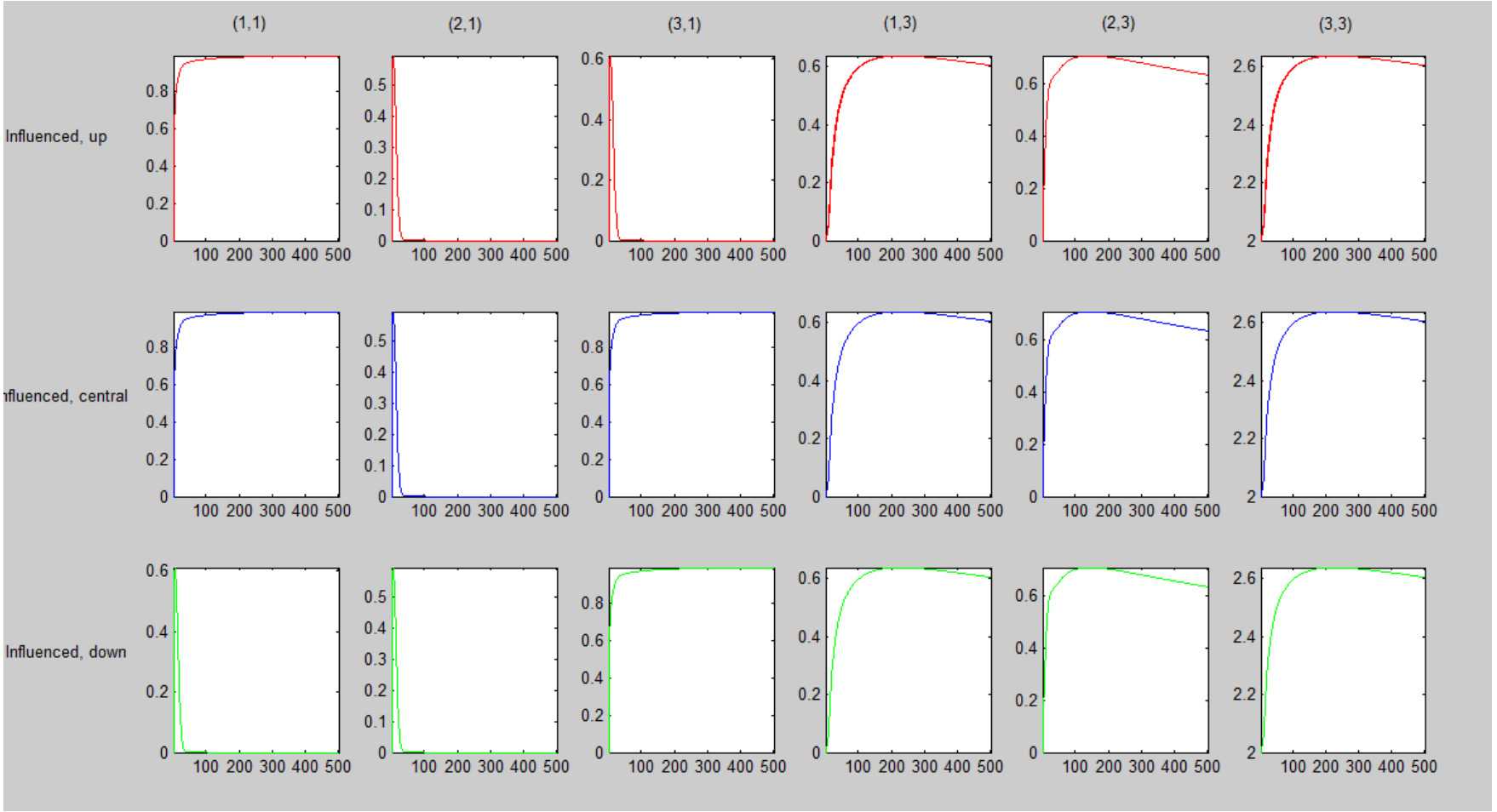
Clustering

- DBScan
- Hierarchical clustering
- K-means
- SOMs

Model checking

Clustering of behaviours

david.gilbert@brunel.ac.uk

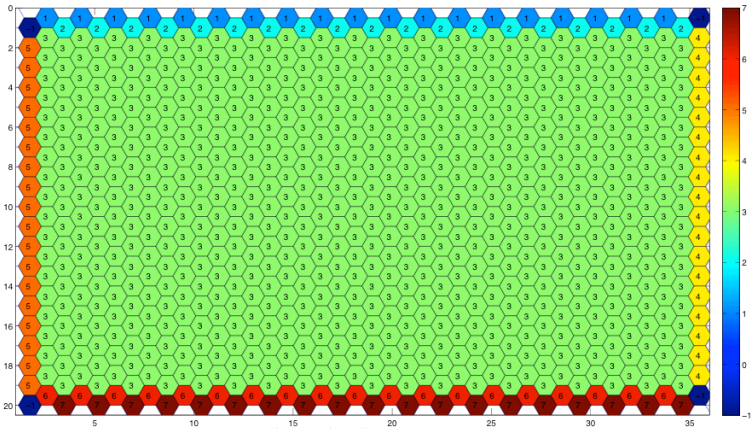


Clustering

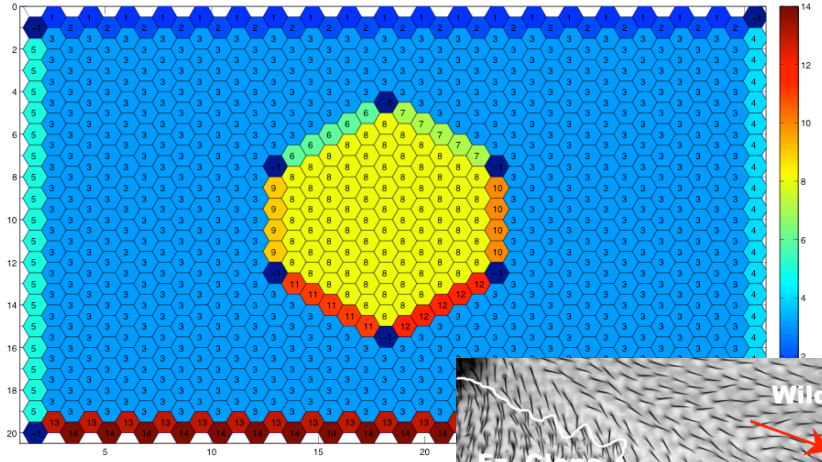
david.gilbert@brunel.ac.uk

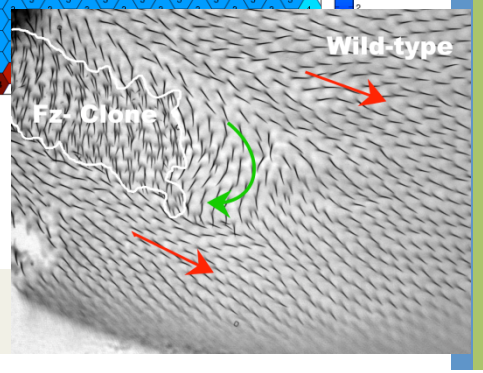
DBScan with Principal Component Analysis (PCA)

Unbiased model:
Grid 40*40 (800 cells)

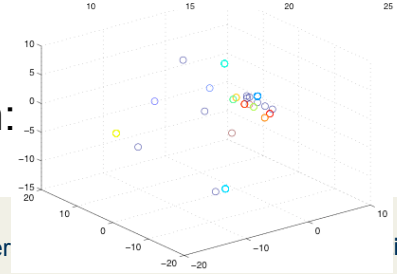


Fz- mutant clone model:
A patch of mutated cells lacking Frizzled (Fz) in a wild-type background





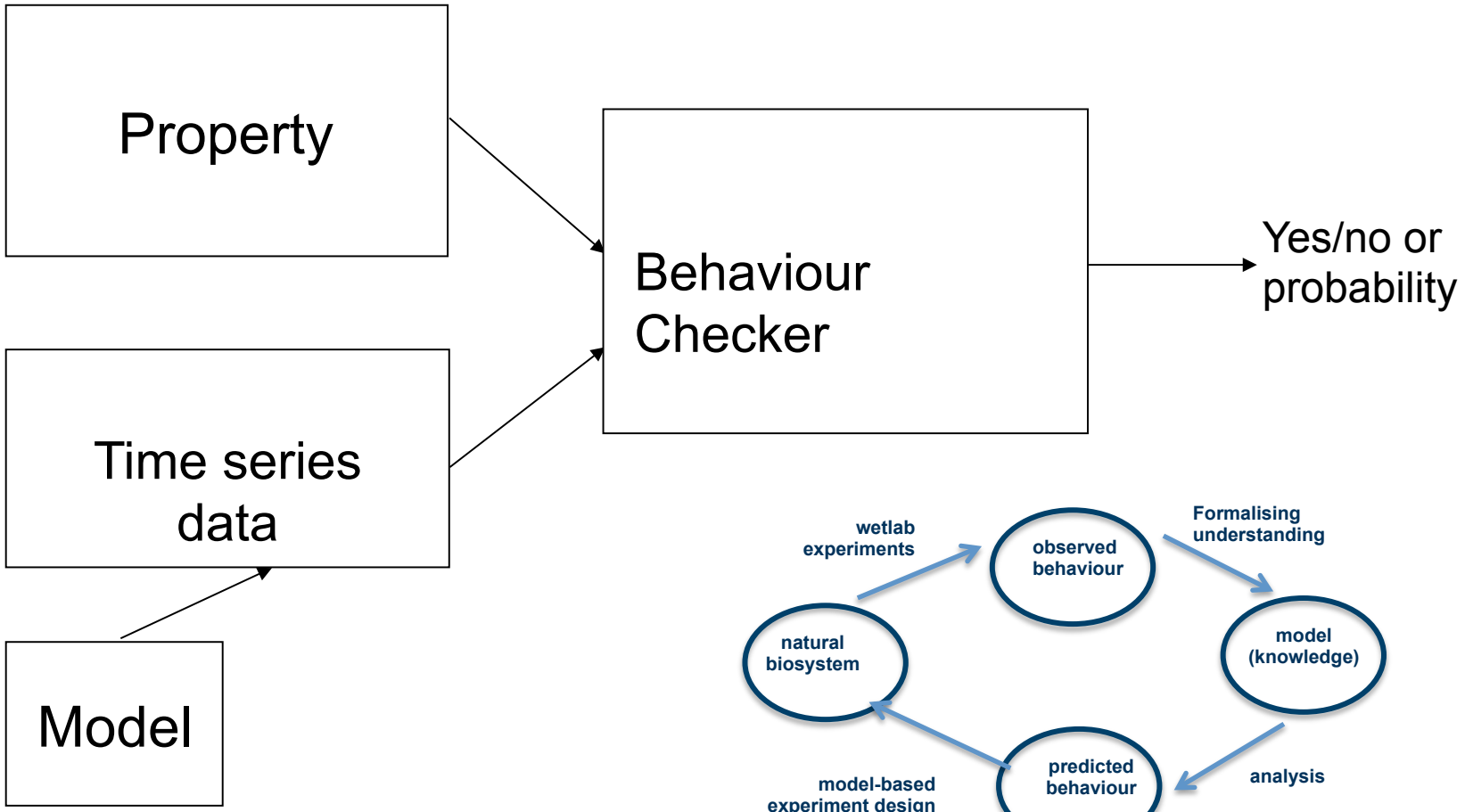
Feature selection: PCA

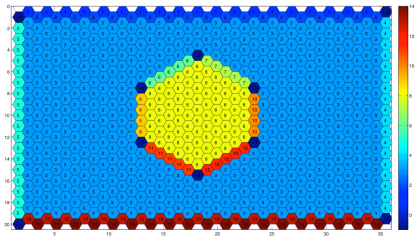


Simulation-based Model Checking

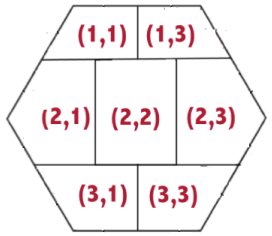
david.gilbert@brunel.ac.uk

- PLTL using MC2 [Donaldson&Gilbert CMSB 2008]





Model Checking
Primary data



Fz- mutant clone model

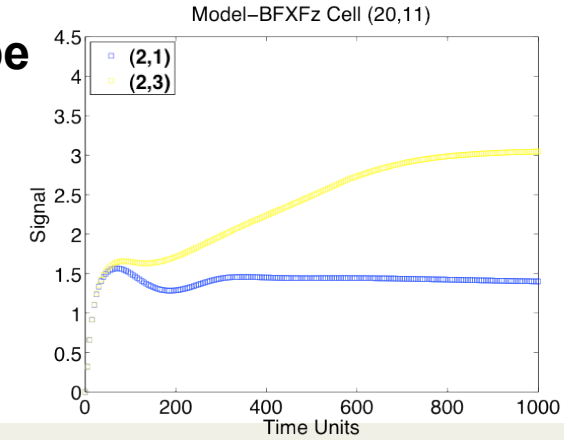
Unlike in the wild-type cells, for the **cells distally neighbouring to the Fz- clone** the concentration of FFD in the middle distal compartment is always lower than that of the middle proximal compartment:

P=? [time > 0 → G(D2 < P2)]

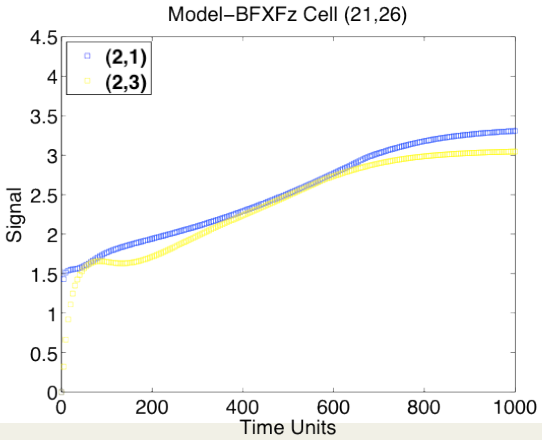
Moreover, the trace of D2 exhibits a peak followed by a trough, which is not true for P2:

P=?[F(d(D2) > 0 ∧ F(d(D2) < 0 ∧ F(d(D2) > 0)))]

Wild-type



Distally neighbouring to the clone

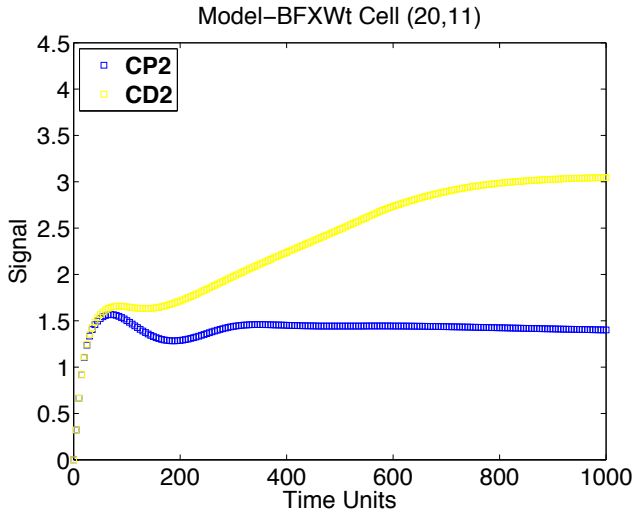


Big idea – check cumulative signal!

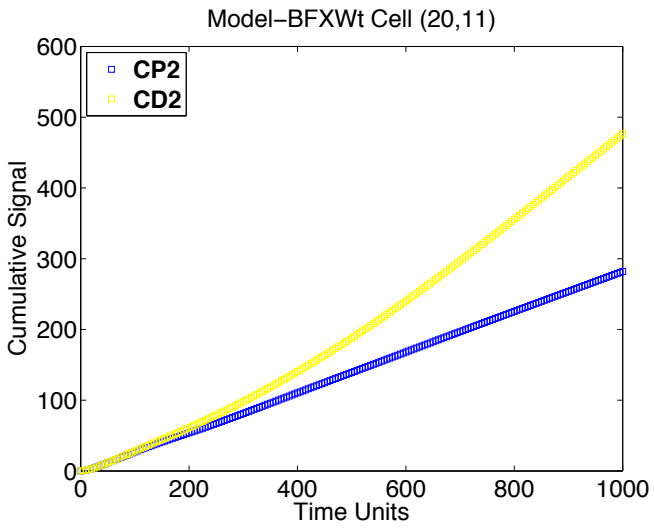
david.gilbert@brunel.ac.uk

- Cumulative signal: time-series of accumulated concentrations of FFD (secondary data)
- Why?

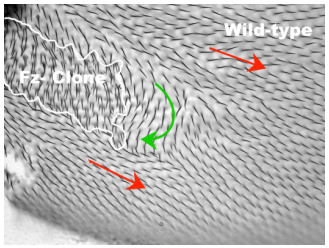
- The localisation of PCP signalling at any given time point is the result of the cumulative effect of the sum over the signalling events until that point.



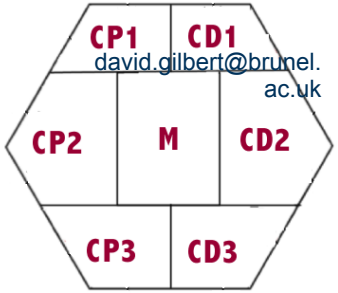
Primary data



Secondary data



Model Checking
Secondary data
Fz- mutant clone model



david.gilbert@brunel.ac.uk

Wild type cells in the tissue (i.e. away from the clone area).
After short initial period: Always middle distal cumulative[FFD] greater than middle proximal cumulative[FFD]

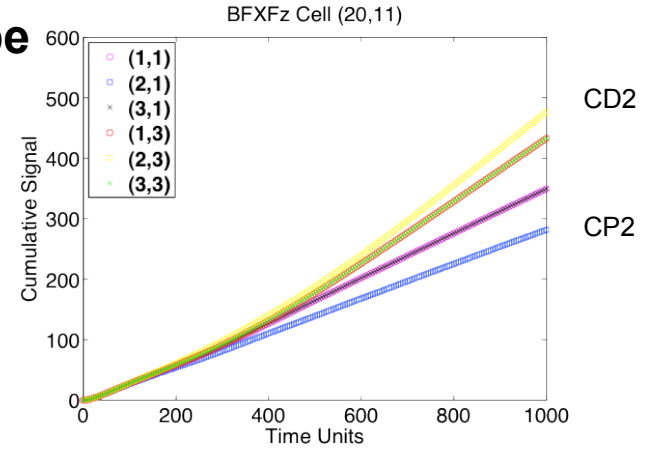
P=? [time > ε → G(CD2 > CP2)]

Wild type cells distally neighbouring to clone in the tissue
After short initial period: Always middle distal cumulative[FFD] less than middle proximal cumulative[FFD]

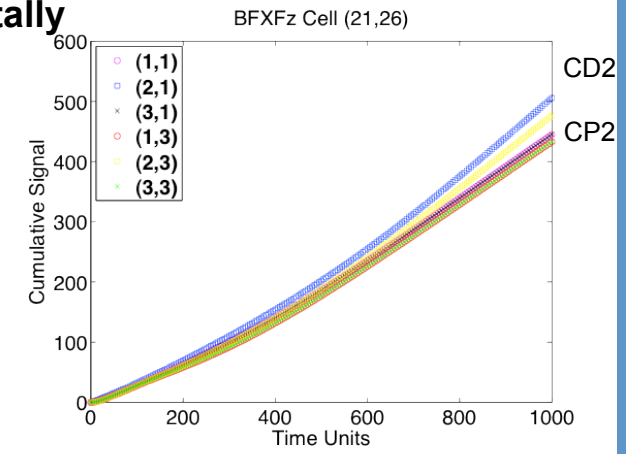
P=? [time > ε → G(CD2 < CP2)]

Hairs grow normally in wild-type, but disturbed in WT distally near clone, influence from the clone

Wild-type



Wild-type distally neighbouring to clone



Automatic Generation of Temporal Logic Descriptions

We can use PLTLc to characterise the clusters of time series

PLTLc statements should be

- *general* enough to describe all the time series in a given cluster
- *discriminative* enough to distinguish between time series of different clusters

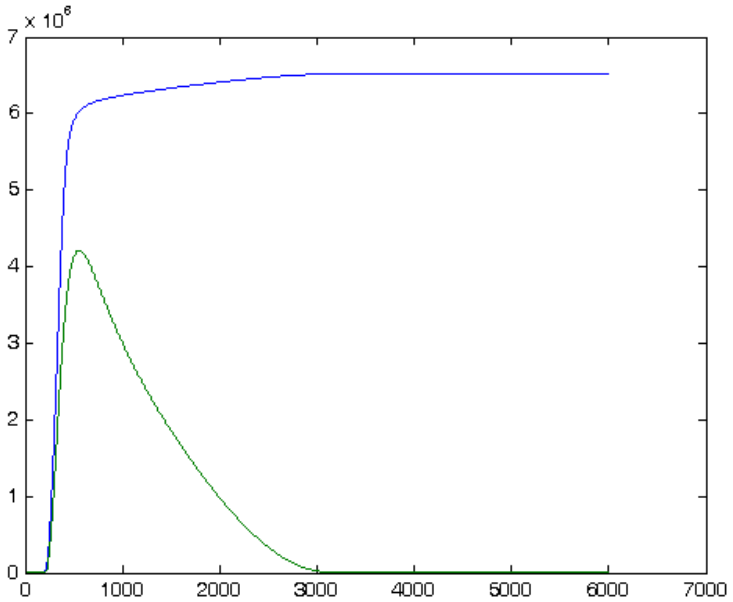
The generation algorithm is based on property patterns (templates)

D. Maccagnola, E. Messina, Q. Gao and D. Gilbert, (2012). A Machine Learning Approach for Generating Temporal Logic Classifications of Complex Model Behaviours. Proc Winter Simulation Conference 2012, IEEE.

Automatic Generation of TL Descriptions

• ***Trend***: describes the trend of a time series as a sequence of direction (“increase”, “steady”, “decrease”)

$$\phi_1 U(\phi_2 U(\ldots U(\phi_{m-1} U(G(\phi_m)))\ldots))$$



If a cluster shows different trends, they are ordered by frequency (F_0 is the most frequent, then F_1 and so on) and the cluster trend is defined by:

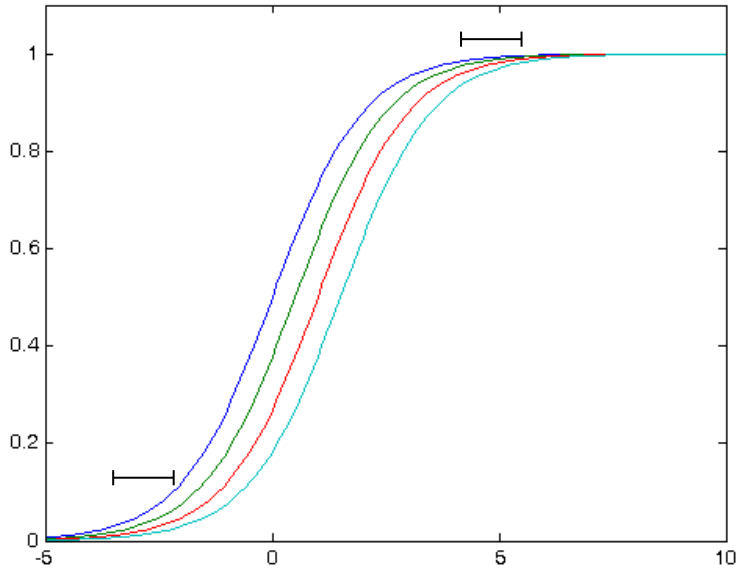
F_0 V F_1 V F_2 V …

Example:
steady-increase-steady OR
steady-increase-decrease-steady

d = 0 U d > 0 U (G(d=0)) V
d = 0 U d > 0 U d<0 U (G(d=0))

Automatic Generation of TL Descriptions

- ***Time***: identifies the time points when the time series changes its direction, i.e. a set of “direction changes”

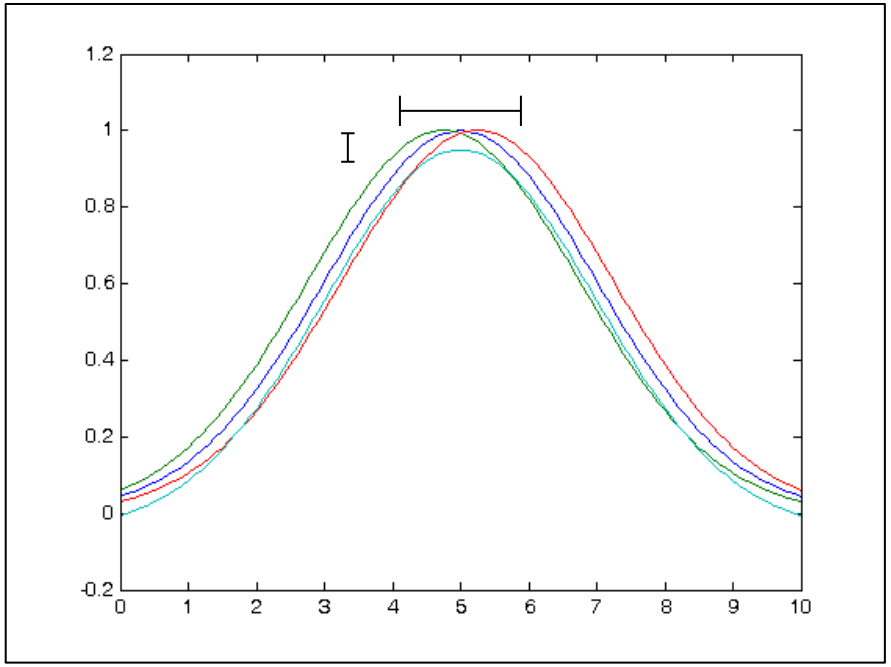


Time series with the same trend may have slightly different time patterns

We compute a set of *time intervals*

Automatic Generation of TL Descriptions

• ***Extrema***: represents the occurrence of all the local minima and maxima of a time series

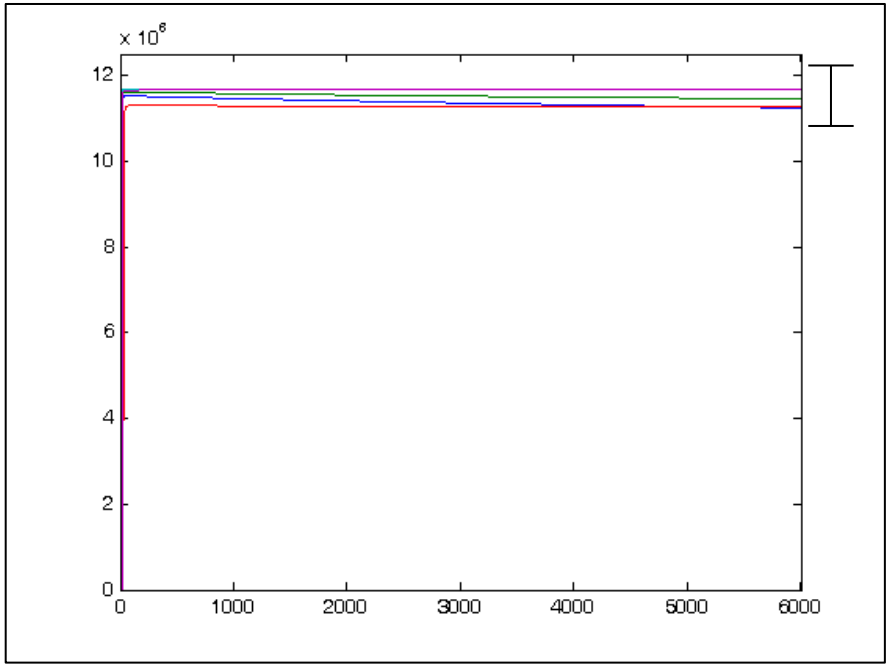


The time and value of each extrema can slightly change among the time series in a cluster

The extrema of a cluster are defined by a sequence of *time* and *value* intervals

Automatic Generation of TL Descriptions

• ***Steady state***: represents the value of the time series steady state (if exists)



The value of each steady state can slightly change among the time series in a cluster

The steady state of a cluste, if exists, is defined by a value interval

Automatic Generation of TL Descriptions

PLTLc GENERATION PROCEDURE:

1. Consider cluster C_i and the set of remaining clusters $\neg C_i$;
2. If C_i and $\neg C_i$ have different trends, stop; otherwise, continue;
3. If C_i and $\neg C_i$ have the same trend with different times, stop; otherwise, continue;
4. If C_i and $\neg C_i$ have at least one different extrema, stop; otherwise, continue;
5. If C_i and $\neg C_i$ have different steady states, stop; otherwise, the clusters are identical and the algorithm cannot return a valid description.

Automatic Generation of TL Descriptions

- The effectiveness of this algorithms is affected by:
    - The cluster's quality
    - The number of "direction changes" of the time series
- The effectiveness of this algorithm is NOT affected by the *number of time series* per cluster

Automatic Generation of TL Descriptions

Evaluation

• To evaluate the PLTLc statement, we test it as a temporal logic query over the clusters

• We use the probability $P_{=?}[\phi_{opt}(C_i)]$ that the statement correctly classifies the time series belonging to cluster *i*

•We associate to each statement a "confidence level" *Conf :*

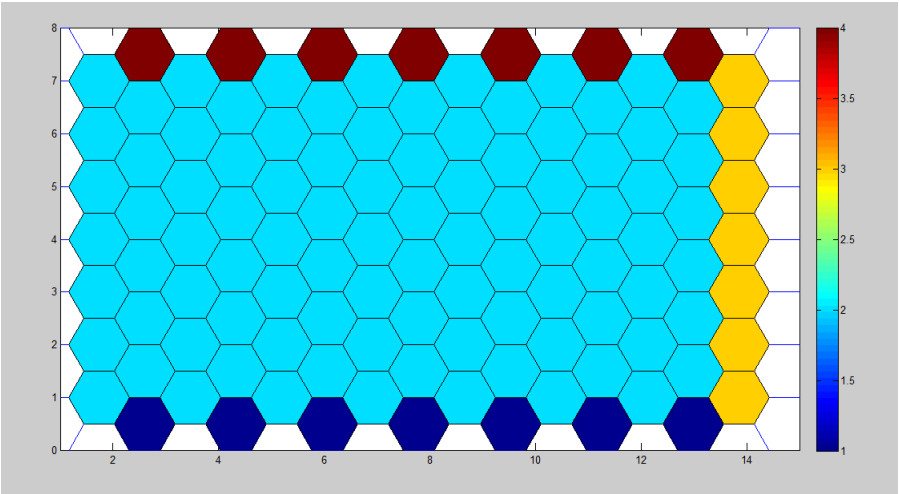
$$Conf(\phi_{opt}(C_i)) = \frac{P_{=?}[\phi_{opt}(C_i)]}{1 + max_{j \neq i} P_{=?}[\phi_{opt}(C_j)]}$$

which indicates its capability to discriminate between time series of cluster i from time series of the most similar cluster j ≠ i.

Results

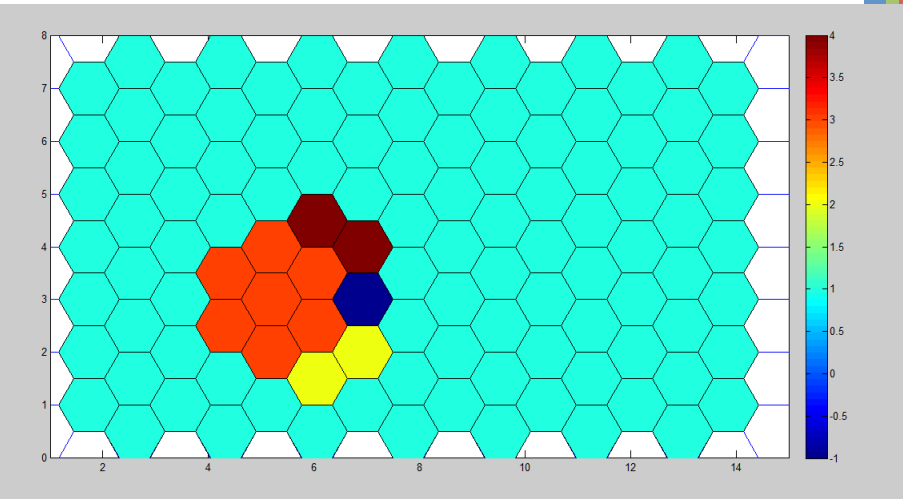
Best clustering result (using DBScan)

Wild Type Tissue



- All the cells have the same behaviour
- The borders are effect of a biased model

Mutated Tissue



- The mutated clone is clearly visible
- Nearby “wild type” cells are INFLUENCED by the mutated clone

DISCOVERED PROPERTIES

PLTLc EXAMPLE:

Behaviour in the INFLUENCED CELLS

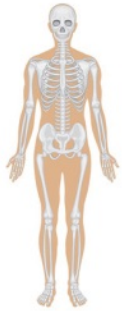
$$P_{=?}[d[FFD] > 0\ U(Time \geq 30 \wedge Time \leq 31 \wedge d[FFD] = 0\ \wedge G(d[FFD] = 0)))]$$

“The concentration of FFD increases from time zero, reaches its peak
between time 30 and 31, and then becomes steady till the end”.

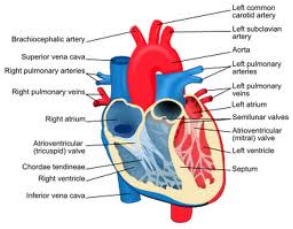
Q. Gao, D. Gilbert, M. Heiner, F. Liu, D. Maccagnola and D. Tree, (2013). Multiscale Modelling and Analysis of Planar Cell Polarity in the Drosophila Wing, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 10:2.

Systems biology

Biological systems are **multidimensional** …

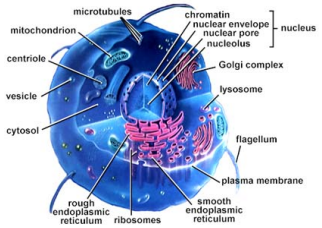


Organism

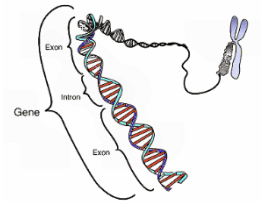


Organ

Tissue



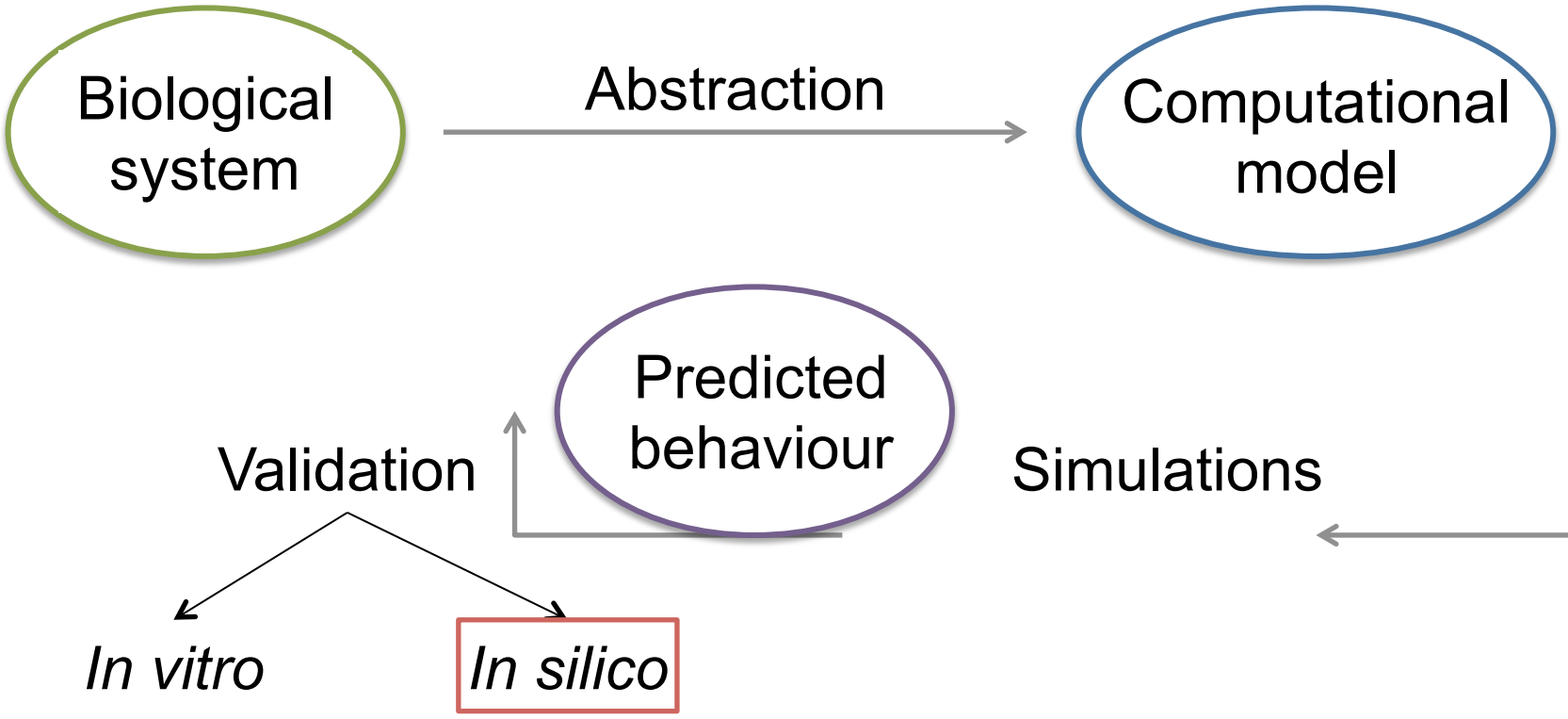
Cell



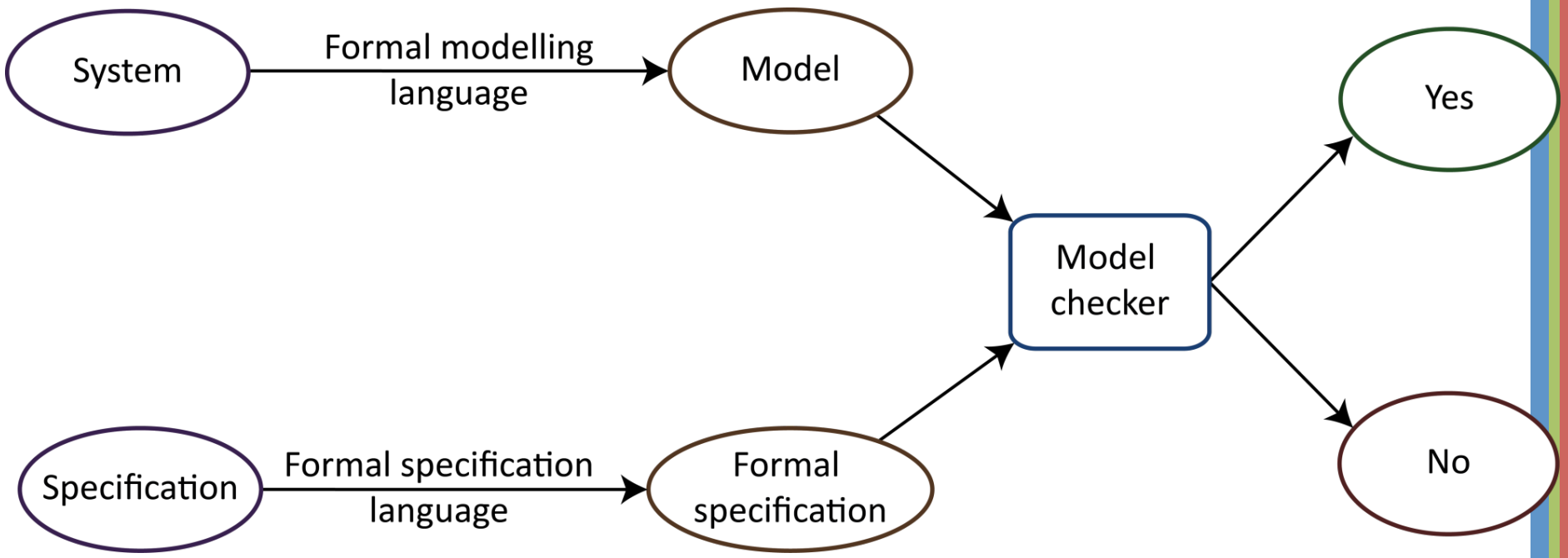
Gene

… and **multiscale**.

Computational models of biological systems

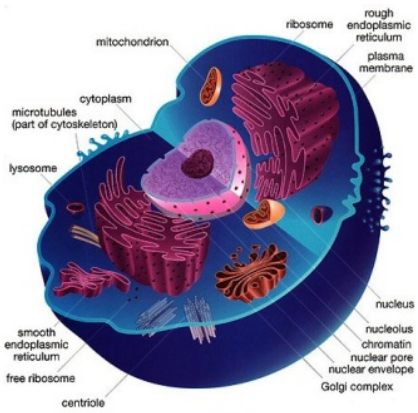


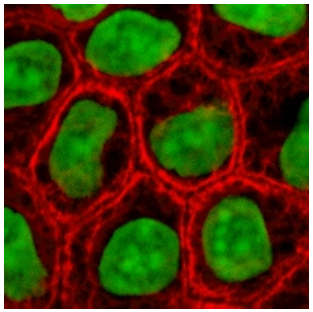
Model checking

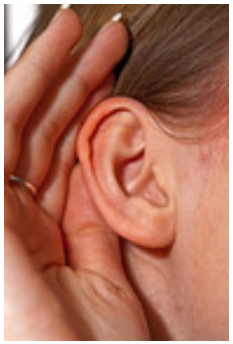


Multiscale!

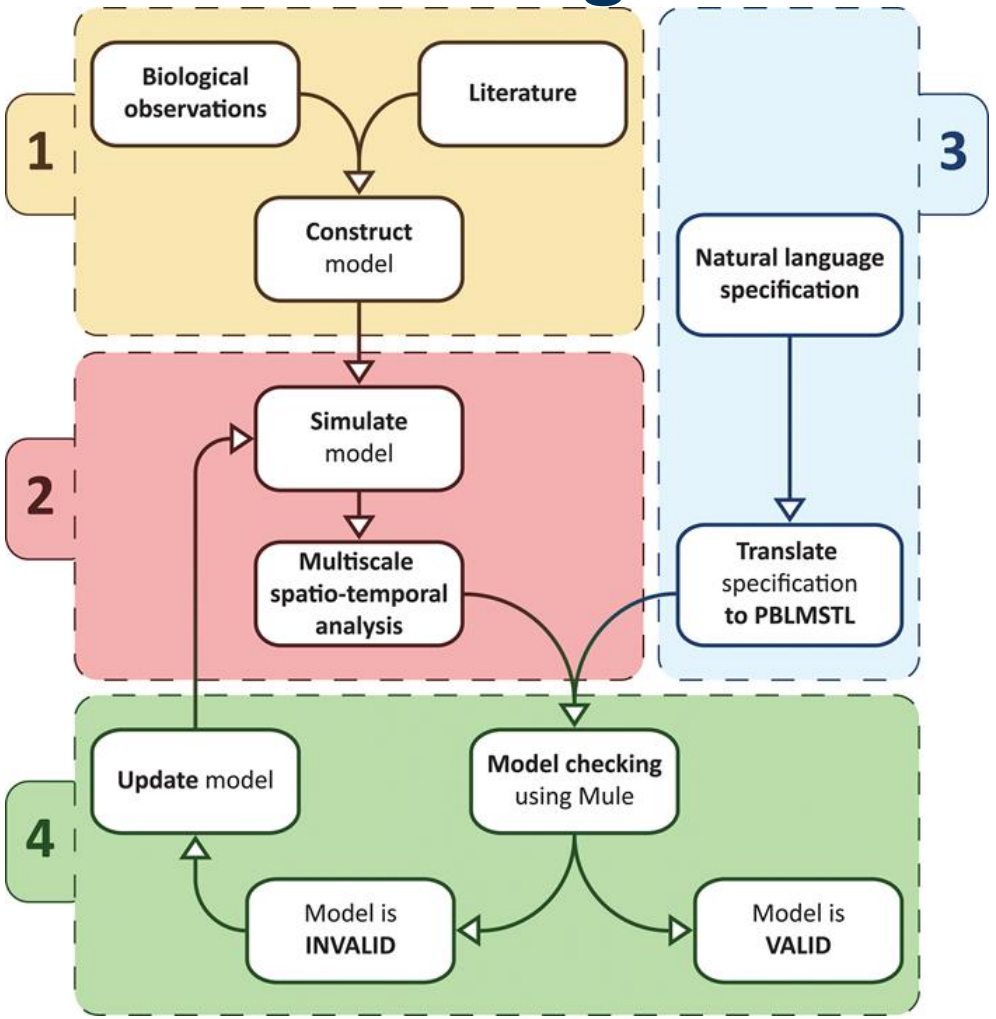
Work by Ovidiu Parvu
PhD student





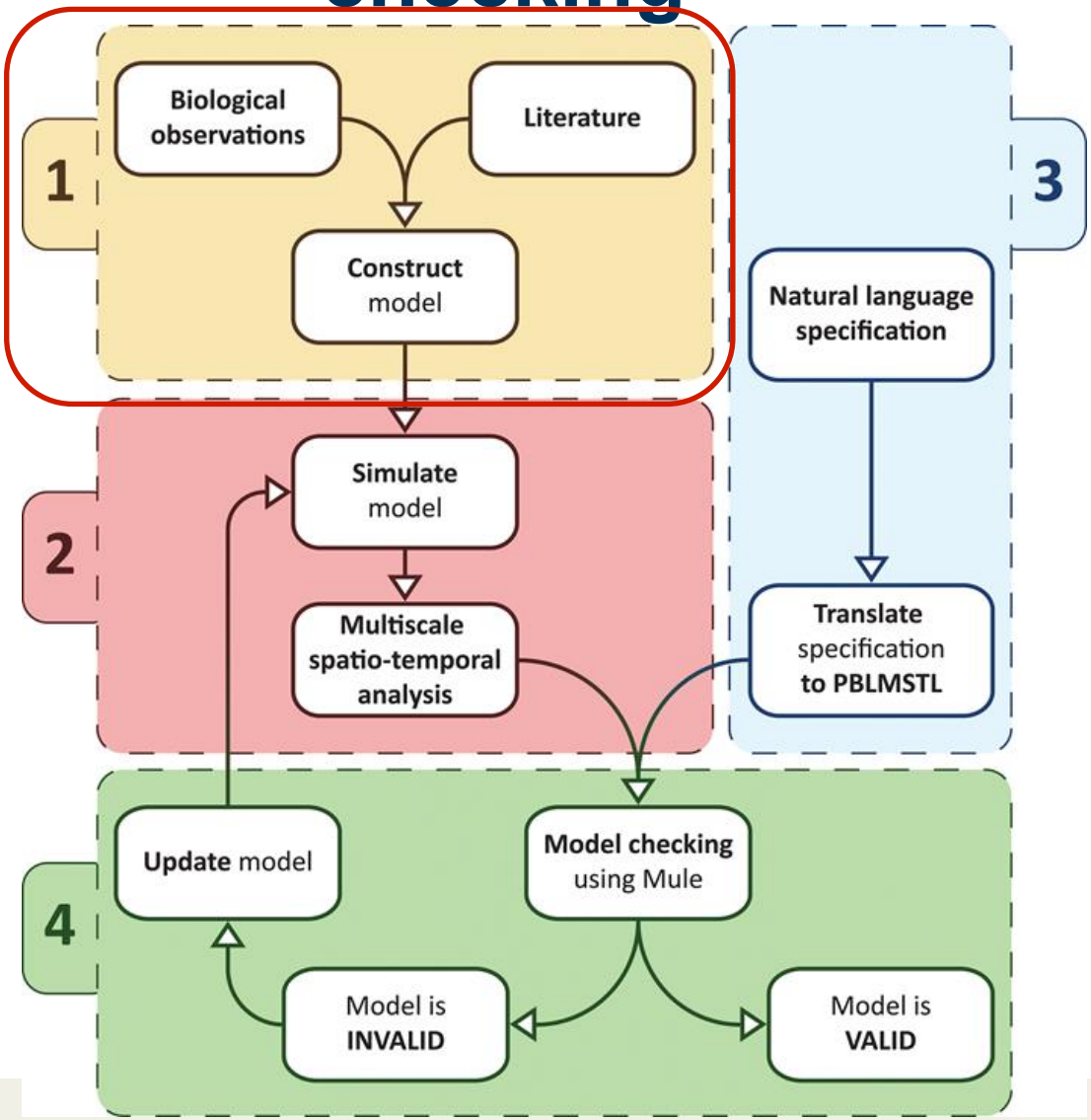


Multiscale spatio-temporal model checking



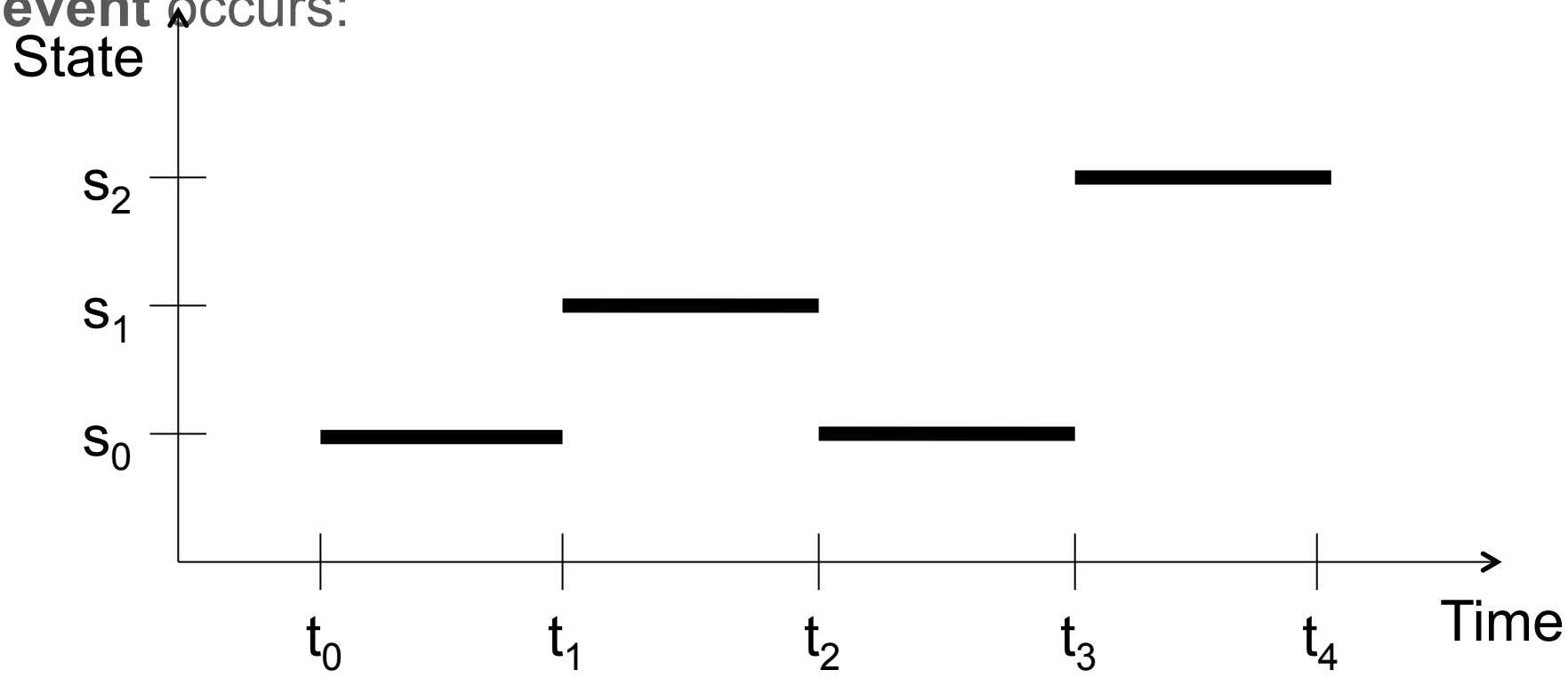
[1] O. Pârvu and D. Gilbert, "A novel method to verify multilevel computational models of biological systems using multiscale spatio-temporal meta model checking," *PLoS ONE* **(under review)**.

Multiscale spatio-temporal model checking



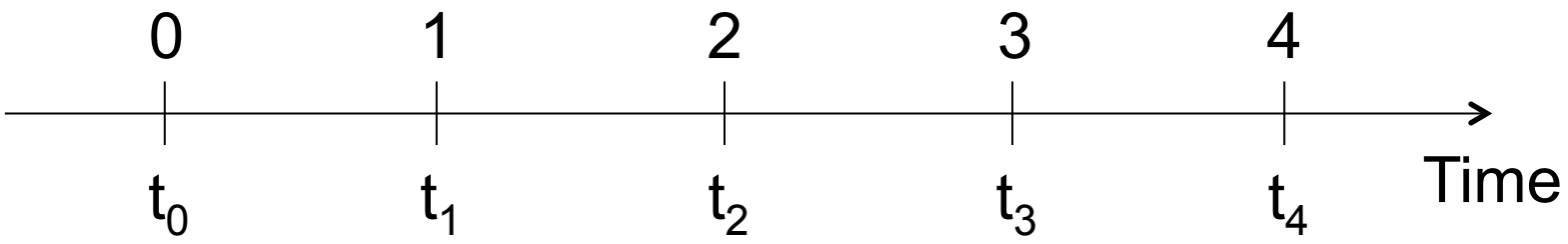
Assumption: stochastic discrete-event systems

The modelled biological systems are assumed to be **stochastic** and transition between states only when an **event** occurs:

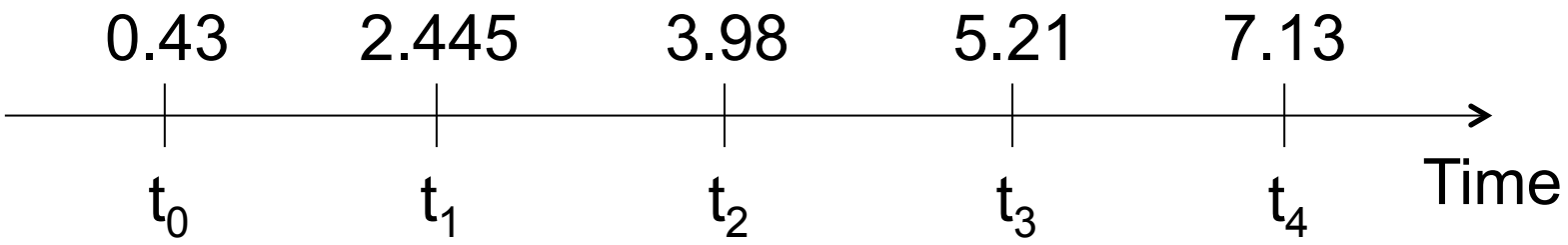


Assumption: Discrete or continuous time

Time is represented in either a **discrete**:

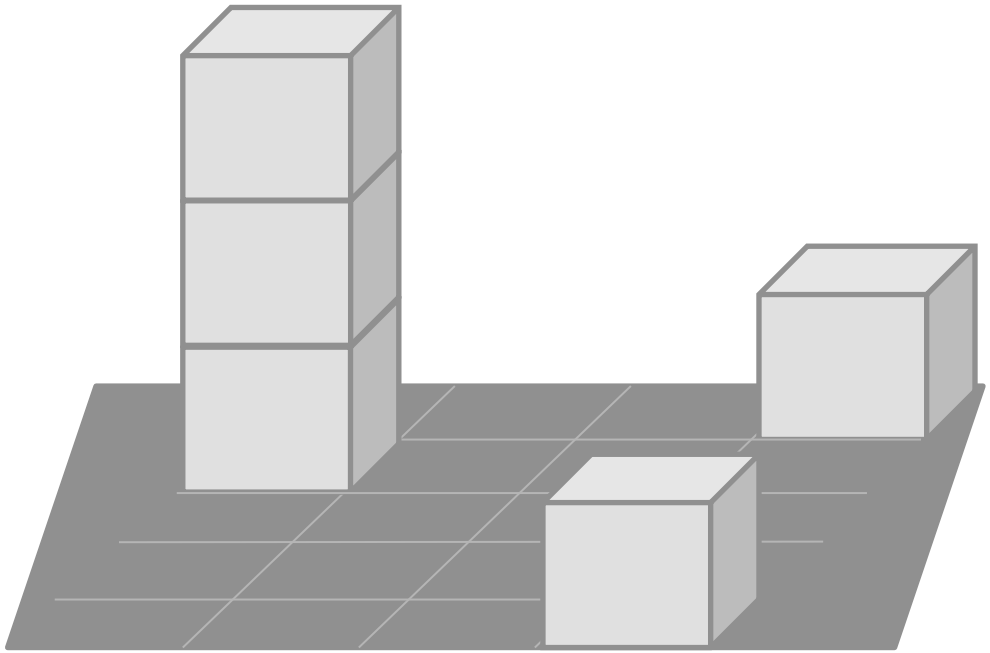


or **continuous** manner:



Assumption:
Pseudo-3D discretised space

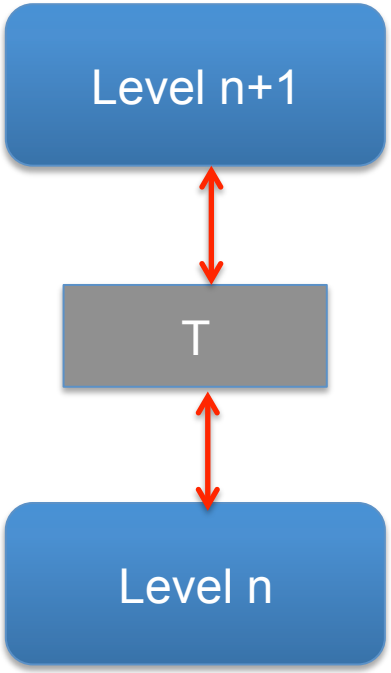
Discretised spatial domain is represented in **pseudo-3D** dimensions i.e. 2D space in which **pile up** is allowed.

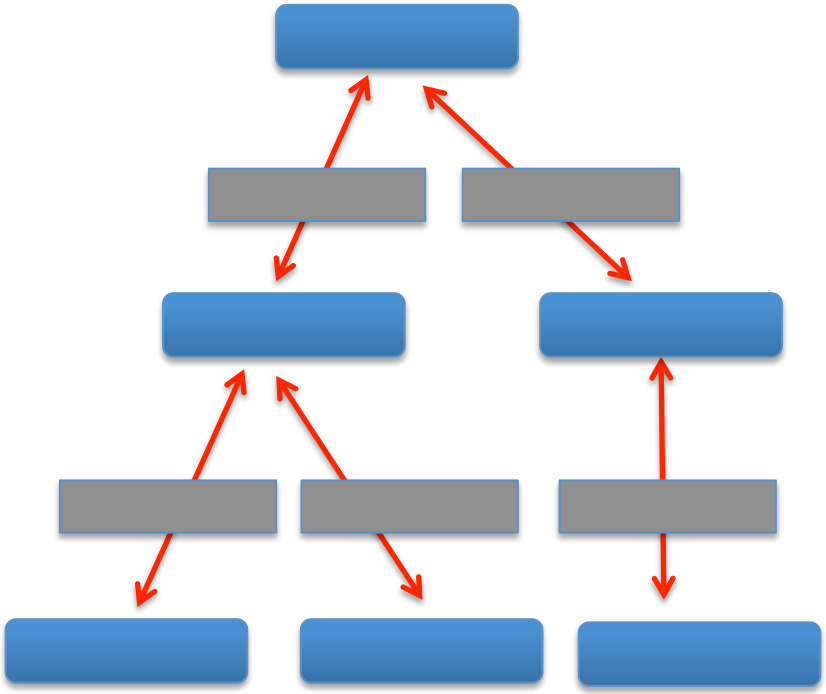


Concepts

- **Multiple scales**: time; space (1, 2, 3D),
- **Development**
- **Hierarchy of organisation** in organisms
- **Levels in hierarchy**: inherently associated with time & space scales?
- *Atomic, molecular, sub/intracellular (organelles, compartments), cellular, intracellular, tissues, organs*,…
- **Hierarchy – tree** (partially ordered upper semi-lattice).

Levels, Hierarchy





General approaches to multiscale modelling

- Models may be at many **different scales in time and space** – these **scales may be fixed or change** through the evolution of the model.

- **Operator splitting** is a powerful approach – splitting the problem into sub-problems and applying them composition-wise in some manner. This splitting generates a splitting error that depends on the stepsize.

- But **how do we split?** How do we control the splitting errors? How do we know that we converge to the right solution? Can we have a general formalism for these ideas? Can we inform different scientific communities so that both benefit.

- **Can we do parameter estimation on the fly** as we evolve from subproblem to subproblem?

Calibrating multi-scale spatial models to observations

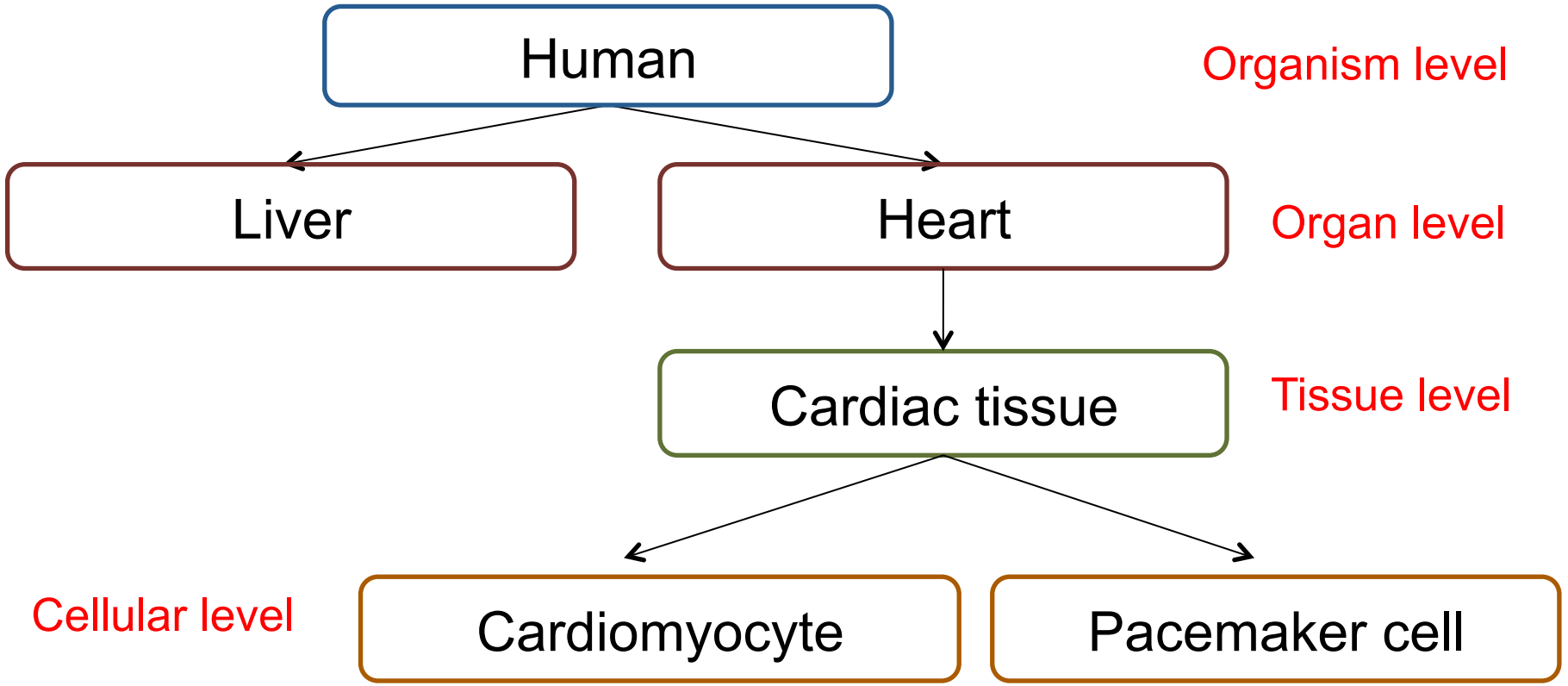
- **Calibrating spatio-temporal multi scale models to observation is hard.** Compared to non-spatial/ single level models, simulation times of spatially extended models are considerably longer, precluding exhaustive parameter sweeps and making local search algorithms impractical.

- **Spatial heterogeneity**: physical parameters of a PDE system are location dependent, so that the parameter estimation much harder function estimation problem.

- Both these **problems are compounded in the case of multi-scale system**: not only the calibration problem has to be solved for each levels, but generally the coupling between the different levels (transfer function) is also unknown.

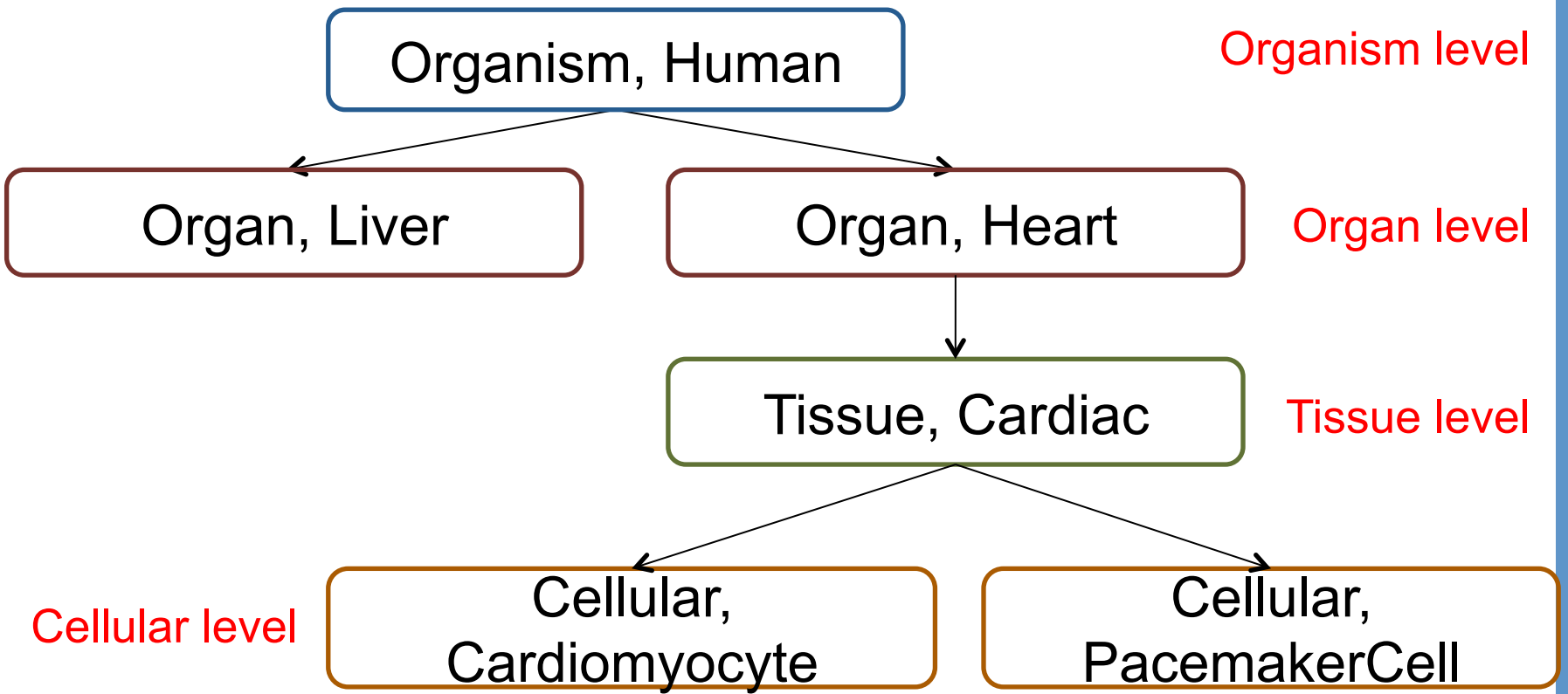
Assumption: Multiscale representation

Biological systems which are **multilevel** (i.e. span multiple levels of organization) are assumed to be inherently **multiscale** (i.e. span multiple spatio-temporal scales)



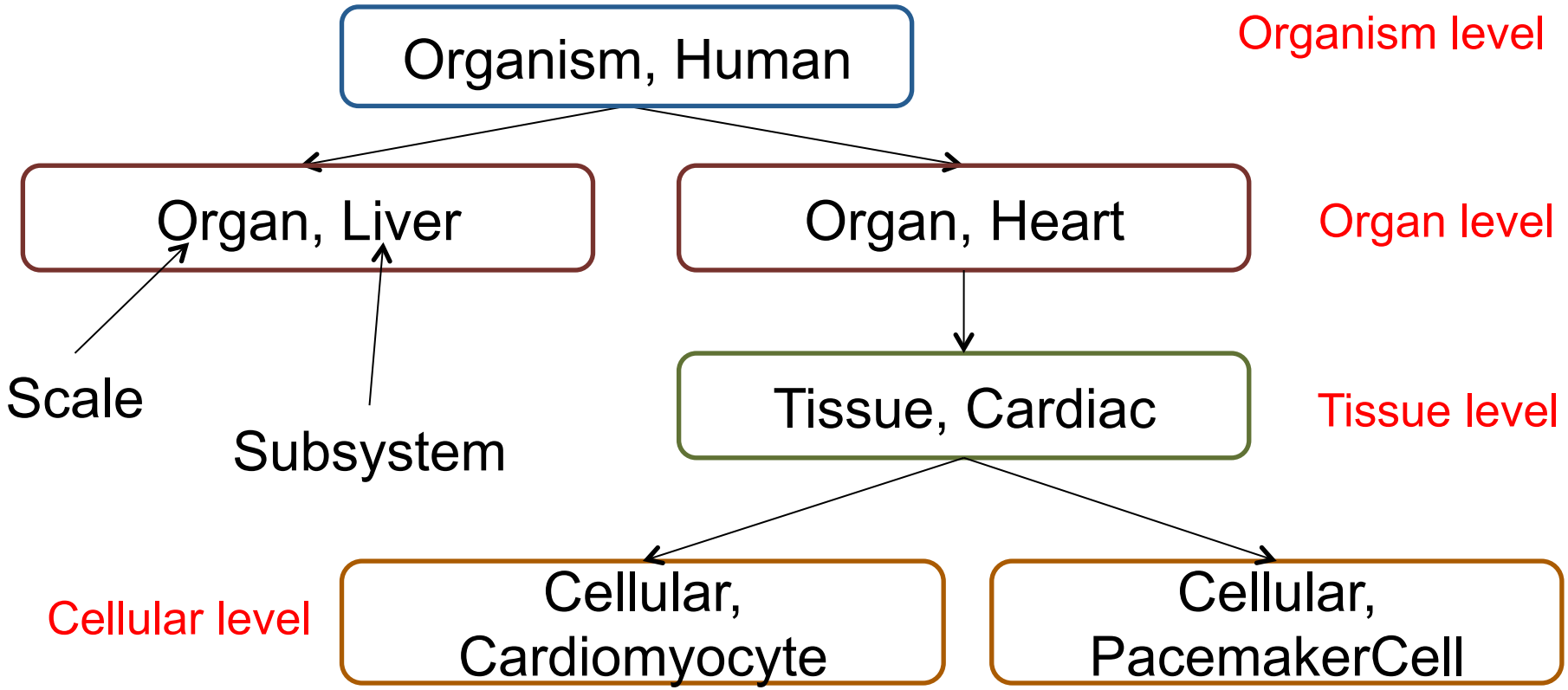
Assumption: Multiscale representation

The multiscale structure of biological systems is encoded as a rooted directed tree $MA = (V_{MA}, E_{MA})$ called the **multiscale architecture** graph



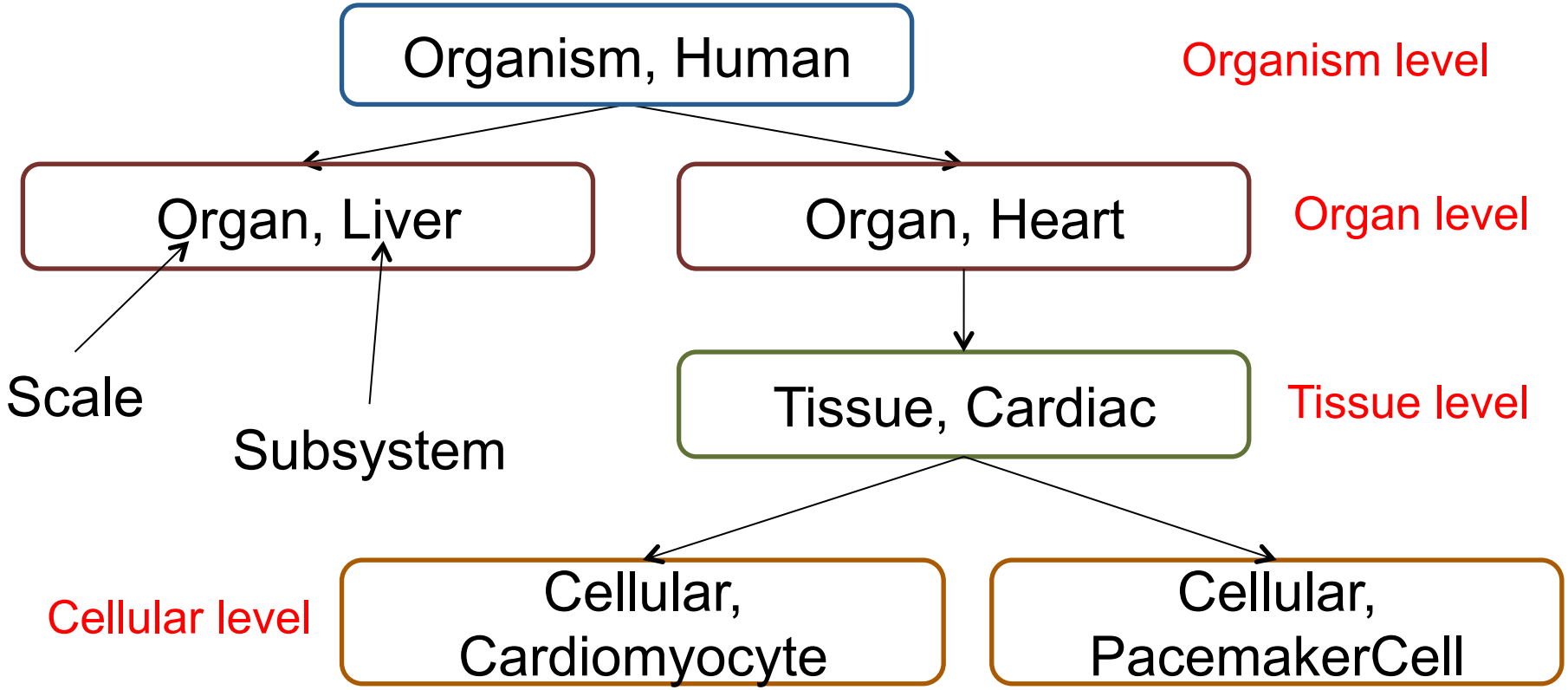
Assumption: Multiscale representation

Each vertex in the set V_{MA} is encoded as a tuple **(Scale, Subsystem)** and relations between scales and subsystems are encoded as edges in the set E_{MA}

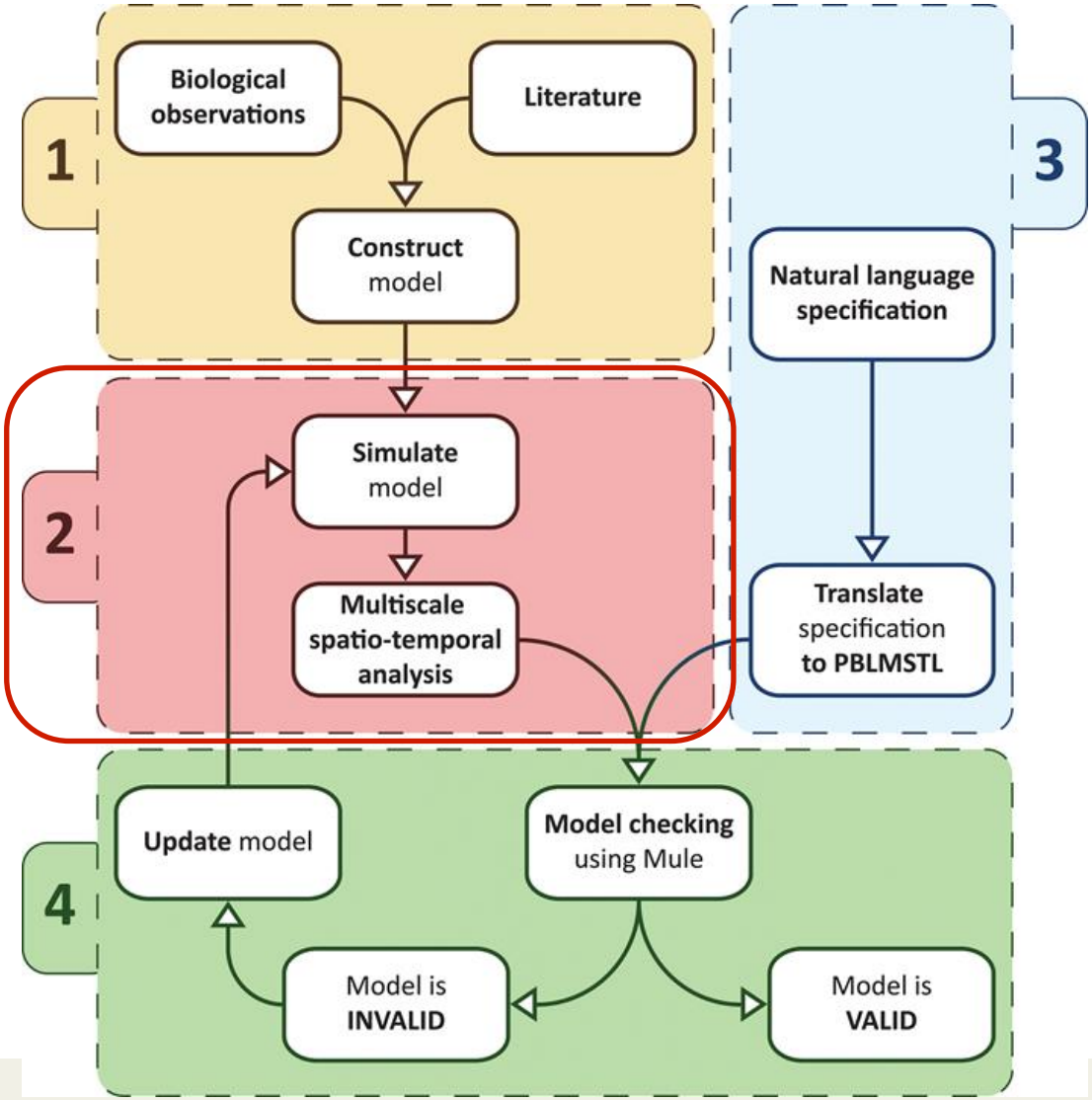


Assumption: Multiscale representation

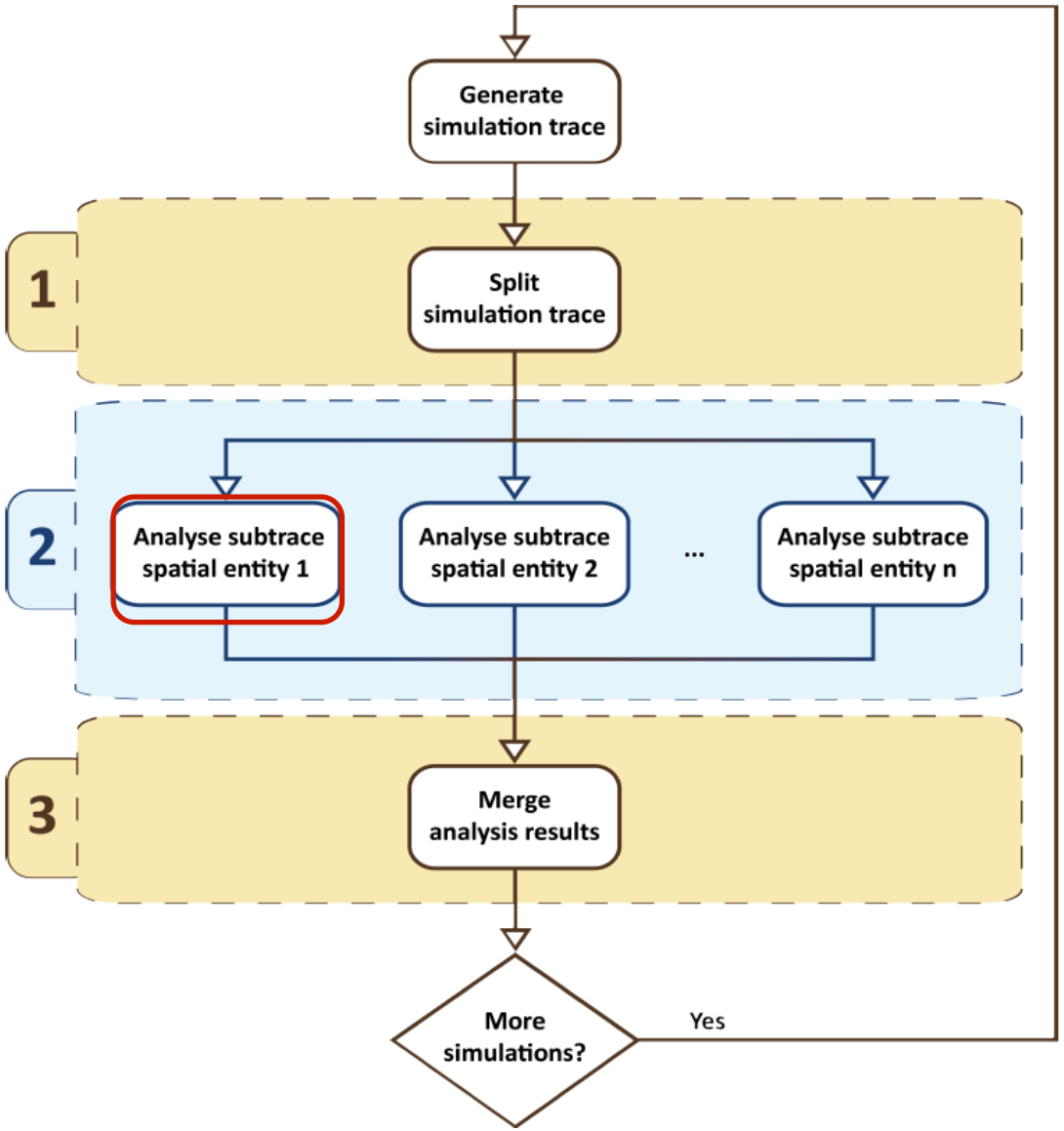
A strict **partial order <** can be defined over the set of vertices
(e.g. (Tissue, Cardiac) < (Organism, Human))



Multiscale spatio-temporal model checking

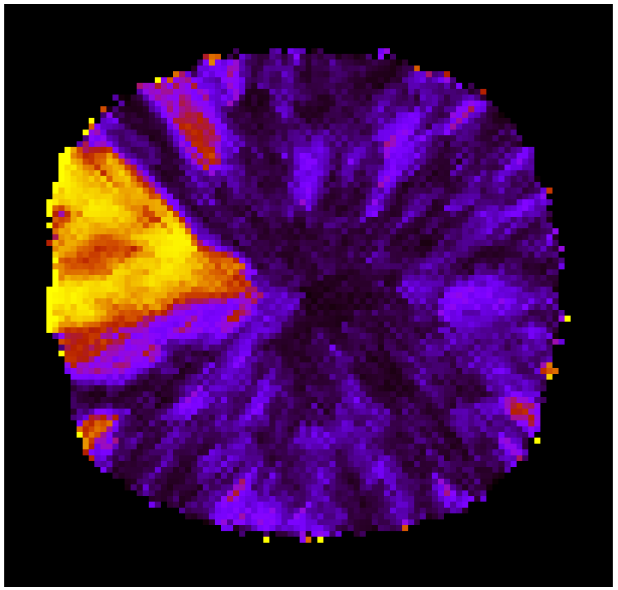


Multiscale spatio-temporal analysis

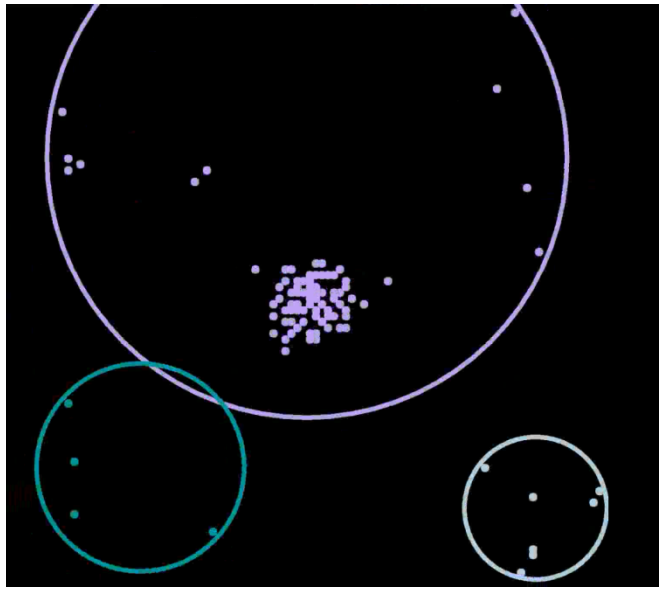


Spatio-temporal detection and analysis

Spatio-temporal detection and analysis modules enable detecting **regions** and **clusters**



Region
(e.g. bacterial colony growth)



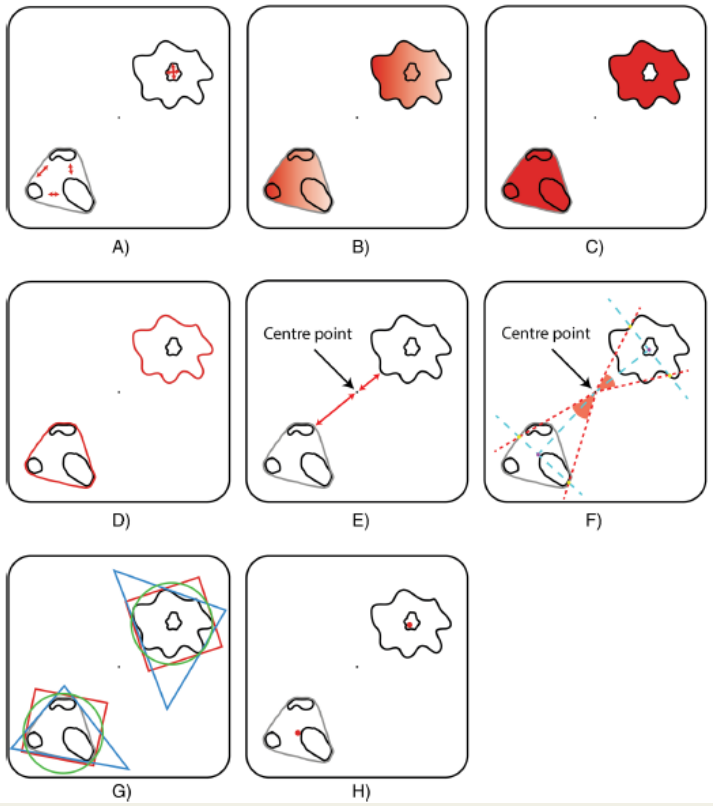
Cluster
(e.g. chemotaxis)

[2] O. Pârvu and D. Gilbert, “Automatic validation of computational models using pseudo-3D spatio-temporal model checking,” *BMC Systems Biology*, vol. 8, no. 1, p. 124, Dec. 2014

Spatio-temporal detection and analysis

For each spatial entity detected the following set of **properties** is computed:

- *Clusteredness*
- *Density*
- *Area*
- *Perimeter*
- *Distance from origin*
- *Angle*
- *Triangular, rectangular and circular measure*
- *Centroid*

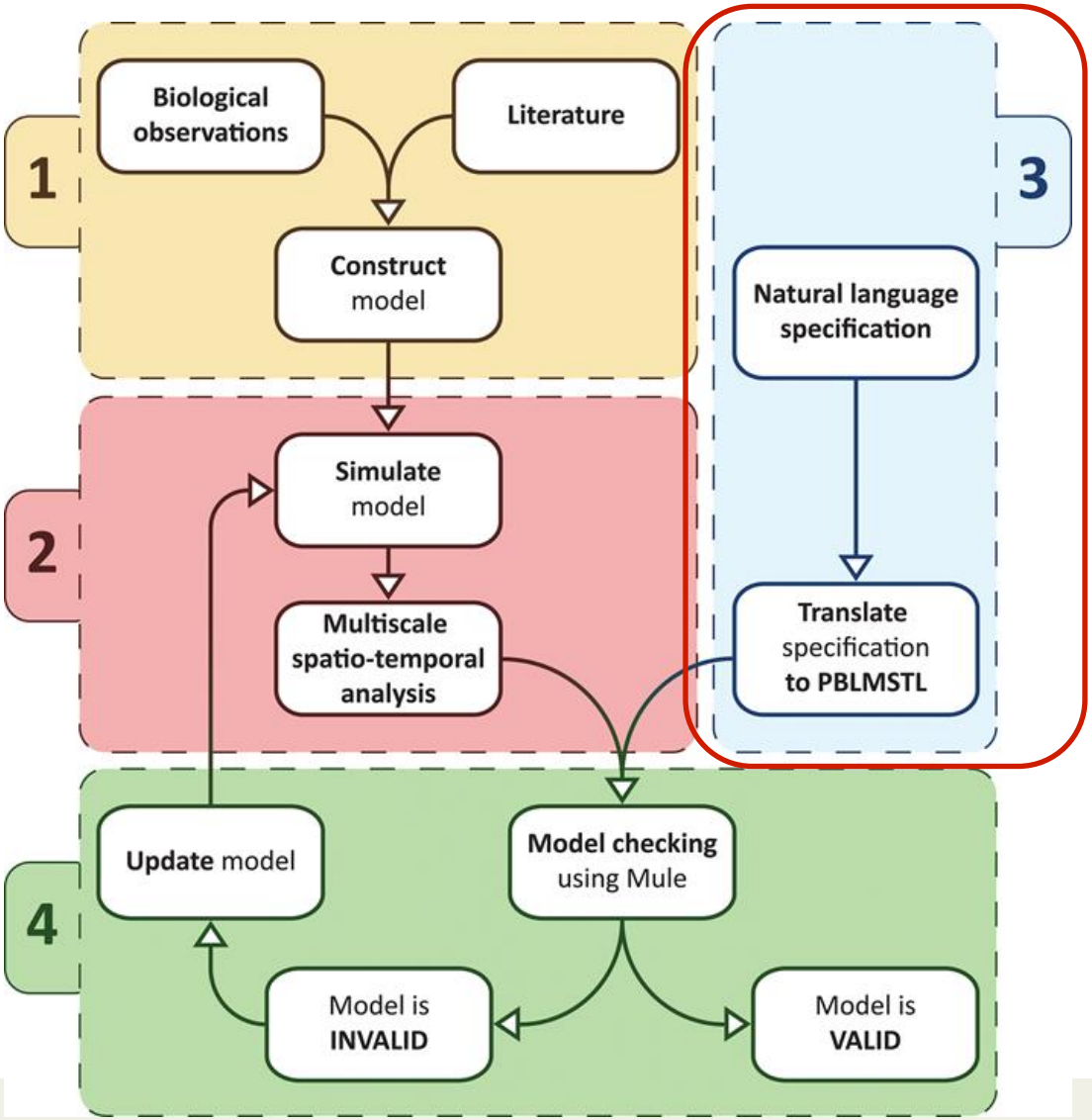


Multiscale model simulation results

Standard representation format for model simulation results: Multiscale Spatial Temporal Markup Language

```
<?xml version="1.0" encoding="utf-8"?>
<experiment>
   <timepoint>
      <spatialEntity semanticType="Organ.Heart"
          spatialType="cluster">
         <clusteredness>0.01</clusteredness>
         <density>5</density>
         <area>15</area>
         <perimeter>28</perimeter>
         <distanceFromOrigin>81</distanceFromOrigin>
         <angle>10.5</angle>
         <triangleMeasure>0.5</triangleMeasure>
         <rectangleMeasure>1.0</rectangleMeasure>
         <circleMeasure>0.1</circleMeasure>
         <centroidX>703.4999</centroidX>
         <centroidY>118.087</centroidY>
      </spatialEntity>
      <numericStateVariable semanticType="Organ.Liver">
         <name>avgClusterednessClusters</name>
   </timepoint>
     ...
</experiment>
```

Formal specification



Formal specification

Formal specification encoded in a temporal logic called **Probabilistic Bounded Linear Multiscale Spatial Temporal Logic** (PBLMSTL)

The probability is greater than 90% that always within time interval [2.1, 99.8] the liver dysfunction (corresponding to scale and subsystem (Organ, Liver)) equals the average area of damaged liver tissues (corresponding to scale and subsystem (Tissue, DamagedLiverTissue)).

```
P > 0.9 [G [2.1, 99.8] ({LiverDysfunction}
    (scaleAndSubsystem = Organ.Liver) =
    avg(area(filter(regions, scaleAndSubsystem =
    Tissue.DamagedLiverTissue))))]
```

Formal specification

Formal specification encoded in a temporal logic called **Probabilistic Bounded Linear Multiscale Spatial Temporal Logic** (PBLMSTL)

The probability is greater than 90% *that always within time interval [2.1, 99.8] the liver dysfunction (corresponding to scale and subsystem (Organ, Liver)) equals the average area of damaged liver tissues (corresponding to scale and subsystem (Tissue, DamagedLiverTissue)).*

P > 0.9 [G [2.1, 99.8] ({LiverDysfunction}
(scaleAndSubsystem = Organ.Liver) =
avg(area(filter(regions, scaleAndSubsystem =
Tissue.DamagedLiverTissue))))]

Formal specification

Formal specification encoded in a temporal logic called **Probabilistic Bounded Linear Multiscale Spatial Temporal Logic** (PBLMSTL)

*The probability is greater than 90% **that always within time interval [2.1, 99.8]** the liver dysfunction (corresponding to scale and subsystem (Organ, Liver)) equals the average area of damaged liver tissues (corresponding to scale and subsystem (Tissue, DamagedLiverTissue)).*

P > 0.9 [**G [2.1, 99.8]** ({LiverDysfunction} (scaleAndSubsystem = Organ.Liver) = avg(area(filter(regions, scaleAndSubsystem = Tissue.DamagedLiverTissue))))]

Formal specification

Formal specification encoded in a temporal logic called **Probabilistic Bounded Linear Multiscale Spatial Temporal Logic** (PBLMSTL)

The probability is greater than 90% that always within time interval [2.1, 99.8] ***the liver dysfunction (corresponding to scale and subsystem (Organ, Liver))*** *equals the average area of damaged liver tissues (corresponding to scale and subsystem (Tissue, DamagedLiverTissue)).*

P > 0.9 [G [2.1, 99.8] (**{LiverDysfunction} (scaleAndSubsystem = Organ.Liver)** = avg(area(filter(regions, scaleAndSubsystem = Tissue.DamagedLiverTissue))))]

Formal specification

Formal specification encoded in a temporal logic called **Probabilistic Bounded Linear Multiscale Spatial Temporal Logic** (PBLMSTL)

*The probability is greater than 90% that always within time interval [2.1, 99.8] the liver dysfunction (corresponding to scale and subsystem (Organ, Liver)) **equals** the average area of damaged liver tissues (corresponding to scale and subsystem (Tissue, DamagedLiverTissue)).*

P > 0.9 [G [2.1, 99.8] ({LiverDysfunction}
(scaleAndSubsystem = Organ.Liver) **=**
avg(area(filter(regions, scaleAndSubsystem =
Tissue.DamagedLiverTissue))))]

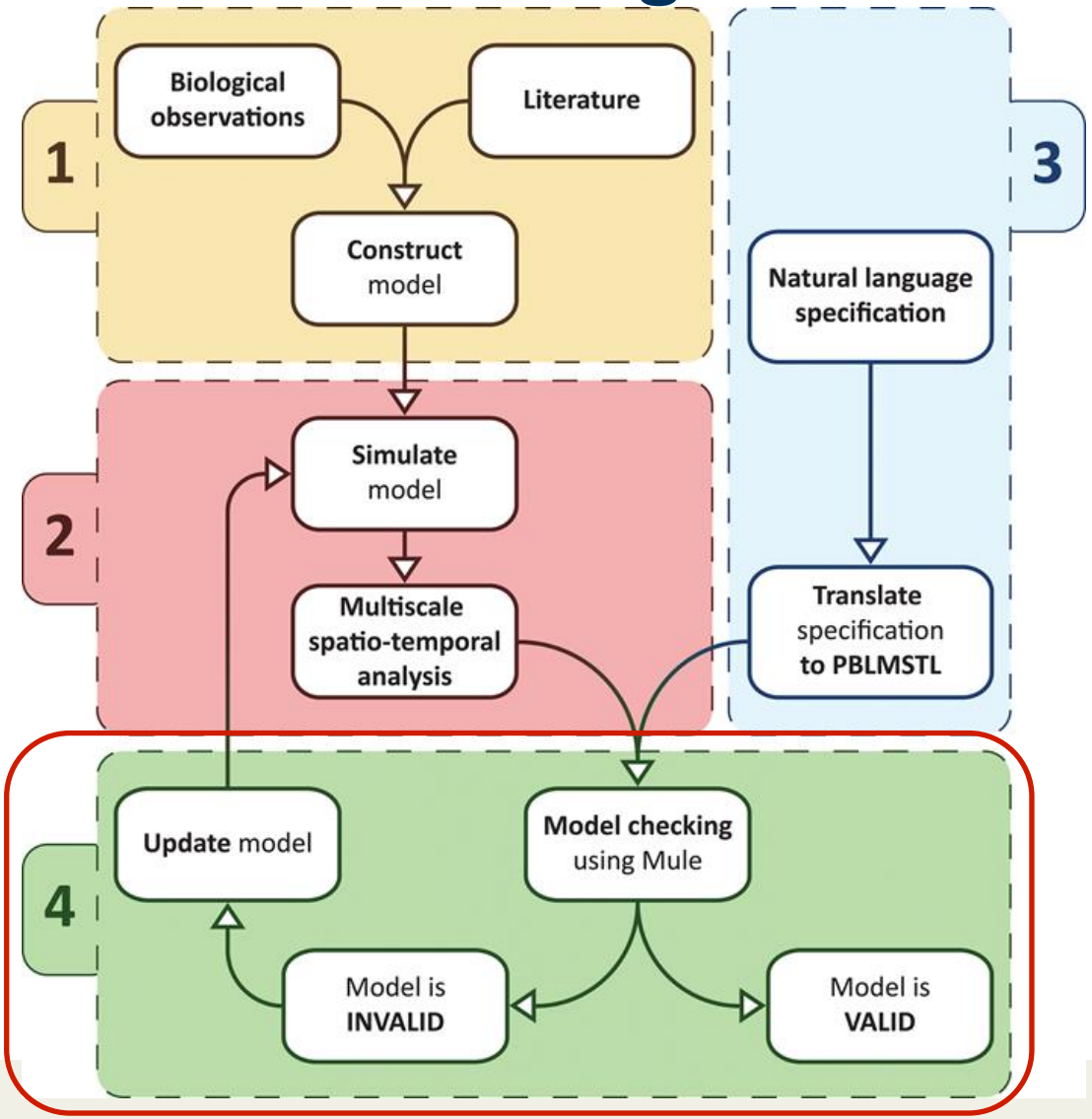
Formal specification

Formal specification encoded in a temporal logic called **Probabilistic Bounded Linear Multiscale Spatial Temporal Logic** (PBLMSTL)

The probability is greater than 90% that always within time interval [2.1, 99.8] the liver dysfunction (corresponding to scale and subsystem (Organ, Liver)) equals ***the average area of damaged liver tissues (corresponding to scale and subsystem (Tissue, DamagedLiverTissue))****.*

P > 0.9 [G [2.1, 99.8] ({LiverDysfunction}
(scaleAndSubsystem = Organ.Liver) =
avg(area(filter(regions, scaleAndSubsystem = Tissue.DamagedLiverTissue))))]

Multiscale spatio-temporal model checking



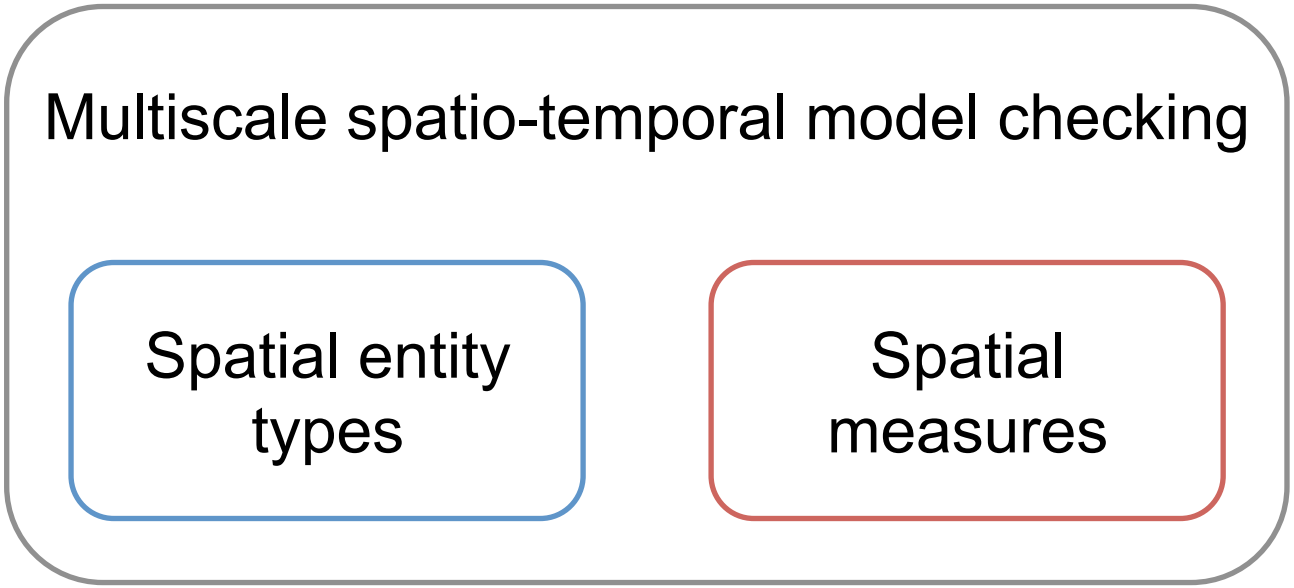
Multiscale spatio-temporal model checking

Approximate probabilistic model checking algorithms supported:

	Frequentist	**Bayesian**
Estimate	Based on Chernoff-Hoeffding bounds	Based on mean and variance
Hypothesis testing	Statistical, Probabilistic black-box	Statistical

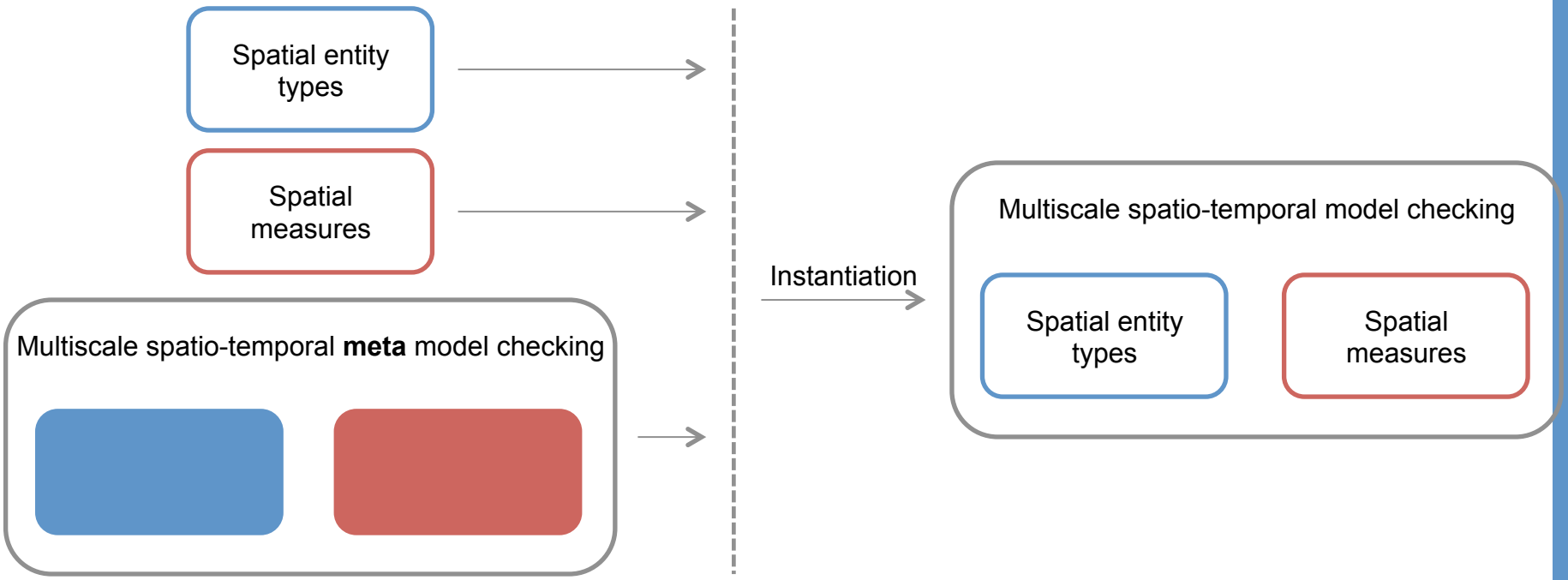
Multiscale spatio-temporal meta model checking

Multiscale spatio-temporal model checking approach as defined so far is **restricted** to particular pseudo-3D **spatial entity types** (e.g. region) and **spatial measures** (e.g. area)

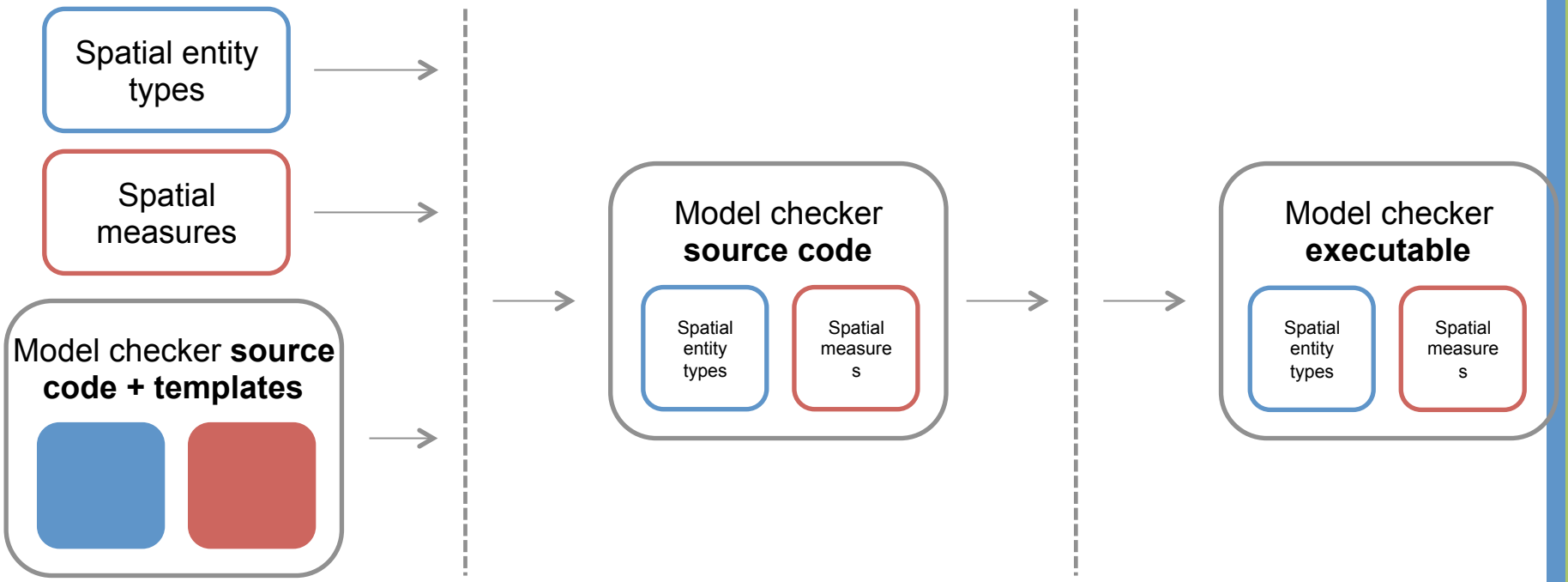


Multiscale spatio-temporal meta model checking

Multiscale spatio-temporal **meta** model checking: Define a generic **family** of multiscale spatio-temporal model checkers that can be **instantiated** for specific spatial entity types and spatial measures



Implementation



Compilation step 1
(Generate source code from templates)

Compilation step 2
(Compile source code into executable)

Multiscale spatio-temporal meta model checker

The multiscale spatio-temporal meta model checker **Mule** is made **freely available online** (binary, source code, Docker image) at http://mule.modelchecking.org

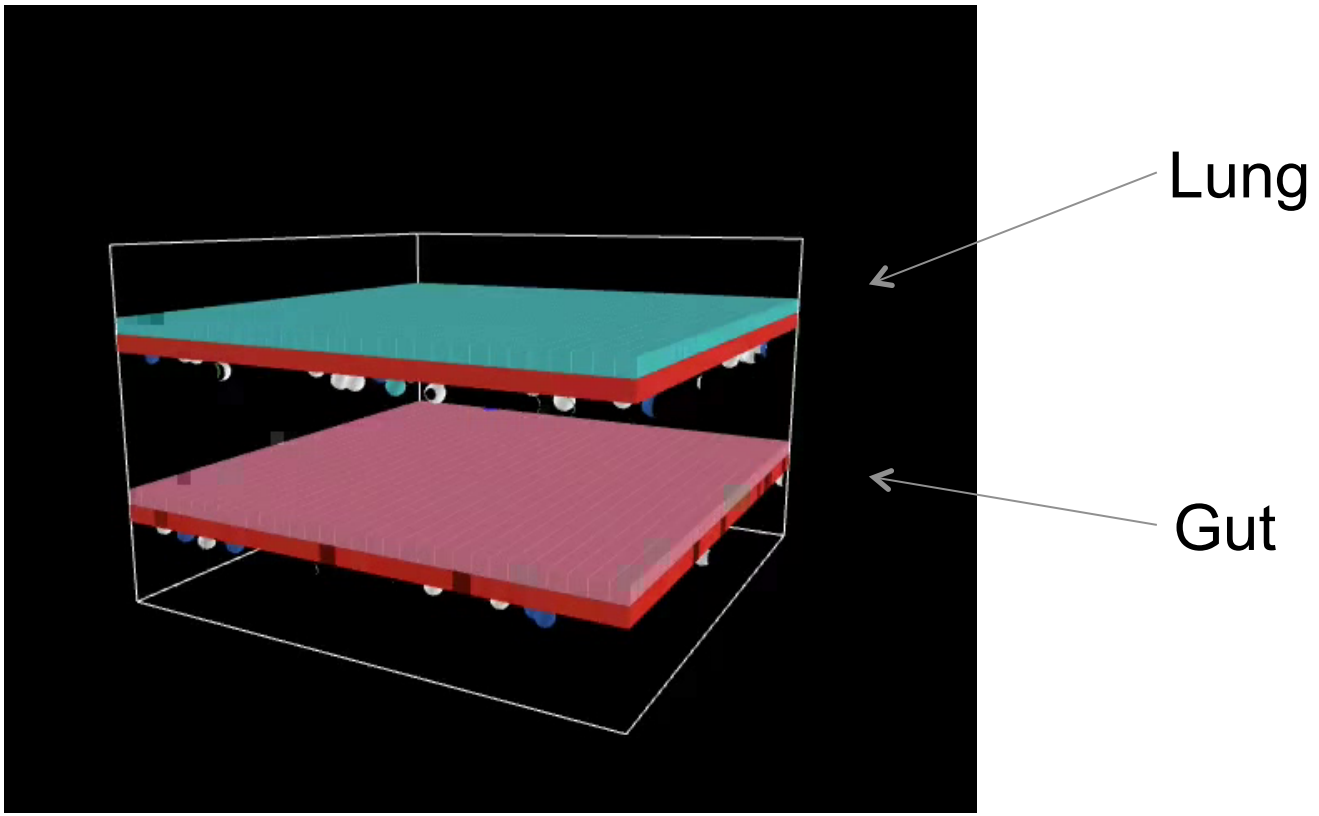
Tutorials: http://mule.modelchecking.org/tutorials

Case studies

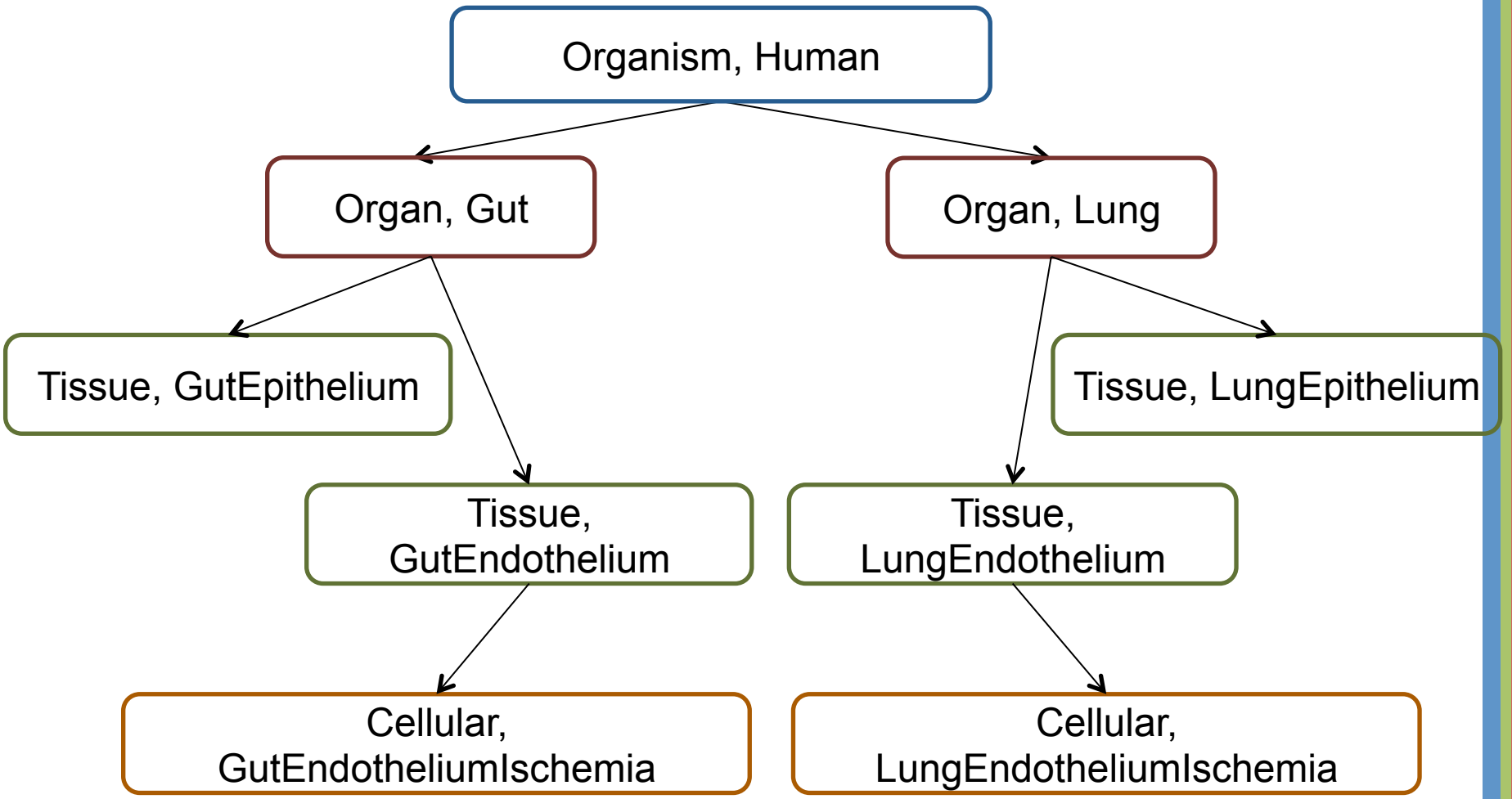
Case studies considered:

- Rat cardiovascular system dynamics [3]
- Uterine contractions of labour [4]
- *Xenopus laevis* oocytes cell cycle [5, 6]
- Acute inflammation of gut and lung [7].

Case study: Acute inflammation of gut and lung



Case study: Acute inflammation of gut and lung



Case study: Acute inflammation of gut and lung

Formal specification – statement 1:

The probability is greater than 0.9 that if the value of cytoplasm occludin in the lung (corresponding to scale and subsystem Tissue.LungEpithelium) decreases then eventually the total area of the regions defined by ischemic endothelial lung cells (corresponding to scale and subsystem Cellular.LungEndotheliumIschemia) will increase.

```
P > 0.9 [F [1, 999] ((d({LungOccludinCytoplasm}
    (scaleAndSubsystem = Tissue.LungEpithelium)) < 0) =>
    (F [1, 999] (d(sum(area(filter(regions,
    scaleAndSubsystem =
    Cellular.LungEndotheliumIschemia)))) > 0)))]
```

Case study: Acute inflammation of gut and lung

Formal specification – statement 2:
The probability is greater than 0.9 that always if the value of the gut cell damage by-product (corresponding to scale and subsystem Tissue.GutEndothelium) increases, then eventually the value of the lung cell damage by-product (corresponding to scale and subsystem Tissue.LungEndothelium) increases.

```
P > 0.9 [G [1, 999] ((d({GutCellDamageByproduct}
    (scaleAndSubsystem = Tissue.GutEndothelium)) > 0) =>
    (F [1, 999] (d({LungCellDamageByproduct}
    (scaleAndSubsystem = Tissue.LungEndothelium)) > 0)))]
```

Case study: Acute inflammation of gut and lung

Formal specification – statement 3:
The probability is greater than 0.9 that if the value of the gut cell wall occludin (corresponding to scale and subsystem Tissue.GutEpithelium) decreases then eventually the value of the gut leak (corresponding to scale and subsystem Organ.Gut) will increase.

```
P > 0.9 [F [1, 999] ((d({GutOccludinCellwall}
    (scaleAndSubsystem = Tissue.GutEpithelium)) < 0) =>
    (F [1, 999] (d({GutLeak} (scaleAndSubsystem =
    Organ.Gut)) > 0)))]
```

Conclusions

Our multiscale spatio-temporal meta model checking approach will enable computational biologists to **efficiently construct reliable** multiscale computational models of biological systems

In the future we would like to employ the meta model checking approach for analysing **real life** data sets, and to verify computational models from **other domains of science**

Contact details:

Professor David Gilbert (david.gilbert@brunel.ac.uk),
Ovidiu Pârvu (ovidiu.parvu@gmail.com), ovidiuparvu.com

Bibliography

[1] O. Pârvu and D. Gilbert, “A novel method to verify multilevel computational models of biological systems using multiscale spatio-temporal meta model checking,” *PLoS ONE* (under review).

[2] O. Pârvu and D. Gilbert, “Automatic validation of computational models using pseudo-3D spatio-temporal model checking,” *BMC Systems Biology*, vol. 8, no. 1, p. 124, Dec. 2014.

[3] D. A. Beard, M. L. Neal, N. Tabesh-Saleki, C. T. Thompson, J. B. Bassingtwaighte, M. Shimoyama, and B. E. Carlson, “Multiscale Modeling and Data Integration in the Virtual Physiological Rat Project,” *Ann Biomed Eng*, vol. 40, no. 11, pp. 2365–2378, Nov. 2012.

[4] R. C. Young and P. Barendse, “Linking Myometrial Physiology to Intrauterine Pressure; How Tissue-Level Contractions Create Uterine Contractions of Labor,” *PLoS Comput Biol*, vol. 10, no. 10, p. e1003850, Oct. 2014.

[5] J. E. Ferrell Jr., T. Y.-C. Tsai, and Q. Yang, “Modeling the Cell Cycle: Why Do Certain Circuits Oscillate?,” *Cell*, vol. 144, no. 6, pp. 874–885, Mar. 2011.

[6] J. Starruß and W. de Back, “Morpheus examples,” *Morpheus*. [Online]. Available: http://imc.zih.tu-dresden.de/wiki/morpheus/doku.php?id=examples:examples. [Accessed: 20-Sep-2015].

[7] G. An, “Introduction of an agent-based multi-scale modular architecture for dynamic knowledge representation of acute inflammation,” *Theoretical Biology and Medical Modelling*, vol. 5, no. 1, p. 11, May 2008.

[8] O. Pârvu, “Computational model validation using a novel multiscale multidimensional spatio-temporal meta model checking approach,” PhD Thesis, Brunel University London, London, UK, (to be examined).

[9] O. Pârvu, D. Gilbert, M. Heiner, F. Liu, N. Saunders, and S. Shaw, “Spatial-Temporal Modelling and Analysis of Bacterial Colonies with Phase Variable Genes,” *ACM Trans. Model. Comput. Simul.*, vol. 25, no. 2, pp. 13:1–13:25, May 2015.

[10] O. Pârvu and D. Gilbert, “Implementation of linear minimum area enclosing triangle algorithm,” *Comp. Appl. Math.*, pp. 1–16, Nov. 2014.

[11] O. Pârvu, D. Gilbert, M. Heiner, F. Liu, and N. Saunders, “Modelling and Analysis of Phase Variation in Bacterial Colony Growth,” in *Computational Methods in Systems Biology*, A. Gupta and T. A. Henzinger, Eds. Springer Berlin Heidelberg, 2013, pp. 78–91.

EPSRC

BBSRC
bioscience for the future

{david.gilbert,ovidiu.p
arvu}@brunel.ac.uk

Technologies

Acknowledgements

Brandenburg
University of Technology
Cottbus - Senftenberg

Monika Heiner
Christian Rohr
Martin Schwarick

Brunel
University
London

Pam Gao
Crina Grosan
Nigel Saunders
David Tree

1920
HIT

Fei Liu

OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

Mary Ann Blätke
Wolfgang Marwan
Jan Wegener

UNIVERSITA' DEGLI STUDI DI MILANO
BICOCCA

Daniele
Maccagnola
Enza Messina

PORT SAID UNIVERSITY

Mostafa Herayj

Acknowledgements

Brunel

Pam Gao

Crina Grosan

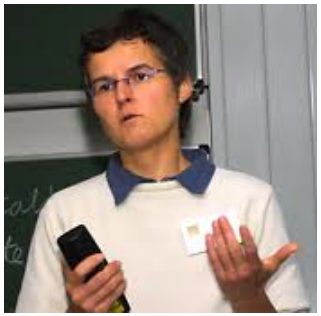
Ovidiu Parvu

Nigel Saunders

David Tree

Cottbus

Monika Heiner



Milan

Daniele Maccagnola

Enza Messina

China

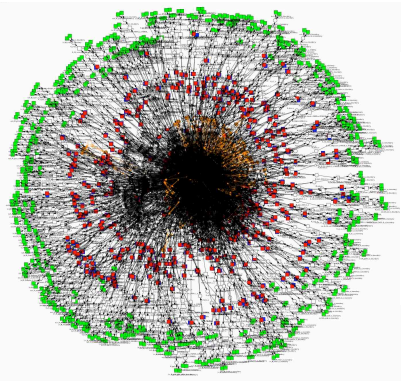
Zujian Wu

Synthetic Biology Computational Design Group

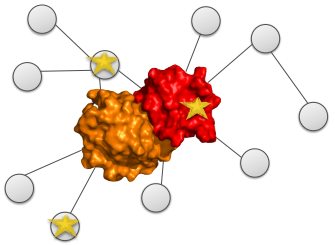
Network-based analysis and design & Protein structure informed design

David Gilbert, Bello Suleiman, Monika Heiner, Alessandro Pandini, Arshad Khan, Nigel J Saunders
CSSM and MBE Clusters, Synthetic Biology Theme, Brunel University London

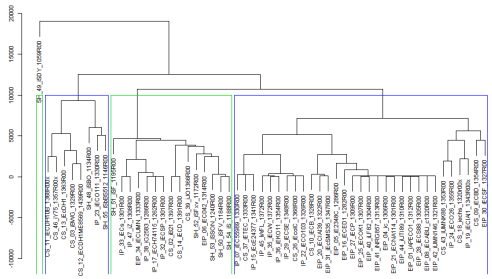
- Development and application of **methodologies for the design of novel microbacterial strains**, collaboration with Microbial BioEngineering Group.
- **Biochemical pathway models as design 'templates'** - guidelines for bioengineering implementation.
- Biochemical reactions of bacterial strains as **systems of continuous or stochastic equations**, bipartite graph structure - Petri net.
- **Model construction**, **analysis** and **modification**.
- **Analysis: static and dynamic properties** - sound and consistent; checked against observations of the bacteria that they describe.
- **Modelling database** to store components from public domain models of bacteria as well as locally generated data which can be reused for model construction. Include phenotypic annotation and direct links to public molecule and reaction databases. **Integration with the database of the Brunel Strain Collection** will facilitate implementation of the process from *in silico* design to *in vitro* / *in vivo* experiments. Extend from gene data to include proteins and metabolites.
- The generation of designs for new synthetic bacterial strains involves the **selection of optimal combinations of chassis (host strain) and genes for transfer, knockout or modification**.
- **Computational analysis will define design solutions over multiscale levels from polymorphism to protein and pathway modules.**
- david.gilbert@brunel.ac.uk
- www.brunel.ac.uk/people/david-gilbert



Whole genome metabolic model, *E.coli K12* for simulation



Protein structure mapped on Protein-Protein Interaction Network



Supervised learning over set of whole genome simulations

Funding: Brunel University – College of Engineering, Design and Physical Sciences