Introduction to Agent-Based Modelling

Outline

- From Cellular Automata to Agent-Based Modelling
- What is an Agent-Based Model
- Properties of Agent-Based Models
- Classification of Agent-Based Models

FROM CELLULAR AUTOMATA TO AGENT-BASED MODELLING

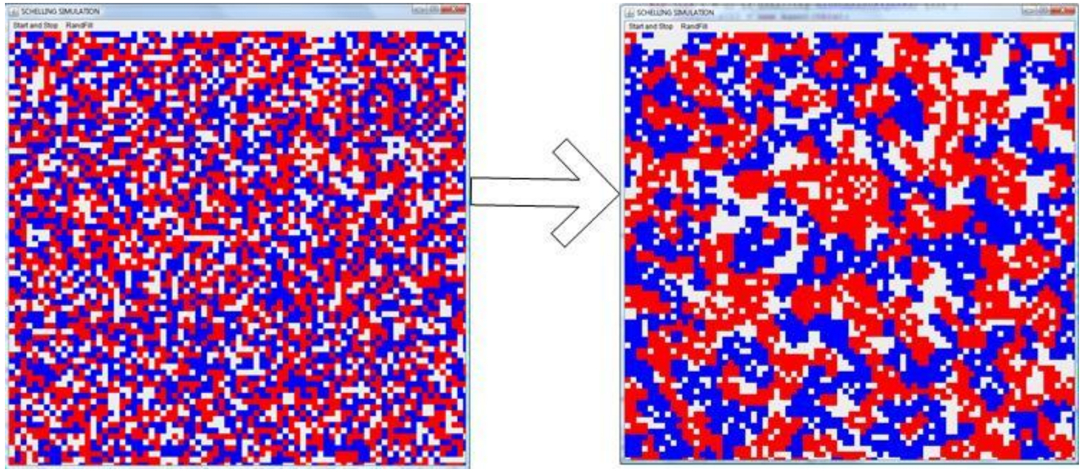
Historical Background

- Agent-based modelling is a comparably young modelling technique.
- Were inspired by Cellular Automata (Von Neumann, Ulam, etc)
- Thomas Schelling‘s Model of Segregation (1971) is broadly denoted as the first agent-based model

Model segregation behaviour between individuals
with different races in US in the 1970s

Historical Background

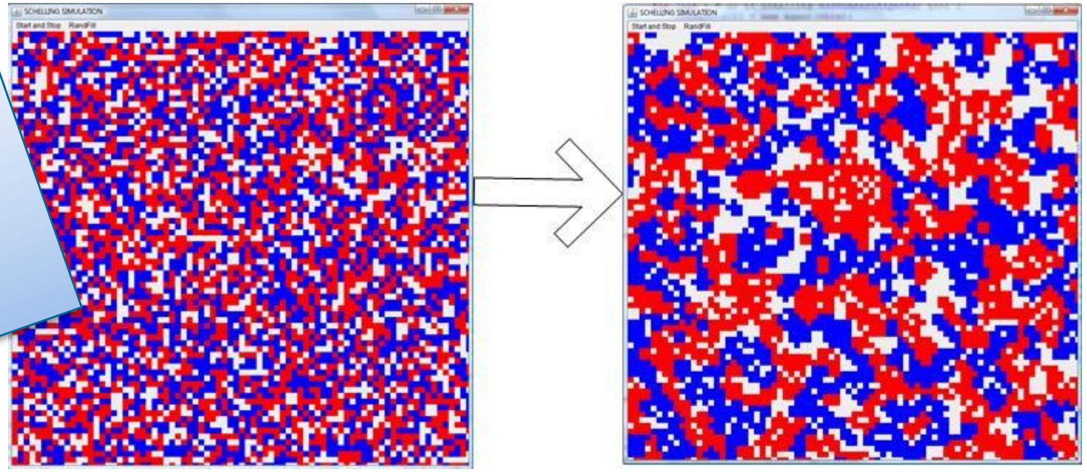
- Agent-based modelling is a comparably young modelling technique.
- Were inspired by Cellular Automata (Von Neumann, Ulam, etc)
- Thomas Schelling‘s Model of Segregation (1971) is broadly denoted as the first agent-based model



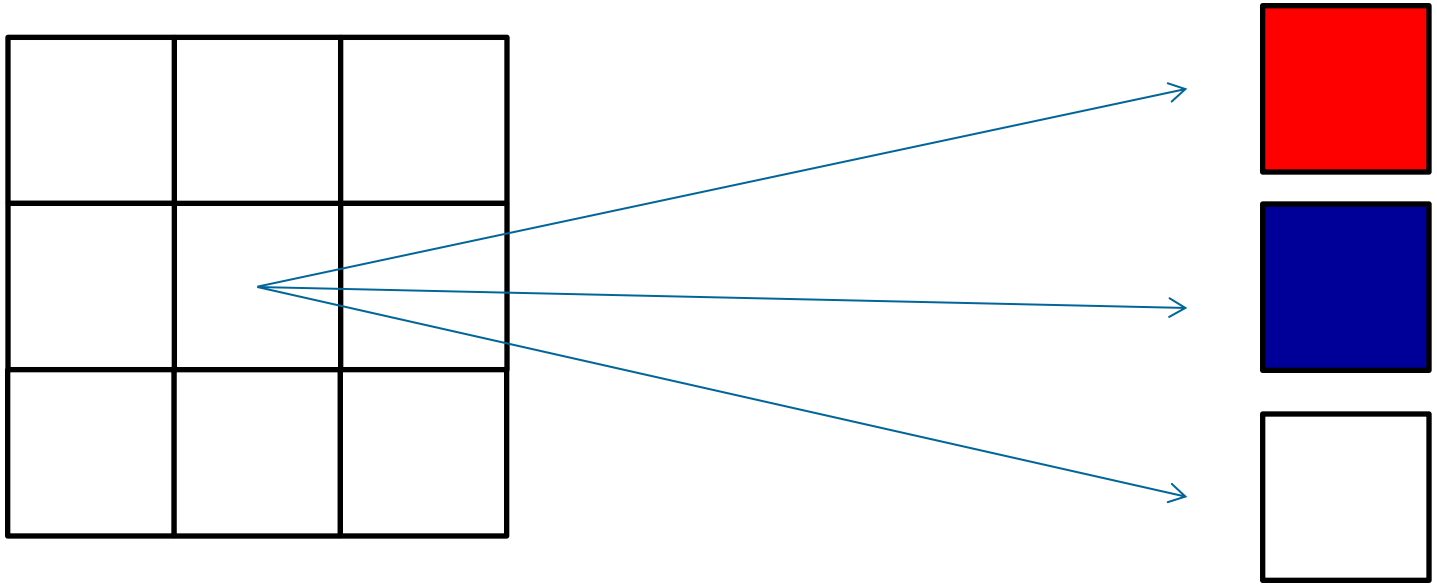
Historical Background

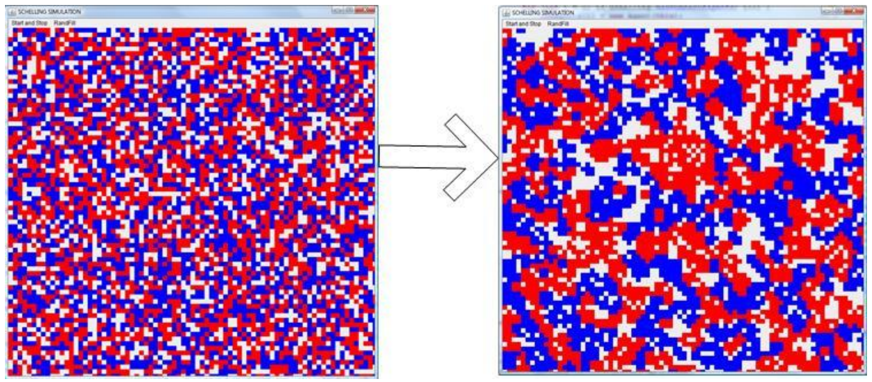
- Agent-based modelling is a comparably young modelling technique.
- Were inspired by Cellular Automata (Von Neumann, Ulam, etc)
- Thomas Schelling‘s Model of Segregation (1971) is broadly denoted as the first agent-based model

Still Looks very much like a CA model?!



A Small but Powerful Difference…

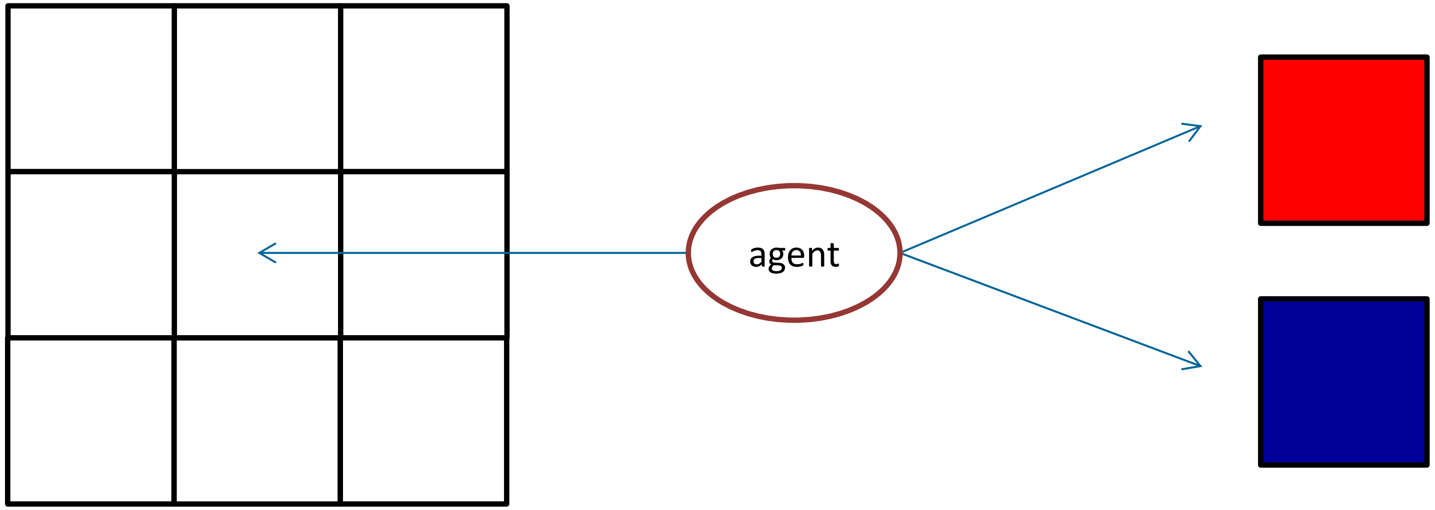


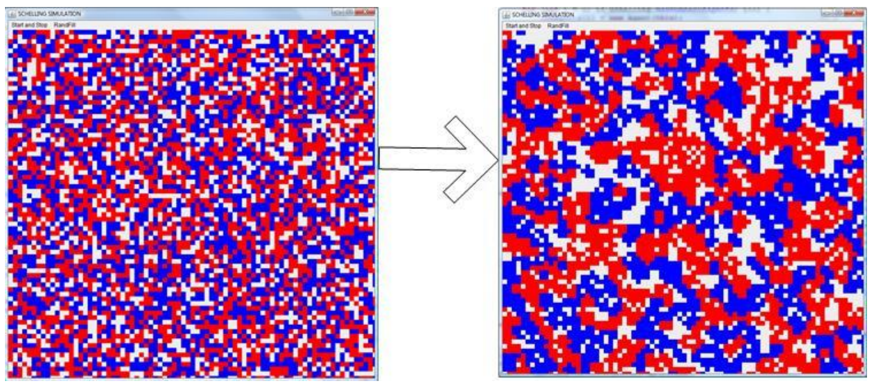


CA Model

Each cell is assigned a colour
(= a person if colour is not white)

A Small but Powerful Difference…





Agent Based Model (ABM)

Each agent (= person) is assigned a colour (blue or red) ***and*** a cell

A Small but Powerful Difference…

In principle both representations make sense for this application. Yet Schelling used the second concept to describe the model for its benefits.

CA Model	ABM

for C in Cellspace:
    if C is not white:
        N(C) = neighbourhood of C
        do update rules with C w.r.t. to N(C)
Update Cellspace
	for A in AgentList:
	    get cell and color of A
	    find neighboured agents N(A)
	    do some actions with A w.r.t. to N(A)
	Update AgentList

Pseudocode representation of a time step in Schelling‘s model.

A Small but Powerful Difference…

In principle both representations make sense for this application. Yet Schelling used the second concept to describe the model for its benefits.

It is easier to explain the model as it is a more natural description!

We do not have to use a dicrete time-step!

It could be some „grayscales“ in between if we want to

ABM

We could distinguish between male and female agents (persons)

```
for A in AgentList:
    get cell and color of A
    find neighboured agents N(A)
    do some actions with A w.r. to N(A)
Update AgentList
```

We do not have to use a cell-space

We could include more realistic distributions

We could add some immigrants

We could introduce death of agents

WHAT IS AN AGENT-BASED MODEL

What is an Agent?

Why Agent?

What is an Agent?

Latin: „agere“ (to act)

What is an Agent?

- Agent – lat. agere (act)
- There is no unique definition. The word is very broadly used.

[Agent-based modelling is...]

„Rather a general concept“

(Winter Simulation Conference 2005 & 2006)

What is an Agent?

- With respect to Winter Simulation Conference (2005 & 2006) an agent has to...

  ... **be uniquely identifiable**
  ... **cohabitate an environment with other agents, and has to be able to communicate with them.**
  ... **be able to act targeted.**
  ... **be autonomous and independent.**
  ... **be able to change its behaviour.**

What is an Agent?

- With respect to Winter Simulation Conference (2005 & 2006) an agent has to...

  ... be uniquely identifiable
  ... cohabitate an environment with other agents, and has to be able to communicate with them.
  ... be able to act targeted.
  ... be autonomous and independent.
  ... be able to change its behaviour.

Optional properties (Wintersimulation Conference 2015)

Short Summary

- Agent-Based modelling is a bottom up modelling approach using a big number of individual system components (agents).
- The components act independently (following given rules)
- As it requires a lot of processing resources ABM is a very young science with high potential.

PROPERTIES OF AGENT-BASED MODELS

Properties of Agent-Based Models

(Bonabeau, 2002)

a. Representation of „emergent phenomena“
b. Flexibility
c. Natural description of the system

Properties of Agent-Based Models

(Bonabeau, 2002)

a. Representation of „emergent phenomena“
b. Flexibility
c. Natural description of the system

Representation of „Emergent Phenomena“

Simple rules for individual agents

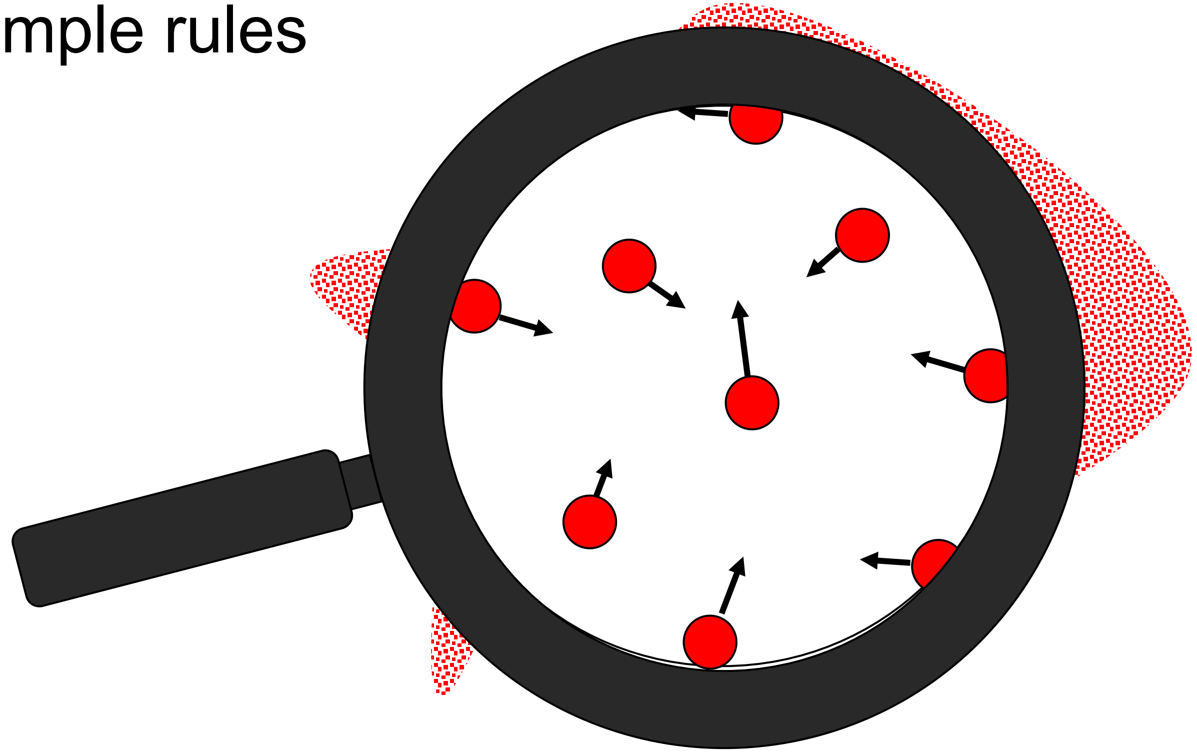
Complex dynamics of the whole system

⬇

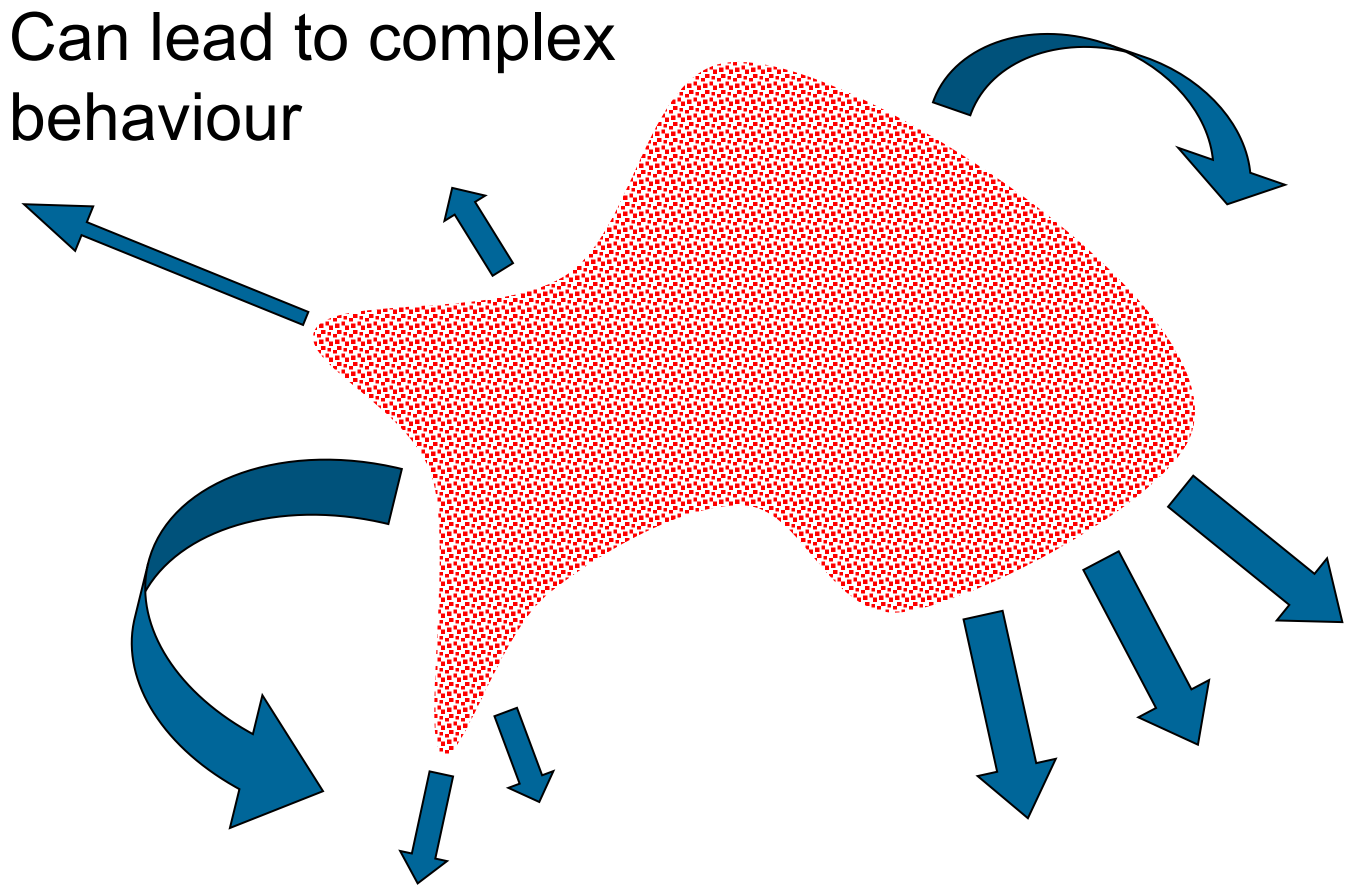
group dynamics / swarm intelligence

Representation of „Emergent Phenomena“

Simple rules



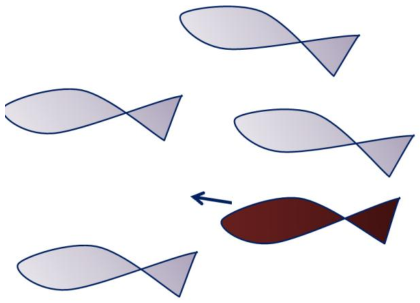
Representation of „Emergent Phenomena“



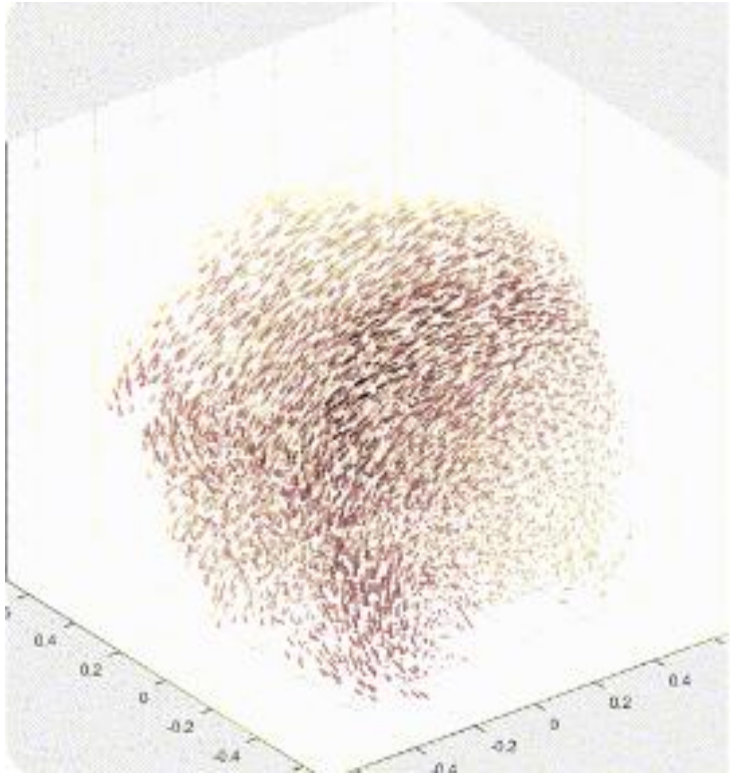
Representation of „Emergent Phenomena“

https://www.youtube.com/watch?v=QOGCSBh3kmM

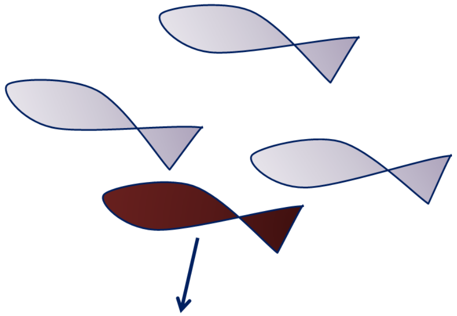
Boids Flock Model

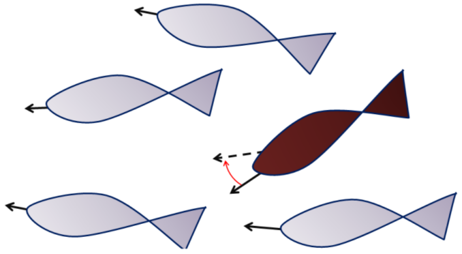


Each agent tends towards
the centre of its neighbours



Keep a distance that is
neither too far nor too small





Swim in the same direction
as your neighbours

Properties of Agent-Based Models

(Bonabeau, 2002)

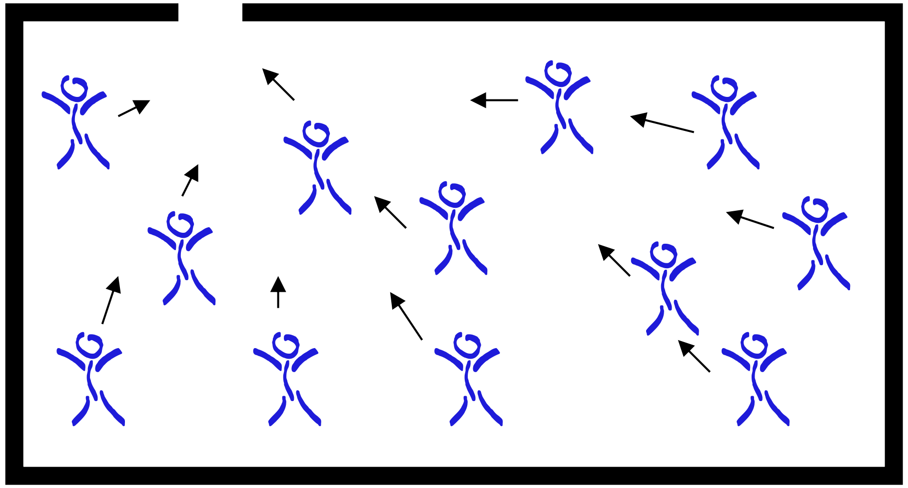
a. Representation of „emergent phenomena“
b. Flexibility
c. Natural description of the system

Flexibility

- Change of details is very easy compared to other (especially macroscopic) modelling approaches.
- Different parameterisation of single agents does not require changes within the system structure.
- Change or addition of (meta) rules for single agents does not influence the system structure as well (as long as they remain compatible with the system).

Example: Emergency exit strategy

Example: Emergency exit strategy



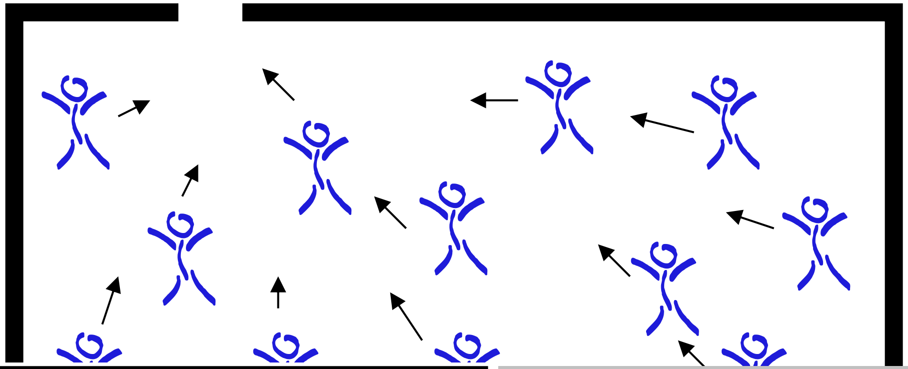
Agent-Based Model

Macroscopic approach

Example: Emergency exit strategy

Agent-Based Model

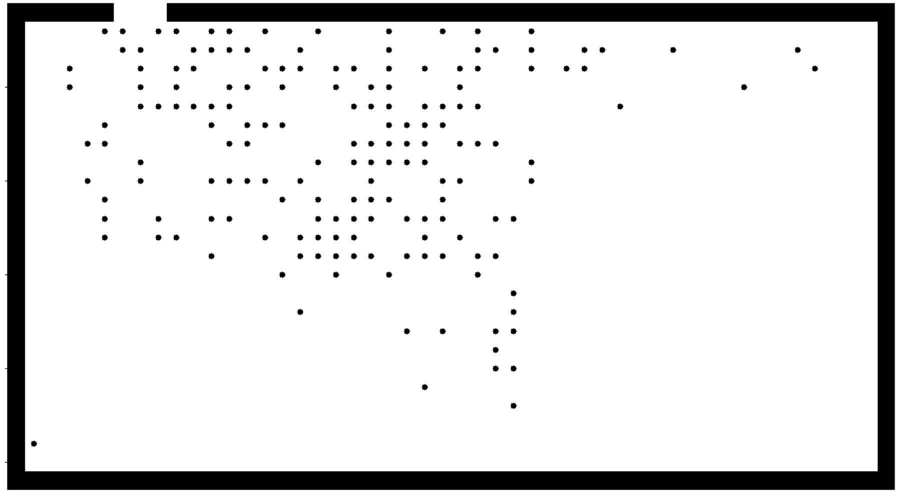
Easy

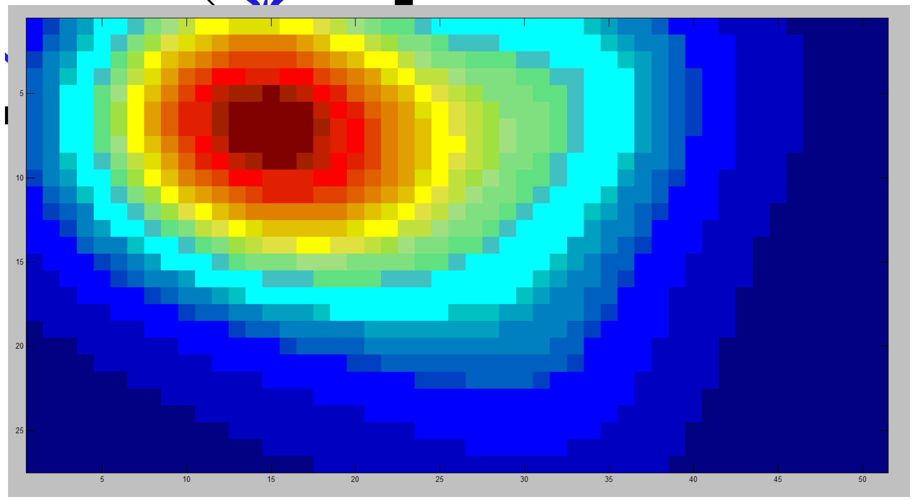


Macroscopic approach

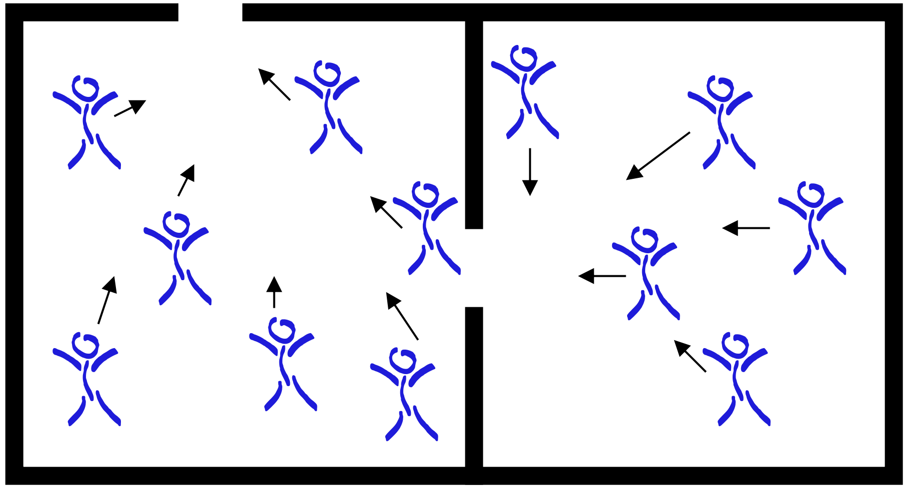
Easy

(Navier Stokes PDE Based Model)





Example: Emergency exit strategy



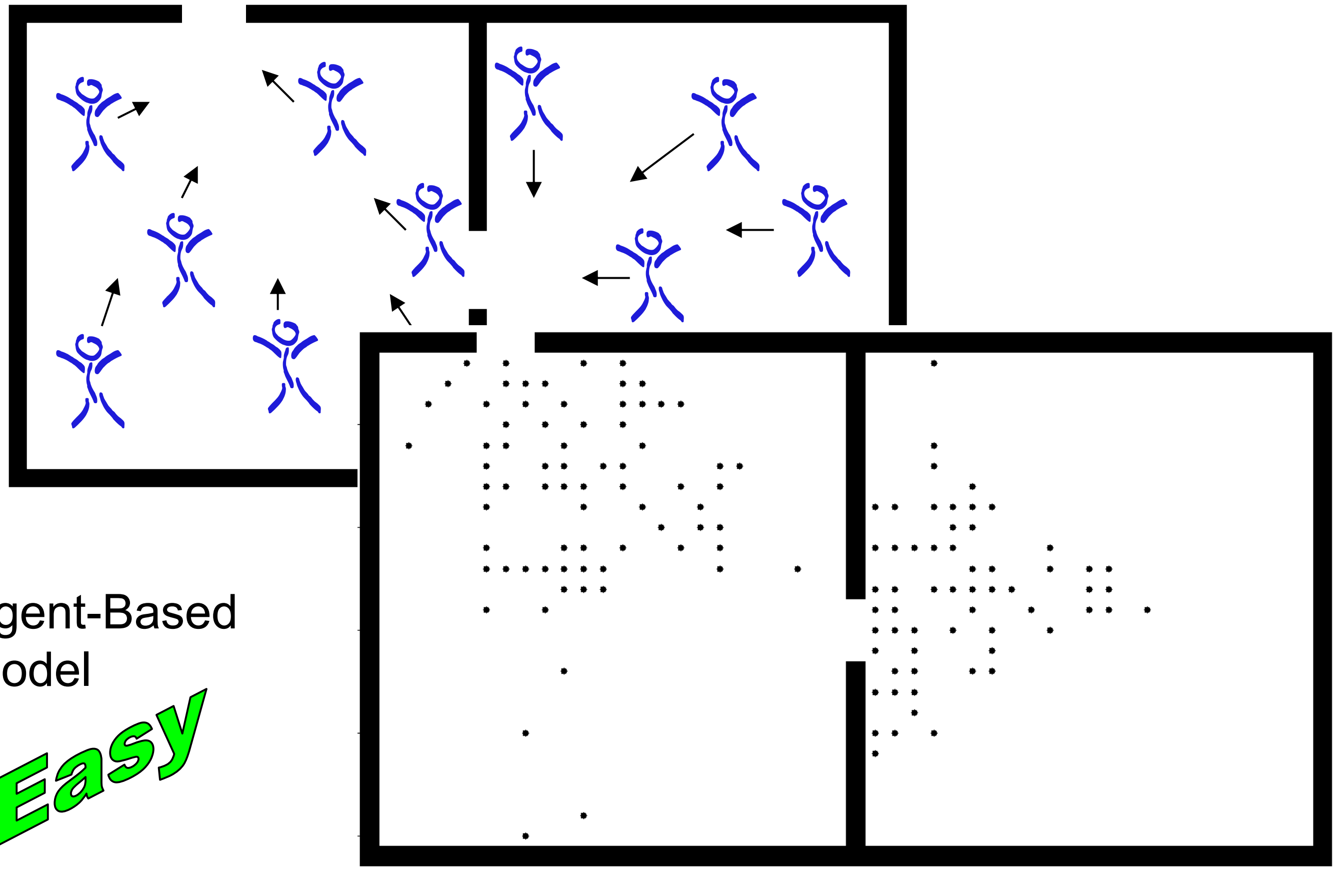
Agent-Based Model

Easy

Macroscopic approach

Tricky

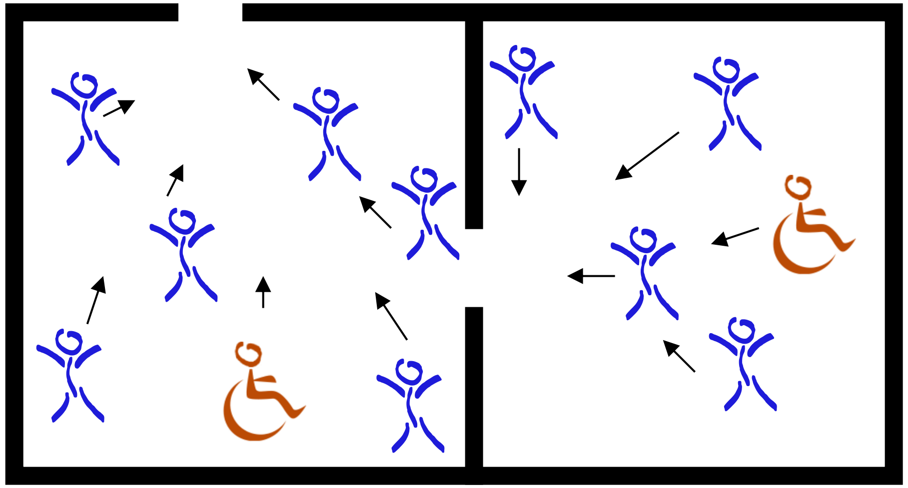
Example: Emergency exit strategy



Agent-Based Model

Easy

Example: Emergency exit strategy



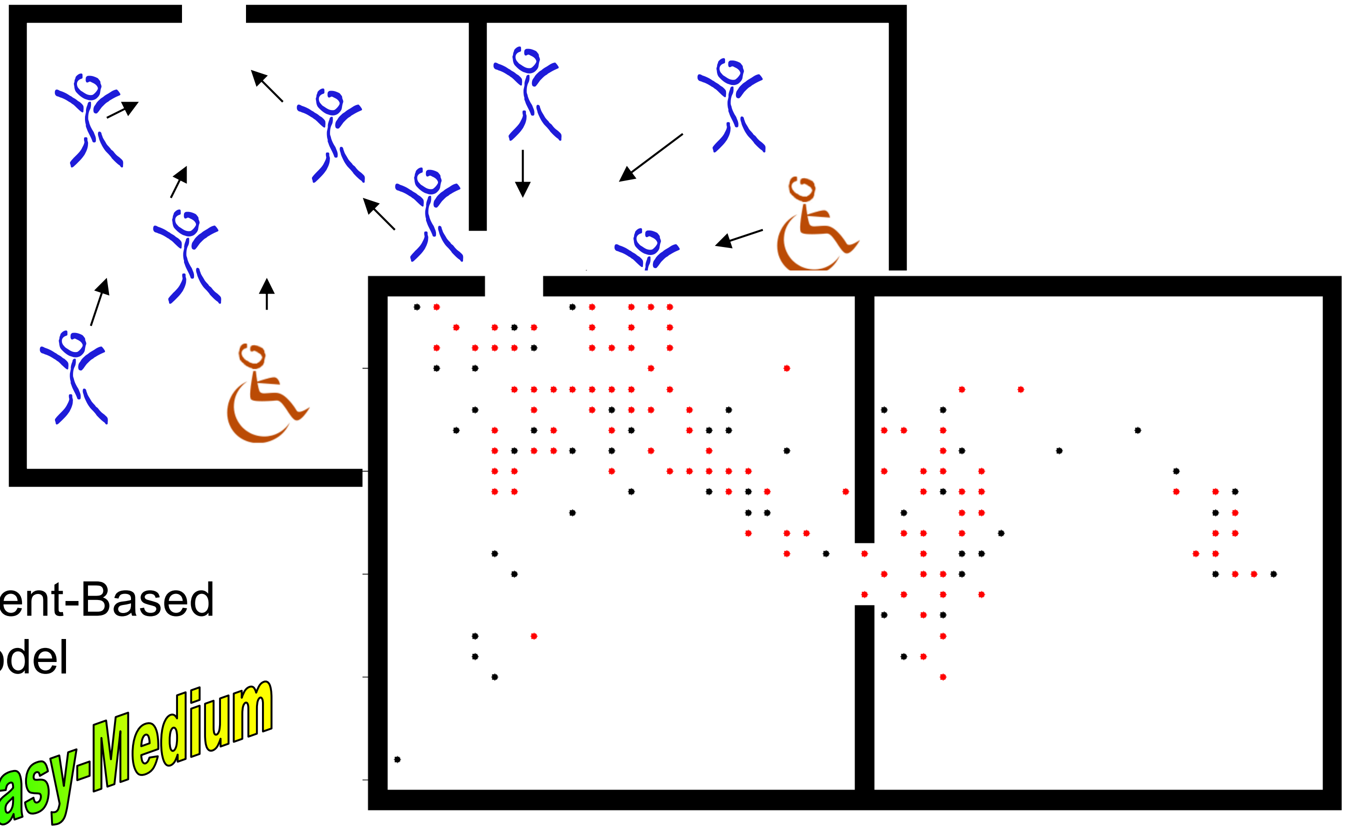
Agent-Based Model

Easy-Medium

Macroscopic approach

Almost Impossible

Example: Emergency exit strategy



Agent-Based
Model

Easy-Medium

Properties of Agent-Based Models

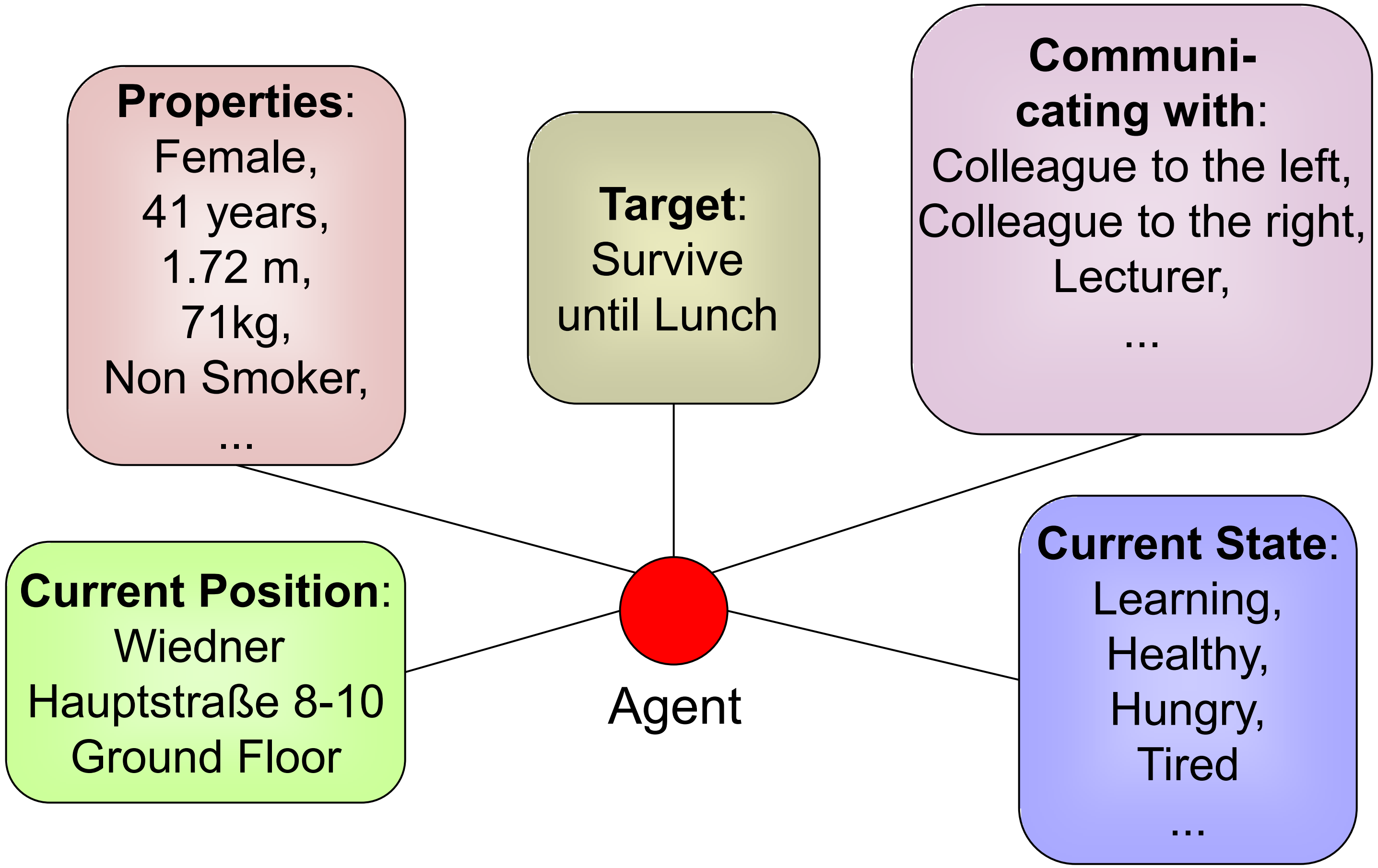
(Bonabeau, 2002)

a. Representation of „emergent phenomena“
b. Flexibility
c. Natural description of the system

Natural description of the System

- Components of the system look like in reality
- Parameters can be seen like data or properties of individuals in reality
- No mathematical background knowledge is required in order to understand the modelling approach

Natural description of the System



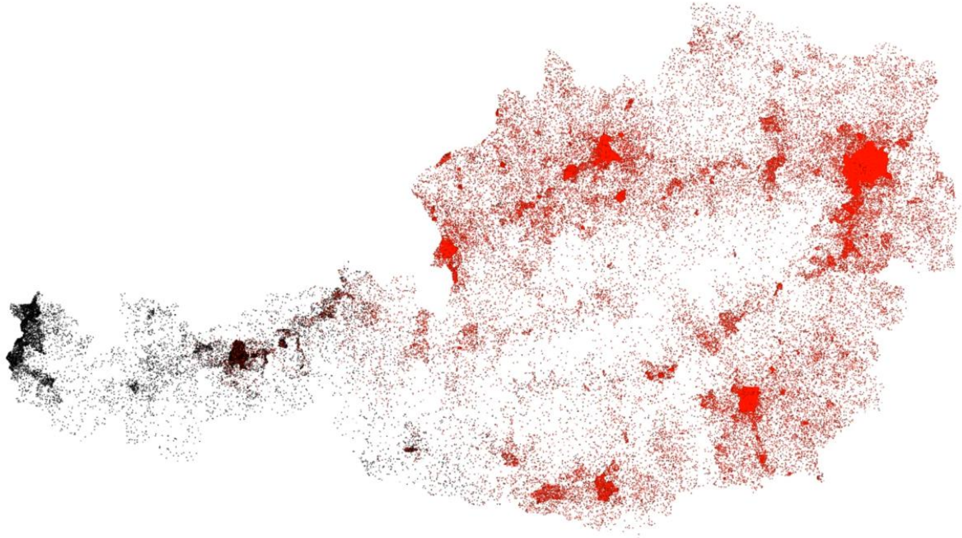
Beware of wrong ideas!

- Natural description of the system makes the model easier to communicate
- Therefore it is often regarded as more credible by non-experts

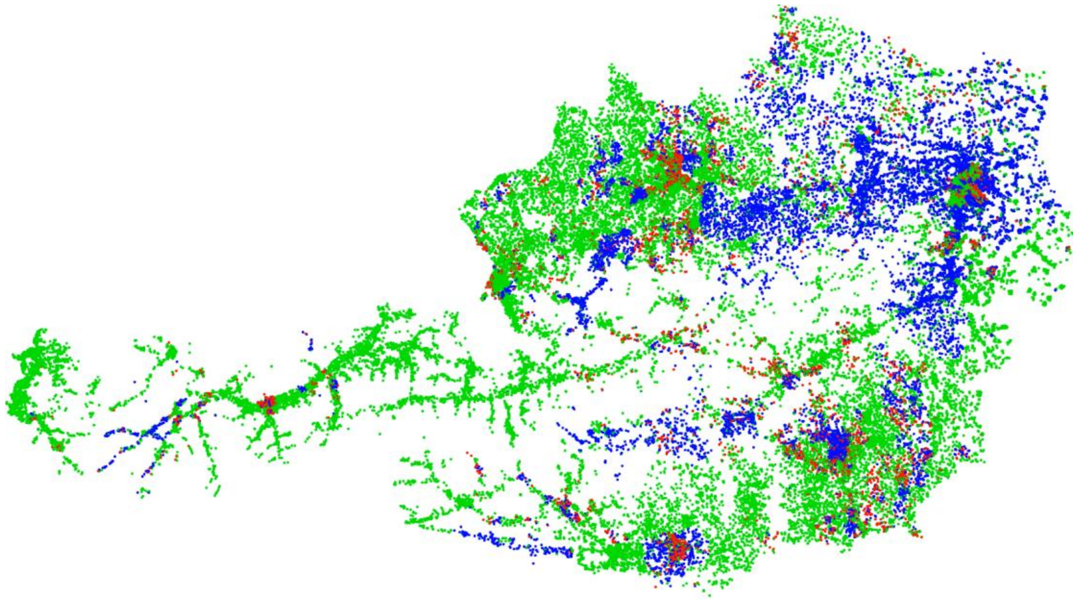
BUT

VALID-LOOKING != VALID

Example: Disease Spread Model



How does an infectious disease spread?



Both models look reasonable…

Interpretation of Agent-Based Model Results

Basically two classes of agent-based models can be observed

ABMs for **qualitative** investigation

- Usually interested in (temporal behaviour) of patterns
- Usually used for foundamental scientific research

ABMs for **quantitative** investigation

- Usually interested in temporal behaviour of aggregate numbers
- Usually used for some kind of resource planning

Research Question?

Interpretation of Agent-Based Model Results

Basically two classes of agent-based models can be observed

ABMs for **qualitative** investigation

- (On purpose) very abstract
- Usually very complex model behaviour
- Hardly any parameters identified with real data

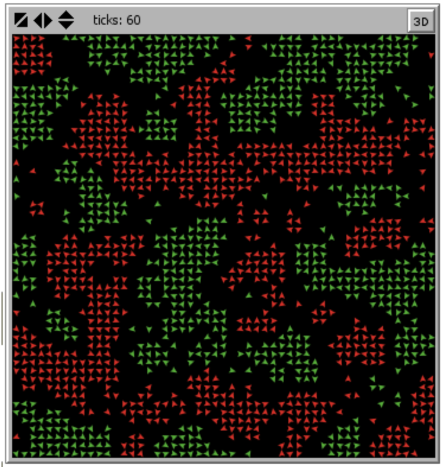
ABMs for **quantitative** investigation

- Rather simple agent interactions
- A lot of data involved for model parametrisation and validation
- Usually less famous

Model?

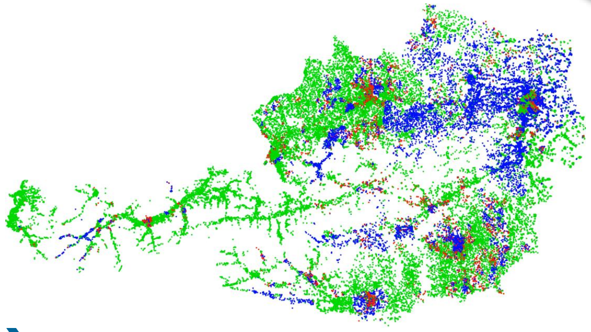
Interpretation of Agent-Based Model Results : Examples

ABMs for **qualitative** investigation



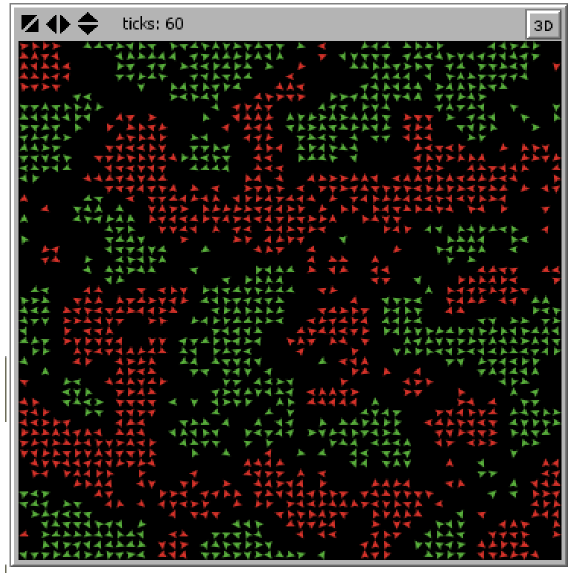
Schelling‘s Segregation Model

ABMs for **quantitative** investigation

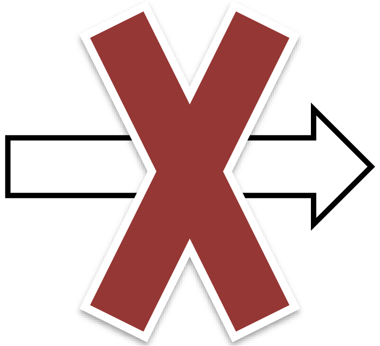


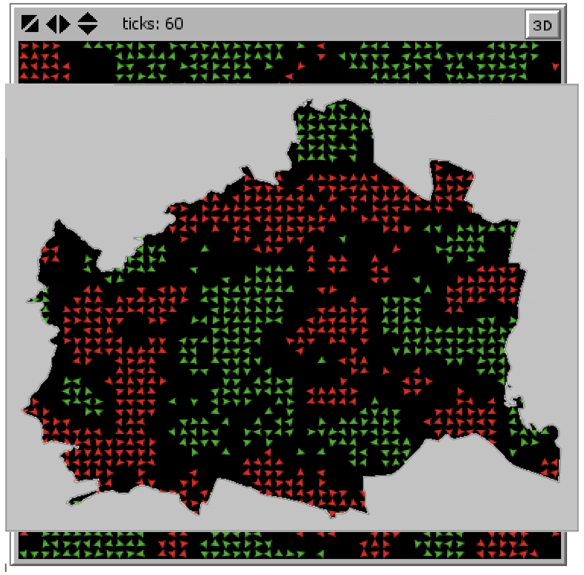
GEPOC / SIR

Interpretation of Agent-Based Model Results : Examples



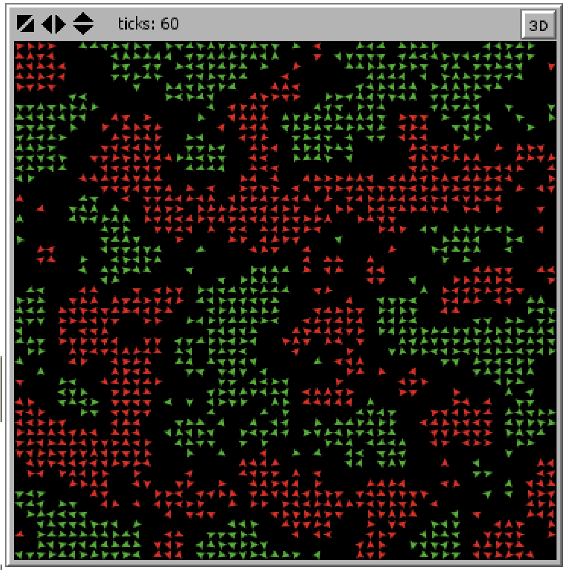
WRONG INTERPRETATION



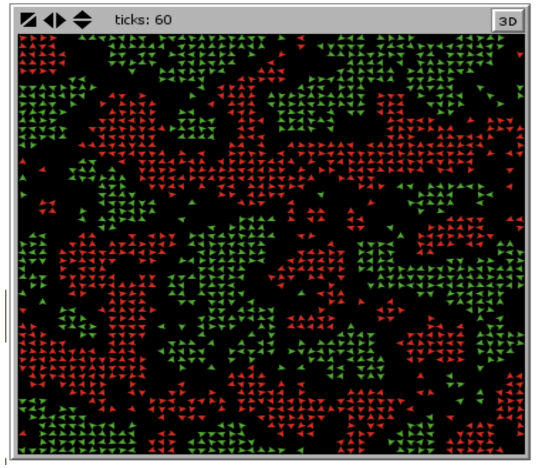


„Schelling‘s model predicts: In a few years only immigrants in Wien Hietzing!“

Interpretation of Agent-Based Model Results : Examples



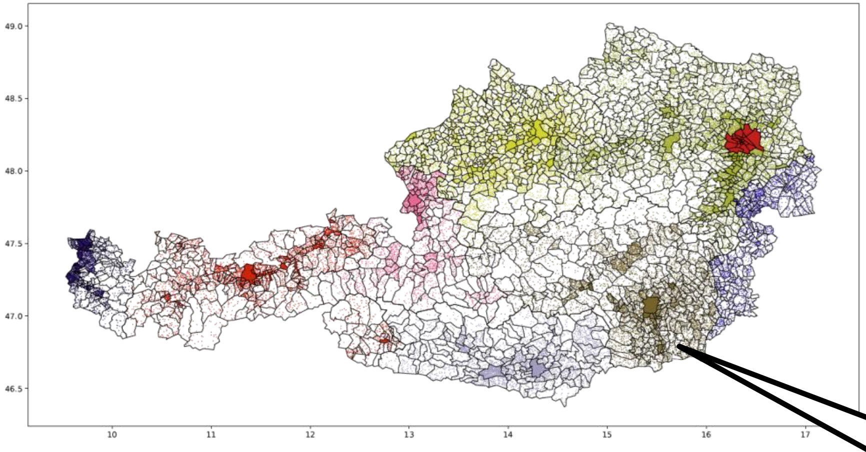
CORRECT INTERPRETATION



„If we do not take care on our migration policy human homophobia might lead to spatially visible ghettoism as seen above in Austria as well!“

Interpretation of Agent-Based Model Results : Examples

WRONG INTERPRETATION

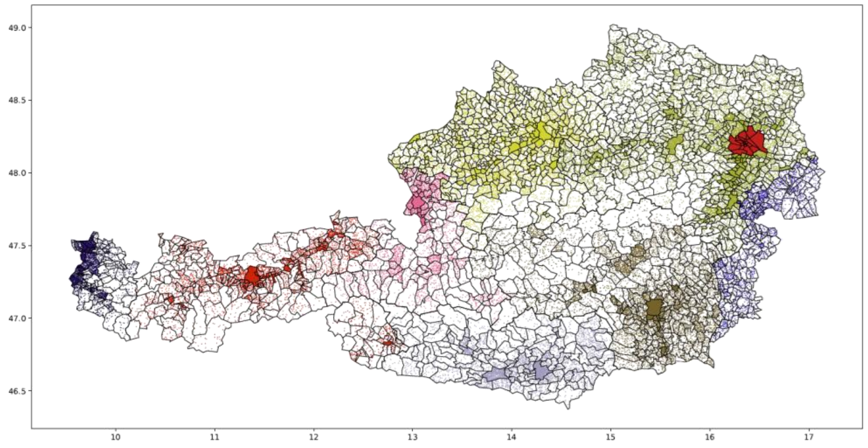


„GEPOC predicts: In two years there will be am infected 50 year old immigrant in Leibnitz“

Hi guys, i‘m Mike

In general: **<u>Never</u>** pick only one agent from an ABM!

Interpretation of Agent-Based Model Results : Examples



CORRECT INTERPRETATION

„GEPOC predicts: Austrian population is assumed to grow to x.x Mio people until 2030.“

Summary: Properties of Agent-Based Models

Agent-based models are good in…

- … analysis and discovery of complex group dynamic behaviour. This must not necessarily be a good thing as emergent behaviour may occur in models even if it is not correct.
- … communitcating models to non-experts.
  The modelling appoach is easy to understand, picturesque and no mathematical background is necessary.

Summary: Properties of Agent-Based Models

Agent-based models are good in…

- … analysis and discovery of complex group dynamic behaviour.
- … communitcating models to non-experts.

Agent-based modelling is problematic …

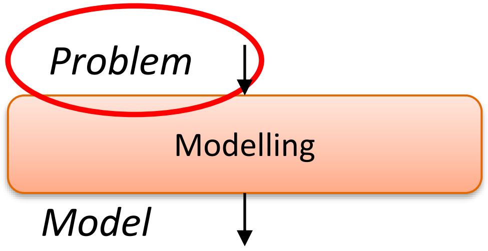
- … regards misinterpretation. If it looks like reality it must not necessarily be a valid model for it.
- … regards the validation process. Validation of ABMs is a difficult task due to complex model behaviour.
- … regards computer ressources. ABMs require high performance CPUs and a lot of RAM.

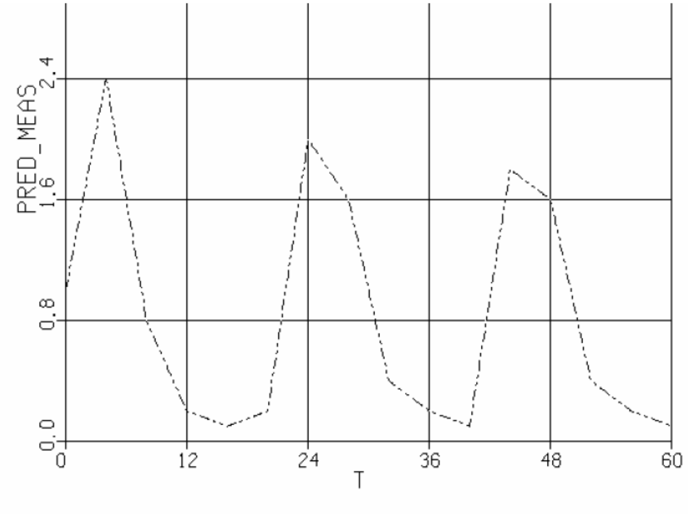
ABM CLASSIFICATION AND CASE STUDIES

Goal

- Get some idea about, how an agent-based model may „look like“
- Get some idea about, how diverse agent-based modelling is
- Classifications of ABMs – clean-up this mess...
- Tips and Tricks

Case Study 1: Predator Prey Model





Dynamics: Predator eats Prey
Predator / Prey births, deaths

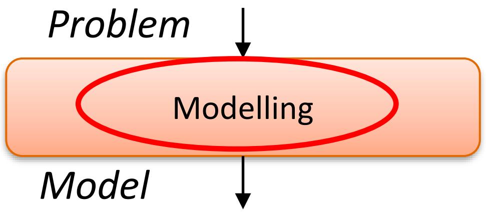
Environment: isolated

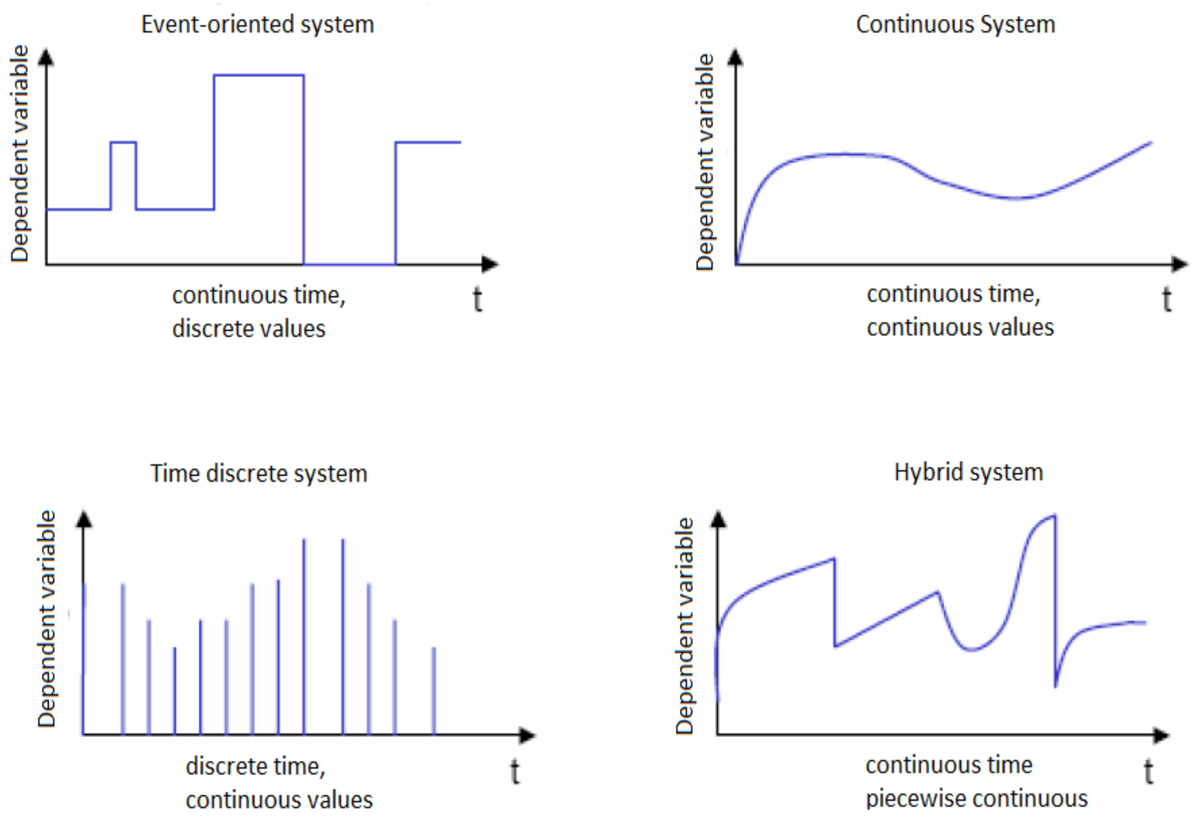
Measurement: Predator Population

5 Years = 60 months, quarterly

Problem: When is a reasonable time to use chemical pesticides to reduce number of predators?

Case Study 1: Predator Prey Model





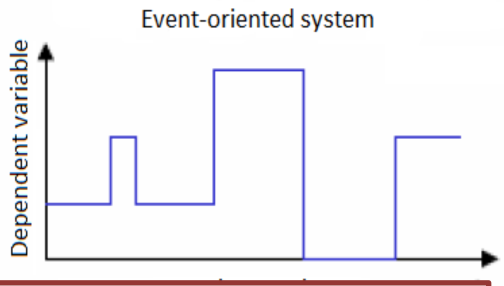
Case Study 1: Predator Prey Model

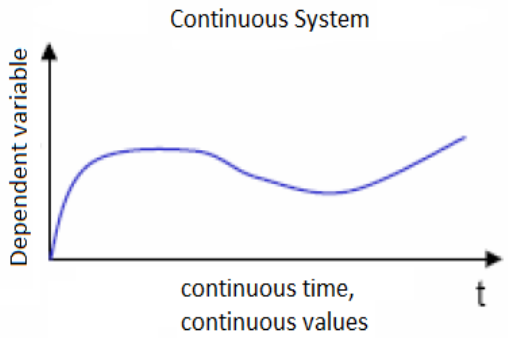
Problem

Modelling

Model

Approach 1 (see 1st lecture)



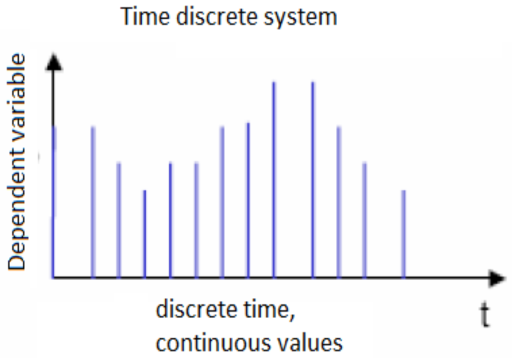


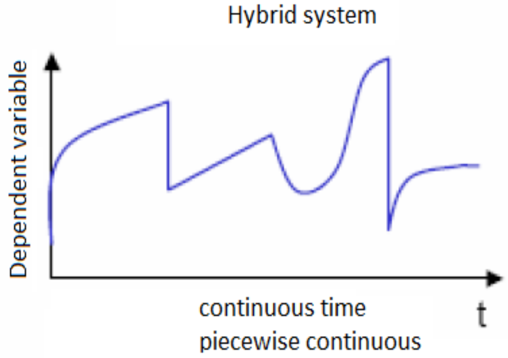
$Y(t)$ ... Prey

$X(t)$ ... Predators

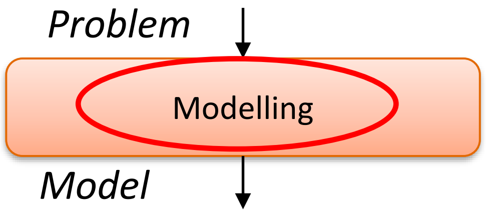
Separation – Isolated environment

Choice - 2 variables = 2 states



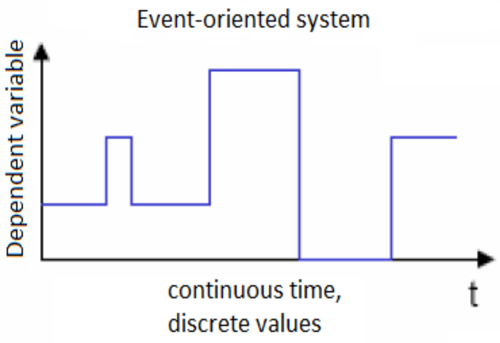


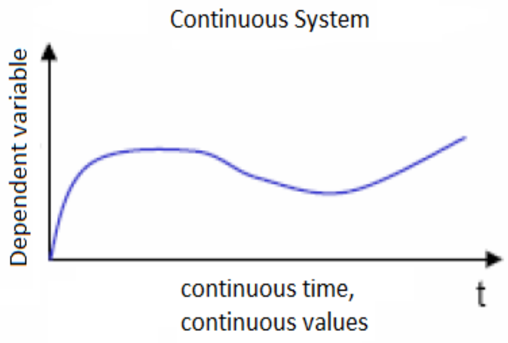
Case Study 1: Predator Prey Model

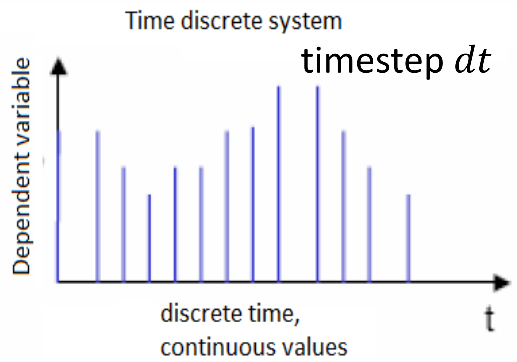


Separation –
Isolated environment
(~ rectangular grid)

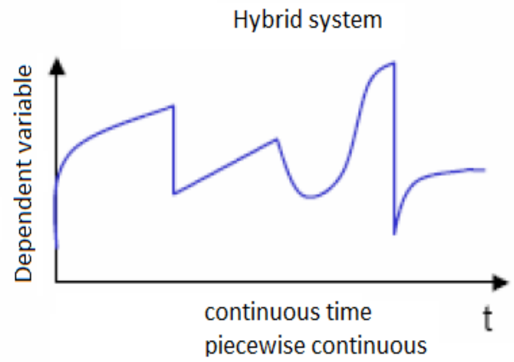
Choice -
$Y(t) = \#\{y_i(t)\}$ prey agents
$X(t) = \#\{x_i(t)\}$ predator agents





$y_i(t)$ ... Prey animal
$x_i(t)$ ... Predator animal

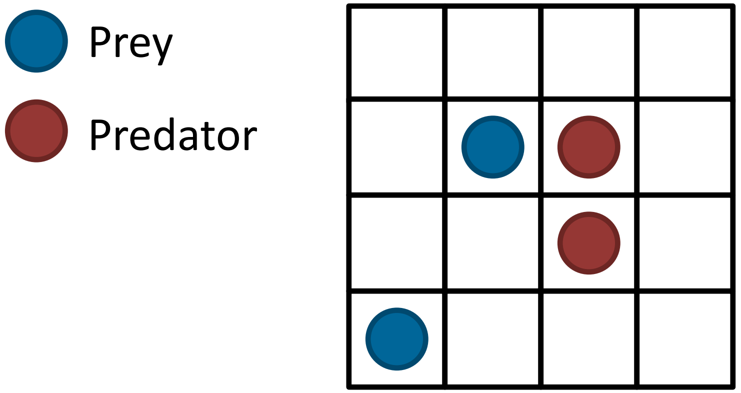


Case Study 1: Predator Prey Model

- Initialisation:
  $Y(0) = Y_0$ prey agents and $X(0) = X_0$ predator agents distributed uniformly on a rectangular grid with $M \cdot N > Y_0 + X_0$ cells
- Time Step Dynamics:
  A time step is split into two phases:
  1. Movement
  2. Population Dynamics

Case Study 1: Predator Prey Model

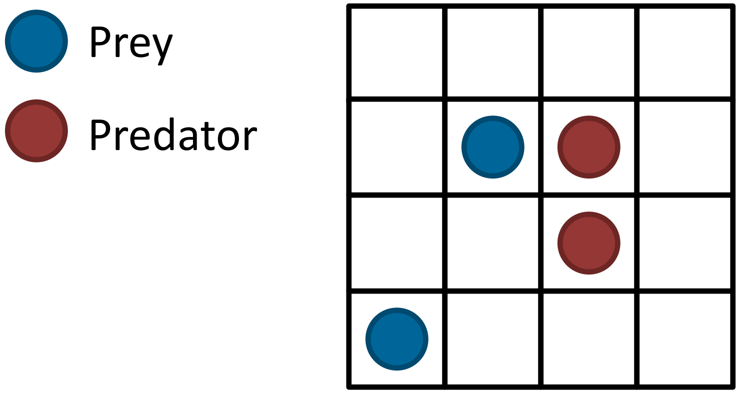
- <u>Time Step Dynamics:</u>
  A time step is split into two phases:
  1. Movement
  2. Population Dynamics



- Every agent moves in a randomly picked neighbour cell (Moore neighbourhood)

Case Study 1: Predator Prey Model

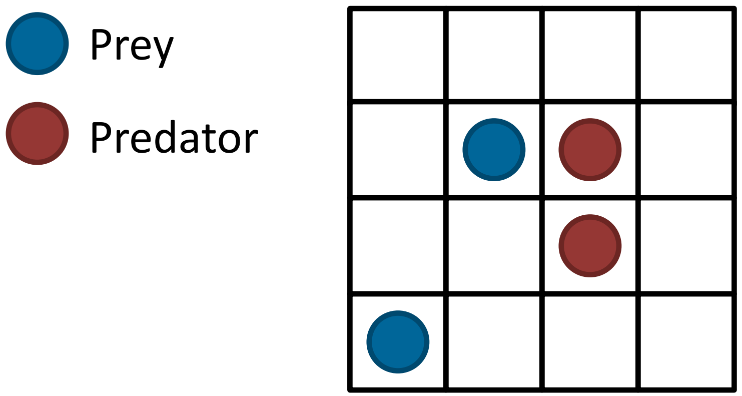
- Time Step Dynamics:
  A time step is split into two phases:
  1. Movement
  2. Population Dynamics



- Every agent moves in a randomly picked neighbour cell (Moore neighbourhood)
- Iterate in random order, periodic boundary conditions

Case Study 1: Predator Prey Model

- <u>Time Step Dynamics:</u>
  A time step is split into two phases:
  1. Movement
  2. Population Dynamics

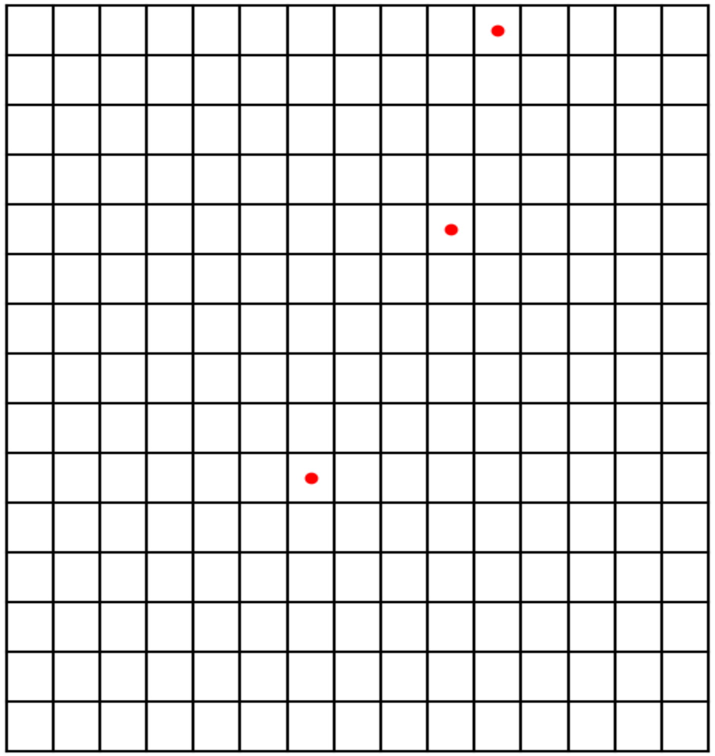


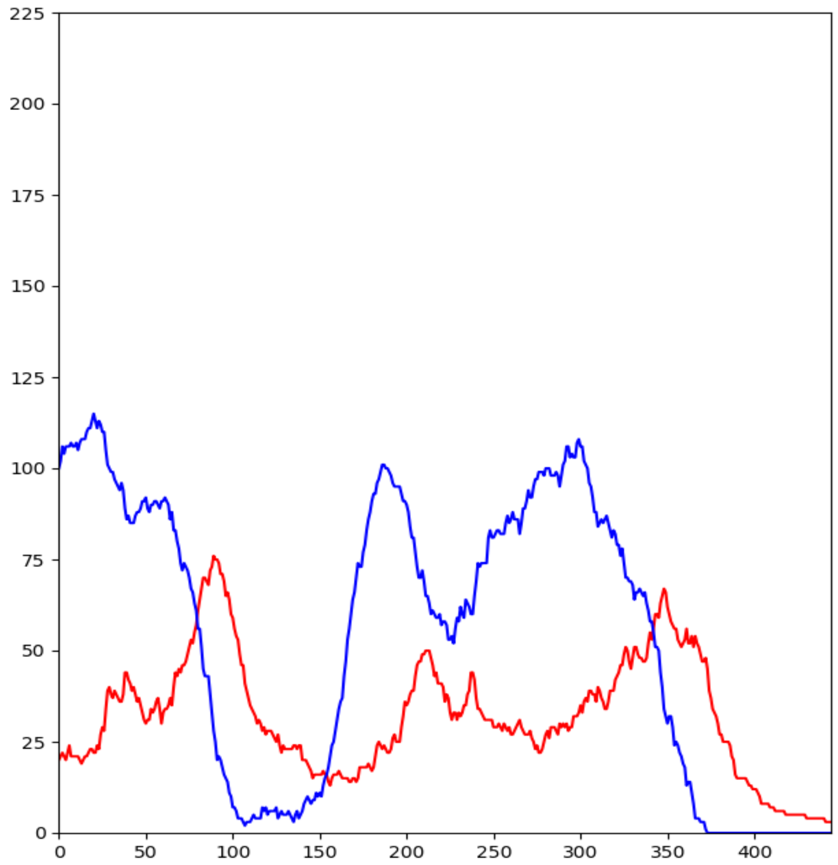
Every time-step agents are iterated in random order:

Predator:
- Every predator dies with probability α
- If prey is around (Moore), the predator successfully catches one of it with probability β and „replaces“ it by one offspring

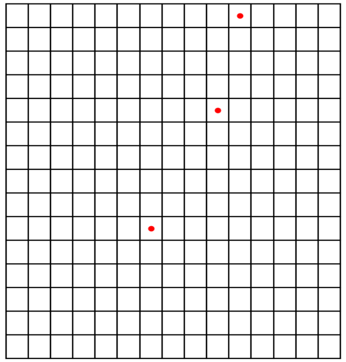
Prey:
- If possible, every prey produces an offspring in one randomly picked neighbour cell with probability γ

Case Study 1: Predator Prey Model

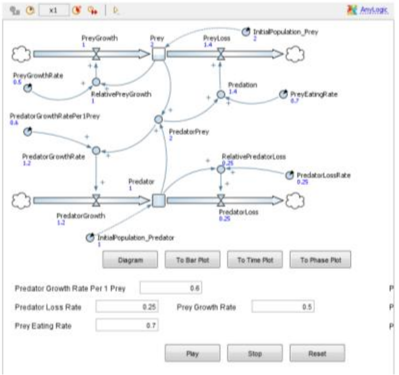


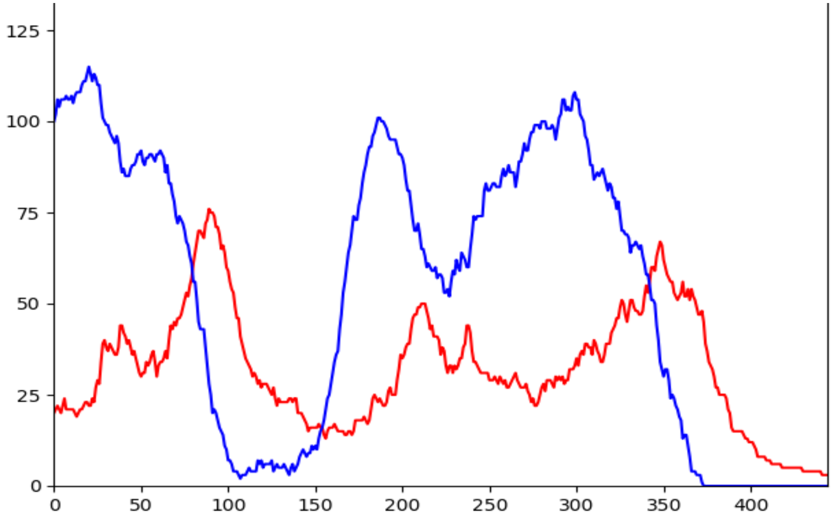


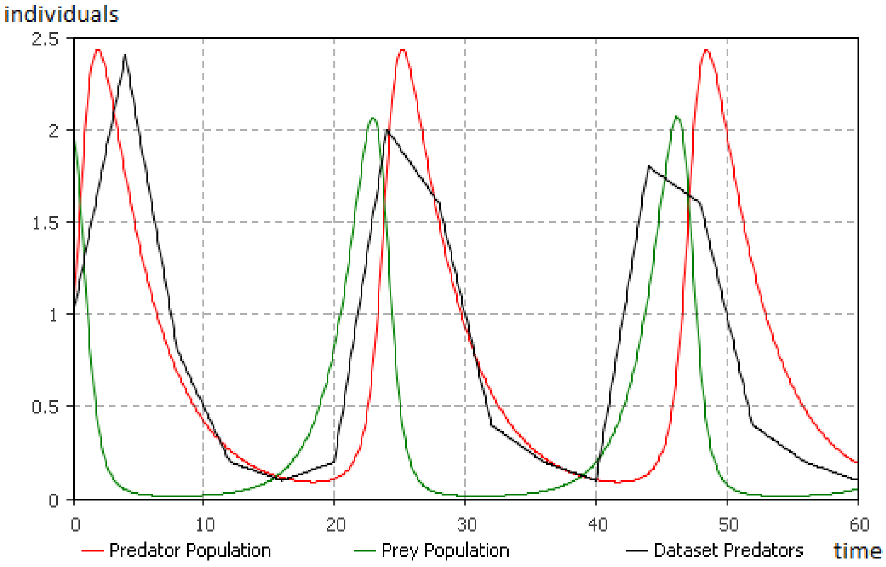
Case Study 1: Predator Prey Model



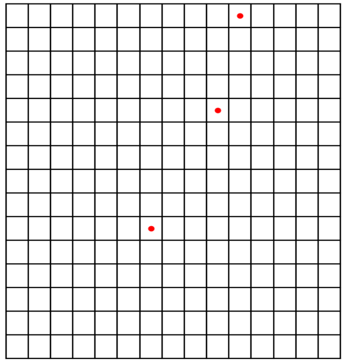
Compare ABM model results with SD model results



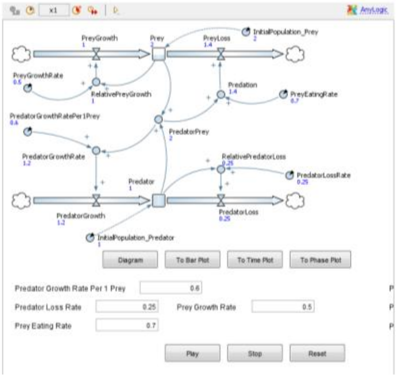


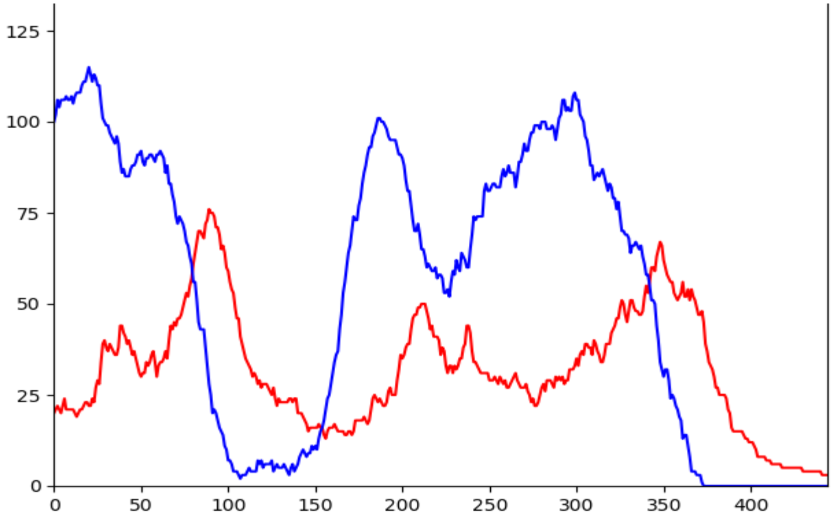


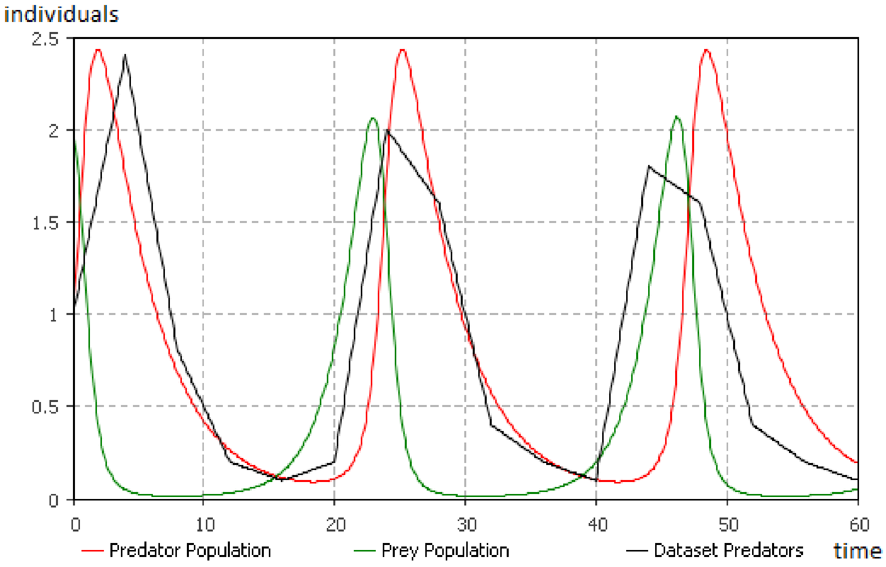
Case Study 1: Predator Prey Model



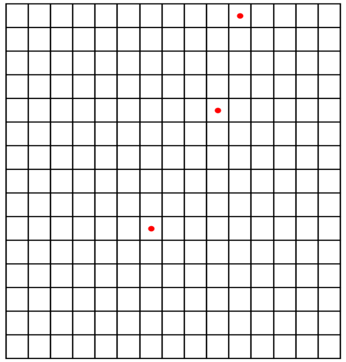
- Fuzzy (randomness)
- Dying out
- Scale



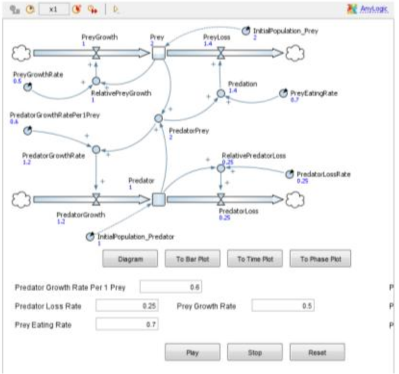


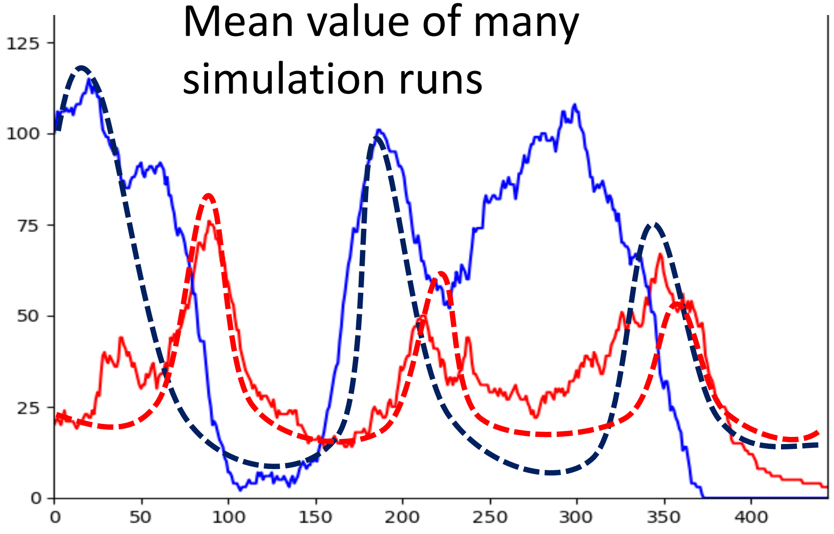


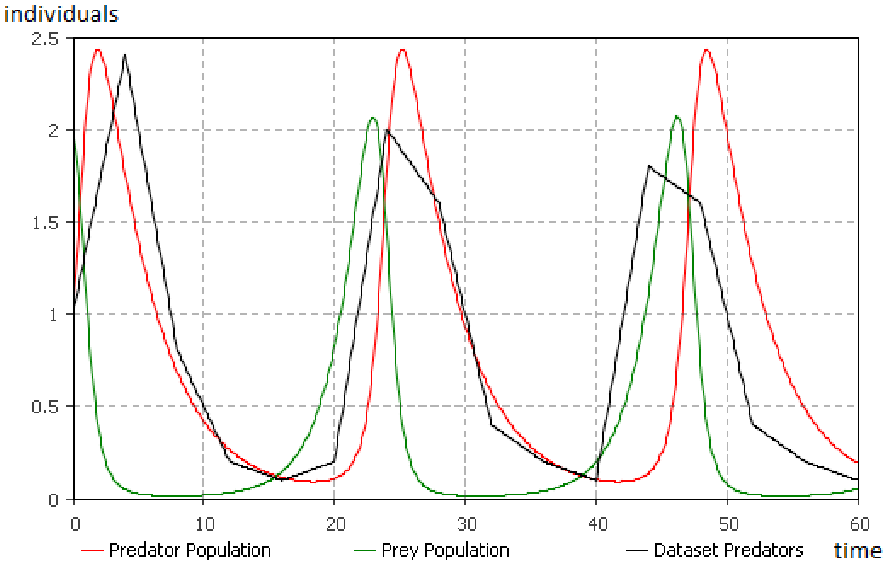
Case Study 1: Predator Prey Model



- Fuzzy (randomness)
- Dying out
- Scale







Case Study 1: Lessons Learned

Lesson 1: Be careful, in which sequence/order agents are addressed to perform actions when updating in steps. Don‘t unintentionally favour some!

You are responsible for correctly ordering events within atomic time-steps!

Case Study 1: Lessons Learned

Lesson 1: Be careful, in which sequence/order agents are addressed to perform actions when updating in steps. Don‘t unintentionally favour some!

Lesson 2: Be careful, when implementing movement on a grid. Don‘t occupy spots twice!

Think about splitting: „movement“ in „attempt movement“ and „execute movement“

Case Study 1: Lessons Learned

Lesson 1: Be careful, in which sequence/order agents are addressed to perform actions when updating in steps. Don‘t unintentionally favour some!

Lesson 2: Be careful, when implementing movement on a grid. Don‘t occupy spots twice!

Lesson 3: Never judge only based on only one simulation result, if randomness is involved!

Perform Monte Carlo simulation

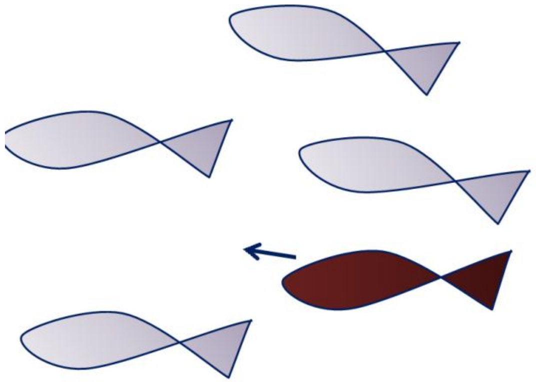
Case Study 2: Boids Flock Model

- Model for simulation of (bird) flocking behaviour
- Craig Reynolds in 1986

- Three simple rules on individual level lead to complex behaviour of the crowd

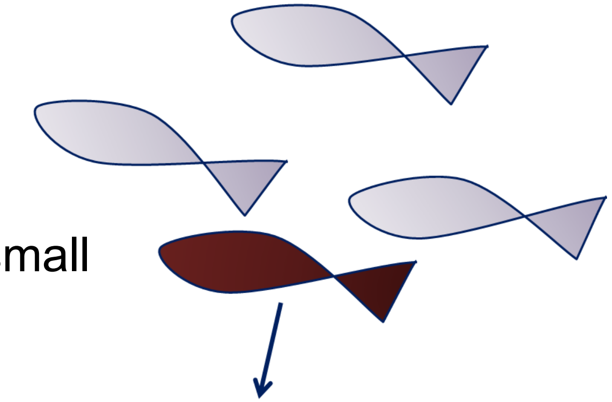
https://www.youtube.com/watch?v=QOGCSBh3kmM

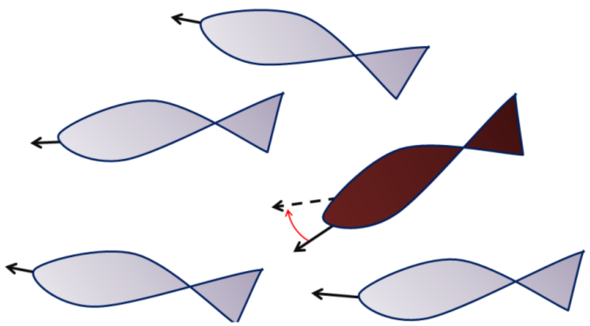
Case Study 2: Boids Flock Model



Each agent tends towards
the centre of its neighbours

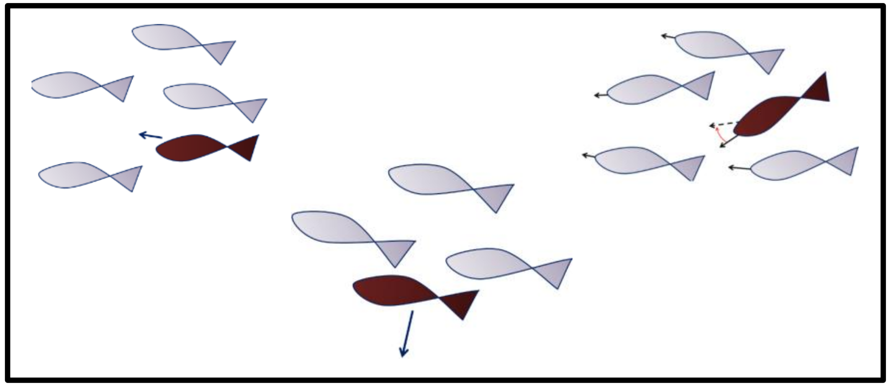
Keep a distance that is
neither too far nor too small

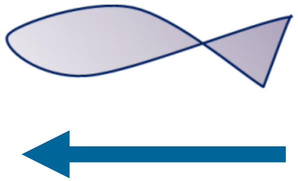




Swim in the same direction
as your neighbours

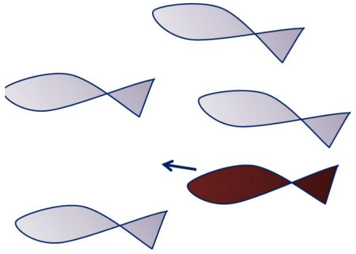
Case Study 2: Boids Flock Model



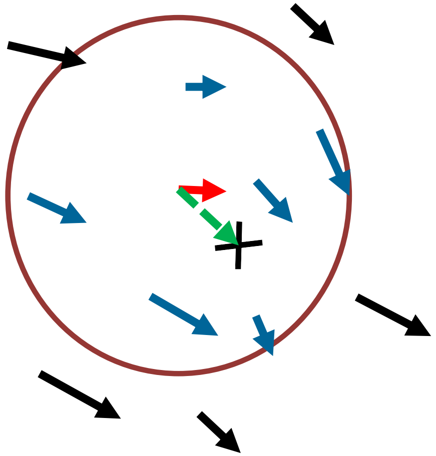


- a_k current position of agent k
- v_k current velocity of agent k

Case Study 2: Boids Flock Model



Each agent tends towards the centre of its neighbours



Let a_k be the position of agent k and let

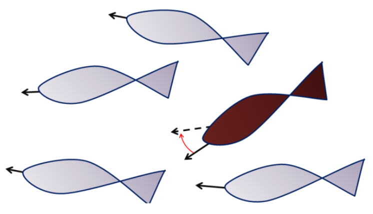
$I := \{k \neq i : ||a_k - a_i|| < Or\}$

for and observation radius Or.

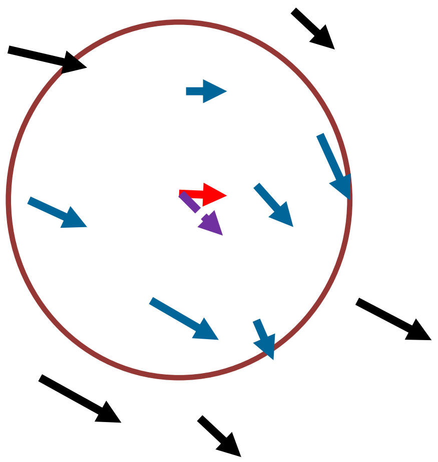
Then:

$$d_1^k = \frac{1}{|I|} \sum_{i \in I} a_i - a_k$$

Case Study 2: Boids Flock Model



Swim in the same direction as your neighbours



Let v_k be the velocity of agent k and let

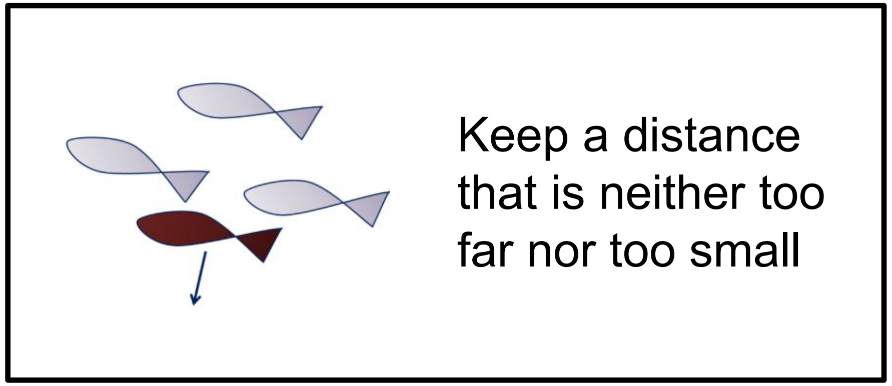
$I := \{k \neq i : ||a_k - a_i|| < Or\}$

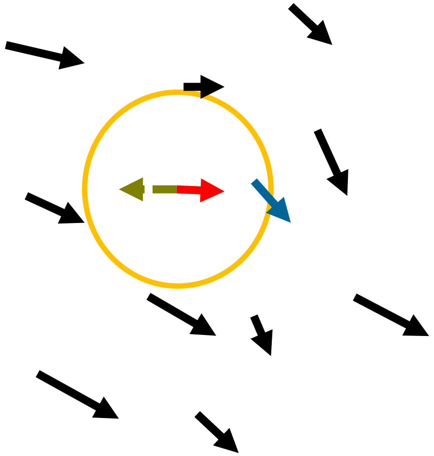
for and observation radius Or.

Then:

$$d_2^k = \frac{1}{|I|} \sum_{i \in I} v_i$$

Case Study 2: Boids Flock Model

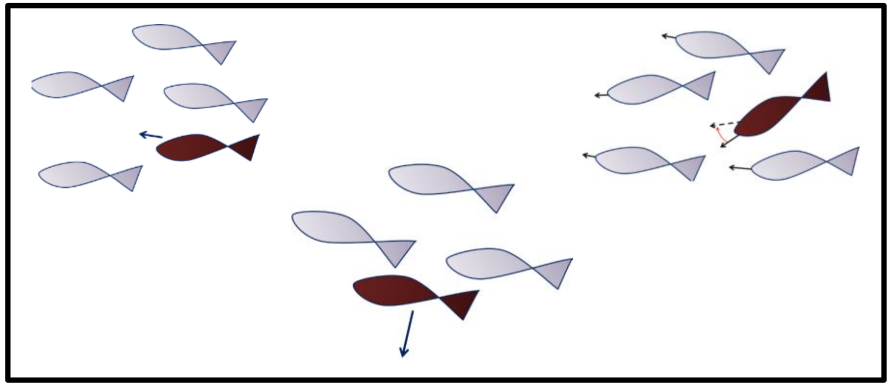


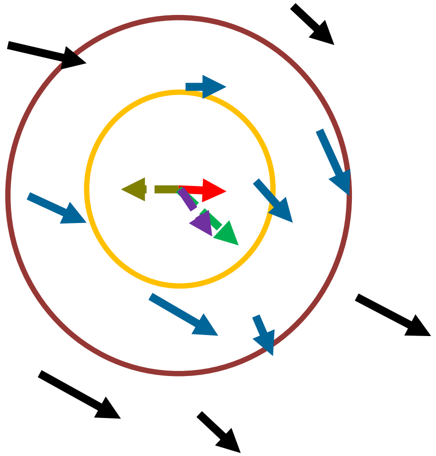


Let a_k be the position of agent k and let
$J := \{k \neq i : ||a_k - a_i|| < Cr\}$
for and collision radius Cr.
Then:

$$d_3^k = a_k - \frac{1}{|J|} \sum_{i \in J} a_i$$

Case Study 2: Boids Flock Model





Update velocity

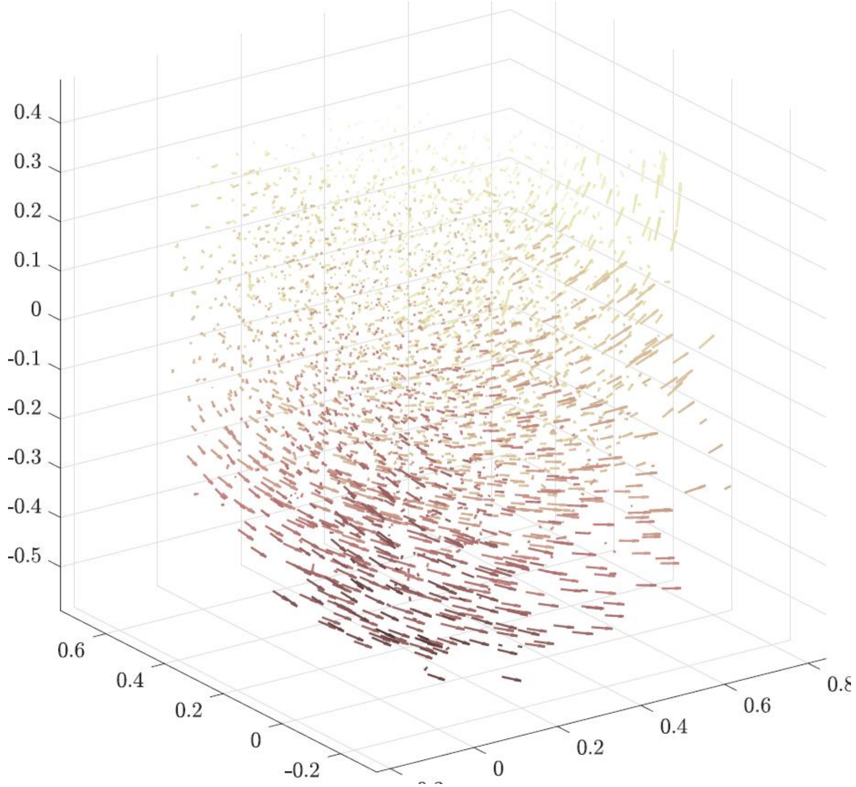
$$\tilde{v}_i = \alpha_0 v_i + \alpha_1 d_1^i + \alpha_2 d_2^i + \alpha_3 d_3^i$$

Update position

$$\tilde{x}_i = x_i + \tilde{v}_i$$

Case Study 2: Boids Flock Model

- Boids model is the most picturesque example for emergence in ABMs
- It is a good test case for agent-based simulators (high computation performance required)



Case Study 2: Lessons Learned

Lesson 4: ABMs are often computationally expensive!
Think about, how to optimize your code performance

Think about benefits vs drawbacks of parallelisation on agent-level.

(at least, try to parallelize Monte Carlo simulation)

Try to avoid $O(N^2)$
Try to avoid „searching“

Case Study 2: Lessons Learned

Lesson 4: ABMs are often computationally expensive!
Think about, how to optimize your code performance

Bad solution:

```
for (agent1:agents) {
        for (agent2:agents) {
                if
(agent1.region==agent2.region){
                        do something…
                }
        }
}
```

Try to avoid $O(N^2)$
Try to avoid „searching“

Case Study 2: Lessons Learned

Lesson 4: ABMs are often computationally expensive!
Think about, how to optimize your code performance

Better solution:

```
regionMap = Map()
for (agent:agents) {
        regionMap[agent.region].add(agent)
}
for (agent:agents) {
        others = regionMap[agent.region]
        for (agent2: others) {
                do something…
        }
}
```

Try to avoid $O(N^2)$
Try to avoid „searching“

Case Study 2: Lessons Learned

Lesson 4: ABMs are often computationally expensive! Think about, how to optimize your code performance

Lesson 5: Think about potentially different individual perceptions: i.e. what every agent knows/sees.

Correctly model individual perceptions is the key to implementing realistic behaviour

Case Study 2: Lessons Learned

Lesson 4: ABMs are often computationally expensive! Think about, how to optimize your code performance

Lesson 5: Think about potentially different individual perceptions: i.e. what every agent knows/sees.

Lesson 6: A fully <u>reproducible</u> model description is difficult and can be long and cconfusing. Think about using a standartised protocol for it.

Usually it is worth to make two documentations: one lay documentation and one technical

Case Study 2: Lessons Learned

- Example: ODD Protocol by Volker Grimm et.al.
- Standartised documentation of agent-based models

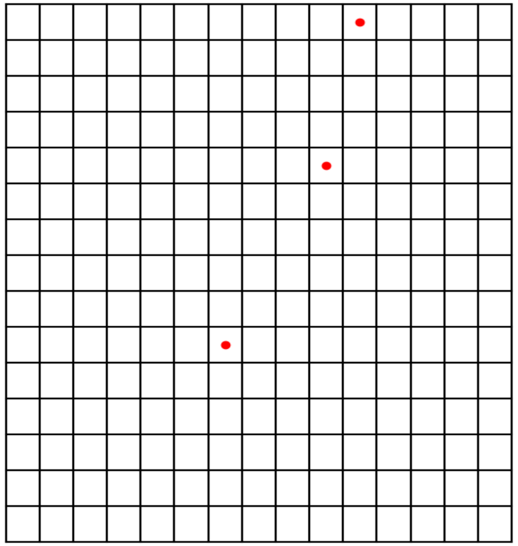
Overview	**Purpose**
	State variables and scales
	Process overview and scheduling
Design concepts	**Design concepts**
Details	**Initialization**
	Input
	Submodels

Fig. 1 – The seven elements of the ODD protocol, which can be grouped into the three blocks: Overview, Design concepts, and Details.

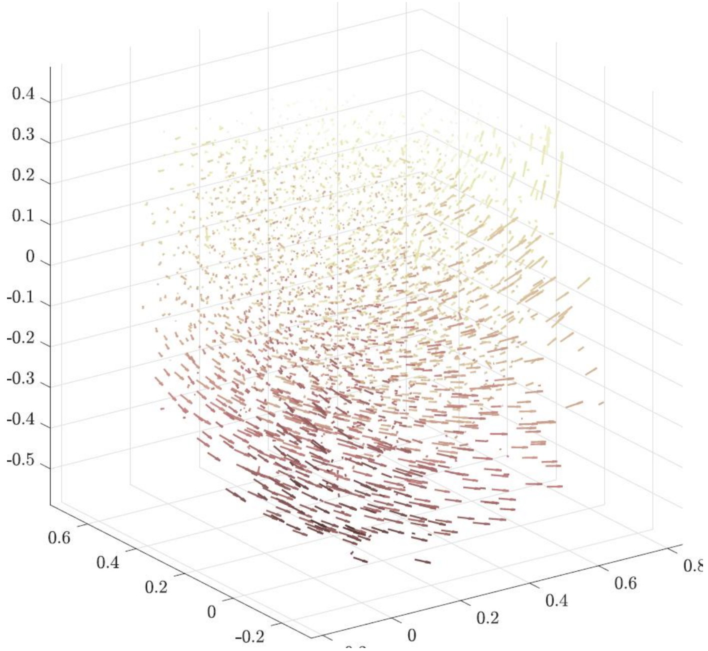
Grimm, Volker, Uta Berger, Finn Bastiansen, Sigrunn Eliassen, Vincent Ginot, Jarl Giske, John Goss-Custard, u. a. „A Standard Protocol for Describing Individual-Based and Agent-Based Models“. *Ecological Modelling* 198, Nr. 1–2 (September 2006): 115–26. https://doi.org/10.1016/j.ecolmodel.2006.04.023.

Classification of ABMs (1)

Differences?



vs.



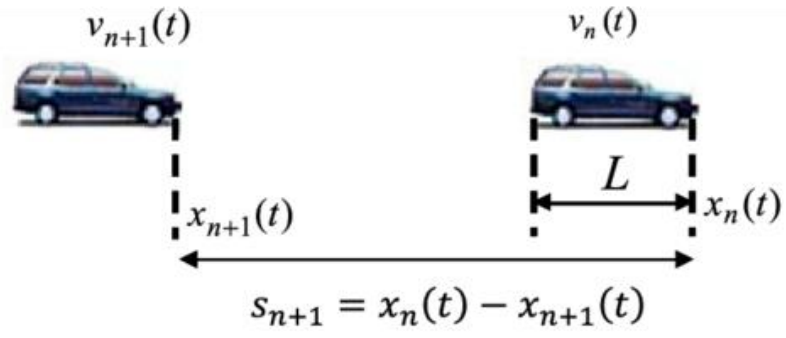
Classification of ABMs (1)

Classification 1	
with respect to modelling purpose (i.e. the research question)	
ABMs for qualitative investigation	ABMs for quantitative investigation
• (On purpose) very abstract • Usually very complex model behaviour • Hardly any parameters identified with real data	• Rather simple agent interactions • A lot of data involved for model parametrisation and validation • Usually less famous

Classification of ABMs (2)

Classification 2			
with respect to agent **environment**			
spatial environment		**abstract** environment	
lattice	**continuous**	**network**	...
• Sometimes equivalent to a CA • Different forms of grids • 1D – 3D	• Often uses distance-metrics for agent interaction • Surprisingly, often easier to handle than lattice models	• Contacts between agents modelled as edges of a network	

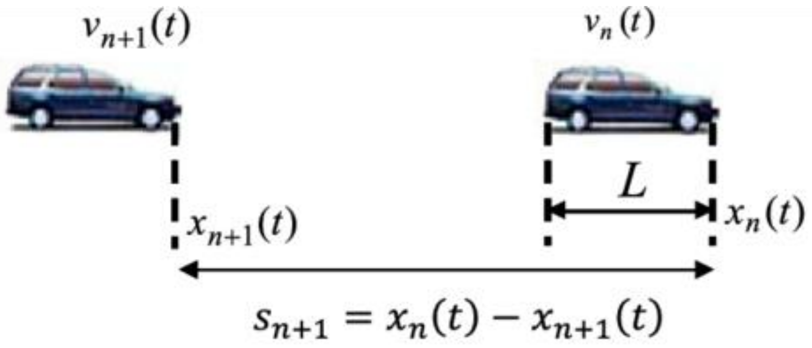
Case Study 3: Gipps 's Car Following Model



- Each car in a one-lane road is represented by an agent
- Each agent i has a certain length L_i, position $x_i(t)$ and velocity $v_i(t)$
- Velocity update is based on a differential equation that includes the distance to and velocity of the car in front

Case Study 3: Gipps 's Car Following Model

- Each car in a one-lane road is represented by an agent
- Each agent i has a certain length L_i, position $x_i(t)$ and velocity $v_i(t)$
- Velocity update is based on a differential equation that includes the distance to and velocity of the car in front

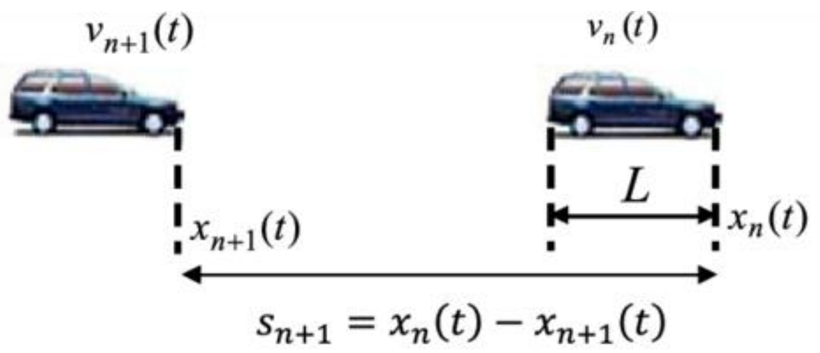


$$\dot{v}_{i+1}(t+\tau) = A \cdot \frac{v_i(t) - v_{i+1}(t)}{x_i(t) - x_{i+1}(t) - L_i}$$

A ... acceleration constant
τ ... reaction time

Case Study 3: Gipps ‘s Car Following Model

- Each car in a one-lane road is represented by an agent
- Each agent i has a certain length L_i, position $x_i(t)$ and velocity $v_i(t)$
- Velocity update is based on a differential equation that includes the distance to and velocity of the car in front



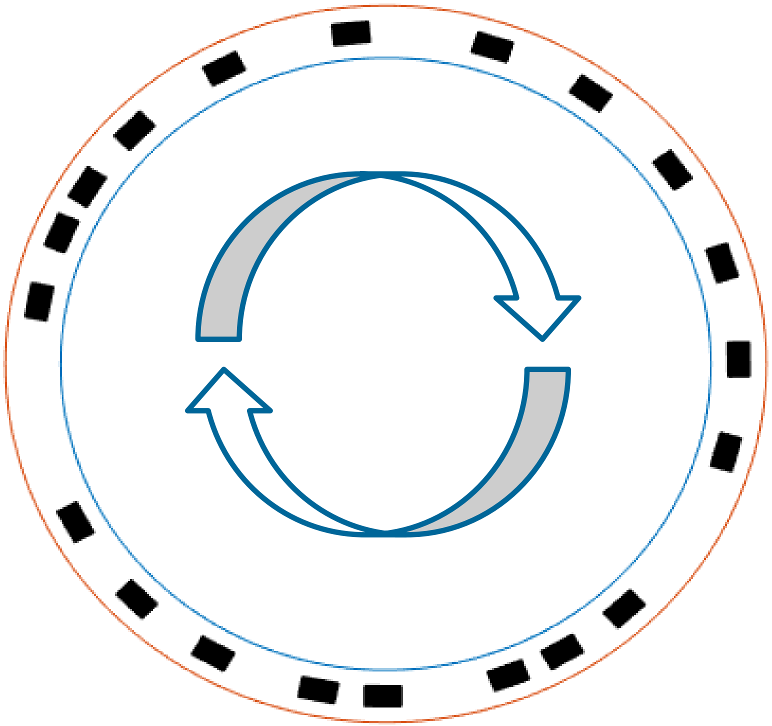
$$\dot{v}_{i+1}(t+\tau) = A \cdot \frac{v_i(t) - v_{i+1}(t)}{x_i(t) - x_{i+1}(t) - L_i}$$

A ... acceleration constant
τ ... reaction time

Some additional parameters:
lingering, maximum velocity, maximum acceleration, maximum break force, length of the road

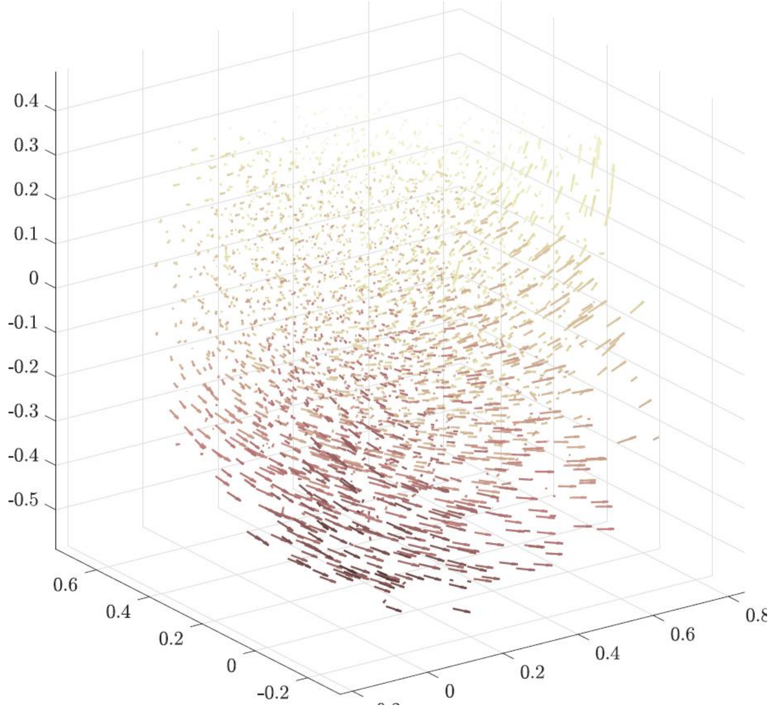
Case Study 3: Gipps 's Car Following Model

- Gipps model poses the base for most modern models for traffic flow
- Alternative approaches: Nagel-Schreckenberg Model, Burgers equation
- Extensions to: multiple lanes, junctions, traffic lights, ...

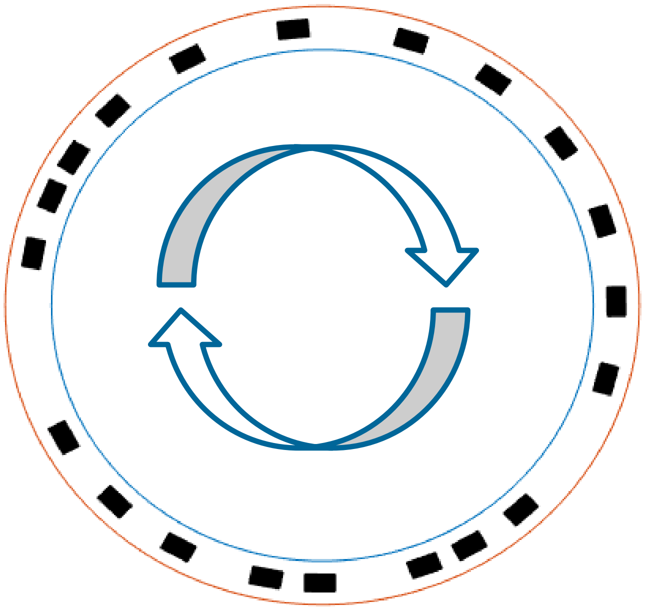


Case Study 3: Gipps ‘s Car Following Model

Differences?



vs.

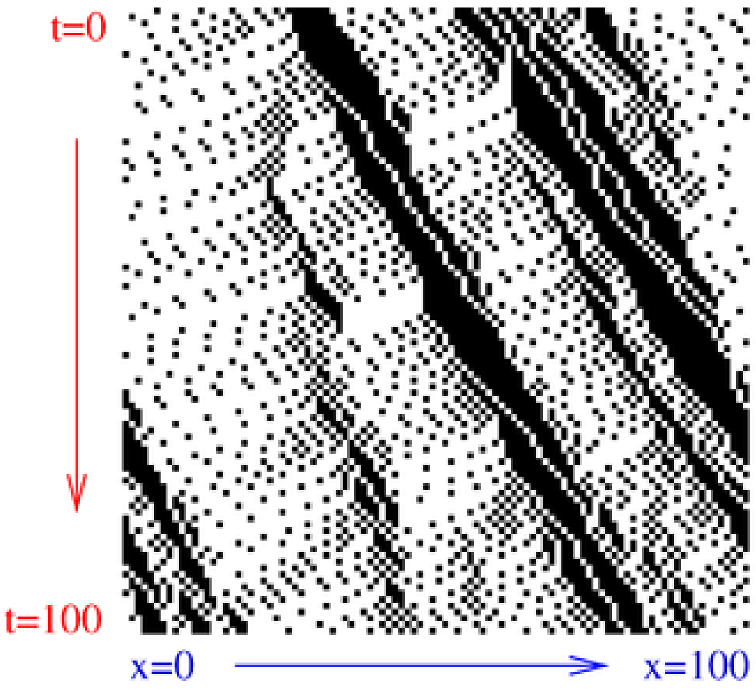


Classification of ABMs (3)

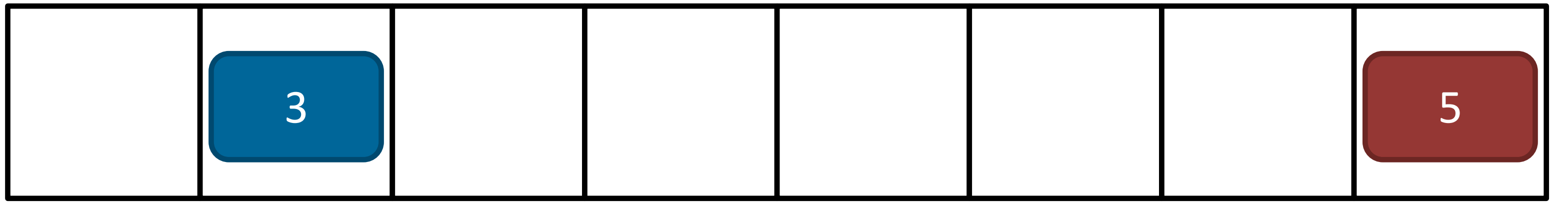
Classification 3		
with respect to **time update**		
time continuous		**time discrete**
differential equation	**event-based**	**time steps**
• Usually used for systems with physical laws	• Often used for scheduling problems	• Most common update strategy. • Needs special care with events happening at the same time

Case Study 4: Nagel Schreckenberg Model

- 1992, Kai Nagel and Michael Schreckenberg
- Same Purpose as Gipps Model
- Discrete 1D Grid insread of continuous road
- One car per grid point

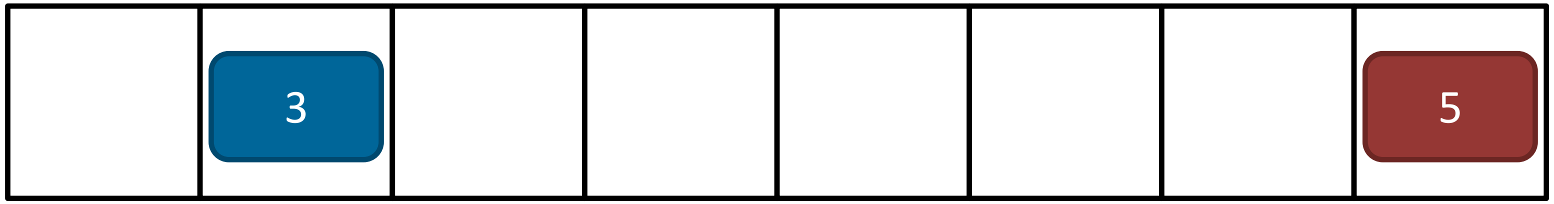


Case Study 4: Nagel Schreckenberg Model



- Agents enter the model from the left (at the left-most cell)
- Each agent has a certain velocity (natural number)
- Model is updated with equidistant time-steps

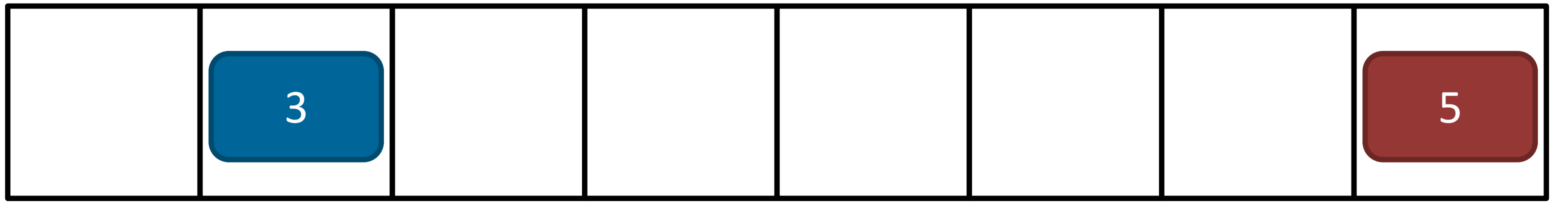
Case Study 4: Nagel Schreckenberg Model



Each time-step, each agent...

- Updates its velocity according to the car in front (if any)
- Drives that many cells to the right

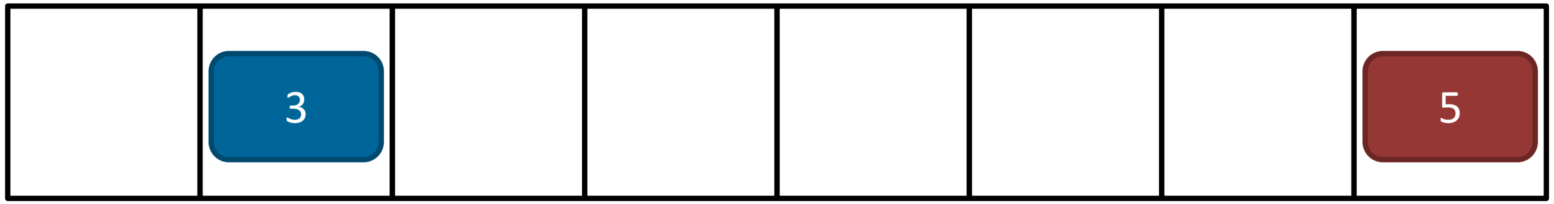
Case Study 4: Nagel Schreckenberg Model



Each time-step, each agent...

- Updates its velocity according to the car in front (if any)
  1. Increases its velocity v by one: $v \leftarrow v + 1$
  2. Checks how many cells to the right are empty (say q)
  3. If $v > q$, then $v \leftarrow q$
  4. With a certain probability: $v \leftarrow \max(v - 1, 0)$

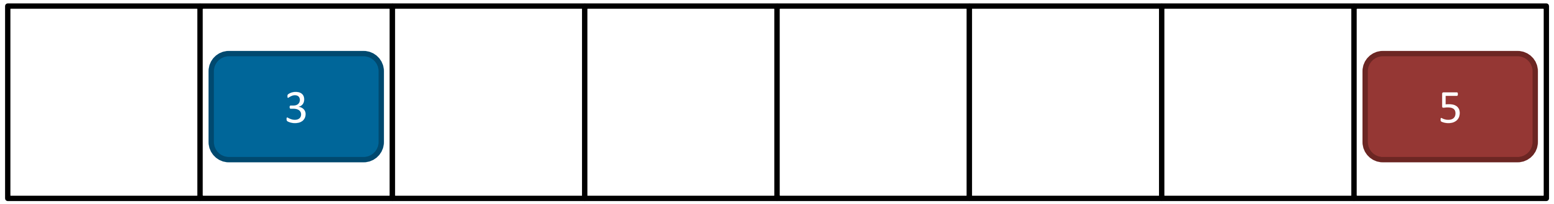
Case Study 4: Nagel Schreckenberg Model



Each time-step, each agent...

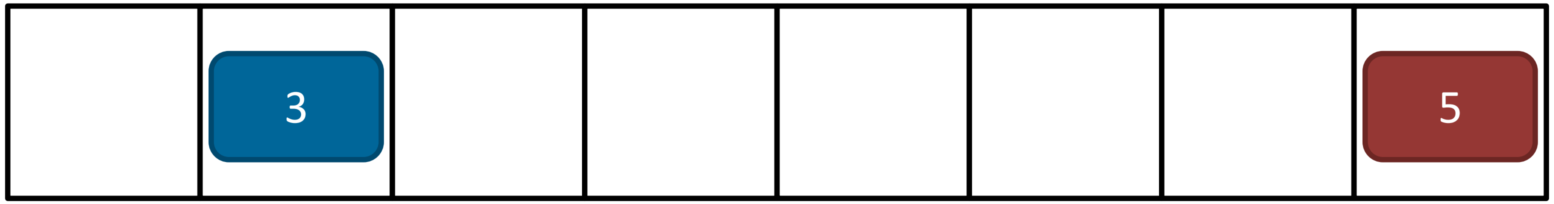
- Updates its velocity according to the car in front (if any)
- Drives that many cells to the right
    1. agent advances v cells

Case Study 4: Nagel Schreckenberg Model

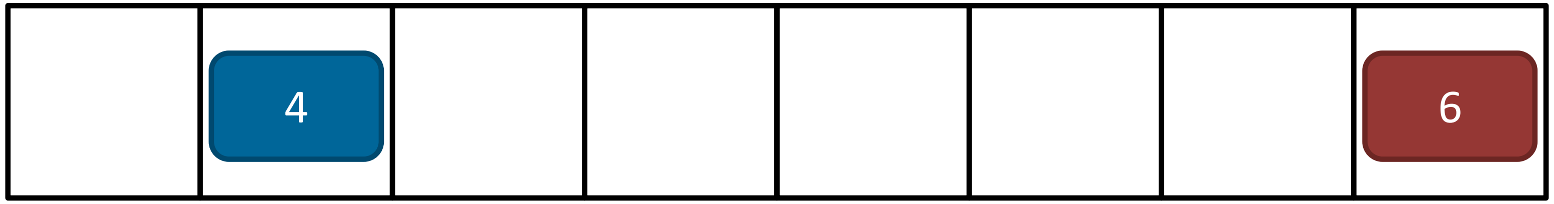


Increases its velocity v by one: $v \leftarrow$
$v + 1$

Case Study 4: Nagel Schreckenberg Model

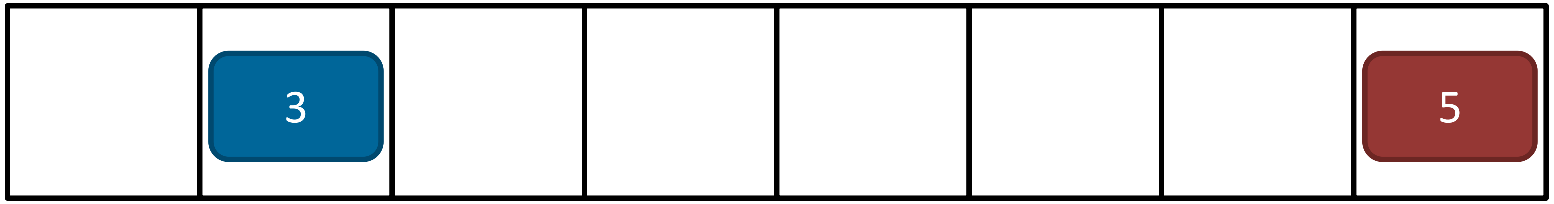


Increases its velocity v by one: $v \leftarrow v + 1$

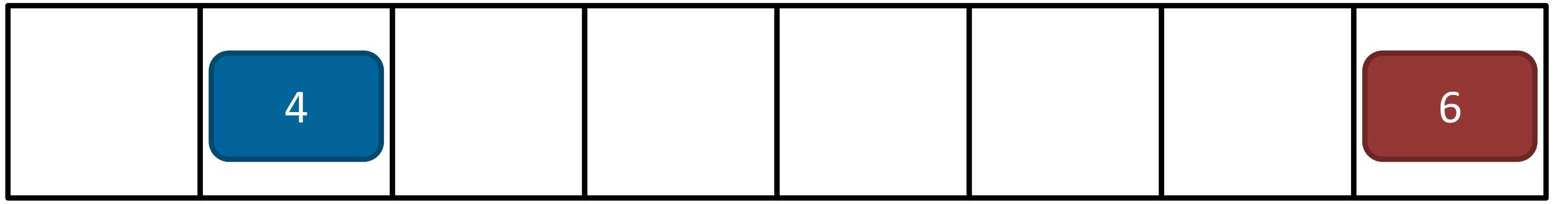


Checks how many cells to the right are empty (say q)

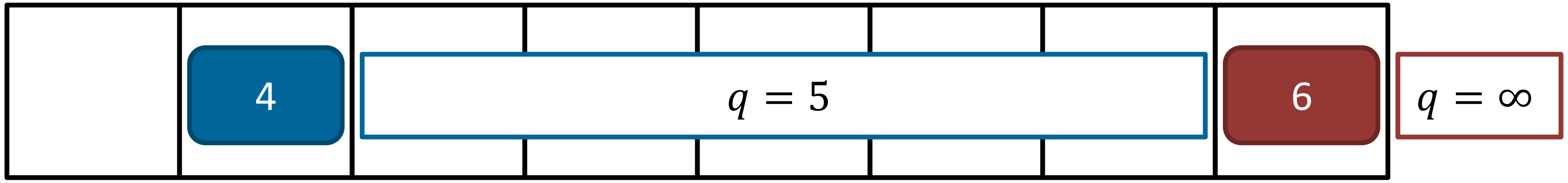
Case Study 4: Nagel Schreckenberg Model



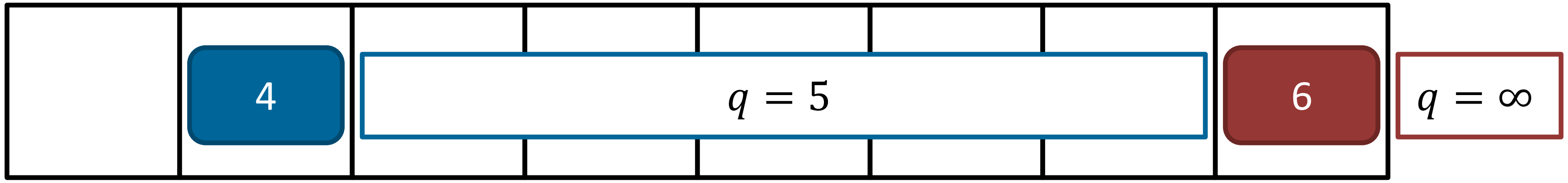
Increases its velocity v by one: $v \leftarrow v + 1$



Checks how many cells to the right are empty (say q)

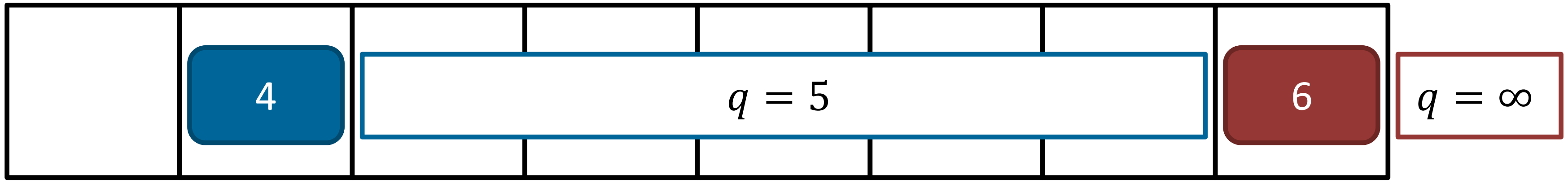


Case Study 4: Nagel Schreckenberg Model

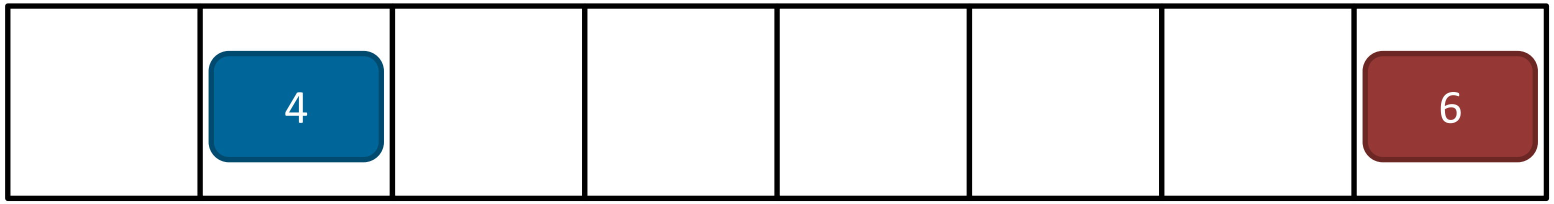


If $v > q$, then $v \leftarrow q$

Case Study 4: Nagel Schreckenberg Model

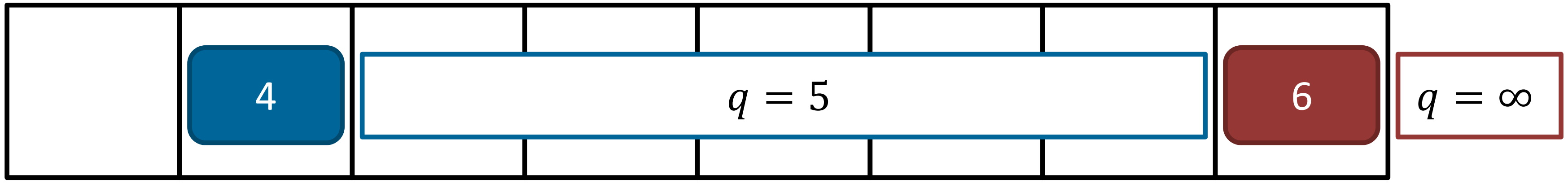


If $v > q$, then $v \leftarrow q$

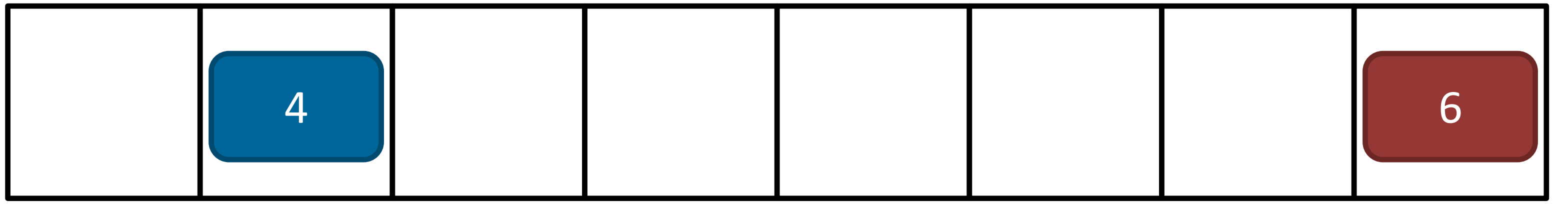


With a certain probability: $v \leftarrow \max(v - 1, 0)$

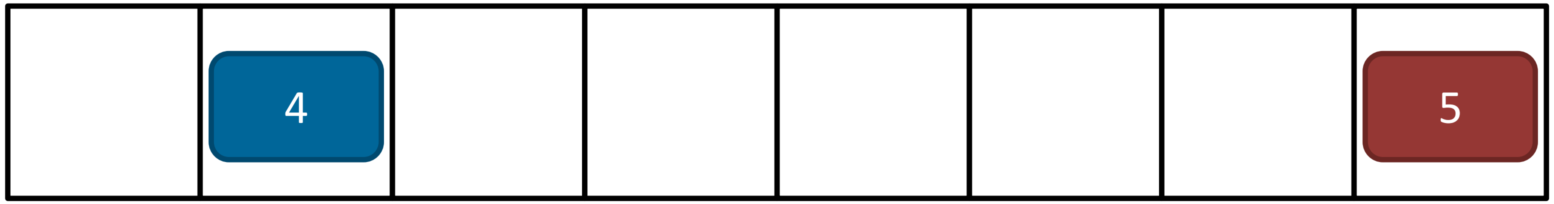
Case Study 4: Nagel Schreckenberg Model



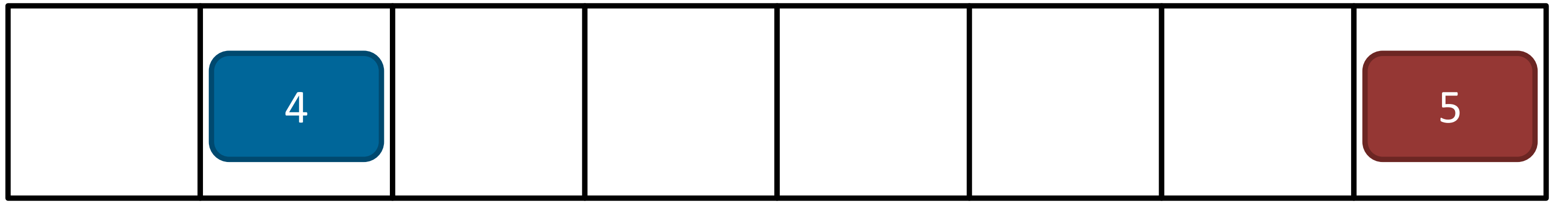
If $v > q$, then $v \leftarrow q$



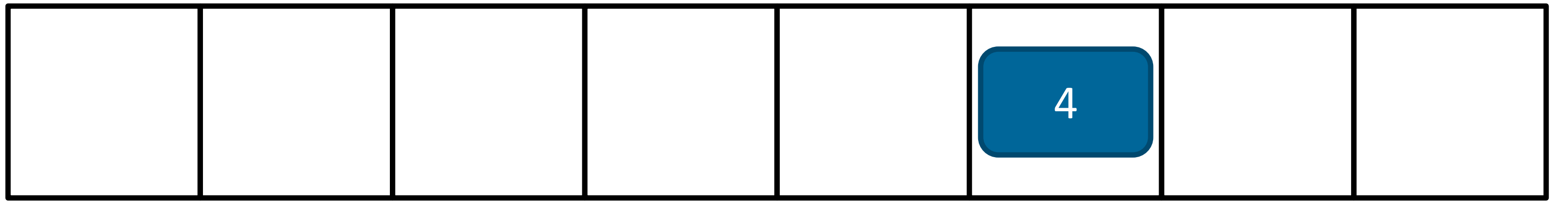
With a certain probability: $v \leftarrow \max(v - 1, 0)$



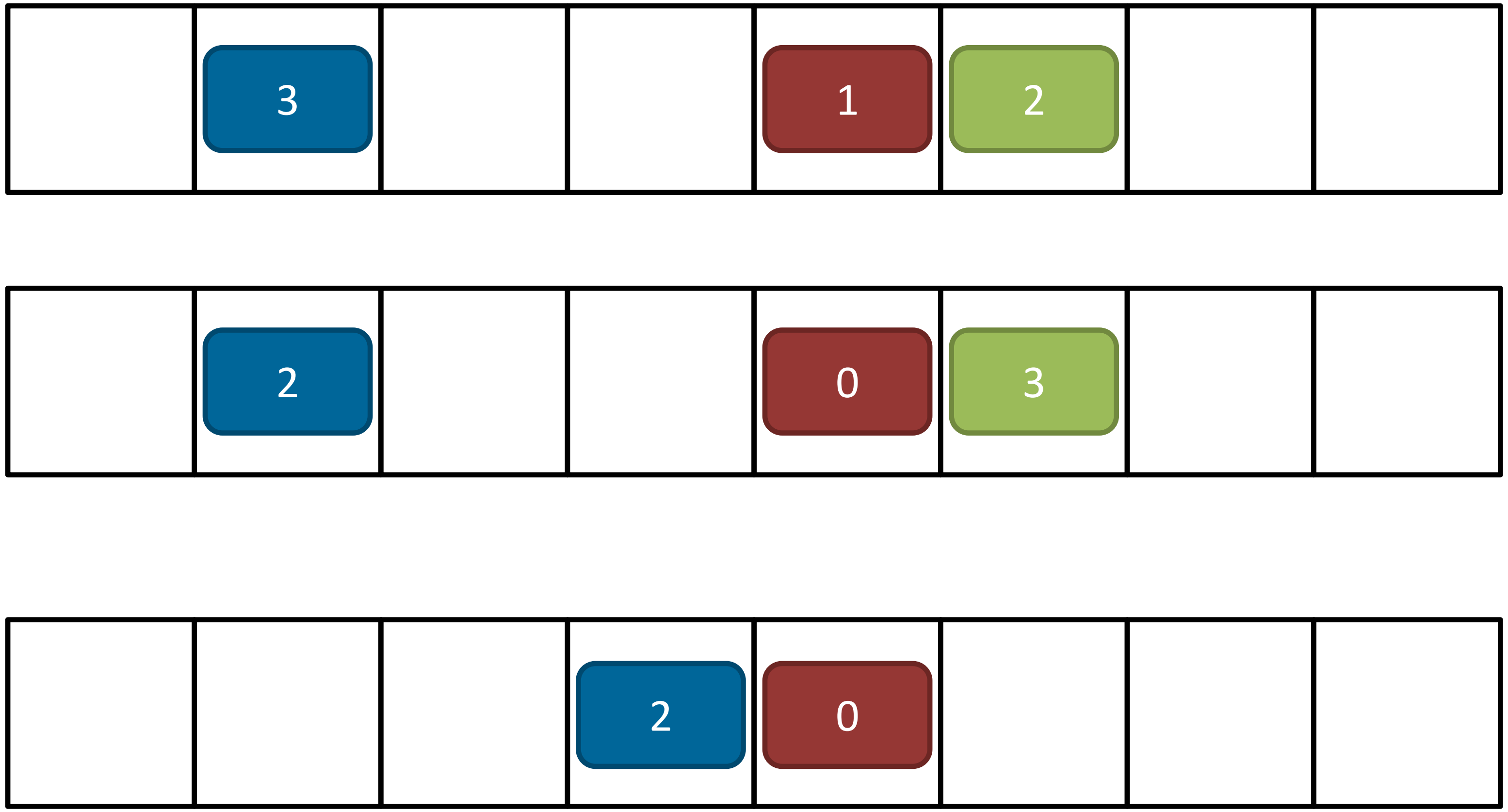
Case Study 4: Nagel Schreckenberg Model



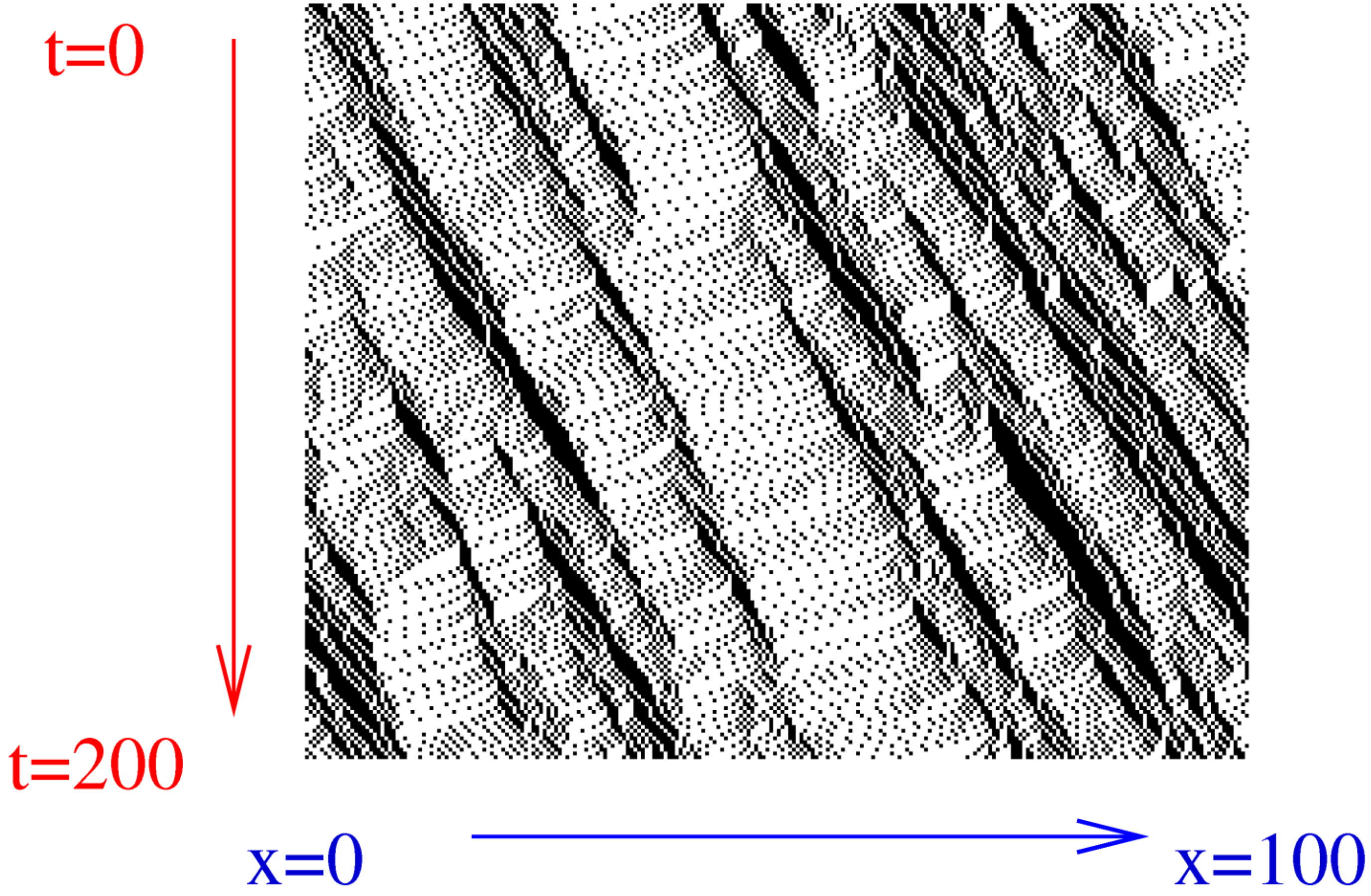
agent advances v cells



Case Study 4: Nagel Schreckenberg Model

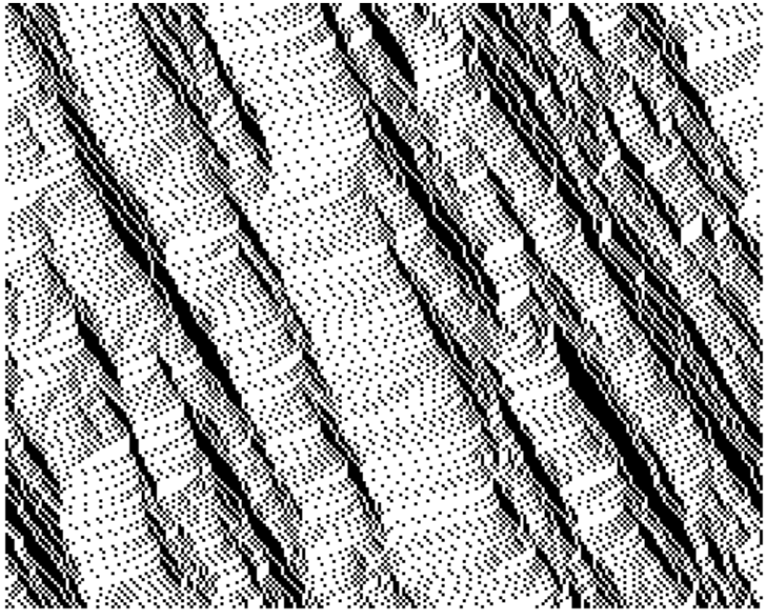


Case Study 4: Nagel Schreckenberg Model

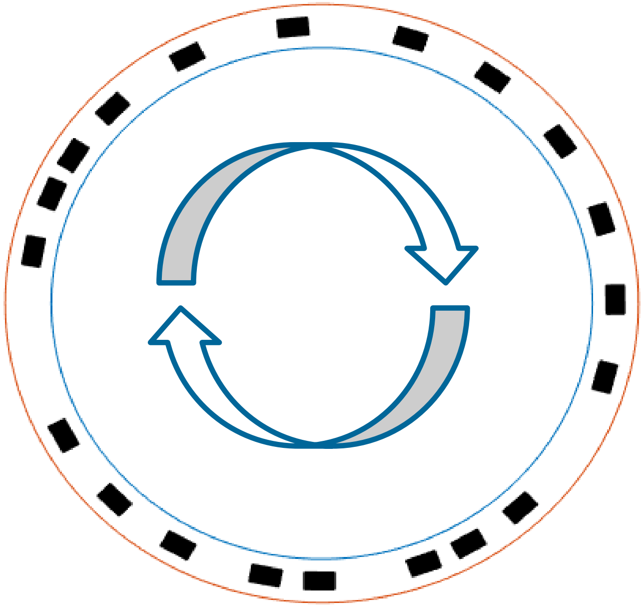


Case Study 4: Nagel Schreckenberg Model

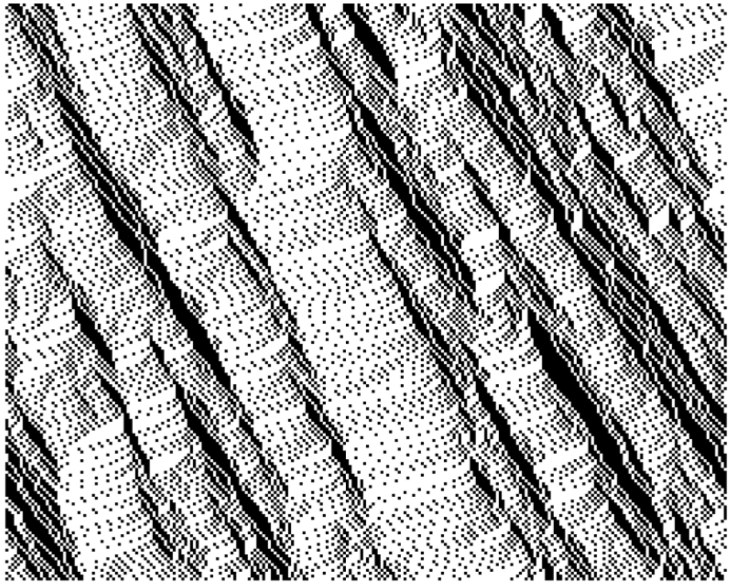
- Model usually described as a cellular automaton
- Model extendable to multiple lanes
- Either torodorial boundary conditions or new generation of cars every time-step



Differences?
(apart from known)

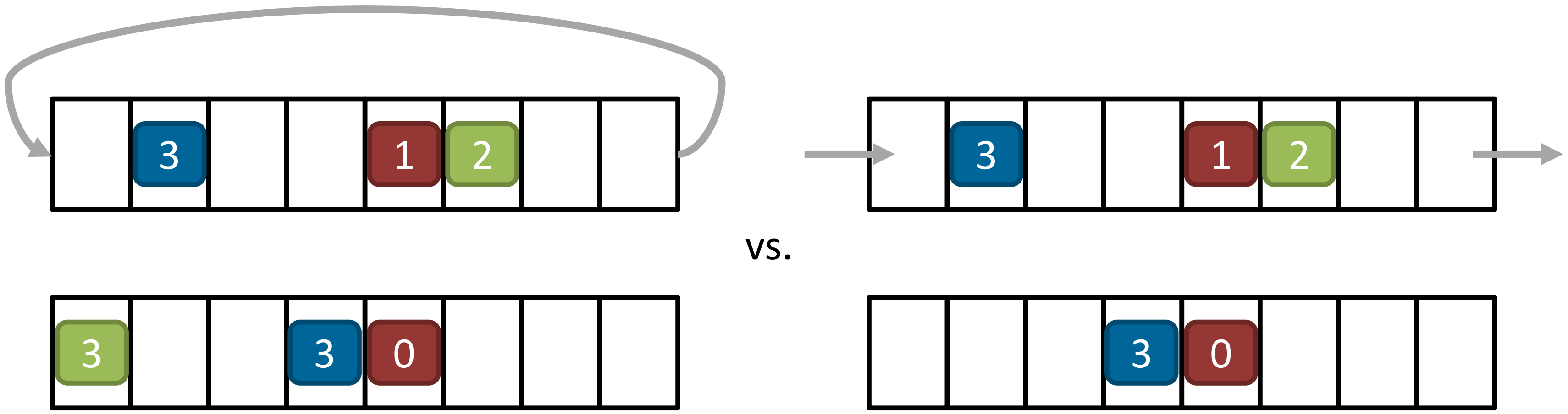


vs.



Case Study 4: Nagel Schreckenberg Model

Differences?



Classification of ABMs (4)

Classification 4	
with respect to <u>agent population</u>	
<u>population static</u>	**<u>population dynamic</u>**
• agents only generated at the beginning of the simulation • system variables only change due to change of agent states	• agents are (can be) generated on run-time • system variables can also change due to change of number of agents • usually more difficult to deal with due to space allocation of vectors

Case Study 4: Lessons Learned

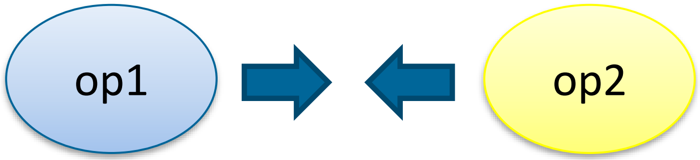
Lesson 7: Careful when implementing population dynamic agent based models: Removal and adding of elements to a list is usually expensive.

Consider, recreating the list every time instead of adding and removal!

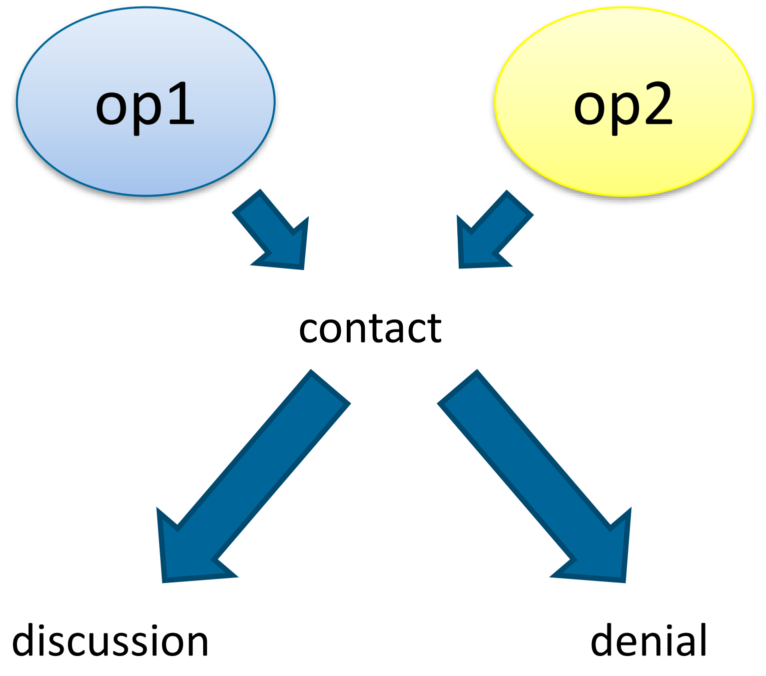
Make some benchmarks here.
You have nothing to lose but much to win!

Case Study 5: Opinion Dynamics

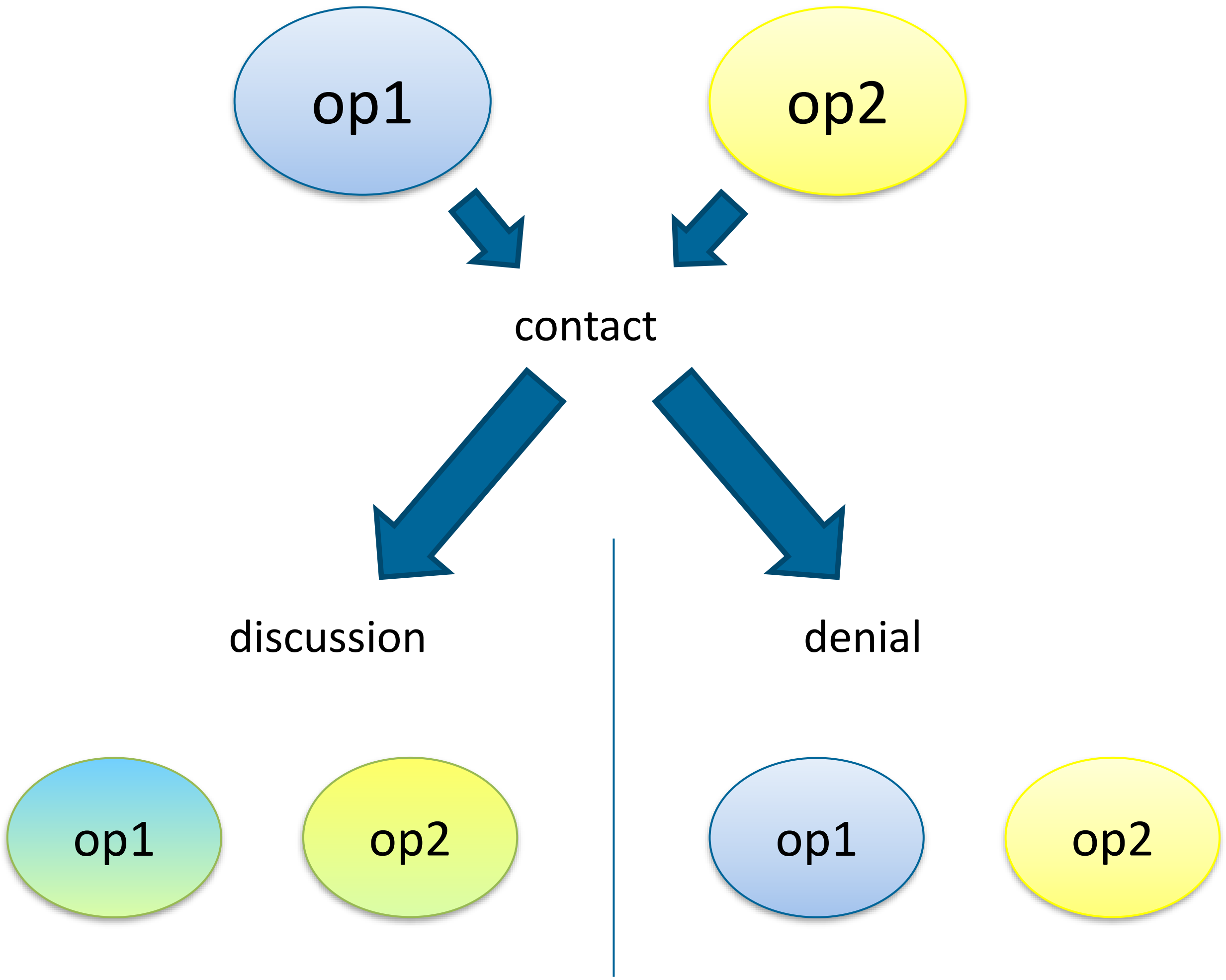
- Specific model from Guillaume Deffuant 2000 (basic concepts much older)
- Simple model that depicts spread and development of different opinions



Case Study 5: Opinion Dynamics



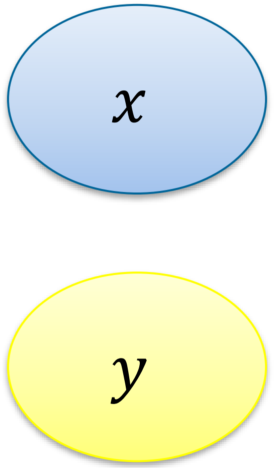
Case Study 5: Opinion Dynamics



Case Study 5: Opinion Dynamics

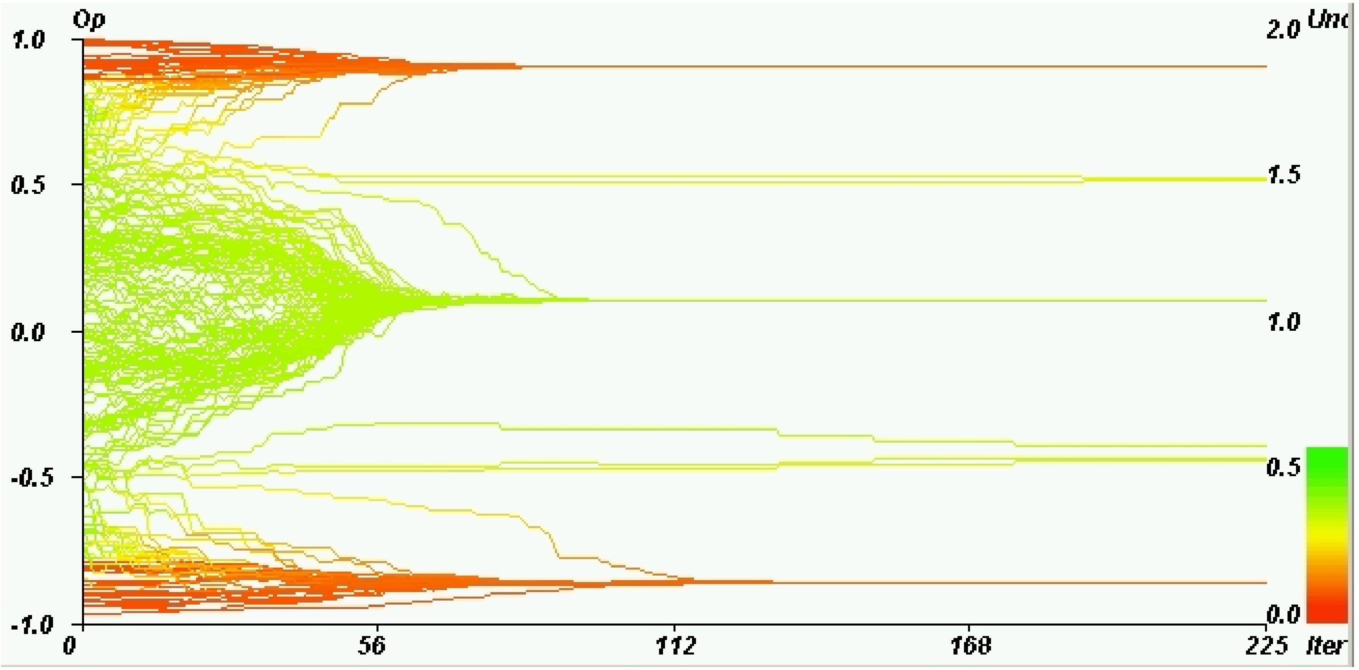
- N agents are initialised
- Every agent is assigned an opinion $x \in [-1,1]$
- Every time step,
  - Split the population into two random but equivalent halves
  - pick $N/2$ random partners from both, say with opinions x and y.
  - If $|x - y| < \tau$, the two start discussing and
    $$x \leftarrow x + \mu(y - x)$$
    $$y \leftarrow y - \mu(y - x)$$
  - Otherwise
    $$x \leftarrow x$$
    $$y \leftarrow y$$

$$\tau \in [0,2]$$
$$\mu \in \left[0, \frac{1}{2}\right]$$



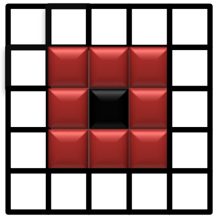
Case Study 5: Opinion Dynamics

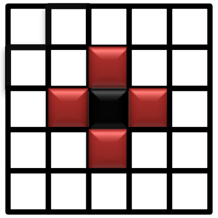
- Picturesque model to show, how communities with different opinions develop (e.g. Political parties,...)

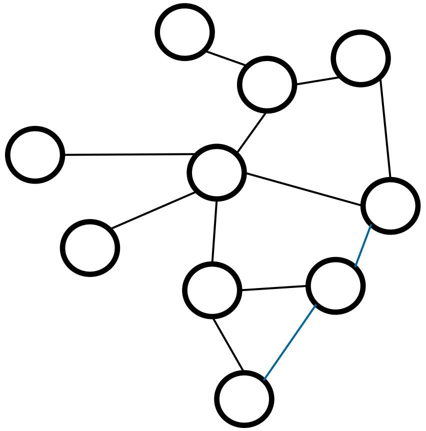


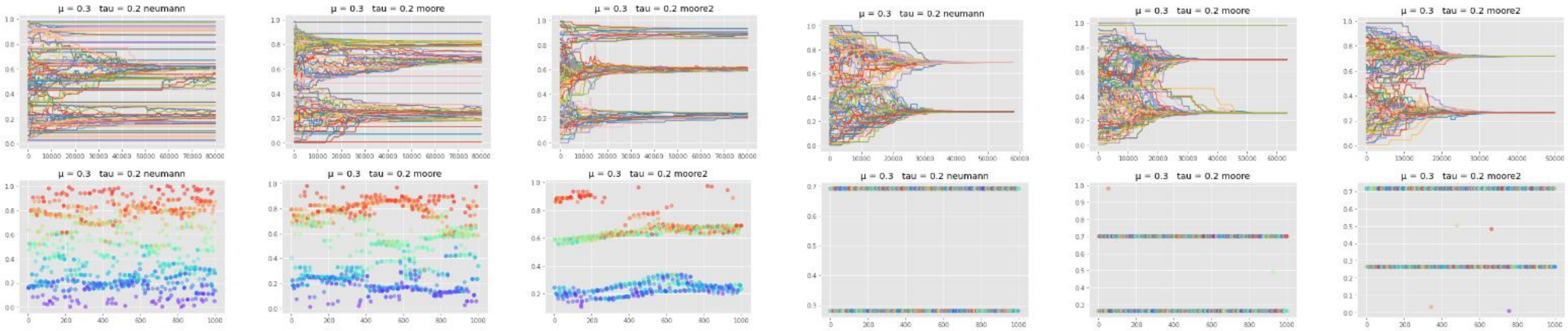
Case Study 5: Opinion Dynamics

- Options to investigate different topological structures of the contact network and contact behaviour









Case Study 5: Lessons Learned

Lesson 8: Use configuration files and unique scenario Ids not to lose track of the different created results. Make sure that the mapping between configuration and result is reproducible

Think about reproducibility of random events

Case Study 5: Lessons Learned

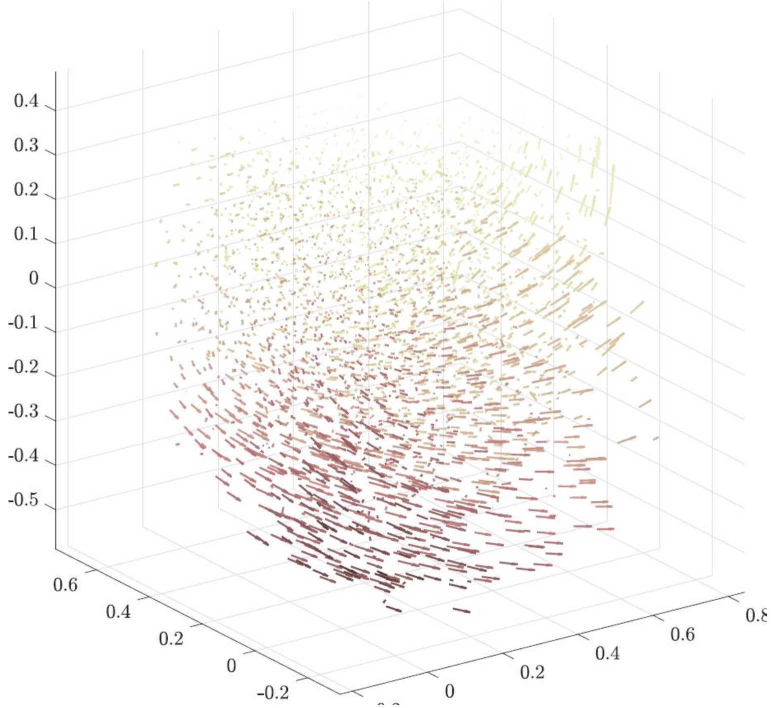
Lesson 8: Use configuration files and unique scenario Ids not to lose track of the different created results. Make sure that the mapping between configuration and result is reproducible

Lesson 9: Think about splitting initialisation and execution process. E.g. generation of random networks or initial sampling might take long.

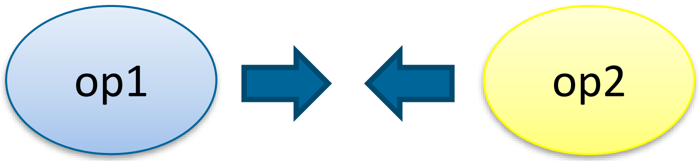
Enable serialisation and deserialisation or agents?

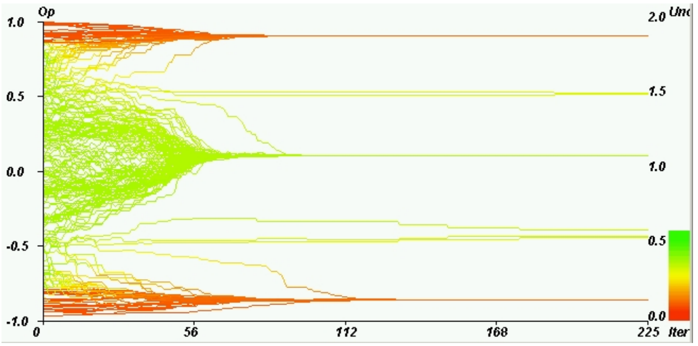
Case Study 5: Opinion Dynamics

any new differences?



vs.





Classification of ABMs (5)

<table>
<tr><th colspan="3">Classification 5</th></tr>
<tr><td colspan="3">with respect to <u>randomness (stochasticity)</u></td></tr>
<tr><td colspan="2"><u>stochastic</u></td><td><u>deterministic</u></td></tr>
<tr><td><u>initial-value stochastic</u></td><td><u>update stochastic</u></td><td></td></tr>
<tr><td>• Initial setting (of agents) is determined using random numbers</td><td>• Update rules user random numbers</td><td>• The outcome of the model is uniquely defined by its initial condition</td></tr>
</table>

Summary

When developing an agent-based model particularly care for ...

- ... order of actions / simultaneous events
- ... correct result interpretation
  (randomness , quantitative/qualitative, ...)
- … reproducible experimental design (configuration files …)
- ... code performance
- ... reproducible documentation(s)